Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

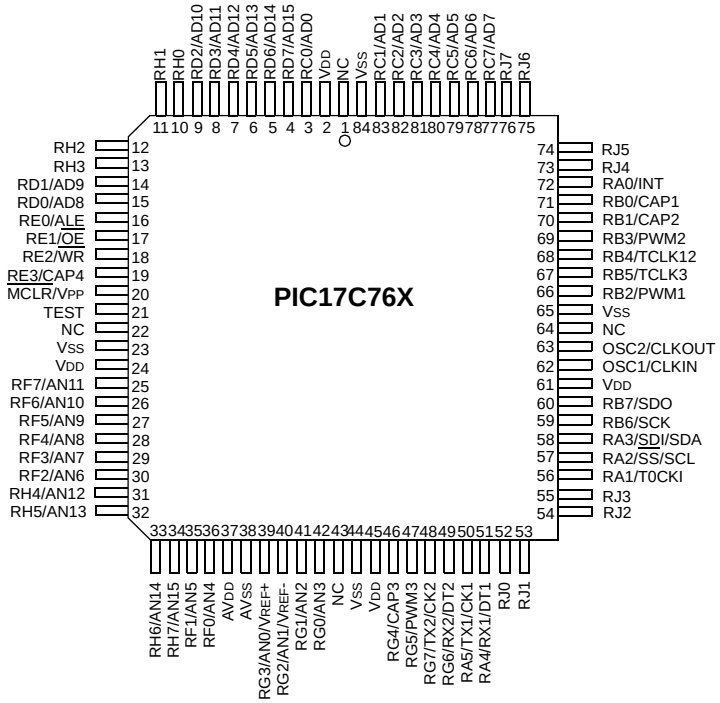
"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" 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"

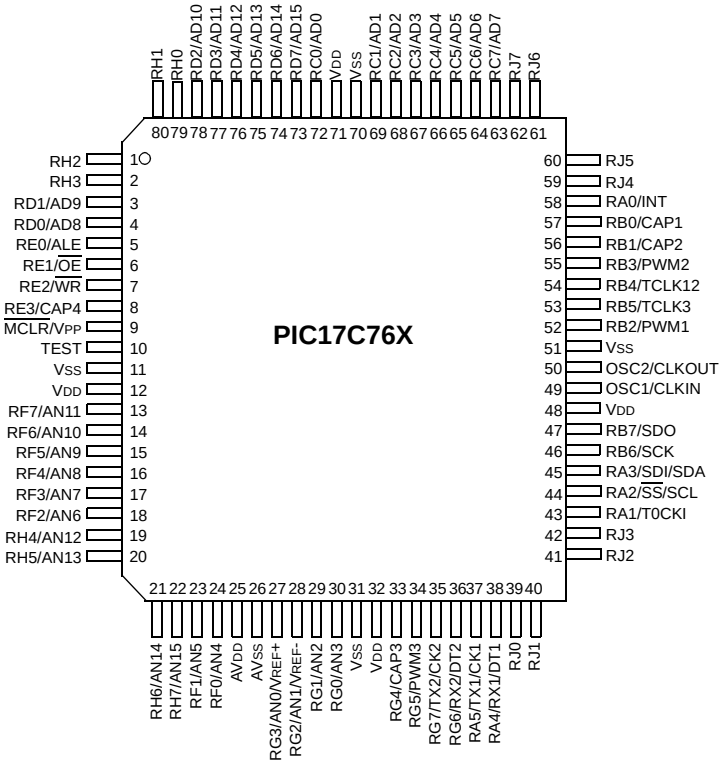
Details	
Product Status	Obsolete
Core Processor	PIC
Core Size	8-Bit
Speed	33MHz
Connectivity	I²C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	66
Program Memory Size	32KB (16K x 16)
Program Memory Type	OTP
EEPROM Size	-
RAM Size	902 x 8
Voltage - Supply (Vcc/Vdd)	4.5V ~ 5.5V
Data Converters	A/D 16x10b
Oscillator Type	External
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	84-LCC (J-Lead)
Supplier Device Package	84-PLCC (29.31x29.31)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic17c766t-33e-l

Pin Diagrams cont.'d

84-pin PLCC



80-Pin TQFP



4.1.6 RC OSCILLATOR

For timing insensitive applications, the RC device option offers additional cost savings. RC oscillator frequency is a function of the supply voltage, the resistor (R_{EXT}) and capacitor (C_{EXT}) values, and the operating temperature. In addition to this, oscillator frequency will vary from unit to unit due to normal process parameter variation. Furthermore, the difference in lead frame capacitance between package types will also affect oscillation frequency, especially for low C_{EXT} values. The user also needs to take into account variation due to tolerance of external R and C components used. Figure 4-7 shows how the R/C combination is connected to the PIC17CXXX. For R_{EXT} values below 2.2 kΩ, the oscillator operation may become unstable, or stop completely. For very high R_{EXT} values (e.g. 1 MΩ), the oscillator becomes sensitive to noise, humidity and leakage. Thus, we recommend to keep R_{EXT} between 3 kΩ and 100 kΩ.

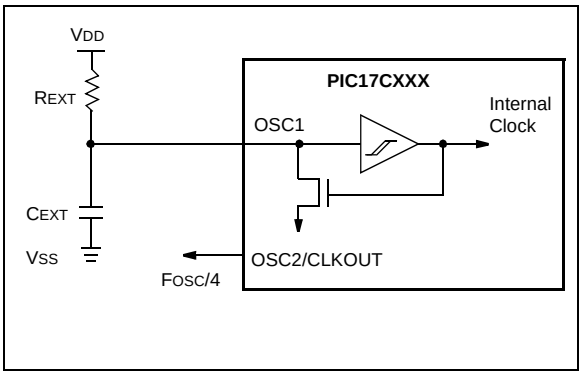
Although the oscillator will operate with no external capacitor (C_{EXT} = 0 pF), we recommend using values above 20 pF for noise and stability reasons. With little or no external capacitance, oscillation frequency can vary dramatically due to changes in external capacitances, such as PCB trace capacitance or package lead frame capacitance.

See Section 21.0 for RC frequency variation from part to part due to normal process variation. The variation is larger for larger R (since leakage current variation will affect RC frequency more for large R) and for smaller C (since variation of input capacitance will affect RC frequency more).

See Section 21.0 for variation of oscillator frequency due to V_{DD} for given R_{EXT}/C_{EXT} values, as well as frequency variation due to operating temperature for given R, C, and V_{DD} values.

The oscillator frequency, divided by 4, is available on the OSC2/CLKOUT pin and can be used for test purposes or to synchronize other logic (see Figure 4-8 for waveform).

FIGURE 4-7: RC OSCILLATOR MODE



4.1.6.1 RC Start-up

As the device voltage increases, the RC will immediately start its oscillations once the pin voltage levels meet the input threshold specifications (parameter #D032 and parameter #D042 in the electrical specification section). The time required for the RC to start oscillating depends on many factors. These include:

- Resistor value used
- Capacitor value used
- Device V_{DD} rise time
- System temperature

TABLE 5-4: INITIALIZATION CONDITIONS FOR SPECIAL FUNCTION REGISTERS

Register	Address	Power-on Reset Brown-out Reset	$\overline{\text{MCLR}}$ Reset WDT Reset	Wake-up from SLEEP through Interrupt
Unbanked				
INDF0	00h	N/A	N/A	N/A
FSR0	01h	xxxx xxxx	uuuu uuuu	uuuu uuuu
PCL	02h	0000h	0000h	PC + 1(2)
PCLATH	03h	0000 0000	uuuu uuuu	uuuu uuuu
ALUSTA	04h	1111 xxxx	1111 uuuu	1111 uuuu
T0STA	05h	0000 000-	0000 000-	0000 000-
CPUSTA(3)	06h	--11 11qq	--11 qquu	--uu qquu
INTSTA	07h	0000 0000	0000 0000	uuuu uuuu(1)
INDF1	08h	N/A	N/A	N/A
FSR1	09h	xxxx xxxx	uuuu uuuu	uuuu uuuu
WREG	0Ah	xxxx xxxx	uuuu uuuu	uuuu uuuu
TMR0L	0Bh	xxxx xxxx	uuuu uuuu	uuuu uuuu
TMR0H	0Ch	xxxx xxxx	uuuu uuuu	uuuu uuuu
TBLPTRL	0Dh	0000 0000	0000 0000	uuuu uuuu
TBLPTRH	0Eh	0000 0000	0000 0000	uuuu uuuu
BSR	0Fh	0000 0000	0000 0000	uuuu uuuu
Bank 0				
PORTA(4,6)	10h	0-xx 11xx	0-uu 11uu	u-uu uuuu
DDRB	11h	1111 1111	1111 1111	uuuu uuuu
PORTB(4)	12h	xxxx xxxx	uuuu uuuu	uuuu uuuu
RCSTA1	13h	0000 -00x	0000 -00u	uuuu -uuu
RCREG1	14h	xxxx xxxx	uuuu uuuu	uuuu uuuu
TXSTA1	15h	0000 --1x	0000 --1u	uuuu --uu
TXREG1	16h	xxxx xxxx	uuuu uuuu	uuuu uuuu
SPBRG1	17h	0000 0000	0000 0000	uuuu uuuu

Legend: u = unchanged, x = unknown, - = unimplemented, read as '0', q = value depends on condition

Note 1: One or more bits in INTSTA, PIR1, PIR2 will be affected (to cause wake-up).

2: When the wake-up is due to an interrupt and the GLINTD bit is cleared, the PC is loaded with the interrupt vector.

3: See Table 5-3 for RESET value of specific condition.

4: This is the value that will be in the port output latch.

5: When the device is configured for Microprocessor or Extended Microcontroller mode, the operation of this port does not rely on these registers.

6: On any device RESET, these pins are configured as inputs.

7.0 MEMORY ORGANIZATION

There are two memory blocks in the PIC17C7XX; program memory and data memory. Each block has its own bus, so that access to each block can occur during the same oscillator cycle.

The data memory can further be broken down into General Purpose RAM and the Special Function Registers (SFRs). The operation of the SFRs that control the "core" are described here. The SFRs used to control the peripheral modules are described in the section discussing each individual peripheral module.

7.1 Program Memory Organization

PIC17C7XX devices have a 16-bit program counter capable of addressing a 64K x 16 program memory space. The RESET vector is at 0000h and the interrupt vectors are at 0008h, 0010h, 0018h, and 0020h (Figure 7-1).

7.1.1 PROGRAM MEMORY OPERATION

The PIC17C7XX can operate in one of four possible program memory configurations. The configuration is selected by configuration bits. The possible modes are:

- Microprocessor
- Microcontroller
- Extended Microcontroller
- Protected Microcontroller

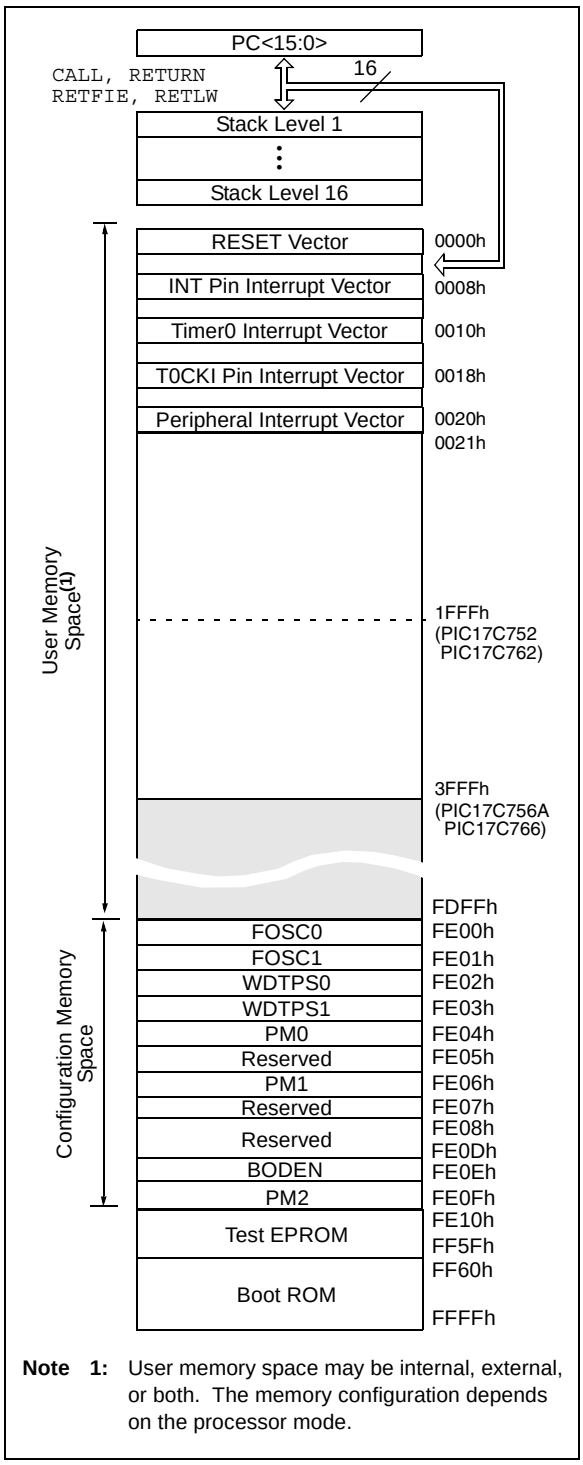
The **Microcontroller** and **Protected Microcontroller** modes only allow internal execution. Any access beyond the program memory reads unknown data. The Protected Microcontroller mode also enables the code protection feature.

The **Extended Microcontroller** mode accesses both the internal program memory, as well as external program memory. Execution automatically switches between internal and external memory. The 16-bits of address allow a program memory range of 64K-words.

The **Microprocessor** mode only accesses the external program memory. The on-chip program memory is ignored. The 16-bits of address allow a program memory range of 64K-words. Microprocessor mode is the default mode of an unprogrammed device.

The different modes allow different access to the configuration bits, test memory and boot ROM. Table 7-1 lists which modes can access which areas in memory. Test Memory and Boot Memory are not required for normal operation of the device. Care should be taken to ensure that no unintended branches occur to these areas.

FIGURE 7-1: PROGRAM MEMORY MAP AND STACK



Note 1: User memory space may be internal, external, or both. The memory configuration depends on the processor mode.

FIGURE 7-5: PIC17C7XX REGISTER FILE MAP

Addr	Unbanked
00h	INDF0
01h	FSR0
02h	PCL
03h	PCLATH
04h	ALUSTA
05h	T0STA
06h	CPUSTA
07h	INTSTA
08h	INDF1
09h	FSR1
0Ah	WREG
0Bh	TMR0L
0Ch	TMR0H
0Dh	TBLPTRL
0Eh	TBLPTRH
0Fh	BSR

	Bank 0	Bank 1(1)	Bank 2(1)	Bank 3(1)	Bank 4(1)	Bank 5(1)	Bank 6(1)	Bank 7(1)	Bank 8(1,4)
10h	PORTA	DDRC	TMR1	PW1DCL	PIR2	DDRF	SSPADD	PW3DCL	DDRH
11h	DDRB	PORTC	TMR2	PW2DCL	PIE2	PORTF	SSPCON1	PW3DCH	PORTH
12h	PORTB	DDRD	TMR3L	PW1DCH	—	DDRG	SSPCON2	CA3L	DDRJ
13h	RCSTA1	PORTD	TMR3H	PW2DCH	RCSTA2	PORTG	SSPSTAT	CA3H	PORTJ
14h	RCREG1	DDRE	PR1	CA2L	RCREG2	ADCON0	SSPBUF	CA4L	—
15h	TXSTA1	PORTE	PR2	CA2H	TXSTA2	ADCON1	—	CA4H	—
16h	TXREG1	PIR1	PR3L/CA1L	TCON1	TXREG2	ADRESL	—	TCON3	—
17h	SPBRG1	PIE1	PR3H/CA1H	TCON2	SPBRG2	ADRESH	—	—	—

	Unbanked
18h	PRODL
19h	PRODH
1Ah 1Fh	General Purpose RAM

	Bank 0(2)	Bank 1(2)	Bank 2(2)	Bank 3(2,3)
20h FFh	General Purpose RAM	General Purpose RAM	General Purpose RAM	General Purpose RAM

Note 1: SFR file locations 10h - 17h are banked. The lower nibble of the BSR specifies the bank. All unbanked SFRs ignore the Bank Select Register (BSR) bits.

2: General Purpose Registers (GPR) locations 20h - FFh, 120h - 1FFh, 220h - 2FFh, and 320h - 3FFh are banked. The upper nibble of the BSR specifies this bank. All other GPRs ignore the Bank Select Register (BSR) bits.

3: RAM bank 3 is not implemented on the PIC17C752 and the PIC17C762. Reading any unimplemented register reads ‘0’s.

4: Bank 8 is only implemented on the PIC17C76X devices.

TABLE 7-3: SPECIAL FUNCTION REGISTERS (CONTINUED)

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	$\overline{\text{MCLR}}$, WDT
Bank 6											
10h	SSPADD	SSP Address Register in I^2C Slave mode. SSP Baud Rate Reload Register in I^2C Master mode								0000 0000	0000 0000
11h	SSPCON1	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	0000 0000
12h	SSPCON2	GCEN	AKSTAT	AKDT	AKEN	RCEN	PEN	RSEN	SEN	0000 0000	0000 0000
13h	SSPSTAT	SMP	CKE	D/$\overline{\text{A}}$	P	S	R/$\overline{\text{W}}$	UA	BF	0000 0000	0000 0000
14h	SSPBUF	Synchronous Serial Port Receive Buffer/Transmit Register								xxxx xxxx	uuuu uuuu
15h	Unimplemented	—	—	—	—	—	—	—	—	---- ----	---- ----
16h	Unimplemented	—	—	—	—	—	—	—	—	---- ----	---- ----
17h	Unimplemented	—	—	—	—	—	—	—	—	---- ----	---- ----
Bank 7											
10h	PW3DCL	DC1	DC0	TM2PW3	—	—	—	—	—	xx0- ----	uu0- ----
11h	PW3DCH	DC9	DC8	DC7	DC6	DC5	DC4	DC3	DC2	xxxx xxxx	uuuu uuuu
12h	CA3L	Capture3 Low Byte								xxxx xxxx	uuuu uuuu
13h	CA3H	Capture3 High Byte								xxxx xxxx	uuuu uuuu
14h	CA4L	Capture4 Low Byte								xxxx xxxx	uuuu uuuu
15h	CA4H	Capture4 High Byte								xxxx xxxx	uuuu uuuu
16h	TCON3	—	CA4OVF	CA3OVF	CA4ED1	CA4ED0	CA3ED1	CA3ED0	PWM3ON	-000 0000	-000 0000
17h	Unimplemented	—	—	—	—	—	—	—	—	---- ----	---- ----
Bank 8(3)											
10h(3)	DDRH	Data Direction Register for PORTH								1111 1111	1111 1111
11h(3)	PORTH(4)	RH7/ AN15	RH6/ AN14	RH5/ AN13	RH4/ AN12	RH3	RH2	RH1	RH0	xxxx xxxx	uuuu uuuu
12h(3)	DDRJ	Data Direction Register for PORTJ								1111 1111	1111 1111
13h(3)	PORTJ(4)	RJ7	RJ6	RJ5	RJ4	RJ3	RJ2	RJ1	RJ0	xxxx xxxx	uuuu uuuu
14h(3)	Unimplemented	—	—	—	—	—	—	—	—	---- ----	---- ----
15h(3)	Unimplemented	—	—	—	—	—	—	—	—	---- ----	---- ----
16h(3)	Unimplemented	—	—	—	—	—	—	—	—	---- ----	---- ----
17h(3)	Unimplemented	—	—	—	—	—	—	—	—	---- ----	---- ----
Unbanked											
18h	PRODL	Low Byte of 16-bit Product (8 x 8 Hardware Multiply)								xxxx xxxx	uuuu uuuu
19h	PRODH	High Byte of 16-bit Product (8 x 8 Hardware Multiply)								xxxx xxxx	uuuu uuuu

Legend: x = unknown, u = unchanged, - = unimplemented, read as '0', q = value depends on condition.
Shaded cells are unimplemented, read as '0'.

Note 1: The upper byte of the program counter is not directly accessible. PCLATH is a holding register for PC<15:8> whose contents are updated from, or transferred to, the upper byte of the program counter.

2: The $\overline{\text{TO}}$ and $\overline{\text{PD}}$ status bits in CPUSTA are not affected by a $\overline{\text{MCLR}}$ Reset.

3: Bank 8 and associated registers are only implemented on the PIC17C76X devices.

4: This is the value that will be in the port output latch.

5: When the device is configured for Microprocessor or Extended Microcontroller mode, the operation of this port does not rely on these registers.

6: On any device RESET, these pins are configured as inputs.

13.0 TIMER1, TIMER2, TIMER3, PWMS AND CAPTURES

The PIC17C7XX has a wealth of timers and time based functions to ease the implementation of control applications. These time base functions include three PWM outputs and four Capture inputs.

Timer1 and Timer2 are two 8-bit incrementing timers, each with an 8-bit period register (PR1 and PR2, respectively) and separate overflow interrupt flags. Timer1 and Timer2 can operate either as timers (increment on internal FOSC/4 clock), or as counters (increment on falling edge of external clock on pin RB4/TCLK12). They are also software configurable to operate as a single 16-bit timer/counter. These timers are also used as the time base for the PWM (Pulse Width Modulation) modules.

Timer3 is a 16-bit timer/counter which uses the TMR3H and TMR3L registers. Timer3 also has two additional registers (PR3H/CA1H:PR3L/CA1L) that are configurable as a 16-bit period register or a 16-bit capture register. TMR3 can be software configured to increment from the internal system clock (FOSC/4), or from an external signal on the RB5/TCLK3 pin. Timer3 is the time base for all of the 16-bit captures.

Six other registers comprise the Capture2, Capture3, and Capture4 registers (CA2H:CA2L, CA3H:CA3L, and CA4H:CA4L).

Figure 13-1, Figure 13-2 and Figure 13-3 are the control registers for the operation of Timer1, Timer2 and Timer3, as well as PWM1, PWM2, PWM3, Capture1, Capture2, Capture3 and Capture4.

Table 13-1 shows the Timer resource requirements for these time base functions. Each timer is an open resource so that multiple functions may operate with it.

TABLE 13-1: TIME-BASE FUNCTION/ RESOURCE REQUIREMENTS

Time Base Function	Timer Resource
PWM1	Timer1
PWM2	Timer1 or Timer2
PWM3	Timer1 or Timer2
Capture1	Timer3
Capture2	Timer3
Capture3	Timer3
Capture4	Timer3

REGISTER 13-1: TCON1 REGISTER (ADDRESS: 16h, BANK 3)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CA2ED1	CA2ED0	CA1ED1	CA1ED0	T16	TMR3CS	TMR2CS	TMR1CS
bit 7							bit 0

bit 7-6 **CA2ED1:CA2ED0**: Capture2 Mode Select bits
00 = Capture on every falling edge
01 = Capture on every rising edge
10 = Capture on every 4th rising edge
11 = Capture on every 16th rising edge

bit 5-4 **CA1ED1:CA1ED0**: Capture1 Mode Select bits
00 = Capture on every falling edge
01 = Capture on every rising edge
10 = Capture on every 4th rising edge
11 = Capture on every 16th rising edge

bit 3 **T16**: Timer2:Timer1 Mode Select bit
1 = Timer2 and Timer1 form a 16-bit timer
0 = Timer2 and Timer1 are two 8-bit timers

bit 2 **TMR3CS**: Timer3 Clock Source Select bit
1 = TMR3 increments off the falling edge of the RB5/TCLK3 pin
0 = TMR3 increments off the internal clock

bit 1 **TMR2CS**: Timer2 Clock Source Select bit
1 = TMR2 increments off the falling edge of the RB4/TCLK12 pin
0 = TMR2 increments off the internal clock

bit 0 **TMR1CS**: Timer1 Clock Source Select bit
1 = TMR1 increments off the falling edge of the RB4/TCLK12 pin
0 = TMR1 increments off the internal clock

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

13.1.3.1 PWM Periods

The period of the PWM1 output is determined by Timer1 and its period register (PR1). The period of the PWM2 and PWM3 outputs can be individually software configured to use either Timer1 or Timer2 as the time-base. For PWM2, when TM2PW2 bit (PW2DCL<5>) is clear, the time base is determined by TMR1 and PR1 and when TM2PW2 is set, the time base is determined by Timer2 and PR2. For PWM3, when TM2PW3 bit (PW3DCL<5>) is clear, the time base is determined by TMR1 and PR1, and when TM2PW3 is set, the time base is determined by Timer2 and PR2.

Running two different PWM outputs on two different timers allows different PWM periods. Running all PWMs from Timer1 allows the best use of resources by freeing Timer2 to operate as an 8-bit timer. Timer1 and Timer2 cannot be used as a 16-bit timer if any PWM is being used.

The PWM periods can be calculated as follows:

$$\text{period of PWM1} = [(PR1) + 1] \times 4T_{OSC}$$

$$\text{period of PWM2} = [(PR1) + 1] \times 4T_{OSC} \text{ or } [(PR2) + 1] \times 4T_{OSC}$$

$$\text{period of PWM3} = [(PR1) + 1] \times 4T_{OSC} \text{ or } [(PR2) + 1] \times 4T_{OSC}$$

The duty cycle of PWMx is determined by the 10-bit value DCx<9:0>. The upper 8-bits are from register PWxDCH and the lower 2-bits are from PWxDCL<7:6> (PWxDCH:PWxDCL<7:6>). Table 13-4 shows the maximum PWM frequency (F_{PWM}), given the value in the period register.

The number of bits of resolution that the PWM can achieve depends on the operation frequency of the device as well as the PWM frequency (F_{PWM}).

Maximum PWM resolution (bits) for a given PWM frequency:

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

where: F_{PWM} = 1 / period of PWM

The PWMx duty cycle is as follows:

$$\text{PWMx Duty Cycle} = (DCx) \times T_{OSC}$$

where DCx represents the 10-bit value from PWxDCH:PWxDCL.

If DCx = 0, then the duty cycle is zero. If PRx = PWxDCH, then the PWM output will be low for one to four Q-clocks (depending on the state of the PWxDCL<7:6> bits). For a duty cycle to be 100%, the PWxDCH value must be greater then the PRx value.

The duty cycle registers for both PWM outputs are double buffered. When the user writes to these registers, they are stored in master latches. When TMR1 (or TMR2) overflows and a new PWM period begins, the master latch values are transferred to the slave latches and the PWMx pin is forced high.

Note:	For PW1DCH, PW1DCL, PW2DCH, PW2DCL, PW3DCH and PW3DCL registers, a write operation writes to the "master latches", while a read operation reads the "slave latches". As a result, the user may not read back what was just written to the duty cycle registers (until transferred to slave latch).

The user should also avoid any "read-modify-write" operations on the duty cycle registers, such as: `ADDWF PW1DCH`. This may cause duty cycle outputs that are unpredictable.

TABLE 13-4: PWM FREQUENCY vs. RESOLUTION AT 33 MHz

PWM Frequency	Frequency (kHz)				
	32.2	**64.5**	**90.66**	**128.9**	**515.6**
PRx Value	0xFF	0x7F	0x5A	0x3F	0x0F
High Resolution	10-bit	9-bit	8.5-bit	8-bit	6-bit
Standard Resolution	8-bit	7-bit	6.5-bit	6-bit	4-bit

13.1.3.2 PWM INTERRUPTS

The PWM modules make use of the TMR1 and/or TMR2 interrupts. A timer interrupt is generated when TMR1 or TMR2 equals its period register and on the following increment is cleared to zero. This interrupt also marks the beginning of a PWM cycle. The user can write new duty cycle values before the timer rollover. The TMR1 interrupt is latched into the TMR1IF bit and the TMR2 interrupt is latched into the TMR2IF bit. These flags must be cleared in software.

13.2 Timer3

Timer3 is a 16-bit timer consisting of the TMR3H and TMR3L registers. TMR3H is the high byte of the timer and TMR3L is the low byte. This timer has an associated 16-bit period register (PR3H/CA1H:PR3L/CA1L). This period register can be software configured to be a another 16-bit capture register.

When the TMR3CS bit (TCON1<2>) is clear, the timer increments every instruction cycle (FOSC/4). When TMR3CS is set, the counter increments on every falling edge of the RB5/TCLK3 pin. In either mode, the TMR3ON bit must be set for the timer/counter to increment. When TMR3ON is clear, the timer will not increment or set flag bit TMR3IF.

Timer3 has two modes of operation, depending on the CA1/PR3 bit (TCON2<3>). These modes are:

- Three capture and one period register mode
- Four capture register mode

The PIC17C7XX has up to four 16-bit capture registers that capture the 16-bit value of TMR3 when events are detected on capture pins. There are four capture pins (RB0/CAP1, RB1/CAP2, RG4/CAP3, and RE3/CAP4), one for each capture register pair. The capture pins are multiplexed with the I/O pins. An event can be:

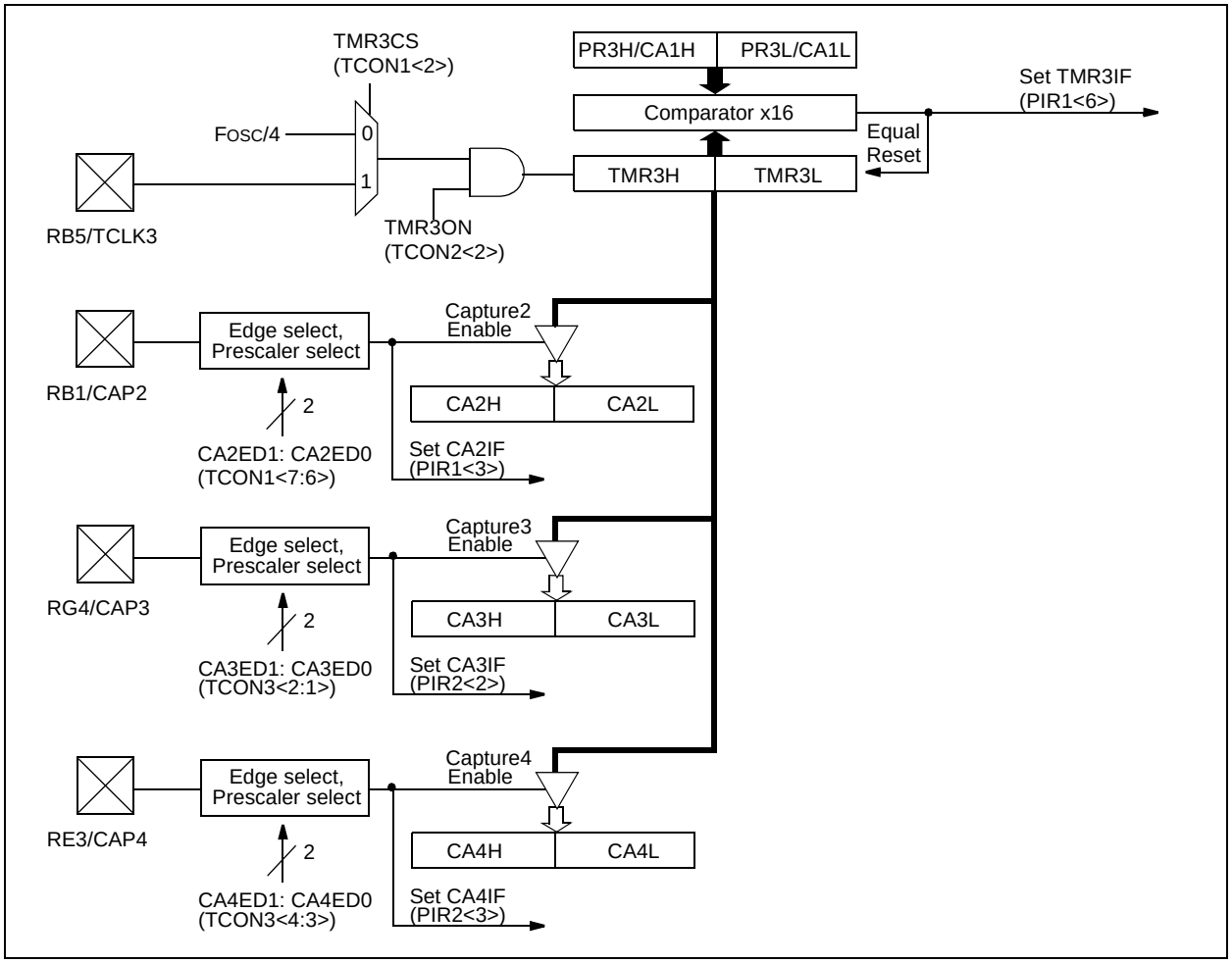
- A rising edge
- A falling edge
- Every 4th rising edge
- Every 16th rising edge

Each 16-bit capture register has an interrupt flag associated with it. The flag is set when a capture is made. The capture modules are truly part of the Timer3 block. Figure 13-5 and Figure 13-6 show the block diagrams for the two modes of operation.

13.2.1 THREE CAPTURE AND ONE PERIOD REGISTER MODE

In this mode, registers PR3H/CA1H and PR3L/CA1L constitute a 16-bit period register. A block diagram is shown in Figure 13-5. The timer increments until it equals the period register and then resets to 0000h on the next timer clock. TMR3 Interrupt Flag bit (TMR3IF) is set at this point. This interrupt can be disabled by clearing the TMR3 Interrupt Enable bit (TMR3IE). TMR3IF must be cleared in software.

FIGURE 13-5: TIMER3 WITH THREE CAPTURE AND ONE PERIOD REGISTER BLOCK DIAGRAM



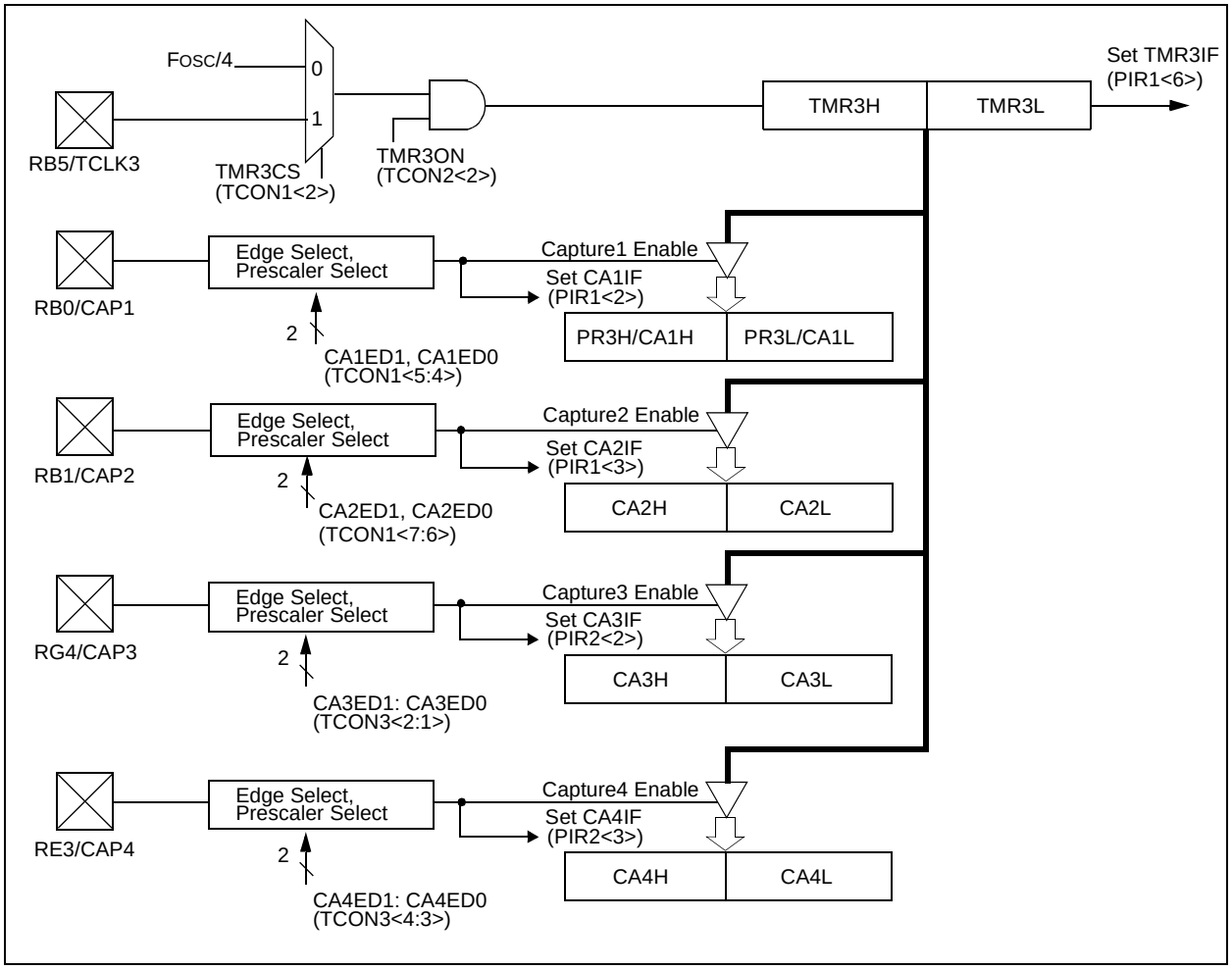
13.2.2 FOUR CAPTURE MODE

This mode is selected by setting bit CA1/$\overline{PR3}$. A block diagram is shown in Figure 13-6. In this mode, TMR3 runs without a period register and increments from 0000h to FFFFh and rolls over to 0000h. The TMR3 interrupt Flag (TMR3IF) is set on this rollover. The TMR3IF bit must be cleared in software.

Registers PR3H/CA1H and PR3L/CA1L make a 16-bit capture register (Capture1). It captures events on pin RB0/CAP1. Capture mode is configured by the CA1ED1 and CA1ED0 bits. Capture1 Interrupt Flag bit (CA1IF) is set upon detection of the capture event. The corresponding interrupt mask bit is CA1IE. The Capture1 Overflow Status bit is CA1OVF.

All the captures operate in the same manner. Refer to Section 13.2.1 for the operation of capture.

FIGURE 13-6: TIMER3 WITH FOUR CAPTURES BLOCK DIAGRAM



14.2 USART Asynchronous Mode

In this mode, the USART uses standard nonreturn-to-zero (NRZ) format (one START bit, eight or nine data bits, and one STOP bit). The most common data format is 8-bits. An on-chip dedicated 8-bit baud rate generator can be used to derive standard baud rate frequencies from the oscillator. The USART's transmitter and receiver are functionally independent but use the same data format and baud rate. The baud rate generator produces a clock x64 of the bit shift rate. Parity is not supported by the hardware, but can be implemented in software (and stored as the ninth data bit). Asynchronous mode is stopped during SLEEP.

The Asynchronous mode is selected by clearing the SYNC bit (TXSTA<4>).

The USART Asynchronous module consists of the following components:

- Baud Rate Generator
- Sampling Circuit
- Asynchronous Transmitter
- Asynchronous Receiver

14.2.1 USART ASYNCHRONOUS TRANSMITTER

The USART transmitter block diagram is shown in Figure 14-1. The heart of the transmitter is the transmit shift register (TSR). The shift register obtains its data from the read/write transmit buffer (TXREG). TXREG is loaded with data in software. The TSR is not loaded until the STOP bit has been transmitted from the previous load. As soon as the STOP bit is transmitted, the TSR is loaded with new data from the TXREG (if available). Once TXREG transfers the data to the TSR (occurs in one TCY at the end of the current BRG cycle), the TXREG is empty and an interrupt bit, TXIF, is set. This interrupt can be enabled/disabled by setting/clearing the TXIE bit. TXIF will be set, regardless of TXIE and cannot be reset in software. It will reset only when new data is loaded into TXREG. While TXIF indicates the status of the TXREG, the TRMT (TXSTA<1>) bit shows the status of the TSR.

TRMT is a read only bit which is set when the TSR is empty. No interrupt logic is tied to this bit, so the user has to poll this bit in order to determine if the TSR is empty.

Note:	The TSR is not mapped in data memory, so it is not available to the user.

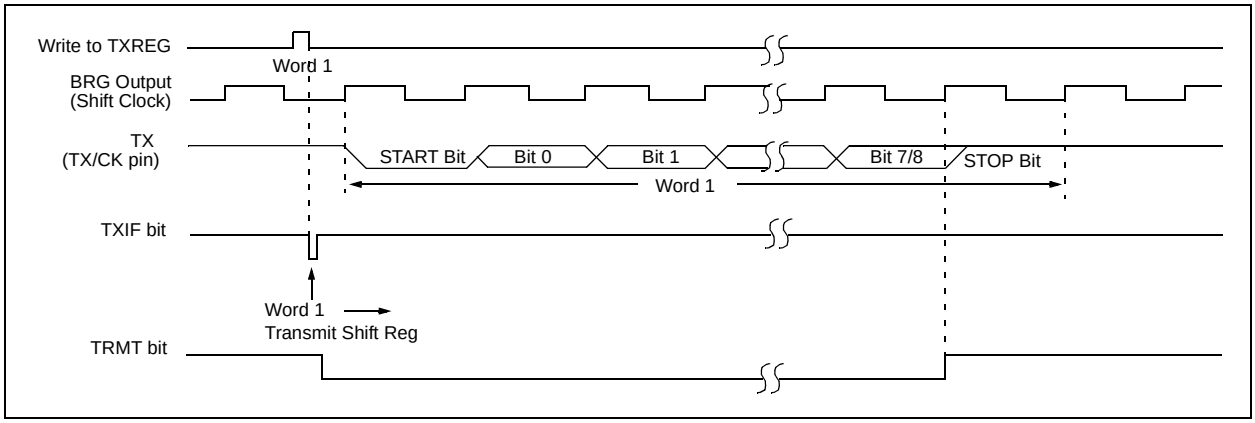
Transmission is enabled by setting the TXEN (TXSTA<5>) bit. The actual transmission will not occur until TXREG has been loaded with data and the baud rate generator (BRG) has produced a shift clock (Figure 14-3). The transmission can also be started by first loading TXREG and then setting TXEN. Normally, when transmission is first started, the TSR is empty, so a transfer to TXREG will result in an immediate transfer to TSR, resulting in an empty TXREG. A back-to-back transfer is thus possible (Figure 14-4). Clearing TXEN during a transmission will cause the transmission to be aborted. This will reset the transmitter and the TX/CK pin will revert to hi-impedance.

In order to select 9-bit transmission, the TX9 (TXSTA<6>) bit should be set and the ninth bit value should be written to TX9D (TXSTA<0>). The ninth bit value must be written before writing the 8-bit data to the TXREG. This is because a data write to TXREG can result in an immediate transfer of the data to the TSR (if the TSR is empty).

Steps to follow when setting up an Asynchronous Transmission:

1. Initialize the SPBRG register for the appropriate baud rate.
2. Enable the asynchronous serial port by clearing the SYNC bit and setting the SPEN bit.
3. If interrupts are desired, then set the TXIE bit.
4. If 9-bit transmission is desired, then set the TX9 bit.
5. If 9-bit transmission is selected, the ninth bit should be loaded in TX9D.
6. Load data to the TXREG register.
7. Enable the transmission by setting TXEN (starts transmission).

FIGURE 14-3: ASYNCHRONOUS MASTER TRANSMISSION



REGISTER 15-3: SSPCON2: SYNC SERIAL PORT CONTROL REGISTER2 (ADDRESS 12h, BANK 6)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
GCEN	ACKSTAT	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN
bit 7							bit 0

bit 7 **GCEN**: General Call Enable bit (in I^2C Slave mode only)
1 = Enable interrupt when a general call address (0000h) is received in the SSPSR
0 = General call address disabled

bit 6 **ACKSTAT**: Acknowledge Status bit (in I^2C Master mode only)

In Master Transmit mode:
1 = Acknowledge was not received from slave
0 = Acknowledge was received from slave

bit 5 **ACKDT**: Acknowledge Data bit (in I^2C Master mode only)

In Master Receive mode:
Value that will be transmitted when the user initiates an Acknowledge sequence at the end of a receive.
1 = Not Acknowledge
0 = Acknowledge

bit 4 **ACKEN**: Acknowledge Sequence Enable bit (in I^2C Master mode only)

In Master Receive mode:
1 = Initiate Acknowledge sequence on SDA and SCL pins and transmit AKDT data bit. Automatically cleared by hardware.
0 = Acknowledge sequence idle

Note: If the I^2C module is not in the IDLE mode, this bit may not be set (no spooling) and the SSPBUF may not be written (or writes to the SSPBUF are disabled).

bit 3 **RCEN**: Receive Enable bit (in I^2C Master mode only)
1 = Enables Receive mode for I^2C
0 = Receive idle

Note: If the I^2C module is not in the IDLE mode, this bit may not be set (no spooling) and the SSPBUF may not be written (or writes to the SSPBUF are disabled).

bit 2 **PEN**: STOP Condition Enable bit (in I^2C Master mode only)

SCK Release Control:
1 = Initiate STOP condition on SDA and SCL pins. Automatically cleared by hardware.
0 = STOP condition idle

Note: If the I^2C module is not in the IDLE mode, this bit may not be set (no spooling) and the SSPBUF may not be written (or writes to the SSPBUF are disabled).

bit 1 **RSEN**: Repeated Start Condition Enabled bit (in I^2C Master mode only)
1 = Initiate Repeated Start condition on SDA and SCL pins. Automatically cleared by hardware.
0 = Repeated Start condition idle

Note: If the I^2C module is not in the IDLE mode, this bit may not be set (no spooling) and the SSPBUF may not be written (or writes to the SSPBUF are disabled).

bit 0 **SEN**: START Condition Enabled bit (In I^2C Master mode only)
1 = Initiate START condition on SDA and SCL pins. Automatically cleared by hardware.
0 = START condition idle.

Note: If the I^2C module is not in the IDLE mode, this bit may not be set (no spooling) and the SSPBUF may not be written (or writes to the SSPBUF are disabled).

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

17.0 SPECIAL FEATURES OF THE CPU

What sets a microcontroller apart from other processors are special circuits to deal with the needs of real-time applications. The PIC17CXXX family has a host of such 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 (Section 4.0)
- RESET (Section 5.0)
  - Power-on Reset (POR)
  - Power-up Timer (PWRT)
  - Oscillator Start-up Timer (OST)
  - Brown-out Reset (BOR)
- Interrupts (Section 6.0)
- Watchdog Timer (WDT)
- SLEEP mode
- Code protection

The PIC17CXXX has a Watchdog Timer which can be shut-off only through EPROM bits. It runs off its own RC oscillator for added reliability. There are two timers that offer necessary delays on POR and BOR. 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 96 ms (nominal) on power-up only, designed to keep the part in RESET while the power supply stabilizes. With these two timers on-chip, most applications need no external RESET circuitry.

The SLEEP mode is designed to offer a very low current power-down mode. The user can wake from SLEEP through external RESET, Watchdog Timer Reset, 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 LF crystal option saves power. Configuration bits are used to select various options. This configuration word has the format shown in Figure 17-1.

REGISTER 17-1: CONFIGURATION WORDS

High (H) Table Read Addr.	U-x	R/P-1	R/P-1	U-x	U-x	U-x	U-x	U-x	U-x
FE0Fh - FE08h	—	PM2	BODEN	—	—	—	—	—	—
	bit 15 bit 8	bit 7							bit 0

Low (L) Table Read Addr.	U-x	U-x	R/P-1	U-x	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
FE07h - FE00h	—	—	PM1	—	PM0	WDTPS1	WDTPS0	FOSC1	FOSC0
	bit 15 bit 8	bit 7							bit 0

bits 7H, 6L, 4L	**PM2, PM1, PM0:** Processor Mode Select bits 111 = Microprocessor mode 110 = Microcontroller mode 101 = Extended Microcontroller mode 000 = Code Protected Microcontroller mode
bit 6H	**BODEN:** Brown-out Detect Enable 1 = Brown-out Detect circuitry is enabled 0 = Brown-out Detect circuitry is disabled
bits 3L:2L	**WDTPS1:WDTPS0**: WDT Postscaler Select bits 11 = WDT enabled, postscaler = 1 10 = WDT enabled, postscaler = 256 01 = WDT enabled, postscaler = 64 00 = WDT disabled, 16-bit overflow timer
bits 1L:0L	**FOSC1:FOSC0**: Oscillator Select bits 11 = EC oscillator 10 = XT oscillator 01 = RC oscillator 00 = LF oscillator
Shaded bits (—)	**Reserved**

RETFIE **Return from Interrupt**

Syntax: [*label*] RETFIE

Operands: None

Operation: TOS → (PC);
0 → GLINTD;
PCLATH is unchanged.

Status Affected: GLINTD

Encoding:

0000	0000	0000	0101

Description: Return from Interrupt. Stack is POP'ed and Top-of-Stack (TOS) is loaded in the PC. Interrupts are enabled by clearing the GLINTD bit. GLINTD is the global interrupt disable bit (CPUSTA<4>).

Words: 1

Cycles: 2

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	No operation	Clear GLINTD	POP PC from stack
No operation	No operation	No operation	No operation

Example: `RETFIE`

After Interrupt
PC = TOS
GLINTD = 0

RETLW **Return Literal to WREG**

Syntax: [*label*] RETLW k

Operands: $0 \le k \le 255$

Operation: k → (WREG); TOS → (PC);
PCLATH is unchanged

Status Affected: None

Encoding:

1011	0110	kkkk	kkkk

Description: WREG is loaded with the eight-bit literal 'k'. The program counter is loaded from the top of the stack (the return address). The high address latch (PCLATH) remains unchanged.

Words: 1

Cycles: 2

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read literal 'k'	Process Data	POP PC from stack, Write to WREG
No operation	No operation	No operation	No operation

Example:

```
    CALL TABLE ; WREG contains table
               ;  offset value
               ;  WREG now has
               ;  table value
    :
TABLE
    ADDWF PC   ; WREG = offset
    RETLW k0   ; Begin table
    RETLW k1   ;
    :
    :
    RETLW kn   ; End of table
```

Before Instruction
WREG = 0x07

After Instruction
WREG = value of k7

20.3 Timing Parameter Symbology

The timing parameter symbols have been created following one of the following formats:

1. TppS2ppS
2. TppS
3. TCC:ST (I²C specifications only)
4. Ts (I²C specifications only)

T			
F	Frequency	T	Time

Lowercase symbols (pp) and their meanings:

pp			
ad	Address/Data	ost	Oscillator Start-Up Timer
al	ALE	pwrt	Power-Up Timer
cc	Capture1 and Capture2	rb	PORTB
ck	CLKOUT or clock	rd	$\overline{RD}$
dt	Data in	rw	$\overline{RD}$ or $\overline{WR}$
in	INT pin	t0	T0CKI
io	I/O port	t123	TCLK12 and TCLK3
mc	$\overline{MCLR}$	wdt	Watchdog Timer
oe	$\overline{OE}$	wr	$\overline{WR}$
os	OSC1		

Uppercase symbols and their meanings:

S			
D	Driven	L	Low
E	Edge	P	Period
F	Fall	R	Rise
H	High	V	Valid
I	Invalid (Hi-impedance)	Z	Hi-impedance

FIGURE 20-11: CAPTURE TIMINGS

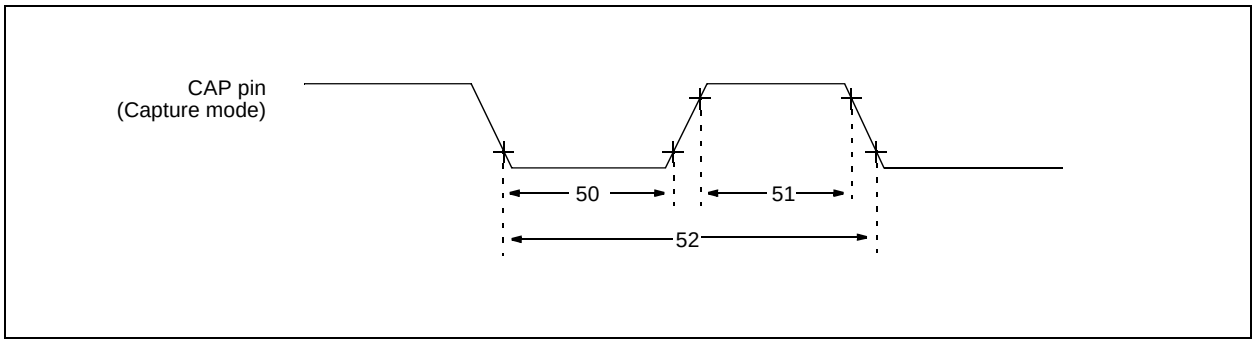


TABLE 20-6: CAPTURE REQUIREMENTS

Param No.	Sym	Characteristic	Min	Typ †	Max	Units	Conditions
50	TccL	Capture pin input low time	10	—	—	ns	
51	TccH	Capture pin input high time	10	—	—	ns	
52	TccP	Capture pin input period	$\frac{2T_{CY}}{N}$	—	—	ns	N = prescale value (4 or 16)

† Data in “Typ” column is at 5V, 25°C unless otherwise stated.

FIGURE 20-12: PWM TIMINGS

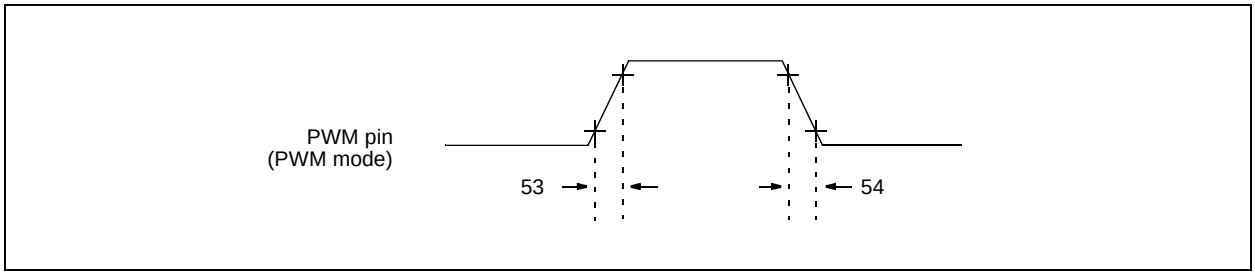


TABLE 20-7: PWM REQUIREMENTS

Param No.	Sym	Characteristic	Min	Typ †	Max	Units	Conditions
53	TccR	PWM pin output rise time	—	10	35	ns	
54	TccF	PWM pin output fall time	—	10	35	ns	

† Data in “Typ” column is at 5V, 25°C unless otherwise stated.

FIGURE 20-16: SPI SLAVE MODE TIMING (CKE = 1)

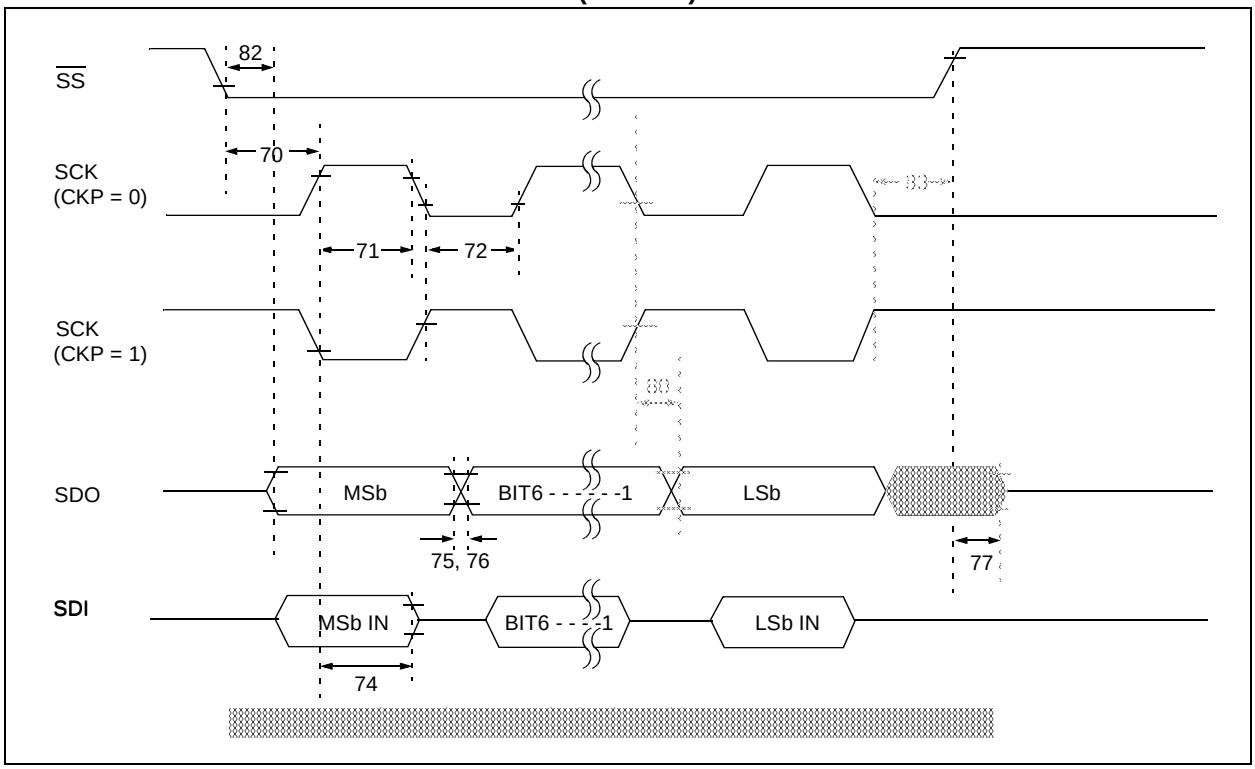


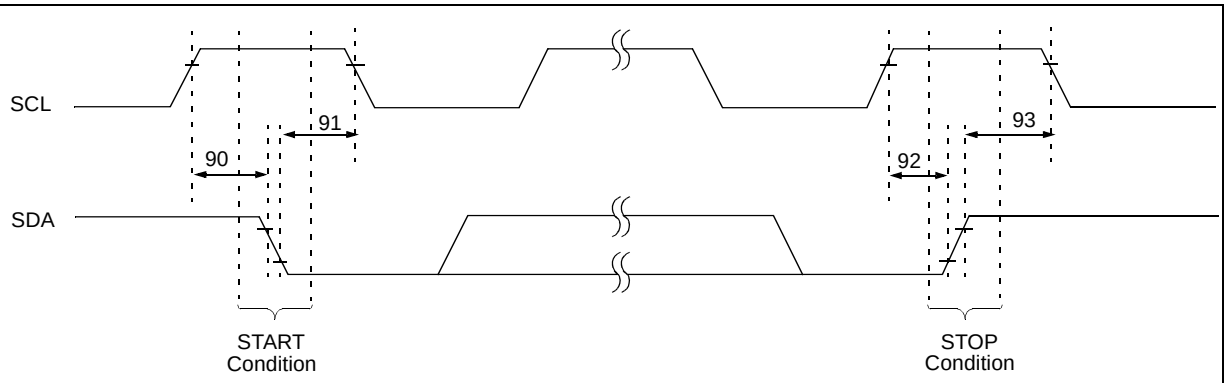
TABLE 20-11: SPI MODE REQUIREMENTS (SLAVE MODE, CKE = 1)

Param. No.	Symbol	Characteristic		Min	Typ†	Max	Units	Conditions
70	TssL2scH, TssL2scL	$\overline{SS}\downarrow$ to SCK↓ or SCK↑ input		Tcy	—	—	ns	
71	TscH	SCK input high time (Slave mode)	Continuous	1.25TCY + 30	—	—	ns	
71A			Single Byte	40	—	—	ns	**(Note 1)**
72	TscL	SCK input low time (Slave mode)	Continuous	1.25TCY + 30	—	—	ns	
72A			Single Byte	40	—	—	ns	**(Note 1)**
73A	TB2B	Last clock edge of Byte1 to the 1st clock edge of Byte2		1.5TCY + 40	—	—	ns	**(Note 1)**
74	TscH2diL, TscL2diL	Hold time of SDI data input to SCK edge		100	—	—	ns	
75	TdoR	SDO data output rise time		—	10	25	ns	
76	TdoF	SDO data output fall time		—	10	25	ns	
77	TssH2doZ	$\overline{SS}\uparrow$ to SDO output hi-impedance		10	—	50	ns	
80	TscH2doV, TscL2doV	SDO data output valid after SCK edge		—	—	50	ns	
82	TssL2doV	SDO data output valid after $\overline{SS}\downarrow$ edge		—	—	50	ns	
83	TscH2ssH, TscL2ssH	$\overline{SS}\uparrow$ after SCK edge		1.5TCY + 40	—	—	ns	

† Data in "Typ" column is at 5V, 25°C unless otherwise stated.

Note 1: Specification 73A is only required if specifications 71A and 72A are used.

FIGURE 20-17: I²C BUS START/STOP BITS TIMING



Note: Refer to Figure 20-5 for load conditions.

TABLE 20-12: I²C BUS START/STOP BITS REQUIREMENTS

Param. No.	Sym	Characteristic		Min	Typ	Max	Units	Conditions
90	Tsu:sta	START condition Setup time	100 kHz mode	2(Tosc)(BRG + 1)	—	—	ns	Only relevant for Repeated Start condition
			400 kHz mode	2(Tosc)(BRG + 1)	—	—		
			1 MHz mode(1)	2(Tosc)(BRG + 1)	—	—		
91	Thd:sta	START condition Hold time	100 kHz mode	2(Tosc)(BRG + 1)	—	—	ns	After this period, the first clock pulse is generated
			400 kHz mode	2(Tosc)(BRG + 1)	—	—		
			1 MHz mode(1)	2(Tosc)(BRG + 1)	—	—		
92	Tsu:sto	STOP condition Setup time	100 kHz mode	2(Tosc)(BRG + 1)	—	—	ns	
			400 kHz mode	2(Tosc)(BRG + 1)	—	—		
			1 MHz mode(1)	2(Tosc)(BRG + 1)	—	—		
93	Thd:sto	STOP condition Hold time	100 kHz mode	2(Tosc)(BRG + 1)	—	—	ns	
			400 kHz mode	2(Tosc)(BRG + 1)	—	—		
			1 MHz mode(1)	2(Tosc)(BRG + 1)	—	—		

Note 1: Maximum pin capacitance = 10 pF for all I²C pins.

FIGURE 20-25: MEMORY INTERFACE READ TIMING

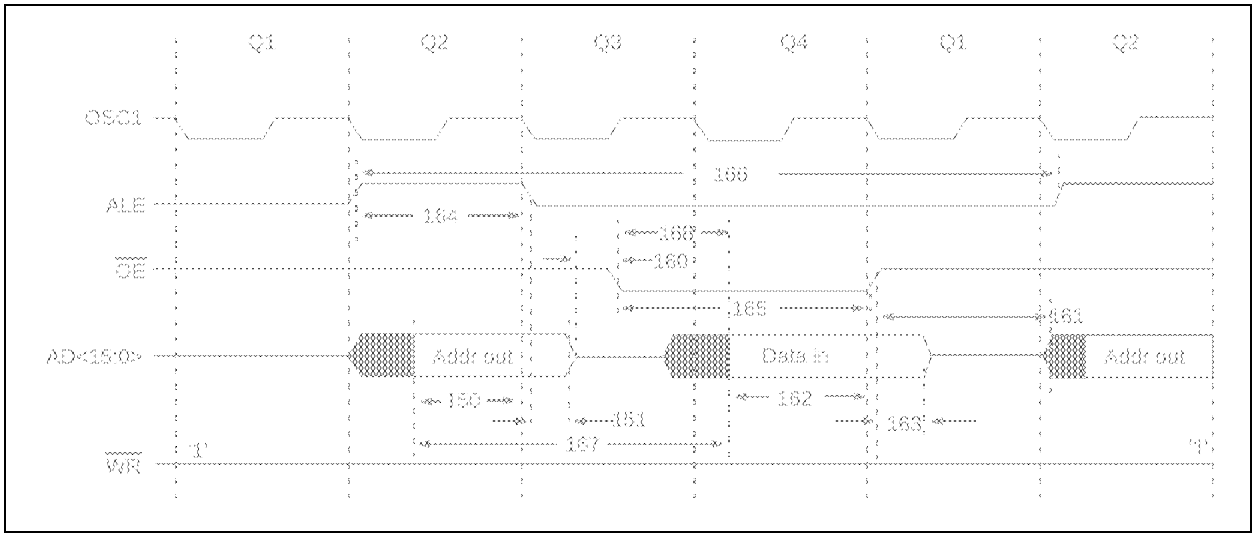


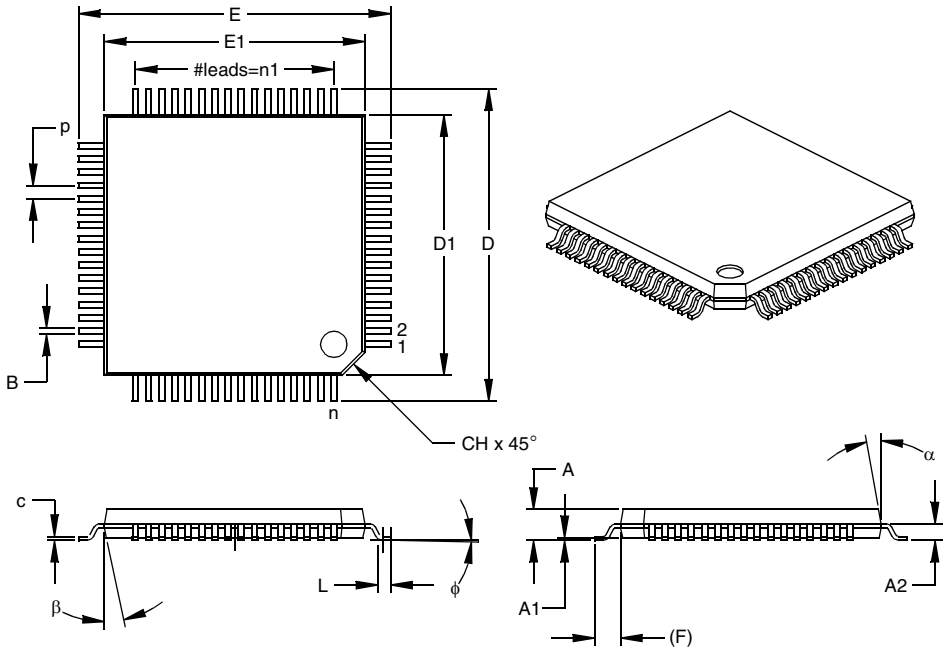
TABLE 20-21: MEMORY INTERFACE READ REQUIREMENTS

Param. No.	Sym	Characteristic		Min	Typ†	Max	Units	Conditions
150	TadV2alL	AD15:AD0 (address) valid to ALE↓ (address setup time)	PIC17**C**XXX	0.25TCY - 10	—	—	ns	
			PIC17**LC**XXX	0.25TCY - 10	—	—		
151	TalL2adI	ALE↓ to address out invalid (address hold time)	PIC17**C**XXX	5	—	—	ns	
			PIC17**LC**XXX	5	—	—		
160	TadZ2oeL	AD15:AD0 hi-impedance to $\overline{OE}$↓	PIC17**C**XXX	0	—	—	ns	
			PIC17**LC**XXX	0	—	—		
161	ToeH2adD	$\overline{OE}$↑ to AD15:AD0 driven	PIC17**C**XXX	0.25TCY - 15	—	—	ns	
			PIC17**LC**XXX	0.25TCY - 15	—	—		
162	TadV2oeH	Data in valid before $\overline{OE}$↑ (data setup time)	PIC17**C**XXX	35	—	—	ns	
			PIC17**LC**XXX	45	—	—		
163	ToeH2adI	$\overline{OE}$↑to data in invalid (data hold time)	PIC17**C**XXX	0	—	—	ns	
			PIC17**LC**XXX	0	—	—		
164	TalH	ALE pulse width	PIC17**C**XXX	—	0.25TCY	—	ns	
			PIC17**LC**XXX	—	0.25TCY	—		
165	ToeL	$\overline{OE}$ pulse width	PIC17**C**XXX	0.5TCY - 35	—	—	ns	
			PIC17**LC**XXX	0.5TCY - 35	—	—		
166	TalH2alH	ALE↑ to ALE↑(cycle time)	PIC17**C**XXX	—	TCY	—	ns	
			PIC17**LC**XXX	—	TCY	—		
167	Tacc	Address access time	PIC17**C**XXX	—	—	0.75TCY - 30	ns	
			PIC17**LC**XXX	—	—	0.75TCY - 45		
168	Toe	Output enable access time ($\overline{OE}$ low to data valid)	PIC17**C**XXX	—	—	0.5TCY - 45	ns	
			PIC17**LC**XXX	—	—	0.5TCY - 75		

† Data in "Typ" column is at 5V, 25°C unless otherwise stated.

64-Lead Plastic Thin Quad Flatpack (PT) 10x10x1 mm Body, 1.0/0.10 mm Lead Form (TQFP)

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



Units		INCHES			MILLIMETERS*		
Dimension Limits		MIN	NOM	MAX	MIN	NOM	MAX
Number of Pins	n		64			64	
Pitch	p		.020			0.50	
Pins per Side	n1		16			16	
Overall Height	A	.039	.043	.047	1.00	1.10	1.20
Molded Package Thickness	A2	.037	.039	.041	0.95	1.00	1.05
Standoff §	A1	.002	.006	.010	0.05	0.15	0.25
Foot Length	L	.018	.024	.030	0.45	0.60	0.75
Footprint (Reference)	(F)		.039			1.00	
Foot Angle	ϕ	0	3.5	7	0	3.5	7
Overall Width	E	.463	.472	.482	11.75	12.00	12.25
Overall Length	D	.463	.472	.482	11.75	12.00	12.25
Molded Package Width	E1	.390	.394	.398	9.90	10.00	10.10
Molded Package Length	D1	.390	.394	.398	9.90	10.00	10.10
Lead Thickness	c	.005	.007	.009	0.13	0.18	0.23
Lead Width	B	.007	.009	.011	0.17	0.22	0.27
Pin 1 Corner Chamfer	CH	.025	.035	.045	0.64	0.89	1.14
Mold Draft Angle Top	α	5	10	15	5	10	15
Mold Draft Angle Bottom	β	5	10	15	5	10	15

* Controlling Parameter
§ Significant Characteristic

Notes:
Dimensions D1 and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010” (0.254mm) per side.
JEDEC Equivalent: MS-026
Drawing No. C04-085