



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.

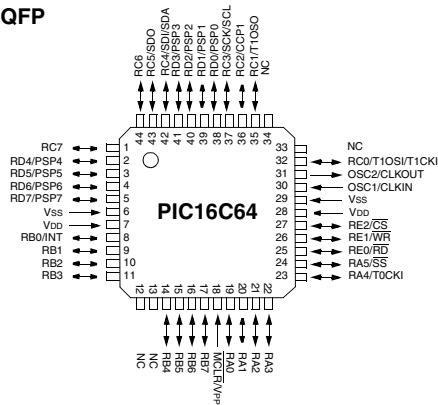
Applications of "[Embedded - Microcontrollers](#)"

Details

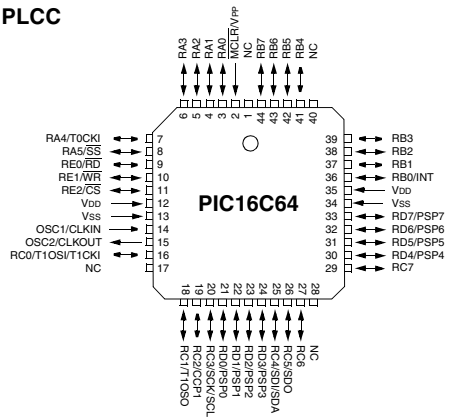
Product Status	Active
Core Processor	PIC
Core Size	8-Bit
Speed	20MHz
Connectivity	I ² C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	33
Program Memory Size	7KB (4K x 14)
Program Memory Type	OTP
EEPROM Size	-
RAM Size	192 x 8
Voltage - Supply (Vcc/Vdd)	4V ~ 6V
Data Converters	-
Oscillator Type	External
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Surface Mount
Package / Case	44-QFP
Supplier Device Package	44-MQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16c65a-20-pq

Pin Diagrams (Cont'd)

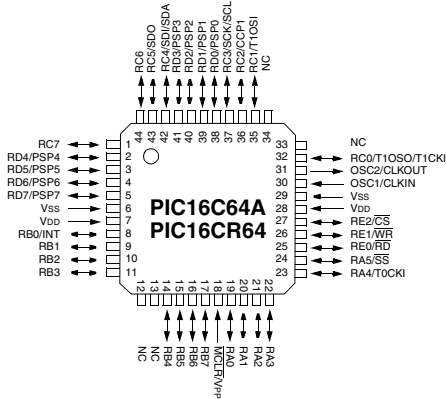
MQFP



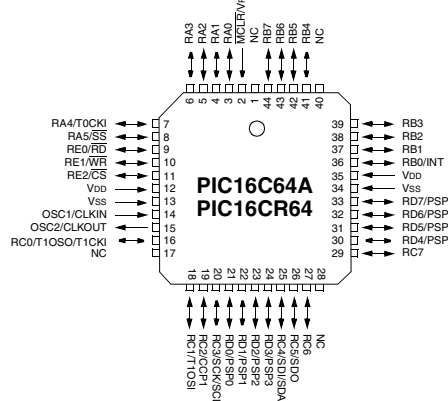
PLCC



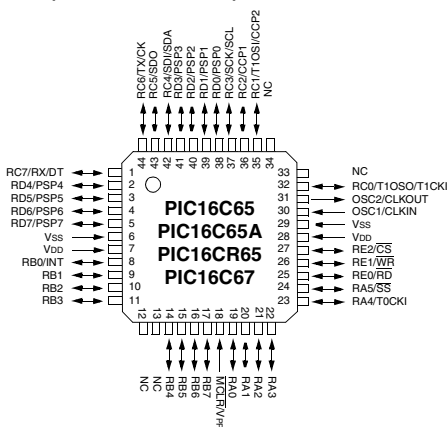
MQFP, TQFP (PIC16C64A only)



PLCC



MQFP, TQFP (Not on PIC16C65)



PLCC

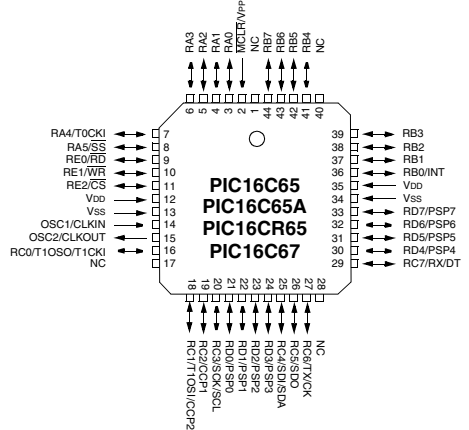


TABLE 3-2: PIC16C62/62A/R62/63/R63/66 PINOUT DESCRIPTION

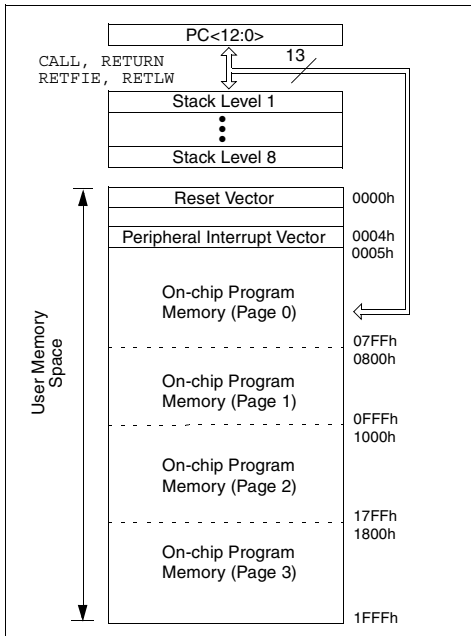
Pin Name	Pin#	Pin Type	Buffer Type	Description
OSC1/CLKIN	9	I	ST/CMOS ⁽³⁾	Oscillator crystal input/external clock source input.
OSC2/CLKOUT	10	O	—	Oscillator crystal output. Connects to crystal or resonator in crystal oscillator mode. In RC mode, the pin outputs CLKOUT which has 1/4 the frequency of OSC1, and denotes the instruction cycle rate.
MCLR/VPP	1	I/P	ST	Master clear reset input or programming voltage input. This pin is an active low reset to the device.
RA0	2	I/O	TTL	PORTA is a bi-directional I/O port. RA4 can also be the clock input to the Timer0 timer/counter. Output is open drain type. RA5 can also be the slave select for the synchronous serial port.
RA1	3	I/O	TTL	
RA2	4	I/O	TTL	
RA3	5	I/O	TTL	
RA4/T0CKI	6	I/O	ST	
RA5/ \overline{SS}	7	I/O	TTL	
RB0/INT	21	I/O	TTL/ST ⁽⁴⁾	PORTB is a bi-directional I/O port. PORTB can be software programmed for internal weak pull-up on all inputs. RB0 can also be the external interrupt pin. Interrupt on change pin. Interrupt on change pin. Interrupt on change pin. Serial programming clock. Interrupt on change pin. Serial programming data.
RB1	22	I/O	TTL	
RB2	23	I/O	TTL	
RB3	24	I/O	TTL	
RB4	25	I/O	TTL	
RB5	26	I/O	TTL	
RB6	27	I/O	TTL/ST ⁽⁵⁾	
RB7	28	I/O	TTL/ST ⁽⁵⁾	
RC0/T1OSO ⁽¹⁾ /T1CKI	11	I/O	ST	PORTC is a bi-directional I/O port. RC0 can also be the Timer1 oscillator output ⁽¹⁾ or Timer1 clock input. RC1 can also be the Timer1 oscillator input ⁽¹⁾ or Capture2 input/Compare2 output/PWM2 output ⁽²⁾ . RC2 can also be the Capture1 input/Compare1 output/PWM1 output. RC3 can also be the synchronous serial clock input/output for both SPI and I ² C modes. RC4 can also be the SPI Data In (SPI mode) or data I/O (I ² C mode). RC5 can also be the SPI Data Out (SPI mode). RC6 can also be the USART Asynchronous Transmit ⁽²⁾ or Synchronous Clock ⁽²⁾ . RC7 can also be the USART Asynchronous Receive ⁽²⁾ or Synchronous Data ⁽²⁾ .
RC1/T1OSI ⁽¹⁾ /CCP2 ⁽²⁾	12	I/O	ST	
RC2/CCP1	13	I/O	ST	
RC3/SCK/SCL	14	I/O	ST	
RC4/SDI/SDA	15	I/O	ST	
RC5/SDO	16	I/O	ST	
RC6/TX/CK ⁽²⁾	17	I/O	ST	
RC7/RX/DT ⁽²⁾	18	I/O	ST	
Vss	8,19	P	—	Ground reference for logic and I/O pins.
VDD	20	P	—	Positive supply for logic and I/O pins.

Legend: I = input O = output I/O = input/output P = power
 — = Not used TTL = TTL input ST = Schmitt Trigger input

- Note 1: Pin functions T1OSO and T1OSI are reversed on the PIC16C62.
 2: The USART and CCP2 are not available on the PIC16C62/62A/R62.
 3: This buffer is a Schmitt Trigger input when configured in RC oscillator mode and a CMOS input otherwise.
 4: This buffer is a Schmitt Trigger input when configured as the external interrupt.
 5: This buffer is a Schmitt Trigger input when used in serial programming mode.

PIC16C6X

FIGURE 4-4: PIC16C66/67 PROGRAM MEMORY MAP AND STACK



4.2 Data Memory Organization

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

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

RP1:RP0 (STATUS<6:5>)

- = 00 → Bank0
- = 01 → Bank1
- = 10 → Bank2
- = 11 → Bank3

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

4.2.1 GENERAL PURPOSE REGISTERS

These registers are accessed either directly or indirectly through the File Select Register (FSR) (Section 4.5).

For the PIC16C61, general purpose register locations 8Ch-AFh of Bank 1 are not physically implemented. These locations are mapped into 0Ch-2Fh of Bank 0.

FIGURE 4-5: PIC16C61 REGISTER FILE MAP

File Address		File Address	
00h	INDF ⁽¹⁾	INDF ⁽¹⁾	80h
01h	TMR0	OPTION	81h
02h	PCL	PCL	82h
03h	STATUS	STATUS	83h
04h	FSR	FSR	84h
05h	PORTA	TRISA	85h
06h	PORTB	TRISB	86h
07h			87h
08h			88h
09h			89h
0Ah	PCLATH	PCLATH	8Ah
0Bh	INTCON	INTCON	8Bh
0Ch	General Purpose Register	Mapped in Bank 0 ⁽²⁾	8Ch
2Fh			AFh
30h			B0h
7Fh			FFh
	Bank 0	Bank 1	

☐ Unimplemented data memory location; read as '0'.
 Note 1: Not a physical register.
 Note 2: These locations are unimplemented in Bank 1. Any access to these locations will access the corresponding Bank 0 register.

TABLE 4-2: SPECIAL FUNCTION REGISTERS FOR THE PIC16C62/62A/R62 (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 ⁽³⁾
Bank 1											
80h ⁽¹⁾	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 ⁽¹⁾	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	0000 0000
83h ⁽¹⁾	STATUS	IRP ⁽⁵⁾	RP1 ⁽⁵⁾	RP0	T0	PD	Z	DC	C	0001 1xxx	000q quuu
84h ⁽¹⁾	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	—	Unimplemented								—	—
89h	—	Unimplemented								—	—
8Ah ^(1,2)	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter					---0 0000	---0 0000
8Bh ⁽¹⁾	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
8Ch	PIE1	(6)	(6)	—	—	SSPIE	CCP1IE	TMR2IE	TMR1IE	00-- 0000	00-- 0000
8Dh	—	Unimplemented								—	—
8Eh	PCON	—	—	—	—	—	—	POR	BOR ⁽⁴⁾	---- --qq	---- --uu
8Fh	—	Unimplemented								—	—
90h	—	Unimplemented								—	—
91h	—	Unimplemented								—	—
92h	PR2	Timer2 Period Register								1111 1111	1111 1111
93h	SSPADD	Synchronous Serial Port (I ² C mode) Address Register								0000 0000	0000 0000
94h	SSPSTAT	—	—	D/Ā	P	S	R/Ṿ	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 PIC16C62, always maintain this bit set.
 5: The IRP and RP1 bits are reserved on the PIC16C62/62A/R62, always maintain these bits clear.
 6: PIE1<7:6> and PIR1<7:6> are reserved on the PIC16C62/62A/R62, always maintain these bits clear.

TABLE 4-3: SPECIAL FUNCTION REGISTERS FOR THE PIC16C63/R63 (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 ⁽³⁾
Bank 1											
80h ⁽¹⁾	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	0000 0000
81h	OPTION	RBP _U	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
82h ⁽¹⁾	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	0000 0000
83h ⁽¹⁾	STATUS	IRP ⁽⁴⁾	RP1 ⁽⁴⁾	RP0	T ₀	P _D	Z	DC	C	0001 1xxx	000q quuu
84h ⁽¹⁾	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	—	Unimplemented								—	—
89h	—	Unimplemented								—	—
8Ah ^(1,2)	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter					---0 0000	---0 0000
8Bh ⁽¹⁾	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
8Ch	PIE1	(5)	(5)	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
8Dh	PIE2	—	—	—	—	—	—	—	CCP2IE	---- --0	---- --0
8Eh	PCON	—	—	—	—	—	—	POR	BOR	---- --qq	---- --uu
8Fh	—	Unimplemented								—	—
90h	—	Unimplemented								—	—
91h	—	Unimplemented								—	—
92h	PR2	Timer2 Period Register								1111 1111	1111 1111
93h	SSPADD	Synchronous Serial Port (I ² C mode) Address Register								0000 0000	0000 0000
94h	SSPSTAT	—	—	D/Ā	P	S	R/Ā	UA	BF	--00 0000	--00 0000
95h	—	Unimplemented								—	—
96h	—	Unimplemented								—	—
97h	—	Unimplemented								—	—
98h ⁽²⁾	TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010
99h ⁽²⁾	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 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 PIC16C63/R63, always maintain these bits clear.

5: PIE1<7:6> and PIR1<7:6> are reserved on the PIC16C63/R63, always maintain these bits clear.

PIC16C6X

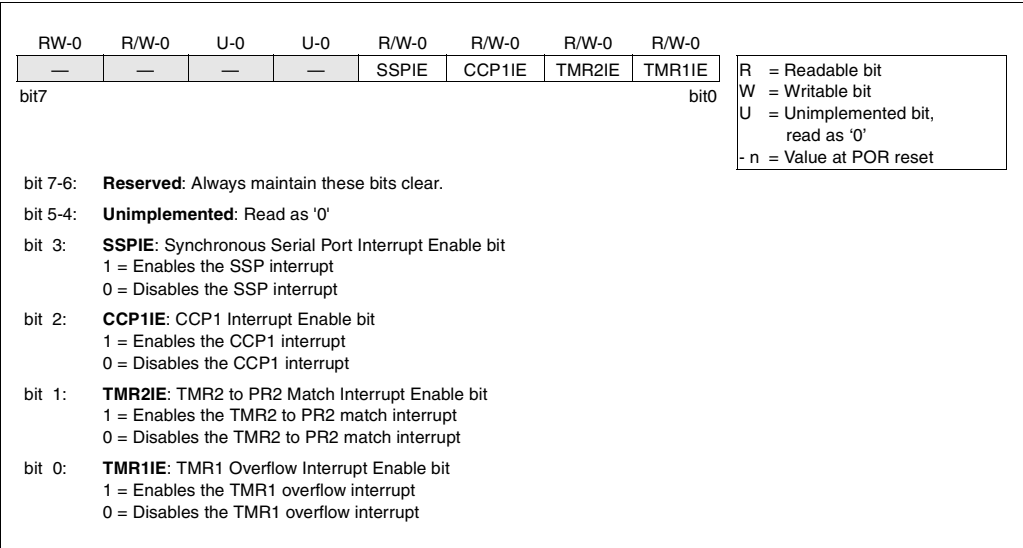
4.2.2.4 PIE1 REGISTER

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

This register contains the individual enable bits for the peripheral interrupts.

Note: Bit PEIE (INTCON<6>) must be set to enable any peripheral interrupt.

FIGURE 4-12: PIE1 REGISTER FOR PIC16C62/62A/R62 (ADDRESS 8Ch)



11.4.2 ADDRESSING I²C 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

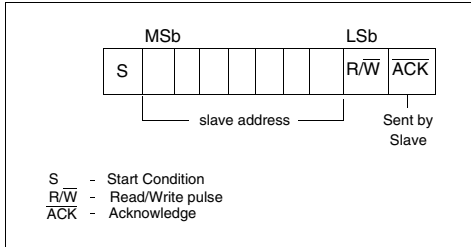
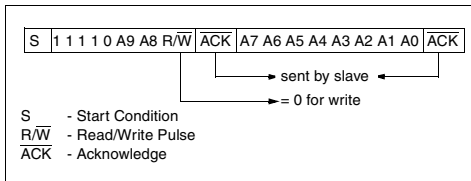


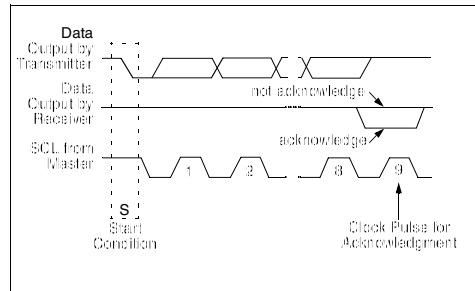
FIGURE 11-16: I²C 10-BIT ADDRESS FORMAT



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).

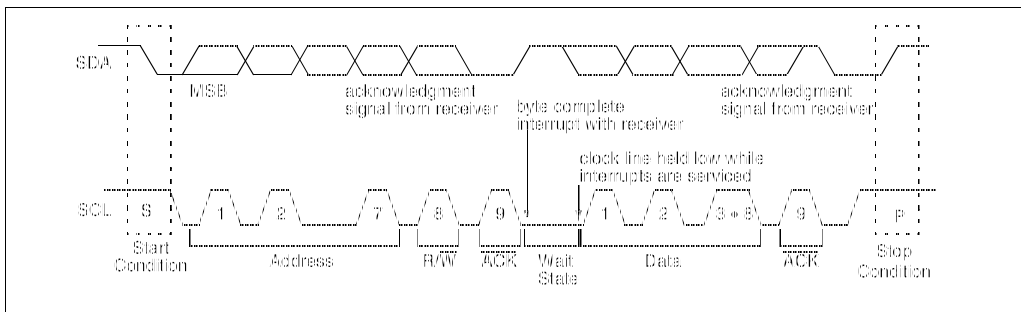
FIGURE 11-17: SLAVE-RECEIVER ACKNOWLEDGE



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



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

FIGURE 13-17: INTERRUPT LOGIC FOR PIC16C61

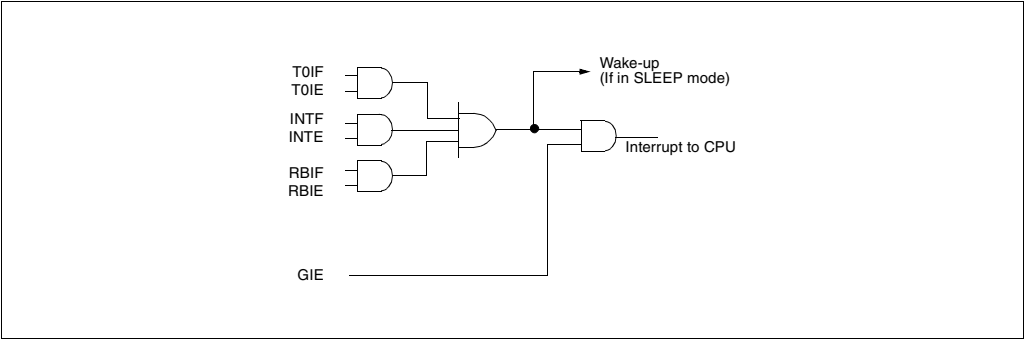
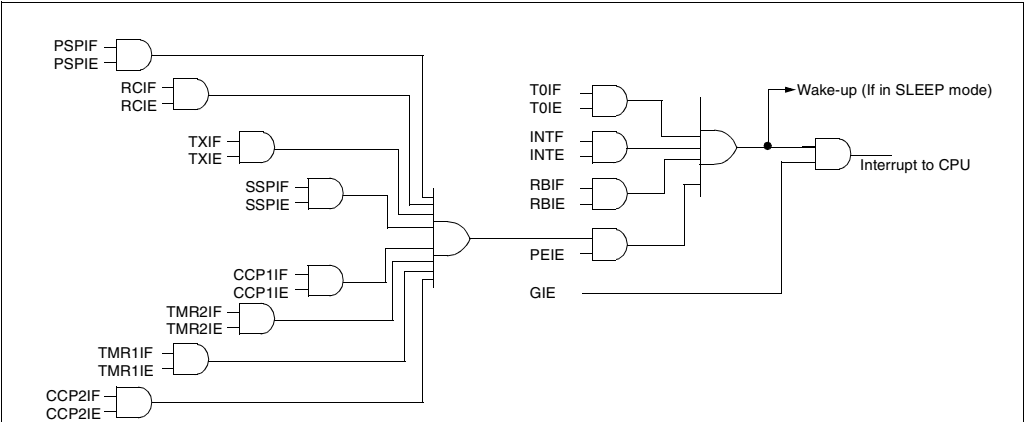


FIGURE 13-18: INTERRUPT LOGIC FOR PIC16C6X



The following table shows which devices have which interrupts.

Device	T0IF	INTF	RBIF	PSPIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	CCP2IF
PIC16C62	Yes	Yes	Yes	-	-	-	Yes	Yes	Yes	Yes	-
PIC16C62A	Yes	Yes	Yes	-	-	-	Yes	Yes	Yes	Yes	-
PIC16CR62	Yes	Yes	Yes	-	-	-	Yes	Yes	Yes	Yes	-
PIC16C63	Yes	Yes	Yes	-	Yes	Yes	Yes	Yes	Yes	Yes	Yes
PIC16CR63	Yes	Yes	Yes	-	Yes	Yes	Yes	Yes	Yes	Yes	Yes
PIC16C64	Yes	Yes	Yes	Yes	-	-	Yes	Yes	Yes	Yes	-
PIC16C64A	Yes	Yes	Yes	Yes	-	-	Yes	Yes	Yes	Yes	-
PIC16C64	Yes	Yes	Yes	Yes	-	-	Yes	Yes	Yes	Yes	-
PIC16C65	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
PIC16C65A	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
PIC16CR65	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
PIC16C66	Yes	Yes	Yes	-	Yes	Yes	Yes	Yes	Yes	Yes	Yes
PIC16C67	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes

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

```

PIC16C6X

BCF		Bit Clear f			
Syntax:	[label] BCF f,b				
Operands:	$0 \leq f \leq 127$ $0 \leq b \leq 7$				
Operation:	$0 \rightarrow (f)$				
Status Affected:	None				
Encoding:	01	00bb	bfff	ffff	
Description:	Bit 'b' in register 'f' is cleared.				
Words:	1				
Cycles:	1				
Q Cycle Activity:	Q1	Q2	Q3	Q4	
	Decode	Read register 'f'	Process data	Write register 'f'	

Example BCF FLAG_REG, 7

Before Instruction
FLAG_REG = 0x07
After Instruction
FLAG_REG = 0x47

BTFSC		Bit Test, Skip if Clear						
Syntax:	[label] BTFSC f,b							
Operands:	0 ≤ f ≤ 127 0 ≤ b ≤ 7							
Operation:	skip if (f) = 0							
Status Affected:	None							
Encoding:	<table><tr><td>01</td><td>10bb</td><td>bfff</td><td>ffff</td></tr></table>				01	10bb	bfff	ffff
01	10bb	bfff	ffff					
Description:	If bit 'b' in register 'f' is '1' then the next instruction is executed. If bit 'b', in register 'f', is '0' then the next instruction is discarded, and a NOP is executed instead, making this a 2Tcy instruction.							
Words:	1							
Cycles:	1(2)							
Q Cycle Activity:	Q1	Q2	Q3	Q4				

If Skip: (2nd Cycle)

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

Example HERE BTFSC FLAG, 1
FALSE GOTO PROCESS_CODE
TRUE •
 •

Before Instruction
PC = address HERE
After Instruction
if FLAG<1> = 0,
PC = address TRUE
if FLAG<1>=1,
PC = address FALSE

BSF		Bit Set f			
Syntax:	[label] BSF f,b				
Operands:	$0 \leq f \leq 127$ $0 \leq b \leq 7$				
Operation:	$1 \rightarrow (f)$				
Status Affected:	None				
Encoding:	01	01bb	bfff	ffff	
Description:	Bit 'b' in register 'f' is set.				
Words:	1				
Cycles:	1				
Q Cycle Activity:	Q1	Q2	Q3	Q4	
	Decode	Read register 'f'	Process data	Write register 'f'	

Example BSF FLAG_REG, 7

Before Instruction
FLAG_REG = 0x0A
After Instruction
FLAG_REG = 0x8A

NOP No Operation

Syntax: [*label*] NOP

Operands: None

Operation: No operation

Status Affected: None

Encoding:

00	0000	0xx0	0000
----	------	------	------

Description: No operation.

Words: 1

Cycles: 1

Q Cycle Activity:

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

Example NOP

RETFIE Return from Interrupt

Syntax: [*label*] RETFIE

Operands: None

Operation: TOS → PC,
1 → GIE

Status Affected: None

Encoding:

00	0000	0000	1001
----	------	------	------

Description: Return from Interrupt. Stack is POPed and Top of Stack (TOS) is loaded in the PC. Interrupts are enabled by setting Global Interrupt Enable bit, GIE (INTCON<7>). This is a two cycle instruction.

Words: 1

Cycles: 2

Q Cycle Activity:

	Q1	Q2	Q3	Q4
1st Cycle	Decode	No-Operation	Set the GIE bit	Pop from the Stack
2nd Cycle	No-Operation	No-Operation	No-Operation	No-Operation

Example RETFIE

After Interrupt

PC = TOS

GIE = 1

OPTION Load Option Register

Syntax: [*label*] OPTION

Operands: None

Operation: (W) → OPTION

Status Affected: None

Encoding:

00	0000	0110	0010
----	------	------	------

Description: The contents of the W register are loaded in the OPTION register. This instruction is supported for code compatibility with PIC16C5X products. Since OPTION is a readable/writable register, the user can directly address it.

Words: 1

Cycles: 1

Example

To maintain upward compatibility with future PIC16CXX products, do not use this instruction.

RLF Rotate Left f through Carry

Syntax: `[label] RLF f,d`

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

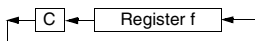
Operation: See description below

Status Affected: C

Encoding:

00	1101	dfff	ffff
----	------	------	------

Description: The contents of register 'f' are rotated one bit to the left through the Carry Flag. If 'd' is 0 the result is placed in the W register. If 'd' is 1 the result is stored back in register 'f'.



Words: 1

Cycles: 1

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

Example

RLF REG1, 0

Before Instruction
 REG1 = 1110 0110
 C = 0

After Instruction
 REG1 = 1110 0110
 W = 1100 1100
 C = 1

RRF Rotate Right f through Carry

Syntax: `[label] RRF f,d`

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

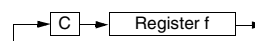
Operation: See description below

Status Affected: C

Encoding:

00	1100	dfff	ffff
----	------	------	------

Description: The contents of register 'f' are rotated one bit to the right through the Carry Flag. If 'd' is 0 the result is placed in the W register. If 'd' is 1 the result is placed back in register 'f'.



Words: 1

Cycles: 1

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

Example

RRF REG1, 0

Before Instruction
 REG1 = 1110 0110
 C = 0

After Instruction
 REG1 = 1110 0110
 W = 0111 0011
 C = 0

15.5 Timing Diagrams and Specifications

FIGURE 15-2: EXTERNAL CLOCK TIMING

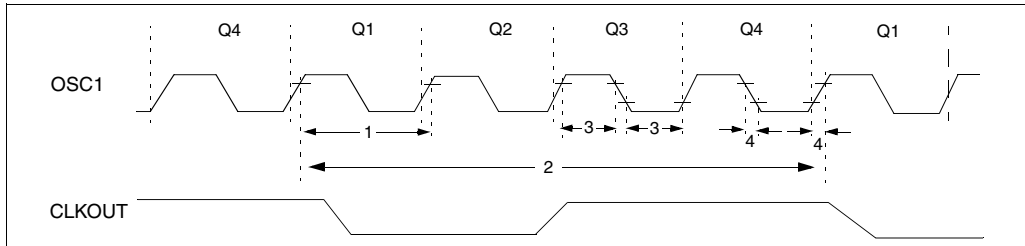


TABLE 15-2: EXTERNAL CLOCK TIMING REQUIREMENTS

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

17.0 ELECTRICAL CHARACTERISTICS FOR PIC16C62/64

Absolute Maximum Ratings †

Ambient temperature under bias	-55°C to +85°C
Storage temperature	-65°C to +150°C
Voltage on any pin with respect to VSS (except VDD, MCLR, and RA4)	-0.3V to (VDD + 0.3V)
Voltage on VDD with respect to VSS	-0.3V to +7.5V
Voltage on MCLR with respect to VSS (Note 2)	0V to +14V
Voltage on RA4 with respect to VSS	0V to +14V
Total power dissipation (Note 1)	1.0W
Maximum current out of VSS pin	300 mA
Maximum current into VDD pin	250 mA
Input clamp current, I _{IK} (V _I < 0 or V _I > VDD)	±20 mA
Output clamp current, I _{OK} (V _O < 0 or V _O > VDD)	±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, PORTB, and PORTE* (combined)	200 mA
Maximum current sourced by PORTA, PORTB, and PORTE* (combined)	200 mA
Maximum current sunk by PORTC and PORTD* (combined)	200 mA
Maximum current sourced by PORTC and PORTD* (combined)	200 mA

* PORTD and PORTE not available on the PIC16C62.

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

Note 2: Voltage spikes below VSS at the MCLR pin, inducing currents greater than 80 mA, may cause latch-up. Thus, a series resistor of 50-100Ω should be used when applying a “low” level to the MCLR pin rather than pulling this pin directly to VSS.

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

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

OSC	PIC16C62-04 PIC16C64-04	PIC16C62-10 PIC16C64-10	PIC16C62-20 PIC16C64-20	PIC16LC62-04 PIC16LC64-04	JW Devices
RC	VDD: 4.0V to 6.0V IDD: 3.8 mA max. at 5.5V IPD: 21 µA max. at 4V Freq:4 MHz max.	VDD: 4.5V to 5.5V IDD: 2.0 mA typ. at 5.5V IPD: 1.5 µA typ. at 4V Freq:4 MHz max.	VDD: 4.5V to 5.5V IDD: 2.0 mA typ. at 5.5V IPD: 1.5 µA typ. at 4V Freq:4 MHz max.	VDD: 3.0V to 6.0V IDD: 3.8 mA max. at 3.0V IPD: 13.5 µA max. at 3V Freq: 4 MHz max.	VDD: 4.0V to 6.0V IDD: 3.8 mA max. at 5.5V IPD: 21 µA max. at 4V Freq:4 MHz max.
XT	VDD: 4.0V to 6.0V IDD: 3.8 mA max. at 5.5V IPD: 21 µA max. at 4V Freq:4 MHz max.	VDD: 4.5V to 5.5V IDD: 2.0 mA typ. at 5.5V IPD: 1.5 µA typ. at 4V Freq:4 MHz max.	VDD: 4.5V to 5.5V IDD: 2.0 mA typ. at 5.5V IPD: 1.5 µA typ. at 4V Freq:4 MHz max.	VDD: 3.0V to 6.0V IDD: 3.8 mA max. at 3.0V IPD: 13.5 µA max. at 3.0V Freq: 4 MHz max.	VDD: 4.0V to 6.0V IDD: 3.8 mA max. at 5.5V IPD: 21 µA max. at 4V Freq:4 MHz max.
HS	VDD: 4.5V to 5.5V IDD: 13.5 mA typ. at 5.5V IPD: 1.5 µA typ. at 4.5V Freq:4 MHz max.	VDD: 4.5V to 5.5V IDD: 15 mA max. at 5.5V IPD: 1.5 µA typ. at 4.5V Freq: 10 MHz max.	VDD: 4.5V to 5.5V IDD: 30 mA max. at 5.5V IPD: 1.5 µA typ. at 4.5V Freq: 20 MHz max.	Not recommended for use in HS mode	VDD: 4.5V to 5.5V IDD: 30 mA max. at 5.5V IPD: 1.5 µA typ. at 4.5V Freq: 20 MHz max.
LP	VDD: 4.0V to 6.0V IDD: 52.5 µA typ. at 32 kHz, 4.0V IPD: 0.9 µA typ. at 4.0V Freq:200 kHz max.	Not recommended for use in LP mode	Not recommended for use in LP mode	VDD: 3.0V to 6.0V IDD: 48 µA max. at 32 kHz, 3.0V IPD: 13.5 µA max. at 3.0V Freq:200 kHz max.	VDD: 3.0V to 6.0V IDD: 48 µA max. at 32 kHz, 3.0V IPD:13.5 µA max. at 3.0V Freq:200 kHz max.

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

PIC16C6X

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

FIGURE 21-6: TIMER0 AND TIMER1 EXTERNAL CLOCK TIMINGS

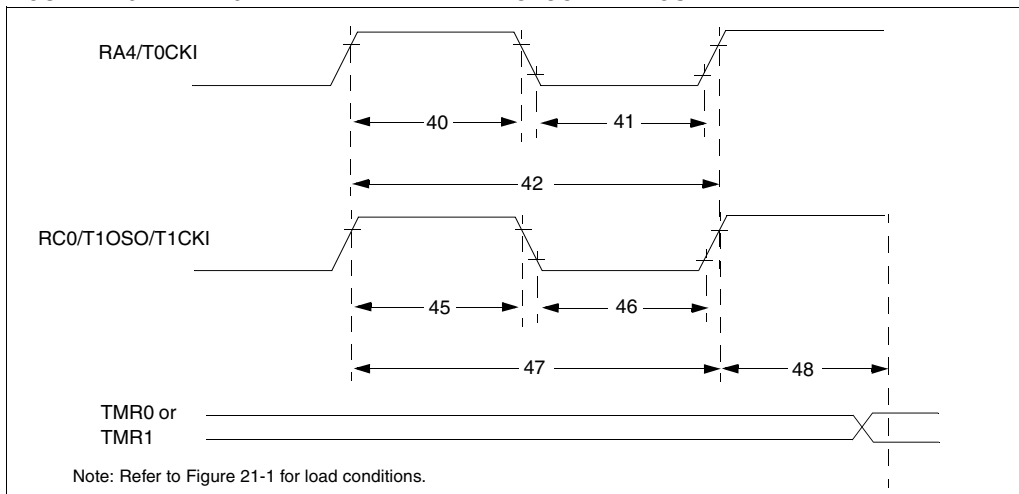


TABLE 21-5: TIMER0 AND TIMER1 EXTERNAL CLOCK REQUIREMENTS

Param No.	Sym	Characteristic		Min	Typ†	Max	Units	Conditions	
40*	Tt0H	T0CKI High Pulse Width	No Prescaler	0.5Tcy + 20	—	—	ns	Must also meet parameter 42	
			With Prescaler	10	—	—	ns		
41*	Tt0L	T0CKI Low Pulse Width	No Prescaler	0.5Tcy + 20	—	—	ns	Must also meet parameter 42	
			With Prescaler	10	—	—	ns		
42*	Tt0P	T0CKI Period	No Prescaler	Tcy + 40	—	—	ns		
			With Prescaler	Greater of: 20 or Tcy + 40 N	—	—	ns	N = prescale value (2, 4, ..., 256)	
45*	Tt1H	T1CKI High Time	Synchronous, Prescaler = 1	0.5Tcy + 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	0.5Tcy + 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	PIC16C6X	Greater of: 30 OR Tcy + 40 N	—	—	ns	N = prescale value (1, 2, 4, 8)
			PIC16LC6X	Greater of: 50 OR Tcy + 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		2Tosc	—	7Tosc	—		

* 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

TXSTA	
Diagram	105
Section	105
Summary	31, 33
W	9
Special Function Registers, Initialization	
Conditions	132
Special Function Registers, Reset Conditions	131
Special Function Register Summary	24, 26, 28, 30, 32
File Maps	21
Resets	128
ROM	7
RP0 bit	20, 35
RP1	35
RX9	106
RX9D	106
S	
S	84, 89
SCI - See Universal Synchronous Asynchronous Receiver Transmitter (USART)	
SCK	86
SCL	100
SDI	86
SDO	86
Serial Port Enable bit, SPEN	106
Serial Programming	142
Serial Programming, Block Diagram	142
Serialized Quick-Turnaround-Production	7
Single Receive Enable bit, SREN	106
Slave Mode	
SCL	100
SDA	100
SLEEP Mode	123, 141
SMP	89
Software Simulator (MPSIM)	161
SPBRG	25, 27, 29, 31, 33, 34
Special Features, Section	123
SPEN	106
SPI	
Block Diagram	86, 91
Master Mode	92
Master Mode Timing	93
Mode	86
Serial Clock	91
Serial Data In	91
Serial Data Out	91
Slave Mode Timing	94
Slave Mode Timing Diagram	93
Slave Select	91
SPI clock	92
SPI Mode	91
SSPCON	90
SSPSTAT	89
SPI Clock Edge Select bit, CKE	89
SPI Data Input Sample Phase Select bit, SMP	89
SPI Mode	86
SREN	106
SS	86
SSP	
Module Overview	83
Section	83
SSPBUF	92
SSPCON	90
SSPSR	92
SSPSTAT	89

SSP in I ² C Mode - See I ² C	
SSPADD	25, 27, 29, 31, 33, 34, 99
SSPBUF	24, 26, 28, 30, 32, 34, 99
SSPCON	24, 26, 28, 30, 32, 34, 85, 90
SSPEN	85, 90
SSPIE	38
SSPIF	41
SSPM3:SSPM0	85, 90
SSPOV	85, 90, 100
SSPSTAT	25, 27, 29, 31, 33, 34, 84, 99
SSPSTAT Register	89
Stack	48
Start bit, S	84, 89
STATUS	24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34
Status bits	130, 131
Status Bits During Various Resets	131
Stop bit, P	84, 89
Switching Prescalers	69
SYNC, USART Mode Select bit, SYNC	105
Synchronizing Clocks, TMR0	67
Synchronous Serial Port (SSP)	
Block Diagram, SPI Mode	86
SPI Master/Slave Diagram	87
SPI Mode	86
Synchronous Serial Port Enable bit, SSPEN	85, 90
Synchronous Serial Port Interrupt Enable bit, SSPIE	38
Synchronous Serial Port Interrupt Flag bit, SSPIF	41
Synchronous Serial Port Mode Select bits,	
SSPM3:SSPM0	85, 90
Synchronous Serial Port Module	83
Synchronous Serial Port Status Register	89
T	
T0CS	36
TOIE	37
TOIF	37
T0SE	36
T1CKPS1:T1CKPS0	71
T1CON	24, 26, 28, 30, 32, 34
T1OSCEN	71
T1SYNC	71
T2CKPS1:T2CKPS0	75
T2CON	24, 26, 28, 30, 32, 34, 75
Time-out	130
Time-out bit	35
Time-out Sequence	130
Timer Modules	
Overview, all	63
Timer0	
Block Diagram	65
Counter Mode	65
External Clock	67
Interrupt	65
Overview	63
Prescaler	68
Section	65
Timer Mode	65
Timing Diagram Timing Diagrams	
Timer0	65
TMR0 register	65
Timer1	
Block Diagram	72
Capacitor Selection	73
Counter Mode, Asynchronous	73
Counter Mode, Synchronous	72
External Clock	73
Oscillator	73

Figure 11-2:	SSPCON: Sync Serial Port Control Register (Address 14h)	85	Figure 13-2:	Configuration Word for PIC16C62/64/65	124
Figure 11-3:	SSP Block Diagram (SPI Mode)	86	Figure 13-3:	Configuration Word for PIC16C62A/R62/63/R63/64A/R64/65A/R65/66/67	124
Figure 11-4:	SPI Master/Slave Connection	87	Figure 13-4:	Crystal/Ceramic Resonator Operation (HS, XT or LP OSC Configuration)	125
Figure 11-5:	SPI Mode Timing, Master Mode or Slave Mode w/o SS Control	88	Figure 13-5:	External Clock Input Operation (HS, XT or LP OSC Configuration)	125
Figure 11-6:	SPI Mode Timing, Slave Mode with SS Control	88	Figure 13-6:	External Parallel Resonant Crystal Oscillator Circuit	127
Figure 11-7:	SSPSTAT: Sync Serial Port Status Register (Address 94h)(PIC16C66/67)	89	Figure 13-7:	External Series Resonant Crystal Oscillator Circuit	127
Figure 11-8:	SSPCON: Sync Serial Port Control Register (Address 14h)(PIC16C66/67)	90	Figure 13-8:	RC Oscillator Mode	127
Figure 11-9:	SSP Block Diagram (SPI Mode) (PIC16C66/67)	91	Figure 13-9:	Simplified Block Diagram of On-chip Reset Circuit	128
Figure 11-10:	SPI Master/Slave Connection (PIC16C66/67)	92	Figure 13-10:	Brown-out Situations	129
Figure 11-11:	SPI Mode Timing, Master Mode (PIC16C66/67)	93	Figure 13-11:	Time-out Sequence on Power-up (MCLR not Tied to VDD): Case 1	134
Figure 11-12:	SPI Mode Timing (Slave Mode With CKE = 0) (PIC16C66/67)	93	Figure 13-12:	Time-out Sequence on Power-up (MCLR Not Tied To VDD): Case 2	134
Figure 11-13:	SPI Mode Timing (Slave Mode With CKE = 1) (PIC16C66/67)	94	Figure 13-13:	Time-out Sequence on Power-up (MCLR Tied to VDD)	134
Figure 11-14:	Start and Stop Conditions	95	Figure 13-14:	External Power-on Reset Circuit (For Slow VDD Power-up)	135
Figure 11-15:	7-bit Address Format	96	Figure 13-15:	External Brown-out Protection Circuit 1	135
Figure 11-16:	I ² C 10-bit Address Format	96	Figure 13-16:	External Brown-out Protection Circuit 2	135
Figure 11-17:	Slave-receiver Acknowledge	96	Figure 13-17:	Interrupt Logic for PIC16C61	137
Figure 11-18:	Data Transfer Wait State	96	Figure 13-18:	Interrupt Logic for PIC16C6X	137
Figure 11-19:	Master-transmitter Sequence	97	Figure 13-19:	INT Pin Interrupt Timing	138
Figure 11-20:	Master-receiver Sequence	97	Figure 13-20:	Watchdog Timer Block Diagram	140
Figure 11-21:	Combined Format	97	Figure 13-21:	Summary of Watchdog Timer Registers	140
Figure 11-22:	Multi-master Arbitration (Two Masters)	98	Figure 13-22:	Wake-up from Sleep Through Interrupt	142
Figure 11-23:	Clock Synchronization	98	Figure 13-23:	Typical In-circuit Serial Programming Connection	142
Figure 11-24:	SSP Block Diagram (I ² C Mode)	99	Figure 14-1:	General Format for Instructions	143
Figure 11-25:	I ² C Waveforms for Reception (7-bit Address)	101	Figure 16-1:	Load Conditions for Device Timing Specifications	168
Figure 11-26:	I ² C Waveforms for Transmission (7-bit Address)	102	Figure 16-2:	External Clock Timing	169
Figure 11-27:	Operation of the I ² C Module in IDLE_MODE, RCV_MODE or XMIT_MODE	104	Figure 16-3:	CLKOUT and I/O Timing	170
Figure 12-1:	TXSTA: Transmit Status and Control Register (Address 98h)	105	Figure 16-4:	Reset, Watchdog Timer, Oscillator Start-up Timer and Power-up Timer Timing	171
Figure 12-2:	RCSTA: Receive Status and Control Register (Address 18h)	106	Figure 16-5:	Timer0 External Clock Timings	172
Figure 12-3:	RX Pin Sampling Scheme (BRGH = 0) (PIC16C63/R63/65/65A/R65)	110	Figure 17-1:	Typical RC Oscillator Frequency vs. Temperature	173
Figure 12-4:	RX Pin Sampling Scheme (BRGH = 1) (PIC16C63/R63/65/65A/R65)	110	Figure 17-2:	Typical RC Oscillator Frequency vs. VDD	174
Figure 12-5:	RX Pin Sampling Scheme (BRGH = 1) (PIC16C63/R63/65/65A/R65)	110	Figure 17-3:	Typical RC Oscillator Frequency vs. VDD	174
Figure 12-6:	RX Pin Sampling Scheme (BRGH = 0 or = 1) (PIC16C66/67)	111	Figure 17-4:	Typical RC Oscillator Frequency vs. VDD	174
Figure 12-7:	USART Transmit Block Diagram	112	Figure 17-5:	Typical IPD vs. VDD Watchdog Timer Disabled 25°C	174
Figure 12-8:	Asynchronous Master Transmission	113	Figure 17-6:	Typical IPD vs. VDD Watchdog Timer Enabled 25°C	175
Figure 12-9:	Asynchronous Master Transmission (Back to Back)	113	Figure 17-7:	Maximum IPD vs. VDD Watchdog Disabled	175
Figure 12-10:	USART Receive Block Diagram	114	Figure 17-8:	Maximum IPD vs. VDD Watchdog Enabled*	176
Figure 12-11:	Asynchronous Reception	114	Figure 17-9:	VTH (Input Threshold Voltage) of I/O Pins vs. VDD	176
Figure 12-12:	Synchronous Transmission	117			
Figure 12-13:	Synchronous Transmission through TXEN	117			
Figure 12-14:	Synchronous Reception (Master Mode, SREN)	119			
Figure 13-1:	Configuration Word for PIC16C61	123			

PIC16C6X

Figure 17-10:	V _{IH} , V _{IL} of $\overline{\text{MCLR}}$, T0CKI and OSC1 (in RC Mode) vs. V _{DD}	177	Figure 20-7:	Parallel Slave Port Timing	226
Figure 17-11:	V _{TH} (Input Threshold Voltage) of OSC1 Input (in XT, HS, and LP Modes) vs. V _{DD}	177	Figure 20-8:	SPI Mode Timing	227
Figure 17-12:	Typical I _{DD} vs. Frequency (External Clock, 25°C)	178	Figure 20-9:	I ² C Bus Start/Stop Bits Timing	228
Figure 17-13:	Maximum I _{DD} vs. Frequency (External Clock, -40° to +85°C)	178	Figure 20-10:	I ² C Bus Data Timing	229
Figure 17-14:	Maximum I _{DD} vs. Frequency (External Clock, -55° to +125°C)	179	Figure 20-11:	USART Synchronous Transmission (Master/Slave) Timing	230
Figure 17-15:	WDT Timer Time-out Period vs. V _{DD}	179	Figure 20-12:	USART Synchronous Receive (Master/Slave) Timing	230
Figure 17-16:	Transconductance (gm) of HS Oscillator vs. V _{DD}	179	Figure 21-1:	Load Conditions for Device Timing Specifications	236
Figure 17-17:	Transconductance (gm) of LP Oscillator vs. V _{DD}	180	Figure 21-2:	External Clock Timing	237
Figure 17-18:	Transconductance (gm) of XT Oscillator vs. V _{DD}	180	Figure 21-3:	CLKOUT and I/O Timing	238
Figure 17-19:	IOH vs. VOH, V _{DD} = 3V	180	Figure 21-4:	Reset, Watchdog Timer, Oscillator Start-up Timer and Power-up Timer Timing	239
Figure 17-20:	IOH vs. VOH, V _{DD} = 5V	180	Figure 21-5:	Brown-out Reset Timing	239
Figure 17-21:	IOL vs. VOL, V _{DD} = 3V	181	Figure 21-6:	Timer0 and Timer1 External Clock Timings	240
Figure 17-22:	IOL vs. VOL, V _{DD} = 5V	181	Figure 21-7:	Capture/Compare/PWM Timings (CCP1 and CCP2)	241
Figure 18-1:	Load Conditions for Device Timing Specifications	188	Figure 21-8:	Parallel Slave Port Timing (PIC16C65A)	242
Figure 18-2:	External Clock Timing	189	Figure 21-9:	SPI Mode Timing	243
Figure 18-3:	CLKOUT and I/O Timing	190	Figure 21-10:	I ² C Bus Start/Stop Bits Timing	244
Figure 18-4:	Reset, Watchdog Timer, Oscillator Start-up Timer and Power-up Timer Timing	191	Figure 21-11:	I ² C Bus Data Timing	245
Figure 18-5:	Timer0 and Timer1 External Clock Timings	192	Figure 21-12:	USART Synchronous Transmission (Master/Slave) Timing	246
Figure 18-6:	Capture/Compare/PWM Timings (CCP1)	193	Figure 21-13:	USART Synchronous Receive (Master/Slave) Timing	246
Figure 18-7:	Parallel Slave Port Timing (PIC16C64)	194	Figure 22-1:	Load Conditions for Device Timing Specifications	252
Figure 18-8:	SPI Mode Timing	195	Figure 22-2:	External Clock Timing	253
Figure 18-9:	I ² C Bus Start/Stop Bits Timing	196	Figure 22-3:	CLKOUT and I/O Timing	254
Figure 18-10:	I ² C Bus Data Timing	197	Figure 22-4:	Reset, Watchdog Timer, Oscillator Start-up Timer and Power-up Timer Timing	255
Figure 19-1:	Load Conditions for Device Timing Specifications	204	Figure 22-5:	Brown-out Reset Timing	255
Figure 19-2:	External Clock Timing	205	Figure 22-6:	Timer0 and Timer1 External Clock Timings	256
Figure 19-3:	CLKOUT and I/O Timing	206	Figure 22-7:	Capture/Compare/PWM Timings (CCP1 and CCP2)	257
Figure 19-4:	Reset, Watchdog Timer, Oscillator Start-up Timer and Power-up Timer Timing	207	Figure 22-8:	Parallel Slave Port Timing (PIC16C65)	258
Figure 19-5:	Brown-out Reset Timing	207	Figure 22-9:	SPI Mode Timing	259
Figure 19-6:	Timer0 and Timer1 External Clock Timings	208	Figure 22-10:	I ² C Bus Start/Stop Bits Timing	260
Figure 19-7:	Capture/Compare/PWM Timings (CCP1)	209	Figure 22-11:	I ² C Bus Data Timing	261
Figure 19-8:	Parallel Slave Port Timing (PIC16C64A/R64)	210	Figure 22-12:	USART Synchronous Transmission (Master/Slave) Timing	262
Figure 19-9:	SPI Mode Timing	211	Figure 22-13:	USART Synchronous Receive (Master/Slave) Timing	262
Figure 19-10:	I ² C Bus Start/Stop Bits Timing	212	Figure 23-1:	Load Conditions for Device Timing Specifications	268
Figure 19-11:	I ² C Bus Data Timing	213	Figure 23-2:	External Clock Timing	269
Figure 20-1:	Load Conditions for Device Timing Specifications	220	Figure 23-3:	CLKOUT and I/O Timing	270
Figure 20-2:	External Clock Timing	221	Figure 23-4:	Reset, Watchdog Timer, Oscillator Start-up Timer and Power-up Timer Timing	271
Figure 20-3:	CLKOUT and I/O Timing	222	Figure 23-5:	Brown-out Reset Timing	271
Figure 20-4:	Reset, Watchdog Timer, Oscillator Start-up Timer and Power-up Timer Timing	223	Figure 23-6:	Timer0 and Timer1 External Clock Timings	272
Figure 20-5:	Timer0 and Timer1 External Clock Timings	224	Figure 23-7:	Capture/Compare/PWM Timings (CCP1 and CCP2)	273
Figure 20-6:	Capture/Compare/PWM Timings (CCP1 and CCP2)	225	Figure 23-8:	Parallel Slave Port Timing (PIC16C67)	274
			Figure 23-9:	SPI Master Mode Timing (CKE = 0)	275
			Figure 23-10:	SPI Master Mode Timing (CKE = 1)	275
			Figure 23-11:	SPI Slave Mode Timing (CKE = 0)	276

READER RESPONSE

Please list the following information, and use this outline to provide us with your comments about this Data Sheet.

Application (optional):

Would you like a reply? ____Y ____N

Device: **PIC16C6X** Literature Number: **DS30234E**

Questions:

1. What are the best features of this document?

2. How does this document meet your hardware and software development needs?

3. Do you find the organization of this data sheet easy to follow? If not, why?

4. What additions to the data sheet do you think would enhance the structure and subject?

5. What deletions from the data sheet could be made without affecting the overall usefulness?

6. Is there any incorrect or misleading information (what and where)?

7. How would you improve this document?

8. How would you improve our software, systems, and silicon products?