



# PIC16C6X

## 8-Bit CMOS Microcontrollers

### Devices included in this data sheet:

- PIC16C61
- PIC16C62
- PIC16C62A
- PIC16CR62
- PIC16C63
- PIC16C64
- PIC16C64A
- PIC16CR64
- PIC16C65
- PIC16C65A

### PIC16C6X Microcontroller Core Features:

- High performance RISC CPU
- Only 35 single word instructions to learn
- All single cycle instructions (200 ns) except for program branches which are two-cycle
- Operating speed: DC - 20 MHz clock input  
DC - 200 ns instruction cycle
- Interrupt capability
- Eight level deep hardware stack
- Direct, indirect, and relative addressing modes
- Power-on Reset (POR)
- Power-up Timer (PWRT) and Oscillator Start-up Timer (OST)
- Watchdog Timer (WDT) with its own on-chip RC oscillator for reliable operation
- Programmable code-protection
- Power saving SLEEP mode

- Selectable oscillator options
- Low-power, high-speed CMOS EPROM/ROM technology
- Fully static design
- Wide operating voltage range: 2.5V to 6.0V
- Commercial, Industrial, and Automotive Temperature Range
- Low-power consumption:
  - < 2 mA @ 5V, 4 MHz
  - 15  $\mu$ A typical @ 3V, 32 kHz
  - < 1  $\mu$ A typical standby current

### PIC16C6X Peripheral Features:

- Timer0: 8-bit timer/counter with prescaler
- Timer1: 16-bit timer/counter with prescaler. TMR1 can be incremented during sleep via external crystal/clock
- Timer2: 8-bit timer/counter with period register, prescaler and postscaler
- Capture/Compare/PWM module(s)
- Capture is 16-bit, max resolution 12.5 ns, compare is 16-bit, max resolution 200 ns, max. PWM resolution is 10-bit.
- Synchronous Serial Port (SSP) with SPI and I<sup>2</sup>C™
- Universal Synchronous Asynchronous Receiver Transmitter (USART/SCI)
- Parallel Slave Port (PSP) 8-bits wide, with external  $\overline{RD}$ ,  $\overline{WR}$  and  $\overline{CS}$  controls
- Brown-out detection circuitry for Brown-out Reset (BOR)

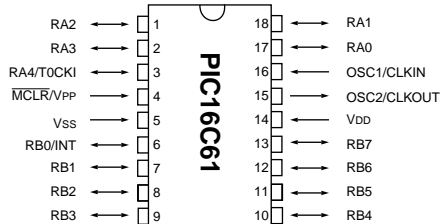
PIC16C6X Features	61	62	62A	R62	63	64	64A	R64	65	65A
Program Memory (EPROM)	1K	2K	2K	—	4K	2K	2K	—	4K	4K
(ROM)	—	—	—	2K	—	—	—	2K	—	—
Data Memory (Bytes)	36	128	128	128	192	128	128	128	192	192
I/O Pins	13	22	22	22	22	33	33	33	33	33
Parallel Slave Port	—	—	—	—	—	Yes	Yes	Yes	Yes	Yes
Capture/Compare/PWM Module	—	1	1	1	2	1	1	1	2	2
Timer Modules	1	3	3	3	3	3	3	3	3	3
Serial Communication	—	SPI/ I <sup>2</sup> C	SPI/ I <sup>2</sup> C	SPI/ I <sup>2</sup> C	SPI/I <sup>2</sup> C, USART	SPI/ I <sup>2</sup> C	SPI/ I <sup>2</sup> C	SPI/ I <sup>2</sup> C	SPI/I <sup>2</sup> C, USART	SPI/I <sup>2</sup> C, USART
In-Circuit Serial Programming	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Brown-out Reset	—	—	Yes	Yes	Yes	—	Yes	Yes	—	Yes
Interrupt Sources	3	7	7	7	10	8	8	8	11	11
Sink/Source Current (mA)	25/20	25/25	25/25	25/25	25/25	25/25	25/25	25/25	25/25	25/25

I<sup>2</sup>C is a trademark of Philips Corporation  
SPI is a trademark of Motorola Corporation.

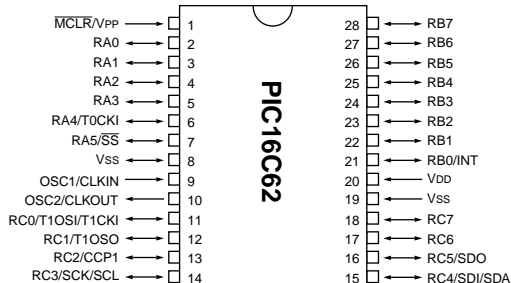
# PIC16C6X

## Pin Diagrams

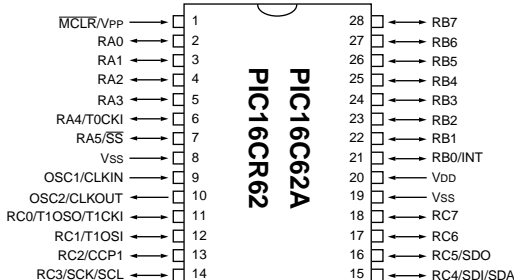
**PDIP, SOIC, Windowed CERDIP**



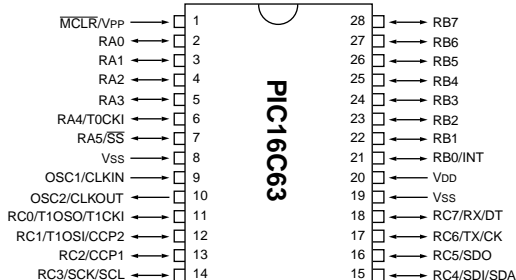
**SDIP, SOIC, SSOP, Windowed CERDIP**



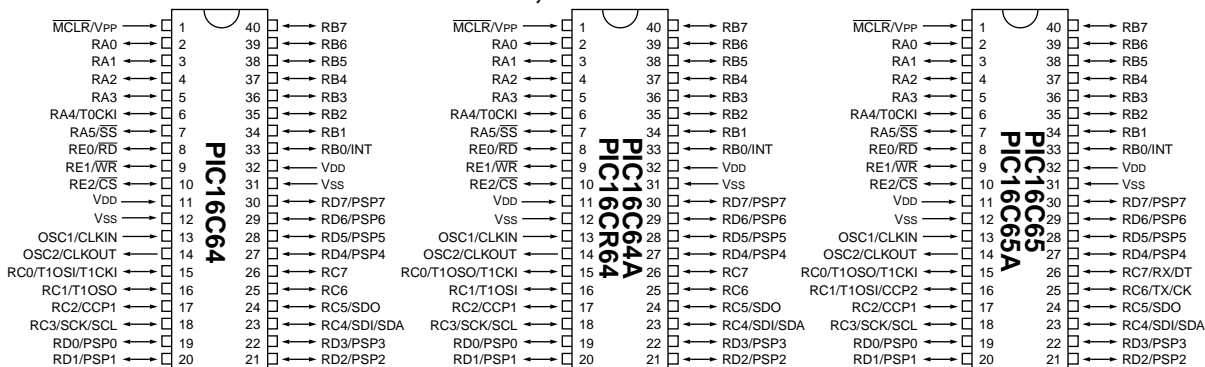
**SDIP, SOIC, SSOP, Windowed CERDIP**



**SDIP, SOIC, Windowed CERDIP**

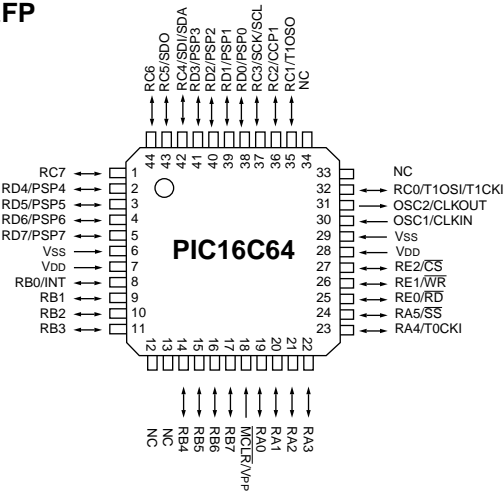


**PDIP, Windowed CERDIP**

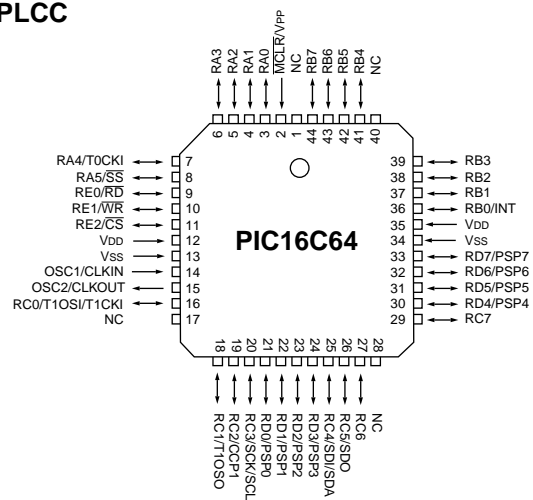


## Pin Diagrams (Cont.'d)

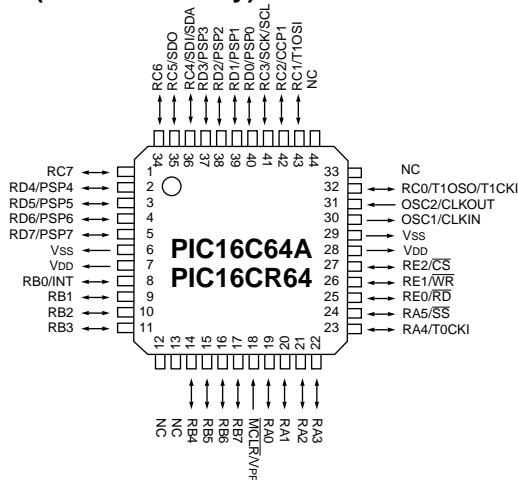
**MQFP**



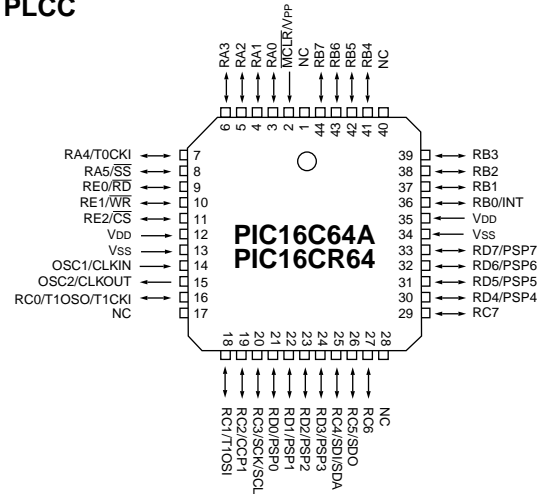
**PLCC**



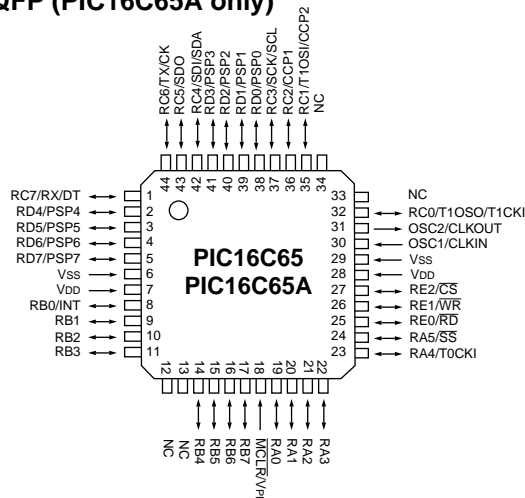
**MQFP, TQFP (PIC16C64A only)**



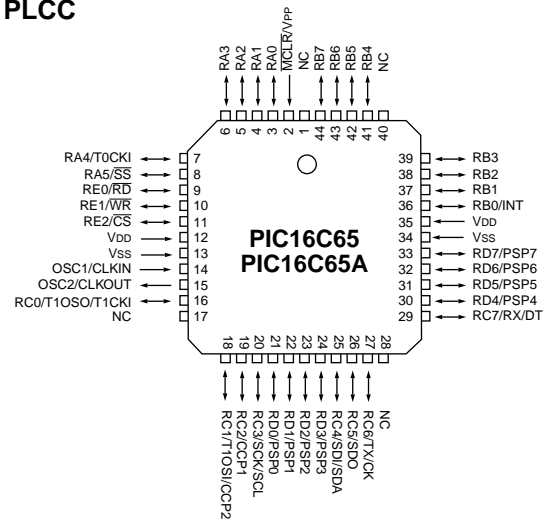
**PLCC**



**MQFP, TQFP (PIC16C65A only)**



**PLCC**



# PIC16C6X

## TABLE OF CONTENTS

1.0	General Description .....	5
2.0	PIC16C6X Device Varieties .....	7
3.0	Architectural Overview .....	9
4.0	Memory Organization .....	19
5.0	I/O Ports.....	47
6.0	Overview of Timer Modules .....	59
7.0	Timer0 Module.....	61
8.0	Timer1 Module.....	67
9.0	Timer2 Module.....	71
10.0	Capture/Compare/PWM (CCP) Module(s) .....	73
11.0	Synchronous Serial Port (SSP) Module.....	79
12.0	Universal Synchronous Asynchronous Receiver Transmitter (USART) Module .....	95
13.0	Special Features of the CPU .....	111
14.0	Instruction Set Summary .....	131
15.0	Development Support .....	143
16.0	Electrical Characteristics for PIC16C61.....	149
17.0	DC and AC Characteristics Graphs and Tables for PIC16C61 .....	159
18.0	Electrical Characteristics for PIC16C62/64.....	171
19.0	DC and AC Characteristics Graphs and Tables for PIC16C62/64 .....	187
20.0	Electrical Characteristics for PIC16C62A/R62/64A/R64.....	189
21.0	DC and AC Characteristics Graphs and Tables for PIC16C62A/R62/64A/R64 .....	207
22.0	Electrical Characteristics for PIC16C65.....	209
23.0	DC and AC Characteristics Graphs and Tables for PIC16C65 .....	225
24.0	Electrical Characteristics for PIC16C63/65A .....	227
25.0	DC and AC Characteristics Graphs and Tables for PIC16C63/65A .....	243
26.0	Packaging Information .....	245
	Index .....	271
	List of Examples .....	277
	List of Figures .....	277
	List of Tables .....	279
	Connecting to Microchip BBS .....	281
	Reader Response.....	282

For register and module descriptions in this data sheet, device legends show which devices apply to those sections. For example, the legend below shows that some features of only the PIC16C62A, PIC16CR62, PIC16C63, PIC16C64A, PIC16CR64, and PIC16C65A are described in this section.

Applicable Devices									
61	62	62A	R62	63	64	64A	R64	65	65A

### *To Our Valued Customers*

We constantly strive to improve the quality of all our products and documentation. We have spent an exceptional amount of time to ensure that these documents are correct. However, we realize that we may have missed a few things. If you find any information that is missing or appears in error, please use the reader response form in the back of this data sheet to inform us. We appreciate your assistance in making this a better document.

## 1.0 GENERAL DESCRIPTION

The PIC16CXX is a family of low-cost, high-performance, CMOS, fully-static, 8-bit microcontrollers.

All PIC16/17 microcontrollers employ an advanced RISC architecture. The PIC16CXX microcontroller family has enhanced core features, eight-level deep stack, and multiple internal and external interrupt sources. The separate instruction and data buses of the Harvard architecture allow a 14-bit wide instruction word with separate 8-bit wide data. The two stage instruction pipeline allows all instructions to execute in a single cycle, except for program branches (which require two cycles). A total of 35 instructions (reduced instruction set) are available. Additionally, a large register set gives some of the architectural innovations used to achieve a very high performance.

PIC16CXX microcontrollers typically achieve a 2:1 code compression and a 4:1 speed improvement over other 8-bit microcontrollers in their class.

The **PIC16C61** device has 36 bytes of RAM and 13 I/O pins. In addition a timer/counter is available.

The **PIC16C62/62A/R62** devices have 128 bytes of RAM and 22 I/O pins. In addition, several peripheral features are available, including: three timer/counters, one Capture/Compare/PWM module and one serial port. The Synchronous Serial Port can be configured as either a 3-wire Serial Peripheral Interface (SPI™) or the two-wire Inter-Integrated Circuit (I²C) bus.

The **PIC16C63** device has 192 bytes of RAM and 22 I/O pins. In addition, several peripheral features are available, including: three timer/counters, two Capture/Compare/PWM modules and two serial ports. The Synchronous Serial Port can be configured as either a 3-wire Serial Peripheral Interface (SPI) or the two-wire Inter-Integrated Circuit (I²C) bus. The Universal Synchronous Asynchronous Receiver Transmitter (USART) is also known as a Serial Communications Interface or SCI.

The **PIC16C64/64A/R64** devices have 128 bytes of RAM and 33 I/O pins. In addition, several peripheral features are available, including: three timer/counters, one Capture/Compare/PWM module and one serial port. The Synchronous Serial Port can be configured as either a 3-wire Serial Peripheral Interface (SPI) or the two-wire Inter-Integrated Circuit (I²C) bus. An 8-bit Parallel Slave Port is also provided.

The **PIC16C65/65A** devices have 192 bytes of RAM and 33 I/O pins. In addition, several peripheral features are available, including: three timer/counters, two Capture/Compare/PWM modules and two serial ports. The Synchronous Serial Port can be configured as either a 3-wire Serial Peripheral Interface (SPI) or the two-wire Inter-Integrated Circuit (I²C) bus. The Universal Synchronous Asynchronous Receiver Transmitter (USART) is also known as a Serial Communications Interface or SCI. An 8-bit Parallel Slave Port is also provided.

The PIC16C6X device family has special features to reduce external components, thus reducing cost, enhancing system reliability and reducing power consumption. There are four oscillator options, of which the single pin RC oscillator provides a low-cost solution, the LP oscillator minimizes power consumption, XT is a standard crystal, and the HS is for High Speed crystals. The SLEEP (power-down) mode offers a power saving mode. The user can wake the chip from SLEEP through several external and internal interrupts, and reset(s).

A highly reliable Watchdog Timer with its own on-chip RC oscillator provides protection against software lock-up.

A UV erasable Cerdip packaged version is ideal for code development, while the cost-effective One-Time-Programmable (OTP) version is suitable for production in any volume.

The PIC16C6X family fits perfectly in applications ranging from high-speed automotive and appliance control to low-power remote sensors, keyboards and telecom processors. The EPROM technology makes customization of application programs (transmitter codes, motor speeds, receiver frequencies, etc.) extremely fast and convenient. The small footprint packages make this microcontroller series perfect for all applications with space limitations. Low-cost, low-power, high performance, ease-of-use, and I/O flexibility make the PIC16C6X very versatile even in areas where no microcontroller use has been considered before (e.g. timer functions, serial communication, capture and compare, PWM functions, and co-processor applications).

### 1.1 Family and Upward Compatibility

Those users familiar with the PIC16C5X family of microcontrollers will realize that this is an enhanced version of the PIC16C5X architecture. Please refer to Appendix A for a detailed list of enhancements. Code written for PIC16C5X can be easily ported to PIC16CXX family of devices (Appendix B).

### 1.2 Development Support

The PIC16CXX family is supported by a full-featured macro assembler, a software simulator, an in-circuit emulator, a low-cost development programmer, and a full-featured programmer. A "C" compiler and fuzzy logic support tools are also available.

---

All PIC16/17 family devices have Power-on Reset, selectable Watchdog Timer, selectable code protect, and high I/O current capability.

All PIC16CXX family devices use serial programming with clock pin RB6 and data pin RB7.

Note 1: Please contact your local sales office for availability of these devices.

## 2.0 PIC16C6X DEVICE VARIETIES

A variety of frequency ranges and packaging options are available. Depending on application and production requirements, the proper device option can be selected using the information in the PIC16C6X Product Identification System section at the end of this data sheet. When placing orders, please use that page of the data sheet to specify the correct part number.

For the PIC16C6X family of devices, there are four device "types" as indicated in the device number:

1. **C**, as in PIC16**C**64. These devices have EPROM type memory and operate over the standard voltage range.
2. **LC**, as in PIC16**LC**64. These devices have EPROM type memory and operate over an extended voltage range.
3. **CR**, as in PIC16**CR**64. These devices have ROM program memory and operate over the standard voltage range.
4. **LCR**, as in PIC16**LCR**64. These devices have ROM program memory and operate over an extended voltage range.

### 2.1 UV Erasable Devices

The UV erasable version, offered in Cerdip package, is optimal for prototype development and pilot programs.

The UV erasable version can be erased and reprogrammed to any of the configuration modes. Microchip's PICSTART® and PRO MATE™ programmers both support the PIC16C6X. Third party programmers also are available; refer to the *Microchip Third Party Guide* for a list of sources.

### 2.2 One-Time-Programmable (OTP) Devices

The availability of OTP devices is especially useful for customers who need the flexibility for frequent code updates and small volume applications.

The OTP devices, packaged in plastic packages, permit the user to program them once. In addition to the program memory, the configuration bits must also be programmed.

### 2.3 Quick-Turnaround-Production (QTP) Devices

Microchip offers a QTP Programming Service for factory production orders. This service is made available for users who choose not to program a medium to high quantity of units and whose code patterns have stabilized. The devices are identical to the OTP devices but with all EPROM locations and configuration options already programmed by the factory. Certain code and prototype verification procedures apply before production shipments are available. Please contact your local Microchip Technology sales office for more details.

### 2.4 Serialized Quick-Turnaround Production (SQTP<sup>SM</sup>) Devices

Microchip offers a unique programming service where a few user-defined locations in each device are programmed with different serial numbers. The serial numbers may be random, pseudo-random or sequential.

Serial programming allows each device to have a unique number which can serve as an entry-code, password or ID number.

ROM devices do not allow serialization information in the program memory space. The user may have this information programmed in the data memory space.

For information on submitting ROM code, please contact your regional sales office.

### 2.5 Read Only Memory (ROM) Devices

Microchip offers masked ROM versions of several of the highest volume parts, thus giving customers a low cost option for high volume, mature products.

For information on submitting ROM code, please contact your regional sales office.

# PIC16C6X

---

NOTES:



## 3.0 ARCHITECTURAL OVERVIEW

The high performance of the PIC16CXX family can be attributed to a number of architectural features commonly found in RISC microprocessors. To begin with, the PIC16CXX uses a Harvard architecture, in which, program and data are accessed from separate memories using separate buses. This improves bandwidth over traditional von Neumann architecture where program and data may be fetched from the same memory using the same bus. Separating program and data buses further allows instructions to be sized differently than 8-bit wide data words. Instruction opcodes are 14-bits wide making it possible to have all single word instructions. A 14-bit wide program memory access bus fetches a 14-bit instruction in a single cycle. A two-stage pipeline overlaps fetch and execution of instructions (Example 3-1). Consequently, all instructions execute in a single cycle (200 ns @ 20 MHz) except for program branches.

The PIC16C61 addresses 1K x 14 of program memory. The PIC16C62/62A/R62/64/64A/R64 addresses 2K x 14 of program memory, and the PIC16C63/65/65A devices address 4K x 14 of program memory. All program memory is internal.

The PIC16CXX can directly or indirectly address its register files or data memory. All special function registers including the program counter are mapped in the data memory. The PIC16CXX has an orthogonal (symmetrical) instruction set that makes it possible to carry out any operation on any register using any addressing mode. This symmetrical nature and lack of "special optimal situations" makes programming with the PIC16CXX simple yet efficient, thus significantly reducing the learning curve.

The PIC16CXX device contains an 8-bit ALU and working register (W). The ALU is a general purpose arithmetic unit. It performs arithmetic and Boolean functions between data in the working register and any register file.

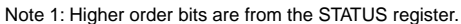
The ALU is 8-bits wide and capable of addition, subtraction, shift, and logical operations. Unless otherwise mentioned, arithmetic operations are two's complement in nature. In two-operand instructions, typically one operand is the working register (W register), the other operand is a file register or an immediate constant. In single operand instructions, the operand is either the W register or a file register.

The W register is an 8-bit working register used for ALU operations. It is not an addressable register.

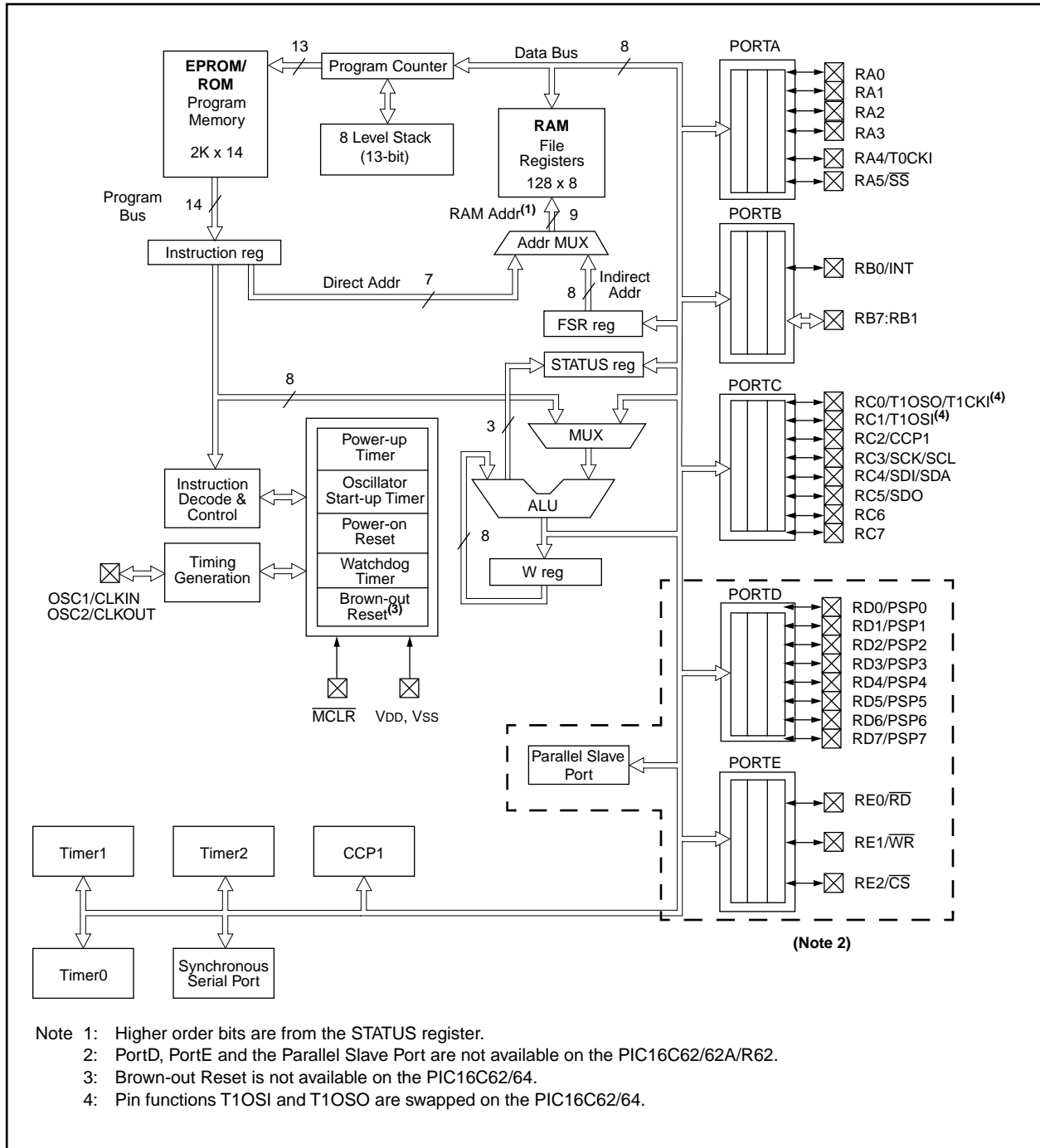
Depending upon the instruction executed, the ALU may affect the values of the Carry (C), Digit Carry (DC), and Zero (Z) bits in the STATUS register. Bits C and DC operate as a borrow and digit borrow out bit, respectively, in subtraction. See the `SUBLW` and `SUBWF` instructions for examples.

---

\_\_\_\_\_

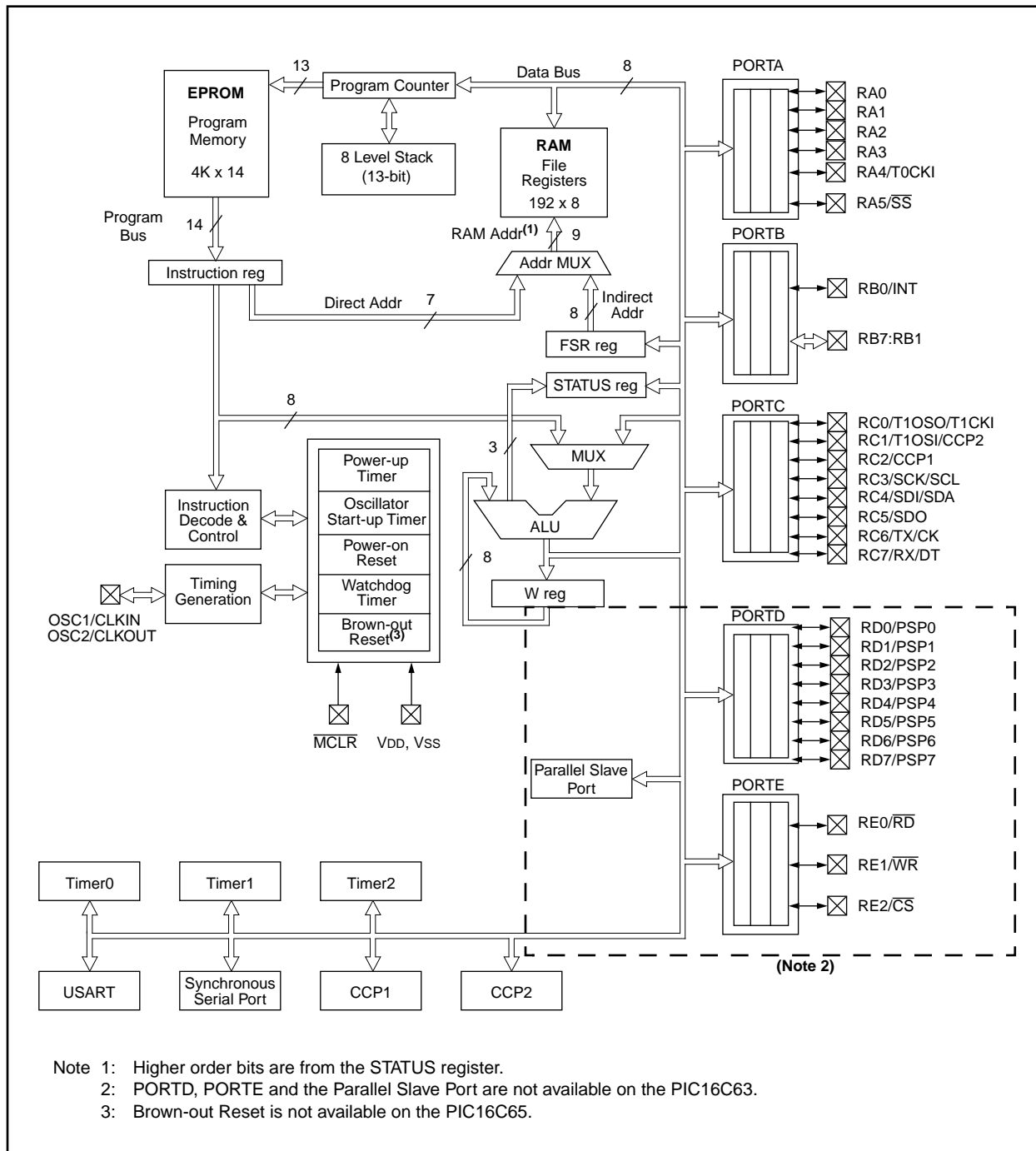


**FIGURE 3-2: PIC16C62/62A/R62/64/64A/R64 BLOCK DIAGRAM**



# PIC16C6X

**FIGURE 3-3: PIC16C63/65/65A BLOCK DIAGRAM**



**TABLE 3-1: PIC16C61 PINOUT DESCRIPTION**

Pin Name	DIP Pin#	SOIC Pin#	Pin Type	Buffer Type	Description
OSC1/CLKIN	16	16	I	ST/CMOS <sup>(1)</sup>	Oscillator crystal input/external clock source input.
OSC2/CLKOUT	15	15	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	4	4	I/P	ST	Master clear reset input/programming voltage input. This pin is an active low reset to the device.
RA0	17	17	I/O	TTL	PORTA is a bi-directional I/O port.  Can also be selected to be the clock input to the Timer0 timer/counter. Output is open drain type.
RA1	18	18	I/O	TTL	
RA2	1	1	I/O	TTL	
RA3	2	2	I/O	TTL	
RA4/T0CKI	3	3	I/O	ST	
RB0/INT	6	6	I/O	TTL/ST <sup>(2)</sup>	PORTB is a bi-directional I/O port. PORTB can be software programmed for internal weak pull-up on all inputs. RB0/INT can also be selected as an 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	7	7	I/O	TTL	
RB2	8	8	I/O	TTL	
RB3	9	9	I/O	TTL	
RB4	10	10	I/O	TTL	
RB5	11	11	I/O	TTL	
RB6	12	12	I/O	TTL/ST <sup>(3)</sup>	
RB7	13	13	I/O	TTL/ST <sup>(3)</sup>	
VSS	5	5	P	—	Ground reference for logic and I/O pins.
VDD	14	14	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: This buffer is a Schmitt Trigger input when configured in RC oscillator mode and a CMOS input otherwise.  
 Note 2: This buffer is a Schmitt Trigger input when configured as the external interrupt.  
 Note 3: This buffer is a Schmitt Trigger input when used in serial programming mode.

# PIC16C6X

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

Pin Name	DIP, SSOP Pin#	Pin Type	Buffer Type	Description
OSC1/CLKIN	9	I	ST/CMOS <sup>(3)</sup>	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/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.  Can also be selected to be the clock input to the Timer0 timer/counter. Output is open drain type. 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/SS	7	I/O	TTL	
RB0/INT	21	I/O	TTL/ST <sup>(4)</sup>	PORTB is a bi-directional I/O port. PORTB can be software programmed for internal weak pull-up on all inputs. RB0/INT can also be selected as an 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 <sup>(5)</sup>	
RB7	28	I/O	TTL/ST <sup>(5)</sup>	
RC0/T1OSO <sup>(1)</sup> /T1CKI	11	I/O	ST	PORTC is a bi-directional I/O port. RC0/T1OSO/T1CKI can also be selected as a Timer1 oscillator output <sup>(1)</sup> /Timer1 clock input. RC1/T1OSI can also be selected as a Timer1 oscillator input <sup>(1)</sup> or Capture2 input/Compare2 output/PWM2 output <sup>(2)</sup> . RC2/CCP1 can also be selected as a Capture1 input/Compare1 output/PWM1 output. RC3/SCK/SCL can also be selected as the synchronous serial clock input/output for both SPI and I <sup>2</sup> C modes. RC4/SDI/SDA can also be selected as the SPI Data In (SPI mode) or data I/O (I <sup>2</sup> C mode). RC5/SDO can also be selected as the SPI Data Out (SPI mode). RC6/TX/CK can also be selected as Asynchronous Transmit <sup>(2)</sup> or USART Synchronous Clock <sup>(2)</sup> . RC7/RX/DT can also be selected as the Asynchronous Receive <sup>(2)</sup> or USART Synchronous Data <sup>(2)</sup> .
RC1/T1OSI <sup>(1)</sup> /CCP2 <sup>(2)</sup>	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 <sup>(2)</sup>	17	I/O	ST	
RC7/RX/DT <sup>(2)</sup>	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.

**TABLE 3-3: PIC16C64/64A/R64/65/65A PINOUT DESCRIPTION**

Pin Name	DIP Pin#	PLCC Pin#	MQFP Pin#	Pin Type	Buffer Type	Description
OSC1/CLKIN	13	14	30	I	ST/CMOS <sup>(3)</sup>	Oscillator crystal input/external clock source input.
OSC2/CLKOUT	14	15	31	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	2	18	I/P	ST	Master clear reset input/programming voltage input. This pin is an active low reset to the device.
RA0	2	3	19	I/O	TTL	PORTA is a bi-directional I/O port.  Can also be selected to be the clock input to the Timer0 timer/counter. Output is open drain type. Slave select for the synchronous serial port.
RA1	3	4	20	I/O	TTL	
RA2	4	5	21	I/O	TTL	
RA3	5	6	22	I/O	TTL	
RA4/T0CKI	6	7	23	I/O	ST	
RA5/SS	7	8	24	I/O	TTL	
RB0/INT	33	36	8	I/O	TTL/ST <sup>(4)</sup>	PORTB is a bi-directional I/O port. PORTB can be software programmed for internal weak pull-up on all inputs. RB0/INT can also be selected as an 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	34	37	9	I/O	TTL	
RB2	35	38	10	I/O	TTL	
RB3	36	39	11	I/O	TTL	
RB4	37	41	14	I/O	TTL	
RB5	38	42	15	I/O	TTL	
RB6	39	43	16	I/O	TTL/ST <sup>(5)</sup>	
RB7	40	44	17	I/O	TTL/ST <sup>(5)</sup>	
RC0/T1OSO <sup>(1)</sup> /T1CKI	15	16	32	I/O	ST	PORTC is a bi-directional I/O port. RC0/T1OSO/T1CKI can also be selected as a Timer1 oscillator output <sup>(1)</sup> /Timer1 clock input. RC1/T1OSI <sup>(1)</sup> /CCP2 <sup>(2)</sup> can also be selected as a Timer1 oscillator input <sup>(1)</sup> or Capture2 input/Compare2 output/PWM2 output <sup>(2)</sup> . RC2/CCP1 can also be selected as a Capture1 input/Compare1 output/PWM1 output. RC3/SCK/SCL can also be selected as the synchronous serial clock input/output for both SPI and I <sup>2</sup> C modes. RC4/SDI/SDA can also be selected as the SPI Data In (SPI mode) or data I/O (I <sup>2</sup> C mode). RC5/SDO can also be selected as the SPI Data Out (SPI mode). RC6/TX/CK <sup>(2)</sup> can also be selected as Asynchronous Transmit <sup>(2)</sup> or USART Synchronous Clock <sup>(2)</sup> . RC7/RX/DT <sup>(2)</sup> can also be selected as the Asynchronous Receive <sup>(2)</sup> or USART Synchronous Data <sup>(2)</sup> .
RC1/T1OSI <sup>(1)</sup> /CCP2 <sup>(2)</sup>	16	18	35	I/O	ST	
RC2/CCP1	17	19	36	I/O	ST	
RC3/SCK/SCL	18	20	37	I/O	ST	
RC4/SDI/SDA	23	25	42	I/O	ST	
RC5/SDO	24	26	43	I/O	ST	
RC6/TX/CK <sup>(2)</sup>	25	27	44	I/O	ST	
RC7/RX/DT <sup>(2)</sup>	26	29	1	I/O	ST	

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 PIC16C64.  
 2: CCP2 and the USART are not available on the PIC16C64/64A/R64.  
 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.  
 6: This buffer is a Schmitt Trigger input when configured as general purpose I/O and a TTL input when used in the Parallel Slave Port mode (for interfacing to a microprocessor bus).

# PIC16C6X

**TABLE 3-3: PIC16C64/64A/R64/65/65A PINOUT DESCRIPTION (Cont'd)**

Pin Name	DIP Pin#	PLCC Pin#	MQFP Pin#	Pin Type	Buffer Type	Description
RD0/PSP0	19	21	38	I/O	ST/TTL <sup>(6)</sup>	PORTD can be a bi-directional I/O port or parallel slave port for interfacing to a microprocessor bus.
RD1/PSP1	20	22	39	I/O	ST/TTL <sup>(6)</sup>	
RD2/PSP2	21	23	40	I/O	ST/TTL <sup>(6)</sup>	
RD3/PSP3	22	24	41	I/O	ST/TTL <sup>(6)</sup>	
RD4/PSP4	27	30	2	I/O	ST/TTL <sup>(6)</sup>	
RD5/PSP5	28	31	3	I/O	ST/TTL <sup>(6)</sup>	
RD6/PSP6	29	32	4	I/O	ST/TTL <sup>(6)</sup>	
RD7/PSP7	30	33	5	I/O	ST/TTL <sup>(6)</sup>	
RE0/ $\overline{RD}$	8	9	25	I/O	ST/TTL <sup>(6)</sup>	PORTE is a bi-directional I/O port. RE0/ $\overline{RD}$ read control for parallel slave port. RE1/ $\overline{WR}$ write control for parallel slave port. RE2/ $\overline{CS}$ select control for parallel slave port.
RE1/ $\overline{WR}$	9	10	26	I/O	ST/TTL <sup>(6)</sup>	
RE2/ $\overline{CS}$	10	11	27	I/O	ST/TTL <sup>(6)</sup>	
Vss	12,31	13,34	6,29	P	—	Ground reference for logic and I/O pins.
VDD	11,32	12,35	7,28	P	—	Positive supply for logic and I/O pins.
NC	—	1,17, 28,40	12,13, 33,34	—	—	These pins are not internally connected. These pins should be left unconnected.

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 PIC16C64.  
 2: CCP2 and the USART are not available on the PIC16C64/64A/R64.  
 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.  
 6: This buffer is a Schmitt Trigger input when configured as general purpose I/O and a TTL input when used in the Parallel Slave Port mode (for interfacing to a microprocessor bus).



## 3.1 Clocking Scheme/Instruction Cycle

The clock input (from OSC1) is internally divided by four to generate four non-overlapping quadrature clocks namely Q1, Q2, Q3, and Q4. Internally, the program counter (PC) is incremented every Q1, the instruction is fetched from the program memory and latched into the instruction register in Q4. The instruction is decoded and executed during the following Q1 through Q4. The clock and instruction execution flow is shown in Figure 3-4.

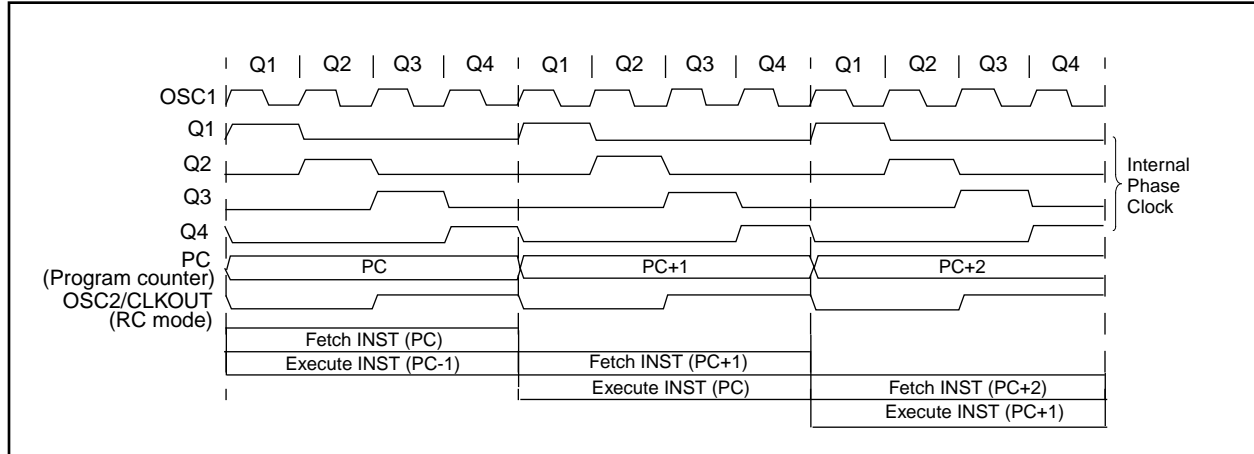
## 3.2 Instruction Flow/Pipelining

An "Instruction Cycle" consists of four Q cycles (Q1, Q2, Q3, and Q4). The instruction fetch and execute are pipelined such that fetch takes one instruction cycle while decode and execute takes another instruction cycle. However, due to the pipelining, each instruction effectively executes in one cycle. If an instruction causes the program counter to change (e.g. GOTO) then two cycles are required to complete the instruction (Example 3-1).

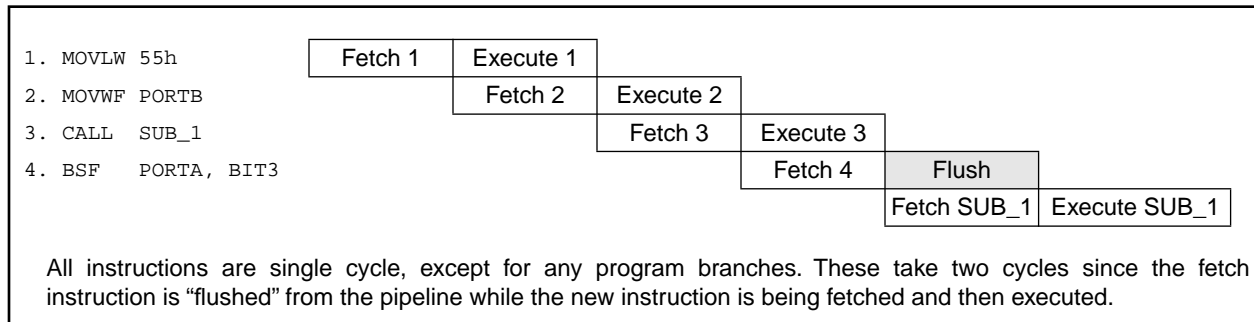
A fetch cycle begins with the program counter (PC) incrementing in Q1.

In the execution cycle, the fetched instruction is latched into the "Instruction Register (IR)" in cycle Q1. This instruction is then decoded and executed during the Q2, Q3, and Q4 cycles. Data memory is read during Q2 (operand read) and written during Q4 (destination write).

**FIGURE 3-4: CLOCK/INSTRUCTION CYCLE**



**EXAMPLE 3-1: INSTRUCTION PIPELINE FLOW**



# PIC16C6X

---

NOTES:

## 4.0 MEMORY ORGANIZATION

### Applicable Devices

61	62	62A	R62	63	64	64A	R64	65	65A
----	----	-----	-----	----	----	-----	-----	----	-----

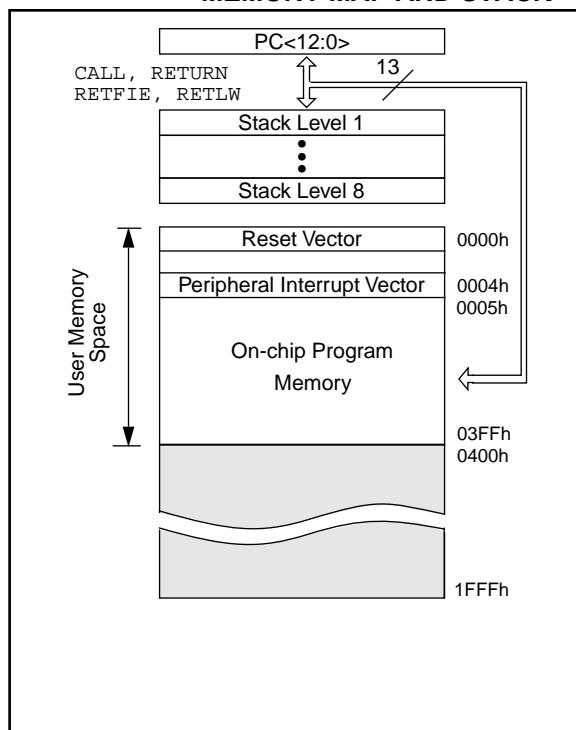
### 4.1 Program Memory Organization

The PIC16C6X family has a 13-bit program counter capable of addressing an 8K x 14 program memory space.

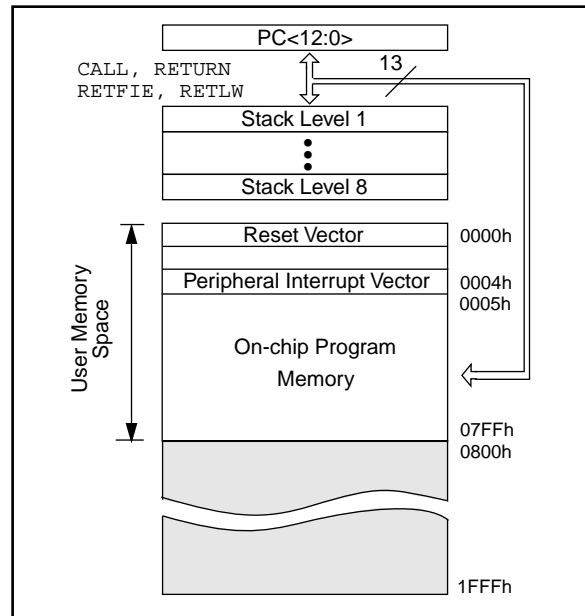
For the PIC16C61, only the first 1K x 14 (0000h-03FFh) is physically implemented. For the PIC16C62/62A/R62/64/64A/R64, only the first 2K x 14 (0000h-07FFh) are physically implemented, and for the PIC16C63/65/65A, only the first 4K x 14 (0000h-0FFFh) are physically implemented. Accessing a location above the physically implemented address will cause a wrap-around.

The reset vector is at 0000h and the interrupt vector is at 0004h.

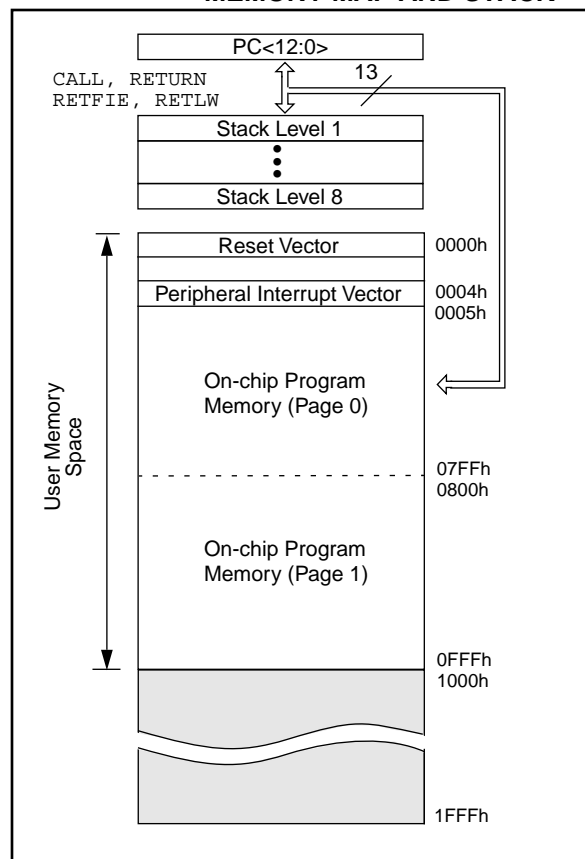
**FIGURE 4-1: PIC16C61 PROGRAM MEMORY MAP AND STACK**



**FIGURE 4-2: PIC16C62/62A/R62/64/64A/R64 PROGRAM MEMORY MAP AND STACK**



**FIGURE 4-3: PIC16C63/65/65A PROGRAM MEMORY MAP AND STACK**



# PIC16C6X

## 4.2 Data Memory Organization

Applicable Devices									
61	62	62A	R62	63	64	64A	R64	65	65A

The data memory is partitioned into two Banks which contain the General Purpose Registers and the Special Function Registers. Bit RP0 is the bank select bit.

RP0 (STATUS<5>) = 1 → Bank 1

RP0 (STATUS<5>) = 0 → Bank 0

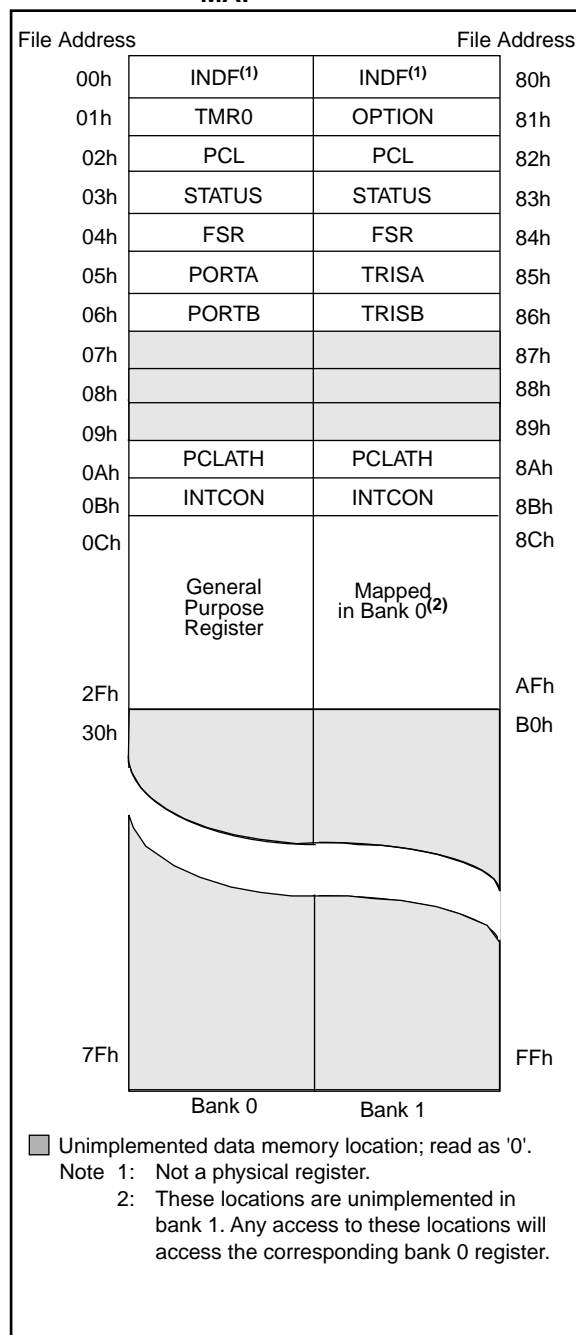
Each Bank extends up to 7Fh (128 bytes). The lower locations of each Bank are reserved for the Special Function Registers. Above the Special Function Registers are General Purpose Registers implemented as static RAM. Both Bank 0 and Bank 1 contain special function registers. Some "high use" special function registers from Bank 0 are mirrored in Bank 1 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).

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

**FIGURE 4-4: PIC16C61 REGISTER FILE MAP**



**FIGURE 4-5: PIC16C62/62A/R62/64/64A/R64 REGISTER FILE MAP**

File Address		File Address	
00h	INDF <sup>(1)</sup>	INDF <sup>(1)</sup>	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	PORTC	TRISC	87h
08h	PORTD <sup>(2)</sup>	TRISD <sup>(2)</sup>	88h
09h	PORTE <sup>(2)</sup>	TRISE <sup>(2)</sup>	89h
0Ah	PCLATH	PCLATH	8Ah
0Bh	INTCON	INTCON	8Bh
0Ch	PIR1	PIE1	8Ch
0Dh			8Dh
0Eh	TMR1L	PCON	8Eh
0Fh	TMR1H		8Fh
10h	T1CON		90h
11h	TMR2		91h
12h	T2CON	PR2	92h
13h	SSPBUF	SSPADD	93h
14h	SSPCON	SSPSTAT	94h
15h	CCPR1L		95h
16h	CCPR1H		96h
17h	CCP1CON		97h
18h			98h
1Fh			9Fh
20h	General Purpose Register	General Purpose Register	A0h
			BFh
			C0h
7Fh			FFh
Bank 0		Bank 1	

Unimplemented data memory location; read as '0'.

Note 1:

Not a physical register.

Note 2:

Pin PORTD and PORTE are not available on the PIC16C62/62A/R62.

**FIGURE 4-6: PIC16C63/65/65A REGISTER FILE MAP**

File Address		File Address	
00h	INDF <sup>(1)</sup>	INDF <sup>(1)</sup>	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	PORTC	TRISC	87h
08h	PORTD <sup>(2)</sup>	TRISD <sup>(2)</sup>	88h
09h	PORTE <sup>(2)</sup>	TRISE <sup>(2)</sup>	89h
0Ah	PCLATH	PCLATH	8Ah
0Bh	INTCON	INTCON	8Bh
0Ch	PIR1	PIE1	8Ch
0Dh	PIR2	PIE2	8Dh
0Eh	TMR1L	PCON	8Eh
0Fh	TMR1H		8Fh
10h	T1CON		90h
11h	TMR2		91h
12h	T2CON	PR2	92h
13h	SSPBUF	SSPADD	93h
14h	SSPCON	SSPSTAT	94h
15h	CCPR1L		95h
16h	CCPR1H		96h
17h	CCP1CON		97h
18h	RCSTA	TXSTA	98h
19h	TXREG	SPBRG	99h
1Ah	RCREG		9Ah
1Bh	CCPR2L		9Bh
1Ch	CCPR2H		9Ch
1Dh	CCP2CON		9Dh
1Eh			9Eh
1Fh			9Fh
20h	General Purpose Register	General Purpose Register	A0h
7Fh			FFh

Bank 0                      Bank 1

☐ Unimplemented data memory location; read as '0'.

Note 1: Not a physical register

Note 2: Pin PORTD and PORTE are not available on the PIC16C63.

# PIC16C6X

## 4.2.2 SPECIAL FUNCTION REGISTERS:

The Special Function Registers are registers used by the CPU and peripheral functions for controlling the desired operation of the device. These registers are implemented as static RAM.

The special function registers can be classified into two sets (core and peripheral). The registers associated with the “core” functions are described in this section and those related to the operation of the peripheral features are described in the section of that peripheral feature.

**TABLE 4-1: SPECIAL FUNCTION REGISTERS FOR THE PIC16C61**

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR	Value on all other resets <sup>(3)</sup>
Bank 0											
00h <sup>(1)</sup>	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	0000 0000
01h	TMR0	Timer0 module's register								xxxx xxxx	uuuu uuuu
02h <sup>(1)</sup>	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	0000 0000
03h <sup>(1)</sup>	STATUS	IRP <sup>(4)</sup>	RP1 <sup>(4)</sup>	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	000q quuu
04h <sup>(1)</sup>	FSR	Indirect data memory address pointer								xxxx xxxx	uuuu uuuu
05h	PORTA	—	—	—	PORTA Data Latch when written: PORTA pins when read					---x xxxx	---u uuuu
06h	PORTB	PORTB Data Latch when written: PORTB pins when read								xxxx xxxx	uuuu uuuu
07h	—	Unimplemented								—	—
08h	—	Unimplemented								—	—
09h	—	Unimplemented								—	—
0Ah <sup>(1,2)</sup>	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter					---0 0000	---0 0000
0Bh <sup>(1)</sup>	INTCON	GIE	—	TOIE	INTE	RBIE	TOIF	INTF	RBIF	0-00 000x	0-00 000u
Bank 1											
80h <sup>(1)</sup>	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	0000 0000
81h	OPTION	$\overline{RBP}_U$	INTEDG	TOCS	TOSE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
82h <sup>(1)</sup>	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	0000 0000
83h <sup>(1)</sup>	STATUS	IRP <sup>(4)</sup>	RP1 <sup>(4)</sup>	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	000q quuu
84h <sup>(1)</sup>	FSR	Indirect data memory address pointer								xxxx xxxx	uuuu uuuu
85h	TRISA	—	—	—	PORTA Data Direction Register					---1 1111	---1 1111
86h	TRISB	PORTB Data Direction Control Register								1111 1111	1111 1111
87h	—	Unimplemented								—	—
88h	—	Unimplemented								—	—
89h	—	Unimplemented								—	—
8Ah <sup>(1,2)</sup>	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter					---0 0000	---0 0000
8Bh <sup>(1)</sup>	INTCON	GIE	—	TOIE	INTE	RBIE	TOIF	INTF	RBIF	0-00 000x	0-00 000u

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

Shaded locations are unimplemented and read as '0'

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

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

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

4: The IRP and RP1 bits are reserved on the PIC16C6X, always maintain these bits clear.

**TABLE 4-2: SPECIAL FUNCTION REGISTERS FOR THE PIC16C62/62A/R62**

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR BOR	Value on all other resets <sup>(3)</sup>
Bank 0											
00h <sup>(1)</sup>	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	0000 0000
01h	TMR0	Timer0 module's register								xxxx xxxx	uuuu uuuu
02h <sup>(1)</sup>	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	0000 0000
03h <sup>(1)</sup>	STATUS	IRP <sup>(5)</sup>	RP1 <sup>(5)</sup>	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	000q quuu
04h <sup>(1)</sup>	FSR	Indirect data memory address pointer								xxxx xxxx	uuuu uuuu
05h	PORTA	—	—	PORTA Data Latch when written: PORTA pins when read						--xx xxxx	--uu uuuu
06h	PORTB	PORTB Data Latch when written: PORTB pins when read								xxxx xxxx	uuuu uuuu
07h	PORTC	PORTC Data Latch when written: PORTC pins when read								xxxx xxxx	uuuu uuuu
08h	—	Unimplemented								—	—
09h	—	Unimplemented								—	—
0Ah <sup>(1,2)</sup>	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter					---0 0000	---0 0000
0Bh <sup>(1)</sup>	INTCON	GIE	PEIE	TOIE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	(6)	(6)	—	—	SSPIF	CCP1IF	TMR2IF	TMR1IF	00-- 0000	00-- 0000
0Dh	—	Unimplemented								—	—
0Eh	TMR1L	Holding register for the Least Significant Byte of the 16-bit TMR1 register								xxxx xxxx	uuuu uuuu
0Fh	TMR1H	Holding register for the Most Significant Byte of the 16-bit TMR1 register								xxxx xxxx	uuuu uuuu
10h	T1CON	—	—	T1CKPS1	T1CKPS0	T1OSCEN	T1SYNC	TMR1CS	TMR1ON	--00 0000	--uu uuuu
11h	TMR2	Timer2 module's register								0000 0000	0000 0000
12h	T2CON	—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	-000 0000	-000 0000
13h	SSPBUF	Synchronous Serial Port Receive Buffer/Transmit Register								xxxx xxxx	uuuu uuuu
14h	SSPCON	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	0000 0000
15h	CCPR1L	Capture/Compare/PWM1 (LSB)								xxxx xxxx	uuuu uuuu
16h	CCPR1H	Capture/Compare/PWM1 (MSB)								xxxx xxxx	uuuu uuuu
17h	CCP1CON	—	—	CCP1X	CCP1Y	CCP1M3	CCP1M2	CCP1M1	CCP1M0	--00 0000	--00 0000
18h-1Fh	—	Unimplemented								—	—

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

Shaded locations are unimplemented, read as '0'.

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

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

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

4: The  $\overline{BOR}$  bit is reserved on the PIC16C62, always maintain this bit set.

5: The IRP and RP1 bits are reserved on the PIC16C6X, 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.

# PIC16C6X

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

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

Shaded locations are unimplemented, read as '0'.

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

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

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

4: The  $\overline{BOR}$  bit is reserved on the PIC16C62, always maintain this bit set.

5: The IRP and RP1 bits are reserved on the PIC16C6X, 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**

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR BOR	Value on all other resets <sup>(3)</sup>
Bank 0											
00h <sup>(1)</sup>	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	0000 0000
01h	TMR0	Timer0 module's register								xxxx xxxx	uuuu uuuu
02h <sup>(1)</sup>	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	0000 0000
03h <sup>(1)</sup>	STATUS	IRP <sup>(4)</sup>	RP1 <sup>(4)</sup>	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	000q quuu
04h <sup>(1)</sup>	FSR	Indirect data memory address pointer								xxxx xxxx	uuuu uuuu
05h	PORTA	—	—	PORTA Data Latch when written: PORTA pins when read						--xx xxxx	--uu uuuu
06h	PORTB	PORTB Data Latch when written: PORTB pins when read								xxxx xxxx	uuuu uuuu
07h	PORTC	PORTC Data Latch when written: PORTC pins when read								xxxx xxxx	uuuu uuuu
08h	—	Unimplemented								—	—
09h	—	Unimplemented								—	—
0Ah <sup>(1,2)</sup>	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter					---0 0000	---0 0000
0Bh <sup>(1)</sup>	INTCON	GIE	PEIE	TOIE	INTE	RBIE	TOIF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	(5)	(5)	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	00-- 0000	00-- 0000
0Dh	PIR2	—	—	—	—	—	—	—	CCP2IF	---- --0	---- --0
0Eh	TMR1L	Holding register for the Least Significant Byte of the 16-bit TMR1 register								xxxx xxxx	uuuu uuuu
0Fh	TMR1H	Holding register for the Most Significant Byte of the 16-bit TMR1 register								xxxx xxxx	uuuu uuuu
10h	T1CON	—	—	T1CKPS1	T1CKPS0	T1OSCEN	$\overline{T1SYN\overline{C}}$	TMR1CS	TMR1ON	--00 0000	--uu uuuu
11h	TMR2	Timer2 module's register								0000 0000	0000 0000
12h	T2CON	—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	-000 0000	-000 0000
13h	SSPBUF	Synchronous Serial Port Receive Buffer/Transmit Register								xxxx xxxx	uuuu uuuu
14h	SSPCON	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	0000 0000
15h	CCPR1L	Capture/Compare/PWM1 (LSB)								xxxx xxxx	uuuu uuuu
16h	CCPR1H	Capture/Compare/PWM1 (MSB)								xxxx xxxx	uuuu uuuu
17h	CCP1CON	—	—	CCP1X	CCP1Y	CCP1M3	CCP1M2	CCP1M1	CCP1M0	--00 0000	--00 0000
18h	RCSTA	SPEN	RX9	SREN	CREN	—	FERR	OERR	RX9D	0000 -00x	0000 -00x
19h	TXREG	USART Transmit Data Register								0000 0000	0000 0000
1Ah	RCREG	USART Receive Data Register								0000 0000	0000 0000
1Bh	CCPR2L	Capture/Compare/PWM2 (LSB)								xxxx xxxx	uuuu uuuu
1Ch	CCPR2H	Capture/Compare/PWM2 (MSB)								xxxx xxxx	uuuu uuuu
1Dh	CCP2CON	—	—	CCP2X	CCP2Y	CCP2M3	CCP2M2	CCP2M1	CCP2M0	--00 0000	--00 0000
1Eh-1Fh	—	Unimplemented								—	—

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

Shaded locations are unimplemented, read as '0'.

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

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 PIC16C6X, always maintain these bits clear.

5: PIE1<7:6> and PIR1<7:6> are reserved, always maintain these bits clear.

# PIC16C6X

**TABLE 4-3: SPECIAL FUNCTION REGISTERS FOR THE PIC16C63 (Cont.'d)**

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR BOR	Value on all other resets <sup>(3)</sup>	
Bank 1												
80h <sup>(1)</sup>	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	0000 0000	
81h	OPTION	RBP <sub>U</sub>	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111	
82h <sup>(1)</sup>	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	0000 0000	
83h <sup>(1)</sup>	STATUS	IRP <sup>(4)</sup>	RP1 <sup>(4)</sup>	RP0	T <sub>0</sub>	P <sub>D</sub>	Z	DC	C	0001 1xxx	000q quuu	
84h <sup>(1)</sup>	FSR	Indirect data memory address pointer								xxxx xxxx	uuuu uuuu	
85h	TRISA	—	—	PORTA Data Direction Register						--11 1111	--11 1111	
86h	TRISB	PORTB Data Direction Register								1111 1111	1111 1111	
87h	TRISC	PORTC Data Direction Register								1111 1111	1111 1111	
88h	—	Unimplemented								—	—	
89h	—	Unimplemented								—	—	
8Ah <sup>(1,2)</sup>	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter				---	0 0000	---	0 0000
8Bh <sup>(1)</sup>	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u	
8Ch	PIE1	(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 <sup>2</sup> C mode) Address Register								0000 0000	0000 0000	
94h	SSPSTAT	—	—	D/A	P	S	R/W	UA	BF	--00 0000	--00 0000	
95h	—	Unimplemented								—	—	
96h	—	Unimplemented								—	—	
97h	—	Unimplemented								—	—	
98h <sup>(2)</sup>	TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010	
99h <sup>(2)</sup>	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 PIC16C6X, always maintain these bits clear.

5: PIE1<7:6> and PIR1<7:6> are reserved, always maintain these bits clear.

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

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR BOR	Value on all other resets <sup>(3)</sup>
Bank 0											
00h <sup>(1)</sup>	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	0000 0000
01h	TMR0	Timer0 module's register								xxxx xxxx	uuuu uuuu
02h <sup>(1)</sup>	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	0000 0000
03h <sup>(1)</sup>	STATUS	IRP <sup>(5)</sup>	RP1 <sup>(5)</sup>	RP0	T <sub>O</sub>	P <sub>D</sub>	Z	DC	C	0001 1xxx	000q quuu
04h <sup>(1)</sup>	FSR	Indirect data memory address pointer								xxxx xxxx	uuuu uuuu
05h	PORTA	—	—	PORTA Data Latch when written: PORTA pins when read						--xx xxxx	--uu uuuu
06h	PORTB	PORTB Data Latch when written: PORTB pins when read								xxxx xxxx	uuuu uuuu
07h	PORTC	PORTC Data Latch when written: PORTC pins when read								xxxx xxxx	uuuu uuuu
08h	PORTD	PORTD Data Latch when written: PORTD pins when read								xxxx xxxx	uuuu uuuu
09h	PORTE	—	—	—	—	—	RE2	RE1	RE0	---- -xxx	---- -uuu
0Ah <sup>(1,2)</sup>	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter					---0 0000	---0 0000
0Bh <sup>(1)</sup>	INTCON	GIE	PEIE	TOIE	INTE	RBIE	TOIF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	PSPIF	(6)	—	—	SSPIF	CCP1IF	TMR2IF	TMR1IF	00-- 0000	00-- 0000
0Dh	—	Unimplemented								—	—
0Eh	TMR1L	Holding register for the Least Significant Byte of the 16-bit TMR1 register								xxxx xxxx	uuuu uuuu
0Fh	TMR1H	Holding register for the Most Significant Byte of the 16-bit TMR1 register								xxxx xxxx	uuuu uuuu
10h	T1CON	—	—	T1CKPS1	T1CKPS0	T1OSCEN	T1SYN <sub>C</sub>	TMR1CS	TMR1ON	--00 0000	--uu uuuu
11h	TMR2	Timer2 module's register								0000 0000	0000 0000
12h	T2CON	—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	-000 0000	-000 0000
13h	SSPBUF	Synchronous Serial Port Receive Buffer/Transmit Register								xxxx xxxx	uuuu uuuu
14h	SSPCON	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	0000 0000
15h	CCPR1L	Capture/Compare/PWM1 (LSB)								xxxx xxxx	uuuu uuuu
16h	CCPR1H	Capture/Compare/PWM1 (MSB)								xxxx xxxx	uuuu uuuu
17h	CCP1CON	—	—	CCP1X	CCP1Y	CCP1M3	CCP1M2	CCP1M1	CCP1M0	--00 0000	--00 0000
18h-1Fh	—	Unimplemented								—	—

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

Shaded locations are unimplemented, read as '0'.

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

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

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

4: The  $\overline{BOR}$  bit is reserved on the PIC16C64, always maintain this bit set.

5: The IRP and RP1 bits are reserved on the PIC16C6X, always maintain these bits clear.

6: PIE1<6> and PIR1<6> are reserved on the PIC16C6X, always maintain these bits clear.

# PIC16C6X

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

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR BOR	Value on all other resets <sup>(3)</sup>
Bank 1											
80h <sup>(1)</sup>	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	0000 0000
81h	OPTION	RBP $\bar{U}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
82h <sup>(1)</sup>	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	0000 0000
83h <sup>(1)</sup>	STATUS	IRP <sup>(5)</sup>	RP1 <sup>(5)</sup>	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	000q quuu
84h <sup>(1)</sup>	FSR	Indirect data memory address pointer								xxxx xxxx	uuuu uuuu
85h	TRISA	—	—	PORTA Data Direction Register						--11 1111	--11 1111
86h	TRISB	PORTB Data Direction Register								1111 1111	1111 1111
87h	TRISC	PORTC Data Direction Register								1111 1111	1111 1111
88h	TRISD	PORTD Data Direction Register								1111 1111	1111 1111
89h	TRISE	IBF	OBF	IBOV	PSPMODE	—	TRISE2	TRISE1	TRISE0	0000 -111	0000 -111
8Ah <sup>(1,2)</sup>	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter					---0 0000	---0 0000
8Bh <sup>(1)</sup>	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
8Ch	PIE1	PSPIE	(6)	—	—	SSPIE	CCP1IE	TMR2IE	TMR1IE	00-- 0000	00-- 0000
8Dh	—	Unimplemented								—	—
8Eh	PCON	—	—	—	—	—	—	$\overline{POR}$	$\overline{BOR}$ <sup>(4)</sup>	---- --qq	---- --uu
8Fh	—	Unimplemented								—	—
90h	—	Unimplemented								—	—
91h	—	Unimplemented								—	—
92h	PR2	Timer2 Period Register								1111 1111	1111 1111
93h	SSPADD	Synchronous Serial Port (I <sup>2</sup> C mode) Address Register								0000 0000	0000 0000
94h	SSPSTAT	—	—	D/ $\overline{A}$	P	S	R/ $\overline{W}$	UA	BF	--00 0000	--00 0000
95h-9Fh	—	Unimplemented								—	—

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

Shaded locations are unimplemented, read as '0'.

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

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

3: Other (non power-up) resets include external reset through  $\bar{M}CLR$  and the Watchdog Timer reset.

4: The  $\bar{B}OR$  bit is reserved on the PIC16C64, always maintain this bit set.

5: The IRP and RP1 bits are reserved on the PIC16C6X, always maintain these bits clear.

6: PIE1<6> and PIR1<6> are reserved on the PIC16C6X, always maintain these bits clear.

**TABLE 4-5: SPECIAL FUNCTION REGISTERS FOR THE PIC16C65/65A**

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR BOR	Value on all other resets <sup>(3)</sup>
Bank 0											
00h <sup>(1)</sup>	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	0000 0000
01h	TMR0	Timer0 module's register								xxxx xxxx	uuuu uuuu
02h <sup>(1)</sup>	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	0000 0000
03h <sup>(1)</sup>	STATUS	IRP <sup>(5)</sup>	RP1 <sup>(5)</sup>	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	000q quuu
04h <sup>(1)</sup>	FSR	Indirect data memory address pointer								xxxx xxxx	uuuu uuuu
05h	PORTA	—	—	PORTA Data Latch when written: PORTA pins when read						--xx xxxx	--uu uuuu
06h	PORTB	PORTB Data Latch when written: PORTB pins when read								xxxx xxxx	uuuu uuuu
07h	PORTC	PORTC Data Latch when written: PORTC pins when read								xxxx xxxx	uuuu uuuu
08h	PORTD	PORTD Data Latch when written: PORTD pins when read								xxxx xxxx	uuuu uuuu
09h	PORTE	—	—	—	—	—	RE2	RE1	RE0	---- -xxx	---- -uuu
0Ah <sup>(1,2)</sup>	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter					---0 0000	---0 0000
0Bh <sup>(1)</sup>	INTCON	GIE	PEIE	TOIE	INTE	RBIE	TOIF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	PSPIF	(6)	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
0Dh	PIR2	—	—	—	—	—	—	—	CCP2IF	---- -000	---- -000
0Eh	TMR1L	Holding register for the Least Significant Byte of the 16-bit TMR1 register								xxxx xxxx	uuuu uuuu
0Fh	TMR1H	Holding register for the Most Significant Byte of the 16-bit TMR1 register								xxxx xxxx	uuuu uuuu
10h	T1CON	—	—	T1CKPS1	T1CKPS0	T1OSCEN	$\overline{T1SYN\overline{C}}$	TMR1CS	TMR1ON	--00 0000	--uu uuuu
11h	TMR2	Timer2 module's register								0000 0000	0000 0000
12h	T2CON	—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	-000 0000	-000 0000
13h	SSPBUF	Synchronous Serial Port Receive Buffer/Transmit Register								xxxx xxxx	uuuu uuuu
14h	SSPCON	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	0000 0000
15h	CCPR1L	Capture/Compare/PWM1 (LSB)								xxxx xxxx	uuuu uuuu
16h	CCPR1H	Capture/Compare/PWM1 (MSB)								xxxx xxxx	uuuu uuuu
17h	CCP1CON	—	—	CCP1X	CCP1Y	CCP1M3	CCP1M2	CCP1M1	CCP1M0	--00 0000	--00 0000
18h	RCSTA	SPEN	RX9	SREN	CREN	—	FERR	OERR	RX9D	0000 -00x	0000 -00x
19h	TXREG	USART Transmit Data Register								0000 0000	0000 0000
1Ah	RCREG	USART Receive Data Register								0000 0000	0000 0000
1Bh	CCPR2L	Capture/Compare/PWM2 (LSB)								xxxx xxxx	uuuu uuuu
1Ch	CCPR2H	Capture/Compare/PWM2 (MSB)								xxxx xxxx	uuuu uuuu
1Dh	CCP2CON	—	—	CCP2X	CCP2Y	CCP2M3	CCP2M2	CCP2M1	CCP2M0	--00 0000	--00 0000
1Eh-1Fh	—	Unimplemented								—	—

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

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

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  $\overline{BOR}$  bit is reserved on the PIC16C65, always maintain this bit set.

5: The IRP and RP1 bits are reserved on the PIC16C6X, always maintain these bits clear.

6: PIE1<6> and PIR1<6> are reserved on the PIC16C6X, always maintain these bits clear.

# PIC16C6X

**TABLE 4-5: SPECIAL FUNCTION REGISTERS FOR THE PIC16C65/65A (Cont'd)**

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR BOR	Value on all other resets <sup>(3)</sup>
Bank 1											
80h <sup>(1)</sup>	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	0000 0000
81h	OPTION	RBP $\overline{U}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
82h <sup>(1)</sup>	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	0000 0000
83h <sup>(1)</sup>	STATUS	IRP <sup>(5)</sup>	RP1 <sup>(5)</sup>	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	000q quuu
84h <sup>(1)</sup>	FSR	Indirect data memory address pointer								xxxx xxxx	uuuu uuuu
85h	TRISA	—	—	PORTA Data Direction Register						--11 1111	--11 1111
86h	TRISB	PORTB Data Direction Register								1111 1111	1111 1111
87h	TRISC	PORTC Data Direction Register								1111 1111	1111 1111
88h	TRISD	PORTD Data Direction Register								1111 1111	1111 1111
89h	TRISE	IBF	OBF	IBOV	PSPMODE	—	TRISE2	TRISE1	TRISE0	0000 -111	0000 -111
8Ah <sup>(1,2)</sup>	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter					---0 0000	---0 0000
8Bh <sup>(1)</sup>	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
8Ch	PIE1	PSPIE	(6)	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
8Dh	PIE2	—	—	—	—	—	—	—	CCP2IE	---- -0	---- -0
8Eh	PCON	—	—	—	—	—	—	POR	$\overline{BOR}$ <sup>(4)</sup>	---- -qq	---- -uu
8Fh	—	Unimplemented								—	—
90h	—	Unimplemented								—	—
91h	—	Unimplemented								—	—
92h	PR2	Timer2 Period Register								1111 1111	1111 1111
93h	SSPADD	Synchronous Serial Port (I <sup>2</sup> C mode) Address Register								0000 0000	0000 0000
94h	SSPSTAT	—	—	D/ $\overline{A}$	P	S	R/ $\overline{W}$	UA	BF	--00 0000	--00 0000
95h	—	Unimplemented								—	—
96h	—	Unimplemented								—	—
97h	—	Unimplemented								—	—
98h	TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010
99h	SPBRG	Baud Rate Generator Register								0000 0000	0000 0000
9Ah	—	Unimplemented								—	—
9Bh	—	Unimplemented								—	—
9Ch	—	Unimplemented								—	—
9Dh	—	Unimplemented								—	—
9Eh	—	Unimplemented								—	—
9Fh	—	Unimplemented								—	—

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

Shaded locations are unimplemented, read as '0'.

Note 1: These registers can be addressed from 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  $\bar{B}OR$  bit is reserved on the PIC16C65, always maintain this bit set.

5: The IRP and RP1 bits are reserved on the PIC16C6X, always maintain these bits clear.

6: PIE1<6> and PIR1<6> are reserved on the PIC16C6X, always maintain these bits clear.

## 4.2.2.1 STATUS REGISTER

Applicable Devices									
61	62	62A	R62	63	64	64A	R64	65	65A

The STATUS register, shown in Figure 4-7, contains the arithmetic status of the ALU, the RESET status and the bank select bits for data memory.

The STATUS register can be the destination for any instruction, as with any other register. If the STATUS register is the destination for an instruction that affects the Z, DC or C bits, then the write to these three bits is disabled. These bits are set or cleared according to the device logic. Furthermore, the  $\overline{TO}$  and  $\overline{PD}$  bits are not writable. Therefore, the result of an instruction with the STATUS register as destination may be different than intended.

For example, `CLRF STATUS` will clear the upper-three bits and set the Z bit. This leaves the STATUS register as `000u u1uu` (where u = unchanged).

It is recommended, therefore, that only `BCF`, `BSF`, `SWAPF` and `MOVWF` instructions are used to alter the STATUS register because these instructions do not affect the Z, C or DC bits from the STATUS register. For other instructions, not affecting any status bits, see the "Instruction Set Summary."

**Note 1:** Bits IRP and RP1 (STATUS<7:6>) are not used by the PIC16C6X and should be maintained clear. Use of these bits as general purpose R/W bits is NOT recommended, since this may affect upward compatibility with future products.

**Note 2:** The C and DC bits operate as a borrow and digit borrow bit, respectively, in subtraction. See the `SUBLW` and `SUBWF` instructions for examples.

**FIGURE 4-7: STATUS REGISTER (ADDRESS 03h, 83h)**

R/W-0	R/W-0	R/W-0	R-1	R-1	R/W-x	R/W-x	R/W-x	
IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	
bit7								bit0
<p>bit 7: <b>IRP:</b> Register Bank Select bit (used for indirect addressing)  1 = Bank 2, 3 (100h - 1FFh)  0 = Bank 0, 1 (00h - FFh)  The IRP bit is not used by the PIC16C6X. The IRP bit is reserved, always maintain this bit clear.</p> <p>bit 6-5: <b>RP1:RP0:</b> Register Bank Select bits (used for direct addressing)  11 = Bank 3 (180h - 1FFh)  10 = Bank 2 (100h - 17Fh)  01 = Bank 1 (80h - FFh)  00 = Bank 0 (00h - 7Fh)  Each bank is 128 bytes.  Only bit RP0 is used by the PIC16C6X. The RP1 bit is reserved, always maintain this bit clear.</p> <p>bit 4: <b><math>\overline{TO}</math>:</b> Time-out bit  1 = After power-up, <code>CLRWDT</code> instruction, or <code>SLEEP</code> instruction  0 = A WDT time-out occurred</p> <p>bit 3: <b><math>\overline{PD}</math>:</b> Power-down bit  1 = After power-up or by the <code>CLRWDT</code> instruction  0 = By execution of the <code>SLEEP</code> instruction</p> <p>bit 2: <b>Z:</b> Zero bit  1 = The result of an arithmetic or logic operation is zero  0 = The result of an arithmetic or logic operation is not zero</p> <p>bit 1: <b>DC:</b> Digit carry/borrow bit (for <code>ADDWF</code>, <code>ADDLW</code>, <code>SUBLW</code>, and <code>SUBWF</code> instructions) (For borrow the polarity is reversed).  1 = A carry-out from the 4th low order bit of the result occurred  0 = No carry-out from the 4th low order bit of the result</p> <p>bit 0: <b>C:</b> Carry/borrow bit (for <code>ADDWF</code>, <code>ADDLW</code>, <code>SUBLW</code>, and <code>SUBWF</code> instructions) (For borrow the polarity is reversed).  1 = A carry-out from the most significant bit of the result occurred  0 = No carry-out from the most significant bit of the result</p> <p>Note: a subtraction is executed by adding the two's complement of the second operand.  For rotate (<code>RRF</code>, <code>RLF</code>) instructions, this bit is loaded with either the high or low order bit of the source register.</p>								
<p>R = Readable bit  W = Writable bit  - n = Value at POR reset  x = unknown</p>								

# PIC16C6X

## 4.2.2.2 OPTION REGISTER

Applicable Devices									
61	62	62A	R62	63	64	64A	R64	65	65A

The OPTION register is a readable and writable register which contains various control bits to configure the TMR0/WDT prescaler, the external INT interrupt, TMR0, and the weak pull-ups on PORTB.

**Note:** To achieve a 1:1 prescaler assignment for TMR0 register, assign the prescaler to the Watchdog Timer by setting the PSA (OPTION<3>) bit.

**FIGURE 4-8: OPTION REGISTER (ADDRESS 81h)**

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1																											
RBP <sub>U</sub>	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0																											
bit7				bit0																														
<div>R = Readable bit W = Writable bit U = Unimplemented bit, read as '0' - n = Value at POR reset</div>																																		
bit 7: <b>RBP<sub>U</sub></b> : PORTB Pull-up Enable bit 1 = PORTB pull-ups are disabled 0 = PORTB pull-ups are enabled by individual port latch values																																		
bit 6: <b>INTEDG</b> : Interrupt Edge Select bit 1 = Interrupt on rising edge of RB0/INT pin 0 = Interrupt on falling edge of RB0/INT pin																																		
bit 5: <b>T0CS</b> : TMR0 Clock Source Select bit 1 = Transition on RA4/T0CKI pin 0 = Internal instruction cycle clock (CLKOUT)																																		
bit 4: <b>T0SE</b> : TMR0 Source Edge Select bit 1 = Increment on high-to-low transition on RA4/T0CKI pin 0 = Increment on low-to-high transition on RA4/T0CKI pin																																		
bit 3: <b>PSA</b> : Prescaler Assignment bit 1 = Prescaler is assigned to the WDT 0 = Prescaler is assigned to the Timer0 module																																		
bit 2-0: <b>PS2:PS0</b> : Prescaler Rate Select bits																																		
<table><tr><th>Bit Value</th><th>TMR0 Rate</th><th>WDT Rate</th></tr><tr><td>000</td><td>1 : 2</td><td>1 : 1</td></tr><tr><td>001</td><td>1 : 4</td><td>1 : 2</td></tr><tr><td>010</td><td>1 : 8</td><td>1 : 4</td></tr><tr><td>011</td><td>1 : 16</td><td>1 : 8</td></tr><tr><td>100</td><td>1 : 32</td><td>1 : 16</td></tr><tr><td>101</td><td>1 : 64</td><td>1 : 32</td></tr><tr><td>110</td><td>1 : 128</td><td>1 : 64</td></tr><tr><td>111</td><td>1 : 256</td><td>1 : 128</td></tr></table>								Bit Value	TMR0 Rate	WDT Rate	000	1 : 2	1 : 1	001	1 : 4	1 : 2	010	1 : 8	1 : 4	011	1 : 16	1 : 8	100	1 : 32	1 : 16	101	1 : 64	1 : 32	110	1 : 128	1 : 64	111	1 : 256	1 : 128
Bit Value	TMR0 Rate	WDT Rate																																
000	1 : 2	1 : 1																																
001	1 : 4	1 : 2																																
010	1 : 8	1 : 4																																
011	1 : 16	1 : 8																																
100	1 : 32	1 : 16																																
101	1 : 64	1 : 32																																
110	1 : 128	1 : 64																																
111	1 : 256	1 : 128																																



## 4.2.2.3 INTCON REGISTER

Applicable Devices									
61	62	62A	R62	63	64	64A	R64	65	65A

The INTCON Register is a readable and writable register which contains the various enable and flag bits for the TMR0 register overflow, RB port change and external RB0/INT pin interrupts.

**Note:** Interrupt flag bits get set when an interrupt condition occurs regardless of the state of its corresponding enable bit or the global enable bit, GIE (INTCON<7>).

**FIGURE 4-9: INTCON REGISTER FOR PIC16C61 (ADDRESS 0Bh, 8Bh)**

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-x
GIE	—	TOIE	INTE	RBIE	TOIF	INTF	RBIF
bit7							bit0

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

bit 7: **GIE:** Global Interrupt Enable bit<sup>(1)</sup>  
1 = Enables all un-masked interrupts  
0 = Disables all interrupts

bit 6: **Unimplemented:** Read as '0'

bit 5: **TOIE:** TMR0 Overflow Interrupt Enable bit  
1 = Enables the TMR0 overflow interrupt  
0 = Disables the TMR0 overflow interrupt

bit 4: **INTE:** RB0/INT External Interrupt Enable bit  
1 = Enables the RB0/INT external interrupt  
0 = Disables the RB0/INT external interrupt

bit 3: **RBIE:** RB Port Change Interrupt Enable bit  
1 = Enables the RB port change interrupt  
0 = Disables the RB port change interrupt

bit 2: **TOIF:** TMR0 Overflow Interrupt Flag bit  
1 = TMR0 register overflowed (must be cleared in software)  
0 = TMR0 register did not overflow

bit 1: **INTF:** RB0/INT External Interrupt Flag bit  
1 = The RB0/INT external interrupt occurred (must be cleared in software)  
0 = The RB0/INT external interrupt did not occur

bit 0: **RBIF:** RB Port Change Interrupt Flag bit  
1 = At least one of the RB7:RB4 pins changed state (see Section 5.2 to clear interrupt)  
0 = None of the RB7:RB4 pins have changed state

Note 1: If an interrupt occurs while the GIE bit is being cleared, the GIE bit may unintentionally be re-enabled by the RETFIE instruction in the user's Interrupt Service Routine. Refer to Section 13.5 for a detailed description.

# PIC16C6X

**FIGURE 4-10: INTCON REGISTER FOR PIC16C62/62A/R62/63/64/64A/R64/65/65A  
(ADDRESS 0Bh, 8Bh)**

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-x
GIE	PEIE	TOIE	INTE	RBIE	TOIF	INTF	RBIF

bit7

bit0

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

bit 7: **GIE:** Global Interrupt Enable bit<sup>(1)</sup>  
1 = Enables all un-masked interrupts  
0 = Disables all interrupts

bit 6: **PEIE:** Peripheral Interrupt Enable bit  
1 = Enables all un-masked peripheral interrupts  
0 = Disables all peripheral interrupts

bit 5: **TOIE:** TMR0 Overflow Interrupt Enable bit  
1 = Enables the TMR0 overflow interrupt  
0 = Disables the TMR0 overflow interrupt

bit 4: **INTE:** RB0/INT External Interrupt Enable bit  
1 = Enables the RB0/INT external interrupt  
0 = Disables the RB0/INT external interrupt

bit 3: **RBIE:** RB Port Change Interrupt Enable bit  
1 = Enables the RB port change interrupt  
0 = Disables the RB port change interrupt

bit 2: **TOIF:** TMR0 Overflow Interrupt Flag bit  
1 = TMR0 register overflowed (must be cleared in software)  
0 = TMR0 register did not overflow

bit 1: **INTF:** RB0/INT External Interrupt Flag bit  
1 = The RB0/INT external interrupt occurred (must be cleared in software)  
0 = The RB0/INT external interrupt did not occur

bit 0: **RBIF:** RB Port Change Interrupt Flag bit  
1 = At least one of the RB7:RB4 pins changed state (see Section 5.2 to clear the interrupt)  
0 = None of the RB7:RB4 pins have changed state

Note 1: For the PIC16C62/64/65 only, if an interrupt occurs while the GIE bit is being cleared, the GIE bit may unintentionally be re-enabled by the RETFIE instruction in the user's Interrupt Service Routine. Refer to Section 13.5 for a detailed description.

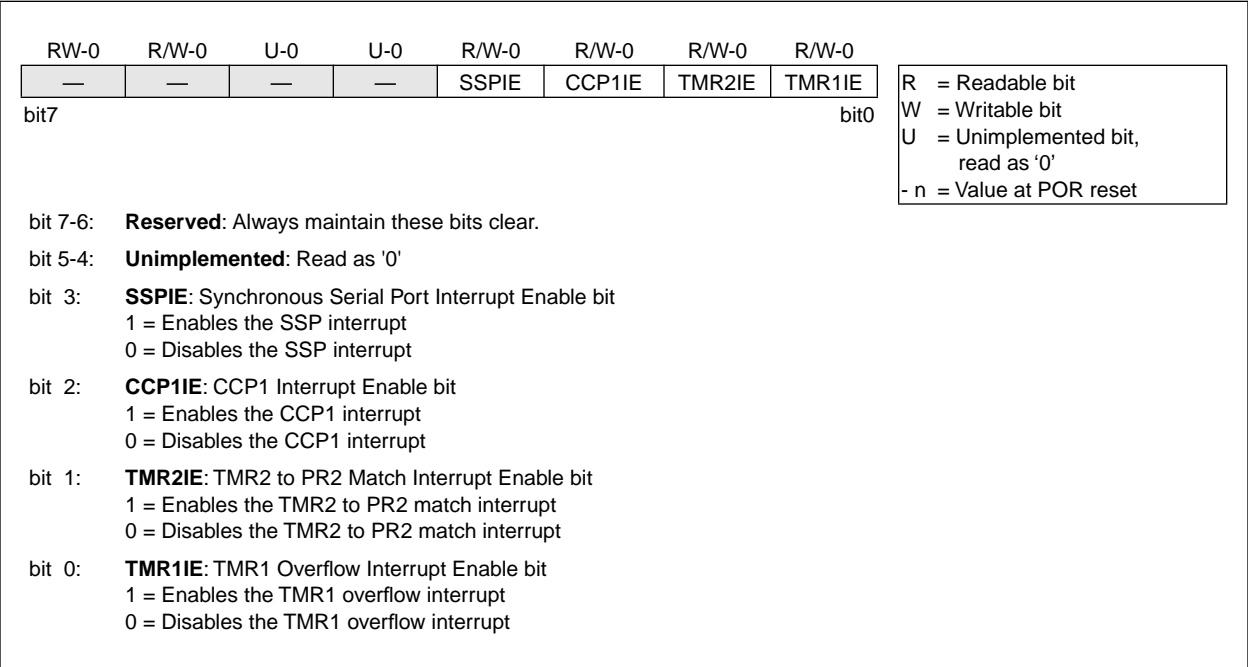
4.2.2.4    PIE1 REGISTER

Applicable Devices									
61	62	62A	R62	63	64	64A	R64	65	65A

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

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

FIGURE 4-11: PIE1 REGISTER FOR PIC16C62/62A/R62 (ADDRESS 8Ch)



# PIC16C6X

**FIGURE 4-12: PIE1 REGISTER FOR PIC16C63 (ADDRESS 8Ch)**

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE
bit7							bit0

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

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

bit 5: **RCIE:** USART Receive Interrupt Enable bit  
1 = Enables the USART receive interrupt  
0 = Disables the USART receive interrupt

bit 4: **TXIE:** USART Transmit Interrupt Enable bit  
1 = Enables the USART transmit interrupt  
0 = Disables the USART transmit interrupt

bit 3: **SSPIE:** Synchronous Serial Port Interrupt Enable bit  
1 = Enables the SSP interrupt  
0 = Disables the SSP interrupt

bit 2: **CCP1IE:** CCP1 Interrupt Enable bit  
1 = Enables the CCP1 interrupt  
0 = Disables the CCP1 interrupt

bit 1: **TMR2IE:** TMR2 to PR2 Match Interrupt Enable bit  
1 = Enables the TMR2 to PR2 match interrupt  
0 = Disables the TMR2 to PR2 match interrupt

bit 0: **TMR1IE:** TMR1 Overflow Interrupt Enable bit  
1 = Enables the TMR1 overflow interrupt  
0 = Disables the TMR1 overflow interrupt

**FIGURE 4-13: PIE1 REGISTER FOR PIC16C64/64A/R64 (ADDRESS 8Ch)**

R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
PSPIE	—	—	—	SSPIE	CCP1IE	TMR2IE	TMR1IE
bit7							bit0

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

bit 7: **PSPIE:** Parallel Slave Port Read/Write Interrupt Enable bit  
1 = Enables the PSP read/write interrupt  
0 = Disables the PSP read/write interrupt

bit 6: **Reserved:** Always maintain this bit clear.

bit 5-4: **Unimplemented:** Read as '0'

bit 3: **SSPIE:** Synchronous Serial Port Interrupt Enable bit  
1 = Enables the SSP interrupt  
0 = Disables the SSP interrupt

bit 2: **CCP1IE:** CCP1 Interrupt Enable bit  
1 = Enables the CCP1 interrupt  
0 = Disables the CCP1 interrupt

bit 1: **TMR2IE:** TMR2 to PR2 Match Interrupt Enable bit  
1 = Enables the TMR2 to PR2 match interrupt  
0 = Disables the TMR2 to PR2 match interrupt

bit 0: **TMR1IE:** TMR1 Overflow Interrupt Enable bit  
1 = Enables the TMR1 overflow interrupt  
0 = Disables the TMR1 overflow interrupt

**FIGURE 4-14: PIE1 REGISTER FOR PIC16C65/65A (ADDRESS 8Ch)**

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PSPIE	—	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE
bit7							bit0

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

bit 7: **PSPIE:** Parallel Slave Port Read/Write Interrupt Enable bit  
1 = Enables the PSP read/write interrupt  
0 = Disables the PSP read/write interrupt

bit 6: **Reserved:** Always maintain this bit clear.

bit 5: **RCIE:** USART Receive Interrupt Enable bit  
1 = Enables the USART receive interrupt  
0 = Disables the USART receive interrupt

bit 4: **TXIE:** USART Transmit Interrupt Enable bit  
1 = Enables the USART transmit interrupt  
0 = Disables the USART transmit interrupt

bit 3: **SSPIE:** Synchronous Serial Port Interrupt Enable bit  
1 = Enables the SSP interrupt  
0 = Disables the SSP interrupt

bit 2: **CCP1IE:** CCP1 Interrupt Enable bit  
1 = Enables the CCP1 interrupt  
0 = Disables the CCP1 interrupt

bit 1: **TMR2IE:** TMR2 to PR2 Match Interrupt Enable bit  
1 = Enables the TMR2 to PR2 match interrupt  
0 = Disables the TMR2 to PR2 match interrupt

bit 0: **TMR1IE:** TMR1 Overflow Interrupt Enable bit  
1 = Enables the TMR1 overflow interrupt  
0 = Disables the TMR1 overflow interrupt

# PIC16C6X

## 4.2.2.5 PIR1 REGISTER

Applicable Devices									
61	62	62A	R62	63	64	64A	R64	65	65A

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

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

**FIGURE 4-15: PIR1 REGISTER FOR PIC16C62/62A/R62 (ADDRESS 0Ch)**

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

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

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

bit 5-4: **Unimplemented:** Read as '0'

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

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

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

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

**FIGURE 4-16: PIR1 REGISTER FOR PIC16C63 (ADDRESS 0Ch)**

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

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

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

bit 5: **RCIF:** USART Receive Interrupt Flag bit  
1 = The USART receive buffer is full (must be cleared in software)  
0 = The USART receive buffer is empty

bit 4: **TXIF:** USART Transmit Interrupt Flag bit  
1 = The USART transmit buffer is full (must be cleared in software)  
0 = The USART transmit buffer is empty

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

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

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

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

# PIC16C6X

**FIGURE 4-17: PIR1 REGISTER FOR PIC16C64/64A/R64 (ADDRESS 0Ch)**

R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
PSPIF	—	—	—	SSPIF	CCP1IF	TMR2IF	TMR1IF
bit7				bit0			

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

bit 7: **PSPIF**: Parallel Slave Port Interrupt Flag bit  
1 = A read or a write operation has taken place (Must be cleared in software)  
0 = No read or write operation has taken place

bit 6: **Reserved**: Always maintain this bit clear.

bit 5-4: **Unimplemented**: Read as '0'

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

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

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

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



**FIGURE 4-18: PIR1 REGISTER FOR PIC16C65/65A (ADDRESS 0Ch)**

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

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

bit 7: **PSPIF:** Parallel Slave Port Interrupt Flag bit  
1 = A read or a write operation has taken place (Must be cleared in software)  
0 = No read or write operation has taken place

bit 6: **Reserved:** Always maintain this bit clear.

bit 5: **RCIF:** USART Receive Interrupt Flag bit  
1 = The USART receive buffer is full (must be cleared in software)  
0 = The USART receive buffer is empty

bit 4: **TXIF:** USART Transmit Interrupt Flag bit  
1 = The USART transmit buffer is full (must be cleared in software)  
0 = The USART transmit buffer is empty

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

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

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

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

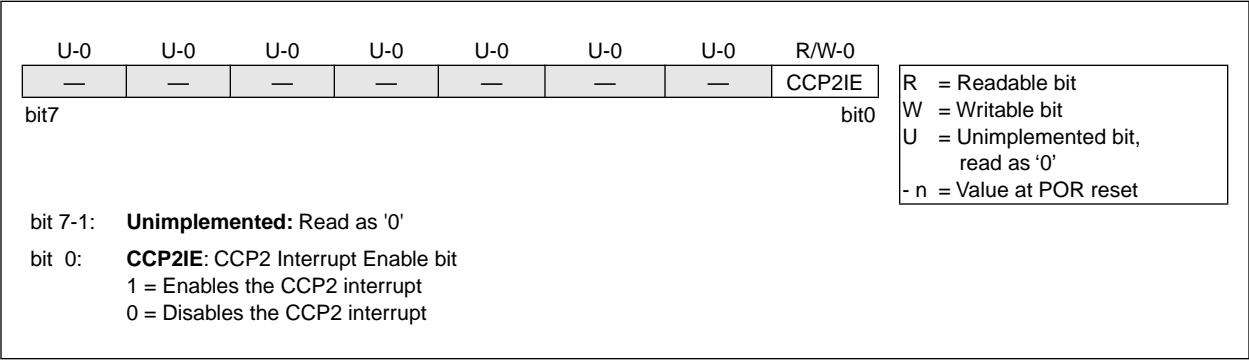
# PIC16C6X

## 4.2.2.6 PIE2 REGISTER

Applicable Devices									
61	62	62A	R62	63	64	64A	R64	65	65A

This register contains the CCP2 interrupt enable bit.

FIGURE 4-19: PIE2 REGISTER (ADDRESS 8Dh)



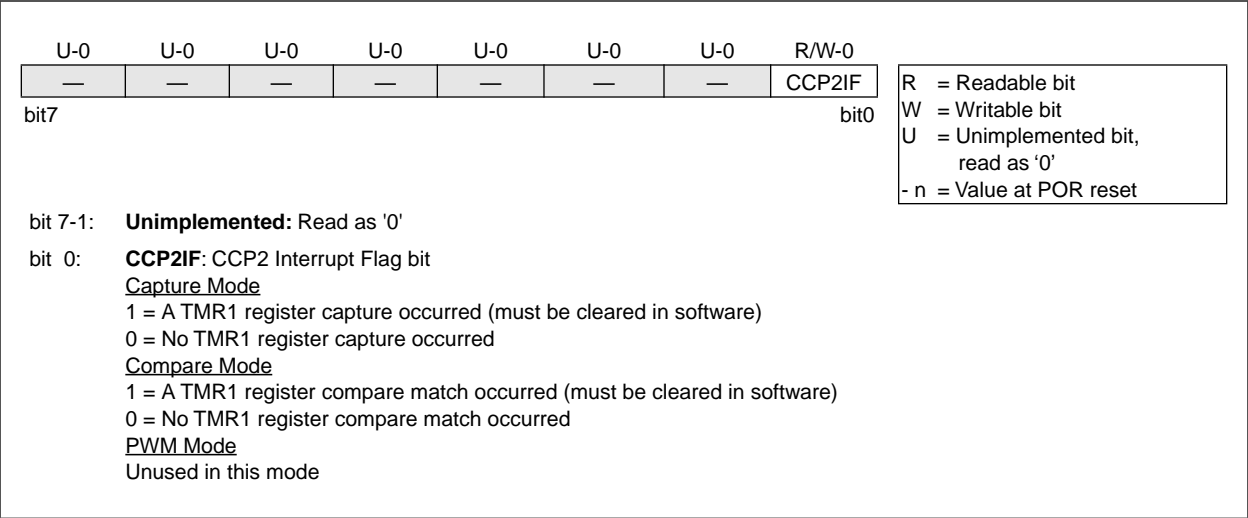
4.2.2.7 PIR2 REGISTER

Applicable Devices									
61	62	62A	R62	63	64	64A	R64	65	65A

This register contains the CCP2 interrupt flag bit.

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

FIGURE 4-20: PIR2 REGISTER (ADDRESS 0Dh)



# PIC16C6X

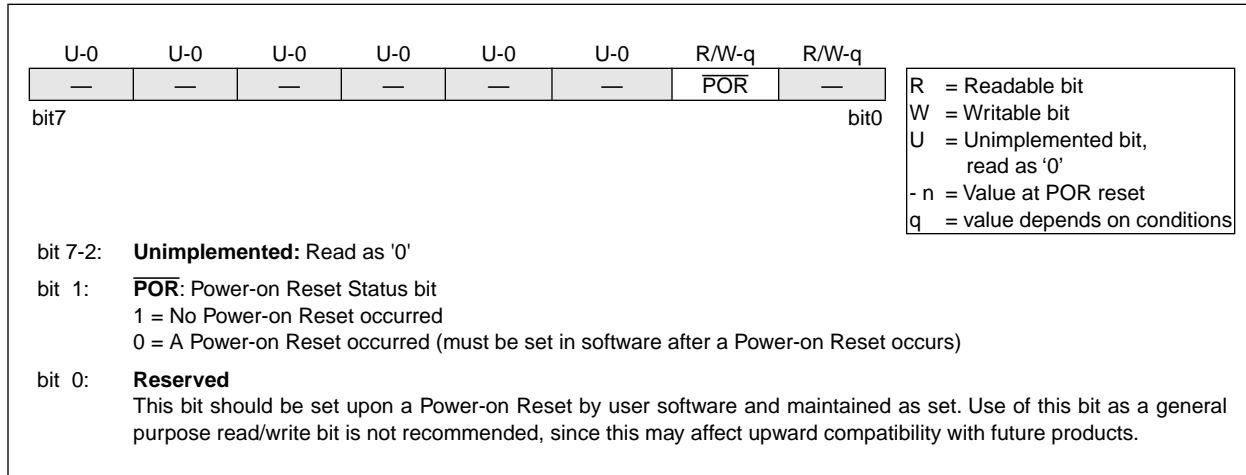
## 4.2.2.8 PCON REGISTER

Applicable Devices									
61	62	62A	R62	63	64	64A	R64	65	65A

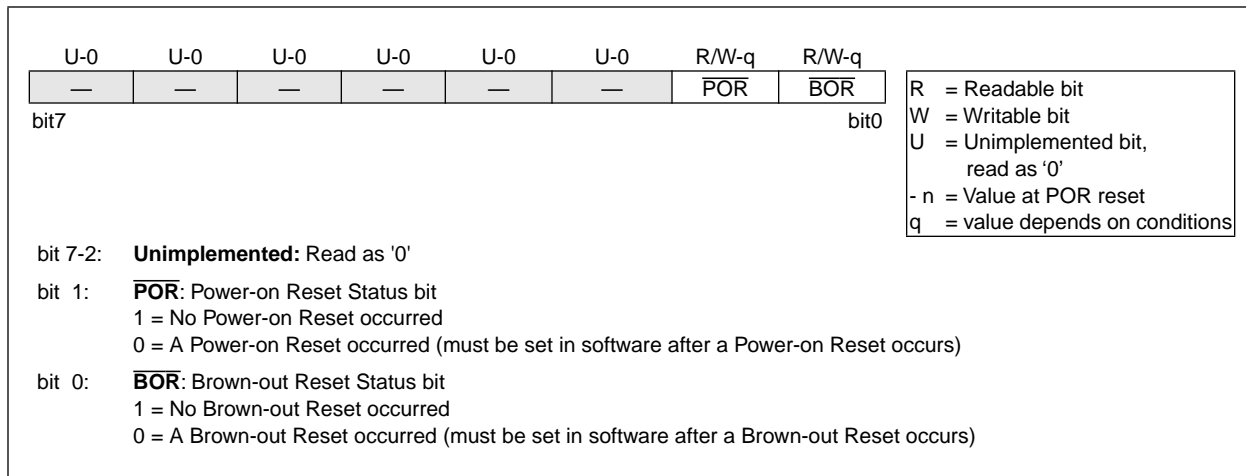
The Power Control register (PCON) contains a flag bit to allow differentiation between a Power-on Reset to an external  $\overline{\text{MCLR}}$  reset or WDT reset. The PIC16C62A/R62/63/64A/R64/65A contains an additional bit to differentiate a Brown-out Reset condition from a Power-on Reset condition.

**Note:**  $\overline{\text{BOR}}$  is unknown on Power-on Reset. It must then be set by the user and checked on subsequent resets to see if  $\overline{\text{BOR}}$  is clear, indicating a brown-out has occurred. The BOR status bit is a “don't care” and is not necessarily predictable if the brown-out circuit is disabled (by clearing the BODEN bit in the Configuration word).

**FIGURE 4-21: PCON REGISTER FOR PIC16C62/64/65 (ADDRESS 8Eh)**



**FIGURE 4-22: PCON REGISTER FOR PIC16C62A/R62/63/64A/R64/65A (ADDRESS 8Eh)**



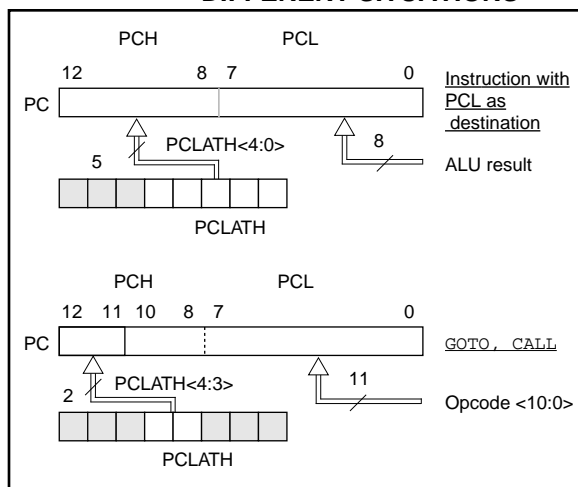
## 4.3 PCL and PCLATH

### Applicable Devices

61	62	62A	R62	63	64	64A	R64	65	65A
----	----	-----	-----	----	----	-----	-----	----	-----

The program counter (PC) is 13-bits wide. The low byte comes from the PCL register, which is a readable and writable register. The high byte (PC<12:8>) not directly readable or writable and comes from PCLATH. On any reset, the high byte of the PC is cleared. Figure 4-23 shows the two situations for the loading of the PC. The upper example in the figure shows how the PC is loaded on a write to PCL (PCLATH<4:0> → PCH). The lower example in the figure shows how the PC is loaded during a CALL or GOTO instruction (PCLATH<4:3> → PCH).

**FIGURE 4-23: LOADING OF PC IN DIFFERENT SITUATIONS**



### 4.3.1 COMPUTED GOTO

A computed GOTO is accomplished by adding an offset to the program counter (ADDWF PCL). When doing a table read using a computed GOTO method, care should be exercised if the table location crosses a PCL memory boundary (each 256 byte block). Refer to the application note "Implementing a Table Read" (AN556).

### 4.3.2 STACK

The PIC16CXX family has an 8 deep x 13-bit wide hardware stack. The stack space is not part of either program or data space and the stack pointer is not readable or writable. The PC is PUSHed onto the stack when a CALL instruction is executed or an interrupt causes a branch. The stack is POPed in the event of a RETURN, RETLW or a RETFIE instruction execution. PCLATH is not affected by a PUSH or a POP operation.

The stack operates as a circular buffer. This means that after the stack has been PUSHed eight times, the ninth push overwrites the value that was stored from the first push. The tenth push overwrites the second push (and so on).

**Note 1:** There are no status bits to indicate stack overflows or stack underflow conditions.

**Note 2:** There are no instructions mnemonics called PUSH or POP. These are actions that occur from the execution of the CALL, RETURN, RETLW, and RETFIE instructions, or the vectoring to an interrupt address

## 4.4 Program Memory Paging

### Applicable Devices

61	62	62A	R62	63	64	64A	R64	65	65A
----	----	-----	-----	----	----	-----	-----	----	-----

The PIC16C63/65/65A devices have 4K of program memory, but the CALL and GOTO instructions only have an 11-bit address range. This 11-bit address range allows a branch within a 2K program memory page size. To allow CALL and GOTO instructions to address the entire 4K program memory address range, there must be another bit to specify the program memory page. This paging bit comes from the PCLATH<3> bit (Figure 4-23). When doing a CALL or GOTO instruction, the user must ensure that this page select bit (PCLATH<3>) is programmed so that the desired program memory page is addressed. If a return from a CALL instruction (or interrupt) is executed, the entire 13-bit PC is pushed onto the stack. Therefore, manipulation of the PCLATH<3> bit is not required for the return instructions (which POPs the address from the stack).

**Note:** The PIC16C6X ignores the PCLATH<4> bit, which is used for program memory pages 2 and 3 (1000h - 1FFFh). The use of PCLATH<4> as a general purpose read/write bit is not recommended since this may affect upward compatibility with future products.

Devices with 2K or less program memory ignore the PCLATH<3> bit, which is used for program memory page 1 (0800h-0FFFh). The use of PCLATH<3> as a general purpose read/write bit for these devices is not recommended since this may affect upward compatibility with future products.

# PIC16C6X

Example 4-1 shows the calling of a subroutine in page 1 of the program memory. This example assumes that the PCLATH is saved and restored by the interrupt service routine (if interrupts are used).

### EXAMPLE 4-1: CALL OF A SUBROUTINE IN PAGE 1 FROM PAGE 0

```
ORG 0x500
BSF    PCLATH,3    ;Select page 1 (800h-FFFh)
CALL   SUB1_P1     ;Call subroutine in
:           ;page 1 (800h-FFFh)
:
:
ORG 0x900
SUB1_P1:           ;called subroutine
:           ;page 1 (800h-FFFh)
:
RETURN          ;return to Call subroutine
:           ;in page 0 (000h-7FFh)
```

### 4.5 Indirect Addressing, INDF and FSR Registers

Applicable Devices									
61	62	62A	R62	63	64	64A	R64	65	65A

The INDF register is not a physical register. Addressing the INDF register will cause indirect addressing.

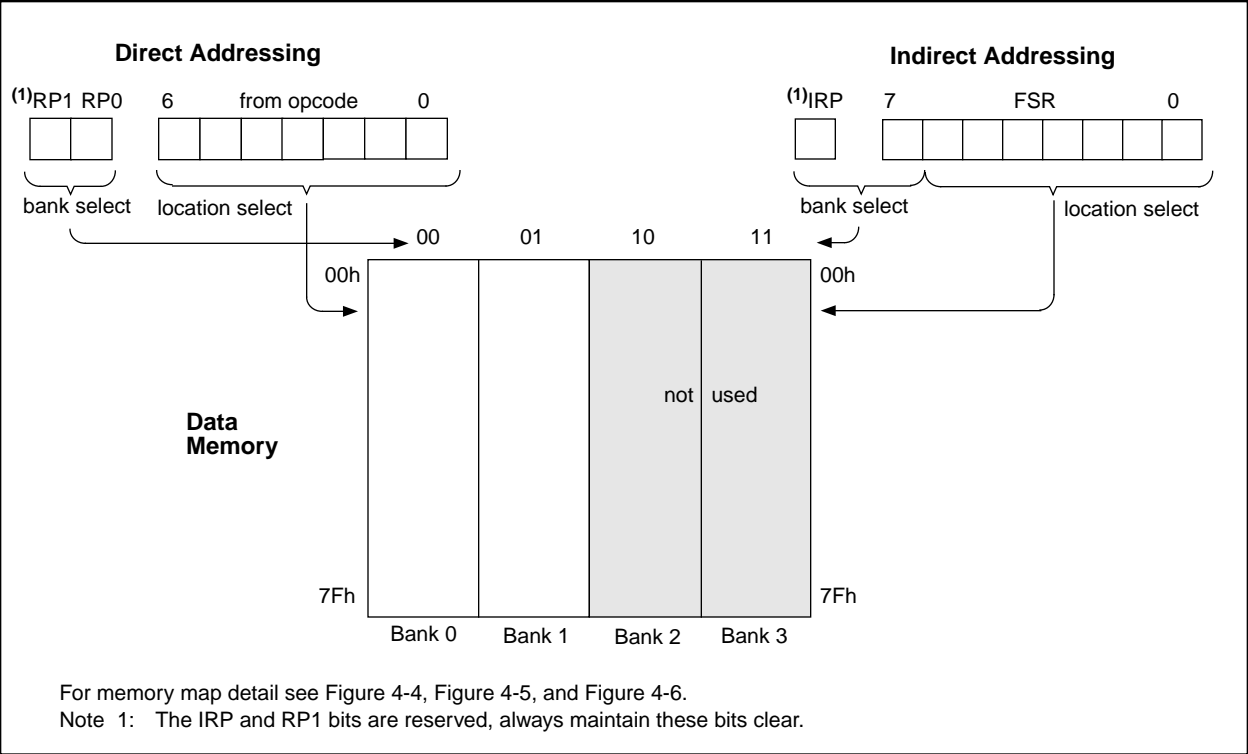
Indirect addressing is possible by using the INDF register. Any instruction using the INDF register actually accesses the register pointed to by the File Select Register, FSR. Reading the INDF register itself indirectly (FSR = '0') will produce 00h. Writing to the INDF register indirectly results in a no-operation (although status bits may be affected). An effective 9-bit address is obtained by concatenating the 8-bit FSR register and the IRP bit (STATUS<7>), as shown in Figure 4-24. However, IRP is not used in the PIC16C6X.

A simple program to clear RAM location 20h-2Fh using indirect addressing is shown in Example 4-2.

### EXAMPLE 4-2: INDIRECT ADDRESSING

```
        movlw 0x20    ;initialize pointer
        movwf FSR     ; to RAM
NEXT     clrf  INDF    ;clear INDF register
        incf  FSR,F   ;inc pointer
        btfss FSR,4   ;all done?
        goto  NEXT    ;NO, clear next
CONTINUE
:           ;YES, continue
```

FIGURE 4-24: DIRECT/INDIRECT ADDRESSING



## 5.0 I/O PORTS

Applicable Devices									
61	62	62A	R62	63	64	64A	R64	65	65A

Some pins for these I/O ports are multiplexed with an alternate function for the peripheral features on the device. In general, when a peripheral is enabled, that pin may not be used as a general purpose I/O pin.

### 5.1 PORTA and TRISA Register

Applicable Devices									
61	62	62A	R62	63	64	64A	R64	65	65A

All devices have a 6-bit wide PORTA, except for the PIC16C61 which has a 5-bit wide PORTA.

Pin RA4/T0CKI is a Schmitt Trigger input and an open drain output. All other RA port pins have TTL input levels and full CMOS output drivers. All pins have data direction bits (TRIS registers) which can configure these pins as output or input.

Setting a bit in the TRISA register puts the corresponding output driver in a hi-impedance mode. Clearing a bit in the TRISA register puts the contents of the output latch on the selected pin.

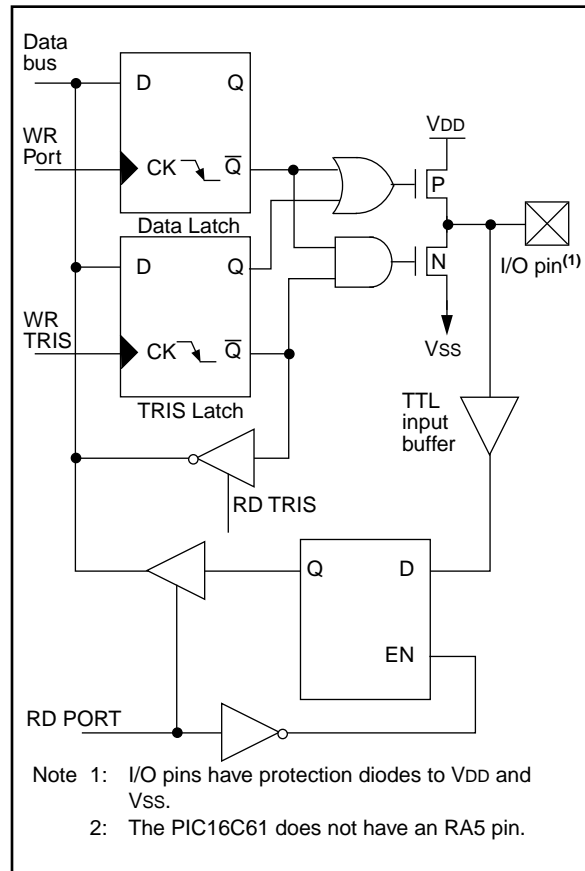
Reading PORTA register reads the status of the pins whereas writing to it will write to the port latch. All write operations are read-modify-write operations. Therefore, a write to a port implies that the port pins are read, this value is modified, and then written to the port data latch.

Pin RA4 is multiplexed with Timer0 module clock input to become the RA4/T0CKI pin.

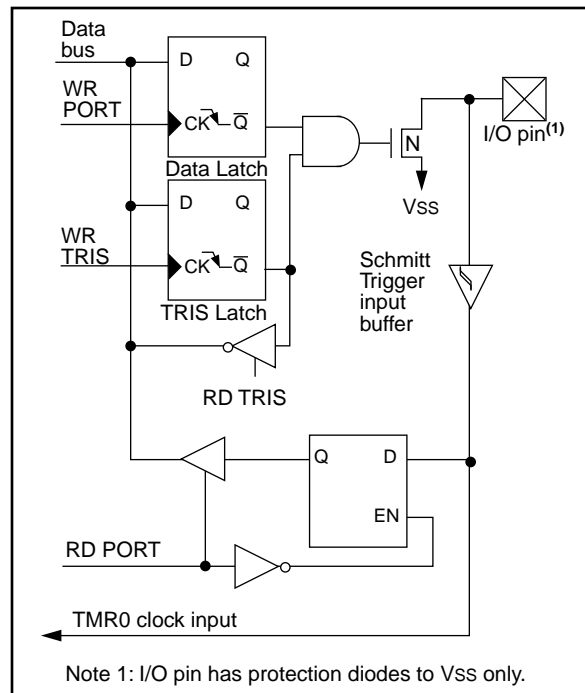
#### EXAMPLE 5-1: INITIALIZING PORTA

```
CLRF    PORTA        ; Initialize PORTA by
                    ; setting output
                    ; data latches
BSF     STATUS, RP0   ; Select Bank 1
MOVLW   0xCF          ; Value used to
                    ; initialize data
                    ; direction
MOVWF   TRISA         ; Set RA<3:0> as inputs
                    ; RA<5:4> as outputs
                    ; TRISA<7:6> are always
                    ; read as '0'.
```

**FIGURE 5-1: BLOCK DIAGRAM OF THE RA3:RA0 PINS AND THE RA5 PIN**



**FIGURE 5-2: BLOCK DIAGRAM OF THE RA4/T0CKI PIN**



# PIC16C6X

**TABLE 5-1: PORTA FUNCTIONS**

Name	Bit#	Buffer Type	Function
RA0	bit0	TTL	Input/output
RA1	bit1	TTL	Input/output
RA2	bit2	TTL	Input/output
RA3	bit3	TTL	Input/output
RA4/T0CKI	bit4	ST	Input/output or external clock input for Timer0. Output is open drain type.
RA5/ $\overline{SS}$ (1)	bit5	TTL	Input/output, slave select input for synchronous serial port.

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

Note 1: The PIC16C61 does not have PORTA<5> or TRISA<5>, read as '0'.

**TABLE 5-2: SUMMARY OF REGISTERS ASSOCIATED WITH PORTA**

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
05h	PORTA	—	—	RA5 <sup>(1)</sup>	RA4	RA3	RA2	RA1	RA0	--xx xxxx	--uu uuuu
85h	TRISA	—	—	TRISA5 <sup>(1)</sup>	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	--11 1111	--11 1111

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

Note 1: PORTA<5> and TRISA<5> are not implemented on the PIC16C61, read as '0'.



## 5.2 PORTB and TRISB Register

Applicable Devices									
61	62	62A	R62	63	64	64A	R64	65	65A

PORTB is an 8-bit wide bi-directional port. The corresponding data direction register is TRISB. Setting a bit in the TRISB register puts the corresponding output driver in a hi-impedance mode. Clearing a bit in the TRISB register puts the contents of the output latch on the selected pin(s).

### EXAMPLE 5-2: INITIALIZING PORTB

```

CLRF   PORTB           ; Initialize PORTB by
                        ; setting output
                        ; data latches
BSF     STATUS, RP0     ; Select Bank 1
MOVLW   0xCF           ; Value used to
                        ; initialize data
                        ; direction
MOVWF   TRISB          ; Set RB<3:0> as inputs
                        ; RB<5:4> as outputs
                        ; RB<7:6> as inputs
    
```

Each of the PORTB pins has a weak internal pull-up. A single control bit can turn on all the pull-ups. This is done by clearing bit  $\overline{\text{RBP}}\text{U}$  (OPTION<7>). The weak pull-up is automatically turned off when the port pin is configured as an output. The pull-ups are also disabled on a Power-on Reset.

Four of PORTB's pins, RB7:RB4, have an interrupt on change feature. Only pins configured as inputs can cause this interrupt to occur (i.e., any RB7:RB4 pin configured as an output is excluded from the interrupt on change comparison). The input pins (of RB7:RB4) are compared with the old value latched on the last read of PORTB. The "mismatch" outputs of RB7:RB4 are OR'ed together to generate the RB port change interrupt with flag bit RBIF (INTCON<0>).

This interrupt can wake the device from SLEEP. The user, in the interrupt service routine, can clear the interrupt in the following manner:

- Any read or write of PORTB. This will end the mismatch condition.
- Clear flag bit RBIF.

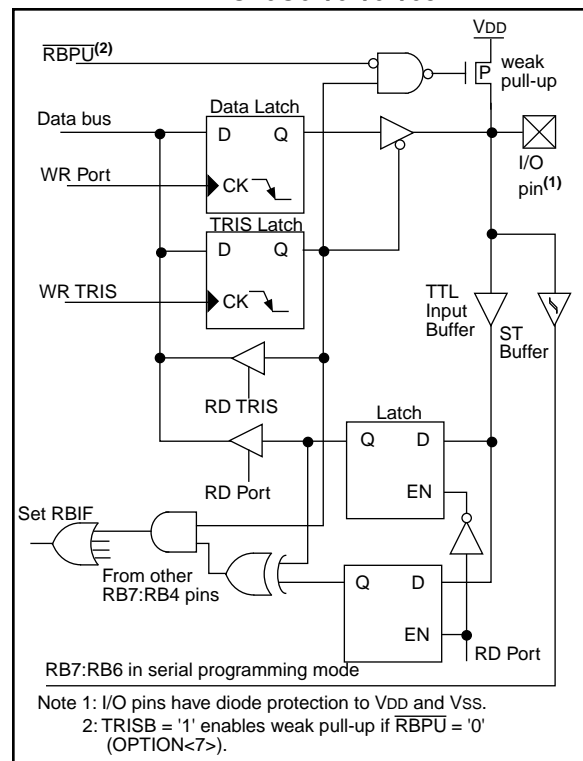
A mismatch condition will continue to set flag bit RBIF. Reading PORTB will end the mismatch condition, and allow flag bit RBIF to be cleared.

This interrupt on mismatch feature, together with software configurable pull-ups on these four pins allow easy interface to a key pad and make it possible for wake-up on key-depression. Refer to the Embedded Control Handbook, Application Note, "Implementing Wake-up on Key Stroke" (AN552).

**Note:** For PIC16C61/62/64/65 only, if a change on the I/O pin should occur when a read operation is being executed (start of the Q2 cycle), then interrupt flag bit RBIF may not get set.

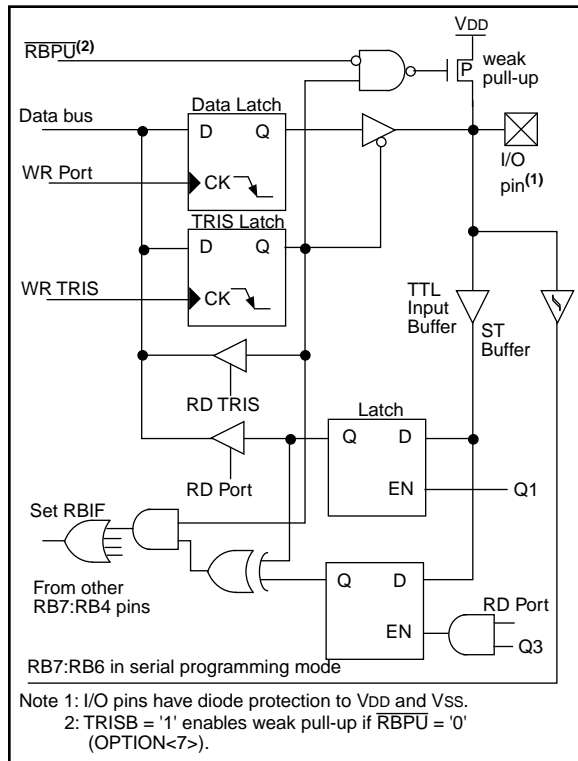
The interrupt on change feature is recommended for wake-up on key depression operation and operations where PORTB is only used for the interrupt on change feature. Polling of PORTB is not recommended while using the interrupt on change feature.

**FIGURE 5-3: BLOCK DIAGRAM OF THE RB7:RB4 PINS FOR PIC16C61/62/64/65**

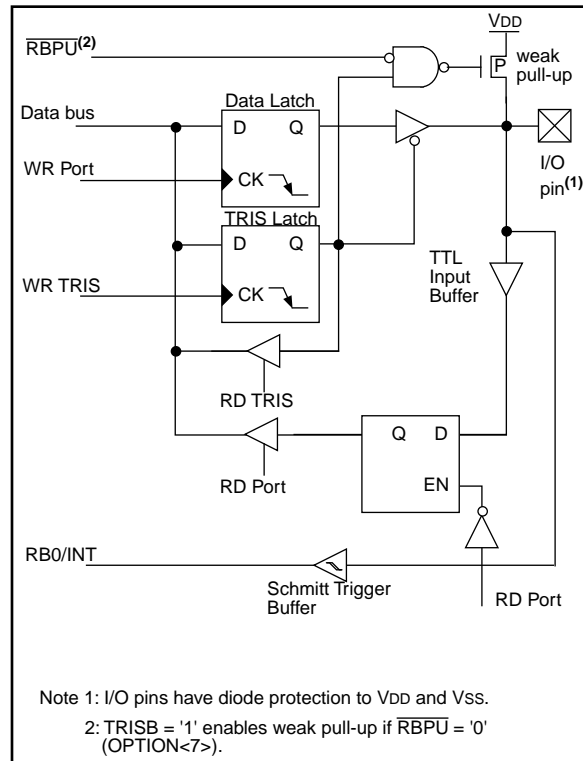


# PIC16C6X

**FIGURE 5-4: BLOCK DIAGRAM OF THE RB7:RB4 PINS FOR PIC16C62A/63/64A/65A**



**FIGURE 5-5: BLOCK DIAGRAM OF THE RB3:RB0 PINS**



**TABLE 5-3: PORTB FUNCTIONS**

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

Legend: TTL = TTL input, ST = Schmitt Trigger

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

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

**TABLE 5-4: SUMMARY OF REGISTERS ASSOCIATED WITH PORTB**

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR BOR	Value on all other resets
06h	PORTB	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	xxxx xxxx	uuuu uuuuu
86h	TRISB	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	1111 1111	1111 1111
81h	OPTION	RBPƯ	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111

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

5.3 PORTC and TRISC Register

Applicable Devices									
61	62	62A	R62	63	64	64A	R64	65	65A

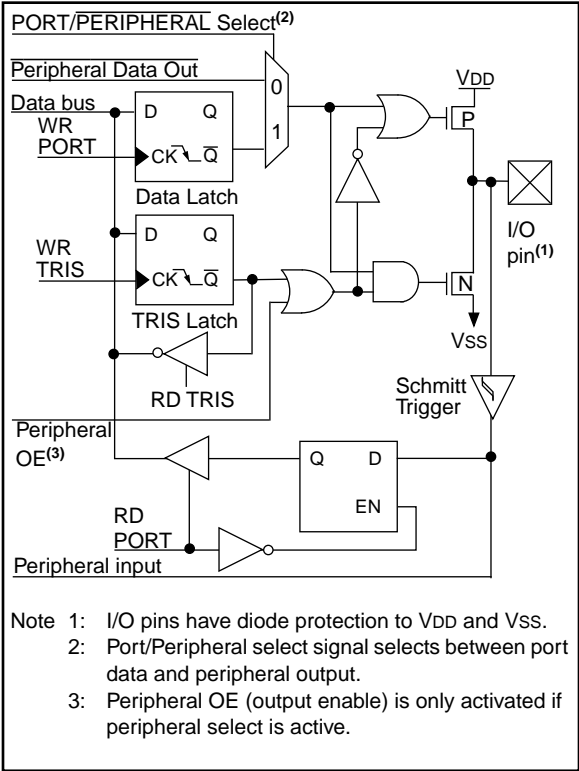
PORTC is an 8-bit bi-directional port. Each pin is individually configurable as an input or output through the TRISC register. PORTC is multiplexed with several peripheral functions (Table 5-5). PORTC pins have Schmitt Trigger input buffers.

When enabling peripheral functions, care should be taken in defining TRIS bits for each PORTC pin. Some peripherals override the TRIS bit to make a pin an output, while other peripherals override the TRIS bit to make a pin an input. Since the TRIS bit override is in effect while the peripheral is enabled, read-modify-write instructions (BSF, BCF, XORWF) with TRISC as destination should be avoided. The user should refer to the corresponding peripheral section for the correct TRIS bit settings.

EXAMPLE 5-3: INITIALIZING PORTC

```
CLRF    PORTC           ; Initialize PORTC by
                        ; setting output
                        ; data latches
BSF     STATUS, RP0      ; Select Bank 1
MOVLW   0xCF             ; Value used to
                        ; initialize data
                        ; direction
MOVWF   TRISC            ; Set RC<3:0> as inputs
                        ; RC<5:4> as outputs
                        ; RC<7:6> as inputs
```

FIGURE 5-6: PORTC BLOCK DIAGRAM (PERIPHERAL OUTPUT OVERRIDE)



- Note 1: I/O pins have diode protection to VDD and VSS.  
Note 2: Port/Peripheral select signal selects between port data and peripheral output.  
Note 3: Peripheral OE (output enable) is only activated if peripheral select is active.

TABLE 5-5: PORTC FUNCTIONS FOR PIC16C62/64

Name	Bit#	Buffer Type	Function
RC0/T1OSI/T1CKI	bit0	ST	Input/output port pin or Timer1 oscillator input or Timer1 clock input
RC1/T1OSO	bit1	ST	Input/output port pin or Timer1 oscillator output
RC2/CCP1	bit2	ST	Input/output port pin or Capture1 input/Compare1 output/PWM1 output
RC3/SCK/SCL	bit3	ST	RC3/SCK/SCL can also be selected as the synchronous serial clock for both SPI and I <sup>2</sup> C modes.
RC4/SDI/SDA	bit4	ST	RC4/SDI/SDA can also be selected as the SPI Data In (SPI mode) or data I/O (I <sup>2</sup> C mode).
RC5/SDO	bit5	ST	Input/output port pin or synchronous serial port data output
RC6	bit6	ST	Input/output port pin
RC7	bit7	ST	Input/output port pin

Legend: ST = Schmitt Trigger Input

# PIC16C6X

**TABLE 5-6: PORTC FUNCTIONS FOR PIC16C62A/R62/64A/R64**

Name	Bit#	Buffer Type	Function
RC0/T1OSO/T1CKI	bit0	ST	Input/output port pin or Timer1 oscillator output or Timer1 clock input
RC1/T1OSI	bit1	ST	Input/output port pin or Timer1 oscillator input
RC2/CCP1	bit2	ST	Input/output port pin or Capture input/Compare output/PWM1 output
RC3/SCK/SCL	bit3	ST	RC3/SCK/SCL can also be selected as the synchronous serial clock for both SPI and I <sup>2</sup> C modes.
RC4/SDI/SDA	bit4	ST	RC4/SDI/SDA can also be selected as the SPI Data In (SPI mode) or data I/O (I <sup>2</sup> C mode).
RC5/SDO	bit5	ST	Input/output port pin or synchronous serial port data output
RC6	bit6	ST	Input/output port pin
RC7	bit7	ST	Input/output port pin

Legend: ST = Schmitt Trigger Input

**TABLE 5-7: PORTC FUNCTIONS FOR PIC16C63/65/65A**

Name	Bit#	Buffer Type	Function
RC0/T1OSO/T1CKI	bit0	ST	Input/output port pin or Timer1 oscillator output or Timer1 clock input
RC1/T1OSI/CCP2	bit1	ST	Input/output port pin, Timer1 oscillator input, Capture2 input/Compare2 output/PWM2 output
RC2/CCP1	bit2	ST	Input/output port pin or Capture1 input/Compare1 output/PWM1 output
RC3/SCK/SCL	bit3	ST	RC3/SCK/SCL can also be selected as the synchronous serial clock for both SPI and I <sup>2</sup> C modes.
RC4/SDI/SDA	bit4	ST	RC4/SDI/SDA can also be selected as the SPI Data In (SPI mode) or data I/O (I <sup>2</sup> C mode).
RC5/SDO	bit5	ST	Input/output port pin or synchronous serial port data output
RC6/TX/CK	bit6	ST	Input/output port pin, USART Asynchronous Transmit, or USART Synchronous Clock
RC7/RX/DT	bit7	ST	Input/output port pin, USART Asynchronous Receive, or USART Synchronous Data

Legend: ST = Schmitt Trigger Input

**TABLE 5-8: SUMMARY OF REGISTERS ASSOCIATED WITH PORTC**

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
07h	PORTC	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0	xxxx xxxx	uuuu uuuu
87h	TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	1111 1111	1111 1111

Legend: x = unknown, u = unchanged.

## 5.4 PORTD and TRISD Register

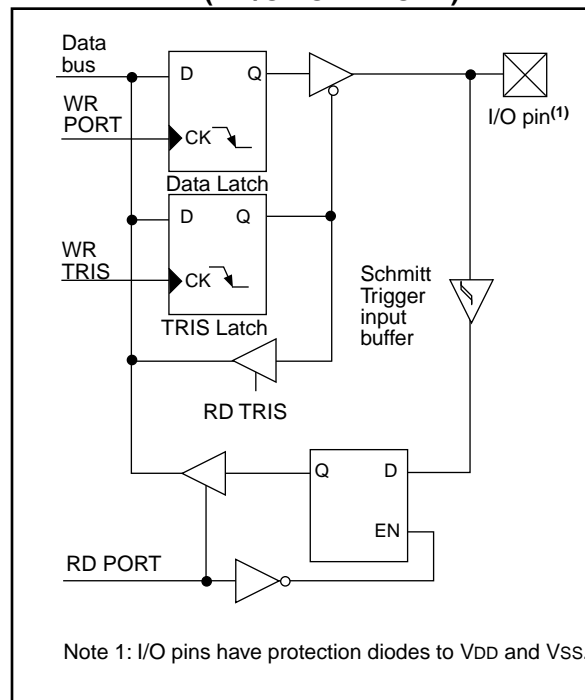
### Applicable Devices

61	62	62A	R62	63	64	64A	R64	65	65A
----	----	-----	-----	----	----	-----	-----	----	-----

PORTD is an 8-bit port with Schmitt Trigger input buffers. Each pin is individually configurable as input or output.

PORTD can be configured as an 8-bit wide microprocessor port (or parallel slave port) by setting control bit PSPMODE (TRISE<4>). In this mode, the input buffers are TTL.

**FIGURE 5-7: PORTD BLOCK DIAGRAM (IN I/O PORT MODE)**



**TABLE 5-9: PORTD FUNCTIONS**

Name	Bit#	Buffer Type	Function
RD0/PSP0	bit0	ST/TTL <sup>(1)</sup>	Input/output port pin or parallel slave port bit0
RD1/PSP1	bit1	ST/TTL <sup>(1)</sup>	Input/output port pin or parallel slave port bit1
RD2/PSP2	bit2	ST/TTL <sup>(1)</sup>	Input/output port pin or parallel slave port bit2
RD3/PSP3	bit3	ST/TTL <sup>(1)</sup>	Input/output port pin or parallel slave port bit3
RD4/PSP4	bit4	ST/TTL <sup>(1)</sup>	Input/output port pin or parallel slave port bit4
RD5/PSP5	bit5	ST/TTL <sup>(1)</sup>	Input/output port pin or parallel slave port bit5
RD6/PSP6	bit6	ST/TTL <sup>(1)</sup>	Input/output port pin or parallel slave port bit6
RD7/PSP7	bit7	ST/TTL <sup>(1)</sup>	Input/output port pin or parallel slave port bit7

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

Note 1: Buffer is a Schmitt Trigger when in I/O mode, and a TTL buffer when in Parallel Slave Port mode.

**TABLE 5-10: SUMMARY OF REGISTERS ASSOCIATED WITH PORTD**

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
08h	PORTD	RD7	RD6	RD5	RD4	RD3	RD2	RD1	RD0	xxxx xxxx	uuuu uuuu
88h	TRISD	TRISD7	TRISD6	TRISD5	TRISD4	TRISD3	TRISD2	TRISD1	TRISD0	1111 1111	1111 1111
89h	TRISE	IBF	OBF	IBOV	PSPMODE	—	TRISE2	TRISE1	TRISE0	0000 -111	0000 -111

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

# PIC16C6X

## 5.5 PORTE and TRISE Register

Applicable Devices									
61	62	62A	R62	63	64	64A	R64	65	65A

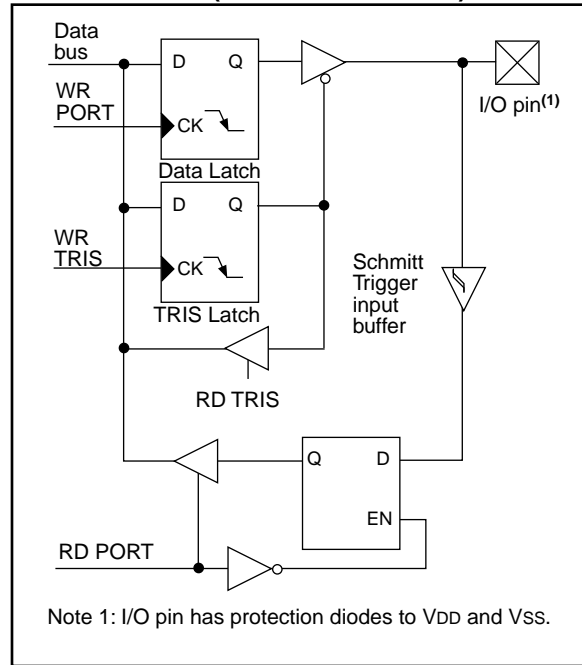
PORTE has three pins, RE2/ $\overline{CS}$ , RE1/ $\overline{WR}$ , and RE0/ $\overline{RD}$  which are individually configurable as inputs or outputs. These pins have Schmitt Trigger input buffers.

I/O PORTE becomes control inputs for the microprocessor port when bit PSPMODE (TRISE<4>) is set. In this mode, the user must make sure that the TRISE<2:0> bits are set (pins are configured as digital inputs). In this mode the input buffers are TTL.

Figure 5-9 shows the TRISE register, which also controls the parallel slave port operation.

TRISE controls the direction of the RE pins.

**FIGURE 5-8: PORTE BLOCK DIAGRAM (IN I/O PORT MODE)**



**FIGURE 5-9: TRISE REGISTER (ADDRESS 89h)**

R-0	R-0	R/W-0	R/W-0	U-0	R/W-1	R/W-1	R/W-1
IBF	OBF	IBOV	PSPMODE	—	TRISE2	TRISE1	TRISE0
bit7							bit0

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

bit 7 : **IBF**: Input Buffer Full Status bit  
1 = A word has been received and is waiting to be read by the CPU  
0 = No word has been received

bit 6 : **OBF**: Output Buffer Full Status bit  
1 = The output buffer still holds a previously written word  
0 = The output buffer has been read

bit 5 : **IBOV**: Input Buffer Overflow Detect bit (in microprocessor mode)  
1 = A write occurred when a previously input word has not been read (must be cleared in software)  
0 = No overflow occurred

bit 4 : **PSPMODE**: Parallel Slave Port Mode Select bit  
1 = Parallel slave port mode  
0 = General purpose I/O mode

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

bit 2 : **TRISE2**: Direction Control bit for pin RE2/ $\overline{CS}$   
1 = Input  
0 = Output

bit 1 : **TRISE1**: Direction Control bit for pin RE1/ $\overline{WR}$   
1 = Input  
0 = Output

bit 0 : **TRISE0**: Direction Control bit for pin RE0/ $\overline{RD}$   
1 = Input  
0 = Output

**TABLE 5-11: PORTE FUNCTIONS**

Name	Bit#	Buffer Type	Function
RE0/ $\overline{RD}$	bit0	ST/TTL <sup>(1)</sup>	Input/output port pin, Read control input in parallel slave port mode. $\overline{RD}$ 1 = Not a read operation 0 = Read operation. The system reads the PORTD register (if chip selected)
RE1/ $\overline{WR}$	bit1	ST/TTL <sup>(1)</sup>	Input/output port pin, Write control input in parallel slave port mode. $\overline{WR}$ 1 = Not a write operation 0 = Write operation. The system writes to the PORTD register (if chip selected)
RE2/ $\overline{CS}$	bit2	ST/TTL <sup>(1)</sup>	Input/output port pin, Chip select control input in parallel slave port mode. $\overline{CS}$ 1 = Device is not selected 0 = Device is selected

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

Note 1: Buffer is a Schmitt Trigger when in I/O mode, and a TTL buffer when in Parallel Slave Port (PSP) mode.

**TABLE 5-12: SUMMARY OF REGISTERS ASSOCIATED WITH PORTE**

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
09h	PORTE	—	—	—	—	—	RE2	RE1	RE0	---- -xxx	---- -uuu
89h	TRISE	IBF	OBF	IBOV	PSPMODE	—	TRISE2	TRISE1	TRISE0	0000 -111	0000 -111

Legend: x = unknown, u = unchanged, - = unimplemented locations read as '0'. Shaded cells not used by PORTE.

## 5.6 I/O Programming Considerations

Applicable Devices									
61	62	62A	R62	63	64	64A	R64	65	65A

### 5.6.1 BI-DIRECTIONAL I/O PORTS

Any instruction which writes, operates internally as a read followed by a write operation. The BCF and BSF instructions, for example, read the register into the CPU, execute the bit operation and write the result back to the register. Caution must be used when these instructions are applied to a port with both inputs and outputs defined. For example, a BSF operation on bit5 of PORTB will cause all eight bits of PORTB to be read into the CPU. Then the BSF operation takes place on bit5 and PORTB is written to the output latches. If another bit of PORTB is used as a bi-directional I/O pin (e.g., bit0) and it is defined as an input at this time, the input signal present on the pin itself would be read into the CPU and rewritten to the data latch of this particular pin, overwriting the previous content. As long as the pin stays in the input mode, no problem occurs. However, if bit0 is switched into output mode later on, the content of the data latch may now be unknown.

Reading the port register, reads the values of the port pins. Writing to the port register writes the value to the port latch. When using read-modify-write instructions (ex. BCF, BSF, etc.) on a port, the value of the port pins is read, the desired operation is done to this value, and this value is then written to the port latch.

Example 5-4 shows the effect of two sequential read-modify-write instructions on an I/O port.

### EXAMPLE 5-4: READ-MODIFY-WRITE INSTRUCTIONS ON AN I/O PORT

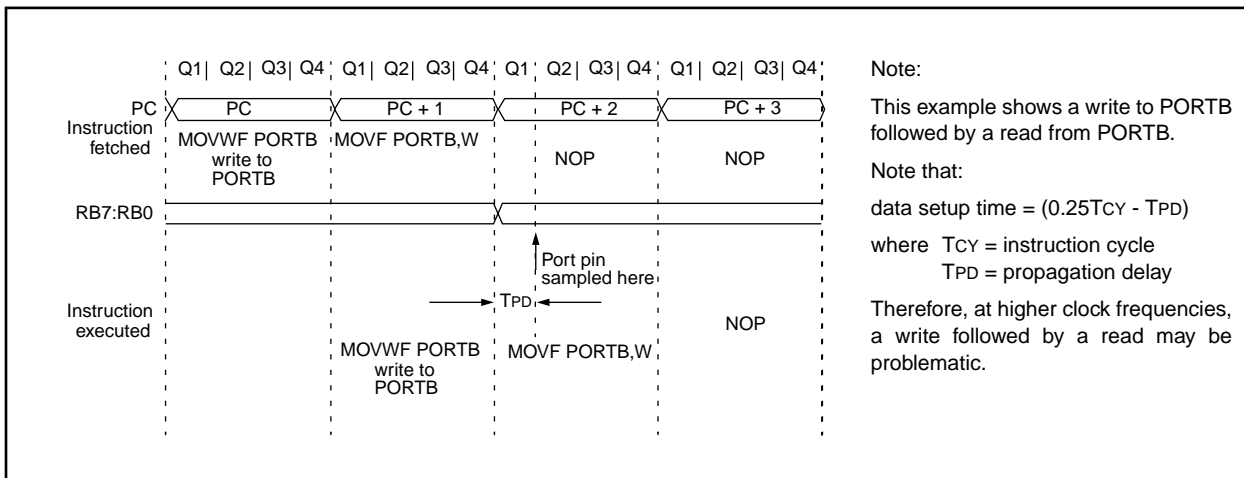
```
;Initial PORT settings: PORTB<7:4> Inputs
;                        PORTB<3:0> Outputs
;PORTB<7:6> have external pull-ups and are
;not connected to other circuitry
;
;
;                        PORT latch  PORT pins
;                        -----
BCF PORTB, 7      ; 01pp pppp  11pp pppp
BCF PORTB, 6      ; 10pp pppp  11pp pppp
BSF STATUS, RP0   ;
BCF TRISB, 7      ; 10pp pppp  11pp pppp
BCF TRISB, 6      ; 10pp pppp  10pp pppp
;
;Note that the user may have expected the
;pin values to be 00pp pppp. The 2nd BCF
;caused RB7 to be latched as the pin value
;(high).
```

A pin actively outputting a Low or High should not be driven from external devices at the same time in order to change the level on this pin ("wired-or", "wired-and"). The resulting high output currents may damage the chip.

### 5.6.2 SUCCESSIVE OPERATIONS ON I/O PORTS

The actual write to an I/O port happens at the end of an instruction cycle, whereas for reading, the data must be valid at the beginning of the instruction cycle (Figure 5-10). Therefore, care must be exercised if a write followed by a read operation is carried out on the same I/O port. The sequence of instructions should be such to allow the pin voltage to stabilize (load dependent) before the next instruction which causes that file to be read into the CPU is executed. Otherwise, the previous state of that pin may be read into the CPU rather than the new state. When in doubt, it is better to separate these instructions with a NOP or another instruction not accessing this I/O port.

FIGURE 5-10: SUCCESSIVE I/O OPERATION





## 5.7 Parallel Slave Port

### Applicable Devices

61	62	62A	R62	63	64	64A	R64	65	65A
----	----	-----	-----	----	----	-----	-----	----	-----

PORTD operates as an 8-bit wide parallel slave port, or microprocessor port when control bit PSMODE (TRISE<4>) is set. In slave mode it is asynchronously readable and writable by the external world through RD control input (RE0/RD) and WR control input pin (RE1/WR).

It can directly interface to an 8-bit microprocessor data bus. The external microprocessor can read or write the PORTD latch as an 8-bit latch. Setting PSMODE enables port pin RE0/RD to be the RD input, RE1/WR to be the WR input and RE2/CS to be the CS (chip select) input. For this functionality, the corresponding data direction bits of the TRISE register (TRISE<2:0>) must be configured as inputs (set).

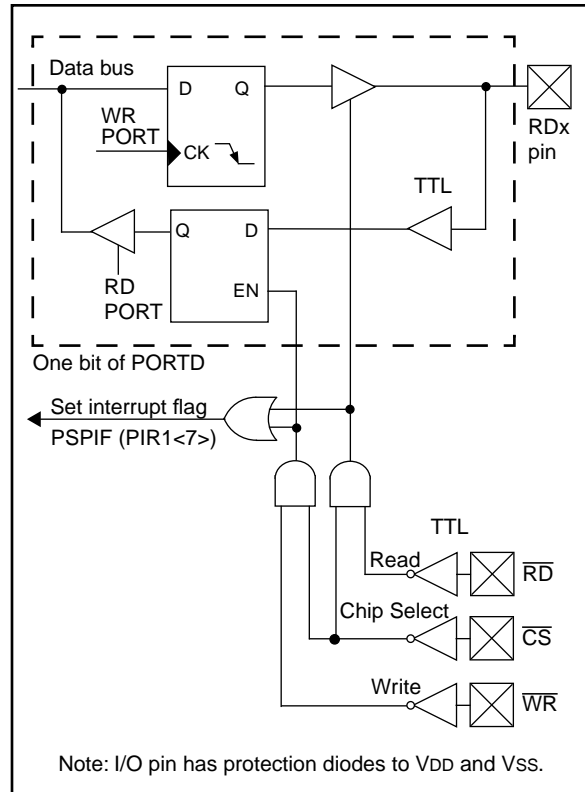
There are actually two 8-bit latches, one for data-out (from the PIC16/17) and one for data input. The user writes 8-bit data to PORTD data latch and reads data from the port pin latch (note that they have the same address). In this mode, the TRISD register is ignored since the microprocessor is controlling the direction of data flow.

Input Buffer Full Status Flag bit IBF (TRISE<7>) is set if a received word is waiting to be read by the CPU. Once the PORTD input latch is read, bit IBF is cleared. IBF is a read only status bit. Output Buffer Full Status Flag bit OBF (TRISE<6>) is set if a word written to PORTD latch is waiting to be read by the external bus. Once the PORTD output latch is read by the microprocessor, bit OBF is cleared. Input Buffer Overflow Status flag bit IBOV (TRISE<5>) is set if a second write to the microprocessor port is attempted when the previous word has not been read by the CPU (the first word is retained in the buffer).

When not in Parallel Slave Port mode, bits IBF (TRISE<7>) and OBF (TRISE<6>) are held clear. However if flag bit IBOV (TRISE<5>) was previously set, it must be cleared in software.

An interrupt is generated and latched into flag bit PSPIF (PIR1<7>) when a read or a write operation is completed. Flag bit PSPIF must be cleared by user software and the interrupt can be disabled by clearing the interrupt enable bit PSPIE (PIE1<7>).

**FIGURE 5-11: PORTD AND PORTE AS A PARALLEL SLAVE PORT**



**TABLE 5-13: REGISTERS ASSOCIATED WITH PARALLEL SLAVE PORT**

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
08h	PORTD	PSP7	PSP6	PSP5	PSP4	PSP3	PSP2	PSP1	PSP0	xxxx xxxx	uuuu uuuu
09h	PORTE	—	—	—	—	—	RE2	RE1	RE0	---- -xxx	---- -uuu
89h	TRISE	IBF	OBF	IBOV	PSMODE	—	TRISE2	TRISE1	TRISE0	0000 -111	0000 -111
0Ch	PIR1	PSPIF	(1)	RCIF(2)	TXIF(2)	SSPIF	CCP1IF	TMR2IF	TRM1IF	0000 0000	0000 0000
8Ch	PIE1	PSPIE	(1)	RCIE(2)	TXIE(2)	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000

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

Note 1: These bits are reserved, always maintain these bits clear.

Note 2: These bits are implemented on the PIC16C63/65/65A only.

# PIC16C6X

---

NOTES:

## 6.0 OVERVIEW OF TIMER MODULES

Applicable Devices									
61	62	62A	R62	63	64	64A	R64	65	65A

All PIC16C6X devices have three timer modules except for the PIC16C61, which has one timer module. Each module can generate an interrupt to indicate that an event has occurred (i.e., timer overflow). Each of these modules are detailed in the following sections. The timer modules are:

- Timer0 module (Section 7.0)
- Timer1 module (Section 8.0)
- Timer2 module (Section 9.0)

### 6.1 Timer0 Overview

Applicable Devices									
61	62	62A	R62	63	64	64A	R64	65	65A

The Timer0 module (previously known as RTCC) is a simple 8-bit overflow counter. The clock source can be either the internal system clock ( $F_{osc}/4$ ) or an external clock. When the clock source is an external clock, the Timer0 module can be selected to increment on either the rising or falling edge.

The Timer0 module also has a programmable prescaler option. This prescaler can be assigned to either the Timer0 module or the Watchdog Timer. Bit PSA ( $OPTION<3>$ ) assigns the prescaler, and bits PS2:PS0 ( $OPTION<2:0>$ ) determine the prescaler value. TMR0 can increment at the following rates: 1:1 when the prescaler is assigned to Watchdog Timer, 1:2, 1:4, 1:8, 1:16, 1:32, 1:64, 1:128, and 1:256.

Synchronization of the external clock occurs after the prescaler. When the prescaler is used, the external clock frequency may be higher than the device's frequency. The maximum frequency is 50 MHz, given the high and low time requirements of the clock.

### 6.2 Timer1 Overview

Applicable Devices									
61	62	62A	R62	63	64	64A	R64	65	65A

Timer1 is a 16-bit timer/counter. The clock source can be either the internal system clock ( $F_{osc}/4$ ), an external clock, or an external crystal. Timer1 can operate as either a timer or a counter. When operating as a counter (external clock source), the counter can either operate synchronized to the device or asynchronously to the device. Asynchronous operation allows Timer1 to operate during sleep, which is useful for applications that require a real-time clock as well as the power savings of SLEEP mode.

Timer1 also has a prescaler option which allows TMR1 to increment at the following rates: 1:1, 1:2, 1:4, and 1:8. TMR1 can be used in conjunction with the Capture/Compare/PWM module. When used with a CCP module, Timer1 is the time-base for 16-bit capture or 16-bit compare and must be synchronized to the device.

### 6.3 Timer2 Overview

Applicable Devices									
61	62	62A	R62	63	64	64A	R64	65	65A

Timer2 is an 8-bit timer with a programmable prescaler and a programmable postscaler, as well as an 8-bit Period Register (PR2). Timer2 can be used with the CCP1 module (in PWM mode) as well as the Baud Rate Generator for the Synchronous Serial Port (SSP). The prescaler option allows Timer2 to increment at the following rates: 1:1, 1:4, and 1:16.

The postscaler allows TMR2 register to match the period register (PR2) a programmable number of times before generating an interrupt. The postscaler can be programmed from 1:1 to 1:16 (inclusive).

### 6.4 CCP Overview

Applicable Devices									
61	62	62A	R62	63	64	64A	R64	65	65A

The CCP module(s) can operate in one of three modes: 16-bit capture, 16-bit compare, or up to 10-bit Pulse Width Modulation (PWM).

Capture mode captures the 16-bit value of TMR1 into the CCPRxH:CCPRxL register pair. The capture event can be programmed for either the falling edge, rising edge, fourth rising edge, or sixteenth rising edge of the CCPx pin.

Compare mode compares the TMR1H:TMR1L register pair to the CCPRxH:CCPRxL register pair. When a match occurs, an interrupt can be generated and the output pin CCP1 can be forced to a given state (High or Low) and Timer1 can be reset (CCP1). This depends on control bits CCPxM3:CCPxM0.

PWM mode compares the TMR2 register to a 10-bit duty cycle register value as well as to an 8-bit Period Register (PR2). When TMR2 register = PR2 register, the TMR2 register is cleared to 00h, an interrupt may be generated, and the CCPx pin (if an output) will be forced high. When the TMR2 register = Duty Cycle register, the CCPx pin will be forced low.

# PIC16C6X

---

NOTES:

## 7.0 TIMER0 MODULE

Applicable Devices									
61	62	62A	R62	63	64	64A	R64	65	65A

The Timer0 module has the following features:

- 8-bit timer/counter register, TMR0
  - Read and write capability
  - Interrupt on overflow from FFh to 00h
- 8-bit software programmable prescaler
- Internal or external clock select
  - Edge select for external clock

Figure 7-1 is a simplified block diagram of the Timer0 module.

Timer mode is selected by clearing bit T0CS (OPTION<5>). In timer mode, the Timer0 module will increment every instruction cycle (without prescaler). If TMR0 register is written, the increment is inhibited for the following two instruction cycles (Figure 7-2 and Figure 7-3). The user can work around this by writing an adjusted value to the TMR0 register.

Counter mode is selected by setting bit T0CS. In this mode, Timer0 will increment either on every rising or falling edge of pin RA4/T0CKI. The incrementing edge is determined by the source edge select bit T0SE

(OPTION<4>). Clearing bit T0SE selects the rising edge. Restrictions on the external clock input are discussed in detail in Section 7.2.

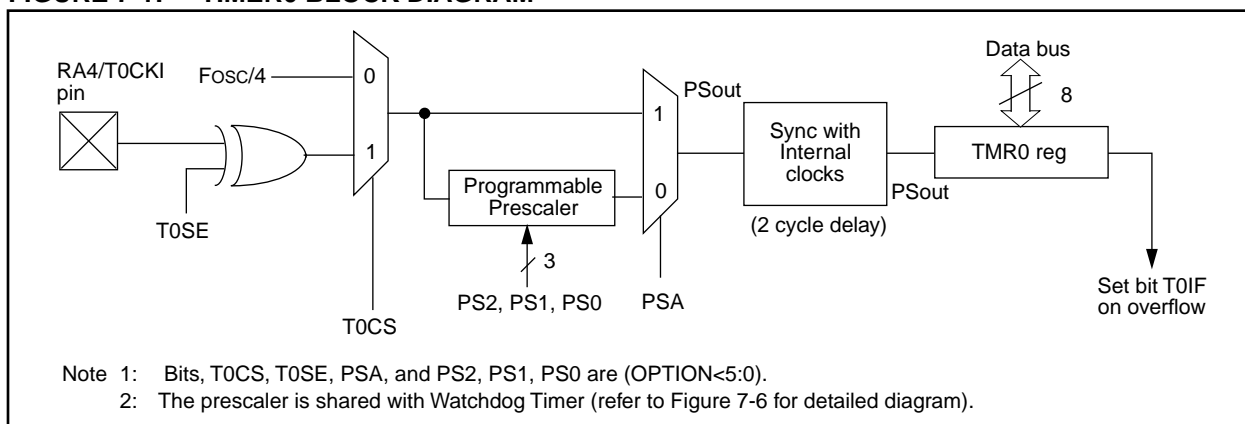
The prescaler is mutually exclusively shared between the Timer0 module and the Watchdog Timer. The prescaler assignment is controlled in software by control bit PSA (OPTION<3>). Clearing bit PSA will assign the prescaler to the Timer0 module. The prescaler is not readable or writable. When the prescaler is assigned to the Timer0 module, prescale values of 1:2, 1:4, ..., 1:256 are selectable. Section 7.3 details the operation of the prescaler.

### 7.1 Timer0 Interrupt

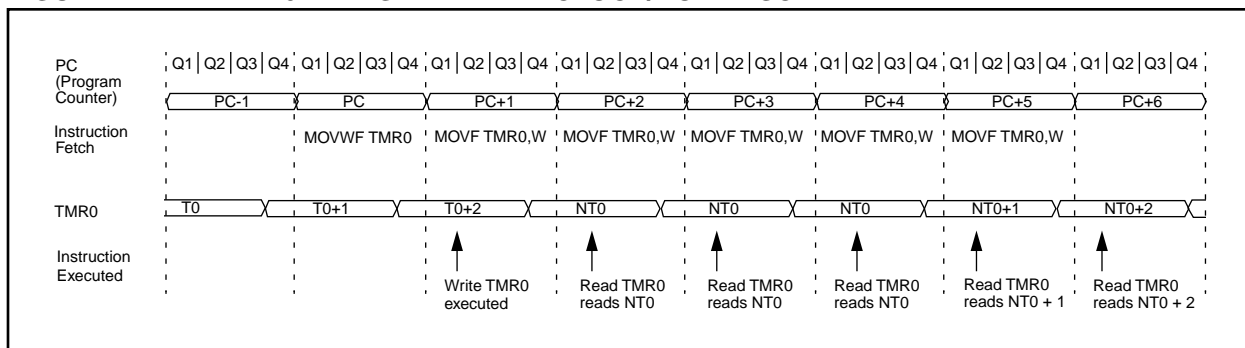
Applicable Devices									
61	62	62A	R62	63	64	64A	R64	65	65A

The TMR0 interrupt is generated when the register (TMR0) overflows from FFh to 00h. This overflow sets interrupt flag bit T0IF (INTCON<2>). The interrupt can be masked by clearing enable bit T0IE (INTCON<5>). Flag bit T0IF must be cleared in software by the Timer0 interrupt service routine before re-enabling this interrupt. The TMR0 interrupt cannot wake the processor from SLEEP since the timer is shut off during SLEEP. Figure 7-4 displays the Timer0 interrupt timing.

**FIGURE 7-1: TIMER0 BLOCK DIAGRAM**



**FIGURE 7-2: TIMER0 TIMING: INTERNAL CLOCK/NO PRESCALER**



# PIC16C6X

FIGURE 7-3: TIMER0 TIMING: INTERNAL CLOCK/PRESCALE 1:2

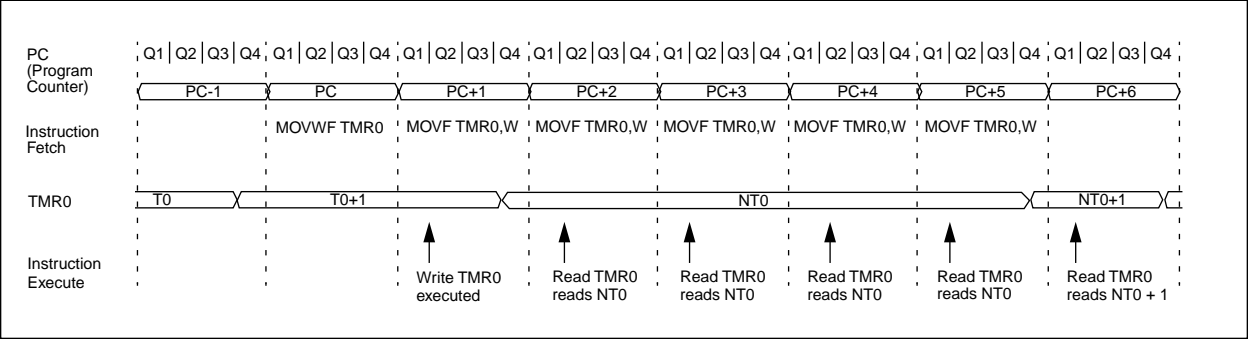
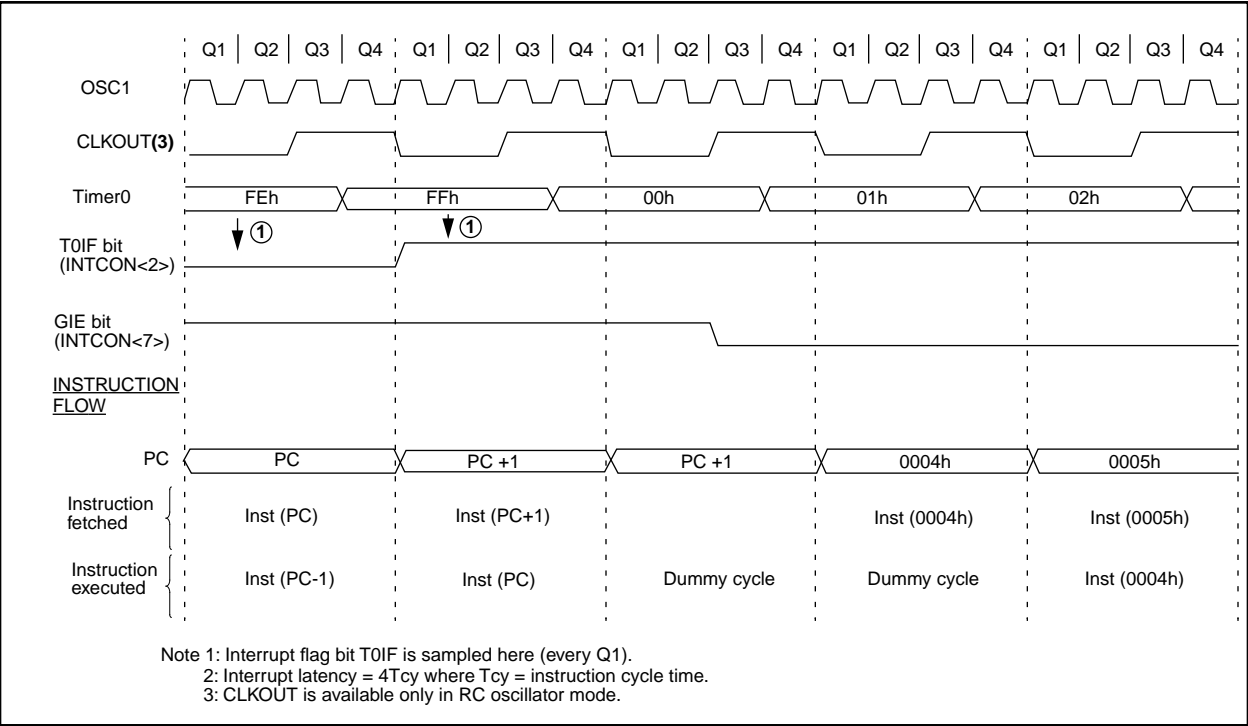


FIGURE 7-4: TIMER0 INTERRUPT TIMING



## 7.2 Using Timer0 with External Clock

### Applicable Devices

61	62	62A	R62	63	64	64A	R64	65	65A
----	----	-----	-----	----	----	-----	-----	----	-----

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

### 7.2.1 EXTERNAL CLOCK SYNCHRONIZATION

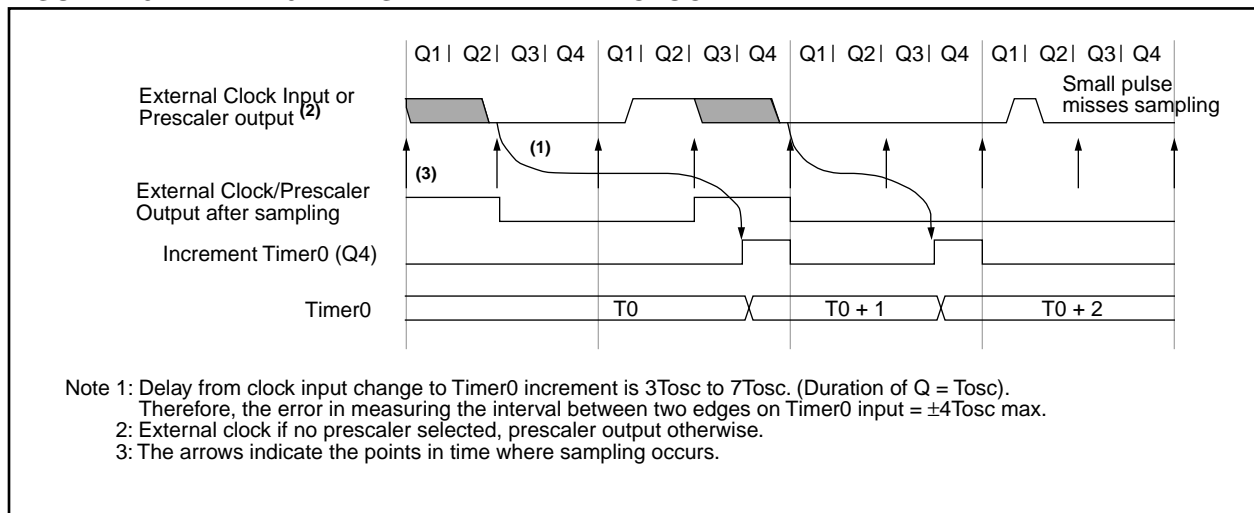
When no prescaler is used, the external clock input is the same as the prescaler output. The synchronization of T0CKI with the internal phase clocks is accomplished by sampling the prescaler output on the Q2 and Q4 cycles of the internal phase clocks (Figure 7-5). Therefore, it is necessary for T0CKI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

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

### 7.2.2 TIMER0 INCREMENT DELAY

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

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



# PIC16C6X

## 7.3 Prescaler

### Applicable Devices

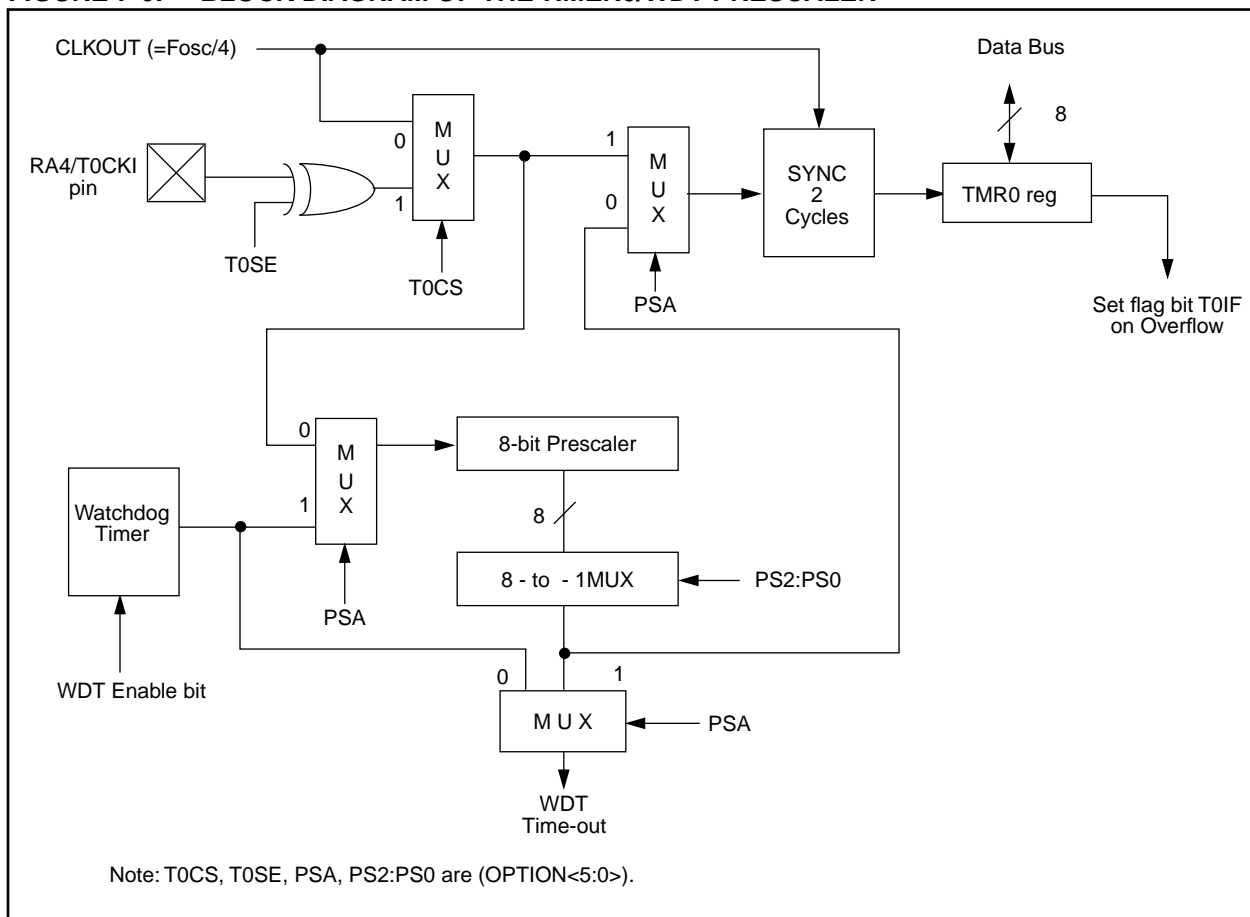
61	62	62A	R62	63	64	64A	R64	65	65A
----	----	-----	-----	----	----	-----	-----	----	-----

An 8-bit counter is available as a prescaler for the Timer0 module or as a postscaler for the Watchdog Timer (WDT), respectively (Figure 7-6). For simplicity, this counter is being referred to as “prescaler” throughout this data sheet. Note that the prescaler may be used by either the Timer0 module or the Watchdog Timer, but not both. Thus, a prescaler assignment for the Timer0 module means that there is no prescaler for the Watchdog Timer, and vice-versa.

The PSA and PS2:PS0 bits (OPTION<3:0>) determine the prescaler assignment and prescale ratio.

When assigned to the Timer0 module, all instructions writing to the TMR0 register (e.g. `CLRF 1`, `MOVWF 1`, `BSF 1,x`) will clear the prescaler count. When assigned to Watchdog Timer, a `CLRWDT` instruction will clear the prescaler count along with the Watchdog Timer. The prescaler is not readable or writable.

**FIGURE 7-6: BLOCK DIAGRAM OF THE TIMER0/WDT PRESCALER**





## 7.3.1 SWITCHING PRESCALER ASSIGNMENT

The prescaler assignment is fully under software control, i.e., it can be changed “on the fly” during program execution.

**Note:** To avoid an unintended device RESET, the following instruction sequence (shown in Example 7-1) must be executed when changing the prescaler assignment from Timer0 to the WDT. This precaution must be followed even if the WDT is disabled.

### EXAMPLE 7-1: CHANGING PRESCALER (TIMER0→WDT)

```
BCF    STATUS, RP0    ;Bank 0
CLRF   TMR0           ;Clear TMR0 & Prescaler
BSF    STATUS, RP0    ;Bank 1
CLRWDT           ;Clears WDT
MOVLW  b'xxxxlxxx'    ;Select new prescale
MOVWF  OPTION_REG     ;value & WDT
BCF    STATUS, RP0    ;Bank 0
```

To change prescaler from the WDT to the Timer0 module, use the sequence shown in Example 7-2.

### EXAMPLE 7-2: CHANGING PRESCALER (WDT→TIMER0)

```
CLRWDT           ;Clear WDT and
                  ;prescaler
BSF     STATUS, RP0 ;Bank 1
MOVLW   b'xxx0xxx' ;Select TMR0, new
                  ;prescale value and
MOVWF   OPTION_REG ;clock source
BCF     STATUS, RP0 ;Bank 0
```

**TABLE 7-1: REGISTERS ASSOCIATED WITH TIMER0**

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
01h	TMR0	Timer0 module's register								xxxx xxxx	uuuu uuuu
0Bh/8Bh	INTCON	GIE	PEIE <sup>(1)</sup>	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
81h	OPTION	RBPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
85h	TRISA	—	—	TRISA5 <sup>(1)</sup>	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	--11 1111	--11 1111

Legend: x = unknown, u = unchanged, - = unimplemented locations read as '0'. Shaded cells are not used by Timer0.

Note 1: TRISA<5> and bit PEIE are not implemented on the PIC16C61, read as '0'.

# PIC16C6X

---

NOTES:

## 8.0 TIMER1 MODULE

### Applicable Devices

61	62	62A	R62	63	64	64A	R64	65	65A
----	----	-----	-----	----	----	-----	-----	----	-----

Timer1 is a 16-bit timer/counter consisting of two 8-bit registers (TMR1H and TMR1L) which are readable and writable. Register TMR1 (TMR1H + TMR1L) increments from 0000h to FFFFh and rolls over to 0000h. The TMR1 Interrupt, if enabled, is generated on overflow which is latched in interrupt flag bit TMR1IF (PIR1<0>). This interrupt can be enabled or disabled using the TMR1 interrupt enable bit TMR1IE (PIE1<0>).

Timer1 can operate in one of two modes:

- As a timer
- As a counter

The operating mode is determined by clock select bit, TMR1CS (T1CON<1>) (Figure 8-2).

In timer mode, Timer1 increments every instruction cycle. In counter mode, it increments on every rising edge of the external clock input.

Timer1 can be turned on or off using the control bit TMR1ON (T1CON<0>).

Timer1 also has an internal "reset input". This reset can be generated by CCP1 or CCP2 (Capture/Compare/PWM) module. See Section 10.0 for details. Figure 8-1 shows the Timer1 control register.

For the PIC16C62A/R62/63/64A/R64/65A, when the Timer1 oscillator is enabled (T1OSCEN is set), the RC1 and RC0 pins become inputs. That is, the TRISC<1:0> value is ignored.

For the PIC16C62/64/65, when the Timer1 oscillator is enabled (T1OSCEN is set), RC1 pin becomes an input, however the RC0 pin will have to be configured as an input by setting the TRISC<0> bit.

The Timer1 module also has a software programmable prescaler.

**FIGURE 8-1: T1CON: TIMER1 CONTROL REGISTER (ADDRESS 10h)**

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	T1CKPS1	T1CKPS0	T1OSCEN	T1SYN $\overline{C}$	TMR1CS	TMR1ON
bit7							bit0

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

bit 7-6: **Unimplemented:** Read as '0'

bit 5-4: **T1CKPS1:T1CKPS0:** Timer1 Input Clock Prescale Select bits  
11 = 1:8 Prescale value  
10 = 1:4 Prescale value  
01 = 1:2 Prescale value  
00 = 1:1 Prescale value

bit 3: **T1OSCEN:** Timer1 Oscillator Enable Control bit  
1 = Oscillator is enabled  
0 = Oscillator is shut off  
Note: The oscillator inverter and feedback resistor are turned off to eliminate power drain.

bit 2: **T1SYN $\overline{C}$ :** Timer1 External Clock Input Synchronization Control bit  
TMR1CS = 1  
1 = Do not synchronize external clock input  
0 = Synchronize external clock input  
TMR1CS = 0  
This bit is ignored. Timer1 uses the external clock when TMR1CS = 0.

bit 1: **TMR1CS:** Timer1 Clock Source Select bit  
1 = External clock from T1OSI (on the rising edge) (See pinouts for pin with T1OSI function)  
0 = Internal clock (Fosc/4)

bit 0: **TMR1ON:** Timer1 On bit  
1 = Enables Timer1  
0 = Stops Timer1

## 8.1 Timer1 Operation in Timer Mode

### Applicable Devices

61	62	62A	R62	63	64	64A	R64	65	65A
----	----	-----	-----	----	----	-----	-----	----	-----

Timer mode is selected by clearing bit TMR1CS (T1CON<1>). In this mode, the input clock to the timer is  $F_{osc}/4$ . The synchronize control bit T1SYNC (T1CON<2>) has no effect since the internal clock is always in sync.

## 8.2 Timer1 Operation in Synchronized Counter Mode

### Applicable Devices

61	62	62A	R62	63	64	64A	R64	65	65A
----	----	-----	-----	----	----	-----	-----	----	-----

Counter mode is selected by setting bit TMR1CS. In this mode the timer increments on every rising edge of clock input on T1OSI when enable bit T1OSCEN is set or pin with T1CKI when bit T1OSCEN is cleared.

**Note:** The T1OSI function is multiplexed to different pins, depending on the device. See the pinout descriptions to see which pin has the T1OSI function.

If T1SYNC is cleared, then the external clock input is synchronized with internal phase clocks. The synchronization is done after the prescaler stage. The prescaler stage is an asynchronous ripple counter.

In this configuration, during SLEEP mode, Timer1 will not increment even if an external clock is present, since the synchronization circuit is shut off. The prescaler, however, will continue to increment.

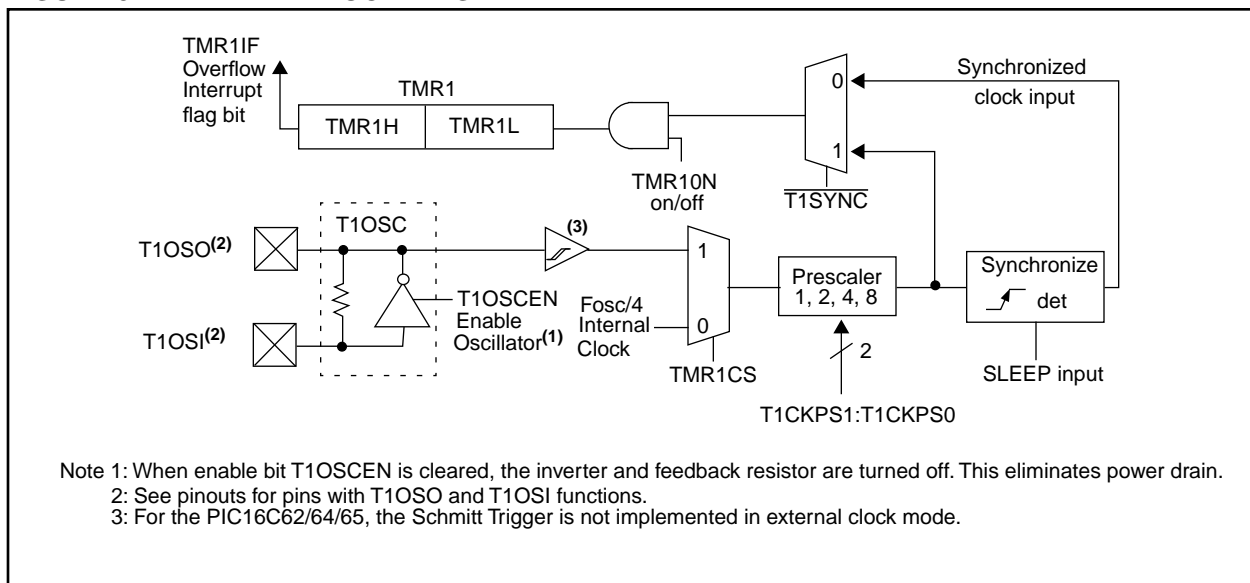
## 8.2.1 EXTERNAL CLOCK INPUT TIMING FOR SYNCHRONIZED COUNTER MODE

When an external clock input is used for Timer1 in synchronized counter mode, it must meet certain requirements. The external clock requirement is due to internal phase clock (Tosc) synchronization. Also, there is a delay in the actual incrementing of TMR1 after synchronization.

When the prescaler is 1:1, the external clock input is the same as the prescaler output. The synchronization of T1CKI with the internal phase clocks is accomplished by sampling the prescaler output on the Q2 and Q4 cycles of the internal phase clocks. Therefore, it is necessary for T1CKI to be high for at least  $2T_{osc}$  (and a small RC delay of 20 ns) and low for at least  $2T_{osc}$  (and a small RC delay of 20 ns). Refer to appropriate electrical specification section, parameters 45, 46, and 47.

When a prescaler other than 1:1 is used, the external clock input is divided by the asynchronous ripple-counter type prescaler so that the prescaler output is symmetrical. In order for the external clock to meet the sampling requirement, the ripple counter must be taken into account. Therefore, it is necessary for T1CKI to have a period of at least  $4T_{osc}$  (and a small RC delay of 40 ns) divided by the prescaler value. The only requirement on T1CKI high and low time is that they do not violate the minimum pulse width requirements of 10 ns). Refer to applicable electrical specification section, parameters 40, 42, 45, 46, and 47.

FIGURE 8-2: TIMER1 BLOCK DIAGRAM



## 8.3 Timer1 Operation in Asynchronous Counter Mode

Applicable Devices									
61	62	62A	R62	63	64	64A	R64	65	65A

If control bit  $\overline{T1SYNC}$  (T1CON<2>) is set, the external clock input is not synchronized. The timer continues to increment asynchronous to the internal phase clocks. The timer will continue to run during SLEEP and generate an interrupt on overflow which will wake the processor. However, special precautions in software are needed to read-from or write-to the Timer1 register pair, TMR1L and TMR1H (Section 8.3.2).

In asynchronous counter mode, Timer1 cannot be used as a time-base for capture or compare operations.

### 8.3.1 EXTERNAL CLOCK INPUT TIMING WITH UNSYNCHRONIZED CLOCK

If control bit  $\overline{T1SYNC}$  is set, the timer will increment completely asynchronously. The input clock must meet certain minimum high time and low time requirements, as specified in timing parameters (45 - 47).

### 8.3.2 READING AND WRITING TMR1 IN ASYNCHRONOUS COUNTER MODE

Reading TMR1H or TMR1L, while the timer is running from an external asynchronous clock, will ensure a valid read (taken care of in hardware). However, the user should keep in mind that reading the 16-bit timer in two 8-bit values itself poses certain problems since the timer may overflow between the reads.

For writes, it is recommended that the user simply stop the timer and write the desired values. A write contention may occur by writing to the timer registers while the register is incrementing. This may produce an unpredictable value in the timer register.

Reading the 16-bit value requires some care. Example 8-1 is an example routine to read the 16-bit timer value. This is useful if the timer cannot be stopped.

## EXAMPLE 8-1: READING A 16-BIT FREE-RUNNING TIMER

```
; All Interrupts are disabled
MOVWF    TMR1H, W    ;Read high byte
MOVWF    TMPH        ;
MOVWF    TMR1L, W    ;Read low byte
MOVWF    TMPL        ;
MOVWF    TMR1H, W    ;Read high byte
SUBWF    TMPH, W      ;Sub 1st read
                     ;with 2nd read
BTFSC    STATUS, Z    ;is result = 0
GOTO     CONTINUE     ;Good 16-bit read
; TMR1L may have rolled over between the read
; of the high and low bytes. Reading the high
; and low bytes now will read a good value.
MOVWF    TMR1H, W    ;Read high byte
MOVWF    TMPH        ;
MOVWF    TMR1L, W    ;Read low byte
MOVWF    TMPL        ;
; Re-enable Interrupt (if required)
CONTINUE                ;Continue with
:                        ;your code
```

## 8.4 Timer1 Oscillator

Applicable Devices									
61	62	62A	R62	63	64	64A	R64	65	65A

A crystal oscillator circuit is built in-between T1OSI (input) and T1OSO (amplifier output). It is enabled by setting control bit T1OSCEN (T1CON<3>). The oscillator is a low power oscillator rated up to 200 kHz. It will continue to run during SLEEP. It is primarily intended for a 32 kHz crystal. Table 8-1 shows the capacitor selection for the Timer1 oscillator.

The Timer1 oscillator is identical to the LP oscillator. The user must allow a software time delay to ensure proper oscillator start-up.

**TABLE 8-1: CAPACITOR SELECTION FOR THE TIMER1 OSCILLATOR**

Osc Type	Freq	C1	C2
LP	32 kHz <sup>(1)</sup>	15 pF	15 pF
	100 kHz	15 pF	15 pF
	200 kHz	0 - 15 pF	0 - 15 pF
Higher capacitance increases the stability of oscillator but also increases the start-up time. These values are for design guidance only. Note 1: For VDD > 4.5V, C1 = C2 ≈ 30 pF is recommended.			
<b>Crystals Tested:</b>			
32.768 kHz	Epson C-001R32.768K-A	± 20 PPM	
100 kHz	Epson C-2 100.00 KC-P	± 20 PPM	
200 kHz	STD XTL 200.000 kHz	± 20 PPM	

# PIC16C6X

## 8.5 Resetting Timer1 using a CCP Trigger Output

Applicable Devices									
61	62	62A	R62	63	64	64A	R64	65	65A

CCP2 is implemented on the PIC16C63/65/65A only. If CCP1 or CCP2 module is configured in Compare mode to generate a “special event trigger” (CCPxM3:CCPxM0 = 1011), this signal will reset Timer1.

**Note:** The “special event trigger” from the CCP1 and CCP2 modules will not set interrupt flag bit TMR1IF (PIR1<0>).

Timer1 must be configured for either timer or synchronized counter mode to take advantage of this feature. If the Timer1 is running in asynchronous counter mode, this reset operation may not work.

In the event that a write to Timer1 coincides with a special event trigger from CCP1 or CCP2, the write will take precedence.

In this mode of operation, the CCPRxH:CCPRxL registers pair effectively becomes the period register for the Timer1 module.

## 8.6 Resetting of TMR1 Register Pair (TMR1H + TMR1L)

Applicable Devices									
61	62	62A	R62	63	64	64A	R64	65	65A

The TMR1H and TMR1L registers are not reset on a POR or any other reset except by the CCP1 special event trigger.

T1CON register is reset to 00h on Power-on Reset or a Brown-out Reset. In all other resets, the register is unaffected.

## 8.7 Timer1 Prescaler

Applicable Devices									
61	62	62A	R62	63	64	64A	R64	65	65A

The prescaler counter is cleared on writes to the TMR1H or TMR1L registers

**TABLE 8-2: REGISTERS ASSOCIATED WITH TIMER1 AS A TIMER/COUNTER**

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	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	PSPIF <sup>(2)</sup>	<sup>(3)</sup>	RCIF <sup>(1)</sup>	TXIF <sup>(1)</sup>	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
8Ch	PIE1	PSPIE <sup>(2)</sup>	<sup>(3)</sup>	RCIE <sup>(1)</sup>	TXIE <sup>(1)</sup>	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
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	T1SYN <sup>C</sup>	TMR1CS	TMR1ON	--00 0000	--uu uuuu

Legend: x = unknown, u = unchanged, – = unimplemented read as '0'. Shaded cells are not used by Timer1 module.

Note 1: The USART is implemented on the PIC16C63/65/65A only.

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

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

## 9.0 TIMER2 MODULE

### Applicable Devices

61	62	62A	R62	63	64	64A	R64	65	65A
----	----	-----	-----	----	----	-----	-----	----	-----

Timer2 is an 8-bit timer with a prescaler and a postscaler. It is especially suitable as PWM time-base for PWM mode of CCP module(s). TMR2 is a readable and writable register, and is cleared on any device reset.

The input clock (FOSC/4) has a prescale option of 1:1, 1:4 or 1:16 (selected by control bits T2CKPS1:T2CKPS0 (T2CON<1:0>).

The Timer2 module has an 8-bit period register, PR2. Timer2 increments from 00h until it matches PR2 and then resets to 00h on the next increment cycle. PR2 is a readable and writable register. The PR2 register is set during a reset.

The match output of the TMR2 register goes through a 4-bit postscaler (which gives a 1:1 to 1:16 scaling, inclusive) to generate a TMR2 interrupt (latched in flag bit TMR2IF (PIR1<1>).

The Timer2 module can be shut off by clearing control bit TMR2ON (T2CON<2>) to minimize power consumption.

Figure 9-2 shows the Timer2 control register.

## 9.1 Timer2 Prescaler and Postscaler

### Applicable Devices

61	62	62A	R62	63	64	64A	R64	65	65A
----	----	-----	-----	----	----	-----	-----	----	-----

The prescaler and postscaler counters are cleared when any of the following occurs:

- a write to the TMR2 register
- a write to the T2CON register
- any device reset (POR, BOR,  $\overline{\text{MCLR}}$  Reset, or WDT Reset).

TMR2 register will not clear when T2CON is written, only for a WDT, BOR, POR, and  $\overline{\text{MCLR}}$  reset.

## 9.2 Output of TMR2

### Applicable Devices

61	62	62A	R62	63	64	64A	R64	65	65A
----	----	-----	-----	----	----	-----	-----	----	-----

The output of TMR2 (before the postscaler) is fed to the Synchronous Serial Port module which optionally uses it to generate shift clock.

FIGURE 9-1: TIMER2 BLOCK DIAGRAM

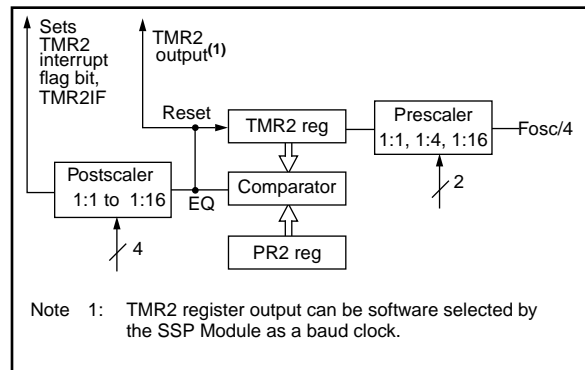


FIGURE 9-2: T2CON: TIMER2 CONTROL REGISTER (ADDRESS 12h)

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0
bit7							bit0

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

bit 6-3: **TOUTPS3:TOUTPS0:** Timer2 Output Postscale Select bits  
0000 = 1:1 postscale  
0001 = 1:2 postscale  
•  
•  
1111 = 1:16 postscale

bit 2: **TMR2ON:** Timer2 On bit  
1 = Timer2 is on  
0 = Timer2 is off

bit 1-0: **T2CKPS1:T2CKPS0:** Timer2 Clock Prescale Select bits  
00 = 1:1 prescale  
01 = 1:4 prescale  
1x = 1:16 prescale

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

# PIC16C6X

**TABLE 9-1: REGISTERS ASSOCIATED WITH TIMER2 AS A TIMER/COUNTER**

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	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	PSPIF <sup>(2)</sup>	<sup>(3)</sup>	RCIF <sup>(1)</sup>	TXIF <sup>(1)</sup>	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
8Ch	PIE1	PSPIE <sup>(2)</sup>	<sup>(3)</sup>	RCIE <sup>(1)</sup>	TXIE <sup>(1)</sup>	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
11h	TMR2	Timer2 module's register								0000 0000	0000 0000
12h	T2CON	—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	–000 0000	–000 0000
92h	PR2	Timer2 Period register								1111 1111	1111 1111

Legend: x = unknown, u = unchanged, – = unimplemented locations read as '0'. Shaded cells are not used by Timer2.

Note 1: The USART is implemented on the PIC16C63/65/65A only.

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

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



## 10.0 CAPTURE/COMPARE/PWM (CCP) MODULE(s)

Applicable Devices										
61	62	62A	R62	63	64	64A	R64	65	65A	CCP1
61	62	62A	R62	63	64	64A	R64	65	65A	CCP2

Each CCP (Capture/Compare/PWM) module contains a 16-bit register which can operate as a 16-bit capture register, as a 16-bit compare register, or as a PWM master/slave duty cycle register. Both the CCP1 and CCP2 modules are identical in operation, with the exception of the operation of the special event trigger. Table 10-1 and Table 10-2 show the resources and interactions of the CCP modules(s). In the following sections, the operation of a CCP module is described with respect to CCP1. CCP2 operates the same as CCP1, except where noted.

### CCP1 module:

Capture/Compare/PWM Register1 (CCPR1) is comprised of two 8-bit registers: CCPR1L (low byte) and CCPR1H (high byte). The CCP1CON register controls the operation of CCP1. All are readable and writable.

### CCP2 module:

Capture/Compare/PWM Register2 (CCPR2) is comprised of two 8-bit registers: CCPR2L (low byte) and CCPR2H (high byte). The CCP2CON register controls the operation of CCP2. All are readable and writable.

For use of the CCP modules, refer to the *Embedded Control Handbook*, "Using the CCP Modules" (AN594).

**TABLE 10-1: CCP MODE - TIMER RESOURCE**

CCP Mode	Timer Resource
Capture	Timer1
Compare	Timer1
PWM	Timer2

**TABLE 10-2: INTERACTION OF TWO CCP MODULES**

CCPx Mode	CCPy Mode	Interaction
Capture	Capture	Same TMR1 time-base.
Capture	Compare	The compare should be configured for the special event trigger, which clears TMR1.
Compare	Compare	The compare(s) should be configured for the special event trigger, which clears TMR1.
PWM	PWM	The PWMs will have the same frequency, and update rate (TMR2 interrupt).
PWM	Capture	None
PWM	Compare	None

# PIC16C6X

**FIGURE 10-1: CCP1CON REGISTER (ADDRESS 17h) / CCP2CON REGISTER (ADDRESS 1Dh)**

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	CCPxX	CCPxY	CCPxM3	CCPxM2	CCPxM1	CCPxM0
bit7							bit0

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

bit 7-6: **Unimplemented:** Read as '0'

bit 5-4: **CCPxX:CCPxY:** PWM Least Significant bits  
Capture Mode  
 Unused  
Compare Mode  
 Unused  
PWM Mode  
 These bits are the two LSbs of the PWM duty cycle. The eight MSbs are found in CCPRxL.

bit 3-0: **CCPxM3:CCPxM0:** CCPx Mode Select bits  
 0000 = Capture/Compare/PWM off (resets CCPx module)  
 0100 = Capture mode, every falling edge  
 0101 = Capture mode, every rising edge  
 0110 = Capture mode, every 4th rising edge  
 0111 = Capture mode, every 16th rising edge  
 1000 = Compare mode, set output on match (bit CCPxIF is set)  
 1001 = Compare mode, clear output on match (bit CCPxIF is set)  
 1010 = Compare mode, generate software interrupt on match (bit CCPxIF is set, CCPx pin is unaffected)  
 1011 = Compare mode, trigger special event (CCPxIF bit is set; CCP1 resets TMR1; CCP2 resets TMR1)  
 11xx = PWM mode

## 10.1 Capture Mode

Applicable Devices									
61	62	62A	R62	63	64	64A	R64	65	65A

In Capture mode, CCPR1H:CCPR1L captures the 16-bit value of the TMR1 register when an event occurs on pin RC2/CCP1 (Figure 10-2). An event is defined as:

- A falling edge
- A rising edge
- Every 4th rising edge
- Every 16th rising edge

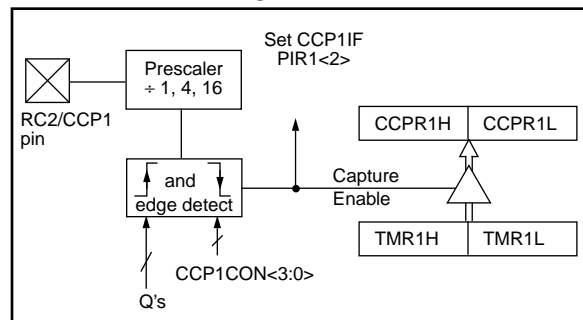
An event is selected by control bits CCP1M3:CCP1M0 (CCP1CON<3:0>). When a capture is made, the interrupt request flag bit CCP1IF (PIR1<2>) is set. It must be reset in software. If another capture occurs before the value in register CCPR1 is read, the old captured value will be lost.

### 10.1.1 CCP PIN CONFIGURATION

In capture mode, the RC2/CCP1 pin should be configured as an input by setting its corresponding TRIS bit.

**Note:** If the RC2/CCP1 pin is configured as an output, a write to the port can cause a capture condition.

**FIGURE 10-2: CAPTURE MODE OPERATION BLOCK DIAGRAM**



### 10.1.2 TIMER1 MODE SELECTION

Timer1 must be running in timer mode or synchronized counter mode for the CCP module to use the capture feature. In asynchronous counter mode the capture operation may not work.

### 10.1.3 SOFTWARE INTERRUPT

When the Capture mode is changed, a false capture interrupt may be generated. The user should keep enable bit CCP1IE (PIE1<2>) clear to avoid false interrupts and should clear flag bit CCP1IF following any such change in operating mode.

## 10.1.4 PRESCALER

There are four prescaler settings, specified by bits CCP1M3:CCP1M0. Whenever the CCP module is turned off, or the CCP module is not in Capture mode, the prescaler counter is cleared. This means that any reset will clear the prescaler counter.

Switching from one capture prescaler to another may generate an interrupt. Also, the prescaler counter will not be cleared, therefore the first capture may be from a non-zero prescaler. Example 10-1 shows the recommended method for switching between capture prescalers. This example also clears the prescaler counter and will not generate the “false” interrupt.

### EXAMPLE 10-1: CHANGING BETWEEN CAPTURE PRESCALERS

```
CLRF    CCP1CON    ; Turn CCP module off
MOVLW   NEW_CAPT_PS ; Load the W reg with
                        ; the new prescaler
                        ; mode value and CCP ON
MOVWF   CCP1CON    ; Load CCP1CON with
                        ; this value
```

## 10.2 Compare Mode

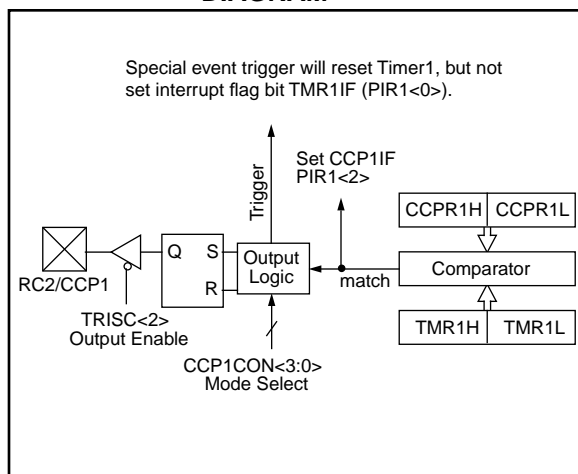
Applicable Devices											
61	62	62A	R62	63	64	64A	R64	65	65A		

In Compare mode, the 16-bit CCPR1 register value is constantly compared against the TMR1 register pair value. When a match occurs, the RC2/CCP1 pin is:

- Driven High
- Driven Low
- Remains Unchanged

The action on the pin is based on the value of control bits CCP1M3:CCP1M0 (CCP1CON<3:0>). At the same time, a compare interrupt is also generated.

**FIGURE 10-3: COMPARE MODE OPERATION BLOCK DIAGRAM**



## 10.2.1 CCP PIN CONFIGURATION

The user must configure the RC2/CCP1 pin as an output by clearing the TRISC<2> bit.

**Note:** Clearing the CCP1CON register will force the RC2/CCP1 compare output latch to the default low level. This is not the data latch.

## 10.2.1 TIMER1 MODE SELECTION

Timer1 must be running in Timer mode or Synchronized Counter mode if the CCP module is using the compare feature. In Asynchronous Counter mode, the compare operation may not work.

## 10.2.2 SOFTWARE INTERRUPT MODE

When Generate Software Interrupt is chosen, the CCP1 pin is not affected. Only a CCP interrupt is generated (if enabled).

## 10.2.3 SPECIAL EVENT TRIGGER

In this mode, an internal hardware trigger is generated which may be used to initiate an action.

The special event trigger output of CCP1 and CCP2 resets the TMR1 register pair. This allows the CCPR1 and CCPR2 registers to effectively be 16-bit programmable period register for Timer1.

For compatibility issues, the special event trigger output of CCP2 on PIC16C7X devices also starts an A/D conversion.

**Note:** The “special event trigger” from the CCP1 and CCP2 modules will not set interrupt flag bit TMR1IF (PIR1<0>).

# PIC16C6X

## 10.3 PWM Mode

### Applicable Devices

61	62	62A	R62	63	64	64A	R64	65	65A
----	----	-----	-----	----	----	-----	-----	----	-----

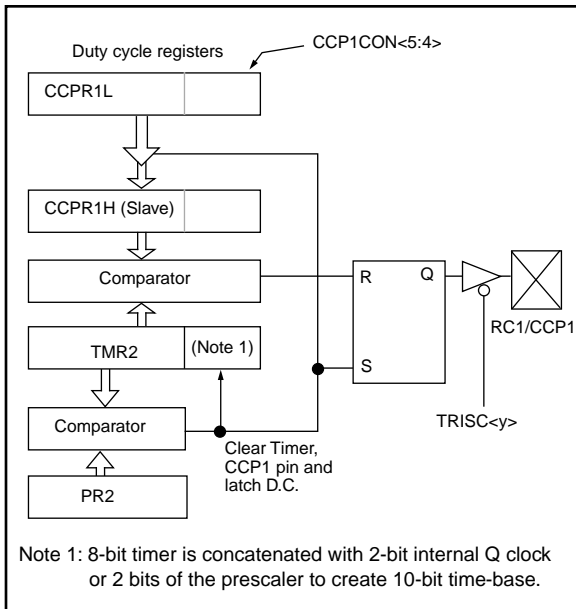
In Pulse Width Modulation (PWM) mode, the CCPx pin produces up to a 10-bit resolution PWM output. Since the CCPx pin is multiplexed with the PORTC data latch, the corresponding TRISC bit must be cleared to make the CCPx pin an output.

**Note:** Clearing the CCP1CON register will force the CCP1 PWM output latch to the default low level. This is not the PORTC I/O data latch.

Figure 10-4 shows a simplified block diagram of the CCP module in PWM mode.

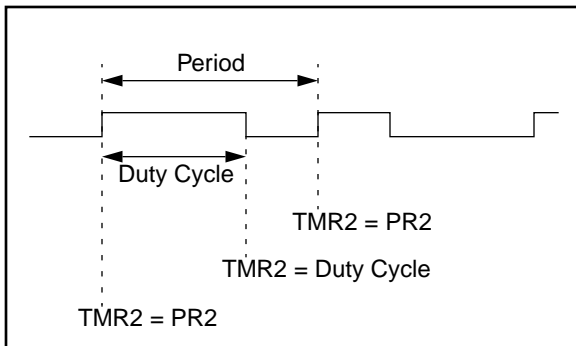
For a step by step procedure on how to set up the CCP module for PWM operation, see Section 10.3.3.

**FIGURE 10-4: SIMPLIFIED PWM BLOCK DIAGRAM**



A PWM output (Figure 10-5) has a time base (period) and a time that the output stays high (duty cycle). The frequency of the PWM is the inverse of the period (1/period).

**FIGURE 10-5: PWM OUTPUT**



### 10.3.1 PWM PERIOD

The PWM period is specified by writing to the PR2 register. The PWM period can be calculated using the following formula:

$$\text{PWM period} = [(PR2) + 1] * 4 * T_{osc} * (\text{TMR2 prescale value})$$

PWM frequency is defined as  $1 / [\text{PWM period}]$ .

When TMR2 is equal to PR2, the following three events occur:

- TMR2 is cleared
- The CCP1 pin is set (exception: if PWM duty cycle = 0%, the CCP1 pin will not be set)
- The PWM duty cycle is latched from CCPR1L into CCPR1H

**Note:** The Timer2 postscaler (see Section 9.1) is not used in the determination of the PWM frequency. The postscaler could be used to have a servo update rate at a different frequency than the PWM output.

### 10.3.2 PWM DUTY CYCLE

The PWM duty cycle is specified by writing to the CCPR1L register and to the CCP1CON<5:4> bits. Up to 10-bit resolution is available: the CCPR1L contains the eight MSBs and the CCP1CON<5:4> contains the two LSbs. This 10-bit value is represented by CCPR1L:CCP1CON<5:4>. The following equation is used to calculate the PWM duty cycle:

$$\text{PWM duty cycle} = (\text{CCPR1L:CCP1CON<5:4>}) * T_{osc} * (\text{TMR2 prescale value})$$

CCPR1L and CCP1CON<5:4> can be written to at any time, but the duty cycle value is not latched into CCPR1H until after a match between PR2 and TMR2 occurs (i.e., the period is complete). In PWM mode, the CCPR1H is a read-only register.

The CCPR1H register and a 2-bit internal latch are used to double buffer the PWM duty cycle. This double buffering is essential for glitchless PWM operation.

When the CCPR1H and 2-bit latch match TMR2 concatenated with an internal 2-bit Q clock or 2 bits of the TMR2 prescaler, the CCP1 pin is cleared.

**Note:** If the PWM duty cycle value is longer than the PWM period (PWM duty cycle = 100%), the CCP1 pin will not be cleared.

## EXAMPLE 10-2: PWM PERIOD AND DUTY CYCLE CALCULATION

Desired PWM frequency is 78.125 kHz,  
Fosc = 20 MHz  
TMR2 prescale = 1

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

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

$$PR2 = 63$$

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

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

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

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

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

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

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. 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 appropriate TRISC bit.
4. Set the TMR2 prescale value and enable Timer2 by writing to T2CON.
5. Configure the CCP 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	INTCON	GIE	PEIE	TOIE	INTE	RBIE	TOIF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	PSPIF <sup>(2)</sup>	<sup>(3)</sup>	RCIF <sup>(1)</sup>	TXIF <sup>(1)</sup>	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
0Dh <sup>(4)</sup>	PIR2	—	—	—	—	—	—	—	CCP2IF	---- --0	---- --0
8Ch	PIE1	PSPIE <sup>(2)</sup>	<sup>(3)</sup>	RCIE <sup>(1)</sup>	TXIE <sup>(1)</sup>	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
8Dh <sup>(4)</sup>	PIE2	—	—	—	—	—	—	—	CCP2IE	---- --0	---- --0
0Eh	TMR1L	Holding register for the Least Significant Byte of the 16-bit TMR1 register								xxxx xxxx	uuuu uuuu
0Fh	TMR1H	Holding register for the Most Significant Byte of the 16-bit TMR1 register								xxxx xxxx	uuuu uuuu
10h	T1CON	—	—	T1CKPS1	T1CKPS0	T1OSCEN	T1SYNC	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 <sup>(4)</sup>	CCPR2L	Capture/Compare/PWM2 (LSB)								xxxx xxxx	uuuu uuuu
1Ch <sup>(4)</sup>	CCPR2H	Capture/Compare/PWM2 (MSB)								xxxx xxxx	uuuu uuuu
1Dh <sup>(4)</sup>	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 PIC16C63/65/65A only.

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

3: These bits are reserved, always maintain these bits clear.

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

# PIC16C6X

**TABLE 10-5: REGISTERS ASSOCIATED WITH PWM AND TIMER2**

Addr	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/8 Bh	INTCON	GIE	PEIE	TOIE	INTE	RBIE	TOIF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	PSPIF <sup>(2)</sup>	<sup>(3)</sup>	RCIF <sup>(1)</sup>	TXIF <sup>(1)</sup>	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
0Dh <sup>(4)</sup>	PIR2	—	—	—	—	—	—	—	CCP2IF	---- --0	---- --0
8Ch	PIE1	PSPIE <sup>(2)</sup>	<sup>(3)</sup>	RCIE <sup>(1)</sup>	TXIE <sup>(1)</sup>	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
8Dh <sup>(4)</sup>	PIE2	—	—	—	—	—	—	—	CCP2IE	---- --0	---- --0
11h	TMR2	Timer2 module's register								0000 0000	0000 0000
92h	PR2	Timer2 module's Period register								1111 1111	1111 1111
12h	T2CON	—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	-000 0000	-000 0000
15h	CCPR1L	Capture/Compare/PWM1 (LSB)								xxxx xxxx	uuuu uuuu
16h	CCPR1H	Capture/Compare/PWM1 (MSB)								xxxx xxxx	uuuu uuuu
17h	CCP1CON	—	—	CCP1X	CCP1Y	CCP1M3	CCP1M2	CCP1M1	CCP1M0	--00 0000	--00 0000
1Bh <sup>(4)</sup>	CCPR2L	Capture/Compare/PWM2 (LSB)								xxxx xxxx	uuuu uuuu
1Ch <sup>(4)</sup>	CCPR2H	Capture/Compare/PWM2 (MSB)								xxxx xxxx	uuuu uuuu
1Dh <sup>(4)</sup>	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 this mode.

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

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

3: These bits are reserved, always maintain these bits clear.

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

## 11.0 SYNCHRONOUS SERIAL PORT (SSP) MODULE

Applicable Devices									
61	62	62A	R62	63	64	64A	R64	65	65A

The Synchronous Serial Port (SSP) module is a serial interface useful for communicating with other peripheral or microcontroller devices. These peripheral devices may be Serial EEPROMs, shift registers, display drivers, A/D converters, etc. The SSP module can operate in one of two modes:

- Serial Peripheral Interface (SPI)
- Inter-Integrated Circuit (I<sup>2</sup>C)

Refer to Application Note AN578, "Use of the SSP Module in the I<sup>2</sup>C Multi-Master Environment."

**FIGURE 11-1: SSPSTAT: SYNC SERIAL PORT STATUS REGISTER (ADDRESS 94h)**

U-0	U-0	R-0	R-0	R-0	R-0	R-0	R-0
—	—	D/ $\overline{A}$	P	S	R/ $\overline{W}$	UA	BF
bit7							bit0

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

bit 7-6: **Unimplemented:** Read as '0'

bit 5: **D/ $\overline{A}$ :** Data/Address bit (I<sup>2</sup>C mode only)  
1 = Indicates that the last byte received or transmitted was data  
0 = Indicates that the last byte received or transmitted was address

bit 4: **P:** Stop bit (I<sup>2</sup>C mode only. This bit is cleared when the SSP module is disabled, SSPEN is cleared)  
1 = Indicates that a stop bit has been detected last (this bit is '0' on RESET)  
0 = Stop bit was not detected last

bit 3: **S:** Start bit (I<sup>2</sup>C mode only. This bit is cleared when the SSP module is disabled, SSPEN is cleared)  
1 = Indicates that a start bit has been detected last (this bit is '0' on RESET)  
0 = Start bit was not detected last

bit 2: **R/ $\overline{W}$ :** Read/Write bit information (I<sup>2</sup>C mode only)  
This bit holds the R/W bit information following the last address match. This bit is valid from the address match to the next start bit, stop bit, or  $\overline{ACK}$  bit.  
1 = Read  
0 = Write

bit 1: **UA:** Update Address (10-bit I<sup>2</sup>C mode only)  
1 = Indicates that the user needs to update the address in the SSPADD register  
0 = Address does not need to be updated

bit 0: **BF:** Buffer Full Status bit  
Receive (SPI and I<sup>2</sup>C modes)  
1 = Receive complete, SSPBUF is full  
0 = Receive not complete, SSPBUF is empty  
Transmit (I<sup>2</sup>C mode only)  
1 = Transmit in progress, SSPBUF is full  
0 = Transmit complete, SSPBUF is empty

# PIC16C6X

**FIGURE 11-2: SSPCON: SYNC SERIAL PORT CONTROL REGISTER (ADDRESS 14h)**

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0
bit7							bit0

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

bit 7: **WCOL**: Write Collision Detect bit  
1 = The SSPBUF register is written while it is still transmitting the previous word (must be cleared in software)  
0 = No collision

bit 6: **SSPOV**: Receive Overflow Detect bit  
In SPI mode  
1 = A new byte is received while the SSPBUF register is still holding the previous data. In case of overflow, the data in SSPSR register is lost. Overflow can only occur in slave mode. The user must read the SSPBUF, even if only transmitting data, to avoid setting overflow. In master mode the overflow bit is not set since each new reception (and transmission) is initiated by writing to the SSPBUF register.  
0 = No overflow  
In I<sup>2</sup>C mode  
1 = A byte is received while the SSPBUF register is still holding the previous byte. SSPOV is a "don't care" in transmit mode. SSPOV must be cleared in software in either mode.  
0 = No overflow

bit 5: **SSPEN**: Synchronous Serial Port Enable bit  
In SPI mode  
1 = Enables serial port and configures SCK, SDO, and SDI as serial port pins  
0 = Disables serial port and configures these pins as I/O port pins  
In I<sup>2</sup>C mode  
1 = Enables the serial port and configures the SDA and SCL pins as serial port pins  
0 = Disables serial port and configures these pins as I/O port pins  
In both modes, when enabled, these pins must be properly configured as input or output.

bit 4: **CKP**: Clock Polarity Select bit  
In SPI mode  
1 = Transmit happens on falling edge, receive on rising edge. Idle state for clock is a high level.  
0 = Transmit happens on rising edge, receive on falling edge. Idle state for clock is a low level.  
In I<sup>2</sup>C mode  
SCK release control  
1 = Enable clock  
0 = Holds clock low (clock stretch) (Used to ensure data setup time)

bit 3-0: **SSPM3:SSPM0**: Synchronous Serial Port Mode Select bits  
0000 = SPI master mode, clock = Fosc/4  
0001 = SPI master mode, clock = Fosc/16  
0010 = SPI master mode, clock = Fosc/64  
0011 = SPI master mode, clock = TMR2 output/2  
0100 = SPI slave mode, clock = SCK pin.  $\overline{SS}$  pin control enabled.  
0101 = SPI slave mode, clock = SCK pin.  $\overline{SS}$  pin control disabled.  $\overline{SS}$  can be used as I/O pin.  
0110 = I<sup>2</sup>C slave mode, 7-bit address  
0111 = I<sup>2</sup>C slave mode, 10-bit address  
1011 = I<sup>2</sup>C start and stop bit interrupts enabled (slave idle)  
1110 = I<sup>2</sup>C slave mode, 7-bit address with start and stop bit interrupts enabled  
1111 = I<sup>2</sup>C slave mode, 10-bit address with start and stop bit interrupts enabled



## 11.1 SPI Mode

Applicable Devices									
61	62	62A	R62	63	64	64A	R64	65	65A

The SPI mode allows 8-bits of data to be synchronously transmitted and received simultaneously. To accomplish communication, typically three pins are used:

- Serial Data Out (SDO)
- Serial Data In (SDI)
- Serial Clock (SCK)

Additionally a fourth pin may be used when in a slave mode of operation:

- Slave Select ( $\overline{SS}$ )

When initializing the SPI, several options need to be specified. This is done by programming the appropriate control bits in the SSPCON register (SSPCON<5:0>). These control bits allow the following to be specified:

- Master Mode (SCK is the clock output)
- Slave Mode (SCK is the clock input)
- Clock Polarity (Output/Input data on the Rising/Falling edge of SCK)
- Clock Rate (Master mode only)
- Slave Select Mode (Slave mode only)

The SSP consists of a transmit/receive Shift Register (SSPSR) and a Buffer register (SSPBUF). The SSPSR shifts the data in and out of the device, MSb first. The SSPBUF holds the data that was written to the SSPSR, until the received data is ready. Once the 8-bits of data have been received, that byte is moved to the SSPBUF register. Then the Buffer Full bit, BF (SSPSTAT<0>) and flag bit SSPIF are set. This double buffering of the received data (SSPBUF) allows the next byte to start reception before reading the data that was just received. Any write to the SSPBUF register during transmission/reception of data will be ignored, and the write collision detect bit, WCOL (SSPCON<7>) will be set. User software must clear bit WCOL so that it can be determined if the following write(s) to the SSPBUF completed successfully. When the application software is expecting to receive valid data, the SSPBUF register should be read before the next byte of data to transfer is written to the SSPBUF register. The Buffer Full bit BF (SSPSTAT<0>) indicates when the SSPBUF register has been loaded with the received data (transmission is complete). When the SSPBUF is read, bit BF is cleared. This data may be irrelevant if the SPI is only a transmitter. Generally the SSP Interrupt is used to determine when the transmission/reception has completed. The SSPBUF register must be read and/or written. If the interrupt method is not going to be used, then software polling can be done to ensure that a write collision does not occur. Example 11-1 shows the loading of the SSPBUF (SSPSR) register for data transmission. The shaded instruction is only required if the received data is meaningful.

### EXAMPLE 11-1: LOADING THE SSPBUF (SSPSR) REGISTER

```

BSF    STATUS, RP0    ;Specify Bank 1
LOOP  BTFSS SSPSTAT, BF ;Has data been
                                ;received
                                ;(transmit
                                ;complete)?

GOTO   LOOP            ;No

BCF    STATUS, RP0    ;Specify Bank 0
MOVWF  SSPBUF, W       ;W reg = contents
                                ;of SSPBUF

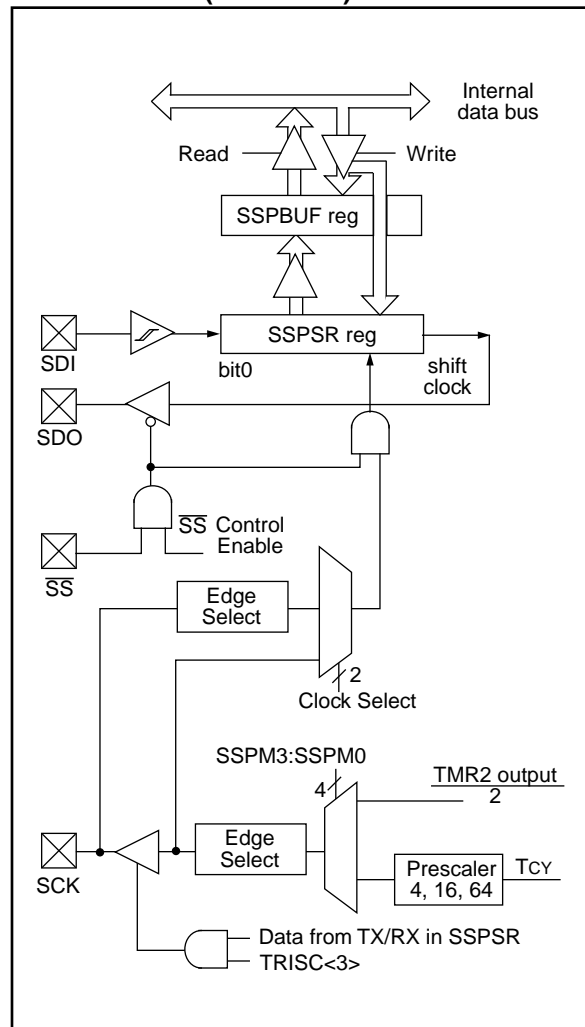
MOVWF  RXDATA          ;Save in user RAM
MOVWF  TXDATA, W       ;W reg = contents
                                ; of TXDATA

MOVWF  SSPBUF          ;New data to xmit

```

The block diagram of the SSP module, when in SPI mode (Figure 11-3), shows that the SSPSR register is not directly readable or writable, and can only be accessed from addressing the SSPBUF register. Additionally, the SSP status register (SSPSTAT) indicates the various status conditions.

**FIGURE 11-3: SSP BLOCK DIAGRAM (SPI MODE)**



# PIC16C6X

To enable the serial port, SSP enable bit SSPEN (SSPCON<5>) must be set. To reset or reconfigure SPI mode, clear enable bit SSPEN, re-initialize SSPCON register, and then set enable bit SSPEN. This configures the SDI, SDO, SCK, and  $\overline{SS}$  pins as serial port pins. For the pins to behave as the serial port function, they must have their data direction bits (in the TRISC register) appropriately programmed. That is:

- SDI must have TRISC<4> set
- SDO must have TRISC<5> cleared
- SCK (Master mode) must have TRISC<3> cleared
- SCK (Slave mode) must have TRISC<3> set
- $\overline{SS}$  must have TRISA<5> set (if implemented)

Any serial port function that is not desired may be overridden by programming the corresponding data direction (TRIS) register to the opposite value. An example would be in master mode where you are only sending data (to a display driver), then both SDI and  $\overline{SS}$  could be used as general purpose outputs by clearing their corresponding TRIS register bits.

Figure 11-4 shows a typical connection between two microcontrollers. The master controller (Processor 1) initiates the data transfer by sending the SCK signal. Data is shifted out of both shift registers on their programmed clock edge, and latched on the opposite edge of the clock. Both processors should be programmed to the same Clock Polarity (CKP), then both controllers would send and receive data at the same time. Whether the data is meaningful (or dummy data) depends on the application software. This leads to three scenarios for data transmission:

- Master sends data — Slave sends dummy data
- Master sends data — Slave sends data
- Master sends dummy data — Slave sends data

The master can initiate the data transfer at any time because it controls the SCK. The master determines when the slave (Processor 2) is to broadcast data by the software protocol.

In master mode the data is transmitted/received as soon as the SSPBUF register is written to. If the SPI is only going to receive, the SCK output could be disabled (programmed as an input). The SSPSR register will continue to shift in the signal present on the SDI pin at the programmed clock rate. As each byte is received, it will be loaded into the SSPBUF register as if a normal received byte (interrupts and status bits appropriately set). This could be useful in receiver applications as a “line activity monitor” mode.

In slave mode, the data is transmitted and received as the external clock pulses appear on SCK. When the last bit is latched interrupt flag bit SSPIF (PIR1<3>) is set.

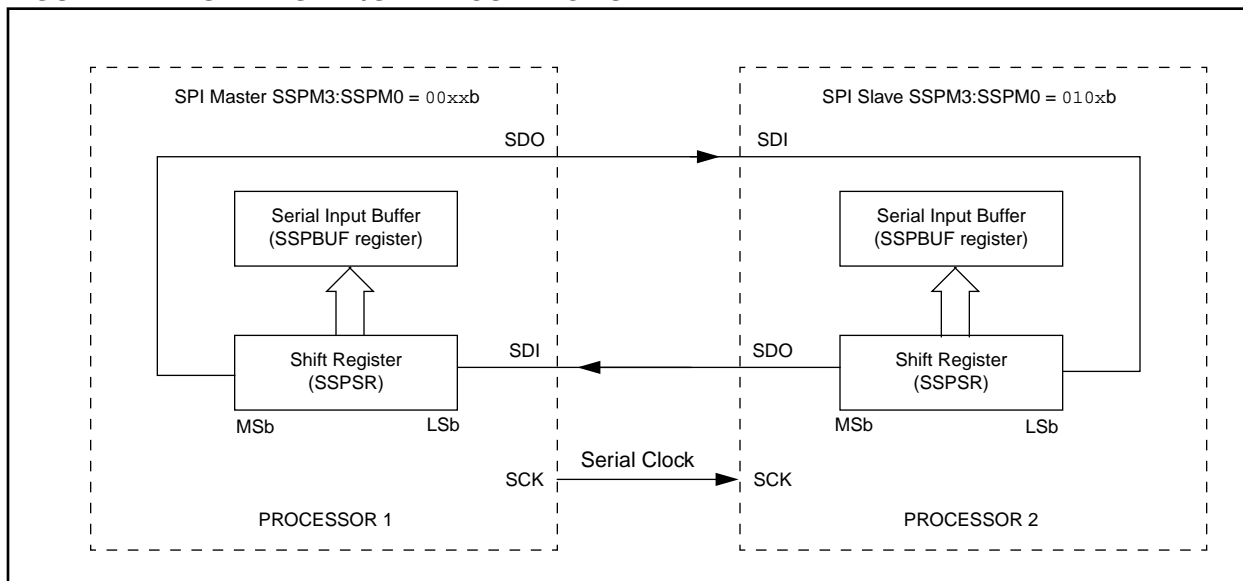
The clock polarity is selected by appropriately programming bit CKP (SSPCON<4>). This then would give waveforms for SPI communication as shown in Figure 11-5 and Figure 11-6 where the MSB is transmitted first. In master mode, the SPI clock rate (bit rate) is user programmable to be one of the following:

- $F_{osc}/4$  (or  $T_{CY}$ )
- $F_{osc}/16$  (or  $4 \cdot T_{CY}$ )
- $F_{osc}/64$  (or  $16 \cdot T_{CY}$ )
- Timer2 output/2

This allows a maximum bit clock frequency (at 20 MHz) of 5 MHz. When in slave mode the external clock must meet the minimum high and low times.

In sleep mode, the slave can transmit and receive data and wake the device from sleep.

**FIGURE 11-4: SPI MASTER/SLAVE CONNECTION**

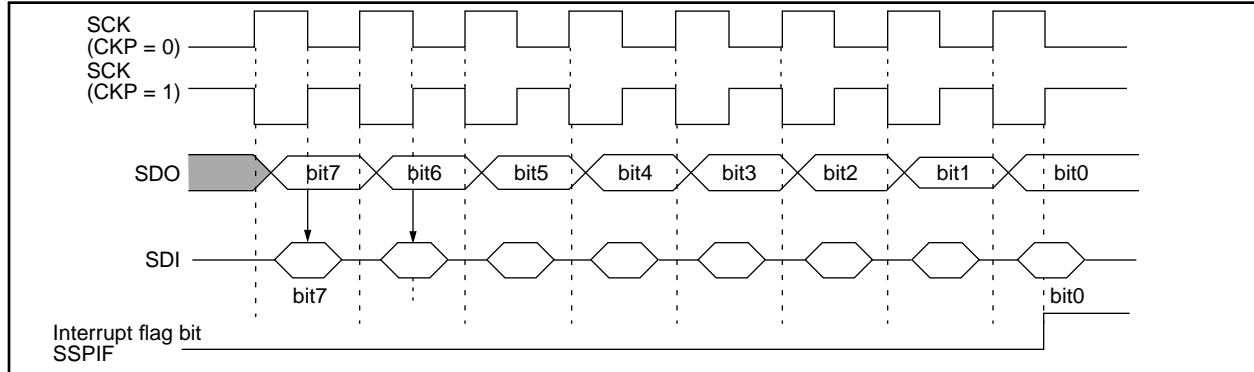


The  $\overline{SS}$  pin allows a synchronous slave mode. The SPI must be in slave mode ( $SSPCON<3:0> = 04h$ ) and the  $TRISA<5>$  bit must be set for synchronous slave mode to be enabled. When the  $\overline{SS}$  pin is low, transmission and reception are enabled and the SDO pin is driven. When the  $\overline{SS}$  pin goes high, the SDO pin is no longer driven, even if in the middle of a transmitted byte, and becomes a floating output. If the  $\overline{SS}$  pin is taken low without resetting SPI mode, the transmission will continue from the point at

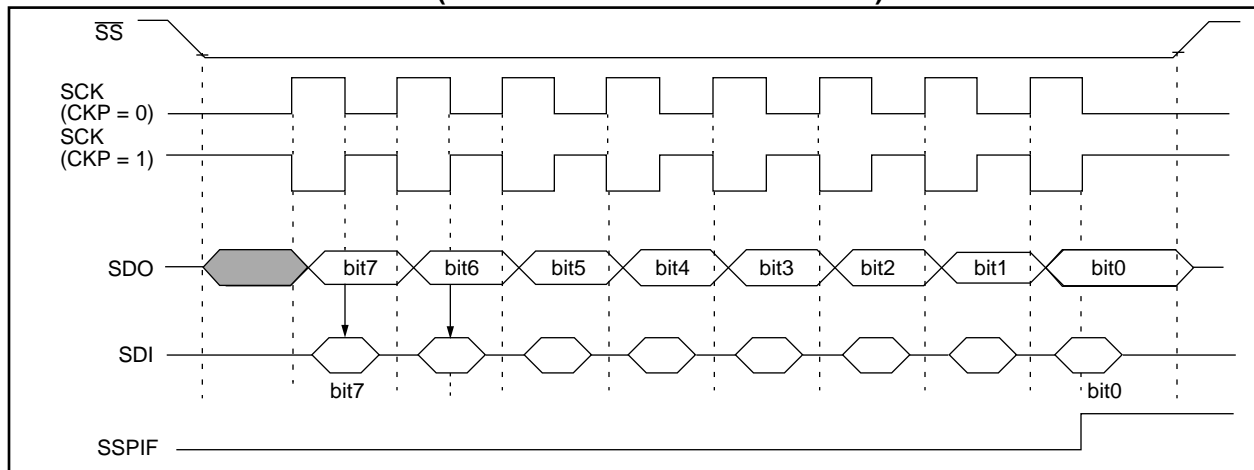
which it was taken high. External pull-up/ pull-down resistors may be desirable, depending on the application.

To emulate two-wire communication, the SDO pin can be connected to the SDI pin. When the SPI needs to operate as a receiver the SDO pin can be configured as an input. This disables transmissions from the SDO. The SDI can always be left as an input (SDI function) since it cannot create a bus conflict.

**FIGURE 11-5: SPI MODE TIMING (MASTER MODE OR SLAVE MODE W/O  $\overline{SS}$  CONTROL)**



**FIGURE 11-6: SPI MODE TIMING (SLAVE MODE WITH  $\overline{SS}$  CONTROL)**



**TABLE 11-1: REGISTERS ASSOCIATED WITH SPI 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	INTCON	GIE	PEIE	TOIE	INTE	RBIE	TOIF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	PSPIF <sup>(2)</sup>	<sup>(3)</sup>	RCIF <sup>(1)</sup>	TXIF <sup>(1)</sup>	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
8Ch	PIE1	PSPIE <sup>(2)</sup>	<sup>(3)</sup>	RCIE <sup>(1)</sup>	TXIE <sup>(1)</sup>	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
13h	SSPBUF	Synchronous Serial Port Receive Buffer/Transmit Register								xxxxx xxxxx	uuuu uuuu
14h	SSPCON	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	0000 0000
85h	TRISA	—	—	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	--11 1111	--11 1111
94h	SSPSTAT	—	—	D/A	P	S	R/W	UA	BF	--00 0000	--00 0000

Legend: x = unknown, u = unchanged, - = unimplemented locations read as '0'. Shaded cells are not used by SSP module in SPI mode.

Note 1: These bits are associated with the USART which is implemented on the PIC16C63/65/65A only.

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

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

## 11.2 I<sup>2</sup>C Overview

Applicable Devices									
61	62	62A	R62	63	64	64A	R64	65	65A

This section provides an overview of the Inter-Integrated Circuit (I<sup>2</sup>C) bus, with Section 11.3 discussing the operation of the SSP module in I<sup>2</sup>C mode.

The I<sup>2</sup>C bus is a two-wire serial interface developed by the Philips Corporation. The original specification, or standard mode, was for data transfers of up to 100 Kbps. An enhanced specification, or fast mode, supports data transmission up to 400 Kbps. Both standard mode and fast mode devices will inter-operate if attached to the same bus.

The I<sup>2</sup>C interface employs a comprehensive protocol to ensure reliable transmission and reception of data. When transmitting data, one device is the “master”, which initiates transfer on the bus and generates the clock signals to permit that transfer, while the other device(s) acts as the “slave.” All portions of the slave protocol are implemented in the SSP module’s hardware, except general call support, while portions of the master protocol need to be addressed in the PIC16CXX software. Table 11-2 defines some of the I<sup>2</sup>C bus terminology. For additional information on the I<sup>2</sup>C interface specification, refer to the Philips Corporation document “*The I<sup>2</sup>C bus and how to use it.*” #939839340011, which can be obtained from the Philips Corporation.

In the I<sup>2</sup>C interface protocol each device has an address. When a master wishes to initiate a data transfer, it first transmits the address of the device that it wishes to “talk” to. All devices “listen” to see if this is their address. Within this address, a bit specifies if the master wishes to read-from / write-to the slave device. The master and slave are always in opposite modes (transmitter/receiver) of operation during a data transfer. That is they can be thought of operating in either of these two relations:

- Master-transmitter and Slave-receiver

- Slave-transmitter and Master-receiver

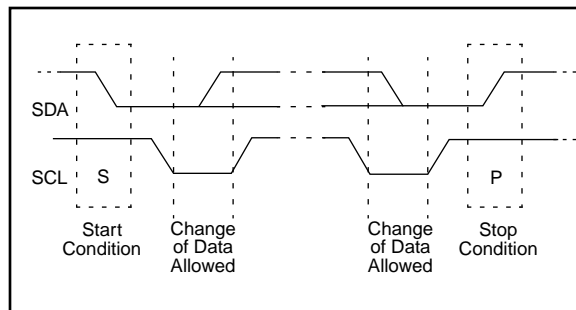
In both cases the master generates the clock signal.

The output stages of the clock (SCL) and data (SDA) lines must have an open-drain or open-collector in order to perform the wired-AND function of the bus. External pull-up resistors are used to ensure a high level when no device is pulling the line down. The number of devices that may be attached to the I<sup>2</sup>C bus is limited only by the maximum bus loading specification of 400 pF.

### 11.2.1 INITIATING AND TERMINATING DATA TRANSFER

During times of no data transfer (idle time), both the clock line (SCL) and the data line (SDA) are pulled high through the external pull-up resistors. The START and STOP conditions determine the start and stop of data transmission. The START condition is defined as a high to low transition of the SDA when the SCL is high. The STOP condition is defined as a low to high transition of the SDA when the SCL is high. Figure 11-7 shows the START and STOP conditions. The master generates these conditions for starting and terminating data transfer. Due to the definition of the START and STOP conditions, when data is being transmitted the SDA line can only change state when the SCL line is low.

**FIGURE 11-7: START AND STOP CONDITIONS**



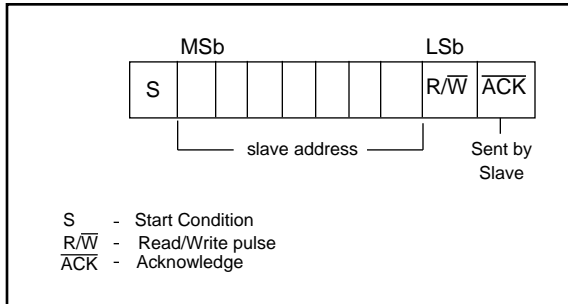
**TABLE 11-2: I<sup>2</sup>C BUS TERMINOLOGY**

Term	Description
Transmitter	The device that sends the data to the bus.
Receiver	The device that receives the data from the bus.
Master	The device which initiates the transfer, generates the clock and terminates the transfer.
Slave	The device addressed by a master.
Multi-master	More than one master device in a system. These masters can attempt to control the bus at the same time without corrupting the message.
Arbitration	Procedure that ensures that only one of the master devices will control the bus. This ensure that the transfer data does not get corrupted.
Synchronization	Procedure where the clock signals of two or more devices are synchronized.

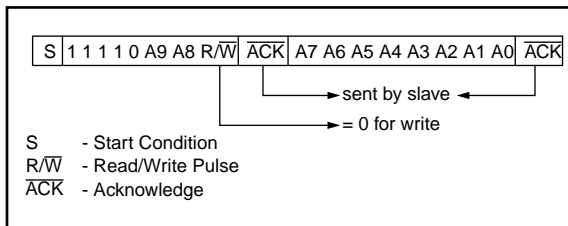
## 11.2.2 ADDRESSING I<sup>2</sup>C DEVICES

There are two address formats. The simplest is the 7-bit address format with a  $R/\overline{W}$  bit (Figure 11-8). The more complex is the 10-bit address with a  $R/\overline{W}$  bit (Figure 11-9). For 10-bit address format, two bytes must be transmitted with the first five bits specifying this to be a 10-bit address.

**FIGURE 11-8: 7-BIT ADDRESS FORMAT**



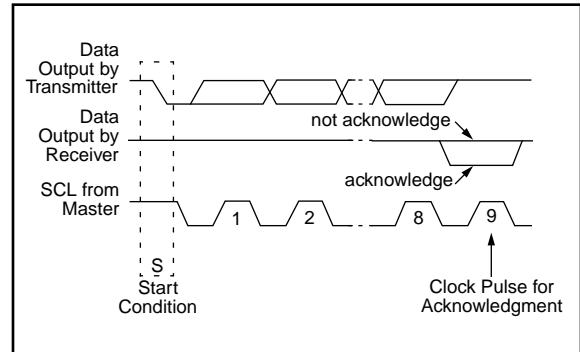
**FIGURE 11-9: I<sup>2</sup>C 10-BIT ADDRESS FORMAT**



## 11.2.3 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 acknowledgment bit ( $\overline{ACK}$ ) (Figure 11-10). When a slave-receiver doesn't acknowledge the slave address or received data, the master must abort the transfer. The slave must leave the SDA line high so that the master can generate the STOP condition (Figure 11-7).

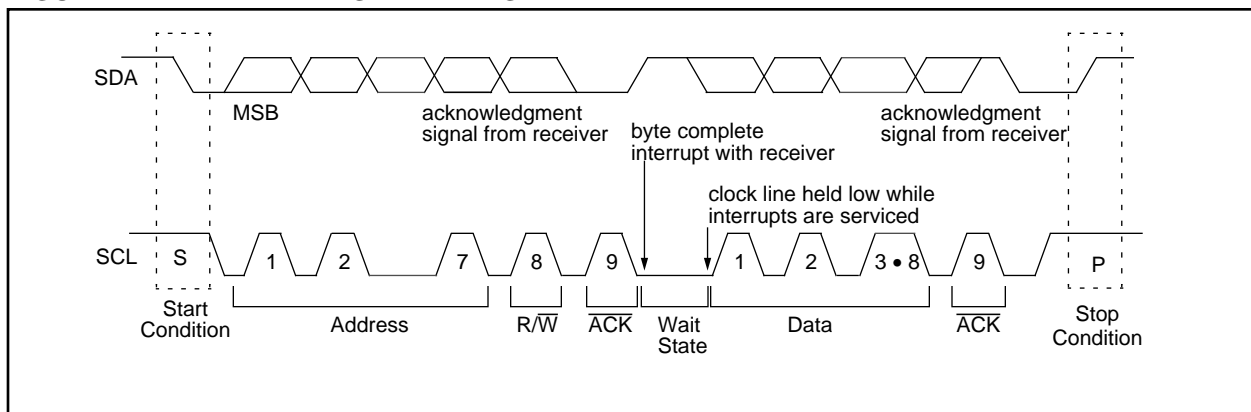
**FIGURE 11-10: SLAVE-RECEIVER ACKNOWLEDGE**



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 11-11. 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 11-11: DATA TRANSFER WAIT STATE**



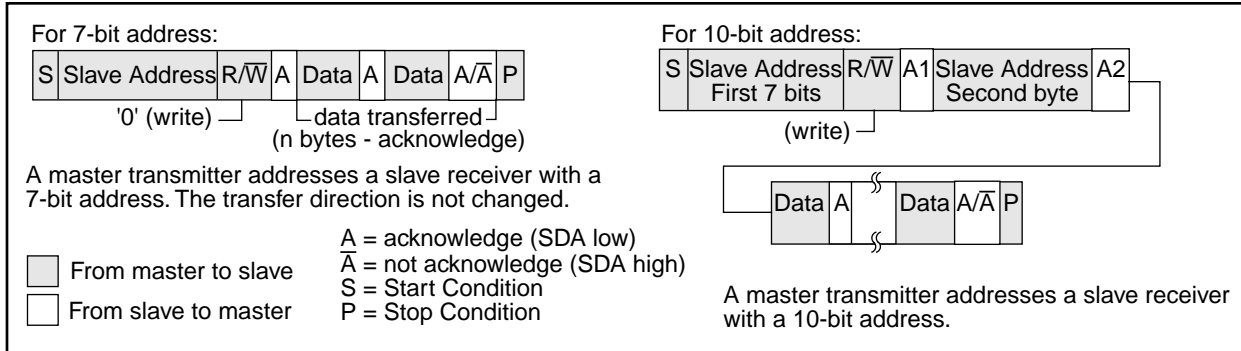
# PIC16C6X

Figure 11-12 and Figure 11-13 show Master-transmitter and Master-receiver data transfer sequences.

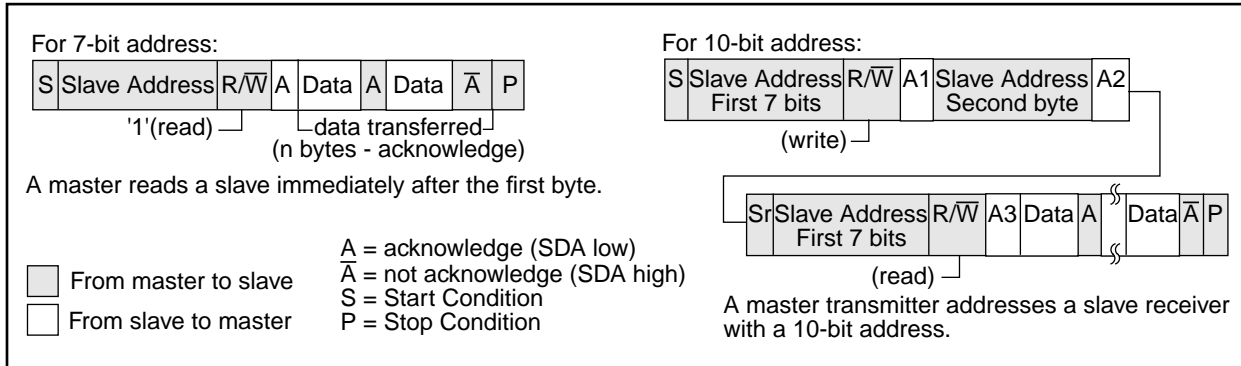
When a master does not wish to relinquish the bus (by generating a STOP condition), a repeated START condition (Sr) must be generated. This condition is identical to the start condition (S) (SDA goes high-to-low while SCL

is high), but occurs after a data transfer acknowledge pulse (not the bus-free state). This allows a master to send "commands" to the slave and then receive the requested information or to address a different slave device. This sequence is shown in Figure 11-14.

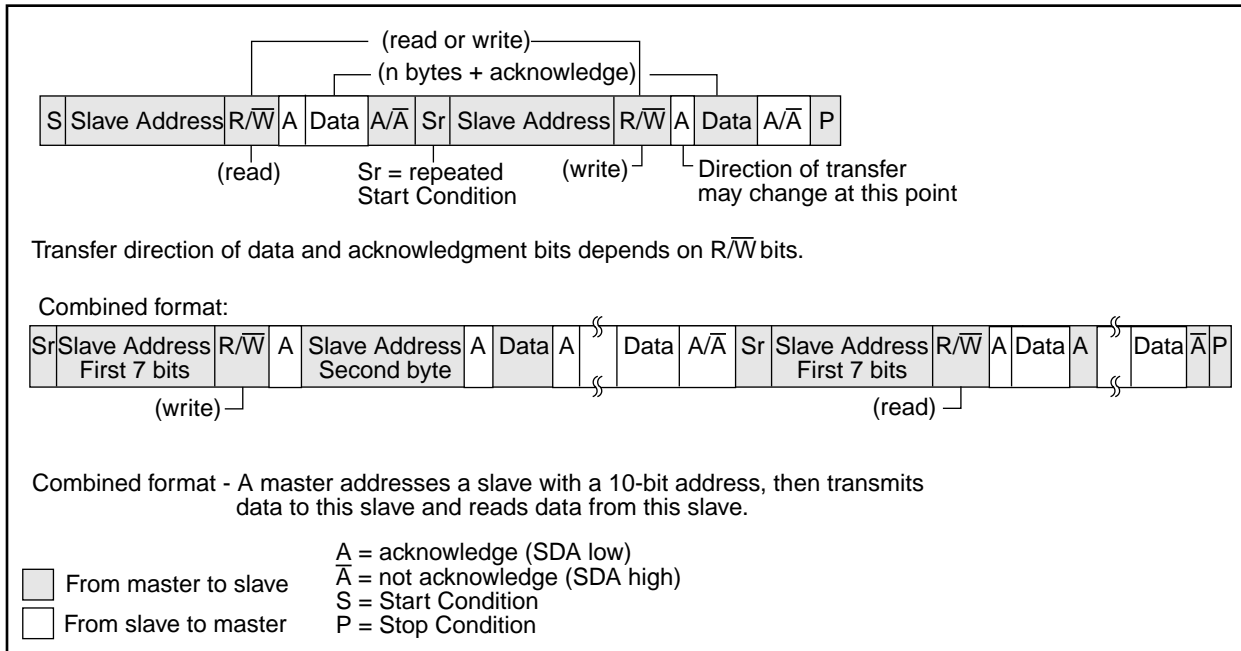
**FIGURE 11-12: MASTER-TRANSMITTER SEQUENCE**



**FIGURE 11-13: MASTER-RECEIVER SEQUENCE**



**FIGURE 11-14: COMBINED FORMAT**



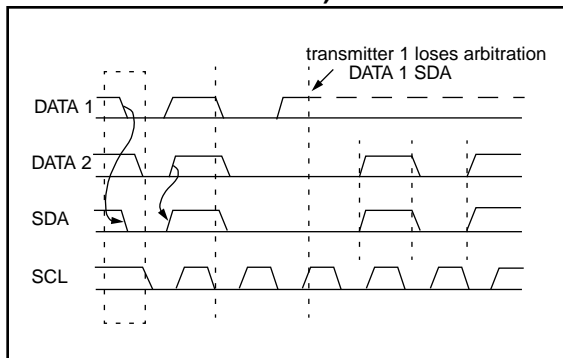
## 11.2.4 MULTI-MASTER

The I<sup>2</sup>C protocol allows a system to have more than one master. This is called multi-master. When two or more masters try to transfer data at the same time, arbitration and synchronization occur.

### 11.2.4.1 ARBITRATION

Arbitration takes place on the SDA line, while the SCL line is high. The master which transmits a high when the other master transmits a low loses arbitration (Figure 11-15), and turns off its data output stage. A master which lost arbitration can generate clock pulses until the end of the data byte where it lost arbitration. When the master devices are addressing the same device, arbitration continues into the data.

**FIGURE 11-15: MULTI-MASTER ARBITRATION (TWO MASTERS)**



Masters that also incorporate the slave function, and have lost arbitration must immediately switch over to slave-receiver mode. This is because the winning master-transmitter may be addressing it.

Arbitration is not allowed between:

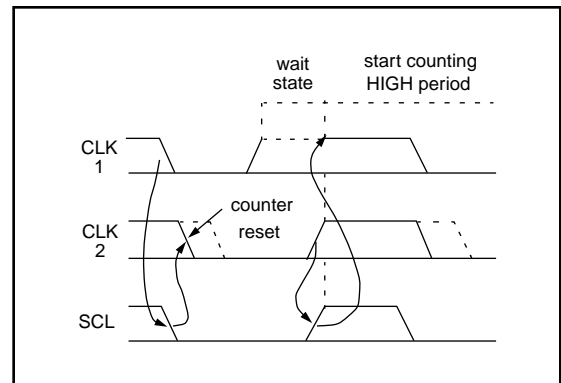
- A repeated START condition
- A STOP condition and a data bit
- A repeated START condition and a STOP condition

Care needs to be taken to ensure that these conditions do not occur.

### 11.2.4.2 Clock Synchronization

Clock synchronization occurs after the devices have started arbitration. This is performed using a wired-AND connection to the SCL line. A high to low transition on the SCL line causes the concerned devices to start counting off their low period. Once a device clock has gone low, it will hold the SCL line low until its SCL high state is reached. The low to high transition of this clock may not change the state of the SCL line, if another device clock is still within its low period. The SCL line is held low by the device with the longest low period. Devices with shorter low periods enter a high wait-state, until the SCL line comes high. When the SCL line comes high, all devices start counting off their high periods. The first device to complete its high period will pull the SCL line low. The SCL line high time is determined by the device with the shortest high period, Figure 11-16.

**FIGURE 11-16: CLOCK SYNCHRONIZATION**

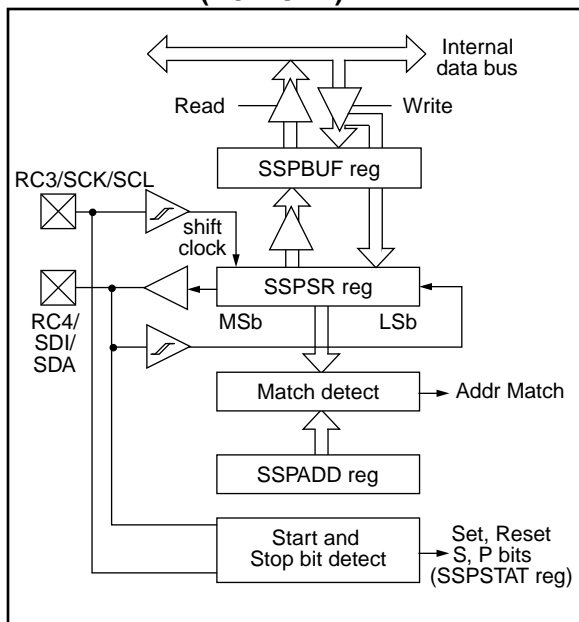


## 11.3 SSP I<sup>2</sup>C Operation

Applicable Devices									
61	62	62A	R62	63	64	64A	R64	65	65A

The SSP module in I<sup>2</sup>C mode fully implements all slave functions (except general call support), and provides interrupts on start and stop bits in hardware to facilitate software implementations of the master functions. The SSP module implements the standard and fast mode specifications as well as 7-bit and 10-bit addressing. Two pins are used for data transfer. These are the RC3/SCK/SCL pin, which is the clock (SCL), and the RC4/SDI/SDA pin, which is the data (SDA). The user must configure these pins as inputs or outputs through the TRISC<4:3> bits. The SSP module functions are enabled by setting SSP Enable bit SSPEN (SSPCON<5>).

**FIGURE 11-17: SSP BLOCK DIAGRAM (I<sup>2</sup>C MODE)**



The SSP module has five registers for I<sup>2</sup>C operation. These are the:

- SSP Control Register (SSPCON)
- SSP Status Register (SSPSTAT)
- Serial Receive/Transmit Buffer (SSPBUF)
- SSP Shift Register (SSPSR) - Not directly accessible
- SSP Address Register (SSPADD)

The SSPCON register allows control of the I<sup>2</sup>C operation. Four mode selection bits (SSPCON<3:0>) allow one of the following I<sup>2</sup>C modes to be selected:

- I<sup>2</sup>C Slave mode (7-bit address)
- I<sup>2</sup>C Slave mode (10-bit address)
- I<sup>2</sup>C Slave mode (7-bit address), with start and stop bit interrupts enabled
- I<sup>2</sup>C Slave mode (10-bit address), with start and stop bit interrupts enabled
- I<sup>2</sup>C start and stop bit interrupts enabled, slave is idle

Selection of any I<sup>2</sup>C mode, with enable bit SSPEN set, forces the SCL and SDA pins to be open drains, provided these pins are programmed to inputs by setting the appropriate TRISC bits.

The SSPSTAT register gives the status of the data transfer. This information includes detection of a START or STOP bit, specifies if the received byte was data or address, if the next byte is the completion of 10-bit address, and if this will be a read or write data transfer. The SSPSTAT register is read only.

The SSPBUF is the register to which transfer data is written-to or read-from. The SSPSR register shifts the data in or out of the device. In receive operations, the SSPBUF and SSPSR create a doubled buffered receiver. This allows reception of the next byte to begin before reading the last byte of received data. When the complete byte is received, it is transferred to the SSPBUF register and flag bit SSPIF is set. If another complete byte is received before the SSPBUF register is read, a receiver overflow has occurred and overflow detect bit SSPOV (SSPCON<6>) is set, and the byte in the SSPSR is lost.

The SSPADD register holds the slave address. In 10-bit mode, the user needs to write the high byte of the address (1111 0 A9 A8 0). Following the high byte address match, the low byte of the address needs to be loaded (A7:A0).



## 11.3.1 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 ( $\overline{ACK}$ ) pulse, and then load the SSPBUF register with the received value in the SSPSR register.

There are certain conditions that will cause the SSP module not to give this  $\overline{ACK}$  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 register, but interrupt flag bit SSPIF (PIR1<3>) is set. Table 11-3 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 overflow bit SSPOV is cleared through software.

The SCL clock input must have a minimum high and low time for proper operation. The high and low times of the I<sup>2</sup>C specification as well as the requirement of the SSP module is shown in timing parameter #100 and parameter #101.

### 11.3.1.1 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 bits BF and SSPOV are clear, the following events occur:

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

In 10-bit address mode, two address bytes need to be received by the slave (Figure 11-9). 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 are 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 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 11-3: DATA TRANSFER RECEIVED BYTE ACTIONS**

Status Bits as Data Transfer is Received		SSPSR → SSPBUF	Generate $\overline{ACK}$ 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

# PIC16C6X

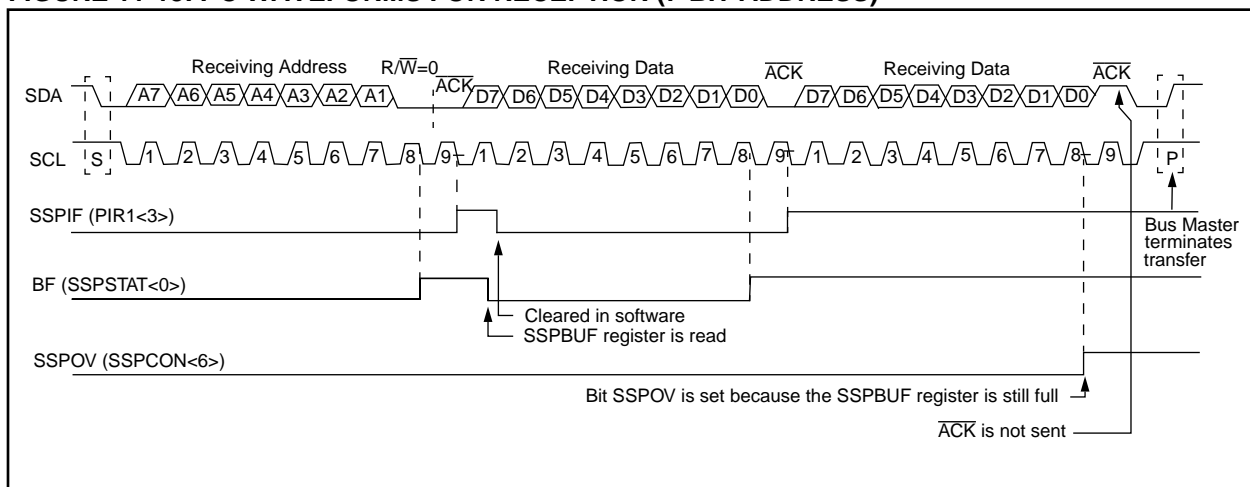
## 11.3.1.2 RECEPTION

When the  $R/\overline{W}$  bit of the address byte is clear and an address match occurs, bit  $R/\overline{W}$  (SSPSTAT<2>) is cleared. The received address is loaded into the SSPBUF register.

When the address byte overflow condition exists then no acknowledge ( $\overline{ACK}$ ) pulse is given. An overflow condition is defined as either bit BF (SSPSTAT<0>) is set or bit SSPOV (SSPCON<6>) is set.

A SSP interrupt is generated for each data transfer byte. Flag bit SSPIF (PIR1<3>) must be cleared in software, and the SSPSTAT register is used to determine the status of the byte.

**FIGURE 11-18: I<sup>2</sup>C WAVEFORMS FOR RECEPTION (7-BIT ADDRESS)**



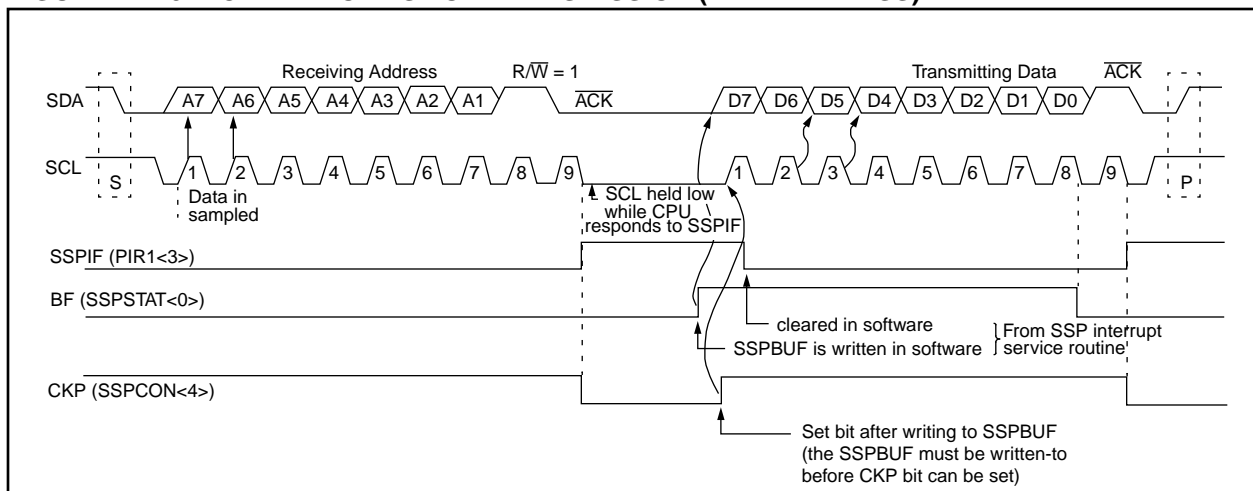
## 11.3.1.3 TRANSMISSION

When the  $R/\overline{W}$  bit of the incoming address byte is set and an address match occurs, bit  $R/\overline{W}$  (SSPSTAT<2>) is set. The received address is loaded into the SSPBUF register. The  $\overline{ACK}$  pulse will be sent on the ninth bit, and the SCL pin is held low. The transmit data must be loaded into the SSPBUF register, which also loads the SSPSR register. Then the SCL pin should be enabled by setting bit CKP (SSPCON<4>). The master must monitor the SCL pin prior to asserting another clock pulse. The slave devices may be holding off the master by stretching the clock. The eight data bits are shifted out on the falling edge of the SCL input. This ensures that the SDA signal is valid during the SCL high time (Figure 11-19).

An SSP interrupt is generated for each data transfer byte. Flag bit SSPIF must be cleared in software, and the SSPSTAT register is used to determine the status of the byte. Flag bit SSPIF is set on the falling edge of the ninth clock pulse.

As a slave-transmitter, the  $\overline{ACK}$  pulse from the master-receiver is latched on the rising edge of the ninth SCL input pulse. If the SDA line was high (not  $\overline{ACK}$ ), then the data transfer is complete. When the  $\overline{ACK}$  is latched by the slave, the slave logic is reset and the slave then monitors for another occurrence of the START bit. If the SDA line was low ( $\overline{ACK}$ ), the transmit data must be loaded into the SSPBUF register, which also loads the SSPSR register. Then the SCL pin should be enabled by setting bit CKP.

**FIGURE 11-19: I<sup>2</sup>C WAVEFORMS FOR TRANSMISSION (7-BIT ADDRESS)**



# PIC16C6X

## 11.3.2 MASTER MODE

Master mode of operation is supported by interrupt generation on the detection of the START and STOP conditions. The STOP (P) and START (S) bits are cleared by a reset or when the SSP module is disabled. Control of the I<sup>2</sup>C 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 the SSP Interrupt Flag bit SSPIF to be set (SSP Interrupt occurs 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.

## 11.3.3 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. Control of the I<sup>2</sup>C bus may be taken when bit P (SSPSTAT<4>) is set, or the bus is idle with both bits S and P 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  $\overline{ACK}$  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 11-4: REGISTERS ASSOCIATED WITH I<sup>2</sup>C 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
0B/8Bh	INTCON	GIE	PEIE	TOIE	INTE	RBIE	TOIF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	PSPIF <sup>(2)</sup>	<sup>(3)</sup>	RCIF <sup>(1)</sup>	TXIF <sup>(1)</sup>	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
8Ch	PIE1	PSPIE <sup>(2)</sup>	<sup>(3)</sup>	RCIE <sup>(1)</sup>	TXIE <sup>(1)</sup>	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 <sup>2</sup> C mode) Address Register								0000 0000	0000 0000
14h	SSPCON	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	0000 0000
87h	TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	1111 1111	1111 1111
94h	SSPSTAT	—	—	D/ $\overline{A}$	P	S	R/ $\overline{W}$	UA	BF	--00 0000	--00 0000

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

Shaded cells are not used by the SSP module in I<sup>2</sup>C mode.

- Note 1: These bits are associated with the USART and are implemented on the PIC16C63/65/65A only.  
 2: PSPIF and PSPIE are reserved on the PIC16C62/62A/R62/63, always maintain these bits clear.  
 3: PIR1<6> and PIE1<6> are reserved, always maintain these bits clear.

**FIGURE 11-20: OPERATION OF THE I<sup>2</sup>C MODULE IN IDLE\_MODE, RCV\_MODE OR XMIT\_MODE**

<b>IDLE_MODE (7-bit):</b> if (Addr_match)		{ Set interrupt; if (R/W = 1) { Send $\overline{ACK}$ = 0; set XMIT_MODE; } else if (R/W = 0) set RCV_MODE; }
<b>RCV_MODE:</b> if ((SSPBUF=Full) OR (SSPOV = 1))		{ Set SSPOV; Do not acknowledge; } else { transfer SSPSR → SSPBUF; send $\overline{ACK}$ = 0; } Receive 8-bits in SSPSR; Set interrupt;
<b>XMIT_MODE:</b> While ((SSPBUF = Empty) AND (CKP=0)) Hold SCL Low; Send byte; Set interrupt;		if ( $\overline{ACK}$ Received = 1) { End of transmission; Go back to IDLE_MODE; } else if ( $\overline{ACK}$ Received = 0) Go back to XMIT_MODE;
<b>IDLE_MODE (10-Bit):</b> If (High_byte_addr_match AND (R/W = 0))		{ PRIOR_ADDR_MATCH = FALSE; Set interrupt; if ((SSPBUF = Full) OR ((SSPOV = 1)) { Set SSPOV; Do not acknowledge; } else { Set UA = 1; Send $\overline{ACK}$ = 0; While (SSPADD not updated) Hold SCL low; Clear UA = 0; Receive Low_addr_byte; Set interrupt; Set UA = 1; If (Low_byte_addr_match) { PRIOR_ADDR_MATCH = TRUE; Send $\overline{ACK}$ = 0; while (SSPADD not updated) Hold SCL low; Clear UA = 0; Set RCV_MODE; } } } else if (High_byte_addr_match AND (R/W = 1)) { if (PRIOR_ADDR_MATCH) { send $\overline{ACK}$ = 0; set XMIT_MODE; } else PRIOR_ADDR_MATCH = FALSE; }

# PIC16C6X

---

NOTES:

## 12.0 UNIVERSAL SYNCHRONOUS ASYNCHRONOUS RECEIVER TRANSMITTER (USART) MODULE

Applicable Devices									
61	62	62A	R62	63	64	64A	R64	65	65A

The Universal Synchronous Asynchronous Receiver Transmitter (USART) module is one of the two serial I/O modules. (USART is also known as a Serial Communications Interface or SCI) The USART can be configured as a full duplex asynchronous system that can communicate with peripheral devices such as CRT terminals and personal computers, or it can be configured as a half duplex synchronous system that can communicate with peripheral devices such as A/D or D/A integrated circuits, Serial EEPROMs etc.

The USART can be configured in the following modes:

- Asynchronous (full duplex)
- Synchronous - Master (half duplex)
- Synchronous - Slave (half duplex)

Bit SPEN (RCSTA<7>) and bits TRISC<7:6> have to be set in order to configure pins RC6/TX/CK and RC7/RX/DT as the Universal Synchronous Asynchronous Receiver Transmitter.

**FIGURE 12-1: TXSTA: TRANSMIT STATUS AND CONTROL REGISTER (ADDRESS 98h)**

R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R-1	R/W-0
CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D
bit7							bit0
<div> <div>R = Readable bit</div> <div>W = Writable bit</div> <div>U = Unimplemented bit, read as '0'</div> <div>- n = Value at POR reset</div> </div>							
bit 7:	<b>CSRC:</b> Clock Source Select bit <u>Asynchronous mode</u> Don't care <u>Synchronous mode</u> 1 = Master mode (Clock generated internally from BRG) 0 = Slave mode (Clock from external source)						
bit 6:	<b>TX9:</b> 9-bit Transmit Enable bit 1 = Selects 9-bit transmission 0 = Selects 8-bit transmission						
bit 5:	<b>TXEN:</b> Transmit Enable bit 1 = Transmit enabled 0 = Transmit disabled Note: SREN/CREN overrides TXEN in SYNC mode.						
bit 4:	<b>SYNC:</b> USART Mode Select bit 1 = Synchronous mode 0 = Asynchronous mode						
bit 3:	<b>Unimplemented:</b> Read as '0'						
bit 2:	<b>BRGH:</b> High Baud Rate Select bit <u>Asynchronous mode</u> 1 = High speed <div> <b>Note:</b> At the time of this printing, the asynchronous high speed mode (BRGH is set) may experience a high rate of receive errors. It is recommended to have the BRGH bit cleared. If you desire a higher baud rate than BRGH=0 can support, refer to the device errata for additional information. </div> 0 = Low speed <u>Synchronous mode</u> Unused in this mode						
bit 1:	<b>TRMT:</b> Transmit Shift Register Status bit 1 = TSR empty 0 = TSR full						
bit 0:	<b>TX9D:</b> 9th bit of transmit data. Can be parity bit.						

# 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)  
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)



## 12.1 USART Baud Rate Generator (BRG)

Applicable Devices									
61	62	62A	R62	63	64	64A	R64	65	65A

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

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

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

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

### EXAMPLE 12-1: CALCULATING BAUD RATE ERROR

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

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

**Note:** At the time of this printing, the asynchronous high speed mode (BRGH is set) may experience a high rate of receive errors. It is recommended to have the BRGH bit cleared. If you desire a higher baud rate than BRGH=0 can support, refer to the device errata for additional information.

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

**TABLE 12-1: BAUD RATE FORMULA**

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

X = value in SPBRG (0 to 255)

**TABLE 12-2: REGISTERS ASSOCIATED WITH BAUD RATE GENERATOR**

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR BOR	Value on all other Resets
98h	TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010
18h	RCSTA	SPEN	RX9	SREN	CREN	—	FERR	OERR	RX9D	0000 -00x	0000 -00x
99h	SPBRG	Baud Rate Generator Register								0000 0000	0000 0000

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

# PIC16C6X

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

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

BAUD RATE (K)	FOSC = 5.0688 MHz			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)
0.3	NA	-	-	NA	-	-	NA	-	-	0.303	+1.14	26
1.2	NA	-	-	NA	-	-	1.202	+0.16	207	1.170	-2.48	6
2.4	NA	-	-	NA	-	-	2.404	+0.16	103	NA	-	-
9.6	9.6	0	131	9.622	+0.23	92	9.615	+0.16	25	NA	-	-
19.2	19.2	0	65	19.04	-0.83	46	19.24	+0.16	12	NA	-	-
76.8	79.2	+3.13	15	74.57	-2.90	11	83.34	+8.51	2	NA	-	-
96	97.48	+1.54	12	99.43	+3.57	8	NA	-	-	NA	-	-
300	316.8	+5.60	3	298.3	-0.57	2	NA	-	-	NA	-	-
500	NA	-	-	NA	-	-	NA	-	-	NA	-	-
HIGH	1267	-	0	894.9	-	0	250	-	0	8.192	-	0
LOW	4.950	-	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			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)
0.3	0.31	+3.13	255	0.301	+0.23	185	0.300	+0.16	51	0.256	-14.67	1
1.2	1.2	0	65	1.190	-0.83	46	1.202	+0.16	12	NA	-	-
2.4	2.4	0	32	2.432	+1.32	22	2.232	-6.99	6	NA	-	-
9.6	9.9	+3.13	7	9.322	-2.90	5	NA	-	-	NA	-	-
19.2	19.8	+3.13	3	18.64	-2.90	2	NA	-	-	NA	-	-
76.8	79.2	+3.13	0	NA	-	-	NA	-	-	NA	-	-
96	NA	-	-	NA	-	-	NA	-	-	NA	-	-
300	NA	-	-	NA	-	-	NA	-	-	NA	-	-
500	NA	-	-	NA	-	-	NA	-	-	NA	-	-
HIGH	79.2	-	0	55.93	-	0	15.63	-	0	0.512	-	0
LOW	0.3094	-	255	0.2185	-	255	0.0610	-	255	0.0020	-	255

**TABLE 12-5: BAUD RATES FOR ASYNCHRONOUS MODE (BRGH = 1)**

BAUD RATE (K)	FOSC = 20 MHz			16 MHz			10 MHz			7.16 MHz		
	KBAUD	%ERROR	SPBRG value (decimal)	KBAUD	%ERROR	SPBRG value (decimal)	KBAUD	%ERROR	SPBRG value (decimal)	KBAUD	%ERROR	SPBRG value (decimal)
9.6	9.615	+0.16	129	9.615	+0.16	103	9.615	+0.16	64	9.520	-0.83	46
19.2	19.230	+0.16	64	19.230	+0.16	51	18.939	-1.36	32	19.454	+1.32	22
38.4	37.878	-1.36	32	38.461	+0.16	25	39.062	+1.7	15	37.286	-2.90	11
57.6	56.818	-1.36	21	58.823	+2.12	16	56.818	-1.36	10	55.930	-2.90	7
115.2	113.636	-1.36	10	111.111	-3.55	8	125	+8.51	4	111.860	-2.90	3
250	250	0	4	250	0	3	NA	-	-	NA	-	-
625	625	0	1	NA	-	-	625	0	0	NA	-	-
1250	1250	0	0	NA	-	-	NA	-	-	NA	-	-

BAUD RATE (K)	FOSC = 5.068 MHz			3.579 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)
9.6	9.6	0	32	9.727	+1.32	22	8.928	-6.99	6	NA	-	-
19.2	18.645	-2.94	16	18.643	-2.90	11	20.833	+8.51	2	NA	-	-
38.4	39.6	+3.12	7	37.286	-2.90	5	31.25	-18.61	1	NA	-	-
57.6	52.8	-8.33	5	55.930	-2.90	3	62.5	+8.51	0	NA	-	-
115.2	105.6	-8.33	2	111.860	-2.90	1	NA	-	-	NA	-	-
250	NA	-	-	223.721	-10.51	0	NA	-	-	NA	-	-
625	NA	-	-	NA	-	-	NA	-	-	NA	-	-
1250	NA	-	-	NA	-	-	NA	-	-	NA	-	-

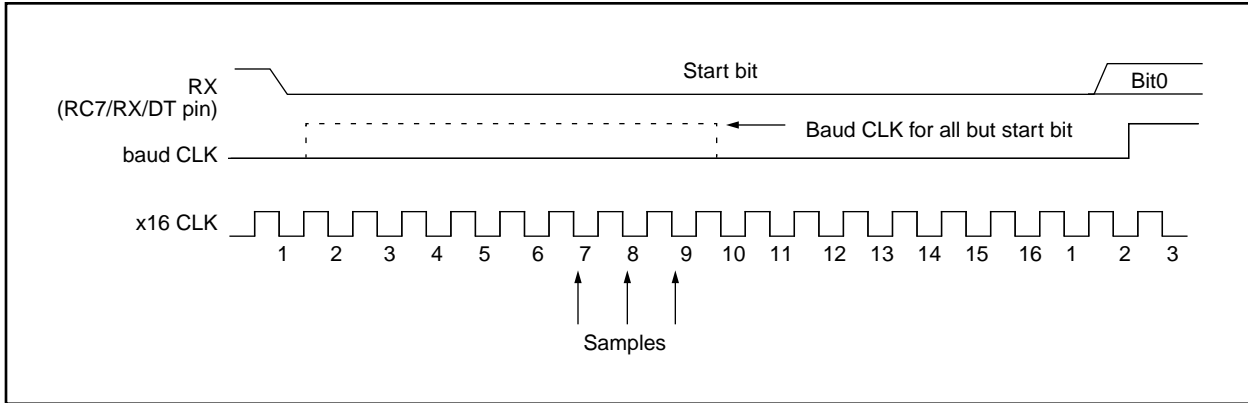
# PIC16C6X

## 12.1.1 SAMPLING

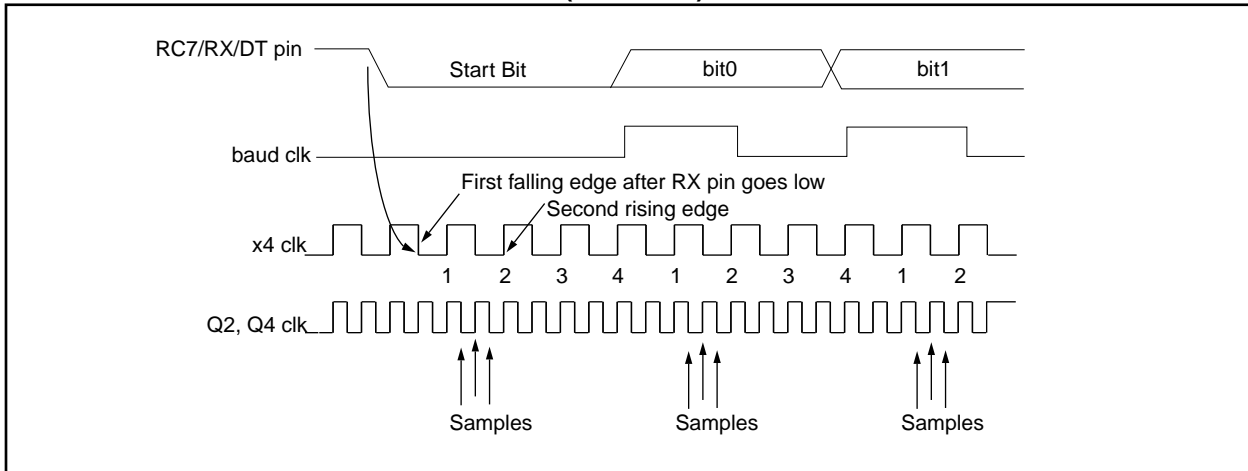
The data on the RC7/RX/DT pin is sampled three times by a majority detect circuit to determine if a high or a low level is present at the RX pin. If bit BRGH (TXSTA<2>) is clear (i.e., at the low baud rates), the sampling is done on the seventh, eighth and ninth falling edges of a x16 clock (Figure 12-3). If bit BRGH is

set (i.e., at the high baud rates), the sampling is done on the 3 clock edges preceding the second rising edge after the first falling edge of a x4 clock (Figure 12-4 and Figure 12-5).

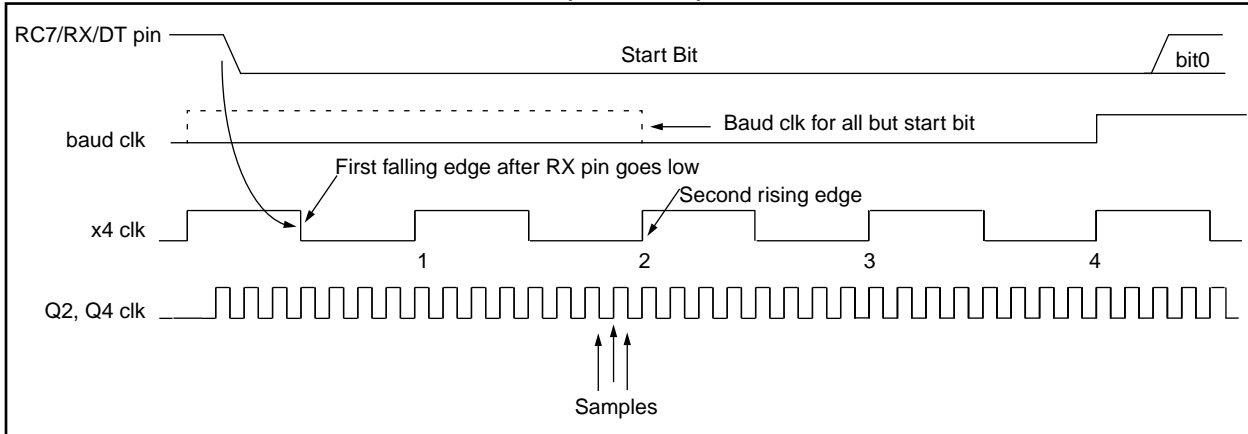
**FIGURE 12-3: RX PIN SAMPLING SCHEME (BRGH = 0)**



**FIGURE 12-4: RX PIN SAMPLING SCHEME (BRGH = 1)**



**FIGURE 12-5: RX PIN SAMPLING SCHEME (BRGH = 1)**



## 12.2 USART Asynchronous Mode

### Applicable Devices

61	62	62A	R62	63	64	64A	R64	65	65A
----	----	-----	-----	----	----	-----	-----	----	-----

In this mode, the USART uses standard nonreturn-to-zero (NRZ) format (one start bit, eight or nine data bits and one stop bit). The most common data format is 8-bits. An on-chip dedicated 8-bit baud rate generator can be used to derive standard baud rate frequencies from the oscillator. The USART transmits and receives the LSb first. The USART's transmitter and receiver are functionally independent but use the same data format and baud rate. The baud rate generator produces a clock either x16 or x64 of the bit shift rate, depending on bit BRGH (TXSTA<2>). Parity is not supported by the hardware, but can be implemented in software (and stored as the ninth data bit). Asynchronous mode is stopped during SLEEP.

Asynchronous mode is selected by clearing bit SYNC (TXSTA<4>).

The USART Asynchronous module consists of the following important elements:

- Baud Rate Generator
- Sampling Circuit
- Asynchronous Transmitter
- Asynchronous Receiver

### 12.2.1 USART ASYNCHRONOUS TRANSMITTER

The USART transmitter block diagram is shown in Figure 12-6. The heart of the transmitter is the transmit (serial) shift register (TSR). The shift register obtains its data from the read/write transmit buffer, TXREG. The TXREG register is loaded with data in software. The TSR register is not loaded until the STOP bit has been transmitted from the previous load. As soon as the STOP bit is transmitted, the TSR is loaded with new data from the TXREG (if available). Once the TXREG register transfers the data to the TSR register (occurs in one Tcy) the TXREG register is empty and flag bit TXIF

(PIR1<4>) is set. This interrupt is enabled/disabled by setting/clearing enable bit TXIE (PIE1<4>). Flag bit TXIF will be set regardless of the state of enable bit TXIE and cannot be cleared in software. It will reset only when new data is loaded into the TXREG register. While flag bit TXIF indicates the status of the TXREG register, another bit, TRMT (TXSTA<1>) shows the status of the TSR register. Status bit TRMT is a read only bit which is set when the TSR register is empty. No interrupt logic is tied to this bit, so the user has to poll this bit in order to determine if the TSR register is empty.

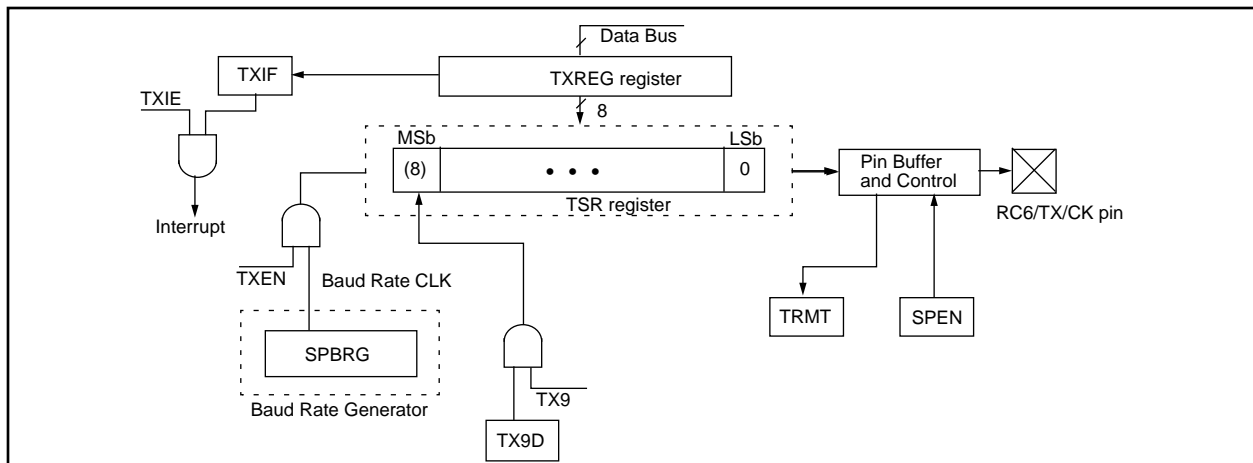
**Note 1:** The TSR register is not mapped in data memory so it is not available to the user.

**Note 2:** Flag bit TXIF is set when enable bit TXEN is set.

Transmission is enabled by setting enable bit TXEN (TXSTA<5>). The actual transmission will not occur until the TXREG register has been loaded with data and the baud rate generator (BRG) has produced a shift clock (Figure 12-6). The transmission can also be started by first loading the TXREG register and then setting enable bit TXEN. Normally when transmission is first started, the TSR register is empty, so a transfer to the TXREG register will result in an immediate transfer to TSR register resulting in an empty TXREG register. A back-to-back transfer is thus possible (Figure 12-8). Clearing enable bit TXEN during a transmission will cause the transmission to be aborted and will reset the transmitter. As a result the RC6/TX/CK pin will revert to hi-impedance.

In order to select 9-bit transmission, transmit bit TX9 (TXSTA<6>) should be set and the ninth bit should be written to bit TX9D (TXSTA<0>). The ninth bit must be written before writing the 8-bit data to the TXREG register. This is because a data write to the TXREG register can result in an immediate transfer of the data to the TSR register (if the TSR is empty). In such a case, an incorrect ninth data bit maybe loaded in the TSR register.

**FIGURE 12-6: USART TRANSMIT BLOCK DIAGRAM**

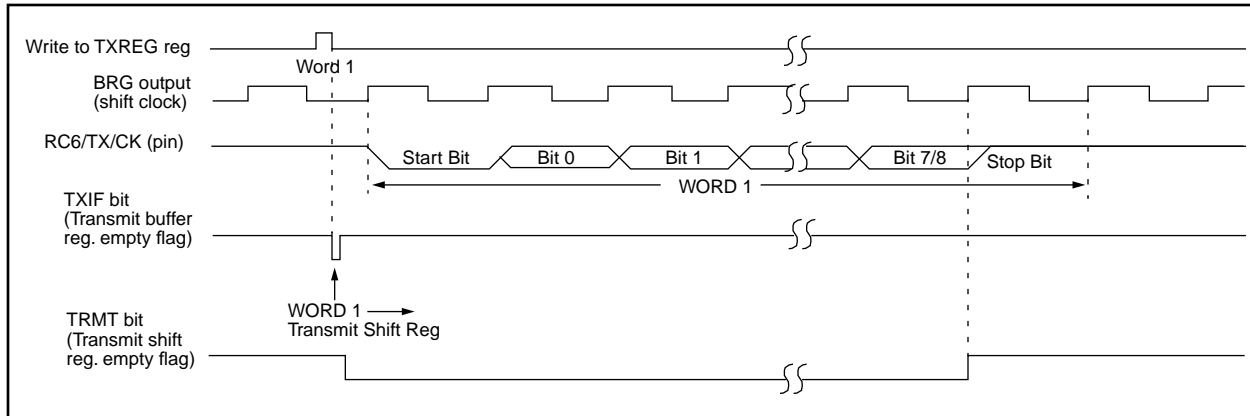


# PIC16C6X

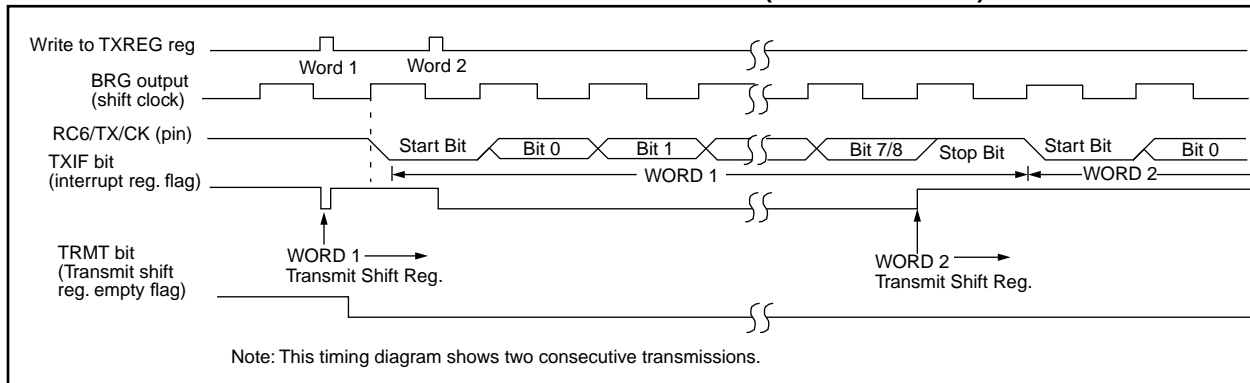
Steps to follow when setting up an Asynchronous Transmission:

1. Initialize the SPBRG register for the appropriate baud rate. If a high speed baud rate is desired, then set bit BRGH. (Section 12.1).
2. Enable the asynchronous serial port by clearing bit SYNC and setting bit SPEN.
3. If interrupts are desired, then set enable bit TXIE.
4. If 9-bit transmission is desired, then set transmit bit TX9.
5. Enable the transmission by setting bit TXEN, which will also set bit TXIF.
6. If 9-bit transmission is selected, the ninth bit should be loaded in bit TX9D.
7. Load data to the TXREG register (starts transmission).

**FIGURE 12-7: ASYNCHRONOUS MASTER TRANSMISSION**



**FIGURE 12-8: ASYNCHRONOUS MASTER TRANSMISSION (BACK TO BACK)**



**TABLE 12-6: REGISTERS ASSOCIATED WITH ASYNCHRONOUS TRANSMISSION**

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
0Ch	PIR1	PSPIF <sup>(1)</sup>	(2)	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
18h	RCSTA	SPEN	RX9	SREN	CREN	—	FERR	OERR	RX9D	0000 -00x	0000 -00x
19h	TXREG	TX7	TX6	TX5	TX4	TX3	TX2	TX1	TX0	0000 0000	0000 0000
8Ch	PIE1	PSPIE <sup>(1)</sup>	(2)	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
98h	TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010
99h	SPBRG	Baud Rate Generator Register								0000 0000	0000 0000

Legend: x = unknown, - = unimplemented locations read as '0'. Shaded cells are not used for Asynchronous Transmission.

Note 1: PSPIF and PSPIE are reserved on the PIC16C63, always maintain these bits clear.

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

## 12.2.2 USART ASYNCHRONOUS RECEIVER

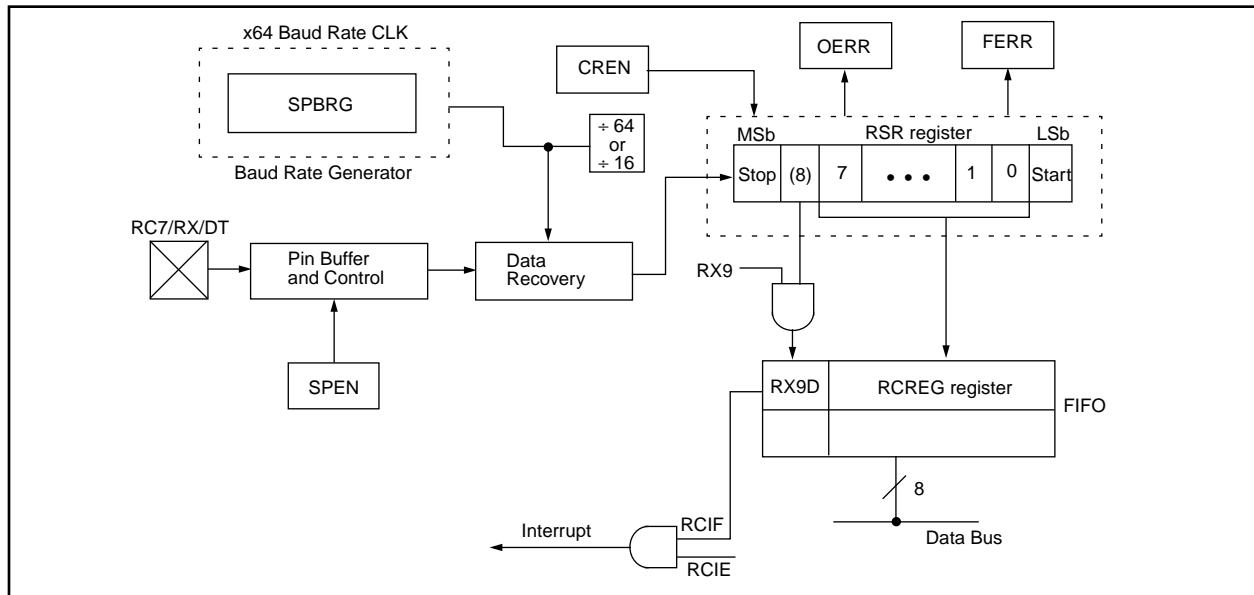
The receiver block diagram is shown in Figure 12-9. The data comes in the RC7/RX/DT pin and drives the data recovery block. The data recovery block is actually a high speed shifter operating at x16 times the baud rate, whereas the main receive serial shifter operates at the bit rate or at FOSC.

Once Asynchronous mode is selected, reception is enabled by setting bit CREN (RCSTA<4>).

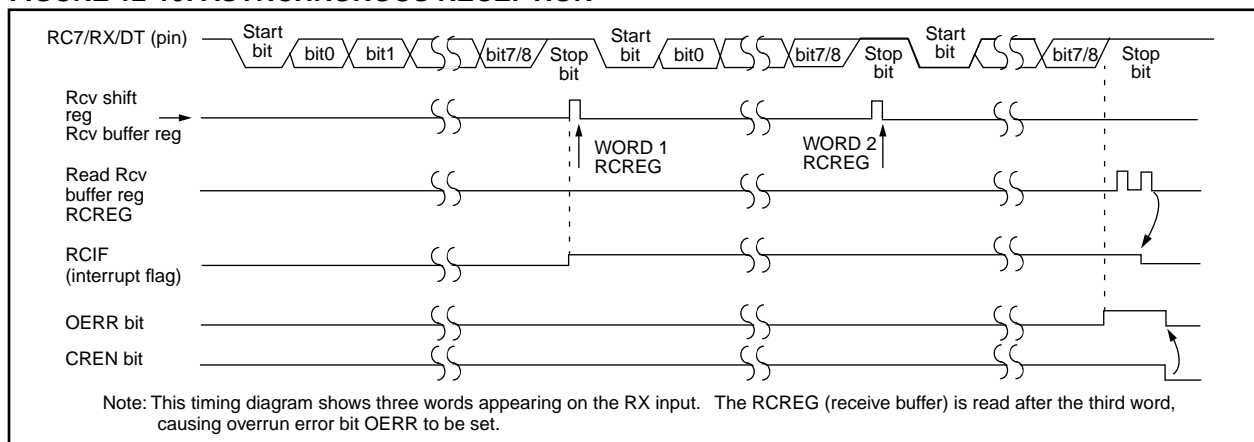
The heart of the receiver is the receive (serial) shift register (RSR). After sampling the STOP bit, the received data in the RSR is transferred to the RCREG register (if it is empty). If the transfer is complete, flag bit RCIF (PIR1<5>) is set. The actual interrupt can be enabled/disabled by setting/clearing enable bit RCIE (PIE1<5>). Flag bit RCIF is a read only bit which is reset by the hardware. It is cleared when the RCREG register has been read and is empty. The RCREG is double buffered register, i.e., it is a two deep FIFO. It is

possible for two bytes of data to be received and transferred to the RCREG FIFO and a third byte begin shifting to the RSR register. On the detection of the STOP bit of the third byte, if the RCREG is still full, then the overrun error bit, OERR (RCSTA<1>) will be set. The word in the RSR register will be lost. The RCREG register can be read twice to retrieve the two bytes in the FIFO. Overrun bit OERR has to be cleared in software. This is done by resetting the receive logic (CREN is cleared and then set). If bit OERR is set, transfers from the RSR register to the RCREG register are inhibited, so it is essential to clear overrun bit OERR if it is set. Framing error bit FERR (RCSTA<2>) is set if a stop bit is detected as clear. Error bit FERR and the 9th receive bit are buffered the same way as the receive data. Reading the RCREG register will load bits RX9D and FERR with new values. Therefore it is essential for the user to read the RCSTA register before reading RCREG in order not to lose the old FERR and RX9D information.

**FIGURE 12-9: USART RECEIVE BLOCK DIAGRAM**



**FIGURE 12-10: ASYNCHRONOUS RECEPTION**



# PIC16C6X

Steps to follow when setting up an Asynchronous Reception:

1. Initialize the SPBRG register for the appropriate baud rate. If a high speed baud rate is desired, set bit BRGH (Section 12.1).
2. Enable the asynchronous serial port by clearing bit SYNC and setting bit SPEN.
3. If interrupts are desired, then set enable bit RCIE.
4. If 9-bit reception is desired, then set bit RX9.
5. Enable the reception by setting enable bit CREN.
6. Flag bit RCIF will be set when reception is complete, and an interrupt will be generated if enable bit RCIE was set.
7. Read the RCSTA register to get the ninth bit (if enabled) and determine if any error occurred during reception.
8. Read the 8-bit received data by reading the RCREG register.
9. If any error occurred, clear the error by clearing enable bit CREN.

**TABLE 12-7: REGISTERS ASSOCIATED WITH ASYNCHRONOUS RECEPTION**

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
0Ch	PIR1	PSPIF <sup>(1)</sup>	(2)	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
18h	RCSTA	SPEN	RX9	SREN	CREN	—	FERR	OERR	RX9D	0000 -00x	0000 -00x
1Ah	RCREG	RX7	RX6	RX5	RX4	RX3	RX2	RX1	RX0	0000 0000	0000 0000
8Ch	PIE1	PSPIE <sup>(1)</sup>	(2)	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
98h	TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010
99h	SPBRG	Baud Rate Generator Register								0000 0000	0000 0000

Legend: x = unknown, - = unimplemented locations read as '0'. Shaded cells are not used for Asynchronous Reception.

Note 1: PSPIE and PSPIF are reserved on the PIC16C63, always maintain these bits clear.

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



## 12.3 USART Synchronous Master Mode

Applicable Devices									
61	62	62A	R62	63	64	64A	R64	65	65A

In Synchronous Master mode the data is transmitted in a half-duplex manner i.e., transmission and reception do not occur at the same time. When transmitting data the reception is inhibited and vice versa. The synchronous mode is entered by setting bit SYNC (TXSTA<4>). In addition enable bit SPEN (RCSTA<7>) is set in order to configure the RC6 and RC7 I/O pins to CK (clock) and DT (data) lines respectively. The Master mode indicates that the processor transmits the master clock on the CK line. The Master mode is entered by setting bit CSRC (TXSTA<7>).

### 12.3.1 USART SYNCHRONOUS MASTER TRANSMISSION

The USART transmitter block diagram is shown in Figure 12-6. The heart of the transmitter is the transmit (serial) shift register (TSR). The shift register obtains its data from the read/write transmit buffer register, TXREG. The TXREG register is loaded with data in software. The TSR register is not loaded until the last bit has been transmitted from the previous load. As soon as the last bit is transmitted, the TSR register is loaded with new data from the TXREG register (if available). Once the TXREG register transfers the data to the TSR register (occurs in one Tcycle), the TXREG register is empty and interrupt flag bit TXIF (PIR1<4>) is set. This interrupt can be enabled/disabled by setting/clearing enable bit TXIE (PIE1<4>). Flag bit TXIF will be set regardless of the status of enable bit TXIE and cannot be cleared in software. It will clear only when new data is loaded into the TXREG register. While flag bit TXIF indicates the status of the TXREG register, another bit, TRMT (TXSTA<1>), shows the status of the TSR register. Status bit TRMT is a read only bit which is set when the TSR register is empty. No interrupt logic is tied to this bit, so the user has to poll this bit in order to determine if the TSR register is empty. The TSR register is not mapped in data memory so it is not available to the user.

Transmission is enabled by setting enable bit TXEN (TXSTA<5>). The actual transmission will not occur until the TXREG register has been loaded with data. The first data bit will be shifted out on the next available rising edge of the clock on the CK line. Data out is stable around the falling edge of the synchronous clock (Figure 12-11). The transmission can also be started by first loading the TXREG register and then setting enable bit TXEN. This is advantageous when slow baud rates are selected, since the BRG is kept in reset when bits TXEN, CREN, and SREN are clear. Setting enable bit TXEN will start the BRG, creating a shift clock immediately. Normally when transmission is first started, the TSR register is empty, so a transfer to the TXREG register will result in an immediate transfer to TSR resulting in an empty TXREG register. Back-to-back transfers are possible.

Clearing enable bit TXEN, during a transmission, will cause the transmission to be aborted and will reset the transmitter. The DT and CK pins will revert to hi-impedance. If, during a transmission, either bit CREN or bit SREN are set the transmission is aborted and the DT pin reverts to a hi-impedance state (for a reception). The CK pin will remain an output if bit CSRC is set (internal clock). The transmitter logic however, is not reset although it is disconnected from the pins. In order to reset the transmitter, the user has to clear enable bit TXEN. If enable bit SREN is set (to interrupt an on going transmission and receive a single word), then after the single word is received, enable bit SREN will be cleared, and the serial port will revert back to transmitting since enable bit TXEN is still set. The DT line will immediately switch from hi-impedance receive mode to transmit and start driving. To avoid this, enable bit TXEN should be cleared.

In order to select 9-bit transmission, bit TX9 (TXSTA<6>) should be set and the ninth bit should be written to bit TX9D (TXSTA<0>). The ninth bit must be written before writing the 8-bit data to the TXREG register. This is because a data write to the TXREG register can result in an immediate transfer of the data to the TSR register (if the TSR is empty). If the TSR register was empty and the TXREG register was written before writing the "new" TX9D, the "present" value of bit TX9D is loaded.

Steps to follow when setting up a Synchronous Master Transmission:

1. Initialize the SPBRG register for the appropriate baud rate (Section 12.1).
2. Enable the synchronous master serial port by setting bits SYNC, SPEN, and CSRC.
3. If interrupts are desired, then set enable bit TXIE.
4. If 9-bit transmission is desired, then set bit TX9.
5. Enable the transmission by setting enable bit TXEN.
6. If 9-bit transmission is selected, the ninth bit should be loaded in bit TX9D.
7. Start transmission by loading data to the TXREG register.

# PIC16C6X

**TABLE 12-8: REGISTERS ASSOCIATED WITH SYNCHRONOUS MASTER TRANSMISSION**

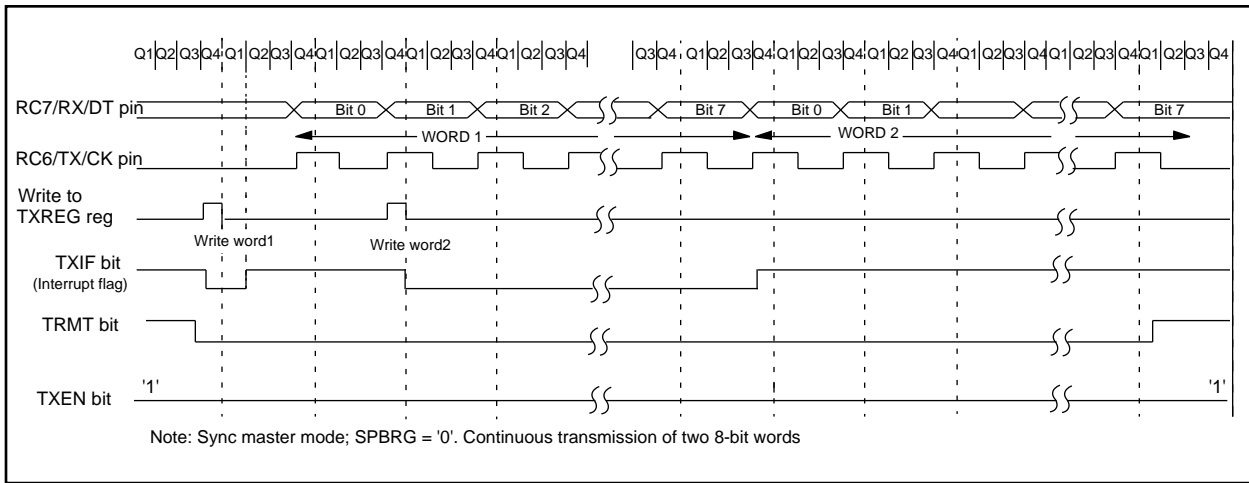
Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR BOR	Value on all other Resets
0Ch	PIR1	PSPIF <sup>(1)</sup>	(2)	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
18h	RCSTA	SPEN	RX9	SREN	CREN	—	FERR	OERR	RX9D	0000 -00x	0000 -00x
19h	TXREG	TX7	TX6	TX5	TX4	TX3	TX2	TX1	TX0	0000 0000	0000 0000
8Ch	PIE1	PSPIE <sup>(1)</sup>	(2)	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
98h	TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010
99h	SPBRG	Baud Rate Generator Register								0000 0000	0000 0000

Legend: x = unknown, - = unimplemented locations read as '0'. Shaded cells are not used for Synchronous Master Transmission.

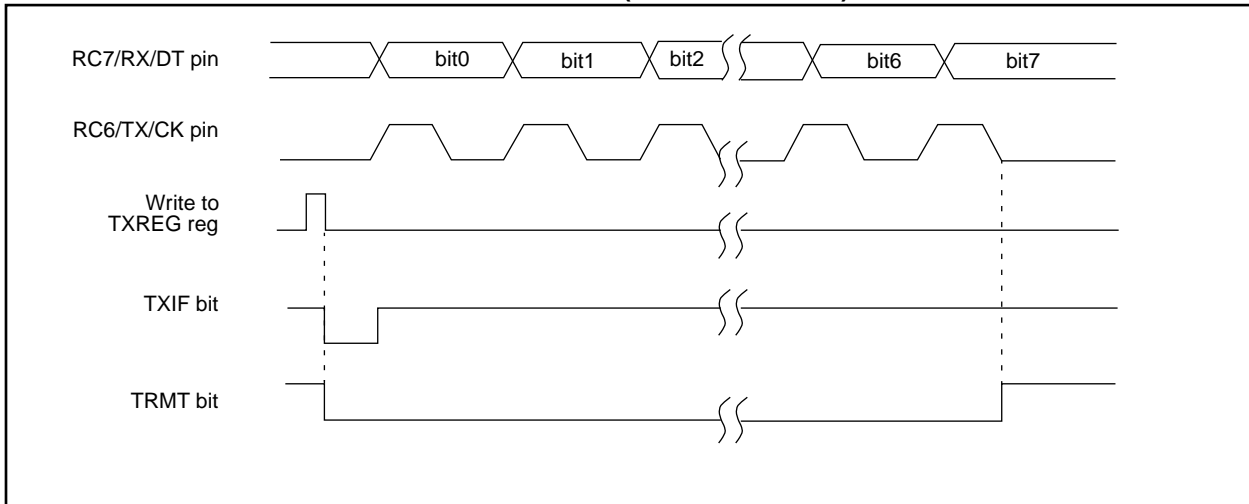
Note 1: PSPIE and PSPIF are reserved on the PIC16C63, always maintain these bits clear.

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

**FIGURE 12-11: SYNCHRONOUS TRANSMISSION**



**FIGURE 12-12: SYNCHRONOUS TRANSMISSION (THROUGH TXEN)**



## 12.3.2 USART SYNCHRONOUS MASTER RECEPTION

Once Synchronous Mode is selected, reception is enabled by setting either enable bit SREN (RCSTA<5>) bit or enable bit CREN (RCSTA<4>). Data is sampled on the DT pin on the falling edge of the clock. If enable bit SREN is set, then only a single word is received. If enable bit CREN is set, the reception is continuous until bit CREN is cleared. If both the bits are set then bit CREN takes precedence. After clocking the last bit, the received data in the Receive Shift Register (RSR) is transferred to the RCREG register (if it is empty). When the transfer is complete, interrupt bit RCIF (PIR1<5>) is set. The actual interrupt can be enabled/disabled by setting/clearing enable bit RCIE (PIE1<5>). Flag bit RCIF is a read only bit which is reset by the hardware. In this case, it is reset when the RCREG register has been read and is empty. The RCREG is a double buffered register, i.e., it is a two deep FIFO. It is possible for two bytes of data to be received and transferred to the RCREG FIFO and a third byte to begin shifting into the RSR register. On the clocking of the last bit of the third byte, if the RCREG register is still full, then overrun error bit, OERR (RCSTA<1>) is set. The word in the RSR register will be lost. The RCREG register can be read twice to retrieve the two bytes in the FIFO. Overrun error bit OERR has to be cleared in software (by clearing bit CREN). If bit OERR is set, transfers from the RSR to the RCREG are inhibited, so it is essential to clear bit OERR if it is set. The 9th receive bit is buffered the same way as the receive data. Reading the RCREG register will load bit RX9D with a new value. Therefore it is essential for the user to read the RCSTA register before reading the RCREG register in order not to lose the old RX9D bit information.

Steps to follow when setting up Synchronous Master Reception:

1. Initialize the SPBRG register for the appropriate baud rate (Section 12.1).
2. Enable the synchronous master serial port by setting bits SYNC, SPEN, and CSRC.
3. Ensure bits CREN and SREN are clear.
4. If interrupts are desired, then set enable bit RCIE.
5. If 9-bit reception is desired, then set bit RX9.
6. If a single reception is required, set enable bit SREN. For continuous reception set enable bit CREN.
7. Flag bit RCIF will be set when reception is complete and an interrupt will be generated if enable bit RCIE was set.
8. Read the RCSTA register to get the ninth bit (if enabled) and determine if any error occurred during reception.
9. Read the 8-bit received data by reading the RCREG register.
10. If any error occurred, clear the error by clearing enable bit CREN.

**TABLE 12-9: REGISTERS ASSOCIATED WITH SYNCHRONOUS MASTER RECEPTION**

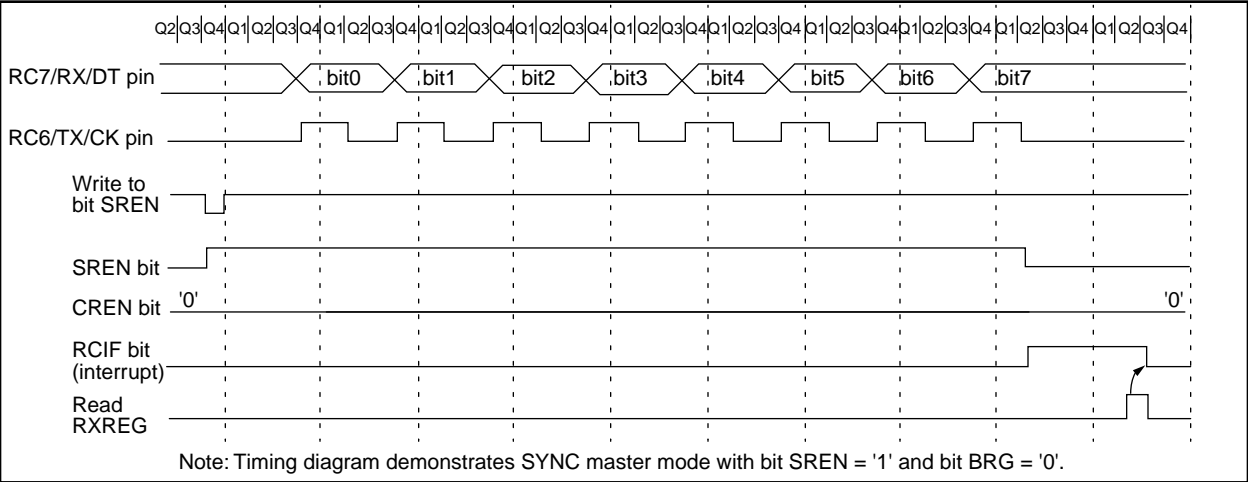
Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR BOR	Value on all other Resets
0Ch	PIR1	PSPIF <sup>(1)</sup>	<sup>(2)</sup>	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
18h	RCSTA	SPEN	RX9	SREN	CREN	—	FERR	OERR	RX9D	0000 -00x	0000 -00x
1Ah	RCREG	RX7	RX6	RX5	RX4	RX3	RX2	RX1	RX0	0000 0000	0000 0000
8Ch	PIE1	PSPIE <sup>(1)</sup>	<sup>(2)</sup>	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
98h	TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010
99h	SPBRG	Baud Rate Generator Register								0000 0000	0000 0000

Legend: x = unknown, - = unimplemented locations read as '0'. Shaded cells are not used for Synchronous Master Reception.

Note 1: PSPIF and PSPIE are reserved on the PIC16C63, always maintain these bits clear.

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

FIGURE 12-13: SYNCHRONOUS RECEPTION (MASTER MODE, SREN)



## 12.4 USART Synchronous Slave Mode

### Applicable Devices

61	62	62A	R62	63	64	64A	R64	65	65A
----	----	-----	-----	----	----	-----	-----	----	-----

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 the 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

**TABLE 12-10: REGISTERS ASSOCIATED WITH SYNCHRONOUS SLAVE TRANSMISSION**

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
0Ch	PIR1	PSPIF <sup>(1)</sup>	(2)	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
18h	RCSTA	SPEN	RX9	SREN	CREN	—	FERR	OERR	RX9D	0000 -00x	0000 -00x
19h	TXREG	TX7	TX6	TX5	TX4	TX3	TX2	TX1	TX0	0000 0000	0000 0000
8Ch	PIE1	PSPIE <sup>(1)</sup>	(2)	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
98h	TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010
99h	SPBRG	Baud Rate Generator Register								0000 0000	0000 0000

Legend: x = unknown, - = unimplemented locations read as '0'. Shaded cells are not used for Synchronous Slave Transmission.

Note 1: PSPIF and PSPIE are reserved on the PIC16C63, always maintain these bits clear.

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

**TABLE 12-11: REGISTERS ASSOCIATED WITH SYNCHRONOUS SLAVE RECEPTION**

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
0Ch	PIR1	PSPIF <sup>(1)</sup>	(2)	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
18h	RCSTA	SPEN	RX9	SREN	CREN	—	FERR	OERR	RX9D	0000 -00x	0000 -00x
1Ah	RCREG	RX7	RX6	RX5	RX4	RX3	RX2	RX1	RX0	0000 0000	0000 0000
8Ch	PIE1	PSPIE <sup>(1)</sup>	(2)	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
98h	TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010
99h	SPBRG	Baud Rate Generator Register								0000 0000	0000 0000

Legend: x = unknown, - = unimplemented locations read as '0'. Shaded cells are not used for Synchronous Slave Reception.

Note 1: PSPIF and PSPIE are reserved on the PIC16C63, always maintain these bits clear.

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

## 13.0 SPECIAL FEATURES OF THE CPU

Applicable Devices									
61	62	62A	R62	63	64	64A	R64	65	65A

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

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

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

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

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

### 13.1 Configuration Bits

Applicable Devices									
61	62	62A	R62	63	64	64A	R64	65	65A

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

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

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

—	—	—	—	—	—	—	—	—	—	CP0	PWRT	WDTE	F0SC1	F0SC0	Register: CONFIG Address 2007h
bit13														bit0	
bit 13-5: <b>Unimplemented:</b> Read as '1'															
bit 4: <b>CP0:</b> Code protection bit 1 = Code protection off 0 = All memory is code protected, but 00h - 3Fh is writable															
bit 3: <b>PWRT:</b> Power-up Timer Enable bit 1 = Power-up Timer enabled 0 = Power-up Timer disabled															
bit 2: <b>WDTE:</b> Watchdog Timer Enable bit 1 = WDT enabled 0 = WDT disabled															
bit 1-0: <b>F0SC1:F0SC0:</b> Oscillator Selection bits 11 = RC oscillator 10 = HS oscillator 01 = XT oscillator 00 = LP oscillator															

# PIC16C6X

**FIGURE 13-2: CONFIGURATION WORD FOR PIC16C62/64/65**

—	—	—	—	—	—	—	—	CP1	CP0	PWRTE	WDTE	F0SC1	F0SC0	Register: CONFIG Address 2007h
bit13													bit0	
bit 13-6:	<b>Unimplemented:</b> Read as '1'													
bit 5-4:	<b>CP1:CP0:</b> Code Protection bits													
	11 = Code protection off													
	10 = Upper half of program memory code protected													
	01 = Upper 3/4th of program memory code protected													
	00 = All memory is code protected													
bit 3:	<b>PWRTE:</b> Power-up Timer Enable bit													
	1 = Power-up Timer enabled													
	0 = Power-up Timer disabled													
bit 2:	<b>WDTE:</b> Watchdog Timer Enable bit													
	1 = WDT enabled													
	0 = WDT disabled													
bit 1-0:	<b>FOSC1:FOSC0:</b> Oscillator Selection bits													
	11 = RC oscillator													
	10 = HS oscillator													
	01 = XT oscillator													
	00 = LP oscillator													

**FIGURE 13-3: CONFIGURATION WORD FOR PIC16C62A/R62/63/64A/R64/65A**

CP1	CP0	CP1	CP0	CP1	CP0	—	BODEN	CP1	CP0	PWRTE	WDTE	F0SC1	F0SC0	Register: CONFIG Address 2007h
bit13													bit0	
bit 13-8:	<b>CP1:CP0:</b> Code Protection bits <sup>(2)</sup>													
bit 5:4:	11 = Code protection off													
	10 = Upper half of program memory code protected													
	01 = Upper 3/4th of program memory code protected													
	00 = All memory is code protected													
bit 7:	<b>Unimplemented:</b> Read as '1'													
bit 6:	<b>BODEN:</b> Brown-out Reset Enable bit <sup>(1)</sup>													
	1 = Brown-out Reset enabled													
	0 = Brown-out Reset disabled													
bit 3:	<b>PWRTE:</b> Power-up Timer Enable bit <sup>(1)</sup>													
	1 = Power-up Timer disabled													
	0 = Power-up Timer enabled													
bit 2:	<b>WDTE:</b> Watchdog Timer Enable bit													
	1 = WDT enabled													
	0 = WDT disabled													
bit 1-0:	<b>FOSC1:FOSC0:</b> Oscillator Selection bits													
	11 = RC oscillator													
	10 = HS oscillator													
	01 = XT oscillator													
	00 = LP oscillator													
Note 1: Enabling Brown-out Reset automatically enables Power-up Timer (PWRT) regardless of the value of bit <b>PWRTE</b> . Ensure the Power-up Timer is enabled anytime Brown-out Reset is enabled.														
2: All of the CP1:CP0 pairs have to be given the same value to implement the code protection scheme listed.														



## 13.2 Oscillator Configurations

Applicable Devices										
61	62	62A	R62	63	64	64A	R64	65	65A	

### 13.2.1 OSCILLATOR TYPES

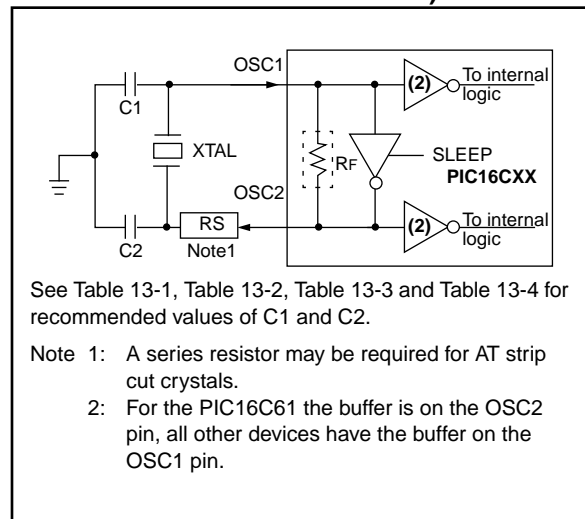
The PIC16CXX can be operated in four different oscillator modes. The user can program two configuration bits (FOSC1 and FOSC0) to select one of these four modes:

- LP Low Power Crystal
- XT Crystal/Resonator
- HS High Speed Crystal/Resonator
- RC Resistor/Capacitor

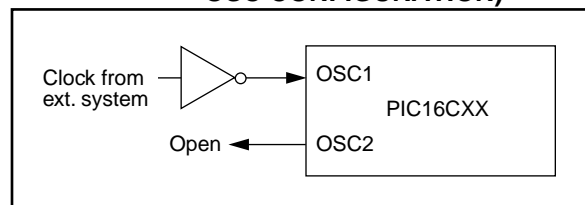
### 13.2.2 CRYSTAL OSCILLATOR/CERAMIC RESONATORS

In LP, XT, or HS modes a crystal or ceramic resonator is connected to the OSC1/CLKIN and OSC2/CLKOUT pins to establish oscillation (Figure 13-4). The PIC16CXX oscillator design requires the use of a parallel cut crystal. Use of a series cut crystal may give a frequency out of the crystal manufacturers specifications. When in LP, XT, or HS modes, the device can have an external clock source to drive the OSC1/CLKIN pin (Figure 13-5).

**FIGURE 13-4: CRYSTAL/CERAMIC RESONATOR OPERATION (HS, XT OR LP OSC CONFIGURATION)**



**FIGURE 13-5: EXTERNAL CLOCK INPUT OPERATION (HS, XT OR LP OSC CONFIGURATION)**



# PIC16C6X

**TABLE 13-1: CERAMIC RESONATORS  
PIC16C61**

Ranges Tested:			
Mode	Freq	OSC1	OSC2
XT	455 kHz	47 - 100 pF	47 - 100 pF
	2.0 MHz	15 - 68 pF	15 - 68 pF
	4.0 MHz	15 - 68 pF	15 - 68 pF
HS	8.0 MHz	15 - 68 pF	15 - 68 pF
	16.0 MHz	10 - 47 pF	10 - 47 pF
<p>Note : Recommended values of C1 and C2 are identical to the ranges tested table.</p> <p>Higher capacitance increases the stability of oscillator but also increases the start-up time. These values are for design guidance only. Since each resonator has its own characteristics, the user should consult the resonator manufacturer for appropriate values of external components.</p>			
Resonators Used:			
455 kHz	Panasonic EFO-A455K04B	± 0.3%	
2.0 MHz	Murata Erie CSA2.00MG	± 0.5%	
4.0 MHz	Murata Erie CSA4.00MG	± 0.5%	
8.0 MHz	Murata Erie CSA8.00MT	± 0.5%	
16.0 MHz	Murata Erie CSA16.00MX	± 0.5%	
All resonators used did not have built-in capacitors.			

**TABLE 13-2: CAPACITOR SELECTION  
FOR CRYSTAL OSCILLATOR  
FOR PIC16C61**

Mode	Freq	OSC1	OSC2
LP	32 kHz	33 - 68 pF	33 - 68 pF
	200 kHz	15 - 47 pF	15 - 47 pF
XT	100 kHz	47 - 100 pF	47 - 100 pF
	500 kHz	20 - 68 pF	20 - 68 pF
	1 MHz	15 - 68 pF	15 - 68 pF
	2 MHz	15 - 47 pF	15 - 47 pF
	4 MHz	15 - 33 pF	15 - 33 pF
HS	8 MHz	15 - 47 pF	15 - 47 pF
	20 MHz	15 - 47 pF	15 - 47 pF
Note: Higher capacitance increases the stability of oscillator but also increases the start-up time. These values are for design guidance only. Rs may be required in HS mode as well as XT mode to avoid overdriving crystals with low drive level specification. Since each crystal has its own characteristics, the user should consult the crystal manufacturer for appropriate values of external components.			

**TABLE 13-3: CERAMIC RESONATORS  
PIC16C62/62A/R62/63/64/64A/  
R64/65/65A**

Ranges Tested:			
Mode	Freq	OSC1	OSC2
XT	455 kHz	68 - 100 pF	68 - 100 pF
	2.0 MHz	15 - 68 pF	15 - 68 pF
	4.0 MHz	15 - 68 pF	15 - 68 pF
HS	8.0 MHz	10 - 68 pF	10 - 68 pF
	16.0 MHz	10 - 22 pF	10 - 22 pF
<p>Note: Recommended values of C1 and C2 are identical to the ranges tested table.</p> <p>Higher capacitance increases the stability of oscillator but also increases the start-up time. These values are for design guidance only. Since each resonator has its own characteristics, the user should consult the resonator manufacturer for appropriate values of external components.</p>			
Resonators Used:			
455 kHz	Panasonic EFO-A455K04B	± 0.3%	
2.0 MHz	Murata Erie CSA2.00MG	± 0.5%	
4.0 MHz	Murata Erie CSA4.00MG	± 0.5%	
8.0 MHz	Murata Erie CSA8.00MT	± 0.5%	
16.0 MHz	Murata Erie CSA16.00MX	± 0.5%	
All resonators used did not have built-in capacitors.			

**TABLE 13-4: CAPACITOR SELECTION  
FOR CRYSTAL OSCILLATOR  
FOR PIC16C62/62A/R62/63/64/  
64A/R64/65/65A**

Mode	Freq	OSC1	OSC2
LP	32 kHz <sup>(1)</sup>	15 - 47 pF	15 - 47 pF
	200 kHz	15 - 33 pF	15 - 33 pF
XT	100 kHz	47 - 100 pF	47 - 100 pF
	500 kHz	20 - 68 pF	20 - 68 pF
	1 MHz	15 - 68 pF	15 - 68 pF
	2 MHz	15 - 47 pF	15 - 47 pF
	4 MHz	15 - 33 pF	15 - 33 pF
HS	8 MHz	15 - 47 pF	15 - 47 pF
	20 MHz	15 - 47 pF	15 - 47 pF
Note: Higher capacitance increases the stability of oscillator but also increases the start-up time. These values are for design guidance only. Rs may be required in HS mode as well as XT mode to avoid overdriving crystals with low drive level specification. Since each crystal has its own characteristics, the user should consult the crystal manufacturer for appropriate values of external components.			
Note 1: For VDD > 4.5V, C1 = C2 ≈ 30 pF is recommended.			

## 13.2.3 EXTERNAL CRYSTAL OSCILLATOR CIRCUIT

Either a prepackaged oscillator can be used or a simple oscillator circuit with TTL gates can be built. Prepackaged oscillators provide a wide operating range and better stability. A well-designed crystal oscillator will provide good performance with TTL gates. Two types of crystal oscillator circuits can be used; one with series resonance, or one with parallel resonance.

Figure 13-6 shows implementation of a parallel resonant oscillator circuit. The circuit is designed to use the fundamental frequency of the crystal. The 74AS04 inverter performs the 180-degree phase shift that a parallel oscillator requires. The 4.7 k $\Omega$  resistor provides the negative feedback for stability. The 10 k $\Omega$  potentiometer biases the 74AS04 in the linear region. This could be used for external oscillator designs.

**FIGURE 13-6: EXTERNAL PARALLEL RESONANT CRYSTAL OSCILLATOR CIRCUIT**

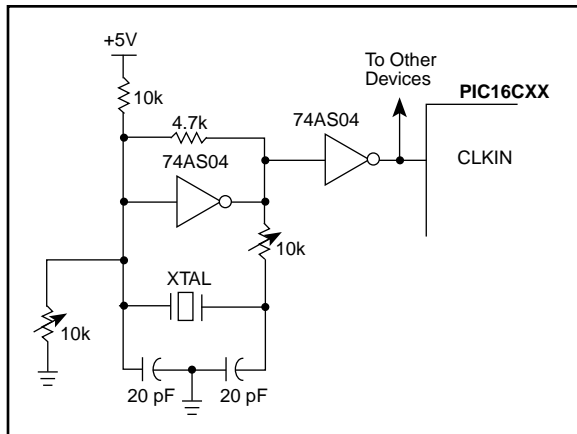
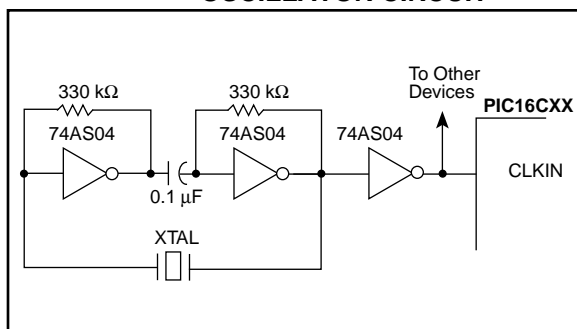


Figure 13-7 shows a series resonant oscillator circuit. This circuit is also designed to use the fundamental frequency of the crystal. The inverter performs a 180-degree phase shift in a series resonant oscillator circuit. The 330 k $\Omega$  resistors provide the negative feedback to bias the inverters in their linear region.

**FIGURE 13-7: EXTERNAL SERIES RESONANT CRYSTAL OSCILLATOR CIRCUIT**



## 13.2.4 RC OSCILLATOR

For timing insensitive applications the RC device option offers additional cost savings. The RC oscillator frequency is a function of the supply voltage, the resistor ( $R_{ext}$ ) and capacitor ( $C_{ext}$ ) values, and the operating temperature. In addition to this, the oscillator frequency will vary from unit to unit due to normal process parameter variation. Furthermore, the difference in lead frame capacitance between package types will also affect the oscillation frequency, especially for low  $C_{ext}$  values. The user also needs to take into account variation due to tolerance of external R and C components used. Figure 13-8 shows how the RC combination is connected to the PIC16CXX. For  $R_{ext}$  values below 2.2 k $\Omega$ , the oscillator operation may become unstable or stop completely. For very high  $R_{ext}$  values (e.g. 1 M $\Omega$ ), the oscillator becomes sensitive to noise, humidity and leakage. Thus, we recommend keeping  $R_{ext}$  between 3 k $\Omega$  and 100 k $\Omega$ .

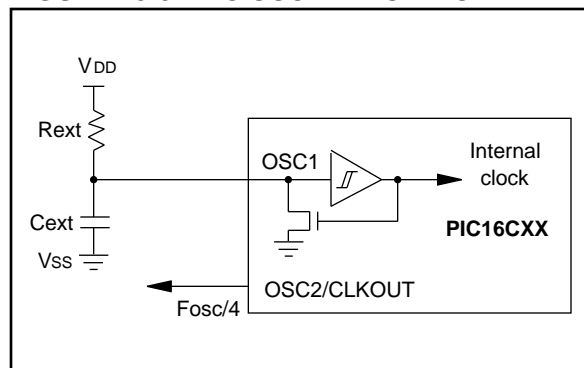
Although the oscillator will operate with no external capacitor ( $C_{ext} = 0$  pF), we recommend using values above 20 pF for noise and stability reasons. With no or small external capacitance, the oscillation frequency can vary dramatically due to changes in external capacitances, such as PCB trace capacitance or package lead frame capacitance.

See characterization data for desired device for RC frequency variation from part to part due to normal process variation. The variation is larger for larger R (since leakage current variation will affect RC frequency more for large R) and for smaller C (since variation of input capacitance will affect RC frequency more).

See characterization data for desired device for variation of oscillator frequency due to  $V_{DD}$  for given  $R_{ext}/C_{ext}$  values as well as frequency variation due to operating temperature for given R, C, and  $V_{DD}$  values.

The oscillator frequency, divided by 4, is available on the OSC2/CLKOUT pin, and can be used for test purposes or to synchronize other logic (see Figure 3-4 for waveform).

**FIGURE 13-8: RC OSCILLATOR MODE**



# PIC16C6X

## 13.3 Reset

### Applicable Devices

61	62	62A	R62	63	64	64A	R64	65	65A
----	----	-----	-----	----	----	-----	-----	----	-----

The PIC16CXX differentiates between various kinds of reset:

- Power-on Reset (POR)
- $\overline{\text{MCLR}}$  reset during normal operation
- $\overline{\text{MCLR}}$  reset during SLEEP
- WDT Reset (normal operation)
- Brown-out Reset (BOR) - PIC16C62A/R62/63/64A/R64/65A

Some registers are not affected in any reset condition, their status is unknown on POR and unchanged in any other reset. Most other registers are reset to a "reset state" on Power-on Reset (POR), on  $\overline{\text{MCLR}}$  or WDT Reset, on  $\overline{\text{MCLR}}$  reset during SLEEP, and on Brown-out Reset (BOR). They are not affected by a WDT Wake-up, which is viewed as the resumption of normal operation.

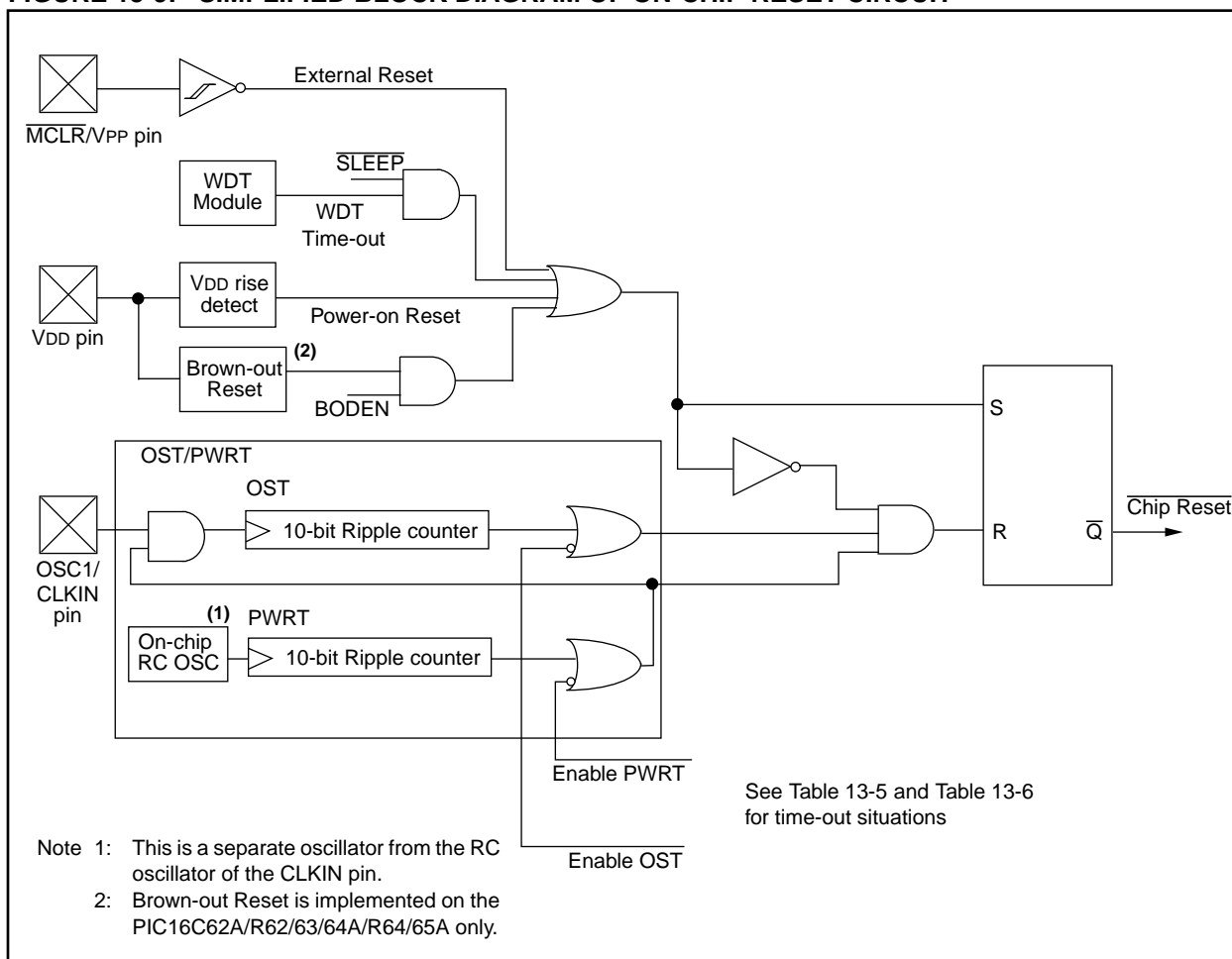
The  $\overline{\text{TO}}$  and  $\overline{\text{PD}}$  bits are set or cleared differently in different reset situations as indicated in Table 13-7, Table 13-8, and Table 13-9. These bits are used in software to determine the nature of the reset. See Table 13-12 for a full description of reset states of all registers.

A simplified block diagram of the on-chip reset circuit is shown in Figure 13-9.

On the PIC16C62A/R62/63/64A/R64/65A, the  $\overline{\text{MCLR}}$  reset path has a noise filter to detect and ignore small pulses. See parameter #34 for pulse width specifications.

It should be noted that a WDT Reset does not drive the  $\overline{\text{MCLR}}$  pin low.

**FIGURE 13-9: SIMPLIFIED BLOCK DIAGRAM OF ON-CHIP RESET CIRCUIT**



## 13.4 Power-on Reset (POR), Power-up Timer (PWRT), Oscillator Start-up Timer (OST) and Brown-out Reset (BOR)

Applicable Devices									
61	62	62A	R62	63	64	64A	R64	65	65A

### 13.4.1 POWER-ON RESET (POR)

A Power-on Reset pulse is generated on-chip when VDD rise is detected (in the range of 1.5V - 2.1V). To take advantage of the POR, just tie the  $\overline{\text{MCLR}}/\text{VPP}$  pin directly (or through a resistor) to VDD. This will eliminate external RC components usually needed to create a Power-on Reset. A maximum rise time for VDD is required. See Electrical Specifications for details.

When the device starts normal operation (exits the reset condition), device operating parameters (voltage, frequency, temperature, ...) must be met to ensure operation. If these conditions are not met, the device must be held in reset until the operating conditions are met.

For additional information, refer to Application Note AN607, "Power-up Trouble Shooting."

### 13.4.2 POWER-UP TIMER (PWRT)

The Power-up Timer provides a fixed 72 ms nominal time-out on power-up only, from POR. The Power-up Timer operates on an internal RC oscillator. The chip is kept in reset as long as PWRT is active. The PWRT's time delay allows VDD to rise to an acceptable level. A configuration bit is provided to enable/disable the PWRT.

The power-up time delay will vary from chip to chip due to VDD, temperature, and process variation. See DC parameters for details.

### 13.4.3 OSCILLATOR START-UP TIMER (OST)

The Oscillator Start-up Timer (OST) provides 1024 oscillator cycle (from OSC1 input) delay after the PWRT delay is over. This ensures the crystal oscillator or resonator has started and stabilized.

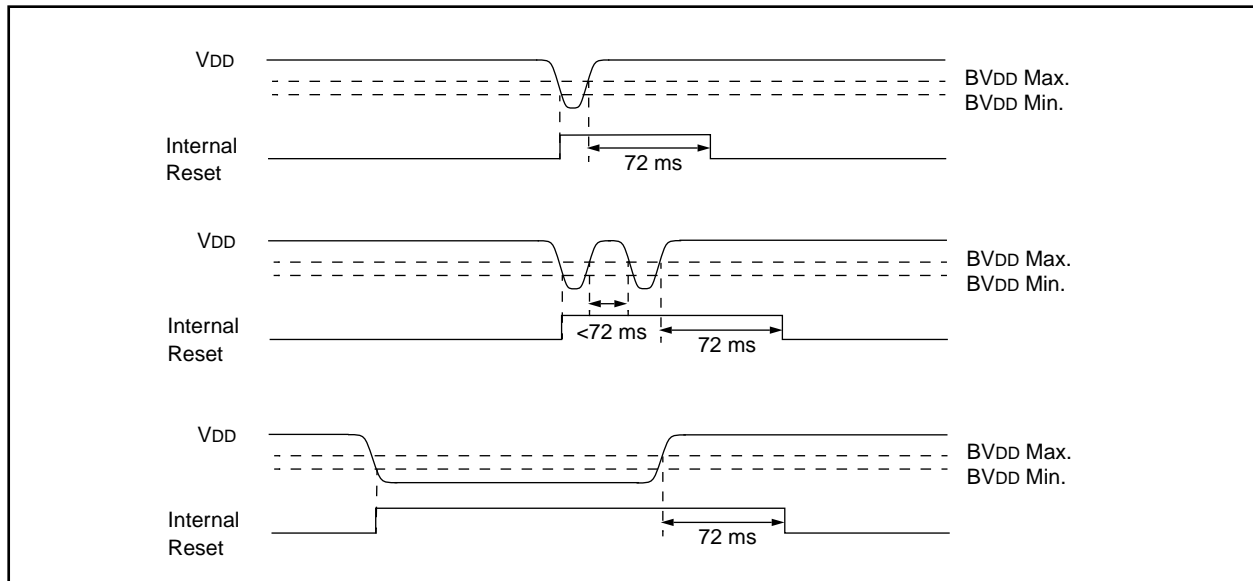
The OST time-out is invoked only for XT, LP and HS modes and only on Power-on Reset or wake-up from SLEEP.

### 13.4.4 BROWN-OUT RESET (BOR)

Applicable Devices									
61	62	62A	R62	63	64	64A	R64	65	65A

A configuration bit, BODEN, can disable (if clear/programmed) or enable (if set) the Brown-out Reset circuitry. If VDD falls below 4.0V (parameter D005 in Electrical Specification section) for greater than parameter #34, the brown-out situation will reset the chip. A reset may not occur if VDD falls below 4.0V for less than parameter #34. The chip will remain in Brown-out Reset until VDD rises above BVDD. The Power-up Timer will now be invoked and will keep the chip in RESET an additional 72 ms. If VDD drops below BVDD while the Power-up Timer is running, the chip will go back into a Brown-out Reset and the Power-up Timer will be initialized. Once VDD rises above BVDD, the Power-up Timer will execute a 72 ms time delay. The Power-up Timer should always be enabled when Brown-out Reset is enabled. Figure 13-10 shows typical brown-out situations.

**FIGURE 13-10: BROWN-OUT SITUATIONS**



# PIC16C6X

## 13.4.5 TIME-OUT SEQUENCE

On power-up the time-out sequence is as follows: First a PWRT time-out is invoked after the POR time delay has expired. Then OST is activated. The total time-out will vary based on oscillator configuration and the status of the PWRT. For example, in RC mode, with the PWRT disabled, there will be no time-out at all. Figure 13-11, Figure 13-12, and Figure 13-13 depict time-out sequences on power-up.

Since the time-outs occur from the POR pulse, if the MCLR/VPP pin is kept low long enough, the time-outs will expire. Then bringing the MCLR/VPP pin high will begin execution immediately (Figure 13-14). This is useful for testing purposes or to synchronize more than one PIC16CXX device operating in parallel.

Table 13-10 and Table 13-11 show the reset conditions for some special function registers, while Table 13-12 shows the reset conditions for all the registers.

## 13.4.6 POWER CONTROL/STATUS REGISTER (PCON)

Applicable Devices									
61	62	62A	R62	63	64	64A	R64	65	65A

The Power Control/Status Register, PCON has up to 2 bits, depending upon the device. Bit0 is not implemented on the PIC16C62/64/65.

Bit0 is  $\overline{\text{BOR}}$  (Brown-out Reset Status bit).  $\overline{\text{BOR}}$  is unknown on Power-on Reset. It must then be set by the user and checked on subsequent resets to see if  $\overline{\text{BOR}}$  cleared, indicating that a brown-out has occurred. The  $\overline{\text{BOR}}$  status bit is a “Don’t Care” and is not necessarily predictable if the Brown-out Reset circuitry is disabled (by clearing bit BODEN in the Configuration Word).

Bit1 is  $\overline{\text{POR}}$  (Power-on Reset Status bit). It is cleared on a Power-on Reset and unaffected otherwise. The user must set this bit following a Power-on Reset.

**TABLE 13-5: TIME-OUT IN VARIOUS SITUATIONS, PIC16C61/62/64/65**

Oscillator Configuration	Power-up		Wake-up from SLEEP
	PWRT = 1	PWRT = 0	
XT, HS, LP	72 ms + 1024Tosc	1024Tosc	1024 TOSC
RC	72 ms	—	—

**TABLE 13-6: TIME-OUT IN VARIOUS SITUATIONS, PIC16C62A/R62/63/64A/R64/65A**

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

**TABLE 13-7: STATUS BITS AND THEIR SIGNIFICANCE, PIC16C61**

$\overline{\text{TO}}$	$\overline{\text{PD}}$	
1	1	Power-on Reset or MCLR reset during normal operation
0	1	WDT Reset
0	0	WDT Wake-up
1	0	MCLR reset during SLEEP or interrupt wake-up from SLEEP

**TABLE 13-8: STATUS BITS AND THEIR SIGNIFICANCE, PIC16C62/64/65**

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

Legend: x = unknown, u = unchanged

**TABLE 13-9: STATUS BITS AND THEIR SIGNIFICANCE FOR PIC16C62A/R62/63/64A/R64/65A**

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

Legend: x = unknown, u = unchanged

**TABLE 13-10: RESET CONDITION FOR SPECIAL REGISTERS ON PIC16C61/62/64/65**

	Program Counter	STATUS	PCON <sup>(2)</sup>
Power-on Reset	000h	0001 1xxx	---- --0-
$\overline{MCLR}$ reset during normal operation	000h	000u uuuu	---- --u-
$\overline{MCLR}$ reset during SLEEP	000h	0001 0uuu	---- --u-
WDT Reset	000h	0000 1uuu	---- --u-
WDT Wake-up	PC + 1	uuu0 0uuu	---- --u-
Interrupt wake-up from SLEEP	PC + 1 <sup>(1)</sup>	uuu1 0uuu	---- --u-

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

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

2: The PCON register is not implemented on the PIC16C61.

**TABLE 13-11: RESET CONDITION FOR SPECIAL REGISTERS ON PIC16C62A/R62/63/64A/R64/65A**

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

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

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

**TABLE 13-12: INITIALIZATION CONDITIONS FOR ALL REGISTERS**

Register	Applicable Devices										Power-on Reset Brown-out Reset	MCLR Reset during: – normal operation – SLEEP WDT Reset	Wake-up via interrupt or WDT Wake-up
W	61	62	62A	R62	63	64	64A	R64	65	65A	xxxx xxxx	uuuu uuuu	uuuu uuuu
INDF	61	62	62A	R62	63	64	64A	R64	65	65A	N/A	N/A	N/A
TMR0	61	62	62A	R62	63	64	64A	R64	65	65A	xxxx xxxx	uuuu uuuu	uuuu uuuu
PCL	61	62	62A	R62	63	64	64A	R64	65	65A	0000h	0000h	PC + 1 <sup>(2)</sup>
STATUS	61	62	62A	R62	63	64	64A	R64	65	65A	0001 1xxx	000q quuu <sup>(3)</sup>	uuuq quuu <sup>(3)</sup>
FSR	61	62	62A	R62	63	64	64A	R64	65	65A	xxxx xxxx	uuuu uuuu	uuuu uuuu
PORTA	61	62	62A	R62	63	64	64A	R64	65	65A	---x xxxx	---u uuuu	---u uuuu
	61	62	62A	R62	63	64	64A	R64	65	65A	--xx xxxx	--uu uuuu	--uu uuuu
PORTB	61	62	62A	R62	63	64	64A	R64	65	65A	xxxx xxxx	uuuu uuuu	uuuu uuuu
PORTC	61	62	62A	R62	63	64	64A	R64	65	65A	xxxx xxxx	uuuu uuuu	uuuu uuuu
PORTD	61	62	62A	R62	63	64	64A	R64	65	65A	xxxx xxxx	uuuu uuuu	uuuu uuuu
PORTE	61	62	62A	R62	63	64	64A	R64	65	65A	---- -xxx	---- -uuu	---- -uuu
PCLATH	61	62	62A	R62	63	64	64A	R64	65	65A	---0 0000	---0 0000	---u uuuu
INTCON	61	62	62A	R62	63	64	64A	R64	65	65A	0000 000x	0000 000u	uuuu uuuu <sup>(1)</sup>
PIR1	61	62	62A	R62	63	64	64A	R64	65	65A	00-- 0000	00-- 0000	uu-- uuuu <sup>(1)</sup>
	61	62	62A	R62	63	64	64A	R64	65	65A	0000 0000	0000 0000	uuuu uuuu <sup>(1)</sup>
PIR2	61	62	62A	R62	63	64	64A	R64	65	65A	---- --0	---- --0	---- --u <sup>(2)</sup>
TMR1L	61	62	62A	R62	63	64	64A	R64	65	65A	xxxx xxxx	uuuu uuuu	uuuu uuuu
TMR1H	61	62	62A	R62	63	64	64A	R64	65	65A	xxxx xxxx	uuuu uuuu	uuuu uuuu
T1CON	61	62	62A	R62	63	64	64A	R64	65	65A	--00 0000	--uu uuuu	--uu uuuu
TMR2	61	62	62A	R62	63	64	64A	R64	65	65A	0000 0000	0000 0000	uuuu uuuu
T2CON	61	62	62A	R62	63	64	64A	R64	65	65A	-000 0000	-000 0000	-uuu uuuu
SSPBUF	61	62	62A	R62	63	64	64A	R64	65	65A	xxxx xxxx	uuuu uuuu	uuuu uuuu
SSPCON	61	62	62A	R62	63	64	64A	R64	65	65A	0000 0000	0000 0000	uuuu uuuu
CCPR1L	61	62	62A	R62	63	64	64A	R64	65	65A	xxxx xxxx	uuuu uuuu	uuuu uuuu
CCPR1H	61	62	62A	R62	63	64	64A	R64	65	65A	xxxx xxxx	uuuu uuuu	uuuu uuuu
CCP1CON	61	62	62A	R62	63	64	64A	R64	65	65A	--00 0000	--00 0000	--uu uuuu
RCSTA	61	62	62A	R62	63	64	64A	R64	65	65A	0000 -00x	0000 -00x	uuuu -uuu
TXREG	61	62	62A	R62	63	64	64A	R64	65	65A	0000 0000	0000 0000	uuuu uuuu
RCREG	61	62	62A	R62	63	64	64A	R64	65	65A	0000 0000	0000 0000	uuuu uuuu
CCPR2L	61	62	62A	R62	63	64	64A	R64	65	65A	xxxx xxxx	uuuu uuuu	uuuu uuuu
CCPR2H	61	62	62A	R62	63	64	64A	R64	65	65A	xxxx xxxx	uuuu uuuu	uuuu uuuu
CCP2CON	61	62	62A	R62	63	64	64A	R64	65	65A	0000 0000	0000 0000	uuuu uuuu
OPTION	61	62	62A	R62	63	64	64A	R64	65	65A	1111 1111	1111 1111	uuuu uuuu
TRISA	61	62	62A	R62	63	64	64A	R64	65	65A	---1 1111	---1 1111	---u uuuu
	61	62	62A	R62	63	64	64A	R64	65	65A	--11 1111	--11 1111	--uu uuuu
TRISB	61	62	62A	R62	63	64	64A	R64	65	65A	1111 1111	1111 1111	uuuu uuuu

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

Note 1: One or more bits in INTCON, PIR1 and/or PIR2 will be affected (to cause wake-up).

2: When the wake-up is due to an interrupt and the global enable bit, GIE is set, the PC is loaded with the interrupt vector (0004h) after execution of PC + 1.

3: See Table 13-10 and Table 13-11 for reset value for specific conditions.



**TABLE 13-12: INITIALIZATION CONDITIONS FOR ALL REGISTERS (Cont.'d)**

Register	Applicable Devices										Power-on Reset Brown-out Reset	MCLR Reset during: – normal operation – SLEEP WDT Reset	Wake-up via interrupt or WDT Wake-up
TRISC	61	62	62A	R62	63	64	64A	R64	65	65A	1111 1111	1111 1111	uuuu uuuu
TRISD	61	62	62A	R62	63	64	64A	R64	65	65A	1111 1111	1111 1111	uuuu uuuu
TRISE	61	62	62A	R62	63	64	64A	R64	65	65A	0000 -111	0000 -111	uuuu -uuu
PIE1	61	62	62A	R62	63	64	64A	R64	65	65A	00-- 0000	00-- 0000	uu-- uuuu
	61	62	62A	R62	63	64	64A	R64	65	65A	0000 0000	0000 0000	uuuu uuuu
PIE2	61	62	62A	R62	63	64	64A	R64	65	65A	---- --0	---- --0	---- --u
PCON	61	62	62A	R62	63	64	64A	R64	65	65A	---- --0u	---- --uu	---- --uu
	61	62	62A	R62	63	64	64A	R64	65	65A	---- --0-	---- --u-	---- --u-
PR2	61	62	62A	R62	63	64	64A	R64	65	65A	1111 1111	1111 1111	1111 1111
SSPADD	61	62	62A	R62	63	64	64A	R64	65	65A	0000 0000	0000 0000	uuuu uuuu
SSPSTAT	61	62	62A	R62	63	64	64A	R64	65	65A	--00 0000	--00 0000	--uu uuuu
TXSTA	61	62	62A	R62	63	64	64A	R64	65	65A	0000 -010	0000 -010	uuuu -uuu
SPBRG	61	62	62A	R62	63	64	64A	R64	65	65A	0000 0000	0000 0000	uuuu uuuu

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

Note 1: One or more bits in INTCON, PIR1 and/or PIR2 will be affected (to cause wake-up).

2: When the wake-up is due to an interrupt and the global enable bit, GIE is set, the PC is loaded with the interrupt vector (0004h) after execution of PC + 1.

3: See Table 13-10 and Table 13-11 for reset value for specific conditions.

FIGURE 13-11: TIME-OUT SEQUENCE ON POWER-UP ( $\overline{\text{MCLR}}$  NOT TIED TO  $V_{DD}$ ): CASE 1

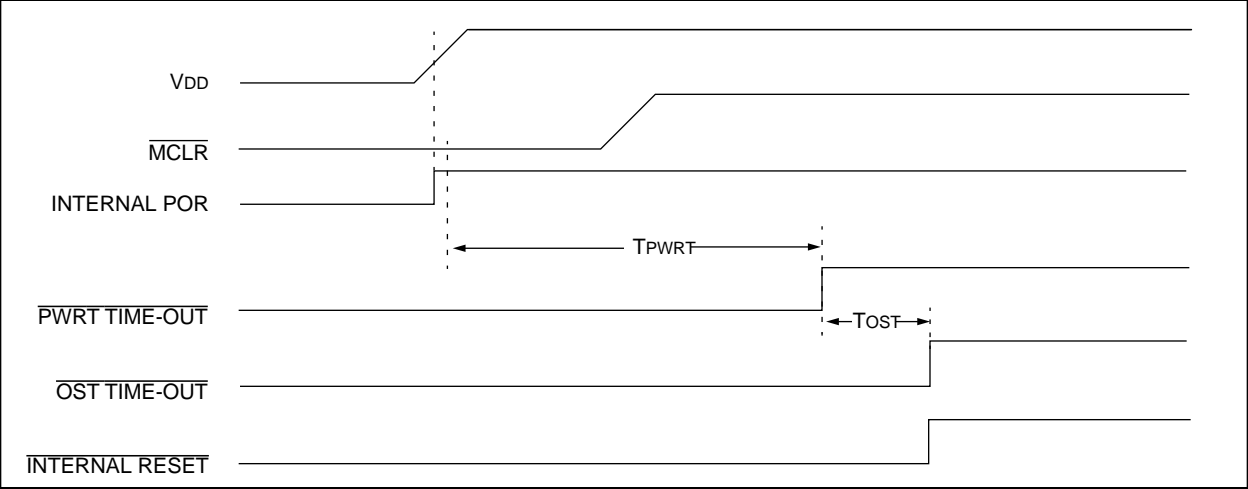


FIGURE 13-12: TIME-OUT SEQUENCE ON POWER-UP ( $\overline{\text{MCLR}}$  NOT TIED TO  $V_{DD}$ ): CASE 2

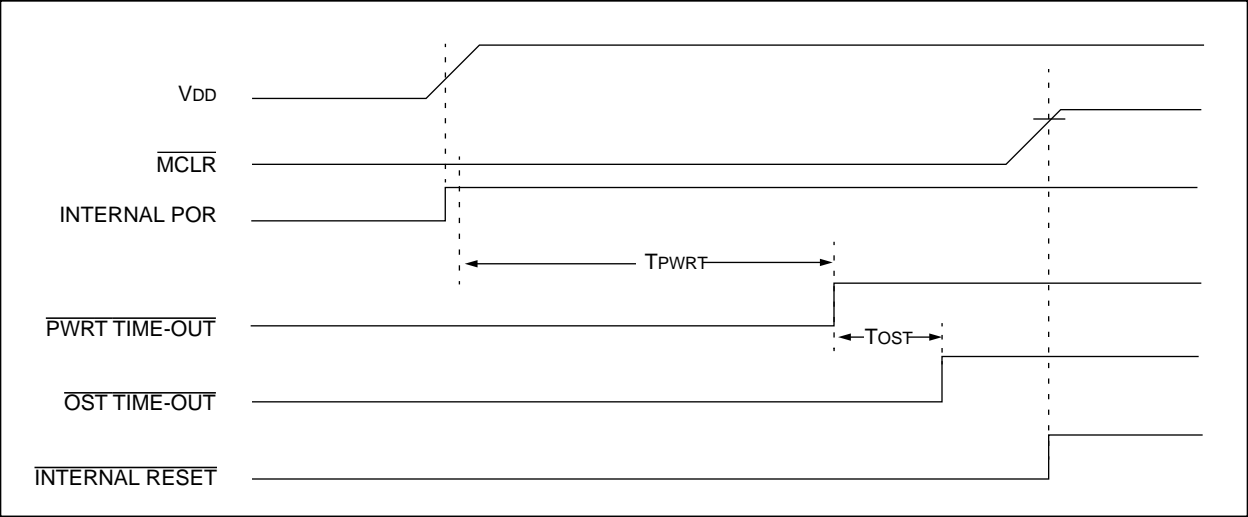
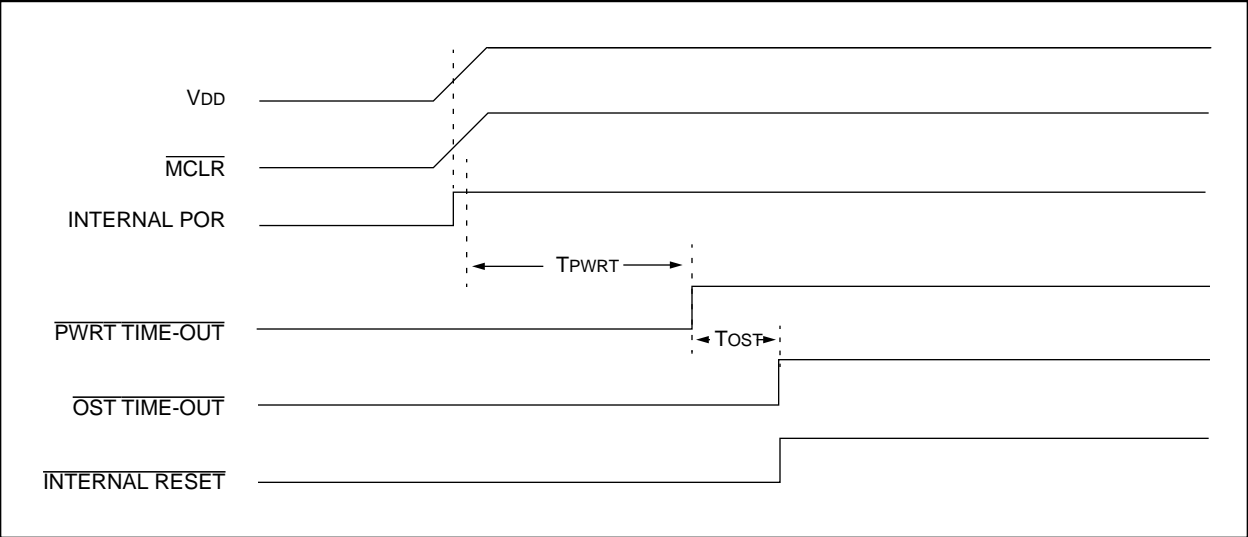
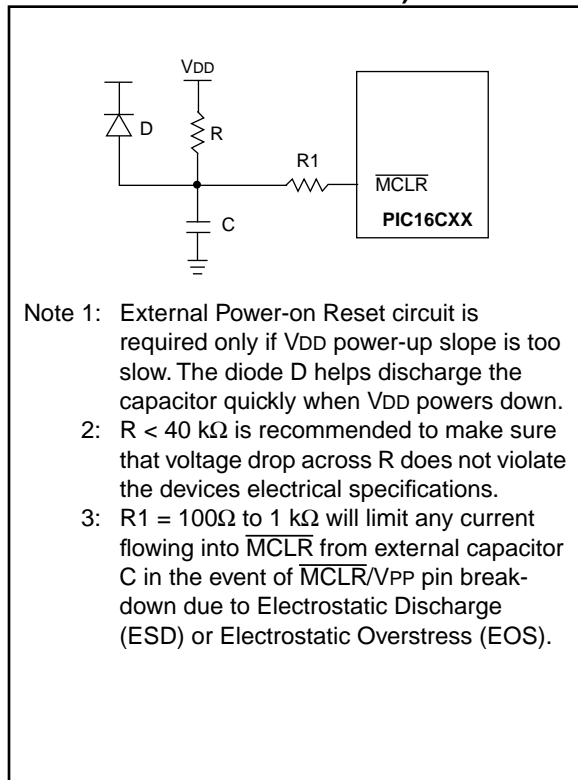


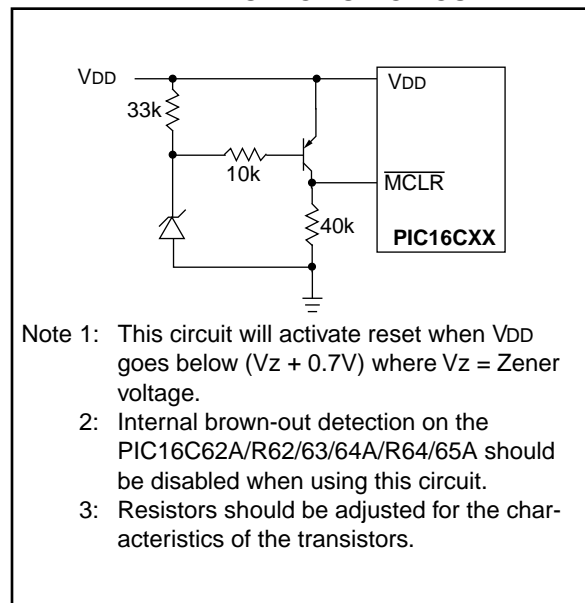
FIGURE 13-13: TIME-OUT SEQUENCE ON POWER-UP ( $\overline{\text{MCLR}}$  TIED TO  $V_{DD}$ )



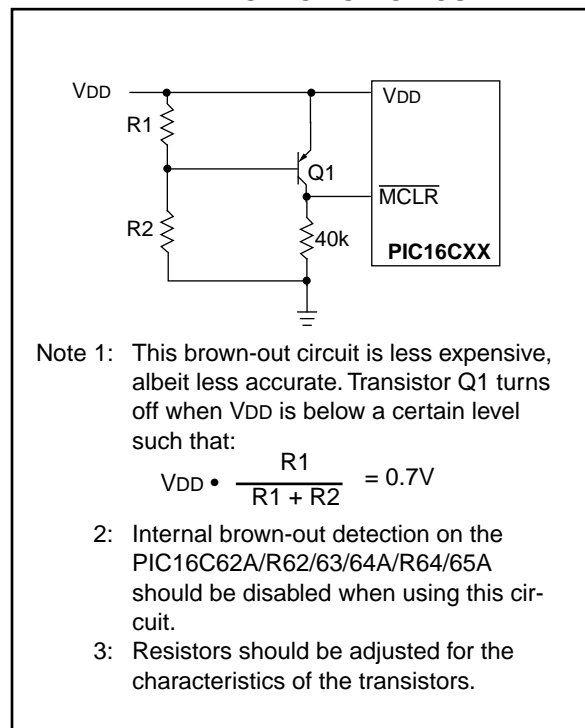
**FIGURE 13-14: EXTERNAL POWER-ON RESET CIRCUIT (FOR SLOW VDD POWER-UP)**



**FIGURE 13-15: EXTERNAL BROWN-OUT PROTECTION CIRCUIT 1**



**FIGURE 13-16: EXTERNAL BROWN-OUT PROTECTION CIRCUIT 2**



# PIC16C6X

## 13.5 Interrupts

Applicable Devices									
61	62	62A	R62	63	64	64A	R64	65	65A

The PIC16C6X family has up to 11 sources of interrupt:

Interrupt Sources	Applicable Devices									
External interrupt RB0/INT	61	62	62A	R62	63	64	64A	R64	65	65A
TMR0 overflow interrupt	61	62	62A	R62	63	64	64A	R64	65	65A
PORTB change interrupt (pins RB7:RB4)	61	62	62A	R62	63	64	64A	R64	65	65A
TMR1 overflow interrupt	61	62	62A	R62	63	64	64A	R64	65	65A
TMR2 matches period interrupt	61	62	62A	R62	63	64	64A	R64	65	65A
CCP1 interrupt	61	62	62A	R62	63	64	64A	R64	65	65A
CCP2 interrupt	61	62	62A	R62	63	64	64A	R64	65	65A
USART Receive	61	62	62A	R62	63	64	64A	R64	65	65A
USART Transmit	61	62	62A	R62	63	64	64A	R64	65	65A
Synchronous serial port interrupt	61	62	62A	R62	63	64	64A	R64	65	65A
Parallel slave port read/write interrupt	61	62	62A	R62	63	64	64A	R64	65	65A

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). 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-

20). 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 PIC16C61/64/65 only, 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:

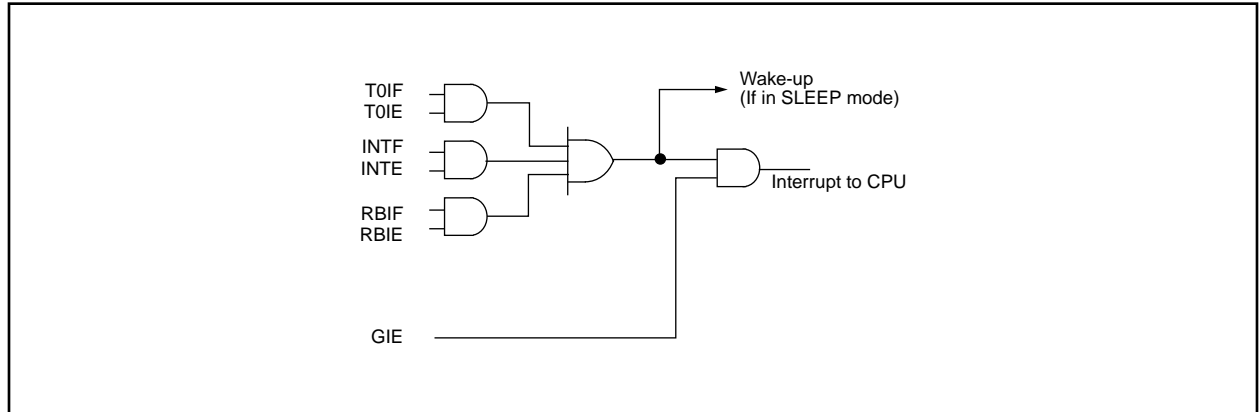
1. An instruction clears the GIE bit while an interrupt is acknowledged
2. The program branches to the Interrupt vector and executes the Interrupt Service Routine.
3. The Interrupt Service Routine completes with the execution of the RETFIE instruction. This causes the GIE bit to be set (enables interrupts), 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.

```

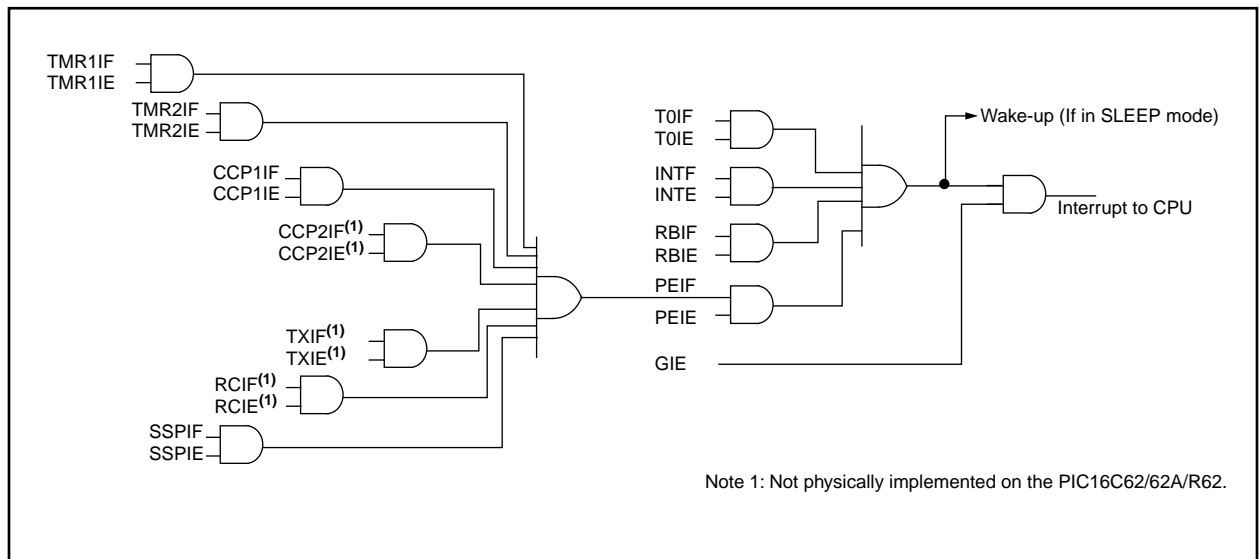
LOOP BCF INTCON,GIE      ;Disable Global
                                ;Interrupt bit
      BTFSC INTCON,GIE    ;Global Interrupt
                                ;Disabled?
      GOTO LOOP           ;NO, try again
      :                   ;Yes, continue
                                ;with program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

```

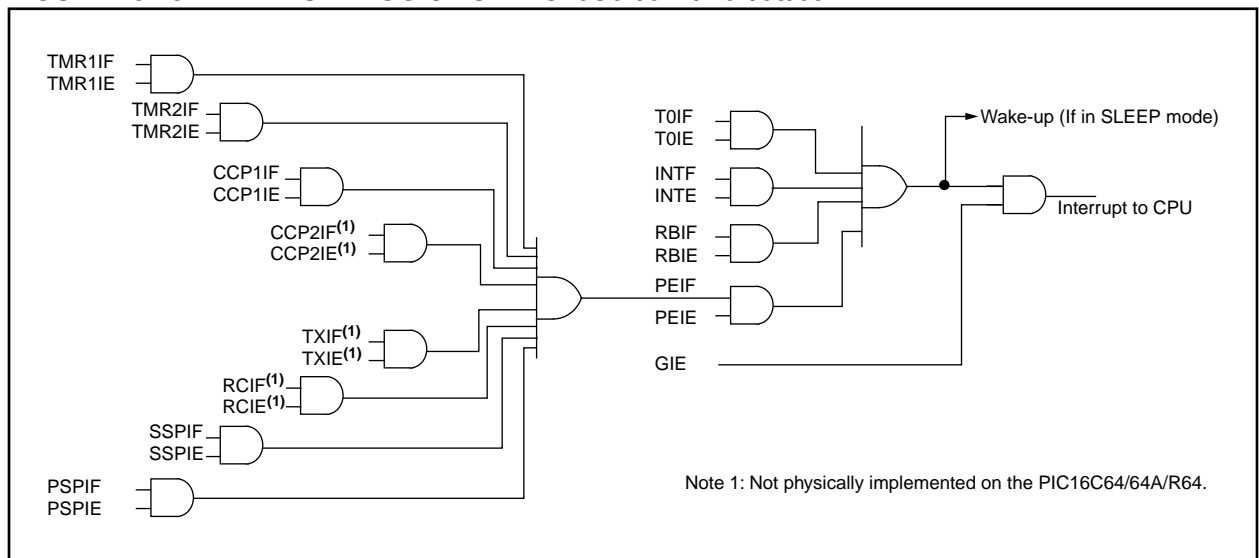
**FIGURE 13-17: INTERRUPT LOGIC FOR PIC16C61**



**FIGURE 13-18: INTERRUPT LOGIC FOR PIC16C62/62A/R62/63**



**FIGURE 13-19: INTERRUPT LOGIC FOR PIC16C64/64A/R64/65/65A**



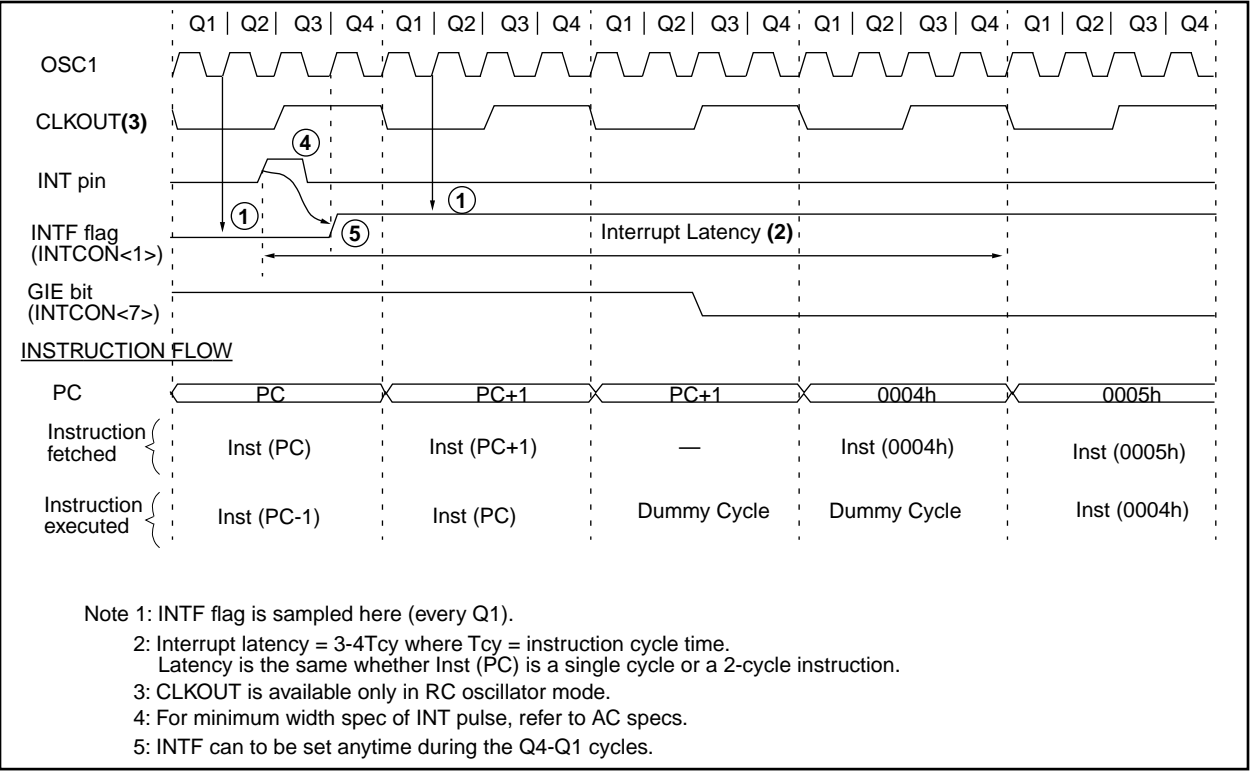
13.5.1 INT INTERRUPT

External interrupt on RB0/INT pin is edge triggered: either rising if edge select bit INTEDG (OPTION<6>) is set, or falling, if bit INTEDG is clear. When a valid edge appears on the RB0/INT pin, flag bit INTF (INTCON<1>) is set. This interrupt can be disabled by clearing enable bit INTE (INTCON<4>). The INTF bit must be cleared in software in the interrupt service routine before re-enabling this interrupt. The INT interrupt can wake the processor from SLEEP, if enable bit INTE was set prior to going into SLEEP. The status of global enable bit GIE decides whether or not the processor branches to the interrupt vector following wake-up. See Section 13.8 for details on SLEEP mode.

13.5.2 TMR0 INTERRUPT

An overflow (FFh → 00h) in the TMR0 register will set flag bit T0IF (INTCON<2>). The interrupt can be enabled/disabled by setting/clearing enable bit T0IE (INTCON<5>) (Section 7.0).

FIGURE 13-20: INT PIN INTERRUPT TIMING



## 13.5.3 PORTB INTERRUPT ON CHANGE

An input change on PORTB<7:4> sets flag bit RBIF (INTCON<0>). The interrupt can be enabled/disabled by setting/clearing enable bit RBIE (INTCON<4>) (Section 5.2).

**Note:** For the PIC16C61/62/64/65 only, if a change on the I/O pin should occur when the read operation is being executed (start of the Q2 cycle), then flag bit RBIF may not get set.

## 13.6 Context Saving During Interrupts

### Applicable Devices

61	62	62A	R62	63	64	64A	R64	65	65A
----	----	-----	-----	----	----	-----	-----	----	-----

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 and Example 13-2 store and restore the STATUS and W registers. For all PIC16C6X devices with greater than 1K of program memory (all devices except PIC16C61), the register, W\_TEMP, must be defined in both 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).

The examples:

- Stores the W register.
- Stores the STATUS register in bank 0.
- Executes ISR code.
- 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 AND W REGISTERS IN RAM (PIC16C62/62A/R62/63/64/64A/R64/65/65A)

```

MOVWF    W_TEMP            ;Copy W to TEMP register, could be bank one or zero
SWAPF    STATUS,W          ;Swap status to be saved into W
BCF      STATUS,RP0        ;Change to bank zero, regardless of current bank
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

```

# PIC16C6X

## 13.7 Watchdog Timer (WDT)

### Applicable Devices

61	62	62A	R62	63	64	64A	R64	65	65A
----	----	-----	-----	----	----	-----	-----	----	-----

The Watchdog Timer is a free running on-chip RC oscillator which does not require any external components. This RC oscillator is separate from the RC oscillator of the OSC1/CLKIN pin. That means that the WDT will run, even if the clock on the OSC1/CLKIN and OSC2/CLKOUT pins of the device has been stopped, for example, by execution of a *SLEEP* instruction. During normal operation, a WDT time-out generates a device reset. If the device is in *SLEEP* mode, a WDT time-out causes the device to wake-up and continue with normal operation (WDT Wake-up). The WDT can be permanently disabled by clearing configuration bit WDTE (Section 13.1).

### 13.7.1 WDT PERIOD

The WDT has a nominal time-out period of 18 ms, (with no prescaler). The time-out periods vary with temperature, VDD and process variations from part to part (see

DC specs). If longer time-out periods are desired, a prescaler with a division ratio of up to 1:128 can be assigned to the WDT under software control by writing to the OPTION register. Thus, time-out periods up to 2.3 seconds can be realized.

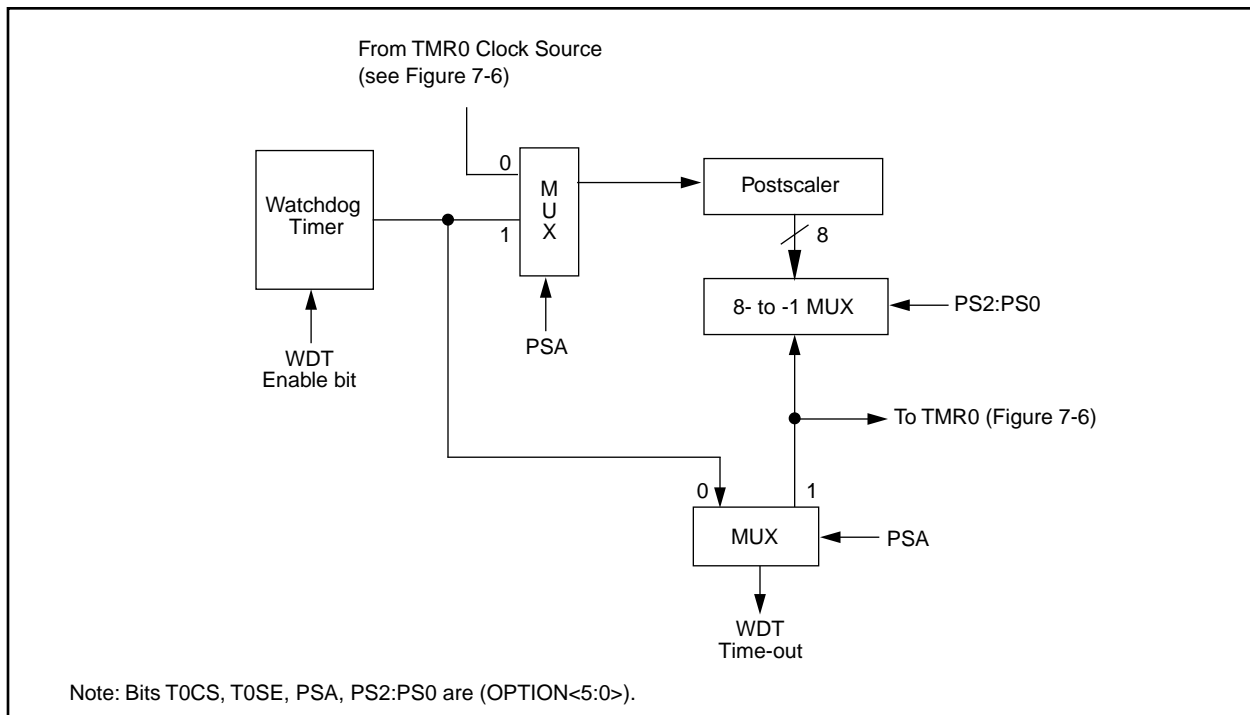
The *CLRWDT* and *SLEEP* instructions clear the WDT and the postscaler, if assigned to the WDT, and prevent it from timing out and generating a device RESET condition.

The  $\overline{TO}$  bit in the STATUS register will be cleared upon a WDT time-out.

### 13.7.2 WDT PROGRAMMING CONSIDERATIONS

It should also be taken in account that under worst case conditions (VDD = Min., Temperature = Max., max. WDT prescaler) it may take several seconds before a WDT time-out occurs.

**FIGURE 13-21: WATCHDOG TIMER BLOCK DIAGRAM**



**FIGURE 13-22: SUMMARY OF WATCHDOG TIMER REGISTERS**

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
2007h	Config. bits	(1)	BODEN <sup>(1)</sup>	CP1	CP0	PWRTE <sup>(1)</sup>	WDTE	FOSC1	FOSC0
81h	OPTION	RBP <sub>U</sub>	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0

Legend: Shaded cells are not used by the Watchdog Timer.

Note 1: See Figure 13-1, Figure 13-2, and Figure 13-3 for details of these bits for the specific device.



## 13.8 Power-down Mode (SLEEP)

Applicable Devices											
61	62	62A	R62	63	64	64A	R64	65	65A		

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

If enabled, the Watchdog Timer will be cleared but keeps running, status bit  $\overline{PD}$  (STATUS<3>) is cleared, status bit  $\overline{TO}$  (STATUS<4>) is set, and the oscillator driver is turned off. The I/O ports maintain the status they had before the SLEEP instruction was executed (driving high, low, or hi-impedance).

For lowest current consumption in this mode, place all I/O pins at either VDD, or VSS, ensure no external circuitry is drawing current from the I/O pin, and disable external clocks. Pull all I/O pins, that are hi-impedance inputs, high or low externally to avoid switching currents caused by floating inputs. The T0CKI input should also be at VDD or VSS for lowest current consumption. The contribution from on-chip pull-ups on PORTB should be considered.

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

### 13.8.1 WAKE-UP FROM SLEEP

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

1. External reset input on  $\overline{MCLR}/VPP$  pin.
2. Watchdog Timer Wake-up (if WDT was enabled).
3. Interrupt from RB0/INT pin, RB port change, or some Peripheral Interrupts.

External  $\overline{MCLR}$  Reset will cause a device reset. All other events are considered a continuation of program execution and cause a "wake-up". The  $\overline{TO}$  and  $\overline{PD}$  bits

in the STATUS register can be used to determine the cause of device reset. The  $\overline{PD}$  bit, which is set on power-up is cleared when SLEEP is invoked. The  $\overline{TO}$  bit is cleared if WDT time-out occurred (and caused wake-up).

The following peripheral interrupts can wake the device from SLEEP:

1. TMR1 interrupt. Timer1 must be operating as an asynchronous counter.
2. SSP (Start/Stop) bit detect interrupt.
3. SSP transmit or receive in slave mode (SPI/I<sup>2</sup>C).
4. CCP capture mode interrupt.
5. Parallel Slave Port read or write.
6. USART TX or RX (synchronous slave mode).

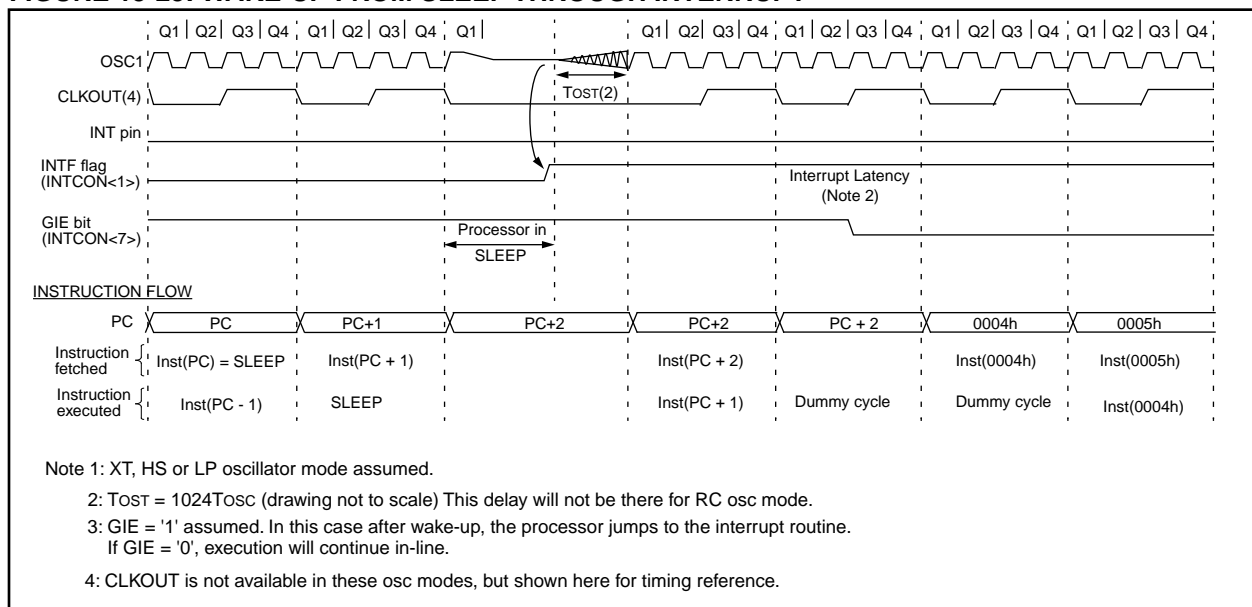
Other peripherals can not generate interrupts since during SLEEP, no on-chip Q clocks are present.

When the SLEEP instruction is being executed, the next instruction (PC + 1) is pre-fetched. For the device to wake-up through an interrupt event, the corresponding interrupt enable bit must be set (enabled). Wake-up is regardless of the state of the GIE bit. If the GIE bit is clear (disabled), the device continues execution at the instruction after the SLEEP instruction. If the GIE bit is set (enabled), the device executes the instruction after the SLEEP instruction and then branches to the interrupt address (0004h). In cases where the execution of the instruction following SLEEP is not desirable, the user should have a NOP after the SLEEP instruction.

**Note:** Interrupts that are capable of waking the device from SLEEP will still set the individual flag bits regardless of the state of the global enable bit, GIE.

The WDT is cleared when the device wakes-up from sleep, regardless of the source of wake-up.

**FIGURE 13-23: WAKE-UP FROM SLEEP THROUGH INTERRUPT**



# PIC16C6X

## 13.9 Program Verification/Code Protection

### Applicable Devices

61	62	62A	R62	63	64	64A	R64	65	65A
----	----	-----	-----	----	----	-----	-----	----	-----

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

**Note:** Microchip does not recommend code protecting windowed devices.

## 13.10 ID Locations

### Applicable Devices

61	62	62A	R62	63	64	64A	R64	65	65A
----	----	-----	-----	----	----	-----	-----	----	-----

Four memory locations (2000h - 2003h) are designated as ID locations where the user can store checksum or other code-identification numbers. These locations are not accessible during normal execution but are readable and writable during program/verify. It is recommended that only the 4 least significant bits of ID location are used.

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

## 13.11 In-Circuit Serial Programming

### Applicable Devices

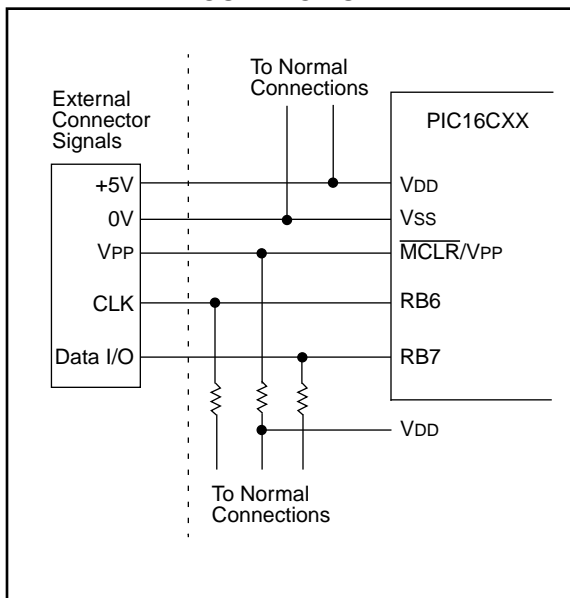
61	62	62A	R62	63	64	64A	R64	65	65A
----	----	-----	-----	----	----	-----	-----	----	-----

The PIC16CXX microcontrollers can be serially programmed while in the end application circuit. This is simply done with two lines for clock and data, and three other lines for power, ground, and the programming voltage. This allows customers to manufacture boards with unprogrammed devices, and then program the microcontroller just before shipping the product. This also allows the most recent firmware or a custom firmware to be programmed.

The device is placed into a program/verify mode by holding pins RB6 and RB7 low while raising the  $\overline{\text{MCLR}}$  (VPP) pin from VIL to VIH (see programming specification). RB6 becomes the programming clock and RB7 becomes the programming data. Both RB6 and RB7 are Schmitt Trigger inputs in this mode.

After reset, to place the device in program/verify mode, the program counter (PC) is at location 00h. A 6-bit command is then supplied to the device. Depending on the command, 14-bits of program data are then supplied to or from the device, depending if the command was a load or a read. For complete details of serial programming, please refer to the PIC16C6X/7X Programming Specifications (Literature #DS30228).

**FIGURE 13-24: TYPICAL IN-CIRCUIT SERIAL PROGRAMMING CONNECTION**



## 14.0 INSTRUCTION SET SUMMARY

Applicable Devices									
61	62	62A	R62	63	64	64A	R64	65	65A

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

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

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

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

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

**TABLE 14-1: OPCODE FIELD DESCRIPTIONS**

Field	Description
f	Register file address (0x00 to 0x7F)
W	Working register (accumulator)
b	Bit address within an 8-bit file register
k	Literal field, constant data or label
x	Don't care location (= 0 or 1) The assembler will generate code with x = 0. It is the recommended form of use for compatibility with all Microchip software tools.
d	Destination select; d = 0: store result in W, d = 1: store result in file register f. Default is d = 1
label	Label name
TOS	Top of Stack
PC	Program Counter
PCLATH	Program Counter High Latch
GIE	Global Interrupt Enable bit
WDT	Watchdog Timer/Counter
TO	Time-out bit
PD	Power-down bit
dest	Destination either the W register or the specified register file location
[ ]	Options
( )	Contents
→	Assigned to
< >	Register bit field
∈	In the set of
<i>italics</i>	User defined term (font is courier)

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

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

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

Table 14-2 lists the instructions recognized by the MPASM assembler.

Figure 14-1 shows the three general formats that the instructions can have.

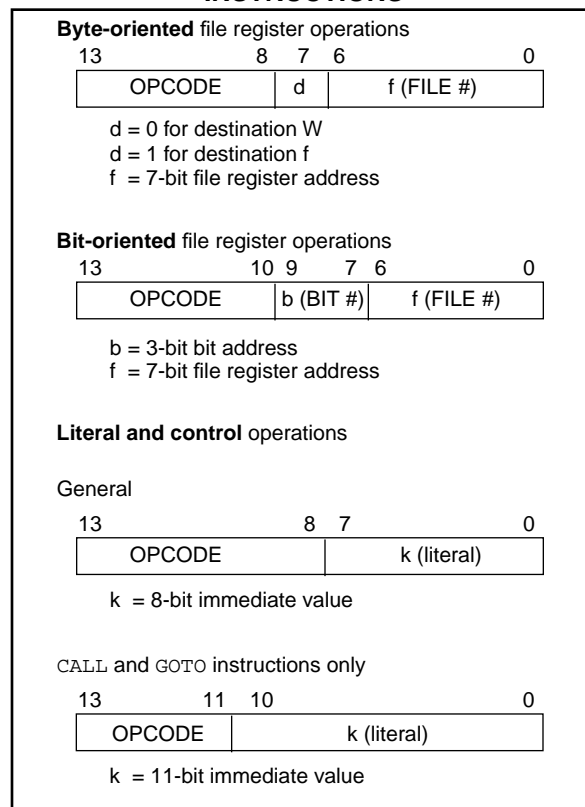
**Note:** To maintain upward compatibility with future PIC16CXX products, do not use the `OPTION` and `TRIS` instructions.

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

0xhh

where h signifies a hexadecimal digit.

**FIGURE 14-1: GENERAL FORMAT FOR INSTRUCTIONS**



# PIC16C6X

**TABLE 14-2: PIC16CXX INSTRUCTION SET**

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

- Note 1: When an I/O register is modified as a function of itself ( e.g., `MOVF PORTB, 1`), the value used will be that value present on the pins themselves. For example, if the data latch is '1' for a pin configured as input and is driven low by an external device, the data will be written back with a '0'.
- 2: If this instruction is executed on the TMR0 register (and, where applicable, d = 1), the prescaler will be cleared if assigned to the Timer0 Module.
- 3: If Program Counter (PC) is modified or a conditional test is true, the instruction requires two cycles. The second cycle is executed as a NOP.

## 14.1 Instruction Descriptions

### ADDLW Add Literal and W

Syntax:	[label] ADDLW k				
Operands:	$0 \leq k \leq 255$				
Operation:	$(W) + k \rightarrow (W)$				
Status Affected:	C, DC, Z				
Encoding:	<table><tr><td>11</td><td>111x</td><td>kkkk</td><td>kkkk</td></tr></table>	11	111x	kkkk	kkkk
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				
Example	<pre>ADDLW    0x15</pre> <div>Before Instruction</div> <div>W = 0x10</div> <div>After Instruction</div> <div>W = 0x25</div>				

### ANDLW AND Literal with W

Syntax:	[label] ANDLW k				
Operands:	$0 \leq k \leq 255$				
Operation:	(W) .AND. (k) $\rightarrow$ (W)				
Status Affected:	Z				
Encoding:	<table border="1"><tr><td>11</td><td>1001</td><td>kkkk</td><td>kkkk</td></tr></table>	11	1001	kkkk	kkkk
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				
Example	<pre>ANDLW    0x5F</pre> <p>Before Instruction</p> <p>W = 0xA3</p> <p>After Instruction</p> <p>W = 0x03</p>				

### ADDWF Add W and f

Syntax:	[label] ADDWF f,d												
Operands:	$0 \leq f \leq 127$ $d \in [0,1]$												
Operation:	$(W) + (f) \rightarrow (dest)$												
Status Affected:	C, DC, Z												
Encoding:	<table><tr><td>00</td><td>0111</td><td>dfff</td><td>ffff</td></tr></table>	00	0111	dfff	ffff								
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												
Example	<pre>ADDWF    FSR, 0</pre> <p>Before Instruction</p> <table><tr><td>W</td><td>=</td><td>0x17</td></tr><tr><td>FSR</td><td>=</td><td>0xC2</td></tr></table> <p>After Instruction</p> <table><tr><td>W</td><td>=</td><td>0xD9</td></tr><tr><td>FSR</td><td>=</td><td>0xC2</td></tr></table>	W	=	0x17	FSR	=	0xC2	W	=	0xD9	FSR	=	0xC2
W	=	0x17											
FSR	=	0xC2											
W	=	0xD9											
FSR	=	0xC2											

### ANDWF AND W with f

Syntax:	[label] ANDWF f,d			
Operands:	$0 \leq f \leq 127$ $d \in [0,1]$			
Operation:	(W) .AND. (f) $\rightarrow$ (dest)			
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			
Example	ANDWF FSR, 1			
	Before Instruction			
	W	=	0x17	
	FSR	=	0xC2	
	After Instruction			
	W	=	0x17	
	FSR	=	0x02	

# PIC16C6X

BCF		Bit Clear f			
Syntax:	[label] BCF f,b				
Operands:	$0 \leq f \leq 127$ $0 \leq b \leq 7$				
Operation:	$0 \rightarrow (f<b>)$				
Status Affected:	None				
Encoding:	01	00bb	bfff	ffff	
Description:	Bit 'b' in register 'f' is cleared.				
Words:	1				
Cycles:	1				
Example	BCF FLAG_REG, 7				
	Before Instruction				
	FLAG_REG = 0xC7				
	After Instruction				
	FLAG_REG = 0x47				

BSF		Bit Set f			
Syntax:	[label] BSF f,b				
Operands:	$0 \leq f \leq 127$ $0 \leq b \leq 7$				
Operation:	$1 \rightarrow (f < b)$				
Status Affected:	None				
Encoding:	01	01bb	bfff	ffff	
Description:	Bit 'b' in register 'f' is set.				
Words:	1				
Cycles:	1				
Example	BSF FLAG_REG, 7				
	Before Instruction				
	FLAG_REG = 0x0A				
	After Instruction				
	FLAG_REG = 0x8A				

BTFSC		Bit Test, Skip if Clear			
Syntax:	[label] BTFSC f,b				
Operands:	0 ≤ f ≤ 127 0 ≤ b ≤ 7				
Operation:	skip if (f<b) = 0				
Status Affected:	None				
Encoding:	01	10bb	bfff	ffff	
Description:	If bit 'b' in register 'f' is '0' then the next instruction is skipped. If bit 'b' is '0' then the next instruction fetched during the current instruction execution is discarded, and a NOP is executed instead, making this a 2 cycle instruction.				
Words:	1				
Cycles:	1(2)				
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				

BTFSS		Bit Test f, Skip if Set			
Syntax:	[label] BTFSS f,b				
Operands:	$0 \leq f \leq 127$ $0 \leq b < 7$				
Operation:	skip if (f<b>) = 1				
Status Affected:	None				
Encoding:	01	11bb	bfff	ffff	
Description:	If bit 'b' in register 'f' is '1' then the next instruction is skipped. If bit 'b' is '1', then the next instruction fetched during the current instruction execution, is discarded and a NOP is executed instead, making this a 2 cycle instruction.				
Words:	1				
Cycles:	1(2)				
Example	HERE	BTFSC	FLAG,1		
	FALSE	GOTO	PROCESS_CODE		
	TRUE	.			
		.			
		.			
Before Instruction					
	PC = address HERE				
After Instruction					
	if FLAG<1> = 0,				
	PC = address FALSE				
	if FLAG<1> = 1,				
	PC = address TRUE				

CALL		Call Subroutine							
Syntax:	[ <i>label</i> ] CALL k								
Operands:	0 ≤ k ≤ 2047								
Operation:	(PC)+ 1 → TOS, k → PC<10:0>, (PCLATH<4:3>) → PC<12:11>								
Status Affected:	None								
Encoding:	<table><tr><td>10</td><td>0kkk</td><td>kkkk</td><td>kkkk</td></tr></table>					10	0kkk	kkkk	kkkk
10	0kkk	kkkk	kkkk						
Description:	Call Subroutine. First, return address (PC+1) is pushed onto the stack. The eleven bit immediate address is loaded into PC bits <10:0>. The upper bits of the PC are loaded from PCLATH. CALL is a two cycle instruction.								
Words:	1								
Cycles:	2								
Example	HERE                  CALL                  THERE								
	Before Instruction								
	PC = Address HERE								
	After Instruction								
	PC = Address THERE								
	TOS = Address HERE+1								

CLRF		Clear f							
Syntax:	[label] CLRF f								
Operands:	$0 \leq f \leq 127$								
Operation:	$00h \rightarrow (f)$ $1 \rightarrow Z$								
Status Affected:	Z								
Encoding:	<table><tr><td>00</td><td>0001</td><td>1fff</td><td>ffff</td></tr></table>					00	0001	1fff	ffff
00	0001	1fff	ffff						
Description:	The contents of register 'f' are cleared and the Z bit is set.								
Words:	1								
Cycles:	1								
Example	<div>CLRF                      FLAG_REG</div> <div>Before Instruction</div> <div>                    FLAG_REG    =    0x5A</div> <div>After Instruction</div> <div>                    FLAG_REG    =    0x00</div> <div>                    Z                =    1</div>								

CLR W					
Syntax:	[ <i>label</i> ] CLRW				
Operands:	None				
Operation:	00h → (W) 1 → Z				
Status Affected:	Z				
Encoding:	<table><tr><td>00</td><td>0001</td><td>0xxx</td><td>xxxx</td></tr></table>	00	0001	0xxx	xxxx
00	0001	0xxx	xxxx		
Description:	W register is cleared. Zero bit (Z) is set.				
Words:	1				
Cycles:	1				
Example	<pre>CLR W</pre> <p>Before Instruction</p> <p>W = 0x5A</p> <p>After Instruction</p> <p>W = 0x00</p> <p>Z = 1</p>				

# PIC16C6X

## CLRWDT Clear Watchdog Timer

Syntax:	[ <i>label</i> ] CLRWDT				
Operands:	None				
Operation:	00h → WDT 0 → WDT prescaler, 1 → $\overline{TO}$ 1 → $\overline{PD}$				
Status Affected:	$\overline{TO}$ , $\overline{PD}$				
Encoding:	<table><tr><td>00</td><td>0000</td><td>0110</td><td>0100</td></tr></table>	00	0000	0110	0100
00	0000	0110	0100		
Description:	CLRWDT instruction resets the Watchdog Timer. It also resets the prescaler of the WDT. Status bits $\overline{TO}$ and $\overline{PD}$ are set.				
Words:	1				
Cycles:	1				
Example	CLRWDT				

Before Instruction  
WDT counter = ?  
After Instruction  
WDT counter = 0x00  
WDT prescaler = 0  
 $\overline{TO}$  = 1  
 $\overline{PD}$  = 1

## COMF Complement f

Syntax:	[ <i>label</i> ] COMF f,d			
Operands:	$0 \leq f \leq 127$ $d \in [0,1]$			
Operation:	$(\bar{f}) \rightarrow (\text{dest})$			
Status Affected:	Z			
Encoding:	00	1001	dfff	ffff
Description:	The contents of register 'f' are complemented. If 'd' is 0 the result is stored in W. If 'd' is 1 the result is stored back in register 'f'.			
Words:	1			
Cycles:	1			
Example	COMF REG1,0			

Before Instruction  
REG1 = 0x13  
After Instruction  
REG1 = 0x13  
W = 0xEC

## DECF Decrement f

Syntax:	[ <i>label</i> ] DECF f,d			
Operands:	$0 \leq f \leq 127$ $d \in [0,1]$			
Operation:	(f) - 1 $\rightarrow$ (dest)			
Status Affected:	Z			
Encoding:	00	0011	dfff	ffff
Description:	Decrement register 'f'. If 'd' is 0 the result is stored in the W register. If 'd' is 1 the result is stored back in register 'f'.			
Words:	1			
Cycles:	1			
Example	DECF CNT, 1			

Before Instruction  
CNT = 0x01  
Z = 0  
After Instruction  
CNT = 0x00  
Z = 1

## DECFSZ Decrement f, Skip if 0

Syntax:	[ <i>label</i> ] DECFSZ f,d			
Operands:	$0 \leq f \leq 127$ $d \in [0,1]$			
Operation:	(f) - 1 $\rightarrow$ (dest); skip if result = 0			
Status Affected:	None			
Encoding:	00	1011	dfff	ffff
Description:	<p>The contents of register 'f' are decremented. If 'd' is 0 the result is placed in the W register. If 'd' is 1 the result is placed back in register 'f'.</p> <p>If the result is 0, the next instruction, which is already fetched, is discarded. A NOP is executed instead making it a two cycle instruction.</p>			
Words:	1			

Before Instruction  
PC = address HERE  
After Instruction  
CNT = CNT - 1  
if CNT = 0,  
PC = address CONTINUE  
if CNT ≠ 0,  
PC = address HERE+1



GOTO		Unconditional Branch							
Syntax:	[ <i>label</i> ] GOTO k								
Operands:	$0 \leq k \leq 2047$								
Operation:	$k \rightarrow PC<10:0>$ $PCLATH<4:3> \rightarrow PC<12:11>$								
Status Affected:	None								
Encoding:	<table><tr><td>10</td><td>1kkk</td><td>kkkk</td><td>kkkk</td></tr></table>					10	1kkk	kkkk	kkkk
10	1kkk	kkkk	kkkk						
Description:	GOTO is an unconditional branch. The eleven bit immediate value is loaded into PC bits <10:0>. The upper bits of PC are loaded from PCLATH<4:3>. GOTO is a two cycle instruction.								
Words:	1								
Cycles:	2								
Example	GOTO THERE								
	After Instruction								
	PC = Address THERE								

INCFSZ		Increment f, Skip if 0					
Syntax:	[ <i>label</i> ] INCFSZ f,d						
Operands:	$0 \leq f \leq 127$ $d \in [0,1]$						
Operation:	$(f) + 1 \rightarrow (\text{dest}), \text{ skip if result} = 0$						
Status Affected:	None						
Encoding:	<table><tr><td>00</td><td>1111</td><td>dfff</td><td>ffff</td></tr></table>			00	1111	dfff	ffff
00	1111	dfff	ffff				
Description:	<p>The contents of register 'f' are incremented. If 'd' is 0 the result is placed in the W register. If 'd' is 1 the result is placed back in register 'f'. If the result is 0, the next instruction, which is already fetched, is discarded. A NOP is executed instead making it a two cycle instruction.</p>						
Words:	1						
Cycles:	1(2)						
Example	HERE	INCFSZ	CNT, 1				

INCF	Increment f												
Syntax:	[ <i>label</i> ] INCF f,d												
Operands:	$0 \leq f \leq 127$ $d \in [0,1]$												
Operation:	$(f) + 1 \rightarrow (\text{dest})$												
Status Affected:	Z												
Encoding:	<table><tr><td>00</td><td>1010</td><td>dfff</td><td>ffff</td></tr></table>	00	1010	dfff	ffff								
00	1010	dfff	ffff										
Description:	The contents of register 'f' are incremented. If 'd' is 0 the result is placed in the W register. If 'd' is 1 the result is placed back in register 'f'.												
Words:	1												
Cycles:	1												
Example	<pre>INCF      CNT, 1</pre> <p>Before Instruction</p> <table><tr><td>CNT</td><td>=</td><td>0xFF</td></tr><tr><td>Z</td><td>=</td><td>0</td></tr></table> <p>After Instruction</p> <table><tr><td>CNT</td><td>=</td><td>0x00</td></tr><tr><td>Z</td><td>=</td><td>1</td></tr></table>	CNT	=	0xFF	Z	=	0	CNT	=	0x00	Z	=	1
CNT	=	0xFF											
Z	=	0											
CNT	=	0x00											
Z	=	1											

IORLW		Inclusive OR Literal with W						
Syntax:	[ <i>label</i> ] IORLW k							
Operands:	$0 \leq k \leq 255$							
Operation:	$(W) .OR. k \rightarrow (W)$							
Status Affected:	Z							
Encoding:	<table border="1"><tr><td>11</td><td>1000</td><td>kkkk</td><td>kkkk</td></tr></table>				11	1000	kkkk	kkkk
11	1000	kkkk	kkkk					
Description:	The contents of the W register is OR'ed with the eight bit literal 'k'. The result is placed in the W register.							
Words:	1							
Cycles:	1							
Example	<pre>IORLW      0x35</pre> <div>Before Instruction</div> <div>W = 0x9A</div> <div>After Instruction</div> <div>W = 0xBF</div> <div>Z = 1</div>							

# PIC16C6X

## IORWF Inclusive OR W with f

Syntax: [ *label* ] IORWF f,d

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

Operation: (W) .OR. (f)  $\rightarrow$  (dest)

Status Affected: Z

Encoding: 

00	0100	dfff	ffff
----	------	------	------

Description: Inclusive OR the W register with register 'f'. 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

Example IORWF RESULT, 0

Before Instruction

RESULT = 0x13

W = 0x91

After Instruction

RESULT = 0x13

W = 0x93

Z = 1

## MOVF Move f

Syntax: [ *label* ] MOVF f,d

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

Operation: (f)  $\rightarrow$  (dest)

Status Affected: Z

Encoding: 

00	1000	dfff	ffff
----	------	------	------

Description: The contents of register f is moved to a destination dependant upon the status of d. If d = 0, destination is W register. If d = 1, the destination is file register f itself. d = 1 is useful to test a file register since status flag Z is affected.

Words: 1

Cycles: 1

Example MOVF FSR, 0

After Instruction

W = value in FSR register

Z = 1

## MOVLW Move Literal to W

Syntax: [ *label* ] MOVLW k

Operands:  $0 \leq k \leq 255$

Operation: k  $\rightarrow$  (W)

Status Affected: None

Encoding: 

11	00xx	kkkk	kkkk
----	------	------	------

Description: The eight bit literal 'k' is loaded into W register. The don't cares will assemble as 0's.

Words: 1

Cycles: 1

Example MOVLW 0x5A

After Instruction

W = 0x5A

## MOVWF Move W to f

Syntax: [ *label* ] MOVWF f

Operands:  $0 \leq f \leq 127$

Operation: (W)  $\rightarrow$  (f)

Status Affected: None

Encoding: 

00	0000	1fff	ffff
----	------	------	------

Description: Move data from W register to register 'f'.

Words: 1

Cycles: 1

Example MOVWF OPTION

Before Instruction

OPTION = 0xFF

W = 0x4F

After Instruction

OPTION = 0x4F

W = 0x4F

## 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

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

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.**

## RETLW Return with Literal in W

Syntax: [ *label* ] RETLW k

Operands:  $0 \leq k \leq 255$

Operation: k → (W);  
TOS → PC

Status Affected: None

Encoding: 

11	01xx	kkkk	kkkk
----	------	------	------

Description: The W register is loaded with the eight bit literal 'k'. The program counter is loaded from the top of the stack (the return address). This is a two cycle instruction.

Words: 1

Cycles: 2

Example

```
CALL TABLE ;W contains table
              ;offset value
              ;W now has table value
.
.
.
TABLE ADDWF PC ;W = offset
      RETLW k1 ;Begin table
      RETLW k2 ;
      .
      .
      .
      RETLW kn ; End of table
```

Before Instruction

W = 0x07

After Instruction

W = value of k8

# PIC16C6X

## RETURN Return from Subroutine

Syntax: [ *label* ] RETURN

Operands: None

Operation: TOS → PC

Status Affected: None

Encoding: 

00	0000	0000	1000
----	------	------	------

Description: Return from subroutine. The stack is POPed and the top of the stack (TOS) is loaded into the program counter. This is a two cycle instruction.

Words: 1

Cycles: 2

Example 

```
RETURN
```

  
After Interrupt  
PC = TOS

## RRF Rotate Right f through Carry

Syntax: [ *label* ] RRF f,d

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

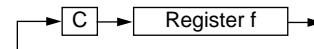
Operation: See description below

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

Example 

```
RRF REG1,0
```

Before Instruction

REG1 = 1110 0110  
C = 0

After Instruction

REG1 = 1110 0110  
W = 0111 0011  
C = 0

## RLF Rotate Left f through Carry

Syntax: [ *label* ] RLF f,d

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

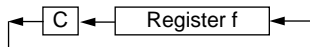
Operation: See description below

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

Example 

```
RLF REG1,0
```

  
Before Instruction  
REG1 = 1110 0110  
C = 0  
After Instruction  
REG1 = 1110 0110  
W = 1100 1100  
C = 1

## SLEEP

Syntax: [ *label* ] SLEEP

Operands: None

Operation: 00h → WDT,  
0 → WDT prescaler,  
1 →  $\overline{\text{TO}}$ ,  
0 →  $\overline{\text{PD}}$

Status Affected:  $\overline{\text{TO}}$ ,  $\overline{\text{PD}}$

Encoding: 

00	0000	0110	0011
----	------	------	------

Description: The power-down status bit,  $\overline{\text{PD}}$  is cleared. Time-out status bit,  $\overline{\text{TO}}$  is set. Watchdog Timer and its prescaler are cleared. The processor is put into SLEEP mode with the oscillator stopped. See Section 13.8 for more details.

Words: 1

Cycles: 1

Example 

```
SLEEP
```

## SUBLW Subtract W from Literal

Syntax: [ *label* ] SUBLW *k*

Operands:  $0 \leq k \leq 255$

Operation:  $k - (W) \rightarrow (W)$

Status C, DC, Z

Affected:

Encoding: 

11	110x	kkkk	kkkk
----	------	------	------

Description: The W register is subtracted (2's complement method) from the eight bit literal 'k'. The result is placed in the W register.

Words: 1

Cycles: 1

Example 1: SUBLW 0x02

Before Instruction

W = 1  
C = ?

After Instruction

W = 1  
C = 1; result is positive

Example 2: Before Instruction

W = 2  
C = ?

After Instruction

W = 0  
C = 1; result is zero

Example 3: Before Instruction

W = 3  
C = ?

After Instruction

W = 0xFF  
C = 0; result is negative

## SUBWF Subtract W from f

Syntax: [ *label* ] SUBWF *f*,*d*

Operands:  $0 \leq f \leq 127$

$d \in [0,1]$

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

Status C, DC, Z

Affected:

Encoding: 

00	0010	ffff	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

Example 1: SUBWF REG1, 1

Before Instruction

REG1 = 3  
W = 2  
C = ?

After Instruction

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

Example 2: Before Instruction

REG1 = 2  
W = 2  
C = ?

After Instruction

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

Example 3: Before Instruction

REG1 = 1  
W = 2  
C = ?

After Instruction

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

# PIC16C6X

SWAPF		Swap Nibbles in f							
Syntax:	[ <i>label</i> ] SWAPF f,d								
Operands:	$0 \leq f \leq 127$ $d \in [0,1]$								
Operation:	$(f<3:0>) \rightarrow (dest<7:4>),$ $(f<7:4>) \rightarrow (dest<3:0>)$								
Status Affected:	None								
Encoding:	<table><tr><td>00</td><td>1110</td><td>dfff</td><td>ffff</td></tr></table>					00	1110	dfff	ffff
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								
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) → TRIS register f;				
Status Affected:	None				
Encoding:	<table><tr><td>00</td><td>0000</td><td>0110</td><td>0fff</td></tr></table>	00	0000	0110	0fff
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	<div>To maintain upward compatibility with future PIC16CXX products, do not use this instruction.</div>				

XORLW	Exclusive OR Literal with W				
Syntax:	[label] XORLW k				
Operands:	0 ≤ k ≤ 255				
Operation:	(W) .XOR. k → (W)				
Status Affected:	Z				
Encoding:	<table><tr><td>11</td><td>1010</td><td>kkkk</td><td>kkkk</td></tr></table>	11	1010	kkkk	kkkk
11	1010	kkkk	kkkk		
Description:	The contents of the W register are XOR'ed with the eight bit literal 'k'. The result is placed in the W register.				
Words:	1				
Cycles:	1				
Example:	XORLW 0xAF				
	Before Instruction				
	W = 0xB5				
	After Instruction				
	W = 0x1A				

XORWF	Exclusive OR W with f												
Syntax:	[label] XORWF f,d												
Operands:	$0 \leq f \leq 127$ $d \in [0,1]$												
Operation:	(W) .XOR. (f) $\rightarrow$ (dest)												
Status Affected:	Z												
Encoding:	<table><tr><td>00</td><td>0110</td><td>dfff</td><td>ffff</td></tr></table>	00	0110	dfff	ffff								
00	0110	dfff	ffff										
Description:	Exclusive OR 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												
Example	<pre>XORWF REG 1</pre> <p>Before Instruction</p> <table><tr><td>REG</td><td>=</td><td>0xAF</td></tr><tr><td>W</td><td>=</td><td>0xB5</td></tr></table> <p>After Instruction</p> <table><tr><td>REG</td><td>=</td><td>0x1A</td></tr><tr><td>W</td><td>=</td><td>0xB5</td></tr></table>	REG	=	0xAF	W	=	0xB5	REG	=	0x1A	W	=	0xB5
REG	=	0xAF											
W	=	0xB5											
REG	=	0x1A											
W	=	0xB5											

## 15.0 DEVELOPMENT SUPPORT

### 15.1 Development Tools

The PIC16/17 microcontrollers are supported with a full range of hardware and software development tools:

- PICMASTER® Real-Time In-Circuit Emulator
- PRO MATE™ Universal Programmer
- PICSTART® Low-Cost Prototype Programmer
- PICDEM-1 Low-Cost Demonstration Board
- PICDEM-2 Low-Cost Demonstration Board
- MPASM Assembler
- MPSIM Software Simulator
- C Compiler (MP-C)
- Fuzzy logic development system (*fuzzyTECH*®-MP)

### 15.2 PICMASTER: High Performance Universal In-Circuit Emulator with MPLAB IDE

The PICMASTER Universal In-Circuit Emulator is intended to provide the product development engineer with a complete microcontroller design tool set for all microcontrollers in the PIC16C5X, PIC16CXX and PIC17CXX families. PICMASTER is supplied with the MPLAB™ Integrated Development Environment (IDE), which allows editing, "make" and download, and source debugging from a single environment. A PICMASTER System configuration is shown in Figure 15-1.

Interchangeable target probes allow the system to be easily reconfigured for emulation of different processors. The universal architecture of the PICMASTER allows expansion to support all new PIC16C5X, PIC16CXX and PIC17CXX microcontrollers.

The PICMASTER Emulator System has been designed as a real-time emulation system with advanced features that are generally found on more expensive development tools. The PC compatible 386 (and better) machine platform and Microsoft Windows™ 3.x environment were chosen to best make these features available to you, the end user.

The PICMASTER Universal Emulator System consists primarily of four major components:

- Host-Interface Card
- Emulator Control Pod
- Target-Specific Emulator Probe
- PC-Host Emulation Control Software

The Windows operating system allows the developer to take full advantage of the many powerful features and functions of the PICMASTER system.

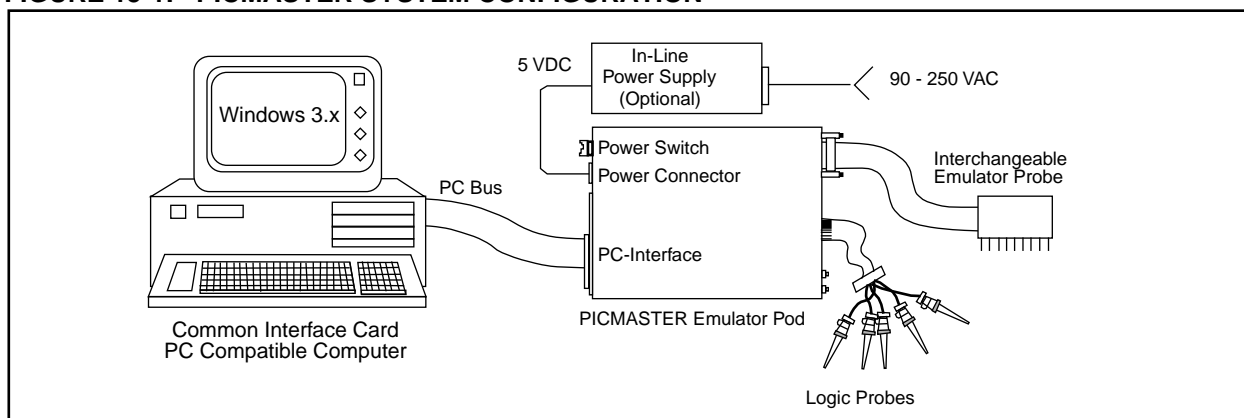
PICMASTER emulation can operate in one window, while a text editor is running in a second window.

PC-Host Emulation Control software takes full advantage of Dynamic Data Exchange (DDE), a feature of Windows. DDE allows data to be dynamically transferred between two or more Windows programs. With this feature, data collected with PICMASTER can be automatically transferred to a spreadsheet or database program for further analysis.

Under Windows, as many as four PICMASTER emulators can be run simultaneously from the same PC making development of multi-microcontroller systems possible (e.g., a system containing a PIC16CXX microcontroller and a PIC17CXX microcontroller).

The PICMASTER probes specifications are shown in Table 15-1.

**FIGURE 15-1: PICMASTER SYSTEM CONFIGURATION**



# PIC16C6X

**TABLE 15-1: PICMASTER PROBE SPECIFICATION**

Devices	PICMASTER PROBE	PROBE	
		Maximum Frequency	Operating Voltage
PIC16C54	PROBE-16D	20 MHz	4.5V - 5.5V
PIC16C54A	PROBE-16D	20 MHz	4.5V - 5.5V
PIC16CR54	PROBE-16D	20 MHz	4.5V - 5.5V
PIC16CR54A	PROBE-16D <sup>(1)</sup>	20 MHz	4.5V - 5.5V
PIC16CR54B	PROBE-16D <sup>(1)</sup>	20 MHz	4.5V - 5.5V
PIC16C55	PROBE-16D	20 MHz	4.5V - 5.5V
PIC16CR55	PROBE-16D <sup>(1)</sup>	20 MHz	4.5V - 5.5V
PIC16C56	PROBE-16D	20 MHz	4.5V - 5.5V
PIC16CR56	PROBE-16D <sup>(1)</sup>	20 MHz	4.5V - 5.5V
PIC16C57	PROBE-16D	20 MHz	4.5V - 5.5V
PIC16CR57A	PROBE-16D	20 MHz	4.5V - 5.5V
PIC16CR57B	PROBE-16D <sup>(1)</sup>	20 MHz	4.5V - 5.5V
PIC16C58A	PROBE-16D	20 MHz	4.5V - 5.5V
PIC16CR58A	PROBE-16D	20 MHz	4.5V - 5.5V
PIC16CR58B	PROBE-16D <sup>(1)</sup>	20 MHz	4.5V - 5.5V
PIC16C61	PROBE-16G	10 MHz	4.5V - 5.5V
PIC16C62	PROBE-16E	10 MHz	4.5V - 5.5V
PIC16C62A	PROBE-16E <sup>(1)</sup>	10 MHz	4.5V - 5.5V
PIC16CR62	PROBE-16E <sup>(1)</sup>	10 MHz	4.5V - 5.5V
PIC16C63	PROBE-16F <sup>(1)</sup>	10 MHz	4.5V - 5.5V
PIC16C64	PROBE-16E	10 MHz	4.5V - 5.5V
PIC16C64A	PROBE-16E <sup>(1)</sup>	10 MHz	4.5V - 5.5V

**TABLE 15-1: PICMASTER PROBE SPECIFICATION CONT.)**

Devices	PICMASTER PROBE	PROBE	
		Maximum Frequency	Operating Voltage
PIC16CR64	PROBE-16E <sup>(1)</sup>	10 MHz	4.5V - 5.5V
PIC16C65	PROBE-16F	10 MHz	4.5V - 5.5V
PIC16C65A	PROBE-16F <sup>(1)</sup>	10 MHz	4.5V - 5.5V
PIC16C620	PROBE-16H	10 MHz	4.5V - 5.5V
PIC16C621	PROBE-16H	10 MHz	4.5V - 5.5V
PIC16C622	PROBE-16H	10 MHz	4.5V - 5.5V
PIC16C70	PROBE-16B <sup>(1)</sup>	10 MHz	4.5V - 5.5V
PIC16C71	PROBE-16B	10 MHz	4.5V - 5.5V
PIC16C71A	PROBE-16B <sup>(1)</sup>	10 MHz	4.5V - 5.5V
PIC16C72	PROBE-16F <sup>(1)</sup>	10 MHz	4.5V - 5.5V
PIC16C73	PROBE-16F	10 MHz	4.5V - 5.5V
PIC16C73A	PROBE-16F <sup>(1)</sup>	10 MHz	4.5V - 5.5V
PIC16C74	PROBE-16F	10 MHz	4.5V - 5.5V
PIC16C74A	PROBE-16F <sup>(1)</sup>	10 MHz	4.5V - 5.5V
PIC16C83	PROBE-16C	10 MHz	4.5V - 5.5V
PIC16C84	PROBE-16C	10 MHz	4.5V - 5.5V
PIC17C42	PROBE-17B	20 MHz	4.5V - 5.5V
PIC17C43	PROBE-17B	20 MHz	4.5V - 5.5V
PIC17C44	PROBE-17B	20 MHz	4.5V - 5.5V

Note 1: This PICMASTER probe can be used to functionally emulate the device listed in the previous column. Contact your Microchip sales office for details.



## 15.3 PRO MATE: Universal Programmer

The PRO MATE Universal Programmer is a full-featured programmer capable of operating in stand-alone mode as well as PC-hosted mode.

The PRO MATE has programmable VDD and VPP supplies which allows it to verify programmed memory at VDD min and VDD max for maximum reliability. It has an LCD display for displaying error messages, keys to enter commands and a modular detachable socket assembly to support various package types. In stand-alone mode the PRO MATE can read, verify or program PIC16C5X, PIC16CXX and PIC17CXX devices. It can also set configuration and code-protect bits in this mode.

In PC-hosted mode, the PRO MATE connects to the PC via one of the COM (RS-232) ports. PC based user-interface software makes using the programmer simple and efficient. The user interface is full-screen and menu-based. Full screen display and editing of data, easy selection of bit configuration and part type, easy selection of VDD min, VDD max and VPP levels, load and store to and from disk files (Intel® hex format) are some of the features of the software. Essential commands such as read, verify, program and blank check can be issued from the screen. Additionally, serial programming support is possible where each part is programmed with a different serial number, sequential or random.

The PRO MATE has a modular “programming socket module”. Different socket modules are required for different processor types and/or package types.

PRO MATE supports all PIC16C5X, PIC16CXX and PIC17CXX processors.

## 15.4 PICSTART Low-Cost Development System

The PICSTART programmer is an easy-to-use, very low-cost prototype programmer. It connects to the PC via one of the COM (RS-232) ports. PC-based user interface software makes using the programmer simple and efficient. The user interface is full-screen and menu-based. PICSTART is not recommended for production programming.

## 15.5 PICDEM-1 Low-Cost PIC16/17 Demonstration Board

The PICDEM-1 is a simple board which demonstrates the capabilities of several of Microchip's microcontrollers. The microcontrollers supported are: PIC16C5X (PIC16C54 to PIC16C58A), PIC16C61, PIC16C62X, PIC16C71, PIC16C8X, PIC17C42, PIC17C43 and PIC17C44. All necessary hardware and software is included to run basic demo programs. The users can program the sample microcontrollers provided with the PICDEM-1 board, on a PRO MATE or PICSTART-16B programmer, and easily test firmware. The user can also connect the PICDEM-1 board to the PICMASTER emulator and download the firmware to the emulator for testing. Additional prototype area is available for the user to build some additional hardware and connect it to the microcontroller socket(s). Some of the features include an RS-232 interface, a potentiometer for simulated analog input, push-button switches and eight LEDs connected to PORTB.

## 15.6 PICDEM-2 Low-Cost PIC16CXX Demonstration Board

The PICDEM-2 is a simple demonstration board that supports the PIC16C62, PIC16C64, PIC16C65, PIC16C73 and PIC16C74 microcontrollers. All the necessary hardware and software is included to run the basic demonstration programs. The user can program the sample microcontrollers provided with the PICDEM-2 board, on a PRO MATE programmer or PICSTART-16C, and easily test firmware. The PICMASTER emulator may also be used with the PICDEM-2 board to test firmware. Additional prototype area has been provided to the user for adding additional hardware and connecting it to the microcontroller socket(s). Some of the features include a RS-232 interface, push-button switches, a potentiometer for simulated analog input, a Serial EEPROM to demonstrate usage of the I<sup>2</sup>C bus and separate headers for connection to an LCD module and a keypad.

## 15.7 MPLAB Integrated Development Environment Software.

The MPLAB IDE Software brings an ease of software development previously unseen in the 8-bit microcontroller market. MPLAB is a windows based application which contains:

- A full featured editor
- Three operating modes
  - editor
  - emulator
  - simulator
- A project manager
- Customizable tool bar and key mapping
- A status bar with project information
- Extensive on-line help

MPLAB allows you to:

- Edit your source files (either assembly or "C")
- One touch assemble (or compile) and download to PIC16/17 tools (automatically updates all project information)
- Debug using:
  - source files
  - absolute listing file
- Transfer data dynamically via DDE (soon to be replaced by OLE)
- Run up to four emulators on the same PC

The ability to use MPLAB with Microchip's simulator allows a consistent platform and the ability to easily switch from the low cost simulator to the full featured emulator with minimal retraining due to development tools.

## 15.8 Assembler (MPASM)

The MPASM Cross Assembler is a PC-hosted symbolic assembler. It supports all microcontroller series including the PIC16C5X, PIC16CXX, and PIC17CXX families.

MPASM offers full featured Macro capabilities, conditional assembly, and several source and listing formats. It generates various object code formats to support Microchip's development tools as well as third party programmers.

MPASM allows full symbolic debugging from the Microchip Universal Emulator System (PICMASTER).

MPASM has the following features to assist in developing software for specific use applications.

- Provides translation of Assembler source code to object code for all Microchip microcontrollers.
- Macro assembly capability.
- Produces all the files (Object, Listing, Symbol, and special) required for symbolic debug with Microchip's emulator systems.
- Supports Hex (default), Decimal and Octal source and listing formats.

MPASM provides a rich directive language to support programming of the PIC16/17. Directives are helpful in making the development of your assemble source code shorter and more maintainable.

- **Data Directives** are those that control the allocation of memory and provide a way to refer to data items symbolically, i.e., by meaningful names.
- **Control Directives** control the MPASM listing display. They allow the specification of titles and subtitles, page ejects and other listing control. This eases the readability of the printed output file.
- **Conditional Directives** permit sections of conditionally assembled code. This is most useful where additional functionality may wished to be added depending on the product (less functionality for the low end product, then for the high end product). Also this is very helpful in the debugging of a program.
- **Macro Directives** control the execution and data allocation within macro body definitions. This makes very simple the re-use of functions in a program as well as between programs.

## 15.9 Software Simulator (MPLAB-SIM)

The MPLAB-SIM Software Simulator allows code development in a PC host environment. It allows the user to simulate the PIC16/17 series microcontrollers on an instruction level. On any given instruction, the user may examine or modify any of the data areas or provide external stimulus to any of the pins. The input/output radix can be set by the user and the execution can be performed in; single step, execute until break, or in a trace mode. MPLAB-SIM fully supports symbolic debugging using MP-C and MPASM. The Software Simulator offers the low cost flexibility to develop and debug code outside of the laboratory environment making it an excellent multi-project software development tool.

## 15.10 C Compiler (MP-C)

The MP-C Code Development System is a complete 'C' compiler and integrated development environment for Microchip's PIC16/17 family of microcontrollers. The compiler provides powerful integration capabilities and ease of use not found with other compilers.

For easier source level debugging, the compiler provides symbol information that is compatible with the PICMASTER Universal Emulator memory display (PICMASTER emulator software versions 1.13 and later).

The MP-C Code Development System is supplied directly by Byte Craft Limited of Waterloo, Ontario, Canada. If you have any questions, please contact your regional Microchip FAE or Microchip technical support personnel at (602) 786-7627.

## 15.11 Fuzzy Logic Development System (fuzzyTECH-MP)

*fuzzyTECH-MP* fuzzy logic development tool is available in two versions - a low cost introductory version, MP Explorer, for designers to gain a comprehensive working knowledge of fuzzy logic system design; and a full-featured version, *fuzzyTECH-MP*, edition for implementing more complex systems.

Both versions include Microchip's *fuzzyLAB™* demonstration board for hands-on experience with fuzzy logic systems implementation.

## 15.12 Development Systems

For convenience, the development tools are packaged into comprehensive systems as listed in Table 15-2.

**TABLE 15-2: DEVELOPMENT SYSTEM PACKAGES**

Item	Name	System Description
1.	PICMASTER System	PICMASTER In-Circuit Emulator, PRO MATE Programmer, Assembler, Software Simulator, Samples and your choice of Target Probe.
2.	PICSTART System	PICSTART Low-Cost Prototype Programmer, Assembler, Software Simulator and Samples.
3.	PRO MATE System	PRO MATE Universal Programmer, full featured stand-alone or PC-hosted programmer, Assembler, Simulator

# PIC16C6X

---

NOTES:

Applicable Devices	61	62	62A	R62	63	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

## 16.0 ELECTRICAL CHARACTERISTICS FOR PIC16C61

### Absolute Maximum Ratings †

Ambient temperature under bias	-55 to +125°C
Storage temperature	-65°C to +150°C
Voltage on any pin with respect to VSS (except VDD and $\overline{\text{MCLR}}$ )	-0.3V to (VDD + 0.3V)
Voltage on VDD with respect to VSS	0 to +7.5V
Voltage on $\overline{\text{MCLR}}$ with respect to VSS (Note 2)	0 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 <sub>IK</sub> (V <sub>I</sub> < 0 or V <sub>I</sub> > VDD)	± 20 mA
Output clamp current, I <sub>OK</sub> (V <sub>O</sub> < 0 or V <sub>O</sub> > VDD)	± 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  $\overline{\text{MCLR}}$  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  $\overline{\text{MCLR}}$  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 16-1: 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.	Do not 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.	Do not 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.

# PIC16C6X

Applicable Devices	61	62	62A	R62	63	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

## 16.1 DC Characteristics:

**PIC16C61-04 (Commercial, Industrial, Automotive)**

**PIC16C61-20 (Commercial, Industrial, Automotive)**

Standard Operating Conditions (unless otherwise stated)							
DC CHARACTERISTICS		Operating temperature					
		-40°C ≤ TA ≤ +125°C for automotive,					
		-40°C ≤ TA ≤ +85°C for industrial and					
0°C ≤ TA ≤ +70°C for commercial							
Param No.	Characteristic	Sym	Min	Typ†	Max	Units	Conditions
D001 D001A	Supply Voltage	VDD	4.0 4.5	- -	6.0 5.5	V V	XT, RC and LP osc configuration HS osc configuration
D002*	RAM Data Retention Voltage (Note 1)	VDR	-	1.5	-	V	Device in SLEEP mode
D003	VDD start voltage to ensure Power-on Reset	VPOR	-	VSS	-	V	See section on Power-on Reset for details
D004*	VDD rise rate to ensure Power-on Reset	SVDD	0.05	-	-	V/ms	See section on Power-on Reset for details
D010  D013	Supply Current (Note 2)	IDD	-  -	1.8  13.5	3.3  30	mA  mA	FOSC = 4 MHz, VDD = 5.5V (Note 4)  HS osc configuration (PIC16C61-20) FOSC = 20 MHz, VDD = 5.5V
D020 D021 D021A D021B	Power-down Current (Note 3)	IPD	- - - -	7 1.0 1.0 1.0	28 14 16 20	μA μA μA μA	VDD = 4.0V, WDT enabled, -40°C to +85°C VDD = 4.0V, WDT disabled, -0°C to +70°C VDD = 4.0V, WDT disabled, -40°C to +85°C VDD = 4.0V, WDT disabled, -40°C to +125°C

\* These parameters are characterized but not tested.

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

Note 1: This is the limit to which VDD can be lowered in SLEEP mode without losing RAM data.

2: The supply current is mainly a function of the operating voltage and frequency. Other factors such as I/O pin loading and switching rate, oscillator type, internal code execution pattern, and temperature also have an impact on the current consumption.

The test conditions for all IDD measurements in active operation mode are:

OSC1 = external square wave, from rail to rail; all I/O pins tristated, pulled to VDD,

MCLR = VDD; WDT enabled/disabled as specified.

3: The power-down current in SLEEP mode does not depend on the oscillator type. Power-down current is measured with the part in SLEEP mode, with all I/O pins in hi-impedance state and tied to VDD and VSS.

4: For RC osc configuration, current through Rext is not included. The current through the resistor can be estimated by the formula  $I_r = V_{DD}/2R_{ext}$  (mA) with Rext in kOhm.

Applicable Devices	61	62	62A	R62	63	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

## 16.2 DC Characteristics: PIC16LC61-04 (Commercial, Industrial, Automotive)

Standard Operating Conditions (unless otherwise stated)							
DC CHARACTERISTICS							
Operating temperature    -40°C    ≤ TA ≤ +125°C for automotive, -40°C    ≤ TA ≤ +85°C for industrial and 0°C        ≤ TA ≤ +70°C for commercial							
Param No.	Characteristic	Sym	Min	Typ†	Max	Units	Conditions
D001	Supply Voltage	VDD	3.0	-	6.0	V	XT, RC, and LP osc configuration
D002*	RAM Data Retention Voltage (Note 1)	VDR	-	1.5	-	V	Device in SLEEP mode
D003	VDD start voltage to ensure Power-on Reset	VPOR	-	VSS	-	V	See section on Power-on Reset for details
D004*	VDD rise rate to ensure Power-on Reset	SVDD	0.05	-	-	V/ms	See section on Power-on Reset for details
D010	Supply Current (Note 2)	IDD	-	1.4	2.5	mA	FOSC = 4 MHz, VDD = 3.0V (Note 4)
D010A			-	15	32	μA	FOSC = 32 kHz, VDD = 3.0V, WDT disabled, LP osc configuration
D020	Power-down Current (Note 3)	IPD	-	5	20	μA	VDD = 3.0V, WDT enabled, -40°C to +85°C
D021			-	0.6	9	μA	VDD = 3.0V, WDT disabled, 0°C to +70°C
D021A			-	0.6	12	μA	VDD = 3.0V, WDT disabled, -40°C to +85°C
D021B			-	0.6	16	μA	VDD = 3.0V, WDT disabled, -40°C to +125°C

\* These parameters are characterized but not tested.

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

Note 1: This is the limit to which VDD can be lowered in SLEEP mode without losing RAM data.

2: The supply current is mainly a function of the operating voltage and frequency. Other factors such as I/O pin loading and switching rate, oscillator type, internal code execution pattern, and temperature also have an impact on the current consumption.

The test conditions for all IDD measurements in active operation mode are:

OSC1 = external square wave, from rail to rail; all I/O pins tristated, pulled to VDD,

MCLR = VDD; WDT enabled/disabled as specified.

3: The power-down current in SLEEP mode does not depend on the oscillator type. Power-down current is measured with the part in SLEEP mode, with all I/O pins in hi-impedance state and tied to VDD and VSS.

4: For RC osc configuration, current through Rext is not included. The current through the resistor can be estimated by the formula  $I_r = V_{DD}/2R_{ext}$  (mA) with Rext in kOhm.

# PIC16C6X

Applicable Devices 61 62 62A R62 63 64 64A R64 65 65A

## 16.3 DC Characteristics:

**PIC16C61-04 (Commercial, Industrial, Automotive)**

**PIC16C61-20 (Commercial, Industrial, Automotive)**

**PIC16LC61-04 (Commercial, Industrial, Automotive)**

DC CHARACTERISTICS		Standard Operating Conditions (unless otherwise stated) Operating temperature    -40°C ≤ TA ≤ +125°C for automotive, -40°C ≤ TA ≤ +85°C for industrial and 0°C ≤ TA ≤ +70°C for commercial Operating voltage VDD range as described in DC spec Section 16.1 and Section 16.2.					
Param No.	Characteristic	Sym	Min	Typ†	Max	Units	Conditions
D030	<b>Input Low Voltage</b> I/O ports	V <sub>IL</sub>	V <sub>SS</sub>	-	0.8V	V	Note1
D031	with TTL buffer		V <sub>SS</sub>	-	0.2V <sub>D</sub>	V	
D032	with Schmitt Trigger buffer		V <sub>SS</sub>	-	0.2V <sub>D</sub>	V	
D033	MCLR, RA4/T0CKI, OSC1 (in RC mode)		V <sub>SS</sub>	-	0.3V <sub>D</sub>	V	
D040	<b>Input High Voltage</b> I/O ports	V <sub>IH</sub>	0.36V <sub>DD</sub>	-	V <sub>DD</sub>	V	4.5V ≤ V <sub>DD</sub> ≤ 5.5V For V <sub>DD</sub> > 5.5V or V <sub>DD</sub> < 4.5V For entire V <sub>DD</sub> range Note1
D040A	with TTL buffer		0.45V <sub>DD</sub>	-	V <sub>DD</sub>	V	
D041	with Schmitt Trigger buffer		0.85V <sub>DD</sub>	-	V <sub>DD</sub>	V	
D042	MCLR, RA4/T0CKI		0.85V <sub>DD</sub>	-	V <sub>DD</sub>	V	
D042A	OSC1 (XT, HS and LP)		0.7V <sub>DD</sub>	-	V <sub>DD</sub>	V	Note1
D043	OSC1 (in RC mode)		0.9V <sub>DD</sub>	-	V <sub>DD</sub>	V	
D070	PORTB weak pull-up current	IP <sub>URB</sub>	50	250	† 400	μA	V <sub>DD</sub> = 5V, V <sub>PIN</sub> = V <sub>SS</sub>
D060	<b>Input Leakage Current</b> (Notes 2, 3) I/O ports	I <sub>IL</sub>	-	-	±1	μA	V <sub>SS</sub> ≤ V <sub>PIN</sub> ≤ V <sub>DD</sub> , Pin at hi-impedance
D061	MCLR, RA4/T0CKI		-	-	±5	μA	V <sub>SS</sub> ≤ V <sub>PIN</sub> ≤ V <sub>DD</sub>
D063	OSC1		-	-	±5	μA	V <sub>SS</sub> ≤ V <sub>PIN</sub> ≤ V <sub>DD</sub> , XT, HS and LP osc configuration
D080	<b>Output Low Voltage</b> I/O ports	V <sub>OL</sub>	-	-	0.6	V	I <sub>OL</sub> = 8.5 mA, V <sub>DD</sub> = 4.5V, -40°C to +85°C
D080A			-	-	0.6	V	I <sub>OL</sub> = 7.0 mA, V <sub>DD</sub> = 4.5V, -40°C to +125°C
D083	OSC2/CLKOUT (RC osc config)		-	-	0.6	V	I <sub>OL</sub> = 1.6 mA, V <sub>DD</sub> = 4.5V, -40°C to +85°C
D083A			-	-	0.6	V	I <sub>OL</sub> = 1.2 mA, V <sub>DD</sub> = 4.5V, -40°C to +125°C

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

Note 1: In RC oscillator configuration, the OSC1/CLKIN pin is a Schmitt Trigger input. It is not recommended that the PIC16C6X be driven with external clock in RC mode.

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

3: Negative current is defined as coming out of the pin.



Applicable Devices	61	62	62A	R62	63	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

<b>DC CHARACTERISTICS</b> <div> <b>Standard Operating Conditions (unless otherwise stated)</b>            Operating temperature    -40°C    ≤ TA ≤ +125°C for automotive,               -40°C    ≤ TA ≤ +85°C for industrial and               0°C      ≤ TA ≤ +70°C for commercial            Operating voltage VDD range as described in DC spec Section 16.1 and            Section 16.2.         </div>							
Param No.	Characteristic	Sym	Min	Typ†	Max	Units	Conditions
D090	<b>Output High Voltage</b> I/O ports (Note 3)	VOH	VDD-0.7	-	-	V	IOH = -3.0 mA, VDD = 4.5V, -40°C to +85°C
D090A			VDD-0.7	-	-	V	IOH = -2.5 mA, VDD = 4.5V, -40°C to +125°C
D092			VDD-0.7	-	-	V	IOH = -1.3 mA, VDD = 4.5V, -40°C to +85°C
D092A			VDD-0.7	-	-	V	IOH = -1.0 mA, VDD = 4.5V, -40°C to +125°C
D100	<b>Capacitive Loading Specs on Output Pins</b> OSC2 pin	COSC2			15	pF	In XT, HS and LP modes when external clock is used to drive OSC1.
D101	All I/O pins and OSC2 (in RC mode)	CIO			50	pF	

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

Note 1: In RC oscillator configuration, the OSC1/CLKIN pin is a Schmitt Trigger input. It is not recommended that the PIC16C6X be driven with external clock in RC mode.

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

3: Negative current is defined as coming out of the pin.

# PIC16C6X

Applicable Devices	61	62	62A	R62	63	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

## 16.4 Timing Parameter Symbolology

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

1. TppS2ppS
2. TppS
3. TCC:ST (I<sup>2</sup>C specifications only)
4. Ts (I<sup>2</sup>C specifications only)

<b>T</b>			
F	Frequency	T	Time

Lowercase letters (pp) and their meanings:

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

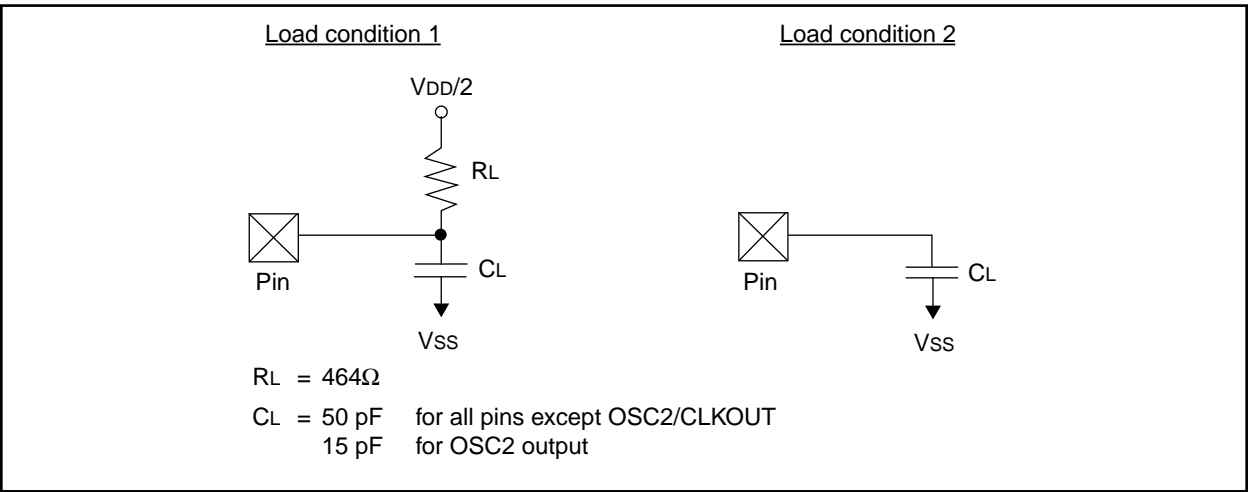
Uppercase letters and their meanings:

<b>S</b>			
F	Fall	P	Period
H	High	R	Rise
I	Invalid (Hi-impedance)	V	Valid
L	Low	Z	Hi-impedance
<b>I<sup>2</sup>C only</b>			
AA	output access	High	High
BUF	Bus free	Low	Low

TCC:ST (I<sup>2</sup>C specifications only)

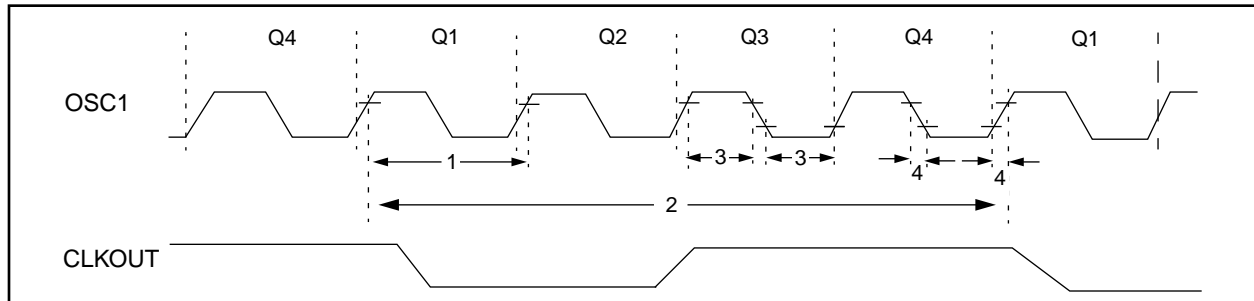
<b>CC</b>			
HD	Hold	SU	Setup
<b>ST</b>			
DAT	DATA input hold	STO	STOP condition
STA	START condition		

FIGURE 16-1: LOAD CONDITIONS FOR DEVICE TIMING SPECIFICATIONS



## 16.5 Timing Diagrams and Specifications

**FIGURE 16-2: EXTERNAL CLOCK TIMING**



**TABLE 16-2: EXTERNAL CLOCK TIMING REQUIREMENTS**

Parameter No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
	Fosc	<b>External CLKIN Frequency (Note 1)</b>	DC	—	4	MHz	XT and RC osc mode
			DC	—	4	MHz	HS osc mode (PIC16C61-04,
			DC	—	20	MHz	HS osc mode (PIC16C61-20)
			DC	—	200	kHz	LP osc mode
		<b>Oscillator Frequency (Note 1)</b>	DC	—	4	MHz	RC osc mode
			0.1	—	4	MHz	XT osc mode
			1	—	4	MHz	HS osc mode (PIC16C61-04,
			1	—	20	MHz	HS osc mode (PIC16C61-20)
1	Tosc	<b>External CLKIN Period (Note 1)</b>	250	—	—	ns	XT and RC osc mode
			250	—	—	ns	HS osc mode (PIC16C61-04,
			50	—	—	ns	HS osc mode (PIC16C61-20)
			5	—	—	μs	LP osc mode
		<b>Oscillator Period (Note 1)</b>	250	—	—	ns	RC osc mode
			250	—	10,000	ns	XT osc mode
			250	—	1,000	ns	HS osc mode (PIC16C61-04,
			50	—	1,000	ns	HS osc mode (PIC16C61-20)
2	Tcy	<b>Instruction Cycle Time (Note 1)</b>	5	—	—	μs	LP osc mode
			1.0	4/Fosc	DC	μs	Tcy = 4/Fosc
			—	—	—	—	—
3	TosL, TosH	<b>External Clock in (OSC1) High or Low Time</b>	50	—	—	ns	XT oscillator
			2.5	—	—	μs	LP oscillator
			10	—	—	ns	HS oscillator
4	TosR, TosF	<b>External Clock in (OSC1) Rise or Fall Time</b>	25	—	—	ns	XT oscillator
			50	—	—	ns	LP oscillator
			15	—	—	ns	HS oscillator

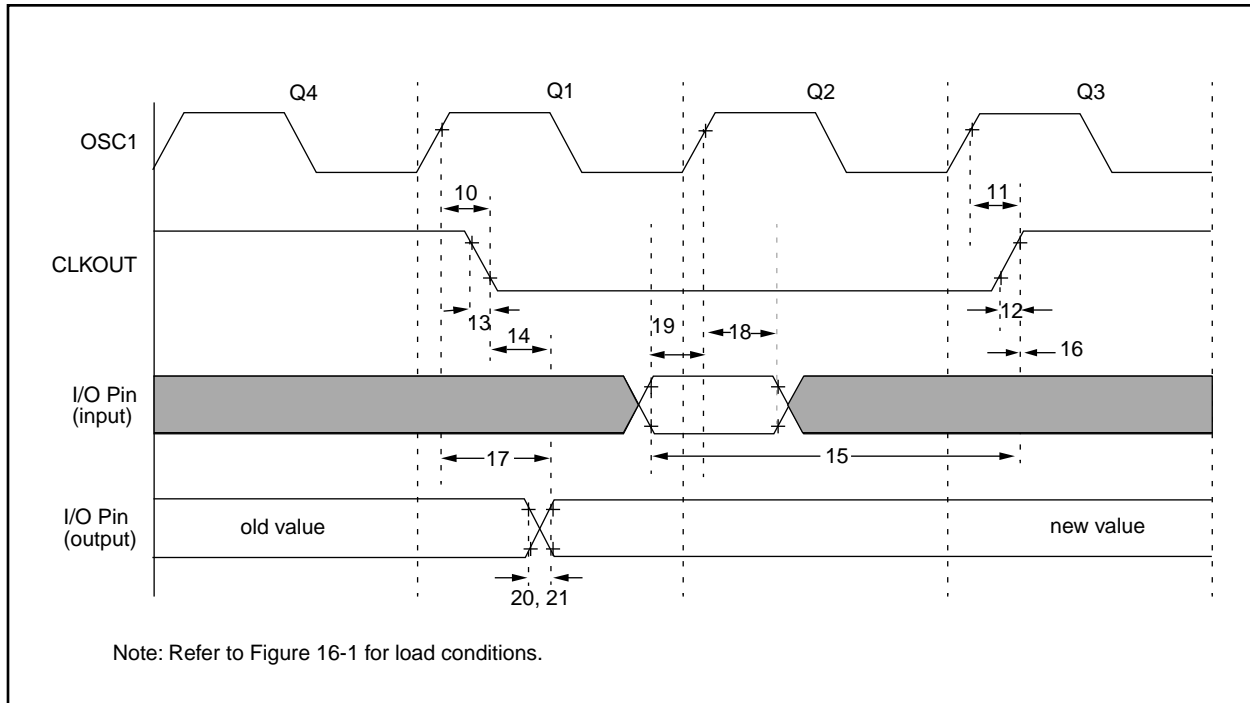
† Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

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

# PIC16C6X

Applicable Devices 61 62 62A R62 63 64 64A R64 65 65A

**FIGURE 16-3: CLKOUT AND I/O TIMING**



**TABLE 16-3: CLKOUT AND I/O TIMING REQUIREMENTS**

Parameter No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
10*	TosH2ckL	OSC1↑ to CLKOUT↓	—	15	30	ns	Note 1
11*	TosH2ckH	OSC1↑ to CLKOUT↑	—	15	30	ns	Note 1
12*	TckR	CLKOUT rise time	—	5	15	ns	Note 1
13*	TckF	CLKOUT fall time	—	5	15	ns	Note 1
14*	TckL2ioV	CLKOUT ↓ to Port out valid	—	—	0.5T <sub>CY</sub> + 20	ns	Note 1
15*	TioV2ckH	Port in valid before CLKOUT ↑	0.25T <sub>CY</sub> + 25	—	—	ns	Note 1
16*	TckH2ioL	Port in hold after CLKOUT ↑	0	—	—	ns	Note 1
17*	TosH2ioV	OSC1↑ (Q1 cycle) to Port out valid	—	—	80 - 100	ns	
18*	TosH2ioL	OSC1↑ (Q2 cycle) to Port input invalid (I/O in hold time)	TBD	—	—	ns	
19*	TioV2osH	Port input valid to OSC1↑ (I/O in setup time)	TBD	—	—	ns	
20*	TioR	Port output rise time	PIC16C61	—	10	25	ns
			PIC16LC61	—	—	60	ns
21*	TioF	Port output fall time	PIC16C61	—	10	25	ns
			PIC16LC61	—	—	60	ns
22††*	Tinp	RB0/INT pin high or low time	20	—	—	ns	
23††*	Trbp	RB7:RB4 change int high or low time	20	—	—	ns	

\* These parameters are characterized but not tested.

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

†† These parameters are asynchronous events not related to any internal clock edges.

Note 1: Measurements are taken in RC Mode where CLKOUT output is 4 x T<sub>OSC</sub>.

Applicable Devices	61	62	62A	R62	63	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

FIGURE 16-4: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER AND POWER-UP  
TIMER TIMING

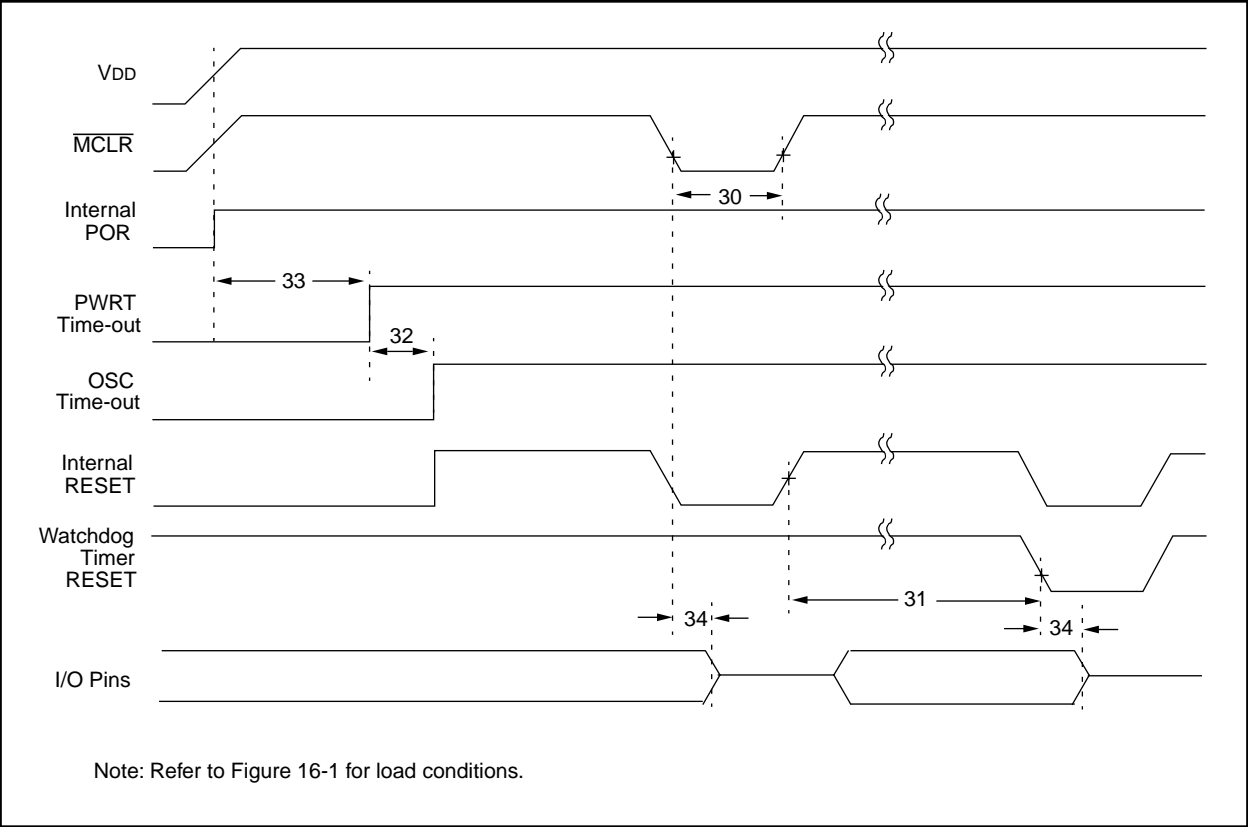


TABLE 16-4: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER AND POWER-UP  
TIMER REQUIREMENTS

Parameter No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
30*	Tmcl	MCLR Pulse Width (low)	200	—	—	ns	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	—	1024Tosc	—		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	—	—	100	ns	

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

# PIC16C6X

Applicable Devices	61	62	62A	R62	63	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

FIGURE 16-5: TIMER0 CLOCK TIMINGS

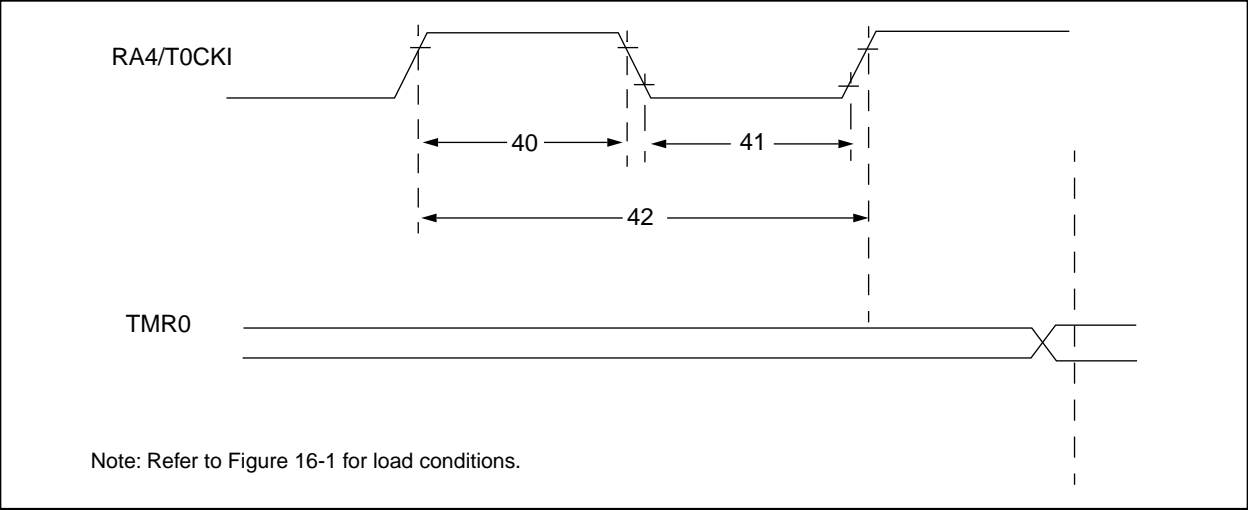


TABLE 16-5: TIMER0 CLOCK REQUIREMENTS

Parameter No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
40*	Tt0H	T0CKI High Pulse Width	No Prescaler	0.5TCY + 20	—	—	ns
			With Prescaler	10	—	—	ns
41*	Tt0L	T0CKI Low Pulse Width	No Prescaler	0.5TCY + 20	—	—	ns
			With Prescaler	10	—	—	ns
42*	Tt0P	T0CKI Period	Greater of: 20 μs or $\frac{TCY + 40}{N}$	—	—	ns	N = prescale value (2, 4, ..., 256)

\* 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.

Applicable Devices	61	62	62A	R62	63	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

17.0 DC AND AC CHARACTERISTICS GRAPHS AND TABLES FOR PIC16C61

The graphs and tables provided in this section are for design guidance and are not tested or guaranteed.

In some graphs or tables the data presented are outside specified operating range (i.e., outside specified VDD range). This is for information only and devices are guaranteed to operate properly only within the specified range.

The data presented in this section is a statistical summary of data collected on units from different lots over a period of time. 'Typical' represents the mean of the distribution while 'max' or 'min' represents (mean +3σ) and (mean -3σ) respectively where σ is standard deviation.

FIGURE 17-1: TYPICAL RC OSCILLATOR FREQUENCY vs. TEMPERATURE

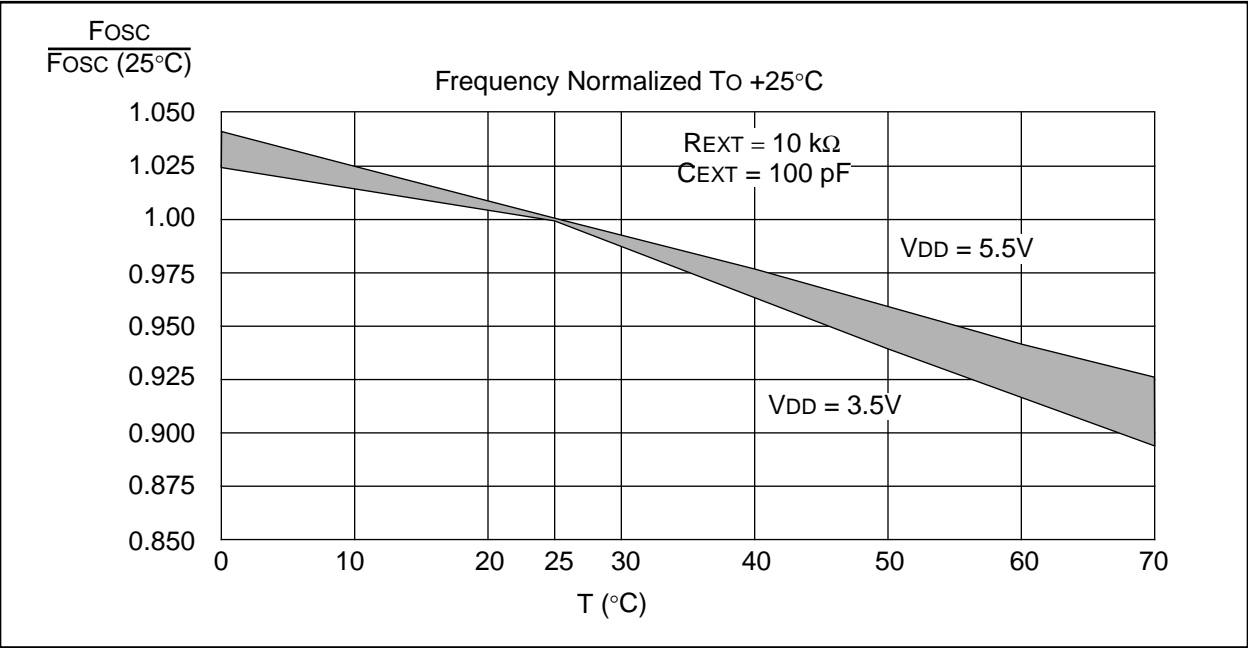


TABLE 17-1: RC OSCILLATOR FREQUENCIES

Cext	Rext	Average Fosc @ 5V, 25°C	
20 pF	4.7k	4.52 MHz	± 17.35%
	10k	2.47 MHz	± 10.10%
	100k	290.86 kHz	± 11.90%
100 pF	3.3k	1.92 MHz	± 9.43%
	4.7k	1.48 MHz	± 9.83%
	10k	788.77 kHz	± 10.92%
	100k	88.11 kHz	± 16.03%
300 pF	3.3k	726.89 kHz	± 10.97%
	4.7k	573.95 kHz	± 10.14%
	10k	307.31 kHz	± 10.43%
	100k	33.82 kHz	± 11.24%

The percentage variation indicated here is part to part variation due to normal process distribution. The variation indicated is ±3 standard deviation from average value for VDD = 5V.

# PIC16C6X

Applicable Devices 61 62 62A R62 63 64 64A R64 65 65A

FIGURE 17-2: TYPICAL RC OSCILLATOR FREQUENCY vs. VDD

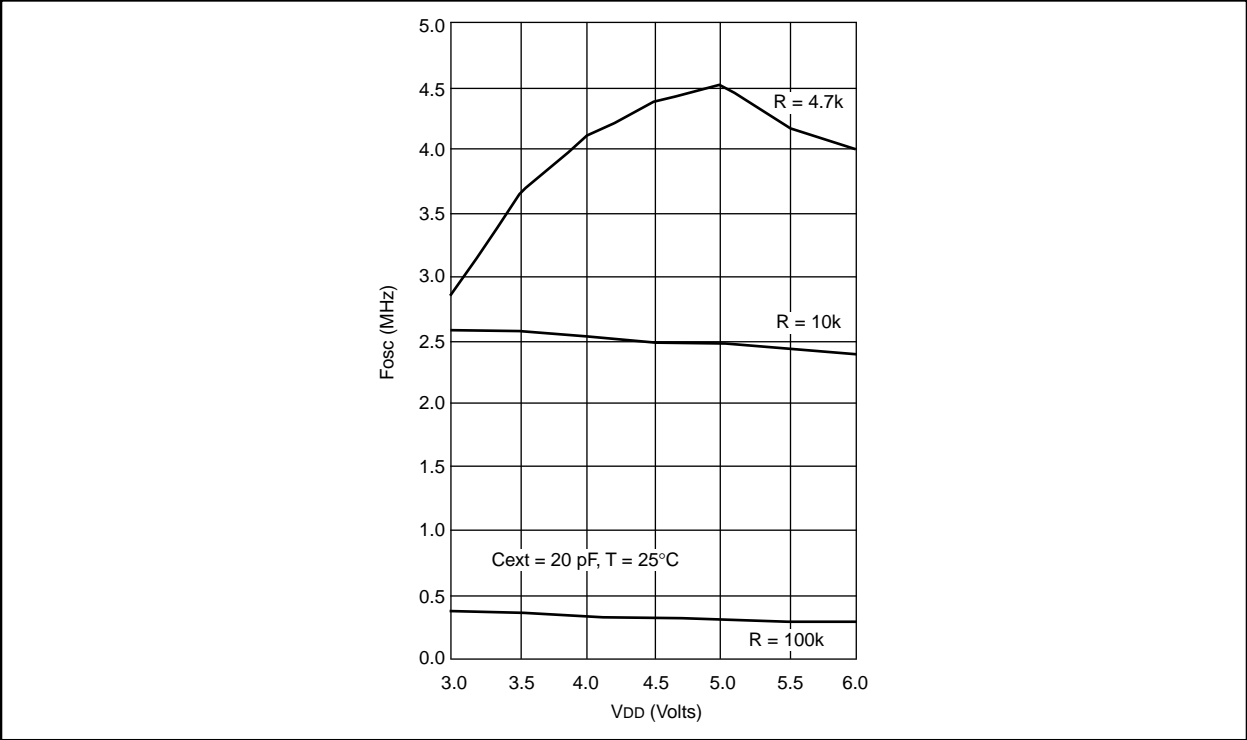
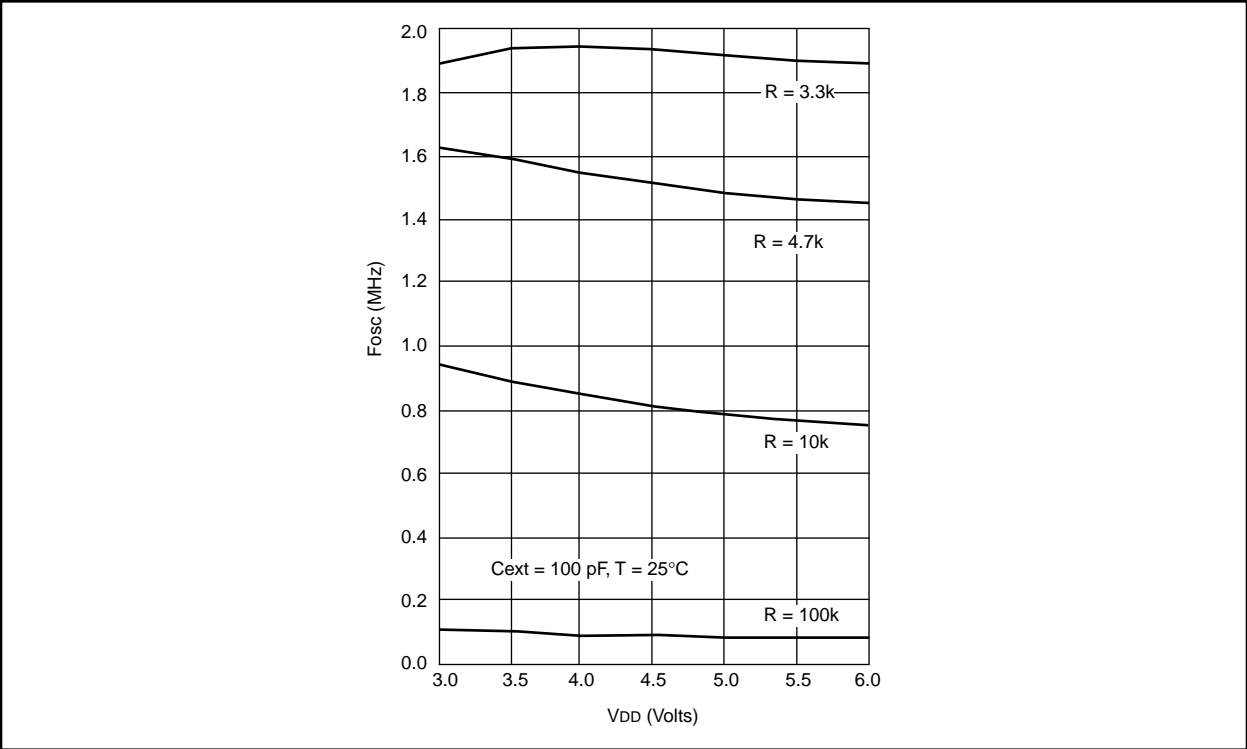
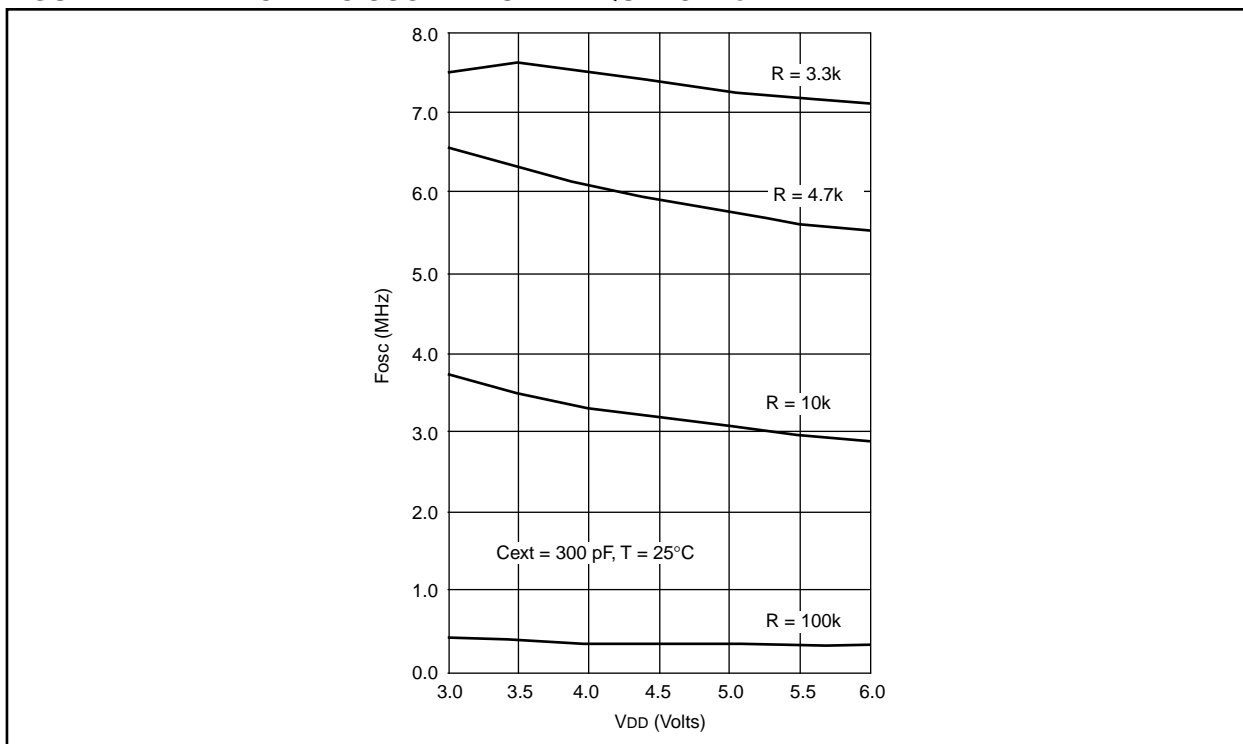


FIGURE 17-3: TYPICAL RC OSCILLATOR FREQUENCY vs. VDD

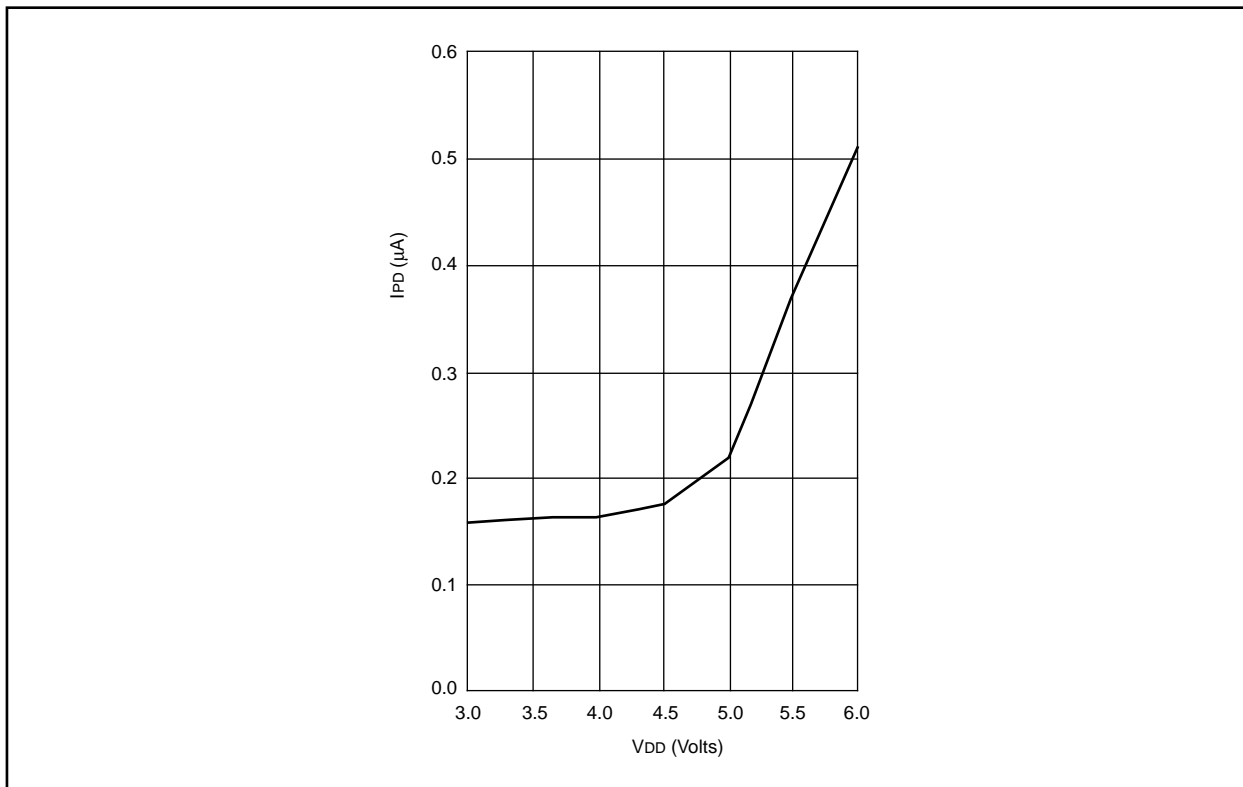




**FIGURE 17-4: TYPICAL RC OSCILLATOR FREQUENCY vs. V<sub>DD</sub>**



**FIGURE 17-5: TYPICAL I<sub>PD</sub> vs. V<sub>DD</sub> WATCHDOG TIMER DISABLED 25°C**



# PIC16C6X

Applicable Devices 61 62 62A R62 63 64 64A R64 65 65A

FIGURE 17-6: TYPICAL  $I_{PD}$  VS.  $V_{DD}$  WATCHDOG TIMER ENABLED 25°C

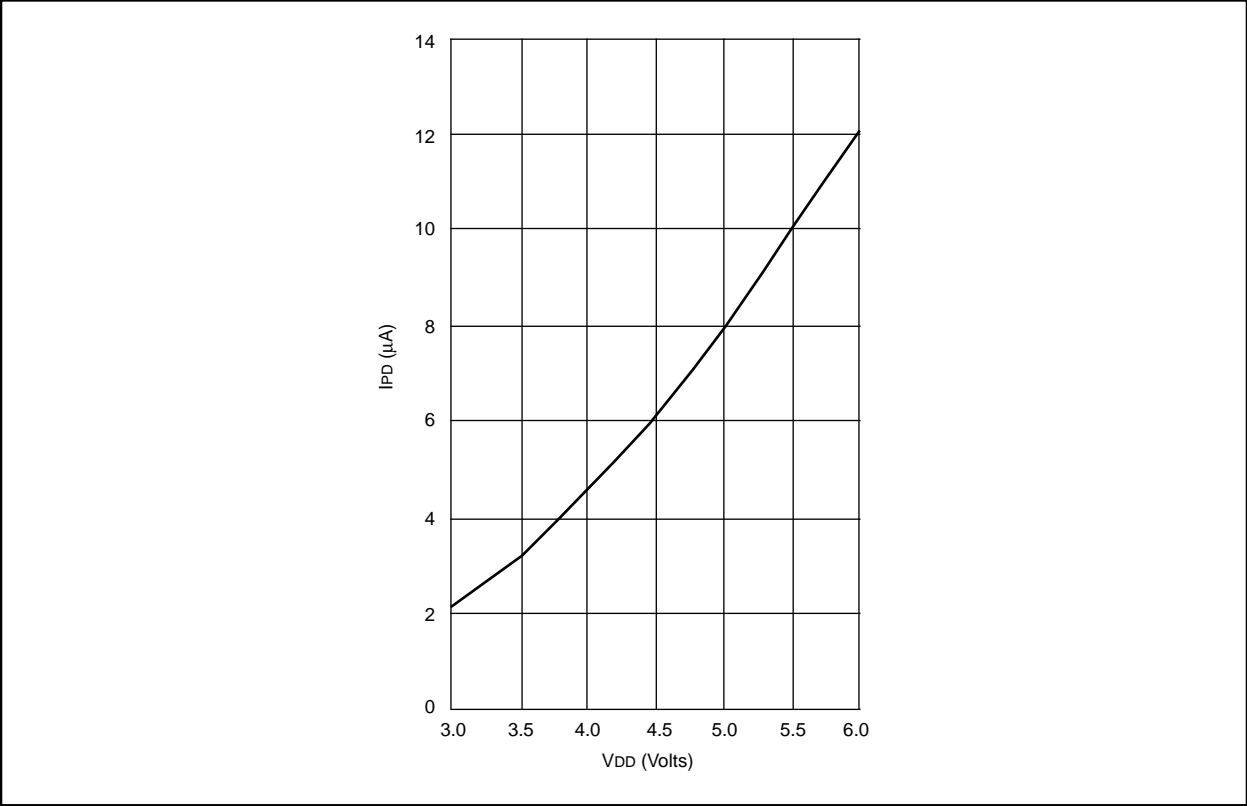
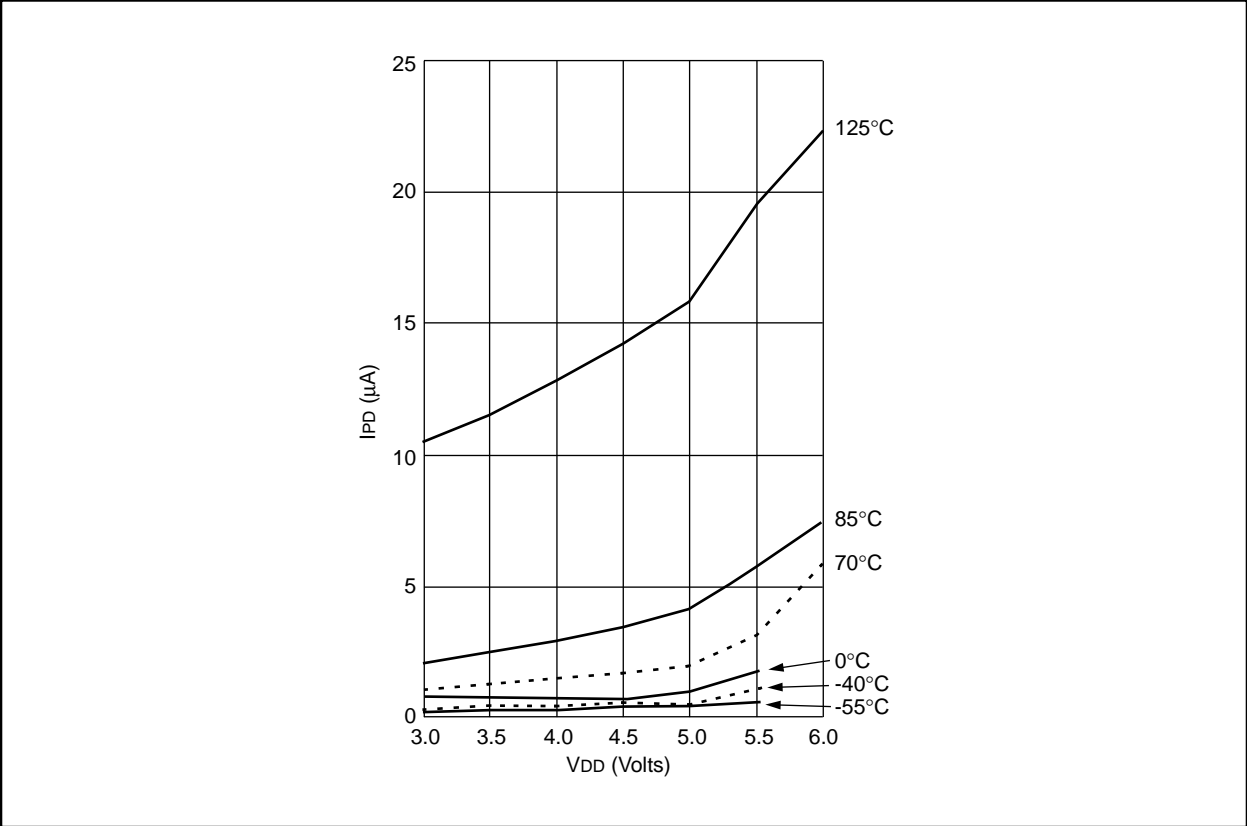
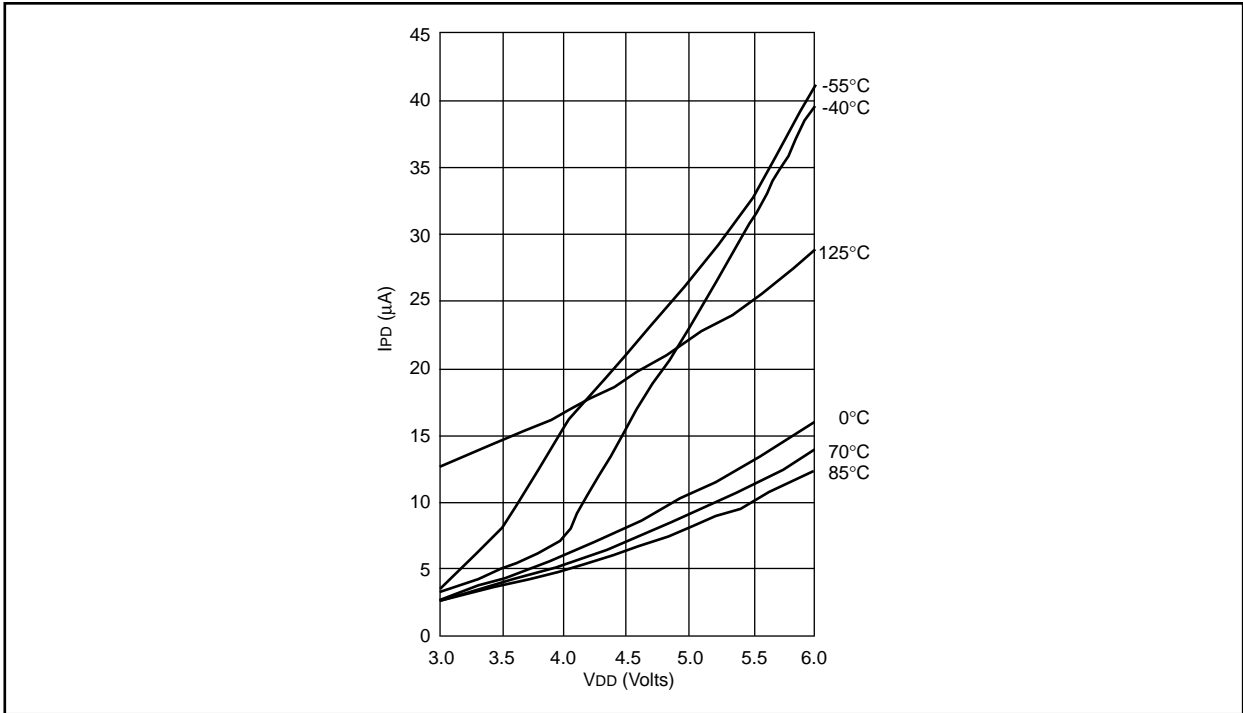


FIGURE 17-7: MAXIMUM  $I_{PD}$  VS.  $V_{DD}$  WATCHDOG DISABLED



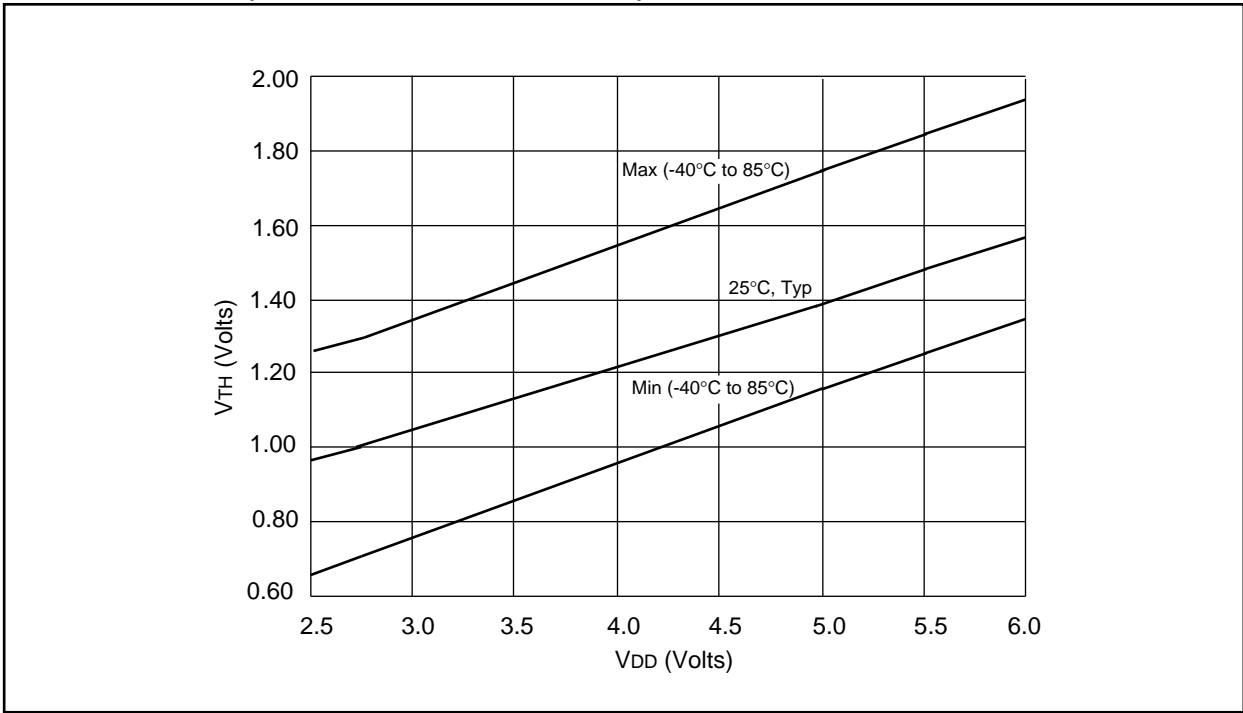
Applicable Devices	61	62	62A	R62	63	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

FIGURE 17-8: MAXIMUM I<sub>PD</sub> vs. V<sub>DD</sub> WATCHDOG ENABLED\*



\*I<sub>PD</sub>, with Watchdog Timer enabled, has two components: The leakage current which increases with higher temperature and the operating current of the Watchdog Timer logic which increases with lower temperature. At -40°C, the latter dominates explaining the apparently anomalous behavior.

FIGURE 17-9: V<sub>TH</sub> (INPUT THRESHOLD VOLTAGE) OF I/O PINS vs. V<sub>DD</sub>



# PIC16C6X

Applicable Devices	61	62	62A	R62	63	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

FIGURE 17-10:  $V_{IH}$ ,  $V_{IL}$  OF  $\overline{MCLR}$ ,  $T0CKI$  AND  $OSC1$  (IN RC MODE) vs.  $V_{DD}$

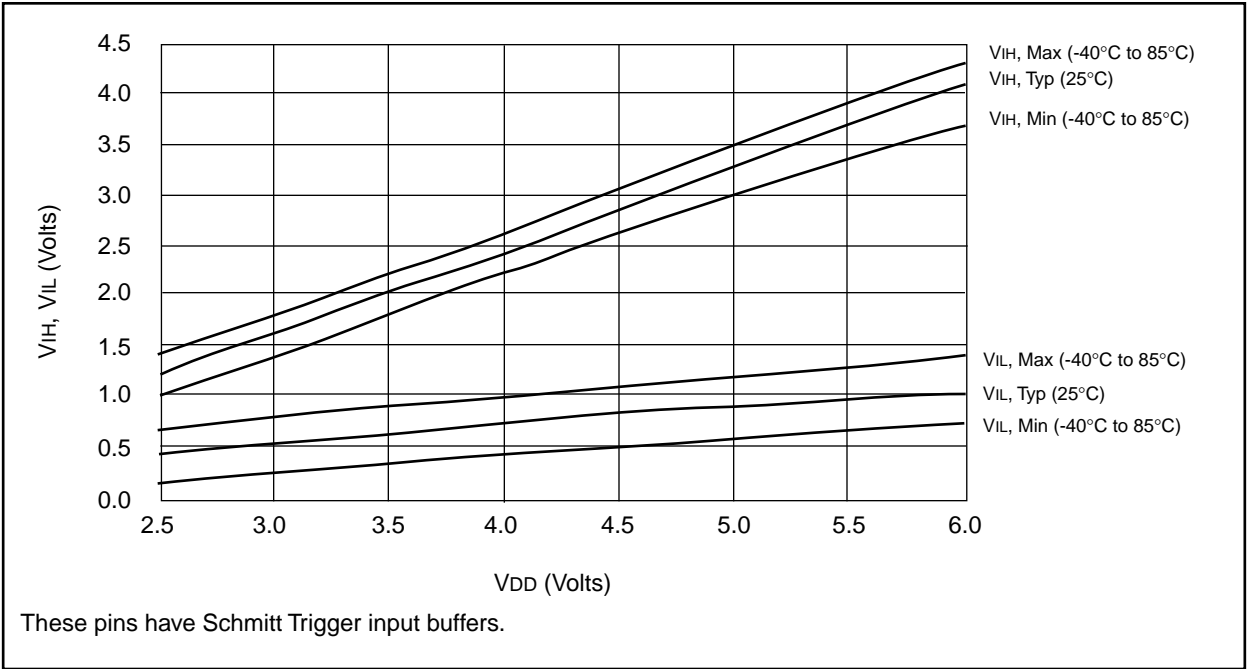
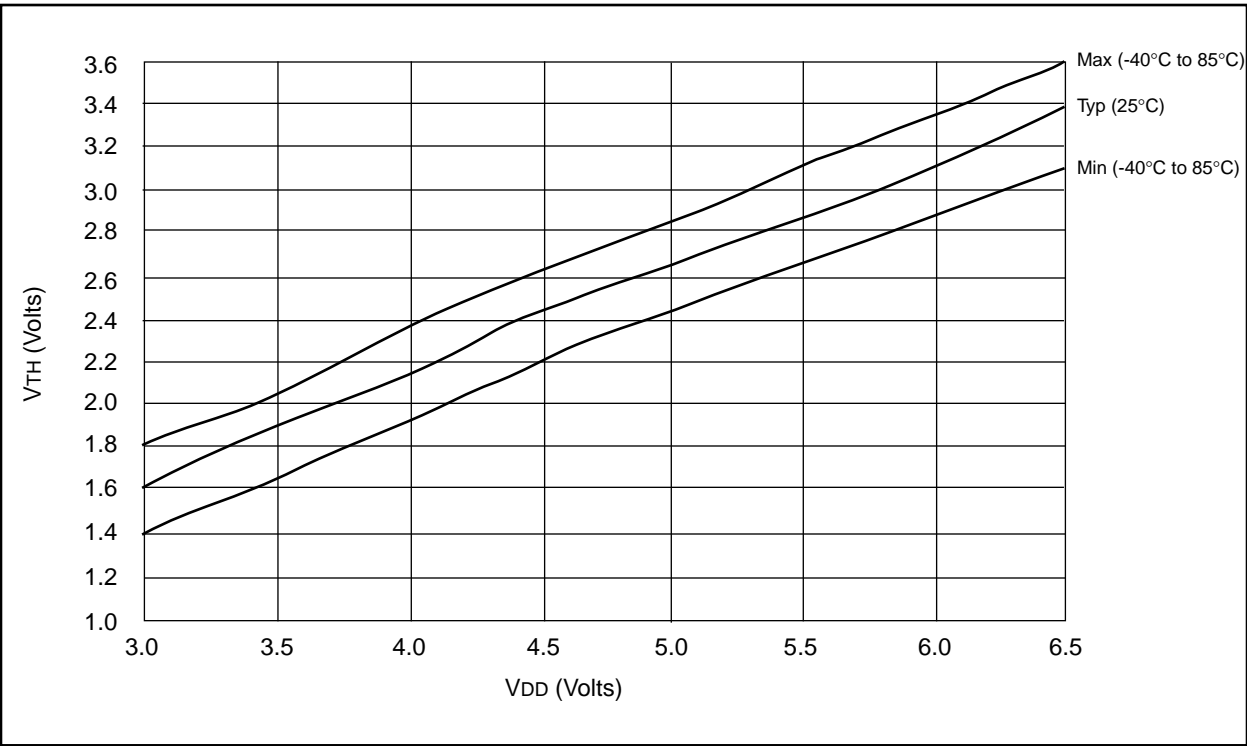
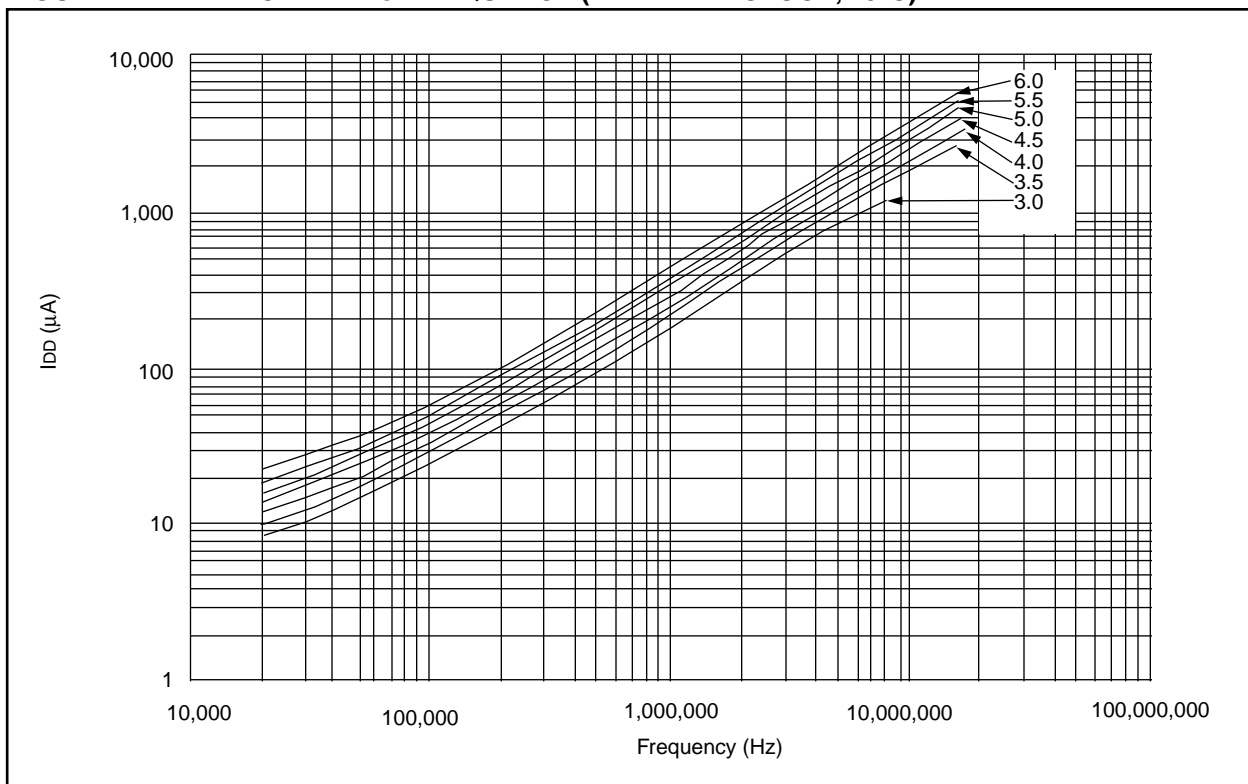


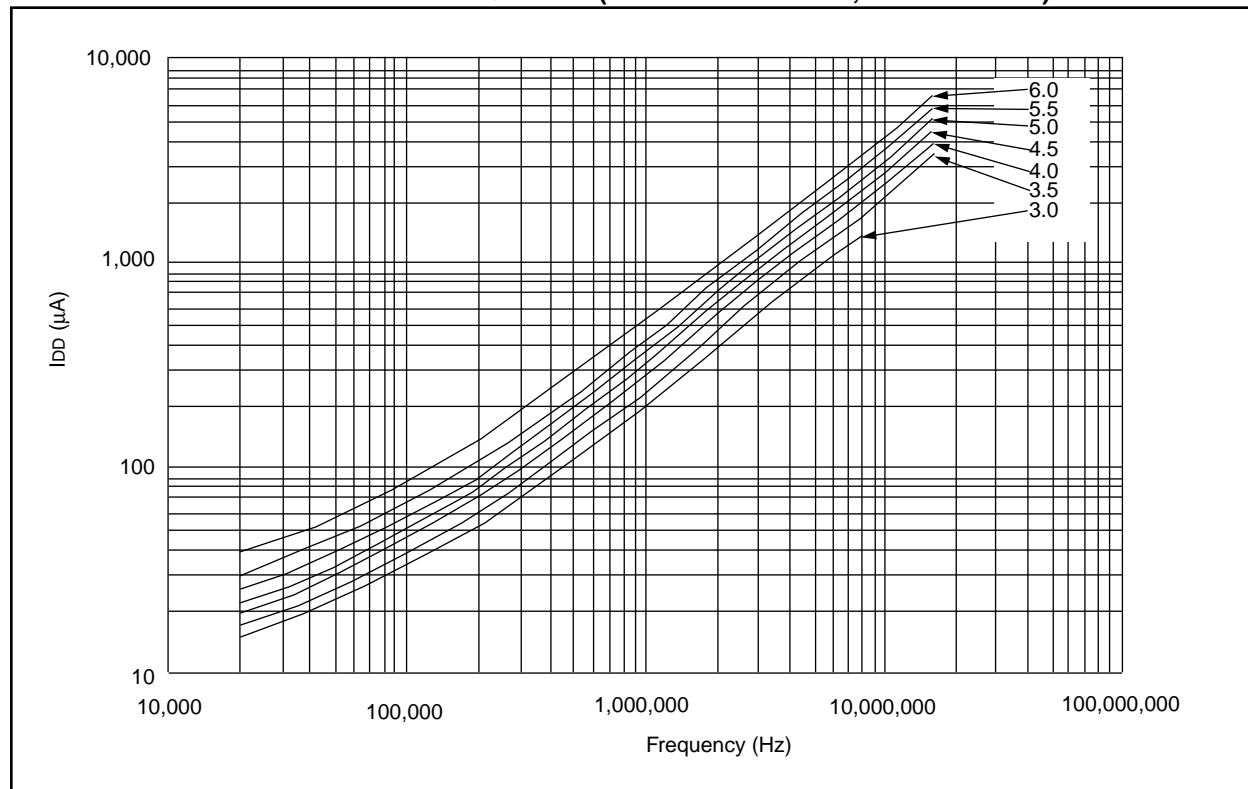
FIGURE 17-11:  $V_{TH}$  (INPUT THRESHOLD VOLTAGE) OF  $OSC1$  INPUT (IN XT, HS, AND LP MODES) vs.  $V_{DD}$



**FIGURE 17-12: TYPICAL  $I_{DD}$  vs. FREQUENCY (EXTERNAL CLOCK, 25°C)**



**FIGURE 17-13: MAXIMUM  $I_{DD}$  vs. FREQUENCY (EXTERNAL CLOCK, -40° TO +85°C)**



# PIC16C6X

Applicable Devices 61 62 62A R62 63 64 64A R64 65 65A

FIGURE 17-14: MAXIMUM I<sub>DD</sub> vs. FREQUENCY (EXTERNAL CLOCK, -55° TO +125°C)

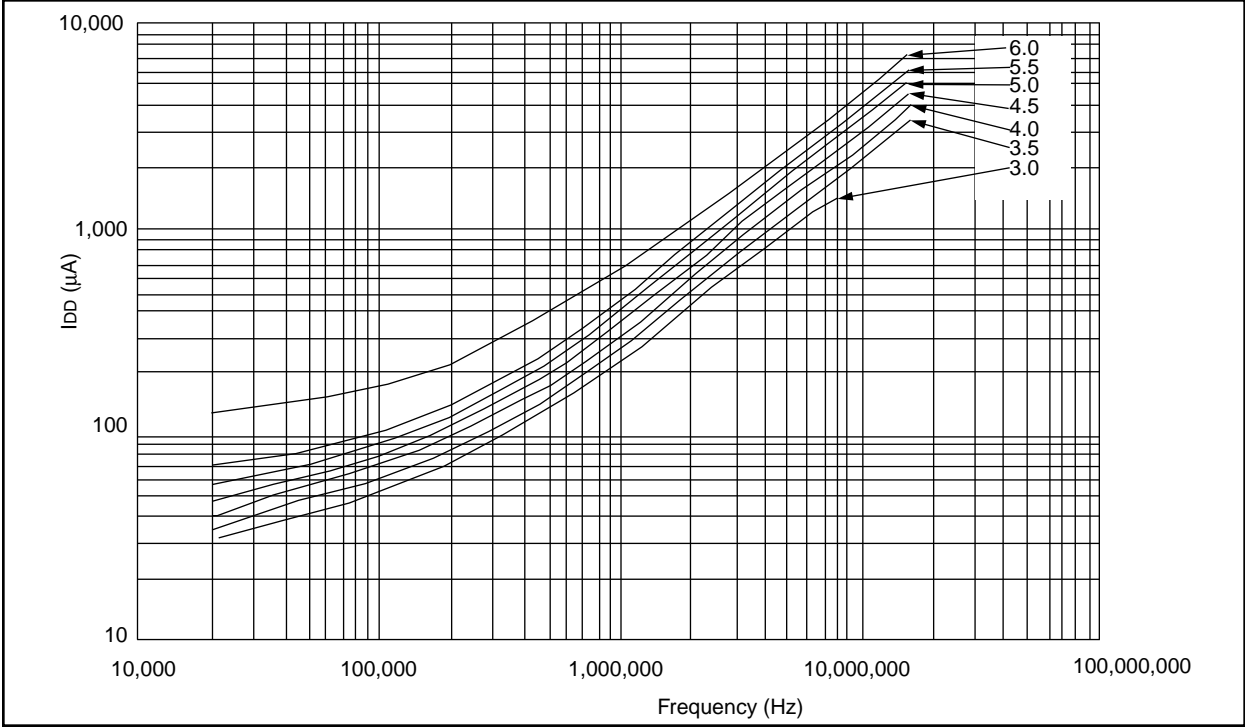
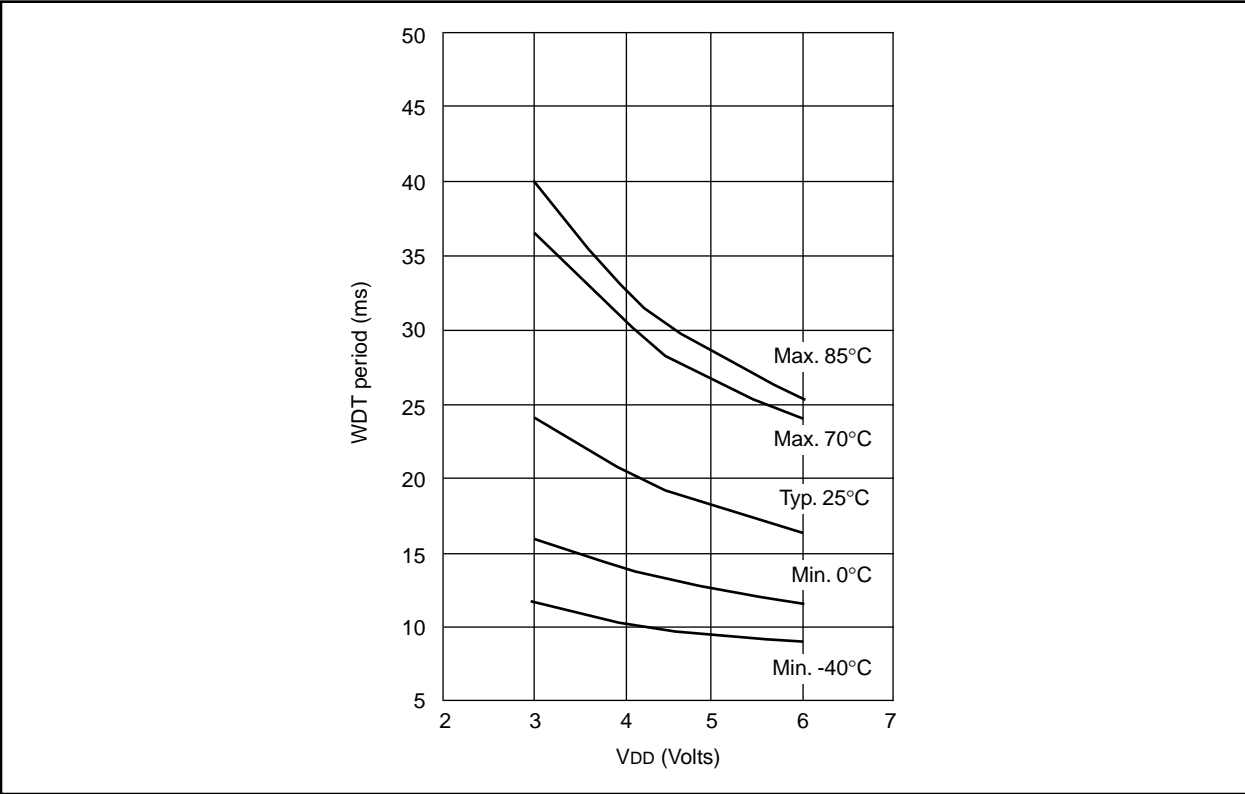


FIGURE 17-15: WDT TIMER TIME-OUT PERIOD vs. V<sub>DD</sub>



Applicable Devices	61	62	62A	R62	63	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

FIGURE 17-16: TRANSCONDUCTANCE (gm) OF HS OSCILLATOR vs. VDD

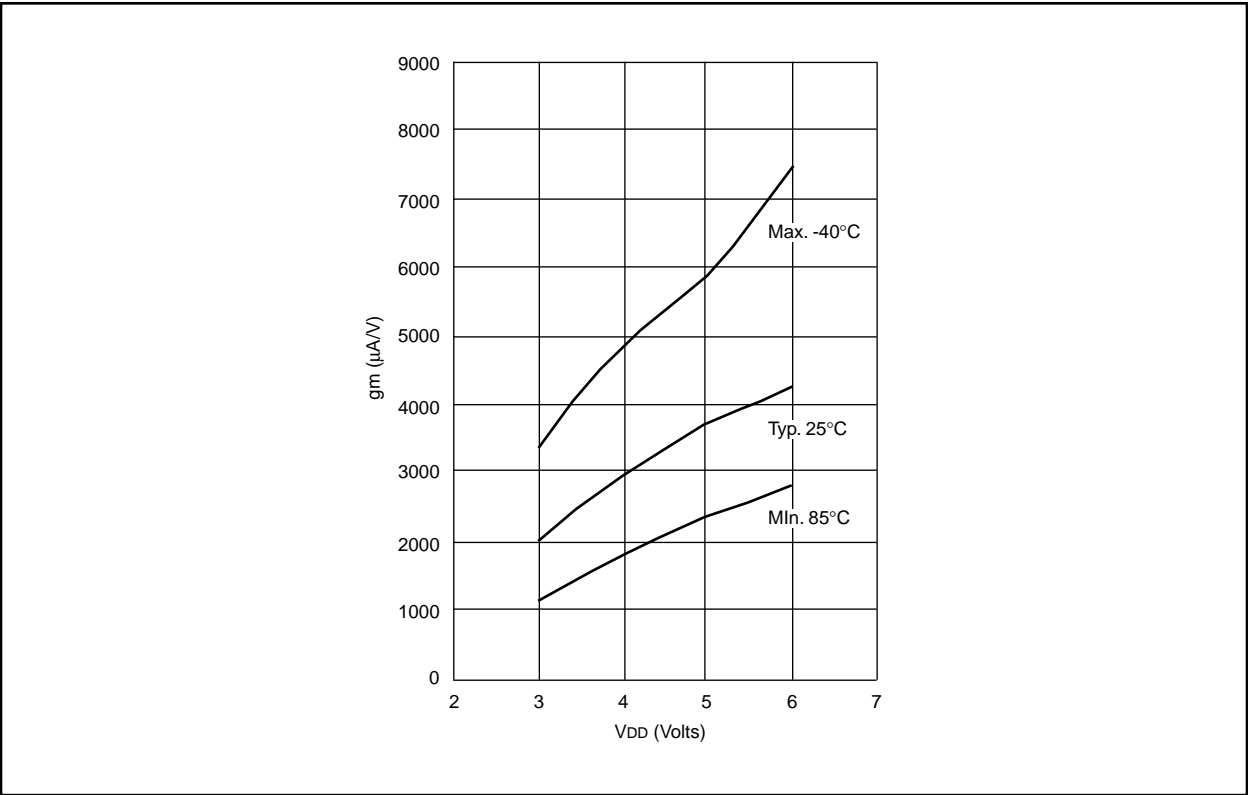
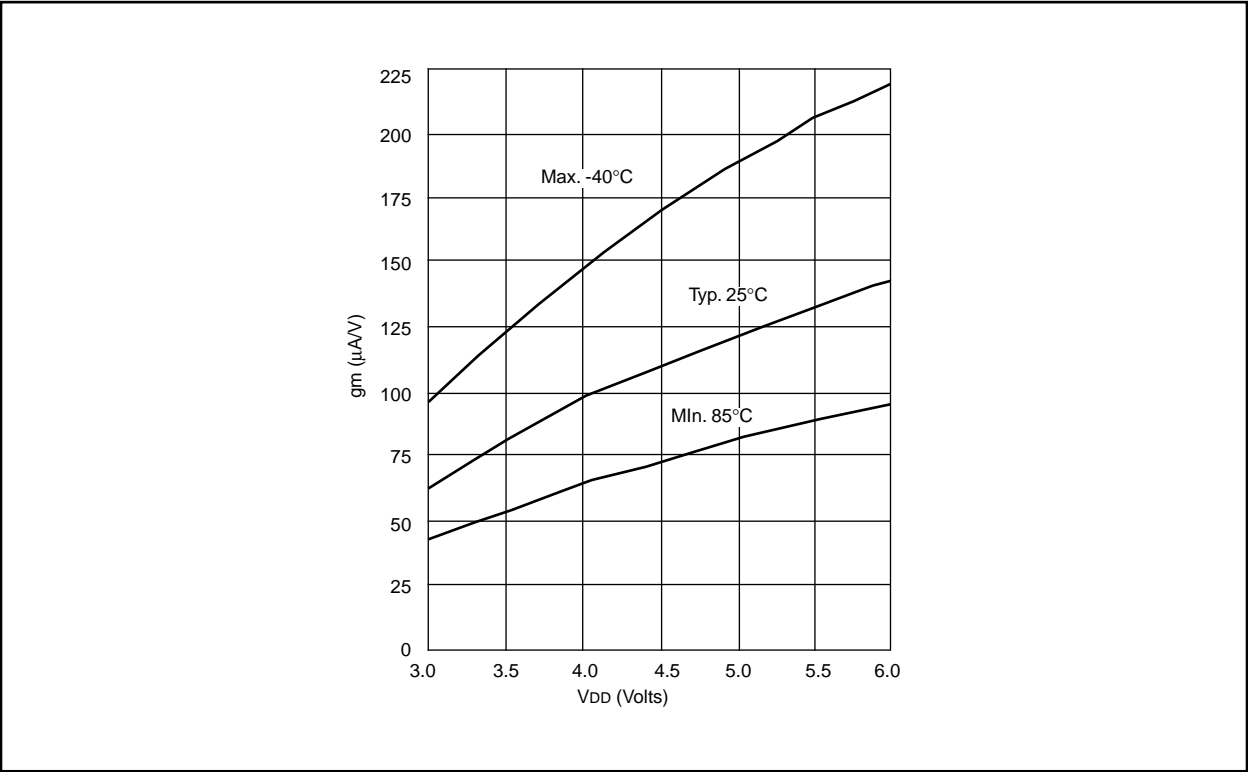


FIGURE 17-17: TRANSCONDUCTANCE (gm) OF LP OSCILLATOR vs. VDD



# PIC16C6X

Applicable Devices 61 62 62A R62 63 64 64A R64 65 65A

FIGURE 17-18: TRANSCONDUCTANCE (gm) OF XT OSCILLATOR vs. VDD

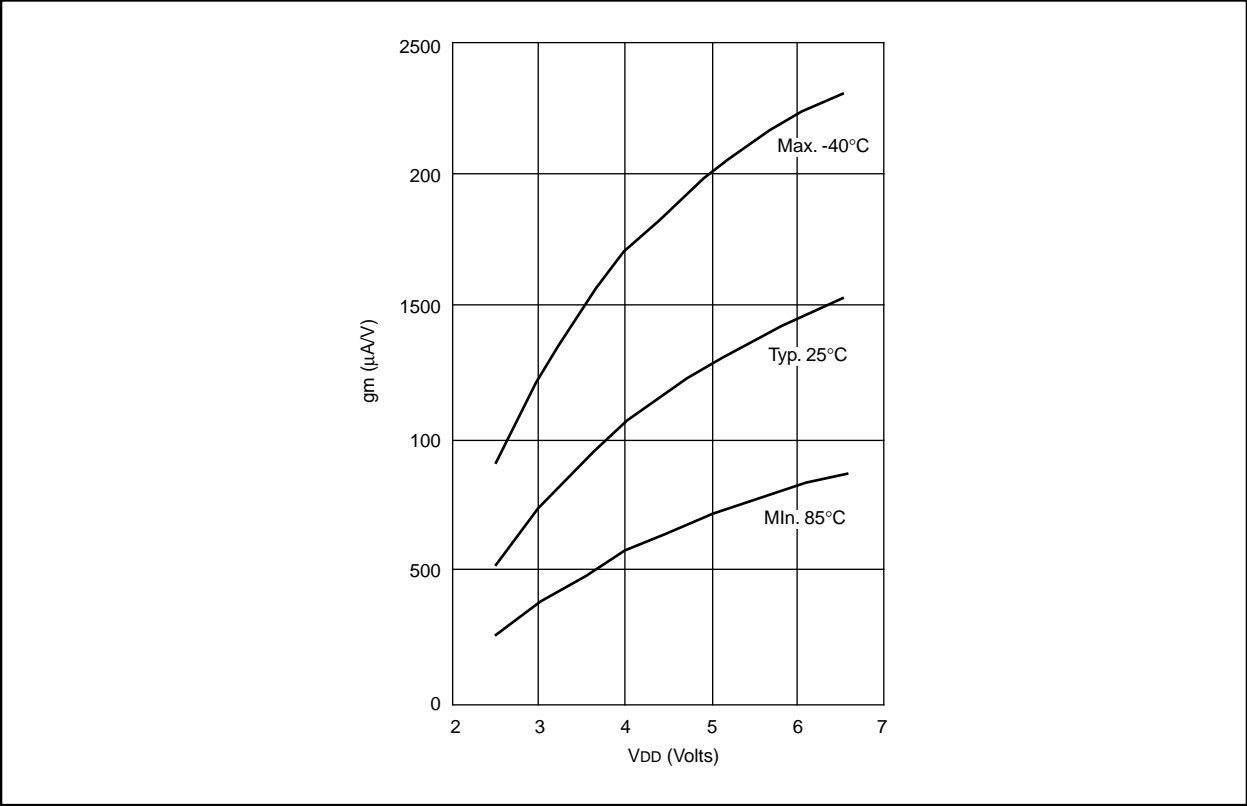
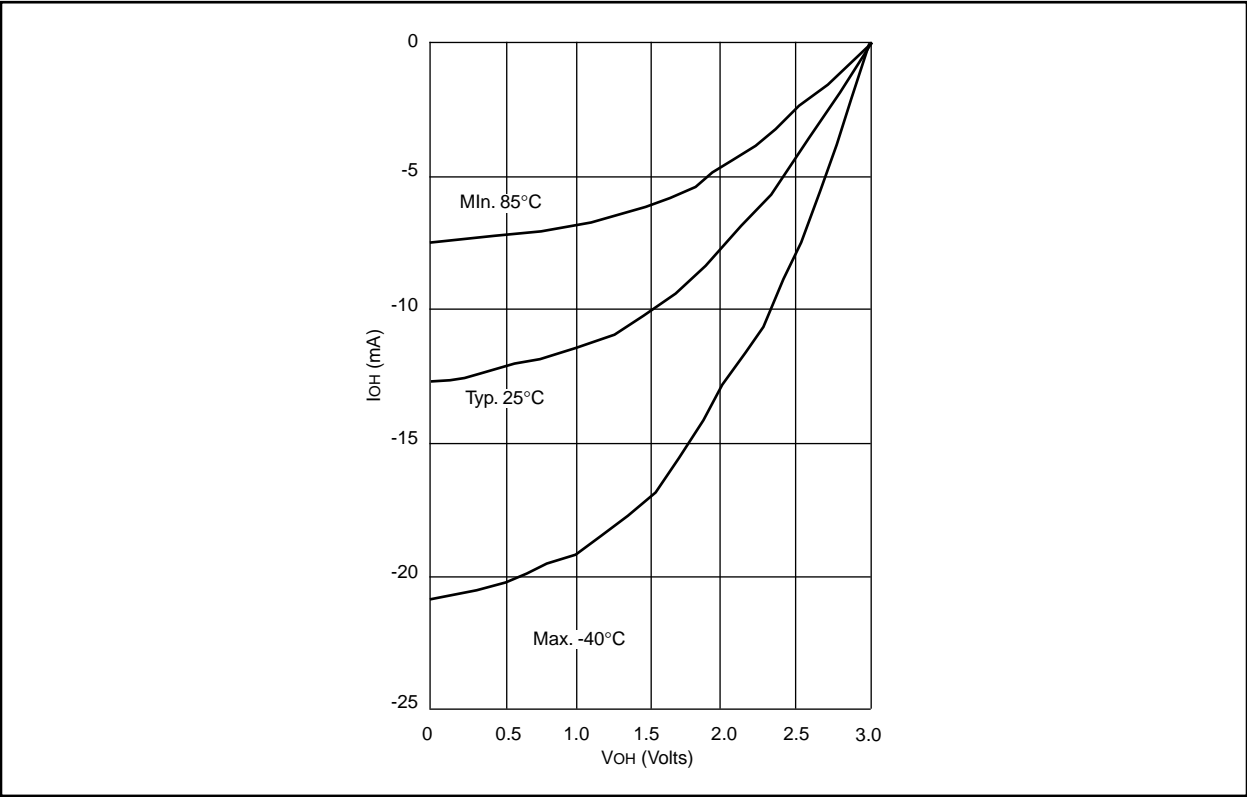


FIGURE 17-19: IOH vs. VOH, VDD = 3V





Applicable Devices	61	62	62A	R62	63	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

FIGURE 17-20: I<sub>OH</sub> vs. V<sub>OH</sub>, V<sub>DD</sub> = 5V

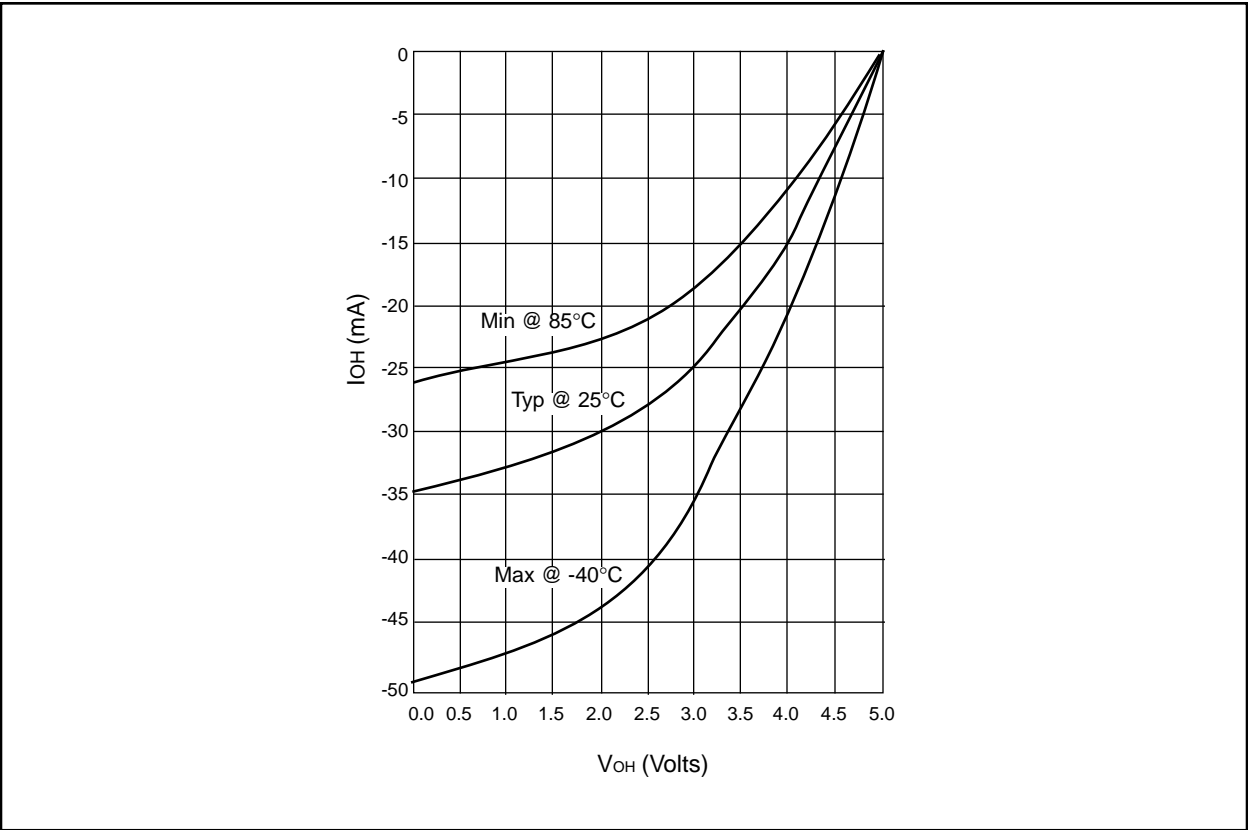
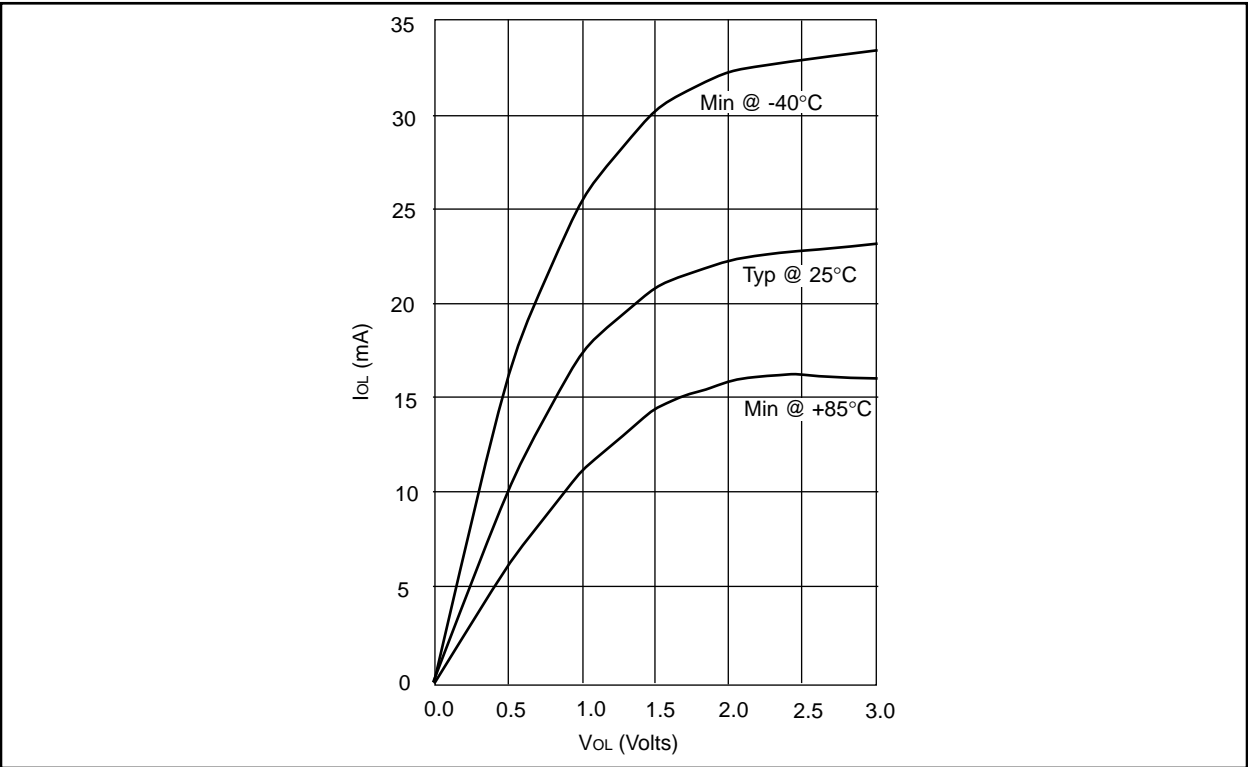


FIGURE 17-21: I<sub>OL</sub> vs. V<sub>OL</sub>, V<sub>DD</sub> = 3V



# PIC16C6X

Applicable Devices 61 62 62A R62 63 64 64A R64 65 65A

FIGURE 17-22: IOL vs. VOL, VDD = 5V

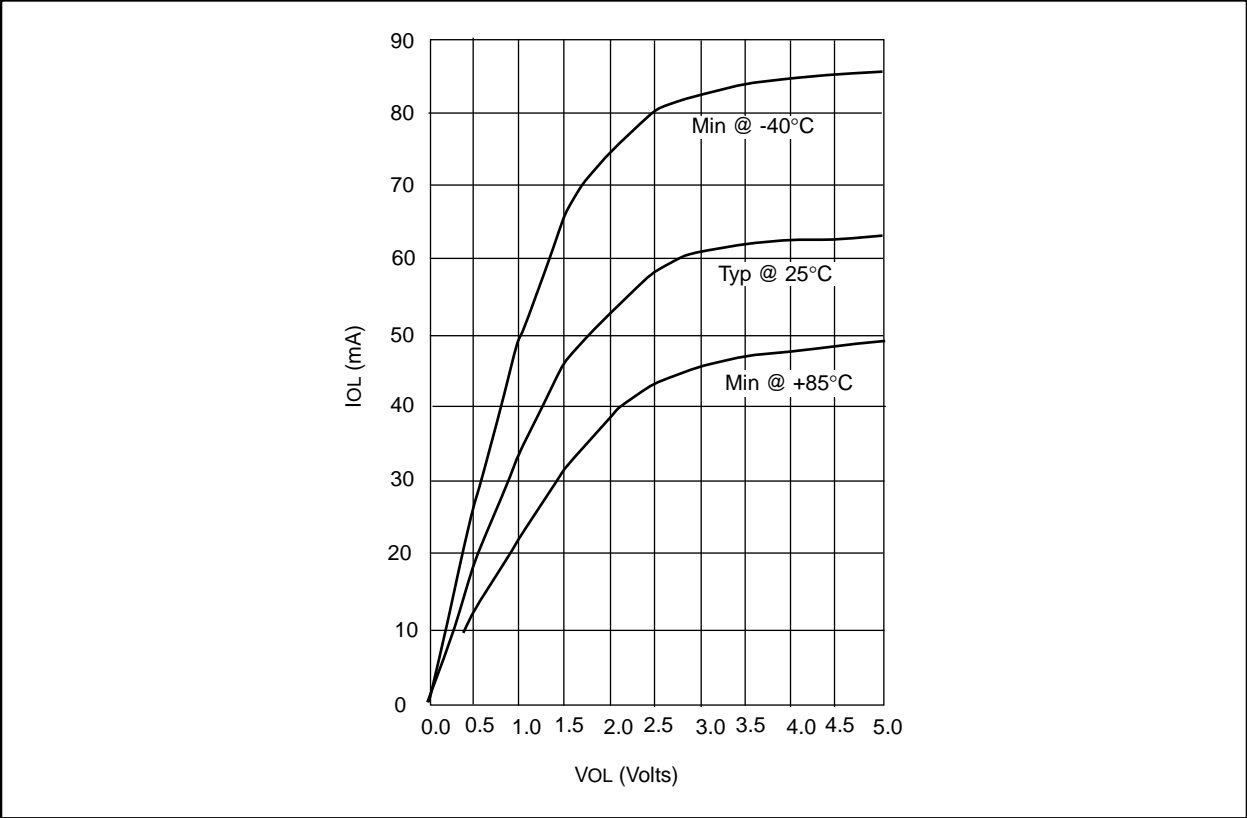


TABLE 17-2: INPUT CAPACITANCE\*

Pin Name	Typical Capacitance (pF)	
	18L PDIP	18L SOIC
RA port	5.0	4.3
RB port	5.0	4.3
MCLR	17.0	17.0
OSC1/CLKIN	4.0	3.5
OSC2/CLKOUT	4.3	3.5
T0CKI	3.2	2.8

\*All capacitance values are typical at 25°C. A part to part variation of ±25% (three standard deviations) should be taken into account.

Applicable Devices	61	62	62A	R62	63	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

## 18.0 ELECTRICAL CHARACTERISTICS FOR PIC16C62/64

### Absolute Maximum Ratings †

Ambient temperature under bias	-55 to +125°C
Storage temperature	-65°C to +150°C
Voltage on any pin with respect to VSS (except VDD and $\overline{\text{MCLR}}$ )	-0.3V to (VDD + 0.3V)
Voltage on VDD with respect to VSS	0 to +7.5V
Voltage on $\overline{\text{MCLR}}$ with respect to VSS (Note 2)	0 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 <sub>IK</sub> (V <sub>I</sub> < 0 or V <sub>I</sub> > VDD)	±20 mA
Output clamp current, I <sub>OK</sub> (V <sub>O</sub> < 0 or V <sub>O</sub> > 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:  $P_{dis} = V_{DD} \times \{I_{DD} - \sum I_{OH}\} + \sum \{(V_{DD} - V_{OH}) \times I_{OH}\} + \sum (V_{OL} \times I_{OL})$

**Note 2:** Voltage spikes below VSS at the  $\overline{\text{MCLR}}$  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  $\overline{\text{MCLR}}$  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 18-1: 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: 2.0 mA typ. at 3.0V IPD: 0.9 µA typ. 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: 2.0 mA typ. at 3.0V IPD: 0.9 µA typ. 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: 30 mA max. at 5.5V IPD: 1.5 µA typ. at 4.5V Freq: 20 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.	Do not 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.	Do not use in LP mode	Do not 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	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

**18.1 DC Characteristics:** **PIC16C62-04, PIC16C64-04 (Commercial, Industrial)**  
**PIC16C62-10, PIC16C64-10 (Commercial, Industrial)**  
**PIC16C62-20, PIC16C64-20 (Commercial, Industrial)**

Standard Operating Conditions (unless otherwise stated)							
DC CHARACTERISTICS							
Operating temperature    -40°C    ≤ TA ≤ +85°C for industrial and 0°C        ≤ TA ≤ +70°C for commercial							
Param No.	Characteristic	Sym	Min	Typ†	Max	Units	Conditions
D001 D001A	Supply Voltage	VDD	4.0 4.5	- -	6.0 5.5	V V	XT, RC and LP osc configuration HS osc configuration
D002*	RAM Data Retention Voltage (Note 1)	VDR	-	1.5	-	V	Device in SLEEP mode
D003	VDD start voltage to ensure Power-on Reset	VPOR	-	VSS	-	V	See section on Power-on Reset for details
D004*	VDD rise rate to ensure Power-on Reset	SVDD	0.05	-	-	V/ms	See section on Power-on Reset for details
D010  D013	Supply Current (Note 2, 5)	IDD	- -	2.7 13.5	5.0 30	mA mA	XT, RC, osc configuration (PIC16C62/64-04) FOSC = 4 MHz, VDD = 5.5V (Note 4)  HS osc configuration (PIC16C62/64-20) FOSC = 20 MHz, VDD = 5.5V
D020 D021 D021A	Power-down Current (Note 3, 5)	IPD	- - -	10.5 1.5 1.5	42 21 24	μA μA μA	VDD = 4.0V, WDT enabled, -40°C to +85°C VDD = 4.0V, WDT disabled, -0°C to +70°C VDD = 4.0V, WDT disabled, -40°C to +85°C

\* These parameters are characterized but not tested.

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

Note 1: This is the limit to which VDD can be lowered in SLEEP mode without losing RAM data.

2: The supply current is mainly a function of the operating voltage and frequency. Other factors such as I/O pin loading and switching rate, oscillator type, internal code execution pattern, and temperature also have an impact on the current consumption.

The test conditions for all IDD measurements in active operation mode are:

OSC1 = external square wave, from rail to rail; all I/O pins tristated, pulled to VDD

MCLR = VDD; WDT enabled/disabled as specified.

3: The power-down current in SLEEP mode does not depend on the oscillator type. Power-down current is measured with the part in SLEEP mode, with all I/O pins in hi-impedance state and tied to VDD and VSS.

4: For RC osc configuration, current through Rext is not included. The current through the resistor can be estimated by the formula  $I_r = V_{DD}/2R_{ext}$  (mA) with Rext in kOhm.

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

Applicable Devices	61	62	62A	R62	63	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

## 18.2 DC Characteristics: PIC16LC62-04, PIC16LC64-04 (Commercial, Industrial)

DC CHARACTERISTICS		Standard Operating Conditions (unless otherwise stated)					
		Operating temperature    -40°C    ≤ TA ≤ +85°C for industrial and 0°C        ≤ TA ≤ +70°C for commercial					
Param No.	Characteristic	Sym	Min	Typ†	Max	Units	Conditions
D001	Supply Voltage	VDD	3.0	-	6.0	V	LP, XT, RC osc configuration (DC - 4 MHz)
D002*	RAM Data Retention Voltage (Note 1)	VDR	-	1.5	-	V	Device in SLEEP mode
D003	VDD start voltage to ensure Power-on Reset	VPOR	-	VSS	-	V	See section on Power-on Reset for details
D004*	VDD rise rate to ensure Power-on Reset	SVDD	0.05	-	-	V/ms	See section on Power-on Reset for details
D010	Supply Current (Note 2, 5)	IDD	-	2.0	3.8	mA	XT, RC osc configuration FOSC = 4 MHz, VDD = 3.0V (Note 4)
D010A			-	22.5	48	μA	LP osc configuration FOSC = 32 kHz, VDD = 3.0V, WDT disabled
D020	Power-down Current (Note 3, 5)	IPD	-	7.5	30	μA	VDD = 3.0V, WDT enabled, -40°C to +85°C
D021			-	0.9	13.5	μA	VDD = 3.0V, WDT disabled, 0°C to +70°C
D021A			-	0.9	18	μA	VDD = 3.0V, WDT disabled, -40°C to +85°C

\* These parameters are characterized but not tested.

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

Note 1: This is the limit to which VDD can be lowered in SLEEP mode without losing RAM data.

2: The supply current is mainly a function of the operating voltage and frequency. Other factors such as I/O pin loading and switching rate, oscillator type, internal code execution pattern, and temperature also have an impact on the current consumption.

The test conditions for all IDD measurements in active operation mode are:

OSC1 = external square wave, from rail to rail; all I/O pins tristated, pulled to VDD

MCLR = VDD; WDT enabled/disabled as specified.

3: The power-down current in SLEEP mode does not depend on the oscillator type. Power-down current is measured with the part in SLEEP mode, with all I/O pins in hi-impedance state and tied to VDD and VSS.

4: For RC osc configuration, current through Rext is not included. The current through the resistor can be estimated by the formula  $I_r = VDD/2R_{ext}$  (mA) with Rext in kOhm.

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

# PIC16C6X

Applicable Devices	61	62	62A	R62	63	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

**18.3 DC Characteristics: PIC16C62-04, PIC16C64-04 (Commercial, Industrial)**  
**PIC16C62-10, PIC16C64-10 (Commercial, Industrial)**  
**PIC16C62-20, PIC16C64-20 (Commercial, Industrial)**  
**PIC16LC62-04, PIC16LC64-04 (Commercial, Industrial)**

<b>DC CHARACTERISTICS</b> <b>Standard Operating Conditions (unless otherwise stated)</b> Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for industrial and $0^{\circ}\text{C} \leq T_A \leq +70^{\circ}\text{C}$ for commercial Operating voltage $V_{DD}$ range as described in DC spec Section 18.1 and Section 18.2							
Param No.	Characteristic	Sym	Min	Typ †	Max	Units	Conditions
D030 D031 D032 D033	<b>Input Low Voltage</b> I/O ports with TTL buffer with Schmitt Trigger buffer $\overline{\text{MCLR}}$ , RA4/T0CKI, OSC1 (in RC mode) OSC1 (in XT, HS and LP)	$V_{IL}$	$V_{SS}$ $V_{SS}$ $V_{SS}$ $V_{SS}$	- - - -	0.5V 0.2V <sub>DD</sub> 0.2V <sub>DD</sub> 0.3V <sub>DD</sub>	V V V V	Note1
D040 D040A D041 D042 D042A D043	<b>Input High Voltage</b> I/O ports with TTL buffer with Schmitt Trigger buffer $\overline{\text{MCLR}}$ , RA4/T0CKI, RC7:RC4, RD7:RD4, RB0/INT RE2:RE0, OSC1 (XT, HS and LP) OSC1 (in RC mode)	$V_{IH}$	2.0 0.8V <sub>DD</sub> 0.8V <sub>DD</sub> 0.8V <sub>DD</sub> 0.7V <sub>DD</sub> 0.9V <sub>DD</sub>	- - - - - -	V <sub>DD</sub> V <sub>DD</sub> V <sub>DD</sub> V <sub>DD</sub> V <sub>DD</sub> V <sub>DD</sub>	V V V V V V	$4.5\text{V} \leq V_{DD} \leq 5.5\text{V}$ For $V_{DD} > 5.5\text{V}$ or $V_{DD} < 4.5\text{V}$ For entire $V_{DD}$ range Note1
D070	PORTB weak pull-up current	IPURB	50	200	†400	μA	$V_{DD} = 5\text{V}$ , $V_{PIN} = V_{SS}$
D060 D061 D063	<b>Input Leakage Current</b> (Notes 2, 3) I/O ports $\overline{\text{MCLR}}$ , RA4/T0CKI OSC1	$I_{IL}$	- - -	- - -	±1 ±5 ±5	μA μA μA	$V_{SS} \leq V_{PIN} \leq V_{DD}$ , Pin at hi-impedance $V_{SS} \leq V_{PIN} \leq V_{DD}$ $V_{SS} \leq V_{PIN} \leq V_{DD}$ , XT, HS and LP osc configuration
D080 D083 D090 D092	<b>Output Low Voltage</b> I/O ports OSC2/CLKOUT (RC osc config)  <b>Output High Voltage</b> I/O ports (Note 3) OSC2/CLKOUT (RC osc config)	$V_{OL}$   $V_{OH}$	- -  $V_{DD}-0.7$	- -  -	0.6 0.6  -	V V  V	$I_{OL} = 8.5\text{ mA}$ , $V_{DD} = 4.5\text{V}$ , -40°C to +85°C $I_{OL} = 1.6\text{ mA}$ , $V_{DD} = 4.5\text{V}$ , -40°C to +85°C $I_{OH} = -3.0\text{ mA}$ , $V_{DD} = 4.5\text{V}$ , -40°C to +85°C $I_{OH} = -1.3\text{ mA}$ , $V_{DD} = 4.5\text{V}$ , -40°C to +85°C

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

- Note 1: In RC oscillator configuration, the OSC1/CLKIN pin is a Schmitt Trigger input. It is not recommended that the PIC16C6X be driven with external clock in RC mode.
- 2: The leakage current on the  $\overline{\text{MCLR}}$ /VPP pin is strongly dependent on the applied voltage level. The specified levels represent normal operating conditions. Higher leakage current may be measured at different input voltages.
- 3: Negative current is defined as coming out of the pin.

# PIC16C6X

<b>Applicable Devices</b>	61	62	62A	R62	63	64	64A	R64	65	65A
---------------------------	----	----	-----	-----	----	----	-----	-----	----	-----

<b>DC CHARACTERISTICS</b>							
<b>Standard Operating Conditions (unless otherwise stated)</b> Operating temperature    -40°C    ≤ TA ≤ +85°C for industrial and 0°C        ≤ TA ≤ +70°C for commercial Operating voltage VDD range as described in DC spec Section 18.1 and Section 18.2							
Param No.	Characteristic	Sym	Min	Typ †	Max	Units	Conditions
D100	<b>Capacitive Loading Specs on Output Pins</b> OSC2 pin	Cosc2	-	-	15	pF	In XT, HS and LP modes when external clock is used to drive OSC1.
D101	All I/O pins and OSC2 (in RC mode)	Cio	-	-	50	pF	
D102	SCL, SDA in I²C mode	Cb	-	-	400	pF	

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

Note 1: In RC oscillator configuration, the OSC1/CLKIN pin is a Schmitt Trigger input. It is not recommended that the PIC16C6X be driven with external clock in RC mode.

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

3: Negative current is defined as coming out of the pin.

# PIC16C6X

Applicable Devices	61	62	62A	R62	63	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

## 18.4 Timing Parameter Symbolology

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

1. TppS2ppS
2. TppS
3. TCC:ST (I<sup>2</sup>C specifications only)
4. Ts (I<sup>2</sup>C specifications only)

<b>T</b>			
F	Frequency	T	Time

Lowercase letters (pp) and their meanings:

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

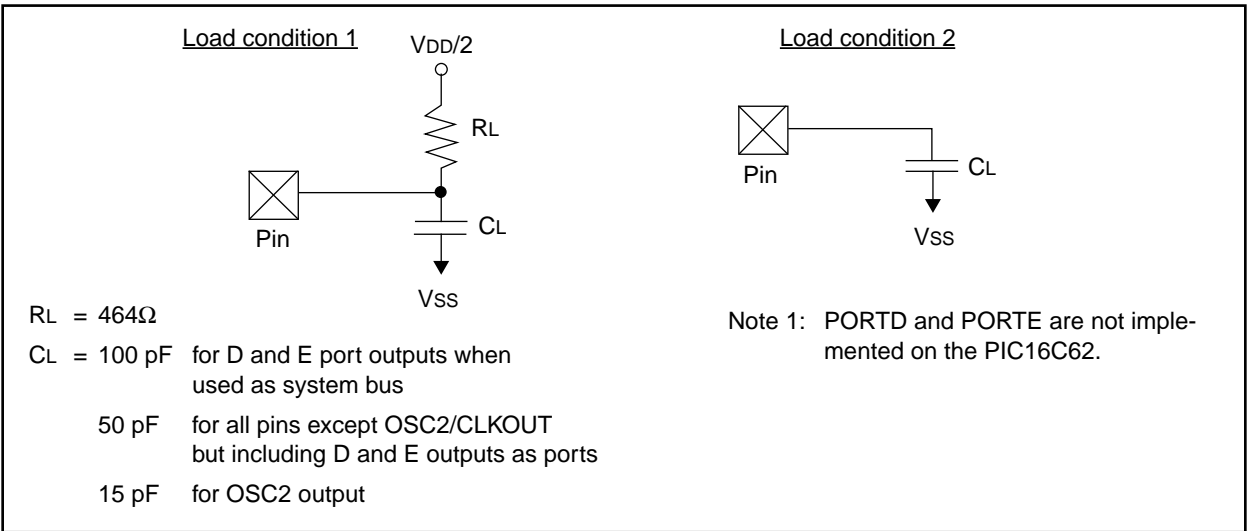
Uppercase letters and their meanings:

<b>S</b>			
F	Fall	P	Period
H	High	R	Rise
I	Invalid (Hi-impedance)	V	Valid
L	Low	Z	Hi-impedance
<b>I<sup>2</sup>C only</b>			
AA	output access	High	High
BUF	Bus free	Low	Low

TCC:ST (I<sup>2</sup>C specifications only)

<b>CC</b>			
HD	Hold	SU	Setup
<b>ST</b>			
DAT	DATA input hold	STO	STOP condition
STA	START condition		

FIGURE 18-1: LOAD CONDITIONS FOR DEVICE TIMING SPECIFICATIONS

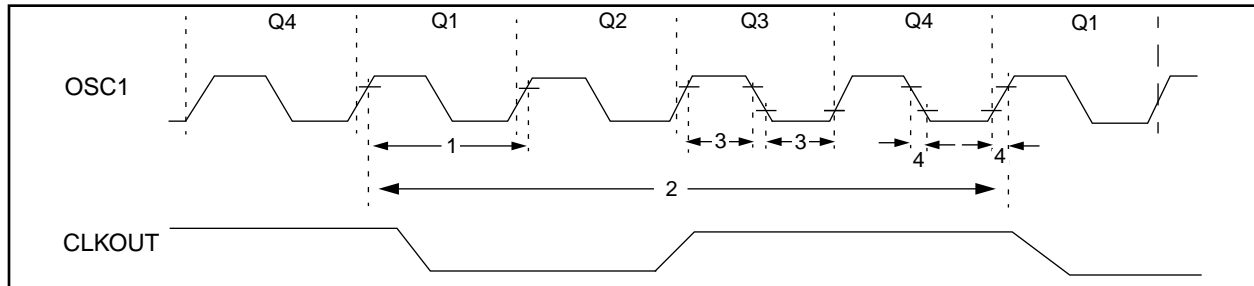




Applicable Devices	61	62	62A	R62	63	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

## 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	<b>External CLKIN Frequency (Note 1)</b>	DC	—	4	MHz	XT and RC osc mode
			DC	—	4	MHz	HS osc mode (PIC16C62/64-04,
			DC	—	20	MHz	HS osc mode (PIC16C62/64-20)
			DC	—	200	kHz	LP osc mode
		<b>Oscillator Frequency (Note 1)</b>	DC	—	4	MHz	RC osc mode
			0.1	—	4	MHz	XT osc mode
			4	—	4	MHz	HS osc mode (PIC16C62/64-04,
			4	—	10	MHz	HS osc mode (PIC16C62/64-10)
			4	—	20	MHz	HS osc mode (PIC16C62/64-20)
			5	—	200	kHz	LP osc mode
1	Tosc	<b>External CLKIN Period (Note 1)</b>	250	—	—	ns	XT and RC osc mode
			250	—	—	ns	HS osc mode (PIC16C62/64-04,
			100	—	—	ns	HS osc mode (PIC16C62/64-10)
			50	—	—	ns	HS osc mode (PIC16C62/64-20)
			5	—	—	μs	LP osc mode
		<b>Oscillator Period (Note 1)</b>	250	—	—	ns	RC osc mode
			250	—	10,000	ns	XT osc mode
			250	—	250	ns	HS osc mode (PIC16C62/64-04,
			100	—	250	ns	HS osc mode (PIC16C62/64-10)
			50	—	1,000	ns	HS osc mode (PIC16C62/64-20)
2	Tcy	<b>Instruction Cycle Time (Note 1)</b>	200	—	DC	ns	Tcy = 4/Fosc
			200	—	DC	ns	Tcy = 4/Fosc
3	TosL, TosH	<b>External Clock in (OSC1) High or Low Time</b>	50	—	—	ns	XT oscillator
			2.5	—	—	μs	LP oscillator
			10	—	—	ns	HS oscillator
4	TosR, TosF	<b>External Clock in (OSC1) Rise or Fall Time</b>	—	—	25	ns	XT oscillator
			—	—	50	ns	LP oscillator
			—	—	15	ns	HS oscillator

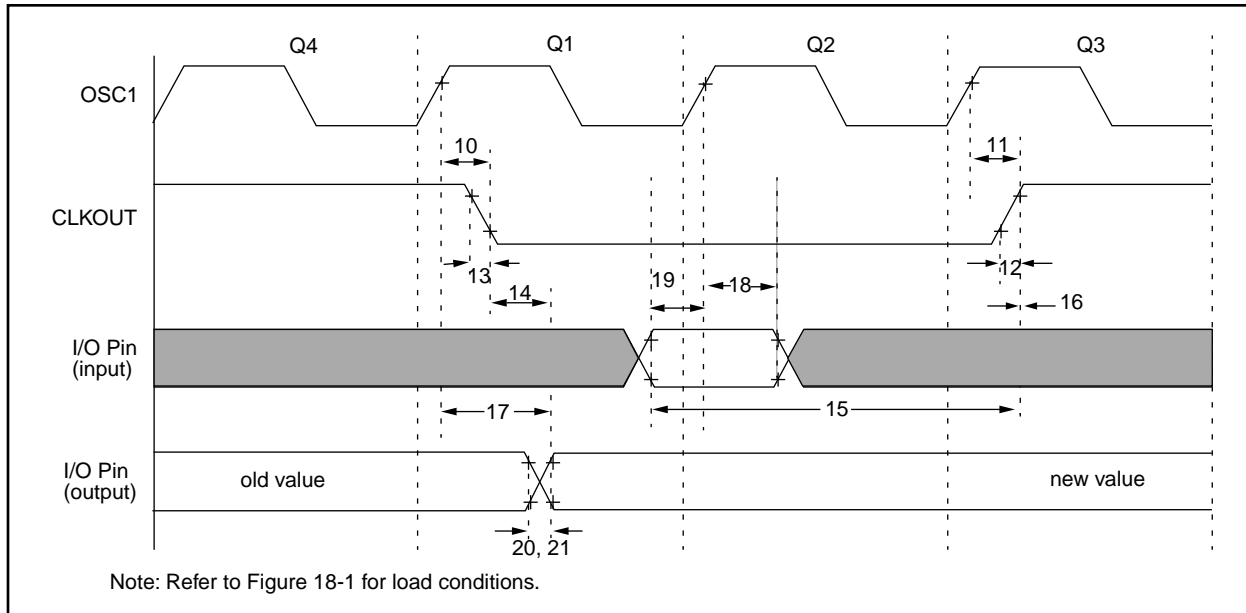
† Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

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

# PIC16C6X

Applicable Devices 61 62 62A R62 63 64 64A R64 65 65A

**FIGURE 18-3: CLKOUT AND I/O TIMING**



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

Parameters	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
10*	TosH2ckL	OSC1↑ to CLKOUT↓	—	15	30	ns	Note 1
11*	TosH2ckH	OSC1↑ to CLKOUT↑	—	15	30	ns	Note 1
12*	TckR	CLKOUT rise time	—	5	15	ns	Note 1
13*	TckF	CLKOUT fall time	—	5	15	ns	Note 1
14*	TckL2ioV	CLKOUT ↓ to Port out valid	—	—	0.5T <sub>CY</sub> + 20	ns	Note 1
15*	TioV2ckH	Port in valid before CLKOUT ↑	0.25T <sub>CY</sub> + 25	—	—	ns	Note 1
16*	TckH2ioL	Port in hold after CLKOUT ↑	0	—	—	ns	Note 1
17*	TosH2ioV	OSC1↑ (Q1 cycle) to Port out valid	—	—	80 - 100	ns	
18*	TosH2ioL	OSC1↑ (Q2 cycle) to Port input invalid (I/O in hold time)	TBD	—	—	ns	
19*	TioV2osH	Port input valid to OSC1↑ (I/O in setup time)	TBD	—	—	ns	
20*	TioR	Port output rise time	PIC16C62/64	—	10	25	ns
			PIC16LC62/64	—	—	60	ns
21*	TioF	Port output fall time	PIC16C62/64	—	10	25	ns
			PIC16LC62/64	—	—	60	ns
22††*	Tinp	INT pin high or low time	20	—	—	ns	
23††*	Trbp	RB7:RB4 change INT high or low time	20	—	—	ns	

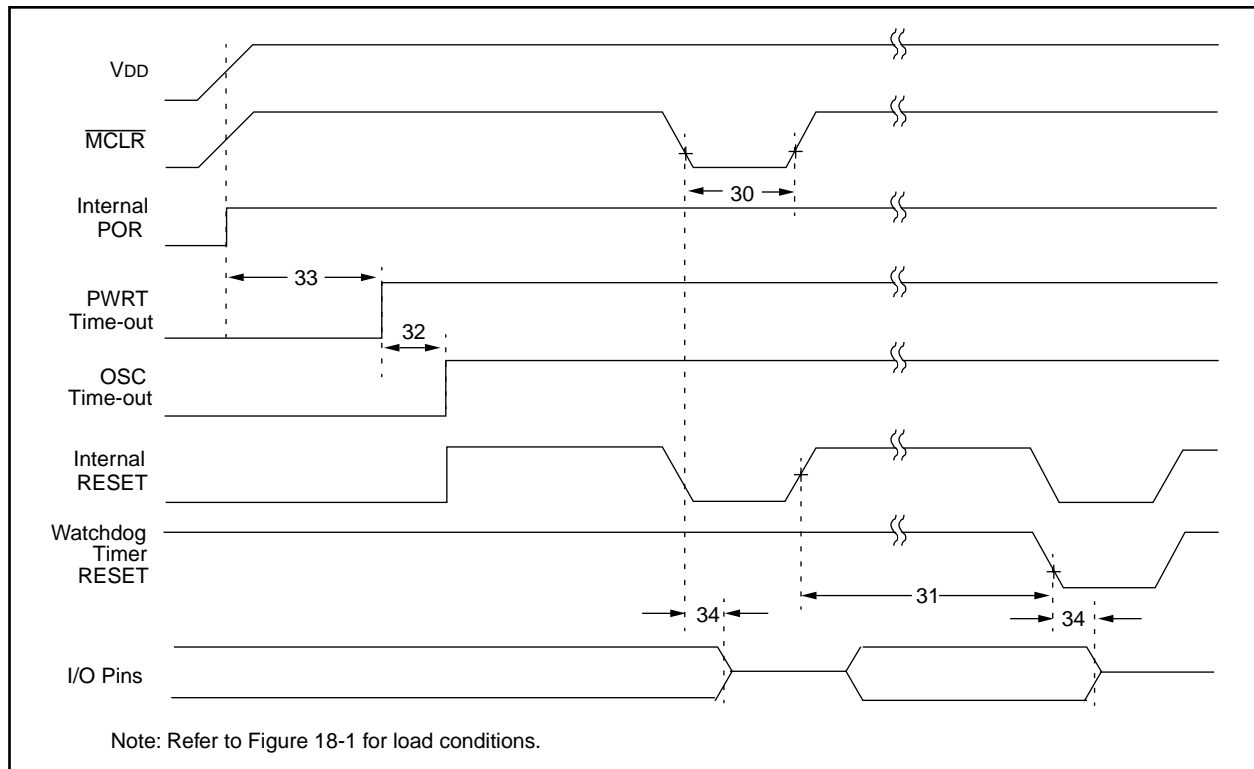
\* These parameters are characterized but not tested.

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

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

Note 1: Measurements are taken in RC Mode where CLKOUT output is 4 x T<sub>osc</sub>.

**FIGURE 18-4: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER AND POWER-UP TIMER TIMING**



**TABLE 18-4: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER AND POWER-UP TIMER REQUIREMENTS**

Parameter No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
30*	Tmcl	MCLR Pulse Width (low)	100	—	—	ns	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	—	1024Tosc	—	—	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	—	—	100	ns	

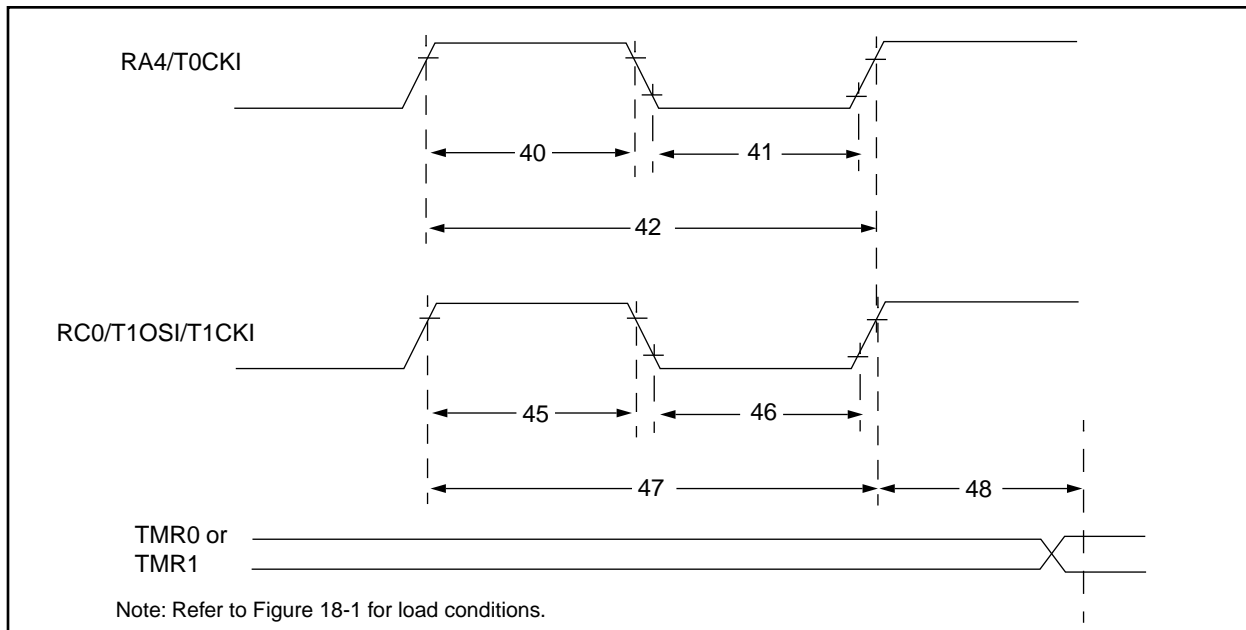
\* These parameters are characterized but not tested.

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

# PIC16C6X

Applicable Devices 61 62 62A R62 63 64 64A R64 65 65A

**FIGURE 18-5: TIMER0 AND TIMER1 CLOCK TIMINGS**



**TABLE 18-5: TIMER0 AND TIMER1 CLOCK REQUIREMENTS**

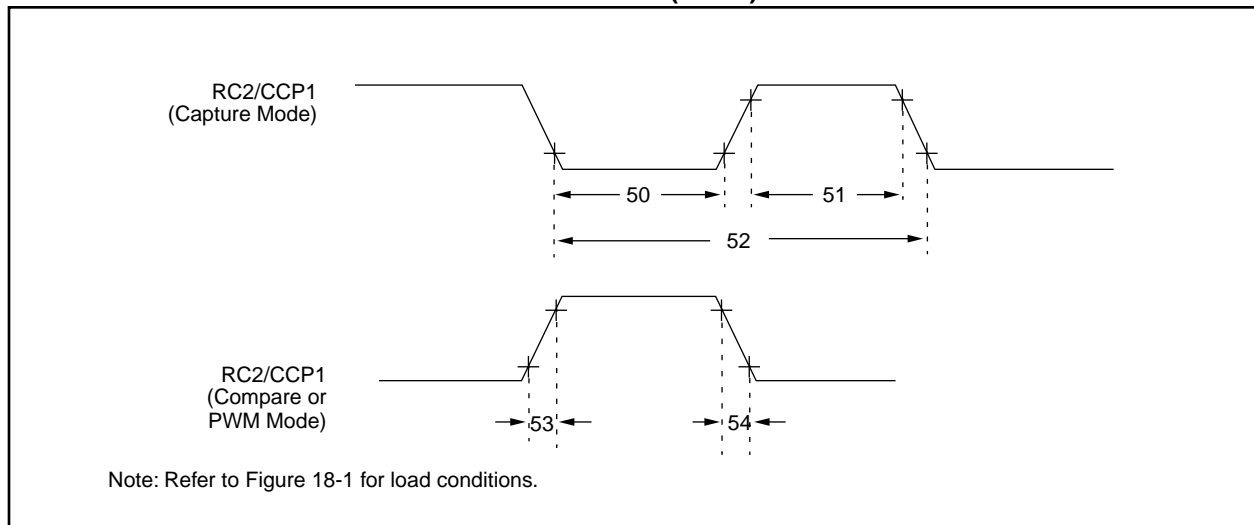
Param No.	Sym	Characteristic		Min	Typ†	Max	Units	Conditions
40*	Tt0H	T0CKI High Pulse Width	No Prescaler	$0.5T_{CY} + 20$	—	—	ns	
			With Prescaler	10	—	—	ns	
41*	Tt0L	T0CKI Low Pulse Width	No Prescaler	$0.5T_{CY} + 20$	—	—	ns	
			With Prescaler	10	—	—	ns	
42*	Tt0P	T0CKI Period		Greater of: $20 \mu s$ or $\frac{T_{CY} + 40}{N}$	—	—	ns	N = prescale value (1, 2, 4, ..., 256)
45*	Tt1H	T1CKI High Time	Synchronous, No Prescaler	$0.5T_{CY} + 20$	—	—	ns	
			Synchronous, PIC16C62/64 With Prescaler	10	—	—	ns	
			PIC16LC62/64	20	—	—	ns	
			Asynchronous	$2T_{CY}$	—	—	ns	
46*	Tt1L	T1CKI Low Time	Synchronous, No Prescaler	$0.5T_{CY} + 20$	—	—	ns	
			Synchronous, PIC16C62/64 With Prescaler	10	—	—	ns	
			PIC16LC62/64	20	—	—	ns	
			Asynchronous	$2T_{CY}$	—	—	ns	
47*	Tt1P	T1CKI input period	Synchronous	Greater of: $20 \mu s$ or $\frac{T_{CY} + 40}{N}$	—	—	ns	N = prescale value (1, 2, 4, 8)
			Asynchronous	Greater of: $20 \mu s$ or $4T_{CY}$	—	—	ns	
	Ft1	Timer1 oscillator input frequency range (oscillator enabled by setting bit T1OSCEN)		DC	—	200	kHz	
48	TCKEztmr1	Delay from external clock edge to timer increment		$2T_{osc}$	—	$7T_{osc}$	—	

\* These parameters are characterized but not tested.

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

Applicable Devices	61	62	62A	R62	63	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

**FIGURE 18-6: CAPTURE/COMPARE/PWM TIMINGS (CCP1)**



**TABLE 18-6: CAPTURE/COMPARE/PWM REQUIREMENTS (CCP1)**

Parameter No.	Sym	Characteristic			Min	Typ†	Max	Units	Conditions
50*	TccL	CCP1 input low time	No Prescaler		0.5Tcy + 20	—	—	ns	
			With Prescaler	PIC16C62/64	10	—	—	ns	
				PIC16LC62/64	20	—	—	ns	
51*	TccH	CCP1 input high time	No Prescaler		0.5Tcy + 20	—	—	ns	
			With Prescaler	PIC16C62/64	10	—	—	ns	
				PIC16LC62/64	20	—	—	ns	
52*	TccP	CCP1 input period			$\frac{3Tcy + 40}{N}$	—	—	ns	N = prescale value (1,4 or 16)
53	TccR	CCP1 output rise time			—	10	25	ns	
54	TccF	CCP1 output fall time			—	10	25	ns	

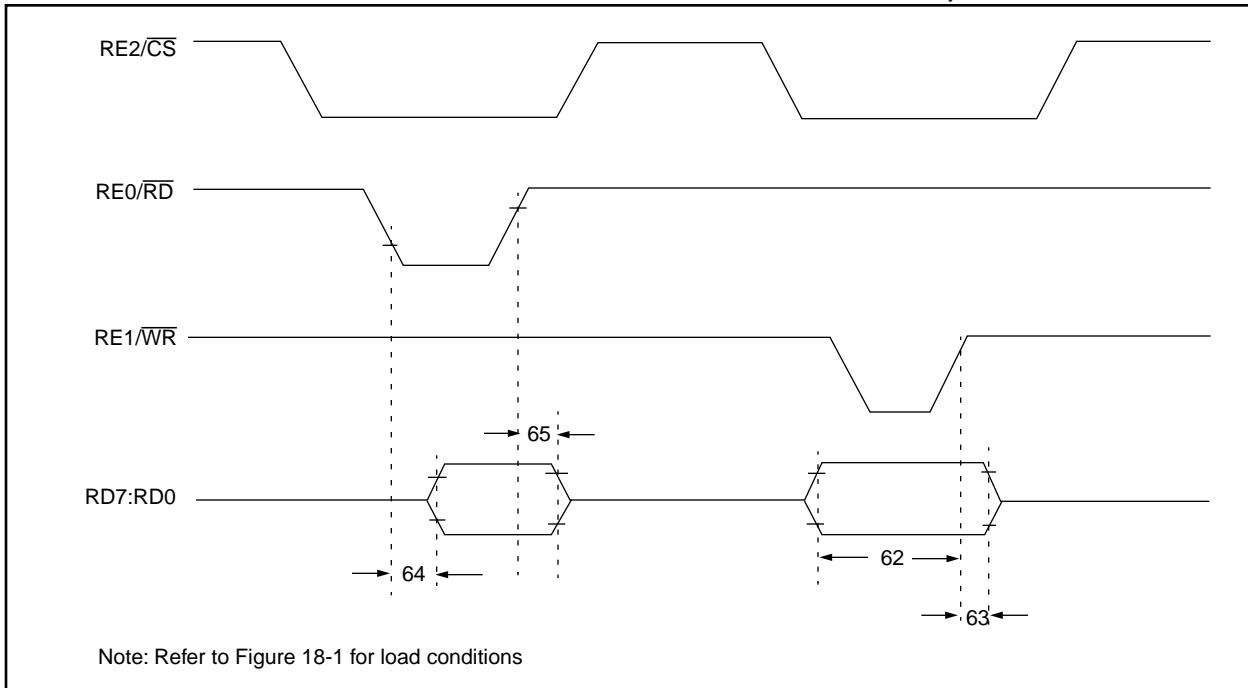
\* These parameters are characterized but not tested.

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

# PIC16C6X

Applicable Devices 61 62 62A R62 63 64 64A R64 65 65A

**FIGURE 18-7: PARALLEL SLAVE PORT TIMING FOR THE PIC16C64 ONLY)**

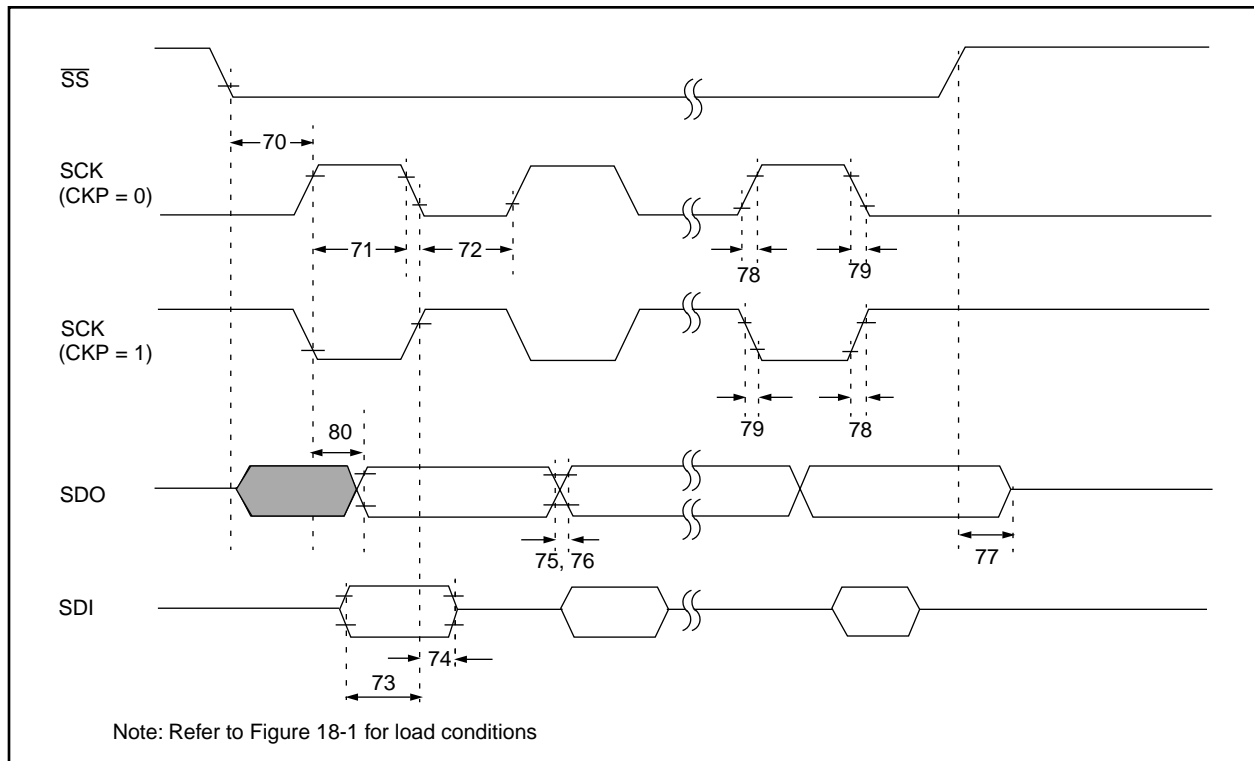


**TABLE 18-7: PARALLEL SLAVE PORT REQUIREMENTS FOR THE PIC16C64 ONLY**

Parameter No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
62	TdtV2wrH	Data in valid before $\overline{WR}\uparrow$ or $\overline{CS}\uparrow$ (setup time)	20	—	—	ns	
63*	TwrH2dtI	$\overline{WR}\uparrow$ or $\overline{CS}\uparrow$ to data-in invalid (hold time)	PIC16C64	20	—	—	ns
			PIC16LC64	35	—	—	ns
64	TrdL2dtV	$\overline{RD}\downarrow$ and $\overline{CS}\downarrow$ to data-out valid	—	—	60	ns	
65	TrdH2dtI	$\overline{RD}\uparrow$ or $\overline{CS}\uparrow$ to data-out invalid	10	—	30	ns	

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

**FIGURE 18-8: SPI MODE TIMING**



**TABLE 18-8: SPI MODE REQUIREMENTS**

Parameter No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
70	TssL2scH, TssL2scL	$\overline{SS}$ ↓ to SCK↓ or SCK↑ input	T <sub>CY</sub>	—	—	ns	
71	TscH	SCK input high time (slave mode)	T <sub>CY</sub> + 20	—	—	ns	
72	TscL	SCK input low time (slave mode)	T <sub>CY</sub> + 20	—	—	ns	
73	TdiV2scH, TdiV2scL	Setup time of SDI data input to SCK edge	100	—	—	ns	
74	Tsch2diL, TscL2diL	Hold time of SDI data input to SCK edge	100	—	—	ns	
75	TdoR	SDO data output rise time	—	10	25	ns	
76	TdoF	SDO data output fall time	—	10	25	ns	
77	TssH2doZ	$\overline{SS}$ ↓ to SDO output hi-impedance	10	—	50	ns	
78	TscR	SCK output rise time (master mode)	—	10	25	ns	
79	TscF	SCK output fall time (master mode)	—	10	25	ns	
80	Tsch2doV, TscL2doV	SDO data output valid after SCK edge	—	—	50	ns	

† 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	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

FIGURE 18-9: I<sup>2</sup>C BUS START/STOP BITS TIMING

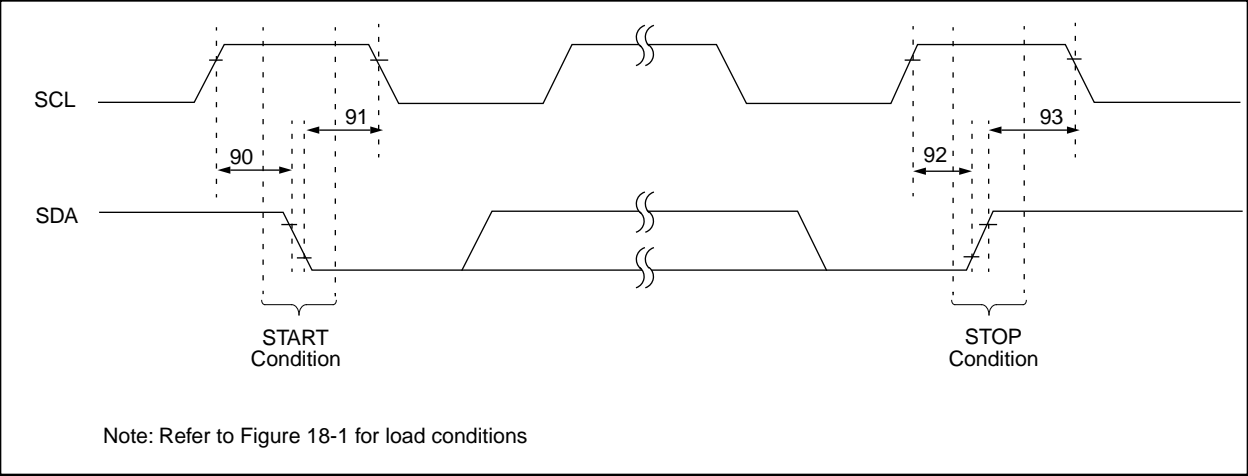


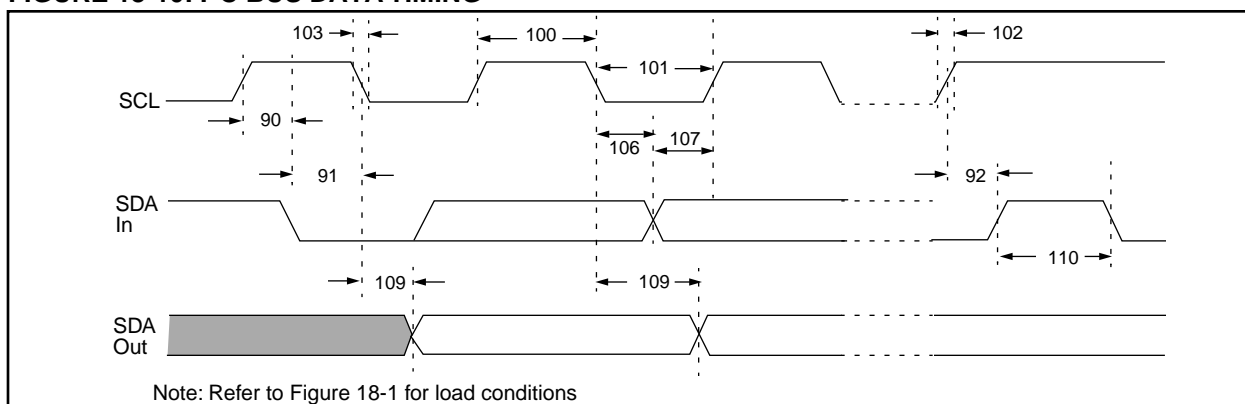
TABLE 18-9: I<sup>2</sup>C BUS START/STOP BITS REQUIREMENTS

Parameter No.	Sym	Characteristic		Min	Typ	Max	Units	Conditions
90	TSU:STA	START condition	100 kHz mode	4700	—	—	ns	Only relevant for repeated START condition
		Setup time	400 kHz mode	600	—	—		
91	THD:STA	START condition	100 kHz mode	4000	—	—	ns	After this period the first clock pulse is generated
		Hold time	400 kHz mode	600	—	—		
92	TSU:STO	STOP condition	100 kHz mode	4700	—	—	ns	
		Setup time	400 kHz mode	600	—	—		
93	THD:STO	STOP condition	100 kHz mode	4000	—	—	ns	
		Hold time	400 kHz mode	600	—	—		



Applicable Devices	61	62	62A	R62	63	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

**FIGURE 18-10: I<sup>2</sup>C BUS DATA TIMING**



**TABLE 18-10: I<sup>2</sup>C BUS DATA REQUIREMENTS**

Parameter No.	Sym	Characteristic		Min	Max	Units	Conditions
100	T <sub>HIGH</sub>	Clock high time	100 kHz mode	4.0	—	μs	PIC16C64 must operate at a minimum of 1.5 MHz
			400 kHz mode	0.6	—	μs	PIC16C64 must operate at a minimum of 10 MHz
			SSP Module	1.5T <sub>CY</sub>	—		
101	T <sub>LOW</sub>	Clock low time	100 kHz mode	4.7	—	μs	PIC16C64 must operate at a minimum of 1.5 MHz
			400 kHz mode	1.3	—	μs	PIC16C64 must operate at a minimum of 10 MHz
			SSP Module	1.5T <sub>CY</sub>	—		
102	T <sub>R</sub>	SDA and SCL rise time	100 kHz mode	—	1000	ns	
			400 kHz mode	20 + 0.1C <sub>b</sub>	300	ns	C <sub>b</sub> is specified to be from 10 to 400 pF
103	T <sub>F</sub>	SDA and SCL fall time	100 kHz mode	—	300	ns	
			400 kHz mode	20 + 0.1C <sub>b</sub>	300	ns	C <sub>b</sub> is specified to be from 10 to 400 pF
90	T <sub>SU:STA</sub>	START condition setup time	100 kHz mode	4.7	—	μs	Only relevant for repeated START condition
			400 kHz mode	0.6	—	μs	
91	T <sub>HD:STA</sub>	START condition hold time	100 kHz mode	4.0	—	μs	After this period the first clock pulse is generated
			400 kHz mode	0.6	—	μs	
106	T <sub>HD:DAT</sub>	Data input hold time	100 kHz mode	0	—	ns	
			400 kHz mode	0	0.9	μs	
107	T <sub>SU:DAT</sub>	Data input setup time	100 kHz mode	250	—	ns	Note 2
			400 kHz mode	100	—	ns	
92	T <sub>SU:STO</sub>	STOP condition setup time	100 kHz mode	4.7	—	μs	
			400 kHz mode	0.6	—	μs	
109	T <sub>AA</sub>	Output valid from clock	100 kHz mode	—	3500	ns	Note 1
			400 kHz mode	—	—	ns	
110	T <sub>BUF</sub>	Bus free time	100 kHz mode	4.7	—	μs	Time the bus must be free before a new transmission can start
			400 kHz mode	1.3	—	μs	
	C <sub>b</sub>	Bus capacitive loading		—	400	pF	

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

2: A fast-mode I<sup>2</sup>C-bus device can be used in a standard-mode I<sup>2</sup>C-bus system, but the requirement t<sub>su</sub>;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 T<sub>R</sub> max.+t<sub>su</sub>;DAT = 1000 + 250 = 1250 ns (according to the standard-mode I<sup>2</sup>C bus specification) before the SCL line is released.

# PIC16C6X

---

NOTES:

Applicable Devices	61	62	62A	R62	63	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

19.0 DC AND AC CHARACTERISTICS GRAPHS AND TABLES FOR PIC16C62/64

NOT AVAILABLE AT THIS TIME

# PIC16C6X

---

NOTES:

Applicable Devices	61	62	62A	R62	63	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

## 20.0 ELECTRICAL CHARACTERISTICS FOR PIC16C62A/R62/64A/R64

### Absolute Maximum Ratings †

Ambient temperature under bias	-55 to +125°C
Storage temperature	-65°C to +150°C
Voltage on any pin with respect to VSS (except VDD and $\overline{\text{MCLR}}$ )	-0.3V to (VDD + 0.3V)
Voltage on VDD with respect to VSS	0 to +7.5V
Voltage on $\overline{\text{MCLR}}$ with respect to VSS (Note 2)	0 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 <sub>IK</sub> (V <sub>I</sub> < 0 or V <sub>I</sub> > VDD)	±20 mA
Output clamp current, I <sub>OK</sub> (V <sub>O</sub> < 0 or V <sub>O</sub> > 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

**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  $\overline{\text{MCLR}}$  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  $\overline{\text{MCLR}}$  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 20-1: CROSS REFERENCE OF DEVICE SPECS FOR OSCILLATOR CONFIGURATIONS AND FREQUENCIES OF OPERATION (COMMERCIAL DEVICES)**

OSC	PIC16C62A-04 PIC16CR62-04 PIC16C64A-04 PIC16CR64-04	PIC16C62A-10 PIC16CR62-10 PIC16C64A-10 PIC16CR64-10	PIC16C62A-20 PIC16CR62-20 PIC16C64A-20 PIC16CR64-20	PIC16LC62A-04 PIC16LCR62-04 PIC16LC64A-04 PIC16LCR64-04	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: 2.5V to 6.0V IDD: 2.0 mA typ. at 3.0V IPD: 0.9 µA typ. 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: 2.5V to 6.0V IDD: 2.0 mA typ. at 3.0V IPD: 0.9 µA typ. 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: 30 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.	Do not 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.	Do not use in LP mode	Do not use in LP mode	VDD: 2.5V 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: 2.5V 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	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

**20.1 DC Characteristics:** **PIC16C62A-04,PIC16C64A-04 (Commercial, Industrial, Automotive)**  
**PIC16CR62-04,PIC16CR64-04 (Commercial, Industrial, Automotive)**  
**PIC16C62A-10,PIC16C64A-10 (Commercial, Industrial, Automotive)**  
**PIC16CR62-10,PIC16CR64-10 (Commercial, Industrial, Automotive)**  
**PIC16C62A-20,PIC16C64A-20 (Commercial, Industrial, Automotive)**  
**PIC16CR62-20,PIC16CR64-20 (Commercial, Industrial, Automotive)**

Standard Operating Conditions (unless otherwise stated)							
DC CHARACTERISTICS							
Operating temperature    -40°C    ≤ TA ≤ +125°C for automotive, -40°C    ≤ TA ≤ +85°C for industrial and 0°C        ≤ TA ≤ +70°C for commercial							
Param No.	Characteristic	Sym	Min	Typ†	Max	Units	Conditions
D001 D001A	Supply Voltage	VDD	4.0 4.5	- -	6.0 5.5	V V	XT, RC and LP osc configuration HS osc configuration
D002*	RAM Data Retention Voltage (Note 1)	VDR	-	1.5	-	V	Device in SLEEP mode
D003	VDD start voltage to ensure Power-on Reset	VPOR	-	VSS	-	V	See section on Power-on Reset for details
D004*	VDD rise rate to ensure Power-on Reset	SVDD	0.05	-	-	V/ms	See section on Power-on Reset for details
D005	Brown-out Reset Voltage	BVDD	3.7 3.7	4.0 4.0	4.3 4.4	V V	Boden configuration bit enabled Automotive Range Only
D010  D013	Supply Current (Note 2, 5)	IDD	- -	2.7 13.5	5 30	mA mA	XT, RC, osc configuration (PIC16C62A/R62/ 64A/R64-04) FOSC = 4 MHz, VDD = 5.5V (Note 4) HS osc configuration (PIC16C62A/R62/64A/ R64-20) FOSC = 20 MHz, VDD = 5.5V
D015*	Brown-out Reset Current (Note 7)	Δ IBOR	-	350	425	μA	BOR enabled, VDD = 5.0V

\* These parameters are characterized but not tested.

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

Note 1: This is the limit to which VDD can be lowered in SLEEP mode without losing RAM data.

2: The supply current is mainly a function of the operating voltage and frequency. Other factors such as I/O pin loading and switching rate, oscillator type, internal code execution pattern, and temperature also have an impact on the current consumption.

The test conditions for all IDD measurements in active operation mode are:

OSC1 = external square wave, from rail to rail; all I/O pins tristated, pulled to VDD

MCLR = VDD; WDT enabled/disabled as specified.

3: The power-down current in SLEEP mode does not depend on the oscillator type. Power-down current is measured with the part in SLEEP mode, with all I/O pins in hi-impedance state and tied to VDD and VSS.

4: For RC osc configuration, current through Rext is not included. The current through the resistor can be estimated by the formula  $I_r = V_{DD}/2R_{ext}$  (mA) with Rext in kOhm.

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

6: The Δ current is the additional current consumed when this peripheral is enabled. This current should be added to the base IDD or IPD measurement.

Applicable Devices	61	62	62A	R62	63	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

Standard Operating Conditions (unless otherwise stated)							
DC CHARACTERISTICS		Operating temperature					
		-40°C ≤ TA ≤ +125°C for automotive,					
		-40°C ≤ TA ≤ +85°C for industrial and					
		0°C ≤ TA ≤ +70°C for commercial					
Param No.	Characteristic	Sym	Min	Typ†	Max	Units	Conditions
D020	Power-down Current (Note 3, 5)	IPD	-	10.5	42	μA	VDD = 4.0V, WDT enabled, -40°C to +85°C
D021			-	1.5	21	μA	VDD = 4.0V, WDT disabled, -0°C to +70°C
D021A			-	1.5	24	μA	VDD = 4.0V, WDT disabled, -40°C to +85°C
D021B			-	2.5	24	μA	VDD = 4.0V, WDT disabled, -40°C to +125°C
D023*	Brown-out Reset Current (Note 6)	Δ IBOR	-	350	425	μA	BOR enabled, VDD = 5.0V

\* These parameters are characterized but not tested.

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

Note 1: This is the limit to which VDD can be lowered in SLEEP mode without losing RAM data.

2: The supply current is mainly a function of the operating voltage and frequency. Other factors such as I/O pin loading and switching rate, oscillator type, internal code execution pattern, and temperature also have an impact on the current consumption.

The test conditions for all IDD measurements in active operation mode are:

OSC1 = external square wave, from rail to rail; all I/O pins tristated, pulled to VDD

MCLR = VDD; WDT enabled/disabled as specified.

3: The power-down current in SLEEP mode does not depend on the oscillator type. Power-down current is measured with the part in SLEEP mode, with all I/O pins in hi-impedance state and tied to VDD and VSS.

4: For RC osc configuration, current through Rext is not included. The current through the resistor can be estimated by the formula  $I_r = V_{DD}/2R_{ext}$  (mA) with Rext in kOhm.

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

6: The Δ current is the additional current consumed when this peripheral is enabled. This current should be added to the base IDD or IPD measurement.

# PIC16C6X

Applicable Devices 61 62 62A R62 63 64 64A R64 65 65A

## 20.2 DC Characteristics: PIC16LC62A-04, PIC16LCR62-04 (Commercial, Industrial, Automotive<sup>(6)</sup>) PIC16LC64A-04, PIC16LCR64-04 (Commercial, Industrial, Automotive<sup>(6)</sup>)

Standard Operating Conditions (unless otherwise stated)							
DC CHARACTERISTICS							
Operating temperature -40°C ≤ TA ≤ +125°C for automotive, -40°C ≤ TA ≤ +85°C for industrial and 0°C ≤ TA ≤ +70°C for commercial							
Param No.	Characteristic	Sym	Min	Typ†	Max	Units	Conditions
D001	Supply Voltage	VDD	2.5	-	6.0	V	LP, XT, RC osc configuration (DC - 4 MHz)
D002*	RAM Data Retention Voltage (Note 1)	VDR	-	1.5	-	V	Device in SLEEP mode
D003	VDD start voltage to ensure Power-on Reset	VPOR	-	VSS	-	V	See section on Power-on Reset for details
D004*	VDD rise rate to ensure Power-on Reset	SVDD	0.05	-	-	V/ms	See section on Power-on Reset for details
D005	Brown-out Reset Voltage	BVDD	3.7	4.0	4.3	V	Boden bit in configuration word enabled
			3.7	4.0	4.4	V	Automotive Range Only
D010	Supply Current (Note 2, 5)	IDD	-	2.0	3.8	mA	XT, RC osc configuration FOSC = 4 MHz, VDD = 3.0V (Note 4)
D010A			-	22.5	48	μA	LP osc configuration FOSC = 32 kHz, VDD = 3.0V, WDT disabled
D015*	Brown-out Reset Current (Note 7)	ΔIBOR	-	350	425	μA	BOR enabled, VDD = 5.0V
D020	Power-down Current (Note 3, 5)	IPD	-	7.5	30	μA	VDD = 3.0V, WDT enabled, -40°C to +85°C
D021			-	0.9	5	μA	VDD = 3.0V, WDT disabled, 0°C to +70°C
D021A			-	0.9	5	μA	VDD = 3.0V, WDT disabled, -40°C to +85°C
D021B			-	0.9	10	μA	VDD = 3.0V, WDT disabled, -40°C to +125°C
D023*	Brown-out Reset Current (Note 7)	ΔIBOR	-	350	425	μA	BOR enabled, VDD = 5.0V

\* These parameters are characterized but not tested.

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

Note 1: This is the limit to which VDD can be lowered in SLEEP mode without losing RAM data.

2: The supply current is mainly a function of the operating voltage and frequency. Other factors such as I/O pin loading and switching rate, oscillator type, internal code execution pattern, and temperature also have an impact on the current consumption.

The test conditions for all IDD measurements in active operation mode are:

OSC1 = external square wave, from rail to rail; all I/O pins tristated, pulled to VDD

MCLR = VDD; WDT enabled/disabled as specified.

3: The power-down current in SLEEP mode does not depend on the oscillator type. Power-down current is measured with the part in SLEEP mode, with all I/O pins in hi-impedance state and tied to VDD and VSS.

4: For RC osc configuration, current through Rext is not included. The current through the resistor can be estimated by the formula  $I_r = V_{DD}/2R_{ext}$  (mA) with Rext in kOhm.

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

6: Automotive operating range is Advanced information for this device.

7: The Δ current is the additional current consumed when this peripheral is enabled. This current should be added to the base IDD or IPD measurement.



# PIC16C6X

<b>Applicable Devices</b>	61	62	62A	R62	63	64	64A	R64	65	65A
---------------------------	----	----	-----	-----	----	----	-----	-----	----	-----

**20.3 DC Characteristics:** PIC16C62A-04,PIC16C64A-04 (Commercial, Industrial, Automotive)  
PIC16CR62-04,PIC16CR64-04 (Commercial, Industrial, Automotive)  
PIC16C62A-10,PIC16C64A-10 (Commercial, Industrial, Automotive)  
PIC16CR62-10,PIC16CR64-10 (Commercial, Industrial, Automotive)  
PIC16C62A-20,PIC16C64A-20 (Commercial, Industrial, Automotive)  
PIC16CR62-20,PIC16CR64-20 (Commercial, Industrial, Automotive)  
PIC16LC62A-04,PIC16LC64A-04 (Commercial, Industrial, Automotive)  
PIC16LCR62-04,PIC16LCR64-04 (Commercial, Industrial, Automotive)

<div> <div>DC CHARACTERISTICS</div> <div> <div>Standard Operating Conditions (unless otherwise stated)</div> <div> Operating temperature    <math>-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}</math> for automotive,  <math>-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}</math> for industrial and  <math>0^{\circ}\text{C} \leq T_A \leq +70^{\circ}\text{C}</math> for commercial </div> </div> <div>Operating voltage VDD range as described in DC spec Section 20.1 and Section 20.2</div> </div>							
Param No.	Characteristic	Sym	Min	Typ †	Max	Units	Conditions
D030 D031 D032 D033	<b>Input Low Voltage</b> I/O ports with TTL buffer with Schmitt Trigger buffer MCLR, RA4/T0CKI, OSC1 (in RC mode) OSC1 (in XT, HS and LP)	VIL	VSS VSS VSS VSS	- - - -	0.5V 0.2VDD 0.2VDD 0.3VDD	V V V V	Note1
D040 D040A D041 D042 D042A D043	<b>Input High Voltage</b> I/O ports with TTL buffer with Schmitt Trigger buffer MCLR, RA4/T0CKI, RC7:RC4, RD7:RD4, RB0/INT, RE2:RE0 OSC1 (XT, HS and LP) OSC1 (in RC mode)	VIH	2.0 0.8VDD 0.8VDD 0.8VDD 0.7VDD 0.9VDD	- - - - - -	VDD VDD VDD VDD VDD VDD	V V V V V V	$4.5\text{V} \leq V_{DD} \leq 5.5\text{V}$ For VDD > 5.5V or VDD < 4.5V For entire VDD range  Note1
D070	PORTB weak pull-up current	IPURB	50	250	†400	μA	VDD = 5V, VPIN = VSS
D060 D061 D063	<b>Input Leakage Current</b> (Notes 2, 3) I/O ports  MCLR, RA4/T0CKI OSC1	IIL	- - -	- - -	±1 ±5 ±5	μA μA μA	VSS ≤ VPIN ≤ VDD, Pin at hi-impedance VSS ≤ VPIN ≤ VDD VSS ≤ VPIN ≤ VDD, XT, HS and LP osc configuration

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

Note 1: In RC oscillator configuration, the OSC1/CLKIN pin is a Schmitt Trigger input. It is not recommended that the PIC16C6X be driven with external clock in RC mode.

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

3: Negative current is defined as coming out of the pin.

# PIC16C6X

Applicable Devices 61 62 62A R62 63 64 64A R64 65 65A

<b>Standard Operating Conditions (unless otherwise stated)</b> Operating temperature    -40°C ≤ TA ≤ +125°C for automotive, -40°C ≤ TA ≤ +85°C for industrial and 0°C ≤ TA ≤ +70°C for commercial  Operating voltage VDD range as described in DC spec Section 20.1 and Section 20.2							
Param No.	Characteristic	Sym	Min	Typ †	Max	Units	Conditions
D080	<b>Output Low Voltage</b> I/O ports	VOL	-	-	0.6	V	IO <sub>L</sub> = 8.5 mA, VDD = 4.5V, -40°C to +85°C
D080A			-	-	0.6	V	IO <sub>L</sub> = 7.0 mA, VDD = 4.5V, -40°C to +125°C
D083	OSC2/CLKOUT (RC osc config)		-	-	0.6	V	IO <sub>L</sub> = 1.6 mA, VDD = 4.5V, -40°C to +85°C
D083A			-	-	0.6	V	IO <sub>L</sub> = 1.2 mA, VDD = 4.5V, -40°C to +125°C
D090	<b>Output High Voltage</b> I/O ports (Note 3)	VOH	VDD-0.7	-	-	V	IO <sub>H</sub> = -3.0 mA, VDD = 4.5V, -40°C to +85°C
D090A			VDD-0.7	-	-	V	IO <sub>H</sub> = -2.5 mA, VDD = 4.5V, -40°C to +125°C
D092	OSC2/CLKOUT (RC osc config)		VDD-0.7	-	-	V	IO <sub>H</sub> = -1.3 mA, VDD = 4.5V, -40°C to +85°C
D092A			VDD-0.7	-	-	V	IO <sub>H</sub> = -1.0 mA, VDD = 4.5V, -40°C to +125°C
D100	<b>Capacitive Loading Specs on Output Pins</b> OSC2 pin	COSC2			15	pF	In XT, HS and LP modes when external clock is used to drive OSC1.
D101	All I/O pins and OSC2 (in RC mode)	CIO			50	pF	
D102	SCL, SDA in I <sup>2</sup> C mode	Cb			400	pF	

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

- Note 1: In RC oscillator configuration, the OSC1/CLKIN pin is a Schmitt Trigger input. It is not recommended that the PIC16C6X be driven with external clock in RC mode.
- 2: The leakage current on the MCLR/VPP pin is strongly dependent on the applied voltage level. The specified levels represent normal operating conditions. Higher leakage current may be measured at different input voltages.
- 3: Negative current is defined as coming out of the pin.

Applicable Devices	61	62	62A	R62	63	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

20.4 Timing Parameter Symbolology

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

1. TppS2ppS
2. TppS
3. TCC:ST (I<sup>2</sup>C specifications only)
4. Ts (I<sup>2</sup>C specifications only)

<b>T</b>			
F	Frequency	T	Time

Lowercase letters (pp) and their meanings:

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

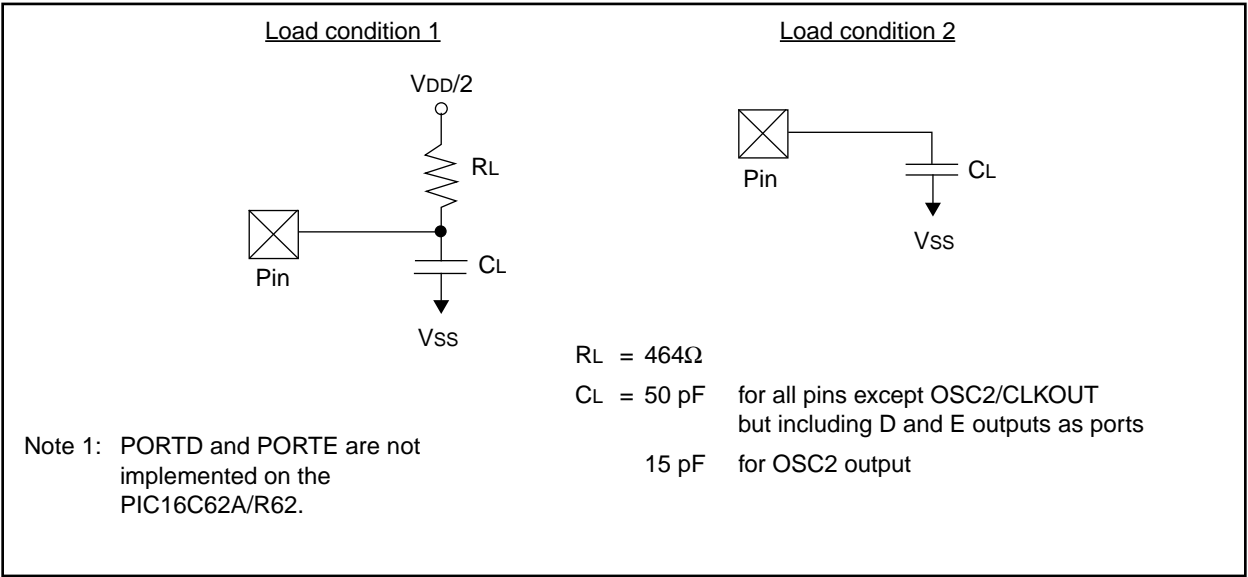
Uppercase letters and their meanings:

<b>S</b>			
F	Fall	P	Period
H	High	R	Rise
I	Invalid (Hi-impedance)	V	Valid
L	Low	Z	Hi-impedance
<b>I<sup>2</sup>C only</b>			
AA	output access	High	High
BUF	Bus free	Low	Low

TCC:ST (I<sup>2</sup>C specifications only)

<b>CC</b>			
HD	Hold	SU	Setup
<b>ST</b>			
DAT	DATA input hold	STO	STOP condition
STA	START condition		

FIGURE 20-1: LOAD CONDITIONS FOR DEVICE TIMING SPECIFICATIONS



# PIC16C6X

Applicable Devices 61 62 62A R62 63 64 64A R64 65 65A

## 20.5 Timing Diagrams and Specifications

FIGURE 20-2: EXTERNAL CLOCK TIMING

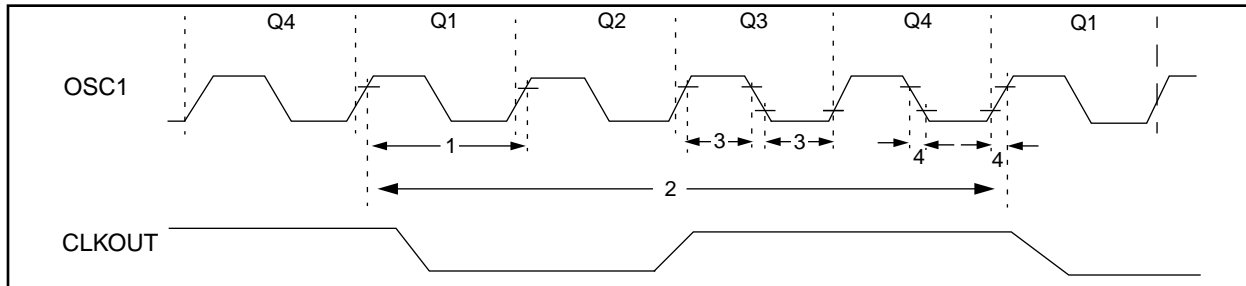


TABLE 20-2: EXTERNAL CLOCK TIMING REQUIREMENTS

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 (PIC16C64A/CR64-04, PIC16LC64A/LCR64-04)
			DC	—	20	MHz	HS osc mode (PIC16C64A/CR64-20)
			DC	—	200	kHz	LP osc mode
		Oscillator Frequency (Note 1)	DC	—	4	MHz	RC osc mode
			0.1	—	4	MHz	XT osc mode
			4	—	4	MHz	HS osc mode (PIC16C64A/CR64-04, PIC16LC64A/LCR64-04)
			4	—	10	MHz	HS osc mode (PIC16C64A/CR64-10)
			4	—	20	MHz	HS osc mode (PIC16C64A/CR64-20)
			5	—	200	kHz	LP osc mode

† 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	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

Parameter No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
1	Tosc	<b>External CLKIN Period (Note 1)</b>	250	—	—	ns	XT and RC osc mode
			250	—	—	ns	HS osc mode (PIC16C64A/CR64-04, PIC16LC64A/LCR64-04)
			100	—	—	ns	HS osc mode (PIC16C64A/CR64-10)
			50	—	—	ns	HS osc mode (PIC16C64A/CR64-20)
			5	—	—	μs	LP osc mode
		<b>Oscillator Period (Note 1)</b>	250	—	—	ns	RC osc mode
			250	—	10,000	ns	XT osc mode
			250	—	250	ns	HS osc mode (PIC16C64A/CR64-04, PIC16LC64A/LCR64-04)
			100	—	250	ns	HS osc mode (PIC16C64A/CR64-10)
			50	—	250	ns	HS osc mode (PIC16C64A/CR64-20)
			5	—	—	μs	LP osc mode
2	TCY	<b>Instruction Cycle Time (Note 1)</b>	200	—	DC	ns	TCY = 4/Fosc
3	TosL, TosH	<b>External Clock in (OSC1) High or Low Time</b>	100	—	—	ns	XT oscillator
			2	—	—	μs	LP oscillator
			20	—	—	ns	HS oscillator
4	TosR, TosF	<b>External Clock in (OSC1) Rise or Fall Time</b>	—	—	25	ns	XT oscillator
			—	—	50	ns	LP oscillator
			—	—	15	ns	HS oscillator

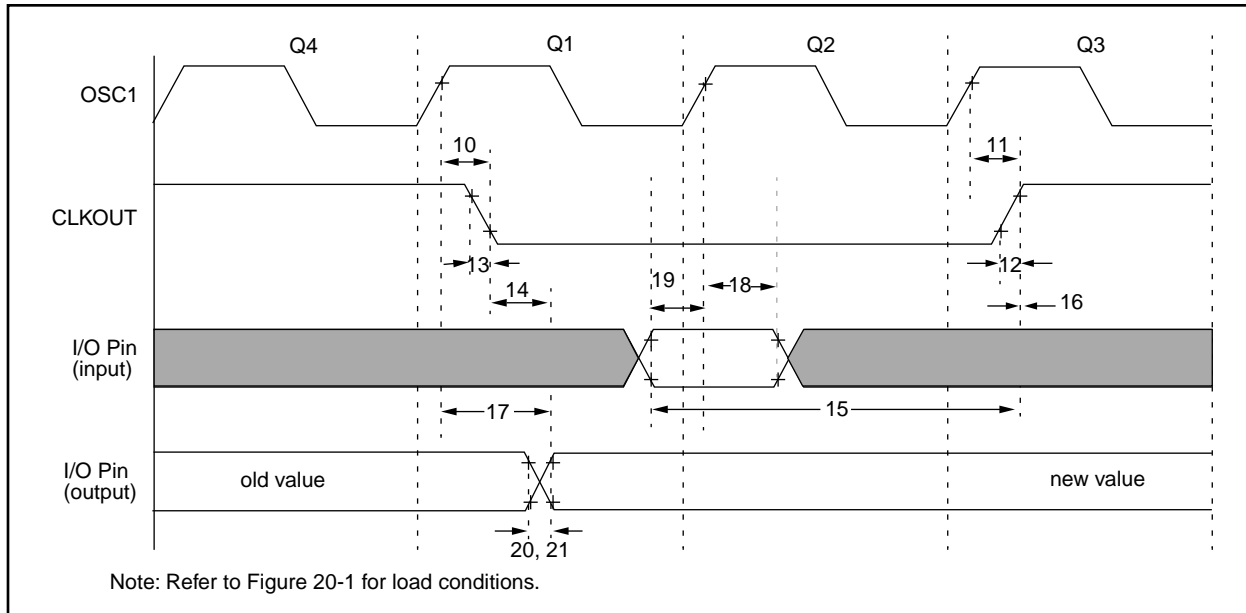
† Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

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

# PIC16C6X

Applicable Devices 61 62 62A R62 63 64 64A R64 65 65A

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



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

Parameters	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
10*	TosH2ckL	OSC1↑ to CLKOUT↓	—	75	200	ns	Note 1
11*	TosH2ckH	OSC1↑ to CLKOUT↑	—	75	200	ns	Note 1
12*	TckR	CLKOUT rise time	—	3	100	ns	Note 1
13*	TckF	CLKOUT fall time	—	3	100	ns	Note 1
14*	TckL2ioV	CLKOUT ↓ to Port out valid	—	—	0.5Tcy + 20	ns	Note 1
15*	TioV2ckH	Port in valid before CLKOUT ↑	Tosc + 200	—	—	ns	Note 1
16*	TckH2iol	Port in hold after CLKOUT ↑	0	—	—	ns	Note 1
17*	TosH2ioV	OSC1↑ (Q1 cycle) to Port out valid	—	50	150	ns	
18*	TosH2iol	OSC1↑ (Q2 cycle) to Port input invalid (I/O in hold time)	100	—	—	ns	
19*	TioV2osH	Port input valid to OSC1↑ (I/O in setup time)	0	—	—	ns	
20*	TioR	Port output rise time	—	PIC16C62A/ R62/64A/R64	10	40	ns
				PIC16LC62A/ R62/64A/R64	—	80	ns
21*	TioF	Port output fall time	—	PIC16C62A/ R62/64A/R64	10	40	ns
				PIC16LC62A/ R62/64A/R64	—	80	ns
22††*	Tinp	RB0/INT pin high or low time	Tcy	—	—	ns	
23††*	Trbp	RB7:RB4 change int high or low time	Tcy	—	—	ns	

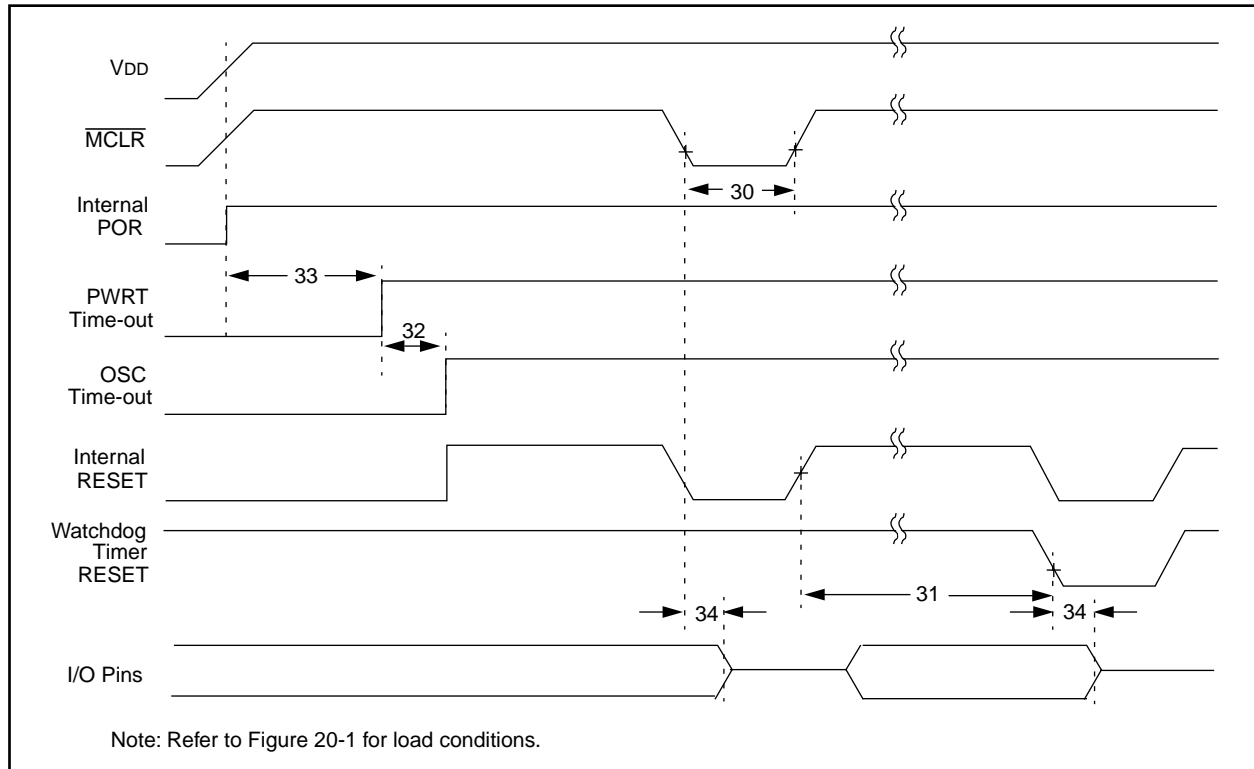
\* These parameters are characterized but not tested.

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

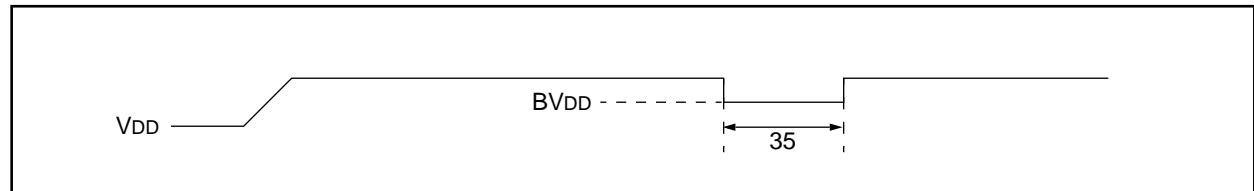
†† These parameters are asynchronous events not related to any internal clock edge.

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

**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	—	1024Tosc	—	—	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)

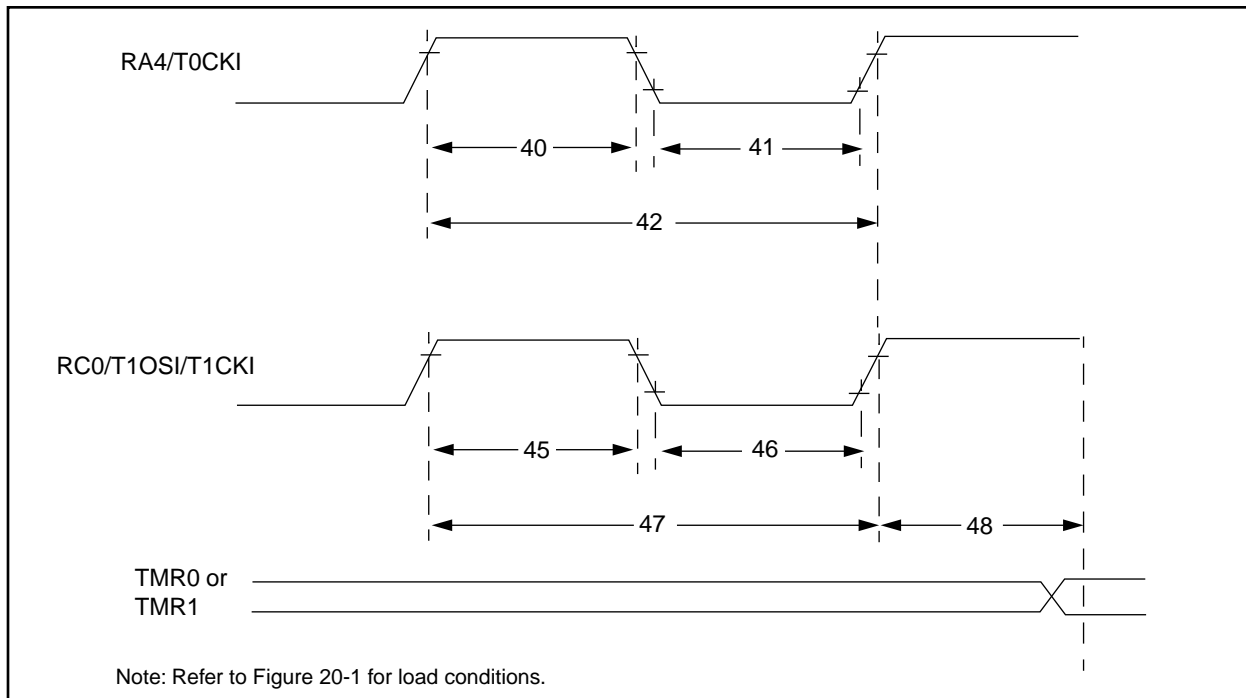
\* These parameters are characterized but not tested.

† 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 64 64A R64 65 65A

**FIGURE 20-6: TIMER0 AND TIMER1 CLOCK TIMINGS**



**TABLE 20-5: TIMER0 AND TIMER1 CLOCK REQUIREMENTS**

Parameter No.	Sym	Characteristic		Min	Typ†	Max	Units	Conditions
40*	Tt0H	T0CKI High Pulse Width	No Prescaler	$0.5T_{CY} + 20$	—	—	ns	
			With Prescaler	10	—	—	ns	
41*	Tt0L	T0CKI Low Pulse Width	No Prescaler	$0.5T_{CY} + 20$	—	—	ns	
			With Prescaler	10	—	—	ns	
42*	Tt0P	T0CKI Period		$\frac{T_{CY} + 40}{N}$	—	—	ns	N = prescale value (1, 2, 4, ..., 256)
45*	Tt1H	T1CKI High Time	Synchronous, No Prescaler	$0.5T_{CY} + 20$	—	—	ns	
		Synchronous, With Prescaler	PIC16C62A/R62/64A/R64	10	—	—	ns	
			PIC16LC62A/R62/64A/R64	20	—	—	ns	
		Asynchronous		$2T_{CY}$	—	—	ns	
46*	Tt1L	T1CKI Low Time	Synchronous, No Prescaler	$0.5T_{CY} + 20$	—	—	ns	
		Synchronous, With Prescaler	PIC16C62A/R62/64A/R64	10	—	—	ns	
			PIC16LC62A/R62/64A/R64	20	—	—	ns	
		Asynchronous		$2T_{CY}$	—	—	ns	
47*	Tt1P	T1CKI input period	Synchronous	$\frac{T_{CY} + 40}{N}$	—	—	ns	N = prescale value (1, 2, 4, 8)
		Asynchronous		$4T_{CY}$	—	—	ns	
	Ft1	Timer1 oscillator input frequency range (oscillator enabled by setting bit T1OSCEN)		DC	—	200	kHz	
48	TCKEztmr1	Delay from external clock edge to timer increment		$2T_{osc}$	—	$7T_{osc}$	—	

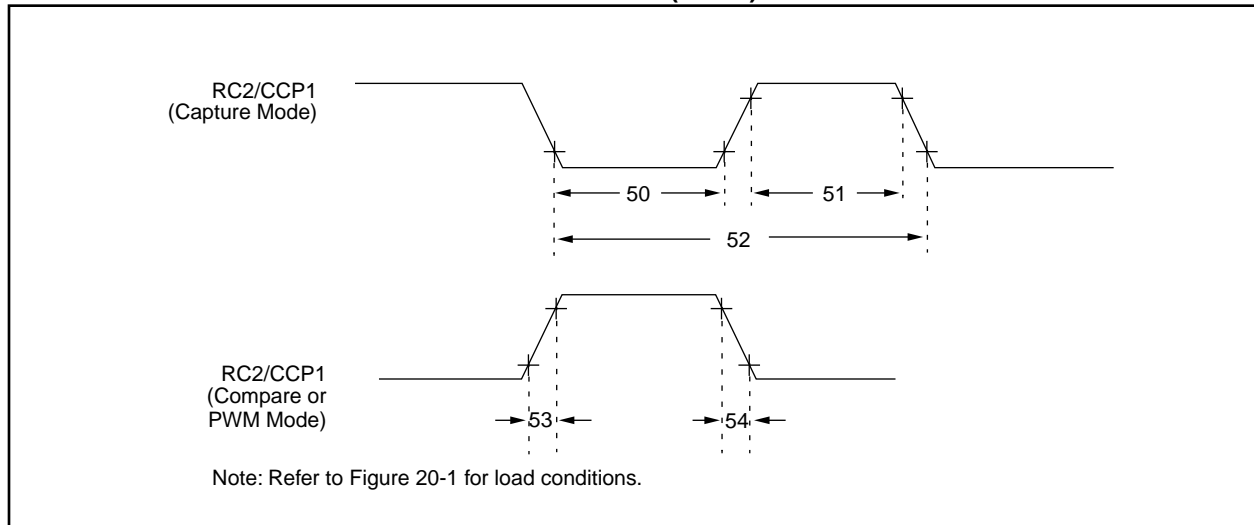
\* These parameters are characterized but not tested.

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



Applicable Devices	61	62	62A	R62	63	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

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



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

Parameter No.	Sym	Characteristic			Min	Typ†	Max	Units	Conditions
50*	TccL	CCP1 input low time	No Prescaler		0.5Tcy + 20	—	—	ns	
			With Prescaler	PIC16C62A/R62/64A/R64	10	—	—	ns	
				PIC16LC62A/R62/64A/R64	20	—	—	ns	
51*	TccH	CCP1 input high time	No Prescaler		0.5Tcy + 20	—	—	ns	
			With Prescaler	PIC16C62A/R62/64A/R64	10	—	—	ns	
				PIC16LC62A/R62/64A/R64	20	—	—	ns	
52*	TccP	CCP1 input period			$\frac{3T_{CY} + 40}{N}$	—	—	ns	N = prescale value (1,4 or 16)
53*	TccR	CCP1 output rise time			—	10	25	ns	
54*	TccF	CCP1 output fall time		PIC16C62A/R62/64A/R64	—	10	25	ns	
				PIC16LC62A/R62/64A/R64	—	—	45	ns	

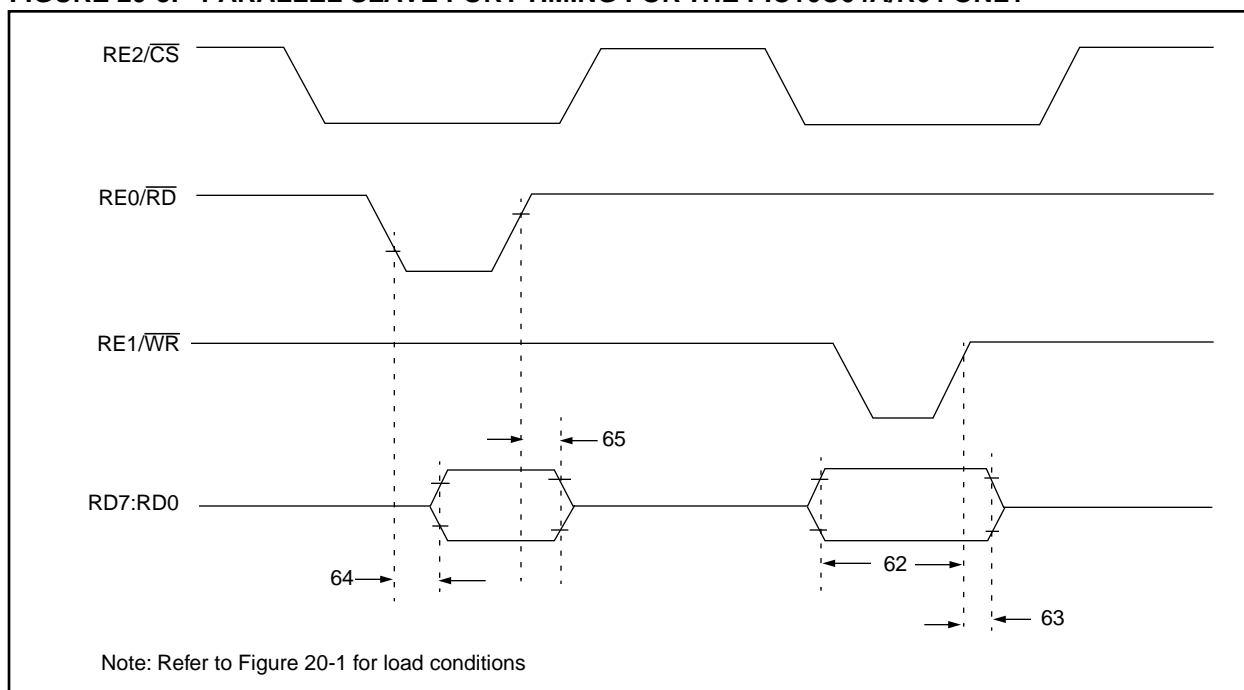
\* These parameters are characterized but not tested.

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

# PIC16C6X

Applicable Devices 61 62 62A R62 63 64 64A R64 65 65A

**FIGURE 20-8: PARALLEL SLAVE PORT TIMING FOR THE PIC16C64A/R64 ONLY**



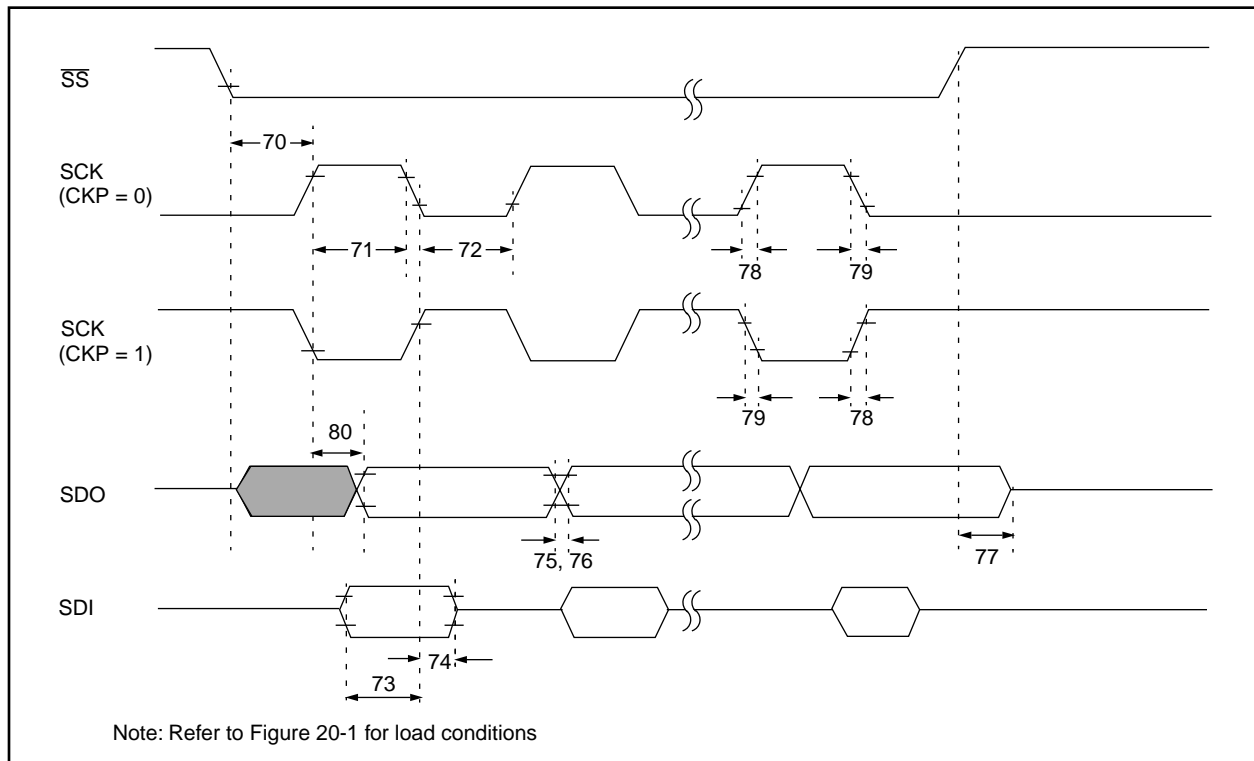
**TABLE 20-7: PARALLEL SLAVE PORT REQUIREMENTS FOR THE PIC16C64A/R64 ONLY**

Parameter No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
62	TdtV2wrH	Data in valid before $\overline{WR}\uparrow$ or $\overline{CS}\uparrow$ (setup time)	20	—	—	ns	Automotive Range Only
			25	—	—	ns	
63*	TwrH2dtl	$\overline{WR}\uparrow$ or $\overline{CS}\uparrow$ to data-in invalid (hold time)	PIC16C64A/R64	20	—	—	ns
			PIC16LC64A/R64	35	—	—	ns
64	TrdL2dtV	$\overline{RD}\downarrow$ and $\overline{CS}\downarrow$ to data-out valid	—	—	80	ns	Automotive Range Only
			—	—	90	ns	
65*	TrdH2dtl	$\overline{RD}\uparrow$ or $\overline{CS}\uparrow$ to data-out invalid	10	—	30	ns	

\* Characterized but not tested.

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

**FIGURE 20-9: SPI MODE TIMING**



**TABLE 20-8: SPI MODE REQUIREMENTS**

Parameter No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
70*	TssL2scH, TssL2scL	$\overline{SS}\downarrow$ to SCK $\downarrow$ or SCK $\uparrow$ input	T <sub>CY</sub>	—	—	ns	
71*	TscH	SCK input high time (slave mode)	T <sub>CY</sub> + 20	—	—	ns	
72*	TscL	SCK input low time (slave mode)	T <sub>CY</sub> + 20	—	—	ns	
73*	TdiV2scH, TdiV2scL	Setup time of SDI data input to SCK edge	100	—	—	ns	
74*	Tsch2diL, TscL2diL	Hold time of SDI data input to SCK edge	100	—	—	ns	
75*	TdoR	SDO data output rise time	—	10	25	ns	
76*	TdoF	SDO data output fall time	—	10	25	ns	
77*	TssH2doZ	$\overline{SS}\downarrow$ to SDO output hi-impedance	10	—	50	ns	
78*	TscR	SCK output rise time (master mode)	—	10	25	ns	
79*	TscF	SCK output fall time (master mode)	—	10	25	ns	
80*	Tsch2doV, TscL2doV	SDO data output valid after SCK edge	—	—	50	ns	

\* 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	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

FIGURE 20-10: I<sup>2</sup>C BUS START/STOP BITS TIMING

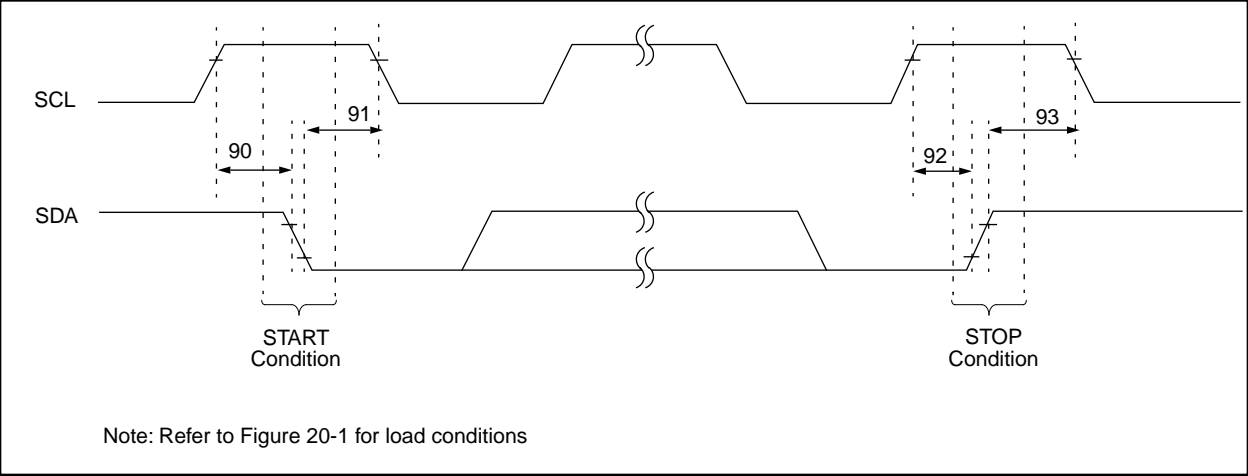


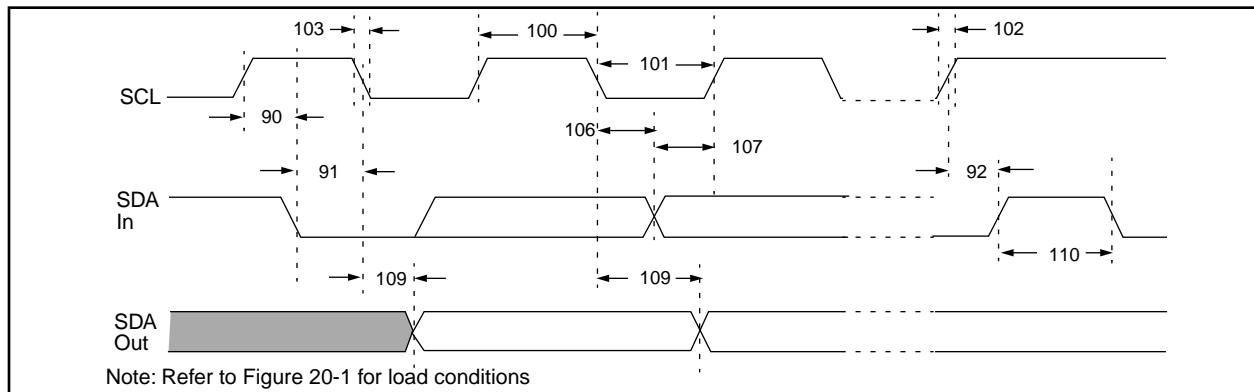
TABLE 20-9: I<sup>2</sup>C BUS START/STOP BITS REQUIREMENTS

Parameter No.	Sym	Characteristic		Min	Typ	Max	Units	Conditions
90*	TSU:STA	START condition	100 kHz mode	4700	—	—	ns	Only relevant for repeated START condition
		Setup time	400 kHz mode	600	—	—		
91*	THD:STA	START condition	100 kHz mode	4000	—	—	ns	After this period the first clock pulse is generated
		Hold time	400 kHz mode	600	—	—		
92*	TSU:STO	STOP condition	100 kHz mode	4700	—	—	ns	
		Setup time	400 kHz mode	600	—	—		
93*	THD:STO	STOP condition	100 kHz mode	4000	—	—	ns	
		Hold time	400 kHz mode	600	—	—		

\*Characterized but not tested.

Applicable Devices	61	62	62A	R62	63	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

**FIGURE 20-11: I<sup>2</sup>C BUS DATA TIMING**



**TABLE 20-10: I<sup>2</sup>C BUS DATA REQUIREMENTS**

Parameter No.	Sym	Characteristic		Min	Max	Units	Conditions
100*	THIGH	Clock high time	100 kHz mode	4.0	—	μs	PIC16C64A/R64 must operate at a minimum of 1.5 MHz
			400 kHz mode	0.6	—	μs	PIC16C64A/R64 must operate at a minimum of 10 MHz
			SSP Module	1.5TCY	—		
101*	TLOW	Clock low time	100 kHz mode	4.7	—	μs	PIC16C64A/R64 must operate at a minimum of 1.5 MHz
			400 kHz mode	1.3	—	μs	PIC16C64A/R64 must operate at a minimum of 10 MHz
			SSP Module	1.5TCY	—		
102*	Tr	SDA and SCL rise time	100 kHz mode	—	1000	ns	
			400 kHz mode	20 + 0.1Cb	300	ns	Cb is specified to be from 10-400 pF
103*	Tf	SDA and SCL fall time	100 kHz mode	—	300	ns	
			400 kHz mode	20 + 0.1Cb	300	ns	Cb is specified to be from 10-400 pF
90*	TSU:STA	START condition setup time	100 kHz mode	4.7	—	μs	Only relevant for repeated START condition
			400 kHz mode	0.6	—	μs	
91*	THD:STA	START condition hold time	100 kHz mode	4.0	—	μs	After this period the first clock pulse is generated
			400 kHz mode	0.6	—	μs	
106*	THD:DAT	Data input hold time	100 kHz mode	0	—	ns	
			400 kHz mode	0	0.9	μs	
107*	TSU:DAT	Data input setup time	100 kHz mode	250	—	ns	Note 2
			400 kHz mode	100	—	ns	
92*	TSU:STO	STOP condition setup time	100 kHz mode	4.7	—	μs	
			400 kHz mode	0.6	—	μs	
109*	TAA	Output valid from clock	100 kHz mode	—	3500	ns	Note 1
			400 kHz mode	—	—	ns	
110*	TBUF	Bus free time	100 kHz mode	4.7	—	μs	Time the bus must be free before a new transmission can start
			400 kHz mode	1.3	—	μs	
	Cb	Bus capacitive loading		—	400	pF	

\* Characterized but not tested.

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

2: A fast-mode I<sup>2</sup>C-bus device can be used in a standard-mode I<sup>2</sup>C-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 I<sup>2</sup>C bus specification) before the SCL line is released.

# PIC16C6X

---

NOTES:

Applicable Devices	61	62	62A	R62	63	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

21.0 DC AND AC CHARACTERISTICS GRAPHS AND TABLES FOR PIC16C62A/  
R62/64A/R64

NOT AVAILABLE AT THIS TIME

# PIC16C6X

---

NOTES:



Applicable Devices	61	62	62A	R62	63	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

## 22.0 ELECTRICAL CHARACTERISTICS FOR PIC16C65

### Absolute Maximum Ratings †

Ambient temperature under bias.....	-55 to +125°C
Storage temperature .....	-65°C to +150°C
Voltage on any pin with respect to VSS (except VDD and $\overline{\text{MCLR}}$ ) .....	-0.3V to (VDD + 0.3V)
Voltage on VDD with respect to VSS .....	0 to +7.5V
Voltage on $\overline{\text{MCLR}}$ with respect to VSS (Note 2) .....	0 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 <sub>IK</sub> (V <sub>I</sub> < 0 or V <sub>I</sub> > VDD) .....	±20 mA
Output clamp current, I <sub>OK</sub> (V <sub>O</sub> < 0 or V <sub>O</sub> > 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

**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  $\overline{\text{MCLR}}$  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  $\overline{\text{MCLR}}$  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 22-1: CROSS REFERENCE OF DEVICE SPECS FOR OSCILLATOR CONFIGURATIONS AND FREQUENCIES OF OPERATION (COMMERCIAL DEVICES)**

OSC	PIC16C65-04	PIC16C65-10	PIC16C65-20	PIC16LC65-04	JW Devices
RC	VDD: 4.0V to 6.0V IDD: 5 mA max. at 5.5V IPD: 21 µA max. at 4V Freq: 4 MHz max.	VDD: 4.5V to 5.5V IDD: 2.7 mA typ. at 5.5V IPD: 1.5 µA typ. at 4V Freq: 4 MHz max.	VDD: 4.5V to 5.5V IDD: 2.7 mA typ. at 5.5V IPD: 1.5 µA typ. at 4V Freq: 4 MHz max.	VDD: 3.0V to 6.0V IDD: 2.0 mA typ. at 3V IPD: 0.9 µA typ. at 3V Freq: 4 MHz max.	VDD: 4.0V to 6.0V IDD: 5 mA max. at 5.5V IPD: 21 µA max. at 4V Freq: 4 MHz max.
XT	VDD: 4.0V to 6.0V IDD: 5 mA max. at 5.5V IPD: 21 µA max. at 4V Freq: 4 MHz max.	VDD: 4.5V to 5.5V IDD: 2.7 mA typ. at 5.5V IPD: 1.5 µA typ. at 4V Freq: 4 MHz max.	VDD: 4.5V to 5.5V IDD: 2.7 mA typ. at 5.5V IPD: 1.5 µA typ. at 4V Freq: 4 MHz max.	VDD: 3.0V to 6.0V IDD: 2.0 mA typ. at 3V IPD: 0.9 µA typ. at 3V Freq: 4 MHz max.	VDD: 4.0V to 6.0V IDD: 5 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: 20 mA max. at 5.5V IPD: 1.0 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.	Do not 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.	Do not use in LP mode	Do not 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	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

## 22.1 DC CHARACTERISTICS: PIC16C65-04 (Commercial, Industrial, Automotive<sup>(6)</sup>) PIC16C65-10 (Commercial, Industrial, Automotive<sup>(6)</sup>) PIC16C65-20 (Commercial, Industrial, Automotive<sup>(6)</sup>)

Standard Operating Conditions (unless otherwise stated)							
DC CHARACTERISTICS							
Operating temperature    -40°C    ≤ TA ≤ +125°C for automotive, -40°C    ≤ TA ≤ +85°C for industrial and 0°C        ≤ TA ≤ +70°C for commercial							
Param No.	Characteristic	Sym	Min	Typ†	Max	Units	Conditions
D001 D001A	Supply Voltage	VDD	4.0 4.5	- -	6.0 5.5	V V	XT, RC and LP osc configuration HS osc configuration
D002*	RAM Data Retention Voltage (Note 1)	VDR	-	1.5	-	V	Device in SLEEP mode
D003	VDD start voltage to ensure Power-on Reset	VPOR	-	VSS	-	V	See section on Power-on Reset for details
D004*	VDD rise rate to ensure Power-on Reset	SVDD	0.05	-	-	V/ms	See section on Power-on Reset for details
D010  D013	Supply Current (Note 2, 5)	IDD	- -	2.7 13.5	5 30	mA mA	XT, RC osc configuration (PIC16C65-04) FOSC = 4 MHz, VDD = 5.5V (Note 4)  HS osc configuration (PIC16C65-20) FOSC = 20 MHz, VDD = 5.5V
D020 D021 D021A D021B	Power-down Current (Note 3, 5)	IPD	- - - -	10.5 1.5 1.5 1.5	800 800 800 800	μA μA μA μA	VDD = 4.0V, WDT enabled, -40°C to +85°C VDD = 4.0V, WDT disabled, -0°C to +70°C VDD = 4.0V, WDT disabled, -40°C to +85°C VDD = 4.0V, WDT disabled, -40°C to +125°C

\* These parameters are characterized but not tested.

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

Note 1: This is the limit to which VDD can be lowered in SLEEP mode without losing RAM data.

2: The supply current is mainly a function of the operating voltage and frequency. Other factors such as I/O pin loading and switching rate, oscillator type, internal code execution pattern, and temperature also have an impact on the current consumption.

The test conditions for all IDD measurements in active operation mode are:

OSC1 = external square wave, from rail to rail; all I/O pins tristated, pulled to VDD,

MCLR = VDD; WDT enabled/disabled as specified.

3: The power down current in SLEEP mode does not depend on the oscillator type. Power-down current is measured with the part in SLEEP mode, with all I/O pins in hi-impedance state and tied to VDD and VSS.

4: For RC osc configuration, current through Rext is not included. The current through the resistor can be estimated by the formula  $I_r = VDD/2R_{ext}$  (mA) with Rext in kOhm.

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

6: Automotive operating range is Advanced information for this device.

# PIC16C6X

<b>Applicable Devices</b>	61	62	62A	R62	63	64	64A	R64	65	65A
---------------------------	----	----	-----	-----	----	----	-----	-----	----	-----

## 22.2 DC CHARACTERISTICS: PIC16LC65-04 (Commercial, Industrial, Automotive<sup>(6)</sup>)

Standard Operating Conditions (unless otherwise stated)							
DC CHARACTERISTICS		Operating temperature -40°C ≤ TA ≤ +125°C for automotive, -40°C ≤ TA ≤ +85°C for industrial and 0°C ≤ TA ≤ +70°C for commercial					
Param No.	Characteristic	Sym	Min	Typ†	Max	Units	Conditions
D001	Supply Voltage	VDD	3.0	-	6.0	V	LP, XT, RC osc configuration (DC - 4 MHz)
D002*	RAM Data Retention Voltage (Note 1)	VDR	-	1.5	-	V	Device in SLEEP mode
D003	VDD start voltage to ensure Power-on Reset	VPOR	-	VSS	-	V	See section on Power-on Reset for details
D004*	VDD rise rate to ensure Power-on Reset	SVDD	0.05	-	-	V/ms	See section on Power-on Reset for details
D010	Supply Current (Note 2, 5)	IDD	-	2.0	3.8	mA	XT, RC osc configuration FOSC = 4 MHz, VDD = 3.0V (Note 4)
D010A			-	22.5	105	μA	LP osc configuration FOSC = 32 kHz, VDD = 4.0V, WDT disabled
D020	Power-down Current (Note 3, 5)	IPD	-	7.5	30	μA	VDD = 3.0V, WDT enabled, -40°C to +85°C
D021			-	0.9	13.5	μA	VDD = 3.0V, WDT disabled, 0°C to +70°C
D021A			-	0.9	18	μA	VDD = 3.0V, WDT disabled, -40°C to +85°C
D021B			-	0.9	24	μA	VDD = 3.0V, WDT disabled, -40°C to +125°C

\* These parameters are characterized but not tested.

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

Note 1: This is the limit to which VDD can be lowered in SLEEP mode without losing RAM data.

2: The supply current is mainly a function of the operating voltage and frequency. Other factors such as I/O pin loading and switching rate, oscillator type, internal code execution pattern, and temperature also have an impact on the current consumption.

The test conditions for all IDD measurements in active operation mode are:

OSC1 = external square wave, from rail to rail; all I/O pins tristated, pulled to V<sub>DD</sub>,

$\overline{\text{MCLR}} = \text{VDD}$ ; WDT enabled/disabled as specified.

3: The power down current in SLEEP mode does not depend on the oscillator type. Power-down current is measured with the part in SLEEP mode, with all I/O pins in hi-impedance state and tied to V<sub>DD</sub> and V<sub>SS</sub>.

4: For RC osc configuration, current through Rext is not included. The current through the resistor can be estimated by the formula  $I_r = V_{DD}/2R_{ext}$  (mA) with Rext in kOhm.

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

6: Automotive operating range is Advanced information for this device.

# PIC16C6X

Applicable Devices	61	62	62A	R62	63	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

**22.3 DC CHARACTERISTICS:**      **PIC16C65-04 (Commercial, Industrial, Automotive<sup>(4)</sup>)**  
**PIC16C65-10 (Commercial, Industrial, Automotive<sup>(4)</sup>)**  
**PIC16C65-20 (Commercial, Industrial, Automotive<sup>(4)</sup>)**  
**PIC16LC65-04 (Commercial, Industrial, Automotive<sup>(4)</sup>)**

<b>DC CHARACTERISTICS</b> <b>Standard Operating Conditions (unless otherwise stated)</b> Operating temperature    -40°C    ≤ TA ≤ +125°C for automotive, -40°C    ≤ TA ≤ +85°C for industrial and 0°C        ≤ TA ≤ +70°C for commercial Operating voltage VDD range as described in DC spec Section 22.1 and Section 22.2							
Param No.	Characteristic	Sym	Min	Typ †	Max	Units	Conditions
D030	<b>Input Low Voltage</b> I/O ports	VIL					
D031	with TTL buffer		VSS	-	0.5V	V	
D032	with Schmitt Trigger buffer		VSS	-	0.2VDD	V	
D032	MCLR, RA4/T0CKI, OSC1 (in RC mode)		VSS	-	0.2VDD	V	
D033	OSC1 (in XT, HS and LP)		VSS	-	0.3VDD	V	Note1
D040	<b>Input High Voltage</b> I/O ports	VIH		-			
D040A	with TTL buffer		2.0	-	VDD	V	4.5V ≤ VDD ≤ 5.5V
D041	with Schmitt Trigger buffer		0.8VDD	-	VDD		For VDD > 5.5V or VDD < 4.5V
D042	MCLR, RA4/T0CKI, RC7:RC4, RD7:RD4, RB0/INT		0.8VDD	-	VDD	V	For entire VDD range
D042A	RE2:RE0, OSC1 (XT, HS and LP)		0.7 VDD	-	VDD	V	Note1
D043	OSC1 (in RC mode)		0.9VDD	-	VDD	V	
D070	PORTB weak pull-up current	IPURB	50	250	†400	µA	VDD = 5V, VPIN = VSS
D060	<b>Input Leakage Current</b> (Notes 2, 3) I/O ports	IIL	-	-	±1	µA	VSS ≤ VPIN ≤ VDD, Pin at hi-impedance
D061	MCLR, RA4/T0CKI		-	-	±5	µA	VSS ≤ VPIN ≤ VDD
D063	OSC1		-	-	±5	µA	VSS ≤ VPIN ≤ VDD, XT, HS, and LP osc configuration
D080	<b>Output Low Voltage</b> I/O ports	VOL	-	-	0.6	V	IOL = 8.5 mA, VDD = 4.5V, -40°C to +85°C
D080A			-	-	0.6	V	IOL = 7.0 mA, VDD = 4.5V, -40°C to +125°C
D083	OSC2/CLKOUT (RC osc config)		-	-	0.6	V	IOL = 1.6 mA, VDD = 4.5V, -40°C to +85°C
D083A			-	-	0.6	V	IOL = 1.2 mA, VDD = 4.5V, -40°C to +125°C

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

- Note 1: In RC oscillator configuration, the OSC1/CLKIN pin is a Schmitt Trigger input. It is not recommended that the PIC16C6X be driven with external clock in RC mode.
- 2: The leakage current on the MCLR/VPP pin is strongly dependent on the applied voltage level. The specified levels represent normal operating conditions. Higher leakage current may be measured at different input voltages.
- 3: Negative current is defined as coming out of the pin.
- 4: Automotive operating range is Advanced information for this device.

Applicable Devices	61	62	62A	R62	63	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

<b>DC CHARACTERISTICS</b> <div> <b>Standard Operating Conditions (unless otherwise stated)</b>            Operating temperature    <math>-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}</math> for automotive,               <math>-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}</math> for industrial and               <math>0^{\circ}\text{C} \leq T_A \leq +70^{\circ}\text{C}</math> for commercial             Operating voltage <math>V_{DD}</math> range as described in DC spec Section 22.1 and            Section 22.2         </div>							
Param No.	Characteristic	Sym	Min	Typ †	Max	Units	Conditions
D090	<b>Output High Voltage</b> I/O ports (Note 3)	VOH	VDD-0.7	-	-	V	IOH = -3.0 mA, VDD = 4.5V, -40°C to +85°C
D090A			VDD-0.7	-	-	V	IOH = -2.5 mA, VDD = 4.5V, -40°C to +125°C
D092	OSC2/CLKOUT (RC osc config)		VDD-0.7	-	-	V	IOH = -1.3 mA, VDD = 4.5V, -40°C to +85°C
D092A			VDD-0.7	-	-	V	IOH = -1.0 mA, VDD = 4.5V, -40°C to +125°C
D100	<b>Capacitive Loading Specs on Output Pins</b> OSC2 pin	COSC2	-	-	15	pF	In XT, HS and LP modes when external clock is used to drive OSC1.
D101	All I/O pins and OSC2 (in RC mode)	Cio	-	-	50	pF	
D102	SCL, SDA in I <sup>2</sup> C mode	Cb	-	-	400	pF	

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

- Note 1: In RC oscillator configuration, the OSC1/CLKIN pin is a Schmitt Trigger input. It is not recommended that the PIC16C6X be driven with external clock in RC mode.
- 2: The leakage current on the  $\overline{\text{MCLR}}/\text{VPP}$  pin is strongly dependent on the applied voltage level. The specified levels represent normal operating conditions. Higher leakage current may be measured at different input voltages.
- 3: Negative current is defined as coming out of the pin.
- 4: Automotive operating range is Advanced information for this device.

# PIC16C6X

Applicable Devices	61	62	62A	R62	63	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

## 22.4 Timing Parameter Symbolology

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

1. TppS2ppS
2. TppS
3. TCC:ST (I<sup>2</sup>C specifications only)
4. Ts (I<sup>2</sup>C specifications only)

<b>T</b>			
F	Frequency	T	Time

Lowercase letters (pp) and their meanings:

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

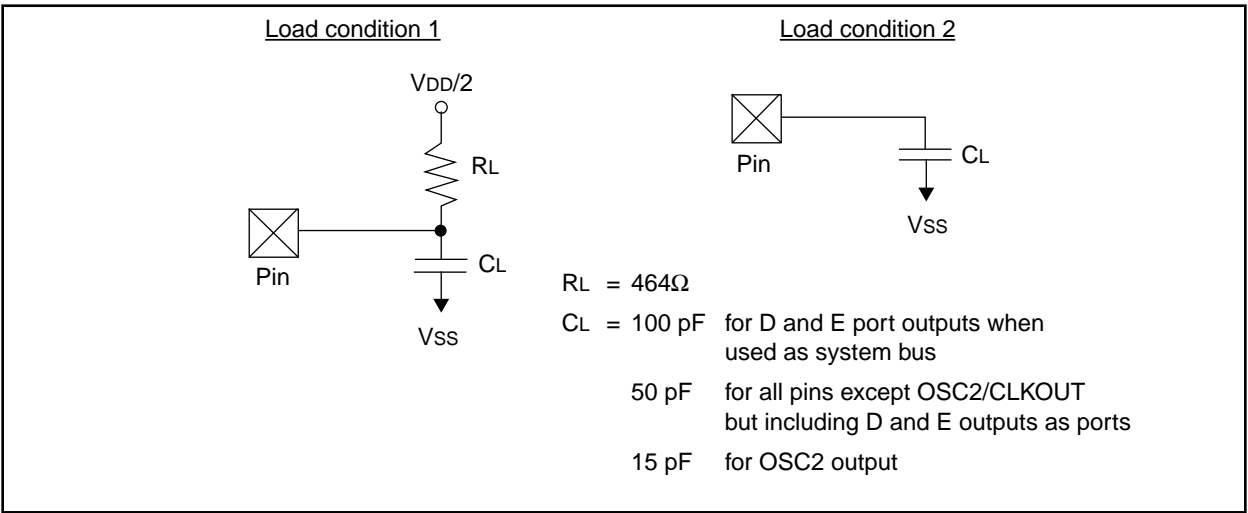
Uppercase letters and their meanings:

<b>S</b>			
F	Fall	P	Period
H	High	R	Rise
I	Invalid (Hi-impedance)	V	Valid
L	Low	Z	Hi-impedance
<b>I<sup>2</sup>C only</b>			
AA	output access	High	High
BUF	Bus free	Low	Low

TCC:ST (I<sup>2</sup>C specifications only)

<b>CC</b>			
HD	Hold	SU	Setup
<b>ST</b>			
DAT	DATA input hold	STO	STOP condition
STA	START condition		

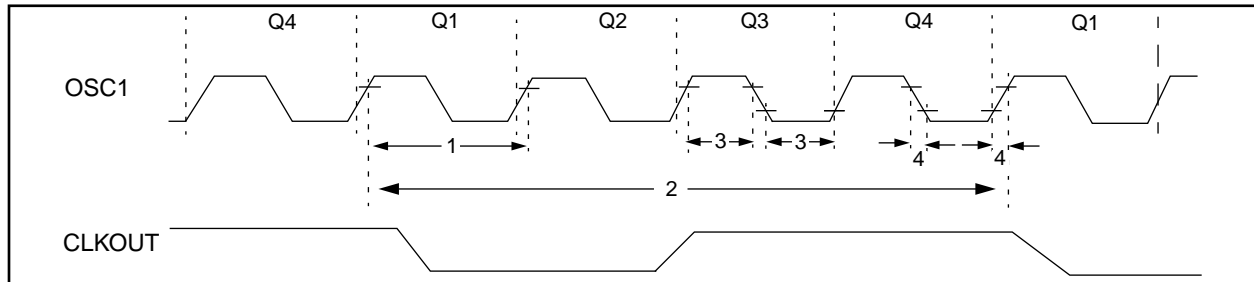
FIGURE 22-1: LOAD CONDITIONS FOR DEVICE TIMING SPECIFICATIONS



Applicable Devices	61	62	62A	R62	63	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

## 22.5 Timing Diagrams and Specifications

**FIGURE 22-2: EXTERNAL CLOCK TIMING**



**TABLE 22-2: EXTERNAL CLOCK TIMING REQUIREMENTS**

Parameter No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
	Fosc	<b>External CLKIN Frequency (Note 1)</b>	DC	—	4	MHz	XT and RC osc mode
			DC	—	4	MHz	HS osc mode (PIC16C65-04, PIC16LC65-04)
			DC	—	20	MHz	HS osc mode (PIC16C65-20)
			DC	—	200	kHz	LP osc mode
		<b>Oscillator Frequency (Note 1)</b>	DC	—	4	MHz	RC osc mode
			0.1	—	4	MHz	XT osc mode
			4	—	4	MHz	HS osc mode (PIC16C65-04, PIC16LC65-04)
			4	—	10	MHz	HS osc mode (PIC16C65-10)
			4	—	20	MHz	HS osc mode (PIC16C65-20)
			5	—	200	kHz	LP osc mode
1	Tosc	<b>External CLKIN Period (Note 1)</b>	250	—	—	ns	XT and RC osc mode
			250	—	—	ns	HS osc mode (PIC16C65-04, PIC16LC65-04)
			100	—	—	ns	HS osc mode (PIC16C65-10)
			50	—	—	ns	HS osc mode (PIC16C65-20)
			5	—	—	μs	LP osc mode
		<b>Oscillator Period (Note 1)</b>	250	—	—	ns	RC osc mode
			250	—	10,000	ns	XT osc mode
			250	—	250	ns	HS osc mode (PIC16C65-04, PIC16LC65-04)
			100	—	250	ns	HS osc mode (PIC16C65-10)
			50	—	250	ns	HS osc mode (PIC16C65-20)
2	Tcy	<b>Instruction Cycle Time (Note 1)</b>	200	—	DC	ns	Tcy = 4/Fosc
			200	—	DC	ns	Tcy = 4/Fosc
			200	—	DC	ns	Tcy = 4/Fosc
3	TosL, TosH	<b>External Clock in (OSC1) High or Low Time</b>	50	—	—	ns	XT oscillator
			2.5	—	—	μs	LP oscillator
			10	—	—	ns	HS oscillator
4	TosR, TosF	<b>External Clock in (OSC1) Rise or Fall Time</b>	—	—	25	ns	XT oscillator
			—	—	50	ns	LP oscillator
			—	—	15	ns	HS oscillator

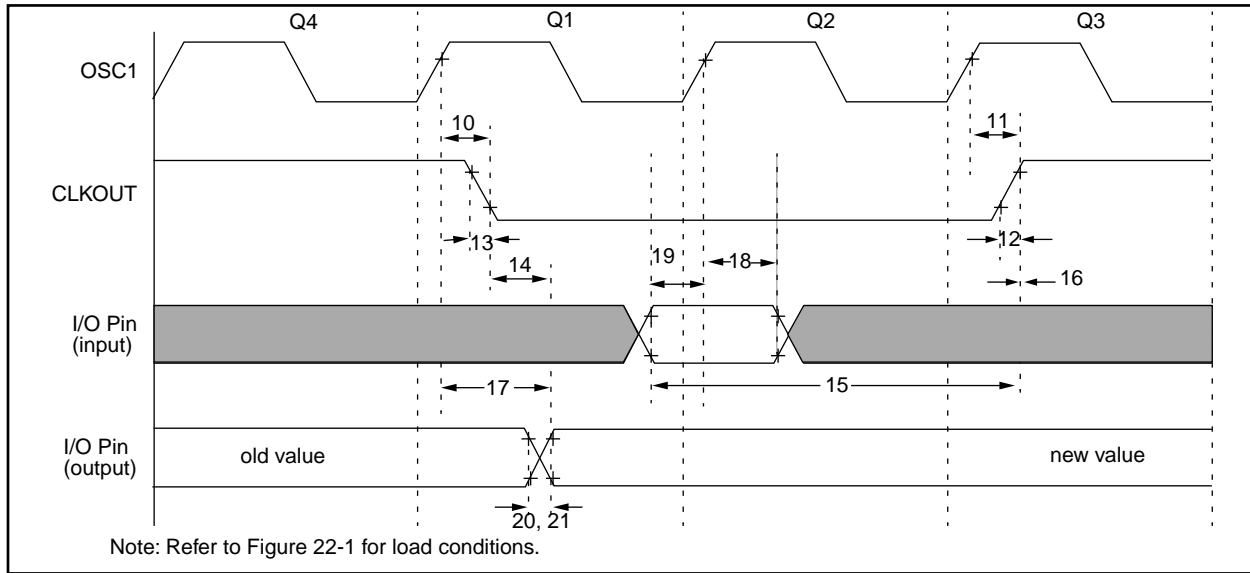
† Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

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

# PIC16C6X

Applicable Devices 61 62 62A R62 63 64 64A R64 65 65A

**FIGURE 22-3: CLKOUT AND I/O TIMING**



**TABLE 22-3: CLKOUT AND I/O TIMING REQUIREMENTS**

Parameter No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
10*	TosH2ckL	OSC1↑ to CLKOUT↓	—	15	30	ns	Note 1
11*	TosH2ckH	OSC1↑ to CLKOUT↑	—	15	30	ns	Note 1
12*	TckR	CLKOUT rise time	—	5	15	ns	Note 1
13*	TckF	CLKOUT fall time	—	5	15	ns	Note 1
14*	TckL2ioV	CLKOUT ↓ to Port out valid	—	—	0.5Tcy + 20	ns	Note 1
15*	TioV2ckH	Port in valid before CLKOUT ↑	0.25Tcy + 25	—	—	ns	Note 1
16*	TckH2ioL	Port in hold after CLKOUT ↑	0	—	—	ns	Note 1
17*	TosH2ioV	OSC1↑ (Q1 cycle) to Port out valid	—	—	80 - 100	ns	
18*	TosH2ioL	OSC1↑ (Q2 cycle) to Port input invalid (I/O in hold time)	TBD	—	—	ns	
19*	TioV2osH	Port input valid to OSC1↑ (I/O in setup time)	TBD	—	—	ns	
20*	TioR	Port output rise time	PIC16C65	—	10	25	ns
			PIC16LC65	—	—	60	ns
21*	TioF	Port output fall time	PIC16C65	—	10	25	ns
			PIC16LC65	—	—	60	ns
22††*	Tinp	RB0/INT pin high or low time	20	—	—	ns	
23††*	Trbp	RB7:RB4 change int high or low time	20	—	—	ns	

\* These parameters are characterized but not tested.

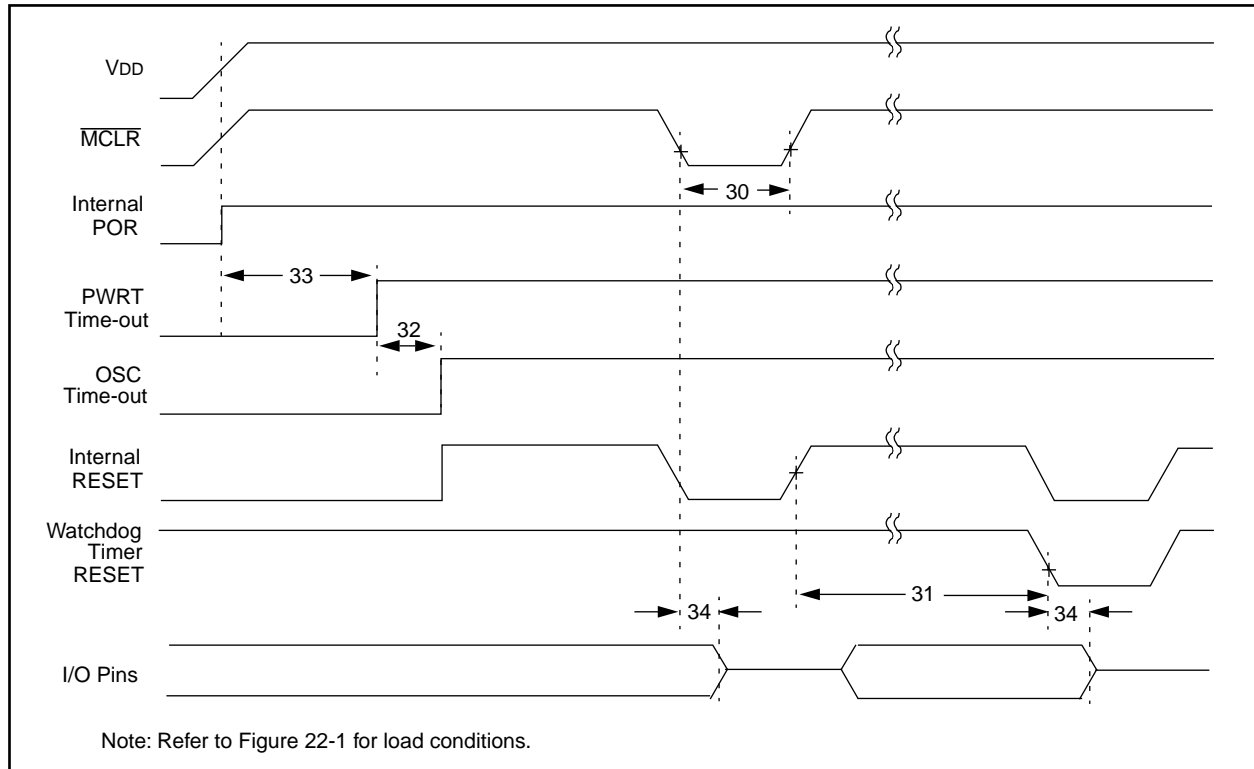
† Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

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

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



**FIGURE 22-4: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER AND POWER-UP TIMER TIMING**



**TABLE 22-4: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER AND POWER-UP TIMER REQUIREMENTS**

Parameter No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
30*	TmCL	MCLR Pulse Width (low)	100	—	—	ns	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	—	1024Tosc	—	—	TOSC = OSC1 period
33*	Tpwrt	Power-up Timer Period or WDT reset	28	72	132	ms	VDD = 5V, -40°C to +125°C
34	Tioz	I/O Hi-impedance from MCLR Low	—	—	100	ns	

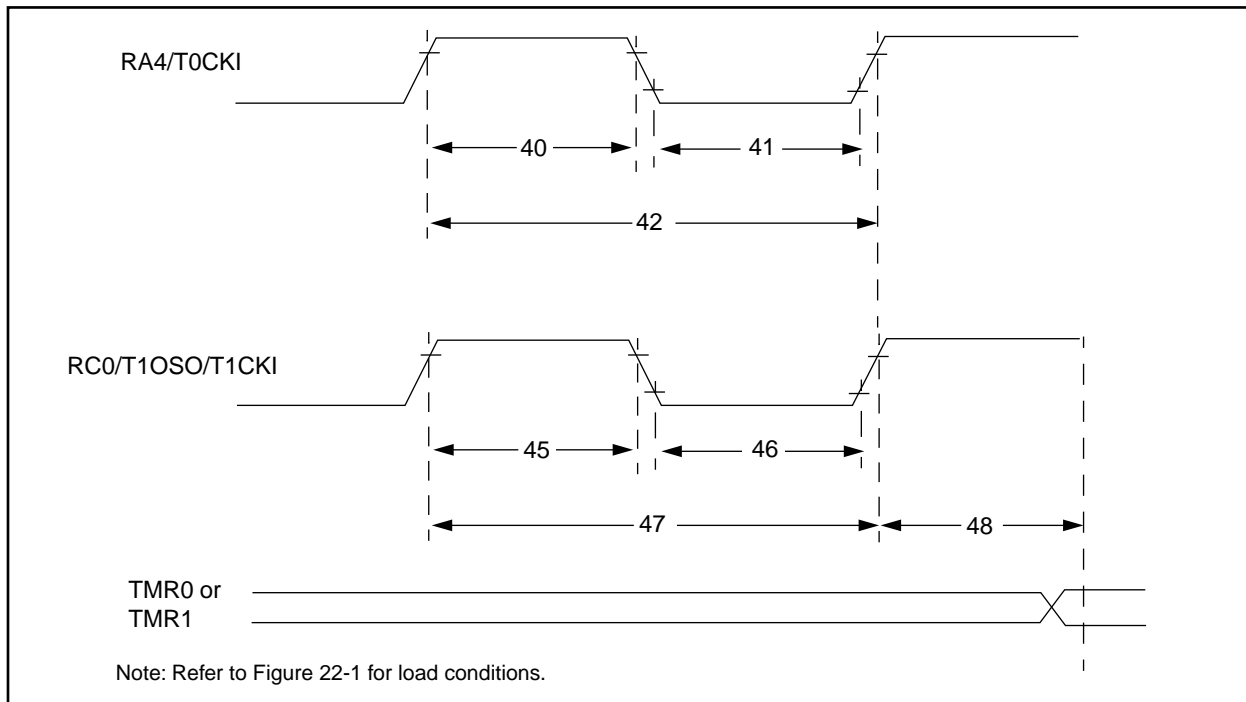
\* These parameters are characterized but not tested.

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

# PIC16C6X

Applicable Devices 61 62 62A R62 63 64 64A R64 65 65A

**FIGURE 22-5: TIMER0 AND TIMER1 CLOCK TIMINGS**



**TABLE 22-5: TIMER0 AND TIMER1 CLOCK REQUIREMENTS**

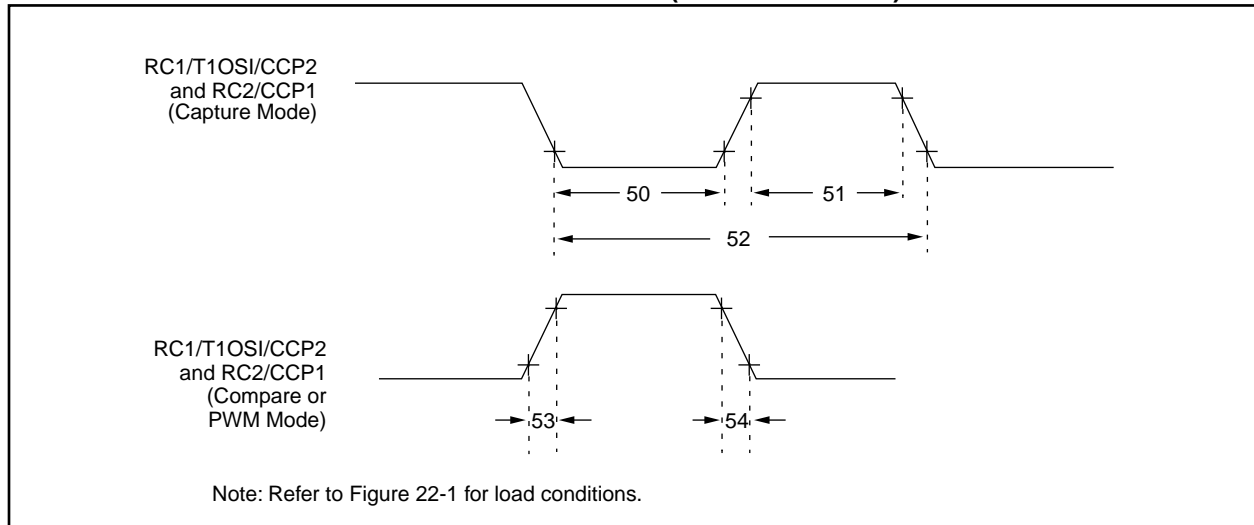
Param No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
40*	Tt0H	T0CKI High Pulse Width	No Prescaler $0.5T_{CY} + 20$ With Prescaler 10	—	—	ns	
41*	Tt0L	T0CKI Low Pulse Width	No Prescaler $0.5T_{CY} + 20$ With Prescaler 10	—	—	ns	
42*	Tt0P	T0CKI Period	Greater of: $20 \mu s$ or $\frac{T_{CY} + 40}{N}$	—	—	ns	N = prescale value (1, 2, 4, ..., 256)
45*	Tt1H	T1CKI High Time	Synchronous, No Prescaler $0.5T_{CY} + 20$ Synchronous, With Prescaler PIC16C65: 10 PIC16LC65: 20 Asynchronous $2 T_{CY}$	—	—	ns	
46*	Tt1L	T1CKI Low Time	Synchronous, No Prescaler $0.5T_{CY} + 20$ Synchronous, With Prescaler PIC16C65: 10 PIC16LC65: 20 Asynchronous $2T_{CY}$	—	—	ns	
47*	Tt1P	T1CKI input period	Synchronous Greater of: $20 \mu s$ or $\frac{T_{CY} + 40}{N}$ Asynchronous Greater of: $20 \mu s$ or $4T_{CY}$	—	—	ns	N = prescale value (1, 2, 4, 8)
	Ft1	Timer1 oscillator input frequency range (oscillator enabled by setting bit T1OSCEN)	DC	—	200	kHz	
48	TCKEztmr1	Delay from external clock edge to timer increment	$2T_{osc}$	—	$7T_{osc}$	—	

\* These parameters are characterized but not tested.

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

Applicable Devices	61	62	62A	R62	63	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

**FIGURE 22-6: CAPTURE/COMPARE/PWM TIMINGS (CCP1 AND CCP2)**



**TABLE 22-6: CAPTURE/COMPARE/PWM REQUIREMENTS (CCP1 AND CCP2)**

Parameter No.	Sym	Characteristic			Min	Typ†	Max	Units	Conditions
50*	TccL	CCP1 and CCP2 input low time	No Prescaler		0.5Tcy + 20	—	—	ns	
			With Prescaler	PIC16C65	10	—	—	ns	
				PIC16LC65	20	—	—	ns	
51*	TccH	CCP1 and CCP2 input high time	No Prescaler		0.5Tcy + 20	—	—	ns	
			With Prescaler	PIC16C65	10	—	—	ns	
				PIC16LC65	20	—	—	ns	
52*	TccP	CCP1 and CCP2 input period			$\frac{3Tcy + 40}{N}$	—	—	ns	N = prescale value (1,4, or 16)
53	TccR	CCP1 and CCP2 output rise time			—	10	25	ns	
54	TccF	CCP1 and CCP2 output fall time			—	10	25	ns	

\* These parameters are characterized but not tested.

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

# PIC16C6X

Applicable Devices	61	62	62A	R62	63	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

FIGURE 22-7: PARALLEL SLAVE PORT TIMING

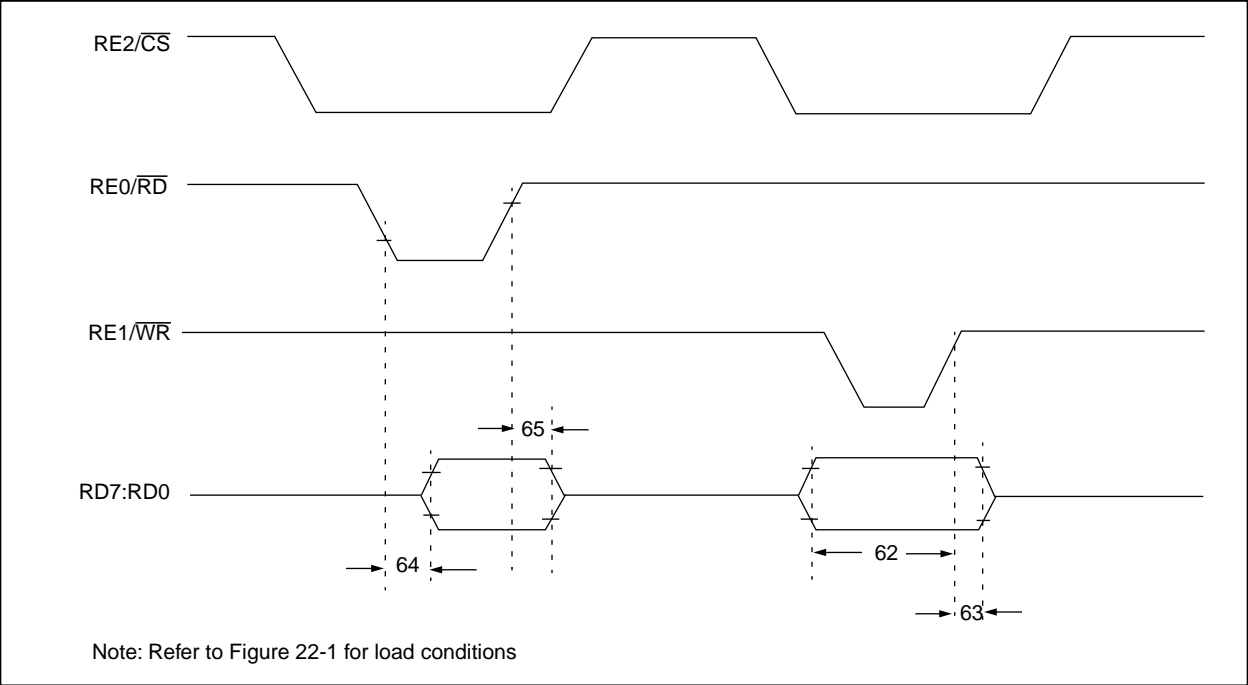
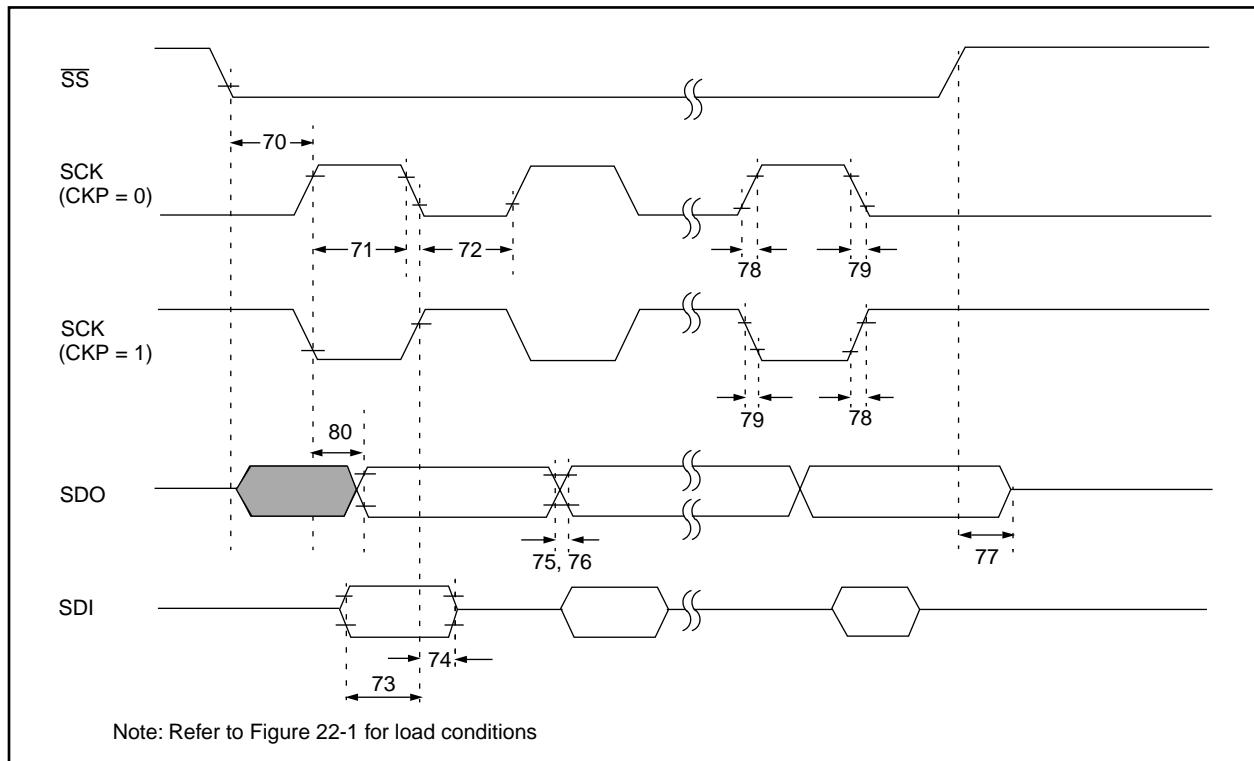


TABLE 22-7: PARALLEL SLAVE PORT REQUIREMENTS

Parameter No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
62	TdtV2wrH	Data in valid before $\overline{WR}\uparrow$ or $\overline{CS}\uparrow$ (setup time)	20 25	— —	— —	ns ns	 Automotive Range Only
63*	TwrH2dtI	$\overline{WR}\uparrow$ or $\overline{CS}\uparrow$ to data-in invalid (hold time)	PIC16C65: 20 PIC16LC65: 35	— —	— —	ns ns	
64	TrdL2dtV	$\overline{RD}\downarrow$ and $\overline{CS}\downarrow$ to data-out valid	— —	— —	80 90	ns ns	 Automotive Range Only
65	TrdH2dtI	$\overline{RD}\uparrow$ or $\overline{CS}\uparrow$ to data-out invalid	10	—	30	ns	

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

**FIGURE 22-8: SPI MODE TIMING**



**TABLE 22-8: SPI MODE REQUIREMENTS**

Parameter No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
70	TssL2scH, TssL2scL	$\overline{SS}$ ↓ to SCK↓ or SCK↑ input	T <sub>CY</sub>	—	—	ns	
71	TscH	SCK input high time (slave mode)	T <sub>CY</sub> + 20	—	—	ns	
72	TscL	SCK input low time (slave mode)	T <sub>CY</sub> + 20	—	—	ns	
73	TdiV2scH, TdiV2scL	Setup time of SDI data input to SCK edge	100	—	—	ns	
74	Tsch2diL, TscL2diL	Hold time of SDI data input to SCK edge	100	—	—	ns	
75	TdoR	SDO data output rise time	—	10	25	ns	
76	TdoF	SDO data output fall time	—	10	25	ns	
77	TssH2doZ	$\overline{SS}$ ↓ to SDO output hi-impedance	10	—	50	ns	
78	TscR	SCK output rise time (master mode)	—	10	25	ns	
79	TscF	SCK output fall time (master mode)	—	10	25	ns	
80	Tsch2doV, TscL2doV	SDO data output valid after SCK edge	—	—	50	ns	

† 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	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

FIGURE 22-9: I<sup>2</sup>C BUS START/STOP BITS TIMING

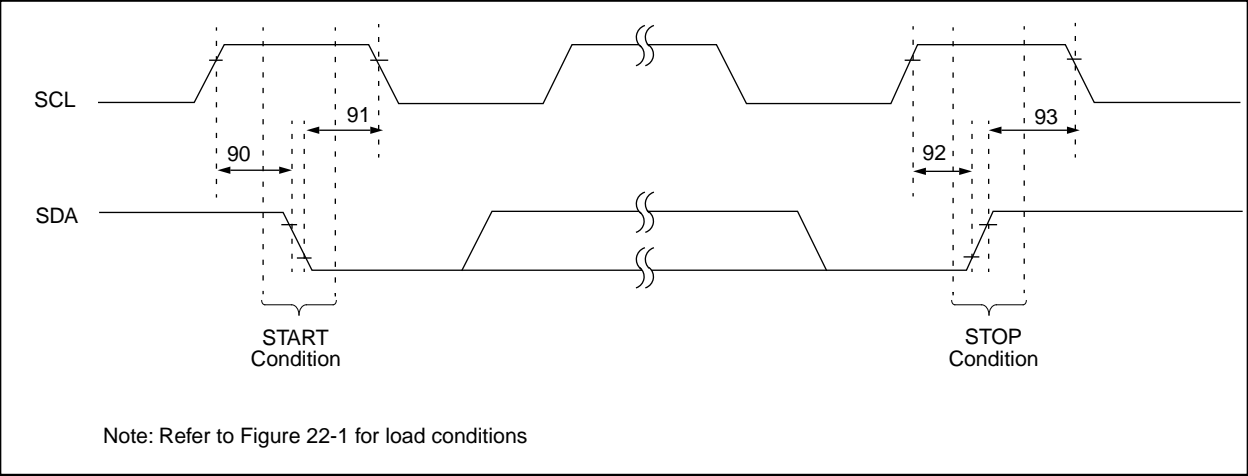
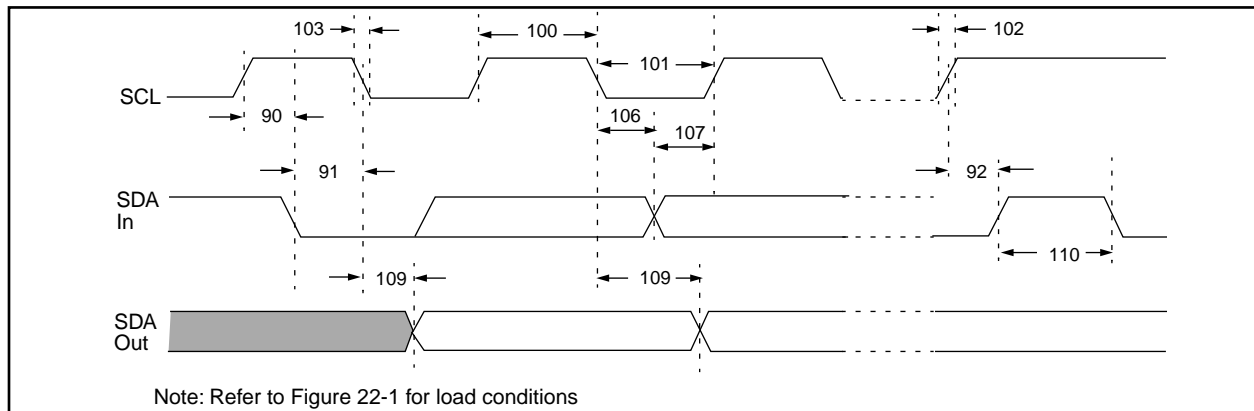


TABLE 22-9: I<sup>2</sup>C BUS START/STOP BITS REQUIREMENTS

Parameter No.	Sym	Characteristic		Min	Typ	Max	Units	Conditions
90	TSU:STA	START condition	100 kHz mode	4700	—	—	ns	Only relevant for repeated START condition
		Setup time	400 kHz mode	600	—	—		
91	THD:STA	START condition	100 kHz mode	4000	—	—	ns	After this period the first clock pulse is generated
		Hold time	400 kHz mode	600	—	—		
92	TSU:STO	STOP condition	100 kHz mode	4700	—	—	ns	
		Setup time	400 kHz mode	600	—	—		
93	THD:STO	STOP condition	100 kHz mode	4000	—	—	ns	
		Hold time	400 kHz mode	600	—	—		

**FIGURE 22-10: I<sup>2</sup>C BUS DATA TIMING**



**TABLE 22-10: I<sup>2</sup>C BUS DATA REQUIREMENTS**

Parameter No.	Sym	Characteristic		Min	Max	Units	Conditions
100	THIGH	Clock high time	100 kHz mode	4.0	—	μs	PIC16C65 must operate at a minimum of 1.5 MHz
			400 kHz mode	0.6	—	μs	PIC16C65 must operate at a minimum of 10 MHz
			SSP Module	1.5TCY	—		
101	TLOW	Clock low time	100 kHz mode	4.7	—	μs	PIC16C65 must operate at a minimum of 1.5 MHz
			400 kHz mode	1.3	—	μs	PIC16C65 must operate at a minimum of 10 MHz
			SSP Module	1.5TCY	—		
102	TR	SDA and SCL rise time	100 kHz mode	—	1000	ns	
			400 kHz mode	20 + 0.1Cb	300	ns	Cb is specified to be from 10-400 pF
103	TF	SDA and SCL fall time	100 kHz mode	—	300	ns	
			400 kHz mode	20 + 0.1Cb	300	ns	Cb is specified to be from 10-400 pF
90	TSU:STA	START condition setup time	100 kHz mode	4.7	—	μs	Only relevant for repeated START condition
			400 kHz mode	0.6	—	μs	
91	THD:STA	START condition hold time	100 kHz mode	4.0	—	μs	After this period the first clock pulse is generated
			400 kHz mode	0.6	—	μs	
106	THD:DAT	Data input hold time	100 kHz mode	0	—	ns	
			400 kHz mode	0	0.9	μs	
107	TSU:DAT	Data input setup time	100 kHz mode	250	—	ns	Note 2
			400 kHz mode	100	—	ns	
92	TSU:STO	STOP condition setup time	100 kHz mode	4.7	—	μs	
			400 kHz mode	0.6	—	μs	
109	TAA	Output valid from clock	100 kHz mode	—	3500	ns	Note 1
			400 kHz mode	—	—	ns	
110	TBUF	Bus free time	100 kHz mode	4.7	—	μs	Time the bus must be free before a new transmission can start
			400 kHz mode	1.3	—	μs	
	Cb	Bus capacitive loading		—	400	pF	

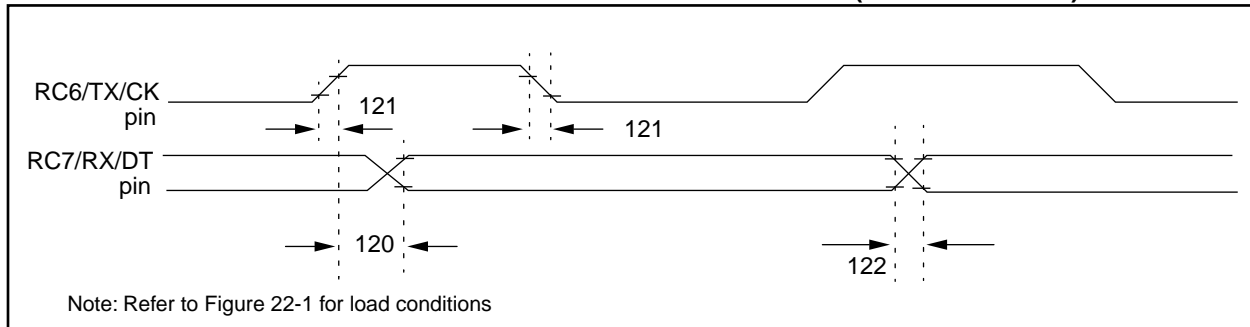
Note 1: As a transmitter, the device must provide this internal minimum delay time to bridge the undefined region (min. 300 ns) of the falling edge of SCL to avoid unintended generation of START or STOP conditions.

Note 2: A fast-mode I<sup>2</sup>C-bus device can be used in a standard-mode I<sup>2</sup>C-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 T<sub>R</sub> max.+tsu:DAT = 1000 + 250 = 1250 ns (according to the standard-mode I<sup>2</sup>C bus specification) before the SCL line is released.

# PIC16C6X

Applicable Devices 61 62 62A R62 63 64 64A R64 65 65A

**FIGURE 22-11: USART MODULE: SYNCHRONOUS TRANSMISSION (MASTER/SLAVE) TIMING**

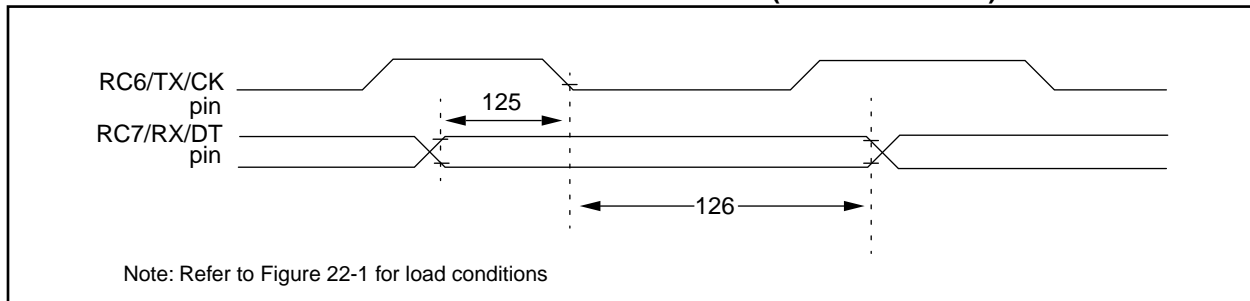


**TABLE 22-11: SERIAL PORT SYNCHRONOUS TRANSMISSION REQUIREMENTS**

Parameter No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
120	tckH2dtV	SYNC XMIT (MASTER & SLAVE) Clock high to data out valid	—	—	50	ns	PIC16C65
			—	—	100	ns	PIC16LC65
121	tckrf	Clock out rise time and fall time (Master Mode)	—	—	25	ns	PIC16C65
			—	—	50	ns	PIC16LC65
122	tdtrf	Data out rise time and fall time	—	—	25	ns	PIC16C65
			—	—	50	ns	PIC16LC65

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

**FIGURE 22-12: USART MODULE: SYNCHRONOUS RECEIVE (MASTER/SLAVE) TIMING**



**TABLE 22-12: SERIAL PORT SYNCHRONOUS RECEIVE REQUIREMENTS**

Parameter No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
125	tdtV2ckl	SYNC RCV (MASTER & SLAVE) Data hold before CK ↓ (DT hold time)	15	—	—	ns	
126	tckL2dtl	Data hold after CK ↓ (DT hold time)	15	—	—	ns	

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



Applicable Devices	61	62	62A	R62	63	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

23.0 DC AND AC CHARACTERISTICS GRAPHS AND TABLES FOR PIC16C65

NOT AVAILABLE AT THIS TIME

# PIC16C6X

---

NOTES:

Applicable Devices	61	62	62A	R62	63	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

## 24.0 ELECTRICAL CHARACTERISTICS FOR PIC16C63/65A

### Absolute Maximum Ratings <sup>(†)</sup>

Ambient temperature under bias.....	-55 to + 125°C
Storage temperature .....	-65°C to + 150°C
Voltage on any pin with respect to VSS (except VDD and $\overline{\text{MCLR}}$ ) .....	-0.3V to (VDD +0.3V)
Voltage on VDD with respect to VSS .....	0 to + 7.5V
Voltage on $\overline{\text{MCLR}}$ with respect to VSS (Note 2) .....	0 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 <sub>IK</sub> (V <sub>I</sub> < 0 or V <sub>I</sub> > VDD) .....	±20 mA
Output clamp current, I <sub>OK</sub> (V <sub>O</sub> < 0 or V <sub>O</sub> > 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 <sup>(*)</sup> (combined) .....	200 mA
Maximum current sourced by PORTA, PORTB, and PORTE <sup>*</sup> (combined) .....	200 mA
Maximum current sunk by PORTC and PORTD <sup>(*)</sup> (combined).....	200 mA
Maximum current sourced by PORTC and PORTD <sup>(*)</sup> (combined) .....	200 mA

\* PORTD and PORTE not available on the PIC16C63.

**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  $\overline{\text{MCLR}}$ /VPP pin, inducing currents greater than 80 mA, may cause latch-up. Thus, a series resistor of 50-100Ω should be used when applying a “low” level to the  $\overline{\text{MCLR}}$ /VPP 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 24-1: CROSS REFERENCE OF DEVICE SPECS FOR OSCILLATOR CONFIGURATIONS AND FREQUENCIES OF OPERATION (COMMERCIAL DEVICES)**

OSC	PIC16C63-04 PIC16C65A-04	PIC16C63-10 PIC16C65A-10	PIC16C63-20 PIC16C65A-20	PIC16LC63-04 PIC16LC65A-04	JW Devices
RC	VDD: 4.0V to 6.0V IDD: 5 mA max. at 5.5V IPD: 21 µA max. at 4V Freq: 4 MHz max.	VDD: 4.5V to 5.5V IDD: 2.7 mA typ. at 5.5V IPD: 1.5 µA typ. at 4V Freq: 4 MHz max.	VDD: 4.5V to 5.5V IDD: 2.7 mA typ. at 5.5V IPD: 1.5 µA typ. at 4V Freq: 4 MHz max.	VDD: 2.5V to 6.0V IDD: 2.0 mA typ. at 3V IPD: 0.9 µA typ. at 3V Freq: 4 MHz max.	VDD: 4.0V to 6.0V IDD: 5 mA max. at 5.5V IPD: 21 µA max. at 4V Freq: 4 MHz max.
XT	VDD: 4.0V to 6.0V IDD: 5 mA max. at 5.5V IPD: 21 µA max. at 4V Freq: 4 MHz max.	VDD: 4.5V to 5.5V IDD: 2.7 mA typ. at 5.5V IPD: 1.5 µA typ. at 4V Freq: 4 MHz max.	VDD: 4.5V to 5.5V IDD: 2.7 mA typ. at 5.5V IPD: 1.5 µA typ. at 4V Freq: 4 MHz max.	VDD: 2.5V to 6.0V IDD: 2.0 mA typ. at 3V IPD: 0.9 µA typ. at 3V Freq: 4 MHz max.	VDD: 4.0V to 6.0V IDD: 5 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: 30 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.	Do not 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.	Do not use in LP mode	Do not use in LP mode	VDD: 2.5V 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: 2.5V 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	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

## 24.1 DC CHARACTERISTICS: PIC16C63-04, PIC16C65A-04 (Commercial, Industrial, Automotive<sup>(6)</sup>) PIC16C63-10, PIC16C65A-10 (Commercial, Industrial, Automotive<sup>(6)</sup>) PIC16C63-20, PIC16C65A-20 (Commercial, Industrial, Automotive<sup>(6)</sup>)

DC CHARACTERISTICS		Standard Operating Conditions (unless otherwise stated)					
		Operating temperature					
		-40°C ≤ TA ≤ +125°C for automotive,					
		-40°C ≤ TA ≤ +85°C for industrial and					
		0°C ≤ TA ≤ +70°C for commercial					
Param No.	Characteristic	Sym	Min	Typ†	Max	Units	Conditions
D001 D001A	Supply Voltage	VDD	4.0 4.5	- -	6.0 5.5	V V	XT, RC and LP osc configuration HS osc configuration
D002*	RAM Data Retention Voltage (Note 1)	VDR	-	1.5	-	V	Device in SLEEP mode
D003	VDD start voltage to ensure Power-on Reset	VPOR	-	VSS	-	V	See section on Power-on Reset for details
D004*	VDD rise rate to ensure Power-on Reset	SVDD	0.05	-	-	V/ms	See section on Power-on Reset for details
D005	Brown-out Reset Voltage	BVDD	3.7 3.7	4.0 4.0	4.3 4.4	V V	BODEN configuration bit is enabled Automotive Range Only
D010	Supply Current (Note 2, 5)	IDD	-	2.7	5	mA	XT, RC, osc configuration (PIC16C63/65A-04) FOSC = 4 MHz, VDD = 5.5V (Note 4)
D013			-	13.5	30	mA	HS osc configuration (PIC16C63/65A-20) FOSC = 20 MHz, VDD = 5.5V
D015*	Brown-out Reset Current (Note 7)	ΔIBOR	-	350	425	μA	BOR enabled, VDD = 5.0V
D020 D021 D021A D021B	Power-down Current (Note 3, 5)	IPD	- - - -	10.5 1.5 1.5 2.5	42 21 24 24	μA μA μA μA	VDD = 4.0V, WDT enabled, -40°C to +85°C VDD = 4.0V, WDT disabled, -0°C to +70°C VDD = 4.0V, WDT disabled, -40°C to +85°C VDD = 4.0V, WDT disabled, -40°C to +125°C
D023*	Brown-out Reset Current (Note 7)	ΔIBOR	-	350	425	μA	BOR enabled, VDD = 5.0V

\* These parameters are characterized but not tested.

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

Note 1: This is the limit to which VDD can be lowered in SLEEP mode without losing RAM data.

2: The supply current is mainly a function of the operating voltage and frequency. Other factors such as I/O pin loading and switching rate, oscillator type, internal code execution pattern, and temperature also have an impact on the current consumption.

The test conditions for all IDD measurements in active operation mode are:

OSC1 = external square wave, from rail to rail; all I/O pins tristated, pulled to VDD,

MCLR = VDD; WDT enabled/disabled as specified.

3: The power down current in SLEEP mode does not depend on the oscillator type. Power-down current is measured with the part in SLEEP mode, with all I/O pins in hi-impedance state and tied to VDD and VSS.

4: For RC osc configuration, current through Rext is not included. The current through the resistor can be estimated by the formula  $I_r = VDD/2R_{ext}$  (mA) with Rext in kOhm.

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

6: Automotive operating range is Advanced information for this device.

7: The Δ current is the additional current consumed when this peripheral is enabled. This current should be added to the base IDD or IPD measurement.

## 24.2 DC CHARACTERISTICS: PIC16LC63-04, PIC16LC65A-04 (Commercial, Industrial, Automotive)<sup>(6)</sup>

Standard Operating Conditions (unless otherwise stated)							
DC CHARACTERISTICS							
Operating temperature    -40°C    ≤ T <sub>A</sub> ≤ +125°C for automotive, -40°C    ≤ T <sub>A</sub> ≤ +85°C for industrial and 0°C        ≤ T <sub>A</sub> ≤ +70°C for commercial							
Param No.	Characteristic	Sym	Min	Typ†	Max	Units	Conditions
D001	Supply Voltage	V <sub>DD</sub>	2.5	-	6.0	V	LP, XT, RC osc configuration (DC - 4 MHz)
D002*	RAM Data Retention Voltage (Note 1)	V <sub>DR</sub>	-	1.5	-	V	Device in SLEEP mode
D003	V <sub>DD</sub> start voltage to ensure Power-on Reset	V <sub>POR</sub>	-	V <sub>SS</sub>	-	V	See section on Power-on Reset for details
D004*	V <sub>DD</sub> rise rate to ensure Power-on Reset	S <sub>VDD</sub>	0.05	-	-	V/ms	See section on Power-on Reset for details
D005	Brown-out Reset Voltage	B <sub>VDD</sub>	3.7 3.7	4.0 4.0	4.3 4.4	V V	BODEN configuration bit is enabled Automotive Range Only
D010	Supply Current (Note 2, 5)	I <sub>DD</sub>	-	2.0	3.8	mA	XT, RC osc configuration FOSC = 4 MHz, V <sub>DD</sub> = 3.0V (Note 4)
D010A						μA	LP osc configuration FOSC = 32 kHz, V <sub>DD</sub> = 3.0V, WDT disabled
D015*	Brown-out Reset Current (Note 7)	ΔI <sub>BOR</sub>	-	350	425	μA	BOR enabled, V <sub>DD</sub> = 5.0V
D020	Power-down Current (Note 3, 5)	I <sub>PD</sub>	-	7.5	30	μA	V <sub>DD</sub> = 3.0V, WDT enabled, -40°C to +85°C
D021			-	0.9	5	μA	V <sub>DD</sub> = 3.0V, WDT disabled, 0°C to +70°C
D021A			-	0.9	5	μA	V <sub>DD</sub> = 3.0V, WDT disabled, -40°C to +85°C
D021B			-	0.9	10	μA	V <sub>DD</sub> = 3.0V, WDT disabled, -40°C to +125°C
D023*	Brown-out Reset Current (Note 7)	ΔI <sub>BOR</sub>	-	350	425	μA	BOR enabled, V <sub>DD</sub> = 5.0V

\* These parameters are characterized but not tested.

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

Note 1: This is the limit to which V<sub>DD</sub> can be lowered in SLEEP mode without losing RAM data.

2: The supply current is mainly a function of the operating voltage and frequency. Other factors such as I/O pin loading and switching rate, oscillator type, internal code execution pattern, and temperature also have an impact on the current consumption.

The test conditions for all I<sub>DD</sub> measurements in active operation mode are:

OSC1 = external square wave, from rail to rail; all I/O pins tristated, pulled to V<sub>DD</sub>,

MCLR = V<sub>DD</sub>; WDT enabled/disabled as specified.

3: The power-down current in SLEEP mode does not depend on the oscillator type. Power-down current is measured with the part in SLEEP mode, with all I/O pins in hi-impedance state and tied to V<sub>DD</sub> and V<sub>SS</sub>.

4: For RC osc configuration, current through R<sub>ext</sub> is not included. The current through the resistor can be estimated by the formula I<sub>r</sub> = V<sub>DD</sub>/2R<sub>ext</sub> (mA) with R<sub>ext</sub> in kOhm.

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

6: Automotive operating range is Advanced information for this device.

7: The Δ current is the additional current consumed when this peripheral is enabled. This current should be added to the base I<sub>DD</sub> or I<sub>PD</sub> measurement.

# PIC16C6X

Applicable Devices	61	62	62A	R62	63	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

## 24.3 DC CHARACTERISTICS: PIC16C63-04, PIC16C65A-04 (Commercial, Industrial, Automotive) PIC16C63-10, PIC16C65A-10 (Commercial, Industrial, Automotive) PIC16C63-20, PIC16C65A-20 (Commercial, Industrial, Automotive) PIC16LC63-04, PIC16LC65A-04 (Commercial, Industrial, Automotive)

DC CHARACTERISTICS		Standard Operating Conditions (unless otherwise stated)					
		Operating temperature    -40°C    ≤ TA ≤ +125°C for automotive, -40°C    ≤ TA ≤ +85°C for industrial and 0°C        ≤ TA ≤ +70°C for commercial					
		Operating voltage VDD range as described in DC spec Section 24.1 and Section 24.2					
Param No.	Characteristic	Sym	Min	Typ †	Max	Units	Conditions
D030 D031 D032 D033	<b>Input Low Voltage</b> I/O ports	VIL					
	with TTL buffer		VSS	-	0.5V	V	
	with Schmitt Trigger buffer		VSS	-	0.2VDD	V	
	MCLR, RA4/T0CKI, OSC1 (in RC mode)		VSS	-	0.2VDD	V	
D033	OSC1 (in XT, HS and LP)		VSS	-	0.3VDD	V	Note1
D040 D040A D041 D042 D042A D043	<b>Input High Voltage</b> I/O ports	VIH					
	with TTL buffer		2.0	-	VDD	V	4.5V ≤ VDD ≤ 5.5V
			0.8VDD	-	VDD	V	For VDD > 5.5V or VDD < 4.5V
	with Schmitt Trigger buffer		0.8VDD	-	VDD	V	For entire VDD range
	MCLR RA4/T0CKI, RC7:RC4, RD7:RD4, RB0/INT, RE2:RE0		0.8VDD	-	VDD	V	
	OSC1 (XT, HS and LP)		0.7VDD	-	VDD	V	Note1
D043	OSC1 (in RC mode)		0.9VDD	-	VDD	V	
D070	PORTB weak pull-up current	IPURB	50	250	400	μA	VDD = 5V, VPIN = VSS
D060 D061 D063	<b>Input Leakage Current</b> (Notes 2, 3) I/O ports	IIL	-	-	±1	μA	VSS ≤ VPIN ≤ VDD, Pin at hi-impedance
	MCLR, RA4/T0CKI		-	-	±5	μA	VSS ≤ VPIN ≤ VDD
	OSC1		-	-	±5	μA	VSS ≤ VPIN ≤ VDD, XT, HS and LP osc configuration
D080 D080A D083 D083A	<b>Output Low Voltage</b> I/O ports	VOL	-	-	0.6	V	IoL = 8.5 mA, VDD = 4.5V, -40°C to +85°C
			-	-	0.6	V	IoL = 7.0 mA, VDD = 4.5V, -40°C to +125°C
	OSC2/CLKOUT (RC osc config)		-	-	0.6	V	IoL = 1.6 mA, VDD = 4.5V, -40°C to +85°C
			-	-	0.6	V	IoL = 1.2 mA, VDD = 4.5V, -40°C to +125°C

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

- Note 1: In RC oscillator configuration, the OSC1/CLKIN pin is a Schmitt Trigger input. It is not recommended that the PIC16C6X be driven with external clock in RC mode.
- 2: The leakage current on the MCLR/VPP pin is strongly dependent on the applied voltage level. The specified levels represent normal operating conditions. Higher leakage current may be measured at different input voltages.
- 3: Negative current is defined as coming out of the pin.

Applicable Devices	61	62	62A	R62	63	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

<b>DC CHARACTERISTICS</b> <div> <b>Standard Operating Conditions (unless otherwise stated)</b>            Operating temperature    <math>-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}</math> for automotive,               <math>-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}</math> for industrial and               <math>0^{\circ}\text{C} \leq T_A \leq +70^{\circ}\text{C}</math> for commercial             Operating voltage <math>V_{DD}</math> range as described in DC spec Section 24.1            and Section 24.2         </div>							
Param No.	Characteristic	Sym	Min	Typ †	Max	Units	Conditions
D090	<b>Output High Voltage</b> I/O ports (Note 3)	VOH	VDD-0.7	-	-	V	IOH = -3.0 mA, VDD = 4.5V, -40°C to +85°C
D090A			VDD-0.7	-	-	V	IOH = -2.5 mA, VDD = 4.5V, -40°C to +125°C
D092			VDD-0.7	-	-	V	IOH = -1.3 mA, VDD = 4.5V, -40°C to +85°C
D092A			VDD-0.7	-	-	V	IOH = -1.0 mA, VDD = 4.5V, -40°C to +125°C
D100	<b>Capacitive Loading Specs on Output Pins</b> OSC2 pin	COSC2	-	-	15	pF	In XT, HS and LP modes when external clock is used to drive OSC1.
D101	All I/O pins and OSC2 (in RC mode)	CIO	-	-	50	pF	
D102	SCL, SDA in I <sup>2</sup> C mode	Cb	-	-	400	pF	

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

- Note 1: In RC oscillator configuration, the OSC1/CLKIN pin is a Schmitt Trigger input. It is not recommended that the PIC16C6X be driven with external clock in RC mode.
- 2: The leakage current on the  $\overline{\text{MCLR}}/\text{VPP}$  pin is strongly dependent on the applied voltage level. The specified levels represent normal operating conditions. Higher leakage current may be measured at different input voltages.
- 3: Negative current is defined as coming out of the pin.

# PIC16C6X

Applicable Devices	61	62	62A	R62	63	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

## 24.4 Timing Parameter Symbolology

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

1. TppS2ppS
2. TppS
3. TCC:ST (I<sup>2</sup>C specifications only)
4. Ts (I<sup>2</sup>C specifications only)

<b>T</b>			
F	Frequency	T	Time

Lowercase letters (pp) and their meanings:

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

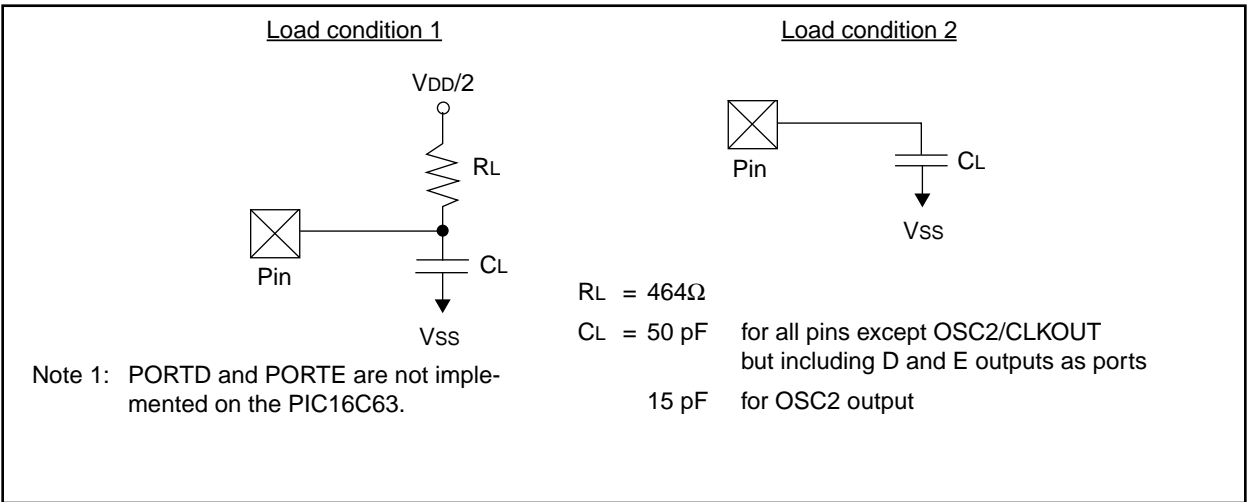
Uppercase letters and their meanings:

<b>S</b>			
F	Fall	P	Period
H	High	R	Rise
I	Invalid (Hi-impedance)	V	Valid
L	Low	Z	Hi-impedance
<b>I<sup>2</sup>C only</b>			
AA	output access	High	High
BUF	Bus free	Low	Low

TCC:ST (I<sup>2</sup>C specifications only)

<b>CC</b>			
HD	Hold	SU	Setup
<b>ST</b>			
DAT	DATA input hold	STO	STOP condition
STA	START condition		

FIGURE 24-1: LOAD CONDITIONS FOR DEVICE TIMING SPECIFICATIONS

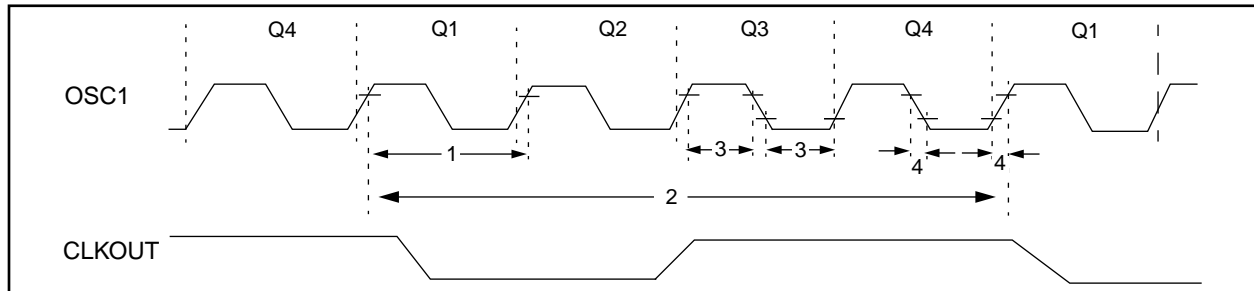




Applicable Devices	61	62	62A	R62	63	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

## 24.5 Timing Diagrams and Specifications

**FIGURE 24-2: EXTERNAL CLOCK TIMING**



**TABLE 24-2: EXTERNAL CLOCK TIMING REQUIREMENTS**

Param No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
	Fosc	<b>External CLKIN Frequency (Note 1)</b>	DC	—	4	MHz	XT and RC osc mode
			DC	—	4	MHz	HS osc mode (PIC16C63/65A-04,
			DC	—	20	MHz	HS osc mode (PIC16C63/65A-20)
			DC	—	200	kHz	LP osc mode
		<b>Oscillator Frequency (Note 1)</b>	DC	—	4	MHz	RC osc mode
			0.1	—	4	MHz	XT osc mode
			4	—	4	MHz	HS osc mode (PIC16C63/65A-04,
			4	—	10	MHz	HS osc mode (PIC16C63/65A-10)
			4	—	20	MHz	HS osc mode (PIC16C63/65A-20)
			5	—	200	kHz	LP osc mode
1	Tosc	<b>External CLKIN Period (Note 1)</b>	250	—	—	ns	XT and RC osc mode
			250	—	—	ns	HS osc mode (PIC16C63/65A-04)
			100	—	—	ns	HS osc mode (PIC16C63/65A-10)
			50	—	—	ns	HS osc mode (PIC16C63/65A-20)
			5	—	—	μs	LP osc mode
		<b>Oscillator Period (Note 1)</b>	250	—	—	ns	RC osc mode
			250	—	10,000	ns	XT osc mode
			250	—	250	ns	HS osc mode (PIC16C63/65A-04)
			100	—	250	ns	HS osc mode (PIC16C63/65A-10)
			50	—	250	ns	HS osc mode (PIC16C63/65A-20)
2	Tcy	<b>Instruction Cycle Time (Note 1)</b>	200	—	DC	ns	Tcy = 4/Fosc
			—	—	—	—	—
3*	TosL, TosH	<b>External Clock in (OSC1) High or Low Time</b>	100	—	—	ns	XT oscillator
			2	—	—	μs	LP oscillator
			20	—	—	ns	HS oscillator
4*	TosR, TosF	<b>External Clock in (OSC1) Rise or Fall Time</b>	—	—	25	ns	XT oscillator
			—	—	50	ns	LP oscillator
			—	—	15	ns	HS oscillator

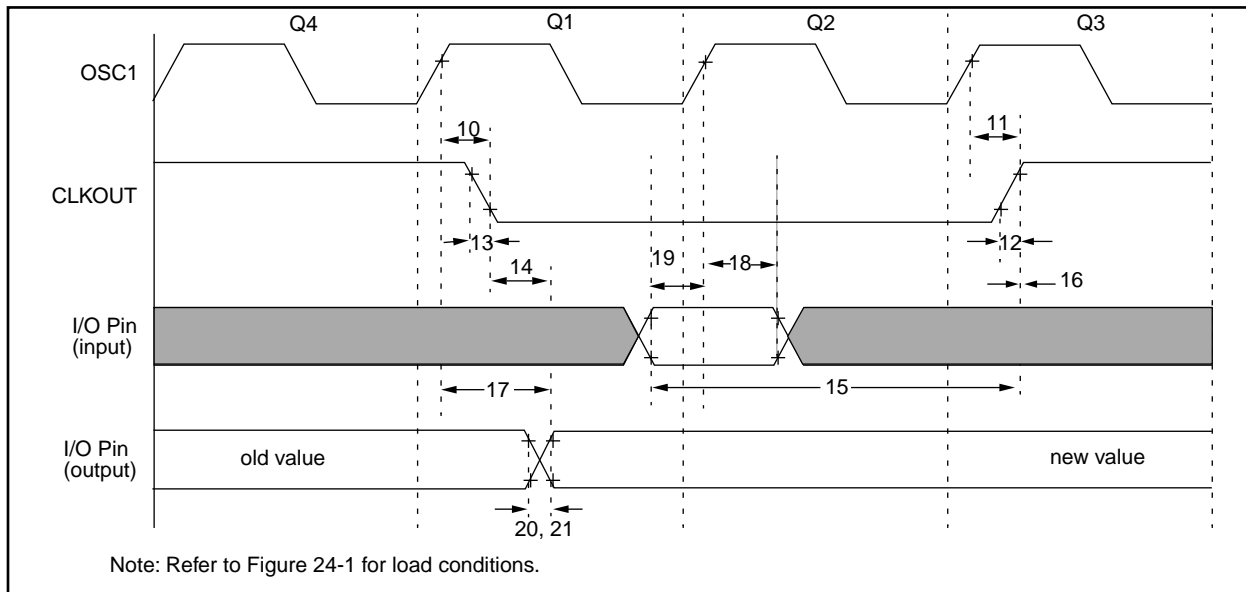
† Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

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

# PIC16C6X

Applicable Devices 61 62 62A R62 63 64 64A R64 65 65A

**FIGURE 24-3: CLKOUT AND I/O TIMING**



**TABLE 24-3: CLKOUT AND I/O TIMING REQUIREMENTS**

Parameter No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
10*	TosH2ckL	OSC1↑ to CLKOUT↓	—	75	200	ns	Note 1
11*	TosH2ckH	OSC1↑ to CLKOUT↑	—	75	200	ns	Note 1
12*	TckR	CLKOUT rise time	—	3	100	ns	Note 1
13*	TckF	CLKOUT fall time	—	3	100	ns	Note 1
14*	TckL2ioV	CLKOUT ↓ to Port out valid	—	—	0.5Tcy + 20	ns	Note 1
15*	TioV2ckH	Port in valid before CLKOUT ↑	Tosc + 200	—	—	ns	Note 1
16*	TckH2iol	Port in hold after CLKOUT ↑	0	—	—	ns	Note 1
17*	TosH2ioV	OSC1↑ (Q1 cycle) to Port out valid	—	50	150	ns	
18*	TosH2iol	OSC1↑ (Q2 cycle) to Port input invalid (I/O in hold time)	100	—	—	ns	
19*	TioV2osH	Port input valid to OSC1↑ (I/O in setup time)	0	—	—	ns	
20*	TioR	Port output rise time	PIC16C63/65A	10	40	ns	
			PIC16LC63/65A	—	80	ns	
21*	TioF	Port output fall time	PIC16C63/65A	10	40	ns	
			PIC16LC63/65A	—	80	ns	
22††*	Tinp	INT pin high or low time	Tcy	—	—	ns	
23††*	Trbp	RB7:RB4 change INT high or low time	Tcy	—	—	ns	

\* These parameters are characterized but not tested.

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

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

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

Applicable Devices	61	62	62A	R62	63	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

FIGURE 24-4: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER AND POWER-UP TIMER TIMING

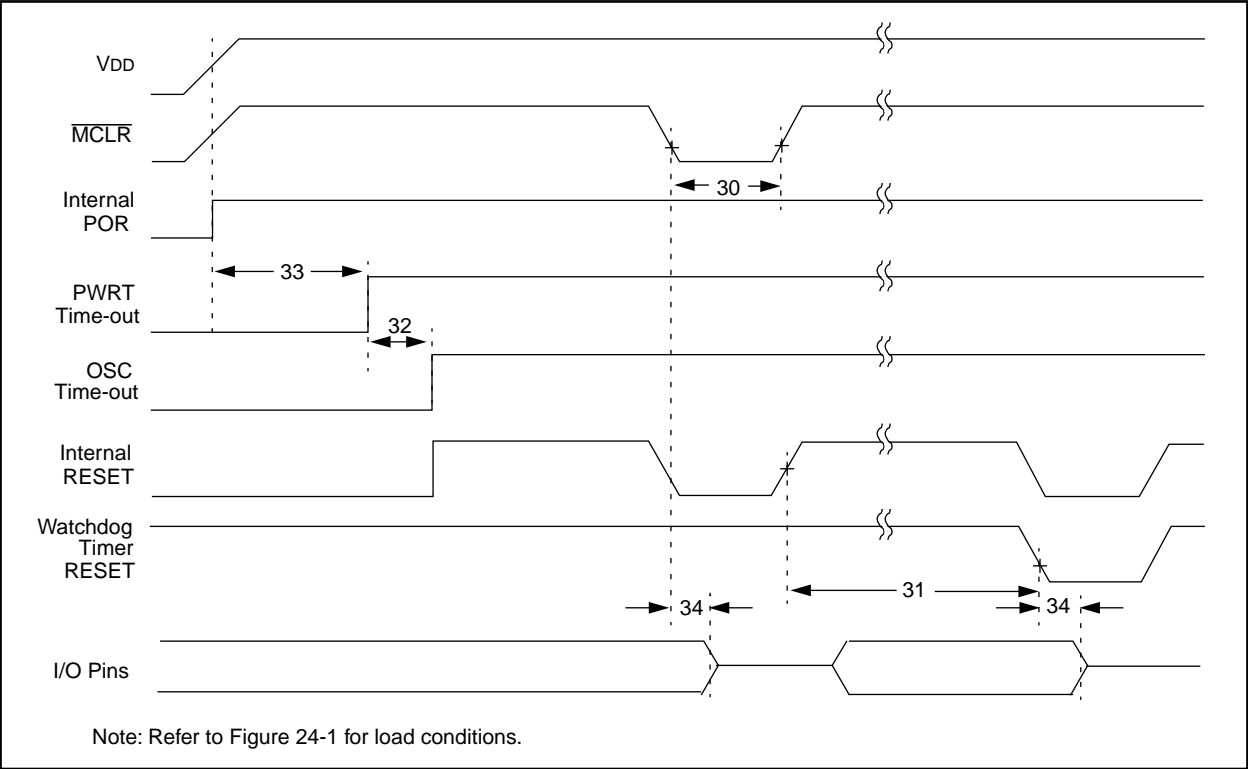


FIGURE 24-5: BROWN-OUT RESET TIMING

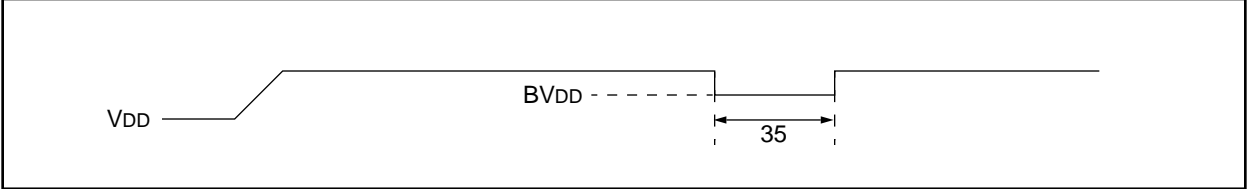


TABLE 24-4: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER, POWER-UP TIMER, AND BROWN-OUT RESET REQUIREMENTS

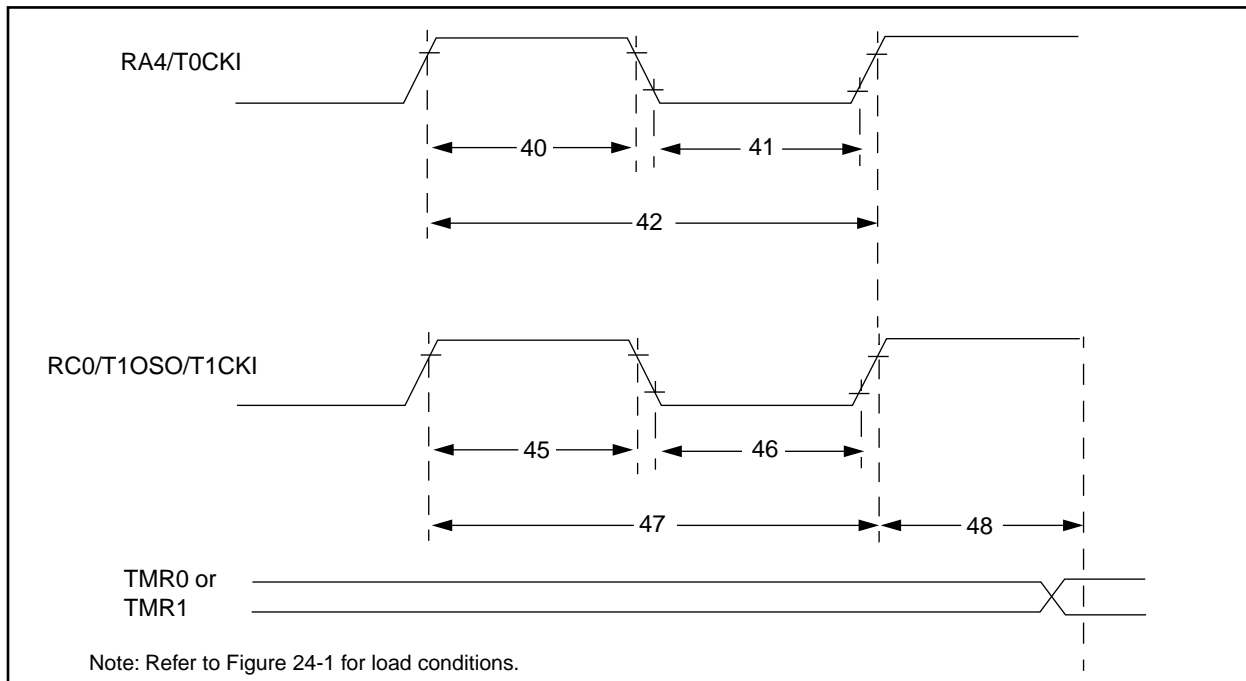
Parameter No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
30	Tmcl	MCLR Pulse Width (low)	2	—	—	μs	VDD = 5V, -40°C to +125°C
31*	Twdt	Watchdog Timer Time-out Period (No Prescaler)	7	18	33	ms	VDD = 5V, -40°C to +125°C
32	Tost	Oscillation Start-up Timer Period	—	1024 TOSC	—	—	TOSC = OSC1 period
33*	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 64 64A R64 65 65A

**FIGURE 24-6: TIMER0 AND TIMER1 CLOCK TIMINGS**



**TABLE 24-5: TIMER0 AND TIMER1 CLOCK REQUIREMENTS**

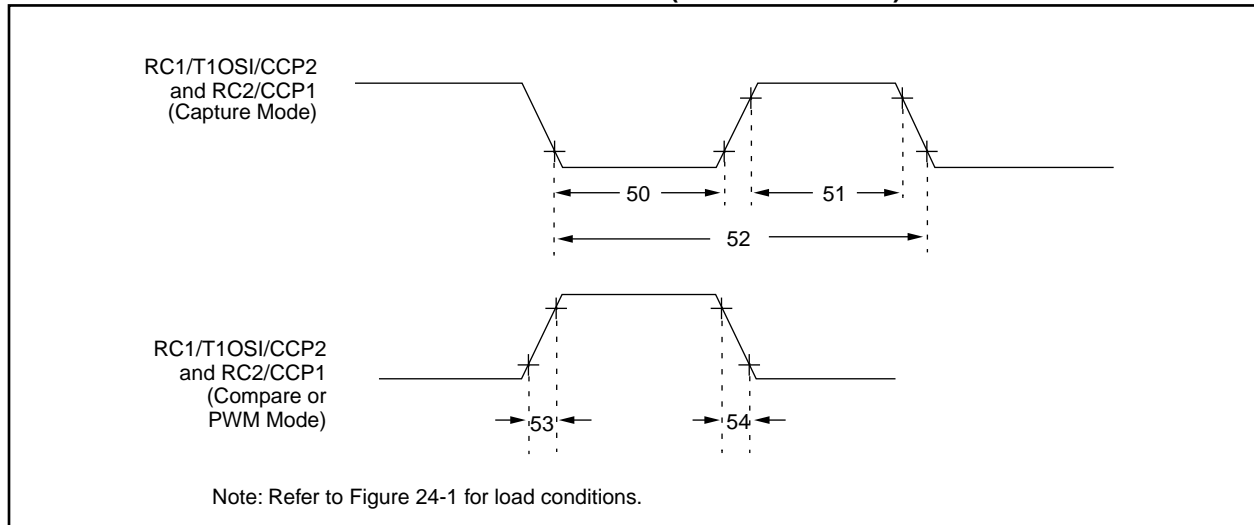
Parameter No.	Sym	Characteristic		Min	Typ†	Max	Units	Conditions
40*	Tt0H	T0CKI High Pulse Width	No Prescaler	$0.5T_{CY} + 20$	—	—	ns	
			With Prescaler	10	—	—	ns	
41*	Tt0L	T0CKI Low Pulse Width	No Prescaler	$0.5T_{CY} + 20$	—	—	ns	
			With Prescaler	10	—	—	ns	
42*	Tt0P	T0CKI Period		$\frac{T_{CY} + 40}{N}$	—	—	ns	N = prescale value (1, 2, 4, ..., 256)
45*	Tt1H	T1CKI High Time	Synchronous, No Prescaler	$0.5T_{CY} + 20$	—	—	ns	
			Synchronous, PIC16C63/65A	10	—	—	ns	
			With Prescaler, PIC16LC63/65A	20	—	—	ns	
			Asynchronous	$2T_{CY}$	—	—	ns	
46*	Tt1L	T1CKI Low Time	Synchronous, No Prescaler	$0.5T_{CY} + 20$	—	—	ns	
			Synchronous, PIC16C63/65A	10	—	—	ns	
			With Prescaler, PIC16LC63/65A	20	—	—	ns	
			Asynchronous	$2T_{CY}$	—	—	ns	
47*	Tt1P	T1CKI input period	Synchronous	$\frac{T_{CY} + 40}{N}$	—	—	ns	N = prescale value (1, 2, 4, 8)
			Asynchronous	$4T_{CY}$	—	—	ns	
	Ft1	Timer1 oscillator input frequency range (oscillator enabled by setting the T1OSCEN bit)		DC	—	200	kHz	
48	TCKEztmr1	Delay from external clock edge to timer increment		$2T_{OSC}$	—	$7T_{OSC}$	—	

\* These parameters are characterized but not tested.

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

Applicable Devices	61	62	62A	R62	63	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

**FIGURE 24-7: CAPTURE/COMPARE/PWM TIMINGS (CCP1 AND CCP2)**



**TABLE 24-6: CAPTURE/COMPARE/PWM REQUIREMENTS (CCP1 AND CCP2)**

Parameter No.	Sym	Characteristic			Min	Typ†	Max	Units	Conditions
50*	TccL	CCP1 and CCP2 input low time	No Prescaler		0.5Tcy + 20	—	—	ns	
			With Prescaler	PIC16C63/65A	10	—	—	ns	
				PIC16LC63/65A	20	—	—	ns	
51*	TccH	CCP1 and CCP2 input high time	No Prescaler		0.5Tcy + 20	—	—	ns	
			With Prescaler	PIC16C63/65A	10	—	—	ns	
				PIC16LC63/65A	20	—	—	ns	
52*	TccP	CCP1 and CCP2 input period			$\frac{3Tcy + 40}{N}$	—	—	ns	N = prescale value (1,4, or 16)
53*	TccR	CCP1 and CCP2 output rise time			—	10	25	ns	
54*	TccF	CCP1 and CCP2 output fall time	PIC16C63/65A		—	10	25	ns	
			PIC16LC63/65A		—	—	45	ns	

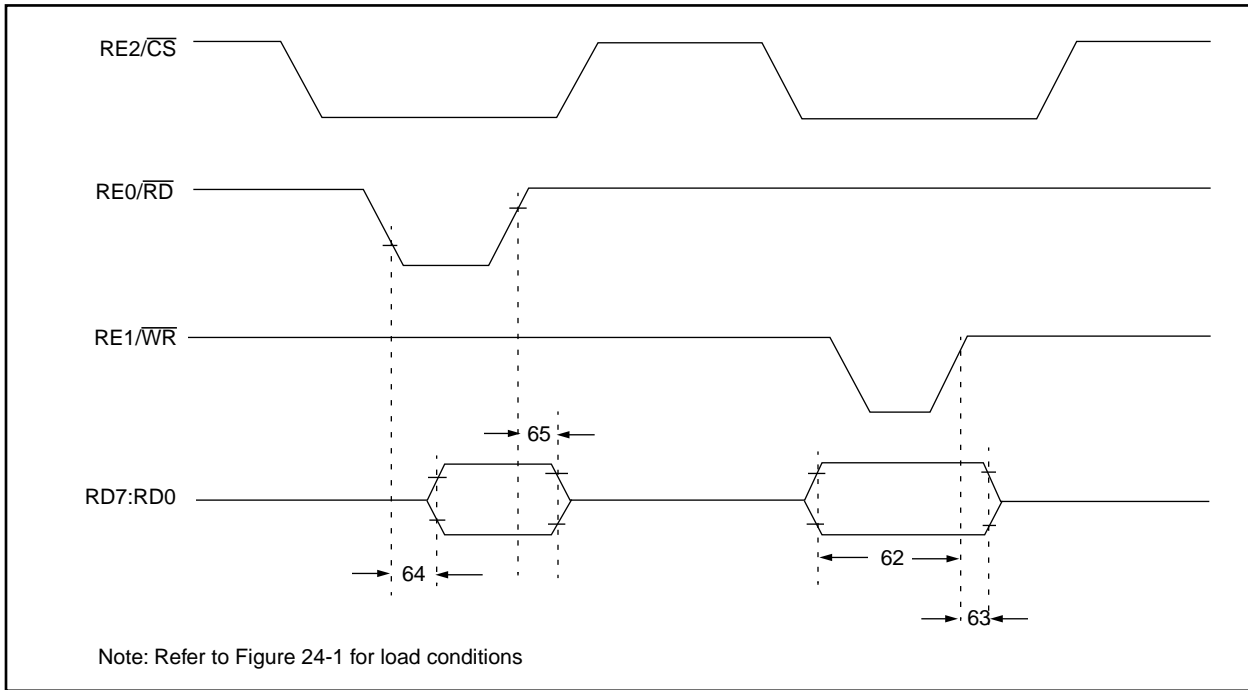
\* These parameters are characterized but not tested.

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

# PIC16C6X

Applicable Devices 61 62 62A R62 63 64 64A R64 65 65A

**FIGURE 24-8: PARALLEL SLAVE PORT TIMING FOR THE PIC16C65A ONLY**



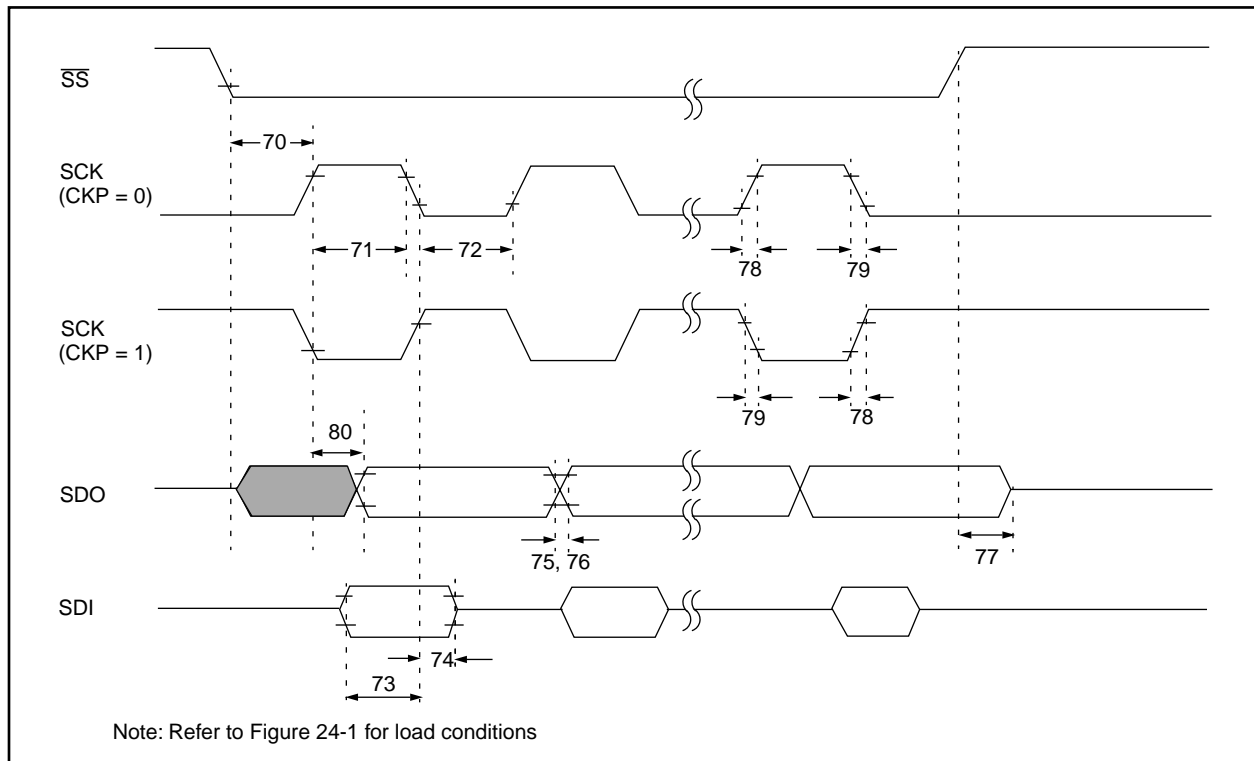
**TABLE 24-7: PARALLEL SLAVE PORT REQUIREMENTS FOR THE PIC16C65A ONLY**

Parameter No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
62*	TdtV2wrH	Data in valid before $\overline{WR}\uparrow$ or $\overline{CS}\uparrow$ (setup time)	20 25	— —	— —	ns ns	 Automotive Range Only
63*	TwrH2dtI	$\overline{WR}\uparrow$ or $\overline{CS}\uparrow$ to data-in invalid (hold time)	PIC16C65A: 20 PIC16LC65A: 35	— —	— —	ns ns	
64	TrdL2dtV	$\overline{RD}\downarrow$ and $\overline{CS}\downarrow$ to data-out valid	— —	— —	80 90	ns ns	 Automotive Range Only
65*	TrdH2dtI	$\overline{RD}\uparrow$ or $\overline{CS}\uparrow$ to data-out invalid	10	—	30	ns	

\* Characterized but not tested

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

**FIGURE 24-9: SPI MODE TIMING**



**TABLE 24-8: SPI MODE REQUIREMENTS**

Parameter No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
70*	TssL2scH, TssL2scL	$\overline{SS}\downarrow$ to SCK $\downarrow$ or SCK $\uparrow$ input	Tcy	—	—	ns	
71*	TscH	SCK input high time (slave mode)	Tcy + 20	—	—	ns	
72*	TscL	SCK input low time (slave mode)	Tcy + 20	—	—	ns	
73*	TdiV2scH, TdiV2scL	Setup time of SDI data input to SCK edge	100	—	—	ns	
74*	Tsch2diL, TscL2diL	Hold time of SDI data input to SCK edge	100	—	—	ns	
75*	TdoR	SDO data output rise time	—	10	25	ns	
76*	TdoF	SDO data output fall time	—	10	25	ns	
77*	TssH2doZ	$\overline{SS}\downarrow$ to SDO output hi-impedance	10	—	50	ns	
78*	TscR	SCK output rise time (master mode)	—	10	25	ns	
79*	TscF	SCK output fall time (master mode)	—	10	25	ns	
80*	Tsch2doV, TscL2doV	SDO data output valid after SCK edge	—	—	50	ns	

\* 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	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

FIGURE 24-10: I<sup>2</sup>C BUS START/STOP BITS TIMING

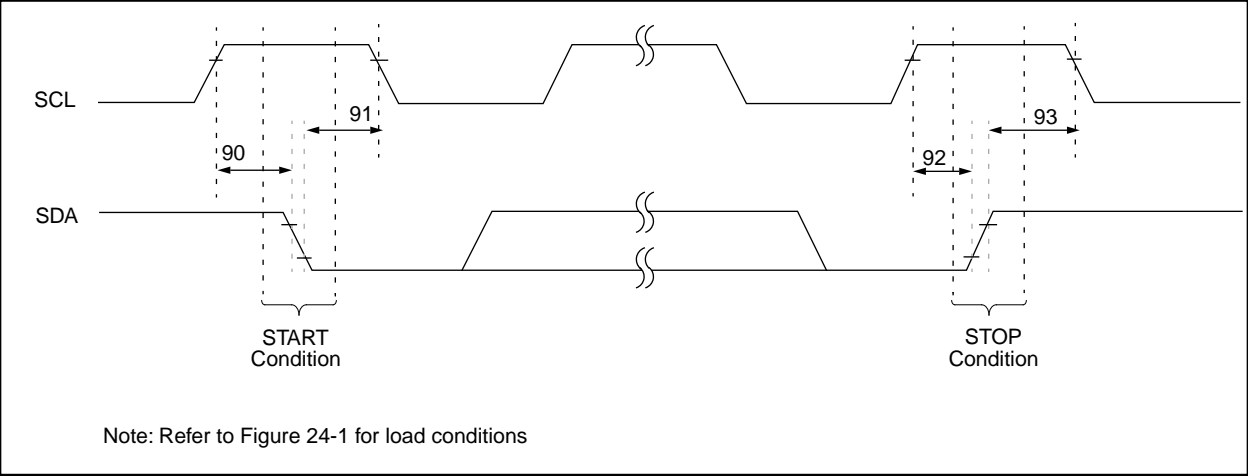


TABLE 24-9: I<sup>2</sup>C BUS START/STOP BITS REQUIREMENTS

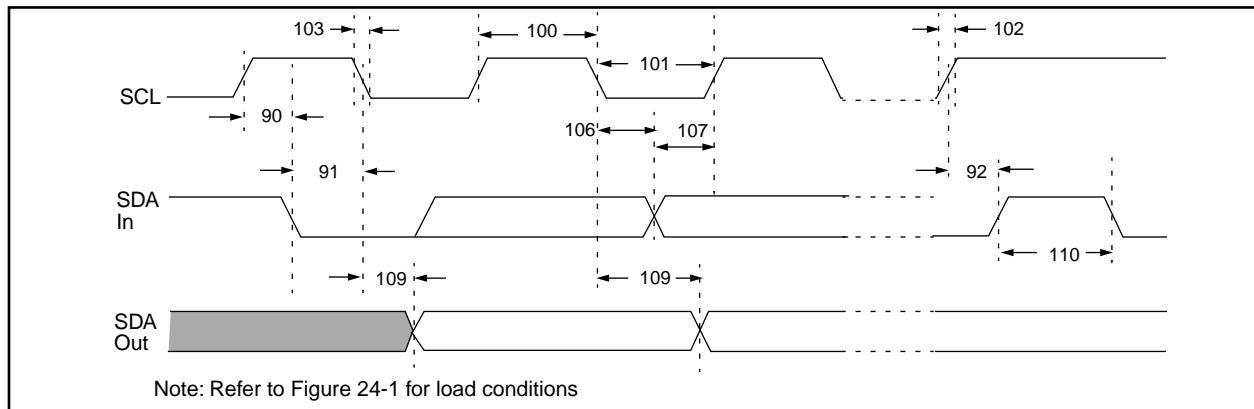
Parameter No.	Sym	Characteristic		Min	Typ	Max	Units	Conditions
90*	TSU:STA	START condition	100 kHz mode	4700	—	—	ns	Only relevant for repeated START condition
		Setup time	400 kHz mode	600	—	—		
91*	THD:STA	START condition	100 kHz mode	4000	—	—	ns	After this period the first clock pulse is generated
		Hold time	400 kHz mode	600	—	—		
92*	TSU:STO	STOP condition	100 kHz mode	4700	—	—	ns	
		Setup time	400 kHz mode	600	—	—		
93	THD:STO	STOP condition	100 kHz mode	4000	—	—	ns	
		Hold time	400 kHz mode	600	—	—		

\*Characterized but not tested



Applicable Devices	61	62	62A	R62	63	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

**FIGURE 24-11: I<sup>2</sup>C BUS DATA TIMING**



**TABLE 24-10: I<sup>2</sup>C BUS DATA REQUIREMENTS**

Parameter No.	Sym	Characteristic		Min	Max	Units	Conditions
100*	THIGH	Clock high time	100 kHz mode	4.0	—	μs	PIC16C65A must operate at a minimum of 1.5 MHz
			400 kHz mode	0.6	—	μs	PIC16C65A must operate at a minimum of 10 MHz
			SSP Module	1.5TCY	—		
101*	TLOW	Clock low time	100 kHz mode	4.7	—	μs	PIC16C65A must operate at a minimum of 1.5 MHz
			400 kHz mode	1.3	—	μs	PIC16C65A must operate at a minimum of 10 MHz
			SSP Module	1.5TCY	—		
102*	Tr	SDA and SCL rise time	100 kHz mode	—	1000	ns	
			400 kHz mode	20 + 0.1Cb	300	ns	Cb is specified to be from 10-400 pF
103*	Tf	SDA and SCL fall time	100 kHz mode	—	300	ns	
			400 kHz mode	20 + 0.1Cb	300	ns	Cb is specified to be from 10-400 pF
90*	TSU:STA	START condition setup time	100 kHz mode	4.7	—	μs	Only relevant for repeated START condition
			400 kHz mode	0.6	—	μs	
91*	THD:STA	START condition hold time	100 kHz mode	4.0	—	μs	After this period the first clock pulse is generated
			400 kHz mode	0.6	—	μs	
106*	THD:DAT	Data input hold time	100 kHz mode	0	—	ns	At 125°C tested to 20 ns min.
			400 kHz mode	0	0.9	μs	
107*	TSU:DAT	Data input setup time	100 kHz mode	250	—	ns	Note 2
			400 kHz mode	100	—	ns	
92*	TSU:STO	STOP condition setup time	100 kHz mode	4.7	—	μs	
			400 kHz mode	0.6	—	μs	
109*	TAA	Output valid from clock	100 kHz mode	—	3500	ns	Note 1
			400 kHz mode	—	—	ns	
110*	TBUF	Bus free time	100 kHz mode	4.7	—	μs	Time the bus must be free before a new transmission can start
			400 kHz mode	1.3	—	μs	
	Cb	Bus capacitive loading		—	400	pF	

\* Characterized but not tested

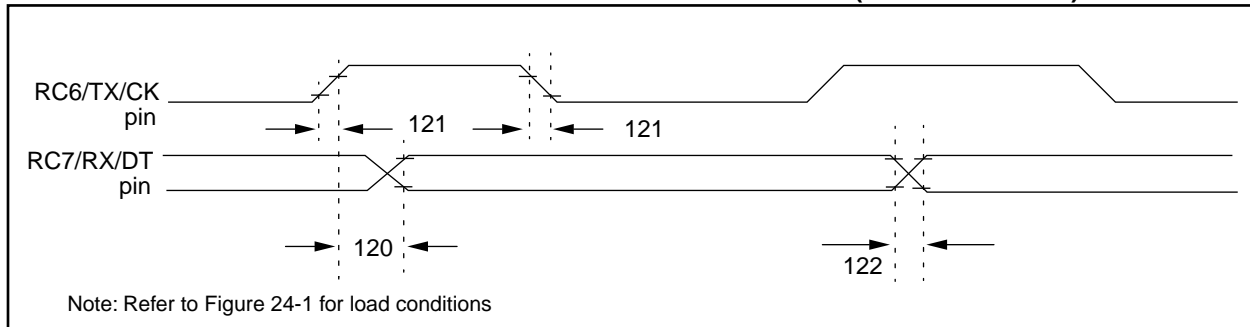
Note 1: As a transmitter, the device must provide this internal minimum delay time to bridge the undefined region (min. 300 ns) of the falling edge of SCL to avoid unintended generation of START or STOP conditions.

Note 2: A fast-mode I<sup>2</sup>C-bus device can be used in a standard-mode I<sup>2</sup>C-bus system, but the requirement T<sub>su</sub>: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 T<sub>r</sub> max.+t<sub>su</sub>:DAT = 1000 + 250 = 1250 ns (according to the standard-mode I<sup>2</sup>C bus specification) before the SCL line is released.

# PIC16C6X

Applicable Devices 61 62 62A R62 63 64 64A R64 65 65A

**FIGURE 24-12: USART MODULE: SYNCHRONOUS TRANSMISSION (MASTER/SLAVE) TIMING**



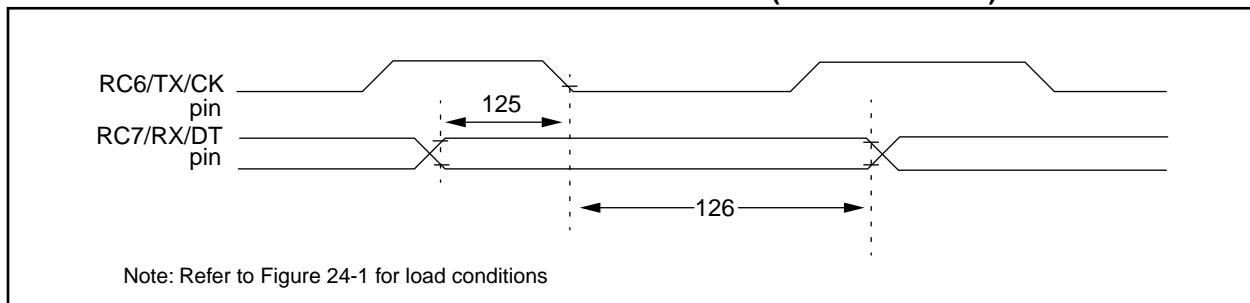
**TABLE 24-11: SERIAL PORT SYNCHRONOUS TRANSMISSION REQUIREMENTS**

Parameter No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
120*	tckH2dtV	<u>SYNC XMIT (MASTER &amp; SLAVE)</u> Clock high to data out valid	—	—	50	ns	
					100	ns	
121*	tckrf	Clock out rise time and fall time (Master Mode)	—	—	25	ns	
			—	—	50	ns	
122*	tdtrf	Data out rise time and fall time	—	—	25	ns	
			—	—	50	ns	

\* Characterized but not tested

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

**FIGURE 24-13: USART MODULE: SYNCHRONOUS RECEIVE (MASTER/SLAVE) TIMING**



**TABLE 24-12: SERIAL PORT SYNCHRONOUS RECEIVE REQUIREMENTS**

Parameter No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
125*	tdtV2ckl	<u>SYNC RCV (MASTER &amp; SLAVE)</u> Data hold before CK ↓ (DT hold time)	15	—	—	ns	
126*	tckL2dtl	Data hold after CK ↓ (DT hold time)	15	—	—	ns	

\* 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.

Applicable Devices	61	62	62A	R62	63	64	64A	R64	65	65A
--------------------	----	----	-----	-----	----	----	-----	-----	----	-----

25.0 DC AND AC CHARACTERISTICS GRAPHS AND TABLES FOR PIC16C63/65A

NOT AVAILABLE AT THIS TIME

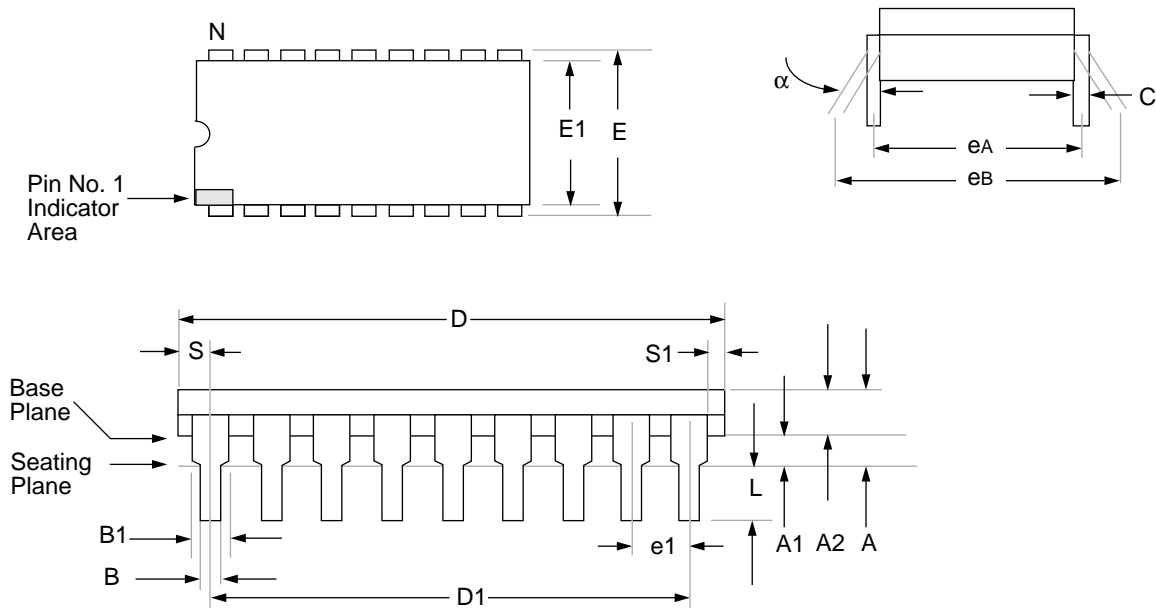
# PIC16C6X

---

NOTES:

## 26.0 PACKAGING INFORMATION

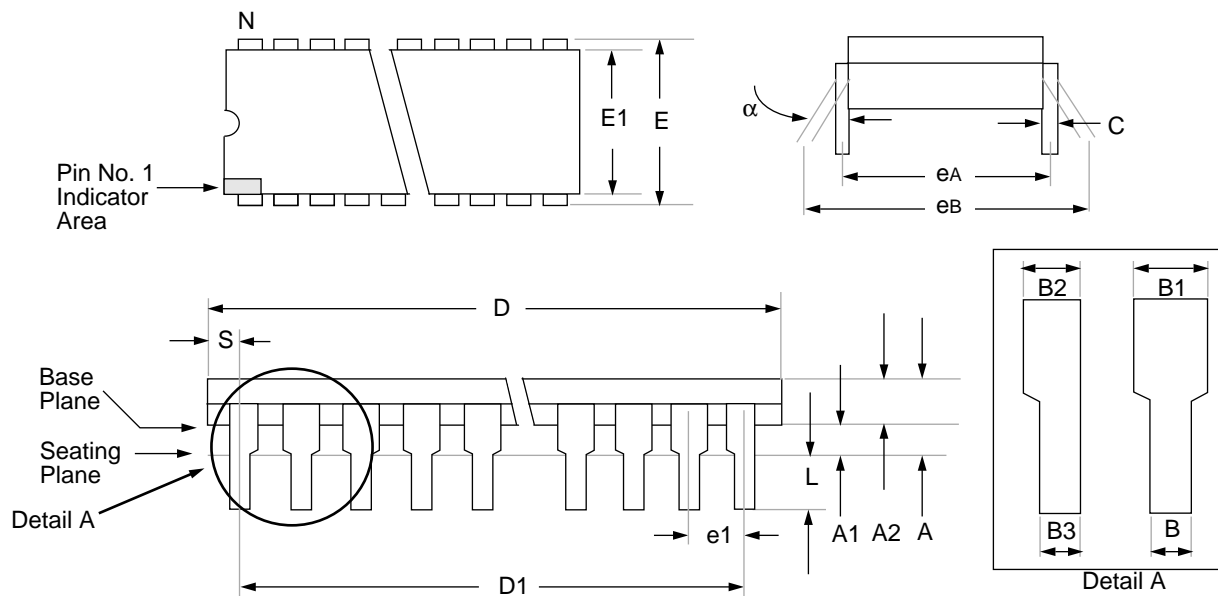
### 26.1 18-Lead Plastic Dual In-line (300 mil)



Package Group: Plastic Dual In-Line (PLA)						
Symbol	Millimeters			Inches		
	Min	Max	Notes	Min	Max	Notes
$\alpha$	0°	10°		0°	10°	
A	—	4.064		—	0.160	
A1	0.381	—		0.015	—	
A2	3.048	3.810		0.120	0.150	
B	0.355	0.559		0.014	0.022	
B1	1.524	1.524	Reference	0.060	0.060	Reference
C	0.203	0.381	Typical	0.008	0.015	Typical
D	22.479	23.495		0.885	0.925	
D1	20.320	20.320	Reference	0.800	0.800	Reference
E	7.620	8.255		0.300	0.325	
E1	6.096	7.112		0.240	0.280	
e1	2.489	2.591	Typical	0.098	0.102	Typical
eA	7.620	7.620	Reference	0.300	0.300	Reference
eB	7.874	9.906		0.310	0.390	
L	3.048	3.556		0.120	0.140	
N	18	18		18	18	
S	0.889	—		0.035	—	
S1	0.127	—		0.005	—	

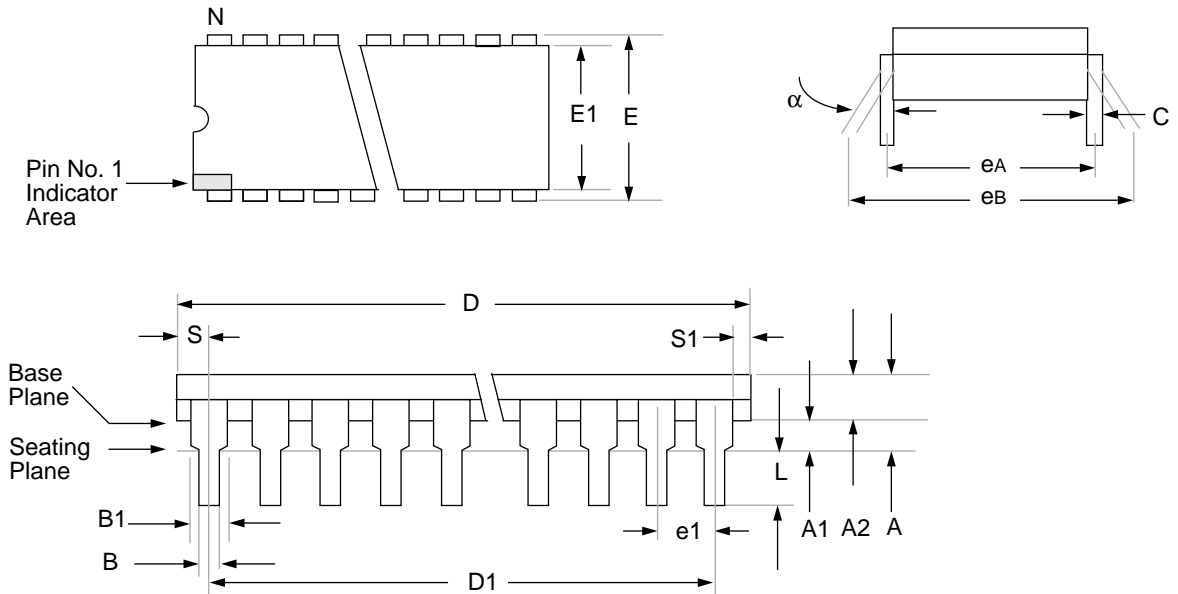
# PIC16C6X

## 26.2 28-Lead Plastic Dual In-line (300 mil)



Package Group: Plastic Dual In-Line (PLA)						
Symbol	Millimeters			Inches		
	Min	Max	Notes	Min	Max	Notes
$\alpha$	0°	10°		0°	10°	
A	3.632	4.572		0.143	0.180	
A1	0.381	—		0.015	—	
A2	3.175	3.556		0.125	0.140	
B	0.406	0.559		0.016	0.022	
B1	1.016	1.651	Typical	0.040	0.065	Typical
B2	0.762	1.016	4 places	0.030	0.040	4 places
B3	0.203	0.508	4 places	0.008	0.020	4 places
C	0.203	0.331	Typical	0.008	0.013	Typical
D	34.163	35.179		1.385	1.395	
D1	33.020	33.020	Reference	1.300	1.300	Reference
E	7.874	8.382		0.310	0.330	
E1	7.112	7.493		0.280	0.295	
e1	2.540	2.540	Typical	0.100	0.100	Typical
eA	7.874	7.874	Reference	0.310	0.310	Reference
eB	8.128	9.652		0.320	0.380	
L	3.175	3.683		0.125	0.145	
N	28	28		28	28	
S	0.584	1.220		0.023	0.048	

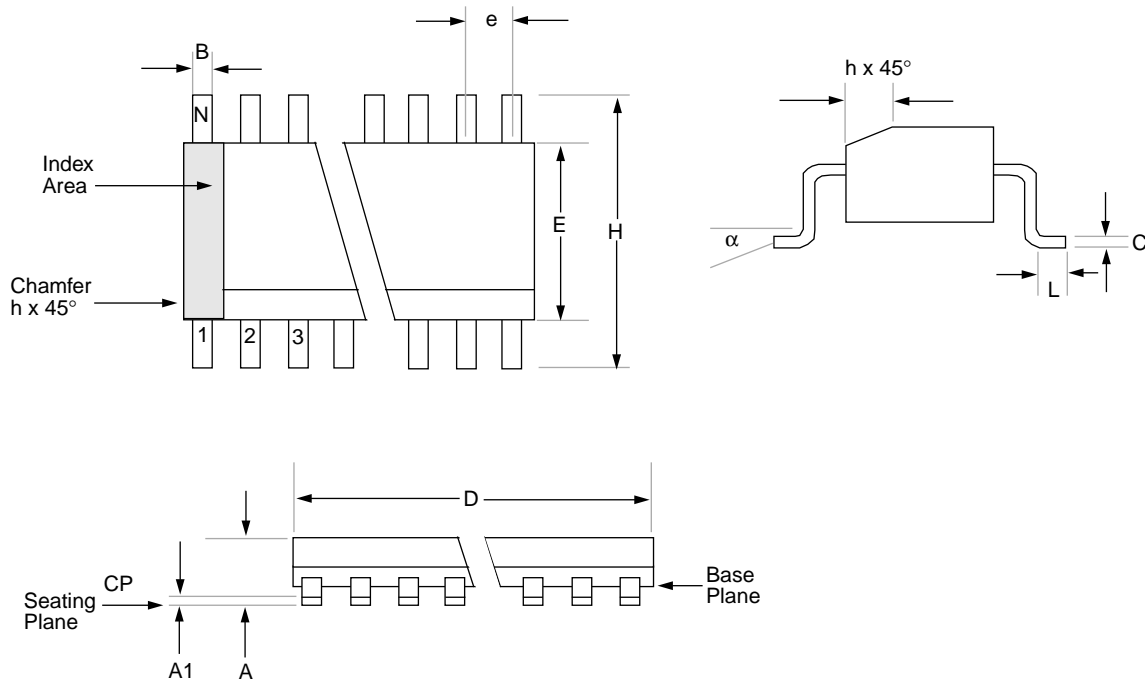
## 26.3 40-Lead Plastic Dual In-line (600 mil)



Package Group: Plastic Dual In-Line (PLA)						
Symbol	Millimeters			Inches		
	Min	Max	Notes	Min	Max	Notes
$\alpha$	0°	10°		0°	10°	
A	—	5.080		—	0.200	
A1	0.381	—		0.015	—	
A2	3.175	4.064		0.125	0.160	
B	0.355	0.559		0.014	0.022	
B1	1.270	1.778	Typical	0.050	0.070	Typical
C	0.203	0.381	Typical	0.008	0.015	Typical
D	51.181	52.197		2.015	2.055	
D1	48.260	48.260	Reference	1.900	1.900	Reference
E	15.240	15.875		0.600	0.625	
E1	13.462	13.970		0.530	0.550	
e1	2.489	2.591	Typical	0.098	0.102	Typical
eA	15.240	15.240	Reference	0.600	0.600	Reference
eB	15.240	17.272		0.600	0.680	
L	2.921	3.683		0.115	0.145	
N	40	40		40	40	
S	1.270	—		0.050	—	
S1	0.508	—		0.020	—	

# PIC16C6X

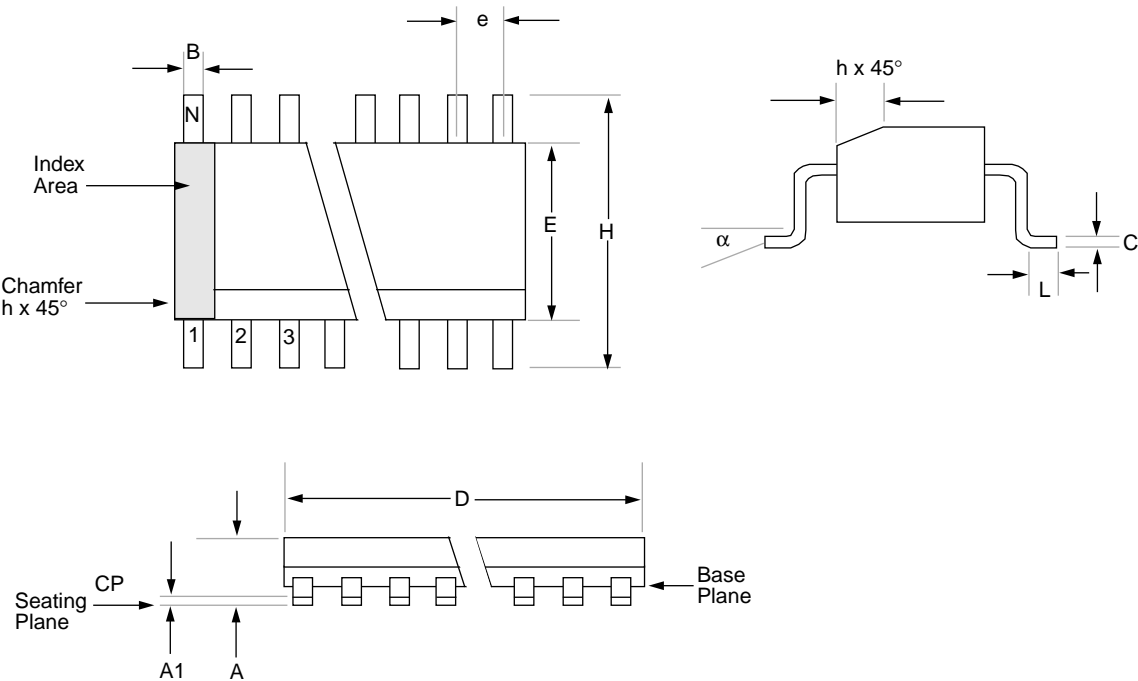
## 26.4 18-Lead Plastic Surface Mount (SOIC - Wide, 300 mil Body)



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



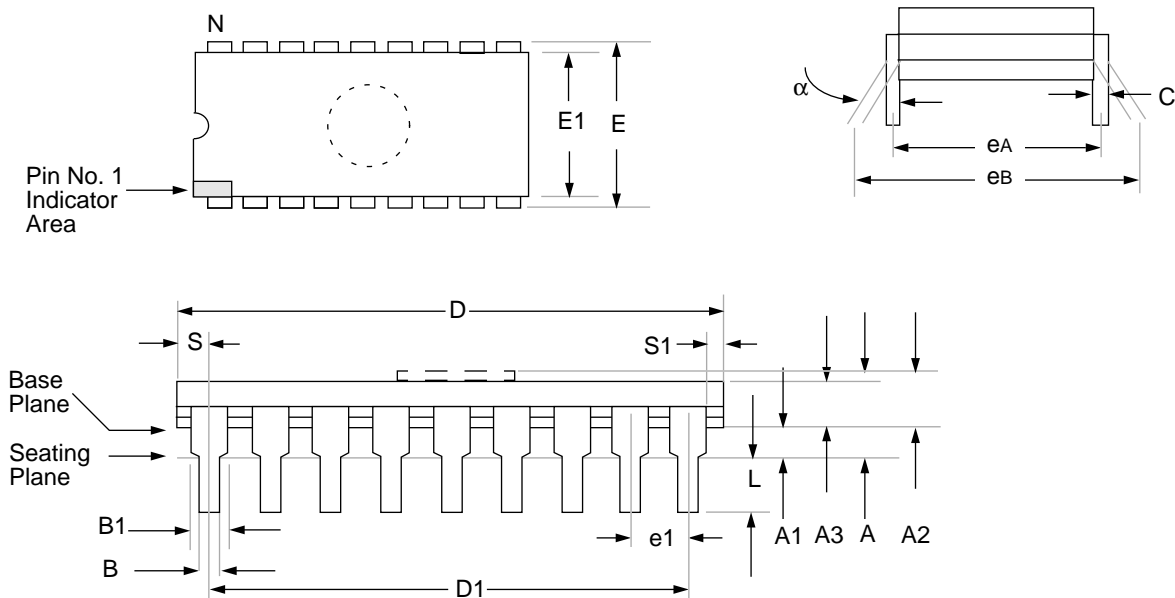
26.5 28-Lead Plastic Surface Mount (SOIC - Wide, 300 mil Body)



Package Group: Plastic SOIC (SO)						
Symbol	Millimeters			Inches		
	Min	Max	Notes	Min	Max	Notes
$\alpha$	0°	8°		0°	8°	
A	2.362	2.642		0.093	0.104	
A1	0.101	0.300		0.004	0.012	
B	0.355	0.483		0.014	0.019	
C	0.241	0.318		0.009	0.013	
D	17.703	18.085		0.697	0.712	
E	7.416	7.595		0.292	0.299	
e	1.270	1.270	Typical	0.050	0.050	Typical
H	10.007	10.643		0.394	0.419	
h	0.381	0.762		0.015	0.030	
L	0.406	1.143		0.016	0.045	
N	28	28		28	28	
CP	—	0.102		—	0.004	

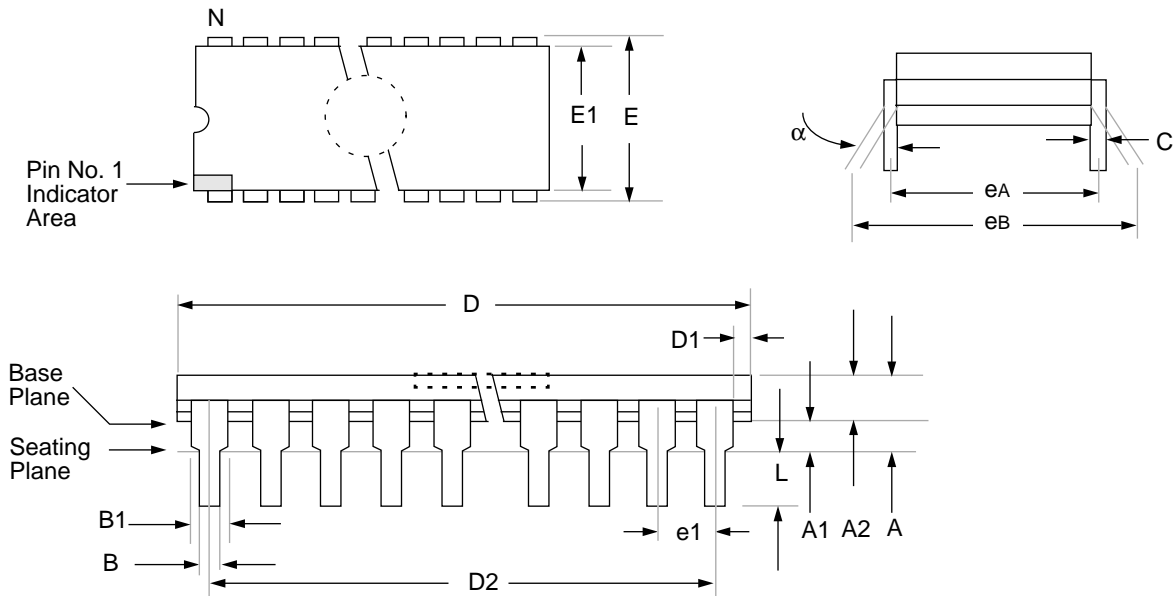
# PIC16C6X

## 26.6 18-Lead Ceramic Cerdip Dual In-line with Window (300 mil)



Package Group: Ceramic Cerdip Dual In-Line (CDP)						
Symbol	Millimeters			Inches		
	Min	Max	Notes	Min	Max	Notes
$\alpha$	0°	10°		0°	10°	
A	—	5.080		—	0.200	
A1	0.381	1.778		0.015	0.070	
A2	3.810	4.699		0.150	0.185	
A3	3.810	4.445		0.150	0.175	
B	0.355	0.585		0.014	0.023	
B1	1.270	1.651	Typical	0.050	0.065	Typical
C	0.203	0.381	Typical	0.008	0.015	Typical
D	22.352	23.622		0.880	0.930	
D1	20.320	20.320	Reference	0.800	0.800	Reference
E	7.620	8.382		0.300	0.330	
E1	5.588	7.874		0.220	0.310	
e1	2.540	2.540	Reference	0.100	0.100	Reference
eA	7.366	8.128	Typical	0.290	0.320	Typical
eB	7.620	10.160		0.300	0.400	
L	3.175	3.810		0.125	0.150	
N	18	18		18	18	
S	0.508	1.397		0.020	0.055	
S1	0.381	1.270		0.015	0.050	

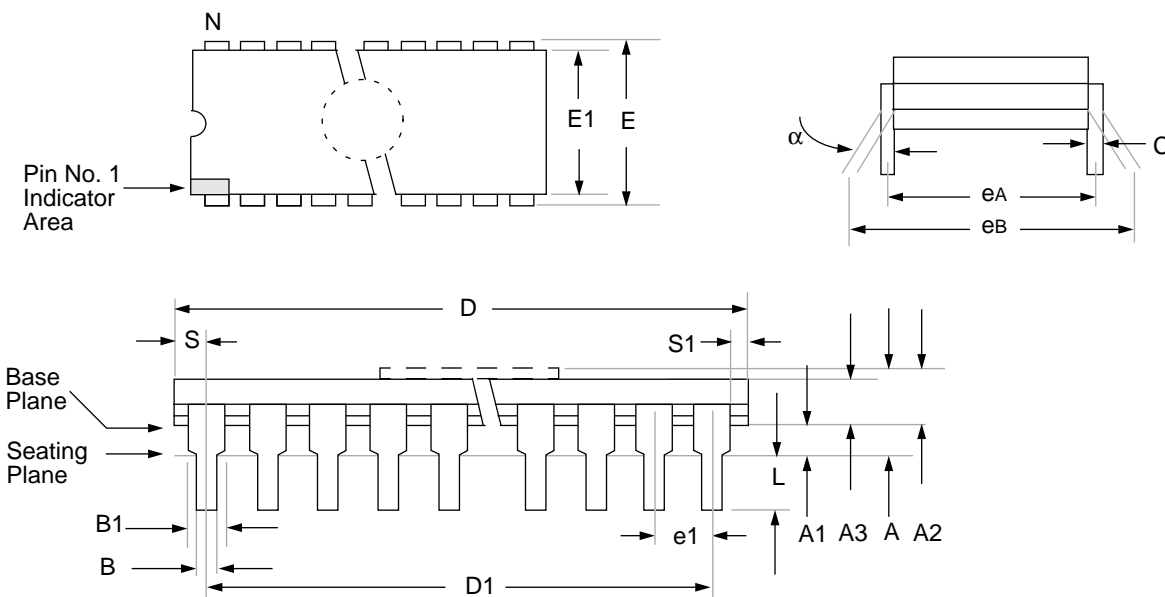
26.7 28-Lead Ceramic Cerdip Dual In-line with Window (300 mil)



Package Group: Ceramic Cerdip Dual In-Line (CDP)						
Symbol	Millimeters			Inches		
	Min	Max	Notes	Min	Max	Notes
$\alpha$	0°	10°		0°	10°	
A	3.30	5.84		.130	0.230	
A1	0.38	—		0.015	—	
A2	2.92	4.95		0.115	0.195	
B	0.35	0.58		0.014	0.023	
B1	1.14	1.78	Typical	0.045	0.070	Typical
C	0.20	0.38	Typical	0.008	0.015	Typical
D	34.54	37.72		1.360	1.485	
D2	32.97	33.07	Reference	1.298	1.302	Reference
E	7.62	8.25		0.300	0.325	
E1	6.10	7.87		0.240	0.310	
e	2.54	2.54	Typical	0.100	0.100	Typical
eA	7.62	7.62	Reference	0.300	0.300	Reference
eB	—	11.43		—	0.450	
L	2.92	5.08		0.115	0.200	
N	28	28		28	28	
D1	0.13	—		0.005	—	

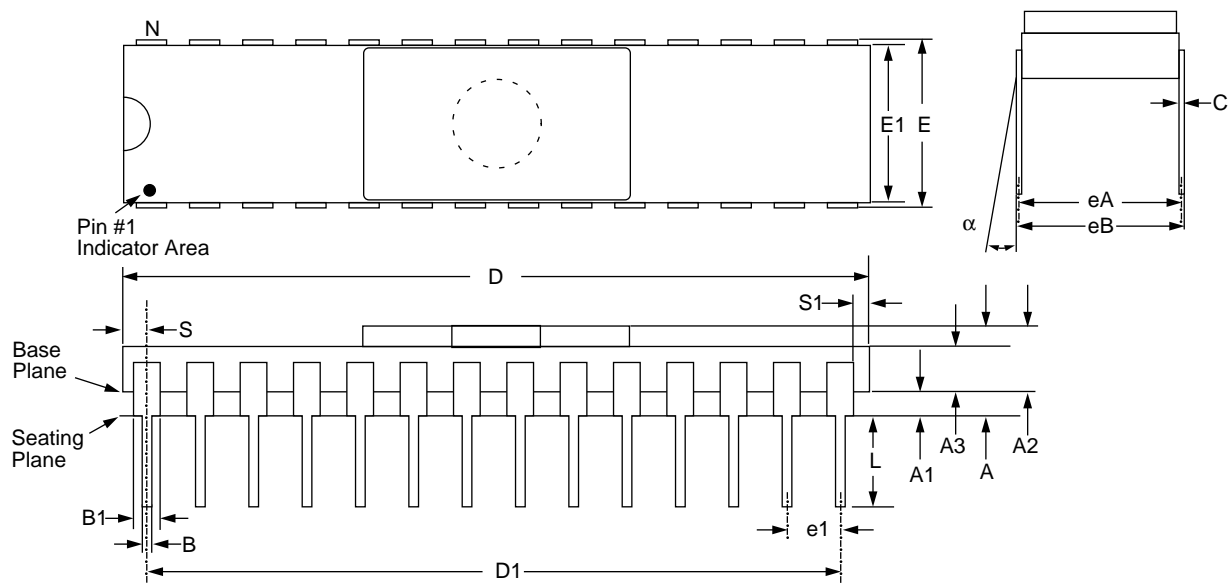
# PIC16C6X

## 26.8 40-Lead Ceramic CERDIP Dual In-line with Window (600 mil)



Package Group: Ceramic CERDIP Dual In-Line (CDP)						
Symbol	Millimeters			Inches		
	Min	Max	Notes	Min	Max	Notes
$\alpha$	0°	10°		0°	10°	
A	4.318	5.715		0.170	0.225	
A1	0.381	1.778		0.015	0.070	
A2	3.810	4.699		0.150	0.185	
A3	3.810	4.445		0.150	0.175	
B	0.355	0.585		0.014	0.023	
B1	1.270	1.651	Typical	0.050	0.065	Typical
C	0.203	0.381	Typical	0.008	0.015	Typical
D	51.435	52.705		2.025	2.075	
D1	48.260	48.260	Reference	1.900	1.900	Reference
E	15.240	15.875		0.600	0.625	
E1	12.954	15.240		0.510	0.600	
e1	2.540	2.540	Reference	0.100	0.100	Reference
eA	14.986	16.002	Typical	0.590	0.630	Typical
eB	15.240	18.034		0.600	0.710	
L	3.175	3.810		0.125	0.150	
N	40	40		40	40	
S	1.016	2.286		0.040	0.090	
S1	0.381	1.778		0.015	0.070	

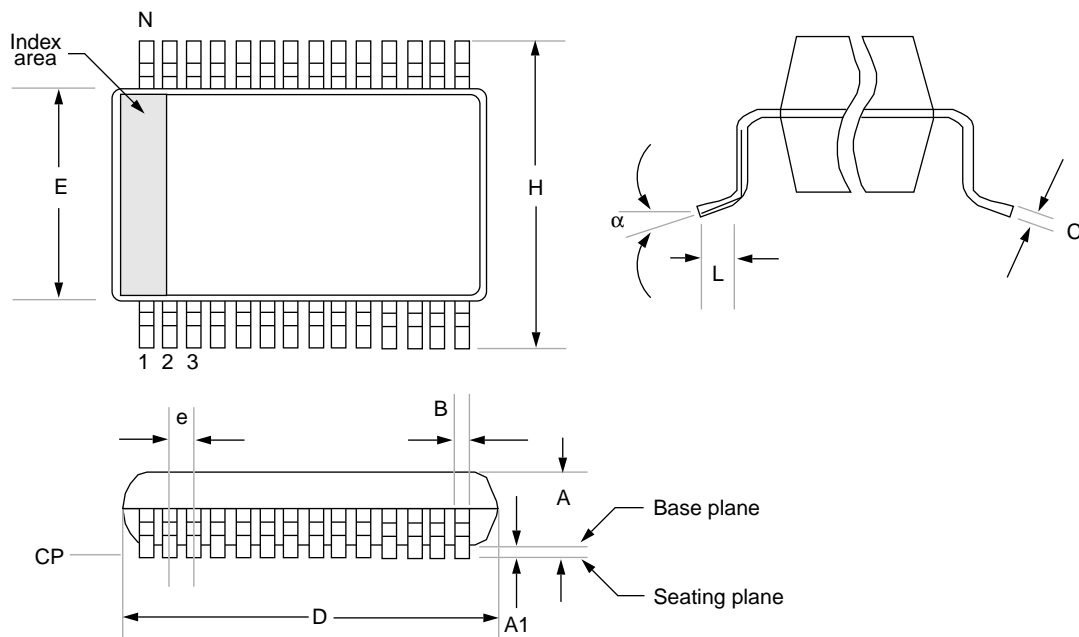
## 26.9 28-Lead Ceramic Side Brazed Dual In-Line with Window (300 mil)



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

# PIC16C6X

## 26.10 28-Lead Plastic Surface Mount (SSOP - 209 mil Body 5.30 mm)

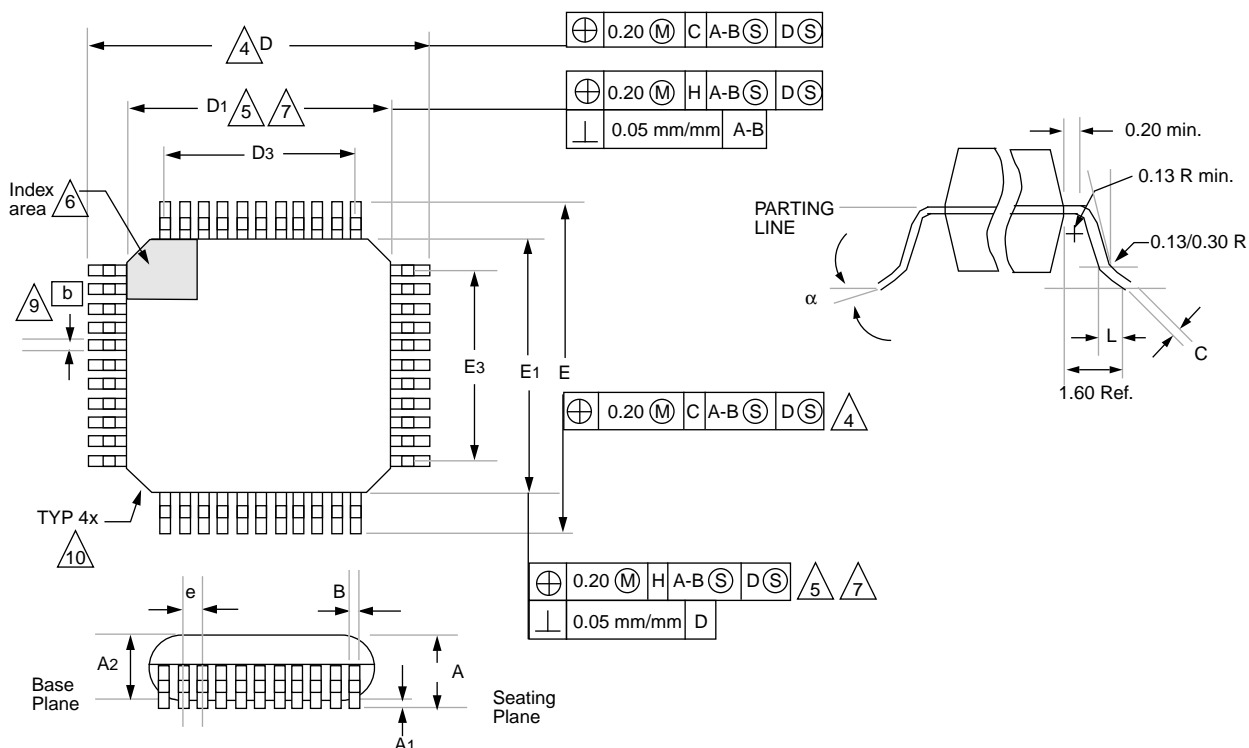


Package Group: Plastic SSOP						
Symbol	Millimeters			Inches		
	Min	Max	Notes	Min	Max	Notes
$\alpha$	0°	8°		0°	8°	
A	1.730	1.990		0.068	0.078	
A1	0.050	0.210		0.002	0.008	
B	0.250	0.380		0.010	0.015	
C	0.130	0.220		0.005	0.009	
D	10.070	10.330		0.396	0.407	
E	5.200	5.380		0.205	0.212	
e	0.650	0.650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



# PIC16C6X

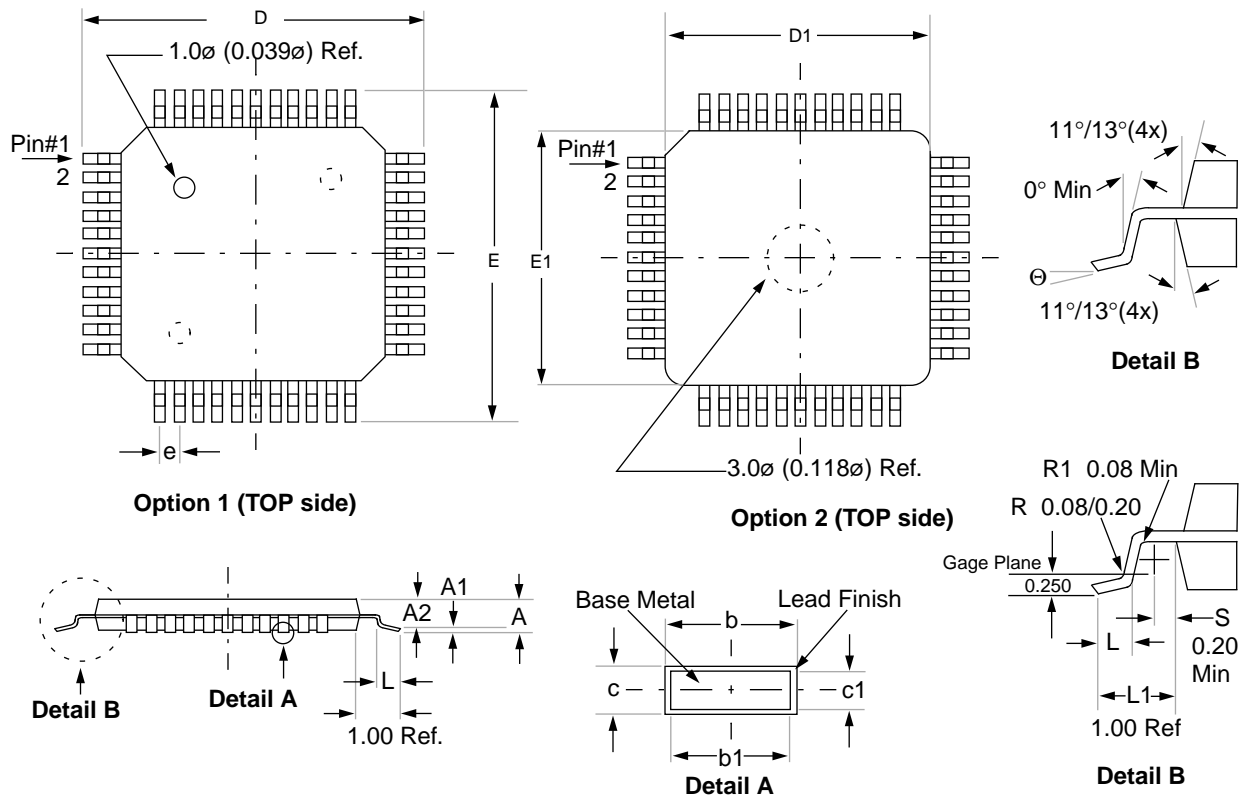
## 26.12 44-Lead Plastic Surface Mount (MQFP 10x10 mm Body 1.6/0.15 mm Lead Form)



Package Group: Plastic MQFP						
Symbol	Millimeters			Inches		
	Min	Max	Notes	Min	Max	Notes
$\alpha$	0°	7°		0°	7°	
A	2.000	2.350		0.078	0.093	
A1	0.050	0.250		0.002	0.010	
A2	1.950	2.100		0.768	0.083	
b	0.300	0.450	Typical	0.011	0.018	Typical
C	0.150	0.180		0.006	0.007	
D	12.950	13.450		0.510	0.530	
D1	9.900	10.100		0.390	0.398	
D3	8.000	8.000	Reference	0.315	0.315	Reference
E	12.950	13.450		0.510	0.530	
E1	9.900	10.100		0.390	0.398	
E3	8.000	8.000	Reference	0.315	0.315	Reference
e	0.800	0.800		0.031	0.032	
L	0.730	1.030		0.028	0.041	
N	44	44		44	44	
CP	0.102	—		0.004	—	



## 26.13 44-Lead Plastic Surface Mount (TQFP 10x10 mm Body 1.0/0.10 mm Lead Form)



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.

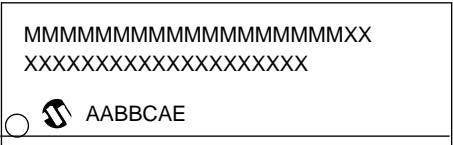
2: Dimension "b" does not include Dambar protrusion, allowable Dambar protrusion shall be 0.08mm (0.003") max.

3: This outline conforms to JEDEC MS-026.

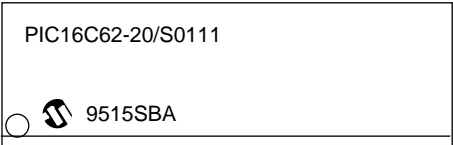


Package Marking Information (Cont'd)

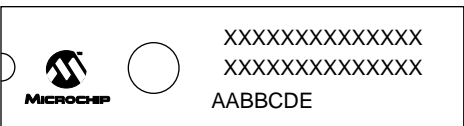
28-Lead SOIC



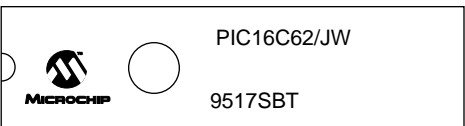
Example



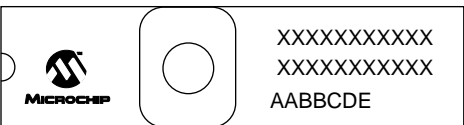
28-Lead CERDIP Skinny Windowed



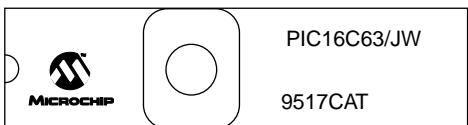
Example



28-Lead Side Brazed Skinny Windowed



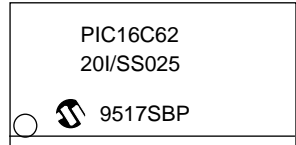
Example



28-Lead SSOP



Example



40-Lead PDIP



Example



Legend:	MM...M	Microchip part number information
	XX...X	Customer specific information*
	AA	Year code (last 2 digits of calendar year)
	BB	Week code (week of January 1 is week '01')
	C	Facility code of the plant at which wafer is manufactured. C = Chandler, Arizona, U.S.A. S = Tempe, Arizona, U.S.A.
	D <sub>1</sub>	Mask revision number for microcontroller
	E	Assembly code of the plant or country of origin in which part was assembled.

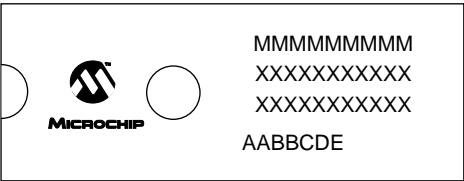
Note: In the event the full Microchip part number cannot be marked on one line, it will be carried over to the next line thus limiting the number of available characters for customer specific information.

\* Standard OTP marking consists of Microchip part number, year code, week code, facility code, mask revision number, and assembly code. For OTP marking beyond this, certain price adders apply. Please check with your Microchip Sales Office. For QTP devices, any special marking adders are included in QTP price.

# PIC16C6X

## Package Marking Information (Cont'd)

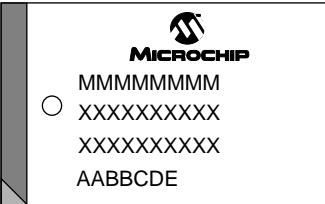
40-Lead CERDIP Windowed



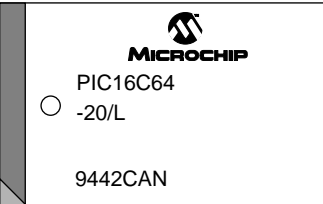
Example



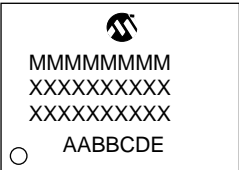
44-Lead PLCC



Example



44-Lead MQFP



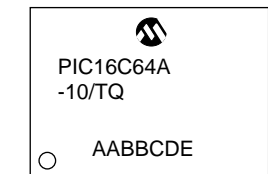
Example



44-Lead TQFP



Example



Legend:	MM...M	Microchip part number information
	XX...X	Customer specific information*
	AA	Year code (last 2 digits of calendar year)
	BB	Week code (week of January 1 is week '01')
	C	Facility code of the plant at which wafer is manufactured. C = Chandler, Arizona, U.S.A. S = Tempe, Arizona, U.S.A.
	D <sub>1</sub>	Mask revision number for microcontroller
	E	Assembly code of the plant or country of origin in which part was assembled.
Note: In the event the full Microchip part number cannot be marked on one line, it will be carried over to the next line thus limiting the number of available characters for customer specific information.		

\* Standard OTP marking consists of Microchip part number, year code, week code, facility code, mask revision number, and assembly code. For OTP marking beyond this, certain price adders apply. Please check with your Microchip Sales Office. For QTP devices, any special marking adders are included in QTP price.

## APPENDIX A: MODIFICATIONS

The following are the list of modifications over the PIC16C5X microcontroller family:

1. Instruction word length is increased to 14-bits. This allows larger page sizes both in program memory (4K now as opposed to 512 before) and register file (192 bytes now versus 32 bytes before).
2. A PC high latch register (PCLATH) is added to handle program memory paging. PA2, PA1, PA0 bits are removed from STATUS register.
3. Data memory paging is redefined slightly. STATUS register is modified.
4. Four new instructions have been added: RETURN, RETFIE, ADDLW, and SUBLW. Two instructions TRIS and OPTION are being phased out although they are kept for compatibility with PIC16C5X.
5. OPTION and TRIS registers are made addressable.
6. Interrupt capability is added. Interrupt vector is at 0004h.
7. Stack size is increased to 8 deep.
8. Reset vector is changed to 0000h.
9. Reset of all registers is revisited. Five different reset (and wake-up) types are recognized. Registers are reset differently.
10. Wake-up from SLEEP through interrupt is added.
11. Two separate timers, Oscillator Start-up Timer (OST) and Power-up Timer (PWRT), are included for more reliable power-up. These timers are invoked selectively to avoid unnecessary delays on power-up and wake-up.
12. PORTB has weak pull-ups and interrupt on change feature.
13. Timer0 pin is also a port pin (RA4/T0CKI) now.
14. FSR is made a full 8-bit register.
15. "In-circuit programming" is made possible. The user can program PIC16CXX devices using only five pins: VDD, VSS, VPP, RB6 (clock) and RB7 (data in/out).
16. Power Control register (PCON) is added with a Power-on Reset status bit ( $\overline{\text{POR}}$ ). (Not on the PIC16C61).
17. Brown-out Reset has been added to the following devices:  
PIC16C62A/R62/63/64A/R64/65A.

## APPENDIX B: COMPATIBILITY

To convert code written for PIC16C5X to PIC16CXX, the user should take the following steps:

1. Remove any program memory page select operations (PA2, PA1, PA0 bits) for CALL, GOTO.
2. Revisit any computed jump operations (write to PC or add to PC, etc.) to make sure page bits are set properly under the new scheme.
3. Eliminate any data memory page switching. Redefine data variables to reallocate them.
4. Verify all writes to STATUS, OPTION, and FSR registers since these have changed.
5. Change reset vector to 0000h.

# PIC16C6X

---

## APPENDIX C: WHAT'S NEW

The format of this data sheet has been changed to be consistent with other product families. This ensures that important topics are covered across all PIC16/17 families. Here is an overview list of new features:

- Added the following new devices:

- PIC16C62
- PIC16C62A
- PIC16CR62
- PIC16C63
- PIC16C64A
- PIC16CR64
- PIC16C65A

A Brown-out Detect Enable Bit (BODEN) has been added to the Configuration Word register for PIC16C62A, PIC16CR62, PIC16C63, PIC16C64A, PIC16CR64, and the PIC16C65A.

A Brown-out Reset detect bit ( $\overline{\text{BOR}}$ ) has been added to the PCON register (for the devices with brown-out detect circuitry).

A  $\overline{\text{MCLR}}$  filter circuit has been added to minimize the influence of pin state changes to the  $\overline{\text{MCLR}}$  line.

Changed low voltage specification for LC devices on the following parts to 2.5V:

- PIC16C62
- PIC16C62A
- PIC16CR62
- PIC16C63
- PIC16C64A
- PIC16CR64
- PIC16C65A

## APPENDIX D: WHAT'S CHANGED

All product and device family tables have been updated for the latest devices and specifications. Added information on ROM devices.

TX8/9 (TXSTA<6>) has been changed to TX9 - 9-bit Transmit Enable bit.

RC8/9 (RCSTA<6>) has been changed to RX9 - 9-bit Receive Enable bit.

RCD8 (RCSTA<0>) has been changed to RX9D.

TXD8 (TXSTA<0>) has been changed to TX9D.

## APPENDIX E: PIC16/17 MICROCONTROLLERS

TABLE E-1: PIC16C5X FAMILY OF DEVICES

	Clock		Memory		Peripherals		Features		
	Maximum Frequency of Operation (MHz)	Program Memory (words)	RAM Data Memory (bytes)		Timer Module(s)	I/O Pins	Voltage Range (Volts)	Number of Instructions	
			ROM	EPROM					
PIC16C54	20	512	—	25	TMR0	12	2.5-6.25	33	18-pin DIP, SOIC; 20-pin SSOP
PIC16C54A	20	512	—	25	TMR0	12	2.5-6.25	33	18-pin DIP, SOIC; 20-pin SSOP
PIC16CR54 <sup>(2)</sup>	20	—	512	25	TMR0	12	2.0-6.25	33	18-pin DIP, SOIC; 20-pin SSOP
PIC16CR54A	20	—	512	25	TMR0	12	2.0-6.25	33	18-pin DIP, SOIC; 20-pin SSOP
PIC16CR54B <sup>(1)</sup>	20	—	512	25	TMR0	12	2.5-6.25	33	18-pin DIP, SOIC; 20-pin SSOP
PIC16C55	20	512	—	24	TMR0	20	2.5-6.25	33	28-pin DIP, SOIC, SSOP
PIC16C56	20	1K	—	25	TMR0	12	2.5-6.25	33	18-pin DIP, SOIC; 20-pin SSOP
PIC16CR56 <sup>(1)</sup>	20	—	1K	25	TMR0	12	2.5-6.25	33	18-pin DIP, SOIC; 20-pin SSOP
PIC16C57	20	2K	—	72	TMR0	20	2.5-6.25	33	28-pin DIP, SOIC, SSOP
PIC16CR57A <sup>(2)</sup>	20	—	2K	72	TMR0	20	2.5-6.25	33	28-pin DIP, SOIC, SSOP
PIC16CR57B	20	—	2K	72	TMR0	20	2.5-6.25	33	28-pin DIP, SOIC, SSOP
PIC16C58A	20	2K	—	73	TMR0	12	2.5-6.25	33	18-pin DIP, SOIC; 20-pin SSOP
PIC16CR58A	20	—	2K	73	TMR0	12	2.5-6.25	33	18-pin DIP, SOIC; 20-pin SSOP
PIC16CR58B <sup>(1)</sup>	20	—	2K	73	TMR0	12	2.5-6.25	33	18-pin DIP, SOIC; 20-pin SSOP

All PIC16/17 Family devices have Power-On Reset, selectable Watchdog Timer, selectable code protect and high I/O current capability.

Note 1: Please contact your local sales office for availability of these devices.

2: Not recommended for new designs.

# PIC16C6X

TABLE E-2: PIC16C62X FAMILY OF DEVICES

Clock			Memory			Peripherals			Features		
PIC16C620	20	512	80	TMR0	2	Yes	4	13	3.0-6.0	Yes	Yes
PIC16C621	20	1K	80	TMR0	2	Yes	4	13	3.0-6.0	Yes	Yes
PIC16C622	20	2K	128	TMR0	2	Yes	4	13	3.0-6.0	Yes	Yes
Maximum Frequency of Operation (MHz)			Data Memory (bytes)			Timer Module(s)			Comparator(s)		
Program Memory			Internal Reference Voltage			Interrupt Sources			I/O Pins		
EPROM			Voltage Range (Volts)			In-Circuit Serial Programming			Brown-out Reset		
									Packages		

All PIC16/17 Family devices have Power-on Reset, selectable Watchdog Timer, selectable code protect and high I/O current capability.  
All PIC16CXX Family devices use serial programming with clock pin RB6 and data pin RB7.



TABLE E-3: PIC16C6X FAMILY OF DEVICES

	Clock		Memory		Peripherals				Features							
	Maximum Frequency of Operation (MHz)		Program Memory		Serial Ports (SPI/I <sup>2</sup> C, USART)				Voltage Range (Volts)							
ROM		Data Memory (bytes)		Timer Module(s)		Parallel Slave Port		Interrupt Sources		In-Circuit Serial Programming		Brown-out Reset		Packages		
EPROM																
PIC16C61	20	1K	—	36	TMR0	—	—	—	3	13	3.0-6.0	Yes	—	18-pin DIP, SOIC		
PIC16C62	20	2K	—	128	TMR0, TMR1, TMR2	1	SPI/I <sup>2</sup> C	—	7	22	3.0-6.0	Yes	—	28-pin SDIP, SOIC, SSOP		
PIC16C62A <sup>(1)</sup>	20	2K	—	128	TMR0, TMR1, TMR2	1	SPI/I <sup>2</sup> C	—	7	22	2.5-6.0	Yes	Yes	28-pin SDIP, SOIC, SSOP		
PIC16CR62 <sup>(1)</sup>	20	—	2K	128	TMR0, TMR1, TMR2	1	SPI/I <sup>2</sup> C	—	7	22	2.5-6.0	Yes	Yes	28-pin SDIP, SOIC, SSOP		
PIC16C63 <sup>(1)</sup>	20	4K	—	192	TMR0, TMR1, TMR2	2	SPI/I <sup>2</sup> C, USART	—	10	22	2.5-6.0	Yes	Yes	28-pin SDIP, SOIC		
PIC16C64	20	2K	—	128	TMR0, TMR1, TMR2	1	SPI/I <sup>2</sup> C	Yes	8	33	3.0-6.0	Yes	—	40-pin DIP; 44-pin PLCC, MQFP		
PIC16C64A <sup>(1)</sup>	20	2K	—	128	TMR0, TMR1, TMR2	1	SPI/I <sup>2</sup> C	Yes	8	33	2.5-6.0	Yes	Yes	40-pin DIP; 44-pin PLCC, MQFP, TQFP		
PIC16CR64 <sup>(1)</sup>	20	—	2K	128	TMR0, TMR1, TMR2	1	SPI/I <sup>2</sup> C	Yes	8	33	2.5-6.0	Yes	Yes	40-pin DIP; 44-pin PLCC, MQFP		
PIC16C65	20	4K	—	192	TMR0, TMR1, TMR2	2	SPI/I <sup>2</sup> C, USART	Yes	11	33	3.0-6.0	Yes	—	40-pin DIP; 44-pin PLCC, MQFP		
PIC16CR65A <sup>(1)</sup>	20	4K	—	192	TMR0, TMR1, TMR2	2	SPI/I <sup>2</sup> C, USART	Yes	11	33	2.5-6.0	Yes	Yes	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 PIC16CXX family devices use serial programming with clock pin RB6 and data pin RB7.

Note 1: Please contact your local sales office for availability of these devices.

# PIC16C6X

TABLE E-4: PIC16C7X FAMILY OF DEVICES

	Clock			Memory			Peripherals					Features		
	Maximum Frequency of Operation (MHz)	Program Memory (Kbytes)	Data Memory (Kbytes)	EEPROM (Kbytes)	Timer Modules	Capable/Compare/PWM Modules	Serial Ports (SPI/I <sup>2</sup> C, USART)	Parallel Slave Port	A/D Converter (8-bit)	Interrupt Sources	I/O Pins	Voltage Range (Volts)	In-Circuit Serial Programming	Brown-out Reset
PIC16C710 <sup>(1)</sup>	20	512	36	TMR0	—	—	—	4	4	13	3.0-6.0	Yes	Yes	18-pin DIP, SOIC; 20-pin SSOP
PIC16C71	20	1K	36	TMR0	—	—	—	4	4	13	3.0-6.0	Yes	—	18-pin DIP, SOIC
PIC16C711 <sup>(1)</sup>	20	1K	68	TMR0	—	—	—	4	4	13	3.0-6.0	Yes	Yes	18-pin DIP, SOIC; 20-pin SSOP
PIC16C72 <sup>(1)</sup>	20	2K	128	TMR0, TMR1, TMR2	1	SPI/I <sup>2</sup> C	—	5	8	22	3.0-6.0	Yes	Yes	28-pin SDIP, SOIC, SSOP
PIC16C73	20	4K	192	TMR0, TMR1, TMR2	2	SPI/I <sup>2</sup> C, USART	—	5	11	22	3.0-6.0	Yes	—	28-pin SDIP, SOIC
PIC16C73A <sup>(1)</sup>	20	4K	192	TMR0, TMR1, TMR2	2	SPI/I <sup>2</sup> C, USART	—	5	11	22	3.0-6.0	Yes	Yes	28-pin SDIP, SOIC
PIC16C74	20	4K	192	TMR0, TMR1, TMR2	2	SPI/I <sup>2</sup> C, USART	Yes	8	12	33	3.0-6.0	Yes	—	40-pin DIP; 44-pin PLCC, MQFP
PIC16C74A <sup>(1)</sup>	20	4K	192	TMR0, TMR1, TMR2	2	SPI/I <sup>2</sup> C, USART	Yes	8	12	33	3.0-6.0	Yes	Yes	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 PIC16CXX Family devices use serial programming with clock pin RB6 and data pin RB7.

Note 1: Please contact your local sales office for availability of these devices.

TABLE E-5: PIC16C8X FAMILY OF DEVICES

	Clock			Memory			Peripherals		Features	
	Maximum Frequency of Operation (MHz)			Program Memory			Timer Module(s)		Voltage Range (Volts)	
	EPROM	Data EEPROM (bytes)	Data Memory (bytes)	ROM	Data EEPROM (bytes)	Data Memory (bytes)	Interrupt Sources	I/O Pins	In-Circuit Serial Programming	Packages
PIC16C83 <sup>(1)</sup>	10	512	—	36	64	TMR0	4	13	Yes	2.0-6.0 18-pin DIP, SOIC
PIC16CR83 <sup>(1)</sup>	10	—	512	36	64	TMR0	4	13	Yes	2.0-6.0 18-pin DIP, SOIC
PIC16C84	10	1K	—	36	64	TMR0	4	13	Yes	2.0-6.0 18-pin DIP, SOIC
PIC16C84A <sup>(1)</sup>	10	1K	—	68	64	TMR0	4	13	Yes	2.0-6.0 18-pin DIP, SOIC
PIC16CR84 <sup>(1)</sup>	10	—	1K	68	64	TMR0	4	13	Yes	2.0-6.0 18-pin DIP, SOIC

All PIC16/17 family devices have Power-on Reset, selectable Watchdog Timer, selectable code protect, and high I/O current capability.

All PIC16CXX family devices use serial programming with clock pin RB6 and data pin RB7.

Note 1: Please contact your local sales office for availability of these devices.

---

All PIC16/17 Family devices have Power-on Reset, selectable Watchdog Timer, selectable code protect and high I/O current capability.

## E.1 Pin Compatibility

Devices that have the same package type and VDD, VSS and MCLR pin locations are said to be pin compatible. This allows these different devices to operate in the same socket. Compatible devices may only require minor software modification to allow proper operation in the application socket (ex., PIC16C56 and PIC16C61 devices). Not all devices in the same package size are pin compatible; for example, the PIC16C62 is compatible with the PIC16C63, but not the PIC16C55.

Pin compatibility does not mean that the devices offer the same features. As an example, the PIC16C54 is pin compatible with the PIC16C71, but does not have an A/D converter, weak pull-ups on PORTB, or interrupts.

**TABLE E-7: PIN COMPATIBLE DEVICES**

Pin Compatible Devices	Package
PIC16C54, PIC16C54A, PIC16CR54, PIC16CR54A, PIC16CR54B, PIC16C56, PIC16CR56, PIC16C58A, PIC16CR58A, PIC16CR58B, PIC16C61, PIC16C620, PIC16C621, PIC16C622, PIC16C70, PIC16C71, PIC16C71A PIC16C83, PIC16CR83, PIC16C84, PIC16C84A, PIC16CR84	18 pin (20 pin)
PIC16C55, PIC16CR55, PIC16C57, PIC16CR57A, PIC16CR57B	28 pin
PIC16C62, PIC16CR62, PIC16C62A, PIC16C63, PIC16C72, PIC16C73, PIC16C73A	28 pin
PIC16C64, PIC16CR64, PIC16C64A, PIC16C65, PIC16C65A, PIC16C74, PIC16C74A	40 pin
PIC17C42, PIC17C43, PIC17C44	40 pin

# PIC16C6X

---

NOTES:

## INDEX

### A

Absolute Maximum Ratings .....	149, 171, 189, 209, 227
ALU .....	9
Application Notes .....	
AN552 .....	49
AN556 .....	45
AN594 .....	73
Architectural Overview .....	9
Assembler .....	146

### B

Baud Rate Formula .....	97
Baud Rate Generator .....	97
Baud Rates .....	
Asynchronous Mode .....	98
Error, Calculating .....	97
RX Pin Sampling, Timing Diagrams.....	100
Sampling .....	100
Synchronous Mode .....	98

### Block Diagrams

Compare Mode .....	75
Crystal Oscillator, Ceramic Resonator .....	113
External Brown-out Protection .....	123
External Parallel Resonant Crystal Circuit .....	115
External Power-on Reset .....	123
External Series Resonant Crystal Circuit.....	115
In-circuit Programming Connections.....	130
Interrupt Logic .....	125
On-chip Reset Circuit.....	116
Parallel Slave Port, PORTD-PORTE .....	57
PIC16C61 .....	10
PIC16C62 .....	11
PIC16C62A .....	11
PIC16C63 .....	12
PIC16C64 .....	11
PIC16C64A .....	11
PIC16C65 .....	12
PIC16C65A .....	12
PIC16CR62.....	11
PIC16CR64.....	11
PORTC .....	51
PORTD (I/O Mode) .....	53
PORTE (I/O Mode) .....	54
PWM .....	76
RA3:RA0 pins .....	47
RA4 .....	47
RA5 pin .....	47
RB3:RB0 pins .....	50
RB7:RB4 pins .....	49, 50
RC Oscillator Mode.....	115
SPI Master/Slave Connection .....	82
SSP, I <sup>2</sup> C Mode.....	88
SSP, SPI Mode.....	81
Timer0.....	61
Timer0/WDT Prescaler .....	64
Timer1.....	68
Timer2.....	71
USART Receive.....	103
USART Transmit.....	101
Watchdog Timer.....	128
BOR .....	119

### C

C Compiler (MP-C) .....	143, 147
Capture/Compare/PWM (CCP) .....	
Capture Mode.....	74
CCP1 .....	73
CCP2 .....	73
Compare Mode.....	75
Compare Mode Block Diagram .....	75
Overview.....	59
Prescaler .....	75
PWM Block Diagram .....	76
PWM Mode.....	76
PWM, Example Frequencies/Resolutions .....	77
Section.....	73
Carry .....	9
CCP1CON .....	23, 25, 27, 29
CCP2CON .....	23, 25, 27, 29
CCPR1H .....	23, 25, 27, 29
CCPR1L .....	23, 25, 27, 29
CCPR2H .....	23, 25, 27, 29
CCPR2L .....	23, 25, 27, 29
Clock Polarity, SPI Mode.....	82
Clocking Scheme.....	17
Code Examples .....	
Changing Between Capture Prescalers .....	75
Changing Prescaler (T0 to WDT) .....	65
Changing Prescaler (WDT to T0) .....	65
Ensuring Interrupts are Globally Disabled .....	124
I <sup>2</sup> C, Idle Mode.....	93
I <sup>2</sup> C, Receive Mode .....	93
I <sup>2</sup> C, Transmit Mode .....	93
Indirect Addressing.....	46
Initializing PORTA .....	47
Initializing PORTB .....	49
Initializing PORTC .....	51
Loading the SSPBUF Register.....	81
Reading a 16-bit Free-running Timer .....	69
Read-Modify-Write on an I/O Port .....	56
Saving Status and W Registers .....	127
Subroutine Call, Page0 to Page1 .....	46
Code Protection.....	130
Computed GOTO .....	45
Configuration Bits .....	111
Configuration Word, Diagram .....	112
Connecting Two Microcontrollers .....	82, 84

### D

Data Memory .....	
Organization .....	20
Section.....	20
Data Sheet .....	
Compatibility .....	261
Modifications.....	261
What's New .....	262
DC CHARACTERISTICS .....	150, 172, 210, 228
DC Characteristics.....	190
Development Support.....	143
Development Systems.....	147
Development Tools.....	143
Device Drawings .....	
18-Lead Ceramic CERDIP Dual In-line with Window (300 mil) .....	250
18-Lead Plastic Dual In-line (300 mil).....	245
18-Lead Plastic Surface Mount (SOIC - Wide, 300 mil Body) .....	248
28-Lead Ceramic CERDIP Dual In-line with Window (300 mil).....	251

# PIC16C6X

28-Lead Ceramic Side Brazed Dual In-Line with Window (300 mil) .....	253
28-Lead Plastic Dual In-line (300 mil) .....	246
28-Lead Plastic Surface Mount (SOIC - Wide, 300 mil Body) .....	249
28-Lead Plastic Surface Mount (SSOP - 209 mil Body 5.30 mm) .....	254
40-Lead Ceramic CERDIP Dual In-line with Window (600 mil) .....	252
40-Lead Plastic Dual In-line (600 mil) .....	247
44-Lead Plastic Leaded Chip Carrier (Square) .....	255
44-Lead Plastic Surface Mount (MQFP 10x10 mm Body 1.6/0.15 mm Lead Form) .....	256, 257
Digit Carry .....	9
Direct Addressing .....	46
Dynamic Data Exchange (DDE) .....	143

## E

Electrical Characteristics .....	149, 171, 189, 209, 227
----------------------------------	-------------------------

## F

Family of Devices .....	
PIC16C5X .....	263
PIC16C62X .....	264
PIC16C6X .....	6, 265
PIC16C7X .....	266
PIC16C8X .....	267
PIC17CXX .....	268
FSR .....	23, 24, 25, 26, 27, 28, 29, 30
Fuzzy Logic Dev. System ( <i>fuzzyTECH</i> ®-MP) .....	143, 147

## G

General Purpose Registers .....	20
Graphs, PIC16C61 .....	159

## I

I/O Ports, Section .....	47
I <sup>2</sup> C - See Synchronous Serial Port (SSP) .....	
ID Locations .....	130
In-circuit Serial Programming .....	130
INDF .....	23, 25, 27, 29
Indirect Addressing .....	46
Instruction Cycle .....	17
Instruction Flow/Pipelining .....	17
Instruction Format .....	131
Instruction Set .....	
ADDLW .....	133
ADDWF .....	133
ANDLW .....	133
ANDWF .....	133
BCF .....	134
BSF .....	134
BTFSC .....	134
BTFSS .....	135
CALL .....	135
CLRF .....	135
CLRW .....	135
CLRWDI .....	136
COMF .....	136
DECF .....	136
DECFSZ .....	136
GOTO .....	137
INCF .....	137
INCFSSZ .....	137
IORLW .....	137
IORWF .....	138
MOVF .....	138
MOVLW .....	138
MOVWF .....	138

NOP .....	139
OPTION .....	139
RETFIE .....	139
RETLW .....	139
RETURN .....	140
RLF .....	140
RRF .....	140
SLEEP .....	140
SUBLW .....	141
SUBWF .....	141
SWAPF .....	142
TRIS .....	142
XORLW .....	142
XORWF .....	142
Section .....	131
Summary Table .....	132
INTCON .....	23, 24, 25, 26, 27, 28, 29, 30
Inter-IC Bus .....	84
Interrupt on Change Feature .....	49
Interrupts .....	
Context Saving .....	127
Port RB .....	49
RB0/INT .....	50, 126
RB0/INT Timing Diagram .....	126
Timer0 .....	61
Timer0, Timing .....	62
Timing Diagram, Wake-up from SLEEP .....	129
TMR0 .....	126
Wake-up from SLEEP .....	129
Section .....	124

## L

Loading the Program Counter .....	45
-----------------------------------	----

## M

MPASM Assembler .....	143, 146
MP-C C Compiler .....	147
MPSIM Software Simulator .....	143, 147

## O

One-Time-Programmable Devices .....	7
OPCODE .....	131
OPTION .....	24, 26, 28, 30
Oscillator Start-up Timer (OST) .....	111, 117
Oscillators .....	
Block Diagram, External Parallel Resonant Crystal .....	115
Capacitor Selection .....	69
Configuration .....	113
External Crystal Circuit .....	115
HS .....	113, 118
LP .....	113, 118
RC, Block Diagram .....	115
RC, Section .....	115
XT .....	113

## P

Packaging Information .....	245
Parallel Slave Port .....	
PORTD .....	53
Section .....	57
PCL .....	23, 24, 25, 26, 27, 28, 29, 30
PCLATH .....	23, 24, 25, 26, 27, 28, 29, 30
PCON .....	24, 26, 28, 30
PD .....	119
PICDEM-1 Low-Cost PIC16/17 Demo Board .....	143, 145
PICDEM-2 Low-Cost PIC16CXX Demo Board .....	143, 145
PICMASTER System Configuration .....	143
PICMASTER™ RT In-Circuit Emulator .....	143
PICSTART™ Low-Cost Development System .....	143, 145



PIE1 .....	24, 26, 28, 30
PIE2 .....	24, 26, 28, 30
Pin Compatible Devices .....	269
Pin Functions	
MCLR/VPP .....	15
OSC1/CLKIN .....	15
OSC2/CLKOUT .....	15
PORTA .....	48
PORTB .....	50
PORTC .....	51
PORTD .....	53
PORTE .....	55
RA4/T0CKI .....	15, 48
RA5/SS .....	15, 48
RB0/INT .....	15, 50
RB6 .....	130
RB7 .....	130
RC0/T1OSI/T1CKI .....	51
RC0/T1OSO/T1CKI .....	15, 51
RC1/T1OSI .....	51
RC1/T1OSI/CCP2 .....	15, 51
RC1/T1OSO .....	51
RC2/CCP1 .....	15, 51, 52
RC3/SCK/SCL .....	15, 51, 52
RC4/SDI/SDA .....	15, 51, 52
RC5/SDO .....	15, 51, 52
RC6/TX/CK .....	15, 51, 52, 95–109
RC7/RX/DT .....	15, 51, 52, 95–109
RD7/PSP7:RD0/PSP0 .....	16, 53
RE0/RD .....	16, 55, 57
RE1/WR .....	16, 55, 57
RE2/CS .....	16, 55, 57
SCK .....	81–83
SDI .....	81–83
SDO .....	81–83
SS .....	81–83
VDD .....	16
Vss .....	16
PIR1 .....	23, 25, 27, 29
PIR2 .....	23, 25, 27, 29
POP .....	45
POR	
Time-Out Sequence on Power-Up .....	122
POR .....	119
Port RB Interrupt .....	49
PORTA .....	23, 25, 27, 29
PORTB .....	23, 25, 27, 29
PORTC .....	23, 25, 27, 29
PORTD .....	23, 25, 27, 29
PORTE .....	23, 25, 27, 29
Ports	
Bi-directional .....	56
I/O Programming Considerations .....	56
PORTA .....	15
PORTB .....	15
PORTC .....	15
PORTD .....	16
PORTE .....	16
Successive Operations on an I/O Port .....	56
Power-down Mode .....	129
Power-on Reset (POR) .....	117
Power-up Timer (PWRT) .....	111, 117
PR2 .....	24, 26, 28, 30
Prescaler .....	64
PRO MATE™ Universal Programmer .....	143, 145
Program Memory	
Map .....	19

Organization .....	19
Paging .....	45
Section .....	19
Programming while In-circuit .....	130
PUSH .....	45
<b>Q</b>	
Quadrature Clocks .....	17
Quick-Turnaround-Production .....	7
<b>R</b>	
RBPU .....	49
RC Oscillator .....	118
RCREG .....	23, 25, 27, 29
RCSTA .....	23, 25, 27, 29
Registers	
CCP1CON	
Diagram .....	74
Section .....	74
Summary .....	23, 25, 27, 29
CCP2CON	
Diagram .....	74
Section .....	74
Summary .....	25, 29
CCPR1H	
Summary .....	23, 25, 27, 29
CCPR1L	
Summary .....	23, 25, 27, 29
CCPR2H	
Summary .....	25, 29
CCPR2L	
Summary .....	25, 29
FSR	
Indirect Addressing .....	46
Summary .....	23, 25, 27, 29
INDF	
Indirect Addressing .....	46
Summary .....	23, 25, 27, 29
INTCON	
Diagram .....	34
Section .....	33
Summary .....	23, 25, 27, 29
OPTION	
Diagram .....	32
Section .....	32
Summary .....	24, 26, 28, 30
PCL	
Section .....	45
Summary .....	23, 25, 27, 29
PCLATH	
Section .....	45
Summary .....	23, 25, 27, 29
PCON	
Diagram .....	44
Section .....	44
Summary .....	24, 26, 28, 30
PIE1	
Diagram .....	37
Section .....	35
Summary .....	24, 26, 28, 30
PIE2	
Diagram .....	42
Section .....	42
Summary .....	26, 30
PIR1	
Diagram .....	41
Section .....	38
Summary .....	23, 25, 27, 29

# PIC16C6X

PIR2			
Diagram .....	43		
Section .....	43		
Summary .....	25, 29		
PORTA			
Section .....	47		
Summary .....	23, 25, 27, 29		
PORTB			
Section .....	49		
Summary .....	23, 25, 27, 29		
PORTC			
Section .....	51		
Summary .....	23, 25, 27, 29		
PORTD			
Section .....	53		
Summary .....	27, 29		
PORTE			
Section .....	54		
Summary .....	27, 29		
PR2			
Summary .....	24, 26, 28, 30		
RCREG			
Summary .....	25, 29		
RCSTA			
Diagram .....	96		
Summary .....	25, 29		
SPBRG			
Summary .....	26, 30		
SSPBUF			
Section .....	81		
Summary .....	23, 25, 27, 29		
SSPCON			
Diagram .....	80		
Summary .....	23, 25, 27, 29		
SSPSR			
Section .....	81		
SSPSTAT			
Diagram .....	79		
Section .....	79		
Summary .....	24, 26, 28, 30		
STATUS			
Diagram .....	31		
Section .....	31		
Summary .....	23, 25, 27, 29		
T1CON			
Diagram .....	67		
Section .....	67		
Summary .....	23, 25, 27, 29		
T2CON			
Diagram .....	71		
Section .....	71		
Summary .....	23, 25, 27, 29		
TMR0			
Summary .....	23, 25, 27, 29		
TMR1H			
Summary .....	23, 25, 27, 29		
TMR1L			
Summary .....	23, 25, 27, 29		
TMR2			
Summary .....	23, 25, 27, 29		
TRISA			
Section .....	47		
Summary .....	24, 26, 28, 30		
TRISB			
Section .....	49		
Summary .....	24, 26, 28, 30		
TRISC			
Section .....	51		
Summary .....	24, 26, 28, 30		
TRISD			
Section .....	53		
Summary .....	28, 30		
TRISE			
Diagram .....	54		
Section .....	54		
Summary .....	28, 30		
TXREG			
Summary .....	25, 29		
TXSTA			
Diagram .....	95		
Section .....	95		
Summary .....	26, 30		
W .....	9		
Special Function Registers, Initialization Conditions .....	120		
Special Function Registers, Reset Conditions .....	119		
Special Function Register Summary .....	23, 25, 27, 29		
File Maps .....	21		
Resets .....	116		
RP0 bit .....	20		
<b>S</b>			
SCI - See Universal Synchronous Asynchronous Receiver Transmitter (USART)			
SCK .....	81		
SDI .....	81		
SDO .....	81		
Serial Programming .....	130		
Serial Programming, Block Diagram .....	130		
Serialized Quick-Turnaround-Production .....	7		
SLEEP Mode .....	111, 129		
Software Simulator (MPSIM) .....	147		
SPBRG .....	24, 26, 28, 30		
Special Features, Section .....	111		
SPI Mode .....	81		
SS .....	81		
SSPADD .....	24, 26, 28, 30		
SSPBUF .....	23, 25, 27, 29		
SSPCON .....	23, 25, 27, 29		
SSPSTAT .....	24, 26, 28, 30		
Stack .....	45		
STATUS .....	23, 24, 25, 26, 27, 28, 29, 30		
Status Bits .....	119		
Status Bits During Various Resets .....	119		
Switching Prescalers .....	65		
Synchronous Serial Port (SSP)			
Block Diagram, SPI Mode .....	81		
I <sup>2</sup> C			
7-bit, 10-bit Address Format .....	85		
Addressing .....	85		
Arbitration .....	87		
Block Diagram .....	88		
Clock Synchronization .....	87		
Combined Master-Receiver/Transmitter Sequence .....	86		
Data Transfer .....	85		
Initiating and Terminating Data Transfer .....	84		
Master Mode .....	92		
Master-Receiver Sequence .....	86		
Master-Transmitter Sequence .....	86		
Multi-Master .....	87		
Multi-Master Mode .....	92		
Operation .....	88		
Slave Mode .....	89		

Slave Mode, Addressing .....	89	I <sup>2</sup> C Bus Data .....	185
Slave Mode, Reception .....	90	I <sup>2</sup> C Bus Start/Stop Bits .....	184
Slave Mode, Transmission .....	91	Oscillator Start-up Timer .....	179
Terminology, Table .....	84	Power-up Timer .....	179
Timing Diagram, Slave Mode Reception .....	90	Reset .....	179
Timing Diagram, Transmission, 7-bit .....	91	SPI Mode .....	183
Transfer Acknowledge .....	85	Timer0 .....	180
I <sup>2</sup> C Section .....	84	Timer1 .....	180
Section .....	79	Watchdog Timer .....	179
SPI Master/Slave Diagram .....	82	PIC16C62A .....	
SPI Mode .....	81	Brown-out Reset .....	199
<b>T</b>		Capture/Compare/PWM .....	201
T1CON .....	23, 25, 27, 29	CLKOUT and I/O .....	198
T2CON .....	23, 25, 27, 29	External Clock .....	196
Time-out Sequence .....	118	I <sup>2</sup> C Bus Data .....	205
Timer Modules		I <sup>2</sup> C Bus Start/Stop Bits .....	204
Overview, all .....	59	Oscillator Start-up Timer .....	199
Timer0		Power-up Timer .....	199
Block Diagram .....	61	Reset .....	199
Counter Mode .....	61	SPI Mode .....	203
External Clock .....	63	Timer0 .....	200
Interrupt .....	61	Timer1 .....	200
Overview .....	59	Watchdog Timer .....	199
Prescaler .....	64	PIC16C63 .....	
Section .....	61	Brown-out Reset .....	235
Timer Mode .....	61	Capture/Compare/PWM .....	237
Timing Diagram .....	61	CLKOUT and I/O .....	234
TMR0 register .....	61	External Clock .....	233
Timer1		I <sup>2</sup> C Bus Data .....	241
Block Diagram .....	68	I <sup>2</sup> C Bus Start/Stop Bits .....	240
Capacitor Selection .....	69	Oscillator Start-up Timer .....	235
Counter Mode, Asynchronous .....	69	Power-up Timer .....	235
Counter Mode, Synchronous .....	68	Reset .....	235
External Clock .....	69	SPI Mode .....	239
Oscillator .....	69	Timer0 .....	236
Overview .....	59	Timer1 .....	236
Prescaler .....	68	USART Synchronous Receive (Master/Slave) .....	242
Section .....	67	USART Synchronous Transmission	
Timer Mode .....	68	(Master/Slave) .....	242
TMR1 Register Pair .....	67	Watchdog Timer .....	235
Timer2		PIC16C64 .....	
Block Diagram .....	71	Capture/Compare/PWM .....	181
Overview .....	59	CLKOUT and I/O .....	178
Postscaler .....	71	External Clock .....	177
Prescaler .....	71	I <sup>2</sup> C Bus Data .....	185
Section .....	71	I <sup>2</sup> C Bus Start/Stop Bits .....	184
TMR2 Register .....	71	Oscillator Start-up Timer .....	179
Timing Diagrams		Parallel Slave Port .....	182
Brown-out Reset .....	117	Power-up Timer .....	179
I <sup>2</sup> C Slave Reception, 7-bit .....	90	Reset .....	179
I <sup>2</sup> C, Clock Synchronization .....	87	SPI Mode .....	183
I <sup>2</sup> C, Data Transfer .....	85	Timer0 .....	180
I <sup>2</sup> C, Multi-Master Arbitration .....	87	Timer1 .....	180
I <sup>2</sup> C, Slave Mode Transmission - 7-bit .....	91	Watchdog Timer .....	179
PIC16C61 .....		PIC16C64A .....	
CLKOUT and I/O .....	156	Brown-out Reset .....	199
External Clock .....	155	Capture/Compare/PWM .....	201
Oscillator Start-up Timer .....	157	CLKOUT and I/O .....	198
Power-up Timer .....	157	External Clock .....	196
Reset .....	157	I <sup>2</sup> C Bus Data .....	205
Timer0 .....	158	I <sup>2</sup> C Bus Start/Stop Bits .....	204
Watchdog Timer .....	157	Oscillator Start-up Timer .....	199
PIC16C62 .....		Parallel Slave Port .....	202
Capture/Compare/PWM .....	181	Power-up Timer .....	199
CLKOUT and I/O .....	178	Reset .....	199
External Clock .....	177	SPI Mode .....	203
		Timer0 .....	200

# PIC16C6X

Timer1.....	200	SPI Mode, Slave Mode With $\overline{SS}$ Control .....	83
Watchdog Timer.....	199	USART, Asynchronous Reception .....	103
PIC16C65		USART, Synchronous Transmission.....	106
Capture/Compare/PWM.....	219	Wake-up from SLEEP Through Interrupts .....	129
CLKOUT and I/O.....	216	TMR0.....	23, 25, 27, 29
External Clock.....	215	TMR1H .....	23, 25, 27, 29
I <sup>2</sup> C Bus Data .....	223	TMR1L.....	23, 25, 27, 29
I <sup>2</sup> C Bus Start/Stop Bits.....	222	TMR2.....	23, 25, 27, 29
Oscillator Start-up Timer .....	217	TO.....	119
Parallel Slave Port .....	220	TRISA.....	24, 26, 28, 30
Reset.....	217	TRISB.....	24, 26, 28, 30
SPI Mode .....	221	TRISC.....	24, 26, 28, 30
Timer0.....	218	TRISD.....	24, 26, 28, 30
Timer1.....	218	TRISE.....	24, 26, 28, 30
USART Synchronous Receive (Master/Slave) ..	224	TXREG .....	23, 25, 27, 29
USART Synchronous Transmission		TXSTA .....	24, 26, 28, 30
(Master/Slave).....	224		
Watchdog Timer.....	217	<b>U</b>	
PIC16C65A		Universal Synchronous Asynchronous Receiver Transmitter	
Brown-out Reset .....	235	(USART)	
Capture/Compare/PWM.....	237	Asynchronous Mode	
CLKOUT and I/O.....	234	Setting Up Transmission .....	102
External Clock.....	233	Timing Diagram, Master Transmission.....	102
I <sup>2</sup> C Bus Data .....	241	Transmitter .....	101
I <sup>2</sup> C Bus Start/Stop Bits.....	240	Asynchronous Receiver	
Oscillator Start-up Timer .....	235	Setting Up Reception.....	104
Parallel Slave Port .....	238	Timing Diagram .....	103
Power-up Timer .....	235	Asynchronous Receiver Mode	
Reset.....	235	Block Diagram .....	103
SPI Mode .....	239	Section.....	103
Timer0.....	236	Section.....	95
Timer1.....	236	Synchronous Master Mode	
USART Synchronous Receive (Master/Slave) ..	242	Reception .....	107
USART Synchronous Transmission		Section.....	105
(Master/Slave).....	242	Setting Up Reception.....	107
Watchdog Timer.....	235	Setting Up Transmission .....	105
PIC16C62		Timing Diagram, Reception .....	108
Capture/Compare/PWM.....	201	Timing Diagram, Transmission.....	106
CLKOUT and I/O.....	198	Transmission .....	105
External Clock.....	196	Synchronous Slave Mode	
I <sup>2</sup> C Bus Data .....	205	Reception .....	109
I <sup>2</sup> C Bus Start/Stop Bits.....	204	Section.....	109
Oscillator Start-up Timer .....	199	Setting Up Reception.....	109
Power-up Timer .....	199	Setting Up Transmission .....	109
Reset.....	199	Transmit.....	109
SPI Mode .....	203	Transmit Block Diagram .....	101
Timer0.....	200	UV Erasable Devices.....	7
Timer1.....	200	<b>W</b>	
Watchdog Timer.....	199	Wake-up from Sleep .....	129
PIC16C64		Wake-up on Key Depression .....	49
Capture/Compare/PWM.....	201	Watchdog Timer (WDT)	
CLKOUT and I/O.....	198	Block Diagram .....	128
External Clock.....	196	Period .....	128
I <sup>2</sup> C Bus Data .....	205	Programming Considerations .....	128
I <sup>2</sup> C Bus Start/Stop Bits.....	204	Section.....	128
Oscillator Start-up Timer .....	199	Weak Internal Pull-ups.....	49
Parallel Slave Port .....	202	<b>X</b>	
Power-up Timer .....	199	XT .....	118
Reset.....	199	<b>Z</b>	
SPI Mode .....	203	Zero bit.....	9
Timer0.....	200		
Timer1.....	200		
Watchdog Timer.....	199		
Power-up Timer .....	217		
RB0/INT Interrupt.....	126		
RX Pin Sampling.....	100		
SPI Mode, Master/Slave Mode, No $\overline{SS}$ Control .....	83		

## LIST OF EXAMPLES

Example 3-1: Instruction Pipeline Flow .....	17
Example 4-1: Call of a Subroutine in Page 1 from Page 0 .....	46
Example 4-2: Indirect Addressing .....	46
Example 5-1: Initializing PORTA .....	47
Example 5-2: Initializing PORTB .....	49
Example 5-3: Initializing PORTC .....	51
Example 5-4: Read-Modify-Write Instructions on an I/O Port .....	56
Example 7-1: Changing Prescaler (Timer0→WDT) .....	65
Example 7-2: Changing Prescaler (WDT→Timer0) .....	65
Example 8-1: Reading a 16-bit Free-running Timer .....	69
Example 10-1: Changing Between Capture Prescalers .....	75
Example 10-2: PWM Period and Duty Cycle Calculation ...	77
Example 11-1: Loading the SSPBUF (SSPSR) Register .....	81
Example 12-1: Calculating Baud Rate Error .....	97
Example 13-1: Saving Status and W Registers in RAM (PIC16C61) .....	127
Example 13-2: Saving Status and W Registers in RAM (PIC16C62/62A/R62/63/64/64A/R64/65/65A) .....	127

## LIST OF FIGURES

Figure 3-1: PIC16C61 Block Diagram .....	10
Figure 3-2: PIC16C62/62A/R62/64/64A/R64 Block Diagram .....	11
Figure 3-3: PIC16C63/65/65A Block Diagram .....	12
Figure 3-4: Clock/Instruction Cycle .....	17
Figure 4-1: PIC16C61 Program Memory Map and Stack .....	19
Figure 4-2: PIC16C62/62A/R62/64/64A/R64 Program Memory Map and Stack .....	19
Figure 4-3: PIC16C63/65/65A Program Memory Map and Stack .....	19
FIGURE 4-4: PIC16C61 Register File Map .....	20
FIGURE 4-5: PIC16C62/62A/R62/64/64A/R64 Register File Map .....	21
Figure 4-6: PIC16C63/65/65A Register File Map .....	21
Figure 4-7: STATUS Register (Address 03h, 83h) .....	31
Figure 4-8: OPTION Register (Address 81h) .....	32
Figure 4-9: INTCON Register for PIC16C61 (Address 0Bh, 8Bh) .....	33
Figure 4-10: INTCON Register for PIC16C62/62A/R62/63/64/64A/R64/65/65A (Address 0Bh, 8Bh) .....	34
Figure 4-11: PIE1 Register for PIC16C62/62A/R62 (Address 8Ch) .....	35
Figure 4-12: PIE1 Register for PIC16C63 (Address 8Ch) .....	36
Figure 4-13: PIE1 Register for PIC16C64/64A/R64 (Address 8Ch) .....	36
Figure 4-14: PIE1 Register for PIC16C65/65A (Address 8Ch) .....	37
Figure 4-15: PIR1 Register for PIC16C62/62A/R62 (Address 0Ch) .....	38
Figure 4-16: PIR1 Register for PIC16C63 (Address 0Ch) .....	39
Figure 4-17: PIR1 Register for PIC16C64/64A/R64 (Address 0Ch) .....	40
Figure 4-18: PIR1 Register for PIC16C65/65A (Address 0Ch) .....	41
Figure 4-19: PIE2 Register (Address 8Dh) .....	42
Figure 4-20: PIR2 Register (Address 0Dh) .....	43

Figure 4-21: PCON Register for PIC16C62/64/65 (Address 8Eh) .....	44
Figure 4-22: PCON Register for PIC16C62A/R62/63/64A/R64/65A (Address 8Eh) .....	44
Figure 4-23: Loading of PC in Different Situations .....	45
Figure 4-24: Direct/Indirect Addressing .....	46
Figure 5-1: Block Diagram of the RA3:RA0 Pins and the RA5 Pin .....	47
Figure 5-2: Block Diagram of the RA4/T0CKI Pin .....	47
Figure 5-3: Block Diagram of the RB7:RB4 Pins FOR PIC16C61/62/64/65 .....	49
Figure 5-4: Block Diagram of the RB7:RB4 Pins FOR PIC16C62A/63/64A/65A .....	50
Figure 5-5: Block Diagram of the RB3:RB0 Pins .....	50
Figure 5-6: PORTC Block Diagram (Peripheral Output Override) .....	51
Figure 5-7: PORTD Block Diagram (In I/O Port Mode) .....	53
Figure 5-8: PORTE Block Diagram (In I/O Port Mode) .....	54
Figure 5-9: TRISE Register (Address 89h) .....	54
Figure 5-10: Successive I/O Operation .....	56
Figure 5-11: PORTD and PORTE as a Parallel Slave Port .....	57
Figure 7-1: Timer0 Block Diagram .....	61
Figure 7-2: Timer0 Timing: Internal Clock/No Prescaler .....	61
Figure 7-3: Timer0 Timing: Internal Clock/Prescale 1:2 .....	62
Figure 7-4: Timer0 Interrupt Timing .....	62
Figure 7-5: Timer0 Timing With External Clock .....	63
Figure 7-6: Block Diagram of the Timer0/WDT Prescaler .....	64
Figure 8-1: T1CON: Timer1 Control Register (Address 10h) .....	67
Figure 8-2: Timer1 Block Diagram .....	68
Figure 9-1: Timer2 Block Diagram .....	71
Figure 9-2: T2CON: Timer2 Control Register (Address 12h) .....	71
Figure 10-1: CCP1CON Register (Address 17h) / CCP2CON Register (Address 1Dh) .....	74
Figure 10-2: Capture Mode Operation Block Diagram .....	74
Figure 10-3: Compare Mode Operation Block Diagram .....	75
Figure 10-4: Simplified PWM Block Diagram .....	76
FIGURE 10-5: PWM Output .....	76
Figure 11-1: SSPSTAT: Sync Serial Port Status Register (Address 94h) .....	79
Figure 11-2: SSPCON: Sync Serial Port Control Register (Address 14h) .....	80
Figure 11-3: SSP Block Diagram (SPI Mode) .....	81
Figure 11-4: SPI Master/Slave Connection .....	82
Figure 11-5: SPI Mode Timing (Master Mode or Slave Mode w/o $\overline{SS}$ Control) .....	83
Figure 11-6: SPI Mode Timing (Slave Mode with $\overline{SS}$ Control) .....	83
Figure 11-7: Start and Stop Conditions .....	84
Figure 11-8: 7-bit Address Format .....	85
Figure 11-9: I <sup>2</sup> C 10-bit Address Format .....	85
Figure 11-10: Slave-Receiver Acknowledge .....	85
Figure 11-11: Data Transfer Wait State .....	85
Figure 11-12: Master-Transmitter Sequence .....	86
Figure 11-13: Master-Receiver Sequence .....	86
Figure 11-14: Combined Format .....	86
Figure 11-15: Multi-master Arbitration (Two Masters) .....	87

# PIC16C6X

Figure 11-16: Clock Synchronization .....	87	Figure 16-1: Load Conditions for Device Timing Specifications .....	154
Figure 11-17: SSP Block Diagram (I <sup>2</sup> C Mode) .....	88	Figure 16-2: External Clock Timing .....	155
Figure 11-18: I <sup>2</sup> C Waveforms for Reception (7-Bit Address) .....	90	Figure 16-3: CLKOUT and I/O Timing .....	156
Figure 11-19: I <sup>2</sup> C Waveforms for Transmission (7-bit Address) .....	91	Figure 16-4: Reset, Watchdog Timer, Oscillator Start-up Timer and Power-up Timer Timing .....	157
Figure 11-20: Operation of the I <sup>2</sup> C Module in IDLE_MODE, RCV_MODE or XMIT_MODE .....	93	Figure 16-5: Timer0 Clock Timings .....	158
Figure 12-1: TXSTA: Transmit Status and Control Register (Address 98h) .....	95	Figure 17-1: Typical RC Oscillator Frequency vs. Temperature .....	159
Figure 12-2: RCSTA: Receive Status and Control Register (Address 18h) .....	96	Figure 17-2: Typical RC Oscillator Frequency vs. VDD .....	160
Figure 12-3: RX Pin Sampling Scheme (BRGH = 0) .....	100	Figure 17-3: Typical RC Oscillator Frequency vs. VDD .....	160
Figure 12-4: RX Pin Sampling Scheme (BRGH = 1) .....	100	Figure 17-4: Typical RC Oscillator Frequency vs. VDD .....	161
Figure 12-5: RX Pin Sampling Scheme (BRGH = 1) .....	100	Figure 17-5: Typical IPD vs. VDD Watchdog Timer Disabled 25°C .....	161
Figure 12-6: USART Transmit Block Diagram .....	101	Figure 17-6: Typical IPD vs. VDD Watchdog Timer Enabled 25°C .....	162
Figure 12-7: Asynchronous Master Transmission .....	102	Figure 17-7: Maximum IPD vs. VDD Watchdog Disabled .....	162
Figure 12-8: Asynchronous Master Transmission (Back to Back) .....	102	Figure 17-8: Maximum IPD vs. VDD Watchdog Enabled* .....	163
Figure 12-9: USART Receive Block Diagram .....	103	Figure 17-9: VTH (Input Threshold Voltage) of I/O Pins vs. VDD .....	163
Figure 12-10: Asynchronous Reception .....	103	Figure 17-10: VIH, VIL of MCLR, T0CKI and OSC1 (in RC Mode) vs. VDD .....	164
Figure 12-11: Synchronous Transmission .....	106	Figure 17-11: VTH (Input Threshold Voltage) of OSC1 Input (in XT, HS, and LP Modes) vs. VDD .....	164
Figure 12-12: Synchronous Transmission (Through TXEN) .....	106	Figure 17-12: Typical IDD vs. Frequency (External Clock, 25°C) .....	165
Figure 12-13: Synchronous Reception (Master Mode, SREN) .....	108	Figure 17-13: Maximum IDD vs. Frequency (External Clock, -40° to +85°C) .....	165
Figure 13-1: Configuration Word for PIC16C61 .....	111	Figure 17-14: Maximum IDD vs. Frequency (External Clock, -55° to +125°C) .....	166
Figure 13-2: Configuration Word for PIC16C62/64/65 .....	112	Figure 17-15: WDT Timer Time-out Period vs. VDD .....	166
Figure 13-3: Configuration Word for PIC16C62A/R62/63/64A/R64/65A .....	112	Figure 17-16: Transconductance (gm) of HS Oscillator vs. VDD .....	167
Figure 13-4: Crystal/Ceramic Resonator Operation (HS, XT or LP OSC Configuration) .....	113	Figure 17-17: Transconductance (gm) of LP Oscillator vs. VDD .....	167
Figure 13-5: External Clock Input Operation (HS, XT or LP OSC Configuration) .....	113	Figure 17-18: Transconductance (gm) of XT Oscillator vs. VDD .....	168
Figure 13-6: External Parallel Resonant Crystal Oscillator Circuit .....	115	Figure 17-19: IOH vs. VOH, VDD = 3V .....	168
Figure 13-7: External Series Resonant Crystal Oscillator Circuit .....	115	Figure 17-20: IOH vs. VOH, VDD = 5V .....	169
Figure 13-8: RC Oscillator Mode .....	115	Figure 17-21: IOL vs. VOL, VDD = 3V .....	169
Figure 13-9: Simplified Block Diagram of On-chip Reset Circuit .....	116	Figure 17-22: IOL vs. VOL, VDD = 5V .....	170
Figure 13-10: Brown-out Situations .....	117	Figure 18-1: Load Conditions for Device Timing Specifications .....	176
Figure 13-11: Time-out Sequence on Power-up (MCLR not Tied to VDD): Case 1 .....	122	Figure 18-2: External Clock Timing .....	177
Figure 13-12: Time-out Sequence on Power-up (MCLR Not Tied To VDD): Case 2 .....	122	Figure 18-3: CLKOUT and I/O Timing .....	178
Figure 13-13: Time-out Sequence on Power-up (MCLR Tied to VDD) .....	122	Figure 18-4: Reset, Watchdog Timer, Oscillator Start-up Timer and Power-up Timer Timing .....	179
Figure 13-14: External Power-on Reset Circuit (For Slow VDD Power-up) .....	123	Figure 18-5: Timer0 and Timer1 Clock Timings .....	180
Figure 13-15: External Brown-out Protection Circuit 1 .....	123	Figure 18-6: Capture/Compare/PWM Timings (CCP1) .....	181
Figure 13-16: External Brown-out Protection Circuit 2 .....	123	Figure 18-7: Parallel Slave Port Timing For The PIC16C64 Only) .....	182
Figure 13-17: Interrupt Logic for PIC16C61 .....	125	Figure 18-8: SPI Mode Timing .....	183
Figure 13-18: Interrupt Logic for PIC16C62/62A/R62/63 .....	125	Figure 18-9: I <sup>2</sup> C Bus Start/Stop Bits Timing .....	184
Figure 13-19: Interrupt Logic for PIC16C64/64A/R64/65/65A .....	125	Figure 18-10: I <sup>2</sup> C Bus Data Timing .....	185
Figure 13-20: INT Pin Interrupt Timing .....	126	Figure 20-1: Load Conditions for Device Timing Specifications .....	195
Figure 13-21: Watchdog Timer Block Diagram .....	128	Figure 20-2: External Clock Timing .....	196
Figure 13-22: Summary of Watchdog Timer Registers .....	128	Figure 20-3: CLKOUT and I/O Timing .....	198
Figure 13-23: Wake-up from Sleep Through Interrupt .....	129	Figure 20-4: Reset, Watchdog Timer, Oscillator Start-up Timer and Power-up Timer Timing .....	199
Figure 13-24: Typical In-circuit Serial Programming Connection .....	130	Figure 20-5: Brown-out Reset Timing .....	199
Figure 14-1: General Format for Instructions .....	131	Figure 20-6: Timer0 and Timer1 Clock Timings .....	200
Figure 15-1: PICMASTER System Configuration .....	143		

Figure 20-7:	Capture/Compare/PWM Timings (CCP1).....	201
Figure 20-8:	Parallel Slave Port Timing For The PIC16C64A/R64 Only.....	202
Figure 20-9:	SPI Mode Timing .....	203
Figure 20-10:	I <sup>2</sup> C Bus Start/Stop Bits Timing.....	204
Figure 20-11:	I <sup>2</sup> C Bus Data Timing .....	205
Figure 22-1:	Load Conditions for Device Timing Specifications.....	214
Figure 22-2:	External Clock Timing.....	215
Figure 22-3:	CLKOUT and I/O Timing.....	216
Figure 22-4:	Reset, Watchdog Timer, Oscillator Start-up Timer and Power-up Timer Timing.....	217
Figure 22-5:	Timer0 and Timer1 Clock Timings.....	218
Figure 22-6:	Capture/Compare/PWM Timings (CCP1 and CCP2) .....	219
Figure 22-7:	Parallel Slave Port Timing .....	220
Figure 22-8:	SPI Mode Timing .....	221
Figure 22-9:	I <sup>2</sup> C Bus Start/Stop Bits Timing.....	222
Figure 22-10:	I <sup>2</sup> C Bus Data Timing .....	223
Figure 22-11:	USART Module: Synchronous Transmission (Master/Slave) Timing .....	224
Figure 22-12:	USART Module: Synchronous Receive (Master/Slave) Timing.....	224
Figure 24-1:	Load Conditions for Device Timing Specifications.....	232
Figure 24-2:	External Clock Timing.....	233
Figure 24-3:	CLKOUT and I/O Timing.....	234
Figure 24-4:	Reset, Watchdog Timer, Oscillator Start-up Timer and Power-up Timer Timing.....	235
Figure 24-5:	Brown-out Reset Timing .....	235
Figure 24-6:	Timer0 and Timer1 Clock Timings.....	236
Figure 24-7:	Capture/Compare/PWM Timings (CCP1 and CCP2) .....	237
Figure 24-8:	Parallel Slave Port Timing For the PIC16C65A Only .....	238
Figure 24-9:	SPI Mode Timing .....	239
Figure 24-10:	I <sup>2</sup> C Bus Start/Stop Bits Timing.....	240
Figure 24-11:	I <sup>2</sup> C Bus Data Timing .....	241
Figure 24-12:	USART Module: Synchronous Transmission (Master/Slave) Timing .....	242
Figure 24-13:	USART Module: Synchronous Receive (Master/Slave) Timing.....	242

## LIST OF TABLES

Table 1-1:	PIC16C6X Family of Devices .....	6
Table 3-1:	PIC16C61 Pinout Description.....	13
Table 3-2:	PIC16C62/62A/R62/63 Pinout Description..	14
Table 3-3:	PIC16C64/64A/R64/65/65A Pinout Description .....	15
Table 4-1:	Special Function Registers for the PIC16C61 .....	22
Table 4-2:	Special Function Registers for the PIC16C62/62A/R62 .....	23
Table 4-3:	Special Function Registers for the PIC16C63.....	25
Table 4-4:	Special Function Registers for the PIC16C64/64A/R64 .....	27
Table 4-5:	Special Function Registers for the PIC16C65/65A .....	29
Table 5-1:	PORTA Functions .....	48
Table 5-2:	Summary of Registers Associated with PORTA .....	48
Table 5-3:	PORTB Functions .....	50
Table 5-4:	Summary of Registers Associated with PORTB .....	50
Table 5-5:	PORTC Functions for PIC16C62/64 .....	51
Table 5-6:	PORTC Functions for PIC16C62A/R62/64A/R64.....	52
Table 5-7:	PORTC Functions for PIC16C63/65/65A .....	52
Table 5-8:	Summary of Registers Associated with PORTC .....	52
Table 5-9:	PORTD Functions .....	53
Table 5-10:	Summary of Registers Associated with PORTD .....	53
Table 5-11:	PORTE Functions .....	55
Table 5-12:	Summary of Registers Associated with PORTE .....	55
Table 5-13:	Registers Associated with Parallel Slave Port.....	57
Table 7-1:	Registers Associated with Timer0 .....	65
Table 8-1:	Capacitor Selection for the Timer1 Oscillator .....	69
Table 8-2:	Registers Associated with Timer1 as a Timer/Counter .....	70
Table 9-1:	Registers Associated with Timer2 as a Timer/Counter .....	72
Table 10-1:	CCP Mode - Timer Resource .....	73
Table 10-2:	Interaction of Two CCP Modules.....	73
Table 10-3:	Example PWM Frequencies and Resolutions at 20 MHz .....	77
Table 10-4:	Registers Associated with Timer1, Capture and Compare.....	77
Table 10-5:	Registers Associated with PWM and Timer2 .....	78
Table 11-1:	Registers Associated with SPI Operation.....	83
Table 11-2:	I <sup>2</sup> C Bus Terminology .....	84
Table 11-3:	Data Transfer Received Byte Actions .....	89
Table 11-4:	Registers Associated with I <sup>2</sup> C Operation .....	92
Table 12-1:	Baud Rate Formula .....	97
Table 12-2:	Registers Associated with Baud Rate Generator .....	97
Table 12-3:	Baud Rates for Synchronous Mode .....	98
Table 12-4:	Baud Rates for Asynchronous Mode (BRGH = 0) .....	98
Table 12-5:	Baud Rates for Asynchronous Mode (BRGH = 1) .....	99
Table 12-6:	Registers Associated with Asynchronous Transmission.....	102

# PIC16C6X

Table 12-7:	Registers Associated with Asynchronous Reception.....	104	Table 20-1:	Cross Reference of Device Specs for Oscillator Configurations and Frequencies of Operation (Commercial Devices) .....	189
Table 12-8:	Registers Associated with Synchronous Master Transmission .....	106	Table 20-2:	External Clock Timing Requirements .....	196
Table 12-9:	Registers Associated with Synchronous Master Reception.....	107	Table 20-3:	CLKOUT and I/O Timing Requirements...	198
Table 12-10:	Registers Associated with Synchronous Slave Transmission .....	110	Table 20-4:	Reset, Watchdog Timer, Oscillator Start-up Timer, Power-up Timer, and Brown-out Reset Requirements .....	199
Table 12-11:	Registers Associated with Synchronous Slave Reception.....	110	Table 20-5:	Timer0 and Timer1 Clock Requirements..	200
Table 13-1:	Ceramic Resonators PIC16C61 .....	114	Table 20-6:	Capture/Compare/PWM Requirements (CCP1) .....	201
Table 13-2:	Capacitor Selection for Crystal Oscillator for PIC16C61 .....	114	Table 20-7:	Parallel Slave Port Requirements for the PIC16C64A/R64 Only .....	202
Table 13-3:	Ceramic Resonators PIC16C62/62A/R62/63/64/64A/R64/65/65A .....	114	Table 20-8:	SPI Mode Requirements .....	203
Table 13-4:	Capacitor Selection for Crystal Oscillator for PIC16C62/62A/R62/63/64/64A/R64/65/65A .....	114	Table 20-9:	I <sup>2</sup> C Bus Start/Stop Bits Requirements.....	204
Table 13-5:	Time-out in Various Situations, PIC16C61/62/64/65 .....	118	Table 20-10:	I <sup>2</sup> C Bus Data Requirements .....	205
Table 13-6:	Time-out in Various Situations, PIC16C62A/R62/63/64A/R64/65A .....	118	Table 22-1:	Cross Reference of Device Specs for Oscillator Configurations and Frequencies of Operation (Commercial Devices) .....	209
Table 13-7:	Status Bits and Their Significance, PIC16C61 .....	118	Table 22-2:	External Clock Timing Requirements .....	215
Table 13-8:	Status bits and Their Significance, PIC16C62/64/65 .....	118	Table 22-3:	CLKOUT and I/O Timing Requirements...	216
Table 13-9:	Status Bits and Their Significance for PIC16C62A/R62/63/64A/R64/65A .....	119	Table 22-4:	Reset, Watchdog Timer, Oscillator Start-up Timer and Power-up Timer Requirements .....	217
Table 13-10:	Reset Condition for Special Registers on PIC16C61/62/64/65 .....	119	Table 22-5:	Timer0 and Timer1 Clock Requirements..	218
Table 13-11:	Reset Condition for Special Registers on PIC16C62A/R62/63/64A/R64/65A .....	119	Table 22-6:	Capture/Compare/PWM Requirements (CCP1 and CCP2).....	219
Table 13-12:	Initialization Conditions for all Registers ...	120	Table 22-7:	Parallel Slave Port Requirements .....	220
Table 14-1:	Opcode Field Descriptions.....	131	Table 22-8:	SPI Mode Requirements .....	221
Table 14-2:	PIC16CXX Instruction Set .....	132	Table 22-9:	I <sup>2</sup> C Bus Start/Stop Bits Requirements.....	222
Table 15-1:	PICMASTER Probe Specification.....	144	Table 22-10:	I <sup>2</sup> C Bus Data Requirements .....	223
Table 15-2:	Development System Packages.....	147	Table 22-11:	Serial Port Synchronous Transmission Requirements .....	224
Table 16-1:	Cross Reference of Device Specs for Oscillator Configurations and Frequencies of Operation (Commercial Devices).....	149	Table 22-12:	Serial Port Synchronous Receive Requirements .....	224
Table 16-2:	External Clock Timing Requirements .....	155	Table 24-1:	Cross Reference of Device Specs for Oscillator Configurations and Frequencies of Operation (Commercial Devices) .....	227
Table 16-3:	CLKOUT and I/O Timing Requirements ...	156	Table 24-2:	External Clock Timing Requirements .....	233
Table 16-4:	Reset, Watchdog Timer, Oscillator Start-up Timer and Power-up Timer Requirements .....	157	Table 24-3:	CLKOUT and I/O Timing Requirements...	234
Table 16-5:	Timer0 Clock Requirements .....	158	Table 24-4:	Reset, Watchdog Timer, Oscillator Start-up Timer, Power-up Timer, and Brown-out Reset Requirements .....	235
Table 17-1:	RC Oscillator Frequencies.....	159	Table 24-5:	Timer0 and Timer1 Clock Requirements..	236
Table 17-2:	Input Capacitance* .....	170	Table 24-6:	Capture/Compare/PWM Requirements (CCP1 and CCP2).....	237
Table 18-1:	Cross Reference of Device Specs for Oscillator Configurations and Frequencies of Operation (Commercial Devices).....	171	Table 24-7:	Parallel Slave Port Requirements for the PIC16C65A Only .....	238
Table 18-2:	External Clock Timing Requirements .....	177	Table 24-8:	SPI Mode Requirements .....	239
Table 18-3:	CLKOUT and I/O Timing Requirements ...	178	Table 24-9:	I <sup>2</sup> C Bus Start/Stop Bits Requirements.....	240
Table 18-4:	Reset, Watchdog Timer, Oscillator Start-up Timer and Power-up Timer Requirements .....	179	Table 24-10:	I <sup>2</sup> C Bus Data Requirements .....	241
Table 18-5:	Timer0 and Timer1 Clock Requirements ..	180	Table 24-11:	Serial Port Synchronous Transmission Requirements .....	242
Table 18-6:	Capture/Compare/PWM Requirements (CCP1).....	181	Table 24-12:	Serial Port Synchronous Receive Requirements .....	242
Table 18-7:	Parallel Slave Port Requirements for the PIC16C64 Only .....	182	Table E-1:	PIC16C5X Family of Devices .....	263
Table 18-8:	SPI Mode Requirements.....	183	Table E-2:	PIC16C62X Family of Devices .....	264
Table 18-9:	I <sup>2</sup> C Bus Start/Stop Bits Requirements .....	184	Table E-3:	PIC16C6X Family of Devices .....	265
Table 18-10:	I <sup>2</sup> C Bus Data Requirements .....	185	Table E-4:	PIC16C7X Family of Devices .....	266
			Table E-5:	PIC16C8X Family of Devices .....	267
			Table E-6:	PIC17CXX Family of Devices.....	268
			Table E-7:	Pin Compatible Devices .....	269



## CONNECTING TO MICROCHIP BBS

Connect worldwide to the Microchip BBS using the CompuServe® communications network. In most cases a local call is your only expense. The Microchip BBS connection does not use CompuServe membership services, therefore **you do not need CompuServe membership to join Microchip's BBS.**

There is **no charge** for connecting to the BBS, except for a toll charge to the CompuServe access number, where applicable. You do not need to be a CompuServe member to take advantage of this connection (you never actually log in to CompuServe).

The procedure to connect will vary slightly from country to country. Please check with your local CompuServe agent for details if you have a problem. CompuServe service allows multiple users at baud rates up to 14400 bps.

The following connect procedure applies in most locations:

1. Set your modem to 8 bit, No parity, and One stop (8N1). This is not the normal CompuServe setting which is 7E1.
2. Dial your local CompuServe access number.
3. Depress **<ENTER>** and a garbage string will appear because CompuServe is expecting a 7E1 setting.
4. Type **+**, depress **<ENTER>** and Host Name: will appear.
5. Type **MCHIPBBS**, depress **< ENTER>** and you will be connected to the Microchip BBS.

In the United States, to find CompuServe's phone number closest to you, set your modem to 7E1 and dial (800) 848-4480 for 300-2400 baud or (800) 331-7166 for 9600-14400 baud connection. After the system responds with Host Name:, type

**NETWORK**, depress **< ENTER>** and follow CompuServe's directions.

For voice information (or calling from overseas), you may call (614) 457-1550 for your local CompuServe number.

### Trademarks:

The Microchip name, logo, PIC, PICMASTER, and PICSTART are registered trademarks of Microchip Technology Incorporated in the U.S.A.

MPLAB, PRO MATE, and *fuzzyLAB*, are trademarks and SQTP is a service mark of Microchip Technology Incorporated.

*fuzzyTECH* is a registered trademark of Inform Software Corporation.

IBM, IBM PC-AT are registered trademarks of International Business Machines Corp.

Pentium is a trademark of Intel Corporation.

MS-DOS and Microsoft Windows are registered trademarks of Microsoft Corporation. Windows is a trademark of Microsoft Corporation.

CompuServe is a registered trademark of CompuServe Incorporated.

All other trademarks mentioned herein are the property of their respective companies.

# PIC16C6X

---

## READER RESPONSE

It is our intention to provide you with the best documentation possible to ensure successful use of your Microchip product. If you wish to provide your comments on organization, clarity, subject matter, and ways in which our documentation can better serve you, please FAX your comments to the Technical Publications Manager at (602) 786-7578.

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

To: Technical Publications Manager  
RE: Reader Response  
From: Name \_\_\_\_\_  
Company \_\_\_\_\_  
Address \_\_\_\_\_  
City / State / ZIP / Country \_\_\_\_\_  
Telephone: (\_\_\_\_\_) \_\_\_\_\_ - \_\_\_\_\_ FAX: (\_\_\_\_\_) \_\_\_\_\_ - \_\_\_\_\_

Application (optional):

Would you like a reply? \_\_\_Y \_\_\_N

Device: PIC16C6X Literature Number: DS30234B

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? \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

NOTES:

# PIC16C6X

---

NOTES:

NOTES:

# PIC16C6X

---

NOTES:

## PIC16C6X Product Identification System

To order or to obtain information, e.g., on pricing or delivery, please use the listed part numbers, and refer to the factory or the listed sales offices.

PART NO.	-XX	X	/XX	XXX	
					<b>Pattern:</b> 3-Digit Pattern Code for QTP (blank otherwise)
					<b>Package:</b>
					L = PLCC
					P = PDIP
					SO = SOIC (Gull Wing, 300 mil body)
					PQ = MQFP (Metric PQFP)
					TQ = TQFP
					JW* = Windowed Cerdip
					SS = Shrink SOIC (Gull Wing, 300 mil body)
					<b>Temperature Range:</b>
					- = 0°C to +70°C (T for tape/reel)
					I = -40°C to +85°C (S for tape/reel)
					E = -40°C to +125°C
					<b>Frequency Range:</b>
					04 = 200 KHz (PIC16C6X-04)
					04 = 4 MHz
					10 = 10 MHz
					20 = 20 MHz
					<b>Device:</b>
					PIC16C6X :V <sub>DD</sub> range 4.0V to 6.0V
					PIC16C6XT :V <sub>DD</sub> range 4.0V to 6.0V (Tape and Reel)
					PIC16LC6X :V <sub>DD</sub> range 2.5V to 6.0V
					PIC16LC6XT :V <sub>DD</sub> range 2.5V to 6.0V (Tape and Reel)
					PIC16CR6X :V <sub>DD</sub> range 4.0V to 6.0V
					PIC16CR6XT :V <sub>DD</sub> range 4.0V to 6.0V (Tape and Reel)
					PIC16LCR6X :V <sub>DD</sub> range 2.5V to 6.0V
					PIC16LCR6XT :V <sub>DD</sub> range 2.5V to 6.0V

**Examples:**

a) PIC16C61 - 04/P 301 = Commercial temp., PDIP package, 4 MHz, normal V<sub>DD</sub> limits, QTP pattern #301

b) PIC16LC64 - 04I/SO = Industrial temp., SOIC package, 4 MHz, extended V<sub>DD</sub> limits

c) PIC16C65 - 10E/P = Automotive temp., PDIP package, 10 MHz, normal V<sub>DD</sub> limits

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

## Sales and Support

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

1. Your local Microchip sales office.
2. The Microchip Corporate Literature Center U.S. FAX: (602) 786-7277
3. The Microchip's Bulletin Board, via your local CompuServe number (CompuServe membership NOT required).

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

For latest version information and upgrade kits for Microchip Development Tools, please call 1-800-755-2345 or 1-602-786-7302.

---

---

# WORLDWIDE SALES & SERVICE

---

---

## AMERICAS

### Corporate Office

Microchip Technology Inc.  
2355 West Chandler Blvd.  
Chandler, AZ 85224-6199  
Tel: 602 786-7200 Fax: 602 786-7277  
Technical Support: 602 786-7627  
Web: <http://www.mchip.com/microchip>

### Atlanta

Microchip Technology Inc.  
500 Sugar Mill Road, Suite 200B  
Atlanta, GA 30350  
Tel: 770 640-0034 Fax: 770 640-0307

### Boston

Microchip Technology Inc.  
5 Mount Royal Avenue  
Marlborough, MA 01752  
Tel: 508 480-9990 Fax: 508 480-8575

### Chicago

Microchip Technology Inc.  
333 Pierce Road, Suite 180  
Itasca, IL 60143  
Tel: 708 285-0071 Fax: 708 285-0075

### Dallas

Microchip Technology Inc.  
14651 Dallas Parkway, Suite 816  
Dallas, TX 75240-8809  
Tel: 214 991-7177 Fax: 214 991-8588

### Dayton

Microchip Technology Inc.  
Suite 150  
Two Prestige Place  
Miamisburg, OH 45342  
Tel: 513 291-1654 Fax: 513 291-9175

### Los Angeles

Microchip Technology Inc.  
18201 Von Karman, Suite 1090  
Irvine, CA 92715  
Tel: 714 263-1888 Fax: 714 263-1338

### New York

Microchip Technology Inc.  
150 Motor Parkway, Suite 416  
Hauppauge, NY 11788  
Tel: 516 273-5305 Fax: 516 273-5335

### San Jose

Microchip Technology Inc.  
2107 North First Street, Suite 590  
San Jose, CA 95131  
Tel: 408 436-7950 Fax: 408 436-7955

## ASIA/PACIFIC

### Hong Kong

Microchip Technology  
Unit No. 3002-3004, Tower 1  
Metroplaza  
223 Hing Fong Road  
Kwai Fong, N.T. Hong Kong  
Tel: 852 2 401 1200 Fax: 852 2 401 3431

### Korea

Microchip Technology  
168-1, Youngbo Bldg. 3 Floor  
Samsung-Dong, Kangnam-Ku,  
Seoul, Korea  
Tel: 82 2 554 7200 Fax: 82 2 558 5934

### Singapore

Microchip Technology  
200 Middle Road  
#10-03 Prime Centre  
Singapore 188980  
Tel: 65 334 8870 Fax: 65 334 8850

### Taiwan

Microchip Technology  
10F-1C 207  
Tung Hua North Road  
Taipei, Taiwan, ROC  
Tel: 886 2 717 7175 Fax: 886 2 545 0139

## EUROPE

### United Kingdom

Arizona Microchip Technology Ltd.  
Unit 6, The Courtyard  
Meadow Bank, Furlong Road  
Bourne End, Buckinghamshire SL8 5AJ  
Tel: 44 1 628 851077 Fax: 44 1 628 850259

### France

Arizona Microchip Technology SARL  
2 Rue du Buisson aux Fraises  
91300 Massy - France  
Tel: 33 1 69 53 63 20 Fax: 33 1 69 30 90 79

### Germany

Arizona Microchip Technology GmbH  
Gustav-Heinemann-Ring 125  
D-81739 Muenchen, Germany  
Tel: 49 89 627 144 0 Fax: 49 89 627 144 44

### Italy

Arizona Microchip Technology SRL  
Centro Direzionale Colleoni  
Palazzo Pegaso Ingresso No. 2  
Via Paracelso 23, 20041  
Agrate Brianza (MI) Italy  
Tel: 39 39 689 9939 Fax: 39 39 689 9883

### JAPAN

Microchip Technology Intl. Inc.  
Benex S-1 6F  
3-18-20, Shin Yokohama  
Kohoku-Ku, Yokohama  
Kanagawa 222 Japan  
Tel: 81 45 471 6166 Fax: 81 45 471 6122

1/05/96



All rights reserved. © 1996, Microchip Technology Incorporated, USA.

Information contained in this publication regarding device applications and the like is intended for suggestion only and may be superseded by updates. No representation or warranty is given and no liability is assumed by Microchip Technology Incorporated with respect to the accuracy or use of such information, or infringement of patents or other intellectual property rights arising from such use, or otherwise. Use of Microchip's products as critical components in medical devices is not authorized except with express written approval by Microchip. No licenses are conveyed, implicitly or otherwise, under any intellectual property rights. The Microchip logo and name are registered trademarks of Microchip Technology Inc. All rights reserved. All other trademarks mentioned herein are the property of their respective companies.