FEATURES

High Performance RISC-like CPU
• Only 33 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
• 12-bit wide instructions
• 8-bit wide data path
• 512 - 2K x 12 on-chip EPROM program memory
• 25 - 72 x 8 general purpose registers (SRAM)
• 7 special function hardware registers
• 2 level deep hardware stack
• Direct, indirect and relative addressing modes for data and instructions

Peripheral Features
• 12 - 20 I/O pins with individual direction control
• 8 bit real time clock/counter (RTCC) with 8-bit programmable prescaler
• Power on reset
• Oscillator start-up timer
• Watchdog timer (WDT) with its own on-chip RC oscillator for reliable operation
• Security EPROM fuse for code-protection
• Power saving SLEEP mode
• EPROM fuse selectable oscillator options:
  - Low cost RC oscillator: RC
  - Standard crystal/resonator: XT
  - High speed crystal/resonator: HS
  - Power saving low frequency crystal: LP

CMOS Technology
• Low power, high speed CMOS EPROM technology
• Fully static design
• Wide operating voltage range:
  - Commercial: 2.5V to 6.25V
  - Industrial: 2.5V to 6.25V
  - Automotive: 2.5V to 6.0V
• Low power consumption
  - < 2 mA typical @ 5V, 4 MHz
  - 15 µA typical @ 3V, 32 KHz
  - < 3 µA typical standby current @ 3V, 0°C to 70°C

FIGURE A - PIN CONFIGURATIONS
1.0 GENERAL DESCRIPTION

The PIC16C5X from Microchip Technology is a family low cost, high performance, 8-bit, fully static, EPROM based CMOS microcontrollers. It employs a RISC-like architecture with only 33 single word/single cycle instructions to learn. All instructions are single cycle (200ns) except for program branches which take two cycles. The PIC16C5X delivers performance an order of magnitude higher than its competitors in similar price category. The 12-bit wide instructions are highly symmetrical resulting in 2:1 code compression over other 8-bit microcontrollers in its class. The easy to use and easy to remember instruction set reduces development time significantly.

The PIC16C5X products are equipped with special microcontroller like features that reduce system cost and power requirements. The power on reset and oscillator start up timer eliminate the need for external reset circuitry. There are four oscillator configurations to choose from, including power saving LP (Low Power) oscillator and cost saving RC oscillator. Power saving SLEEP mode, watchdog timer and code protection features improves system cost, power and reliability.

The UV-erasable cerdip-packaged versions are ideal for code development while the cost-effective One Time Programmable (OTP) versions are suitable for production in any volume. The customer can take full advantage of Microchip's price leadership in OTP microcontroller while benefiting from the OTP flexibility.

The PIC16C5X products are supported by an assembler, a software simulator, an in-circuit emulator and a production quality programmer. All the tools are supported by IBM PC and compatible machines.

1.1 APPLICATIONS

The PIC16C5X series fits perfectly in applications ranging from high speed automotive and appliance motor control to low-power remote transmitters/receivers, pointing devices, 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 for through hole or surface mounting 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 PIC16C5X series very versatile even in areas where no microcontroller use has been considered before (e.g. timer functions, replacement of "glue" logic in larger systems, co-processor applications).

### TABLE 1.0.1 - OVERVIEW OF PIC16C5X DEVICES

<table>
<thead>
<tr>
<th>Part #</th>
<th>EPROM</th>
<th>RAM*</th>
<th>I/O†</th>
<th>Package Options</th>
</tr>
</thead>
<tbody>
<tr>
<td>PIC16C54</td>
<td>512 x 12</td>
<td>32 x 8</td>
<td>13</td>
<td>18L windowed CERDIP, 18L PDIP, 18L SOIC (300 mil), 20L SSOP</td>
</tr>
<tr>
<td>PIC16C55</td>
<td>512 x 12</td>
<td>32 x 8</td>
<td>21</td>
<td>28L windowed CERDIP, 28L PDIP (600 mil), 28L PDIP (300 mil), 28L SOIC (300 mil), 28L SSOP</td>
</tr>
<tr>
<td>PIC16C56</td>
<td>1K x 12</td>
<td>32 x 8</td>
<td>13</td>
<td>18L windowed CERDIP, 18L PDIP, 18L SOIC (300 mil), 20L SSOP</td>
</tr>
<tr>
<td>PIC16C57</td>
<td>2K x 12</td>
<td>80 x 8</td>
<td>21</td>
<td>28L windowed CERDIP, 28L PDIP (600 mil), 28L PDIP (300 mil), 28L SOIC (300 mil), 28L SSOP</td>
</tr>
</tbody>
</table>

* Including special function registers.
** The industrial versions and the HS version operates for VDD range of 4.5 V to 5.5 V (see DC specs).
† Includes RTCC pin.

2.0 ARCHITECTURAL DESCRIPTION

2.1 Harvard Architecture

The PIC16C5X single-chip microcomputers are low-power, high-speed, full static CMOS devices containing EPROM, RAM, I/O, and a central processing unit on a single chip.

The architecture is based on a register file concept with separate bus and memories for data and instructions (Harvard architecture). The data bus and memory (RAM) are 8-bits wide while the program bus and program memory (EPROM) have a width of 12-bits. This concept allows a simple yet powerful instruction set designed to emphasize bit, byte and register operations under high speed with overlapping instruction fetch and execution cycles. That means that, while one instruction is executed, the following instruction is already being read from the program memory. A block diagram of the PIC16C5X series is given in Figure 2.1.1.

2.2 Clocking Scheme/Instruction Cycle

The clock input (from pin OSC1) is internally divided by four to generate four non overlapping quadrature clocks namely Q1, Q2, Q3 and Q4. Internally, PC is incremented every Q1, instruction is fetched from program memory and latched into instruction register in Q4. It is decoded and executed during the following Q1 through Q4. The clocks and instruction execution flow is shown in Figure 2.2.1.
TABLE 2.1.1 - PIN FUNCTIONS

<table>
<thead>
<tr>
<th>Name</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>RA0 - RA3</td>
<td>I/O PORT A</td>
</tr>
<tr>
<td>RB0 - RB7</td>
<td>I/O PORT B</td>
</tr>
<tr>
<td>RC0 - RC7</td>
<td>I/O PORT C (C55/57 only)</td>
</tr>
<tr>
<td>RTCC</td>
<td>Real Time Clock/Counter</td>
</tr>
<tr>
<td>MCLR</td>
<td>Master Clear</td>
</tr>
<tr>
<td>OSC1</td>
<td>Oscillator (input)</td>
</tr>
<tr>
<td>OSC2/CLKOUT</td>
<td>Oscillator (output)</td>
</tr>
<tr>
<td>VDD</td>
<td>Power supply</td>
</tr>
<tr>
<td>VSS</td>
<td>Ground</td>
</tr>
<tr>
<td>N/C</td>
<td>No (internal) Connection</td>
</tr>
</tbody>
</table>

2.3 Data Register File
The 8-bit data bus connects two basic functional elements together: the Register File composed of up to 80 addressable 8-bit registers including the I/O Ports, and an 8-bit wide Arithmetic Logic Unit. The 32 bytes of RAM are directly addressable while a "banking" scheme, with banks of 16 bytes each, is employed to address larger data memories (Figure 4.2.1). Data can be addressed direct, or indirect using the file select register (f4). Immediate data addressing is supported by special "literal" instructions which load data from program memory into the W register.

2.4 Arithmetic/Logic Unit (ALU)
The 8-bit wide ALU contains one temporary working register (W Register). It performs arithmetic and Boolean functions between data held in the W Register and any file register. It also does single operand operations on either the W register or any file register.

2.5 Program Memory
Up to 512 words of 12-bit wide on-chip program memory (EPROM) can be directly addressed. Larger program memories can be addressed by selecting one of up to four available pages with 512 words each (Figure 4.3.1). Sequencing of microinstructions is controlled via the
Program Counter (PC) which automatically increments to execute in-line programs. Program control operations, supporting direct, indirect, relative addressing modes, can be performed by Bit Test and Skip instructions, Call instructions, Jump instructions or by loading computed addresses into the PC. In addition, an on-chip two-level stack is employed to provide easy to use subroutine nesting.

3.0 PIC16C5X SERIES OVERVIEW

A wide variety of EPROM and RAM sizes, number of I/O pins, oscillator types, frequency ranges, and packaging options is available. Depending on application and production requirements the proper device option can be selected using the information and tables in this section. When placing orders, please use the "PIC16C5X Product Identification System" on the back page of this data sheet to specify the correct part number.

3.1 UV Erasable Devices

Four different device versions, as listed in Table 3.1.1, are available to accommodate the different EPROM, RAM, and I/O configurations. These devices are optimal for prototype development and pilot series. The desired oscillator configuration is EPROM programmable as "RC", "XT", "HS" or "LP". An erased device is configured as "RC" type by default. Depending on the selected oscillator type and frequency, the operating supply voltage must be within the same range as a OTP/QTP part would be specified for.

The available PIC development tools "PICPRO™ and PICPRO™II can program all PIC16C5X devices for prototyping and pilot series up to low-volume production.

3.2 One-Time-Programmable (OTP) Devices

The availability of OTP devices is especially useful for customers expecting frequent code changes and updates. OTP devices have the oscillator type pre-configured by the factory, and they are tested only for this special configuration (including voltage and frequency ranges, current consumption). Table 3.2.1 gives an overview about devices available now and planned for future release.

The program EPROM is erased, allowing the user to write the application code into it. In addition, the watchdog timer can be disabled, and/or the code protection logic can be activated by programming special EPROM fuses. The sixteen special EPROM bits for ID code storage are also user programmable.

3.3 Quick-Turnaround-Production (QTP) Devices

Microchip offers a QTP Programming Service for factory production orders. This service is made available for users who chose not to program a medium to high quantity of units and whose code patterns have stabilized. The devices are identical to the OTP devices (see Table 3.2.1) but with all EPROM locations and fuse options already programmed by the factory. Certain code and prototype verification procedures do apply before production shipments are available. Please contact your Microchip Technology sales office for more details.
4.0 OPERATIONAL REGISTER FILES

4.1 f0 Indirect Data Addressing

This is not a physically implemented register. Addressing f0 calls for the contents of the File Select Register to be used to select a file register. For example, in the instruction ADDWF f0, W will add the contents of the register pointed to by the FSR (f4) to the content of the W Register and place the result in W.

If f0 itself is read through indirect addressing (i.e. FSR = 0h), then OOh is read. If f0 is written to via indirect addressing, the result will be a NOP.

4.2 f1 Real Time Clock/Counter Register (RTCC)

This register can be loaded and read by the program as any other register. In addition, its contents can be incremented by an external signal edge applied to the RTCC pin, or by the internal instruction cycle clock (CLKOUT = fosc/4). Figure 4.1.1 is a simplified block diagram of RTCC.

An 8-bit prescaler can be assigned to the RTCC by writing the proper values to the PSA bit and the PS bits in the OPTION register. OPTION register is a special register (not mapped in data memory) addressable using the 'OPTION' instruction. See section 7.5 for details. If the prescaler is assigned to the RTCC, instructions writing to f1 (e.g. CLRF f1, or BSF f1,5,... etc.) clear the prescaler.

The bit "RTS" (RTCC signal Source) in the OPTION register determines, if f1 is incremented internally or externally.

As long as clocks are applied to the RTCC (from internal or external source, with or without prescaler), f1 keeps incrementing and just rolls over when the value "FFh" is reached. All increment pulses for f1 are delayed by two instruction cycles. After writing to f1, for example, no increment takes place for the following two instruction cycles. This is independent if internal or external clock source is selected. If a prescaler is assigned to the RTCC, the output of the prescaler will be delayed by two cycles before f1 is incremented. This is true for instructions that either write to or read-modify-write RTCC (e.g. MOVF f1, CLRF f1). For applications where RTCC needs to be tested for zero without affecting its count, use of MOVF f1, W instruction is recommended. Timing diagrams in Figure 4.2.2 show RTCC read, write and increment timing.

4.2.1 USING RTCC WITH EXTERNAL CLOCK

When external clock input is used for RTCC, it is synchronized with the internal phase clocks. Therefore, the external clock input must meet certain requirements.
FIGURE 4.2.1 - PIC16C5X DATA MEMORY MAP

FILE ADDRESS
00 7 6 5 4 3 2 1 0
01 INDIRECT ADDR. (*)
02 RTCC CALL
03 PC RETLW
04 STACK 1
05 ADDRESS
06 10 9 8 7 6 5 4 3 2 1 0
07 10 9 8 7 6 5 4 3 2 1 0
08 STACK 2
09 7 6 5 4 3 2 1 0
0A TRISB
0B TRISB
0C TRISC
0D TRISC
0E TRISC
0F TRISC

(*) NOT A PHYSICALLY IMPLEMENTED REGISTER. SEE SECTION 4.0 OR DETAILS

(**) FILE 17 IS A GENERAL PURPOSE REGISTER ON THE PIC16C54/C56

(***) BANK 0 IS AVAILABLE ON ALL MICROCONTROLLERS WHILE BANK 1 TO BANK 3 ARE ONLY AVAILABLE ON THE PIC16C57. (SEE SECTION 4.6 FOR DETAILS)
Also there is some delay from the occurrence of the external clock edge to the actual incrementing of RTCC. Referring to Figure 4.1.1, the synchronization is done after the prescaler. The output of the prescaler is sampled twice in every instruction cycle to detect rising or falling edges. Therefore, it is necessary for Psout to be high for at least 2 tosc and low for at least 2 tosc where tosc = oscillator time period.

When no prescaler is used, Psout (Prescaler output, see Figure 5) is the same as RTCC clock input and therefore the requirements are:

\[ TRTH = \frac{RTCC \text{ high time}}{2} + 20 \text{ ns} \]
\[ TRTL = \frac{RTCC \text{ low time}}{2} + 20 \text{ ns} \]

When prescaler is used, the RTCC input is divided by the asynchronous ripple-counter-type prescaler and so the prescaler output is symmetrical. Then:

\[ PSOUT \text{ high time} = \frac{N \cdot TRT}{2} \]
\[ PSOUT \text{ low time} = \frac{N \cdot TRT}{2} \]

where \( TRT = RTCC \text{ input period} \) and \( N = \text{ prescale value} \) (2, 4, ..., 256). The requirement is, therefore:

\[ \frac{4 \cdot \text{tosc} + 40 \text{ ns}}{N} \geq 2 \text{ tosc} + 20 \text{ ns} \]

\[ \text{or } TRT \geq \frac{4 \cdot \text{tosc} + 40 \text{ ns}}{2} \]

The requirement is then:

\[ TRTH = \text{RTCC high time} \geq \frac{2 \cdot \text{tosc} + 20 \text{ ns}}{N} \]
\[ TRTL = \text{RTCC low time} \geq \frac{2 \cdot \text{tosc} + 20 \text{ ns}}{N} \]

The user will notice that no requirement on RTCC high or low time is specified. However, if the high time or low time on RTCC is too small then the pulse may not be detected, hence a minimum high or low time of 10 ns is required. In summary, the RTCC input requirements are:

\[ TRT = \frac{RTCC \text{ period}}{4} \cdot \frac{(4 \cdot \text{tosc} + 40 \text{ ns})}{N} \]
\[ TRTH = \text{RTCC high time} \geq \frac{10 \text{ ns}}{2} \]
\[ TRTL = \text{RTCC low time} \geq \frac{10 \text{ ns}}{2} \]

Delay from external clock edge: 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 RTCC is actually incremented. Referring to Figure 4.2.3, the reader can see that this delay is between 3 tosc and 7 tosc. Thus, for example, measuring the interval between two edges (e.g. period) will be accurate within \( \pm 4 \text{ tosc} \) (\( \pm 200 \text{ ns} @ 20 \text{ MHz} \)).

### 4.3 f2 Program Counter

The program counter generates the addresses for up to 2048 x 12 on-chip EPROM cells containing the program instruction words (Figure 4.3.1).

Depending on the device type, the program counter and its associated two-level hardware stack is 9 - 11-bits wide.

**TABLE 4.3.1 - PROGRAM COUNTER STACK WIDTH**

<table>
<thead>
<tr>
<th>Part #</th>
<th>PC width</th>
<th>Stack width</th>
</tr>
</thead>
<tbody>
<tr>
<td>PIC16C54/PIC16C55</td>
<td>9 bit</td>
<td>9 bit</td>
</tr>
<tr>
<td>PIC16C56</td>
<td>10 bit</td>
<td>10 bit</td>
</tr>
<tr>
<td>PIC16C57</td>
<td>11 bit</td>
<td>11 bit</td>
</tr>
</tbody>
</table>

The program counter is set to all "1's upon a RESET condition. During program execution it is auto incremented with each instruction unless the result of that instruction changes the PC itself.

**FIGURE 4.2.2A - RTCC TIMING: INT CLOCK/NO PRESCALE**

**FIGURE 4.2.2B - RTCC TIMING: INT CLOCK/PRESCALE 1:2**
a) "GOTO" instructions allow the direct loading of the lower 9 program counter bits (PC<8:0>). In case of PIC16C56/PIC16C57, the upper two bits of PC (PC<10:9>) are loaded with page select bits PA1, PA0 (bits 6,5 status register). Thus GOTO allows jump to any location on any page.

b) "CALL" instructions load the lower 8-bit of the PC directly while the ninth bit is cleared to "0". The PC value, incremented by one, will be pushed into the stack. In case of PIC16C56, PIC16C57, the upper two bits of PC (PC<10:9>) are loaded with Page Select bits PA1, PA0 (bits 6,5 status register).

c) "RETLW" instructions load the program counter with the top of stack contents.

d) If PC is the destination in any instruction (e.g. MOVWF 2, ADDWF 2, or BSF 2,5) then the computed 8-bit result will be loaded into the low 8-bits of program counter. The ninth bit of PC will be cleared. In case of PIC16C56/PIC16C57, PC<10:9> will be loaded with Page Select bits PA1, PA0 (bits 6,5 in status register).

It should be noted that because bit 8 (ninth bit) of PC is cleared in CALL instruction or any instruction which writes to the PC (e.g. MOVWF 2), all subroutine calls or computed jumps are limited to the first 256 locations of any program memory page (512 words long).

**MORE ON PROGRAM MEMORY PAGE SELECT (PIC16C56/PIC16C57 ONLY):**

Incrementing the program counter when it is pointing to the last address of a selected memory page is also possible and will cause the program to continue in the next higher page. However, the page pre-select bits in f3 will not be changed, and the next "GOTO", "CALL", "ADDWF 2", "MOVWF 2" instruction will return to the previous page, unless the page pre-select bits have been updated under program control. For example, a "NOP" at location "1FF" (page 0) increments the PC to "200" (page 1). A "GOTO xxx" at "200" will return the program to address "xxx" on page "0" (assuming that the page preselect bits in file register f3 are "0").

Upon a RESET condition, page 0 is pre-selected while the program counter addresses the last location in the last page. Thus, a "GOTO" instruction at this location will automatically cause the program to continue in page 0.

### 4.4 Stack

The PIC16C5X series employs a two level hardware push/pop stack (Figure 4.3.1).

**CALL** instructions push the current program counter value, incremented by "1", into stack level 1. Stack level 1 is automatically pushed to level 2. If more than 2 subsequent "CALL"s are executed, only the most recent two return addresses are stored.

For the PIC16C56 and PIC16C57, the page preselect bits of f3 will be loaded into the most significant bits of the program counter. The ninth bit is always cleared to "0" upon a CALL instruction. This means that subroutine entry addresses have to be located always within the lower half of a memory page (addresses 000-0FF, 200-2FF, 400-4FF, 600-6FF). However, as the stack has always the same width as the PC, subroutines can be called from anywhere in the program.

**RETLW** instructions load the contents of stack level 1 into the program counter while stack level 2 gets copied into level 1. If more than 2 subsequent "RETLW"s are executed, the stack will be filled with the address previously stored in level 2. For the PIC16C56 and PIC16C57, the return will be always to the page from where the subroutine was called, regardless of the current setting of the page pre-select bits in file register f3. Note that the W register will be loaded with the literal value specified in the RETLW instruction. This is particularly useful for the implementation of "data" tables within the program memory.

**FIGURE 4.2.3 - RTCC TIMING WITH EXTERNAL CLOCK**

---

© 1992 Microchip Technology Inc.
FIGURE 4.3.1 - PROGRAM MEMORY ORGANIZATION

GOTO, CALL, INST WITH PC AS DESTINATION ............... FROM f3, BIT 6 (PIC16C57 ONLY)
GOTO, CALL, INST WITH PC AS DESTINATION ............... FROM f3, BIT 5 (PIC16C56/C57 ONLY)
GOTO .................................................. DIRECT FROM INSTRUCTION WORD
CALL, INST WITH PC AS DESTINATION ......... ALWAYS '0'
GOTO, CALL ........................................... DIRECT FROM INSTRUCTION WORD
INST WITH PC AS DESTINATION ............. FROM ALU

MAX. EPROM ADDRESS FOR:

- PIC16C54/PIC16C55
- PIC16C56
- PIC16C57
4.5 f3 Status Word Register

This register contains the arithmetic status of the ALU, the RESET status, and the page preselect bits for larger program memories than 512 words (PIC16C56, PIC16C57).

The status register (f3) can be destination for any instruction like any other register. However, the status bits are set after the following write. Furthermore, TO and PD bits are not writable. Therefore, the result of an instruction with status register as destination may be different than intended. For example, CLRF f3 will clear all bits except for TO and PD and then set the Z bit and leave status register as 0000U100 (where U = unchanged).

It is recommended, therefore, that only BCF, BSF and MOVWF instructions are used to alter the status registers because these instructions do not affect any status bit.

For other instructions, affecting any status bits, see section "Instruction Set Summary" (Table 10.0.1).

FIGURE 4.5.1 - STATUS WORD REGISTER f3

<table>
<thead>
<tr>
<th>7</th>
<th>6</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>PA2</td>
<td>PA1</td>
<td>PA0</td>
<td>TO</td>
<td>PD</td>
<td>Z</td>
<td>DC</td>
<td>C</td>
</tr>
</tbody>
</table>

RESET CONDITION:
PA2, PA1, PA0 cleared to '0'.
TO, PD are set or reset as shown in Table 4.5.1.1.
Z, DC, C are unknown on power on reset and unchanged in any other reset.

CARRY/BORROW BIT:
For ADDWF and SUBWF instructions, this bit is set if there is a carry out from the most significant bit of the resultant. Note that a subtraction is executed by adding the two's complement of the second operand. For rotate (RRF, RLF) instructions, this bit is loaded with either the high or low order bit of the source register.

DIGIT CARRY/BORROW BIT:
For ADDWF and SUBWF instructions, this bit is set if there is a carry out from the 4th low order bit of the resultant.

ZERO BIT:
Set if the result of an arithmetic or logic operation is zero.

POWER DOWN BIT:
Set to "1" during power up or by a CLRWDT command. This bit is reset to "0" by a SLEEP instruction.

TIME-OUT BIT:
Set to "1" during power up and by the CLRWDT and SLEEP command. This bit is reset to "0" by a watchdog timer time out.

PIC16C54/C55:
Two general purpose read/write bits

PIC16C56:
BIT 5... Page preselect bit
  0 = Page 0 (000 - 1FF)
  1 = Page 1 (200 - 3FF)

BIT 6... General purpose read/write bit

PIC16C57:
Two page preselect bits
  00 = Page 0 (000 - 1FF)
  01 = Page 1 (200 - 3FF)
  10 = Page 2 (400 - 5FF)
  11 = Page 3 (600 - 7FF)

BIT 7: General purpose read/write bit
(reserved for future use)
4.5.1 CARRY/BORROW AND DIGIT CARRY/BORROW BITS:

The Carry bit (C) is a carry out in addition operation (ADDWF) and a borrow out in subtract operation (SUBWF).

It is also affected by RRF and RLF instructions. The following examples explain carry/borrow bit operation:

; SUBWF Example #1
clrf 0x20 ; f(20h)=0
movlw 1 ; wreg=1
subwf 0x20 ; f(20h)=f(20h)-wreg=0-1=FFh
; Carry=0: Result is negative

; SUBWF Example #2
movlw 0xFF ;
movwf 0x20 ; f(20h)=FFh
clrw ; wreg=0
subwf 0x20 ; f(20h)=f(20h)-wreg=FFh-0=FFh
; Carry=1: Result is positive

The digit carries operates in the same way as the carry bit, i.e. it is a borrow in subtract operation.

4.5.2 TIME OUT AND POWER DOWN STATUS BITS (TO, PD)

The "TO" and "PD" bits in the status register F3 can be tested to determine if a RESET condition has been caused by a watchdog timer time-out, a power-up condition, or a wake-up from SLEEP by the watchdog timer or MCLR pin.

These status bits are only affected by events listed in Table 4.5.2.1.

TABLE 4.5.2.1 - EVENTS AFFECTING PD/TO STATUS BITS

<table>
<thead>
<tr>
<th>Event</th>
<th>TO</th>
<th>PD</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Power-up</td>
<td>1</td>
<td>1</td>
<td>No effect on PD</td>
</tr>
<tr>
<td>WDT Timeout</td>
<td>0</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>SLEEP instruction</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>CLRWDT instruction</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>

Note: A WDT timeout will occur regardless of the status of the TO bit. A SLEEP instruction will be executed, regardless of the status of the PD bit. Table 4.5.2.2 reflects the status of PD and TO after the corresponding event.

<table>
<thead>
<tr>
<th>PD</th>
<th>TO</th>
<th>RESET was caused by</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>WDT wake-up from SLEEP</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>WDT time-out (not during SLEEP)</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>MCLR wake-up from SLEEP</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>Power-up</td>
</tr>
<tr>
<td>X</td>
<td>X</td>
<td>= Low pulse on MCLR input</td>
</tr>
</tbody>
</table>

Note: The PD and TO bit maintain their status (X) until an event of Table 4.5.2.1 occurs. A low pulse on the MCLR input does not change the PD and TO status bits.

4.5.3 PROGRAM PAGE PRESELECT (PIC16C56, PIC16C57 ONLY)

Bits 5-6 of the STATUS register are defined as PAGE address bits PA0 - PA1, and are used to preselect a program memory page. When executing a GOTO, CALL, or an instruction with PC (f2) as destination (e.g. MOVWF 2), PA0 - PA1 are loaded into bit A9-A10 of the program counter, selecting one of the available program memory pages. The direct address specified in the instruction is only valid within this particular memory page.

RETLW instructions do not change the page preselect bits.

Upon a RESET condition, PA0-PA2 are cleared to "O"s.

4.6 f4 File Select Register (FSR)

PIC16C54/C55/C56

Bits 0-4 select one of the 32 available file registers in the indirect addressing mode (that is, calling for file f0 in any of the file oriented instructions).

Bits 5-7 of the FSR are read-only and are always read as "one"s.

If no indirect addressing is used, the FSR can be used as a 5-bit wide general purpose register.

PIC16C57 ONLY

Bit 5 and 6 of the FSR select the current data memory bank (Figure 4.2.1).

The lower 16 bytes of each bank are physically identical and are always selected when bit 4 of the FSR (in case of indirect addressing) is "0", or bit 4 of the direct file register address of the currently executing instruction is "0" (e.g. MOVWF 08).

Only if bit 4 in the above mentioned cases is "1", bits 5 and 6 of the FSR select one of the four available register banks with 16 bytes each.

Bit 7 is read-only and is always read as "one."
5.0 I/O REGISTERS (PORTS)

The I/O registers can be written and read under program control like any other register of the register file. However, "read" instructions (e.g. MOVF 6,W) always read the I/O pins, regardless if a pin is defined as "input" or "output." Upon a RESET condition, all I/O ports are defined as "input" (= high impedance mode) as the I/O control registers (TRISA, TRISB, TRISC) are all set to "ones."

The execution of a "TRIS f" instruction with corresponding "zeros" in the W-register is necessary to define any of the I/O pins as output.

5.1 f5 (Port A)

4-bit I/O register. Low order 4 bits only are used (RA0 - RA3). Bit 4 - 7 are unimplemented and read as "zeros."

5.2 f6 (Port B)

8-bit I/O register.

5.3 f7 (Port C)

PIC16C55/C57: 8-bit I/O register.
Pic16C54/C56: General purpose register.

5.4 I/O INTERFACING

The equivalent circuit for an I/O port bit is shown in Figure 5.4.1. All ports may be used for both input and output operations. For input operations these ports are non-latching. Any input must be present until read by an input instruction (e.g. MOVF 6, W). The outputs are latched and remain unchanged until the output latch is rewritten. To use a port pin as output, the corresponding direction control bit (in TRISA, TRISB, TRISC) must be set to zero. For use as an input, the corresponding TRIS bit must be "one". Any I/O pin can be programmed individually as input or output.

FIGURE 5.4.1 - EQUIVALENT CIRCUIT FOR A SINGLE I/O PIN

![Equivalent Circuit for a Single I/O Pin](image-url)
5.5 I/O PROGRAMMING CONSIDERATIONS

5.5.1 BIDIRECTIONAL I/O PORTS

a) Some instructions operate internally as read followed by write operations. The BCF and BSF instructions, for example, read the entire port into the CPU, execute the bit operation, and re-output the result. Caution must be used when these instructions are applied to a port where one or more pins are used as input/outputs. For example, a BSF operation on bit 5 of f6 (Port B) will cause all eight bits of f6 to be read into the CPU. Then the BSF operation takes place on bit 5 and f6 is re-output to the output latches. If another bit of f6 is used as a bidirectional I/O pin (say bit 0) 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 re-written 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 bit 0 is switched into output mode later on, the content of the data latch may now be unknown.

b) A pin actively outputting a "0" or "1" 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.

For "wired-or" outputs (assuming negative logic), it is recommended to use external pull-up resistors on the corresponding pins. The pin should be left in high-impedance mode, unless a "0" has to be output. Thus, external devices can drive this pin "0" as well. "Wired-and" outputs can be realized in the same way, but with external pull-down resistors and only actively driving the "1" level from the PIC. The resistor values are user selectable, but should not force output currents above the specified limits (see DC Characteristics).

5.5.2 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 (see figure 5.4.2.1). 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 an other instruction not accessing this I/O port.

5.5.3 OPERATION IN NOISY ENVIRONMENT

In noisy application environments, such as keyboards which are exposed to ESD (Electro Static Discharge), register contents can get corrupted due to noise spikes. The on-chip watchdog timer will take care of all situations involving program sequence "lock-ups." However, if an I/O control register gets corrupted, the program sequence may still be executed properly although an input pin may have switched unintentionally to an output. In this case, the program would always read the same value on this pin. This may result, for example, in a keyboard "lock-up" situation without leading to a watchdog timer timeout. Thus, it is recommended to redefine all I/O pins in regular time intervals (inputs as well as outputs). The optimal strategy is to update the I/O control register every time before reading input data or writing output data.

FIGURE 5.5.2.1 - I/O PORT READ/WRITE TIMING

This example shows write to port B followed by a read from port B. Note that the data setup time = (0.25 TCY - TPD) where TCY = instruction cycle. Therefore, at higher clock frequencies, write followed by a read may be problematic.
6.0 GENERAL PURPOSE REGISTERS

PIC16C54/C55/C56:

f08h - f1Fh: are general purpose register files.

PIC16C57 only:

f08h - f0Fh: are general purpose register files which
are always selected, independent of bank select.
f10h - f1Fh: general purpose register files in memory
bank 0.
f20h - f2Fh: physically identical to f00 - f0F.
f30h - f3Fh: general purpose register files in memory
bank 1.
f40h - f4Fh: physically identical to f00 - f0F.
f50h - f5Fh: general purpose register files in memory
bank 2.
f60h - f6Fh: physically identical to f00 - f0.
f70h - f7Fh: general purpose register files in memory
bank 3.

7.0 SPECIAL PURPOSE REGISTERS

7.1 W Working Register

Holds second operand in two operand instructions and/
or supports the internal data transfer.

7.2 TRISA I/O Control Register For Port A (f5)

Only bits 0 - 3 are available. The corresponding I/O port (f5) is only 4-bit wide.

7.3 TRISB I/O Control Register For Port B (f6)

7.4 TRISC I/O Control Register For Port C (f7)

The I/O control registers will be loaded with the content
of the W register by executing of the TRIS f instruction.
A "1" in the I/O control register puts the corresponding
I/O pin into a high impedance mode. A "0" puts the
contents of file register f5, f6, or f7, respectively, out on
the selected I/O pins.
These registers are "write-only" and are set to all "ones"
upon a RESET condition.

7.5 OPTION Prescaler/RTCC Option Register

Defines prescaler assignment (RTCC or WDT), pres­
cal value, signal source and signal edge for the RTCC.
The OPTION register is "write-only" and is 6 bit wide.
By executing the "OPTION" instruction, the contents of
the "W" register will be transferred to the option register.
Upon a RESET condition, the option register is set to all
"ones."

FIGURE 7.5.1 - OPTION REGISTER

<table>
<thead>
<tr>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>RTS</td>
<td>RTE</td>
<td>PSA</td>
<td>PS2</td>
<td>PS1</td>
<td>PS0</td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>RESET VALUE: 111111b</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>PRESCALER VALUE</th>
<th>RTCC RATE</th>
<th>WDT RATE</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 0 0</td>
<td>1 : 2</td>
<td>1 : 1</td>
</tr>
<tr>
<td>0 0 1</td>
<td>1 : 4</td>
<td>1 : 2</td>
</tr>
<tr>
<td>0 1 0</td>
<td>1 : 8</td>
<td>1 : 4</td>
</tr>
<tr>
<td>0 1 1</td>
<td>1 : 16</td>
<td>1 : 8</td>
</tr>
<tr>
<td>1 0 0</td>
<td>1 : 32</td>
<td>1 : 16</td>
</tr>
<tr>
<td>1 0 1</td>
<td>1 : 64</td>
<td>1 : 32</td>
</tr>
<tr>
<td>1 1 0</td>
<td>1 : 128</td>
<td>1 : 64</td>
</tr>
<tr>
<td>1 1 1</td>
<td>1 : 256</td>
<td>1 : 128</td>
</tr>
</tbody>
</table>

PRESCALER ASSIGNMENT BIT:
0 .... RTCC
1 .... WDT

RTCC SIGNAL EDGE:
0 .... INCREMENT ON LOW-TO-HIGH TRANSITION ON RTCC PIN
1 .... INCREMENT ON HIGH-TO-LOW TRANSITION ON RTCC PIN

RTCC SIGNAL SOURCE:
0 .... INTERNAL INSTRUCTION CYCLE CLOCK (CLKOUT)
1 .... TRANSITION ON RTCC PIN
8.0 RESET CONDITION

A RESET condition can be caused by applying power to the chip (power-up), pulling the MCLR input "low", or by a Watchdog timer timeout. The device will stay in RESET as long as the oscillator start-up timer (OST) is active or the MCLR input is "low."

The oscillator start-up timer is activated as soon as MCLR input is sensed to be high. This implies that in case of power on reset with MCLR tied to VDD the OST starts from power-up. In case of WDT time-out, it will start at the end of the time-out (since MCLR is high). In case of MCLR reset, the OST will start when MCLR goes high. The nominal OST time-out period is 18 ms. See section 13.0 for detailed information on OST and power on reset.

During a RESET condition the state of the PIC is defined as:

- The oscillator is running, or will be started (power-up or wake-up from SLEEP).
- All I/O port pins (RA0 - RA3, RB0 - RB7, RC0 - RC7) are put into the high-impedance state by setting the "TRIS" registers to all "ones" (= input mode).
- The Program Counter is set to all "ones" (1FFh in PIC16C54/55, 3FFh in PIC16C56 and 7FFh in PIC16C57).
- The OPTION register is set to all "ones".
- The Watchdog Timer and its prescaler are cleared.
- The upper three bits (page select bits) in the Status Register (F3) are cleared to "zero."
- "RC" devices only: The "CLKOUT" signal on the OSC2 pin is held at a "low" level.

9.0 PRESCALER

An 8-bit counter is available as a prescaler for the RTCC, or as a post-scaler for the watchdog timer, respectively (Figure 9.0.1). For simplicity, this counter is being referred to as "prescaler" throughout this data sheet. Note that there is only one prescaler available which is mutually exclusively shared between the RTCC and the watchdog timer. Thus, a prescaler assignment for the RTCC means that there is no prescaler for the watchdog timer, and vice versa.

The PSA and PSO-PS2 bits in the OPTION register determine the prescaler assignment and pre-scale ratio.

When assigned to the RTCC, all instructions writing to the RTCC (e.g. CLR W 1, MOVWF 1, BSF 1,x ...etc.) will clear the prescaler. When assigned to WDT, a CLRWDT instruction will clear the prescaler along with the watchdog timer.

9.1 Switching Prescaler Assignment

CHANGING PRESCALER FROM RTCC TO WDT

The prescaler assignment is fully under software control, i.e., it can be changed "on the fly" during program execution. To avoid an unintended device RESET, the following instruction sequence must be executed when changing the prescaler assignment from RTCC to WDT:

1. MOVLW B'xx00xxxx' ; Select internal clock and select new prescaler value. If new prescale value is '000' or '001', then select any other prescale value temporarily.
2. OPTION
3. CLR W 1 ; Clear RTCC and prescaler.
4. MOVLW B'xxxx1xxx' ; Select WDT, do not change prescale value.
5. OPTION
6. CLRWDT ; Clears WDT and prescaler.
7. MOVLW B'xxxx1xxx' ; Select new prescale value.
8. OPTION

Step 1 and 2 are only required if an external RTCC source is used. Steps 7 and 8 are necessary only if the desired prescale value is '000' or '001'.

CHANGING PRESCALER FROM WDT TO RTCC

To change prescaler from WDT to RTCC use the following sequence:

1. CLRWDT ; Clear WDT and prescaler
2. MOVLW B'xxxx0xxx' ; Select RTCC, new prescale value and clock source.
3. OPTION ;
10.0 BASIC INSTRUCTION SET SUMMARY

Each PIC instruction is a 12-bit word divided into an OP CODE which specifies the instruction type and one or more operands which further specify the operation of the instruction. The PIC instruction set summary in Table 10.0.1 lists byte-oriented, bit-oriented, and literal and control operations.

For byte-oriented instructions, "f" represents a file register designator and "d" represents a destination designator. The file register designator specifies which one of the 32 PIC file registers is to be utilized by the instruction. For the PIC16C57, bit 5 and 6 in the FSR determine the selected register bank.

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 nine bit constant or literal value.

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. One instruction cycle consists of four oscillator periods. Thus, for an oscillator frequency of 4 MHz, the normal instruction execution time is 1 µsec. If a conditional test is true or the program counter is changed as a result of an instruction, the instruction execution time is 2 µsec.

Notes to Table 10.0.1

Note 1: The 9th bit of the program counter will be forced to a "zero" by any instruction that writes to the PC (f2) except for GOTO (e.g. CALL, MOVWF 2 etc.). See section 4.3 on page 8 for details.

Note 2: When an I/O register is modified as a function of itself (e.g. MOVF 6, 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 output and is driven low by an external device, the data will be written back with a '0'.

Note 3: The instruction "TRIS f", where f = 5, 6, or 7 causes the contents of the W register to be written to the tristate latches of the specified file (port). A "one" forces the pin to a high impedance state and disables the output buffers.

Note 4: If this instruction is executed on file register f1 (and, where applicable, d=1), the prescaler will be cleared if assigned to the RTCC.
### TABLE 10.0.1 - INSTRUCTION SET SUMMARY

#### BYTE-ORIENTED FILE REGISTER OPERATIONS

<table>
<thead>
<tr>
<th>Instruction-Binary (Hex)</th>
<th>Name</th>
<th>Mnemonic, Operands</th>
<th>Operation</th>
<th>Status Affected</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>0000 0100 0000 040</td>
<td>Clear W</td>
<td>CLRWF</td>
<td>f</td>
<td>0 → f</td>
<td>Z</td>
</tr>
<tr>
<td>0010 0100 0000 040</td>
<td>Clear W</td>
<td>CLRWF</td>
<td>f</td>
<td>0 → f</td>
<td>Z</td>
</tr>
<tr>
<td>0010 0100 0000 06f</td>
<td>Clear f</td>
<td>CLRF</td>
<td>f</td>
<td>0 → f</td>
<td>Z</td>
</tr>
<tr>
<td>0001 11df ffff 1cf</td>
<td>Add W and f</td>
<td>ADDWF</td>
<td>f, d</td>
<td>W + f → d</td>
<td>C, DC, Z</td>
</tr>
<tr>
<td>0000 011f ffff 06f</td>
<td>Clear f</td>
<td>CLRF</td>
<td>f</td>
<td>0 → f</td>
<td>Z</td>
</tr>
<tr>
<td>0010 0100 0000 040</td>
<td>Clear W</td>
<td>CLRWF</td>
<td>f</td>
<td>0 → W</td>
<td>Z</td>
</tr>
<tr>
<td>0010 10df ffff 2cf</td>
<td>Decrement f, Skip if Zero</td>
<td>DECFSZ</td>
<td>f, d</td>
<td>1 - 1 → d, skip if zero</td>
<td>None</td>
</tr>
<tr>
<td>0010 10df ffff 0cf</td>
<td>Decrement f</td>
<td>DECf</td>
<td>f, d</td>
<td>f - 1 → d</td>
<td>Z</td>
</tr>
<tr>
<td>0011 10df ffff 3cf</td>
<td>Increment f, Skip if Zero</td>
<td>INCF</td>
<td>f, d</td>
<td>f + 1 → d</td>
<td>Z</td>
</tr>
<tr>
<td>0000 00df ffff 1cf</td>
<td>Add W and f</td>
<td>ADDWF</td>
<td>f, d</td>
<td>W + f → d</td>
<td>C, DC, Z</td>
</tr>
<tr>
<td>0000 0000 0000 000</td>
<td>No Operation</td>
<td>NOP</td>
<td>-</td>
<td>-</td>
<td>None</td>
</tr>
<tr>
<td>0010 10df ffff 08f</td>
<td>Subtract W from f</td>
<td>SUBWF</td>
<td>f, d</td>
<td>f - W → f</td>
<td>Z</td>
</tr>
<tr>
<td>0010 10df ffff 38f</td>
<td>Swap halves f</td>
<td>SWAPF</td>
<td>f, d</td>
<td>f(0-3) ↔ f(4-7)</td>
<td>None</td>
</tr>
<tr>
<td>0000 001f ffff 02f</td>
<td>Move W to f</td>
<td>MOVWF</td>
<td>f</td>
<td>W → f</td>
<td>None</td>
</tr>
<tr>
<td>0010 10df ffff 0cf</td>
<td>Decrement f</td>
<td>DECf</td>
<td>f, d</td>
<td>f - 1 → d</td>
<td>Z</td>
</tr>
<tr>
<td>0000 0000 0000 000</td>
<td>No Operation</td>
<td>NOP</td>
<td>-</td>
<td>-</td>
<td>None</td>
</tr>
<tr>
<td>0111 00df ffff 7bf</td>
<td>Exclusive OR W and f</td>
<td>XORWF</td>
<td>f, d</td>
<td>W @ f → d</td>
<td>Z</td>
</tr>
<tr>
<td>0010 10df ffff 0cf</td>
<td>Decrement f</td>
<td>DECf</td>
<td>f, d</td>
<td>f - 1 → d</td>
<td>Z</td>
</tr>
<tr>
<td>0000 001f ffff 02f</td>
<td>Move W to f</td>
<td>MOVWF</td>
<td>f</td>
<td>W → f</td>
<td>None</td>
</tr>
<tr>
<td>0000 0000 0000 000</td>
<td>No Operation</td>
<td>NOP</td>
<td>-</td>
<td>-</td>
<td>None</td>
</tr>
<tr>
<td>0001 10df ffff 1cf</td>
<td>Add W and f</td>
<td>ADDWF</td>
<td>f, d</td>
<td>W + f → d</td>
<td>C, DC, Z</td>
</tr>
<tr>
<td>0001 11df ffff 1cf</td>
<td>Add W and f</td>
<td>ADDWF</td>
<td>f, d</td>
<td>W + f → d</td>
<td>C, DC, Z</td>
</tr>
</tbody>
</table>

#### BIT-ORIENTED FILE REGISTER OPERATIONS

<table>
<thead>
<tr>
<th>Instruction-Binary (Hex)</th>
<th>Name</th>
<th>Mnemonic, Operands</th>
<th>Operation</th>
<th>Status Affected</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>0100 bbff ffff 4bf</td>
<td>Bit Clear f</td>
<td>BCF</td>
<td>f, b</td>
<td>0 → f(b)</td>
<td>None</td>
</tr>
<tr>
<td>0110 bbff ffff 8bf</td>
<td>Subtract W from f</td>
<td>SUBWF</td>
<td>f, d</td>
<td>f - W → f</td>
<td>Z</td>
</tr>
<tr>
<td>0111 bbff ffff 7bf</td>
<td>Exclusive OR W and f</td>
<td>XORWF</td>
<td>f, d</td>
<td>W @ f → d</td>
<td>Z</td>
</tr>
</tbody>
</table>

#### LITERAL AND CONTROL OPERATIONS

<table>
<thead>
<tr>
<th>Instruction-Binary (Hex)</th>
<th>Name</th>
<th>Mnemonic, Operands</th>
<th>Operation</th>
<th>Status Affected</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1111 kkkk kkkk Ekk</td>
<td>AND Literal and W</td>
<td>ANDLW</td>
<td>k</td>
<td>k &amp; W → W</td>
<td>Z</td>
</tr>
<tr>
<td>1001 kkkk kkkk 9kk</td>
<td>Call subroutine</td>
<td>CALL</td>
<td>k</td>
<td>PC + 1 → Stack, k → PC</td>
<td>None</td>
</tr>
<tr>
<td>0000 0000 0100 004</td>
<td>Clear Watchdog timer</td>
<td>CLRWDT</td>
<td>-</td>
<td>0 → WDT (and prescaler, if assigned)</td>
<td>TO, PD</td>
</tr>
<tr>
<td>101k kkkk kkkk A0k</td>
<td>Go To address (k is 9 bit)</td>
<td>GOTO</td>
<td>k</td>
<td>k → PC (9 bits)</td>
<td>None</td>
</tr>
<tr>
<td>1101 kkkk kkkk D0k</td>
<td>Incl. OR Literal and W</td>
<td>IORLW</td>
<td>k</td>
<td>k v W → W</td>
<td>Z</td>
</tr>
<tr>
<td>1100 kkkk kkkk C0k</td>
<td>Move Literal to W</td>
<td>MOVWL</td>
<td>k</td>
<td>k → W</td>
<td>None</td>
</tr>
<tr>
<td>0000 0000 0010 002</td>
<td>Load OPTION register</td>
<td>OPTION</td>
<td>-</td>
<td>W → OPTION register</td>
<td>None</td>
</tr>
<tr>
<td>1000 kkkk kkkk 8kk</td>
<td>Return, place Literal in W</td>
<td>RETLW</td>
<td>k</td>
<td>k → W, Stack → PC</td>
<td>None</td>
</tr>
<tr>
<td>0000 0000 0011 003</td>
<td>Go into standby mode</td>
<td>SLEEP</td>
<td>-</td>
<td>0 → WDT, stop oscillator</td>
<td>TO, PD</td>
</tr>
<tr>
<td>0000 0000 00ff 00f</td>
<td>Tristate port f</td>
<td>TRIS</td>
<td>f</td>
<td>W → I/O control register f</td>
<td>None</td>
</tr>
<tr>
<td>1111 kkkk kkkk Fkk</td>
<td>Excl. OR Literal and W</td>
<td>XORLW</td>
<td>k</td>
<td>k @ W → W</td>
<td>Z</td>
</tr>
</tbody>
</table>

Notes: See previous page
10.1 INSTRUCTION DESCRIPTION

ADDWF  ADD W to f

Syntax:  ADDWF f,d
Encoding:  

Words:  1
Cycles:  1
Operation:  (W + f) → d
Status bits:  C, DC, Z
Description:  Add the contents of the W register to 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".

ANDLW  AND Literal and W

Syntax:  ANDLW k
Encoding:  

Words:  1
Cycles:  1
Operation:  (W .AND. k) → W
Status bits:  Z
Description:  The contents of W register are AND'ed with the eight bit literal "k". The result is placed in the W register.

ANDWF  AND W with f

Syntax:  ANDWF f,d
Encoding:  

Words:  1
Cycles:  1
Operation:  (W .AND. f) → d
Status bits:  Z
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".

BCF  Bit Clear f

Syntax:  BCF f,b
Encoding:  

Words:  1
Cycles:  1
Operation:  0 → f(b)
Status bits:  None
Description:  Bit "b" in register "f" is reset to 0.

BSF  Bit Set f

Syntax:  BSF f,b
Encoding:  

Words:  1
Cycles:  1
Operation:  1 → f(b)
Status bits:  None
Description:  Bit "b" in register "f" is set to 1.

BTFSC  Bit Test, skip if Clear

Syntax:  BTFSC f,b
Encoding:  

Words:  1
Cycles:  1(2)
Operation:  skip if f(b) = 0
Status bits:  None
Description:  If bit "b" in register "f" is "0" then the next instruction is skipped.

BTFSS  Bit Test, skip if Set

Syntax:  BTFSS f,b
Encoding:  

Words:  1
Cycles:  1 (2)
Operation:  skip if f(b) = 1
Status bits:  None
Description:  If bit "b" in register "f" is "1" then the next instruction is skipped.

CALL  Subroutine Call

Syntax:  CALL k
Encoding:  

Words:  1
Cycles:  2
Operation:  PC + 1 → TOS; k → PC<7:0>, '0' → PC<8>, PA2, PA1, PA0 → PC<11:9>
Status bits:  None
Description: Subroutine call. First, return address (PC + 1) is pushed into the stack. The eight bit value is loaded into PC bits <7:0>. PC bit 9 is cleared. PC <2:0> bits are loaded into PC <11:9>. CALL is a two cycle instruction.

**CLRF** Clear f and Clear d

**Syntax:** CLRF f,d
**Encoding:**
```
0000 011f ffff
```
**Words:** 1
**Cycles:** 1
**Operation:** 00h → f, 00h → d
**Status bits:** None
**Description:** The contents of register "f" are set to 0. If "d" is 0 the contents of both data memory location "f" and W register are set to 0. If "d" is 1 the only contents of register "f" are set to 0.

**CLRW** Clear W Register

**Syntax:** CLRW
**Encoding:**
```
0000 0100 0000
```
**Words:** 1
**Cycles:** 1
**Operation:** 00h → W
**Status bits:** Z
**Description:** W register is cleared. Zero bit (Z) is set.

**CLRWDT** Clear Watchdog Timer

**Syntax:** CLRWDT
**Encoding:**
```
0000 0000 0100
```
**Words:** 1
**Cycles:** 1
**Operation:** 00h → WDT, 0 → WDT prescaler,
**Status bits:** 1 → TO, 1 → PD
**Description:** CLRWDT instruction resets the watchdog timer. It also resets the prescaler of the WDT. Status bits TO and PD are set.

**COMF** Complement f

**Syntax:** COMF f,d
**Encoding:**
```
0010 01df ffff
```
**Words:** 1
**Cycles:** 1
**Operation:** f → d
**Status bits:** Z
**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".

**DECFSZ** Decrement f, skip if 0

**Syntax:** DECFSZ f,d
**Encoding:**
```
0010 lOdf ffff
```
**Words:** 1
**Cycles:** 1 (2)
**Operation:** (f - 1) → d; skip if result = 0
**Status bits:** None
**Description:** 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". If the result is 0 the next instruction is skipped.

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.

**GOTO** Unconditional Branch

**Syntax:** GOTO k
**Encoding:**
```
101k kkkk kkkk
```
**Words:** 1
**Cycles:** 2
**Operation:** k → PC<8:0>, PA2, PA1, PA0 → PC<11:9>
**Status bits:** None
**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.

**INCF** Increment f

**Syntax:** INCF f,d
**Encoding:**
```
0010 10df ffff
```
**Words:** 1
**Cycles:** 1
**Operation:** (f + 1) → d
**Status bits:** C, DC, Z

© 1992 Microchip Technology Inc.
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”.

**INCFSZ** Increment f, skip if 0
Syntax: INCFSZ f,d
Encoding: 0011 1ldf ffff
Words: 1
Cycles: 1
Operation: (f + 1) → d, skip if result = 0
Status bits: None

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”. 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.

**IORLW** Inclusive OR Literal with W
Syntax: IORLW k
Encoding: 1101 kkkk kkkk
Words: 1
Cycles: 1
Operation: (W .OR. k) → W
Status bits: Z
Description: The eight bit literal “k” is loaded into W register.

**MOVWF** Move W to f
Syntax: MOVWF f,d
Encoding: 0010 00df ffff
Words: 1
Cycles: 1
Operation: (f) → d
Status bits: Z
Description: Move data from W register to register “f”.

**MOVLW** Move Literal to W
Syntax: MOVLW k
Encoding: 1100 kkkk kkkk
Words: 1
Cycles: 1
Operation: k → W
Status bits: None
Description: The eight bit literal “k” is loaded into W register.

**MOVWF** Move W to f
Syntax: MOVWF f
Encoding: 0000 001f ffff
Words: 1
Cycles: 1
Operation: W → f
Status bits: None
Description: Move data from W register to register “f”.

**NOP** No Operation
Syntax: NOP
Encoding: 0000 0000 0000
Words: 1
Cycles: 1
Operation: No operation
Status bits: None
Description: No operation

**OPTION** Load Option Register
Syntax: OPTION
Encoding: 0000 0000 001d
Words: 1
Cycles: 1
Operation: W → OPTION;
Status bits: None
Description: The contents of the W register is loaded in the OPTION register.
RETlw  
**Return Literal to W**

**Syntax:** 
RETlw k

**Encoding:**

```
1000  kkkk  kkkk
```

**Words:** 1

**Cycles:** 2

**Operation:** k → W; TOS → PC;

**Status bits:** None

**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.

RLF  
**Rotate Left f through Carry**

**Syntax:** 
RLF f.d

**Encoding:**

```
0011  01df  ffff
```

**Words:** 1

**Cycles:** 1

**Operation:** f<n> → d<n+1>; f<7> → C; C → d<0>;

**Status bits:** C

**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".

RRF  
**Rotate Right f through Carry**

**Syntax:** 
RRF f.d

**Encoding:**

```
0011  00df  ffff
```

**Words:** 1

**Cycles:** 1

**Operation:** f<n> → d<n-1>; f<0> → C; C → d<7>;

**Status bits:** C

**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".

SLEEP  
**Load TRIS Register**

**Syntax:** 
SLEEP

**Encoding:**

```
0000  0000  0011
```

**Words:** 1

**Cycles:** 1

**Operation:** 0 → PD, 1 → TO; 00h → WDT, 0 → WDT prescaler;

**Status bits:** TO, PD

**Description:** The processor is put into SLEEP mode with the oscillator stopped. See section on SLEEP mode for more details.

SUBWF  
**Subtract W from f**

**Syntax:** 
SUBWF f,d

**Encoding:**

```
0000  10df  ffff
```

**Words:** 1

**Cycles:** 1

**Operation:** (f-W) → d

**Status bits:** C, DC, Z

**Description:** Subtract (2's complement method) the 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".

SWAPF  
**Swap f**

**Syntax:** 
SWAPF f,d

**Encoding:**

```
0011  10df  ffff
```

**Words:** 1

**Cycles:** 1

**Operation:** f<3:0> → d<4:7>, f<4:7> → d<0:3>;

**Status bits:** None

**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".

TRIS  
**Load TRIS Register**

**Syntax:** 
TRIS f

**Encoding:**

```
0000  0000  0fff
```

**Words:** 1

**Cycles:** 1

**Operation:** W → TRIS register f;

**Status bits:** None

**Description:** TRIS register f (f = 5, 6 or 7) is loaded with the contents of the W register.
**Exclusive OR literal with W**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>XORLW k</th>
</tr>
</thead>
<tbody>
<tr>
<td>Encoding</td>
<td>1111 kkkk kkkk</td>
</tr>
<tr>
<td>Words</td>
<td>1</td>
</tr>
<tr>
<td>Cycles</td>
<td>1</td>
</tr>
<tr>
<td>Operation</td>
<td>(W .XOR. k) → W</td>
</tr>
<tr>
<td>Status bits</td>
<td>Z</td>
</tr>
<tr>
<td>Description:</td>
<td>The contents of the W register are XOR'ed with the eight bit literal &quot;k&quot;. The result is placed in the W register.</td>
</tr>
</tbody>
</table>

**Exclusive OR W with f**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>XORWF f,d</th>
</tr>
</thead>
<tbody>
<tr>
<td>Encoding</td>
<td>0001 10df ffff</td>
</tr>
<tr>
<td>Words</td>
<td>1</td>
</tr>
<tr>
<td>Cycles</td>
<td>1</td>
</tr>
<tr>
<td>Operation</td>
<td>(W .XOR. f) → d</td>
</tr>
<tr>
<td>Status bits</td>
<td>Z</td>
</tr>
<tr>
<td>Description:</td>
<td>Exclusive OR the contents of the W register with register &quot;f&quot;. If &quot;d&quot; is 0 the result is stored in the W register. If &quot;d&quot; is 1 the result is stored back in register &quot;f&quot;.</td>
</tr>
</tbody>
</table>

### 11.0 WATCHDOG TIMER (WDT)

The watchdog timer is realized as a free running on-chip RC oscillator which does not require any external components. That means that the WDT will run, even if the clock on the OSC1/OSC2 pins of the device has been stopped, for example, by execution of a SLEEP instruction. A WDT timeout generates a device RESET condition. The WDT can be permanently disabled by programming a "zero" into a special EPROM fuse which is not part of the normal program memory EPROM. The PIC development tools (e.g. PICMASTER, PIC-PAK, PICPRO) provide special commands to select the desired oscillator configuration. On OTP and QTP devices, the oscillator configuration is programmed by the factory and the parts are tested only to the according specifications.

---

**11.1 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 2.5 seconds can be realized. The "CLRWDT" and "SLEEP" instructions clear the WDT and the prescaler, if assigned to the WDT, and prevent it from timing out and generating a device RESET condition.

---

The status bit "TO" in file register f3 will be cleared upon a watchdog timer timeout. The WDT period is a function of the supply voltage, operating temperature, and will also vary from unit to unit due to variations in the manufacturing process. Please refer to the graphs in section 18.0 and DC specs for more details.

### 11.2 WDT Programming Considerations

In a noisy application environment the OPTION register can get corrupted. The OPTION register should be updated at regular intervals. 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 timeout occurs.

### 12.0 OSCILLATOR CIRCUITS

#### 12.1 Oscillator Types

The PIC16C5X series is available with 4 different oscillator options. On windowed devices, a particular oscillator circuit can be selected by programming the configuration EPROM accordingly. The PIC development tools (e.g. PICMASTER, PIC-PAK, PICPRO) provide special commands to select the desired oscillator configuration.

**Crystal Oscillator**

The PIC16C5X-XT, -HS, or LP needs a crystal or ceramic resonator connected to the OSC1 and OSC2 pins to establish oscillation (Figure 12.2.1). XT = Standard crystal oscillator, HS = High speed crystal oscillator. The series resistor Rs may be required for the "HS" oscillator, especially at lower than 20 MHz oscillation frequency. It may also be required in XT mode with AT strip-cut type crystals to avoid overdriving.

---

**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 (Rext) and capacitor (Cext) values, and the operation 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 Cext values. The user also needs to take into account variation to due tolerance of external R and C components used. Figure 12.3.1 shows how the R/C combination is connected to the PIC16C5X. For Rext values below 2.2 kOhm, the
oscillator operation may become unstable, or stop completely. For very high Rext values (e.g. 1 MOhm), the oscillator becomes sensitive to noise, humidity and leakage. Thus, we recommend to keep Rext between 5 kOhm and 100 kOhm.

Although the oscillator will operate with no external capacitor (Cext = 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 table in section 18.0 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 characteristics in section 18.0 for variation of oscillator frequency due to VDD for given Rext/Cext values as well as frequency variation due to operating temperature for given R, C, and VDD 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 2.2.1 for timing).

FIGURE 12.2.1 - CRYSTAL OPERATION (OR CERAMIC RESONATOR) (HS, XT OR LP TYPES ONLY)

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.

FIGURE 12.2.2 - EXTERNAL CLOCK INPUT OPERATION (HS, XT, OR LP TYPES ONLY)

TABLE 12.2.1 - CAPACITOR SELECTION FOR CERAMIC RESONATORS

<table>
<thead>
<tr>
<th>Oscillator Type</th>
<th>Resonator Frequency</th>
<th>Capacitor Range C1 = C2</th>
</tr>
</thead>
<tbody>
<tr>
<td>XT</td>
<td>455 KHz</td>
<td>150 - 330 pF</td>
</tr>
<tr>
<td></td>
<td>2.0 MHz</td>
<td>20 - 330 pF</td>
</tr>
<tr>
<td></td>
<td>4.0 MHz</td>
<td>20 - 330 pF</td>
</tr>
<tr>
<td>HS</td>
<td>8.0 MHz</td>
<td>20 - 200 pF</td>
</tr>
</tbody>
</table>

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.

FIGURE 12.3.1 - RC OSCILLATOR (RC TYPE ONLY)
13.0 OSCILLATOR START-UP TIMER (OST)

Oscillator circuits based on crystals or ceramic resonators require a certain time after power-up to establish a stable oscillation. An on-chip oscillator start-up timer is provided which keeps the device in a RESET condition for approximately 18 ms after the voltage on the MCLR pin has reached a logic high (VIL/MCL) level. Thus, external RC networks connected to the MCLR input are not required in most cases, allowing for savings in cost-sensitive and/or space restricted applications.

The OST will also be triggered upon a watchdog timer timeout. This is particularly important for applications using the WDT to awake the PIC16C5X from SLEEP mode automatically.

The OST is not adequate for low frequency crystals which require much longer than 18 ms to start up and stabilize.

13.1 Power On Reset (POR)

The PIC16C5X incorporates an on chip Power On Reset (POR) circuitry which provides internal chip reset for most power-up situations. To use this feature the user merely needs to tie MCLR pin to VDD. A simplified block diagram of the on-chip power on reset circuit is shown in Figure 13.1.4. The power on reset circuit and the oscillator start-up timer circuit are closely related. On power-up the reset latch is set and the start-up timer begins counting once it detects MCLR to be high. After the time-out period, which is typically 18 ms, it will reset the reset-latch and thus end the on-chip reset signal.

Figures 13.1.5 and 13.1.6 are two power-up situations with relatively fast rise time on VDD. In Figure 13.1.5, VDD is allowed to rise and stabilize before bringing MCLR high. The chip will actually come out of reset 18 ms after MCLR goes high. In Figure 13.1.6, the on chip power-on reset feature is being utilized (MCLR and VDD are tied together). The VDD is stable before the startup timer times out and there is no problem in getting a proper reset. Figure 13.1.7 depicts a potentially problematic situation where VDD rises too slowly. In this situation, when the start-up timer times out, VDD has not reached the VDD (min) value and the chip is therefore not guaranteed to function correctly.

To summarize, the on chip power-on reset is guaranteed to work if the rate of rise of VDD is no slower than 0.05 V/ms. It is also necessary that the VDD starts from 0 V. The on chip power on reset is also not adequate for low frequency crystals which require much longer than 18 ms to start up and stabilize. For such situations, we recommend that external RC circuits are used for longer power on reset.
FIGURE 13.1.4 - SIMPLIFIED POWER ON RESET BLOCK DIAGRAM

FIGURE 13.1.5 - USING EXTERNAL RESET INPUT

Note 1: The tost time-out is invoked every time the chip comes out of reset.

FIGURE 13.1.6 - USING ON-CHIP POR (FAST VDD RISE TIME)
14.0 POWER DOWN MODE (SLEEP)

The power down mode is entered by executing a SLEEP instruction.

If enabled, the watchdog timer will be cleared but keeps running, the bit "PD" in the status register (f3) is cleared, the "TO" bit is set, and the oscillator driver is turned off. The I/O ports maintain the status they had, before the SLEEP command was executed (driving high, low, or high-impedance).

For lowest current consumption in this mode, all I/O pins should be either at VDD, or Vss, with no external circuitry drawing current from the I/O pin. I/O pins that are in the High-Z mode should be pulled high or low externally to avoid switching currents caused by floating inputs. The RTCC input should also be at VDD or Vss for lowest current consumption.

The MCLR pin must be at Vihmc.

14.1 Wake-Up

The device can be awakened by a watchdog timer timeout (if it is enabled) or an externally applied "low" pulse at the MCLR pin. In both cases the PIC will stay in RESET mode for one oscillator start-up timer period (triggered from rising edge on MCLR or WDT timeout) before normal program execution resumes.

The "PD" bit in the STATUS register, which is set to one during power on, but cleared by the "SLEEP" command, can be used to determine if the processor was powered up or awakened from the power down mode (Table 4.5.1.2). The TO bit in the Status register can be used to determine, if the "wake up" was caused by an external MCLR signal or a watchdog timer time out.

NOTE: Some applications may require external R/C networks on the MCLR pin in order to allow for oscillator startup times longer than one OST period. In this case, a WDT wake up from power down mode is not recommended, because a RESET generated by a WDT time out does not discharge the external capacitor, and the PIC will be in RESET only for the oscillator start-up timer period.

15.0 CONFIGURATION FUSES

The configuration EPROM consists of four EPROM fuses which are not part of the normal EPROM for program storage.

Two are for the selection of the oscillator type, one is the watchdog timer enable fuse, and one is the code protection fuse.

The PIC development tools (PICMASTER™, PICPAK-11™, PICPRO™, PICPRO II and PRO MASTER) allow the setting of these with special commands.

OTP or QTP devices have the oscillator configuration programmed by the factory and the parts are tested accordingly. The packages are marked with the suffixes "XT", "RC", "HS" or "LP" following the part number to identify the oscillator type and operating range.

15.1 Customer ID Code

The PIC16C5X series has 16 special EPROM bits which are not part of the normal program memory. These bits are available to the user to store an Identifier (ID) code, checksum, or other informative data. They cannot be accessed during normal program execution. The PIC16C5X programmers (e.g. PICPRO or PICPRO II) provide special commands to read or write these ID bits.
15.2 Code Protection

The program code written into the EPROM can be protected by programming the code protection fuse with "0".
When code protected, the contents of the program EPROM cannot be read out in a way that the program code can be reconstructed. In addition, all memory locations starting at 040h and above are protected against programming. It is still possible to program locations 000h - 03Fh, the ID locations and the configuration fuses. Note that the configuration fuses and the ID bits can still be read, even if the code protection logic is active.

15.2.1 VERIFYING A CODE-PROTECTED PIC

When code protected verifying any program memory location will read a scrambled output which looks like "00000000XXXX" (binary) where X is 1 or 0. To verify a device after code protection, follow this procedure:

a. First, program and verify a good device without code protecting it.
b. Next, blow its code protection fuse and then load its contents in a file.
c. Verify any code-protected PIC against this file.
16.0 ELECTRICAL CHARACTERISTICS

16.1 Absolute Maximum Ratings*

Ambient temperature under bias ...........-55°C to +125°C
Storage Temperature ..................-65°C to +150°C
Voltage on any pin with respect to Vss
(except VDD and MCLR) ............-0.6V to VDD +0.6V
Voltage on VDD with respect to Vss ..........0 to +9.5 V
Voltage on MCLR with respect to Vss
(Note 2) ...........................................0 to +14 V
Total power Dissipation (Note 1) ....................800 mW
Max. Current out of Vss pin .................150 mA
Max. Current into VDD pin ..................50 mA
Max. Current into an input pin ..............±500 µA
Max. Output Current sourced by any I/O pin .......20 mA
Max. Output Current by a single I/O port (Port A, B, or C) ..............40 mA
Max. Output Current by a single I/O port (Port A, B, or C) ..............50 mA

*Notice: Stresses above those listed under “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.

Notes: 1. Total power dissipation should not exceed 800 mW for the package. Power dissipation is calculated as follows:

\[ P_{dis} = V_{DD} \times (I_{DD} - \sum I_{O}) + \sum (V_{DD} - V_{O}) \times I_{O} + \sum (V_{O} \times I_{O}) \]

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

<table>
<thead>
<tr>
<th>Name</th>
<th>Function</th>
<th>Observation</th>
</tr>
</thead>
<tbody>
<tr>
<td>RA0 - RA3</td>
<td>I/O PORT A</td>
<td>4 input/output lines.</td>
</tr>
<tr>
<td>RB0 - RB7</td>
<td>I/O PORT B</td>
<td>8 input/output lines.</td>
</tr>
<tr>
<td>RC0 - RC7</td>
<td>I/O PORT C</td>
<td>8 input/output lines, (PIC16C55/C57 only). Schmitt Trigger Input.</td>
</tr>
<tr>
<td>RTCC</td>
<td>Real Time Clock/Counter</td>
<td>Clock input to RTCC register. Must be tied to Vss or VDD if not in use to avoid unintended entering of test modes and to reduce current consumption. Schmitt Trigger Input.</td>
</tr>
<tr>
<td>MCLR</td>
<td>Master Clear</td>
<td>A &quot;Low&quot; voltage on this input generates a RESET condition for the PIC16C5X microcontroller. A rising voltage triggers the on-chip oscillator start-up timer which keeps the chip in RESET mode for about 18ms. This input must be tied directly, or via a pull-up resistor, to VDD. &quot;XT&quot;, &quot;HS&quot; and &quot;LP&quot; devices: Input terminal for crystal, ceramic resonator, or external clock generator. &quot;RC&quot; devices : Driver terminal for external RC combination to establish oscillation. For &quot;XT&quot;, &quot;HS&quot; and &quot;LP&quot; devices: Output terminal for crystal and ceramic resonator. Do not connect any other load to this output. Leave open if external clock generator is used. For &quot;RC&quot; devices : A &quot;CLKOUT&quot; signal with a frequency of 1/4 Fosc1 is put out on this pin.</td>
</tr>
<tr>
<td>OSC1</td>
<td>Oscillator (input)</td>
<td></td>
</tr>
<tr>
<td>OSC2/CLKOUT</td>
<td>Oscillator (output)</td>
<td></td>
</tr>
<tr>
<td>VDD</td>
<td>Power supply</td>
<td></td>
</tr>
<tr>
<td>VSS</td>
<td>Ground</td>
<td></td>
</tr>
<tr>
<td>N/C</td>
<td>No (internal) Connection</td>
<td></td>
</tr>
</tbody>
</table>
16.3 DC CHARACTERISTICS: PIC16C5X-RC, XT, HS, LP (COMMERCIAL)

<table>
<thead>
<tr>
<th>DC CHARACTERISTICS, POWER SUPPLY PINS</th>
<th>Standard Operating Conditions</th>
<th>Operating voltage $V_{DD} = 3.0V$ to $5.5V$ unless otherwise stated</th>
</tr>
</thead>
<tbody>
<tr>
<td>Characteristic</td>
<td>Sym</td>
<td>Min (Note 1)</td>
</tr>
<tr>
<td>Supply Voltage</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PIC16C5X-XT</td>
<td>$V_{DDxt}$</td>
<td>3.0</td>
</tr>
<tr>
<td>PIC16C5X-RC</td>
<td>$V_{DDrc}$</td>
<td>3.0</td>
</tr>
<tr>
<td>PIC16C5X-HS</td>
<td>$V_{DDhs}$</td>
<td>4.5</td>
</tr>
<tr>
<td>PIC16C5X-LP</td>
<td>$V_{DDlp}$</td>
<td>2.5</td>
</tr>
<tr>
<td>RAM Data Retention Voltage (Note 3)</td>
<td>$V_{DR}$</td>
<td>1.5</td>
</tr>
<tr>
<td>Vdd start voltage to guarantee power on reset</td>
<td>$V_{POR}$</td>
<td>$V_{SS}$</td>
</tr>
<tr>
<td>Vdd rise rate to guarantee power on reset</td>
<td>$S_{VDD}$</td>
<td>0.05*</td>
</tr>
<tr>
<td>Supply Current (Note 2)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PIC16C5X-XT</td>
<td>$I_{DDxt}$</td>
<td>1.8</td>
</tr>
<tr>
<td>PIC16C5X-RC (Note 5)</td>
<td>$I_{DDrc}$</td>
<td>1.8</td>
</tr>
<tr>
<td>PIC16C5X-HS</td>
<td>$I_{DDhs1}$</td>
<td>4.8</td>
</tr>
<tr>
<td>PIC16C5X-LP</td>
<td>$I_{DDhs2}$</td>
<td>9.0</td>
</tr>
<tr>
<td>Power Down Current (Note 4)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PIC16C5X</td>
<td>$I_{PD1}$</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>$I_{PD2}$</td>
<td>0.6</td>
</tr>
</tbody>
</table>

* These parameters are based on characterization and are not tested.

Note 1: Data in the column labeled "Typical" is based on characterization results at 25°C. This data is for design guidance only and is not tested for, or guaranteed by Microchip Technology.

Note 2: The supply current is mainly a function of the operating voltage and frequency. Other factors such as bus loading, oscillator type, bus rate, internal code execution pattern, and temperature also have an impact on the current consumption.

a) The test conditions for all $I_{DD}$ measurements in active operation mode are:

OSC1=external square wave, from rail to rail; all I/O pins tristated, pulled to $V_{DD}$, RT = $V_{DD}$, MCLR = $V_{DD}$; WDT enabled/disabled as specified.

b) For stand-by current measurements, the conditions are the same, except that the device is in SLEEP mode.

Note 3: This is the limit to which $V_{DD}$ can be lowered in SLEEP mode without losing RAM data.

Note 4: 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 $V_{DD}$ and $V_{SS}$.

Note 5: Does not include current through $R_{ext}$. The current through the resistor can be estimated by the formula $I_{R} = V_{DD}/2R_{ext}$ (mA) with $R_{ext}$ in kΩ.
### 16.4 DC CHARACTERISTICS: PIC16C5X-RC, XT, HS, LP (INDUSTRIAL)

#### DC CHARACTERISTICS, POWER SUPPLY PINS

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Sym</th>
<th>Min</th>
<th>Typ (Note 1)</th>
<th>Max</th>
<th>Units</th>
<th>Conditions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Supply Voltage</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PIC16C5X-XT</td>
<td>VDDSxt</td>
<td>3.0</td>
<td>6.25</td>
<td>V</td>
<td></td>
<td>Fosc = DC to 4 MHz</td>
</tr>
<tr>
<td>PIC16C5X-RC</td>
<td>VDDRC</td>
<td>3.0</td>
<td>6.25</td>
<td>V</td>
<td></td>
<td>Fosc = DC to 4 MHz</td>
</tr>
<tr>
<td>PIC16C5X-HS</td>
<td>VDDHS</td>
<td>4.5</td>
<td>5.5</td>
<td>V</td>
<td></td>
<td>Fosc = DC to 20 MHz</td>
</tr>
<tr>
<td>PIC16C5X-LP</td>
<td>VDDLP</td>
<td>2.5</td>
<td>6.25</td>
<td>V</td>
<td></td>
<td>Fosc = DC to 40 KHz</td>
</tr>
<tr>
<td>RAM Data Retention Voltage</td>
<td>VDR</td>
<td>1.5</td>
<td>V</td>
<td></td>
<td></td>
<td>Device in SLEEP mode</td>
</tr>
<tr>
<td>Voltage (Note 3)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Vdd start voltage to</td>
<td>VPOR</td>
<td></td>
<td>V</td>
<td></td>
<td></td>
<td>See section 13.1 for details on power on reset</td>
</tr>
<tr>
<td>guarantee power on reset</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Vdd rise rate to guarantee</td>
<td>SVDD</td>
<td>0.05*</td>
<td>V/ms</td>
<td></td>
<td></td>
<td>See section 13.1 for details on power on reset</td>
</tr>
<tr>
<td>power on reset</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Supply Current (Note 2)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PIC16C5X-XT</td>
<td>IODXT</td>
<td>1.8</td>
<td>3.3</td>
<td>mA</td>
<td></td>
<td>Fosc = 4 MHz, VDD = 5.5V</td>
</tr>
<tr>
<td>PIC16C5X-RC (Note 5)</td>
<td>IODRC</td>
<td>1.8</td>
<td>3.3</td>
<td>mA</td>
<td></td>
<td>Fosc = 4 MHz, VDD = 5.5V</td>
</tr>
<tr>
<td>PIC16C5X-HS</td>
<td>IODHS1</td>
<td>4.8</td>
<td>10.0</td>
<td>mA</td>
<td></td>
<td>Fosc = 10 MHz, VDD = 5.5V</td>
</tr>
<tr>
<td>PIC16C5X-LP</td>
<td>IODHS2</td>
<td>9.0</td>
<td>20.0</td>
<td>mA</td>
<td></td>
<td>Fosc = 20 MHz, VDD = 5.5V</td>
</tr>
<tr>
<td>Power Down Current</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Note 4)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PIC16C5X</td>
<td>IPD1</td>
<td>5</td>
<td>14</td>
<td>µA</td>
<td></td>
<td>VDD = 3.0V, WDT enabled</td>
</tr>
<tr>
<td>Pic16C5X</td>
<td>IPD2</td>
<td>0.8</td>
<td>12</td>
<td>µA</td>
<td></td>
<td>VDD = 3.0V, WDT disabled</td>
</tr>
</tbody>
</table>

* These parameters are based on characterization and are not tested.

**Note 1:** Data in the column labeled "Typical" is based on characterization results at 25°C. This data is for design guidance only and is not tested for, or guaranteed by Microchip Technology.

**Note 2:** The supply current is mainly a function of the operating voltage and frequency. Other factors such as bus loading, oscillator type, bus rate, internal code execution pattern, and temperature also have an impact on the current consumption.

- a) 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, RT = VDD, MCLR = VDD; WDT enabled/disabled as specified.
  - b) For stand-by current measurements, the conditions are the same, except that the device is in SLEEP mode.

**Note 3:** This is the limit to which VDD can be lowered in SLEEP mode without losing RAM data.

**Note 4:** 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.

**Note 5:** Does not include current through Rext. The current through the resistor can be estimated by the formula \( I_r = \frac{V_{DD}}{2R_{ext}} \) (mA) with Rext in kOhm.
## 16.5 DC CHARACTERISTICS: PIC16C5X-RC, XT, HS, LP (AUTOMOTIVE)

### DC CHARACTERISTICS, POWER SUPPLY PINS

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Sym</th>
<th>Min</th>
<th>Typ (Note 1)</th>
<th>Max</th>
<th>Units</th>
<th>Conditions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Supply Voltage</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PIC16C5X-XT</td>
<td>VDDx</td>
<td>3.25</td>
<td>6.0</td>
<td>V</td>
<td>Fosc = DC to 4 MHz</td>
<td></td>
</tr>
<tr>
<td>PIC16C5X-RC</td>
<td>VDDr</td>
<td>3.25</td>
<td>6.0</td>
<td>V</td>
<td>Fosc = DC to 4 MHz</td>
<td></td>
</tr>
<tr>
<td>PIC16C5X-HS</td>
<td>VDDh</td>
<td>4.5</td>
<td>5.5</td>
<td>V</td>
<td>Fosc = DC to 16 MHz</td>
<td></td>
</tr>
<tr>
<td>PIC16C5X-LP</td>
<td>VDDl</td>
<td>2.5</td>
<td>6.0</td>
<td>V</td>
<td>Fosc = DC to 40 KHz</td>
<td></td>
</tr>
</tbody>
</table>

| RAM Data Retention Voltage (Note 3) | VDR | 1.5 | V | Device in SLEEP mode |

| Voltage (Note 3) | VROR | Vss | V | See section 13.1 for details on power on reset |

| Voltage (Note 3) | VDD rise rate to guarantee power on reset | VDD | 0.05* | V/μs | See section 13.1 for details on power on reset |

| Supply Current (Note 2) | IDDX | 1.8 | 3.3 | mA | Fosc = 4 MHz, VDD = 5.5V |
| PIC16C5X-XT (Note 5)    | IDDc | 1.8 | 3.3 | mA | Fosc = 4 MHz, VDD = 5.5V |
| PIC16C5X-HS              | IDD1 | 4.8 | 10.0 | mA | Fosc = 10 MHz, VDD = 5.5V |
| PIC16C5X-LP              | IDD2 | 9.0 | 20.0 | mA | Fosc = 16 MHz, VDD = 5.5V |
|                         | IDDp | 25  | 55   | μA | Fosc = 32 KHz, VDD = 3.25V, WDT disabled |

| Power Down Current (Note 4) | IPD1 | 5   | 22  | μA | VDD = 3.25V, WDT enabled |
|                            | IPD2 | 0.8 | 18  | μA | VDD = 3.25V, WDT disabled |

* These parameters are based on characterization and are not tested.

**Note 1:** Data in the column labeled "Typical" is based on characterization results at 25°C. This data is for design guidance only and is not tested for, or guaranteed by Microchip Technology.

**Note 2:** The supply current is mainly a function of the operating voltage and frequency. Other factors such as bus loading, oscillator type, bus rate, internal code execution pattern, and temperature also have an impact on the current consumption.

  a) 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, RT = VDD, MCLR = VDD; WDT enabled/disabled as specified.

  b) For stand-by current measurements, the conditions are the same, except that the device is in SLEEP mode.

**Note 3:** This is the limit to which VDD can be lowered in SLEEP mode without losing RAM data.

**Note 4:** 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-impedence state and tied to VDD and VSS.

**Note 5:** Does not include current through Rext. The current through the resistor can be estimated by the formula \( I_R = \frac{VDD}{2 \times Rext} \) (mA) with Rext in kOhm.
16.6 DC CHARACTERISTICS: PIC16C5X-RC, XT, HS, LP (COMMERCIAL)  
PIC16C5XI-RC, XT, HS, LP (INDUSTRIAL)

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Sym</th>
<th>Min</th>
<th>Typ (Note 1)</th>
<th>Max</th>
<th>Units</th>
<th>Conditions</th>
</tr>
</thead>
</table>

**DC CHARACTERISTICS,**
**ALL PINS EXCEPT POWER SUPPLY**

**Standard Operating Conditions (unless otherwise stated)**
Operating temperature \(-40 < T_A < +85^\circ\text{C}\) 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 tables  
16.3 and 16.4

**Input Low Voltage**
- I/O ports: \(V_{IL} \rightarrow V_{ss}\) 0.2 \(V_{DD}\) V  
- MCLR (Schmitt trigger): \(V_{ILMC} \rightarrow V_{ss}\) 0.15 \(V_{DD}\) V  
- RTCC (Schmitt trigger): \(V_{ILRT} \rightarrow V_{ss}\) 0.15 \(V_{DD}\) V  
- OSC1 (Schmitt trigger): \(V_{ILOSC} \rightarrow V_{ss}\) 0.15 \(V_{DD}\) V  
- PIC16C5XRC only (Note 5)

**Input High Voltage**
- I/O ports: \(V_{IH} \rightarrow V_{DD}\) 0.45 \(V_{DD}\) V  
- MCLR (Schmitt trigger): \(V_{IHMC} \rightarrow V_{DD}\) V  
- RTCC (Schmitt trigger): \(V_{IHRT} \rightarrow V_{DD}\) V  
- OSC1 (Schmitt trigger): \(V_{IHOSC} \rightarrow V_{DD}\) V  
- PIC16C5X-RC only (Note 5)

**Input Leakage Current**
- (Notes 3, 4)
- I/O ports: \(I_{IL}\) -1 0.5 +1 \(\mu\text{A}\)  
- MCLR: \(I_{ILMCL}\) -5 \(\mu\text{A}\)  
- \(V_{PIN} = V_{ss} + 0.25V\)  
- MCLR: \(I_{ILMCLH}\) 0.5 +5 \(\mu\text{A}\)  
- \(V_{PIN} = V_{DD}\)  
- RTCC: \(I_{ILRT}\) -3 0.5 +3 \(\mu\text{A}\)  
- \(V_{SS} \leq V_{PIN} \leq V_{DD}\)  
- OSC1: \(I_{ILOSC}\) -3 0.5 +3 \(\mu\text{A}\)  
- PIC16C5X-RC only (Note 5)

**Output Low Voltage**
- I/O Ports: \(V_{OL}\) 0.6 \(V_{DD}\) V  
- OSC2/CLKOUT (PIC16C5X-RC): \(V_{OL}\) 0.6 \(V_{DD}\) V

**Output High Voltage**
- I/O Ports (Note 4): \(V_{OH}\) VDD-0.7 \(V_{DD}\) V  
- OSC2/CLKOUT (PIC16C5X-RC): \(V_{OH}\) VDD-0.7 \(V_{DD}\) V

Note 1: Data in the column labeled "Typical" is based on characterization results at 25 \(^\circ\text{C}\). This data is for design guidance only and is not tested for, or guaranteed by Microchip Technology.

Note 2: Total power dissipation as stated under absolute maximum ratings must not be exceeded.

Note 3: The leakage current on the MCLR 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.

Note 4: Negative current is defined as coming out of the pin.

Note 5: For PIC16C5XRC devices, the OSC1 pin is a Schmitt trigger input. It is not recommended that the PIC16C5X be driven with external clock in RC mode.

Note 6: The user may use better of the two specifications.
### 16.7 DC CHARACTERISTICS: PIC16C5X-RC, XT, HS, LP (AUTOMOTIVE)

#### DC CHARACTERISTICS, ALL PINS EXCEPT POWER SUPPLY

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Sym</th>
<th>Min</th>
<th>Typ (Note 1)</th>
<th>Max</th>
<th>Units</th>
<th>Conditions</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Input Low Voltage</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>I/O ports</td>
<td>VIL</td>
<td>VSS</td>
<td>0.15 VDD</td>
<td></td>
<td>V</td>
<td>Pin at hi-impedance</td>
</tr>
<tr>
<td>MCLR (Schmitt trigger)</td>
<td>VILMC</td>
<td>VSS</td>
<td>0.15 VDD</td>
<td></td>
<td>V</td>
<td></td>
</tr>
<tr>
<td>RTCC (Schmitt trigger)</td>
<td>VILRT</td>
<td>VSS</td>
<td>0.15 VDD</td>
<td></td>
<td>V</td>
<td></td>
</tr>
<tr>
<td>OSC1 (Schmitt trigger)</td>
<td>VILOSC</td>
<td>VSS</td>
<td>0.15 VDD</td>
<td></td>
<td>V</td>
<td></td>
</tr>
<tr>
<td>OSC1</td>
<td>VILOSC</td>
<td>VSS</td>
<td>0.3 VDD</td>
<td></td>
<td>V</td>
<td></td>
</tr>
<tr>
<td><strong>Input High Voltage</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>I/O ports</td>
<td>VIH</td>
<td>0.45 VDD</td>
<td>VDD</td>
<td></td>
<td>V</td>
<td>For all VDD (Note 6)</td>
</tr>
<tr>
<td>MCLR (Schmitt trigger)</td>
<td>VIHMC</td>
<td>VDD</td>
<td>VDD</td>
<td></td>
<td>V</td>
<td>4.0 V &lt; VDD ≤ 5.5 V (Note 6)</td>
</tr>
<tr>
<td>RTCC (Schmitt trigger)</td>
<td>VIHRT</td>
<td>VDD</td>
<td>VDD</td>
<td></td>
<td>V</td>
<td>VDD &gt; 5.5 V</td>
</tr>
<tr>
<td>OSC1 (Schmitt trigger)</td>
<td>VIHOSC</td>
<td>VDD</td>
<td>VDD</td>
<td></td>
<td>V</td>
<td>For PIC16C5X-RC only (Note 5)</td>
</tr>
<tr>
<td>OSC1</td>
<td>VIHOSC</td>
<td>VDD</td>
<td>VDD</td>
<td></td>
<td>V</td>
<td>For PIC16C5X-XT, HS, LP</td>
</tr>
<tr>
<td><strong>Input Leakage Current</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>I/O ports</td>
<td>IIL</td>
<td>-1</td>
<td>0.5</td>
<td>+1</td>
<td>μA</td>
<td>VSS ≤ VPIN ≤ VDD, Pin at hi-impedance</td>
</tr>
<tr>
<td>MCLR</td>
<td>IILMCL</td>
<td>-5</td>
<td>0.5</td>
<td>+5</td>
<td>μA</td>
<td>VPIN = VSS + 0.25V</td>
</tr>
<tr>
<td>RTCC</td>
<td>IILMCH</td>
<td>-3</td>
<td>0.5</td>
<td>+3</td>
<td>μA</td>
<td>VPIN = VDD</td>
</tr>
<tr>
<td>OSC1</td>
<td>IILRT</td>
<td>-3</td>
<td>0.5</td>
<td>+3</td>
<td>μA</td>
<td>VSS ≤ VPIN ≤ VDD, PIC16C5X-XT, HS, LP</td>
</tr>
<tr>
<td><strong>Output Low Voltage</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>I/O Ports</td>
<td>VOL</td>
<td></td>
<td>0.6</td>
<td></td>
<td>V</td>
<td>IOL = 8.7 mA, VDD = 4.5V</td>
</tr>
<tr>
<td>OSC2/CLKOUT (PIC16C5X-RC)</td>
<td>VOL</td>
<td></td>
<td>0.6</td>
<td></td>
<td>V</td>
<td>IOL = 1.6 mA, VDD = 4.5V</td>
</tr>
<tr>
<td><strong>Output High Voltage</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>I/O Ports (Note 4)</td>
<td>VOH</td>
<td>VDD-0.7</td>
<td>VDD-0.7</td>
<td>V</td>
<td>IOL = -5.4 mA, VDD = 4.5V</td>
<td></td>
</tr>
<tr>
<td>OSC2/CLKOUT (PIC16C5X-RC)</td>
<td>VOH</td>
<td>VDD-0.7</td>
<td>VDD-0.7</td>
<td>V</td>
<td>IOL = -1.0 mA, VDD = 4.5V</td>
<td></td>
</tr>
</tbody>
</table>

Note 1: Data in the column labeled "Typical" is based on characterization results at 25 °C. This data is for design guidance only and is not tested for, or guaranteed by Microchip Technology.

Note 2: Total power dissipation as stated under absolute maximum ratings must not be exceeded.

Note 3: The leakage current on the MCLR 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.

Note 4: Negative current is defined as coming out of the pin.

Note 5: For PIC16C5XRC devices, the OSC1 pin is a Schmitt trigger input. It is not recommended that the PIC16C5X be driven with external clock in RC mode.

Note 6: The user may use better of the two specifications.
### 16.8 AC CHARACTERISTICS: PIC16C5X-RC, XT, HS, LP (COMMERCIAL)
PIC16C5XI-RC, XT, HS, LP (INDUSTRIAL)
PIC16C5XI-RC, XT, HS, LP (AUTOMOTIVE)

#### AC CHARACTERISTICS

**Standard Operating Conditions (unless otherwise stated)**
- Operating temperature $T_A = -40°C$ to $+85°C$ (industrial), $T_A = -40°C$ to $+125°C$ (automotive) and $0°C ≤ T_A ≤ +70°C$ (commercial).
- Operating voltage $V_{DD}$ range as described in DC spec tables 16.3 and 16.4.

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Sym</th>
<th>Min</th>
<th>Typ (Note 1)</th>
<th>Max</th>
<th>Units</th>
<th>Conditions</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>External CLOCKIN Frequency (Note 2)</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DC</td>
<td>4 MHz</td>
<td>RC mode</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>XT</td>
<td>4 MHz</td>
<td>XT mode</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HS</td>
<td>20 MHz</td>
<td>HS mode (Com/Ind)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LP</td>
<td>16 MHz</td>
<td>HS mode (Automotive)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FOSC, LP</td>
<td>40 kHz</td>
<td></td>
<td>LP mode</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Oscillator Frequency (Note 2)</strong></td>
<td>FOSC, RC</td>
<td>DC</td>
<td>4 MHz</td>
<td>RC mode</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FOSC, XT</td>
<td>0.1 MHz</td>
<td>XT mode</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FOSC, HS</td>
<td>4 MHz</td>
<td>HS mode (Com/Ind)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FOSC, LP</td>
<td>4 MHz</td>
<td>HS mode (Automotive)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FOSCLP</td>
<td>DC</td>
<td></td>
<td>LP mode</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Instruction Cycle Time (Note 2)</strong></td>
<td>TCY, RC</td>
<td>1.0 µs</td>
<td>DC</td>
<td>µs</td>
<td>RC mode</td>
<td></td>
</tr>
<tr>
<td>TCY, XT</td>
<td>1.0 µs</td>
<td>DC</td>
<td>µs</td>
<td>XT mode</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TCY, HS</td>
<td>0.2 µs</td>
<td>DC</td>
<td>µs</td>
<td>HS mode</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TCY, LP</td>
<td>100 µs</td>
<td>DC</td>
<td>µs</td>
<td>LP mode</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>External Clock in Timing (Note 4)</strong></td>
<td>TCKHL, XT</td>
<td>50 ns</td>
<td>ns</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TCKHL, LP</td>
<td>2 ns</td>
<td>µs</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TCKHL, HS</td>
<td>20 ns</td>
<td>ns</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>RESET Timing</strong></td>
<td>TMCLR</td>
<td>100 ns</td>
<td>ns</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>RTCC Input Timing, No Prescaler</strong></td>
<td>TRTH, RC</td>
<td>0.5 TCY + 20 ns</td>
<td>ns</td>
<td>Note 3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TRTL, RC</td>
<td>0.5 TCY + 20 ns</td>
<td>ns</td>
<td>Note 3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>RTCC Input Timing, With Prescaler</strong></td>
<td>TRTH, RC</td>
<td>10 ns</td>
<td>ns</td>
<td>Note 3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TRTL, RC</td>
<td>10 ns</td>
<td>ns</td>
<td>Note 3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TRTP, RC</td>
<td>TCY + 40 ns</td>
<td>N</td>
<td>ns</td>
<td>Note 3. Where $N = \text{prescale value (2, 4, ..., 256)}$</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Watchdog Timer Timeout Period (No Prescaler)</strong></td>
<td>TWDT, RC</td>
<td>9 ms</td>
<td>18 ms</td>
<td>30 ms</td>
<td>ms</td>
<td>$V_{DD} = 5.0V$</td>
</tr>
<tr>
<td><strong>Oscillation Start-up Timer Period</strong></td>
<td>TOST, RC</td>
<td>9 ms</td>
<td>18 ms</td>
<td>30 ms</td>
<td>ms</td>
<td>$V_{DD} = 5.0V$</td>
</tr>
<tr>
<td><strong>I/O Timing</strong></td>
<td>TDS, RC</td>
<td>0.25 TCY + 30 ns</td>
<td>ns</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TDH, RC</td>
<td>0 ns</td>
<td></td>
<td>ns</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TPD, RC</td>
<td>40 ns</td>
<td>ns</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

* Guaranteed by characterization, but not tested.
(Notes on next page)
NOTES TO AC CHARACTERISTICS:
PIC16C5X-RC, XT, HS, LP (COMMERCIAL)
PIC16C5X1-RC, XT, HS, LP (INDUSTRIAL)

1. Data in the column labeled "Typical" is based on characterization results at 25°C. This data is for design guidance only and is not tested for, or guaranteed by Microchip Technology.

2. 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 pin. When an external clock input is used, the "Max." cycle time limit is "DC" (no clock) for all devices.

3. For a detailed explanation of RTCC input clock requirements see section 4.2.1.

4. Clock-in high-time is the duration for which clock input is at Vihosc or higher. Clock-in low-time is the duration for which clock input is at Viloosc or lower.

16.9 Electrical Structure of Pins

FIGURE 16.9.1 - ELECTRICAL STRUCTURE OF I/O PINS (RA, RB, RC)

FIGURE 16.9.2 - ELECTRICAL STRUCTURE OF MCLR AND RTCC PINS

17.0 TIMING DIAGRAMS

FIGURE 17.0.1 - RTCC TIMING

FIGURE 17.0.2 - OSCILLATOR START-UP TIMING (PIC16C5XRC)

Notes to figures 16.9.1 and 16.9.2: The diodes and the grounded gate (or output driver) NMOS device are carefully designed to protect against ESD (Electrostatic discharge) and EOS (Electrical overstress). Rin is a small resistance to further protect the input buffer from ESD.

FIGURE 17.0.3 - INPUT/OUTPUT TIMING FOR I/O PORTS (PIC16C5XRC*)

* The CLKOUT output is available only on PIC16C5XRC devices.
18.0 DC & AC CHARACTERISTICS GRAPHS/TABLES:

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 (e.g. 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 18.0.4 - TYPICAL RC OSCILLATOR FREQUENCY vs VDD

TABLE 18.0.1 - RC OSCILLATOR FREQUENCIES

<table>
<thead>
<tr>
<th>Cext</th>
<th>Rext</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Fosc @ 5V, 25°C</td>
</tr>
<tr>
<td></td>
<td></td>
<td>± percentage variation</td>
</tr>
<tr>
<td>20pf</td>
<td>3.3k</td>
<td>4.71 MHz ± 28%</td>
</tr>
<tr>
<td>5k</td>
<td></td>
<td>3.31 MHz ± 25%</td>
</tr>
<tr>
<td>10k</td>
<td></td>
<td>1.91 MHz ± 24%</td>
</tr>
<tr>
<td>100k</td>
<td></td>
<td>207.76 KHz ± 39%</td>
</tr>
<tr>
<td>100pf</td>
<td>3.3k</td>
<td>1.65 MHz ± 18%</td>
</tr>
<tr>
<td>5k</td>
<td></td>
<td>1.23 MHz ± 21%</td>
</tr>
<tr>
<td>10k</td>
<td></td>
<td>711.54 KHz ± 18%</td>
</tr>
<tr>
<td>100k</td>
<td></td>
<td>75.62 KHz ± 28%</td>
</tr>
<tr>
<td>300pf</td>
<td>3.3k</td>
<td>672.78 KHz ± 14%</td>
</tr>
<tr>
<td>5k</td>
<td></td>
<td>489.49 KHz ± 13%</td>
</tr>
<tr>
<td>10k</td>
<td></td>
<td>275.73 KHz ± 13%</td>
</tr>
<tr>
<td>100k</td>
<td></td>
<td>28.12 KHz ± 23%</td>
</tr>
</tbody>
</table>

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 full VDD range.

FIGURE 18.0.5 - TYPICAL IpD vs VDD
WATCHDOG DISABLED 25°C

FIGURE 18.0.6 - TYPICAL IpD vs VDD
WATCHDOG ENABLED 25°C

Note: The gray shaded regions are outside the normal PIC operating range. Do not operate in these regions.
FIGURE 18.0.7 - MAXIMUM IpD vs VDD
WATCHDOG DISABLED

FIGURE 18.0.8 - MAXIMUM IpD vs VDD
WATCHDOG ENABLED*

* IPD, 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.

Note: The gray shaded regions are outside the normal PIC operating range. Do not operate in these regions.

FIGURE 18.0.9 - VTH (INPUT THRESHOLD VOLTAGE) OF I/O PINS vs VDD
FIGURE 18.0.10 - \( V_{IH}, V_{IL} \) OF MCLR, RTCC AND OSC1 (IN RC MODE) vs \( V_{DD} \)

Note: These input pins have Schmitt trigger input buffer.

FIGURE 18.0.11 - \( V_{TH} \) (INPUT THRESHOLD VOLTAGE) OF OSC1 INPUT (IN XT, HS, AND LP MODES) vs \( V_{DD} \)
FIGURE 18.0.12 - TYPICAL $I_{DD}$ vs FREQ (EXT CLOCK, 25°C)

FIGURE 18.0.13 - MAXIMUM $I_{DD}$ vs FREQ (EXT CLOCK, -40° to +85°C)
FIGURE 18.0.14 - MAXIMUM IDD vs FREQ (EXT CLOCK, -55° to +125°C)

FIGURE 18.0.15 - WDT Timer Time-out Period vs VDD

FIGURE 18.0.16 - Transconductance (gm) of HS Oscillator vs VDD

Note: The gray shaded regions are outside the normal PIC operating range. Do not operate in these regions.
FIGURE 18.0.17 - Transconductance (gm) of LP Oscillator vs VDD

Note: The gray shaded regions are outside the normal PIC operating range. Do not operate in these regions.

FIGURE 18.0.18 - Transconductance (gm) of XT Oscillator vs VDD

FIGURE 18.0.19 - IOH vs VOH, VDD = 3V

FIGURE 18.0.20 - IOH vs VOH, VDD = 5V
Note: The gray shaded regions are outside the normal PIC operating range. Do not operate in these regions.

**TABLE 18.0.2 - INPUT CAPACITANCE FOR PIC16C54/56**

<table>
<thead>
<tr>
<th>Pin Name</th>
<th>Typical Capacitance (pF)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>18L PDIP</td>
</tr>
<tr>
<td>RA port</td>
<td>5.0</td>
</tr>
<tr>
<td>RB port</td>
<td>5.0</td>
</tr>
<tr>
<td>MCLR</td>
<td>17.0</td>
</tr>
<tr>
<td>OSC1</td>
<td>4.0</td>
</tr>
<tr>
<td>OSC2/CLKOUT</td>
<td>4.3</td>
</tr>
<tr>
<td>RTCC</td>
<td>3.2</td>
</tr>
</tbody>
</table>

* All capacitance values are typical at 25°C and measured at 1 MHz. A part to part variation of ±25% (three standard deviations) should be taken into account.
19.0 PACKAGING DIAGRAMS AND DIMENSIONS

19.1 18-LEAD PLASTIC DUAL IN-LINE (300 mil)

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Millimeters</th>
<th>Inches</th>
</tr>
</thead>
<tbody>
<tr>
<td>α</td>
<td>0° - 10°</td>
<td>0° - 10°</td>
</tr>
<tr>
<td>A</td>
<td>4.064 - 0.160</td>
<td>0.160</td>
</tr>
<tr>
<td>A1</td>
<td>0.381 - 0.015 -</td>
<td>0.015 -</td>
</tr>
<tr>
<td>A2</td>
<td>3.048 - 0.150</td>
<td></td>
</tr>
<tr>
<td>B</td>
<td>0.356 - 0.022</td>
<td></td>
</tr>
<tr>
<td>B1</td>
<td>1.524 - 0.060</td>
<td>0.060</td>
</tr>
<tr>
<td>C</td>
<td>0.203 - 0.008</td>
<td>0.080</td>
</tr>
<tr>
<td>D</td>
<td>22.479 - 0.885</td>
<td>0.885</td>
</tr>
<tr>
<td>D1</td>
<td>20.320 - 0.800</td>
<td>0.800</td>
</tr>
<tr>
<td>E</td>
<td>7.620 - 0.325</td>
<td></td>
</tr>
<tr>
<td>E1</td>
<td>6.096 - 0.280</td>
<td></td>
</tr>
<tr>
<td>e1</td>
<td>2.489 - 0.102</td>
<td>Typical</td>
</tr>
<tr>
<td>eA</td>
<td>7.620 - 0.300</td>
<td>0.300</td>
</tr>
<tr>
<td>eB</td>
<td>7.874 - 0.390</td>
<td></td>
</tr>
<tr>
<td>L</td>
<td>3.048 - 0.140</td>
<td></td>
</tr>
<tr>
<td>N</td>
<td>18 - 18</td>
<td></td>
</tr>
<tr>
<td>S</td>
<td>0.889 - 0.035</td>
<td></td>
</tr>
<tr>
<td>S1</td>
<td>0.127 - 0.005</td>
<td></td>
</tr>
</tbody>
</table>

Package Group: Plastic Dual In-line (PLA)

© 1992 Microchip Technology Inc.
19.2 28-LEAD DUAL IN-LINE PLASTIC (600 mil)

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Millimeters</th>
<th>Inches</th>
<th>Notes</th>
<th>Millimeters</th>
<th>Inches</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>α</td>
<td>0°</td>
<td>10°</td>
<td>Notes</td>
<td>0°</td>
<td>10°</td>
<td>Notes</td>
</tr>
<tr>
<td>A</td>
<td>–</td>
<td>5.080</td>
<td>–</td>
<td>0.200</td>
<td></td>
<td></td>
</tr>
<tr>
<td>A₁</td>
<td>0.508</td>
<td>–</td>
<td>0.020</td>
<td>–</td>
<td></td>
<td></td>
</tr>
<tr>
<td>A₂</td>
<td>3.175</td>
<td>4.064</td>
<td>0.125</td>
<td>0.160</td>
<td>0.014</td>
<td>0.022</td>
</tr>
<tr>
<td>B</td>
<td>0.356</td>
<td>0.559</td>
<td>0.014</td>
<td>0.022</td>
<td></td>
<td></td>
</tr>
<tr>
<td>B₁</td>
<td>1.270</td>
<td>1.778</td>
<td>Typical</td>
<td>0.050</td>
<td>0.070</td>
<td>Typical</td>
</tr>
<tr>
<td>C</td>
<td>0.2032</td>
<td>0.381</td>
<td>Typical</td>
<td>0.008</td>
<td>0.015</td>
<td>Typical</td>
</tr>
<tr>
<td>D</td>
<td>35.560</td>
<td>37.084</td>
<td></td>
<td>1.400</td>
<td>1.460</td>
<td></td>
</tr>
<tr>
<td>D₁</td>
<td>33.020</td>
<td>33.020</td>
<td>Reference</td>
<td>1.300</td>
<td>1.300</td>
<td>Reference</td>
</tr>
<tr>
<td>E</td>
<td>15.240</td>
<td>15.875</td>
<td>0.600</td>
<td>0.625</td>
<td></td>
<td></td>
</tr>
<tr>
<td>E₁</td>
<td>12.827</td>
<td>13.970</td>
<td>0.505</td>
<td>0.550</td>
<td></td>
<td></td>
</tr>
<tr>
<td>e₁</td>
<td>2.489</td>
<td>2.591</td>
<td>Typical</td>
<td>0.098</td>
<td>0.102</td>
<td>Typical</td>
</tr>
<tr>
<td>eA</td>
<td>15.240</td>
<td>15.240</td>
<td>Reference</td>
<td>0.600</td>
<td>0.600</td>
<td>Reference</td>
</tr>
<tr>
<td>eB</td>
<td>15.240</td>
<td>17.272</td>
<td>0.600</td>
<td>0.680</td>
<td></td>
<td></td>
</tr>
<tr>
<td>L</td>
<td>2.921</td>
<td>3.683</td>
<td>0.115</td>
<td>0.145</td>
<td></td>
<td></td>
</tr>
<tr>
<td>N</td>
<td>28</td>
<td>28</td>
<td>28</td>
<td>28</td>
<td></td>
<td></td>
</tr>
<tr>
<td>S</td>
<td>0.889</td>
<td>–</td>
<td>0.035</td>
<td>–</td>
<td></td>
<td></td>
</tr>
<tr>
<td>S₁</td>
<td>0.508</td>
<td>–</td>
<td>0.020</td>
<td>–</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Package Group: Plastic Dual-In-Line (PLA)

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Millimeters</th>
<th>Inches</th>
<th>Notes</th>
<th>Millimeters</th>
<th>Inches</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>α</td>
<td>0°</td>
<td>10°</td>
<td></td>
<td>0°</td>
<td>10°</td>
<td></td>
</tr>
<tr>
<td>A</td>
<td>3.63</td>
<td>4.57</td>
<td>.143</td>
<td>.180</td>
<td></td>
<td></td>
</tr>
<tr>
<td>A1</td>
<td>.38</td>
<td></td>
<td>.015</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>A2</td>
<td>3.25</td>
<td>3.65</td>
<td>.128</td>
<td>.140</td>
<td></td>
<td></td>
</tr>
<tr>
<td>B</td>
<td>.41</td>
<td>.56</td>
<td>.016</td>
<td>.022</td>
<td></td>
<td>Typical</td>
</tr>
<tr>
<td>B1</td>
<td>1.02</td>
<td>1.65</td>
<td></td>
<td>.040</td>
<td>.065</td>
<td></td>
</tr>
<tr>
<td>B2</td>
<td>.762</td>
<td>1.02</td>
<td></td>
<td>.030</td>
<td>.040</td>
<td>4 places</td>
</tr>
<tr>
<td>B3</td>
<td>.20</td>
<td>.51</td>
<td></td>
<td>.008</td>
<td>.020</td>
<td>4 places</td>
</tr>
<tr>
<td>C</td>
<td>.20</td>
<td>.33</td>
<td></td>
<td>.008</td>
<td>.013</td>
<td>Typical</td>
</tr>
<tr>
<td>D</td>
<td>34.16</td>
<td>35.43</td>
<td>1.345</td>
<td>1.395</td>
<td></td>
<td></td>
</tr>
<tr>
<td>D1</td>
<td>33.02</td>
<td>33.02</td>
<td></td>
<td>1.300</td>
<td>1.300</td>
<td>Reference</td>
</tr>
<tr>
<td>E</td>
<td>7.87</td>
<td>8.38</td>
<td>.310</td>
<td>.330</td>
<td></td>
<td></td>
</tr>
<tr>
<td>E1</td>
<td>7.1</td>
<td>7.52</td>
<td>.280</td>
<td>.296</td>
<td></td>
<td></td>
</tr>
<tr>
<td>e1</td>
<td>2.54</td>
<td>2.54</td>
<td></td>
<td>.100</td>
<td>.100</td>
<td>Typical</td>
</tr>
<tr>
<td>eA</td>
<td>7.87</td>
<td>7.87</td>
<td></td>
<td>.310</td>
<td>.310</td>
<td>Reference</td>
</tr>
<tr>
<td>eB</td>
<td>8.64</td>
<td>9.65</td>
<td>.340</td>
<td>.380</td>
<td></td>
<td></td>
</tr>
<tr>
<td>L</td>
<td>3.18</td>
<td>3.68</td>
<td>.125</td>
<td>.145</td>
<td></td>
<td></td>
</tr>
<tr>
<td>N</td>
<td>28</td>
<td>-</td>
<td>28</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>S</td>
<td>.58</td>
<td>1.22</td>
<td>.023</td>
<td>.048</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
PACKAGING DIAGRAMS AND DIMENSIONS (CONT.)

19.4 18-LEAD PLASTIC SURFACE MOUNT (SOIC - WIDE, 300 mil BODY)

--- Diagram of 18-lead plastic SOIC-Wide packaging ---

--- Table of dimensions and tolerances ---

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Min (mm)</th>
<th>Max (mm)</th>
<th>Notes</th>
<th>Min (in)</th>
<th>Max (in)</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>α</td>
<td>0°</td>
<td>8°</td>
<td>Notes</td>
<td>0°</td>
<td>8°</td>
<td>Notes</td>
</tr>
<tr>
<td>A</td>
<td>2.3622</td>
<td>2.6416</td>
<td></td>
<td>0.093</td>
<td>0.104</td>
<td></td>
</tr>
<tr>
<td>A₁</td>
<td>0.1016</td>
<td>0.2997</td>
<td></td>
<td>0.004</td>
<td>0.0118</td>
<td></td>
</tr>
<tr>
<td>B</td>
<td>0.3556</td>
<td>0.4826</td>
<td></td>
<td>0.014</td>
<td>0.019</td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>0.2413</td>
<td>0.3175</td>
<td></td>
<td>0.0095</td>
<td>0.0125</td>
<td></td>
</tr>
<tr>
<td>D</td>
<td>11.3538</td>
<td>11.7348</td>
<td></td>
<td>0.447</td>
<td>0.462</td>
<td></td>
</tr>
<tr>
<td>E</td>
<td>7.4168</td>
<td>7.5946</td>
<td></td>
<td>0.292</td>
<td>0.299</td>
<td></td>
</tr>
<tr>
<td>e</td>
<td>1.270</td>
<td>1.270</td>
<td>Typical</td>
<td>0.050</td>
<td>0.050</td>
<td>Typical</td>
</tr>
<tr>
<td>H</td>
<td>10.0076</td>
<td>10.6426</td>
<td></td>
<td>0.394</td>
<td>0.419</td>
<td></td>
</tr>
<tr>
<td>h</td>
<td>0.381</td>
<td>0.762</td>
<td></td>
<td>0.015</td>
<td>0.030</td>
<td></td>
</tr>
<tr>
<td>L</td>
<td>0.4064</td>
<td>1.143</td>
<td></td>
<td>0.016</td>
<td>0.045</td>
<td></td>
</tr>
<tr>
<td>N</td>
<td>18</td>
<td>18</td>
<td></td>
<td>18</td>
<td>18</td>
<td></td>
</tr>
<tr>
<td>CP</td>
<td>–</td>
<td>0.1016</td>
<td></td>
<td>–</td>
<td>0.004</td>
<td></td>
</tr>
</tbody>
</table>

--- Notes ---

- Millimeters and inches are given for reference.
- Typical values are provided for certain dimensions.
- Notes in parentheses indicate additional information.

--- Package Group: Plastic SOIC (SO) ---

© 1992 Microchip Technology Inc.
PACKAGE OUTLINES (CONT.)

19.5 28-LEAD PLASTIC SURFACE MOUNT (SOIC - WIDE, 300 mil BODY)

Package Group: Plastic SOIC (SO)

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Millimeters</th>
<th>Inches</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Min</td>
<td>Max</td>
</tr>
<tr>
<td>α</td>
<td>0°</td>
<td>8°</td>
</tr>
<tr>
<td>A</td>
<td>2.3622</td>
<td>2.6416</td>
</tr>
<tr>
<td>A₁</td>
<td>0.1016</td>
<td>0.2997</td>
</tr>
<tr>
<td>B</td>
<td>0.3556</td>
<td>0.4826</td>
</tr>
<tr>
<td>C</td>
<td>0.2413</td>
<td>0.3175</td>
</tr>
<tr>
<td>D</td>
<td>17.7038</td>
<td>18.0848</td>
</tr>
<tr>
<td>E</td>
<td>7.4168</td>
<td>7.5946</td>
</tr>
<tr>
<td>e</td>
<td>1.270</td>
<td>1.270</td>
</tr>
<tr>
<td>H</td>
<td>10.0076</td>
<td>10.6426</td>
</tr>
<tr>
<td>h</td>
<td>0.381</td>
<td>0.762</td>
</tr>
<tr>
<td>L</td>
<td>0.4064</td>
<td>1.143</td>
</tr>
<tr>
<td>N</td>
<td>28</td>
<td>28</td>
</tr>
<tr>
<td>CP</td>
<td>–</td>
<td>0.1016</td>
</tr>
</tbody>
</table>

© 1992 Microchip Technology Inc.
### 20-LEAD PLASTIC SURFACE MOUNT (SSOP - .209 mil BODY 5.30 mm)

#### Symbol List for Shrink Small Outline Package Parameter

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description of Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>α</td>
<td>Angular spacing between min. and max. lead positions measured at the gauge plane</td>
</tr>
<tr>
<td>A</td>
<td>Distance between seating plane to highest point of body</td>
</tr>
<tr>
<td>A₁</td>
<td>Distance between seating plane and base plane</td>
</tr>
<tr>
<td>B</td>
<td>Width of terminals</td>
</tr>
<tr>
<td>C</td>
<td>Thickness of terminals</td>
</tr>
<tr>
<td>D</td>
<td>Largest overall package parameter of length</td>
</tr>
<tr>
<td>E</td>
<td>Largest overall package width parameter not including leads</td>
</tr>
<tr>
<td>e</td>
<td>Linear spacing of true minimum lead position center line to center line</td>
</tr>
<tr>
<td>H</td>
<td>Largest overall package dimension of width</td>
</tr>
<tr>
<td>L</td>
<td>Length of terminal for soldering to a substrate</td>
</tr>
<tr>
<td>N</td>
<td>Total number of potentially useable lead positions</td>
</tr>
<tr>
<td>CP</td>
<td>Seating plane coplanarity</td>
</tr>
</tbody>
</table>

#### Package Group: Plastic SSOP

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Millimeters</th>
<th>Inches</th>
</tr>
</thead>
<tbody>
<tr>
<td>α</td>
<td>Min</td>
<td>Max</td>
</tr>
<tr>
<td>A</td>
<td>1.73</td>
<td>1.99</td>
</tr>
<tr>
<td>A₁</td>
<td>0.05</td>
<td>0.21</td>
</tr>
<tr>
<td>B</td>
<td>0.25</td>
<td>0.38</td>
</tr>
<tr>
<td>C</td>
<td>0.13</td>
<td>0.22</td>
</tr>
<tr>
<td>D</td>
<td>7.07</td>
<td>7.33</td>
</tr>
<tr>
<td>E</td>
<td>5.20</td>
<td>5.35</td>
</tr>
<tr>
<td>e</td>
<td>0.65</td>
<td>0.65</td>
</tr>
<tr>
<td>H</td>
<td>7.65</td>
<td>7.90</td>
</tr>
<tr>
<td>L</td>
<td>0.55</td>
<td>0.95</td>
</tr>
<tr>
<td>N</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td>CP</td>
<td>-</td>
<td>0.1016</td>
</tr>
</tbody>
</table>

#### Notes:
1. Controlling parameter: mm.
2. All packages are gull wing lead form.
3. "D" and "E" are reference datums and do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .006 package ends and .010 on sides.
4. The chamfer on the body is optional. If it is not present, a visual index feature must be located within the crosshatched area to indicate pin 1 position.
5. Terminal numbers are shown for reference.
PACKAGE OUTLINES (CONT.)

19.7 28-LEAD PLASTIC SURFACE MOUNT (SSOP - .209 mil BODY 5.30 mm)

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Millimeters</th>
<th>Inches</th>
</tr>
</thead>
<tbody>
<tr>
<td>α</td>
<td>0° to 8°</td>
<td>0° to 8°</td>
</tr>
<tr>
<td>A</td>
<td>1.73 to 1.99</td>
<td>0.68 to 0.78</td>
</tr>
<tr>
<td>A₁</td>
<td>0.05 to 0.21</td>
<td>0.002 to 0.008</td>
</tr>
<tr>
<td>B</td>
<td>0.25 to 0.38</td>
<td>0.010 to 0.015</td>
</tr>
<tr>
<td>C</td>
<td>0.13 to 0.22</td>
<td>0.005 to 0.009</td>
</tr>
<tr>
<td>D</td>
<td>10.07 to 10.33</td>
<td>0.397 to 0.407</td>
</tr>
<tr>
<td>E</td>
<td>5.20 to 5.38</td>
<td>0.205 to 0.212</td>
</tr>
<tr>
<td>e</td>
<td>0.65 to 0.65</td>
<td>Typical 0.256 to 0.256</td>
</tr>
<tr>
<td>H</td>
<td>7.65 to 7.90</td>
<td>0.301 to 0.311</td>
</tr>
<tr>
<td>L</td>
<td>0.55 to 0.95</td>
<td>0.022 to 0.037</td>
</tr>
<tr>
<td>N</td>
<td>28 to 28</td>
<td>28 to 28</td>
</tr>
<tr>
<td>CP</td>
<td>-</td>
<td>0.1016</td>
</tr>
</tbody>
</table>

Symbol List for Shrink Small Outline Package Parameter

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description of Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>α</td>
<td>Angular spacing between min. and max. lead positions measured at the guage plane</td>
</tr>
<tr>
<td>A</td>
<td>Distance between seating plane to highest point of body</td>
</tr>
<tr>
<td>A₁</td>
<td>Distance between seating plane and base plane</td>
</tr>
<tr>
<td>B</td>
<td>Width of terminals</td>
</tr>
<tr>
<td>C</td>
<td>Thickness of terminals</td>
</tr>
<tr>
<td>D</td>
<td>Largest overall package parameter of length</td>
</tr>
<tr>
<td>E</td>
<td>Largest overall package width parameter not including leads</td>
</tr>
<tr>
<td>e</td>
<td>Linear spacing of true minimum lead position center line to center line</td>
</tr>
<tr>
<td>H</td>
<td>Largest overall package dimension of width</td>
</tr>
<tr>
<td>L</td>
<td>Length of terminal for soldering to a substrate</td>
</tr>
<tr>
<td>N</td>
<td>Total number of potentially useable lead positions</td>
</tr>
<tr>
<td>CP</td>
<td>Seating plane coplanarity</td>
</tr>
</tbody>
</table>

Notes:
1. Controlling parameter: mm.
2. All packages are gull wing lead form.
3. "D" and "E" are reference datums and do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .006 package ends and .010 on sides.
4. The chamfer on the body is optional. If it is not present, a visual index feature must be located within the crosshatched area to indicate pin 1 position.
5. Terminal numbers are shown for reference.
19.8 PACKAGE MARKING INFORMATION

**18L PDIP**

```
MMMMMMMMMMMMXXX
MMMMMMMMMMMMXXX
AABB CDE
```

**Example**

```
PIC16C56-RCI/P456
9123 CBA
```

**18L SOIC**

```
MMMMMMMMM
MMMMMMMMM
AABB CDE
```

**Example**

```
PIC16C54-XTI/S0218
9118 CDK
```

**28L SOIC**

```
XXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXX
AABB CDE
```

**Example**

```
PIC16C57-XT/S0
9051 CBK
```

**28L PDIP (.300 mil)**

```
XXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXX
AABB CDE
```

**Example**

```
PIC16C56-RCI/P456
9123 CBA
```

**28L SSOP**

```
XXXXXXXXXXXX
XXXXXXXXXXXX
AABB CDE
```

**Example**

```
PIC16C57-XT
9225 CBK
```

**20L SSOP**

```
XXXXXXXX
XXXXXXXX
AABB CDE
```

**Example**

```
PIC16C54
XTI/218
9051 CBP
```

---

**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. |
| D      | Mask revision number |
| 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 rev #, 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.
19.8 PACKAGE MARKING INFORMATION (Cont.)

**28L PDIP (.600 mil)**

```
MMMMMMMMMMMMXXX
MMMMMMMMXXXXXXX
XXXXXXXXXXXXXXXXXX
AABB CDE
```

**Example**

```
PIC16C55-XTI/P126
9042 CDA
```

**18L Cerdip**

```
Microchip
```

**Example**

```
PIC16C54
9101 CBA
```

**28L Cerdip**

```
Microchip
```

**Example**

```
PIC16C57
9038 CBA
```

**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.
- **D** Mask revision number
- **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 rev #, 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.
20.0 DEVELOPMENT SUPPORT

Overview: The PIC16C5X family is supported by a variety of development tools:
- High Performance In-Circuit Emulator (PICMASTER)
- Low Cost Production Quality Programmer (PICPRO II)
- Microchip’s Universal Production Quality Programmer (PRO MASTER)
- PC Based Assembler (PICALC)
- PC Based Simulator (PICSIM)

Microchip offers several bundled development tool systems for convenience and cost benefit to the customer. For PIC16C5X the following are available:

- PICPAK II: Assembler, Simulator, PICPRO II Programmer and windowed samples for development.
- PICMASTER-16: Assembler, Simulator, PICPRO II Programmer, PICMASTER emulator and windowed samples for development.

20.1 PICMASTER™: High Performance Universal In-Circuit Emulator System

The PICMASTER Universal In-Circuit Emulator System is intended to provide the product development engineer with a complete microcontroller design tool set for all microcontrollers in the PIC16CXX and PIC17CXX families. This system currently supports the PIC16CR54, PIC16C54, PIC16C55, PIC16C56 and PIC16C57, and PIC17C42 processors. PIC16C71 support is planned.

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 PIC16CXX and PIC17CXX microcontrollers.

The Emulator System is designed to operate on low-cost PC compatible machines ranging from 80286-AT class ISA-bus systems through the new 80486 EISA-bus machines. The development software runs in the Microsoft Windows® 3.X environment, allowing the operator access to a wide range of supporting software and accessories.

Provided with the PICMASTER System is a high performance real-time In-Circuit Emulator, a programmer unit and a macro assembler program.

Coupled with the user’s choice of text editor, the system is ready for development of products containing any of Microchip’s microcontroller products.

A “Quick Start” PIC Product Sample Pak containing user programmable parts is included for additional convenience.

Microchip provides additional customer support to developers through an Electronic Bulletin Board System (EBBS). Customers have access to the latest updates in software as well as application source code examples. Consult your local sales representative for information on accessing the BBS system.

20.1.1 Host System Requirements:

The PICMASTER has been designed as a real-time emulation system with advanced features generally found on more expensive development tools. The AT platform and Windows 3.X environment was chosen to best make these features available to you the end user. To properly take advantage of these features, PICMASTER requires installation on a system having the following minimum configuration:

- PC AT compatible machine: 80286, 386SX, 386DX, or 80486 with ISA or EISA Bus.
- EGA, VGA, 8514/A, Hercules graphic card (EGA or higher recommended).
- MSDOS / PCDOS version 3.1 or greater.
- Microsoft Windows® version 3.0 or greater operating in either standard or 386 enhanced mode.
- 1 Mbyte RAM (2 Mbytes recommended).
- One 5.25" floppy disk drive.
- One 8-bit PC AT (ISA) I/O expansion slot (half size)
- Microsoft® mouse or compatible (highly recommended).

20.1.2 Emulator System Components:

The PICMASTER Emulator Universal System consists primarily of 4 major components:

- Host-Interface Card: The PC Host Interface Card connects the emulator system to an IBM PC compatible system. This high-speed parallel interface requires a single half-size standard AT / ISA slot in the host system. A 37-conductor cable connects the interface card to the external Emulator Control Pod.
- Emulator Control Pod: The Emulator Control Pod contains all emulation and control logic common to all microcontroller devices. Emulation memory, trace memory, event and cycle timers, and trace/breakpoint logic are contained here. The Pod controls and interfaces to an interchangeable target-specific emulator probe via a 14" precision ribbon cable.
- Target-specific Emulator Probe: A probe specific to microcontroller family to be emulated is installed on the ribbon cable coming from the control pod. This probe configures the universal system for emulation of a specific microcontroller.
- PC Host Emulation Control Software: Host software necessary to control and provide a working user interface is the last major component of the system. The emulation software runs in the Windows 3.X environment, and provides the user with full display, alter, and control of the system under emulation. The Control Software is also universal to all microcontroller families.
The Windows 3.X System is a multitasking operating system which will allow 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. Dynamic Data Exchange (DDE), a feature of Windows 3.X, will be available in this and future versions of the software. 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 3.X, two or more PICMASTER emulators can run simultaneously on the same PC making development of multi-microcontroller systems possible (e.g., a system containing a PIC16Cxx processor and a PIC17Cxx processor).

FIGURE 20.1.1 - PICMASTER

![PICMASTER Emulator Pod](image)

FIGURE 20.1.2 - PICMASTER SYSTEM CONFIGURATION

![PICMASTER System Configuration Diagram](image)
20.2 PICPAK-11™ Development Kit

When real time or in-circuit emulation is not required for code development the PICPAK-11 (PIC Applications Kit) offers the right solution. This very low cost PC hosted software development tool combines the power and versatility of the PICALC assembler and PICSIM simulator software tools to compile, execute, debug and analyze microcode in a PC hosted environment. Microcode debugging capability includes software trace, breakpoints, symbolic debug and stimulus file generation. An EPROM PIC Programmer (PICPRO II) and "Quick Start" PIC16C5X product sample PAK is included for final code verification.

20.3 PICALC Cross-Assembler

The PIC Cross Assembler PICALC is a PC hosted software development tool supporting the PIC16C5X CMOS series and PIC16C5X/7X NMOS series microcontrollers. PICALC offers a full featured Macro and Conditional assembly capability. It can also generate various object code formats including several Hex formats to support Microchip's proprietary development tools as well as third party tools. Also supports Hex (default), Decimal and Octal Source and listing formats. An assembler users manual is available for detailed support.

20.4 PICSIM™ Software Simulator

PICSIM is a software tool which allows for PIC16C5X code development in a PC host environment. The PICSIM software allows you to simulate the PIC16C5X series microcontroller products on an instruction level. On any given instruction, the user may examine or modify any of the data areas within the PIC 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. PICSIM uses two forms of symbolic debugging: an internal symbol table for disassembling opcodes and the displaying of source code from a listing file. PICSIM offers the low cost flexibility to develop and debug code outside of the laboratory environment making it an excellent multi project software development tool.

20.5 PICPRO™ II Programmer

PICPRO II is a production quality programmer with both stand-alone and PC based operation. The PICPRO II will program the entire family of PIC16C5X series of 8-bit microcontrollers. It can read, program, verify, and code protect parts without the need of a PC host. Its EEPROM memory holds programming data and parametric information even when power is removed making it ideal for duplicating ICs. It can also operate with a PC host and do complete read, program, verify, as in stand-alone mode with the additional features of program buffer editing, serialization of both code-protected and non code-protected parts. The PICPRO II comes with both 28 and 18 pin zero insertion force programming sockets on a removable socket module. Additional socket modules are available for the SOIC and PLCC packages. The PICPRO II conforms fully to Microchip's Programming Algorithm. Its programmable Vcc and VPP supplies allow the PICPRO II to support PIC microcontrollers with various operating voltage ranges.

20.6 PRO MASTER™

The PRO MASTER programmer is a production quality programmer capable of operating in stand alone mode as well as PC-hosted mode.

The PRO MASTER has programmable Vdd and Vpp supplies which allows it to verify the PIC 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 MASTER can read, verify or program a part. It can also set fuse configuration and code-protect in this mode. It's EEPROM memory holds data and parametric information even when powered down. It is ideal for low to moderate volume production.

In PC-hosted mode, the PRO MASTER connects to the PC via one of the COM (RS232) ports. A 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 fuse 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, 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 MASTER has a modular "programming socket module". Different socket modules are required for different processor types and/or package types. It is planned that the PRO MASTER will support all current and future PIC16CXX and PIC17CXX processors. Currently socket modules are available for the PIC16C54, PIC16C55, PIC16C56, PIC16C57, PIC17C42 and the PIC16C71.

21.0 EPROM PROGRAMMING

21.1 Prototype Programmers

Microchip's proprietary PIC16C5X series Development System and PICPRO, were not designed for high volume production programming but were designed strictly to support engineering development level programming of PIC16C5X EPROM and OTP units. Microchip assumes no responsibility for the replacement of programming rejects when these development tools are used to support high volume production level programming.

21.2 Production Quality Programmers

Microchip's proprietary PICPRO II programmer can be used for reliable programming for production. High volume programming is also supported by production quality programmers from third party sources. See table 21.2.1.

Microchip assumes no responsibility for replacing defective units related to mechanical and/or electrical problems of any third party programming equipment or the improper use of such equipment.

Programming of the code protection bit (also called "security bit" or "security fuse") implies that the contents of the PIC16C5X EPROM can no longer be verified, thus making programming related failure analysis an impossibility.

Microchip warrants that PIC16C5X units will not exceed a programming failure rate of 1% of shipment quality. Programming related failures beyond this level can be returned for replacement, again, if the security bit has not been programmed.

21.3 Gang Programmers

Gang programmers are available from third party sources. See table 21.2.1.

21.4 Factory Programming

High volume factory programming (QTP) is an available service from Microchip Technology. A small price adder and minimum quantity requirements apply.

### TABLE 21.2.1 LIST OF 3RD PARTY PROGRAMMERS*

<table>
<thead>
<tr>
<th>Company</th>
<th>Model</th>
<th>Contact</th>
<th>Company</th>
<th>Model</th>
<th>Contact</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADVIN Systems, Inc.</td>
<td>PILOT™-U40</td>
<td>408-984-8600 U.S.</td>
<td>HI-LOT</td>
<td>ALL-03</td>
<td>02 7640215 Taiwan</td>
</tr>
<tr>
<td>Bardine Products Ltd.</td>
<td>Micro-Burner™</td>
<td>604-988-9853 Canada</td>
<td>Logical Devices, Inc.</td>
<td>ALLPRO™-88</td>
<td>800-331-7766 U.S.</td>
</tr>
<tr>
<td>BP Microsystems</td>
<td>CP-1128™</td>
<td>800-225-2102 U.S.</td>
<td></td>
<td></td>
<td>305-974-0967 U.S.</td>
</tr>
<tr>
<td>Citadel Products Ltd.</td>
<td>PC-82</td>
<td>44-819-511-848 U.K.</td>
<td>Maple Technology, Ltd</td>
<td>MOP-200</td>
<td>44-666-825-146 U.K.</td>
</tr>
<tr>
<td>Elan Digital Systems Ltd.</td>
<td>EF-PER™ 5000 Series Gang Programmer</td>
<td>0489 579 799 U.K.</td>
<td>(408) 946 3864 U.S.</td>
<td>SMS</td>
<td>Sprint™ Expert</td>
</tr>
</tbody>
</table>

* As of July, 1992
† Product is available but Microchip has not evaluated it yet (as of July, 1992).
Index

Absolute maximum ratings .................................................29
AC characteristics (XT,RC,HS,LP) COM/IND ..................35
Block diagrams:
  Chip ........................................................................4
  I/O Pin ..................................................................13
  Power On Reset ....................................................26
  RTCC (Simplified) .................................................6
  RTCC & WDT ........................................................17
Brown-out protection circuit ...........................................25
Code protection ..........................................................28
Configuration fuses ......................................................27
Data memory map ........................................................7
DC characteristics (XT,RC,HS,LP) COM ..................30,33,35
DC characteristics (XT,RC,HS,LP) IND ..................31,33,35
Development tools .......................................................54
External power on reset circuit .....................................25
Features overview .......................................................1
File register descriptions
  f0 ..........................................................................6
  f1 ..........................................................................6
  f2 ..........................................................................8
  f3 ..........................................................................11
  f4 ..........................................................................12
I/O ports .............................................................13,14
ID locations ............................................................27
Indirect addressing .......................................................6
Instruction set ............................................................18
OPTION register .......................................................15
Oscillator ............................................................23,24,25
Oscillator Start-up Timer ..........................................25
OTP devices ............................................................5
Package information .................................................45,46,47,48,49,50,51
Page select (Program memory) ....................................9,10
PD bit .....................................................................12
Pin-out information .....................................................1,29
Power Down mode (SLEEP) .......................................27
Power On Reset .......................................................25,26,27
Prescaler (RTCC/WDT) .............................................16,17
Program Counter ......................................................16,18
Program memory map ................................................10
Programming information ..........................................57
QTP devices ............................................................5,6
Real Time Clock/Counter (RTCC) ...............................8,9,16,17
RESET ....................................................................16
SLEEP ....................................................................27
Stack ....................................................................9
Status register ..........................................................11
Timing diagrams
  I/O pin .............................................................13,36
  Oscillator Start-up timing .......................................36
  Power On Reset ....................................................25,26
  RTCC timing ........................................................8,9,36
TO bit .....................................................................12
TRIS registers ........................................................15

Typical characteristics graphs
  IDD vs freq ..........................................................41
  IOH vs VOH .........................................................43
  IOL vs VOL ........................................................43
  IPD vs VDD ........................................................38,39
  RC osc freq vs temp .........................................37
  RC osc freq vs VDD ...........................................37,38
  VTH of OSC1 Input vs Vdd .................................39
  Transconductance of HS Oscillator vs Vdd ........42
  Transconductance of LP Oscillator vs Vdd ..........42
  Transconductance of XT Oscillator vs Vdd ........42
  WDT Timer Time-out Period vs VDD .............42

UV Erasable devices ....................................................5,6
W register ..............................................................15
WDT .................................................................23

Trademarks:
PIC is a registered trademark of Microchip Technology Incorporated.
PIC-PAK, PICPRO, PICMASTER, PRO MASTER, and PICSIM are trademarks of Microchip Technology Incorporated.
IBM PC is a trademark of IBM Corporation.
MS DOS and Microsoft Windows are registered trademarks of Microsoft Corporation.
SALES AND SUPPORT

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. For the currently available code-combinations, refer to previous page.

PART NO. - XX X /XX XXX

Pattern: 3-Digit Pattern Code for QTP (blank for OTP and Window Parts)

Package:
P = PDIP
SO = 300 mil SOIC (Gull Wing Lead)
SP = 28L PDIP (300 mil)
SS = SSOP (209 mil)

Temperature:
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

Oscillator:
RC = Commercial temp.
XT = XT oscillator, commercial temp., PDIP, QTP pattern
HS = HS oscillator
LP = LP oscillator

Examples:
a) PIC16C54 - XT/PXXX = "XT" oscillator, commercial temp., PDIP, QTP pattern
b) PIC16C55 - XTI/SS = "XT" oscillator, industrial temp., SOIC (OTP device)
c) PIC16C55 - JW = Commercial temp.
JW* = CERDIP Window
e) PIC16C54-LPE/SS = "LP oscillator", Automotive temp., SSOP package

Notes: * For UV-erasable devices, the oscillator type is RC by default (= erased device). The user can select XT, HS, or LP oscillators by programming the appropriate fuses.