



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	Obsolete
Core Processor	PIC
Core Size	8-Bit
Speed	4MHz
Connectivity	I ² C, SPI
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	22
Program Memory Size	3.5KB (2K x 14)
Program Memory Type	OTP
EEPROM Size	-
RAM Size	128 x 8
Voltage - Supply (Vcc/Vdd)	2.5V ~ 6V
Data Converters	-
Oscillator Type	External
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Surface Mount
Package / Case	28-SOIC (0.295", 7.50mm Width)
Supplier Device Package	28-SOIC
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16lc62at-04-so

PIC16C6X

TABLE 1-1: PIC16C6X FAMILY OF DEVICES

		PIC16C61	PIC16C62A	PIC16CR62	PIC16C63	PIC16CR63
Clock	Maximum Frequency of Operation (MHz)	20	20	20	20	20
Memory	EPROM Program Memory (x14 words)	1K	2K	—	4K	—
	ROM Program Memory (x14 words)	—	—	2K	—	4K
	Data Memory (bytes)	36	128	128	192	192
Peripherals	Timer Module(s)	TMR0	TMR0, TMR1, TMR2	TMR0, TMR1, TMR2	TMR0, TMR1, TMR2	TMR0, TMR1, TMR2
	Capture/Compare/PWM Module(s)	—	1	1	2	2
	Serial Port(s) (SPI/I ² C, USART)	—	SPI/I ² C	SPI/I ² C	SPI/I ² C, USART	SPI/I ² C, USART
	Parallel Slave Port	—	—	—	—	—
Features	Interrupt Sources	3	7	7	10	10
	I/O Pins	13	22	22	22	22
	Voltage Range (Volts)	3.0-6.0	2.5-6.0	2.5-6.0	2.5-6.0	2.5-6.0
	In-Circuit Serial Programming	Yes	Yes	Yes	Yes	Yes
	Brown-out Reset	—	Yes	Yes	Yes	Yes
	Packages	18-pin DIP, SO	28-pin SDIP, SOIC, SSOP	28-pin SDIP, SOIC, SSOP	28-pin SDIP, SOIC	28-pin SDIP, SOIC

		PIC16C64A	PIC16CR64	PIC16C65A	PIC16CR65	PIC16C66	PIC16C67
Clock	Maximum Frequency of Operation (MHz)	20	20	20	20	20	20
Memory	EPROM Program Memory (x14 words)	2K	—	4K	—	8K	8K
	ROM Program Memory (x14 words)	—	2K	—	4K	—	—
	Data Memory (bytes)	128	128	192	192	368	368
Peripherals	Timer Module(s)	TMR0, TMR1, TMR2	TMR0, TMR1, TMR2	TMR0, TMR1, TMR2	TMR0, TMR1, TMR2	TMR0, TMR1, TMR2	TMR0, TMR1, TMR2
	Capture/Compare/PWM Module(s)	1	1	2	2	2	2
	Serial Port(s) (SPI/I ² C, USART)	SPI/I ² C	SPI/I ² C	SPI/I ² C, USART	SPI/I ² C, USART	SPI/I ² C, USART	SPI/I ² C, USART
	Parallel Slave Port	Yes	Yes	Yes	Yes	—	Yes
Features	Interrupt Sources	8	8	11	11	10	11
	I/O Pins	33	33	33	33	22	33
	Voltage Range (Volts)	2.5-6.0	2.5-6.0	2.5-6.0	2.5-6.0	2.5-6.0	2.5-6.0
	In-Circuit Serial Programming	Yes	Yes	Yes	Yes	Yes	Yes
	Brown-out Reset	Yes	Yes	Yes	Yes	Yes	Yes
	Packages	40-pin DIP; 44-pin PLCC, MQFP, TQFP	40-pin DIP; 44-pin PLCC, MQFP, TQFP	40-pin DIP; 44-pin PLCC, MQFP, TQFP	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 PIC16C6X Family devices use serial programming with clock pin RB6 and data pin RB7.

EXAMPLE 10-2: PWM PERIOD AND DUTY CYCLE CALCULATION

Desired PWM frequency is 78.125 kHz,

Fosc = 20 MHz

TMR2 prescale = 1

$$1/78.125 \text{ kHz} = [(PR2) + 1] \cdot 4 \cdot 1/20 \text{ MHz} \cdot 1$$

$$12.8 \mu\text{s} = [(PR2) + 1] \cdot 4 \cdot 50 \text{ ns} \cdot 1$$

$$PR2 = 63$$

Find the maximum resolution of the duty cycle that can be used with a 78.125 kHz frequency and 20 MHz oscillator:

$$1/78.125 \text{ kHz} = 2^{\text{PWM RESOLUTION}} \cdot 1/20 \text{ MHz} \cdot 1$$

$$12.8 \mu\text{s} = 2^{\text{PWM RESOLUTION}} \cdot 50 \text{ ns} \cdot 1$$

$$256 = 2^{\text{PWM RESOLUTION}}$$

$$\log(256) = (\text{PWM Resolution}) \cdot \log(2)$$

$$8.0 = \text{PWM Resolution}$$

At most, an 8-bit resolution duty cycle can be obtained from a 78.125 kHz frequency and a 20 MHz oscillator, i.e., $0 \leq \text{CCPR1L:CCP1CON}\langle 5:4 \rangle \leq 255$. Any value greater than 255 will result in a 100% duty cycle.

In order to achieve higher resolution, the PWM frequency must be decreased. In order to achieve higher PWM frequency, the resolution must be decreased.

Table 10-3 lists example PWM frequencies and resolutions for Fosc = 20 MHz. The TMR2 prescaler and PR2 values are also shown.

10.3.3 SET-UP FOR PWM OPERATION

The following steps should be taken when configuring the CCP module for PWM operation:

1. Set the PWM period by writing to the PR2 register.
2. Set the PWM duty cycle by writing to the CCPR1L register and CCP1CON<5:4> bits.
3. Make the CCP1 pin an output by clearing the TRISC<2> bit.
4. Set the TMR2 prescale value and enable Timer2 by writing to T2CON.
5. Configure the CCP1 module for PWM operation.

TABLE 10-3: EXAMPLE PWM FREQUENCIES AND RESOLUTIONS AT 20 MHz

PWM Frequency	1.22 kHz	4.88 kHz	19.53 kHz	78.12 kHz	156.3 kHz	208.3 kHz
Timer Prescaler (1, 4, 16)	16	4	1	1	1	1
PR2 Value	0xFF	0xFF	0xFF	0x3F	0x1F	0x17
Maximum Resolution (bits)	10	10	10	8	7	5.5

TABLE 10-4: REGISTERS ASSOCIATED WITH TIMER1, CAPTURE AND COMPARE

Add	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
0Bh,8Bh 10Bh,18Bh	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBFIF	0000 000x	0000 000u
0Ch	PIR1	PSPIF ⁽²⁾	⁽³⁾	RCIF ⁽¹⁾	TXIF ⁽¹⁾	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
0Dh ⁽⁴⁾	PIR2	—	—	—	—	—	—	—	CCP2IF	---- --0	---- --0
8Ch	PIE1	PSPIE ⁽²⁾	⁽³⁾	RCIE ⁽¹⁾	TXIE ⁽¹⁾	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
8Dh ⁽⁴⁾	PIE2	—	—	—	—	—	—	—	CCP2IE	---- --0	---- --0
87h	TRISC	PORTC Data Direction register								1111 1111	1111 1111
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	T1SYNCR	TMR1CS	TMR1ON	--00 0000	--uu uuuu
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
1Bh ⁽⁴⁾	CCPR2L	Capture/Compare/PWM2 (LSB)								xxxx xxxx	uuuu uuuu
1Ch ⁽⁴⁾	CCPR2H	Capture/Compare/PWM2 (MSB)								xxxx xxxx	uuuu uuuu
1Dh ⁽⁴⁾	CCP2CON	—	—	CCP2X	CCP2Y	CCP2M3	CCP2M2	CCP2M1	CCP2M0	--00 0000	--00 0000

Legend: x = unknown, u = unchanged, - = unimplemented locations read as '0'. Shaded cells are not used in these modes.

Note 1: These bits are associated with the USART module, which is implemented on the PIC16C63/R63/65/65A/R65/66/67 only.

2: Bits PSPIE and PSPIF are reserved on the PIC16C62/62A/R62/63/R63/66, always maintain these bits clear.

3: The PIR1<6> and PIE1<6> bits are reserved, always maintain these bits clear.

4: These registers are associated with the CCP2 module, which is only implemented on the PIC16C63/R63/65/65A/R65/66/67.

11.5.1 SLAVE MODE

In slave mode, the SCL and SDA pins must be configured as inputs (TRISC<4:3> set). The SSP module will override the input state with the output data when required (slave-transmitter).

When an address is matched or the data transfer after an address match is received, the hardware automatically will generate the acknowledge (ACK) pulse, and then load the SSPBUF register with the received value currently in the SSPSR register.

There are certain conditions that will cause the SSP module not to give this ACK pulse. These are if either (or both):

- The buffer full bit BF (SSPSTAT<0>) was set before the transfer was received.
- The overflow bit SSPOV (SSPCON<6>) was set before the transfer was received.

In this case, the SSPSR register value is not loaded into the SSPBUF, but bit SSPIF (PIR1<3>) is set. Table 11-4 shows what happens when a data transfer byte is received, given the status of bits BF and SSPOV. The shaded cells show the condition where user software did not properly clear the overflow condition. Flag bit BF is cleared by reading the SSPBUF register while bit SSPOV is cleared through software.

The SCL clock input must have a minimum high and low for proper operation. The high and low times of the I²C specification as well as the requirement of the SSP module is shown in timing parameter #100 and parameter #101.

11.5.1.1 ADDRESSING

Once the SSP module has been enabled, it waits for a START condition to occur. Following the START condition, the 8-bits are shifted into the SSPSR register. All incoming bits are sampled with the rising edge of the clock (SCL) line. The value of register SSPSR<7:1> is compared to the value of the SSPADD register. The

address is compared on the falling edge of the eighth clock (SCL) pulse. If the addresses match, and the BF and SSPOV bits are clear, the following events occur:

- The SSPSR register value is loaded into the SSPBUF register.
- The buffer full bit, BF is set.
- An ACK pulse is generated.
- SSP interrupt flag bit, SSPIF (PIR1<3>) is set (interrupt is generated if enabled) - on the falling edge of the ninth SCL pulse.

In 10-bit address mode, two address bytes need to be received by the slave (Figure 11-16). The five Most Significant bits (MSBs) of the first address byte specify if this is a 10-bit address. Bit R/W (SSPSTAT<2>) must specify a write so the slave device will receive the second address byte. For a 10-bit address the first byte would equal '1111 0 A9 A8 0', where A9 and A8 are the two MSBs of the address. The sequence of events for 10-bit address is as follows, with steps 7- 9 for slave-transmitter:

- Receive first (high) byte of Address (bits SSPIF, BF, and bit UA (SSPSTAT<1>) are set).
- Update the SSPADD register with second (low) byte of Address (clears bit UA and releases the SCL line).
- Read the SSPBUF register (clears bit BF) and clear flag bit SSPIF.
- Receive second (low) byte of Address (bits SSPIF, BF, and UA are set).
- Update the SSPADD register with the first (high) byte of Address, if match releases SCL line, this will clear bit UA.
- Read the SSPBUF register (clears bit BF) and clear flag bit SSPIF.
- Receive repeated START condition.
- Receive first (high) byte of Address (bits SSPIF and BF are set).
- Read the SSPBUF register (clears bit BF) and clear flag bit SSPIF.

TABLE 11-4: DATA TRANSFER RECEIVED BYTE ACTIONS

Status Bits as Data Transfer is Received		SSPSR → SSPBUF	Generate ACK Pulse	Set bit SSPIF (SSP Interrupt occurs if enabled)
BF	SSPOV			
0	0	Yes	Yes	Yes
1	0	No	No	Yes
1	1	No	No	Yes
0	1	No	No	Yes

11.5.2 MASTER MODE

Master mode of operation is supported in firmware using interrupt generation on the detection of the START and STOP conditions. The STOP (P) and START (S) bits are cleared from a reset or when the SSP module is disabled. The STOP (P) and START (S) bits will toggle based on the START and STOP conditions. Control of the I²C bus may be taken when the P bit is set, or the bus is idle and both the S and P bits are clear.

In master mode the SCL and SDA lines are manipulated by clearing the corresponding TRISC<4:3> bit(s). The output level is always low, irrespective of the value(s) in PORTC<4:3>. So when transmitting data, a '1' data bit must have the TRISC<4> bit set (input) and a '0' data bit must have the TRISC<4> bit cleared (output). The same scenario is true for the SCL line with the TRISC<3> bit.

The following events will cause SSP Interrupt Flag bit, SSPIF, to be set (SSP Interrupt if enabled):

- START condition
- STOP condition
- Data transfer byte transmitted/received

Master mode of operation can be done with either the slave mode idle (SSPM3:SSPM0 = 1011) or with the slave active. When both master and slave modes are enabled, the software needs to differentiate the source(s) of the interrupt.

11.5.3 MULTI-MASTER MODE

In multi-master mode, the interrupt generation on the detection of the START and STOP conditions allows the determination of when the bus is free. The STOP (P) and START (S) bits are cleared from a reset or when the SSP module is disabled. The STOP (P) and START (S) bits will toggle based on the START and STOP conditions. Control of the I²C bus may be taken when bit P (SSPSTAT<4>) is set, or the bus is idle and both the S and P bits clear. When the bus is busy, enabling the SSP Interrupt will generate the interrupt when the STOP condition occurs.

In multi-master operation, the SDA line must be monitored to see if the signal level is the expected output level. This check only needs to be done when a high level is output. If a high level is expected and a low level is present, the device needs to release the SDA and SCL lines (set TRISC<4:3>). There are two stages where this arbitration can be lost, these are:

- Address Transfer
- Data Transfer

When the slave logic is enabled, the slave continues to receive. If arbitration was lost during the address transfer stage, communication to the device may be in progress. If addressed an $\overline{\text{ACK}}$ pulse will be generated. If arbitration was lost during the data transfer stage, the device will need to re-transfer the data at a later time.

TABLE 11-5: REGISTERS ASSOCIATED WITH I²C OPERATION

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
0Bh, 8Bh, 10Bh, 18Bh	INTCON	GIE	PEIE	TOIE	INTE	RBIE	TOIF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	PSPIF ⁽¹⁾	(2)	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
8Ch	PIE1	PSPIE ⁽¹⁾	(2)	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
13h	SSPBUF	Synchronous Serial Port Receive Buffer/Transmit Register								xxxx xxxx	uuuu uuuu
93h	SSPADD	Synchronous Serial Port (I ² C mode) Address Register								0000 0000	0000 0000
14h	SSPCON	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	0000 0000
94h	SSPSTAT	SMP ⁽³⁾	CKE ⁽³⁾	D/ $\overline{\text{A}}$	P	S	R/ $\overline{\text{W}}$	UA	BF	0000 0000	0000 0000
87h	TRISC	PORTC Data Direction register								1111 1111	1111 1111

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

Shaded cells are not used by SSP module in SPI mode.

Note 1: PSPIF and PSPIE are reserved on the PIC16C66, always maintain these bits clear.

2: PIR1<6> and PIE1<6> are reserved, always maintain these bits clear.

3: The SMP and CKE bits are implemented on the PIC16C66/67 only. All other PIC16C6X devices have these two bits unimplemented, read as '0'.

PIC16C6X

FIGURE 12-2: RCSTA: RECEIVE STATUS AND CONTROL REGISTER (ADDRESS 18h)

R/W-0	R/W-0	R/W-0	R/W-0	U-0	R-0	R-0	R-x
SPEN	RX9	SREN	CREN	—	FERR	OERR	RX9D
bit7							bit0

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

bit 7: **SPEN**: Serial Port Enable bit
(Configures RC7/RX/DT and RC6/TX/CK pins as serial port pins when bits TRISC<7:6> are set)
1 = Serial port enabled
0 = Serial port disabled

bit 6: **RX9**: 9-bit Receive Enable bit
1 = Selects 9-bit reception
0 = Selects 8-bit reception

bit 5: **SREN**: Single Receive Enable bit
Asynchronous mode
Don't care
Synchronous mode - master
1 = Enables single receive
0 = Disables single receive
This bit is cleared after reception is complete.
Synchronous mode - slave
Unused in this mode

bit 4: **CREN**: Continuous Receive Enable bit
Asynchronous mode
1 = Enables continuous receive
0 = Disables continuous receive
Synchronous mode
1 = Enables continuous receive until enable bit CREN is cleared (CREN overrides SREN)
0 = Disables continuous receive

bit 3: **Unimplemented**: Read as '0'

bit 2: **FERR**: Framing Error bit
1 = Framing error (Can be updated by reading RCREG register and receive next valid byte)
0 = No framing error

bit 1: **OERR**: Overrun Error bit
1 = Overrun error (Can be cleared by clearing bit CREN)
0 = No overrun error

bit 0: **RX9D**: 9th bit of received data (Can be parity bit)

12.4 USART Synchronous Slave Mode

Applicable Devices

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

Synchronous Slave Mode differs from Master Mode in the fact that the shift clock is supplied externally at the CK pin (instead of being supplied internally in master mode). This allows the device to transfer or receive data while in SLEEP mode. Slave mode is entered by clearing bit CSRC (TXSTA<7>).

12.4.1 USART SYNCHRONOUS SLAVE TRANSMIT

The operation of the synchronous master and slave modes are identical except in the case of the SLEEP mode.

If two words are written to the TXREG and then the SLEEP instruction is executed, the following will occur:

- The first word will immediately transfer to the TSR register and transmit.
- The second word will remain in TXREG register.
- Flag bit TXIF will not be set.
- When the first word has been shifted out of TSR, the TXREG register will transfer the second word to the TSR and flag bit TXIF will now be set.
- If enable bit TXIE is set, the interrupt will wake the chip from SLEEP and if the global interrupt is enabled, the program will branch to the interrupt vector (0004h).

Steps to follow when setting up Synchronous Slave Transmission:

- Enable the synchronous slave serial port by setting bits SYNC and SPEN, and clearing bit CSRC.
- Clear bits CREN and SREN.
- If interrupts are desired, then set enable bit TXIE.
- If 9-bit transmission is desired, then set bit TX9.
- Enable the transmission by setting bit TXEN.
- If 9-bit transmission is selected, the ninth bit should be loaded in bit TX9D.
- Start transmission by loading data to the TXREG register.

12.4.2 USART SYNCHRONOUS SLAVE RECEPTION

The operation of the synchronous master and slave modes is identical except in the case of the SLEEP mode. Also, enable bit SREN is a don't care in slave mode.

If receive is enabled by setting bit CREN prior to the SLEEP instruction, then a word may be received during SLEEP. On completely receiving the word, the RSR register will transfer the data to the RCREG register and if enable bit RCIE is set, the interrupt generated will wake the chip from SLEEP. If the global interrupt is enabled, the program will branch to the interrupt vector (0004h).

Steps to follow when setting up a Synchronous Slave Reception:

- Enable the synchronous master serial port by setting bits SYNC and SPEN, and clearing bit CSRC.
- If interrupts are desired, then set enable bit RCIE.
- If 9-bit reception is desired, then set bit RX9.
- To enable reception, set enable bit CREN.
- Flag bit RCIF will be set when reception is complete, and an interrupt will be generated if enable bit RCIE was set.
- Read the RCSTA register to get the ninth bit (if enabled) and determine if any error occurred during reception.
- Read the 8-bit received data by reading the RCREG register.
- If any error occurred, clear the error by clearing enable bit CREN.

PIC16C6X

13.5 Interrupts

Applicable Devices

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

The PIC16C6X family has up to 11 sources of interrupt. The interrupt control register (INTCON) records individual interrupt requests in flag bits. It also has individual and global interrupt enable bits.

Note: Individual interrupt flag bits are set regardless of the status of their corresponding mask bit or global enable bit, GIE.

Global interrupt enable bit, GIE (INTCON<7>) enables (if set) all un-masked interrupts or disables (if cleared) all interrupts. When bit GIE is enabled, and an interrupt flag bit and mask bit are set, the interrupt will vector immediately. Individual interrupts can be disabled through their corresponding enable bits in the INTCON register. GIE is cleared on reset.

The “return from interrupt” instruction, RETFIE, exits the interrupt routine as well as sets the GIE bit, which re-enable interrupts.

The RB0/INT pin interrupt, the RB port change interrupt and the TMR0 overflow interrupt flag bits are contained in the INTCON register.

The peripheral interrupt flag bits are contained in special function registers PIR1 and PIR2. The corresponding interrupt enable bits are contained in special function registers PIE1 and PIE2 and the peripheral interrupt enable bit is contained in special function register INTCON.

When an interrupt is responded to, bit GIE is cleared to disable any further interrupts, the return address is pushed onto the stack and the PC is loaded with 0004h. Once in the interrupt service routine the source(s) of the interrupt can be determined by polling the interrupt flag bits. The interrupt flag bit(s) must be cleared in software before re-enabling interrupts to avoid recursive interrupts.

For external interrupt events, such as the RB0/INT pin or RB port change interrupt, the interrupt latency will be three or four instruction cycles. The exact latency depends when the interrupt event occurs (Figure 13-19). The latency is the same for one or two cycle instructions. Once in the interrupt service routine the source(s) of the interrupt can be determined by polling the interrupt flag bits. The interrupt flag bit(s) must be cleared in software before re-enabling interrupts to

avoid infinite interrupt requests. Individual interrupt flag bits are set regardless of the status of their corresponding mask bit or the GIE bit.

Note: For the PIC16C61/62/64/65, if an interrupt occurs while the Global Interrupt Enable bit, GIE is being cleared, bit GIE may unintentionally be re-enabled by the user's Interrupt Service Routine (the RETFIE instruction). The events that would cause this to occur are:

1. An instruction clears the GIE bit while an interrupt is acknowledged
2. The program branches to the Interrupt vector and executes the Interrupt Service Routine.
3. The Interrupt Service Routine completes with the execution of the RETFIE instruction. This causes the GIE bit to be set (enables interrupts), and the program returns to the instruction after the one which was meant to disable interrupts.
4. Perform the following to ensure that interrupts are globally disabled.

```
LOOP BCF INTCON,GIE      ;Disable Global
                          ;Interrupt bit
      BTFSC INTCON,GIE    ;Global Interrupt
                          ;Disabled?
      GOTO LOOP          ;NO, try again
      :                  ;Yes, continue
                          ;with program flow
```


14.1 Instruction Descriptions

ADDLW Add Literal and W

Syntax:	[<i>label</i>] ADDLW k			
Operands:	$0 \leq k \leq 255$			
Operation:	$(W) + k \rightarrow (W)$			
Status Affected:	C, DC, Z			
Encoding:	11	111x	kkkk	kkkk
Description:	The contents of the W register are added to the eight bit literal 'k' and the result is placed in the W register.			
Words:	1			
Cycles:	1			
Q Cycle Activity:	Q1	Q2	Q3	Q4
	Decode	Read literal 'k'	Process data	Write to W

Example: ADDLW 0x15

Before Instruction

 W = 0x10

After Instruction

 W = 0x25

ADDWF Add W and f

Syntax:	[<i>label</i>] ADDWF f,d			
Operands:	$0 \leq f \leq 127$ $d \in [0,1]$			
Operation:	$(W) + (f) \rightarrow (\text{destination})$			
Status Affected:	C, DC, Z			
Encoding:	00	0111	dfff	ffff
Description:	Add the contents of the W register with register 'f'. If 'd' is 0 the result is stored in the W register. If 'd' is 1 the result is stored back in register 'f'.			
Words:	1			
Cycles:	1			
Q Cycle Activity:	Q1	Q2	Q3	Q4
	Decode	Read register 'f'	Process data	Write to destination

Example ADDWF FSR, 0

Before Instruction

 W = 0x17

 FSR = 0xC2

After Instruction

 W = 0xD9

 FSR = 0xC2

ANDLW AND Literal with W

Syntax:	[<i>label</i>] ANDLW k			
Operands:	$0 \leq k \leq 255$			
Operation:	$(W) .\text{AND.} (k) \rightarrow (W)$			
Status Affected:	Z			
Encoding:	11	1001	kkkk	kkkk
Description:	The contents of W register are AND'ed with the eight bit literal 'k'. The result is placed in the W register.			
Words:	1			
Cycles:	1			
Q Cycle Activity:	Q1	Q2	Q3	Q4
	Decode	Read literal "k"	Process data	Write to W

Example ANDLW 0x5F

Before Instruction

 W = 0xA3

After Instruction

 W = 0x03

ANDWF AND W with f

Syntax:	[<i>label</i>] ANDWF f,d			
Operands:	$0 \leq f \leq 127$ $d \in [0,1]$			
Operation:	$(W) .\text{AND.} (f) \rightarrow (\text{destination})$			
Status Affected:	Z			
Encoding:	00	0101	dfff	ffff
Description:	AND the W register with register 'f'. If 'd' is 0 the result is stored in the W register. If 'd' is 1 the result is stored back in register 'f'.			
Words:	1			
Cycles:	1			
Q Cycle Activity:	Q1	Q2	Q3	Q4
	Decode	Read register 'f'	Process data	Write to destination

Example ANDWF FSR, 1

Before Instruction

 W = 0x17

 FSR = 0xC2

After Instruction

 W = 0x17

 FSR = 0x02

SUBWF Subtract W from f

Syntax: [label] SUBWF f,d

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

Operation: $(f) - (W) \rightarrow (\text{destination})$

Status Affected: C, DC, Z

Encoding:

00	0010	dfff	ffff
----	------	------	------

Description: Subtract (2's complement method) W register from register 'f'. If 'd' is 0 the result is stored in the W register. If 'd' is 1 the result is stored back in register 'f'.

Words: 1

Cycles: 1

Q Cycle Activity:

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

Example 1: SUBWF REG1, 1

Before Instruction

REG1 = 3
W = 2
C = ?
Z = ?

After Instruction

REG1 = 1
W = 2
C = 1; result is positive
Z = 0

Example 2: Before Instruction

REG1 = 2
W = 2
C = ?
Z = ?

After Instruction

REG1 = 0
W = 2
C = 1; result is zero
Z = 1

Example 3: Before Instruction

REG1 = 1
W = 2
C = ?
Z = ?

After Instruction

REG1 = 0xFF
W = 2
C = 0; result is negative
Z = 0

SWAPF Swap Nibbles in f

Syntax: [label] SWAPF f,d

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

Operation: $(f<3:0>) \rightarrow (\text{destination}<7:4>)$,
 $(f<7:4>) \rightarrow (\text{destination}<3:0>)$

Status Affected: None

Encoding:

00	1110	dfff	ffff
----	------	------	------

Description: The upper and lower nibbles of register 'f' are exchanged. If 'd' is 0 the result is placed in W register. If 'd' is 1 the result is placed in register 'f'.

Words: 1

Cycles: 1

Q Cycle Activity:

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

Example SWAPF REG, 0

Before Instruction

REG1 = 0xA5

After Instruction

REG1 = 0xA5
W = 0x5A

TRIS Load TRIS Register

Syntax: [label] TRIS f

Operands: $5 \leq f \leq 7$

Operation: $(W) \rightarrow \text{TRIS register } f$;

Status Affected: None

Encoding:

00	0000	0110	0fff
----	------	------	------

Description: The instruction is supported for code compatibility with the PIC16C5X products. Since TRIS registers are readable and writable, the user can directly address them.

Words: 1

Cycles: 1

Example

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

15.0 ELECTRICAL CHARACTERISTICS FOR PIC16C61

Absolute Maximum Ratings †

Ambient temperature under bias.....	-55°C to +125°C
Storage temperature	-65°C to +150°C
Voltage on any pin with respect to 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 pin with respect to Vss	0V to +14V
Total power dissipation (Note 1).....	800 mW
Maximum current out of VSS pin	150 mA
Maximum current into VDD pin	100 mA
Input clamp current, I _{IK} (V _I < 0 or V _I > V _{DD}).....	± 20 mA
Output clamp current, I _{OK} (V _O < 0 or V _O > V _{DD})	± 20 mA
Maximum output current sunk by any I/O pin.....	25 mA
Maximum output current sourced by any I/O pin	20 mA
Maximum current sunk by PORTA	80 mA
Maximum current sourced by PORTA.....	50 mA
Maximum current sunk by PORTB.....	150 mA
Maximum current sourced by PORTB.....	100 mA

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

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 15-1: CROSS REFERENCE OF DEVICE SPECS FOR OSCILLATOR CONFIGURATIONS AND FREQUENCIES OF OPERATION (COMMERCIAL DEVICES)

OSC	PIC16C61-04	PIC16C61-20	PIC16LC61-04	JW Devices
RC	VDD: 4.0V to 6.0V IDD: 3.3 mA max. at 5.5V IPD: 14 µA max. at 4V Freq: 4 MHz max.	VDD: 4.5V to 5.5V IDD: 1.8 mA typ. at 5.5V IPD: 1.0 µA typ. at 4V Freq: 4 MHz max.	VDD: 3.0V to 6.0V IDD: 1.4 mA typ. at 3.0V IPD: 0.6 µA typ. at 3V Freq: 4 MHz max.	VDD: 4.0V to 6.0V IDD: 3.3 mA max. at 5.5V IPD: 14 µA max. at 4V Freq: 4 MHz max.
XT	VDD: 4.0V to 6.0V IDD: 3.3 mA max. at 5.5V IPD: 14 µA max. at 4V Freq: 4 MHz max.	VDD: 4.5V to 5.5V IDD: 1.8 mA typ. at 5.5V IPD: 1.0 µA typ. at 4V Freq: 4 MHz max.	VDD: 3.0V to 6.0V IDD: 1.4 mA typ. at 3.0V IPD: 0.6 µA typ. at 3V Freq: 4 MHz max.	VDD: 4.0V to 6.0V IDD: 3.3 mA max. at 5.5V IPD: 14 µA max. at 4V Freq: 4 MHz max.
HS	VDD: 4.5V to 5.5V IDD: 13.5 mA typ. at 5.5V IPD: 1.0 µA typ. at 4.5V Freq: 4 MHz max.	VDD: 4.5V to 5.5V IDD: 30 mA max. at 5.5V IPD: 1.0 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: 1.0 µA typ. at 4.5V Freq: 20 MHz max.
LP	VDD: 4.0V to 6.0V IDD: 15 µA typ. at 32 kHz, 4.0V IPD: 0.6 µA typ. at 4.0V Freq: 200 kHz max.	Not recommended for use in LP mode	VDD: 3.0V to 6.0V IDD: 32 µA max. at 32 kHz, 3.0V IPD: 9 µA max. at 3.0V Freq: 200 kHz max.	VDD: 3.0V to 6.0V IDD: 32 µA max. at 32 kHz, 3.0V IPD: 9 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.

FIGURE 16-14: MAXIMUM I_{DD} vs. FREQUENCY (EXTERNAL CLOCK, -55° TO +125°C)

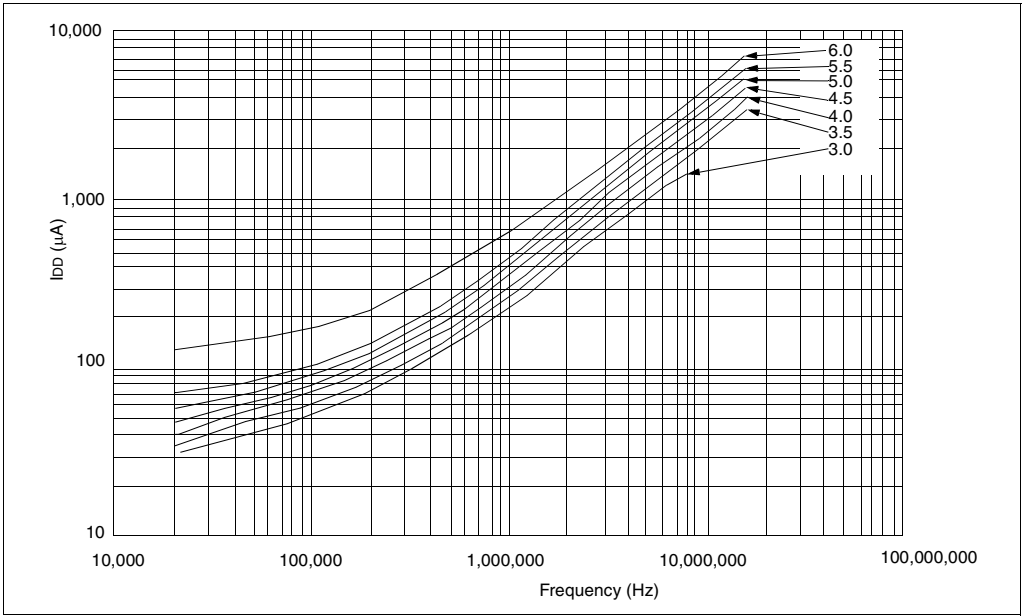


FIGURE 16-15: WDT TIMER TIME-OUT PERIOD vs. V_{DD}

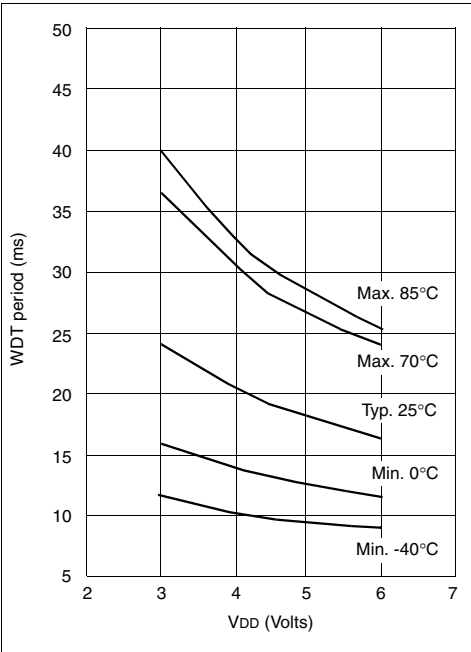
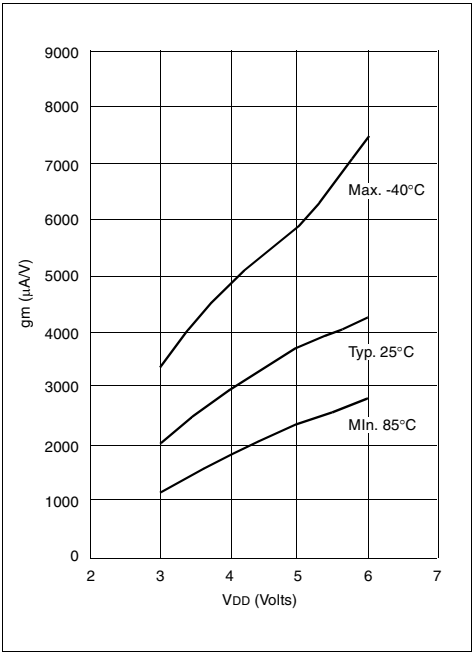


FIGURE 16-16: TRANSCONDUCTANCE (gm) OF HS OSCILLATOR vs. V_{DD}



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

18.5 Timing Diagrams and Specifications

FIGURE 18-2: EXTERNAL CLOCK TIMING

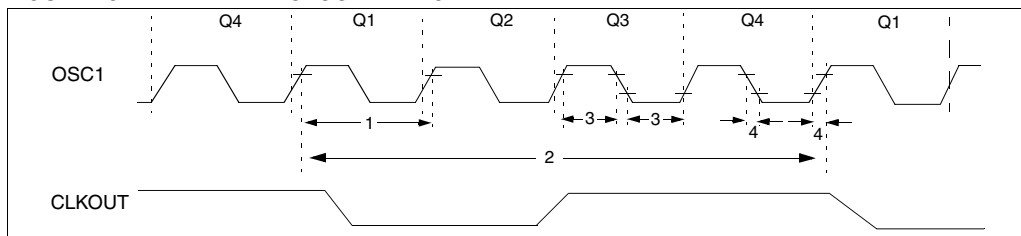


TABLE 18-2: EXTERNAL CLOCK TIMING REQUIREMENTS

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	—	10	MHz	HS osc mode (-10)
			DC	—	20	MHz	HS osc mode (-20)
			DC	—	200	kHz	LP osc mode
		Oscillator Frequency (Note 1)	DC	—	4	MHz	RC osc mode
			0.1	—	4	MHz	XT osc mode
			4	—	20	MHz	HS osc mode
			5	—	200	kHz	LP osc mode
			5	—	—	μs	LP osc mode
1	Tosc	External CLKIN Period (Note 1)	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
		Oscillator Period (Note 1)	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	Instruction Cycle Time (Note 1)	200	Tcy	DC	ns	Tcy = 4/Fosc
			200	Tcy	DC	ns	Tcy = 4/Fosc
3	TosL, TosH	External Clock in (OSC1) High or Low Time	100	—	—	ns	XT oscillator
			2.5	—	—	μs	LP oscillator
			15	—	—	ns	HS oscillator
4	TosR, TosF	External Clock in (OSC1) Rise or Fall Time	—	—	25	ns	XT oscillator
			—	—	50	ns	LP oscillator
			—	—	15	ns	HS oscillator

† 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.

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

FIGURE 18-11: I²C BUS DATA TIMING

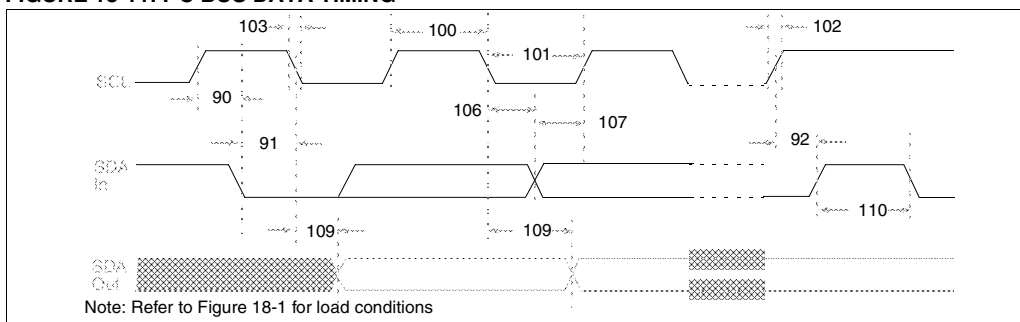


TABLE 18-10: I²C BUS DATA REQUIREMENTS

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

* These parameters are characterized but not tested.

- Note 1: As a transmitter, the device must provide this internal minimum delay time to bridge the undefined region (min. 300 ns) of the falling edge of SCL to avoid unintended generation of START or STOP conditions.
- Note 2: A fast-mode (400 kHz) I²C-bus device can be used in a standard-mode (100 kHz) I²C-bus system, but the requirement tsu;DAT ≥ 250 ns must then be met. This will automatically be the case if the device does not stretch the LOW period of the SCL signal. If such a device does stretch the LOW period of the SCL signal, it must output the next data bit to the SDA line TR max.+tsu;DAT = 1000 + 250 = 1250 ns (according to the standard-mode I²C bus specification) before the SCL line is released.

20.5 Timing Diagrams and Specifications

FIGURE 20-2: EXTERNAL CLOCK TIMING

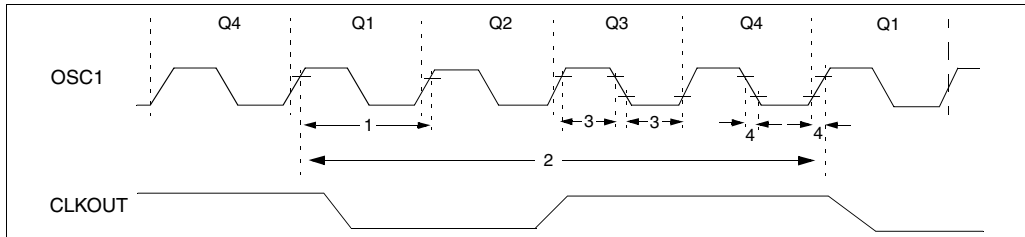


TABLE 20-2: EXTERNAL CLOCK TIMING REQUIREMENTS

Param 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	—	10	MHz	HS osc mode (-10)
			DC	—	20	MHz	HS osc mode (-20)
			DC	—	200	kHz	LP osc mode
		Oscillator Frequency (Note 1)	DC	—	4	MHz	RC osc mode
			0.1	—	4	MHz	XT osc mode
			4	—	20	MHz	HS osc mode
1	Tosc	External CLKIN Period (Note 1)	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
		Oscillator Period (Note 1)	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	Instruction Cycle Time (Note 1)	200	Tcy	DC	ns	Tcy = 4/Fosc
			—	—	—	—	—
3*	TosL, TosH	External Clock in (OSC1) High or Low Time	100	—	—	ns	XT oscillator
			2.5	—	—	μs	LP oscillator
			15	—	—	ns	HS oscillator
4*	TosR, TosF	External Clock in (OSC1) Rise or Fall Time	—	—	25	ns	XT oscillator
			—	—	50	ns	LP oscillator
			—	—	15	ns	HS oscillator

* These parameters are characterized but not tested.

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

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

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

FIGURE 20-4: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER AND POWER-UP TIMER TIMING

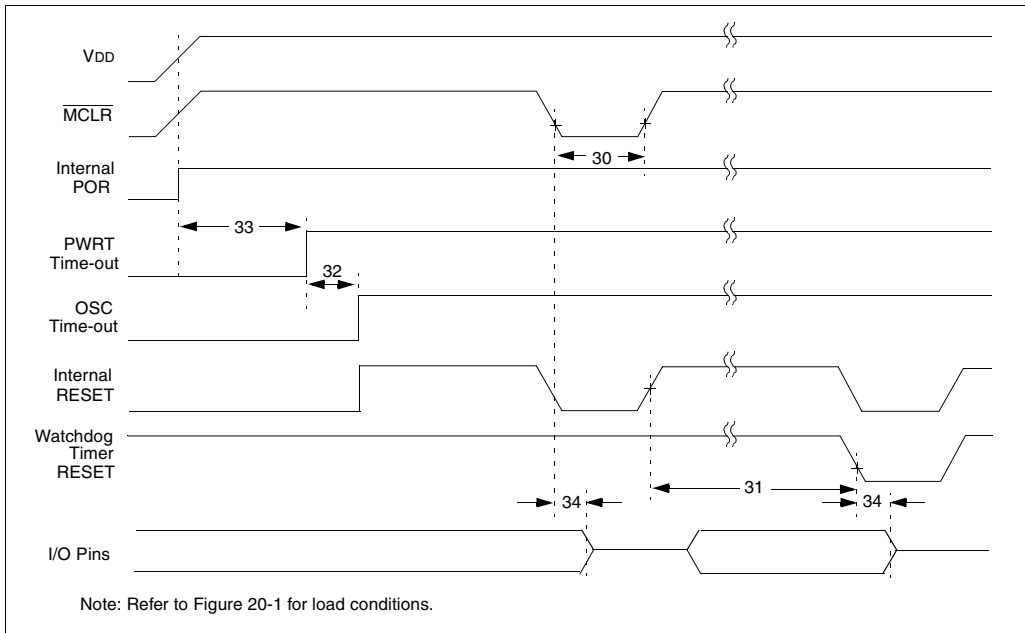


FIGURE 20-5: BROWN-OUT RESET TIMING

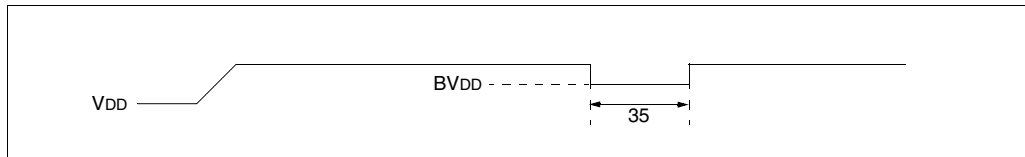


TABLE 20-4: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER, POWER-UP TIMER, AND BROWN-OUT RESET REQUIREMENTS

Parameter No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
30	Tmcl	MCLR Pulse Width (low)	2	—	—	μs	VDD = 5V, -40°C to +125°C
31*	Twdt	Watchdog Timer Time-out Period (No Prescaler)	7	18	33	ms	VDD = 5V, -40°C to +125°C
32	Tost	Oscillation Start-up Timer Period	—	1024 TOSC	—	—	TOSC = OSC1 period
33*	Tpwrt	Power-up Timer Period	28	72	132	ms	VDD = 5V, -40°C to +125°C
34	Tioz	I/O Hi-impedance from MCLR Low or WDT reset	—	—	2.1	μs	
35	TBOR	Brown-out Reset Pulse Width	100	—	—	μs	VDD ≤ BVDD (D005)

* 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

22.4 Timing Parameter Symbology

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

1. TppS2ppS
2. TppS
3. TCC:ST (I²C specifications only)
4. Ts (I²C specifications only)

T		T	
F	Frequency	T	Time

Lowercase letters (pp) and their meanings:

pp		osc	OSC1
cc	CCP1	rd	\overline{RD}
ck	CLKOUT	rw	\overline{RD} or \overline{WR}
cs	\overline{CS}	sc	SCK
di	SDI	ss	\overline{SS}
do	SDO	t0	T0CKI
dt	Data in	t1	T1CKI
io	I/O port	wr	\overline{WR}
mc	\overline{MCLR}		

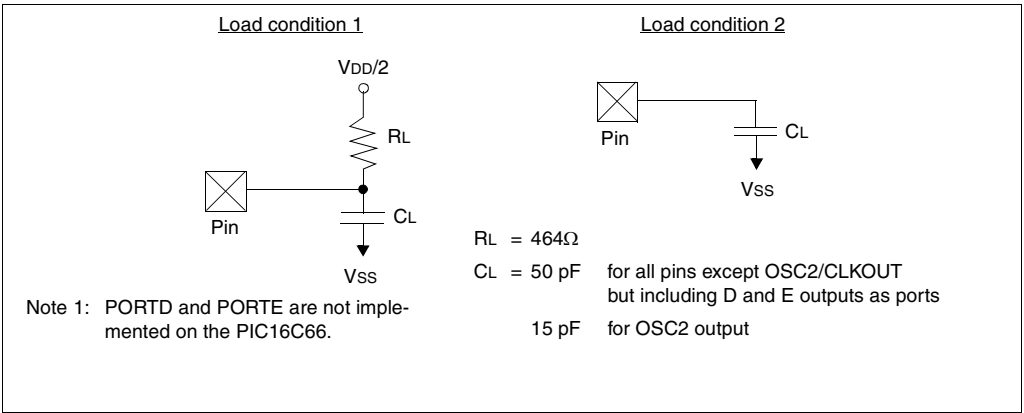
Uppercase letters and their meanings:

S		P	Period
F	Fall	R	Rise
H	High	V	Valid
I	Invalid (Hi-impedance)	Z	Hi-impedance
L	Low		
I²C only		High	High
AA	output access	Low	Low
BUF	Bus free		

TCC:ST (I²C specifications only)

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

FIGURE 22-1: LOAD CONDITIONS FOR DEVICE TIMING SPECIFICATIONS



PIC16C6X

Applicable Devices

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

FIGURE 23-12: TYPICAL I_{DD} vs. FREQUENCY (RC MODE @ 22 pF, 25°C)

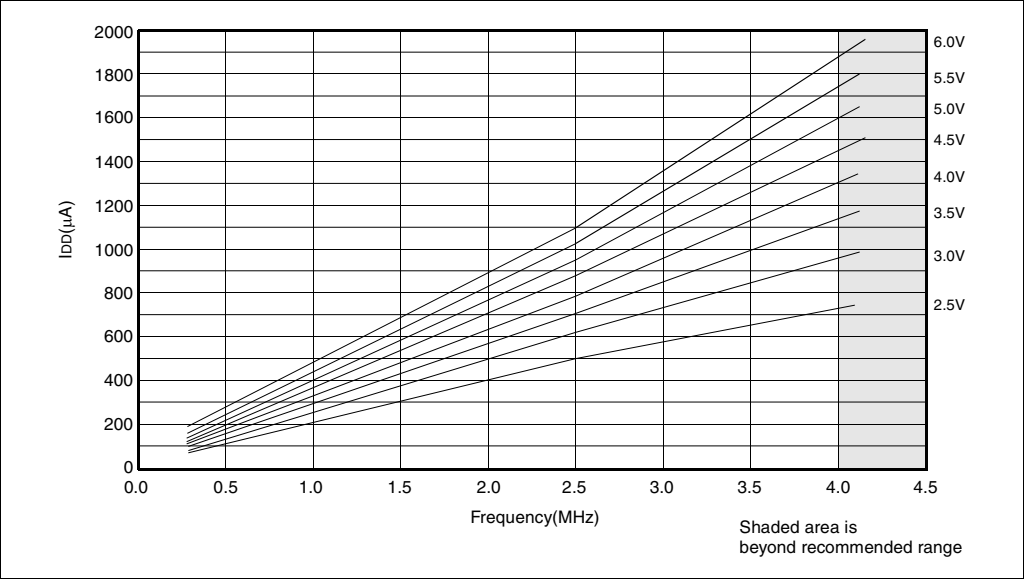
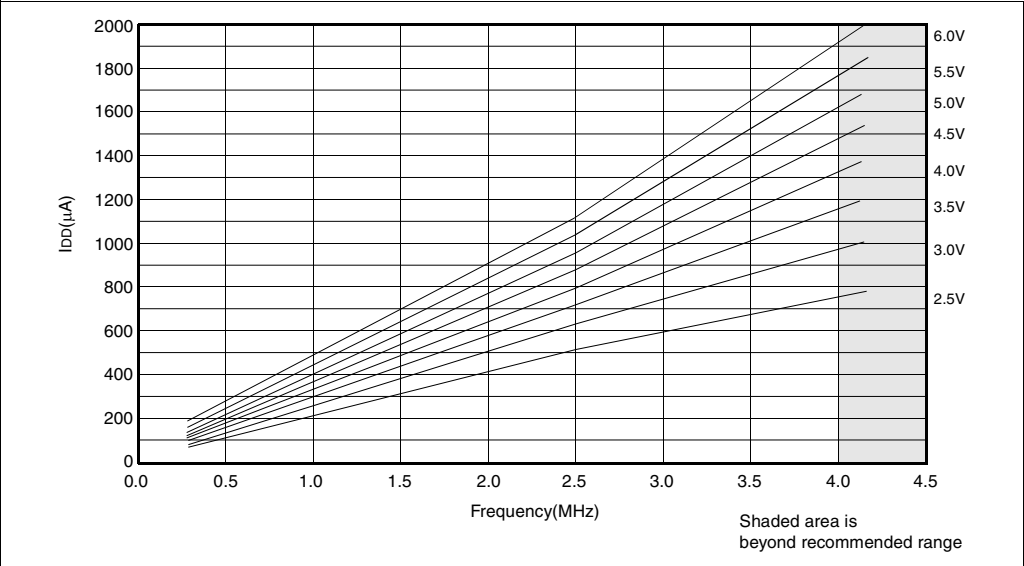


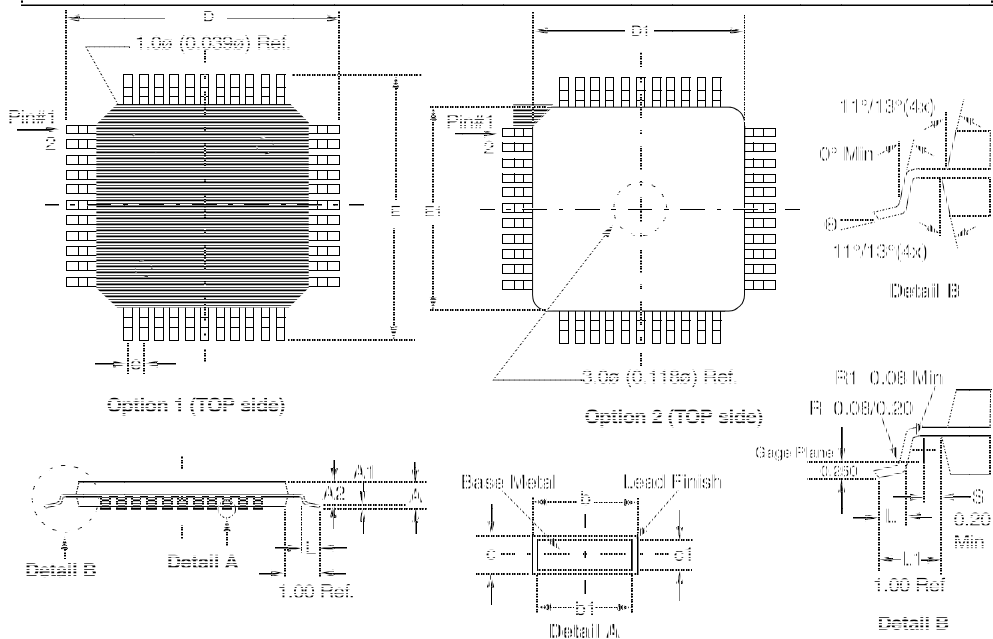
FIGURE 23-13: MAXIMUM I_{DD} vs. FREQUENCY (RC MODE @ 22 pF, -40°C TO 85°C)



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

24.13 44-Lead Plastic Surface Mount (TQFP 10x10 mm Body 1.0/0.10 mm Lead Form) (TQ)

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



Package Group: Plastic TQFP						
Symbol	Millimeters			Inches		
	Min	Max	Notes	Min	Max	Notes
A	1.00	1.20		0.039	0.047	
A1	0.05	0.15		0.002	0.006	
A2	0.95	1.05		0.037	0.041	
D	11.75	12.25		0.463	0.482	
D1	9.90	10.10		0.390	0.398	
E	11.75	12.25		0.463	0.482	
E1	9.90	10.10		0.390	0.398	
L	0.45	0.75		0.018	0.030	
e	0.80 BSC			0.031 BSC		
b	0.30	0.45		0.012	0.018	
b1	0.30	0.40		0.012	0.016	
c	0.09	0.20		0.004	0.008	
c1	0.09	0.16		0.004	0.006	
N	44	44		44	44	
θ	0°	7°		0°	7°	

Note 1: Dimensions D1 and E1 do not include mold protrusion. Allowable mold protrusion is 0.25mm (0.010") per side. D1 and E1 dimensions including mold mismatch.

2: Dimension "b" does not include Dambar protrusion, allowable Dambar protrusion shall be 0.08mm (0.003")max.

3: This outline conforms to JEDEC MS-026.

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

PIC16C6X

LIST OF TABLES

Table 1-1:	PIC16C6X Family of Devices	6
Table 3-1:	PIC16C61 Pinout Description	14
Table 3-2:	PIC16C62/62A/R62/63/R63/66 Pinout Description	15
Table 3-3:	PIC16C64/64A/R64/65/65A/R65/67 Pinout Description	16
Table 4-1:	Special Function Registers for the PIC16C61	23
Table 4-2:	Special Function Registers for the PIC16C62/62A/R62	24
Table 4-3:	Special Function Registers for the PIC16C63/R63	26
Table 4-4:	Special Function Registers for the PIC16C64/64A/R64	28
Table 4-5:	Special Function Registers for the PIC16C65/65A/R65	30
Table 4-6:	Special Function Registers for the PIC16C66/67	32
Table 5-1:	PORTA Functions	52
Table 5-2:	Registers/Bits Associated with PORTA	52
Table 5-3:	PORTB Functions	54
Table 5-4:	Summary of Registers Associated with PORTB	54
Table 5-5:	PORTC Functions for PIC16C62/64	55
Table 5-6:	PORTC Functions for PIC16C62A/R62/64A/R64	56
Table 5-7:	PORTC Functions for PIC16C63/R63/65/65A/R65/66/67	56
Table 5-8:	Summary of Registers Associated with PORTC	56
Table 5-9:	PORTD Functions	57
Table 5-10:	Summary of Registers Associated with PORTD	57
Table 5-11:	PORTE Functions	59
Table 5-12:	Summary of Registers Associated with PORTE	59
Table 5-13:	Registers Associated with Parallel Slave Port	62
Table 7-1:	Registers Associated with Timer0	69
Table 8-1:	Capacitor Selection for the Timer1 Oscillator	73
Table 8-2:	Registers Associated with Timer1 as a Timer/Counter	74
Table 9-1:	Registers Associated with Timer2 as a Timer/Counter	76
Table 10-1:	CCP Mode - Timer Resource	77
Table 10-2:	Interaction of Two CCP Modules	77
Table 10-3:	Example PWM Frequencies and Resolutions at 20 MHz	81
Table 10-4:	Registers Associated with Timer1, Capture and Compare	81
Table 10-5:	Registers Associated with PWM and Timer2	82
Table 11-1:	Registers Associated with SPI Operation	88
Table 11-2:	Registers Associated with SPI Operation (PIC16C66/67)	94
Table 11-3:	I ² C Bus Terminology	95
Table 11-4:	Data Transfer Received Byte Actions	100
Table 11-5:	Registers Associated with I ² C Operation	103
Table 12-1:	Baud Rate Formula	107
Table 12-2:	Registers Associated with Baud Rate Generator	107
Table 12-3:	Baud Rates for Synchronous Mode	108
Table 12-4:	Baud Rates for Asynchronous Mode (BRGH = 0)	108
Table 12-5:	Baud Rates for Asynchronous Mode (BRGH = 1)	109
Table 12-6:	Registers Associated with Asynchronous Transmission	113
Table 12-7:	Registers Associated with Asynchronous Reception	115
Table 12-8:	Registers Associated with Synchronous Master Transmission	117
Table 12-9:	Registers Associated with Synchronous Master Reception	118
Table 12-10:	Registers Associated with Synchronous Slave Transmission	121
Table 12-11:	Registers Associated with Synchronous Slave Reception	121
Table 13-1:	Ceramic Resonators PIC16C61	126
Table 13-2:	Ceramic Resonators PIC16C62/62A/R62/63/R63/ 64/64A/R64/65/65A/R65/66/67	126
Table 13-3:	Capacitor Selection for Crystal Oscillator for PIC16C61	126
Table 13-4:	Capacitor Selection for Crystal Oscillator for PIC16C62/62A/R62/63/R63/ 64/64A/R64/65/65A/R65/66/67	126
Table 13-5:	Time-out in Various Situations, PIC16C61/62/64/65	130
Table 13-6:	Time-out in Various Situations, PIC16C62A/R62/63/R63/ 64A/R64/65A/R65/66/67	130
Table 13-7:	Status Bits and Their Significance, PIC16C61	130
Table 13-8:	Status bits and Their Significance, PIC16C62/64/65	130
Table 13-9:	Status Bits and Their Significance for PIC16C62A/R62/63/R63/ 64A/R64/65A/R65/66/67	131
Table 13-10:	Reset Condition for Special Registers on PIC16C61/62/64/65	131
Table 13-11:	Reset Condition for Special Registers on PIC16C62A/R62/63/R63/ 64A/R64/65A/R65/66/67	131
Table 13-12:	Initialization Conditions for all Registers	132
Table 14-1:	Opcode Field Descriptions	143
Table 14-2:	PIC16CXX Instruction Set	144
Table 15-1:	Development Tools from Microchip	162
Table 16-1:	Cross Reference of Device Specs for Oscillator Configurations and Frequencies of Operation (Commercial Devices)	163
Table 16-2:	External Clock Timing Requirements	169
Table 16-3:	CLKOUT and I/O Timing Requirements	170
Table 16-4:	Reset, Watchdog Timer, Oscillator Start-up Timer and Power-up Timer Requirements	171
Table 16-5:	Timer0 External Clock Requirements	172
Table 17-1:	RC Oscillator Frequencies	173
Table 17-2:	Input Capacitance*	181