



Welcome to [E-XFL.COM](https://www.e-xfl.com)

### What is "[Embedded - Microcontrollers](#)"?

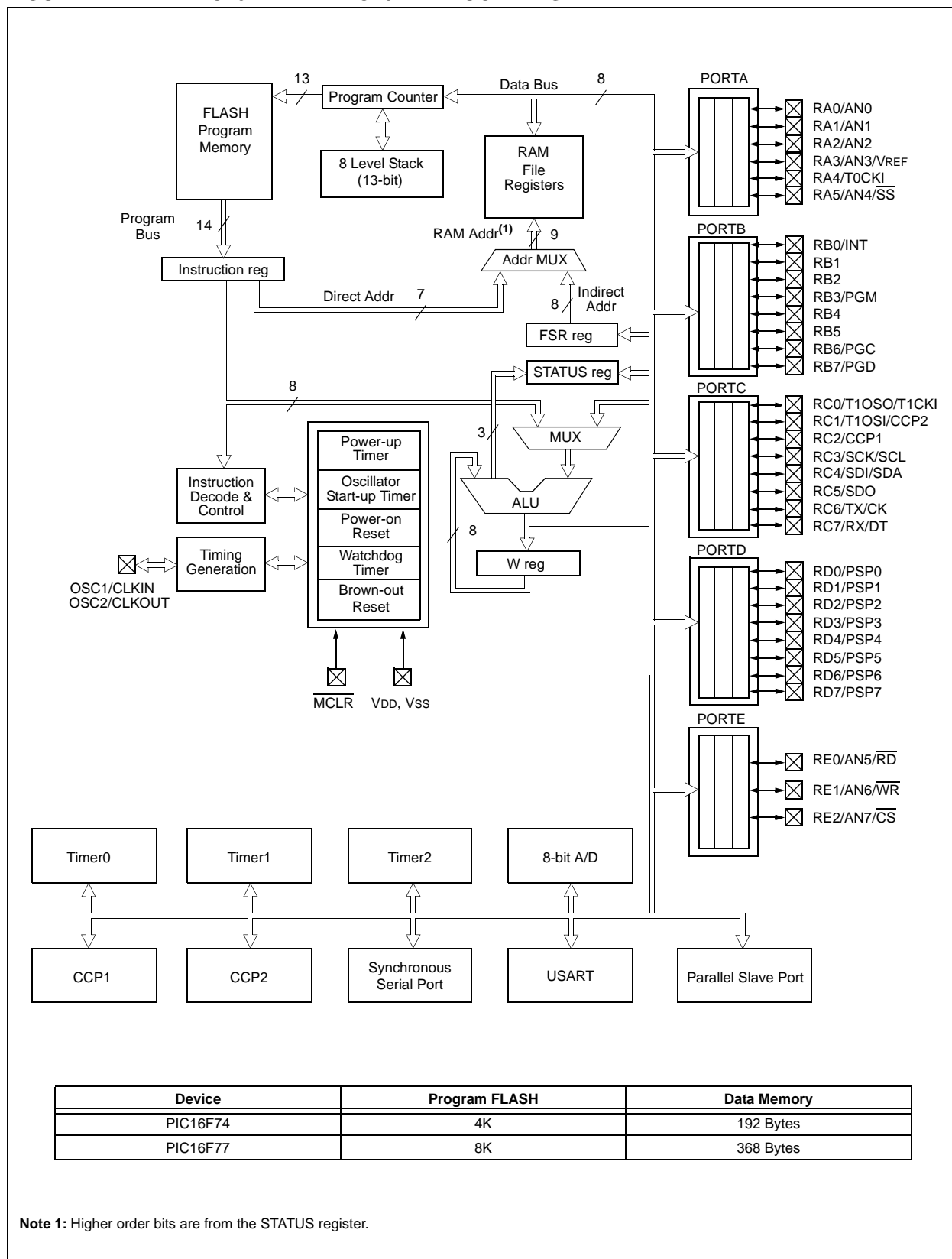
"[Embedded - Microcontrollers](#)" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" 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	20MHz
Connectivity	I <sup>2</sup> C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	33
Program Memory Size	7KB (4K x 14)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	192 x 8
Voltage - Supply (Vcc/Vdd)	2V ~ 5.5V
Data Converters	A/D 8x8b
Oscillator Type	External
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	44-LCC (J-Lead)
Supplier Device Package	44-PLCC (16.59x16.59)
Purchase URL	<a href="https://www.e-xfl.com/product-detail/microchip-technology/pic16lf74-i-l">https://www.e-xfl.com/product-detail/microchip-technology/pic16lf74-i-l</a>

**FIGURE 1-2: PIC16F74 AND PIC16F77 BLOCK DIAGRAM**



**FIGURE 2-3: PIC16F74/73 REGISTER FILE MAP**

File Address	File Address	File Address	File Address
Indirect addr.(*) 00h	Indirect addr.(*) 80h	Indirect addr.(*) 100h	Indirect addr.(*) 180h
TMR0 01h	OPTION_REG 81h	TMR0 101h	OPTION_REG 181h
PCL 02h	PCL 82h	PCL 102h	PCL 182h
STATUS 03h	STATUS 83h	STATUS 103h	STATUS 183h
FSR 04h	FSR 84h	FSR 104h	FSR 184h
PORTA 05h	TRISA 85h		
PORTB 06h	TRISB 86h	PORTB 106h	TRISB 186h
PORTC 07h	TRISC 87h		
PORTD <sup>(1)</sup> 08h	TRISD <sup>(1)</sup> 88h		
PORTE <sup>(1)</sup> 09h	TRISE <sup>(1)</sup> 89h		
PCLATH 0Ah	PCLATH 8Ah	PCLATH 10Ah	PCLATH 18Ah
INTCON 0Bh	INTCON 8Bh	INTCON 10Bh	INTCON 18Bh
PIR1 0Ch	PIE1 8Ch	PMDATA 10Ch	PMCON1 18Ch
PIR2 0Dh	PIE2 8Dh	PMADR 10Dh	
TMR1L 0Eh	PCON 8Eh	PMDATH 10Eh	
TMR1H 0Fh		PMADRH 10Fh	
T1CON 10h			
TMR2 11h			
T2CON 12h	PR2 92h		
SSPBUF 13h	SSPADDD 93h		
SSPCON 14h	SSPSTAT 94h		
CCPR1L 15h			
CCPR1H 16h			
CCP1CON 17h			
RCSTA 18h	TXSTA 98h		
TXREG 19h	SPBRG 99h		
RCREG 1Ah			
CCPR2L 1Bh			
CCPR2H 1Ch			
CCP2CON 1Dh			
ADRES 1Eh			
ADCON0 1Fh	ADCON1 9Fh		
General Purpose Register 96 Bytes	General Purpose Register 96 Bytes	accesses 20h-7Fh	accesses A0h - FFh
Bank 0 7Fh	Bank 1 FFh	Bank 2 17Fh	Bank 3 1FFh

■ Unimplemented data memory locations, read as '0'.  
 \* Not a physical register.

**Note 1:** These registers are not implemented on 28-pin devices.

# PIC16F7X

## 2.2.2.4 PIE1 Register

The PIE1 register contains the individual enable bits for the peripheral interrupts.

**Note:** Bit PEIE (INTCON<6>) must be set to enable any peripheral interrupt.

### REGISTER 2-4: PIE1 REGISTER (ADDRESS 8Ch)

	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	
	PSPIE <sup>(1)</sup>	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE
bit 7							bit 0	
bit 7	<b>PSPIE<sup>(1)</sup>:</b> Parallel Slave Port Read/Write Interrupt Enable bit 1 = Enables the PSP read/write interrupt 0 = Disables the PSP read/write interrupt							
bit 6	<b>ADIE:</b> A/D Converter Interrupt Enable bit 1 = Enables the A/D converter interrupt 0 = Disables the A/D converter interrupt							
bit 5	<b>RCIE:</b> USART Receive Interrupt Enable bit 1 = Enables the USART receive interrupt 0 = Disables the USART receive interrupt							
bit 4	<b>TXIE:</b> USART Transmit Interrupt Enable bit 1 = Enables the USART transmit interrupt 0 = Disables the USART transmit interrupt							
bit 3	<b>SSPIE:</b> Synchronous Serial Port Interrupt Enable bit 1 = Enables the SSP interrupt 0 = Disables the SSP interrupt							
bit 2	<b>CCP1IE:</b> CCP1 Interrupt Enable bit 1 = Enables the CCP1 interrupt 0 = Disables the CCP1 interrupt							
bit 1	<b>TMR2IE:</b> TMR2 to PR2 Match Interrupt Enable bit 1 = Enables the TMR2 to PR2 match interrupt 0 = Disables the TMR2 to PR2 match interrupt							
bit 0	<b>TMR1IE:</b> TMR1 Overflow Interrupt Enable bit 1 = Enables the TMR1 overflow interrupt 0 = Disables the TMR1 overflow interrupt							

**Note 1:** PSPIE is reserved on 28-pin devices; always maintain this bit clear.

#### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

- n = Value at POR reset

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

# PIC16F7X

---

NOTES:

## 3.0 READING PROGRAM MEMORY

The FLASH Program Memory is readable during normal operation over the entire VDD range. It is indirectly addressed through Special Function Registers (SFR). Up to 14-bit numbers can be stored in memory for use as calibration parameters, serial numbers, packed 7-bit ASCII, etc. Executing a program memory location containing data that forms an invalid instruction results in a NOP.

There are five SFRs used to read the program and memory. These registers are:

- PMCON1
- PMDATA
- PMDATH
- PMADR
- PMADRH

The program memory allows word reads. Program memory access allows for checksum calculation and reading calibration tables.

When interfacing to the program memory block, the PMDATH:PMDATA registers form a two-byte word, which holds the 14-bit data for reads. The PMADRH:PMADR registers form a two-byte word, which holds the 13-bit address of the FLASH location being accessed. These devices can have up to 8K words of program FLASH, with an address range from 0h to 3FFFh. The unused upper bits in both the PMDATH and PMADRH registers are not implemented and read as "0's".

### 3.1 PMADR

The address registers can address up to a maximum of 8K words of program FLASH.

When selecting a program address value, the MSByte of the address is written to the PMADRH register and the LSByte is written to the PMADR register. The upper MSbits of PMADRH must always be clear.

### 3.2 PMCON1 Register

PMCON1 is the control register for memory accesses.

The control bit RD initiates read operations. This bit cannot be cleared, only set, in software. It is cleared in hardware at the completion of the read operation.

#### REGISTER 3-1: PMCON1 REGISTER (ADDRESS 18Ch)

R-1	U-0	U-0	U-0	U-x	U-0	U-0	R/S-0
reserved	—	—	—	—	—	—	RD
bit 7							bit 0

bit 7 **Reserved:** Read as '1'

bit 6-1 **Unimplemented:** Read as '0'

bit 0 **RD:** Read Control bit

1 = Initiates a FLASH read, RD is cleared in hardware. The RD bit can only be set (not cleared) in software.

0 = FLASH read completed

#### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

- n = Value at POR reset

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

# PIC16F7X

## 3.3 Reading the FLASH Program Memory

A program memory location may be read by writing two bytes of the address to the PMADR and PMADRH registers and then setting control bit RD (PMCON1<0>). Once the read control bit is set, the microcontroller will use the next two instruction cycles to read the data. The data is available in the PMDATA and PMDATH registers after the second NOP instruction. Therefore, it can be read as two bytes in the following instructions. The PMDATA and PMDATH registers will hold this value until the next read operation.

## 3.4 Operation During Code Protect

FLASH program memory has its own code protect mechanism. External Read and Write operations by programmers are disabled if this mechanism is enabled.

The microcontroller can read and execute instructions out of the internal FLASH program memory, regardless of the state of the code protect configuration bits.

### EXAMPLE 3-1: FLASH PROGRAM READ

	BSF	STATUS, RP1	;
	BCF	STATUS, RP0	; Bank 2
	MOVF	ADDRH, W	;
	MOVWF	PMADRH	; MSByte of Program Address to read
	MOVF	ADDRL, W	;
	MOVWF	PMADR	; LSByte of Program Address to read
	BSF	STATUS, RP0	; Bank 3 Required
Required Sequence	BSF	PMCON1, RD	; EEPROM Read Sequence
	NOP		; memory is read in the next two cycles after BSF PMCON1,RD
	NOP		;
	BCF	STATUS, RP0	; Bank 2
	MOVF	PMDATA, W	; W = LSByte of Program PMDATA
	MOVF	PMDATH, W	; W = MSByte of Program PMDATA

TABLE 3-1: REGISTERS ASSOCIATED WITH PROGRAM FLASH

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other RESETS
10Dh	PMADR	Address Register Low Byte								xxxx xxxx	uuuu uuuu
10Fh	PMADRH	—	—	—	Address Register High Byte					xxxx xxxx	uuuu uuuu
10Ch	PMDATA	Data Register Low Byte								xxxx xxxx	uuuu uuuu
10Eh	PMDATH	—	—	Data Register High Byte						xxxx xxxx	uuuu uuuu
18Ch	PMCON1	— <sup>(1)</sup>	—	—	—	—	—	—	RD	1--- ---0	1--- ---0

Legend: x = unknown, u = unchanged, r = reserved, - = unimplemented read as '0'. Shaded cells are not used during FLASH access.

**Note 1:** This bit always reads as a '1'.

## 5.0 TIMER0 MODULE

The Timer0 module timer/counter has the following features:

- 8-bit timer/counter
- Readable and writable
- 8-bit software programmable prescaler
- Internal or external clock select
- Interrupt on overflow from FFh to 00h
- Edge select for external clock

Additional information on the Timer0 module is available in the PICmicro™ Mid-Range MCU Family Reference Manual (DS33023).

Figure 5-1 is a block diagram of the Timer0 module and the prescaler shared with the WDT.

Timer0 operation is controlled through the OPTION\_REG register (Register 5-1 on the following page). Timer mode is selected by clearing bit T0CS (OPTION\_REG<5>). In Timer mode, the Timer0 module will increment every instruction cycle (without prescaler). If the TMR0 register is written, the increment is inhibited for the following two instruction cycles. The user can work around this by writing an adjusted value to the TMR0 register.

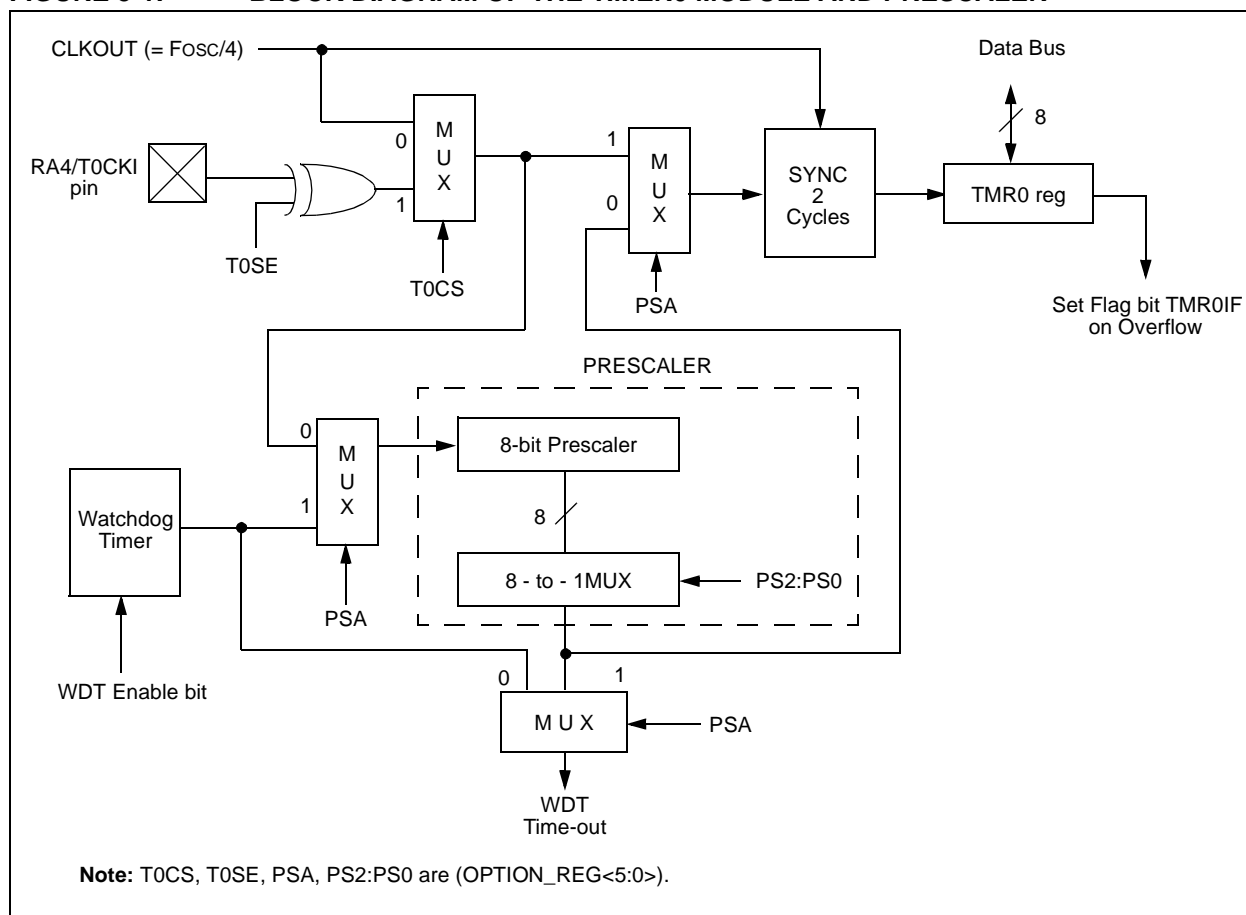
Counter mode is selected by setting bit T0CS (OPTION\_REG<5>). In Counter mode, Timer0 will increment, either on every rising or falling edge of pin RA4/T0CKI. The incrementing edge is determined by the Timer0 Source Edge Select bit T0SE (OPTION\_REG<4>). Clearing bit T0SE selects the rising edge. Restrictions on the external clock input are discussed in detail in Section 5.2.

The prescaler is mutually exclusively shared between the Timer0 module and the Watchdog Timer. The prescaler is not readable or writable. Section 5.3 details the operation of the prescaler.

### 5.1 Timer0 Interrupt

The TMR0 interrupt is generated when the TMR0 register overflows from FFh to 00h. This overflow sets bit TMR0IF (INTCON<2>). The interrupt can be masked by clearing bit TMR0IE (INTCON<5>). Bit TMR0IF must be cleared in software by the Timer0 module Interrupt Service Routine, before re-enabling this interrupt. The TMR0 interrupt cannot awaken the processor from SLEEP, since the timer is shut-off during SLEEP.

**FIGURE 5-1: BLOCK DIAGRAM OF THE TIMER0 MODULE AND PRESCALER**





# PIC16F7X

---

NOTES:

# PIC16F7X

## 6.1 Timer1 Operation in Timer Mode

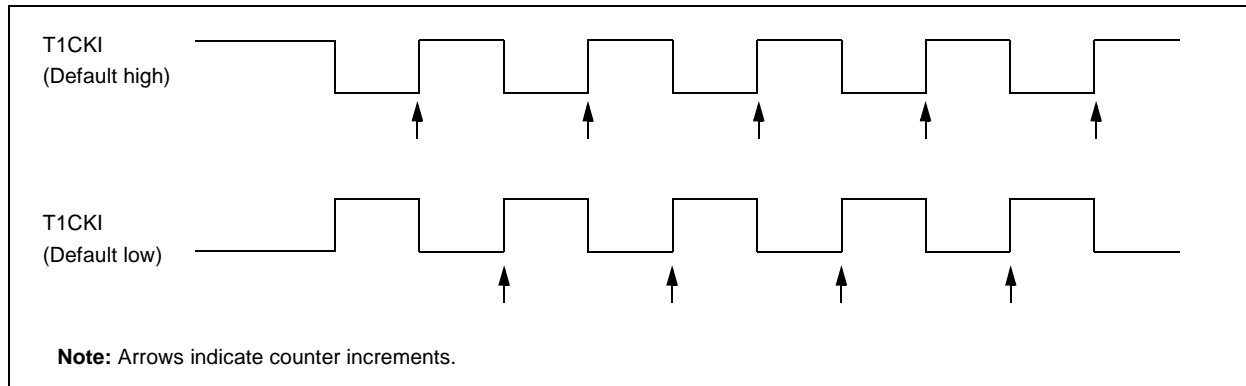
Timer mode is selected by clearing the TMR1CS (T1CON<1>) bit. In this mode, the input clock to the timer is  $F_{OSC}/4$ . The synchronize control bit  $\overline{T1SYNC}$  (T1CON<2>) has no effect, since the internal clock is always in sync.

## 6.2 Timer1 Counter Operation

Timer1 may operate in Asynchronous or Synchronous mode, depending on the setting of the TMR1CS bit.

When Timer1 is being incremented via an external source, increments occur on a rising edge. After Timer1 is enabled in Counter mode, the module must first have a falling edge before the counter begins to increment.

**FIGURE 6-1: TIMER1 INCREMENTING EDGE**



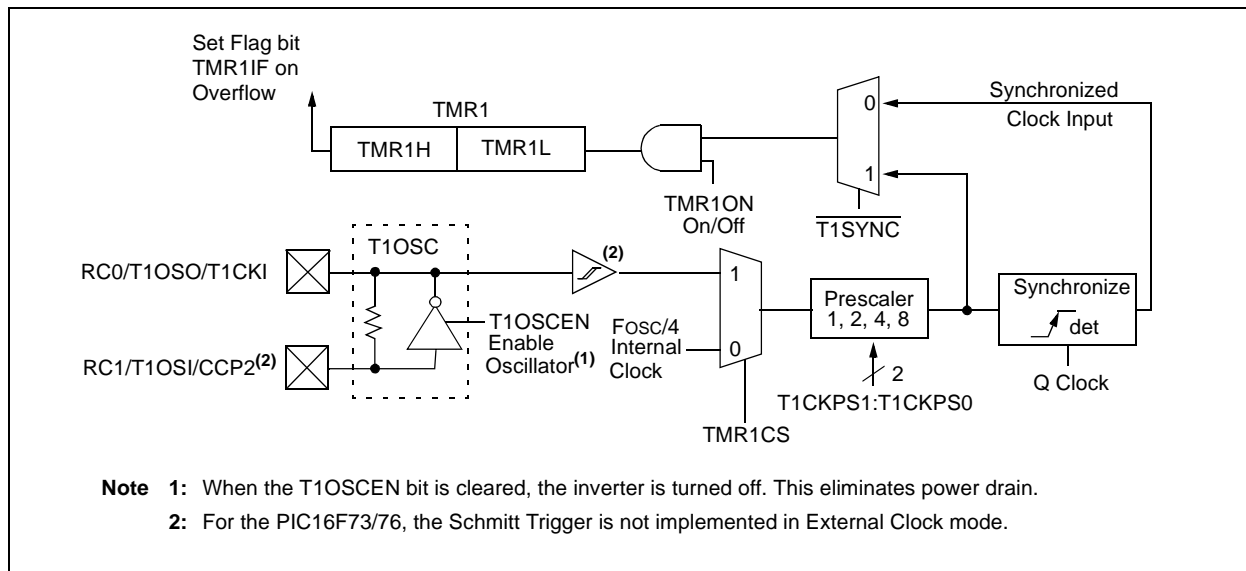
## 6.3 Timer1 Operation in Synchronized Counter Mode

Counter mode is selected by setting bit TMR1CS. In this mode, the timer increments on every rising edge of clock input on pin RC1/T1OSI/CCP2, when bit T1OSCEN is set, or on pin RC0/T1OSO/T1CKI, when bit T1OSCEN is cleared.

If  $\overline{T1SYNC}$  is cleared, then the external clock input is synchronized with internal phase clocks. The synchronization is done after the prescaler stage. The prescaler stage is an asynchronous ripple counter.

In this configuration, during SLEEP mode, Timer1 will not increment even if the external clock is present, since the synchronization circuit is shut-off. The prescaler, however, will continue to increment.

**FIGURE 6-2: TIMER1 BLOCK DIAGRAM**



# PIC16F7X

## 8.4.1 CCP PIN CONFIGURATION

The user must configure the RC2/CCP1 pin as an output by clearing the TRISC<2> bit.

**Note:** Clearing the CCP1CON register will force the RC2/CCP1 compare output latch to the default low level. This is not the PORTC I/O data latch.

## 8.4.2 TIMER1 MODE SELECTION

Timer1 must be running in Timer mode or Synchronized Counter mode if the CCP module is using the compare feature. In Asynchronous Counter mode, the compare operation may not work.

## 8.4.3 SOFTWARE INTERRUPT MODE

When Generate Software Interrupt mode is chosen, the CCP1 pin is not affected. The CCP1IF or CCP2IF bit is set, causing a CCP interrupt (if enabled).

## 8.4.4 SPECIAL EVENT TRIGGER

In this mode, an internal hardware trigger is generated, which may be used to initiate an action.

The special event trigger output of CCP1 resets the TMR1 register pair. This allows the CCPR1 register to effectively be a 16-bit programmable period register for Timer1.

The special event trigger output of CCP2 resets the TMR1 register pair and starts an A/D conversion (if the A/D module is enabled).

**Note:** The special event trigger from the CCP1 and CCP2 modules will not set interrupt flag bit TMR1IF (PIR1<0>).

**TABLE 8-3: REGISTERS ASSOCIATED WITH CAPTURE, COMPARE, AND TIMER1**

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other RESETS
0Bh,8Bh, 10Bh,18Bh	INTCON	GIE	PEIE	TMR0IE	INTE	RBIE	TMR0IF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	PSPIF <sup>(1)</sup>	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
0Dh	PIR2	—	—	—	—	—	—	—	CCP2IF	---- --0	---- --0
8Ch	PIE1	PSPIE <sup>(1)</sup>	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
8Dh	PIE2	—	—	—	—	—	—	—	CCP2IE	---- --0	---- --0
87h	TRISC	PORTC Data Direction Register								1111 1111	1111 1111
0Eh	TMR1L	Holding Register for the Least Significant Byte of the 16-bit TMR1 Register								xxxx xxxx	uuuu uuuu
0Fh	TMR1H	Holding Register for the Most Significant Byte of the 16-bit TMR1 Register								xxxx xxxx	uuuu uuuu
10h	T1CON	—	—	T1CKPS1	T1CKPS0	T1OSCEN	T1SYNC	TMR1CS	TMR1ON	--00 0000	--uu uuuu
15h	CCPR1L	Capture/Compare/PWM Register1 (LSB)								xxxx xxxx	uuuu uuuu
16h	CCPR1H	Capture/Compare/PWM Register1 (MSB)								xxxx xxxx	uuuu uuuu
17h	CCP1CON	—	—	CCP1X	CCP1Y	CCP1M3	CCP1M2	CCP1M1	CCP1M0	--00 0000	--00 0000
1Bh	CCPR2L	Capture/Compare/PWM Register2 (LSB)								xxxx xxxx	uuuu uuuu
1Ch	CCPR2H	Capture/Compare/PWM Register2 (MSB)								xxxx xxxx	uuuu uuuu
1Dh	CCP2CON	—	—	CCP2X	CCP2Y	CCP2M3	CCP2M2	CCP2M1	CCP2M0	--00 0000	--00 0000

Legend: x = unknown, u = unchanged, - = unimplemented, read as '0'. Shaded cells are not used by Capture and Timer1.

**Note 1:** The PSP is not implemented on the PIC16F73/76; always maintain these bits clear.

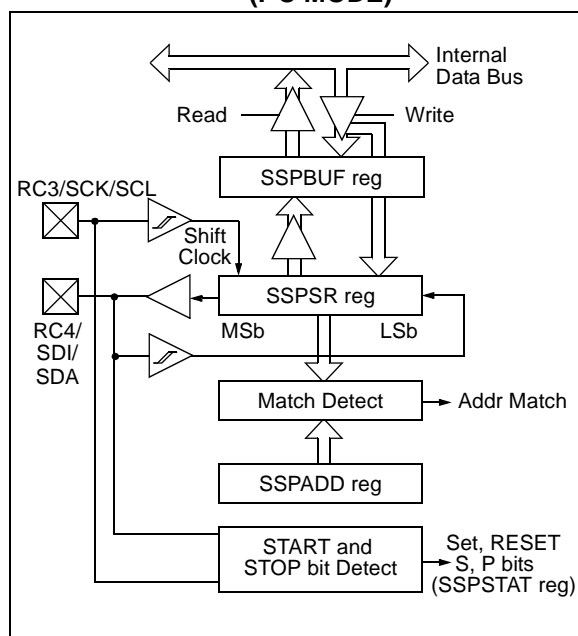
## 9.3 SSP I<sup>2</sup>C Operation

The SSP module in I<sup>2</sup>C mode, fully implements all slave functions, except general call support, and provides interrupts on START and STOP bits in hardware to facilitate firmware implementations of the master functions. The SSP module implements the standard mode specifications as well as 7-bit and 10-bit addressing.

Two pins are used for data transfer. These are the RC3/SCK/SCL pin, which is the clock (SCL), and the RC4/SDI/SDA pin, which is the data (SDA). The user must configure these pins as inputs or outputs through the TRISC<4:3> bits.

The SSP module functions are enabled by setting SSP enable bit SSPEN (SSPCON<5>).

**FIGURE 9-5: SSP BLOCK DIAGRAM (I<sup>2</sup>C MODE)**



The SSP module has five registers for I<sup>2</sup>C operation. These are the:

- SSP Control Register (SSPCON)
- SSP Status Register (SSPSTAT)
- Serial Receive/Transmit Buffer (SSPBUF)
- SSP Shift Register (SSPSR) - Not directly accessible
- SSP Address Register (SSPADD)

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

- I<sup>2</sup>C Slave mode (7-bit address)
- I<sup>2</sup>C Slave mode (10-bit address)
- I<sup>2</sup>C Slave mode (7-bit address), with START and STOP bit interrupts enabled to support Firmware Master mode
- I<sup>2</sup>C Slave mode (10-bit address), with START and STOP bit interrupts enabled to support Firmware Master mode
- I<sup>2</sup>C START and STOP bit interrupts enabled to support Firmware Master mode, Slave is IDLE

Selection of any I<sup>2</sup>C mode with the SSPEN bit set, forces the SCL and SDA pins to be open drain, provided these pins are programmed to inputs by setting the appropriate TRISC bits. Pull-up resistors must be provided externally to the SCL and SDA pins for proper operation of the I<sup>2</sup>C module.

Additional information on SSP I<sup>2</sup>C operation can be found in the PICmicro™ Mid-Range MCU Family Reference Manual (DS33023A).

### 9.3.1 SLAVE MODE

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

When an address is matched, or the data transfer after an address match is received, the hardware automatically will generate the Acknowledge (ACK) pulse, and then load the SSPBUF register with the received value currently in the SSPSR register.

There are certain conditions that will cause the SSP module not to give this ACK pulse. They include (either or both):

- The buffer full bit BF (SSPSTAT<0>) was set before the transfer was received.
- The overflow bit SSPOV (SSPCON<6>) was set before the transfer was received.

In this case, the SSPSR register value is not loaded into the SSPBUF, but bit SSPIF (PIR1<3>) is set. Table 9-2 shows what happens when a data transfer byte is received, given the status of bits BF and SSPOV. The shaded cells show the condition where user software did not properly clear the overflow condition. Flag bit BF is cleared by reading the SSPBUF register, while bit SSPOV is cleared through software.

The SCL clock input must have a minimum high and low for proper operation. The high and low times of the I<sup>2</sup>C specification, as well as the requirements of the SSP module, are shown in timing parameter #100 and parameter #101.

# PIC16F7X

## 9.3.2 MASTER MODE

Master mode of operation is supported in firmware using interrupt generation on the detection of the START and STOP conditions. The STOP (P) and START (S) bits are cleared from a RESET or when the SSP module is disabled. The STOP (P) and START (S) bits will toggle based on the START and STOP conditions. Control of the I<sup>2</sup>C bus may be taken when the P bit is set, or the bus is IDLE and both the S and P bits are clear.

In Master mode, the SCL and SDA lines are manipulated by clearing the corresponding TRISC<4:3> bit(s). The output level is always low, irrespective of the value(s) in PORTC<4:3>. So when transmitting data, a '1' data bit must have the TRISC<4> bit set (input) and a '0' data bit must have the TRISC<4> bit cleared (output). The same scenario is true for the SCL line with the TRISC<3> bit. Pull-up resistors must be provided externally to the SCL and SDA pins for proper operation of the I<sup>2</sup>C module.

The following events will cause SSP Interrupt Flag bit, SSPIF, to be set (SSP Interrupt will occur if enabled):

- START condition
- STOP condition
- Data transfer byte transmitted/received

Master mode of operation can be done with either the Slave mode IDLE (SSPM3:SSPM0 = 1011), or with the Slave active. When both Master and Slave modes are enabled, the software needs to differentiate the source(s) of the interrupt.

## 9.3.3 MULTI-MASTER MODE

In Multi-Master mode, the interrupt generation on the detection of the START and STOP conditions, allows the determination of when the bus is free. The STOP (P) and START (S) bits are cleared from a RESET or when the SSP module is disabled. The STOP (P) and START (S) bits will toggle based on the START and STOP conditions. Control of the I<sup>2</sup>C bus may be taken when bit P (SSPSTAT<4>) is set, or the bus is IDLE and both the S and P bits clear. When the bus is busy, enabling the SSP Interrupt will generate the interrupt when the STOP condition occurs.

In Multi-Master operation, the SDA line must be monitored to see if the signal level is the expected output level. This check only needs to be done when a high level is output. If a high level is expected and a low level is present, the device needs to release the SDA and SCL lines (set TRISC<4:3>). There are two stages where this arbitration can be lost, these are:

- Address Transfer
- Data Transfer

When the slave logic is enabled, the slave continues to receive. If arbitration was lost during the address transfer stage, communication to the device may be in progress. If addressed, an ACK pulse will be generated. If arbitration was lost during the data transfer stage, the device will need to retransfer the data at a later time.

**TABLE 9-3: REGISTERS ASSOCIATED WITH I<sup>2</sup>C OPERATION**

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other RESETS
0Bh, 8Bh, 10Bh, 18Bh	INTCON	GIE	PEIE	TMR0IE	INTE	RBIE	TMR0IF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	PSPIF <sup>(1)</sup>	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
8Ch	PIE1	PSPIE <sup>(1)</sup>	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
13h	SSPBUF	Synchronous Serial Port Receive Buffer/Transmit Register								xxxx xxxx	uuuu uuuu
93h	SSPAD	Synchronous Serial Port (I <sup>2</sup> C mode) Address Register								0000 0000	0000 0000
14h	SSPCON	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	0000 0000
94h	SSPSTAT	SMP <sup>(2)</sup>	CKE <sup>(2)</sup>	D/A	P	S	R/W	UA	BF	0000 0000	0000 0000
87h	TRISC	PORTC Data Direction Register								1111 1111	1111 1111

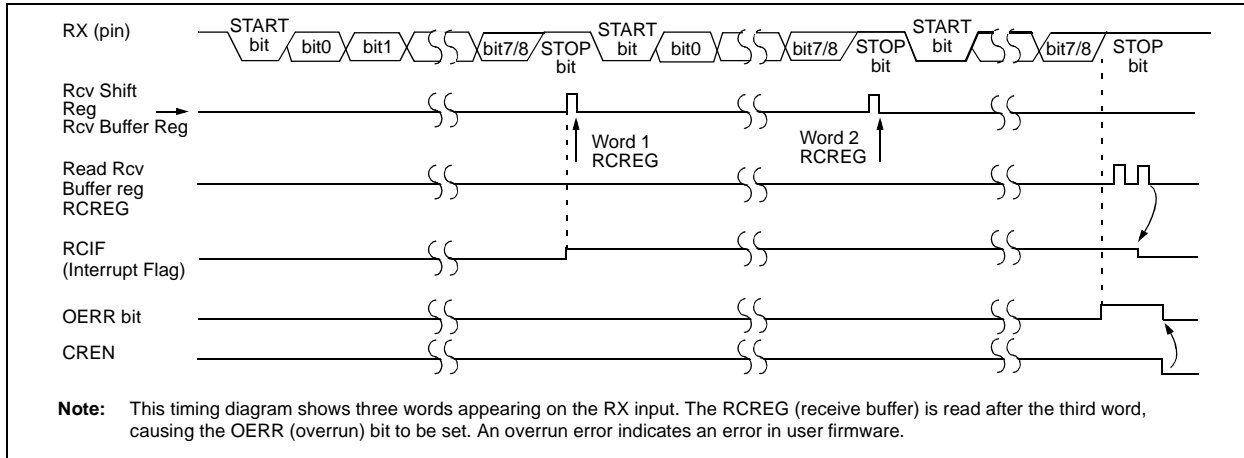
Legend: x = unknown, u = unchanged, - = unimplemented locations read as '0'. Shaded cells are not used by SSP module in I<sup>2</sup>C mode.

**Note 1:** PSPIF and PSPIE are reserved on the PIC16F73/76; always maintain these bits clear.

**2:** Maintain these bits clear in I<sup>2</sup>C mode.

# PIC16F7X

**FIGURE 10-5: ASYNCHRONOUS RECEPTION**



Steps to follow when setting up an Asynchronous Reception:

1. Initialize the SPBRG register for the appropriate baud rate. If a high speed baud rate is desired, set bit BRGH (Section 10.1).
2. Enable the asynchronous serial port by clearing bit SYNC and setting bit SPEN.
3. If interrupts are desired, then set enable bit RCIE.
4. If 9-bit reception is desired, then set bit RX9.
5. Enable the reception by setting bit CREN.
6. Flag bit RCIF will be set when reception is complete and an interrupt will be generated if enable bit RCIE is set.
7. Read the RCSTA register to get the ninth bit (if enabled) and determine if any error occurred during reception.
8. Read the 8-bit received data by reading the RCREG register.
9. If any error occurred, clear the error by clearing enable bit CREN.
10. If using interrupts, ensure that GIE and PEIE in the INTCON register are set.

**TABLE 10-6: REGISTERS ASSOCIATED WITH ASYNCHRONOUS RECEPTION**

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other RESETS
0Bh, 8Bh, 10Bh, 18Bh	INTCON	GIE	PEIE	TMR0IE	INTE	RBIE	TMR0IF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	PSPIF <sup>(1)</sup>	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
18h	RCSTA	SPEN	RX9	SREN	CREN	—	FERR	OERR	RX9D	0000 -00x	0000 -00x
1Ah	RCREG	USART Receive Register								0000 0000	0000 0000
8Ch	PIE1	PSPIE <sup>(1)</sup>	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
98h	TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010
99h	SPBRG	Baud Rate Generator Register								0000 0000	0000 0000

Legend: x = unknown, - = unimplemented locations read as '0'. Shaded cells are not used for asynchronous reception.

**Note 1:** Bits PSPIE and PSPIF are reserved on the PIC16F73/76 devices; always maintain these bits clear.

## 12.0 SPECIAL FEATURES OF THE CPU

These devices have a host of features intended to maximize system reliability, minimize cost through elimination of external components, provide power saving operating modes and offer code protection. These are:

- Oscillator Selection
- RESET
  - Power-on Reset (POR)
  - Power-up Timer (PWRT)
  - Oscillator Start-up Timer (OST)
  - Brown-out Reset (BOR)
- Interrupts
- Watchdog Timer (WDT)
- SLEEP
- Code Protection
- ID Locations
- In-Circuit Serial Programming

These devices have a Watchdog Timer, which can be enabled or disabled, using a configuration bit. It runs off its own RC oscillator for added reliability.

There are two timers that offer necessary delays on power-up. One is the Oscillator Start-up Timer (OST), intended to keep the chip in RESET until the crystal oscillator is stable. The other is the Power-up Timer (PWRT), which provides a fixed delay of 72 ms (nominal) on power-up only. It is designed to keep the part in RESET while the power supply stabilizes, and is enabled or disabled, using a configuration bit. With these two timers on-chip, most applications need no external RESET circuitry.

SLEEP mode is designed to offer a very low current power-down mode. The user can wake-up from SLEEP through external RESET, Watchdog Timer Wake-up, or through an interrupt.

Several oscillator options are also made available to allow the part to fit the application. The RC oscillator option saves system cost while the LP crystal option saves power. Configuration bits are used to select the desired oscillator mode.

Additional information on special features is available in the PICmicro™ Mid-Range Reference Manual (DS33023).

### 12.1 Configuration Bits

The configuration bits can be programmed (read as '0'), or left unprogrammed (read as '1'), to select various device configurations. These bits are mapped in program memory location 2007h.

The user will note that address 2007h is beyond the user program memory space, which can be accessed only during programming.

## 12.2 Oscillator Configurations

### 12.2.1 OSCILLATOR TYPES

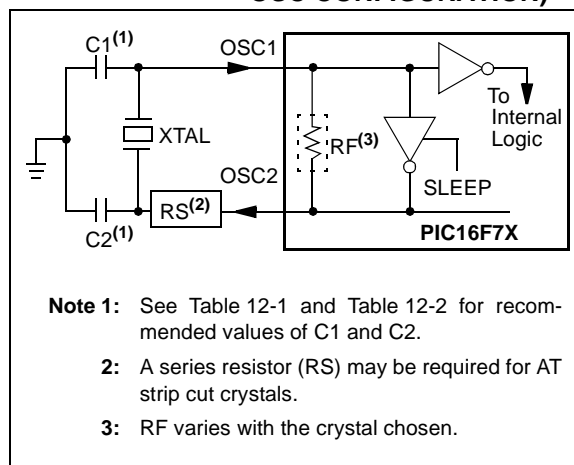
The PIC16F7X can be operated in four different oscillator modes. The user can program two configuration bits (FOSC1 and FOSC0) to select one of these four modes:

- LP Low Power Crystal
- XT Crystal/Resonator
- HS High Speed Crystal/Resonator
- RC Resistor/Capacitor

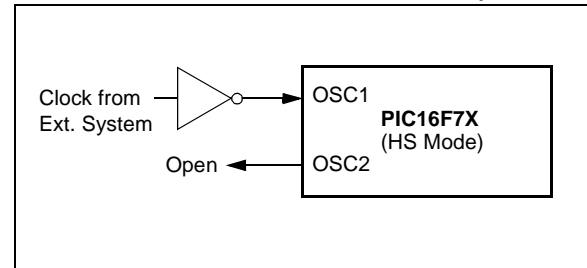
### 12.2.2 CRYSTAL OSCILLATOR/CERAMIC RESONATORS

In XT, LP or HS modes, a crystal or ceramic resonator is connected to the OSC1/CLKIN and OSC2/CLKOUT pins to establish oscillation (Figure 12-1). The PIC16F7X oscillator design requires the use of a parallel cut crystal. Use of a series cut crystal may give a frequency out of the crystal manufacturers specifications. When in HS mode, the device can accept an external clock source to drive the OSC1/CLKIN pin (Figure 12-2). See Figure 15-1 or Figure 15-2 (depending on the part number and VDD range) for valid external clock frequencies.

**FIGURE 12-1: CRYSTAL/CERAMIC RESONATOR OPERATION (HS, XT OR LP OSC CONFIGURATION)**



**FIGURE 12-2: EXTERNAL CLOCK INPUT OPERATION (HS OSC CONFIGURATION)**



**TABLE 12-1: CERAMIC RESONATORS (FOR DESIGN GUIDANCE ONLY)**

Typical Capacitor Values Used:			
Mode	Freq	OSC1	OSC2
XT	455 kHz	56 pF	56 pF
	2.0 MHz	47 pF	47 pF
	4.0 MHz	33 pF	33 pF
HS	8.0 MHz	27 pF	27 pF
	16.0 MHz	22 pF	22 pF

**Capacitor values are for design guidance only.**

These capacitors were tested with the resonators listed below for basic start-up and operation. These values were not optimized.

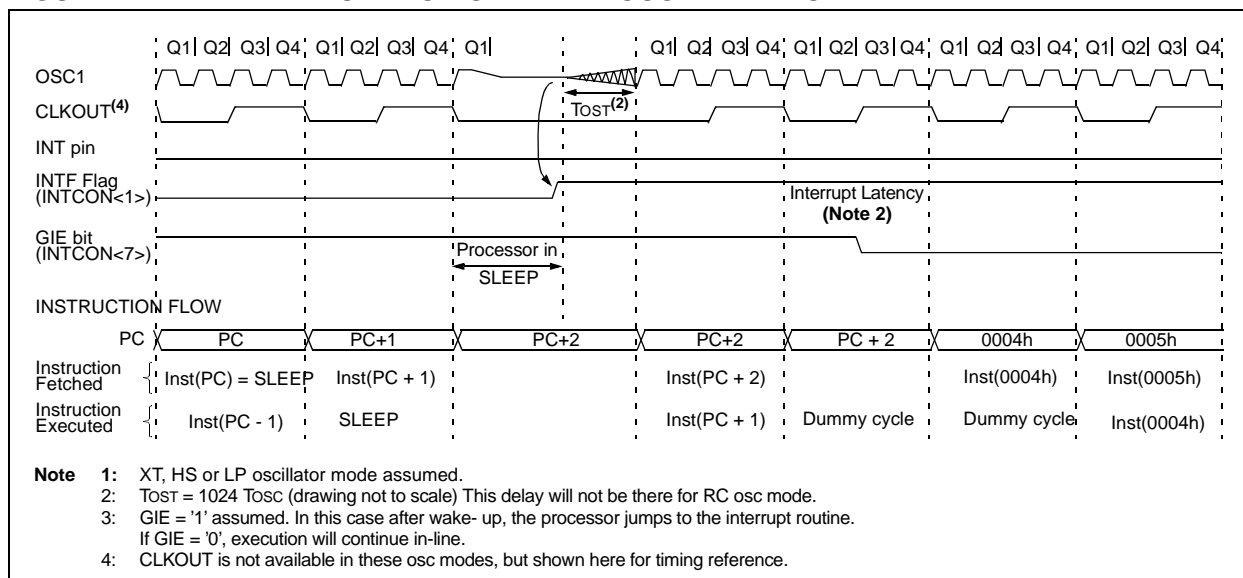
Different capacitor values may be required to produce acceptable oscillator operation. The user should test the performance of the oscillator over the expected VDD and temperature range for the application.

See the notes at the bottom of page 92 for additional information.

Resonators Used:	
455 kHz	Panasonic EFO-A455K04B
2.0 MHz	Murata Erie CSA2.00MG
4.0 MHz	Murata Erie CSA4.00MG
8.0 MHz	Murata Erie CSA8.00MT
16.0 MHz	Murata Erie CSA16.00MX



**FIGURE 12-12: WAKE-UP FROM SLEEP THROUGH INTERRUPT**



## 12.15 Program Verification/Code Protection

If the code protection bit(s) have not been programmed, the on-chip program memory can be read out for verification purposes.

## 12.16 ID Locations

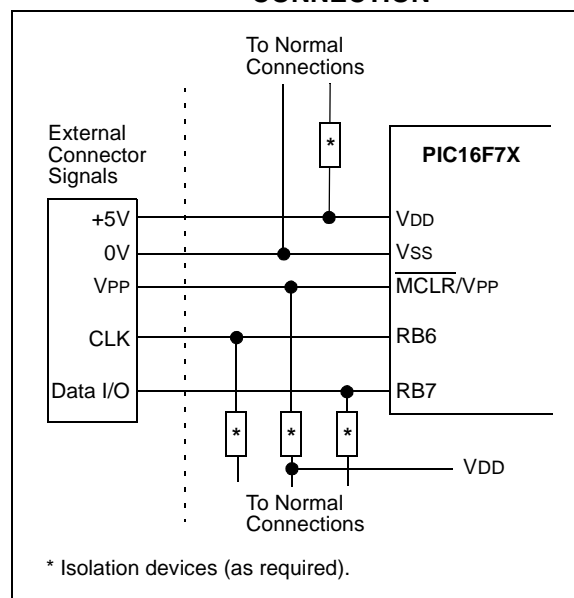
Four memory locations (2000h - 2003h) are designated as ID locations, where the user can store checksum or other code identification numbers. These locations are not accessible during normal execution, but are readable and writable during program/verify. It is recommended that only the 4 Least Significant bits of the ID location are used.

## 12.17 In-Circuit Serial Programming

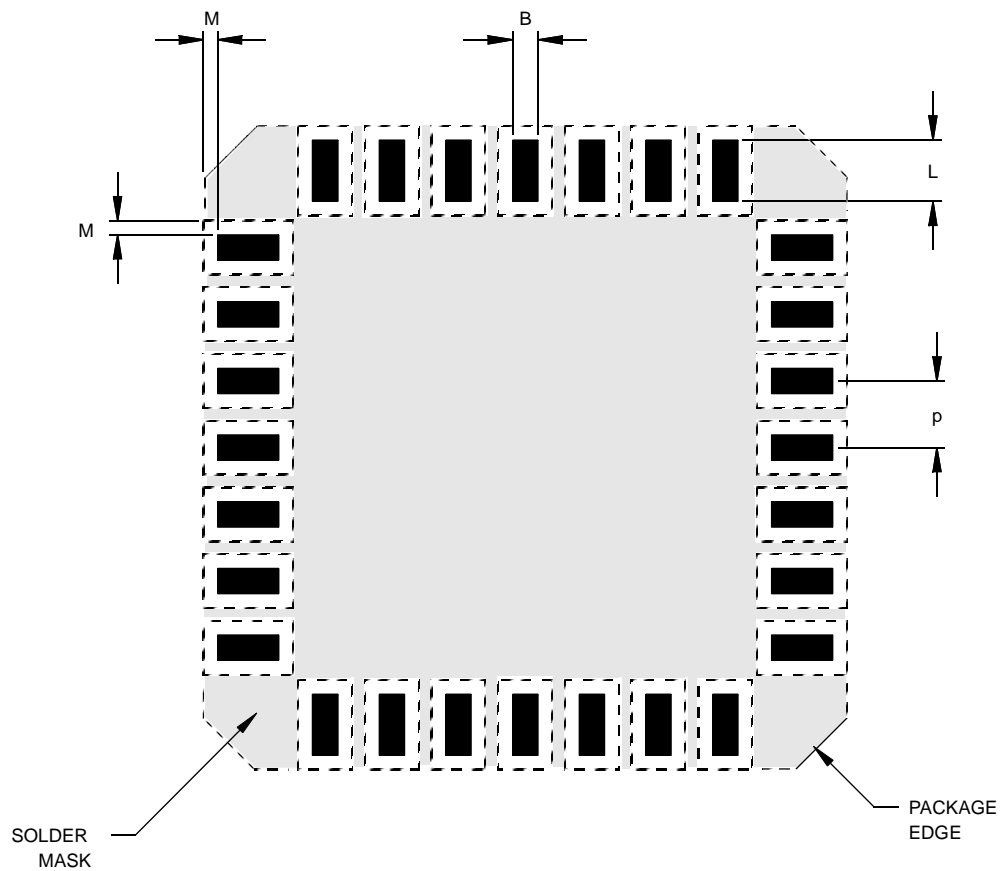
PIC16F7X microcontrollers can be serially programmed while in the end application circuit. This is simply done, with two lines for clock and data and three other lines for power, ground, and the programming voltage (see Figure 12-13 for an example). This allows customers to manufacture boards with unprogrammed devices, and then program the microcontroller just before shipping the product. This also allows the most recent firmware or a custom firmware to be programmed.

For general information of serial programming, please refer to the In-Circuit Serial Programming (ICSP™) Guide (DS30277). For specific details on programming commands and operations for the PIC16F7X devices, please refer to the latest version of the PIC16F7X FLASH Program Memory Programming Specification (DS30324).

**FIGURE 12-13: TYPICAL IN-CIRCUIT SERIAL PROGRAMMING CONNECTION**



## 28-Lead Plastic Micro Leadframe Package (MF) 6x6 mm Body (MLF) (Continued)



Units		INCHES			MILLIMETERS*		
Dimension Limits		MIN	NOM	MAX	MIN	NOM	MAX
Pitch	P	.026 BSC			0.65 BSC		
Pad Width	B	.009	.011	.014	0.23	0.28	0.35
Pad Length	L	.020	.024	.030	0.50	0.60	0.75
Pad to Solder Mask	M	.005		.006	0.13		0.15

\*Controlling Parameter

Drawing No. C04-2114

## APPENDIX A: REVISION HISTORY

Version	Date	Revision Description
A	2000	This is a new data sheet. However, these devices are similar to the PIC16C7X devices found in the PIC16C7X Data Sheet (DS30390) or the PIC16F87X devices (DS30292).
B	2001	Final data sheet. Includes device characterization data. Addition of extended temperature devices. Addition of 28-pin MLF package. Minor typographic revisions throughout.

## APPENDIX B: DEVICE DIFFERENCES

The differences between the devices in this data sheet are listed in Table B-1.

**TABLE B-1: DEVICE DIFFERENCES**

Difference	PIC16F73	PIC16F74	PIC16F76	PIC16F77
FLASH Program Memory (14-bit words)	4K	4K	8K	8K
Data Memory (bytes)	192	192	368	368
I/O Ports	3	5	3	5
A/D	5 channels, 8 bits	8 channels, 8 bits	5 channels, 8 bits	8 channels, 8 bits
Parallel Slave Port	no	yes	no	yes
Interrupt Sources	11	12	11	12
Packages	28-pin PDIP 28-pin SOIC 28-pin SSOP 28-pin MLF	40-pin PDIP 44-pin TQFP 44-pin PLCC	28-pin PDIP 28-pin SOIC 28-pin SSOP 28-pin MLF	40-pin PDIP 44-pin TQFP 44-pin PLCC

## INDEX

### A

#### A/D

A/D Conversion Status (GO/DONE Bit)	83
Acquisition Requirements	86
ADCON0 Register	83
ADCON1 Register	83
ADRES Register	83
Analog Port Pins	8, 10, 12, 39
Analog-to-Digital Converter	83
Associated Registers	88
Configuring Analog Port Pins	87
Configuring the Interrupt	85
Configuring the Module	85
Conversion Clock	87
Conversion Requirements	139
Conversions	87
Converter Characteristics	138
Effects of a RESET	87
Faster Conversion - Lower Resolution	
Trade-off	87
Internal Sampling Switch (Rss) Impedance	86
Operation During SLEEP	87
Source Impedance	86
Using the CCP Trigger	88
Absolute Maximum Ratings	119
ACK Pulse	65, 66
ADCON0 Register	83
GO/DONE Bit	83
ADCON1 Register	83
ADRES Register	83
Analog Port Pins. See A/D	
Application Notes	
AN552 (Implementing Wake-up on Key Strokes	
Using PIC16F7X)	33
AN556 (Implementing a Table Read)	26
AN578 (Use of the SSP Module in the I <sup>2</sup> C	
Multi-Master Environment)	59
AN607 (Power-up Trouble Shooting)	94
Assembler	
MPASM Assembler	113

### B

Banking, Data Memory	13
BF bit	60
Block Diagrams	
A/D	85
Analog Input Model	86
Capture Mode Operation	55
Compare	55
Crystal/Ceramic Resonator Operation (HS, XT	
or LP Osc Configuration)	91
External Clock Input Operation	
(HS Osc Configuration)	91
Interrupt Logic	99
PIC16F73 and PIC16F76	6
PIC16F74 and PIC16F77	7
PORTA	
RA3:RA0 and RA5 Port Pins	31
RA4/T0CKI Pin	31
PORTB	
RB3:RB0 Port Pins	33
RB7:RB4 Port Pins	33
PORTC (Peripheral Output Override)	35

PORTD (In I/O Port Mode)	36
PORTD and PORTE (Parallel Slave Port)	40
PORTE (In I/O Port Mode)	37
PWM Mode	57
RC Oscillator Mode	92
Recommended MCLR Circuit	94
Reset Circuit	93
SSP (I <sup>2</sup> C Mode)	65
SSP (SPI Mode)	62
Timer0/WDT Prescaler	43
Timer1	48
Timer2	51
Typical In-Circuit Serial Programming	
Connection	103
USART	
Receive	75
USART Transmit	73
Watchdog Timer (WDT)	101
BOR. See Brown-out Reset	
BRGH bit	71
Brown-out Reset (BOR)	89, 93, 94, 95, 96

### C

Capture/Compare/PWM (CCP)	
Associated Registers	56, 58
Capture Mode	55
Prescaler	55
CCP Pin Configuration	55, 56
CCP1	
RC2/CCP1 Pin	9, 11
CCP2	
RC1/T1OSI/CCP2 Pin	9, 11
Compare Mode	55
Software Interrupt Mode	56
Special Trigger Output	56
Timer1 Mode Selection	56
Example PWM Frequencies and Resolutions	58
Interaction of Two CCP Modules	53
PWM Duty Cycle	57
PWM Mode	57
PWM Period	57
Setup for PWM Operation	58
Special Event Trigger and A/D Conversions	56
Timer Resources	53
CCP1 Module	53
CCP2 Module	53
CCPR1H Register	53
CCPR1L Register	53
CCPxM<3:0> bits	54
CCPxX and CCPxY bits	54
CKE bit	60
CKP bit	61
Code Examples	
Call of a Subroutine in Page 1 from Page 0	26
Changing Between Capture Prescalers	55
Changing Prescaler Assignment to Timer0	45
Changing Prescaler Assignment to WDT	45
FLASH Program Read	30
Indirect Addressing	27
Initializing PORTA	31
Reading a 16-bit Free-Running Timer	49
Saving STATUS, W, and PCLATH Registers	
in RAM	100
Writing a 16-bit Free-Running Timer	49

# PIC16F7X

PORTE Register .....	37	RCSTA Register .....	
Postscaler, WDT .....		CREN bit .....	70
Assignment (PSA bit) .....	20	OERR bit .....	70
Rate Select (PS2:PS0 bits) .....	20	SPEN bit .....	69
Power-down Mode. See SLEEP .....		SREN bit .....	70
Power-on Reset (POR) .....	89, 93, 95, 96	RD0/PSP0 Pin .....	12
Oscillator Start-up Timer (OST) .....	89, 94	RD1/PSP1 Pin .....	12
POR Status ( $\overline{\text{POR}}$ bit) .....	25	RD2/PSP2 Pin .....	12
Power Control (PCON) Register .....	95	RD3/PSP3 Pin .....	12
Power-down ( $\overline{\text{PD}}$ bit) .....	93	RD4/PSP4 Pin .....	12
Power-up Timer (PWRT) .....	89, 94	RD5/PSP5 Pin .....	12
Time-out (TO bit) .....	19, 93	RD6/PSP6 Pin .....	12
PR2 Register .....	51	RD7/PSP7 Pin .....	12
Prescaler, Timer0 .....		RE0/RD/AN5 Pin .....	12
Assignment (PSA bit) .....	20	RE1/WR/AN6 Pin .....	12
Rate Select (PS2:PS0 bits) .....	20	RE2/ $\overline{\text{CS}}$ /AN7 Pin .....	12
PRO MATE II Universal Device Programmer .....	115	Read-Modify-Write Operations .....	105
Program Counter .....		Receive Overflow Indicator bit (SSPOV) .....	61
RESET Conditions .....	95	Register File .....	13
Program Memory .....	29	Registers .....	
Associated Registers .....	30	ADCON0 (A/D Control 0) .....	83
Interrupt Vector .....	13	ADCON0 (A/D Control 0) Register .....	83
Memory and Stack Maps .....	13	ADCON1 (A/D Control 1) .....	83
Operation During Code Protect .....	30	ADCON1 (A/D Control 1) Register .....	84
Organization .....	13	ADRES (A/D Result) .....	83
Paging .....	26	CCP1CON/CCP2CON (CCP Control) Registers .....	54
PMADR Register .....	29	Configuration Word Register .....	90
PMADRH Register .....	29	Initialization Conditions (table) .....	96–97
Reading FLASH .....	30	INTCON (Interrupt Control) .....	21
Reading, PMADR Register .....	29	INTCON (Interrupt Control) Register .....	21
Reading, PMADRH Register .....	29	OPTION_REG .....	20
Reading, PMCON1 Register .....	29	OPTION_REG Register .....	20, 44
Reading, PMDATA Register .....	29	PCON (Power Control) .....	25
Reading, PMDATH Register .....	29	PCON (Power Control) Register .....	25
RESET Vector .....	13	PIE1 (Peripheral Interrupt Enable 1) .....	22
Program Verification .....	103	PIE1 (Peripheral Interrupt Enable 1) Register .....	22
Programming Pin (VPP) .....	8, 10	PIE2 (Peripheral Interrupt Enable 2) .....	24
Programming, Device Instructions .....	105	PIE2 (Peripheral Interrupt Enable 2) Register .....	24
PUSH .....	26	PIR1 (Peripheral Interrupt Request 1) .....	23
		PIR1 (Peripheral Interrupt Request 1) Register .....	23
		PIR2 (Peripheral Interrupt Request 2) .....	24
		PIR2 (Peripheral Interrupt Request 2) Register .....	24
		PMCON1 (Program Memory Control 1) .....	
		Register .....	29
		RCSTA (Receive Status and Control) Register .....	70
		Special Function, Summary .....	16–18
		SSPCON (Sync Serial Port Control) Register .....	61
		SSPSTAT (Sync Serial Port Status) Register .....	60
		STATUS Register .....	19
		T1CON (Timer 1 Control) Register .....	47
		T2CON (Timer2 Control) Register .....	52
		TRISE Register .....	38
		TXSTA (Transmit Status and Control) Register .....	69
		RESET .....	89, 93
		Brown-out Reset (BOR). See Brown-out Reset (BOR) .....	
		MCLR Reset. See MCLR .....	
		Power-on Reset (POR). See Power-on Reset (POR) .....	
		RESET Conditions for All Registers .....	96
		RESET Conditions for PCON Register .....	95
		RESET Conditions for Program Counter .....	95
		RESET Conditions for STATUS Register .....	95
		RESET .....	
		WDT Reset. See Watchdog Timer (WDT) .....	
		Revision History .....	161