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	5
Program Memory Size	768B (512 x 12)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	25 x 8
Voltage - Supply (Vcc/Vdd)	2V ~ 5.5V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	8-TSSOP, 8-MSOP (0.118", 3.00mm Width)
Supplier Device Package	8-MSOP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic12f508t-i-ms

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."

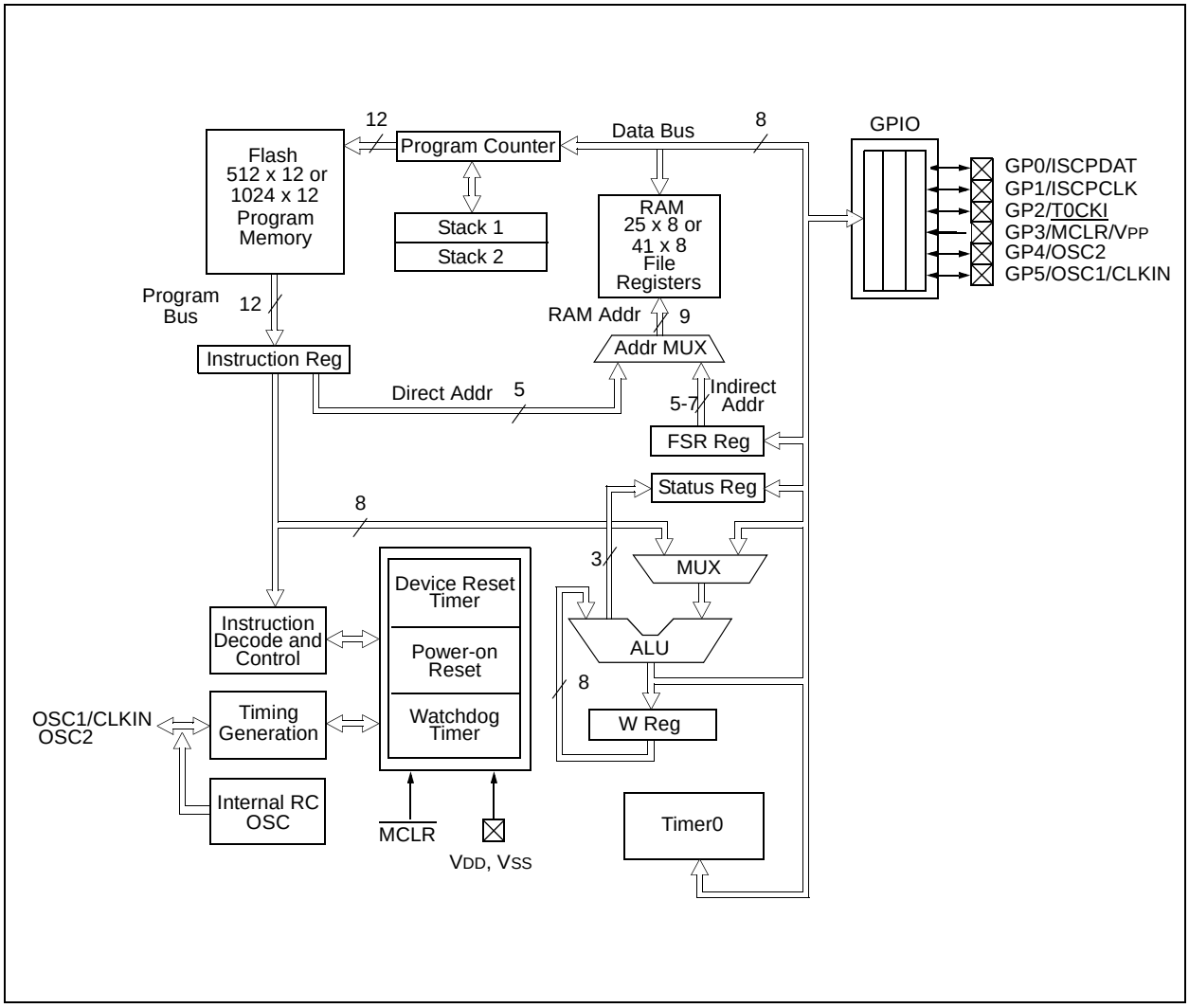
Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your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

Trademarks

QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
ISO/TS 16949:2002

Microchip received ISO/TS-16949:2002 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.

FIGURE 3-1: PIC12F508/509 BLOCK DIAGRAM

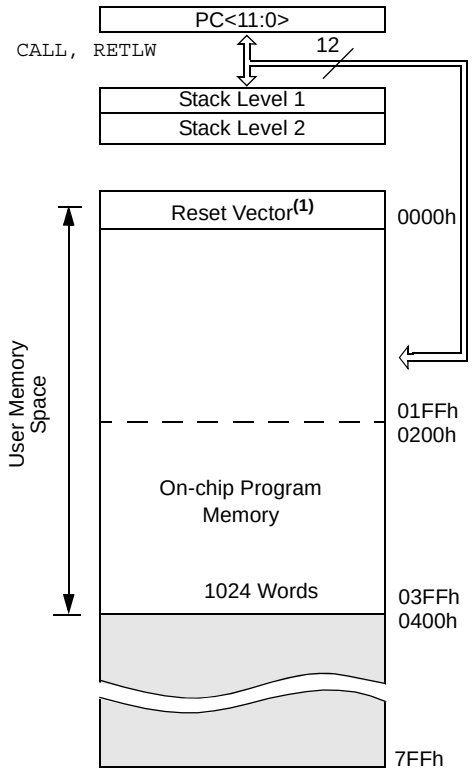


4.2 Program Memory Organization For The PIC16F505

The PIC16F505 device has a 11-bit Program Counter (PC) capable of addressing a 2K x 12 program memory space.

The 1K x 12 (0000h-03FFh) for the PIC16F505 are physically implemented. Refer to Figure 4-2. Accessing a location above this boundary will cause a wrap-around within the first 1K x 12 space. The effective Reset vector is at 0000h (see Figure 4-2). Location 03FFh contains the internal oscillator calibration value. This value should never be overwritten.

FIGURE 4-2: PROGRAM MEMORY MAP AND STACK FOR THE PIC16F505



Note 1: Address 0000h becomes the effective Reset vector. Location 03FFh contains the `MOVLW XX` internal oscillator calibration value.

4.3 Data Memory Organization

Data memory is composed of registers or bytes of RAM. Therefore, data memory for a device is specified by its register file. The register file is divided into two functional groups: Special Function Registers (SFR) and General Purpose Registers (GPR).

The Special Function Registers include the TMR0 register, the Program Counter (PCL), the STATUS register, the I/O registers (ports) and the File Select Register (FSR). In addition, Special Function Registers are used to control the I/O port configuration and prescaler options.

The General Purpose Registers are used for data and control information under command of the instructions.

For the PIC12F508/509, the register file is composed of 7 Special Function Registers, 9 General Purpose Registers and 16 or 32 General Purpose Registers accessed by banking (see Figure 4-3 and Figure 4-4).

For the PIC16F505, the register file is composed of 8 Special Function Registers, 8 General Purpose Registers and 64 General Purpose Registers accessed by banking (Figure 4-5).

4.3.1 GENERAL PURPOSE REGISTER FILE

The General Purpose Register file is accessed, either directly or indirectly, through the File Select Register (FSR). See **Section 4.9 “Indirect Data Addressing: INDF and FSR Registers”**.

4.3.2 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 (PIC12F508/509)

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on Power-On Reset(2)	Page #
00h	INDF	Uses Contents of FSR to Address Data Memory (not a physical register)								xxxx xxxx	28
01h	TMR0	8-bit Real-Time Clock/Counter								xxxx xxxx	35
02h(1)	PCL	Low-order 8 bits of PC								1111 1111	27
03h	STATUS	GPWUF	—	PA0(5)	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0-01 1xxx(3)	22
04h	FSR	Indirect Data Memory Address Pointer								111x xxxx	28
04h(4)	FSR	Indirect Data Memory Address Pointer								110x xxxx	28
05h	OSCCAL	CAL6	CAL5	CAL4	CAL3	CAL2	CAL1	CAL0	—	1111 111-	26
06h	GPIO	—	—	GP5	GP4	GP3	GP2	GP1	GP0	--xx xxxx	31
N/A	TRISGPIO	—	—	I/O Control Register						--11 1111	31
N/A	OPTION	$\overline{GPWU}$	$\overline{GPPU}$	TOCS	TOSE	PSA	PS2	PS1	PS0	1111 1111	24

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

Note 1: The upper byte of the Program Counter is not directly accessible. See **Section 4.7 “Program Counter”** for an explanation of how to access these bits.

2: Other (non Power-up) Resets include external Reset through $\overline{MCLR}$, Watchdog Timer and wake-up on pin change Reset.

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

4: PIC12F509 only.

5: This bit is used on the PIC12F509. For code compatibility do not use this bit on the PIC12F508.

TABLE 4-2: SPECIAL FUNCTION REGISTER (SFR) SUMMARY (PIC16F505)

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on Power-On Reset(2)	Page #
00h	INDF	Uses Contents of FSR to Address Data Memory (not a physical register)								xxxx xxxx	28
01h	TMR0	8-bit Real-Time Clock/Counter								xxxx xxxx	35
02h(1)	PCL	Low-order 8 bits of PC								1111 1111	27
03h	STATUS	RBWUF	—	PA0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0-01 1xxx	22
04h	FSR	Indirect Data Memory Address Pointer								100x xxxx	28
05h	OSCCAL	CAL6	CAL5	CAL4	CAL3	CAL2	CAL1	CAL0	—	1111 111-	26
06h	PORTB	—	—	RB5	RB4	RB3	RB2	RB1	RB0	--xx xxxx	31
07h	PORTC	—	—	RC5	RC4	RC3	RC2	RC1	RC0	--xx xxxx	31
N/A	TRISB	—	—	I/O Control Register						--11 1111	31
N/A	TRISC	—	—	I/O Control Register						--11 1111	31
N/A	OPTION	$\overline{RBWU}$	$\overline{RBPU}$	TOCS	TOSE	PSA	PS2	PS1	PS0	1111 1111	25

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

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

2: Other (non Power-up) Resets include external reset through $\overline{MCLR}$, Watchdog Timer and wake-up on pin change Reset.

NOTES:

5.0 I/O PORT

As with any other register, the I/O register(s) can be written and read under program control. However, read instructions (e.g., `MOVF PORTB,W`) always read the I/O pins independent of the pin's Input/Output modes. On Reset, all I/O ports are defined as input (inputs are at high-impedance) since the I/O control registers are all set.

Note:	On the PIC12F508/509, I/O PORTB is referenced as GPIO. On the PIC16F505, I/O PORTB is referenced as PORTB.

5.1 PORTB/GPIO

PORTB/GPIO is an 8-bit I/O register. Only the low-order 6 bits are used (RB/GP<5:0>). Bits 7 and 6 are unimplemented and read as '0's. Please note that RB3/GP3 is an input only pin. The Configuration Word can set several I/O's to alternate functions. When acting as alternate functions, the pins will read as '0' during a port read. Pins RB0/GP0, RB1/GP1, RB3/GP3 and RB4 can be configured with weak pull-ups and also for wake-up on change. The wake-up on change and weak pull-up functions are not pin selectable. If RB3/GP3/$\overline{MCLR}$ is configured as $\overline{MCLR}$, weak pull-up is always on and wake-up on change for this pin is not enabled.

5.2 PORTC (PIC16F505 Only)

PORTC is an 8-bit I/O register. Only the low-order 6 bits are used (RC<5:0>). Bits 7 and 6 are unimplemented and read as '0's.

Note:	On power-up, TOCKI functionality is enabled in the OPTION register and must be disabled to allow RC5 to be used as general purpose I/O.

5.3 TRIS Registers

The Output Driver Control register is loaded with the contents of the W register by executing the `TRIS f` instruction. A '1' from a TRIS register bit puts the corresponding output driver in a High-Impedance mode. A '0' puts the contents of the output data latch on the selected pins, enabling the output buffer. The exceptions are RB3/GP3, which is input only and the T0CKI pin, which may be controlled by the OPTION register. See Register 4-3 and Register 4-4.

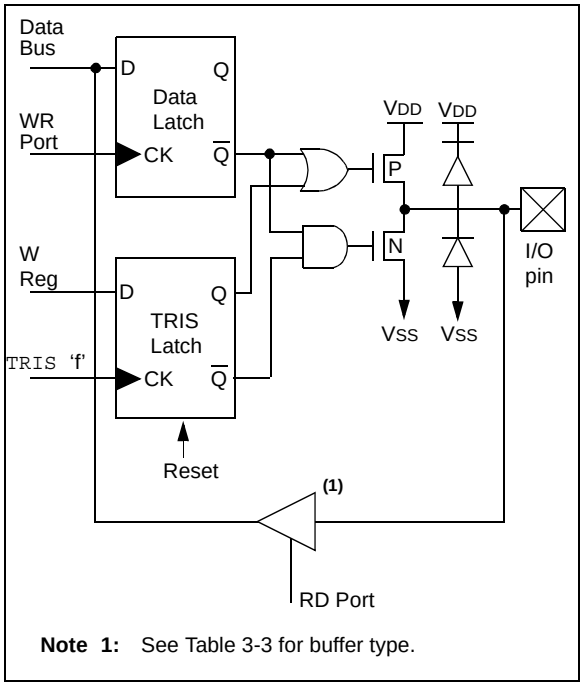
Note:	A read of the ports reads the pins, not the output data latches. That is, if an output driver on a pin is enabled and driven high, but the external system is holding it low, a read of the port will indicate that the pin is low.

The TRIS registers are "write-only" and are set (output drivers disabled) upon Reset.

5.4 I/O Interfacing

The equivalent circuit for an I/O port pin is shown in Figure 5-2. All port pins, except RB3/GP3 which is input only, 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 PORTB, 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 TRIS must be cleared (= `0`). For use as an input, the corresponding TRIS bit must be set. Any I/O pin (except RB3/GP3) can be programmed individually as input or output.

FIGURE 5-1: PIC12F508/509/16F505 EQUIVALENT CIRCUIT FOR A SINGLE I/O PIN



Note 1: See Table 3-3 for buffer type.

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)	$\overline{\text{GPWU}}$	$\overline{\text{GPPU}}$	TOCS	TOSE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
N/A	OPTION(2)	$\overline{\text{RBWU}}$	$\overline{\text{RBPU}}$	TOCS	TOSE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
03h	STATUS(1)	GPWUF	—	PAO	$\overline{\text{TO}}$	$\overline{\text{PD}}$	Z	DC	C	0-01 1xxx	q00q quuu(3)
03h	STATUS(2)	RBWUF	—	PAO	$\overline{\text{TO}}$	$\overline{\text{PD}}$	Z	DC	C	0-01 1xxx	q00q 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: Shaded cells are not used by Port registers, read as '0'. – = unimplemented, read as '0', x = unknown, u = unchanged, q = depends on condition.

Note 1: PIC12F508/509 only.

2: PIC16F505 only.

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

6.2 Prescaler

An 8-bit counter is available as a prescaler for the Timer0 module or as a postscaler for the Watchdog Timer (WDT), respectively (see **Section 7.6 "Watchdog Timer (WDT)"**). For simplicity, this counter is being referred to as "prescaler" throughout this data sheet.

Note:	The prescaler may be used by either the Timer0 module or the WDT, but not both. Thus, a prescaler assignment for the Timer0 module means that there is no prescaler for the WDT and vice versa.

The PSA and PS<2:0> bits (OPTION<3:0>) determine prescaler assignment and prescale ratio.

When assigned to the Timer0 module, all instructions writing to the TMR0 register (e.g., `CLRF 1`, `MOVWF 1`, `BSF 1, x`, etc.) will clear the prescaler. When assigned to WDT, a `CLRWDT` instruction will clear the prescaler along with the WDT. The prescaler is neither readable nor writable. On a Reset, the prescaler contains all '0's.

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)

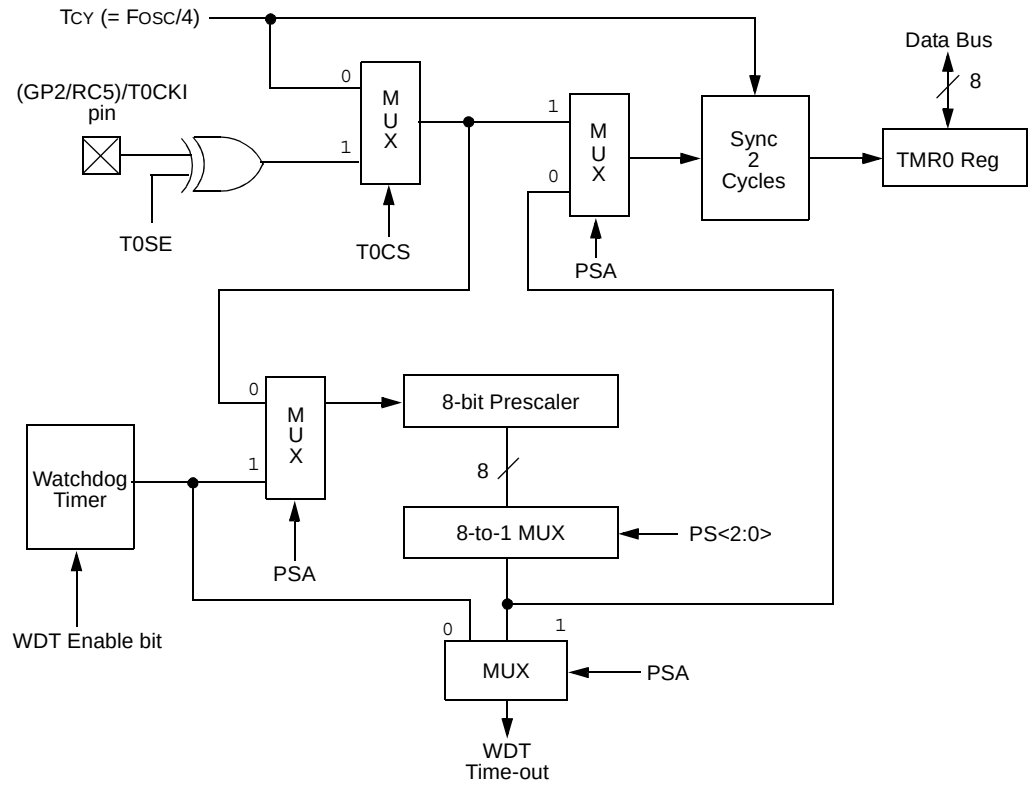
```
CLRWDT              ;Clear WDT
CLRF   TMR0         ;Clear TMR0 & Prescaler
MOVLW  '00xx1111'b  ;These 3 lines (5, 6, 7)
OPTION              ;are required only if
                    ;desired
CLRWDT              ;PS<2:0> are 000 or 001
MOVLW  '00xx1xxx'b  ;Set Postscaler to
OPTION              ;desired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

To change the 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(1), (2)



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

2: T0CKI is shared with pin RC5 on the PIC16F505 and pin GP2 on the PIC12F508/509.

TABLE 8-2: INSTRUCTION SET SUMMARY

Mnemonic, Operands		Description	Cycles	12-Bit Opcode			Status Affected	Notes
				MSb		LSb		
ADDWF	f, d	Add W and f	1	0001	11df	ffff	C, DC, Z	**1, 2, 4**
ANDWF	f, d	AND W with f	1	0001	01df	ffff	Z	**2, 4**
CLRF	f	Clear f	1	0000	011f	ffff	Z	**4**
CLRW	—	Clear W	1	0000	0100	0000	Z	
COMF	f, d	Complement f	1	0010	01df	ffff	Z	
DECF	f, d	Decrement f	1	0000	11df	ffff	Z	**2, 4**
DECFSZ	f, d	Decrement f, Skip if 0	1**(2)**	0010	11df	ffff	None	**2, 4**
INCF	f, d	Increment f	1	0010	10df	ffff	Z	**2, 4**
INCFSZ	f, d	Increment f, Skip if 0	1**(2)**	0011	11df	ffff	None	**2, 4**
IORWF	f, d	Inclusive OR W with f	1	0001	00df	ffff	Z	**2, 4**
MOVF	f, d	Move f	1	0010	00df	ffff	Z	**2, 4**
MOVWF	f	Move W to f	1	0000	001f	ffff	None	**1, 4**
NOP	—	No Operation	1	0000	0000	0000	None	
RLF	f, d	Rotate left f through Carry	1	0011	01df	ffff	C	**2, 4**
RRF	f, d	Rotate right f through Carry	1	0011	00df	ffff	C	**2, 4**
SUBWF	f, d	Subtract W from f	1	0000	10df	ffff	C, DC, Z	**1, 2, 4**
SWAPF	f, d	Swap f	1	0011	10df	ffff	None	**2, 4**
XORWF	f, d	Exclusive OR W with f	1	0001	10df	ffff	Z	**2, 4**
BIT-ORIENTED FILE REGISTER OPERATIONS								
BCF	f, b	Bit Clear f	1	0100	bbbf	ffff	None	**2, 4**
BSF	f, b	Bit Set f	1	0101	bbbf	ffff	None	**2, 4**
BTFSC	f, b	Bit Test f, Skip if Clear	1**(2)**	0110	bbbf	ffff	None	
BTFSS	f, b	Bit Test f, Skip if Set	1**(2)**	0111	bbbf	ffff	None	
LITERAL AND CONTROL OPERATIONS								
ANDLW	k	AND literal with W	1	1110	kkkk	kkkk	Z	
CALL	k	Call Subroutine	2	1001	kkkk	kkkk	None	**1**
CLRWDT	—	Clear Watchdog Timer	1	0000	0000	0100	$\overline{TO}$, $\overline{PD}$	
GOTO	k	Unconditional branch	2	101k	kkkk	kkkk	None	
IORLW	k	Inclusive OR literal with W	1	1101	kkkk	kkkk	Z	
MOVLW	k	Move literal to W	1	1100	kkkk	kkkk	None	
OPTION	—	Load OPTION register	1	0000	0000	0010	None	
RETLW	k	Return, place literal in W	2	1000	kkkk	kkkk	None	
SLEEP	—	Go into Standby mode	1	0000	0000	0011	$\overline{TO}$, $\overline{PD}$	
TRIS	f	Load TRIS register	1	0000	0000	0fff	None	**3**
XORLW	k	Exclusive OR literal to W	1	1111	kkkk	kkkk	Z	

Note 1: The 9th bit of the program counter will be forced to a '0' by any instruction that writes to the PC except for GOTO. See **Section 4.7 "Program Counter"**.

2: When an I/O register is modified as a function of itself (e.g. MOVF PORTB, 1), the value used will be that value present on the pins themselves. For example, if the data latch is '1' for a pin configured as input and is driven low by an external device, the data will be written back with a '0'.

3: The instruction TRIS f, where f = 6, causes the contents of the W register to be written to the tri-state latches of PORTB. A '1' forces the pin to a high-impedance state and disables the output buffers.

4: If this instruction is executed on the TMR0 register (and, where applicable, d = 1), the prescaler will be cleared (if assigned to TMR0).

IORWF	Inclusive OR W with f
Syntax:	[*label*] IORWF f,d
Operands:	$0 \le f \le 31$ $d \in [0,1]$
Operation:	(W).OR. (f) → (dest)
Status Affected:	Z
Description:	Inclusive OR the W register with register 'f'. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is placed back in register 'f'.

MOVF	Move f
Syntax:	[*label*] MOVF f,d
Operands:	$0 \le f \le 31$ $d \in [0,1]$
Operation:	(f) → (dest)
Status Affected:	Z
Description:	The contents of register 'f' are moved to destination 'd'. If 'd' is '0', destination is the W register. If 'd' is '1', the destination is file register 'f'. 'd' = 1 is useful as a test of a file register, since status flag Z is affected.

MOVLW	Move Literal to W
Syntax:	[*label*] MOVLW k
Operands:	$0 \le k \le 255$
Operation:	k → (W)
Status Affected:	None
Description:	The eight-bit literal 'k' is loaded into the W register. The "don't cares" will assembled as '0's.

MOVWF	Move W to f
Syntax:	[*label*] MOVWF f
Operands:	$0 \le f \le 31$
Operation:	(W) → (f)
Status Affected:	None
Description:	Move data from the W register to register 'f'.

NOP	No Operation
Syntax:	[*label*] NOP
Operands:	None
Operation:	No operation
Status Affected:	None
Description:	No operation.

OPTION	Load OPTION Register
Syntax:	[*label*] OPTION
Operands:	None
Operation:	(W) → OPTION
Status Affected:	None
Description:	The content of the W register is loaded into the OPTION register.

9.0 DEVELOPMENT SUPPORT

The PIC® microcontrollers are supported with a full range of hardware and software development tools:

- Integrated Development Environment
  - MPLAB® IDE Software
- Assemblers/Compilers/Linkers
  - MPASM™ Assembler
  - MPLAB C18 and MPLAB C30 C Compilers
  - MPLINK™ Object Linker/ MPLIB™ Object Librarian
  - MPLAB ASM30 Assembler/Linker/Library
- Simulators
  - MPLAB SIM Software Simulator
- Emulators
  - MPLAB ICE 2000 In-Circuit Emulator
  - MPLAB REAL ICE™ In-Circuit Emulator
- In-Circuit Debugger
  - MPLAB ICD 2
- Device Programmers
  - PICSTART® Plus Development Programmer
  - MPLAB PM3 Device Programmer
  - PICkit™ 2 Development Programmer
- Low-Cost Demonstration and Development Boards and Evaluation Kits

9.1 MPLAB Integrated Development Environment Software

The MPLAB IDE software brings an ease of software development previously unseen in the 8/16-bit microcontroller market. The MPLAB IDE is a Windows® operating system-based application that contains:

- A single graphical interface to all debugging tools
  - Simulator
  - Programmer (sold separately)
  - Emulator (sold separately)
  - In-Circuit Debugger (sold separately)
- A full-featured editor with color-coded context
- A multiple project manager
- Customizable data windows with direct edit of contents
- High-level source code debugging
- Visual device initializer for easy register initialization
- Mouse over variable inspection
- Drag and drop variables from source to watch windows
- Extensive on-line help
- Integration of select third party tools, such as HI-TECH Software C Compilers and IAR C Compilers

The MPLAB IDE allows you to:

- Edit your source files (either assembly or C)
- One touch assemble (or compile) and download to PIC MCU emulator and simulator tools (automatically updates all project information)
- Debug using:
  - Source files (assembly or C)
  - Mixed assembly and C
  - Machine code

MPLAB IDE supports multiple debugging tools in a single development paradigm, from the cost-effective simulators, through low-cost in-circuit debuggers, to full-featured emulators. This eliminates the learning curve when upgrading to tools with increased flexibility and power.

TABLE 10-2: PULL-UP RESISTOR RANGES – PIC12F508/509/16F505

VDD (Volts)	Temperature (°C)	Min.	Typ.	Max.
GP0(RBO)/GP1(RB1)				
2.0	–40	73K	105K	186K
	25	73K	113K	187K
	85	82K	123K	190K
	125	86K	132k	190K
5.5	–40	15K	21K	33K
	25	15K	22K	34K
	85	19K	26k	35K
	125	23K	29K	35K
GP3(RB3)				
2.0	–40	63K	81K	96K
	25	77K	93K	116K
	85	82K	96k	116K
	125	86K	100K	119K
5.5	–40	16K	20k	22K
	25	16K	21K	23K
	85	24K	25k	28K
	125	26K	27K	29K

* These parameters are characterized but not tested.

10.3 Timing Parameter Symbology and Load Conditions – PIC12F508/509/16F505

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

1. TppS2ppS
2. TppS

T	
F Frequency	T Time

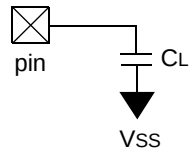
Lowercase subscripts (pp) and their meanings:

pp			
2	to	mc	$\overline{\text{MCLR}}$
ck	CLKOUT	osc	Oscillator
cy	Cycle time	os	OSC1
drt	Device Reset Timer	t0	T0CKI
io	I/O port	wdt	Watchdog Timer

Uppercase letters and their meanings:

S			
F	Fall	P	Period
H	High	R	Rise
I	Invalid (high-impedance)	V	Valid
L	Low	Z	High-impedance

FIGURE 10-3: LOAD CONDITIONS – PIC12F508/509/16F505



Legend:

C_L = 50 pF for all pins except OSC2

15 pF for OSC2 in XT, HS or LP modes when external clock is used to drive OSC1

FIGURE 10-4: EXTERNAL CLOCK TIMING – PIC12F508/509/16F505

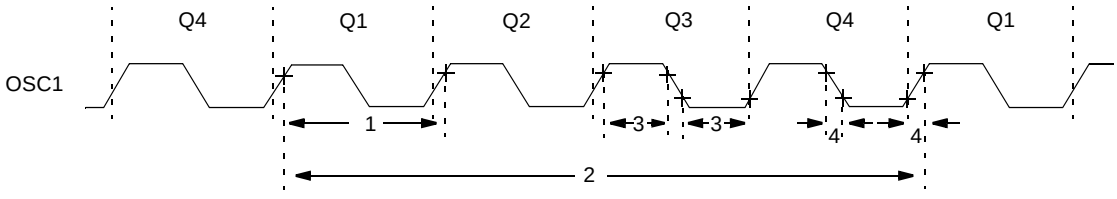


TABLE 10-3: EXTERNAL CLOCK TIMING REQUIREMENTS – PIC12F508/509/16F505

AC CHARACTERISTICS			Standard Operating Conditions (unless otherwise specified) Operating Temperature -40°C ≤ TA ≤ +85°C (industrial), -40°C ≤ TA ≤ +125°C (extended) Operating Voltage VDD range is described in **Section 10.1 "Power-on Reset (POR)"**				
Param No.	**Sym.**	**Characteristic**	**Min.**	**Typ(1)**	**Max.**	**Units**	**Conditions**
1A	FOSC	External CLKIN Frequency(2)	DC	—	4	MHz	XT Oscillator mode
			DC	—	20	MHz	EC, HS Oscillator mode (PIC16F505 only)
			DC	—	200	kHz	LP Oscillator mode
		Oscillator Frequency(2)	—	—	4	MHz	EXTRC Oscillator mode
			0.1	—	4	MHz	XT Oscillator mode
			4	—	20	MHz	HS Oscillator mode (PIC16F505 only)
			—	—	200	kHz	LP Oscillator mode
1	TOSC	External CLKIN Period(2)	250	—	—	ns	XT Oscillator mode
			50	—	—	ns	EC, HS Oscillator mode (PIC16F505 only)
			5	—	—	µs	LP Oscillator mode
		Oscillator Period(2)	250	—	—	ns	EXTRC Oscillator mode
			250	—	10,000	ns	XT Oscillator mode
			50	—	250	ns	HS Oscillator mode (PIC16F505 only)
			5	—	—	µs	LP Oscillator mode
2	TCY	Instruction Cycle Time	200	4/FOSC	—	ns	
3	TosL, TosH	Clock in (OSC1) Low or High Time	50*	—	—	ns	XT Oscillator
			2*	—	—	µs	LP Oscillator
			10*	—	—	ns	EC, HS Oscillator (PIC16F505 only)
4	TosR, TosF	Clock in (OSC1) Rise or Fall Time	—	—	25*	ns	XT Oscillator
			—	—	50*	ns	LP Oscillator
			—	—	15*	ns	EC, HS Oscillator (PIC16F505 only)

* 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.

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

TABLE 10-4: CALIBRATED INTERNAL RC FREQUENCIES – PIC12F508/509/16F505

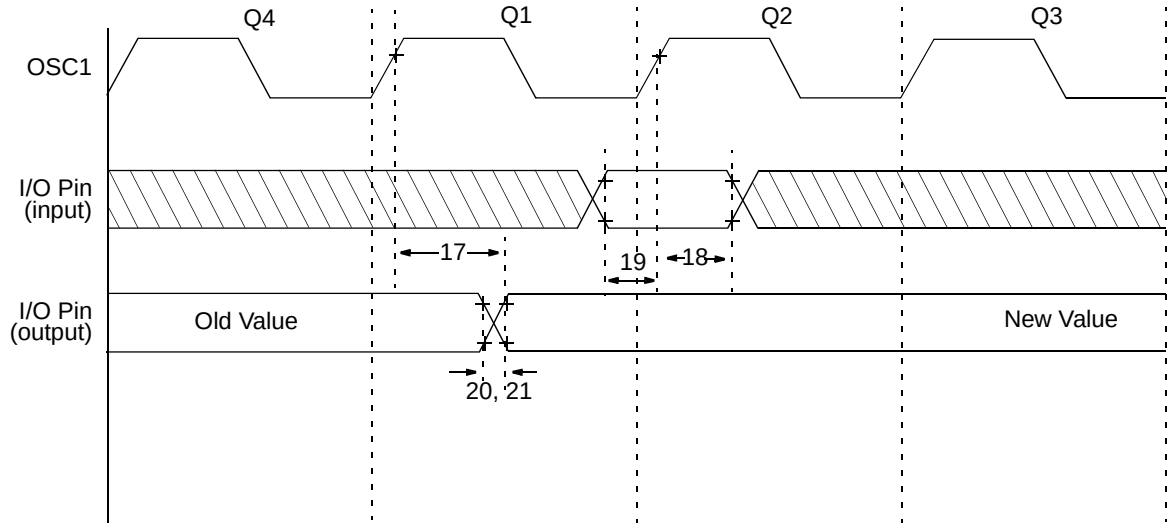
AC CHARACTERISTICS			Standard Operating Conditions (unless otherwise specified) Operating Temperature -40°C ≤ TA ≤ +85°C (industrial), -40°C ≤ TA ≤ +125°C (extended)					
Param No.	**Sym.**	**Characteristic**	**Freq Tolerance**	**Min.**	**Typ†**	**Max.**	**Units**	**Conditions**
F10	FOSC	Internal Calibrated INTOSC Frequency(1)	± 1%	3.96	4.00	4.04	MHz	VDD = 3.5V, TA = 25°C
			± 2%	3.92	4.00	4.08	MHz	2.5V ≤ VDD ≤ 5.5V 0°C ≤ TA ≤ +85°C
			± 5%	3.80	4.00	4.20	MHz	2.0V ≤ VDD ≤ 5.5V -40°C ≤ TA ≤ +85°C (Ind.) -40°C ≤ TA ≤ +125°C (Ext.)

* These parameters are characterized but not tested.

† 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.

Note 1: To ensure these oscillator frequency tolerances, VDD and VSS must be capacitively decoupled as close to the device as possible. 0.1 uF and 0.01 uF values in parallel are recommended.

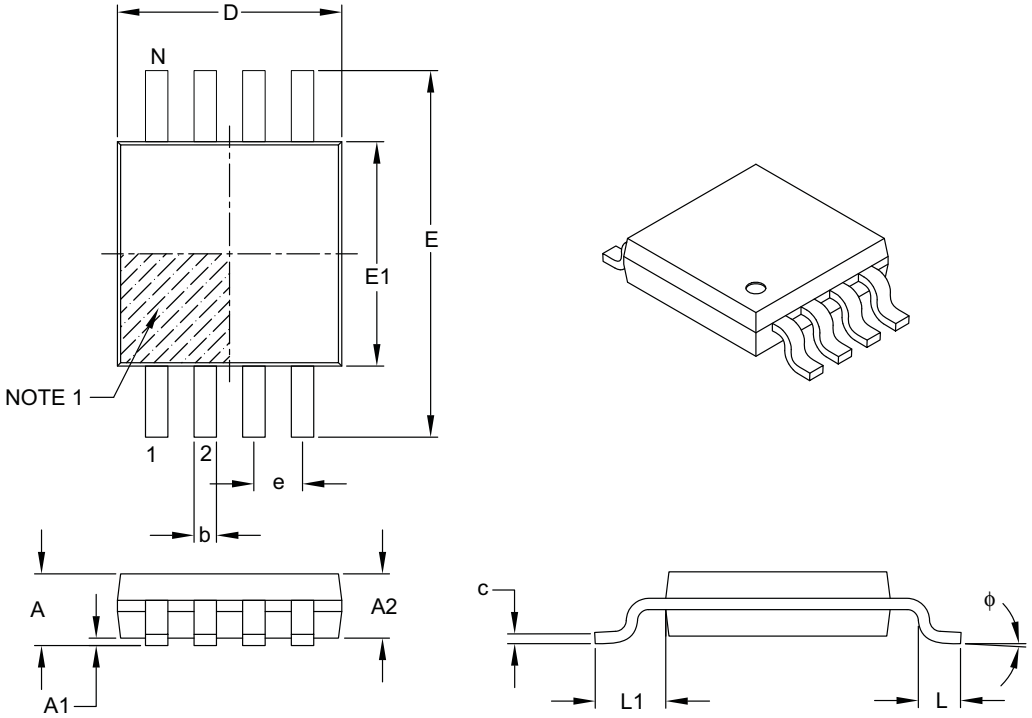
FIGURE 10-5: I/O TIMING – PIC12F508/509/16F505



Note: All tests must be done with specified capacitive loads (see data sheet) 50 pF on I/O pins and CLKOUT.

8-Lead Plastic Micro Small Outline Package (MS) [MSOP]

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



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N		8	
Pitch	e		0.65 BSC	
Overall Height	A	–	–	1.10
Molded Package Thickness	A2	0.75	0.85	0.95
Standoff	A1	0.00	–	0.15
Overall Width	E		4.90 BSC	
Molded Package Width	E1		3.00 BSC	
Overall Length	D		3.00 BSC	
Foot Length	L	0.40	0.60	0.80
Footprint	L1		0.95 REF	
Foot Angle	ϕ	0°	–	8°
Lead Thickness	c	0.08	–	0.23
Lead Width	b	0.22	–	0.40

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.15 mm per side.
3. 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.

Microchip Technology Drawing C04-111B

APPENDIX A: REVISION HISTORY

Revision A (April 2004)

Original data sheet for PIC12F508/509/16F505 devices

Revision B (June 2005)

Update packages

Revision C (03/2007)

Revised Table 3-2 Legend; Revised Table 3-3 RB3 and Legend; Revised Table 10-4 F10; Replaced Package Drawings (Rev. AN); Added DFN package; Replaced Development Support Section; Revised Product ID System.

Revision D (12/2007)

Revised Title; Operating Current; Table 1-1 added DFN and revised note; Revised Section 3.0, last paragraph; Revised Figure 4-4; Revised Table 4-2 (FSR); Revised Register 7-1 and Register 7-2; Revised Section 7.2.2; Revised Table 7-3, Note 2; Revised Table 7-4 (FSR) and Note 2; Deleted Section 7.3.1: External Clock In and Figure 7-6; Revised new Section 7.3.1; Replaced TBD with new data in Tables 10-4 and 10-5; Revised Tables 10-1 (Industrial), 10-2 (Extended), and Tables 10-1 (Industrial, Extended) and 10-2 (Pull-up Resistor Ranges), 10-3, 10-4 and 10-6; Revised Figure 10-1, Figure 10-2; Section 11.0, Added Char data; Revised Package Marking Information; Revised Product ID System.

Revision E (08/2009)

Added PIC16F505 16-Pin diagram (QFN); Added Note after subsection 5.2 PORTC; Updated Note 4 and deleted Note 5, Table 10-1; Deleted Param. No. D061 (Table 10-1) and Param. No. D061A becomes D061; Added QFN Package Information; Revised Product Identification System; Added Figures 11-14, 11-15, 11-16, 11-7 to Char Data section; Other minor corrections; Removed Preliminary status.

INDEX