



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

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

"[Embedded - Microcontrollers](#)" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

### Applications of "[Embedded - Microcontrollers](#)"

#### Details

Product Status	Active
Core Processor	PIC
Core Size	8-Bit
Speed	4MHz
Connectivity	I <sup>2</sup> C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	33
Program Memory Size	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	-40°C ~ 85°C (TA)
Mounting Type	Through Hole
Package / Case	40-DIP (0.600", 15.24mm)
Supplier Device Package	40-PDIP
Purchase URL	<a href="https://www.e-xfl.com/product-detail/microchip-technology/pic16c67-04i-p">https://www.e-xfl.com/product-detail/microchip-technology/pic16c67-04i-p</a>

**TABLE 4-3: SPECIAL FUNCTION REGISTERS FOR THE PIC16C63/R63**

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 <sup>(3)</sup>
Bank 0											
00h <sup>(1)</sup>	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 <sup>(1)</sup>	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	0000 0000
03h <sup>(1)</sup>	STATUS	IRP <sup>(4)</sup>	RP1 <sup>(4)</sup>	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	000q quuu
04h <sup>(1)</sup>	FSR	Indirect data memory address pointer								xxxx xxxx	uuuu uuuu
05h	PORTA	—	—	PORTA Data Latch when written: PORTA pins when read						- -xx xxxx	- -uu uuuu
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	—	Unimplemented								—	—
09h	—	Unimplemented								—	—
0Ah <sup>(1,2)</sup>	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter					- - - 0 0000	- - - 0 0000
0Bh <sup>(1)</sup>	INTCON	GIE	PEIE	TOIE	INTE	RBIE	TOIF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	(5)	(5)	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/PWM1 (LSB)								xxxx xxxx	uuuu uuuu
16h	CCPR1H	Capture/Compare/PWM1 (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/PWM2 (LSB)								xxxx xxxx	uuuu uuuu
1Ch	CCPR2H	Capture/Compare/PWM2 (MSB)								xxxx xxxx	uuuu uuuu
1Dh	CCP2CON	—	—	CCP2X	CCP2Y	CCP2M3	CCP2M2	CCP2M1	CCP2M0	- - 00 0000	- - 00 0000
1Eh-1Fh	—	Unimplemented								—	—

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

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

- 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>)
- Other (non power-up) resets include external reset through  $\overline{MCLR}$  and the Watchdog Timer reset.
- The IRP and RP1 bits are reserved on the PIC16C63/R63, always maintain these bits clear.
- PIE1<7:6> and PIR1<7:6> are reserved on the PIC16C63/R63, always maintain these bits clear.

# PIC16C6X

**TABLE 4-4: SPECIAL FUNCTION REGISTERS FOR THE PIC16C64/64A/R64**

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 <sup>(3)</sup>
Bank 0											
00h <sup>(1)</sup>	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 uu
02h <sup>(1)</sup>	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	0000 0000
03h <sup>(1)</sup>	STATUS	IRP <sup>(5)</sup>	RP1 <sup>(5)</sup>	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	000q quuu
04h <sup>(1)</sup>	FSR	Indirect data memory address pointer								xxxx xxxx	uuuu uuuu
05h	PORTA	—	—	PORTA Data Latch when written: PORTA pins when read						- -xx xxxx	- -uu uuuu
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	PORTD	PORTD Data Latch when written: PORTD pins when read								xxxx xxxx	uuuu uuuu
09h	PORTE	—	—	—	—	—	RE2	RE1	RE0	---- -xxx	---- -uuu
0Ah <sup>(1,2)</sup>	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter					---0 0000	---0 0000
0Bh <sup>(1)</sup>	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	PSPIF	(6)	—	—	SSPIF	CCP1IF	TMR2IF	TMR1IF	00-- 0000	00-- 0000
0Dh	—	Unimplemented								—	—
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/PWM1 (LSB)								xxxx xxxx	uuuu uuuu
16h	CCPR1H	Capture/Compare/PWM1 (MSB)								xxxx xxxx	uuuu uuuu
17h	CCP1CON	—	—	CCP1X	CCP1Y	CCP1M3	CCP1M2	CCP1M1	CCP1M0	--00 0000	--00 0000
18h-1Fh	—	Unimplemented								—	—

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

Shaded locations are unimplemented, 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 BOR bit is reserved on the PIC16C64, always maintain this bit set.

5: The IRP and RP1 bits are reserved on the PIC16C64/64A/R64, always maintain these bits clear.

6: PIE1<6> and PIR1<6> are reserved on the PIC16C64/64A/R64, always maintain these bits clear.

**TABLE 4-4: SPECIAL FUNCTION REGISTERS FOR THE PIC16C64/64A/R64 (Cont'd)**

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 <sup>(3)</sup>
Bank 1											
80h <sup>(1)</sup>	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	0000 0000
81h	OPTION	RBPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
82h <sup>(1)</sup>	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	0000 0000
83h <sup>(1)</sup>	STATUS	IRP <sup>(5)</sup>	RP1 <sup>(6)</sup>	RP0	T0	PD	Z	DC	C	0001 1xxx	000q quuu
84h <sup>(1)</sup>	FSR	Indirect data memory address pointer								xxxx xxxx	uuuu uuuu
85h	TRISA	—	—	PORTA Data Direction Register						--11 1111	--11 1111
86h	TRISB	PORTB Data Direction Register								1111 1111	1111 1111
87h	TRISC	PORTC Data Direction Register								1111 1111	1111 1111
88h	TRISD	PORTD Data Direction Register								1111 1111	1111 1111
89h	TRISE	IBF	OBF	IBOV	PSPMODE	—	PORTE Data Direction Bits			0000 -111	0000 -111
8Ah <sup>(1,2)</sup>	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter					---0 0000	---0 0000
8Bh <sup>(1)</sup>	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
8Ch	PIE1	PSPIE	(6)	—	—	SSPIE	CCP1IE	TMR2IE	TMR1IE	00-- 0000	00-- 0000
8Dh	—	Unimplemented								—	—
8Eh	PCON	—	—	—	—	—	—	POR	BOR <sup>(4)</sup>	---- -gq	---- -uu
8Fh	—	Unimplemented								—	—
90h	—	Unimplemented								—	—
91h	—	Unimplemented								—	—
92h	PR2	Timer2 Period Register								1111 1111	1111 1111
93h	SSPADD	Synchronous Serial Port (I <sup>2</sup> C mode) Address Register								0000 0000	0000 0000
94h	SSPSTAT	—	—	D/A	P	S	R/W	UA	BF	--00 0000	--00 0000
95h-9Fh	—	Unimplemented								—	—

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

Shaded locations are unimplemented, 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 BOR bit is reserved on the PIC16C64, always maintain this bit set.  
 5: The IRP and RP1 bits are reserved on the PIC16C64/64A/R64, always maintain these bits clear.  
 6: PIE1<6> and PIR1<6> are reserved on the PIC16C64/64A/R64, always maintain these bits clear.

**TABLE 4-6: SPECIAL FUNCTION REGISTERS FOR THE PIC16C66/67 (Cont'd)**

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 <sup>(3)</sup>	
Bank 1												
80h <sup>(1)</sup>	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	0000 0000	
81h	OPTION	RBP $\overline{U}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111	
82h <sup>(1)</sup>	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	0000 0000	
83h <sup>(1)</sup>	STATUS	IRP	RP1	RP0	$\overline{T0}$	$\overline{PD}$	Z	DC	C	0001 1xxx	000q quuu	
84h <sup>(1)</sup>	FSR	Indirect data memory address pointer								xxxx xxxx	uuuu uuuu	
85h	TRISA	—	—	PORTA Data Direction Register						--11 1111	--11 1111	
86h	TRISB	PORTB Data Direction Register								1111 1111	1111 1111	
87h	TRISC	PORTC Data Direction Register								1111 1111	1111 1111	
88h <sup>(5)</sup>	TRISD	PORTD Data Direction Register								1111 1111	1111 1111	
89h <sup>(5)</sup>	TRISE	IBF	OBF	IBOV	PSPMODE	—	PORTE Data Direction Bits			0000 -111	0000 -111	
8Ah <sup>(1,2)</sup>	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter						---0 0000	---0 0000
8Bh <sup>(1)</sup>	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u	
8Ch	PIE1	PSPIE <sup>(6)</sup>	(4)	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000	
8Dh	PIE2	—	—	—	—	—	—	—	CCP2IE	---- --0	---- --0	
8Eh	PCON	—	—	—	—	—	—	POR	$\overline{BOR}$	---- --qq	---- --uu	
8Fh	—	Unimplemented								—	—	
90h	—	Unimplemented								—	—	
91h	—	Unimplemented								—	—	
92h	PR2	Timer2 Period Register								1111 1111	1111 1111	
93h	SSPADD	Synchronous Serial Port (I <sup>2</sup> C mode) Address Register								0000 0000	0000 0000	
94h	SSPSTAT	SMP	CKE	D/ $\overline{A}$	P	S	R/ $\overline{W}$	UA	BF	0000 0000	0000 0000	
95h	—	Unimplemented								—	—	
96h	—	Unimplemented								—	—	
97h	—	Unimplemented								—	—	
98h	TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010	
99h	SPBRG	Baud Rate Generator Register								0000 0000	0000 0000	
9Ah	—	Unimplemented								—	—	
9Bh	—	Unimplemented								—	—	
9Ch	—	Unimplemented								—	—	
9Dh	—	Unimplemented								—	—	
9Eh	—	Unimplemented								—	—	
9Fh	—	Unimplemented								—	—	

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

Shaded locations are unimplemented, read as '0'.

Note 1: These registers can be addressed from any 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: PIE1<6> and PIR1<6> are reserved on the PIC16C66/67, always maintain these bits clear.

5: PORTD, PORTE, TRISD, and TRISE are not implemented on the PIC16C66, read as '0'.

6: PSPIF (PIR1<7>) and PSPIE (PIE1<7>) are reserved on the PIC16C66, maintain these bits clear.

# PIC16C6X

**FIGURE 4-17: PIR1 REGISTER FOR PIC16C63/R63/66 (ADDRESS 0Ch)**

R/W-0	R/W-0	R-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF
bit7							bit0

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

bit 7-6: **Reserved:** Always maintain these bits clear.

bit 5: **RCIF:** USART Receive Interrupt Flag bit  
1 = The USART receive buffer is full (cleared by reading RCREG)  
0 = The USART receive buffer is empty

bit 4: **TXIF:** USART Transmit Interrupt Flag bit  
1 = The USART transmit buffer is empty (cleared by writing to TXREG)  
0 = The USART transmit buffer is full

bit 3: **SSPIF:** Synchronous Serial Port Interrupt Flag bit  
1 = The transmission/reception is complete (must be cleared in software)  
0 = Waiting to transmit/receive

bit 2: **CCP1IF:** CCP1 Interrupt Flag bit  
Capture Mode  
1 = A TMR1 register capture occurred (must be cleared in software)  
0 = No TMR1 register capture occurred  
Compare Mode  
1 = A TMR1 register compare match occurred (must be cleared in software)  
0 = No TMR1 register compare match occurred  
PWM Mode  
Unused in this mode

bit 1: **TMR2IF:** TMR2 to PR2 Match Interrupt Flag bit  
1 = TMR2 to PR2 match occurred (must be cleared in software)  
0 = No TMR2 to PR2 match occurred

bit 0: **TMR1IF:** TMR1 Overflow Interrupt Flag bit  
1 = TMR1 register overflow occurred (must be cleared in software)  
0 = No TMR1 register overflow occurred

Interrupt flag bits get set when an interrupt condition occurs regardless of the state of its corresponding enable bit or the global enable bit, GIE (INTCON<7>). User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.

## 7.2 Using Timer0 with External Clock

### Applicable Devices

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

When an external clock input is used for Timer0, it must meet certain requirements. The requirements ensure the external clock can be synchronized with the internal phase clock (Tosc). Also, there is a delay in the actual incrementing of Timer0 after synchronization.

### 7.2.1 EXTERNAL CLOCK SYNCHRONIZATION

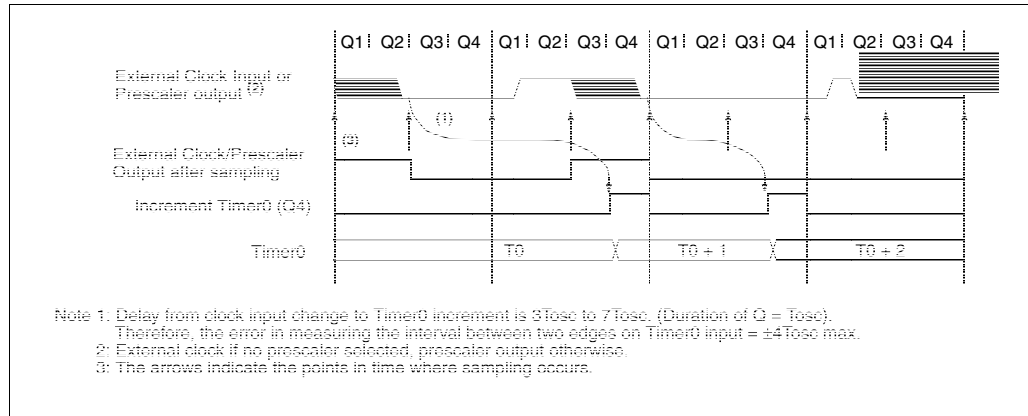
When no prescaler is used, the external clock input is the same as the prescaler output. The synchronization of T0CKI with the internal phase clocks is accomplished by sampling the prescaler output on the Q2 and Q4 cycles of the internal phase clocks (Figure 7-5). Therefore, it is necessary for T0CKI to be high for at least 2Tosc (and a small RC delay of 20 ns) and low for at least 2Tosc (and a small RC delay of 20 ns). Refer to the electrical specification of the desired device.

When a prescaler is used, the external clock input is divided by the asynchronous ripple-counter type prescaler so that the prescaler output is symmetrical. For the external clock to meet the sampling requirement, the ripple-counter must be taken into account. Therefore, it is necessary for T0CKI to have a period of at least 4Tosc (and a small RC delay of 40 ns) divided by the prescaler value. The only requirement on T0CKI high and low time is that they do not violate the minimum pulse width requirement of 10 ns. Refer to parameters 40, 41 and 42 in the electrical specification of the desired device.

### 7.2.2 TIMER0 INCREMENT DELAY

Since the prescaler output is synchronized with the internal clocks, there is a small delay from the time the external clock edge occurs to the time the Timer0 module is actually incremented. Figure 7-5 shows the delay from the external clock edge to the timer incrementing.

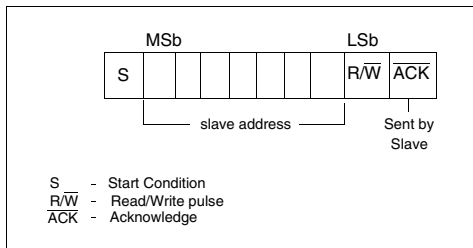
**FIGURE 7-5: TIMER0 TIMING WITH EXTERNAL CLOCK**



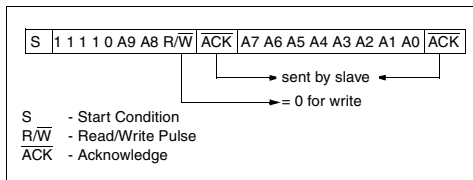
## 11.4.2 ADDRESSING I<sup>2</sup>C DEVICES

There are two address formats. The simplest is the 7-bit address format with a R/W bit (Figure 11-15). The more complex is the 10-bit address with a R/W bit (Figure 11-16). For 10-bit address format, two bytes must be transmitted with the first five bits specifying this to be a 10-bit address.

**FIGURE 11-15: 7-BIT ADDRESS FORMAT**



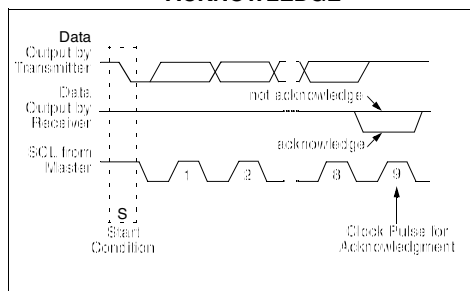
**FIGURE 11-16: I<sup>2</sup>C 10-BIT ADDRESS FORMAT**



## 11.4.3 TRANSFER ACKNOWLEDGE

All data must be transmitted per byte, with no limit to the number of bytes transmitted per data transfer. After each byte, the slave-receiver generates an acknowledge bit (ACK) (Figure 11-17). When a slave-receiver doesn't acknowledge the slave address or received data, the master must abort the transfer. The slave must leave SDA high so that the master can generate the STOP condition (Figure 11-14).

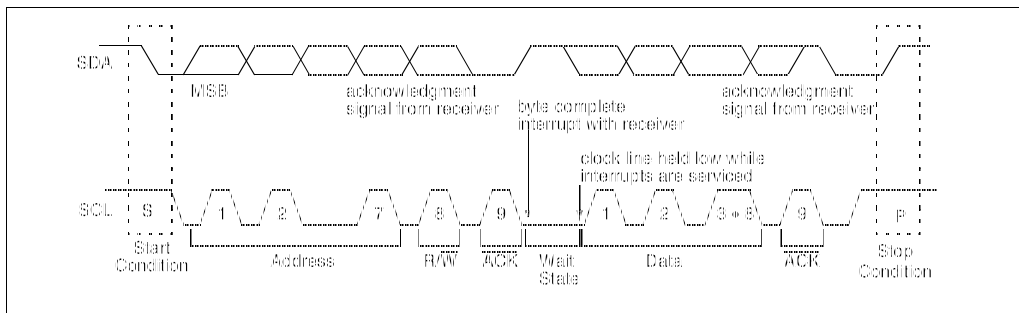
**FIGURE 11-17: SLAVE-RECEIVER ACKNOWLEDGE**



If the master is receiving the data (master-receiver), it generates an acknowledge signal for each received byte of data, except for the last byte. To signal the end of data to the slave-transmitter, the master does not generate an acknowledge (not acknowledge). The slave then releases the SDA line so the master can generate the STOP condition. The master can also generate the STOP condition during the acknowledge pulse for valid termination of data transfer.

If the slave needs to delay the transmission of the next byte, holding the SCL line low will force the master into a wait state. Data transfer continues when the slave releases the SCL line. This allows the slave to move the received data or fetch the data it needs to transfer before allowing the clock to start. This wait state technique can also be implemented at the bit level, Figure 11-18. The slave will inherently stretch the clock, when it is a transmitter, but will not when it is a receiver. The slave will have to clear the SSPCON<4> bit to enable clock stretching when it is a receiver.

**FIGURE 11-18: DATA TRANSFER WAIT STATE**





## 12.1 USART Baud Rate Generator (BRG)

### Applicable Devices

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  $F_{osc}$ , 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:

$F_{osc} = 16 \text{ MHz}$   
 Desired Baud Rate = 9600  
 BRGH = 0  
 SYNC = 0

## EXAMPLE 12-1: CALCULATING BAUD RATE ERROR

$$\begin{aligned} \text{Desired Baud rate} &= F_{osc} / (64 (X + 1)) \\ 9600 &= 16000000 / (64 (X + 1)) \\ X &= \lfloor 25.042 \rfloor = 25 \\ \text{Calculated Baud Rate} &= 16000000 / (64 (25 + 1)) \\ &= 9615 \\ \text{Error} &= \frac{(\text{Calculated Baud Rate} - \text{Desired Baud Rate})}{\text{Desired Baud Rate}} \\ &= (9615 - 9600) / 9600 \\ &= 0.16\% \end{aligned}$$

It may be advantageous to use the high baud rate (BRGH = 1) even for slower baud clocks. This is because the  $F_{osc}/(16(X + 1))$  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 = $F_{osc}/(64(X+1))$	Baud Rate = $F_{osc}/(16(X+1))$
1	(Synchronous) Baud Rate = $F_{osc}/(4(X+1))$	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 Rate Generator Register								0000 0000	0000 0000

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

# PIC16C6X

**TABLE 12-3: BAUD RATES FOR SYNCHRONOUS MODE**

BAUD RATE (K)	FOSC = 20 MHz			16 MHz			10 MHz			7.15909 MHz		
	KBAUD	% ERROR	SPBRG value (decimal)	KBAUD	% ERROR	SPBRG value (decimal)	KBAUD	% ERROR	SPBRG value (decimal)	KBAUD	% ERROR	SPBRG value (decimal)
0.3	NA	-	-	NA	-	-	NA	-	-	NA	-	-
1.2	NA	-	-	NA	-	-	NA	-	-	NA	-	-
2.4	NA	-	-	NA	-	-	NA	-	-	NA	-	-
9.6	NA	-	-	NA	-	-	9.766	+1.73	255	9.622	+0.23	185
19.2	19.53	+1.73	255	19.23	+0.16	207	19.23	+0.16	129	19.24	+0.23	92
76.8	76.92	+0.16	64	76.92	+0.16	51	75.76	-1.36	32	77.82	+1.32	22
96	96.15	+0.16	51	95.24	-0.79	41	96.15	+0.16	25	94.20	-1.88	18
300	294.1	-1.96	16	307.69	+2.56	12	312.5	+4.17	7	298.3	-0.57	5
500	500	0	9	500	0	7	500	0	4	NA	-	-
HIGH	5000	-	0	4000	-	0	2500	-	0	1789.8	-	0
LOW	19.53	-	255	15.625	-	255	9.766	-	255	6.991	-	255

BAUD RATE (K)	FOSC = 5.0688 MHz			4 MHz			3.579545 MHz			1 MHz			32.768 kHz		
	KBAUD	% ERROR	SPBRG value (decimal)	KBAUD	% ERROR	SPBRG value (decimal)	KBAUD	% ERROR	SPBRG value (decimal)	KBAUD	% ERROR	SPBRG value (decimal)	KBAUD	% ERROR	SPBRG value (decimal)
0.3	NA	-	-	NA	-	-	NA	-	-	NA	-	-	0.303	+1.14	26
1.2	NA	-	-	NA	-	-	NA	-	-	1.202	+0.16	207	1.170	-2.48	6
2.4	NA	-	-	NA	-	-	NA	-	-	2.404	+0.16	103	NA	-	-
9.6	9.6	0	131	9.615	+0.16	103	9.622	+0.23	92	9.615	+0.16	25	NA	-	-
19.2	19.2	0	65	19.231	+0.16	51	19.04	-0.83	46	19.24	+0.16	12	NA	-	-
76.8	79.2	+3.13	15	76.923	+0.16	12	74.57	-2.90	11	83.34	+8.51	2	NA	-	-
96	97.48	+1.54	12	1000	+4.17	9	99.43	+3.57	8	NA	-	-	NA	-	-
300	316.8	+5.60	3	NA	-	-	298.3	-0.57	2	NA	-	-	NA	-	-
500	NA	-	-	NA	-	-	NA	-	-	NA	-	-	NA	-	-
HIGH	1267	-	0	100	-	0	894.9	-	0	250	-	0	8.192	-	0
LOW	4.950	-	255	3.906	-	255	3.496	-	255	0.9766	-	255	0.032	-	255

**TABLE 12-4: BAUD RATES FOR ASYNCHRONOUS MODE (BRGH = 0)**

BAUD RATE (K)	FOSC = 20 MHz			16 MHz			10 MHz			7.15909 MHz		
	KBAUD	% ERROR	SPBRG value (decimal)	KBAUD	% ERROR	SPBRG value (decimal)	KBAUD	% ERROR	SPBRG value (decimal)	KBAUD	% ERROR	SPBRG value (decimal)
0.3	NA	-	-	NA	-	-	NA	-	-	NA	-	-
1.2	1.221	+1.73	255	1.202	+0.16	207	1.202	+0.16	129	1.203	+0.23	92
2.4	2.404	+0.16	129	2.404	+0.16	103	2.404	+0.16	64	2.380	-0.83	46
9.6	9.469	-1.36	32	9.615	+0.16	25	9.766	+1.73	15	9.322	-2.90	11
19.2	19.53	+1.73	15	19.23	+0.16	12	19.53	+1.73	7	18.64	-2.90	5
76.8	78.13	+1.73	3	83.33	+8.51	2	78.13	+1.73	1	NA	-	-
96	104.2	+8.51	2	NA	-	-	NA	-	-	NA	-	-
300	312.5	+4.17	0	NA	-	-	NA	-	-	NA	-	-
500	NA	-	-	NA	-	-	NA	-	-	NA	-	-
HIGH	312.5	-	0	250	-	0	156.3	-	0	111.9	-	0
LOW	1.221	-	255	0.977	-	255	0.6104	-	255	0.437	-	255

BAUD RATE (K)	FOSC = 5.0688 MHz			4 MHz			3.579545 MHz			1 MHz			32.768 kHz		
	KBAUD	% ERROR	SPBRG value (decimal)	KBAUD	% ERROR	SPBRG value (decimal)	KBAUD	% ERROR	SPBRG value (decimal)	KBAUD	% ERROR	SPBRG value (decimal)	KBAUD	% ERROR	SPBRG value (decimal)
0.3	0.31	+3.13	255	0.3005	-0.17	207	0.301	+0.23	185	0.300	+0.16	51	0.256	-14.67	1
1.2	1.2	0	65	1.202	+1.67	51	1.190	-0.83	46	1.202	+0.16	12	NA	-	-
2.4	2.4	0	32	2.404	+1.67	25	2.432	+1.32	22	2.232	-6.99	6	NA	-	-
9.6	9.9	+3.13	7	NA	-	-	9.322	-2.90	5	NA	-	-	NA	-	-
19.2	19.8	+3.13	3	NA	-	-	18.64	-2.90	2	NA	-	-	NA	-	-
76.8	79.2	+3.13	0	NA	-	-	NA	-	-	NA	-	-	NA	-	-
96	NA	-	-	NA	-	-	NA	-	-	NA	-	-	NA	-	-
300	NA	-	-	NA	-	-	NA	-	-	NA	-	-	NA	-	-
500	NA	-	-	NA	-	-	NA	-	-	NA	-	-	NA	-	-
HIGH	79.2	-	0	62.500	-	0	55.93	-	0	15.63	-	0	0.512	-	0
LOW	0.3094	-	255	3.906	-	255	0.2185	-	255	0.0610	-	255	0.0020	-	255

## 13.0 SPECIAL FEATURES OF THE CPU

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

What sets a microcontroller apart from other processors are special circuits to deal with the needs of real-time applications. The PIC16CXX family has a host of such features intended to maximize system reliability, minimize cost through elimination of external components, provide power saving operating modes and offer code protection. These are:

- Oscillator selection
- Reset
  - Power-on Reset (POR)
  - Power-up Timer (PWRT)
  - Oscillator Start-up Timer (OST)
  - Brown-out Reset (BOR)
- Interrupts
- Watchdog Timer (WDT)
- SLEEP mode
- Code protection
- ID locations
- In-circuit serial programming

The PIC16CXX has a Watchdog Timer which can be shut off only through configuration bits. It runs off its own RC oscillator for added reliability. There are two

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

SLEEP mode is designed to offer a very low current power-down mode. The user can wake from SLEEP through external reset, Watchdog Timer Wake-up or through an interrupt. Several oscillator options are also made available to allow the part to fit the application. The RC oscillator option saves system cost while the LP crystal option saves power. A set of configuration bits are used to select various options.

### 13.1 Configuration Bits

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

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

The user will note that address 2007h is beyond the user program memory space. In fact, it belongs to the special test/configuration memory space (2000h - 3FFFh), which can be accessed only during programming.

**FIGURE 13-1: CONFIGURATION WORD FOR PIC16C61**

—	—	—	—	—	—	—	—	—	CP0	PWRT	WDTE	FOSC1	FOSC0	Register: CONFIG	Address: 2007h
bit 13													bit 0		
bit 13-5:	<b>Unimplemented:</b> Read as '1'														
bit 4:	<b>CP0:</b> Code protection bit 1 = Code protection off 0 = All memory is code protected, but 00h - 3Fh is writable														
bit 3:	<b>PWRT:</b> Power-up Timer Enable bit 1 = Power-up Timer enabled 0 = Power-up Timer disabled														
bit 2:	<b>WDTE:</b> Watchdog Timer Enable bit 1 = WDT enabled 0 = WDT disabled														
bit 1-0:	<b>FOSC1:FOSC0:</b> Oscillator Selection bits 11 = RC oscillator 10 = HS oscillator 01 = XT oscillator 00 = LP oscillator														

## 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:

- Stores the W register
- Stores the STATUS register in bank 0
- Stores PCLATH
- Executes ISR code
- Restores PCLATH
- Restores STATUS register (and bank select bit)
- 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
MOVWF    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
SWAPF    W_TEMP,W         ;Swap W_TEMP into W

```

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

```

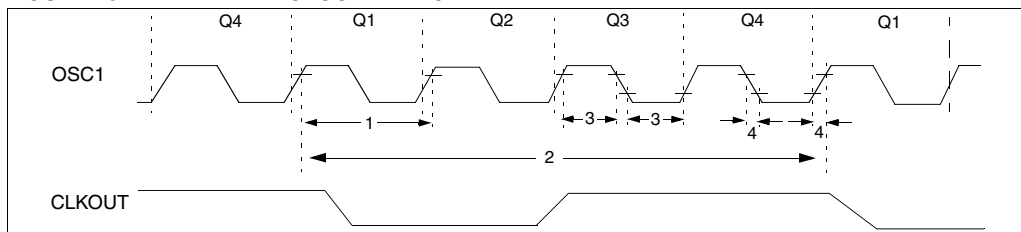
MOVWF    W_TEMP           ;Copy W to TEMP register, could be bank one or zero
SWAPF    STATUS,W         ;Swap status to be saved into W
CLRF     STATUS           ;bank 0, regardless of current bank, Clears IRP,RP1,RP0
MOVWF    STATUS_TEMP      ;Save status to bank zero STATUS_TEMP register
MOVF     PCLATH, W        ;Only required if using pages 1, 2 and/or 3
MOVWF    PCLATH_TEMP      ;Save PCLATH into W
CLRF     PCLATH           ;Page zero, regardless of current page
BCF      STATUS, IRP      ;Return to Bank 0
MOVF     FSR, W           ;Copy FSR to W
MOVWF    FSR_TEMP         ;Copy FSR from W to FSR_TEMP
: (ISR)
:
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
SWAPF    W_TEMP,W         ;Swap W_TEMP into W

```

5: Timer1 oscillator (when enabled) adds approximately 20  $\mu$ A to the specification. This value is from characterization and is for design guidance only. This is not tested.

## 19.5 Timing Diagrams and Specifications

**FIGURE 19-2: EXTERNAL CLOCK TIMING**



**TABLE 19-2: EXTERNAL CLOCK TIMING REQUIREMENTS**

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

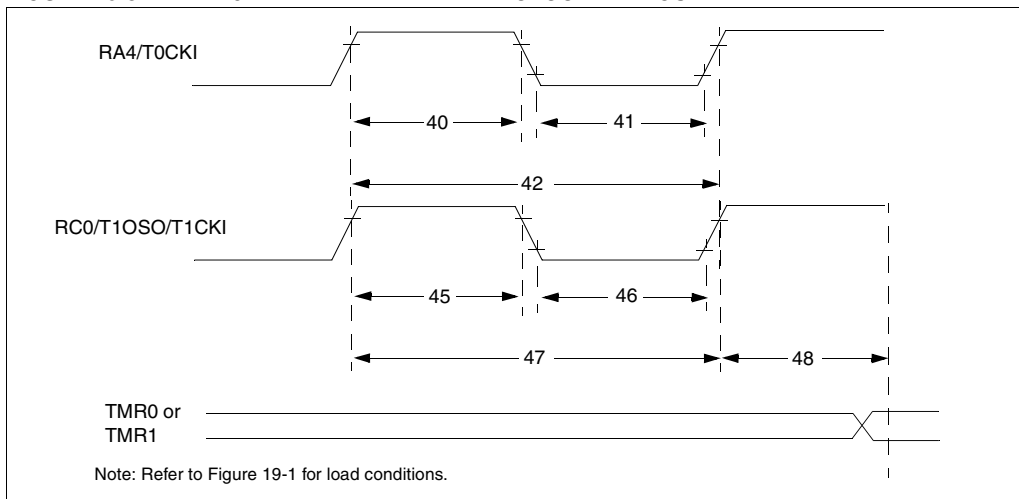
† Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

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.

# PIC16C6X

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

**FIGURE 19-5: TIMER0 AND TIMER1 EXTERNAL CLOCK TIMINGS**



**TABLE 19-5: TIMER0 AND TIMER1 EXTERNAL CLOCK REQUIREMENTS**

Param No.	Sym	Characteristic		Min	Typ†	Max	Units	Conditions
40*	Tt0H	T0CKI High Pulse Width	No Prescaler	$0.5T_{CY} + 20$	—	—	ns	Must also meet parameter 42
			With Prescaler	10	—	—	ns	
41*	Tt0L	T0CKI Low Pulse Width	No Prescaler	$0.5T_{CY} + 20$	—	—	ns	Must also meet parameter 42
			With Prescaler	10	—	—	ns	
42*	Tt0P	T0CKI Period	No Prescaler	$T_{CY} + 40$	—	—	ns	
			With Prescaler	Greater of: $20$ or $T_{CY} + 40$ N	—	—	ns	N = prescale value (2, 4, ..., 256)
45*	Tt1H	T1CKI High Time	Synchronous, Prescaler = 1	$0.5T_{CY} + 20$	—	—	ns	Must also meet parameter 47
		Synchronous, Prescaler = 2,4,8	PIC16C6X	15	—	—	ns	
			PIC16LC6X	25	—	—	ns	
		Asynchronous	PIC16C6X	30	—	—	ns	
			PIC16LC6X	50	—	—	ns	
46*	Tt1L	T1CKI Low Time	Synchronous, Prescaler = 1	$0.5T_{CY} + 20$	—	—	ns	Must also meet parameter 47
		Synchronous, Prescaler = 2,4,8	PIC16C6X	15	—	—	ns	
			PIC16LC6X	25	—	—	ns	
		Asynchronous	PIC16C6X	30	—	—	ns	
			PIC16LC6X	50	—	—	ns	
47*	Tt1P	T1CKI input period	Synchronous	Greater of: $30$ OR $T_{CY} + 40$ N	—	—	ns	N = prescale value (1, 2, 4, 8)
			PIC16LC6X	Greater of: $50$ OR $T_{CY} + 40$ N				N = prescale value (1, 2, 4, 8)
		Asynchronous	PIC16C6X	60	—	—	ns	
			PIC16LC6X	100	—	—	ns	
	Ft1	Timer1 oscillator input frequency range (oscillator enabled by setting bit T1OSCEN)		DC	—	200	kHz	
48	TCKEZtmr1	Delay from external clock edge to timer increment		$2T_{osc}$	—	$7T_{osc}$	—	

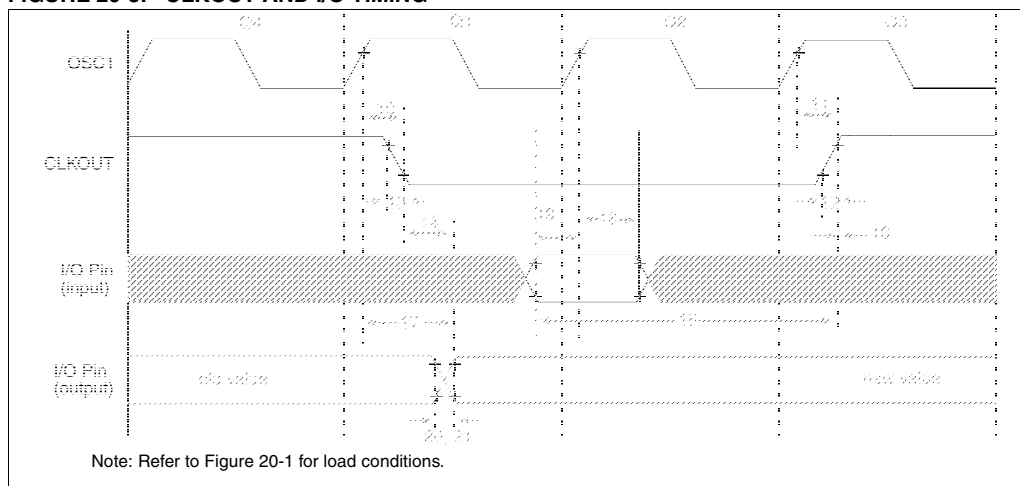
\* These parameters are characterized but not tested.

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

# PIC16C6X

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

**FIGURE 20-3: CLKOUT AND I/O TIMING**



**TABLE 20-3: CLKOUT AND I/O TIMING REQUIREMENTS**

Parameter No.	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 valid	—	50	150	ns	
18*	TosH2ioI	OSC1↑ (Q2 cycle) to Port input invalid (I/O in hold time)	PIC16C63/65A	100	—	—	ns
			PIC16LC63/65A	200	—	—	ns
19*	TioV2osH	Port input valid to OSC1↑ (I/O in setup time)	0	—	—	ns	
20*	TioR	Port output rise time	PIC16C63/65A	—	10	40	ns
			PIC16LC63/65A	—	—	80	ns
21*	TioF	Port output fall time	PIC16C63/65A	—	10	40	ns
			PIC16LC63/65A	—	—	80	ns
22††*	Tinp	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.



# PIC16C6X

## F.7 PIC16C7XX Family of Devices

		PIC16C710	PIC16C71	PIC16C711	PIC16C715	PIC16C72	PIC16C72 <sup>(1)</sup>
<b>Clock</b>	Maximum Frequency of Operation (MHz)	20	20	20	20	20	20
<b>Memory</b>	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
<b>Peripherals</b>	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 <sup>2</sup> C, USART)	—	—	—	—	SPI/I <sup>2</sup> C	SPI/I <sup>2</sup> C
	Parallel Slave Port	—	—	—	—	—	—
	A/D Converter (8-bit) Channels	4	4	4	4	5	5
<b>Features</b>	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
<b>Clock</b>	Maximum Frequency of Operation (MHz)	20	20	20	20
<b>Memory</b>	EPROM Program Memory (x14 words)	4K	4K	8K	8K
	Data Memory (bytes)	192	192	368	368
<b>Peripherals</b>	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 <sup>2</sup> C, USART)	SPI/I <sup>2</sup> C, USART	SPI/I <sup>2</sup> C, USART	SPI/I <sup>2</sup> C, USART	SPI/I <sup>2</sup> C, USART
	Parallel Slave Port	—	Yes	—	Yes
	A/D Converter (8-bit) Channels	5	8	5	8
<b>Features</b>	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.

**NOTES:**

## INDEX

### Numerics

9-bit Receive Enable bit, RX9 .....	106
9-bit Transmit Enable bit, TX9 .....	105
9th bit of received data, RX9D .....	106
9th bit of transmit data, TX9D .....	105

### A

Absolute Maximum .....	
Ratings .....	163, 183, 199, 215, 231, 247, 263
ACK .....	96, 100, 101
ALU .....	9
Application Notes .....	
AN552 (Implementing Wake-up on Key Stroke) .....	53
AN556 (Implementing a Table Read) .....	48
AN594 (Using the CCP Modules) .....	77
Architectural Overview .....	9

### B

Baud Rate Formula .....	107
Baud Rate Generator .....	107
Baud Rates .....	
Asynchronous Mode .....	108
Error, Calculating .....	107
RX Pin Sampling, Timing Diagrams .....	110, 111
Sampling .....	110
Synchronous Mode .....	108
BF .....	84, 89, 100

### Block Diagrams

Capture Mode Operation .....	78
Compare Mode .....	79
Crystal Oscillator, Ceramic Resonator .....	125
External Brown-out Protection .....	135
External Parallel Resonant Crystal Circuit .....	127
External Power-on Reset .....	135
External Series Resonant Crystal Circuit .....	127
I <sup>2</sup> C Mode .....	99
In-circuit Programming Connections .....	142
Interrupt Logic .....	137
On-chip Reset Circuit .....	128
Parallel Slave Port, PORTD-PORTE .....	61
PIC16C61 .....	10
PIC16C62 .....	11
PIC16C62A .....	11
PIC16C63 .....	12
PIC16C64 .....	11
PIC16C64A .....	11
PIC16C65 .....	12
PIC16C65A .....	12
PIC16C66 .....	13
PIC16C67 .....	13
PIC16CR62 .....	11
PIC16CR63 .....	12
PIC16CR64 .....	11
PIC16CR65 .....	12
PORTC .....	55
PORTD (I/O Mode) .....	57
PORTE (I/O Mode) .....	58
PWM .....	80
RA3:RA0 pins .....	51
RA4/TOCK1 pin .....	51
RA5 pin .....	51
RB3:RB0 pins .....	54
RB7:RB4 pins .....	53, 54
RC Oscillator Mode .....	127

SPI Master/Slave Connection .....	87
SSP in I <sup>2</sup> C Mode .....	99
SSP in SPI Mode .....	86, 91
Timer0 .....	65
Timer0/WDT Prescaler .....	68
Timer1 .....	72
Timer2 .....	75
USART Receive .....	114
USART Transmit .....	112
Watchdog Timer .....	140
BOR .....	129
BOR .....	47, 131
BRGH .....	105
Brown-out Reset (BOR) .....	129
Brown-out Reset Status bit, BOR .....	47
Buffer Full Status bit, BF .....	84, 89

### C

C .....	35
C Compiler .....	161
Capture .....	
Block Diagram .....	78
Mode .....	78
Pin Configuration .....	78
Prescaler .....	79
Software Interrupt .....	78
Capture Interrupt .....	78
Capture/Compare/PWM (CCP) .....	
Capture Mode .....	78
Capture Mode Block Diagram .....	78
CCP1 .....	77
CCP2 .....	77
Compare Mode .....	79
Compare Mode Block Diagram .....	79
Overview .....	63
Prescaler .....	79
PWM Block Diagram .....	80
PWM Mode .....	80
PWM, Example Frequencies/Resolutions .....	81
Section .....	77
Carry .....	9
Carry bit .....	35
CCP Module Interaction .....	77
CCP pin Configuration .....	78
CCP to Timer Resource Use .....	77
CCP1 Interrupt Enable bit, CCP1IE .....	38
CCP1 Interrupt Flag bit, CCP1IF .....	41
CCP1 Mode Select bits .....	78
CCP1CON .....	24, 26, 28, 30, 32, 34
CCP1IE .....	38
CCP1IF .....	41
CCP1M3:CCM1M0 .....	78
CCP1X:CCP1Y .....	78
CCP2 Interrupt Enable bit, CCP2IE .....	45
CCP2 Interrupt Flag bit, CCP2IF .....	46
CCP2 Mode Select bits .....	78
CCP2CON .....	24, 26, 28, 30, 32, 34
CCP2IE .....	45
CCP2IF .....	46
CCP2M3:CCP2M0 .....	78
CCP2X:CCP2Y .....	78
CCPR1H .....	24, 26, 28, 30, 32, 34
CCPR1L .....	24, 26, 28, 30, 32, 34
CCPR2H .....	24, 26, 28, 30, 32, 34
CCPR2L .....	24, 26, 28, 30, 32, 34
CKE .....	89
CKP .....	85, 90

# PIC16C6X

Clearing Interrupts.....	53
Clock Polarity Select bit, CKP .....	85, 90
Clock Polarity, SPI Mode .....	87
Clock Source Select bit, CSRC .....	105
Clocking Scheme .....	18
Code Examples	
Changing Between Capture Prescalers .....	79
Ensuring Interrupts are Globally Disabled .....	136
Indirect Addressing .....	49
Initializing PORTA .....	51
Initializing PORTB .....	53
Initializing PORTC .....	55
Loading the SSPBUF Register .....	86
Loading the SSPBUF register .....	91
Reading a 16-bit Free-running Timer .....	73
Read-Modify-Write on an I/O Port .....	60
Saving Status, W, and PCLATH Registers .....	139
Subroutine Call, Page0 to Page1 .....	49
Code Protection .....	142
Compare	
Block Diagram .....	79
Mode .....	79
Pin Configuration .....	79
Software Interrupt .....	79
Special Event Trigger .....	79
Computed GOTO .....	48
Configuration Bits .....	123
Configuration Word, Diagram .....	124
Connecting Two Microcontrollers .....	87
Continuous Receive Enable bit, CREN .....	106
CREN .....	106
CSRC .....	105
<b>D</b>	
D/Ā .....	84, 89
Data/Address bit, D/Ā .....	84, 89
Data Memory	
Organization .....	20
Section .....	20
Data Sheet	
Compatibility .....	307
Modifications .....	307
What's New .....	308
DC .....	35
DC CHARACTERISTICS .. 164, 184, 200, 216, 232, 248, 264	
Development Support .....	159
Development Tools .....	159
Device Drawings	
18-Lead Ceramic CERDIP Dual In-line	
with Window (300 mil) .....	296
18-Lead Plastic Dual In-line (300 mil) .....	291
18-Lead Plastic Surface Mount	
(SOIC - Wide, 300 mil Body) .....	294
28-Lead Ceramic CERDIP Dual In-line with	
Window (300 mil) .....	297
28-Lead Ceramic Side Brazed Dual In-Line	
with Window (300 mil) .....	299
28-Lead Plastic Dual In-line (300 mil) .....	292
28-Lead Plastic Surface Mount	
(SOIC - Wide, 300 mil Body) .....	295
28-Lead Plastic Surface Mount	
(SSOP - 209 mil Body 5.30 mm) .....	300
40-Lead Ceramic CERDIP Dual In-line	
with Window (600 mil) .....	298
40-Lead Plastic Dual In-line (600 mil) .....	293
44-Lead Plastic Leaded Chip Carrier (Square) .....	301

44-Lead Plastic Surface Mount (MQFP	
10x10 mm Body 1.6/0.15 mm Lead Form) .....	302, 303
Device Varieties .....	7
Digit Carry .....	9
Digit Carry bit .....	35
Direct Addressing .....	49

## E

Electrical Characteristics .. 163, 183, 199, 215, 231, 247, 263	
External Clock Synchronization, TMR0 .....	67

## F

Family of Devices	
PIC12CXXX .....	309
PIC14C000 .....	309
PIC16C15X .....	310
PIC16C55X .....	311
PIC16C5X .....	310
PIC16C62X and PIC16C64X .....	311
PIC16C6X .....	6
PIC16C7XX .....	312
PIC16C8X .....	313
PIC16C9XX .....	313
PIC17CXX .....	314
FERR .....	106
Framing Error bit, FERR .....	106
FSR .....	24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34
Fuzzy Logic Dev. System ( <i>fuzzyTECH</i> ®-MP) .....	159, 161

## G

General Description .....	5
General Purpose Registers .....	20
GIE .....	37
Global Interrupt Enable bit, GIE .....	37
Graphs	
PIC16C6X .....	281
PIC16C61 .....	173

## H

High Baud Rate Select bit, BRGH .....	105
---------------------------------------	-----

## I

I/O Ports, Section .....	51
I <sup>2</sup> C	
Addressing .....	100
Addressing I <sup>2</sup> C Devices .....	96
Arbitration .....	98
Block Diagram .....	99
Clock Synchronization .....	98
Combined Format .....	97
I <sup>2</sup> C Operation .....	99
I <sup>2</sup> C Overview .....	95
Initiating and Terminating Data Transfer .....	95
Master Mode .....	103
Master-Receiver Sequence .....	97
Master-Transmitter Sequence .....	97
Mode .....	99
Mode Selection .....	99
Multi-master .....	98
Multi-Master Mode .....	103
Reception .....	101
Reception Timing Diagram .....	101
SCL and SDA pins .....	100
Slave Mode .....	100
START .....	95
STOP .....	95, 96

## Registers

CCP1CON	
Diagram .....	78
Section .....	78
Summary .....	24, 26, 28, 30, 32
CCP2CON	
Diagram .....	78
Section .....	78
Summary .....	26, 30, 32
CCPR1H	
Summary .....	24, 26, 28, 30, 32
CCPR1L	
Summary .....	24, 26, 28, 30, 32
CCPR2H	
Summary .....	26, 30, 32
CCPR2L	
Summary .....	26, 30, 32
FSR	
Indirect Addressing .....	49
Summary .....	24, 26, 28, 30, 32, 34
INDF	
Indirect Addressing .....	49
Summary .....	24, 26, 28, 30, 32, 34
INTCON	
Diagram .....	37
Section .....	37
Summary .....	24, 26, 28, 30, 32, 34
OPTION	
Diagram .....	36
Section .....	36
Summary .....	25, 27, 29, 31, 33, 34
PCL	
Section .....	48
Summary .....	24, 26, 28, 30, 32, 34
PCLATH	
Section .....	48
Summary .....	24, 26, 28, 30, 32, 34
PCON	
Diagram .....	47
Section .....	47
Summary .....	25, 27, 29, 31, 33
PIE1	
Diagram .....	40
Section .....	38
Summary .....	25, 27, 29, 31, 33
PIE2	
Diagram .....	45
Section .....	45
Summary .....	27, 31, 33
PIR1	
Diagram .....	44
Section .....	41
Summary .....	24, 26, 28, 30, 32
PIR2	
Diagram .....	46
Section .....	46
Summary .....	26, 30, 32
PORTA	
Section .....	51
Summary .....	24, 26, 28, 30, 32
PORTB	
Section .....	53
Summary .....	24, 26, 28, 30, 32, 34
PORTC	
Section .....	55
Summary .....	24, 26, 28, 30, 32

PORTD	
Section .....	57
Summary .....	28, 30, 32
PORTE	
Section .....	58
Summary .....	28, 30, 32
PR2	
Summary .....	25, 27, 29, 31, 33
RCREG	
Summary .....	26, 30, 32
RCSTA	
Diagram .....	106
Summary .....	26, 30, 32
SPBRG	
Summary .....	27, 31, 33
SSPBUF	
Section .....	86
Summary .....	24, 26, 28, 30, 32
SSPCON	
Diagram .....	85
Summary .....	24, 26, 28, 30, 32
SSPSR	
Section .....	86
SSPSTAT	
Diagram .....	84
Section .....	84
Summary .....	25, 27, 29, 31, 33
STATUS	
Diagram .....	35
Section .....	35
Summary .....	24, 26, 28, 30, 32, 34
T1CON	
Diagram .....	71
Section .....	71
Summary .....	24, 26, 28, 30, 32
T2CON	
Diagram .....	75
Section .....	75
Summary .....	24, 26, 28, 30, 32
TMR0	
Summary .....	24, 26, 28, 30, 32, 34
TMR1H	
Summary .....	24, 26, 28, 30, 32
TMR1L	
Summary .....	24, 26, 28, 30, 32
TMR2	
Summary .....	24, 26, 28, 30, 32
TRISA	
Section .....	51
Summary .....	25, 27, 29, 31, 33
TRISB	
Section .....	53
Summary .....	25, 27, 29, 31, 33, 34
TRISC	
Section .....	55
Summary .....	25, 27, 29, 31, 33
TRISD	
Section .....	57
Summary .....	29, 31, 33
TRISE	
Diagram .....	58
Section .....	58
Summary .....	29, 31, 33
TXREG	
Summary .....	26, 30, 32