Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

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

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	PIC
Core Size	8-Bit
Speed	4MHz
Connectivity	-
Peripherals	POR, WDT
Number of I/O	11
Program Memory Size	1.5KB (1K x 12)
Program Memory Type	OTP
EEPROM Size	-
RAM Size	72 x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 5.5V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	14-SOIC (0.154", 3.90mm Width)
Supplier Device Package	14-SOIC
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16c505-04e-sl

TABLE 3-1: PIC16C505 PINOUT DESCRIPTION

Name	DIP Pin #	SOIC Pin #	I/O/P Type	Buffer Type	Description
RB0	13	13	I/O	TTL/ST	Bi-directional I/O port/ serial programming data. Can be software programmed for internal weak pull-up and wake-up from SLEEP on pin change. This buffer is a Schmitt Trigger input when used in serial programming mode.
RB1	12	12	I/O	TTL/ST	Bi-directional I/O port/ serial programming clock. Can be software programmed for internal weak pull-up and wake-up from SLEEP on pin change. This buffer is a Schmitt Trigger input when used in serial programming mode.
RB2	11	11	I/O	TTL	Bi-directional I/O port.
RB3/$\overline{\text{MCLR}}$/VPP	4	4	I	TTL/ST	Input port/master clear (reset) input/programming voltage input. When configured as $\overline{\text{MCLR}}$, this pin is an active low reset to the device. Voltage on $\overline{\text{MCLR}}$/VPP must not exceed VDD during normal device operation. Can be software programmed for internal weak pull-up and wake-up from SLEEP on pin change. Weak pull-up only when configured as RB3. ST when configured as $\overline{\text{MCLR}}$.
RB4/OSC2/CLKOUT	3	3	I/O	TTL	Bi-directional I/O port/oscillator crystal output. Connections to crystal or resonator in crystal oscillator mode (XT and LP modes only, RB4 in other modes). Can be software programmed for internal weak pull-up and wake-up from SLEEP on pin change. In EXTRC and INTRC modes, the pin output can be configured to CLKOUT, which has 1/4 the frequency of OSC1 and denotes the instruction cycle rate.
RB5/OSC1/CLKIN	2	2	I/O	TTL/ST	Bidirectional IO port/oscillator crystal input/external clock source input (RB5 in Internal RC mode only, OSC1 in all other oscillator modes). TTL input when RB5, ST input in external RC oscillator mode.
RC0	10	10	I/O	TTL	Bi-directional I/O port.
RC1	9	9	I/O	TTL	Bi-directional I/O port.
RC2	8	8	I/O	TTL	Bi-directional I/O port.
RC3	7	7	I/O	TTL	Bi-directional I/O port.
RC4	6	6	I/O	TTL	Bi-directional I/O port.
RC5/T0CKI	5	5	I/O	ST	Bi-directional I/O port. Can be configured as T0CKI.
VDD	1	1	P	—	Positive supply for logic and I/O pins
VSS	14	14	P	—	Ground reference for logic and I/O pins

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

4.2.2 SPECIAL FUNCTION REGISTERS

The Special Function Registers (SFRs) are registers used by the CPU and peripheral functions to control the operation of the device (Table 4-1).

The Special Function Registers can be classified into two sets. The Special Function Registers associated with the “core” functions are described in this section. Those related to the operation of the peripheral features are described in the section for each peripheral feature.

TABLE 4-1: SPECIAL FUNCTION REGISTER (SFR) SUMMARY

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on Power-On Reset	Value on All Other Resets(2)
00h	INDF	Uses contents of FSR to address data memory (not a physical register)								xxxx xxxx	uuuu uuuu
01h	TMR0	8-bit real-time clock/counter								xxxx xxxx	uuuu uuuu
02h(1)	PCL	Low order 8 bits of PC								1111 1111	1111 1111
03h	STATUS	RBWUF	—	PAO	$\overline{\text{TO}}$	$\overline{\text{PD}}$	Z	DC	C	0001 1xxx	q00q quuu(1)
04h	FSR	Indirect data memory address pointer								110x xxxx	11uu uuuu
05h	OSCCAL	CAL5	CAL4	CAL3	CAL2	CAL1	CAL0	—	—	1000 00--	uuuu uu--
N/A	TRISB	—	—	I/O control registers						--11 1111	--11 1111
N/A	TRISC	—	—	I/O control registers						--11 1111	--11 1111
N/A	OPTION	$\overline{\text{RBWU}}$	$\overline{\text{RBPU}}$	TOCS	TOSE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
06h	PORTB	—	—	RB5	RB4	RB3	RB2	RB1	RB0	--xx xxxx	--uu uuuu
07h	PORTC	—	—	RC5	RC4	RC3	RC2	RC1	RC0	--xx xxxx	--uu uuuu

Legend: Shaded cells not used by Port Registers, read as ‘0’, — = unimplemented, read as ‘0’, x = unknown, u = unchanged, q = depends on condition.

Note 1: If reset was due to wake-up on pin change, then bit 7 = 1. All other rests will cause bit 7 = 0.

Note 2: Other (non-power-up) resets include external reset through $\overline{\text{MCLR}}$, watchdog timer and wake-up on pin change reset.

4.6 Program Counter

As a program instruction is executed, the Program Counter (PC) will contain the address of the next program instruction to be executed. The PC value is increased by one every instruction cycle, unless an instruction changes the PC.

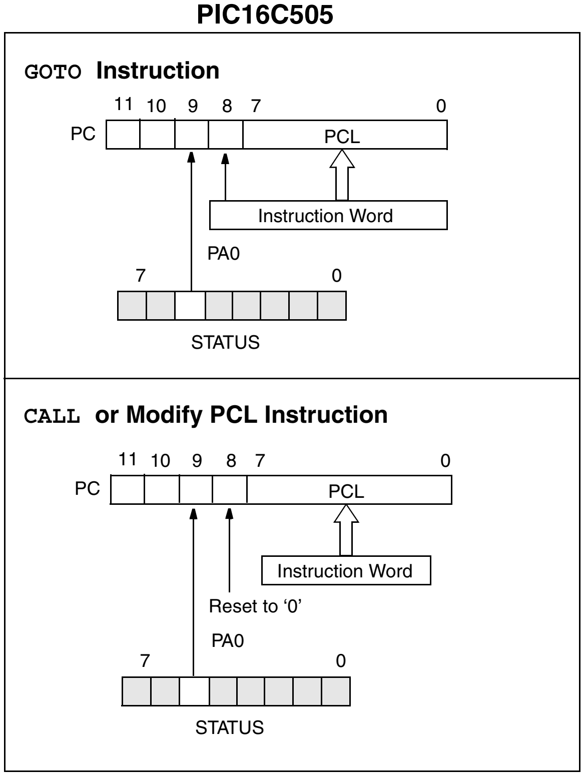
For a `GOTO` instruction, bits 8:0 of the PC are provided by the `GOTO` instruction word. The PC Latch (PCL) is mapped to PC<7:0>. Bit 5 of the STATUS register provides page information to bit 9 of the PC (Figure 4-3).

For a `CALL` instruction, or any instruction where the PCL is the destination, bits 7:0 of the PC again are provided by the instruction word. However, PC<8> does not come from the instruction word, but is always cleared (Figure 4-3).

Instructions where the PCL is the destination, or Modify PCL instructions, include `MOVWF PC`, `ADDWF PC`, and `BSF PC,5`.

Note:	Because PC<8> is cleared in the `CALL` instruction or any Modify PCL instruction, all subroutine calls or computed jumps are limited to the first 256 locations of any program memory page (512 words long).

FIGURE 4-3: LOADING OF PC BRANCH INSTRUCTIONS - PIC16C505



4.6.1 EFFECTS OF RESET

The Program Counter is set upon a RESET, which means that the PC addresses the last location in the last page (i.e., the oscillator calibration instruction.) After executing `MOVLW XX`, the PC will roll over to location 00h and begin executing user code.

The STATUS register page preselect bits are cleared upon a RESET, which means that page 0 is preselected.

Therefore, upon a RESET, a `GOTO` instruction will automatically cause the program to jump to page 0 until the value of the page bits is altered.

4.7 Stack

PIC16C505 devices have a 12-bit wide hardware push/pop stack.

A `CALL` instruction will `push` the current value of stack 1 into stack 2 and then `push` the current program counter value, incremented by one, into stack level 1. If more than two sequential `CALL`’s are executed, only the most recent two return addresses are stored.

A `RETLW` instruction will `pop` the contents of stack level 1 into the program counter and then copy stack level 2 contents into level 1. If more than two sequential `RETLW`’s are executed, the stack will be filled with the address previously stored in level 2. Note that the W register will be loaded with the literal value specified in the instruction. This is particularly useful for the implementation of data look-up tables within the program memory.

Note 1:	There are no STATUS bits to indicate stack overflows or stack underflow conditions.
Note 2:	There are no instructions mnemonics called `PUSH` or `POP`. These are actions that occur from the execution of the `CALL`, `RETLW`, and instructions.

© 1999-2012 Microchip Technology Inc.

6.1 Using Timer0 with an External Clock

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

6.1.1 EXTERNAL CLOCK SYNCHRONIZATION

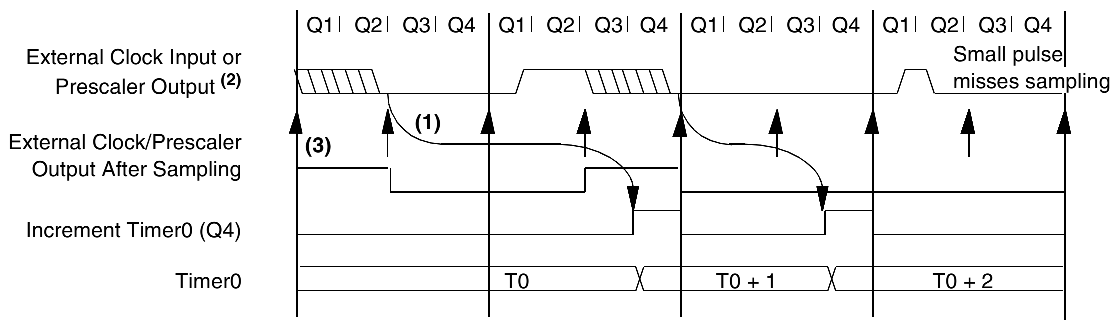
When no prescaler is used, the external clock input is the same as the prescaler output. The synchronization of T0CKI with the internal phase clocks is accomplished by sampling the prescaler output on the Q2 and Q4 cycles of the internal phase clocks (Figure 6-4). Therefore, it is necessary for T0CKI to be high for at least 2TOSC (and a small RC delay of 20 ns) and low for at least 2TOSC (and a small RC delay of 20 ns). Refer to the electrical specification of the desired device.

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

6.1.2 TIMER0 INCREMENT DELAY

Since the prescaler output is synchronized with the internal clocks, there is a small delay from the time the external clock edge occurs to the time the Timer0 module is actually incremented. Figure 6-4 shows the delay from the external clock edge to the timer incrementing.

FIGURE 6-4: TIMER0 TIMING WITH EXTERNAL CLOCK



Note 1: Delay from clock input change to Timer0 increment is 3Tosc to 7Tosc. (Duration of Q = Tosc). Therefore, the error in measuring the interval between two edges on Timer0 input = ± 4Tosc max.

2: External clock if no prescaler selected; prescaler output otherwise.

3: The arrows indicate the points in time where sampling occurs.

7.0 SPECIAL FEATURES OF THE CPU

What sets a microcontroller apart from other processors are special circuits to deal with the needs of real-time applications. The PIC16C505 microcontroller has a host of such features intended to maximize system reliability, minimize cost through elimination of external components, provide power saving operating modes and offer code protection. These features are:

- Oscillator selection
- Reset
  - Power-On Reset (POR)
  - Device Reset Timer (DRT)
  - Wake-up from SLEEP on pin change
- Watchdog Timer (WDT)
- SLEEP
- Code protection
- ID locations
- In-circuit Serial Programming
- Clock Out

The PIC16C505 has a Watchdog Timer, which can be shut off only through configuration bit WDTE. It runs off of its own RC oscillator for added reliability. If using HS, XT or LP selectable oscillator options, there is always an 18 ms (nominal) delay provided by the Device Reset Timer (DRT), intended to keep the chip in reset until the crystal oscillator is stable. If using INTRC or EXTRC, there is an 18 ms delay only on VDD power-up. With this timer on-chip, most applications need no external reset circuitry.

The SLEEP mode is designed to offer a very low current power-down mode. The user can wake-up from SLEEP through a change on input pins or through a Watchdog Timer time-out. Several oscillator options are also made available to allow the part to fit the application, including an internal 4 MHz oscillator. The EXTRC oscillator option saves system cost while the LP crystal option saves power. A set of configuration bits are used to select various options.

7.1 Configuration Bits

The PIC16C505 configuration word consists of 12 bits. Configuration bits can be programmed to select various device configurations. Three bits are for the selection of the oscillator type, one bit is the Watchdog Timer enable bit, and one bit is the $\overline{\text{MCLR}}$ enable bit. Seven bits are for code protection (Register 7-1).

REGISTER 7-1: CONFIGURATION WORD FOR PIC16C505

CP	CP	CP	CP	CP	CP	MCLRE	CP	WDTE	FOSC2	FOSC1	FOSC0
bit11	10	9	8	7	6	5	4	3	2	1	bit0

Register: CONFIG
Address(2): 0FFFh

bit 11-6, 4: **CP Code Protection bits** (1)(2)(3)

bit 5: **MCLRE:** RB3/$\overline{\text{MCLR}}$ pin function select
1 = RB3/$\overline{\text{MCLR}}$ pin function is $\overline{\text{MCLR}}$
0 = RB3/$\overline{\text{MCLR}}$ pin function is digital I/O, $\overline{\text{MCLR}}$ internally tied to VDD

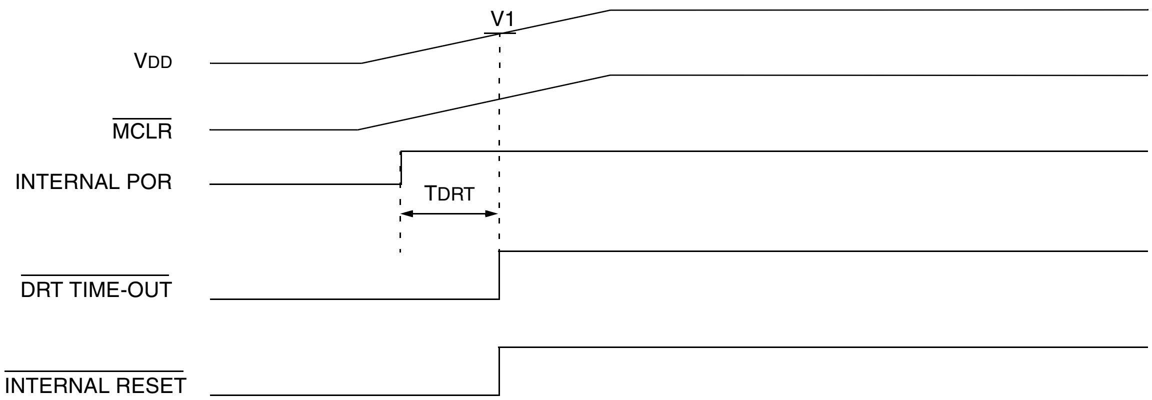
bit 3: **WDTE:** Watchdog timer enable bit
1 = WDT enabled
0 = WDT disabled

bit 2-0: **FOSC<1:0>:** Oscillator Selection bits
111 = external RC oscillator/CLKOUT function on RB4/OSC2/CLKOUT pin
110 = external RC oscillator/RB4 function on RB4/OSC2/CLKOUT pin
101 = internal RC oscillator/CLKOUT function on RB4/OSC2/CLKOUT pin
100 = internal RC oscillator/RB4 function on RB4/OSC2/CLKOUT pin
011 = invalid selection
010 = HS oscillator
001 = XT oscillator
000 = LP oscillator

Note 1: 03FFh is always uncode protected on the PIC16C505. This location contains the `MOVLWxx` calibration instruction for the INTRC.
2: Refer to the PIC16C505 Programming Specifications to determine how to access the configuration word. This register is not user addressable during device operation.
3: All code protect bits must be written to the same value.

© 1999-2012 Microchip Technology Inc.

FIGURE 7-10: TIME-OUT SEQUENCE ON POWER-UP ($\overline{MCLR}$ TIED TO V_{DD}): SLOW V_{DD} RISE TIME



Note: When V_{DD} rises slowly, the T_{DRT} time-out expires long before V_{DD} has reached its final value. In this example, the chip will reset properly if, and only if, $V1 \geq V_{DD}$ min.

7.5 Device Reset Timer (DRT)

In the PIC16C505, the DRT runs any time the device is powered up. DRT runs from RESET and varies based on oscillator selection and reset type (see Table 7-5).

The DRT operates on an internal RC oscillator. The processor is kept in RESET as long as the DRT is active. The DRT delay allows V_{DD} to rise above V_{DD} min. and for the oscillator to stabilize.

Oscillator circuits based on crystals or ceramic resonators require a certain time after power-up to establish a stable oscillation. The on-chip DRT keeps the device in a RESET condition for approximately 18 ms after $\overline{MCLR}$ has reached a logic high ($V_{IH}\overline{MCLR}$) level. Thus, programming RB3/$\overline{MCLR}$/V_{PP} as $\overline{MCLR}$ and using an external RC network connected to the $\overline{MCLR}$ input is not required in most cases, allowing for savings in cost-sensitive and/or space restricted applications, as well as allowing the use of the RB3/$\overline{MCLR}$/V_{PP} pin as a general purpose input.

The Device Reset time delay will vary from chip to chip due to V_{DD}, temperature and process variation. See AC parameters for details.

The DRT will also be triggered upon a Watchdog Timer time-out. This is particularly important for applications using the WDT to wake from SLEEP mode automatically.

Reset sources are POR, $\overline{MCLR}$, WDT time-out and Wake-up on pin change. (See Section 7.9.2, Notes 1, 2, and 3, page 37.)

7.6 Watchdog Timer (WDT)

The Watchdog Timer (WDT) is a free running on-chip RC oscillator, which does not require any external components. This RC oscillator is separate from the external RC oscillator of the RB5/OSC1/CLKIN pin and the internal 4 MHz oscillator. That means that the WDT will run even if the main processor clock has been stopped, for example, by execution of a `SLEEP` instruction. During normal operation or SLEEP, a WDT reset or wake-up reset generates a device RESET.

The $\overline{TO}$ bit (STATUS<4>) will be cleared upon a Watchdog Timer reset.

The WDT can be permanently disabled by programming the configuration bit WDTE as a '0' (Section 7.1). Refer to the PIC16C505 Programming Specifications to determine how to access the configuration word.

TABLE 7-5: DRT (DEVICE RESET TIMER PERIOD)

Oscillator Configuration	POR Reset	Subsequent Resets
IntRC & ExtRC	18 ms (typical)	300 µs (typical)
HS, XT & LP	18 ms (typical)	18 ms (typical)

 © 1999-2012 Microchip Technology Inc.

7.6.1 WDT PERIOD

The WDT has a nominal time-out period of 18 ms, (with no prescaler). If a longer time-out period is 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, a time-out period of a nominal 2.3 seconds can be realized. These periods vary with temperature, VDD and part-to-part process variations (see DC specs).

Under worst case conditions (VDD = Min., Temperature = Max., max. WDT prescaler), it may take several seconds before a WDT time-out occurs.

7.6.2 WDT PROGRAMMING CONSIDERATIONS

The `CLRWDT` instruction clears the WDT and the postscaler, if assigned to the WDT, and prevents it from timing out and generating a device RESET.

The `SLEEP` instruction resets the WDT and the postscaler, if assigned to the WDT. This gives the maximum SLEEP time before a WDT wake-up reset.

FIGURE 7-11: WATCHDOG TIMER BLOCK DIAGRAM

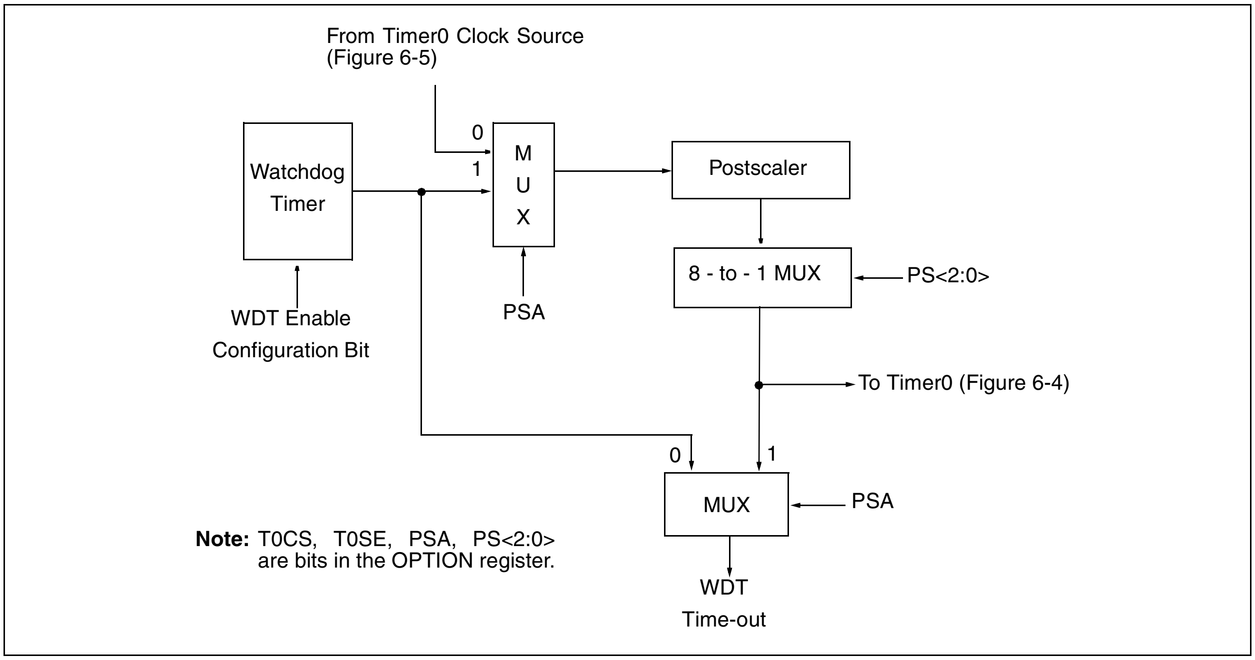


TABLE 7-6: SUMMARY OF REGISTERS ASSOCIATED WITH THE WATCHDOG TIMER

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on Power-On Reset	Value on All Other Resets
N/A	OPTION	$\overline{\text{RBWU}}$	$\overline{\text{RBPU}}$	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111

Legend: Shaded boxes = Not used by Watchdog Timer, — = unimplemented, read as '0', u = unchanged.

7.9 Power-Down Mode (SLEEP)

A device may be powered down (SLEEP) and later powered up (Wake-up from SLEEP).

7.9.1 SLEEP

The Power-Down mode is entered by executing a `SLEEP` instruction.

If enabled, the Watchdog Timer will be cleared but keeps running, the $\overline{\text{TO}}$ bit (STATUS<4>) is set, the $\overline{\text{PD}}$ bit (STATUS<3>) is cleared and the oscillator driver is turned off. The I/O ports maintain the status they had before the `SLEEP` instruction was executed (driving high, driving low or hi-impedance).

It should be noted that a RESET generated by a WDT time-out does not drive the $\overline{\text{MCLR}}$ pin low.

For lowest current consumption while powered down, the T0CKI input should be at V_{DD} or V_{SS} and the RB3/$\overline{\text{MCLR}}$/V_{PP} pin must be at a logic high level (V_{IHMC}) if $\overline{\text{MCLR}}$ is enabled.

7.9.2 WAKE-UP FROM SLEEP

The device can wake-up from SLEEP through one of the following events:

1. An external reset input on RB3/$\overline{\text{MCLR}}$/V_{PP} pin, when configured as $\overline{\text{MCLR}}$.
2. A Watchdog Timer time-out reset (if WDT was enabled).
3. A change on input pin RB0, RB1, RB3 or RB4 when wake-up on change is enabled.

These events cause a device reset. The $\overline{\text{TO}}$, $\overline{\text{PD}}$, and RBWUF bits can be used to determine the cause of device reset. The $\overline{\text{TO}}$ bit is cleared if a WDT time-out occurred (and caused wake-up). The $\overline{\text{PD}}$ bit, which is set on power-up, is cleared when `SLEEP` is invoked. The RBWUF bit indicates a change in state while in SLEEP at pins RB0, RB1, RB3 or RB4 (since the last file or bit operation on RB port).

Caution:	Right before entering SLEEP, read the input pins. When in SLEEP, wake up occurs when the values at the pins change from the state they were in at the last reading. If a wake-up on change occurs and the pins are not read before reentering SLEEP, a wake-up will occur immediately even if no pins change while in SLEEP mode.

The WDT is cleared when the device wakes from sleep, regardless of the wake-up source.

7.10 Program Verification/Code Protection

If the code protection bit has not been programmed, the on-chip program memory can be read out for verification purposes.

The first 64 locations and the last location (OSCCAL) can be read, regardless of the code protection bit setting.

7.11 ID Locations

Four memory locations are designated as ID locations where the user can store checksum or other code-identification numbers. These locations are not accessible during normal execution, but are readable and writable during program/verify.

Use only the lower 4 bits of the ID locations and always program the upper 8 bits as '0's.

8.0 INSTRUCTION SET SUMMARY

Each PIC16C505 instruction is a 12-bit word divided into an OPCODE, which specifies the instruction type, and one or more operands which further specify the operation of the instruction. The PIC16C505 instruction set summary in Table 8-2 groups the instructions into byte-oriented, bit-oriented, and literal and control operations. Table 8-1 shows the opcode field descriptions.

For **byte-oriented** instructions, 'f' represents a file register designator and 'd' represents a destination designator. The file register designator is used to specify which one of the 32 file registers is to be used by the instruction.

The destination designator specifies where the result of the operation is to be placed. If 'd' is '0', the result is placed in the W register. If 'd' is '1', 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 8 or 9-bit constant or literal value.

TABLE 8-1: OPCODE FIELD DESCRIPTIONS

Field	Description
`f`	Register file address (0x00 to 0x7F)
`W`	Working register (accumulator)
`b`	Bit address within an 8-bit file register
`k`	Literal field, constant data or label
`x`	Don't care location (= 0 or 1) The assembler will generate code with x = 0. It is the recommended form of use for compatibility with all Microchip software tools.
`d`	Destination select; d = 0 (store result in W) d = 1 (store result in file register 'f') Default is d = 1
`label`	Label name
`TOS`	Top of Stack
`PC`	Program Counter
`WDT`	Watchdog Timer Counter
$\overline{\texttt{TO}}$	Time-Out bit
$\overline{\texttt{PD}}$	Power-Down bit
`dest`	Destination, either the W register or the specified register file location
[ ]	Options
( )	Contents
→	Assigned to
< >	Register bit field
∈	In the set of
italics	User defined term (font is courier)

All instructions are executed within a 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 μs. If a conditional test is true or the program counter is changed as a result of an instruction, the instruction execution time is 2 μs.

Figure 8-1 shows the three general formats that the instructions can have. All examples in the figure use the following format to represent a hexadecimal number:

0xhhh

where 'h' signifies a hexadecimal digit.

FIGURE 8-1: GENERAL FORMAT FOR INSTRUCTIONS

Byte-oriented file register operations

11 – 6	5	4 – 0
OPCODE	d	f (FILE #)

d = 0 for destination W
d = 1 for destination f
f = 5-bit file register address

Bit-oriented file register operations

11 – 8	7 – 5	4 – 0
OPCODE	b (BIT #)	f (FILE #)

b = 3-bit bit address
f = 5-bit file register address

Literal and control operations (except `GOTO`)

11 – 8	7 – 0
OPCODE	k (literal)

k = 8-bit immediate value

Literal and control operations - `GOTO` instruction

11 – 9	8 – 0
OPCODE	k (literal)

k = 9-bit immediate value

MOVWF **Move W to f**

Syntax:	[*label*] MOVWF f
Operands:	$0 \leq f \leq 31$
Operation:	(W) → (f)
Status Affected:	None
Encoding:	`0000` `001f` `ffff`
Description:	Move data from the W register to register 'f'.
Words:	1
Cycles:	1
Example:	`MOVWF TEMP_REG`

Before Instruction
TEMP_REG = 0xFF
W = 0x4F

After Instruction
TEMP_REG = 0x4F
W = 0x4F

NOP **No Operation**

Syntax:	[*label*] NOP
Operands:	None
Operation:	No operation
Status Affected:	None
Encoding:	`0000` `0000` `0000`
Description:	No operation.
Words:	1
Cycles:	1
Example:	`NOP`

OPTION **Load OPTION Register**

Syntax:	[*label*] OPTION
Operands:	None
Operation:	(W) → OPTION
Status Affected:	None
Encoding:	`0000` `0000` `0010`
Description:	The content of the W register is loaded into the OPTION register.
Words:	1
Cycles:	1
Example	`OPTION`

Before Instruction
W = 0x07

After Instruction
OPTION = 0x07

RETLW **Return with Literal in W**

Syntax:	[*label*] RETLW k
Operands:	$0 \leq k \leq 255$
Operation:	k → (W); TOS → PC
Status Affected:	None
Encoding:	`1000` `kkkk` `kkkk`
Description:	The W register is loaded with the eight bit literal 'k'. The program counter is loaded from the top of the stack (the return address). This is a two cycle instruction.
Words:	1
Cycles:	2
Example:	

```
            CALL TABLE ;W contains
                       ;table offset
                       ;value.
            •          ;W now has table
            •          ;value.
            •
TABLE       ADDWF PC   ;W = offset
            RETLW k1   ;Begin table
            RETLW k2   ;
            •
            •
            •
            RETLW kn   ; End of table
```

Before Instruction
W = 0x07

After Instruction
W = value of k8

SLEEP	**Enter SLEEP Mode**
Syntax:	[*label*] SLEEP
Operands:	None
Operation:	00h $\rightarrow$ WDT; 0 $\rightarrow$ WDT prescaler; 1 $\rightarrow$ $\overline{TO}$; 0 $\rightarrow$ $\overline{PD}$
Status Affected:	$\overline{TO}$, $\overline{PD}$, RBWUF
Encoding:	`0000` `0000` `0011`
Description:	Time-out status bit ($\overline{TO}$) is set. The power down status bit ($\overline{PD}$) is cleared. RBWUF is unaffected. The WDT and its prescaler are cleared. The processor is put into SLEEP mode with the oscillator stopped. See section on SLEEP for more details.
Words:	1
Cycles:	1
Example:	`SLEEP`

SUBWF	**Subtract W from f**
Syntax:	[*label*] SUBWF f,d
Operands:	$0 \leq f \leq 31$ $d \in [0,1]$
Operation:	(f) – (W) $\rightarrow$ (dest)
Status Affected:	C, DC, Z
Encoding:	`0000` `10df` `ffff`
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'.
Words:	1
Cycles:	1

Example 1: `SUBWF REG1, 1`

Before Instruction
REG1 = 3
W = 2
C = ?

After Instruction
REG1 = 1
W = 2
C = 1 ; result is positive

Example 2:

Before Instruction
REG1 = 2
W = 2
C = ?

After Instruction
REG1 = 0
W = 2
C = 1 ; result is zero

Example 3:

Before Instruction
REG1 = 1
W = 2
C = ?

After Instruction
REG1 = FF
W = 2
C = 0 ; result is negative

9.7 MPLAB SIM Software Simulator

The MPLAB SIM Software Simulator allows code development in a PC-hosted environment by simulating the PIC MCUs and dsPIC® DSCs on an instruction level. On any given instruction, the data areas can be examined or modified and stimuli can be applied from a comprehensive stimulus controller. Registers can be logged to files for further run-time analysis. The trace buffer and logic analyzer display extend the power of the simulator to record and track program execution, actions on I/O, most peripherals and internal registers.

The MPLAB SIM Software Simulator fully supports symbolic debugging using the MPLAB C Compilers, and the MPASM and MPLAB Assemblers. The software simulator offers the flexibility to develop and debug code outside of the hardware laboratory environment, making it an excellent, economical software development tool.

9.8 MPLAB REAL ICE In-Circuit Emulator System

MPLAB REAL ICE In-Circuit Emulator System is Microchip's next generation high-speed emulator for Microchip Flash DSC and MCU devices. It debugs and programs PIC® Flash MCUs and dsPIC® Flash DSCs with the easy-to-use, powerful graphical user interface of the MPLAB Integrated Development Environment (IDE), included with each kit.

The emulator is connected to the design engineer's PC using a high-speed USB 2.0 interface and is connected to the target with either a connector compatible with in-circuit debugger systems (RJ11) or with the new high-speed, noise tolerant, Low-Voltage Differential Signal (LVDS) interconnection (CAT5).

The emulator is field upgradable through future firmware downloads in MPLAB IDE. In upcoming releases of MPLAB IDE, new devices will be supported, and new features will be added. MPLAB REAL ICE offers significant advantages over competitive emulators including low-cost, full-speed emulation, run-time variable watches, trace analysis, complex breakpoints, a ruggedized probe interface and long (up to three meters) interconnection cables.

9.9 MPLAB ICD 3 In-Circuit Debugger System

MPLAB ICD 3 In-Circuit Debugger System is Microchip's most cost effective high-speed hardware debugger/programmer for Microchip Flash Digital Signal Controller (DSC) and microcontroller (MCU) devices. It debugs and programs PIC® Flash microcontrollers and dsPIC® DSCs with the powerful, yet easy-to-use graphical user interface of MPLAB Integrated Development Environment (IDE).

The MPLAB ICD 3 In-Circuit Debugger probe is connected to the design engineer's PC using a high-speed USB 2.0 interface and is connected to the target with a connector compatible with the MPLAB ICD 2 or MPLAB REAL ICE systems (RJ-11). MPLAB ICD 3 supports all MPLAB ICD 2 headers.

9.10 PICkit 3 In-Circuit Debugger/ Programmer and PICkit 3 Debug Express

The MPLAB PICkit 3 allows debugging and programming of PIC® and dsPIC® Flash microcontrollers at a most affordable price point using the powerful graphical user interface of the MPLAB Integrated Development Environment (IDE). The MPLAB PICkit 3 is connected to the design engineer's PC using a full speed USB interface and can be connected to the target via an Microchip debug (RJ-11) connector (compatible with MPLAB ICD 3 and MPLAB REAL ICE). The connector uses two device I/O pins and the reset line to implement in-circuit debugging and In-Circuit Serial Programming™.

The PICkit 3 Debug Express include the PICkit 3, demo board and microcontroller, hookup cables and CDROM with user's guide, lessons, tutorial, compiler and MPLAB IDE software.

DC CHARACTERISTICS		**Standard Operating Conditions (unless otherwise specified)** Operating temperature 0°C ≤ TA ≤ +70°C (commercial) –40°C ≤ TA ≤ +85°C (industrial) –40°C ≤ TA ≤ +125°C (extended) Operating voltage VDD range as described in DC spec Section 10.1 and Section 10.3.					
Param No.	**Characteristic**	**Sym**	**Min**	**Typ†**	**Max**	**Units**	**Conditions**
	Output High Voltage						
D090	I/O ports/CLKOUT (Note 3)	V_{OH}	V_{DD} - 0.7	—	—	V	I_{OH} = -3.0 mA, V_{DD} = 4.5V, –40°C to +85°C
D090A			V_{DD} - 0.7	—	—	V	I_{OH} = -2.5 mA, V_{DD} = 4.5V, –40°C to +125°C
D092	OSC2		V_{DD} - 0.7	—	—	V	I_{OH} = -1.3 mA, V_{DD} = 4.5V, –40°C to +85°C
D092A			V_{DD} - 0.7	—	—	V	I_{OH} = -1.0 mA, V_{DD} = 4.5V, –40°C to +125°C
	Capacitive Loading Specs on Output Pins						
D100	OSC2 pin	C_{OSC2}	—	—	15	pF	In XT, HS and LP modes when external clock is used to drive OSC1.
D101	All I/O pins and OSC2	C_{IO}	—	—	50	pF	

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

Note 1: In EXTRC oscillator configuration, the OSC1/CLKIN pin is a Schmitt Trigger input. It is not recommended that the PIC16C505 be driven with external clock in RC mode.

2: The leakage current on the $\overline{MCLR}$ 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.

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

4: Does not include GP3. For GP3 see parameters D061 and D061A.

5: This spec. applies to GP3/$\overline{MCLR}$ configured as external $\overline{MCLR}$ and GP3/$\overline{MCLR}$ configured as input with internal pull-up enabled.

6: This spec. applies when GP3/$\overline{MCLR}$ is configured as an input with pull-up disabled. The leakage current of the $\overline{MCLR}$ circuit is higher than the standard I/O logic.

TABLE 10-1: PULL-UP RESISTOR RANGES - PIC16C505

VDD (Volts)	Temperature (°C)	Min	Typ	Max	Units
RB0/RB1/RB4					
2.5	–40	38K	42K	63K	W
	25	42K	48K	63K	W
	85	42K	49K	63K	W
	125	50K	55K	63K	W
5.5	–40	15K	17K	20K	W
	25	18K	20K	23K	W
	85	19K	22K	25K	W
	125	22K	24K	28K	W
RB3					
2.5	–40	285K	346K	417K	W
	25	343K	414K	532K	W
	85	368K	457K	532K	W
	125	431K	504K	593K	W
5.5	–40	247K	292K	360K	W
	25	288K	341K	437K	W
	85	306K	371K	448K	W
	125	351K	407K	500K	W

* These parameters are characterized but not tested.

© 1999-2012 Microchip Technology Inc.

11.0 DC AND AC CHARACTERISTICS - PIC16C505

The graphs and tables provided in this section are for design guidance and are not tested. 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 will 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 11-1: CALIBRATED INTERNAL RC FREQUENCY RANGE VS. TEMPERATURE (VDD = 5.0V) (INTERNAL RC IS CALIBRATED TO 25°C, 5.0V)

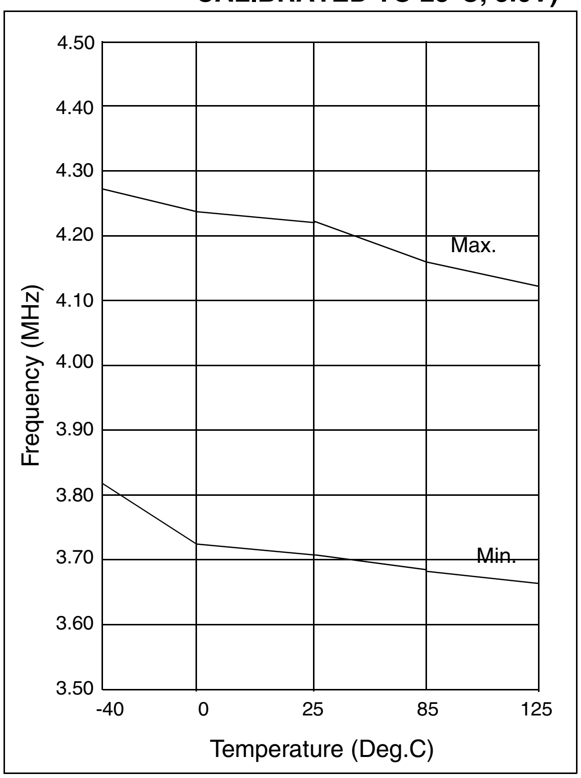


FIGURE 11-2: CALIBRATED INTERNAL RC FREQUENCY RANGE VS. TEMPERATURE (VDD = 2.5V) (INTERNAL RC IS CALIBRATED TO 25°C, 5.0V)

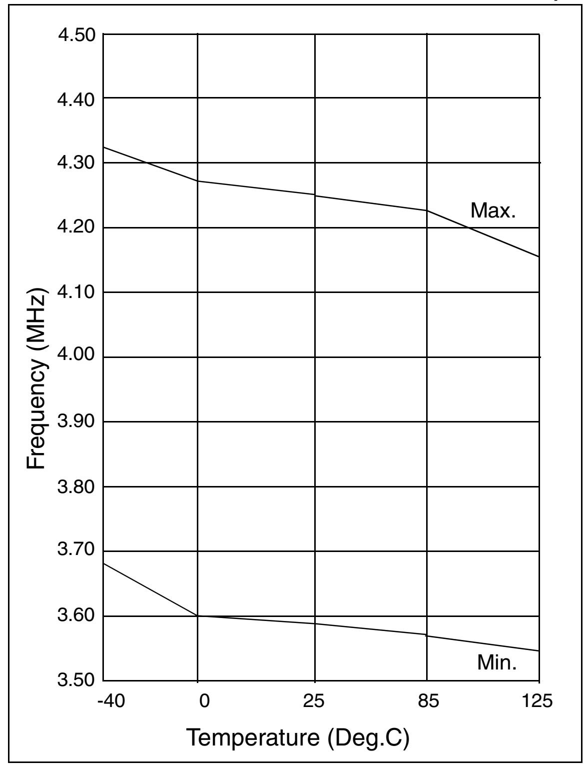


TABLE 11-1: DYNAMIC IDD (TYPICAL) - WDT ENABLED, 25°C

Oscillator	Frequency	VDD = 3.0V(1)	VDD = 5.5V
External RC	4 MHz	240 µA(2)	800 µA(2)
Internal RC	4 MHz	320 µA	800 µA
XT	4 MHz	300 µA	800 µA
LP	32 kHz	19 µA	50 µA
HS	20 MHz	N/A	4.5 mA

Note 1: LP oscillator based on VDD = 2.5V
2: Does not include current through external R&C.

FIGURE 11-3: WDT TIMER TIME-OUT PERIOD vs. VDD

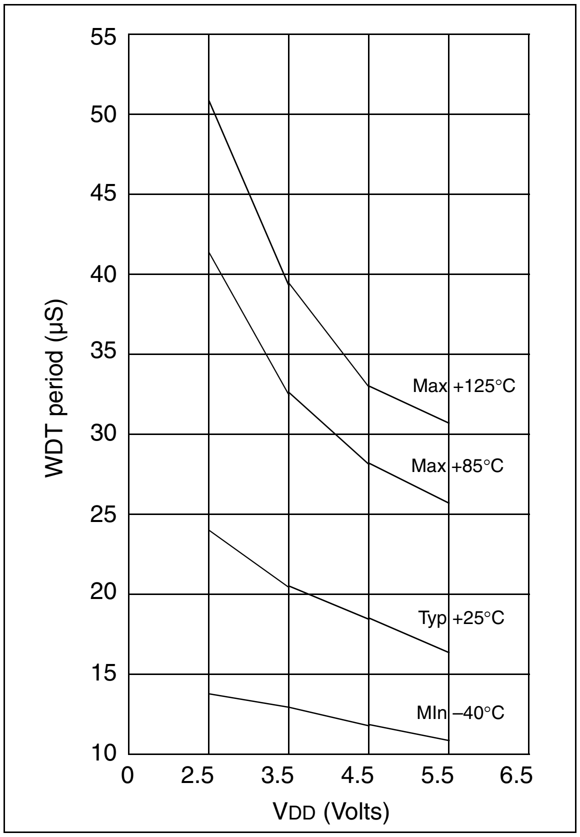
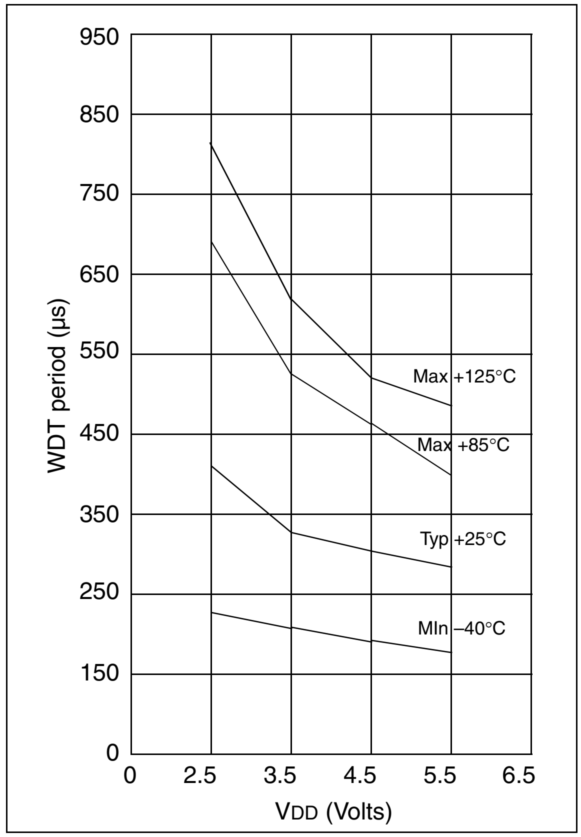


FIGURE 11-4: SHORT DRT PERIOD VS. VDD



 © 1999-2012 Microchip Technology Inc.

FIGURE 11-5: IOH vs. VOH, VDD = 2.5 V

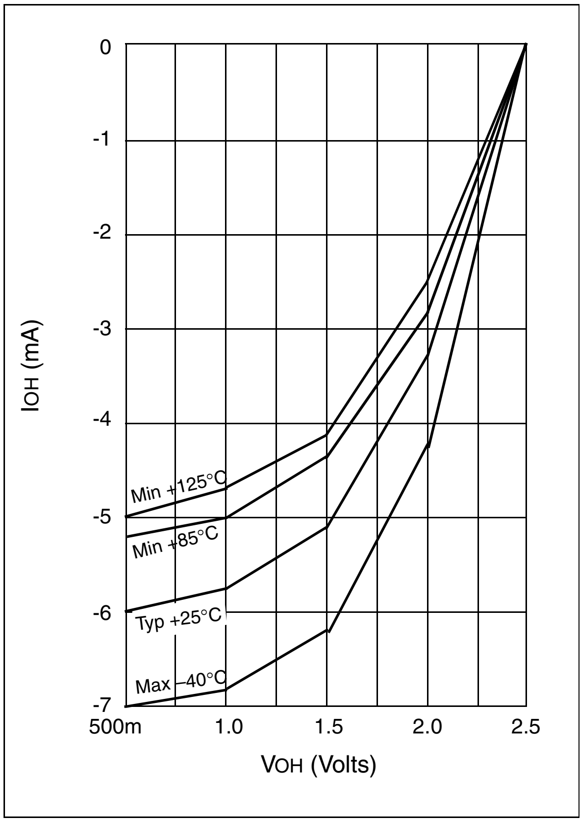


FIGURE 11-6: IOH vs. VOH, VDD = 5.5 V

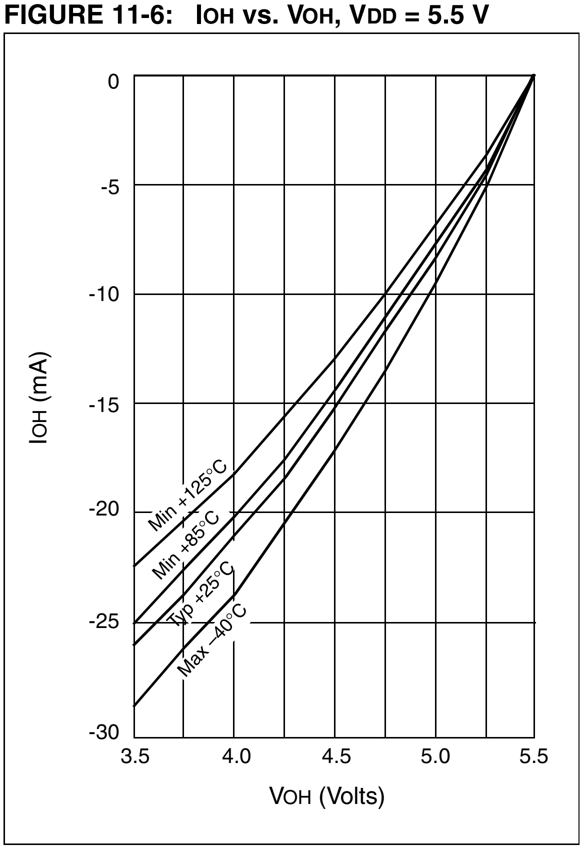


FIGURE 11-7: IOL vs. VOL, VDD = 2.5 V

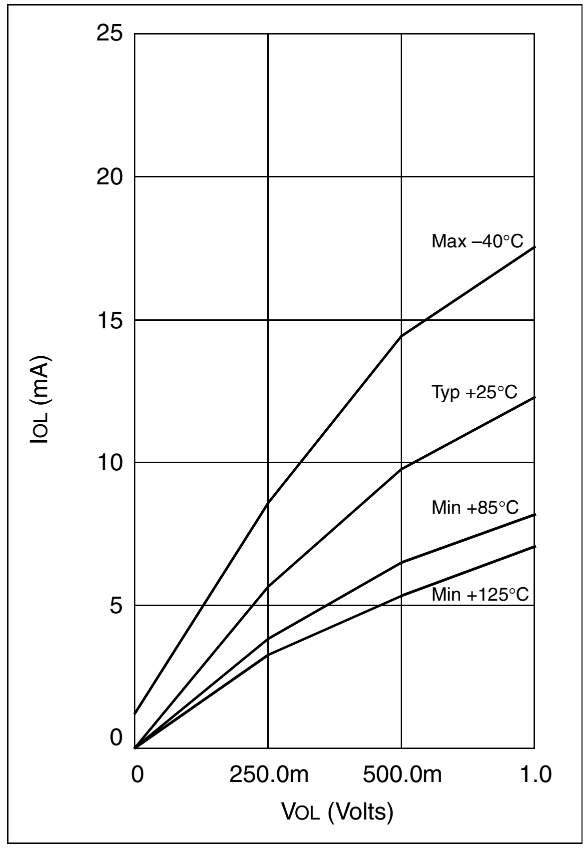
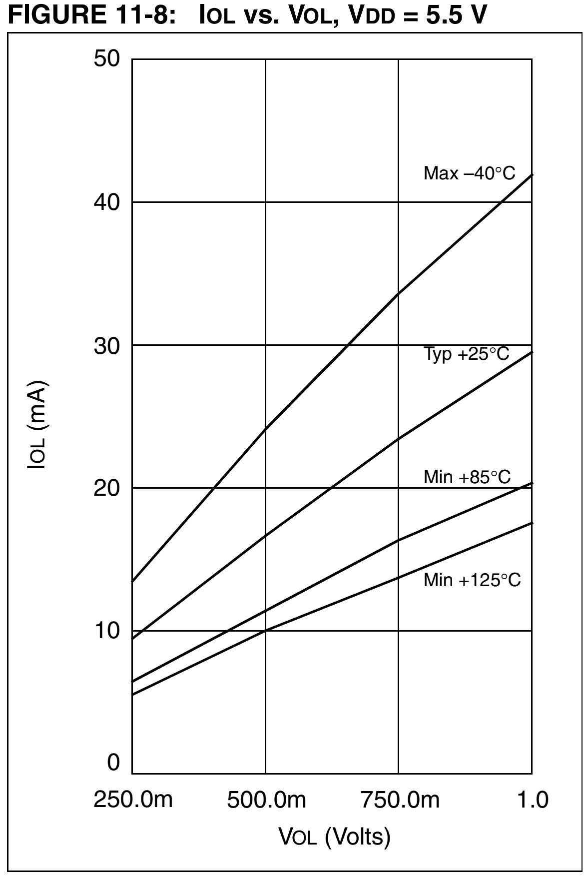


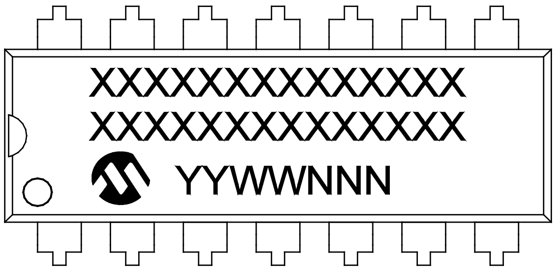
FIGURE 11-8: IOL vs. VOL, VDD = 5.5 V



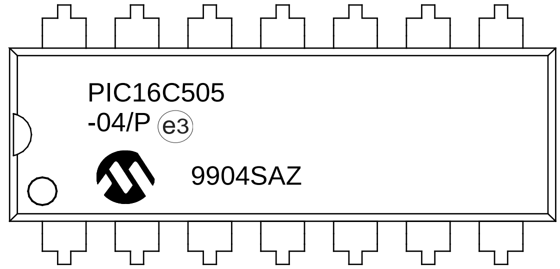
12.0 PACKAGING INFORMATION

12.1 Package Marking Information

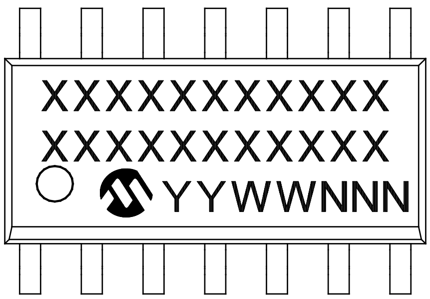
14-Lead PDIP (300 mil)



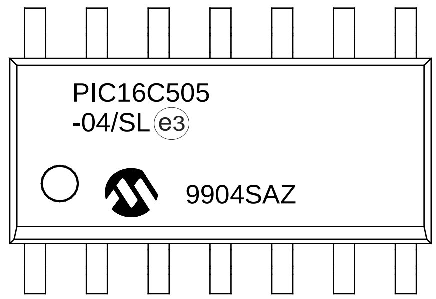
Example



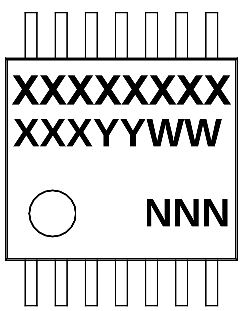
14-Lead SOIC (3.90 mm)



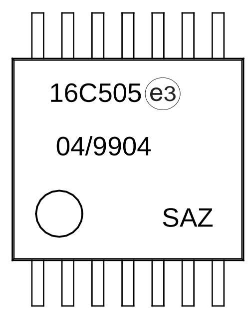
Example



14-Lead TSSOP (4.4 mm)



Example



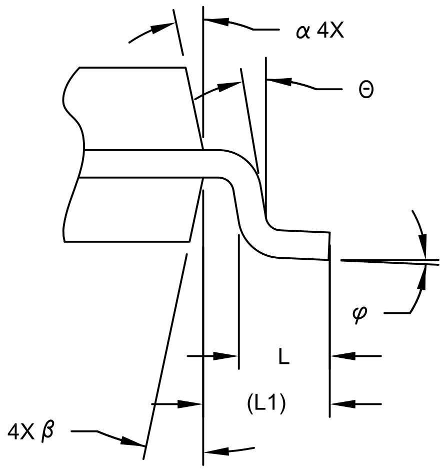
Legend:	XX...X	Customer-specific information
	Y	Year code (last digit of calendar year)
	YY	Year code (last 2 digits of calendar year)
	WW	Week code (week of January 1 is week '01')
	NNN	Alphanumeric traceability code
	e3	Pb-free JEDEC designator for Matte Tin (Sn)
	*	This package is Pb-free. The Pb-free JEDEC designator (e3) can be found on the outer packaging for this package.

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.

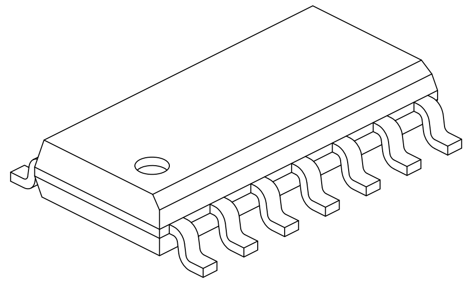
© 1999-2012 Microchip Technology Inc.

14-Lead Plastic Small Outline (SL) - Narrow, 3.90 mm Body [SOIC]

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



VIEW C



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N	14		
Pitch	e	1.27 BSC		
Overall Height	A	-	-	1.75
Molded Package Thickness	A2	1.25	-	-
Standoff §	A1	0.10	-	0.25
Overall Width	E	6.00 BSC		
Molded Package Width	E1	3.90 BSC		
Overall Length	D	8.65 BSC		
Chamfer (Optional)	h	0.25	-	0.50
Foot Length	L	0.40	-	1.27
Footprint	L1	1.04 REF		
Lead Angle	Θ	0°	-	-
Foot Angle	φ	0°	-	8°
Lead Thickness	c	0.10	-	0.25
Lead Width	b	0.31	-	0.51
Mold Draft Angle Top	α	5°	-	15°
Mold Draft Angle Bottom	β	5°	-	15°

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. § Significant Characteristic
3. Dimension D does not include mold flash, protrusions or gate burrs, which shall not exceed 0.15 mm per end. Dimension E1 does not include interlead flash or protrusion, which shall not exceed 0.25 mm per side.
4. Dimensioning and tolerancing per ASME Y14.5M

   BSC: Basic Dimension. Theoretically exact value shown without tolerances.

   REF: Reference Dimension, usually without tolerance, for information purposes only.
5. Datums A & B to be determined at Datum H.

Microchip Technology Drawing No. C04-065C Sheet 2 of 2

© 1999-2012 Microchip Technology Inc.

Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as "unbreakable."

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

Information contained in this publication regarding device applications and the like is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. MICROCHIP MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION, INCLUDING BUT NOT LIMITED TO ITS CONDITION, QUALITY, PERFORMANCE, MERCHANTABILITY OR FITNESS FOR PURPOSE**.** Microchip disclaims all liability arising from this information and its use. Use of Microchip devices in life support and/or safety applications is entirely at the buyer's risk, and the buyer agrees to defend, indemnify and hold harmless Microchip from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Microchip intellectual property rights.

QUALITY MANAGEMENT SYSTEM CERTIFIED BY DNV

ISO/TS 16949

Trademarks

The Microchip name and logo, the Microchip logo, dsPIC, KEELOQ, KEELOQ logo, MPLAB, PIC, PICmicro, PICSTART, PIC32 logo, rfPIC and UNI/O are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

FilterLab, Hampshire, HI-TECH C, Linear Active Thermistor, MXDEV, MXLAB, SEEVAL and The Embedded Control Solutions Company are registered trademarks of Microchip Technology Incorporated in the U.S.A.

Analog-for-the-Digital Age, Application Maestro, chipKIT, chipKIT logo, CodeGuard, dsPICDEM, dsPICDEM.net, dsPICworks, dsSPEAK, ECAN, ECONOMONITOR, FanSense, HI-TIDE, In-Circuit Serial Programming, ICSP, Mindi, MiWi, MPASM, MPLAB Certified logo, MPLIB, MPLINK, mTouch, Omniscient Code Generation, PICC, PICC-18, PICDEM, PICDEM.net, PICkit, PICtail, REAL ICE, rfLAB, Select Mode, Total Endurance, TSHARC, UniWinDriver, WiperLock and ZENA are trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

SQTP is a service mark of Microchip Technology Incorporated in the U.S.A.

All other trademarks mentioned herein are property of their respective companies.

© 1999-2012, Microchip Technology Incorporated, Printed in the U.S.A., All Rights Reserved.

Printed on recycled paper.

ISBN: 978-1-62076-395-7

Microchip received ISO/TS-16949:2009 certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona; Gresham, Oregon and design centers in California and India. The Company's quality system processes and procedures are for its PIC® MCUs and dsPIC® DSCs, KEELOQ® code hopping devices, Serial EEPROMs, microperipherals, nonvolatile memory and analog products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001:2000 certified.