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
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	22
Program Memory Size	3.5KB (2K x 14)
Program Memory Type	OTP
EEPROM Size	-
RAM Size	128 x 8
Voltage - Supply (Vcc/Vdd)	4V ~ 5.5V
Data Converters	A/D 5x8b
Oscillator Type	External
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	28-SSOP (0.209", 5.30mm Width)
Supplier Device Package	28-SSOP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16c72at-20i-ss

2.2.2.6 PCON REGISTER

The Power Control register (PCON) contains flag bits to allow differentiation between a Power-on Reset (POR), Brown-Out Reset (BOR) and resets from other sources. .

Note:	On Power-on Reset, the state of the $\overline{BOR}$ bit is unknown and is not predictable. If the BODEN bit in the configuration word is set, the user must first set the BOR bit on a POR, and check it on subsequent resets. If BOR is cleared while POR remains set, a Brown-out reset has occurred. If the BODEN bit is clear, the BOR bit may be ignored.

REGISTER 2-6: PCON REGISTER (ADDRESS 8Eh)

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-q
—	—	—	—	—	—	$\overline{POR}$	$\overline{BOR}$
bit7							bit0

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

bit 7-2: **Unimplemented:** Read as '0'

bit 1: **$\overline{POR}$**: Power-on Reset Status bit
1 = No Power-on Reset occurred
0 = A Power-on Reset occurred (must be set in software after a Power-on Reset occurs)

bit 0: **$\overline{BOR}$**: Brown-out Reset Status bit
1 = No Brown-out Reset occurred
0 = A Brown-out Reset occurred (must be set in software after a Brown-out Reset occurs)

TABLE 3-3 PORTB FUNCTIONS

Name	Bit#	Buffer	Function
RB0/INT	bit0	TTL/ST(1)	Input/output pin or external interrupt input. Internal software programmable weak pull-up.
RB1	bit1	TTL	Input/output pin. Internal software programmable weak pull-up.
RB2	bit2	TTL	Input/output pin. Internal software programmable weak pull-up.
RB3	bit3	TTL	Input/output pin. Internal software programmable weak pull-up.
RB4	bit4	TTL	Input/output pin (with interrupt on change). Internal software programmable weak pull-up.
RB5	bit5	TTL	Input/output pin (with interrupt on change). Internal software programmable weak pull-up.
RB6	bit6	TTL/ST(2)	Input/output pin (with interrupt on change). Internal software programmable weak pull-up. Serial programming clock.
RB7	bit7	TTL/ST(2)	Input/output pin (with interrupt on change). Internal software programmable weak pull-up. Serial programming data.

Legend: TTL = TTL input, ST = Schmitt Trigger input

Note 1: This buffer is a Schmitt Trigger input when configured as the external interrupt.

2: This buffer is a Schmitt Trigger input when used in serial programming mode.

TABLE 3-4 SUMMARY OF REGISTERS ASSOCIATED WITH PORTB

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
06h	PORTB	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	xxxx xxxx	uuuu uuuu
86h	TRISB	PORTB Data Direction Register								1111 1111	1111 1111
81h	OPTION_REG	$\overline{\text{RBPU}}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111

Legend: x = unknown, u = unchanged. Shaded cells are not used by PORTB.

REGISTER 8-1: SSPSTAT: SYNC SERIAL PORT STATUS REGISTER (ADDRESS 94h)

R/W-0	R/W-0	R-0	R-0	R-0	R-0	R-0	R-0
SMP	CKE	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: **SMP:** SPI data input sample phase
SPI Master Operation
1 = Input data sampled at end of data output time
0 = Input data sampled at middle of data output time
SPI Slave Mode
SMP must be cleared when SPI is used in slave mode
I^2C Mode
This bit must be maintained clear

bit 6: **CKE**: SPI Clock Edge Select
SPI Mode
CKP = 0
1 = Data transmitted on rising edge of SCK
0 = Data transmitted on falling edge of SCK
CKP = 1
1 = Data transmitted on falling edge of SCK
0 = Data transmitted on rising edge of SCK
I^2C Mode
This bit must be maintained clear

bit 5: **D/$\overline{A}$**: Data/$\overline{Address}$ 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, or when the Start bit is detected last, 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, or when the Stop bit is detected last, 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 only 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

NOTES:

10.8 Time-out Sequence

When a POR reset occurs, the PWRT delay starts (if enabled). When PWRT ends, the OST counts 1024 oscillator cycles (LP, XT, HS modes only). When OST completes, the device comes out of reset. The total time-out will vary based on oscillator configuration and the status of the PWRT. For example, in RC mode with the PWRT disabled, there will be no time-out at all.

If $\overline{MCLR}$ is kept low long enough, the time-outs will expire. Bringing $\overline{MCLR}$ high will begin execution immediately. This is useful for testing purposes or to synchronize more than one PIC16CXXX device operating in parallel.

Table 10-5 shows the reset conditions for the STATUS, PCON and PC registers, while Table 10-6 shows the reset conditions for all the registers.

10.9 Power Control/Status Register (PCON)

The $\overline{BOR}$ bit is unknown on Power-on Reset. If the Brown-out Reset circuit is used, the $\overline{BOR}$ bit must be set by the user and checked on subsequent resets to see if it was cleared, indicating a Brown-out has occurred.

$\overline{POR}$ (Power-on Reset Status bit) is cleared on a Power-on Reset and unaffected otherwise. The user

Status Register

IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C

PCON Register

						$\overline{POR}$	$\overline{BOR}$

TABLE 10-3 TIME-OUT IN VARIOUS SITUATIONS

Oscillator Configuration	Power-up		Brown-out	Wake-up from SLEEP
	$\overline{PWRTE}$ = 0	$\overline{PWRTE}$ = 1		
XT, HS, LP	72 ms + 1024TOSC	1024TOSC	72 ms + 1024TOSC	1024TOSC
RC	72 ms	—	72 ms	—

TABLE 10-4 STATUS BITS AND THEIR SIGNIFICANCE

$\overline{POR}$	$\overline{BOR}$	$\overline{TO}$	$\overline{PD}$	
0	x	1	1	Power-on Reset
0	x	0	x	Illegal, $\overline{TO}$ is set on $\overline{POR}$
0	x	x	0	Illegal, $\overline{PD}$ is set on $\overline{POR}$
1	0	1	1	Brown-out Reset
1	1	0	1	WDT Reset
1	1	0	0	WDT Wake-up
1	1	u	u	$\overline{MCLR}$ Reset during normal operation
1	1	1	0	$\overline{MCLR}$ Reset during SLEEP or interrupt wake-up from SLEEP

TABLE 10-5 RESET CONDITION FOR SPECIAL REGISTERS

Condition	Program Counter	STATUS Register	PCON Register
Power-on Reset	000h	0001 1xxx	---- --0x
$\overline{MCLR}$ Reset during normal operation	000h	000u uuuu	---- --uu
$\overline{MCLR}$ Reset during SLEEP	000h	0001 0uuu	---- --uu
WDT Reset	000h	0000 1uuu	---- --uu
WDT Wake-up	PC + 1	uuu0 0uuu	---- --uu
Brown-out Reset	000h	0001 1uuu	---- --u0
Interrupt wake-up from SLEEP	PC + 1(1)	uuu1 0uuu	---- --uu

Legend: u = unchanged, x = unknown, - = unimplemented bit read as '0'.

Note 1: When the wake-up is due to an interrupt and the GIE bit is set, the PC is loaded with the interrupt vector (0004h).

TABLE 10-6 INITIALIZATION CONDITIONS FOR ALL REGISTERS

Register	Applicable Devices		Power-on Reset, Brown-out Reset	$\overline{\text{MCLR}}$ Resets WDT Reset	Wake-up via WDT or Interrupt
W	62B	72A	xxxx xxxx	uuuu uuuu	uuuu uuuu
INDF	62B	72A	N/A	N/A	N/A
TMR0	62B	72A	xxxx xxxx	uuuu uuuu	uuuu uuuu
PCL	62B	72A	0000h	0000h	PC + 1(2)
STATUS	62B	72A	0001 1xxx	000q quuu(3)	uuuq quuu(3)
FSR	62B	72A	xxxx xxxx	uuuu uuuu	uuuu uuuu
PORTA(4)	62B	72A	--0x 0000	--0u 0000	--uu uuuu
PORTB(5)	62B	72A	xxxx xxxx	uuuu uuuu	uuuu uuuu
PORTC(5)	62B	72A	xxxx xxxx	uuuu uuuu	uuuu uuuu
PCLATH	62B	72A	---0 0000	---0 0000	---u uuuu
INTCON	62B	72A	0000 000x	0000 000u	uuuu uuuu(1)
PIR1	62B	72A	---- 0000	---- 0000	---- uuuu(1)
	62B	72A	-0-- 0000	-0-- 0000	-u-- uuuu(1)
TMR1L	62B	72A	xxxx xxxx	uuuu uuuu	uuuu uuuu
TMR1H	62B	72A	xxxx xxxx	uuuu uuuu	uuuu uuuu
T1CON	62B	72A	--00 0000	--uu uuuu	--uu uuuu
TMR2	62B	72A	0000 0000	0000 0000	uuuu uuuu
T2CON	62B	72A	-000 0000	-000 0000	-uuu uuuu
SSPBUF	62B	72A	xxxx xxxx	uuuu uuuu	uuuu uuuu
SSPCON	62B	72A	0000 0000	0000 0000	uuuu uuuu
CCPR1L	62B	72A	xxxx xxxx	uuuu uuuu	uuuu uuuu
CCPR1H	62B	72A	xxxx xxxx	uuuu uuuu	uuuu uuuu
CCP1CON	62B	72A	--00 0000	--00 0000	--uu uuuu
ADRES	62B	72A	xxxx xxxx	uuuu uuuu	uuuu uuuu
ADCON0	62B	72A	0000 00-0	0000 00-0	uuuu uu-u
OPTION_REG	62B	72A	1111 1111	1111 1111	uuuu uuuu
TRISA	62B	72A	--11 1111	--11 1111	--uu uuuu
TRISB	62B	72A	1111 1111	1111 1111	uuuu uuuu
TRISC	62B	72A	1111 1111	1111 1111	uuuu uuuu
PIE1	62B	72A	---- 0000	---- 0000	---- uuuu
	62B	72A	-0-- 0000	-0-- 0000	-u-- uuuu
PCON	62B	72A	---- --0q	---- --uq	---- --uq
PR2	62B	72A	1111 1111	1111 1111	1111 1111
SSPADD	62B	72A	0000 0000	0000 0000	uuuu uuuu
SSPSTAT	62B	72A	0000 0000	0000 0000	uuuu uuuu
ADCON1	62B	72A	---- -000	---- -000	---- -uuu

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

Note 1: One or more bits in INTCON and/or PIR1 will be affected (to cause wake-up).

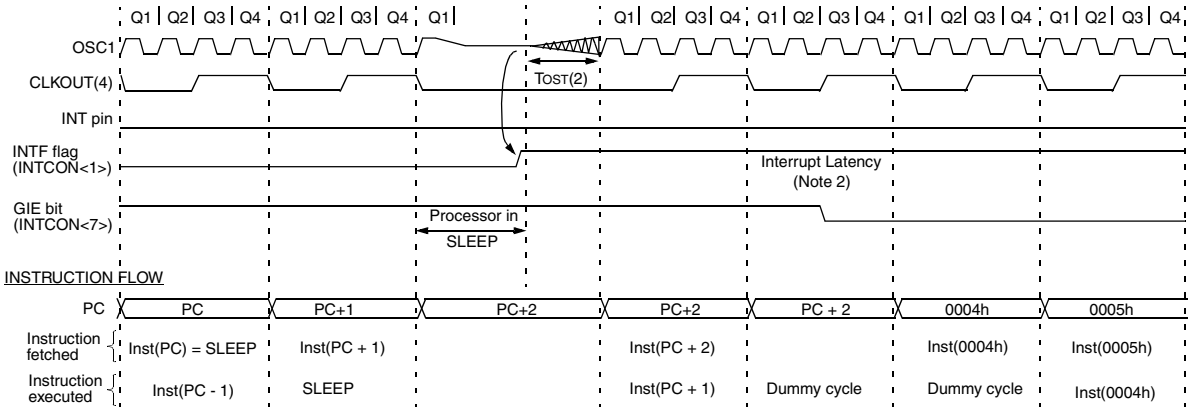
2: When the wake-up is due to an interrupt and the GIE bit is set, the PC is loaded with the interrupt vector (0004h).

3: See Table 10-5 for reset value for specific condition.

4: On any device reset, these pins are configured as inputs.

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

FIGURE 10-10: WAKE-UP FROM SLEEP THROUGH INTERRUPT



Note 1: XT, HS or LP oscillator mode assumed.
2: TOST = 1024TOSC (drawing not to scale) This delay will not be there for RC osc mode.
3: GIE = '1' assumed. In this case after wake- up, the processor jumps to the interrupt routine. If GIE = '0', execution will continue in-line.
4: CLKOUT is not available in these osc modes, but shown here for timing reference.

10.14 Program Verification/Code Protection

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

Note:	Microchip does not recommend code protecting windowed devices.

10.15 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.

For ROM devices, these values are submitted along with the ROM code.

10.16 In-Circuit Serial Programming™

PIC16CXXX microcontrollers can be serially programmed while in the end application circuit. This is simply done with two lines for clock and data, and three more lines for power, ground and the programming voltage. 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 complete details of serial programming, please refer to the In-Circuit Serial Programming (ICSP™) Guide, DS30277.

11.0 INSTRUCTION SET SUMMARY

Each PIC16CXXX instruction is a 14-bit word divided into an OPCODE which specifies the instruction type and one or more operands which further specify the operation of the instruction. The PIC16CXX instruction set summary in Table 11-2 lists **byte-oriented**, **bit-oriented**, and **literal and control** operations. Table 11-1 shows the opcode field descriptions.

For **byte-oriented** instructions, 'f' represents a file register designator and 'd' represents a destination designator. The file register designator specifies which file register is to be used by the instruction.

The destination designator specifies where the result of the operation is to be placed. If 'd' is zero, the result is placed in the W register. If 'd' is one, the result is placed in the file register specified in the instruction.

For **bit-oriented** instructions, 'b' represents a bit field designator which selects the number of the bit affected by the operation, while 'f' represents the number of the file in which the bit is located.

For **literal and control** operations, 'k' represents an eight or eleven bit constant or literal value.

TABLE 11-1 OPCODE FIELD DESCRIPTIONS

Field	Description
f	Register file address (0x00 to 0x7F)
W	Working register (accumulator)
b	Bit address within an 8-bit file register
k	Literal field, constant data or label
x	Don't care location (= 0 or 1) The assembler will generate code with x = 0. It is the recommended form of use for compatibility with all Microchip software tools.
d	Destination select; d = 0: store result in W, d = 1: store result in file register f. Default is d = 1
PC	Program Counter
$\overline{TO}$	Time-out bit
$\overline{PD}$	Power-down bit
Z	Zero bit
DC	Digit Carry bit
C	Carry bit

The instruction set is highly orthogonal and is grouped into three basic categories:

- **Byte-oriented** operations
- **Bit-oriented** operations
- **Literal and control** operations

All instructions are executed within one single instruction cycle, unless a conditional test is true or the program counter is changed as a result of an instruction. In this case, the execution takes two instruction cycles with the second cycle executed as a NOP. One instruction cycle consists of four oscillator periods. Thus, for an oscillator frequency of 4 MHz, the normal instruction execution time is 1 μs. If a conditional test is true or the program counter is changed as a result of an instruction, the instruction execution time is 2 μs.

Table 11-2 lists the instructions recognized by the MPASM assembler.

Figure 11-1 shows the general formats that the instructions can have.

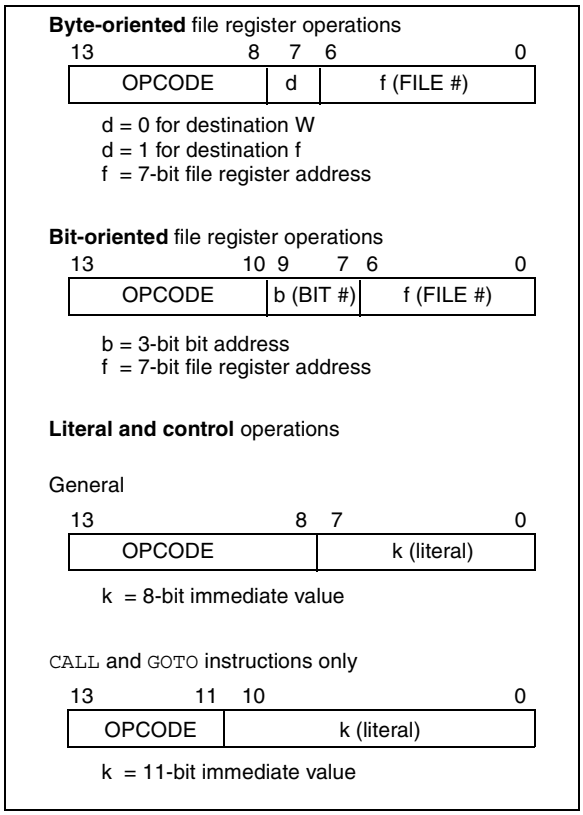
Note: To maintain upward compatibility with future PIC16CXXX products, do not use the OPTION and TRIS instructions.

All examples use the following format to represent a hexadecimal number:

0xhh

where h signifies a hexadecimal digit.

FIGURE 11-1: GENERAL FORMAT FOR INSTRUCTIONS



A description of each instruction is available in the PIC® MCU Mid-Range Reference Manual, (DS33023).

TABLE 11-2 PIC16CXXX INSTRUCTION SET

Mnemonic, Operands		Description	Cycles	14-Bit Opcode MSb			LSb	Status Affected	Notes
BYTE-ORIENTED FILE REGISTER OPERATIONS									
ADDWF	**f, d**	Add W and f	1	00	0111	dfff	ffff	C,DC,Z	1,2
ANDWF	**f, d**	AND W with f	1	00	0101	dfff	ffff	Z	1,2
CLRF	**f**	Clear f	1	00	0001	lfff	ffff	Z	2
CLRW	**-**	Clear W	1	00	0001	0000	0011	Z	
COMF	**f, d**	Complement f	1	00	1001	dfff	ffff	Z	1,2
DECF	**f, d**	Decrement f	1	00	0011	dfff	ffff	Z	1,2
DECFSZ	**f, d**	Decrement f, Skip if 0	1(2)	00	1011	dfff	ffff		1,2,3
INCF	**f, d**	Increment f	1	00	1010	dfff	ffff	Z	1,2
INCFSZ	**f, d**	Increment f, Skip if 0	1(2)	00	1111	dfff	ffff		1,2,3
IORWF	**f, d**	Inclusive OR W with f	1	00	0100	dfff	ffff	Z	1,2
MOVF	**f, d**	Move f	1	00	1000	dfff	ffff	Z	1,2
MOVWF	**f**	Move W to f	1	00	0000	lfff	ffff		
NOP	**-**	No Operation	1	00	0000	0xx0	0000		
RLF	**f, d**	Rotate Left f through Carry	1	00	1101	dfff	ffff	C	1,2
RRF	**f, d**	Rotate Right f through Carry	1	00	1100	dfff	ffff	C	1,2
SUBWF	**f, d**	Subtract W from f	1	00	0010	dfff	ffff	C,DC,Z	1,2
SWAPF	**f, d**	Swap nibbles in f	1	00	1110	dfff	ffff		1,2
XORWF	**f, d**	Exclusive OR W with f	1	00	0110	dfff	ffff	Z	1,2
BIT-ORIENTED FILE REGISTER OPERATIONS									
BCF	**f, b**	Bit Clear f	1	01	00bb	bfff	ffff		1,2
BSF	**f, b**	Bit Set f	1	01	01bb	bfff	ffff		1,2
BTFSC	**f, b**	Bit Test f, Skip if Clear	1 (2)	01	10bb	bfff	ffff		3
BTFSS	**f, b**	Bit Test f, Skip if Set	1 (2)	01	11bb	bfff	ffff		3
LITERAL AND CONTROL OPERATIONS									
ADDLW	**k**	Add literal and W	1	11	111x	kkkk	kkkk	C,DC,Z	
ANDLW	**k**	AND literal with W	1	11	1001	kkkk	kkkk	Z	
CALL	**k**	Call subroutine	2	10	0kkk	kkkk	kkkk		
CLRWDT	**-**	Clear Watchdog Timer	1	00	0000	0110	0100	$\overline{TO}$,$\overline{PD}$	
GOTO	**k**	Go to address	2	10	1kkk	kkkk	kkkk		
IORLW	**k**	Inclusive OR literal with W	1	11	1000	kkkk	kkkk	Z	
MOVLW	**k**	Move literal to W	1	11	00xx	kkkk	kkkk		
RETFIE	**-**	Return from interrupt	2	00	0000	0000	1001		
RETLW	**k**	Return with literal in W	2	11	01xx	kkkk	kkkk		
RETURN	**-**	Return from Subroutine	2	00	0000	0000	1000		
SLEEP	**-**	Go into standby mode	1	00	0000	0110	0011	$\overline{TO}$,$\overline{PD}$	
SUBLW	**k**	Subtract W from literal	1	11	110x	kkkk	kkkk	C,DC,Z	
XORLW	**k**	Exclusive OR literal with W	1	11	1010	kkkk	kkkk	Z	

Note 1: When an I/O register is modified as a function of itself (e.g., `MOVF PORTB, 1`), the value used will be that value present on the pins themselves. For example, if the data latch is '1' for a pin configured as input and is driven low by an external device, the data will be written back with a '0'.

2: If this instruction is executed on the TMR0 register (and, where applicable, d = 1), the prescaler will be cleared if assigned to the Timer0 Module.

3: If Program Counter (PC) is modified or a conditional test is true, the instruction requires two cycles. The second cycle is executed as a NOP.

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
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'.

GOTO	Unconditional Branch
Syntax:	[*label*] GOTO k
Operands:	$0 \le k \le 2047$
Operation:	$k \rightarrow$ PC<10:0> PCLATH<4:3> $\rightarrow$ PC<12:11>
Status Affected:	None
Description:	GOTO is an unconditional branch. The eleven bit immediate value is loaded into PC bits <10:0>. The upper bits of PC are loaded from PCLATH<4:3>. GOTO is a two cycle instruction.

DECF	Decrement f
Syntax:	[*label*] DECF f,d
Operands:	$0 \le f \le 127$ $d \in [0,1]$
Operation:	(f) - 1 $\rightarrow$ (destination)
Status Affected:	Z
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'.

INCF	Increment f
Syntax:	[*label*] INCF f,d
Operands:	$0 \le f \le 127$ $d \in [0,1]$
Operation:	(f) + 1 $\rightarrow$ (destination)
Status Affected:	Z
Description:	The contents of register 'f' are incremented. If 'd' is 0, the result is placed in the W register. If 'd' is 1, the result is placed back in register 'f'.

DECFSZ	Decrement f, Skip if 0
Syntax:	[*label*] DECFSZ f,d
Operands:	$0 \le f \le 127$ $d \in [0,1]$
Operation:	(f) - 1 $\rightarrow$ (destination); skip if result = 0
Status Affected:	None
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.

INCFSZ	Increment f, Skip if 0
Syntax:	[*label*] INCFSZ f,d
Operands:	$0 \le f \le 127$ $d \in [0,1]$
Operation:	(f) + 1 $\rightarrow$ (destination), skip if result = 0
Status Affected:	None
Description:	The contents of register 'f' are incremented. 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, a NOP is executed instead making it a 2TCY instruction.

12.0 DEVELOPMENT SUPPORT

The PIC® microcontrollers are supported with a full range of hardware and software development tools:

- Integrated Development Environment
  - MPLAB™ IDE Software
- Assemblers/Compilers/Linkers
  - MPASM Assembler
  - MPLAB-C17 and MPLAB-C18 C Compilers
  - MPLINK/MPLIB Linker/Librarian
- Simulators
  - MPLAB-SIM Software Simulator
- Emulators
  - MPLAB-ICE Real-Time In-Circuit Emulator
  - PICMASTER®/PICMASTER-CE In-Circuit Emulator
  - ICEPIC™
- In-Circuit Debugger
  - MPLAB-ICD for PIC16F877
- Device Programmers
  - PRO MATE® II Universal Programmer
  - PICSTART® Plus Entry-Level Prototype Programmer
- Low-Cost Demonstration Boards
  - SIMICE
  - PICDEM-1
  - PICDEM-2
  - PICDEM-3
  - PICDEM-17
  - SEEVAL®
  - KEELOQ®

12.1 MPLAB Integrated Development Environment Software

  - The MPLAB IDE software brings an ease of software development previously unseen in the 8-bit microcontroller market. MPLAB is a Windows®-based application which contains:
- Multiple functionality
  - editor
  - simulator
  - programmer (sold separately)
  - emulator (sold separately)
- A full featured editor
- A project manager
- Customizable tool bar and key mapping
- A status bar
- On-line help

MPLAB allows you to:

- Edit your source files (either assembly or 'C')
- One touch assemble (or compile) and download to PIC MCU tools (automatically updates all project information)
- Debug using:
  - source files
  - absolute listing file
  - object code

The ability to use MPLAB with Microchip's simulator, MPLAB-SIM, allows a consistent platform and the ability to easily switch from the cost-effective simulator to the full featured emulator with minimal retraining.

12.2 MPASM Assembler

MPASM is a full featured universal macro assembler for all PIC MCUs. It can produce absolute code directly in the form of HEX files for device programmers, or it can generate relocatable objects for MPLINK.

MPASM has a command line interface and a Windows shell and can be used as a standalone application on a Windows 3.x or greater system. MPASM generates relocatable object files, Intel standard HEX files, MAP files to detail memory usage and symbol reference, an absolute LST file which contains source lines and generated machine code, and a COD file for MPLAB debugging.

MPASM features include:

- MPASM and MPLINK are integrated into MPLAB projects.
- MPASM allows user defined macros to be created for streamlined assembly.
- MPASM allows conditional assembly for multi purpose source files.
- MPASM directives allow complete control over the assembly process.

12.3 MPLAB-C17 and MPLAB-C18 C Compilers

The MPLAB-C17 and MPLAB-C18 Code Development Systems are complete ANSI 'C' compilers and integrated development environments for Microchip's PIC17CXXX and PIC18CXXX family of microcontrollers, respectively. These compilers provide powerful integration capabilities and ease of use not found with other compilers.

For easier source level debugging, the compilers provide symbol information that is compatible with the MPLAB IDE memory display.

12.4 MPLINK/MPLIB Linker/Librarian

MPLINK is a relocatable linker for MPASM and MPLAB-C17 and MPLAB-C18. It can link relocatable objects from assembly or C source files along with precompiled libraries using directives from a linker script.

TABLE 12-1: DEVELOPMENT TOOLS FROM MICROCHIP

		PIC12CXXX	PIC14000	PIC16C5X	PIC16C6X	PIC16CXXX	PIC16F62X	PIC16C7X	PIC16C7XX	PIC16C8X	PIC16F8XX	PIC16C9XX	PIC17C4X	PIC17C7XX	PIC18CXX2	24CXX/ 25CXX/ 93CXX	HCSXXX	MCRFXXX	MCP2510
Software Tools	MPLAB™ Integrated Development Environment	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓				
	MPLAB™ C17 Compiler												✓	✓					
	MPLAB™ C18 Compiler														✓				
	MPASM/MPLINK	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		
Emulators	MPLAB™-ICE	✓	✓	✓	✓	✓	✓**	✓	✓	✓	✓	✓	✓	✓	✓				
	PICMASTER/PICMASTER-CE	✓	✓	✓	✓	✓		✓	✓	✓		✓	✓	✓					
	ICEPIC™ Low-Cost In-Circuit Emulator	✓		✓	✓	✓		✓	✓	✓		✓							
Debugger	MPLAB-ICD In-Circuit Debugger				✓*			✓*			✓								
Programmers	PICSTART®Plus Low-Cost Universal Dev. Kit	✓	✓	✓	✓	✓	✓**	✓	✓	✓	✓	✓	✓	✓	✓				
	PRO MATE® II Universal Programmer	✓	✓	✓	✓	✓	✓**	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		
Demo Boards and Eval Kits	SIMICE	✓		✓															
	PICDEM-1			✓		✓		✓†		✓			✓						
	PICDEM-2				✓†			✓†							✓				
	PICDEM-3											✓							
	PICDEM-14A		✓																
	PICDEM-17													✓					
	KEELOQ® Evaluation Kit																✓		
	KEELOQ Transponder Kit																✓		
	microID™ Programmer's Kit																	✓	
	125 kHz microID Developer's Kit																	✓	
	125 kHz Anticollision microID Developer's Kit																	✓	
	13.56 MHz Anticollision microID Developer's Kit																	✓	
	MCP2510 CAN Developer's Kit																		✓

* Contact the Microchip Technology Inc. web site at www.microchip.com for information on how to use the MPLAB-ICD In-Circuit Debugger (DV164001) with PIC16C62, 63, 64, 65, 72, 73, 74, 76, 77

** Contact Microchip Technology Inc. for availability date.

† Development tool is available on select devices.

NOTES:

13.0 ELECTRICAL CHARACTERISTICS

Absolute Maximum Ratings (†)

Ambient temperature under bias ..... -55°C 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} ..... -0.3V to +7.5V

Voltage on $\overline{MCLR}$ with respect to V_{SS} (Note 2) ..... 0V to +13.25V

Voltage on RA4 with respect to Vss ..... 0V to +8.5V

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: $P_{dis} = V_{DD} \times \{I_{DD} - \sum I_{OH}\} + \sum \{(V_{DD}-V_{OH}) \times I_{OH}\} + \sum(V_{ol} \times I_{OL})$

2: Voltage spikes below V_{SS} at the $\overline{MCLR}/V_{PP}$ 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 $\overline{MCLR}/V_{PP}$ 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.

13.3 DC Characteristics: PIC16C62B/72A-04 (Commercial, Industrial, Extended) PIC16C62B/72A-20 (Commercial, Industrial, Extended) PIC16LC62B/72A-04 (Commercial, Industrial)

DC CHARACTERISTICS			Standard Operating Conditions (unless otherwise stated) Operating temperature 0°C ≤ TA ≤ +70°C for commercial -40°C ≤ TA ≤ +85°C for industrial -40°C ≤ TA ≤+125°C for extended Operating voltage VDD range as described in DC spec Section 13.1 and Section 13.2				
Param No.	**Sym**	**Characteristic**	**Min**	**Typ†**	**Max**	**Units**	**Conditions**
		Input Low Voltage					
	VIL	I/O ports					
D030		with TTL buffer	VSS	-	0.15VDD	V	For entire VDD range
D030A			Vss	-	0.8V	V	4.5V ≤ VDD ≤ 5.5V
D031		with Schmitt Trigger buffer	VSS	-	0.2VDD	V	
D032		$\overline{MCLR}$, OSC1 (in RC mode)	Vss	-	0.2VDD	V	
D033		OSC1 (in XT, HS and LP modes)	Vss	-	0.3VDD	V	Note1
		Input High Voltage					
	VIH	I/O ports		-			
D040		with TTL buffer	2.0	-	VDD	V	4.5V ≤ VDD ≤ 5.5V
D040A			0.25VDD + 0.8V	-	Vdd	V	For entire VDD range
D041		with Schmitt Trigger buffer	0.8VDD	-	VDD	V	For entire VDD range
D042		MCLR	0.8VDD	-	VDD	V	
D042A		OSC1 (XT, HS and LP modes)	0.7VDD	-	VDD	V	Note1
D043		OSC1 (in RC mode)	0.9VDD	-	Vdd	V	
		Input Leakage Current (Notes 2, 3)					
D060	IIL	I/O ports	-	-	±1	µA	Vss ≤ VPIN ≤ VDD, Pin at hi-impedance
D061		$\overline{MCLR}$, RA4/T0CKI	-	-	±5	µA	Vss ≤ VPIN ≤ VDD
D063		OSC1	-	-	±5	µA	Vss ≤ VPIN ≤ VDD, XT, HS and LP osc modes
D070	IPURB	**PORTB weak pull-up current**	50	250	400	µA	VDD = 5V, VPIN = VSS
		Output Low Voltage					
D080	VOL	I/O ports	-	-	0.6	V	IOL = 8.5 mA, VDD = 4.5V, -40°C to +85°C

* These parameters are characterized but not tested.

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

Note 1: In RC oscillator mode, the OSC1/CLKIN pin is a Schmitt Trigger input. It is not recommended that the device be driven with external clock in RC mode.

2: The leakage current on the $\overline{MCLR}$/VPP pin is strongly dependent on the applied voltage level. The specified levels represent normal operating conditions. Higher leakage current may be measured at different input voltages.

3: Negative current is defined as current sourced by the pin.

13.4.3 TIMING DIAGRAMS AND SPECIFICATIONS

FIGURE 13-5: EXTERNAL CLOCK TIMING

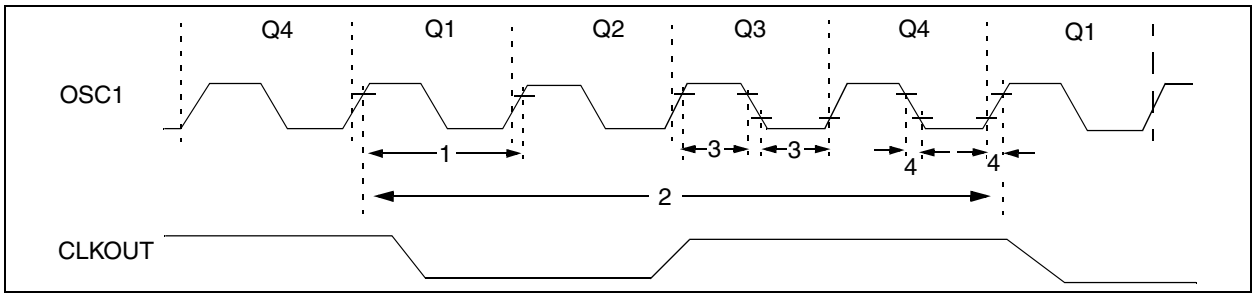


TABLE 13-2: EXTERNAL CLOCK TIMING REQUIREMENTS

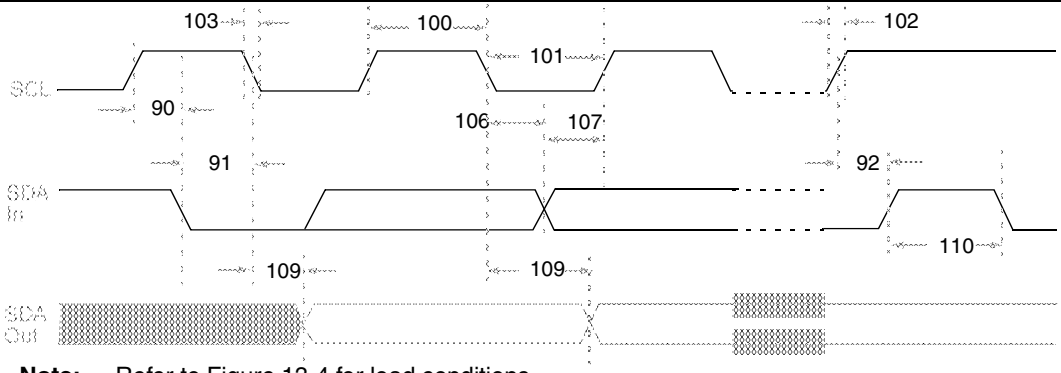
Param No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
1A	Fosc	External CLKIN Frequency (Note 1)	DC	—	4	MHz	RC and XT osc modes
			DC	—	4	MHz	HS osc mode (-04)
			DC	—	20	MHz	HS osc mode (-20)
			DC	—	200	kHz	LP osc mode
		Oscillator Frequency (Note 1)	DC	—	4	MHz	RC osc mode
			0.1	—	4	MHz	XT osc mode
			4	—	20	MHz	HS osc mode
			5	—	200	kHz	LP osc mode
1	Tosc	External CLKIN Period (Note 1)	250	—	—	ns	RC and XT osc modes
			250	—	—	ns	HS osc mode (-04)
			50	—	—	ns	HS osc mode (-20)
			5	—	—	µs	LP osc mode
		Oscillator Period (Note 1)	250	—	—	ns	RC osc mode
			250	—	10,000	ns	XT osc mode
			250	—	250	ns	HS osc mode (-04)
			50	—	250	ns	HS osc mode (-20)
			5	—	—	µs	LP osc mode
2	TCY	Instruction Cycle Time (Note 1)	200	—	DC	ns	TCY = 4/FOSC
3*	TosL, TosH	External Clock in (OSC1) High or Low Time	100	—	—	ns	XT oscillator
			2.5	—	—	µs	LP oscillator
			15	—	—	ns	HS oscillator
4*	TosR, TosF	External Clock in (OSC1) Rise or Fall Time	—	—	25	ns	XT oscillator
			—	—	50	ns	LP oscillator
			—	—	15	ns	HS oscillator

* These parameters are characterized but not tested.

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

Note 1: Instruction cycle period (TCY) equals four times the input oscillator time-base period. All specified values are based on characterization data for that particular oscillator type under standard operating conditions with the device executing code. Exceeding these specified limits may result in an unstable oscillator operation and/or higher than expected current consumption. All devices are tested to operate at "min." values with an external clock applied to the OSC1/CLKIN pin.
When an external clock input is used, the "Max." cycle time limit is "DC" (no clock) for all devices.

FIGURE 13-16: I²C BUS DATA TIMING



Note: Refer to Figure 13-4 for load conditions.

TABLE 13-12: I²C BUS DATA REQUIREMENTS

Param. No.	Sym	Characteristic		Min	Max	Units	Conditions
100*	THIGH	Clock high time	100 kHz mode	4.0	—	µs	Device must operate at a minimum of 1.5 MHz
			400 kHz mode	0.6	—	µs	Device must operate at a minimum of 10 MHz
			SSP Module	1.5TCY	—		
101*	TLOW	Clock low time	100 kHz mode	4.7	—	µs	Device must operate at a minimum of 1.5 MHz
			400 kHz mode	1.3	—	µs	Device must operate at a minimum of 10 MHz
			SSP Module	1.5TCY	—		
102*	TR	SDA and SCL rise time	100 kHz mode	—	1000	ns	
			400 kHz mode	20 + 0.1Cb	300	ns	Cb is specified to be from 10-400 pF
103*	TF	SDA and SCL fall time	100 kHz mode	—	300	ns	
			400 kHz mode	20 + 0.1Cb	300	ns	Cb is specified to be from 10-400 pF
90*	TSU:STA	START condition setup time	100 kHz mode	4.7	—	µs	Only relevant for repeated START condition
			400 kHz mode	0.6	—	µs	
91*	THD:STA	START condition hold time	100 kHz mode	4.0	—	µs	After this period the first clock pulse is generated
			400 kHz mode	0.6	—	µs	
106*	THD:DAT	Data input hold time	100 kHz mode	0	—	ns	
			400 kHz mode	0	0.9	µs	
107*	TSU:DAT	Data input setup time	100 kHz mode	250	—	ns	Note 2
			400 kHz mode	100	—	ns	
92*	TSU:STO	STOP condition setup time	100 kHz mode	4.7	—	µs	
			400 kHz mode	0.6	—	µs	
109*	TAA	Output valid from clock	100 kHz mode	—	3500	ns	Note 1
			400 kHz mode	—	—	ns	
110*	TBUF	Bus free time	100 kHz mode	4.7	—	µs	Time the bus must be free before a new transmission can start
			400 kHz mode	1.3	—	µs	
	Cb	Bus capacitive loading		—	400	pF	

* These parameters are characterized but not tested.

Note 1: As a transmitter, the device must provide this internal minimum delay time to bridge the undefined region (min. 300 ns) of the falling edge of SCL to avoid unintended generation of START or STOP conditions.

2: A fast-mode (400 kHz) I²C-bus device can be used in a standard-mode (100 kHz) I²C-bus system, but the requirement Tsu:DAT ≥ 250 ns must then be met. This will automatically be the case if the device does not stretch the LOW period of the SCL signal. If such a device does stretch the LOW period of the SCL signal, it must output the next data bit to the SDA line TR max.+tsu;DAT = 1000 + 250 = 1250 ns (according to the standard-mode I²C bus specification) before the SCL line is released.

APPENDIX C: MIGRATION FROM BASE-LINE TO MID-RANGE DEVICES

This section discusses how to migrate from a baseline device (i.e., PIC16C5X) to a mid-range device (i.e., PIC16CXXX).

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_REG 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.

To convert code written for PIC16C5X to PIC16CXXX, 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.

Q

R

S

PIC16C62B/72A PRODUCT IDENTIFICATION SYSTEM

To order or obtain information, e.g., on pricing or delivery refer to the factory or the listed sales office.

PART NO. **-XX** **X** **/XX** **XXX**

Pattern:	QTP, SQTP, Code or Special Requirements	
Package:	JW	= Windowed CERDIP
	SO	= SOIC
	SP	= Skinny plastic dip
	P	= PDIP
	SS	= SSOP
Temperature Range:	-	= 0°C to +70°C
	I	= -40°C to +85°C
	E	= -40°C to +125°C
Frequency Range:	04	= 4 MHz
	10	= 10 MHz
	20	= 20 MHz
Device	PIC16C62B:	VDD range 4.0V to 5.5V
	PIC16C62BT:	VDD range 4.0V to 5.5V (Tape/Reel)
	PIC16LC62B:	VDD range 2.5V to 5.5V
	PIC16LC62BT:	VDD range 2.5V to 5.5V (Tape/Reel)

Examples

a) PIC16C72A-04/P 301 Commercial Temp., PDIP Package, 4 MHz, normal VDD limits, QTP pattern #301

* JW Devices are UV erasable and can be programmed to any device configuration. JW Devices meet the electrical requirement of each oscillator type (including LC devices).

Sales and Support

Data Sheets
Products supported by a preliminary Data Sheet may have an errata sheet describing minor operational differences and recommended workarounds. To determine if an errata sheet exists for a particular device, please contact one of the following:

1. Your local Microchip sales office
2. The Microchip Corporate Literature Center U.S. FAX: (480) 786-7277
3. The Microchip Worldwide Site (www.microchip.com)

Please specify which device, revision of silicon and Data Sheet (include Literature #) you are using.

New Customer Notification System
Register on our web site (www.microchip.com/cn) to receive the most current information on our products.