



Welcome to [E-XFL.COM](https://www.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	OTP
EEPROM Size	-
RAM Size	25 x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 5.5V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	8-SOIC (0.209", 5.30mm Width)
Supplier Device Package	8-SOIJ
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic12c508at-04i-sm

TABLE OF CONTENTS

1.0	General Description	4
2.0	PIC12C5XX Device Varieties	7
3.0	Architectural Overview	9
4.0	Memory Organization	13
5.0	I/O Port	21
6.0	Timer0 Module and TMR0 Register	25
7.0	EEPROM Peripheral Operation	29
8.0	Special Features of the CPU	35
9.0	Instruction Set Summary	47
10.0	Development Support	59
11.0	Electrical Characteristics - PIC12C508/PIC12C509	65
12.0	DC and AC Characteristics - PIC12C508/PIC12C509	75
13.0	Electrical Characteristics PIC12C508A/PIC12C509A/PIC12LC508A/PIC12LC509A/PIC12CR509A/ PIC12CE518/PIC12CE519/ PIC12LCE518/PIC12LCE519/PIC12LCR509A	79
14.0	DC and AC Characteristics PIC12C508A/PIC12C509A/PIC12LC508A/PIC12LC509A/PIC12CE518/PIC12CE519/PIC12CR509A/ PIC12LCE518/PIC12LCE519/ PIC12LCR509A	93
15.0	Packaging Information	99
	Index	105
	PIC12C5XX Product Identification System	109
	Sales and Support:	109

To Our Valued Customers

Most Current Data Sheet

To obtain the most up-to-date version of this data sheet, please register at our Worldwide Web site at:

<http://www.microchip.com>

You can determine the version of a data sheet by examining its literature number found on the bottom outside corner of any page. The last character of the literature number is the version number. e.g., DS30000A is version A of document DS30000.

Errata

An errata sheet may exist for current devices, describing minor operational differences (from the data sheet) and recommended workarounds. As device/documentation issues become known to us, we will publish an errata sheet. The errata will specify the revision of silicon and revision of document to which it applies.

To determine if an errata sheet exists for a particular device, please check with one of the following:

- Microchip's Worldwide Web site; <http://www.microchip.com>
- Your local Microchip sales office (see last page)
- The Microchip Corporate Literature Center; U.S. FAX: (602) 786-7277

When contacting a sales office or the literature center, please specify which device, revision of silicon and data sheet (include literature number) you are using.

Corrections to this Data Sheet

We constantly strive to improve the quality of all our products and documentation. We have spent a great deal of time to ensure that this document is correct. However, we realize that we may have missed a few things. If you find any information that is missing or appears in error, please:

- Fill out and mail in the reader response form in the back of this data sheet.
- E-mail us at webmaster@microchip.com.

We appreciate your assistance in making this a better document.

PIC12C5XX

NOTES:

4.3 STATUS Register

This register contains the arithmetic status of the ALU, the RESET status, and the page preselect bit for program memories larger than 512 words.

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

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

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

FIGURE 4-4: STATUS REGISTER (ADDRESS:03h)

R/W-0	R/W-0	R/W-0	R-1	R-1	R/W-x	R/W-x	R/W-x
GPWUF	—	PA0	\overline{TO}	\overline{PD}	Z	DC	C
bit7	6	5	4	3	2	1	bit0
<p>bit 7: GPWUF: GPIO reset bit 1 = Reset due to wake-up from SLEEP on pin change 0 = After power up or other reset</p> <p>bit 6: Unimplemented</p> <p>bit 5: PA0: Program page preselect bits 1 = Page 1 (200h - 3FFh) - PIC12C509, PIC12C509A, PIC12CR509A and PIC12CE519 0 = Page 0 (000h - 1FFh) - PIC12C5XX Each page is 512 bytes. Using the PA0 bit as a general purpose read/write bit in devices which do not use it for program page preselect is not recommended since this may affect upward compatibility with future products.</p> <p>bit 4: \overline{TO}: Time-out bit 1 = After power-up, <code>CLRWDT</code> instruction, or <code>SLEEP</code> instruction 0 = A WDT time-out occurred</p> <p>bit 3: \overline{PD}: Power-down bit 1 = After power-up or by the <code>CLRWDT</code> instruction 0 = By execution of the <code>SLEEP</code> instruction</p> <p>bit 2: Z: Zero bit 1 = The result of an arithmetic or logic operation is zero 0 = The result of an arithmetic or logic operation is not zero</p> <p>bit 1: DC: Digit carry/borrow bit (for <code>ADDWF</code> and <code>SUBWF</code> instructions) ADDWF 1 = A carry from the 4th low order bit of the result occurred 0 = A carry from the 4th low order bit of the result did not occur SUBWF 1 = A borrow from the 4th low order bit of the result did not occur 0 = A borrow from the 4th low order bit of the result occurred</p> <p>bit 0: C: Carry/borrow bit (for <code>ADDWF</code>, <code>SUBWF</code> and <code>RRF</code>, <code>RLF</code> instructions) ADDWF 1 = A carry occurred 0 = A carry did not occur SUBWF 1 = A borrow did not occur 0 = A borrow occurred RRF or RLF Load bit with LSB or MSB, respectively</p>							

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

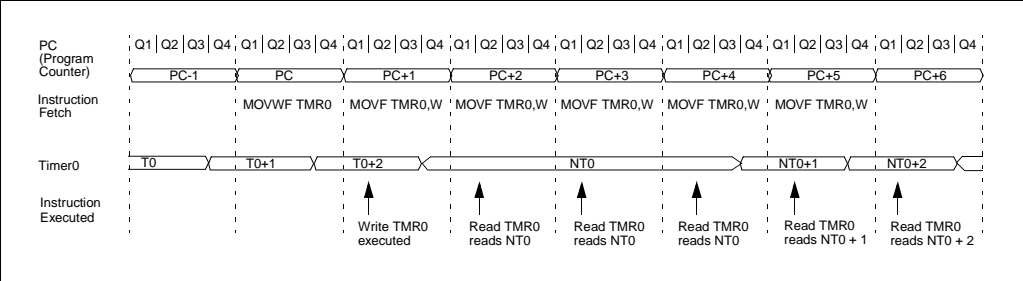


FIGURE 6-3: TIMER0 TIMING: INTERNAL CLOCK/PRESCALE 1:2

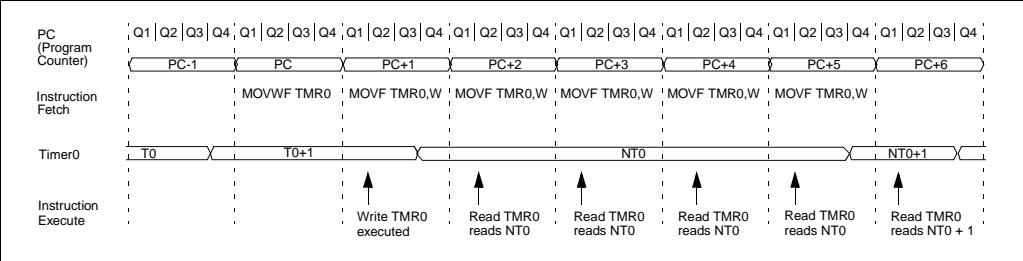


TABLE 6-1: REGISTERS ASSOCIATED WITH TIMER0

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
01h	TMR0	Timer0 - 8-bit real-time clock/counter								xxxx xxxx	uuuu uuuu
N/A	OPTION	GPWU	GPPU	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
N/A	TRIS	—	—	GP5	GP4	GP3	GP2	GP1	GP0	--11 1111	--11 1111

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

Figure 7-1: Block diagram of GPIO6 (SDA line)

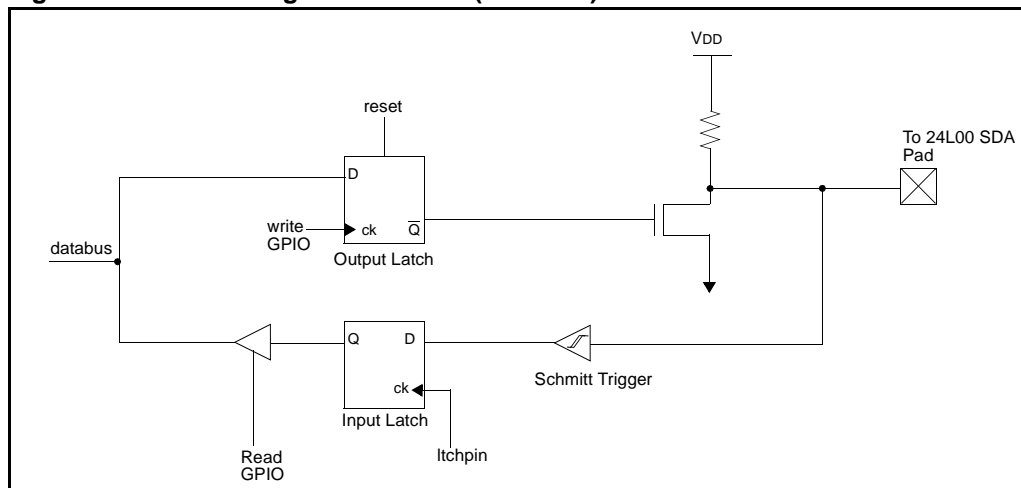
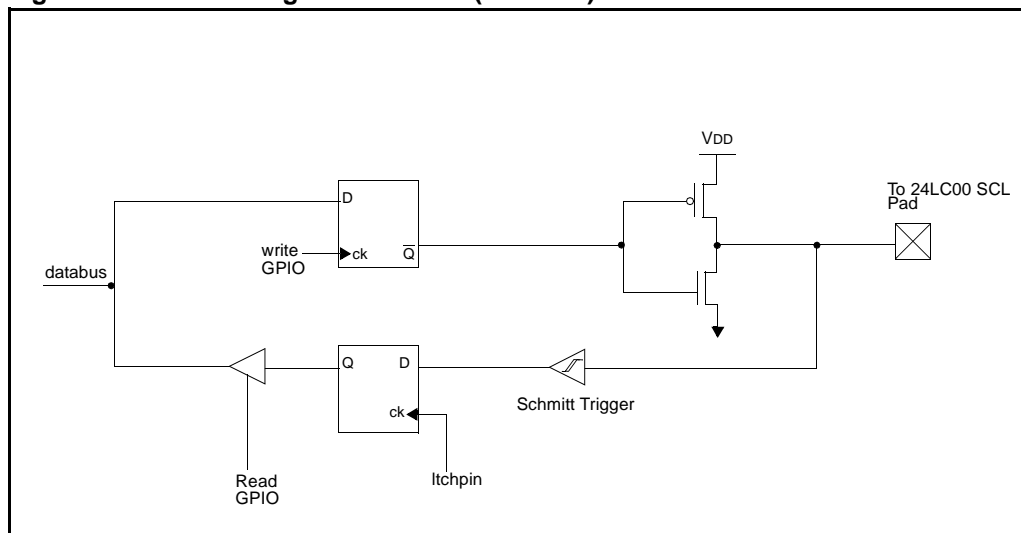


Figure 7-2: Block diagram of GPIO7 (SCL line)



7.5 READ OPERATIONS

Read operations are initiated in the same way as write operations with the exception that the R/W bit of the slave address is set to one. There are three basic types of read operations: current address read, random read, and sequential read.

7.5.1 CURRENT ADDRESS READ

It contains an address counter that maintains the address of the last word accessed, internally incremented by one. Therefore, if the previous read access was to address n , the next current address read operation would access data from address $n + 1$. Upon receipt of the slave address with the R/W bit set to one, the device issues an acknowledge and transmits the eight bit data word. The master will not acknowledge the transfer but does generate a stop condition and the device discontinues transmission (Figure 7-8).

7.5.2 RANDOM READ

Random read operations allow the master to access any memory location in a random manner. To perform this type of read operation, first the word address must be set. This is done by sending the word address to the

device as part of a write operation. After the word address is sent, the master generates a start condition following the acknowledge. This terminates the write operation, but not before the internal address pointer is set. Then the master issues the control byte again but with the R/W bit set to a one. It will then issue an acknowledge and transmits the eight bit data word. The master will not acknowledge the transfer but does generate a stop condition and the device discontinues transmission (Figure 7-9). After this command, the internal address counter will point to the address location following the one that was just read.

7.5.3 SEQUENTIAL READ

Sequential reads are initiated in the same way as a random read except that after the device transmits the first data byte, the master issues an acknowledge as opposed to a stop condition in a random read. This directs the device to transmit the next sequentially addressed 8-bit word (Figure 7-10).

To provide sequential reads, it contains an internal address pointer which is incremented by one at the completion of each read operation. This address pointer allows the entire memory contents to be serially read during one operation.

FIGURE 7-8: CURRENT ADDRESS READ

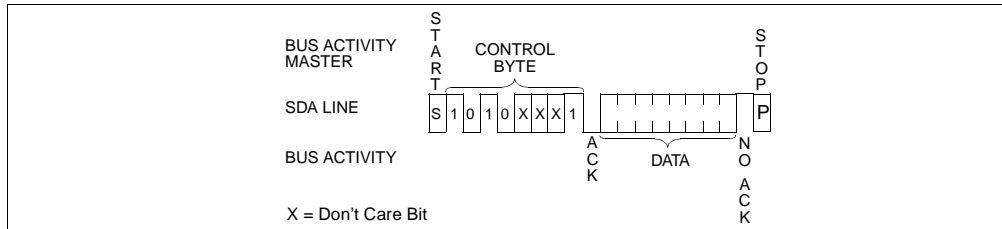


FIGURE 7-9: RANDOM READ

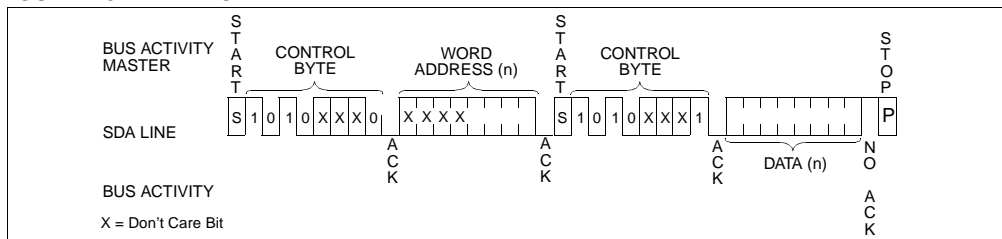
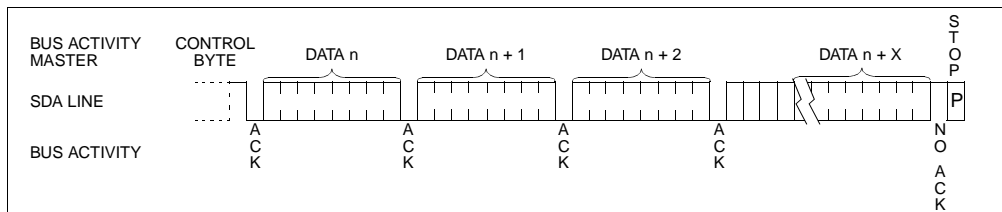


FIGURE 7-10: SEQUENTIAL READ



8.0 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 PIC12C5XX family of microcontrollers 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

The PIC12C5XX 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 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.

8.1 Configuration Bits

The PIC12C5XX configuration word consists of 12 bits. Configuration bits can be programmed to select various device configurations. Two bits are for the selection of the oscillator type, one bit is the Watchdog Timer enable bit, and one bit is the MCLR enable bit.

FIGURE 8-1: CONFIGURATION WORD FOR PIC12C5XX

—	—	—	—	—	—	—	MCLRE	CP	WDTE	FOSC1	FOSC0	Register: CONFIG
bit11	10	9	8	7	6	5	4	3	2	1	bit0	Address ⁽¹⁾ : FFFh
bit 11-5: Unimplemented												
bit 4: MCLRE: MCLR enable bit.												
1 = MCLR pin enabled												
0 = MCLR tied to VDD, (Internally)												
bit 3: CP: Code protection bit.												
1 = Code protection off												
0 = Code protection on												
bit 2: WDTE: Watchdog timer enable bit												
1 = WDT enabled												
0 = WDT disabled												
bit 1-0: FOSC1:FOSC0: Oscillator selection bits												
11 = EXTRC - external RC oscillator												
10 = INTRC - internal RC oscillator												
01 = XT oscillator												
00 = LP oscillator												
Note 1: Refer to the PIC12C5XX Programming Specifications to determine how to access the configuration word. This register is not user addressable during device operation.												

8.9 Power-Down Mode (SLEEP)

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

8.9.1 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{TO} bit (STATUS<4>) is set, the \overline{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{MCLR} pin low.

For lowest current consumption while powered down, the $T0CKI$ input should be at V_{DD} or V_{SS} and the $GP3/\overline{MCLR}/V_{PP}$ pin must be at a logic high level (V_{IHMC}) if \overline{MCLR} is enabled.

8.9.2 WAKE-UP FROM SLEEP

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

1. An external reset input on $GP3/\overline{MCLR}/V_{PP}$ pin, when configured as \overline{MCLR} .
2. A Watchdog Timer time-out reset (if WDT was enabled).
3. A change on input pin $GP0$, $GP1$, or $GP3/\overline{MCLR}/V_{PP}$ when wake-up on change is enabled.

These events cause a device reset. The \overline{TO} , \overline{PD} , and $GPWUF$ bits can be used to determine the cause of device reset. The \overline{TO} bit is cleared if a WDT time-out occurred (and caused wake-up). The \overline{PD} bit, which is set on power-up, is cleared when **SLEEP** is invoked. The $GPWUF$ bit indicates a change in state while in SLEEP at pins $GP0$, $GP1$, or $GP3$ (since the last time there was a file or bit operation on GP 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.

8.10 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 can be read by the PIC12C5XX regardless of the code protection bit setting.

The last memory location cannot be read if code protection is enabled on the PIC12C508/509.

The last memory location can be read regardless of the code protection bit setting on the PIC12C508A/509A/CR509A/CE518/CE519.

8.11 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.

8.12 In-Circuit Serial Programming

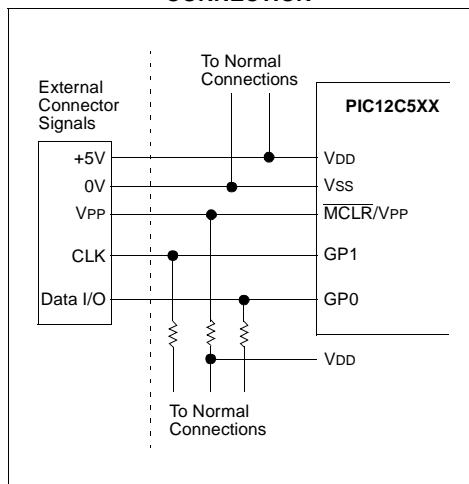
The PIC12C5XX microcontrollers with EPROM program memory can be serially programmed while in the end application circuit. This is simply done with two lines for clock and data, and three other lines for power, ground, and the programming voltage. This allows customers to manufacture boards with unprogrammed devices, and then program the microcontroller just before shipping the product. This also allows the most recent firmware or a custom firmware to be programmed.

The device is placed into a program/verify mode by holding the GP1 and GP0 pins low while raising the MCLR (VPP) pin from V_{IL} to V_{IH} (see programming specification). GP1 becomes the programming clock and GP0 becomes the programming data. Both GP1 and GP0 are Schmitt Trigger inputs in this mode.

After reset, a 6-bit command is then supplied to the device. Depending on the command, 14-bits of program data are then supplied to or from the device, depending if the command was a load or a read. For complete details of serial programming, please refer to the PIC12C5XX Programming Specifications.

A typical in-circuit serial programming connection is shown in Figure 8-16.

FIGURE 8-16: TYPICAL IN-CIRCUIT SERIAL PROGRAMMING CONNECTION



INCF	Increment f			
Syntax:	[<i>label</i>] INCF f,d			
Operands:	$0 \leq f \leq 31$ $d \in [0,1]$			
Operation:	$(f) + 1 \rightarrow (\text{dest})$			
Status Affected:	Z			
Encoding:	<table border="1"><tr><td>0010</td><td>10df</td><td>ffff</td></tr></table>	0010	10df	ffff
0010	10df	ffff		
Description:	The contents of register 'f' are incremented. If 'd' is 0 the result is placed in the W register. If 'd' is 1 the result is placed back in register 'f'.			
Words:	1			
Cycles:	1			
Example:	INCF CNT, 1			

Before Instruction
 CNT = 0xFF
 Z = 0

After Instruction
 CNT = 0x00
 Z = 1

INCFSZ		Increment f, Skip if 0				
Syntax:	[<i>label</i>] INCFSZ f,d					
Operands:	$0 \leq f \leq 31$ $d \in [0,1]$					
Operation:	$(f) + 1 \rightarrow (\text{dest})$, skip if result = 0					
Status Affected:	None					
Encoding:	<table border="1"><tr><td>0011</td><td>11df</td><td>ffff</td></tr></table>			0011	11df	ffff
0011	11df	ffff				
Description:	<p>The contents of register 'f' are incremented. If 'd' is 0 the result is placed in the W register. If 'd' is 1 the result is placed back in register 'f'.</p> <p>If the result is 0, then the next instruction, which is already fetched, is discarded and an NOP is executed instead making it a two cycle instruction.</p>					
Words:	1					
Cycles:	1(2)					
Example:	HERE	INCFSZ	CNT, 1			
		GOTO	LOOP			

Before Instruction
 PC = address (HERE)

After Instruction
 CNT = CNT + 1;
 if CNT = 0,
 PC = address (CONTINUE);
 if CNT \neq 0,
 PC = address (HERE + 1)

IORLW	Inclusive OR literal with W			
Syntax:	[<i>label</i>] IORLW k			
Operands:	0 ≤ k ≤ 255			
Operation:	(W) .OR. (k) → (W)			
Status Affected:	Z			
Encoding:	<table border="1"><tr><td>1101</td><td>kkkk</td><td>kkkk</td></tr></table>	1101	kkkk	kkkk
1101	kkkk	kkkk		
Description:	The contents of the W register are OR'ed with the eight bit literal 'k'. The result is placed in the W register.			
Words:	1			
Cycles:	1			
Example:	IORLW 0x35			
Before Instruction	W = 0x9A			
After Instruction	W = 0xBF Z = 0			

IORWF		Inclusive OR W with f				
Syntax:	[<i>label</i>] IORWF f,d					
Operands:	0 ≤ f ≤ 31 d ∈ [0,1]					
Operation:	(W).OR. (f) → (dest)					
Status Affected:	Z					
Encoding:	<table border="1"><tr><td>0001</td><td>00df</td><td>ffff</td></tr></table>			0001	00df	ffff
0001	00df	ffff				
Description:	Inclusive OR the W register with register 'f'. If 'd' is 0 the result is placed in the W register. If 'd' is 1 the result is placed back in register 'f'.					
Words:	1					
Cycles:	1					
Example:	IORWF RESULT, 0					
Before Instruction						
RESULT = 0x13						
W = 0x91						
After Instruction						
RESULT = 0x13						
W = 0x93						
Z = 0						

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: $k \rightarrow (W)$;
 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

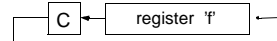
Before Instruction
 W = 0x07
 After Instruction
 W = value of k8

RLF Rotate Left f through Carry

Syntax: [label] RLF f,d
 Operands: $0 \leq f \leq 31$
 $d \in [0,1]$
 Operation: See description below
 Status Affected: C
 Encoding:

0011	01df	ffff
------	------	------

 Description: The contents of register 'f' are rotated one bit to the left through the Carry Flag. If 'd' is 0 the result is placed in the W register. If 'd' is 1 the result is stored back in register 'f'.



Words: 1
 Cycles: 1
 Example: RLF REG1,0

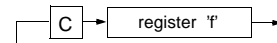
Before Instruction
 REG1 = 1110 0110
 C = 0
 After Instruction
 REG1 = 1110 0110
 W = 1100 1100
 C = 1

RRF Rotate Right f through Carry

Syntax: [label] RRF f,d
 Operands: $0 \leq f \leq 31$
 $d \in [0,1]$
 Operation: See description below
 Status Affected: C
 Encoding:

0011	00df	ffff
------	------	------

 Description: The contents of register 'f' are rotated one bit to the right through the Carry Flag. If 'd' is 0 the result is placed in the W register. If 'd' is 1 the result is placed back in register 'f'.



Words: 1
 Cycles: 1
 Example: RRF REG1,0

Before Instruction
 REG1 = 1110 0110
 C = 0
 After Instruction
 REG1 = 1110 0110
 W = 0111 0011
 C = 0

10.10 MPLAB Integrated Development Environment Software

The MPLAB IDE Software brings an ease of software development previously unseen in the 8-bit microcontroller market. MPLAB is a windows based application which contains:

- A full featured editor
- Three operating modes
 - editor
 - emulator
 - simulator
- A project manager
- Customizable tool bar and key mapping
- A status bar with project information
- Extensive on-line help

MPLAB allows you to:

- Edit your source files (either assembly or 'C')
- One touch assemble (or compile) and download to PICmicro[®] tools (automatically updates all project information)
- Debug using:
 - source files
 - absolute listing file

The ability to use MPLAB with Microchip's simulator allows a consistent platform and the ability to easily switch from the low cost simulator to the full featured emulator with minimal retraining due to development tools.

10.11 Assembler (MPASM)

The MPASM Universal Macro Assembler is a PC-hosted symbolic assembler. It supports all microcontroller series including the PIC12C5XX, PIC14000, PIC16C5X, PIC16CXXX, and PIC17CXX families.

MPASM offers full featured Macro capabilities, conditional assembly, and several source and listing formats. It generates various object code formats to support Microchip's development tools as well as third party programmers.

MPASM allows full symbolic debugging from MPLAB-ICE, Microchip's Universal Emulator System.

MPASM has the following features to assist in developing software for specific use applications.

- Provides translation of Assembler source code to object code for all Microchip microcontrollers.
- Macro assembly capability.
- Produces all the files (Object, Listing, Symbol, and special) required for symbolic debug with Microchip's emulator systems.
- Supports Hex (default), Decimal and Octal source and listing formats.

MPASM provides a rich directive language to support programming of the PICmicro[®]. Directives are helpful in making the development of your assemble source code shorter and more maintainable.

10.12 Software Simulator (MPLAB-SIM)

The MPLAB-SIM Software Simulator allows code development in a PC host environment. It allows the user to simulate the PICmicro[®] series microcontrollers on an instruction level. On any given instruction, the user may examine or modify any of the data areas or provide external stimulus to any of the pins. The input/output radix can be set by the user and the execution can be performed in; single step, execute until break, or in a trace mode.

MPLAB-SIM fully supports symbolic debugging using MPLAB-C17 and MPASM. The Software Simulator offers the low cost flexibility to develop and debug code outside of the laboratory environment making it an excellent multi-project software development tool.

10.13 MPLAB-C17 Compiler

The MPLAB-C17 Code Development System is a complete ANSI 'C' compiler and integrated development environment for Microchip's PIC17CXXX family of microcontrollers. The compiler provides powerful integration capabilities and ease of use not found with other compilers.

For easier source level debugging, the compiler provides symbol information that is compatible with the MPLAB IDE memory display.

10.14 Fuzzy Logic Development System (fuzzyTECH-MP)

fuzzyTECH-MP fuzzy logic development tool is available in two versions - a low cost introductory version, MP Explorer, for designers to gain a comprehensive working knowledge of fuzzy logic system design; and a full-featured version, *fuzzyTECH-MP*, Edition for implementing more complex systems.

Both versions include Microchip's *fuzzyLAB™* demonstration board for hands-on experience with fuzzy logic systems implementation.

10.15 SEEVAL[®] Evaluation and Programming System

The SEEVAL SEEPROM Designer's Kit supports all Microchip 2-wire and 3-wire Serial EEPROMs. The kit includes everything necessary to read, write, erase or program special features of any Microchip SEEPROM product including Smart Serials™ and secure serials. The Total Endurance™ Disk is included to aid in trade-off analysis and reliability calculations. The total kit can significantly reduce time-to-market and result in an optimized system.

NOTES:

TABLE 11-5: RESET, WATCHDOG TIMER, AND DEVICE RESET TIMER - PIC12C508/C509

AC Characteristics Standard Operating Conditions (unless otherwise specified) Operating Temperature $0^{\circ}\text{C} \leq T_A \leq +70^{\circ}\text{C}$ (commercial) $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ (industrial) $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ (extended) Operating Voltage VDD range is described in Section 11.1							
Parameter No.	Sym	Characteristic	Min	Typ ⁽¹⁾	Max	Units	Conditions
30	TmCL	MCLR Pulse Width (low)	2000*	—	—	ns	VDD = 5 V
31	Twdt	Watchdog Timer Time-out Period (No Prescaler)	9*	18*	30*	ms	VDD = 5 V (Commercial)
32	TDRT	Device Reset Timer Period ⁽²⁾	9*	18*	30*	ms	VDD = 5 V (Commercial)
34	TioZ	I/O Hi-impedance from 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.

Note 2: See Table 11-6.

TABLE 11-6: DRT (DEVICE RESET TIMER PERIOD - PIC12C508/C509)

Oscillator Configuration	POR Reset	Subsequent Resets
IntRC & ExtRC	18 ms (typical)	300 μ s (typical)
XT & LP	18 ms (typical)	18 ms (typical)

PIC12C5XX

TABLE 13-1: PULL-UP RESISTOR RANGES* - PIC12C508A, PIC12C509A, PIC12CR509A, PIC12CE518, PIC12CE519, PIC12LC508A, PIC12LC509A, PIC12LCR509A, PIC12LCE518 and PIC12LCE519

V _{DD} (Volts)	Temperature (°C)	Min	Typ	Max	Units
GP0/GP1					
2.5	–40	38K	42K	63K	Ω
	25	42K	48K	63K	Ω
	85	42K	49K	63K	Ω
	125	50K	55K	63K	Ω
5.5	–40	15K	17K	20K	Ω
	25	18K	20K	23K	Ω
	85	19K	22K	25K	Ω
	125	22K	24K	28K	Ω
GP3					
2.5	–40	285K	346K	417K	Ω
	25	343K	414K	532K	Ω
	85	368K	457K	532K	Ω
	125	431K	504K	593K	Ω
5.5	–40	247K	292K	360K	Ω
	25	288K	341K	437K	Ω
	85	306K	371K	448K	Ω
	125	351K	407K	500K	Ω

* These parameters are characterized but not tested.

TABLE 13-3: CALIBRATED INTERNAL RC FREQUENCIES - PIC12C508A, PIC12C509A, PIC12CE518, PIC12CE519, PIC12LC508A, PIC12LC509A, PIC12LCR509A, PIC12LCE518 and PIC12LCE519

AC Characteristics		Standard Operating Conditions (unless otherwise specified)					
		Operating Temperature $0^{\circ}\text{C} \leq T_A \leq +70^{\circ}\text{C}$ (commercial), $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ (industrial), $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ (extended)					
		Operating Voltage VDD range is described in Section 10.1					
Parameter No.	Sym	Characteristic	Min*	Typ ⁽¹⁾	Max*	Units	Conditions
		Internal Calibrated RC Frequency	3.65	4.00	4.28	MHz	VDD = 5.0V
		Internal Calibrated RC Frequency	3.55	—	4.31	MHz	VDD = 2.5V

* 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.

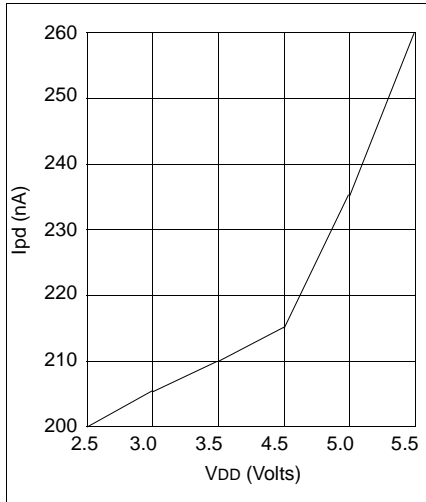
TABLE 13-8: EEPROM MEMORY BUS TIMING REQUIREMENTS - PIC12CE5XX ONLY.

AC Characteristics	Standard Operating Conditions (unless otherwise specified)				
	Operating Temperature $0^{\circ}\text{C} \leq T_A \leq +70^{\circ}\text{C}$, $V_{CC} = 3.0\text{V to } 5.5\text{V}$ (commercial) $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$, $V_{CC} = 3.0\text{V to } 5.5\text{V}$ (industrial) $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$, $V_{CC} = 4.5\text{V to } 5.5\text{V}$ (extended) Operating Voltage V_{DD} range is described in Section 13.1				
Parameter	Symbol	Min	Max	Units	Conditions
Clock frequency	FCLK	—	100	kHz	$4.5\text{V} \leq V_{CC} \leq 5.5\text{V}$ (E Temp range)
		—	100		$3.0\text{V} \leq V_{CC} \leq 4.5\text{V}$
		—	400		$4.5\text{V} \leq V_{CC} \leq 5.5\text{V}$
Clock high time	T _{HIGH}	4000	—	ns	$4.5\text{V} \leq V_{CC} \leq 5.5\text{V}$ (E Temp range)
		4000	—		$3.0\text{V} \leq V_{CC} \leq 4.5\text{V}$
		600	—		$4.5\text{V} \leq V_{CC} \leq 5.5\text{V}$
Clock low time	T _{LOW}	4700	—	ns	$4.5\text{V} \leq V_{CC} \leq 5.5\text{V}$ (E Temp range)
		4700	—		$3.0\text{V} \leq V_{CC} \leq 4.5\text{V}$
		1300	—		$4.5\text{V} \leq V_{CC} \leq 5.5\text{V}$
SDA and SCL rise time (Note 1)	T _R	—	1000	ns	$4.5\text{V} \leq V_{CC} \leq 5.5\text{V}$ (E Temp range)
		—	1000		$3.0\text{V} \leq V_{CC} \leq 4.5\text{V}$
		—	300		$4.5\text{V} \leq V_{CC} \leq 5.5\text{V}$
SDA and SCL fall time	T _F	—	300	ns	(Note 1)
START condition hold time	T _{HD:STA}	4000	—	ns	$4.5\text{V} \leq V_{CC} \leq 5.5\text{V}$ (E Temp range)
		4000	—		$3.0\text{V} \leq V_{CC} \leq 4.5\text{V}$
		600	—		$4.5\text{V} \leq V_{CC} \leq 5.5\text{V}$
START condition setup time	T _{SU:STA}	4700	—	ns	$4.5\text{V} \leq V_{CC} \leq 5.5\text{V}$ (E Temp range)
		4700	—		$3.0\text{V} \leq V_{CC} \leq 4.5\text{V}$
		600	—		$4.5\text{V} \leq V_{CC} \leq 5.5\text{V}$
Data input hold time	T _{HD:DAT}	0	—	ns	(Note 2)
Data input setup time	T _{SU:DAT}	250	—	ns	$4.5\text{V} \leq V_{CC} \leq 5.5\text{V}$ (E Temp range)
		250	—		$3.0\text{V} \leq V_{CC} \leq 4.5\text{V}$
		100	—		$4.5\text{V} \leq V_{CC} \leq 5.5\text{V}$
STOP condition setup time	T _{SU:STO}	4000	—	ns	$4.5\text{V} \leq V_{CC} \leq 5.5\text{V}$ (E Temp range)
		4000	—		$3.0\text{V} \leq V_{CC} \leq 4.5\text{V}$
		600	—		$4.5\text{V} \leq V_{CC} \leq 5.5\text{V}$
Output valid from clock (Note 2)	T _{AA}	—	3500	ns	$4.5\text{V} \leq V_{CC} \leq 5.5\text{V}$ (E Temp range)
		—	3500		$3.0\text{V} \leq V_{CC} \leq 4.5\text{V}$
		—	900		$4.5\text{V} \leq V_{CC} \leq 5.5\text{V}$
Bus free time: Time the bus must be free before a new transmis- sion can start	T _{BUF}	4700	—	ns	$4.5\text{V} \leq V_{CC} \leq 5.5\text{V}$ (E Temp range)
		4700	—		$3.0\text{V} \leq V_{CC} \leq 4.5\text{V}$
		1300	—		$4.5\text{V} \leq V_{CC} \leq 5.5\text{V}$
Output fall time from V _{IH} minimum to V _{IL} maximum	T _{oF}	20+0.1 CB	250	ns	(Note 1), CB ≤ 100 pF
Input filter spike suppression (SDA and SCL pins)	T _{SP}	—	50	ns	(Notes 1, 3)
Write cycle time	T _{WC}	—	4	ms	
Endurance		1M	—	cycles	25°C, V _{CC} = 5.0V, Block Mode (Note 4)

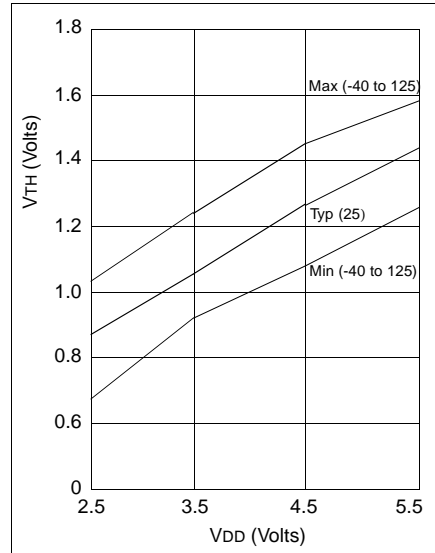
Note 1: Not 100% tested. CB = total capacitance of one bus line in pF.

- 2:** As a transmitter, the device must provide an internal minimum delay time to bridge the undefined region (minimum 300 ns) of the falling edge of SCL to avoid unintended generation of START or STOP conditions.
- 3:** The combined T_{SP} and V_{HYS} specifications are due to new Schmitt trigger inputs which provide improved noise spike suppression. This eliminates the need for a TI specification for standard operation.
- 4:** This parameter is not tested but guaranteed by characterization. For endurance estimates in a specific application, please consult the Total Endurance Model which can be obtained on Microchip's website.

**FIGURE 14-13: TYPICAL IPD VS. VDD,
WATCHDOG DISABLED (25°C)**



**FIGURE 14-14: VTH (INPUT THRESHOLD
VOLTAGE) OF GPIO PINS
VS. VDD**



PIC12C5XX

NOTES: