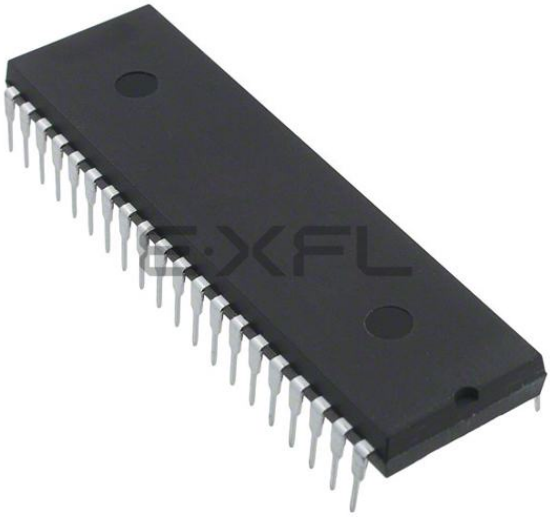


Welcome to E-XFL.COM

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	20MHz
Connectivity	I²C, 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	OTP
EEPROM Size	-
RAM Size	192 x 8
Voltage - Supply (Vcc/Vdd)	4V ~ 6V
Data Converters	A/D 8x8b
Oscillator Type	External
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Through Hole
Package / Case	40-DIP (0.600", 15.24mm)
Supplier Device Package	40-PDIP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16c74a-20-p

TABLE 1-1: PIC16C7XX FAMILY OF DEVCES

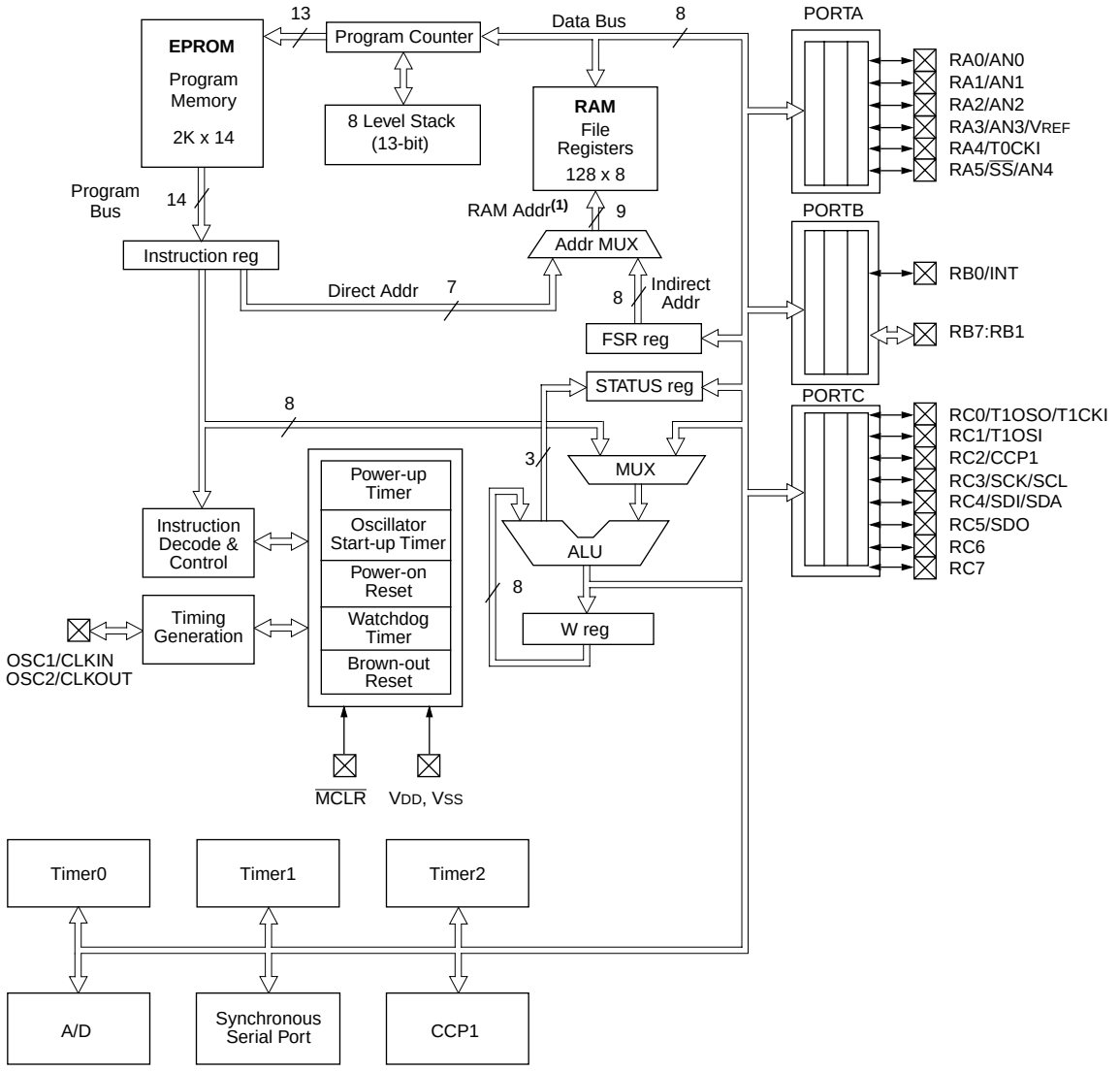
		PIC16C710	PIC16C71	PIC16C711	PIC16C715	PIC16C72	PIC16CR72(1)
Clock	Maximum Frequency of Operation (MHz)	20	20	20	20	20	20
Memory	EPROM Program Memory (x14 words)	512	1K	1K	2K	2K	—
	ROM Program Memory (14K words)	—	—	—	—	—	2K
	Data Memory (bytes)	36	36	68	128	128	128
Peripherals	Timer Module(s)	TMR0	TMR0	TMR0	TMR0	TMR0, TMR1, TMR2	TMR0, TMR1, TMR2
	Capture/Compare/ PWM Module(s)	—	—	—	—	1	1
	Serial Port(s) (SPI/I^2C, USART)	—	—	—	—	SPI/I^2C	SPI/I^2C
	Parallel Slave Port	—	—	—	—	—	—
	A/D Converter (8-bit) Channels	4	4	4	4	5	5
Features	Interrupt Sources	4	4	4	4	8	8
	I/O Pins	13	13	13	13	22	22
	Voltage Range (Volts)	3.0-6.0	3.0-6.0	3.0-6.0	3.0-5.5	2.5-6.0	3.0-5.5
	In-Circuit Serial Programming	Yes	Yes	Yes	Yes	Yes	Yes
	Brown-out Reset	Yes	—	Yes	Yes	Yes	Yes
	Packages	18-pin DIP, SOIC; 20-pin SSOP	18-pin DIP, SOIC	18-pin DIP, SOIC; 20-pin SSOP	18-pin DIP, SOIC; 20-pin SSOP	28-pin SDIP, SOIC, SSOP	28-pin SDIP, SOIC, SSOP

		PIC16C73A	PIC16C74A	PIC16C76	PIC16C77
Clock	Maximum Frequency of Operation (MHz)	20	20	20	20
Memory	EPROM Program Memory (x14 words)	4K	4K	8K	8K
	Data Memory (bytes)	192	192	368	368
Peripherals	Timer Module(s)	TMR0, TMR1, TMR2	TMR0, TMR1, TMR2	TMR0, TMR1, TMR2	TMR0, TMR1, TMR2
	Capture/Compare/PWM Module(s)	2	2	2	2
	Serial Port(s) (SPI/I^2C, USART)	SPI/I^2C, USART	SPI/I^2C, USART	SPI/I^2C, USART	SPI/I^2C, USART
	Parallel Slave Port	—	Yes	—	Yes
	A/D Converter (8-bit) Channels	5	8	5	8
Features	Interrupt Sources	11	12	11	12
	I/O Pins	22	33	22	33
	Voltage Range (Volts)	2.5-6.0	2.5-6.0	2.5-6.0	2.5-6.0
	In-Circuit Serial Programming	Yes	Yes	Yes	Yes
	Brown-out Reset	Yes	Yes	Yes	Yes
	Packages	28-pin SDIP, SOIC	40-pin DIP; 44-pin PLCC, MQFP, TQFP	28-pin SDIP, SOIC	40-pin DIP; 44-pin PLCC, MQFP, TQFP

All PIC16/17 Family devices have Power-on Reset, selectable Watchdog Timer, selectable code protect and high I/O current capability. All PIC16C7XX Family devices use serial programming with clock pin RB6 and data pin RB7.

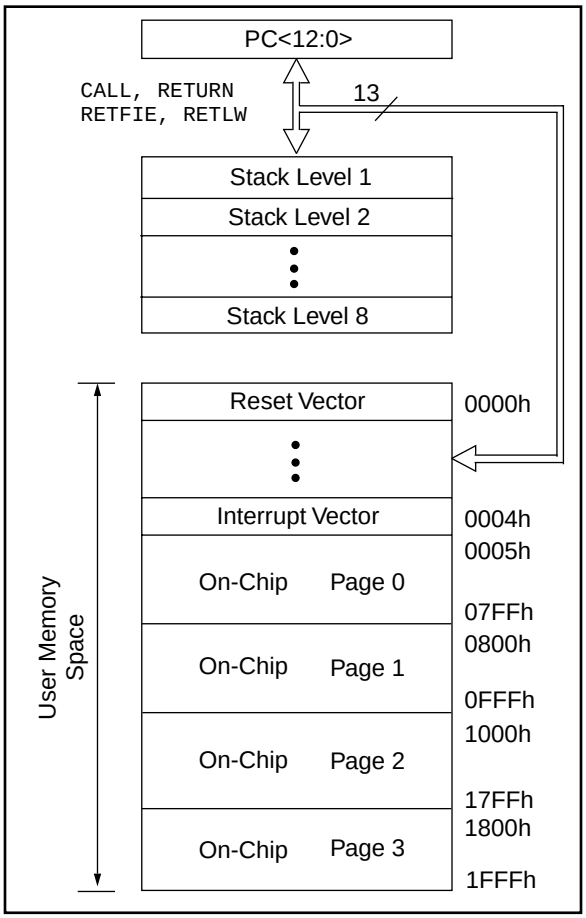
Note 1: Please contact your local Microchip sales office for availability of these devices.

FIGURE 3-1: PIC16C72 BLOCK DIAGRAM



Note 1: Higher order bits are from the STATUS register.

FIGURE 4-3: PIC16C76/77 PROGRAM MEMORY MAP AND STACK



4.2 Data Memory Organization

Applicable Devices						
72	73	73A	74	74A	76	77

The data memory is partitioned into multiple banks which contain the General Purpose Registers and the Special Function Registers. Bits RP1 and RP0 are the bank select bits.

RP1:RP0 (STATUS<6:5>)
= 00 → Bank0
= 01 → Bank1
= 10 → Bank2
= 11 → Bank3

Each bank extends up to 7Fh (128 bytes). The lower locations of each bank are reserved for the Special Function Registers. Above the Special Function Registers are General Purpose Registers, implemented as static RAM. All implemented banks contain special function registers. Some “high use” special function registers from one bank may be mirrored in another bank for code reduction and quicker access.

4.2.1 GENERAL PURPOSE REGISTER FILE

The register file can be accessed either directly, or indirectly through the File Select Register FSR (Section 4.5).

TABLE 4-3: PIC16C76/77 SPECIAL FUNCTION REGISTER SUMMARY

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 (2)
Bank 0											
00h(4)	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	0000 0000
01h	TMR0	Timer0 module's register								xxxx xxxx	uuuu uuuu
02h(4)	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	0000 0000
03h(4)	STATUS	IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	000q quuu
04h(4)	FSR	Indirect data memory address pointer								xxxx xxxx	uuuu uuuu
05h	PORTA	—	—	PORTA Data Latch when written: PORTA pins when read						--0x 0000	--0u 0000
06h	PORTB	PORTB Data Latch when written: PORTB pins when read								xxxx xxxx	uuuu uuuu
07h	PORTC	PORTC Data Latch when written: PORTC pins when read								xxxx xxxx	uuuu uuuu
08h(5)	PORTD	PORTD Data Latch when written: PORTD pins when read								xxxx xxxx	uuuu uuuu
09h(5)	PORTE	—	—	—	—	—	RE2	RE1	RE0	---- -xxx	---- -uuu
0Ah(1,4)	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter					---0 0000	---0 0000
0Bh(4)	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	PSPIF(3)	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
0Dh	PIR2	—	—	—	–	—	—	—	CCP2IF	---- ---0	---- ---0
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	$\overline{T1SYNC}$	TMR1CS	TMR1ON	--00 0000	--uu uuuu
11h	TMR2	Timer2 module's register								0000 0000	0000 0000
12h	T2CON	—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	-000 0000	-000 0000
13h	SSPBUF	Synchronous Serial Port Receive Buffer/Transmit Register								xxxx xxxx	uuuu uuuu
14h	SSPCON	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	0000 0000
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
18h	RCSTA	SPEN	RX9	SREN	CREN	—	FERR	OERR	RX9D	0000 -00x	0000 -00x
19h	TXREG	USART Transmit Data Register								0000 0000	0000 0000
1Ah	RCREG	USART Receive Data Register								0000 0000	0000 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
1Eh	ADRES	A/D Result Register								xxxx xxxx	uuuu uuuu
1Fh	ADCON0	ADCS1	ADCS0	CHS2	CHS1	CHS0	GO/$\overline{DONE}$	—	ADON	0000 00-0	0000 00-0

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

Note 1: The upper byte of the program counter is not directly accessible. PCLATH is a holding register for the PC<12:8> whose contents are transferred to the upper byte of the program counter.

2: Other (non power-up) resets include external reset through $\overline{MCLR}$ and Watchdog Timer Reset.

3: Bits PSPIE and PSPIF are reserved on the PIC16C76, always maintain these bits clear.

4: These registers can be addressed from any bank.

5: PORTD and PORTE are not physically implemented on the PIC16C76, read as '0'.

NOTES:

5.2 PORTB and TRISB Registers

Applicable Devices						
72	73	73A	74	74A	76	77

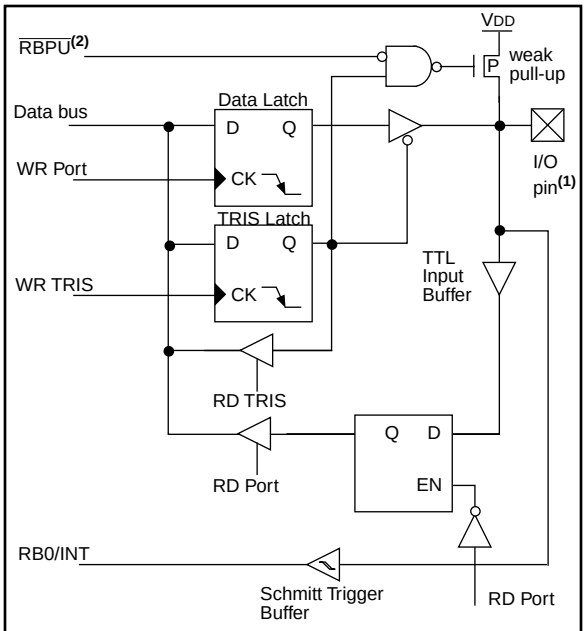
PORTB is an 8-bit wide bi-directional port. The corresponding data direction register is TRISB. Setting a bit in the TRISB register puts the corresponding output driver in a hi-impedance input mode. Clearing a bit in the TRISB register puts the contents of the output latch on the selected pin(s).

EXAMPLE 5-2: INITIALIZING PORTB

```
BCF    STATUS, RP0  ;
CLRF   PORTB        ; Initialize PORTB by
                    ; clearing output
                    ; data latches
BSF    STATUS, RP0  ; Select Bank 1
MOVLW  0xCF         ; Value used to
                    ; initialize data
                    ; direction
MOVWF  TRISB        ; Set RB<3:0> as inputs
                    ; RB<5:4> as outputs
                    ; RB<7:6> as inputs
```

Each of the PORTB pins has a weak internal pull-up. A single control bit can turn on all the pull-ups. This is performed by clearing bit $\overline{\text{RBPU}}$ (OPTION<7>). The weak pull-up is automatically turned off when the port pin is configured as an output. The pull-ups are disabled on a Power-on Reset.

FIGURE 5-3: BLOCK DIAGRAM OF RB3:RB0 PINS



Note 1: I/O pins have diode protection to VDD and VSS.
2: To enable weak pull-ups, set the appropriate TRIS bit(s) and clear the $\overline{\text{RBPU}}$ bit (OPTION<7>).

Four of PORTB's pins, RB7:RB4, have an interrupt on change feature. Only pins configured as inputs can cause this interrupt to occur (i.e. any RB7:RB4 pin configured as an output is excluded from the interrupt on change comparison). The input pins (of RB7:RB4) are compared with the old value latched on the last read of PORTB. The "mismatch" outputs of RB7:RB4 are OR'ed together to generate the RB Port Change Interrupt with flag bit RBIF (INTCON<0>).

This interrupt can wake the device from SLEEP. The user, in the interrupt service routine, can clear the interrupt in the following manner:

a) Any read or write of PORTB. This will end the mismatch condition.
b) Clear flag bit RBIF.

A mismatch condition will continue to set flag bit RBIF. Reading PORTB will end the mismatch condition, and allow flag bit RBIF to be cleared.

This interrupt on mismatch feature, together with software configurable pull-ups on these four pins allow easy interface to a keypad and make it possible for wake-up on key-depression. Refer to the Embedded Control Handbook, *"Implementing Wake-Up on Key Stroke"* (AN552).

Note:	For the PIC16C73/74, if a change on the I/O pin should occur when the read operation is being executed (start of the Q2 cycle), then interrupt flag bit RBIF may not get set.

The interrupt on change feature is recommended for wake-up on key depression operation and operations where PORTB is only used for the interrupt on change feature. Polling of PORTB is not recommended while using the interrupt on change feature.

7.0 TIMER0 MODULE

Applicable Devices						
72	73	73A	74	74A	76	77

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

Figure 7-1 is a simplified block diagram of the Timer0 module.

Timer mode is selected by clearing bit T0CS (OPTION<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 (Figure 7-2 and Figure 7-3). The user can work around this by writing an adjusted value to the TMR0 register.

Counter mode is selected by setting bit T0CS (OPTION<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<4>). Clearing bit T0SE selects the rising edge. Restrictions on the external clock input are discussed in detail in Section 7.2.

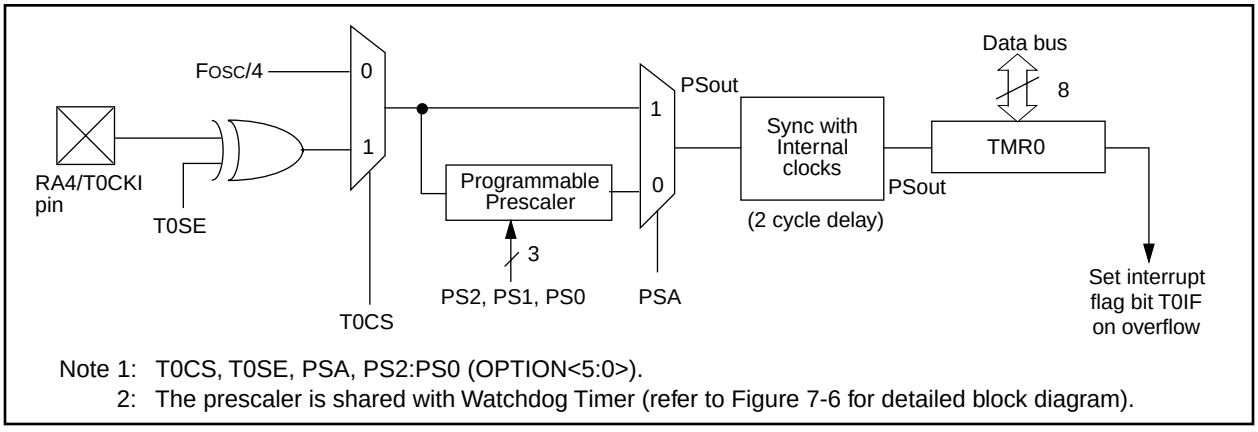
The prescaler is mutually exclusively shared between the Timer0 module and the Watchdog Timer. The prescaler assignment is controlled in software by control bit PSA (OPTION<3>). Clearing bit PSA will assign the prescaler to the Timer0 module. The prescaler is not readable or writable. When the prescaler is assigned to the Timer0 module, prescale values of 1:2, 1:4, ..., 1:256 are selectable. Section 7.3 details the operation of the prescaler.

7.1 Timer0 Interrupt

Applicable Devices						
72	73	73A	74	74A	76	77

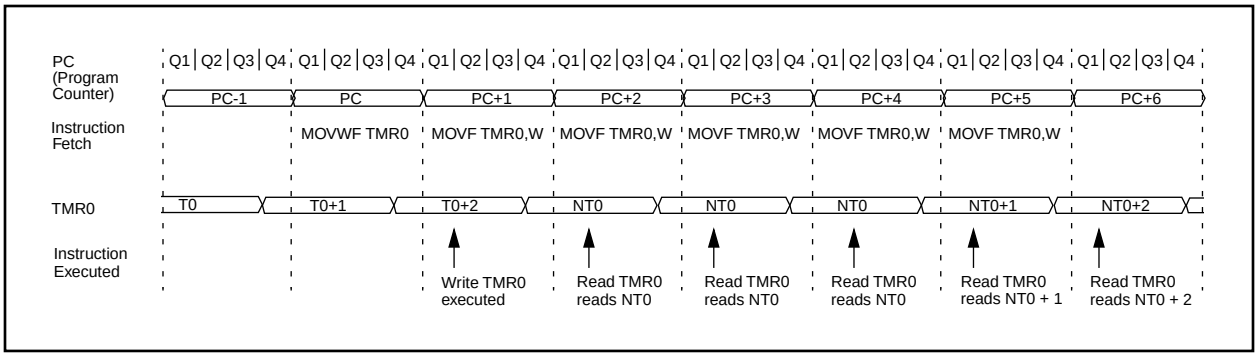
The TMR0 interrupt is generated when the TMR0 register overflows from FFh to 00h. This overflow sets bit T0IF (INTCON<2>). The interrupt can be masked by clearing bit T0IE (INTCON<5>). Bit T0IF must be cleared in software by the Timer0 module interrupt service routine before re-enabling this interrupt. The TMR0 interrupt cannot awaken the processor from SLEEP since the timer is shut off during SLEEP. See Figure 7-4 for Timer0 interrupt timing.

FIGURE 7-1: TIMER0 BLOCK DIAGRAM



Note 1: T0CS, T0SE, PSA, PS2:PS0 (OPTION<5:0>).
2: The prescaler is shared with Watchdog Timer (refer to Figure 7-6 for detailed block diagram).

FIGURE 7-2: TIMER0 TIMING: INTERNAL CLOCK/NO PRESCALE



11.2 SPI Mode for PIC16C72/73/73A/74/74A

This section contains register definitions and operational characteristics of the SPI module for the PIC16C72, PIC16C73, PIC16C73A, PIC16C74, PIC16C74A.

FIGURE 11-1: SSPSTAT: SYNC SERIAL PORT STATUS REGISTER (ADDRESS 94h)

U-0	U-0	R-0	R-0	R-0	R-0	R-0	R-0
—	—	D/$\overline{A}$	P	S	R/$\overline{W}$	UA	BF
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as '0'
- n = Value at POR reset

bit 7-6: **Unimplemented:** Read as '0'

bit 5: **D/$\overline{A}$**: Data/Address bit (I^2C mode only)
1 = Indicates that the last byte received or transmitted was data
0 = Indicates that the last byte received or transmitted was address

bit 4: **P**: Stop bit (I^2C mode only. This bit is cleared when the SSP module is disabled, SSPEN is cleared)
1 = Indicates that a stop bit has been detected last (this bit is '0' on RESET)
0 = Stop bit was not detected last

bit 3: **S**: Start bit (I^2C mode only. This bit is cleared when the SSP module is disabled, SSPEN is cleared)
1 = Indicates that a start bit has been detected last (this bit is '0' on RESET)
0 = Start bit was not detected last

bit 2: **R/$\overline{W}$**: Read/Write bit information (I^2C mode only)
This bit holds the R/W bit information following the last address match. This bit is valid from the address match to the next start bit, stop bit, or $\overline{ACK}$ bit.
1 = Read
0 = Write

bit 1: **UA**: Update Address (10-bit I^2C mode only)
1 = Indicates that the user needs to update the address in the SSPADD register
0 = Address does not need to be updated

bit 0: **BF**: Buffer Full Status bit

Receive (SPI and I^2C modes)
1 = Receive complete, SSPBUF is full
0 = Receive not complete, SSPBUF is empty

Transmit (I^2C mode only)
1 = Transmit in progress, SSPBUF is full
0 = Transmit complete, SSPBUF is empty

FIGURE 11-8: SSPCON: SYNC SERIAL PORT CONTROL REGISTER (ADDRESS 14h)(PIC16C76/77)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as '0'
- n = Value at POR reset

bit 7: **WCOL**: Write Collision Detect bit
1 = The SSPBUF register is written while it is still transmitting the previous word (must be cleared in software)
0 = No collision

bit 6: **SSPOV**: Receive Overflow Indicator bit

In SPI mode
1 = A new byte is received while the SSPBUF register is still holding the previous data. In case of overflow, the data in SSPSR is lost. Overflow can only occur in slave mode. The user must read the SSPBUF, even if only transmitting data, to avoid setting overflow. In master mode the overflow bit is not set since each new reception (and transmission) is initiated by writing to the SSPBUF register.
0 = No overflow

In I^2C mode
1 = A byte is received while the SSPBUF register is still holding the previous byte. SSPOV is a "don't care" in transmit mode. SSPOV must be cleared in software in either mode.
0 = No overflow

bit 5: **SSPEN**: Synchronous Serial Port Enable bit

In SPI mode
1 = Enables serial port and configures SCK, SDO, and SDI as serial port pins
0 = Disables serial port and configures these pins as I/O port pins

In I^2C mode
1 = Enables the serial port and configures the SDA and SCL pins as serial port pins
0 = Disables serial port and configures these pins as I/O port pins
In both modes, when enabled, these pins must be properly configured as input or output.

bit 4: **CKP**: Clock Polarity Select bit
In SPI mode
1 = Idle state for clock is a high level
0 = Idle state for clock is a low level
In I^2C mode
SCK release control
1 = Enable clock
0 = Holds clock low (clock stretch) (Used to ensure data setup time)

bit 3-0: **SSPM3:SSPM0**: Synchronous Serial Port Mode Select bits
0000 = SPI master mode, clock = FOSC/4
0001 = SPI master mode, clock = FOSC/16
0010 = SPI master mode, clock = FOSC/64
0011 = SPI master mode, clock = TMR2 output/2
0100 = SPI slave mode, clock = SCK pin. $\overline{SS}$ pin control enabled.
0101 = SPI slave mode, clock = SCK pin. $\overline{SS}$ pin control disabled. $\overline{SS}$ can be used as I/O pin
0110 = I^2C slave mode, 7-bit address
0111 = I^2C slave mode, 10-bit address
1011 = I^2C firmware controlled master mode (slave idle)
1110 = I^2C slave mode, 7-bit address with start and stop bit interrupts enabled
1111 = I^2C slave mode, 10-bit address with start and stop bit interrupts enabled

13.8 Use of the CCP Trigger

Applicable Devices						
72	73	73A	74	74A	76	77

Note: In the PIC16C72, the "special event trigger" is implemented in the CCP1 module.

An A/D conversion can be started by the “special event trigger” of the CCP2 module (CCP1 on the PIC16C72 only). This requires that the CCP2M3:CCP2M0 bits (CCP2CON<3:0>) be programmed as 1011 and that the A/D module is enabled (ADON bit is set). When the trigger occurs, the GO/$\overline{\text{DONE}}$ bit will be set, starting the A/D conversion, and the Timer1 counter will be reset to zero. Timer1 is reset to automatically repeat the A/D acquisition period with minimal software overhead (moving the ADRES to the desired location). The appropriate analog input channel must be selected and the minimum acquisition done before the “special event trigger” sets the GO/$\overline{\text{DONE}}$ bit (starts a conversion).

If the A/D module is not enabled (ADON is cleared), then the “special event trigger” will be ignored by the A/D module, but will still reset the Timer1 counter.

13.9 Connection Considerations

Applicable Devices						
72	73	73A	74	74A	76	77

If the input voltage exceeds the rail values (V_{SS} or V_{DD}) by greater than 0.2V, then the accuracy of the conversion is out of specification.

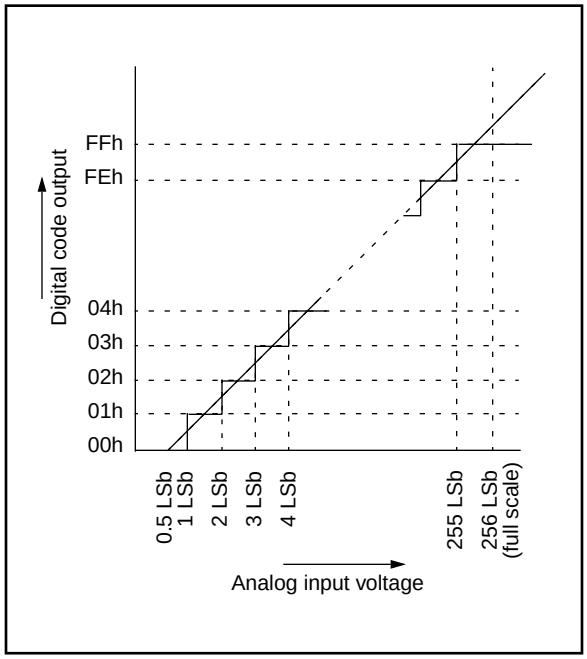
An external RC filter is sometimes added for anti-aliasing of the input signal. The R component should be selected to ensure that the total source impedance is kept under the 10 kΩ recommended specification. Any external components connected (via hi-impedance) to an analog input pin (capacitor, zener diode, etc.) should have very little leakage current at the pin.

13.10 Transfer Function

Applicable Devices						
72	73	73A	74	74A	76	77

The ideal transfer function of the A/D converter is as follows: the first transition occurs when the analog input voltage (V_{AIN}) is Analog V_{REF}/256 (Figure 13-5).

FIGURE 13-5: A/D TRANSFER FUNCTION



13.11 References

A very good reference for understanding A/D converters is the "Analog-Digital Conversion Handbook" third edition, published by Prentice Hall (ISBN 0-13-03-2848-0).

14.2.3 EXTERNAL CRYSTAL OSCILLATOR CIRCUIT

Either a prepackaged oscillator can be used or a simple oscillator circuit with TTL gates can be built. Prepackaged oscillators provide a wide operating range and better stability. A well-designed crystal oscillator will provide good performance with TTL gates. Two types of crystal oscillator circuits can be used; one with series resonance, or one with parallel resonance.

Figure 14-5 shows implementation of a parallel resonant oscillator circuit. The circuit is designed to use the fundamental frequency of the crystal. The 74AS04 inverter performs the 180-degree phase shift that a parallel oscillator requires. The 4.7 kΩ resistor provides the negative feedback for stability. The 10 kΩ potentiometer biases the 74AS04 in the linear region. This could be used for external oscillator designs.

FIGURE 14-5: EXTERNAL PARALLEL RESONANT CRYSTAL OSCILLATOR CIRCUIT

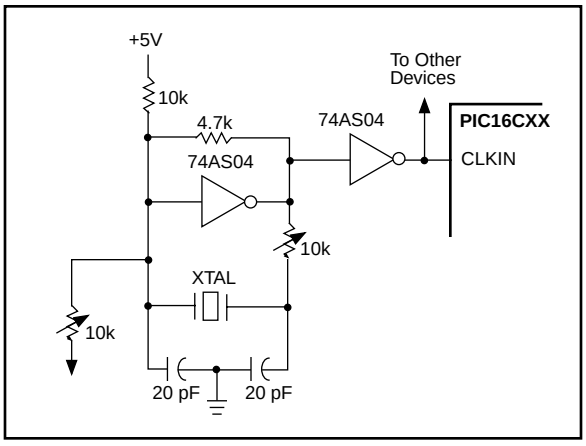
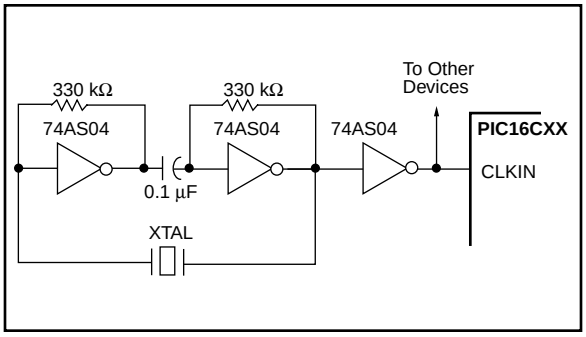


Figure 14-6 shows a series resonant oscillator circuit. This circuit is also designed to use the fundamental frequency of the crystal. The inverter performs a 180-degree phase shift in a series resonant oscillator circuit. The 330 kΩ resistors provide the negative feedback to bias the inverters in their linear region.

FIGURE 14-6: EXTERNAL SERIES RESONANT CRYSTAL OSCILLATOR CIRCUIT



14.2.4 RC OSCILLATOR

For timing insensitive applications the "RC" device option offers additional cost savings. The RC oscillator frequency is a function of the supply voltage, the resistor (Rext) and capacitor (Cext) values, and the operating temperature. In addition to this, the 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 the oscillation frequency, especially for low Cext values. The user also needs to take into account variation due to tolerance of external R and C components used. Figure 14-7 shows how the R/C combination is connected to the PIC16CXX. For Rext values below 2.2 kΩ, the oscillator operation may become unstable, or stop completely. For very high Rext values (e.g. 1 MΩ), the oscillator becomes sensitive to noise, humidity and leakage. Thus, we recommend to keep Rext between 3 kΩ and 100 kΩ.

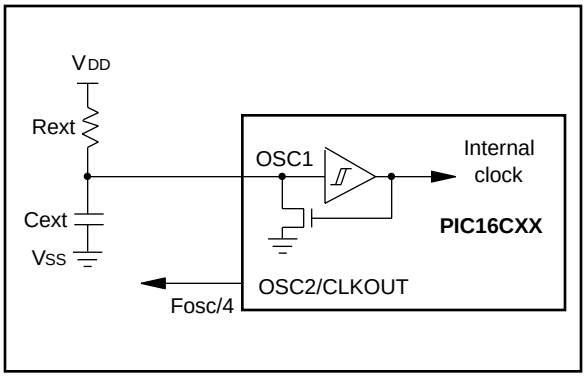
Although the oscillator will operate with no external capacitor (Cext = 0 pF), we recommend using values above 20 pF for noise and stability reasons. With no or small external capacitance, the oscillation frequency can vary dramatically due to changes in external capacitances, such as PCB trace capacitance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

See characterization data for desired device 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 characterization data for desired device for variation of oscillator frequency due to V_{DD} for given Rext/Cext 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 3-4 for waveform).

FIGURE 14-7: RC OSCILLATOR MODE



14.8 Power-down Mode (SLEEP)

Applicable Devices						
72	73	73A	74	74A	76	77

Power-down mode is entered by executing a SLEEP instruction.

If enabled, the Watchdog Timer will be cleared but keeps running, the $\overline{PD}$ bit (STATUS<3>) is cleared, the $\overline{TO}$ (STATUS<4>) bit is set, and the oscillator driver is turned off. The I/O ports maintain the status they had, before the SLEEP instruction was executed (driving high, low, or hi-impedance).

For lowest current consumption in this mode, place all I/O pins at either V_{DD}, or V_{SS}, ensure no external circuitry is drawing current from the I/O pin, power-down the A/D, disable external clocks. Pull all I/O pins, that are hi-impedance inputs, high or low externally to avoid switching currents caused by floating inputs. The T0CKI input should also be at V_{DD} or V_{SS} for lowest current consumption. The contribution from on-chip pull-ups on PORTB should be considered.

The $\overline{MCLR}$ pin must be at a logic high level (V_{IHMC}).

14.8.1 WAKE-UP FROM SLEEP

The device can wake up from SLEEP through one of the following events:

1. External reset input on $\overline{MCLR}$ pin.
2. Watchdog Timer Wake-up (if WDT was enabled).
3. Interrupt from INT pin, RB port change, or some Peripheral Interrupts.

External $\overline{MCLR}$ Reset will cause a device reset. All other events are considered a continuation of program execution and cause a "wake-up". The $\overline{TO}$ and $\overline{PD}$ bits in the STATUS register can be used to determine the cause of device reset. The $\overline{PD}$ bit, which is set on power-up, is cleared when SLEEP is invoked. The $\overline{TO}$ bit is cleared if a WDT time-out occurred (and caused wake-up).

The following peripheral interrupts can wake the device from SLEEP:

1. TMR1 interrupt. Timer1 must be operating as an asynchronous counter.
2. SSP (Start/Stop) bit detect interrupt.
3. SSP transmit or receive in slave mode (SPI/I^2C).
4. CCP capture mode interrupt.
5. Parallel Slave Port read or write.
6. A/D conversion (when A/D clock source is RC).
7. Special event trigger (Timer1 in asynchronous mode using an external clock).
8. USART TX or RX (synchronous slave mode).

Other peripherals cannot generate interrupts since during SLEEP, no on-chip Q clocks are present.

When the SLEEP instruction is being executed, the next instruction (PC + 1) is pre-fetched. For the device to wake-up through an interrupt event, the corresponding interrupt enable bit must be set (enabled). Wake-up is regardless of the state of the GIE bit. If the GIE bit is clear (disabled), the device continues execution at the instruction after the SLEEP instruction. If the GIE bit is set (enabled), the device executes the instruction after the SLEEP instruction and then branches to the interrupt address (0004h). In cases where the execution of the instruction following SLEEP is not desirable, the user should have a NOP after the SLEEP instruction.

14.8.2 WAKE-UP USING INTERRUPTS

When global interrupts are disabled (GIE cleared) and any interrupt source has both its interrupt enable bit and interrupt flag bit set, one of the following will occur:

- If the interrupt occurs **before** the execution of a SLEEP instruction, the SLEEP instruction will complete as a NOP. Therefore, the WDT and WDT postscaler will not be cleared, the $\overline{TO}$ bit will not be set and $\overline{PD}$ bits will not be cleared.
- If the interrupt occurs **during or after** the execution of a SLEEP instruction, the device will immediately wake up from sleep. The SLEEP instruction will be completely executed before the wake-up. Therefore, the WDT and WDT postscaler will be cleared, the $\overline{TO}$ bit will be set and the $\overline{PD}$ bit will be cleared.

Even if the flag bits were checked before executing a SLEEP instruction, it may be possible for flag bits to become set before the SLEEP instruction completes. To determine whether a SLEEP instruction executed, test the $\overline{PD}$ bit. If the $\overline{PD}$ bit is set, the SLEEP instruction was executed as a NOP.

To ensure that the WDT is cleared, a CLRWDT instruction should be executed before a SLEEP instruction.

COMF **Complement f**

Syntax: [*label*] COMF f,d

Operands: $0 \le f \le 127$
$d \in [0,1]$

Operation: $(\bar{f}) \rightarrow$ (destination)

Status Affected: Z

Encoding:

00	1001	dfff	ffff

Description: The contents of register 'f' are complemented. If 'd' is 0 the result is stored in W. If 'd' is 1 the result is stored back in register 'f'.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register 'f'	Process data	Write to destination

Example
```
COMF    REG1,0
```
Before Instruction
REG1 = 0x13
After Instruction
REG1 = 0x13
W = 0xEC

DECF **Decrement f**

Syntax: [*label*] DECF f,d

Operands: $0 \le f \le 127$
$d \in [0,1]$

Operation: (f) - 1 → (destination)

Status Affected: Z

Encoding:

00	0011	dfff	ffff

Description: Decrement register 'f'. If 'd' is 0 the result is stored in the W register. If 'd' is 1 the result is stored back in register 'f'.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register 'f'	Process data	Write to destination

Example
```
DECF    CNT, 1
```
Before Instruction
CNT = 0x01
Z = 0
After Instruction
CNT = 0x00
Z = 1

DECFSZ **Decrement f, Skip if 0**

Syntax: [*label*] DECFSZ f,d

Operands: $0 \le f \le 127$
$d \in [0,1]$

Operation: (f) - 1 → (destination); skip if result = 0

Status Affected: None

Encoding:

00	1011	dfff	ffff

Description: The contents of register 'f' are decremented. If 'd' is 0 the result is placed in the W register. If 'd' is 1 the result is placed back in register 'f'.
If the result is 1, the next instruction, is executed. If the result is 0, then a NOP is executed instead making it a 2TCY instruction.

Words: 1

Cycles: 1(2)

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register 'f'	Process data	Write to destination

If Skip: (2nd Cycle)

Q1	Q2	Q3	Q4
No-Operation	No-Operation	No-Operation	No-Operation

Example
```
HERE    DECFSZ  CNT, 1
        GOTO    LOOP
CONTINUE •
         •
         •
```
Before Instruction
PC = address HERE
After Instruction
CNT = CNT - 1
if CNT = 0,
PC = address CONTINUE
if CNT ≠ 0,
PC = address HERE+1

Applicable Devices	72	73	73A	74	74A	76	77

17.0 ELECTRICAL CHARACTERISTICS FOR PIC16C72

Absolute Maximum Ratings †

Ambient temperature under bias ..... -55 to +125˚C

Storage temperature ..... -65˚C to +150˚C

Voltage on any pin with respect to V_{SS} (except V_{DD}, $\overline{MCLR}$, and RA4) ..... -0.3V to (V_{DD} + 0.3V)

Voltage on V_{DD} with respect to V_{SS} ..... -0.3 to +7.5V

Voltage on $\overline{MCLR}$ with respect to V_{SS} (Note 2) ..... 0 to +14V

Voltage on RA4 with respect to Vss ..... 0 to +14V

Total power dissipation (Note 1) 1.0W

Maximum current out of V_{SS} pin ..... 300 mA

Maximum current into V_{DD} pin ..... 250 mA

Input clamp current, I_{IK} (V_I < 0 or V_I > V_{DD}) ..... ±20 mA

Output clamp current, I_{OK} (V_O < 0 or V_O > V_{DD}) ..... ±20 mA

Maximum output current sunk by any I/O pin ..... 25 mA

Maximum output current sourced by any I/O pin ..... 25 mA

Maximum current sunk by PORTA and PORTB (combined) ..... 200 mA

Maximum current sourced by PORTA and PORTB (combined) ..... 200 mA

Maximum current sunk by PORTC ..... 200 mA

Maximum current sourced by PORTC ..... 200 mA

Note 1: Power dissipation is calculated as follows: $Pdis = V_{DD} \times \{I_{DD} - \sum I_{OH}\} + \sum \{(V_{DD} - V_{OH}) \times I_{OH}\} + \sum(Vol \times I_{OL})$

Note 2: Voltage spikes below V_{SS} at the $\overline{MCLR}$ pin, inducing currents greater than 80 mA, may cause latch-up. Thus, a series resistor of 50-100Ω should be used when applying a "low" level to the $\overline{MCLR}$ pin rather than pulling this pin directly to V_{SS}.

† NOTICE: Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at those or any other conditions above those indicated in the operation listings of this specification is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.

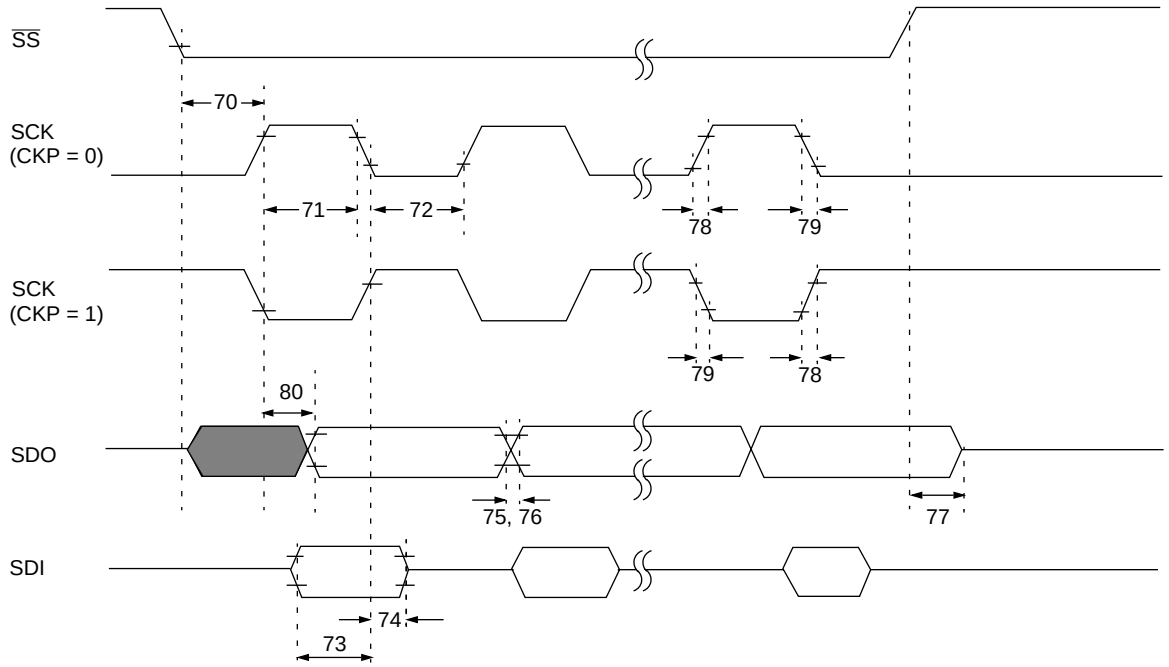
TABLE 17-1: CROSS REFERENCE OF DEVICE SPECS FOR OSCILLATOR CONFIGURATIONS AND FREQUENCIES OF OPERATION (COMMERCIAL DEVICES)

OSC	PIC16C72-04	PIC16C72-10	PIC16C72-20	PIC16LC72-04	JW Devices
RC	V_{DD}: 4.0V to 6.0V I_{DD}: 5 mA max. at 5.5V I_{PD}: 16 µA max. at 4V Freq: 4 MHz max.	V_{DD}: 4.5V to 5.5V I_{DD}: 2.7 mA typ. at 5.5V I_{PD}: 1.5 µA typ. at 4V Freq: 4 MHz max.	V_{DD}: 4.5V to 5.5V I_{DD}: 2.7 mA typ. at 5.5V I_{PD}: 1.5 µA typ. at 4V Freq: 4 MHz max.	V_{DD}: 2.5V to 6.0V I_{DD}: 3.8 mA max. at 3.0V I_{PD}: 5.0 µA max. at 3V Freq: 4 MHz max.	V_{DD}: 4.0V to 6.0V I_{DD}: 5 mA max. at 5.5V I_{PD}: 16 µA max. at 4V Freq: 4 MHz max.
XT	V_{DD}: 4.0V to 6.0V I_{DD}: 5 mA max. at 5.5V I_{PD}: 16 µA max. at 4V Freq: 4 MHz max.	V_{DD}: 4.5V to 5.5V I_{DD}: 2.7 mA typ. at 5.5V I_{PD}: 1.5 µA typ. at 4V Freq: 4 MHz max.	V_{DD}: 4.5V to 5.5V I_{DD}: 2.7 mA typ. at 5.5V I_{PD}: 1.5 µA typ. at 4V Freq: 4 MHz max.	V_{DD}: 2.5V to 6.0V I_{DD}: 3.8 mA max. at 3.0V I_{PD}: 5.0 µA max. at 3V Freq: 4 MHz max.	V_{DD}: 4.0V to 6.0V I_{DD}: 5 mA max. at 5.5V I_{PD}: 16 µA max. at 4V Freq: 4 MHz max.
HS	V_{DD}: 4.5V to 5.5V I_{DD}: 13.5 mA typ. at 5.5V I_{PD}: 1.5 µA typ. at 4.5V Freq: 4 MHz max.	V_{DD}: 4.5V to 5.5V I_{DD}: 10 mA max. at 5.5V I_{PD}: 1.5 µA typ. at 4.5V Freq: 10 MHz max.	V_{DD}: 4.5V to 5.5V I_{DD}: 20 mA max. at 5.5V I_{PD}: 1.5 µA typ. at 4.5V Freq: 20 MHz max.	Not recommended for use in HS mode	V_{DD}: 4.5V to 5.5V I_{DD}: 20 mA max. at 5.5V I_{PD}: 1.5 µA typ. at 4.5V Freq: 20 MHz max.
LP	V_{DD}: 4.0V to 6.0V I_{DD}: 52.5 µA typ. at 32 kHz, 4.0V I_{PD}: 0.9 µA typ. at 4.0V Freq: 200 kHz max.	Not recommended for use in LP mode	Not recommended for use in LP mode	V_{DD}: 2.5V to 6.0V I_{DD}: 48 µA max. at 32 kHz, 3.0V I_{PD}: 5.0 µA max. at 3.0V Freq: 200 kHz max.	V_{DD}: 2.5V to 6.0V I_{DD}: 48 µA max. at 32 kHz, 3.0V I_{PD}: 5.0 µA max. at 3.0V Freq: 200 kHz max.

The shaded sections indicate oscillator selections which are tested for functionality, but not for MIN/MAX specifications.
It is recommended that the user select the device type that ensures the specifications required.

Applicable Devices	72	73	73A	74	74A	76	77

FIGURE 17-8: SPI MODE TIMING



Note: Refer to Figure 17-1 for load conditions

TABLE 17-7: SPI MODE REQUIREMENTS

Parameter No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
70	TssL2scH, TssL2scL	$\overline{SS}\downarrow$ to SCK$\downarrow$ or SCK$\uparrow$ input	TCY	—	—	ns	
71	TscH	SCK input high time (slave mode)	TCY + 20	—	—	ns	
72	TscL	SCK input low time (slave mode)	TCY + 20	—	—	ns	
73	TdiV2scH, TdiV2scL	Setup time of SDI data input to SCK edge	50	—	—	ns	
74	TscH2diL, TscL2diL	Hold time of SDI data input to SCK edge	50	—	—	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	
78	TscR	SCK output rise time (master mode)	—	10	25	ns	
79	TscF	SCK output fall time (master mode)	—	10	25	ns	
80	TscH2doV, TscL2doV	SDO data output valid after SCK edge	—	—	50	ns	

† Data in "Typ" column is at 5V, 25˚C unless otherwise stated. These parameters are for design guidance only and are not tested.

Applicable Devices	72	73	73A	74	74A	76	77

FIGURE 21-3: TYPICAL IPD vs. VDD @ 25°C (WDT ENABLED, RC MODE)

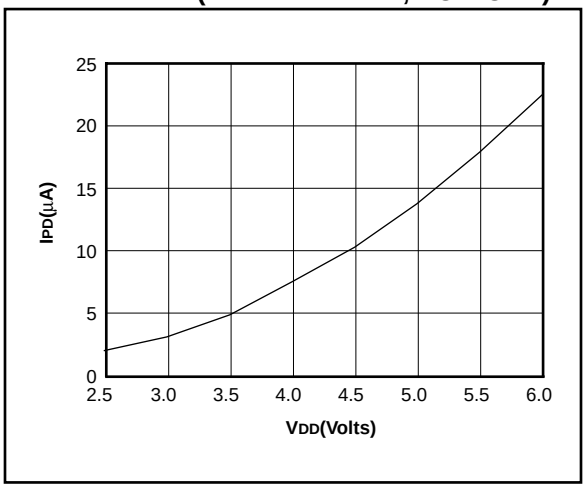


FIGURE 21-4: MAXIMUM IPD vs. VDD (WDT ENABLED, RC MODE)

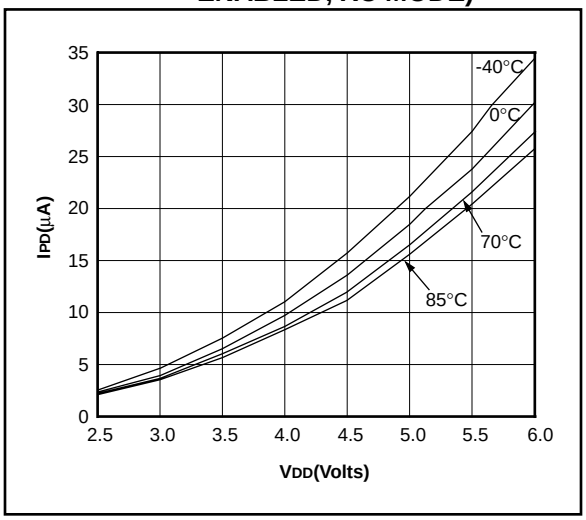
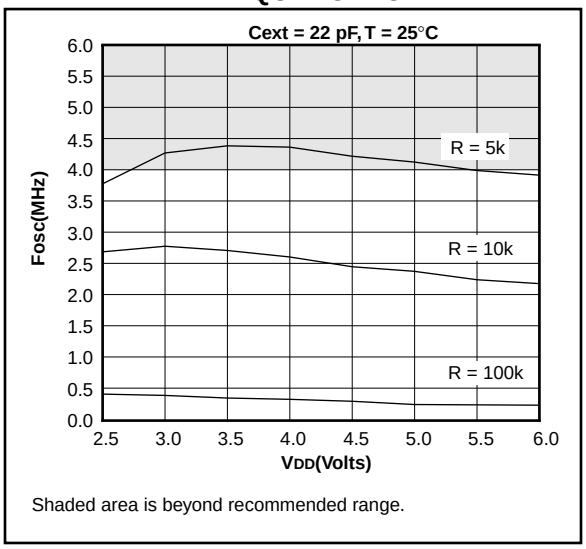


FIGURE 21-5: TYPICAL RC OSCILLATOR FREQUENCY vs. VDD



Shaded area is beyond recommended range.

FIGURE 21-6: TYPICAL RC OSCILLATOR FREQUENCY vs. VDD

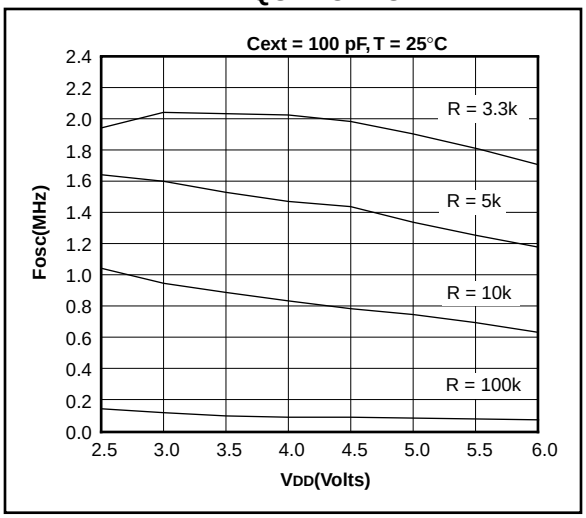
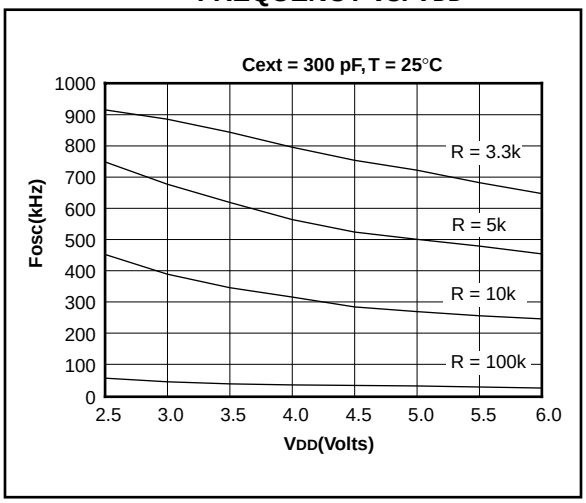


FIGURE 21-7: TYPICAL RC OSCILLATOR FREQUENCY vs. VDD



Data based on matrix samples. See first page of this section for details.

Applicable Devices	72	73	73A	74	74A	76	77

FIGURE 21-25: TYPICAL IDD vs. FREQUENCY (LP MODE, 25°C)

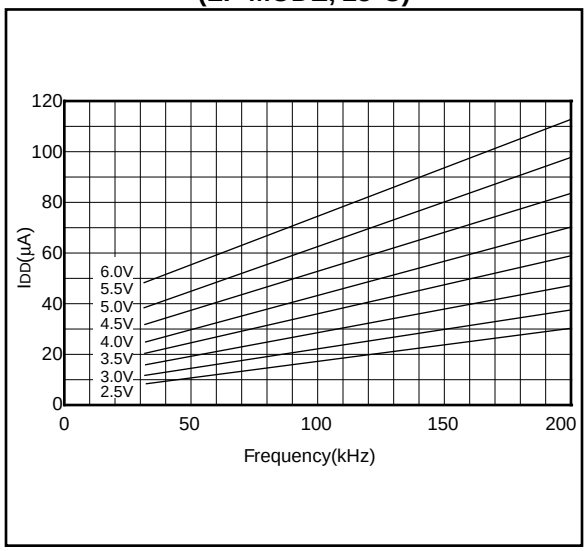


FIGURE 21-26: MAXIMUM IDD vs. FREQUENCY (LP MODE, 85°C TO -40°C)

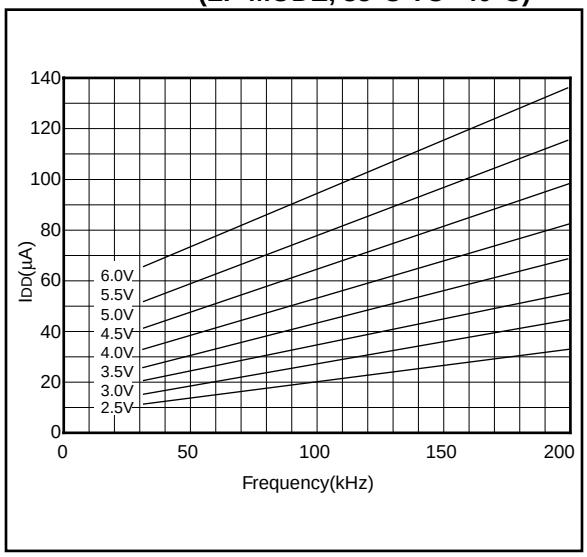


FIGURE 21-27: TYPICAL IDD vs. FREQUENCY (XT MODE, 25°C)

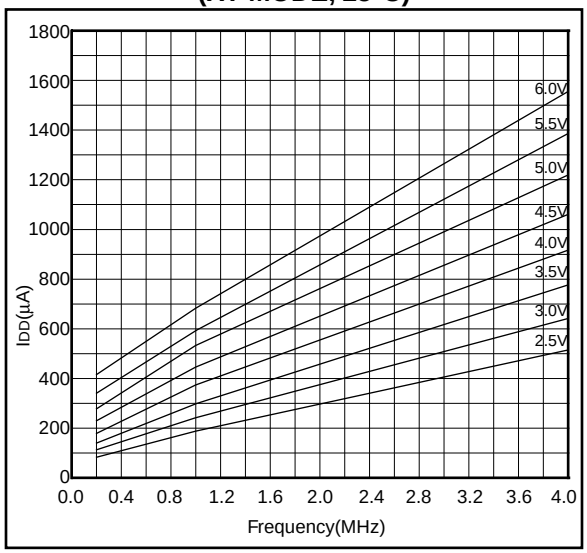
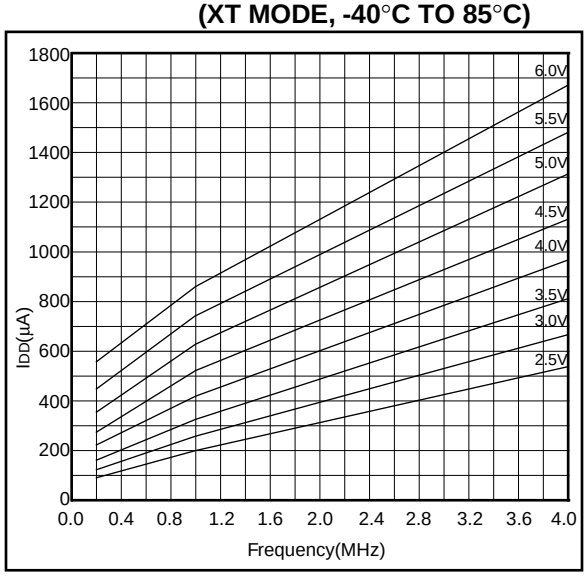
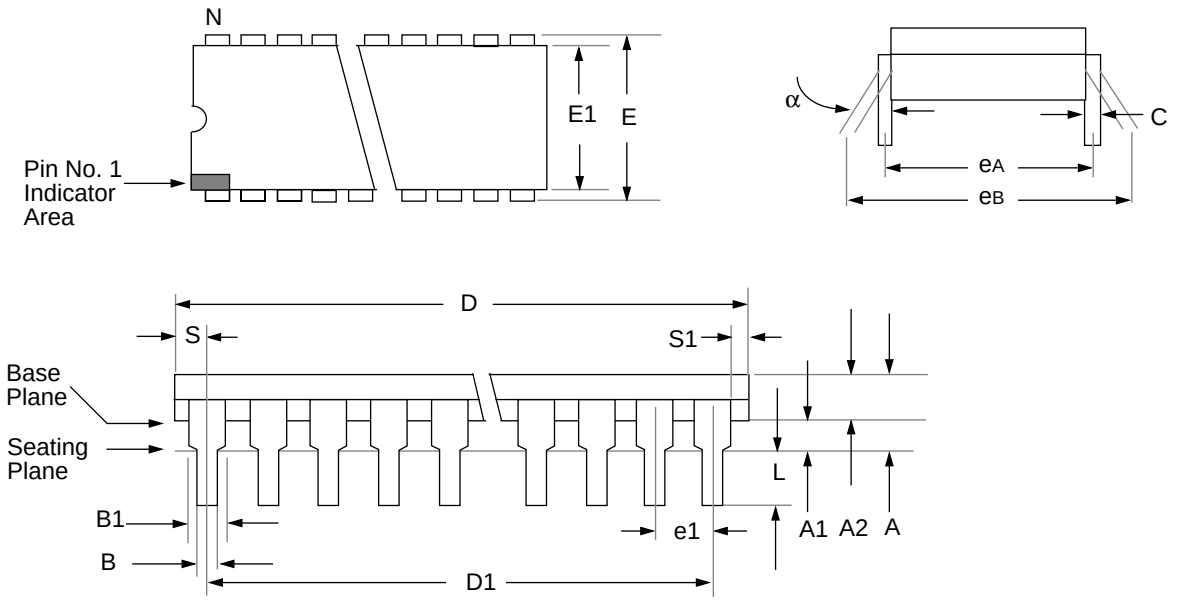


FIGURE 21-28: MAXIMUM IDD vs. FREQUENCY (XT MODE, -40°C TO 85°C)



Data based on matrix samples. See first page of this section for details.

22.4 40-Lead Plastic Dual In-line (600 mil) (P)



Package Group: Plastic Dual In-Line (PLA)						
	Millimeters			Inches		
Symbol	Min	Max	Notes	Min	Max	Notes
α	0°	10°		0°	10°	
A	–	5.080		–	0.200	
A1	0.381	–		0.015	–	
A2	3.175	4.064		0.125	0.160	
B	0.355	0.559		0.014	0.022	
B1	1.270	1.778	**Typical**	0.050	0.070	**Typical**
C	0.203	0.381	**Typical**	0.008	0.015	**Typical**
D	51.181	52.197		2.015	2.055	
D1	48.260	48.260	**Reference**	1.900	1.900	**Reference**
E	15.240	15.875		0.600	0.625	
E1	13.462	13.970		0.530	0.550	
e1	2.489	2.591	**Typical**	0.098	0.102	**Typical**
eA	15.240	15.240	**Reference**	0.600	0.600	**Reference**
eB	15.240	17.272		0.600	0.680	
L	2.921	3.683		0.115	0.145	
N	40	40		40	40	
S	1.270	–		0.050	–	
S1	0.508	–		0.020	–	

APPENDIX A:

The following are the list of modifications over the PIC16C5X microcontroller family:

1. Instruction word length is increased to 14-bits. This allows larger page sizes both in program memory (2K now as opposed to 512 before) and register file (128 bytes now versus 32 bytes before).
2. A PC high latch register (PCLATH) is added to handle program memory paging. Bits PA2, PA1, PA0 are removed from STATUS register.
3. Data memory paging is redefined slightly. STATUS register is modified.
4. Four new instructions have been added: `RETURN, RETFIE, ADDLW`, and `SUBLW`. Two instructions `TRIS` and `OPTION` are being phased out although they are kept for compatibility with PIC16C5X.
5. OPTION and TRIS registers are made addressable.
6. Interrupt capability is added. Interrupt vector is at 0004h.
7. Stack size is increased to 8 deep.
8. Reset vector is changed to 0000h.
9. Reset of all registers is revisited. Five different reset (and wake-up) types are recognized. Registers are reset differently.
10. Wake up from SLEEP through interrupt is added.
11. Two separate timers, Oscillator Start-up Timer (OST) and Power-up Timer (PWRT) are included for more reliable power-up. These timers are invoked selectively to avoid unnecessary delays on power-up and wake-up.
12. PORTB has weak pull-ups and interrupt on change feature.
13. T0CKI pin is also a port pin (RA4) now.
14. FSR is made a full eight bit register.
15. “In-circuit serial programming” is made possible. The user can program PIC16CXX devices using only five pins: V_{DD}, V_{SS}, $\overline{MCLR}/V_{PP}$, RB6 (clock) and RB7 (data in/out).
16. PCON status register is added with a Power-on Reset status bit ($\overline{POR}$).
17. Code protection scheme is enhanced such that portions of the program memory can be protected, while the remainder is unprotected.
18. Brown-out protection circuitry has been added. Controlled by configuration word bit BODEN. Brown-out reset ensures the device is placed in a reset condition if V_{DD} dips below a fixed setpoint.

APPENDIX B: COMPATIBILITY

To convert code written for PIC16C5X to PIC16CXX, the user should take the following steps:

1. Remove any program memory page select operations (PA2, PA1, PA0 bits) for `CALL`, `GOTO`.
2. Revisit any computed jump operations (write to PC or add to PC, etc.) to make sure page bits are set properly under the new scheme.
3. Eliminate any data memory page switching. Redefine data variables to reallocate them.
4. Verify all writes to STATUS, OPTION, and FSR registers since these have changed.
5. Change reset vector to 0000h.

E.5 PIC16C55X Family of Devices

		PIC16C554	PIC16C556(1)	PIC16C558
Clock	Maximum Frequency of Operation (MHz)	20	20	20
Memory	EPROM Program Memory (x14 words)	512	1K	2K
	Data Memory (bytes)	80	80	128
Peripherals	Timer Module(s)	TMR0	TMR0	TMR0
	Comparators(s)	—	—	—
	Internal Reference Voltage	—	—	—
Features	Interrupt Sources	3	3	3
	I/O Pins	13	13	13
	Voltage Range (Volts)	2.5-6.0	2.5-6.0	2.5-6.0
	Brown-out Reset	—	—	—
	Packages	18-pin DIP, SOIC; 20-pin SSOP	18-pin DIP, SOIC; 20-pin SSOP	18-pin DIP, SOIC; 20-pin SSOP

All PIC16/17 Family devices have Power-on Reset, selectable Watchdog Timer, selectable code protect and high I/O current capability. All PIC16C5XX Family devices use serial programming with clock pin RB6 and data pin RB7.

Note 1: Please contact your local Microchip sales office for availability of these devices.

E.6 PIC16C62X and PIC16C64X Family of Devices

		PIC16C620	PIC16C621	PIC16C622	PIC16C642	PIC16C662
Clock	Maximum Frequency of Operation (MHz)	20	20	20	20	20
Memory	EPROM Program Memory (x14 words)	512	1K	2K	4K	4K
	Data Memory (bytes)	80	80	128	176	176
Peripherals	Timer Module(s)	TMR0	TMR0	TMR0	TMR0	TMR0
	Comparators(s)	2	2	2	2	2
	Internal Reference Voltage	Yes	Yes	Yes	Yes	Yes
Features	Interrupt Sources	4	4	4	4	5
	I/O Pins	13	13	13	22	33
	Voltage Range (Volts)	2.5-6.0	2.5-6.0	2.5-6.0	3.0-6.0	3.0-6.0
	Brown-out Reset	Yes	Yes	Yes	Yes	Yes
	Packages	18-pin DIP, SOIC; 20-pin SSOP	18-pin DIP, SOIC; 20-pin SSOP	18-pin DIP, SOIC; 20-pin SSOP	28-pin PDIP, SOIC, Windowed CDIP	40-pin PDIP, Windowed CDIP; 44-pin PLCC, MQFP

All PIC16/17 Family devices have Power-on Reset, selectable Watchdog Timer, selectable code protect and high I/O current capability. All PIC16C62X and PIC16C64X Family devices use serial programming with clock pin RB6 and data pin RB7.