Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

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

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	PIC
Core Size	8-Bit
Speed	20MHz
Connectivity	-
Peripherals	POR, WDT
Number of I/O	11
Program Memory Size	1.5KB (1K x 12)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	67 x 8
Voltage - Supply (Vcc/Vdd)	2V ~ 5.5V
Data Converters	A/D 4x8b
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/pic16f506-e-sl

NOTES:

3.0 ARCHITECTURAL OVERVIEW

The high performance of the PIC12F510/16F506 devices can be attributed to a number of architectural features commonly found in RISC microprocessors. The PIC12F510/16F506 devices use a Harvard architecture in which program and data are accessed on separate buses. This improves bandwidth over traditional von Neumann architectures where program and data are fetched on the same bus. Separating program and data memory further allows instructions to be sized differently than the 8-bit wide data word. Instruction opcodes are 12 bits wide, making it possible to have all single-word instructions. A 12-bit wide program memory access bus fetches a 12-bit instruction in a single cycle. A two-stage pipeline overlaps fetch and execution of instructions. Consequently, all instructions (33) execute in a single cycle (200 ns @ 20 MHz, 1 µs @ 4 MHz) except for program branches.

Table 3-1 lists program memory (Flash) and data memory (RAM) for the PIC12F510/16F506 devices.

TABLE 3-1: PIC12F510/16F506 MEMORY

Device	Memory	
	Program	Data
PIC12F510	1024 x 12	38 x 8
PIC16F506	1024 x 12	67 x 8

The PIC12F510/16F506 devices can directly or indirectly address its register files and data memory. All Special Function Registers (SFRs), including the PC, are mapped in the data memory. The PIC12F510/16F506 devices have a highly orthogonal (symmetrical) instruction set that makes it possible to carry out any operation, on any register, using any addressing mode. This symmetrical nature and lack of “special optimal situations” make programming with the PIC12F510/16F506 devices simple, yet efficient. In addition, the learning curve is reduced significantly.

The PIC12F510/16F506 devices contain an 8-bit ALU and working register. The ALU is a general purpose arithmetic unit. It performs arithmetic and Boolean functions between data in the working register and any register file.

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

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

Depending on the instruction executed, the ALU may affect the values of the Carry (C), Digit Carry (DC) and Zero (Z) bits in the STATUS register. The C and DC bits operate as a $\overline{\text{borrow}}$ and $\overline{\text{digit borrow}}$ out bit, respectively, in subtraction. See the `SUBWF` and `ADDWF` instructions for examples.

A simplified block diagram is shown in Figure 3-1 for PIC12F510 with the corresponding device pins described in Table 3-2. A simplified block diagram for PIC16F506 is shown in Figure 3-2 with the corresponding device pins described in Table 3-3.

3.1 Clocking Scheme/Instruction Cycle

The clock input (OSC1/CLKIN pin) is internally divided by four to generate four non-overlapping quadrature clocks, namely Q1, Q2, Q3 and Q4. Internally, the PC is incremented every Q1 and the instruction is fetched from program memory and latched into the 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 3-3 and Example 3-1.

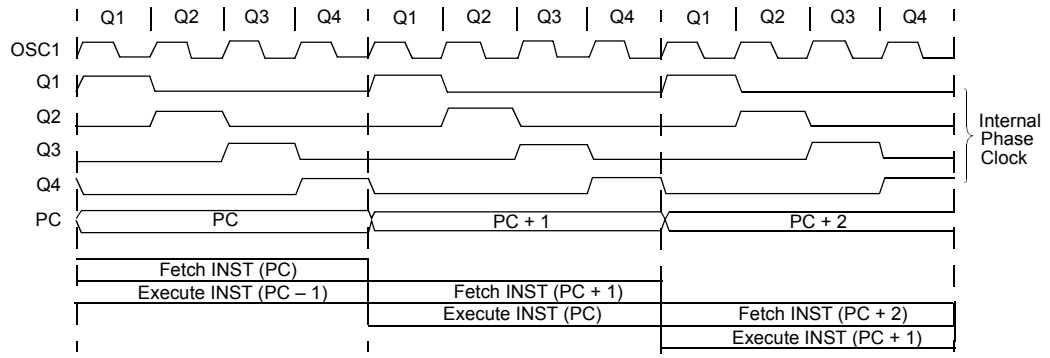
3.2 Instruction Flow/Pipelining

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

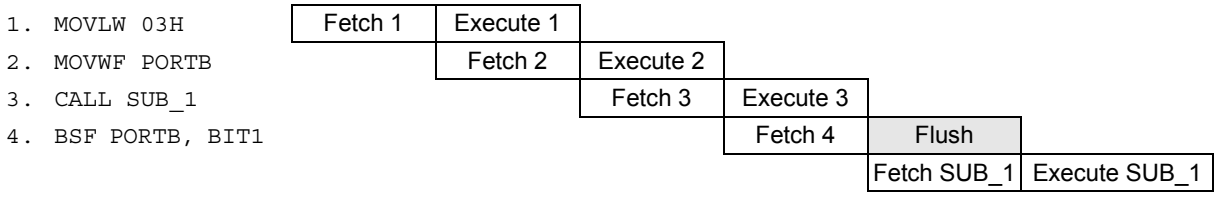
A fetch cycle begins with the PC incrementing in Q1.

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

FIGURE 3-3: CLOCK/INSTRUCTION CYCLE



EXAMPLE 3-1: INSTRUCTION PIPELINE FLOW



All instructions are single cycle, except for any program branches. These take two cycles, since the fetch instruction is “flushed” from the pipeline, while the new instruction is being fetched and then executed.

4.4 OPTION Register

The OPTION register is a 8-bit wide, write-only register, that contains various control bits to configure the Timer0/WDT prescaler and Timer0.

By executing the OPTION instruction, the contents of the W register will be transferred to the OPTION register. A Reset sets the OPTION<7:0> bits.

Note 1:	If TRIS bit is set to '0', the wake-up on change and pull-up functions are disabled for that pin (i.e., note that TRIS overrides Option control of $\overline{\text{GPPU}}$/RBPU and $\overline{\text{GPWU}}$/$\overline{\text{RBWU}}$).
2:	If the T0CS bit is set to '1', it will override the TRIS function on the T0CKI pin.

REGISTER 4-3: OPTION_REG: OPTION REGISTER (PIC12F510)

W-1	W-1	W-1	W-1	W-1	W-1	W-1	W-1
$\overline{\text{GPWU}}$	$\overline{\text{GPPU}}$	T0CS	T0SE	PSA	PS2	PS1	PS0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7 **$\overline{\text{GPWU}}$:** Enable Wake-up On Pin Change bit (GP0, GP1, GP3)
1 = Disabled
0 = Enabled

bit 6 **$\overline{\text{GPPU}}$:** Enable Weak Pull-Ups bit (GP0, GP1, GP3)
1 = Disabled
0 = Enabled

bit 5 **T0CS:** Timer0 Clock Source Select bit
1 = Transition on T0CKI pin
0 = Internal instruction cycle clock (CLKOUT)

bit 4 **T0SE:** Timer0 Source Edge Select bit
1 = Increment on high-to-low transition on T0CKI pin
0 = Increment on low-to-high transition on T0CKI pin

bit 3 **PSA:** Prescaler Assignment bit
1 = Prescaler assigned to the WDT
0 = Prescaler assigned to Timer0

bit 2-0 **PS<2:0>:** Prescaler Rate Select bits

Bit Value	Timer0 Rate	WDT Rate
000	1 : 2	1 : 1
001	1 : 4	1 : 2
010	1 : 8	1 : 4
011	1 : 16	1 : 8
100	1 : 32	1 : 16
101	1 : 64	1 : 32
110	1 : 128	1 : 64
111	1 : 256	1 : 128

TABLE 5-1: SUMMARY OF PORT REGISTERS

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	TRISGPIO(1)	—	—	I/O Control Register						--11 1111	--11 1111
N/A	TRISB(2)	—	—	I/O Control Register						--11 1111	--11 1111
N/A	TRISC(2)	—	—	I/O Control Register						--11 1111	--11 1111
N/A	OPTION(1)	$\overline{GPWU}$	$\overline{GPPU}$	T0CS	TOSE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
N/A	OPTION(2)	$\overline{RBWU}$	$\overline{RBPU}$	T0CS	TOSE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
03h	STATUS(1)	GPWUF	CWUF	PA0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	qq0q quuu(3)
03h	STATUS(2)	RBWUF	CWUF	PA0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	qq0q quuu(3)
06h	GPIO(1)	—	—	GP5	GP4	GP3	GP2	GP1	GP0	--xx xxxx	--uu uuuu
06h	PORTB(2)	—	—	RB5	RB4	RB3	RB2	RB1	RB0	--xx xxxx	--uu uuuu
07h	PORTC(2)	—	—	RC5	RC4	RC3	RC2	RC1	RC0	--xx xxxx	--uu uuuu

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

Note 1: PIC12F510 only.

2: PIC16F506 only.

3: If Reset was due to wake-up on pin change, then bit 7 = 1. All other Resets will cause bit 7 = 0.

TABLE 5-2: I/O PIN FUNCTION ORDER OF PRECEDENCE (PIC16F506)

Priority	RB0	RB1	RB2	RB3	RB4	RB5
1	AN0/C1IN+	AN1/C1IN-	AN2	Input/$\overline{MCLR}$	OSC2/CLKOUT	OSC1/CLKIN
2	TRISB	TRISB	C1OUT	—	TRISB	TRISB
3	—	—	TRISB	—	—	—

TABLE 5-3: I/O PIN FUNCTION ORDER OF PRECEDENCE (PIC16F506)

Priority	RC0	RC1	RC2	RC3	RC4	RC5
1	C2IN+	C2IN-	CV_{REF}	TRISC	C2OUT	T0CKI
2	TRISC	TRISC	TRISC	—	TRISC	TRISC

TABLE 5-4: I/O PIN FUNCTION ORDER OF PRECEDENCE (PIC12F510)

Priority	GP0	GP1	GP2	GP3	GP4	GP5
1	AN0/C1IN+	AN1/C1IN-	AN2	Input/$\overline{MCLR}$	OSC2	OSC1/CLKIN
2	TRISIO	TRISIO	C1OUT	—	TRISIO	TRISIO
3	—	—	T0CKI	—	—	—
4	—	—	TRISIO	—	—	—

6.0 TMR0 MODULE AND TMR0 REGISTER

The Timer0 module has the following features:

- 8-bit timer/counter register, TMR0
- Readable and writable
- 8-bit software programmable prescaler
- Internal or external clock select:
  - Edge select for external clock
  - External clock from either the T0CKI pin or from the output of the comparator

Figure 6-1 is a simplified block diagram of the Timer0 module.

Timer mode is selected by clearing the T0CS bit (OPTION<5>). In Timer mode, the Timer0 module will increment every instruction cycle (without prescaler). If TMR0 register is written, the increment is inhibited for the following two cycles (Figure 6-2 and Figure 6-3). The user can work around this by writing an adjusted value to the TMR0 register.

There are two types of Counter mode. The first Counter mode uses the T0CKI pin to increment Timer0. It is selected by setting the T0CKI bit (OPTION<5>), setting the $\overline{\text{C1T0CS}}$ bit (CM1CON0<4>) and setting the $\overline{\text{C1OUTEN}}$ bit (CM1CON0<6>). In this mode, Timer0 will increment either on every rising or falling edge of pin T0CKI. The T0SE bit (OPTION<4>) determines the source edge. Clearing the T0SE bit selects the rising edge. Restrictions on the external clock input are discussed in detail in **Section 6.1 "Using Timer0 With An External Clock"**.

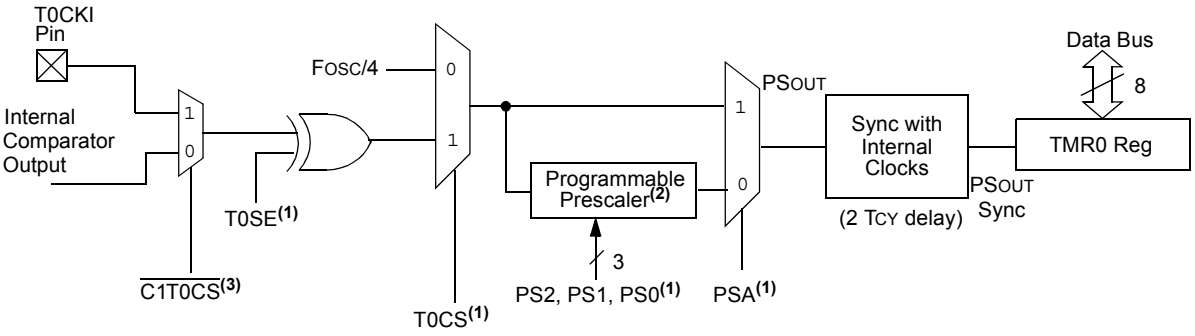
The second Counter mode uses the output of the comparator to increment Timer0. It can be entered in two different ways. The first way is selected by setting the T0CS bit (OPTION<5>), and clearing the $\overline{\text{C1T0CS}}$ bit (CM1CON0<4>) ($\overline{\text{C1OUTEN}}$ [CM1CON0<6>] does not affect this mode of operation). This enables an internal connection between the comparator and the Timer0.

The second way is selected by setting the T0CS bit (OPTION<5>), setting the $\overline{\text{C1T0CS}}$ bit (CM1CON0) and clearing the $\overline{\text{C1OUTEN}}$ bit (CM1CON0<6>). This allows the output of the comparator onto the T0CKI pin, while keeping the T0CKI input active. Therefore, any comparator change on the COUT pin is fed back into the T0CKI input. The T0SE bit (OPTION<4>) determines the source edge. Clearing the T0SE bit selects the rising edge. Restrictions on the external clock input as discussed in **Section 6.1 "Using Timer0 With An External Clock"**.

The prescaler may be used by either the Timer0 module or the Watchdog Timer, but not both. The prescaler assignment is controlled in software by the control bit PSA (OPTION<3>). Clearing the PSA bit will assign the prescaler to Timer0. The prescaler is not readable or writable. When the prescaler is assigned to the Timer0 module, prescale values of 1:2, 1:4,..., 1:256 are selectable. **Section 6.2 "Prescaler"** details the operation of the prescaler.

A summary of registers associated with the Timer0 module is found in Table 6-1.

FIGURE 6-1: TIMER0 BLOCK DIAGRAM



Note 1: Bits T0CS, T0SE, PSA, PS2, PS1 and PS0 are located in the OPTION register.
2: The prescaler is shared with the Watchdog Timer (Figure 6-5).
3: Bit $\overline{\text{C1T0CS}}$ is located in the CM1CON0 register, CM1CON0<4>.

FIGURE 6-2: TIMER0 TIMING: INTERNAL CLOCK/NO PRESCALE

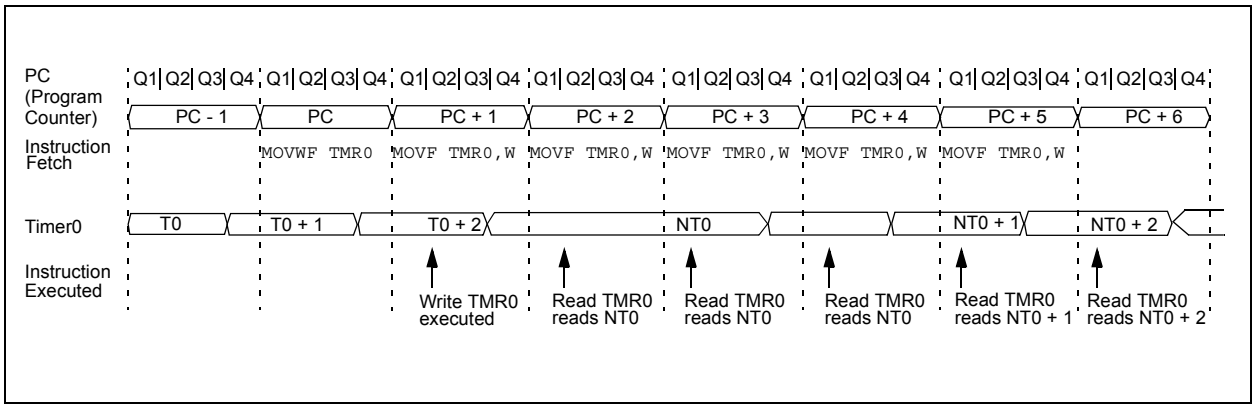


FIGURE 6-3: TIMER0 TIMING: INTERNAL CLOCK/PRESCALE 1:2

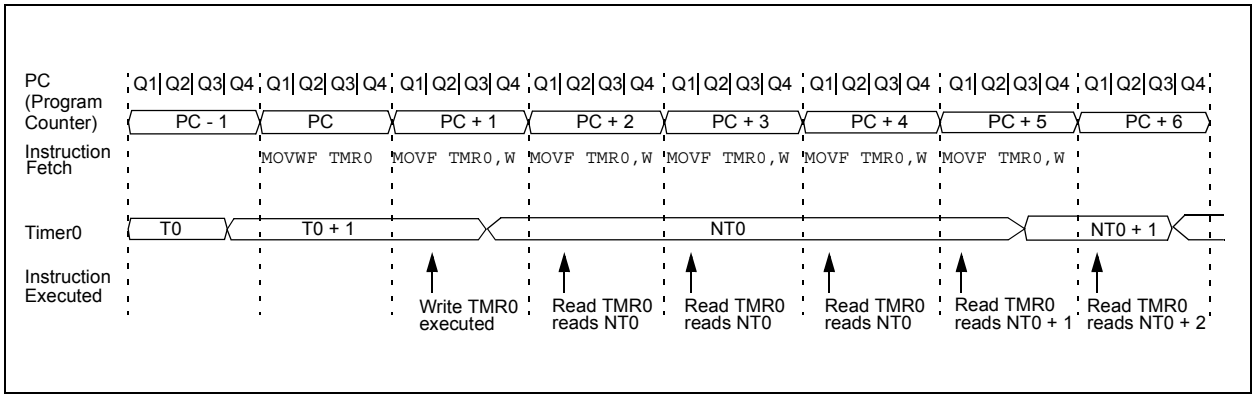


TABLE 6-1: REGISTERS ASSOCIATED WITH TIMER0

Addr	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
01h	TMR0	Timer0 – 8-bit Real-Time Clock/Counter								xxxx xxxx	uuuu uuuu
07h	CM1CON0(2)	C1OUT	$\overline{\text{C1OUTEN}}$	C1POL	$\overline{\text{C1T0CS}}$	C1ON	C1NREF	C1PREF	$\overline{\text{C1WU}}$	1111 1111	uuuu uuuu
08h	CM1CON0(3)	C1OUT	$\overline{\text{C1OUTEN}}$	C1POL	$\overline{\text{C1T0CS}}$	C1ON	C1NREF	C1PREF	$\overline{\text{C1WU}}$	1111 1111	uuuu uuuu
N/A	OPTION	$\overline{\text{GPWU}}$	$\overline{\text{GPPU}}$	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
N/A	TRISGPIO(1)	—	—	I/O Control Register						---- 1111	--11 1111

Legend: Shaded cells not used by Timer0, – = unimplemented, x = unknown, u = unchanged.

Note 1: The TRIS of the T0CKI pin is overridden when T0CS = 1.

2: For PIC12F510.

3: For PIC16F506.

6.2.1 SWITCHING PRESCALER ASSIGNMENT

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 (Example 6-1) must be executed when changing the prescaler assignment from Timer0 to the WDT.

EXAMPLE 6-1: CHANGING PRESCALER (TIMER0 → WDT)

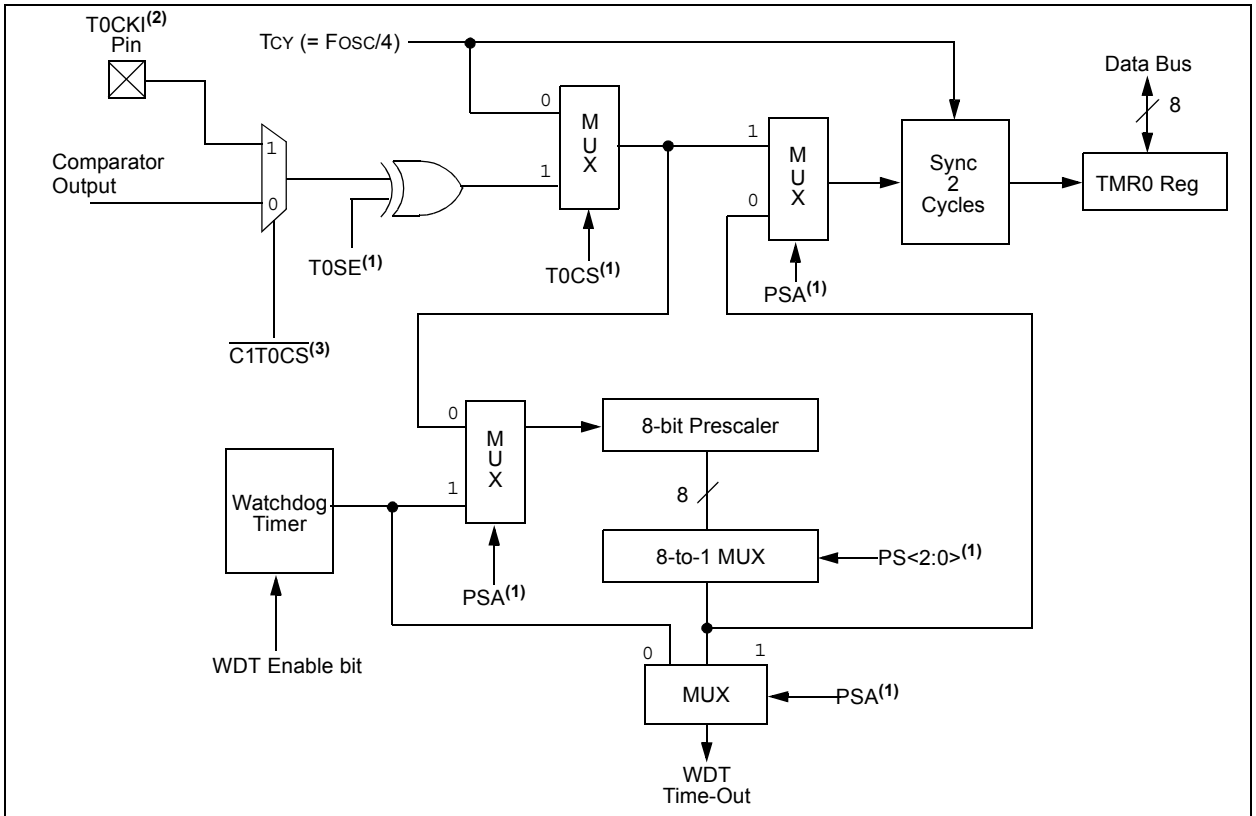
```
CLRWDT              ;Clear WDT
CLRF   TMR0         ;Clear TMR0 & Prescaler
MOVLW  '00xx1111'b  ;These 3 lines (5, 6, 7)
OPTION              ;are required only if
                    ;desired
CLRWDT              ;PS<2:0> are 000 or 001
MOVLW  '00xx1xxx'b  ;Set Postscaler to
OPTION              ;desired WDT rate
```

To change prescaler from the WDT to the Timer0 module, use the sequence shown in Example 6-2. This sequence must be used even if the WDT is disabled. A CLRWDT instruction should be executed before switching the prescaler.

EXAMPLE 6-2: CHANGING PRESCALER (WDT→TIMER0)

```
CLRWDT              ;Clear WDT and
                    ;prescaler
MOVLW  'xxxx0xxx'   ;Select TMR0, new
                    ;prescale value and
                    ;clock source
OPTION
```

FIGURE 6-5: BLOCK DIAGRAM OF THE TIMER0/WDT PRESCALER



Note 1: T0CS, T0SE, PSA, PS<2:0> are bits in the OPTION register.

2: T0CKI is shared with pin GP2 on the PIC12F510 and shared with RC5 on the PIC16F506.

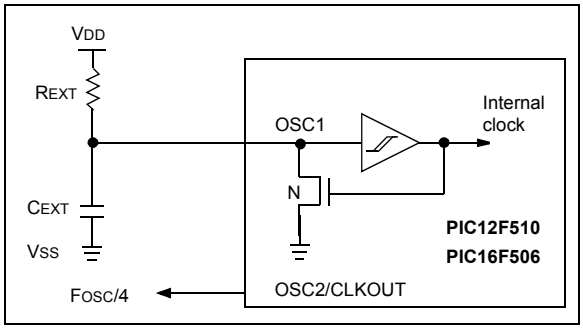
3: Bit $\overline{C1T0CS}$ is located in the CM1CON0 register.

Although the oscillator will operate with no external capacitor (C_{EXT} = 0 pF), we recommend using values above 20 pF for noise and stability reasons. With no capacitance 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.

Section 13.0 "Electrical Characteristics", shows RC frequency variation from part-to-part due to normal process variation. The variation is larger for larger values of R (since leakage current variation will affect RC frequency more for large R) and for smaller values of C (since variation of input capacitance will affect RC frequency more).

Also, see the Electrical Specifications section for variation of oscillator frequency due to V_{DD} for given R_{EXT}/C_{EXT} values, as well as frequency variation due to operating temperature for given R, C and V_{DD} values.

FIGURE 10-5: EXTERNAL RC OSCILLATOR MODE



10.2.5 INTERNAL 4/8 MHz RC OSCILLATOR

The internal RC oscillator provides a fixed 4/8 MHz (nominal) system clock (see **Section 13.0 "Electrical Characteristics"** for information on variation over voltage and temperature).

In addition, a calibration instruction is programmed into the last address of memory, which contains the calibration value for the internal RC oscillator. This location is always uncode protected, regardless of the code-protect settings. This value is programmed as a `MOVLW XX` instruction where `XX` is the calibration value, and is placed at the Reset vector. This will load the W register with the calibration value upon Reset and the PC will then roll over to the users program at address 0x000. The user then has the option of writing the value to the OSCCAL Register (05h) or ignoring it.

OSCCAL, when written to with the calibration value, will "trim" the internal oscillator to remove process variation from the oscillator frequency.

Note:	Erasing the device will also erase the pre-programmed internal calibration value for the internal oscillator. The calibration value must be read prior to erasing the part so it can be reprogrammed correctly later.

For the PIC12F510/16F506 devices, only bits <7:1> of OSCCAL are used for calibration. See Register 4-5 for more information.

Note:	The 0 bit of OSCCAL is unimplemented and should be written as '0' when modifying OSCCAL for compatibility with future devices.

11.0 INSTRUCTION SET SUMMARY

The PIC16 instruction set is highly orthogonal and is comprised of three basic categories.

- **Byte-oriented** operations
- **Bit-oriented** operations
- **Literal and control** operations

Each PIC16 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 formats for each of the categories is presented in Figure 11-1, while the various opcode fields are summarized in Table 11-1.

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

The destination designator specifies where the result of the operation is to be placed. If 'd' is '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 bits 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 11-1: 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
$\overline{TO}$	Time-out bit
$\overline{PD}$	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 11-1 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 11-1: GENERAL FORMAT FOR INSTRUCTIONS

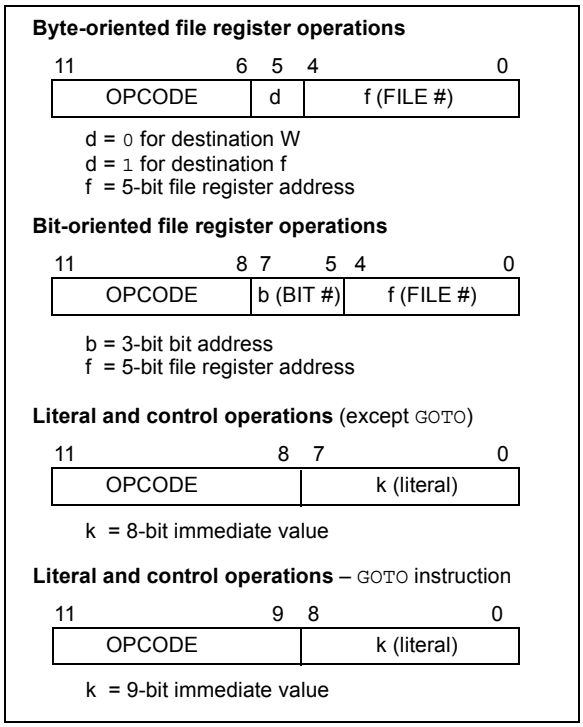


FIGURE 13-3: VOLTAGE FREQUENCY GRAPH, -40°C ≤ TA ≤ +125°C (PIC16F506)

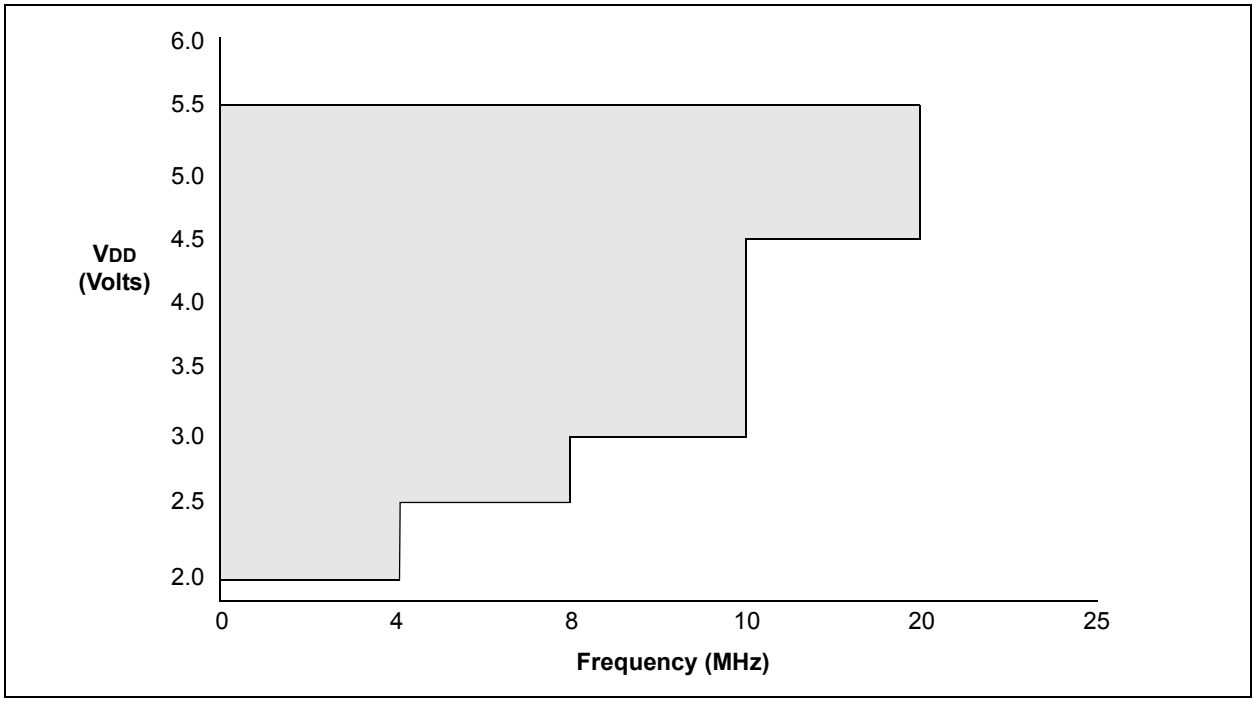


FIGURE 13-4: MAXIMUM OSCILLATOR FREQUENCY TABLE (PIC16F506)

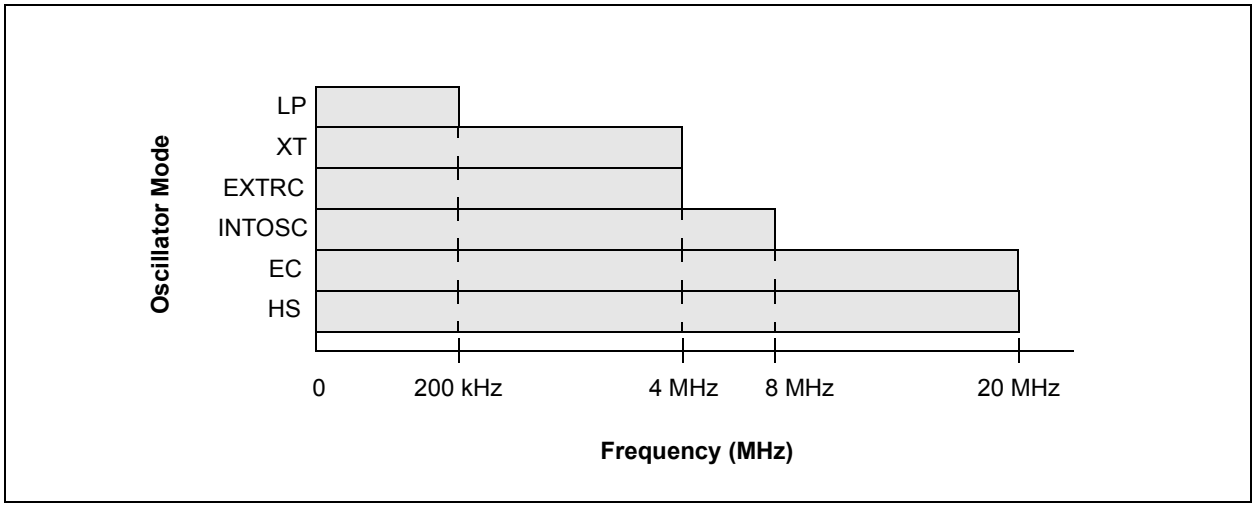


TABLE 13-7: RESET, WATCHDOG TIMER AND DEVICE RESET TIMER

AC CHARACTERISTICS			Standard Operating Conditions (unless otherwise specified) Operating Temperature -40°C ≤ TA ≤ +85°C (industrial) -40°C ≤ TA ≤ +125°C (extended)				
Param No.	**Sym**	**Characteristic**	**Min**	**Typ(1)**	**Max**	**Units**	**Conditions**
30	TMCL	$\overline{MCLR}$ Pulse Width (low)	2000*	—	—	ns	VDD = 5.0V
31	TWDT	Watchdog Timer Time-out Period (No Prescaler)	9* 9*	18* 18*	30* 40*	ms ms	VDD = 5.0V (Industrial) VDD = 5.0V (Extended)
32	TDRT	Device Reset Timer Period					
		Standard	9* 9*	18* 18*	30* 40*	ms ms	VDD = 5.0V (Industrial) VDD = 5.0V (Extended)
		Short	0.5* 0.5*	1.125* 1.125*	2* 2.5*	ms ms	VDD = 5.0V (Industrial) VDD = 5.0V (Extended)
34	TIOZ	I/O high-impedance from $\overline{MCLR}$ low	—	—	2000*	ns	

* These parameters are characterized but not tested.

Note 1: Data in the Typical ("Typ") column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

FIGURE 13-9: TIMER0 CLOCK TIMINGS

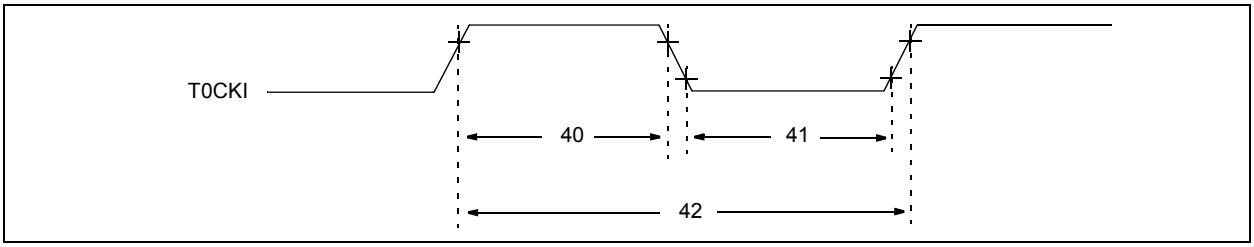


TABLE 13-8: TIMER0 CLOCK REQUIREMENTS

AC CHARACTERISTICS				Standard Operating Conditions (unless otherwise specified) Operating Temperature -40°C ≤ TA ≤ +85°C (industrial) -40°C ≤ TA ≤ +125°C (extended)				
Parm No.	**Sym**	**Characteristic**		**Min**	**Typ(1)**	**Max**	**Units**	**Conditions**
40	Tt0H	T0CKI High Pulse Width	No Prescaler	0.5 TCY + 20*	—	—	ns	
			With Prescaler	10*	—	—	ns	
41	Tt0L	T0CKI Low Pulse Width	No Prescaler	0.5 TCY + 20*	—	—	ns	
			With Prescaler	10*	—	—	ns	
42	Tt0P	T0CKI Period		20 or TCY + 40* N	—	—	ns	Whichever is greater. N = Prescale Value (1, 2, 4,..., 256)

* These parameters are characterized but not tested.

Note 1: Data in the Typical ("Typ") column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

NOTES:

FIGURE 14-6: MAXIMUM WDT IPD vs. VDD OVER TEMPERATURE

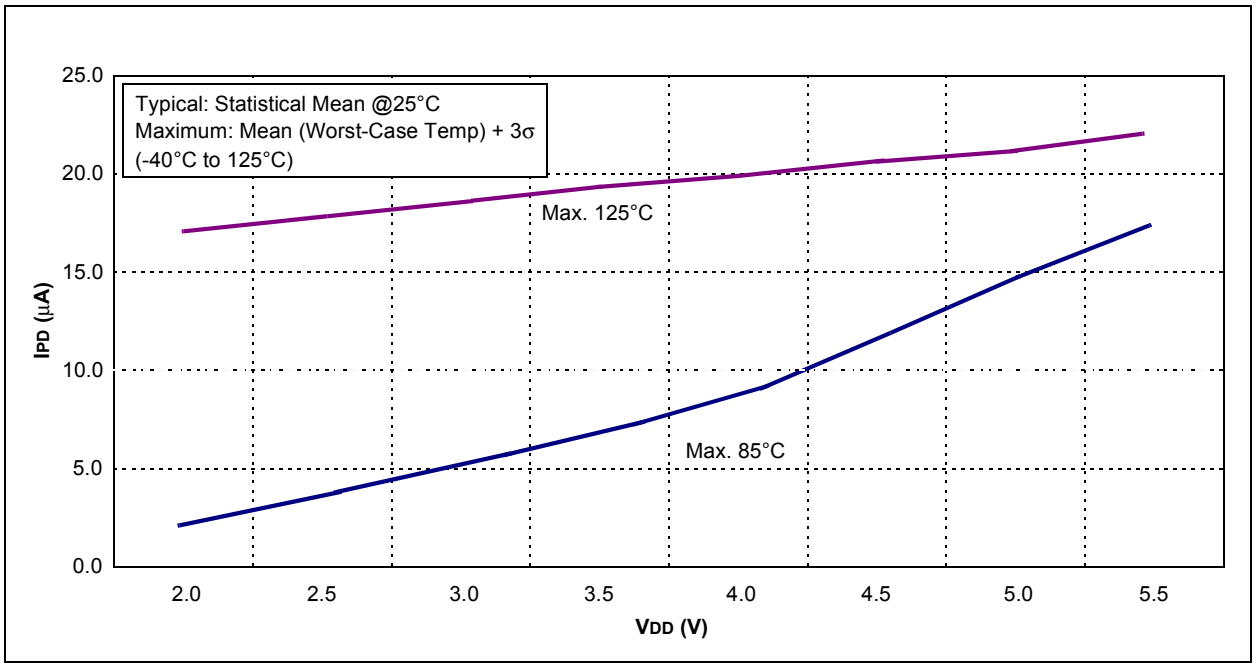


FIGURE 14-7: WDT TIME-OUT vs. VDD OVER TEMPERATURE (NO PRESCALER)

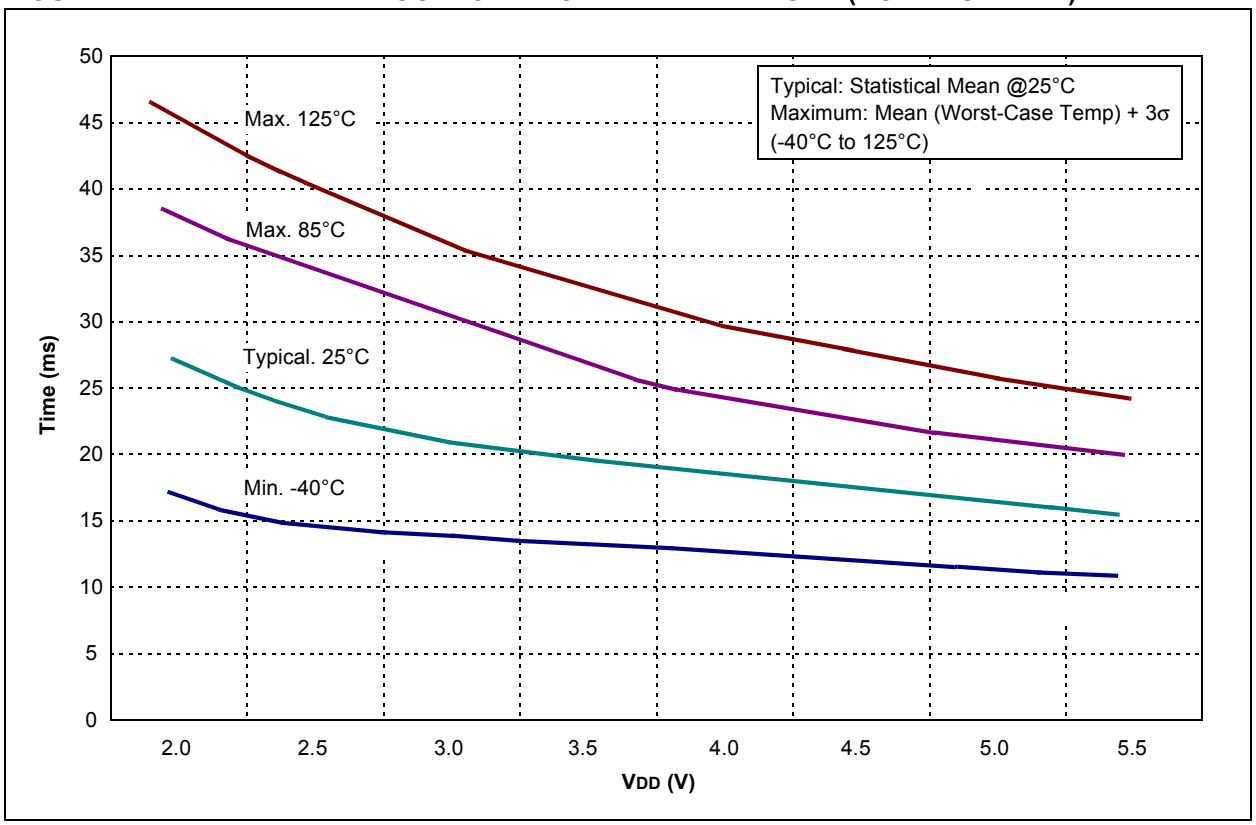
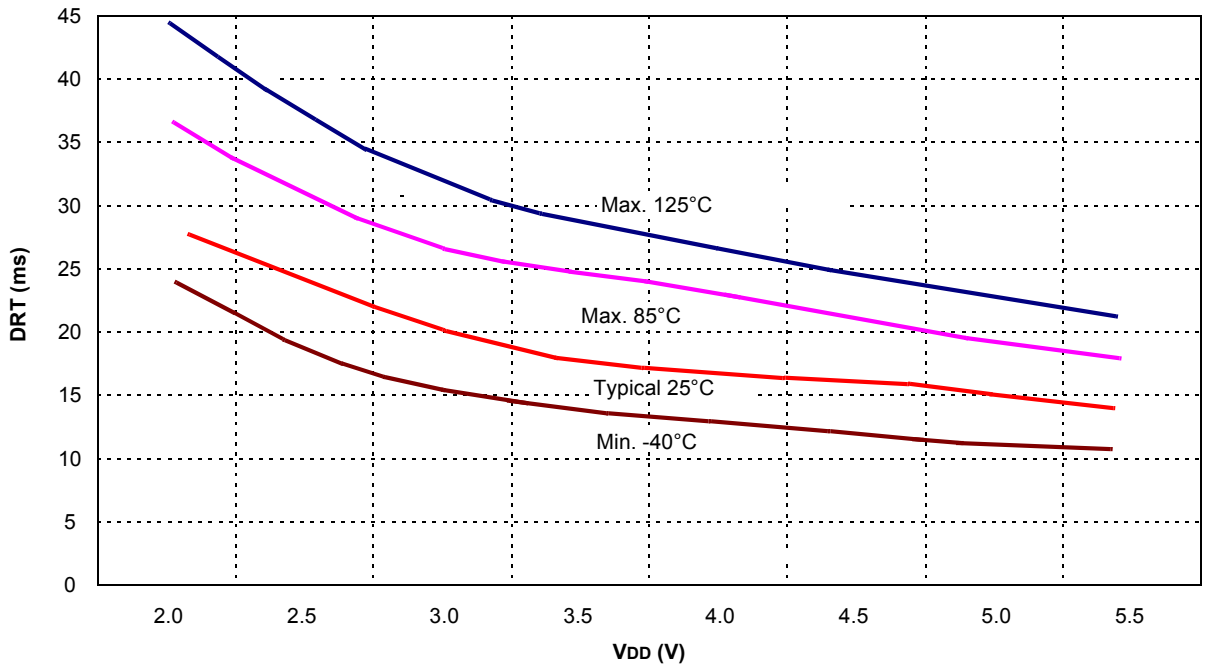


FIGURE 14-14: DEVICE RESET TIMER (HS, XT AND LP) vs. VDD

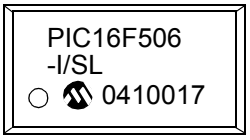


Note: See Table 13-7 if another clock mode is selected.

15.2 Package Marking Information (Cont'd)

14-Lead SOIC (3.90 mm)

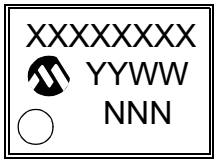
Example



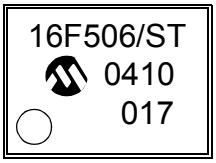
8-Lead MSOP

Example

14-Lead TSSOP (4.4 mm)

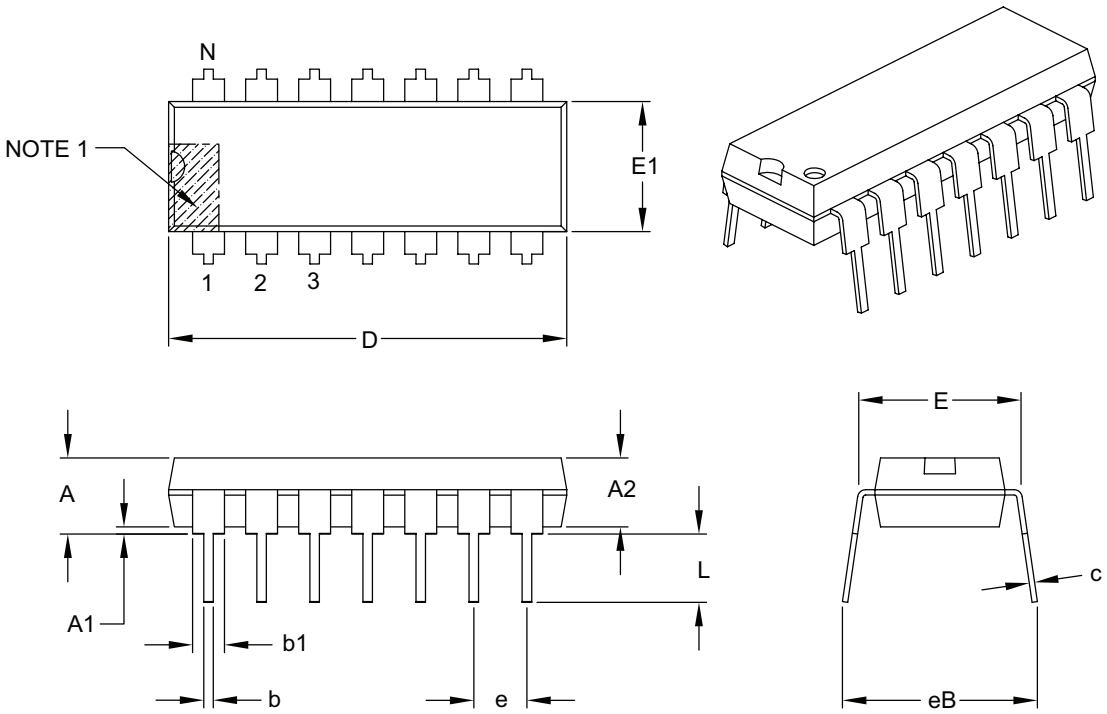


Example



14-Lead Plastic Dual In-Line (P) – 300 mil Body [PDIP]

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



Units		INCHES		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N		14	
Pitch	e		.100 BSC	
Top to Seating Plane	A	–	–	.210
Molded Package Thickness	A2	.115	.130	.195
Base to Seating Plane	A1	.015	–	–
Shoulder to Shoulder Width	E	.290	.310	.325
Molded Package Width	E1	.240	.250	.280
Overall Length	D	.735	.750	.775
Tip to Seating Plane	L	.115	.130	.150
Lead Thickness	c	.008	.010	.015
Upper Lead Width	b1	.045	.060	.070
Lower Lead Width	b	.014	.018	.022
Overall Row Spacing §	eB	–	–	.430

Notes:

1. Pin 1 visual index feature may vary, but must be located with the hatched area.
2. § Significant Characteristic.
3. Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010" per side.
4. Dimensioning and tolerancing per ASME Y14.5M.

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

Microchip Technology Drawing C04-005B

APPENDIX A: REVISION HISTORY

Revision A

Original release.

Revision B

Page 3 – Special Microcontroller Features and Low-Power Features sections.

PIC12F510 Pin Diagram.

Section 3.0 – Figure 3-1, Figure 3-2, Table 3-2, Table 3-3.

Section 4.0 – First paragraph, Section 4.2 - Figure references, Tables 4-1 and 4-2 (Note 1).

Section 5.0 – Table 5-2, Table 5-6 Title.

Section 6.0

Section 7.0 – First paragraph, Section 7.7, Register 7-1, Register 7-2, Register 7-3, Figure 7-1, Figure 7-2, Sections 7.4 through 7.7, Table 7-1.

Section 8.0 – Sections 8.0 through 8.2, Figure 8-1, Table 8-1.

Section 9.0 – Table 9-2, Register 9-1, Register 9-2, Table 9-3.

Section 10.0 – Registers 10-1 and 10-2 (Note 1), Table 10-2 (Note 2), Section 10.2.5, Section 10.3, Table 10-3, Table 10-4, Table 10-5, Section 10.4, Section 10.5, Section 10.6.1, Section 10.9, 10.9.1, 10.9.2, Section 10.11.

Section 13.0 – 13.1 DC Characteristics, 13.2 DC Characteristics, Table 13-1, Table 13-3, Table 13-4.

Revision C (03/2007)

Revised Table 3-2 GP3 and Legend; Revised Table 3-3 RB3 and Legend; Updated Registers to new format; Revised Section 9.1; Revised Table 9-2; Revised 13.1 DC Characteristics D025; Revised Table 13-2 and Table 13-3 and Notes; Replaced Package Drawings (Rev. AN); Added DFN package; Replaced Development Support Section; Revised Product ID System.

Revision D (11/2007)

Revised Table 1-1; Table 4-1, Table 4-2; Figure 4-5; Register 7-1 (Note 1); Register 8-1; Figure 13-4; 13.1 - 13.3; Table 13-1, Table 13-3, Table 13-6, Table 13-7, Table 13-9; Figure 14-4, Figure 14-14; Section 14.0; Packaging; Product ID System.

NOTES:

INDEX