



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 "<u>Embedded - Microcontrollers</u>"

Details	
Product Status	Active
Core Processor	PIC
Core Size	8-Bit
Speed	10MHz
Connectivity	I ² C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	33
Program Memory Size	14KB (8K x 14)
Program Memory Type	OTP
EEPROM Size	-
RAM Size	368 x 8
Voltage - Supply (Vcc/Vdd)	4V ~ 6V
Data Converters	-
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/pic16c67-10-p

NOTES:

4.2.2 SPECIAL FUNCTION REGISTERS:

The Special Function Registers are registers used by the CPU and peripheral modules for controlling the desired operation of the device. These registers are implemented as static RAM. The special function registers can be classified into two sets (core and peripheral). The registers associated with the "core" functions are described in this section and those related to the operation of the peripheral features are described in the section of that peripheral feature.

TABLE 4-1: SPECIAL FUNCTION REGISTERS FOR THE PIC16C61

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR	Value on all other resets ⁽³⁾
Bank 0											
00h ⁽¹⁾	INDF	Addressing	this location	uses conten	ts of FSR to	address data	a memory (n	ot a physical	register)	0000 0000	0000 0000
01h	TMR0	Timer0 mod	imer0 module's register							xxxx xxxx	uuuu uuuu
02h ⁽¹⁾	PCL	Program Co	ounter's (PC)	Least Signif	icant Byte					0000 0000	0000 0000
03h ⁽¹⁾	STATUS	IRP ⁽⁴⁾	IRP ⁽⁴⁾ RP1 ⁽⁴⁾ RP0 TO PD Z DC C								000q quuu
04h ⁽¹⁾	FSR	Indirect data	a memory ad	dress pointe	r					xxxx xxxx	uuuu uuuu
05h	PORTA	_	_	_	PORTA Dat	a Latch wher	n written: PC	RTA pins wh	en read	x xxxx	u uuuu
06h	PORTB	PORTB Dat	a Latch whe	n written: PC	RTB pins wi	nen read				xxxx xxxx	uuuu uuuu
07h	_	Unimpleme	nted							_	_
08h	_	Unimpleme	nted							_	_
09h	_	Unimpleme	Unimplemented							_	_
0Ah ^(1,2)	PCLATH	_	_	_	Write Buffer	for the uppe	er 5 bits of the	e Program C	ounter	0 0000	0 0000
0Bh ⁽¹⁾	INTCON	GIE	_	TOIE	INTE	RBIE	TOIF	INTF	RBIF	0-00 000x	0-00 000u
Bank 1											
80h ⁽¹⁾	INDF	Addressing	this location	uses conten	ts of FSR to	address data	a memory (n	ot a physical	register)	0000 0000	0000 0000
81h	OPTION	RBPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
82h ⁽¹⁾	PCL	Program Co	ounter's (PC)	Least Signif	icant Byte					0000 0000	0000 0000
83h ⁽¹⁾	STATUS	IRP ⁽⁴⁾	RP1 ⁽⁴⁾	RP0	TO	PD	Z	DC	С	0001 1xxx	000q quuu
84h ⁽¹⁾	FSR	Indirect data	a memory ad	dress pointe	r					xxxx xxxx	uuuu uuuu
85h	TRISA	_	_	_	PORTA Dat	a Direction F	Register			1 1111	1 1111
86h	TRISB	PORTB Dat	a Direction C	Control Regis	ster					1111 1111	1111 1111
87h	-	Unimplemented							_	_	
88h	-	- Unimplemented								_	_
89h	-	Unimplemented								_	_
8Ah ^(1,2)	PCLATH	_	_	_	Write Buffer	for the uppe	er 5 bits of the	e Program C	ounter	0 0000	0 0000
8Bh ⁽¹⁾	INTCON	GIE	_	TOIE	INTE	RBIE	TOIF	INTF	RBIF	0-00 000x	0-00 000u

 $\label{eq:localization} \textbf{Legend:} \quad \textbf{x} = \textbf{unknown}, \textbf{u} = \textbf{unchanged}, \textbf{q} = \textbf{value} \ \textbf{depends} \ \textbf{on condition}, \ \textbf{-} = \textbf{unimplemented locations read as '0'}.$

Shaded locations are unimplemented and read as '0'

Note 1: These registers can be addressed from either bank.

^{2:} The upper byte of the Program Counter (PC) is not directly accessible. PCLATH is a holding register for the PC whose contents are transferred to the upper byte of the program counter. (PC<12:8>)

^{3:} Other (non power-up) resets include external reset through MCLR and the Watchdog Timer Reset.

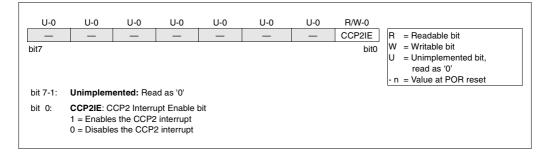
^{4:} The IRP and RP1 bits are reserved on the PIC16C61, always maintain these bits clear.

4.2.2.6 PIE2 REGISTER

Applicable Devices
61 62 62A R62 63 R63 64 64A R64 65 65A R65 66 67

This register contains the CCP2 interrupt enable bit.

FIGURE 4-20: PIE2 REGISTER (ADDRESS 8Dh)



12.1 USART Baud Rate Generator (BRG)

Αp	pli	cable	e Dev	vice	es								
61	62	62A	R62	63	R63	64	64A	R64	65	65A	R65	66	67

The BRG supports both the Asynchronous and Synchronous modes of the USART. It is a dedicated 8-bit baud rate generator. The SPBRG register controls the period of a free running 8-bit timer. In asynchronous mode bit BRGH (TXSTA<2>) also controls the baud rate. In synchronous mode bit BRGH is ignored. Table 12-1 shows the formula for computation of the baud rate for different USART modes which only apply in master mode (internal clock).

Given the desired baud rate and Fosc, the nearest integer value for the SPBRG register can be calculated using the formula in Table 12-1. From this, the error in baud rate can be determined.

Example 12-1 shows the calculation of the baud rate error for the following conditions:

Fosc = 16 MHz Desired Baud Rate = 9600 BRGH = 0 SYNC = 0

EXAMPLE 12-1: CALCULATING BAUD RATE FRROR

Desired Baud rate = Fosc / (64 (X + 1))

9600 = 16000000 / (64 (X + 1)) $X = \lfloor 25.042 \rfloor = 25$

Calculated Baud Rate=16000000 / (64 (25 + 1))

= 9615

Error = (Calculated Baud Rate - Desired Baud Rate)

Desired Baud Rate

= (9615 - 9600) / 9600

= 0.16%

It may be advantageous to use the high baud rate (BRGH = 1) even for slower baud clocks. This is because the FOSC/(16(X+1)) equation can reduce the baud rate error in some cases.

Note: For the PIC16C63/R63/65/65A/R65 the asynchronous high speed mode (BRGH = 1) may experience a high rate of receive errors. It is recommended that BRGH = 0. If you desire a higher baud rate than BRGH = 0 can support, refer to the device errata for additional information or use the PIC16C66/67.

Writing a new value to the SPBRG register, causes the BRG timer to be reset (or cleared), this ensures that the BRG does not wait for a timer overflow before outputting the new baud rate.

TABLE 12-1: BAUD RATE FORMULA

SYNC	BRGH = 0 (Low Speed)	BRGH = 1 (High Speed)
0	(Asynchronous) Baud Rate = Fosc/(64(X+1))	Baud Rate = Fosc/(16(X+1))
1	(Synchronous) Baud Rate = Fosc/(4(X+1))	N/A

X = value in SPBRG (0 to 255)

TABLE 12-2: REGISTERS ASSOCIATED WITH BAUD RATE GENERATOR

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
98h	TXSTA	CSRC	TX9	TXEN	SYNC	-	BRGH	TRMT	TX9D	0000 -010	0000 -010
18h	RCSTA	SPEN	RX9	SREN	CREN	_	FERR	OERR	RX9D	0000 -00x	0000 -00x
99h	SPBRG	Baud Rat	aud Rate Generator Register								0000 0000

Legend: x = unknown, - = unimplemented read as '0'. Shaded cells are not used by the BRG.

13.6 Context Saving During Interrupts

Applicable Devices

61 62 62A R62 63 R63 64 64A R64 65 65A R65 66 67

During an interrupt, only the return PC value is saved on the stack. Typically, users may wish to save key registers during an interrupt i.e., W register and STATUS register. This will have to be implemented in software.

Example 13-1 stores and restores the STATUS and W registers. Example 13-2 stores and restores the STATUS, W, and PCLATH registers (Devices with paged program memory). For all PIC16C6X devices with greater than 1K of program memory (all devices except PIC16C61), the register, W_TEMP, must be

defined in banks and must be defined at the same offset from the bank base address (i.e., if W_TEMP is defined at 0x20 in bank 0, it must also be defined at 0xA0 in bank 1. 0x120 in bank 2. and 0x1A0 in bank 3).

The examples:

- a) Stores the W register
- b) Stores the STATUS register in bank 0
- c) Stores PCLATH
- d) Executes ISR code
- e) Restores PCLATH
- f) Restores STATUS register (and bank select bit)
- g) Restores W register

EXAMPLE 13-1: SAVING STATUS AND W REGISTERS IN RAM (PIC16C61)

```
MOVWF
         W TEMP
                           ;Copy W to TEMP register, could be bank one or zero
SWAPF
         STATUS, W
                           ;Swap status to be saved into W
MOVWE
         STATUS TEMP
                           ; Save status to bank zero STATUS TEMP register
: (ISR)
SWAPF
         STATUS TEMP, W
                           ;Swap STATUS TEMP register into W
                           ; (sets bank to original state)
MOVWF
         STATUS
                           ; Move W into STATUS register
SWAPF
         W TEMP, F
                           ;Swap W TEMP
                           ;Swap W TEMP into W
SWAPF
         W TEMP, W
```

EXAMPLE 13-2: SAVING STATUS, W, AND PCLATH REGISTERS IN RAM (ALL OTHER PIC16C6X DEVICES)

```
;Copy W to TEMP register, could be bank one or zero
MOVWF
        W TEMP
SWAPF
        STATUS, W
                          ;Swap status to be saved into W
CLRF
        STATUS
                         ;bank 0, regardless of current bank, Clears IRP, RP1, RP0
MOVWF
        STATUS TEMP
                         ; Save status to bank zero STATUS TEMP register
        PCLATH, W
MOVE
                         ;Only required if using pages 1, 2 and/or 3
                         ;Save PCLATH into W
      PCLATH TEMP
MOVWF
CLRF
       PCLATH
                         ; Page zero, regardless of current page
BCF
       STATUS, IRP
                         ;Return to Bank 0
       FSR, W
                         ;Copy FSR to W
MOVWF
        FSR TEMP
                         ;Copy FSR from W to FSR TEMP
· (TSR)
MOVF
        PCLATH TEMP, W
                         ;Restore PCLATH
MOVWF
        PCLATH
                          ; Move W into PCLATH
SWAPF
        STATUS TEMP, W
                         ;Swap STATUS TEMP register into W
                          : (sets bank to original state)
MOVWF
        STATUS
                         ; Move W into STATUS register
SWAPF
        W TEMP,F
                         ;Swap W TEMP
        W_TEMP,W
SWAPF
                         ;Swap W TEMP into W
```

13.8 Power-down Mode (SLEEP)

Applicable Devices

61 62 62A R62 63 R63 64 64A R64 65 65A R65 66 67

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

If enabled, the Watchdog Timer will be cleared but keeps running, status bit \overline{PD} (STATUS<3>) is cleared, status bit \overline{TO} (STATUS<4>) 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 VDD, or Vss, ensure no external circuitry is drawing current from the I/O pin, and 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 TOCKI input should also be at VDD or Vss for lowest current consumption. The contribution from on-chip pull-ups on PORTB should be considered

The $\overline{\text{MCLR}}/\text{VPP}$ pin must be at a logic high level (VIHMC).

13.8.1 WAKE-UP FROM SLEEP

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

- External reset input on MCLR/VPP pin.
- Watchdog Timer Wake-up (if WDT was enabled).
- Interrupt from RB0/INT pin, RB port change, or some peripheral interrupts.

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

The following peripheral interrupts can wake the device from SLEEP:

- 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/I²C).
- 4. CCP capture mode interrupt.
- 5. Parallel Slave Port read or write.
- 6. USART TX or RX (synchronous slave mode).

Other peripherals can not 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.

13.8.2 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 TO bit will not be set and 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 TO bit will be set and the 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

14.1 <u>Instruction Descriptions</u>

ADDLW	Add Literal and W	ANDLW	AND Literal with W
Syntax:	[<i>label</i>] ADDLW k	Syntax:	[label] ANDLW k
Operands:	$0 \leq k \leq 255$	Operands:	$0 \leq k \leq 255$
Operation:	$(W) + k \rightarrow (W)$	Operation:	(W) .AND. (k) \rightarrow (W)
Status Affected:	C, DC, Z	Status Affected:	Z
Encoding:	11 111x kkkk kkkk	Encoding:	11 1001 kkkk kkkk
Description:	The contents of the W register are added to the eight bit literal 'k' and the result is placed in the W register.	Description:	The contents of W register are AND'ed with the eight bit literal 'k'. The result is placed in the W register.
Words:	1	Words:	1
Cycles:	1	Cycles:	1
Q Cycle Activity:	Q1 Q2 Q3 Q4	Q Cycle Activity:	Q1 Q2 Q3 Q4
	Decode Read Process Write to data W		Decode Read Process Write to data W
Example:	ADDLW 0x15	Example	ANDLW 0x5F
	Before Instruction W = 0x10 After Instruction W = 0x25		Before Instruction W = 0xA3 After Instruction W = 0x03

ADDWF	Add W and f	ANDWF	AND W with f
Syntax:	[label] ADDWF f,d	Syntax:	[label] ANDWF f,d
Operands:	$0 \le f \le 127$ $d \in [0,1]$	Operands:	$0 \le f \le 127$ $d \in [0,1]$
Operation:	(W) + (f) \rightarrow (destination)	Operation:	(W) .AND. (f) \rightarrow (destination)
Status Affected:	C, DC, Z	Status Affected:	Z
Encoding:	00 0111 dfff ffff	Encoding:	00 0101 dfff ffff
Description:	Add the contents of the W register with 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'.	Description:	AND the W register with 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	Words:	1
Cycles:	1	Cycles:	1
Q Cycle Activity:	Q1 Q2 Q3 Q4	Q Cycle Activity:	Q1 Q2 Q3 Q4
	Decode Read register data Write to destination		Decode Read register data Write to destination
Example	ADDWF FSR, 0	Example	ANDWF FSR, 1
	Before Instruction W = 0x17 FSR = 0xC2 After Instruction W = 0xD9 FSR = 0xC2		Before Instruction W = 0x17 FSR = 0xC2 After Instruction W = 0x17 FSR = 0x02

15.5 <u>Timing Diagrams and Specifications</u>

FIGURE 15-2: EXTERNAL CLOCK TIMING

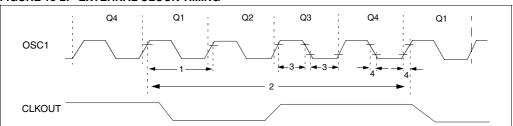


TABLE 15-2: EXTERNAL CLOCK TIMING REQUIREMENTS

Parameter No.	Sym	Characteristic	Min	Тур†	Max	Units	Conditions
	Fosc	External CLKIN Frequency	DC	_	4	MHz	XT and RC osc mode
		(Note 1)	DC	_	4	MHz	HS osc mode (-04)
			DC	_	20	MHz	HS osc mode (-20)
			DC	_	200	kHz	LP osc mode
		Oscillator Frequency	DC	_	4	MHz	RC osc mode
		(Note 1)	0.1	_	4	MHz	XT osc mode
			1	_	4	MHz	HS osc mode (-04)
			1		20	MHz	HS osc mode (-20)
1	Tosc	External CLKIN Period	250	_	_	ns	XT and RC osc mode
		(Note 1)	250	_	_	ns	HS osc mode (-04)
			50	_	_	ns	HS osc mode (-20)
			5	_	_	μS	LP osc mode
		Oscillator Period	250	_	_	ns	RC osc mode
		(Note 1)	250	_	10,000	ns	XT osc mode
			250	_	1,000	ns	HS osc mode (-04)
			50	_	1,000	ns	HS osc mode (-20)
			5		_	μS	LP osc mode
2	Tcy	Instruction Cycle Time (Note 1)	1.0	Tcy	DC	μS	Tcy = 4/Fosc
3	TosL,	External Clock in (OSC1) High or	50	_	_	ns	XT oscillator
	TosH	Low Time	2.5	_	_	μS	LP oscillator
			10		_	ns	HS oscillator
4	TosR,	External Clock in (OSC1) Rise or	25	_	_	ns	XT oscillator
	TosF	Fall Time	50		_	ns	LP oscillator
			15	_	_	ns	HS oscillator

[†] 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 17-4: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER AND POWER-UP TIMER TIMING

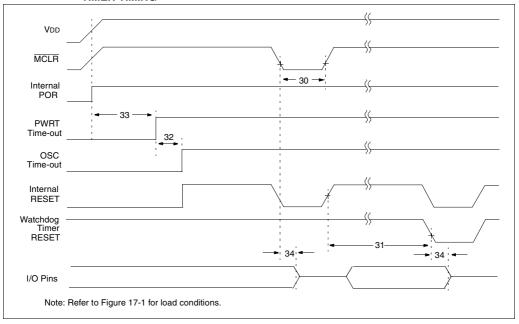


TABLE 17-4: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER AND POWER-UP TIMER REQUIREMENTS

Parameter No.	Sym	Characteristic	Min	Тур†	Max	Units	Conditions
30*	TmcL	MCLR Pulse Width (low)	100	_	_	ns	VDD = 5V, -40°C to +85°C
31*	Twdt	Watchdog Timer Time-out Period (No Prescaler)	7	18	33	ms	VDD = 5V, -40°C to +85°C
32	Tost	Oscillation Start-up Timer Period	_	1024Tosc	_	_	Tosc = OSC1 period
33*	Tpwrt	Power-up Timer Period	28	72	132	ms	VDD = 5V, -40°C to +85°C
34*	Tıoz	I/O Hi-impedance from MCLR Low	_	_	100	ns	

^{*} 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.

FIGURE 18-3: CLKOUT AND I/O TIMING

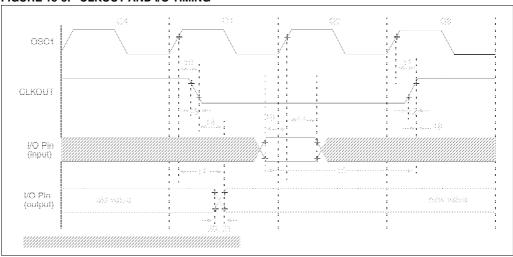


TABLE 18-3: CLKOUT AND I/O TIMING REQUIREMENTS

Parameters	Sym	Characteristic		Min	Typ†	Max	Units	Conditions
10*	TosH2ckL	OSC1↑ to CLKOUT↓		_	75	200	ns	Note 1
11*	TosH2ckH	OSC1↑ to CLKOUT↑	_	75	200	ns	Note 1	
12*	TckR	CLKOUT rise time		_	35	100	ns	Note 1
13*	TckF	CLKOUT fall time		_	35	100	ns	Note 1
14*	TckL2ioV	CLKOUT ↓ to Port out valid		_	_	0.5Tcy + 20	ns	Note 1
15*	TioV2ckH	Port in valid before CLKOUT ↑		Tosc + 200	_	_	ns	Note 1
16*	TckH2ioI	Port in hold after CLKOUT ↑		0	-		ns	Note 1
17*	TosH2ioV	OSC1↑ (Q1 cycle) to Port out va	lid	_	50	150	ns	
18*	TosH2ioI	OSC1 [↑] (Q2 cycle) to Port input invalid (I/O in hold time)	PIC16 C 62A/ R62/64A/R64	100			ns	
			PIC16 LC 62A/ R62/64A/R64	200	_	_	ns	
19*	TioV2osH	Port input valid to OSC1 [↑] (I/O in	setup time)	0	_	_	ns	
20*	TioR	Port output rise time	PIC16 C 62A/ R62/64A/R64	_	10	40	ns	
			PIC16 LC 62A/ R62/64A/R64	_	_	80	ns	
21*	TioF	Port output fall time	PIC16 C 62A/ R62/64A/R64	_	10	40	ns	
			PIC16 LC 62A/ R62/64A/R64	_	-	80	ns	
22††*	Tinp	RB0/INT pin high or low time	Tcy	_	_	ns		
23††*	Trbp	RB7:RB4 change int high or low	time	Tcy	_	_	ns	

^{*} 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.

^{††} These parameters are asynchronous events not related to any internal clock edge.

Note 1: Measurements are taken in RC Mode where CLKOUT output is 4 x TOSC.

19.2 DC Characteristics: PIC16LC65-04 (Commercial, Industrial)

DC CH		Standar Operatir	•	•		°C ≤	Inless otherwise stated) TA ≤ +85°C for industrial and TA ≤ +70°C for commercial
Param No.	Characteristic	Sym	Min	Тур†	Max	Units	Conditions
D001	Supply Voltage	VDD	3.0	-	6.0	٧	LP, XT, RC osc configuration (DC - 4 MHz)
D002*	RAM Data Retention Voltage (Note 1)	VDR	-	1.5	-	V	
D003	VDD start voltage to ensure internal Power-on Reset signal	VPOR	-	Vss	-	V	See section on Power-on Reset for details
D004*	VDD rise rate to ensure internal Power-on Reset signal	SVDD	0.05	-	-	V/ms	See section on Power-on Reset for details
D010	Supply Current (Note 2, 5)	IDD	-	2.0	3.8	mA	XT, RC osc configuration FOSC = 4 MHz, VDD = 3.0V (Note 4)
D010A			-	22.5	105	μА	LP osc configuration FOSC = 32 kHz, VDD = 4.0V, WDT disabled
D020	Power-down Current	IPD	-	7.5	800	μА	VDD = 3.0V, WDT enabled, -40°C to +85°C
D021	(Note 3, 5)		-	0.9	800	μA	VDD = 3.0V, WDT disabled, 0°C to +70°C
D021A			-	0.9	800	μΑ	VDD = 3.0V, WDT disabled, -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: This is the limit to which VDD can be lowered without losing RAM data.
 - 2: The supply current is mainly a function of the operating voltage and frequency. Other factors such as I/O pin loading and switching rate, oscillator type, internal code execution pattern, and temperature also have an impact on the current consumption.
 - The test conditions for all IDD measurements in active operation mode are:
 - OSC1 = external square wave, from rail to rail; all I/O pins tristated, pulled to VDD,
 - MCLR = VDD; WDT enabled/disabled as specified.
 - 3: The power down current in SLEEP mode does not depend on the oscillator type. Power-down current is measured with the part in SLEEP mode, with all I/O pins in hi-impedance state and tied to VDD and Vss.
 - 4: For RC osc configuration, current through Rext is not included. The current through the resistor can be estimated by the formula Ir = VDD/2Rext (mA) with Rext in kOhm.
 - 5: Timer1 oscillator (when enabled) adds approximately 20 μA to the specification. This value is from characterization and is for design guidance only. This is not tested.

Applicable Devices 61 62 62A R62 63 R63 64 64A R64 65 65A R65 66 67

19.4 **Timing Parameter Symbology**

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

1. TppS2	ppS	3. Tcc:st	(I ² C specifications only)
2. TppS		4. Ts	(I ² C specifications only)
T			
F	Frequency	Т	Time
Lowerd	case letters (pp) and their meanings:		
pp			
СС	CCP1	osc	OSC1
ck	CLKOUT	rd	RD
cs	CS	rw	RD or WR
di	SDI	sc	SCK
do	SDO	SS	SS
dt	Data in	t0	T0CKI
io	I/O port	t1	T1CKI
mc	MCLR	wr	WR
Upperd	case letters and their meanings:	•	
S			
F	Fall	Р	Period
Н	High	R	Rise
I	Invalid (Hi-impedance)	V	Valid
L	Low	Z	Hi-impedance
I ² C only			

Bus free Tcc:st (I²C specifications only)

output access

AA

BUF

CC			
HD	Hold	SU	Setup
ST			
DAT	DATA input hold	STO	STOP condition
STA	START condition		

High

Low

High

Low

FIGURE 19-1: LOAD CONDITIONS FOR DEVICE TIMING SPECIFICATIONS

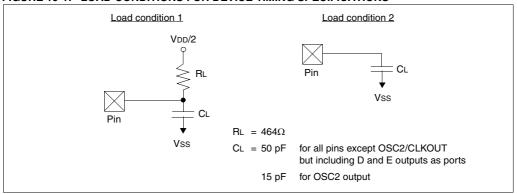


FIGURE 20-7: CAPTURE/COMPARE/PWM TIMINGS (CCP1 AND CCP2)

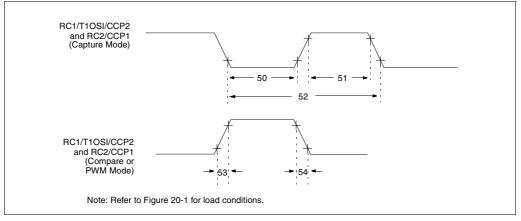


TABLE 20-6: CAPTURE/COMPARE/PWM REQUIREMENTS (CCP1 AND CCP2)

Parameter No.	Sym	Characteristic			Min	Тур†	Max	Units	Conditions	
50*	TccL	CCP1 and CCP2	No Prescaler		0.5Tcy + 20	_	_	ns		
		input low time	With Prescaler	PIC16 C 63/65A	10	_		ns		
				PIC16 LC 63/65A	20	_	-	ns		
51*	TccH		CCP1 and CCP2	No Prescaler		0.5Tcy + 20	_	_	ns	
			input high time	With Prescaler	PIC16 C 63/65A	10	_	_	ns	
				PIC16 LC 63/65A	20	_	_	ns		
52*	TccP	CCP1 and CCP2 in	CCP1 and CCP2 input period		3Tcy + 40 N	_	-	ns	N = prescale value (1,4, or 16)	
53*	TccR	CCP1 and CCP2 o	utput rise time	PIC16 C 63/65A	_	10	25	ns		
				PIC16 LC 63/65A	_	25	45	ns		
54*	TccF	CCP1 and CCP2 o	utput fall time	PIC16 C 63/65A	_	10	25	ns		
				PIC16 LC 63/65A	_	25	45	ns		

^{*} 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.

do

dt

io

Applicable Devices 61 62 62A R62 63 R63 64 64A R64 65 65A R65 66 67

22.4 <u>Timing Parameter Symbology</u>

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

1. TppS2	2ppS	3. Tcc:st	(I ² C specifications only)	
2. TppS		4. Ts	(I ² C specifications only)	
Т				
F	Frequency	Т	Time	
Lower	case letters (pp) and their meanings:			
рр				
СС	CCP1	osc	OSC1	
ck	CLKOUT	rd	RD	
cs	CS	rw	RD or WR	
di	SDI	sc	SCK	

SS

t0

t1

wr

SS

T0CKI

T1CKI

WR

Uppercase letters and their meanings:

SDO

Data in

I/O port

MCLR

S			
F	Fall	Р	Period
Н	High	R	Rise
1	Invalid (Hi-impedance)	V	Valid
L	Low	Z	Hi-impedance
I ² C only			
AA	output access	High	High
BUF	Bus free	Low	Low

Tcc:st (I²C specifications only)

CC			
HD	Hold	SU	Setup
ST			
DAT	DATA input hold	STO	STOP condition
STA	START condition		

FIGURE 22-1: LOAD CONDITIONS FOR DEVICE TIMING SPECIFICATIONS

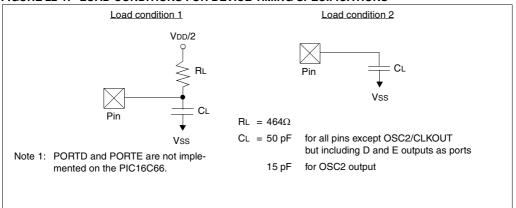


FIGURE 23-8: TYPICAL IPD vs. VDD BROWN-OUT DETECT ENABLED (RC MODE)

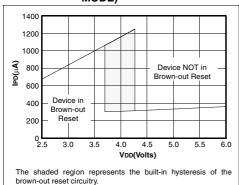


FIGURE 23-9: MAXIMUM IPD vs. VDD BROWN-OUT DETECT ENABLED (85°C TO -40°C, RC MODE)

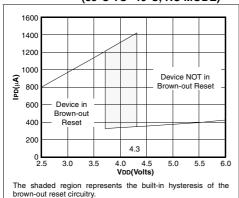


FIGURE 23-10: TYPICAL IPD vs. TIMER1 ENABLED (32 kHz, RC0/RC1 = 33 pF/33 pF, RC MODE)

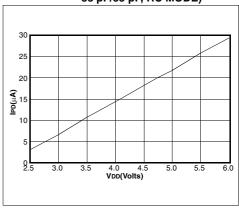
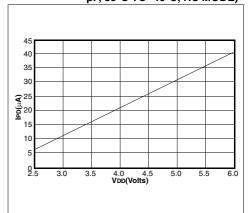
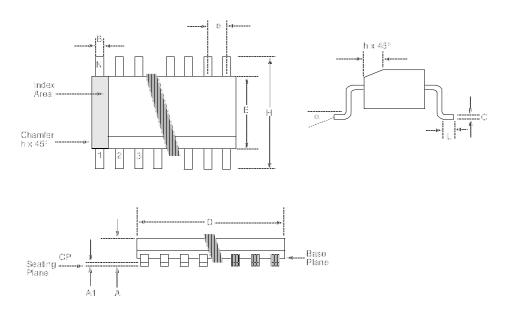


FIGURE 23-11: MAXIMUM IPD vs. TIMER1 ENABLED (32 kHz, RC0/RC1 = 33 pF/33 pF, 85°C TO -40°C, RC MODE)



24.4 18-Lead Plastic Surface Mount (SOIC - Wide, 300 mil Body) (SO)

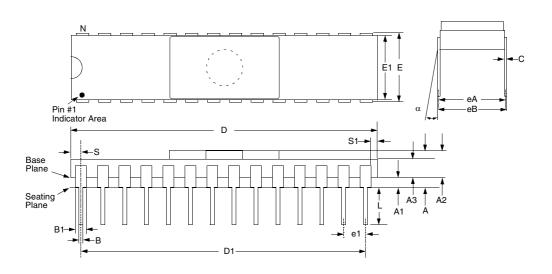
Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



	Package Group: Plastic SOIC (SO)						
		Millimeters			Inches		
Symbol	Min	Max	Notes	Min	Max	Notes	
α	0°	8°		0°	8°		
Α	2.362	2.642		0.093	0.104		
A1	0.101	0.300		0.004	0.012		
В	0.355	0.483		0.014	0.019		
С	0.241	0.318		0.009	0.013		
D	11.353	11.735		0.447	0.462		
E	7.416	7.595		0.292	0.299		
е	1.270	1.270	Reference	0.050	0.050	Reference	
Н	10.007	10.643		0.394	0.419		
h	0.381	0.762		0.015	0.030		
L	0.406	1.143		0.016	0.045		
N	18	18		18	18		
CP	_	0.102		_	0.004		

24.9 28-Lead Ceramic Side Brazed Dual In-Line with Window (300 mil) (JW)

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

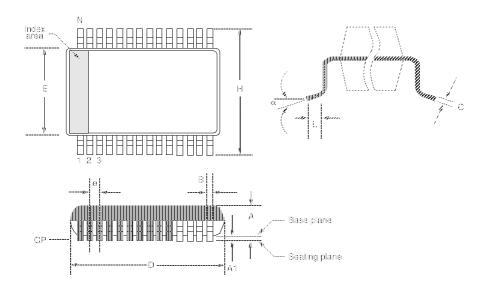


Package Group: Ceramic Side Brazed Dual In-Line (CER)							
0		Millimeters			Inches		
Symbol	Min	Max	Notes	Min	Max	Notes	
α	0°	10°		0°	10°		
Α	3.937	5.030		0.155	0.198		
A1	1.016	1.524		0.040	0.060		
A2	2.921	3.506		0.115	0.138		
A3	1.930	2.388		0.076	0.094		
В	0.406	0.508		0.016	0.020		
B1	1.219	1.321	Typical	0.048	0.052		
С	0.228	0.305	Typical	0.009	0.012		
D	35.204	35.916		1.386	1.414		
D1	32.893	33.147	Reference	1.295	1.305		
E	7.620	8.128		0.300	0.320		
E1	7.366	7.620		0.290	0.300		
e1	2.413	2.667	Typical	0.095	0.105		
eA	7.366	7.874	Reference	0.290	0.310		
eB	7.594	8.179		0.299	0.322		
L	3.302	4.064		0.130	0.160		
N	28	28		28	28		
S	1.143	1.397		0.045	0.055		
S1	0.533	0.737		0.021	0.029		

Mote:

24.10 28-Lead Plastic Surface Mount (SSOP - 209 mil Body 5.30 mm) (SS)

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



	Package Group: Plastic SSOP							
	Millimeters				Inches			
Symbol	Min	Max	Notes	Min	Max	Notes		
α	0°	8°		0°	8°			
Α	1.730	1.990		0.068	0.078			
A1	0.050	0.210		0.002	0.008			
В	0.250	0.380		0.010	0.015			
С	0.130	0.220		0.005	0.009			
D	10.070	10.330		0.396	0.407			
E	5.200	5.380		0.205	0.212			
е	0.650	0.650	Reference	0.026	0.026	Reference		
Н	7.650	7.900		0.301	0.311			
L	0.550	0.950		0.022	0.037			
N	28	28		28	28			
CP	-	0.102		-	0.004			

	I ² C Bus Start/Stop Bits244
	Oscillator Start-up Timer
	Parallel Slave Port242
	Power-up Timer
	Reset
	SPI Mode243
	Timer0240
	Timer1240
	USART Synchronous Receive
	(Master/Slave)246
	Watchdog Timer239
PIC	16C66
1 10	
	Brown-out Reset
	Capture/Compare/PWM273
	CLKOUT and I/O270
	External Clock269
	I ² C Bus Data279
	I ² C Bus Start/Stop Bits278
	Oscillator Start-up Timer
	•
	Power-up Timer271
	Reset271
	Timer0272
	Timer1272
	USART Synchronous Receive
	(Master/Slave)
B10	Watchdog Timer271
PIC	16C67
	Brown-out Reset271
	Capture/Compare/PWM273
	CLKOUT and I/O270
	External Clock
	I ² C Bus Data
	I ² C Bus Start/Stop Bits
	Oscillator Start-up Timer271
	Parallel Slave Port274
	Power-up Timer271
	Power-up Timer
	•
	Reset
	Reset 271 Timer0 272 Timer1 272
	Reset 271 Timer0 272 Timer1 272 USART Synchronous Receive
	Reset 271 Timer0 272 Timer1 272 USART Synchronous Receive (Master/Slave) 280 280
	Reset 271 Timer0 272 Timer1 272 USART Synchronous Receive (Master/Slave) 280 Watchdog Timer 271
PIC	Reset 271 Timer0 272 Timer1 272 USART Synchronous Receive (Master/Slave) 280 Watchdog Timer 271 16CR62
PIC	Reset 271 Timer0 272 Timer1 272 USART Synchronous Receive (Master/Slave) 280 Watchdog Timer 271
PIC	Reset 271 Timer0 272 Timer1 272 USART Synchronous Receive (Master/Slave) 280 Watchdog Timer 271 16CR62 Capture/Compare/PWM 209
PIC	Reset 271 Timer0 272 Timer1 272 USART Synchronous Receive (Master/Slave) (Master/Slave) 280 Watchdog Timer 271 16CR62 2 Capture/Compare/PWM 209 CLKOUT and I/O 206
PIC	Reset 271 Timer0 272 Timer1 272 USART Synchronous Receive (Master/Slave) Watchdog Timer 271 16CR62 209 Capture/Compare/PWM 209 CLKOUT and I/O 206 External Clock 205
PIC	Reset 271 Timer0 272 Timer1 272 USART Synchronous Receive (Master/Slave) Watchdog Timer 271 16CR62 209 CLKOUT and I/O 206 External Clock 205 I²C Bus Data 213
PIC	Reset 271 Timer0 272 Timer1 272 USART Synchronous Receive (Master/Slave) 280 Watchdog Timer 271 16CR62 26 Capture/Compare/PWM 209 CLKOUT and I/O 206 External Clock 205 I ² C Bus Data 213 I ² C Bus Start/Stop Bits 212
PIC	Reset 271 Timer0 272 Timer1 272 USART Synchronous Receive (Master/Slave) (Master/Slave) 280 Watchdog Timer 271 16CR62 20 Capture/Compare/PWM 209 CLKOUT and I/O 206 External Clock 205 I ² C Bus Data 213 I ² C Bus Start/Stop Bits 212 Oscillator Start-up Timer 207
PIC	Reset 271 Timer0 272 Timer1 272 USART Synchronous Receive (Master/Slave) Watchdog Timer 271 16CR62 20 Capture/Compare/PWM 209 CLKOUT and I/O 206 External Clock 205 1 ² C Bus Data 213 1 ² C Bus Start/Stop Bits 212 Oscillator Start-up Timer 207 Power-up Timer 207
PIC	Reset 271 Timer0 272 Timer1 272 USART Synchronous Receive (Master/Slave) Watchdog Timer 271 16CR62 20 Capture/Compare/PWM 209 CLKOUT and I/O 206 External Clock 205 I²C Bus Data 213 I²C Bus Start/Stop Bits 212 Oscillator Start-up Timer 207 Power-up Timer 207 Reset 207
PIC	Reset 271 Timer0 272 Timer1 272 USART Synchronous Receive (Master/Slave) Watchdog Timer 271 16CR62 20 Capture/Compare/PWM 209 CLKOUT and I/O 206 External Clock 205 1 ² C Bus Data 213 1 ² C Bus Start/Stop Bits 212 Oscillator Start-up Timer 207 Power-up Timer 207
PIC	Reset 271 Timer0 272 Timer1 272 USART Synchronous Receive (Master/Slave) Watchdog Timer 271 16CR62 20 Capture/Compare/PWM 209 CLKOUT and I/O 206 External Clock 205 I²C Bus Data 213 I²C Bus Start/Stop Bits 212 Oscillator Start-up Timer 207 Power-up Timer 207 Reset 207
PIC	Reset 271 Timer0 272 Timer1 272 USART Synchronous Receive (Master/Slave) (Master/Slave) 280 Watchdog Timer 271 16CR62 20 Capture/Compare/PWM 209 CLKOUT and I/O 206 External Clock 205 I²C Bus Data 213 I²C Bus Start/Stop Bits 212 Oscillator Start-up Timer 207 Power-up Timer 207 Reset 207 SPI Mode 211 Timer0 208
PIC	Reset 271 Timer0 272 Timer1 272 USART Synchronous Receive (Master/Slave) (Master/Slave) 280 Watchdog Timer 271 16CR62 2 Capture/Compare/PWM 209 CLKOUT and I/O 206 External Clock 205 I²C Bus Data 213 I²C Bus Start/Stop Bits 212 Oscillator Start-up Timer 207 Power-up Timer 207 Reset 207 SPI Mode 211 Timer0 208 Timer1 208
PIC	Reset 271 Timer0 272 Timer1 272 USART Synchronous Receive (Master/Slave) (Master/Slave) 280 Watchdog Timer 271 16CR62 20 Capture/Compare/PWM 209 CLKOUT and I/O 206 External Clock 205 I²C Bus Data 213 I²C Bus Start/Stop Bits 212 Oscillator Start-up Timer 207 Power-up Timer 207 Reset 207 SPI Mode 211 Timer0 208

PIC16CR63	
Brown-out Reset	255
Capture/Compare/PWM	
CLKOUT and I/O	254
External Clock	
I ² C Bus Data	
I ² C Bus Start/Stop Bits	
Oscillator Start-up Timer	
Power-up Timer	
Reset	
SPI Mode	
Timer0 Timer1	
USART Synchronous Receive	250
(Master/Slave)	262
Watchdog Timer	
PIC16CR64	200
Capture/Compare/PWM	209
CLKOUT and I/O	
External Clock	
I ² C Bus Data	
I ² C Bus Start/Stop Bits	
Oscillator Start-up Timer	
Parallel Slave Port	
Power-up Timer	
Reset	
SPI Mode	
Timer0	
Timer1	208
Watchdog Timer	
PIC16CR65	
Brown-out Reset	255
Capture/Compare/PWM	257
CLKOUT and I/O	254
External Clock	253
I ² C Bus Data	261
I ² C Bus Start/Stop Bits	
Oscillator Start-up Timer	
Parallel Slave Port	
Power-up Timer	
Reset	
SPI Mode	
Timer0	
Timer1	256
USART Synchronous Receive	
(Master/Slave)	
Watchdog Timer	
Power-up Timer	
PWM Output RB0/INT Interrupt	
RX Pin Sampling	
SPI Master Mode	
SPI Mode, Master/Slave Mode,	93
No SS Control	QΩ
SPI Mode, Slave Mode With SS Control	ου
SPI Slave Mode (CKE = 1)	
SPI Slave Mode Timing (CKE = 0)	
Timer0 with External Clock	
TMR0 Interrupt Timing	
USART Asynchronous Master Transmission	
USART Asynchronous Master Transmission	
(Back to Back)	113
USART Asynchronous Reception	
USART Synchronous Reception in	
Master Mode	119
USART Synchronous Tranmission	
Wake-up from SLEEP Through Interrupts	142

LIST OF	EQUATION AND EXAMPLES	Figure 4-15:	PIE1 Register for PIC16C65/65A/R65/67 (Address 8Ch)	40
Example 3-1:	Instruction Pipeline Flow18	Figure 4-16:	PIR1 Register for PIC16C62/62A/R62	. 40
	Call of a Subroutine in Page 1		(Address 0Ch)	. 41
	from Page 049	Figure 4-17:	PIR1 Register for PIC16C63/R63/66	
Example 4-2:	Indirect Addressing49		Address 0Ch)	. 42
Example 5-1:	Initializing PORTA51	Figure 4-18:	PIR1 Register for PIC16C64/64A/R64	
Example 5-2:	Initializing PORTB53		(Address 0Ch)	. 43
Example 5-3:	Initializing PORTC55	Figure 4-19:	PIR1 Register for PIC16C65/65A/R65/67	
Example 5-4:	Read-Modify-Write Instructions on an		(Address 0Ch)	
	I/O Port60	Figure 4-20:	PIE2 Register (Address 8Dh)	
	Changing Prescaler (Timer0→WDT)69	Figure 4-21:	PIR2 Register (Address 0Dh)	. 46
	Changing Prescaler (WDT→Timer0)69	Figure 4-22:	PCON Register for PIC16C62/64/65	4-7
Example 8-1:	Reading a 16-bit	Figure 4 00	(Address 8Eh)	. 47
F	Free-running Timer	Figure 4-23:	PCON Register for PIC16C62A/R62/63/	
Example 10-1:	Changing Between		R63/64A/R64/65A/R65/66/67 (Address 8Eh)	17
Evample 10 2	Capture Prescalers79 PWM Period and Duty	Figure 4-24:	Loading of PC in Different Situations	
Example 10-2.	Cycle Calculation81	Figure 4-25:	Direct/Indirect Addressing	
Evample 11-1:	Loading the SSPBUF	Figure 5-1:	Block Diagram of the	. 40
Lxample 11-1.	(SSPSR) Register86	rigaro o r.	RA3:RA0 Pins and the RA5 Pin	51
Example 11-2	Loading the SSPBUF	Figure 5-2:	Block Diagram of the RA4/T0CKI Pin	
Example 11 E.	(SSPSR) Register (PIC16C66/67)91	Figure 5-3:	Block Diagram of the	
Example 12-1:	Calculating Baud Rate Error107	3	RB7:RB4 Pins for PIC16C61/62/64/65	. 53
	Saving Status and W	Figure 5-4:	Block Diagram of the	
	Registers in RAM139	· ·	RB7:RB4 Pins for PIC16C62A/63/R63/	
Example 13-2:	Saving Status, W, and		64A/65A/R65/66/67	. 54
•	PCLATH Registers in RAM	Figure 5-5:	Block Diagram of the	
	(All other PIC16C6X devices)139		RB3:RB0 Pins	. 54
		Figure 5-6:	PORTC Block Diagram	. 55
		Figure 5-7:	PORTD Block Diagram	
LIST OF	FIGURES		(In I/O Port Mode)	. 57
		Figure 5-8:	PORTE Block Diagram	
Figure 3-1:	PIC16C61 Block Diagram10	F: 50	(In I/O Port Mode)	
Figure 3-2:	PIC16C62/62A/R62/64/64A/R64	Figure 5-9:	TRISE Register (Address 89h)	
Fig 0. 0.	Block Diagram11	Figure 5-10:	Successive I/O Operation	. 60
Figure 3-3:	PIC16C63/R63/65/65A/R65	Figure 5-11:	PORTD and PORTE as a Parallel	61
Figure 3-4:	Block Diagram	Figure 5-12:	Slave Port Parallel Slave Port Write Waveforms	
Figure 3-4:	Clock/Instruction Cycle18	Figure 5-12:	Parallel Slave Port Read Waveforms	
Figure 4-1:	PIC16C61 Program Memory Map	Figure 7-1:	Timer0 Block Diagram	
94.0	and Stack19	Figure 7-2:	Timer0 Timing: Internal Clock/No	
Figure 4-2:	PIC16C62/62A/R62/64/64A/	ga. 0 /	Prescaler	. 65
	R64 Program Memory Map and Stack 19	Figure 7-3:	Timer0 Timing: Internal	
Figure 4-3:	PIC16C63/R63/65/65A/R65 Program	3	Clock/Prescale 1:2	. 66
•	Memory Map and Stack19	Figure 7-4:	TMR0 Interrupt Timing	
Figure 4-4:	PIC16C66/67 Program Memory	Figure 7-5:	Timer0 Timing With External Clock	. 67
	Map and Stack20	Figure 7-6:	Block Diagram of the Timer0/WDT	
Figure 4-5:	PIC16C61 Register File Map20		Prescaler	. 68
Figure 4-6:	PIC16C62/62A/R62/64/64A/	Figure 8-1:	T1CON: Timer1 Control Register	
	R64 Register File Map21		(Address 10h)	
Figure 4-7:	PIC16C63/R63/65/65A/R65	Figure 8-2:	Timer1 Block Diagram	
	Register File Map21	Figure 9-1:	Timer2 Block Diagram	. 75
Figure 4-8:	PIC16C66/67 Data Memory Map22	Figure 9-2:	T2CON: Timer2 Control Register	
Figure 4-9:	STATUS Register	E' 10.1	(Address 12h)	. /5
E' 4.40	(Address 03h, 83h, 103h, 183h)	Figure 10-1:	CCP1CON Register (Address 17h) /	70
Figure 4-10:	OPTION Register	Fig 10.0.	CCP2CON Register (Address 1Dh)	. /8
Ciaura 4 11.	(Address 81h, 181h)	Figure 10-2:	Capture Mode Operation	70
Figure 4-11:	INTCON Register (Address ORb. 8Rb. 10Rb.18Rb) 37	Figure 10 2	Block Diagram	. 78
Figure 4-12:	(Address 0Bh, 8Bh, 10Bh 18Bh)37 PIE1 Register for PIC16C62/62A/R62	Figure 10-3:	Compare Mode Operation Block Diagram	70
1 19u16 4-12.	(Address 8Ch)38	Figure 10-4:	Simplified PWM Block Diagram	
Figure 4-13:	PIE1 Register for PIC16C63/R63/66	Figure 10-4.	PWM Output	
guic 4-10.	(Address 8Ch)39	Figure 10-5.	SSPSTAT: Sync Serial Port Status	. 50
Figure 4-14:	PIE1 Register for PIC16C64/64A/R64	riguio i i i.	Register (Address 94h)	. 84
J	(Address 8Ch)39		- 3 (