

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 "<u>Embedded -</u> <u>Microcontrollers</u>"

Details

-XF

Product Status	Active
Core Processor	PIC
Core Size	8-Bit
Speed	4MHz
Connectivity	-
Peripherals	POR, WDT
Number of I/O	5
Program Memory Size	1.5KB (1K x 12)
Program Memory Type	OTP
EEPROM Size	-
RAM Size	41 × 8
Voltage - Supply (Vcc/Vdd)	2.5V ~ 5.5V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	8-SOIC (0.154", 3.90mm Width)
Supplier Device Package	8-SOIC
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic12lc509a-04i-sn

Email: info@E-XFL.COM

Address: Room A, 16/F, Full Win Commercial Centre, 573 Nathan Road, Mongkok, Hong Kong

1.0 GENERAL DESCRIPTION

The PIC12C5XX from Microchip Technology is a family of low-cost, high performance, 8-bit, fully static, EEPROM/EPROM/ROM-based CMOS microcontrollers. It employs a RISC architecture with only 33 single word/single cycle instructions. All instructions are single cycle (1 μ s) except for program branches which take two cycles. The PIC12C5XX delivers performance an order of magnitude higher than its competitors in the same price category. The 12-bit wide instructions are highly symmetrical resulting in 2:1 code compression over other 8-bit microcontrollers in its class. The easy to use and easy to remember instruction set reduces development time significantly.

The PIC12C5XX products are equipped with special features that reduce system cost and power requirements. The Power-On Reset (POR) and Device Reset Timer (DRT) eliminate the need for external reset circuitry. There are four oscillator configurations to choose from, including INTRC internal oscillator mode and the power-saving LP (Low Power) oscillator mode. Power saving SLEEP mode, Watchdog Timer and code protection features also improve system cost, power and reliability.

The PIC12C5XX are available in the cost-effective One-Time-Programmable (OTP) versions which are suitable for production in any volume. The customer can take full advantage of Microchip's price leadership in OTP microcontrollers while benefiting from the OTP's flexibility.

The PIC12C5XX products are supported by a full-featured macro assembler, a software simulator, an in-circuit emulator, a 'C' compiler, fuzzy logic support tools, a low-cost development programmer, and a full featured programmer. All the tools are supported on IBM[®] PC and compatible machines.

1.1 Applications

The PIC12C5XX series fits perfectly in applications ranging from personal care appliances and security systems to low-power remote transmitters/receivers. The EPROM technology makes customizing application programs (transmitter codes, appliance settings, receiver frequencies. etc.) extremely fast and convenient, while the EEPROM data memory technology allows for the changing of calibration factors and security codes. The small footprint packages, for through hole or surface mounting, make this microcontroller series perfect for applications with space limitations. Low-cost, low-power, high performance, ease of use and I/O flexibility make the PIC12C5XX series very versatile even in areas where no microcontroller use has been considered before (e.g., timer functions, replacement of "glue" logic and PLD's in larger systems, coprocessor applications).

Name	DIP Pin #	SOIC Pin #	I/O/P Type	Buffer Type	Description
GP0	7	7	I/O	TTL/ST	Bi-directional I/O port/ serial programming data. Can be software programmed for internal weak pull-up and wake-up from SLEEP on pin change. This buffer is a Schmitt Trigger input when used in serial programming mode.
GP1	6	6	I/O	TTL/ST	Bi-directional I/O port/ serial programming clock. Can be software programmed for internal weak pull-up and wake-up from SLEEP on pin change. This buffer is a Schmitt Trigger input when used in serial programming mode.
GP2/T0CKI	5	5	I/O	ST	Bi-directional I/O port. Can be configured as T0CKI.
GP3/MCLR/Vpp	4	4	1	TTL/ST	Input port/master clear (reset) input/programming volt- age input. When configured as MCLR, this pin is an active low reset to the device. Voltage on MCLR/VPP must not exceed VDD during normal device operation or the device will enter programming mode. Can be software programmed for internal weak pull-up and wake-up from SLEEP on pin change. Weak pull-up always on if configured as MCLR. ST when in MCLR mode.
GP4/OSC2	3	3	I/O	TTL	Bi-directional I/O port/oscillator crystal output. Con- nections to crystal or resonator in crystal oscillator mode (XT and LP modes only, GPIO in other modes).
GP5/OSC1/CLKIN	2	2	I/O	TTL/ST	Bidirectional IO port/oscillator crystal input/external clock source input (GPIO in Internal RC mode only, OSC1 in all other oscillator modes). TTL input when GPIO, ST input in external RC oscillator mode.
VDD	1	1	Р	_	Positive supply for logic and I/O pins
Vss	8	8	Р		Ground reference for logic and I/O pins

TABLE 3-1:	PIC12C5XX	PINOUT	DESCRIPTION

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

4.5 <u>OSCCAL Register</u>

The Oscillator Calibration (OSCCAL) register is used to calibrate the internal 4 MHz oscillator. It contains four to six bits for calibration. Increasing the cal value increases the frequency. See Section 7.2.5 for more information on the internal oscillator.

FIGURE 4-6: OSCCAL REGISTER (ADDRESS 05h) FOR PIC12C508 AND PIC12C509

R/W-0	R/W-1	R/W-1	R/W-1	R/W-0	R/W-0	U-0	U-0	
CAL3	CAL2	CAL1	CAL0	-		—	—	R = Readable bit
bit7							bit0	 W = Writable bit U = Unimplemented bit, read as '0' n = Value at POR reset
bit 7-4:	CAL<3:0:	Calibrat	tion					
bit 3-0:	Unimpler	nented: F	Read as '0'					

FIGURE 4-7: OSCCAL REGISTER (ADDRESS 05h) FOR PIC12C508A/C509A/CR509A/12CE518/ 12CE519

CAL5 CAL4 CAL3 CAL2 CAL1 CAL0 — — R = Readable bit bit7 bit0 bit0 U = Writable bit U = Unimplemented bit, read en (10)	<u>.1 CAL0 — </u>	R = Rea	CAL0	CAL1	041.0			
bit7 bit0 W = Writable bit U = Unimplemented bit, trad ec '0'	bit0 W = Writable bit	bit0 W = Writ		-	CALZ	CAL3	CAL4	CAL5
- n = Value at POR reset	read as '0' - n = Value at POR rese	U = Unir read - n = Valu				•		bit7
bit 7-2: CAL<5:0>: Calibration					tion	>: Calibra	CAL<5:0:	bit 7-2:
bit 1-0: Unimplemented: Read as '0'					Read as '0	nented: F	Unimpler	bit 1-0:

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	TRIS	—	Ι							11 1111	11 1111
N/A	OPTION	GPWU	GPPU	TOCS	TOSE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
03H	STATUS	GPWUF		PAO	TO	PD	Z	DC	С	0001 1xxx	q00q quuu ⁽¹⁾
06h	GPIO (PIC12C508/ PIC12C509/ PIC12C508A/ PIC12C509A/ PIC12CR509A)	_		GP5	GP4	GP3	GP2	GP1	GP0	xx xxxx	uu uuuu
06h	GPIO (PIC12CE518/ PIC12CE519)	SCL	SDA	GP5	GP4	GP3	GP2	GP1	GP0	11xx xxxx	11uu uuuu

Legend: Shaded cells not used by Port Registers, read as '0', — = unimplemented, read as '0', x = unknown, u = unchanged, g = see tables in Section 8.7 for possible values.

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

5.4 I/O Programming Considerations

5.4.1 BI-DIRECTIONAL I/O PORTS

Some instructions operate internally as read followed by write operations. The BCF and BSF instructions, for example, read the entire port into the CPU, execute the bit operation and re-write the result. Caution must be used when these instructions are applied to a port where one or more pins are used as input/outputs. For example, a BSF operation on bit5 of GPIO will cause all eight bits of GPIO to be read into the CPU, bit5 to be set and the GPIO value to be written to the output latches. If another bit of GPIO is used as a bidirectional I/O pin (say bit0) and it is defined as an input at this time, the input signal present on the pin itself would be read into the CPU and rewritten to the data latch of this particular pin, overwriting the previous content. As long as the pin stays in the input mode, no problem occurs. However, if bit0 is switched into output mode later on, the content of the data latch may now be unknown.

Example 5-1 shows the effect of two sequential read-modify-write instructions (e.g., ${\tt BCF}$, ${\tt BSF}$, etc.) on an I/O port.

A pin actively outputting a high or a low should not be driven from external devices at the same time in order to change the level on this pin ("wired-or", "wiredand"). The resulting high output currents may damage the chip.

EXAMPLE 5-1: READ-MODIFY-WRITE INSTRUCTIONS ON AN I/O PORT

; 1 ; ;	Initial GPIO<5 GPIO<2	L GPIO S 5:3> Inp 2:0> Out	Settings puts puts		
;					
;			GPI) latch	GPIO pins
;					
	BCF	GPIO, 5	5 ;01	-ppp	11 pppp
	BCF	GPIO, 4	i ;10	-ppp	11 pppp
	MOVLW	007h	;		
	TRIS	GPIO	;10	-ppp	11 pppp

;Note that the user may have expected the pin ;values to be --00 pppp. The 2nd BCF caused ;GP5 to be latched as the pin value (High).

5.4.2 SUCCESSIVE OPERATIONS ON I/O PORTS

The actual write to an I/O port happens at the end of an instruction cycle, whereas for reading, the data must be valid at the beginning of the instruction cycle (Figure 5-2). Therefore, care must be exercised if a write followed by a read operation is carried out on the same I/O port. The sequence of instructions should allow the pin voltage to stabilize (load dependent) before the next instruction, which causes that file to be read into the CPU, is executed. Otherwise, the previous state of that pin may be read into the CPU rather than the new state. When in doubt, it is better to separate these instructions with a NOP or another instruction not accessing this I/O port.

7.0 EEPROM PERIPHERAL OPERATION

This section applies to PIC12CE518 and PIC12CE519 only.

The PIC12CE518 and PIC12CE519 each have 16 bytes of EEPROM data memory. The EEPROM memory has an endurance of 1,000,000 erase/write cycles and a data retention of greater than 40 years. The EEPROM data memory supports a bi-directional 2-wire bus and data transmission protocol. These two-wires are serial data (SDA) and serial clock (SCL), that are mapped to bit6 and bit7, respectively, of the GPIO register (SFR 06h). Unlike the GP0-GP5 that are connected to the internal EEPROM peripheral. For most applications, all that is required is calls to the following functions:

; Byte_Write: Byte write routine Inputs: EEPROM Address EEADDR : ; EEPROM Data EEDATA Outputs: Return 01 in W if OK, else ; return 00 in W ; ; Read_Current: Read EEPROM at address currently held by EE device. Inputs: NONE ; Outputs: EEPROM Data EEDATA ; Return 01 in W if OK, else ; return 00 in W ; ; Read_Random: Read EEPROM byte at supplied address Inputs: EEPROM Address : FFADDR ; Outputs: EEPROM Data EEDATA Return 01 in W if OK, ; else return 00 in W

The code for these functions is available on our website www.microchip.com. The code will be accessed by either including the source code FL51XINC.ASM or by linking FLASH5IX.ASM.

It is very important to check the return codes when using these calls, and retry the operation if unsuccessful. Unsuccessful return codes occur when the EE data memory is busy with the previous write, which can take up to 4 mS.

7.0.1 SERIAL DATA

SDA is a bi-directional pin used to transfer addresses and data into and data out of the device.

For normal data transfer SDA is allowed to change only during SCL low. Changes during SCL high are reserved for indicating the START and STOP conditions.

The EEPROM interface is a 2-wire bus protocol consisting of data (SDA) and a clock (SCL). Although these lines are mapped into the GPIO register, they are not accessible as external pins; only to the internal EEPROM peripheral. SDA and SCL operation is also slightly different than GPO-GP5 as listed below. Namely, to avoid code overhead in modifying the TRIS register, both SDA and SCL are always outputs. To read data from the EEPROM peripheral requires outputting a '1' on SDA placing it in high-Z state, where only the internal 100K pull-up is active on the SDA line.

SDA:

Built-in 100K (typical) pull-up to VDD Open-drain (pull-down only) Always an output Outputs a '1' on reset

SCL: Full CMOS output Always an output Outputs a '1' on reset

The following example requires:

- · Code Space: 77 words
- RAM Space: 5 bytes (4 are overlayable)
- Stack Levels:1 (The call to the function itself. The functions do not call any lower level functions.)
- Timing:
 - WRITE_BYTE takes 328 cycles
 - READ_CURRENT takes 212 cycles
 - READ_RANDOM takes 416 cycles.
- IO Pins: 0 (No external IO pins are used)

This code must reside in the lower half of a page. The code achieves it's small size without additional calls through the use of a sequencing table. The table is a list of procedures that must be called in order. The table uses an ADDWF PCL,F instruction, effectively a computed goto, to sequence to the next procedure. However the ADDWF PCL,F instruction yields an 8 bit address, forcing the code to reside in the first 256 addresses of a page.

^{© 1999} Microchip Technology Inc.

FIGURE 7-3: DATA TRANSFER SEQUENCE ON THE SERIAL BUS





7.2 Device Addressing

After generating a START condition, the bus master transmits a control byte consisting of a slave address and a Read/Write bit that indicates what type of operation is to be performed. The slave address consists of a 4-bit device code (1010) followed by three don't care bits.

The last bit of the control byte determines the operation to be performed. When set to a one a read operation is selected, and when set to a zero a write operation is selected. (Figure 7-5). The bus is monitored for its corresponding slave address all the time. It generates an acknowledge bit if the slave address was true and it is not in a programming mode.

FIGURE 7-5: CONTROL BYTE FORMAT



8.2.5 INTERNAL 4 MHz RC OSCILLATOR

The internal RC oscillator provides a fixed 4 MHz (nominal) system clock at VDD = 5V and 25°C, see "Electrical Specifications" section for information on variation over voltage and temperature.

In addition, a calibration instruction is programmed into the top of memory which contains the calibration value for the internal RC oscillator. This location is never code 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: Please note that 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 PIC12C508A, PIC12C509A, PIC12CE518, PIC12CE519, and PIC12CR509A, bits <7:2>, CAL5-CAL0 are used for calibration. Adjusting CAL5-0 from 000000 to 111111 yields a higher clock speed. Note that bits 1 and 0 of OSCCAL are unimplemented and should be written as 0 when modifying OSCCAL for compatibility with future devices.

For the PIC12C508 and PIC12C509, the upper 4 bits of the register are used. Writing a larger value in this location yields a higher clock speed.

8.3 <u>RESET</u>

The device differentiates between various kinds of reset:

- a) Power on reset (POR)
- b) MCLR reset during normal operation
- c) MCLR reset during SLEEP
- d) WDT time-out reset during normal operation
- e) WDT time-out reset during SLEEP
- f) Wake-up from SLEEP on pin change

Some registers are not reset in any way; they are unknown on POR and unchanged in any other reset. Most other registers are reset to "reset state" on poweron reset (POR), \overline{MCLR} , WDT or wake-up on pin change reset during normal operation. They are not affected by a WDT reset during SLEEP or \overline{MCLR} reset during SLEEP, since these resets are viewed as resumption of normal operation. The exceptions to this are \overline{TO} , \overline{PD} , and GPWUF bits. They are set or cleared differently in different reset situations. These bits are used in software to determine the nature of reset. See Table 8-3 for a full description of reset states of all registers.

FIGURE 8-11: TIME-OUT SEQUENCE ON POWER-UP (MCLR TIED TO VDD): SLOW VDD RISE TIME



8.5 Device Reset Timer (DRT)

In the PIC12C5XX, DRT runs from RESET and varies based on oscillator selection (see Table 8-5.)

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

Oscillator circuits based on crystals or ceramic resonators require a certain time after power-up to establish a stable oscillation. The on-chip DRT keeps the device in a RESET condition for approximately 18 ms after MCLR has reached a logic high (VIHMCLR) level. Thus, programming GP3/MCLR/VPP as MCLR and using an external RC network connected to the MCLR input is not required in most cases, allowing for savings in cost-sensitive and/or space restricted applications, as well as allowing the use of the GP3/MCLR/VPP pin as a general purpose input.

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

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

8.6 Watchdog Timer (WDT)

The Watchdog Timer (WDT) is a free running on-chip RC oscillator which does not require any external components. This RC oscillator is separate from the external RC oscillator of the GP5/OSC1/CLKIN pin and the internal 4 MHz oscillator. That means that the WDT will run even if the main processor clock has been stopped, for example, by execution of a SLEEP instruction. During normal operation or SLEEP, a WDT reset or wake-up reset generates a device RESET.

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

The WDT can be permanently disabled by programming the configuration bit WDTE as a '0' (Section 8.1). Refer to the PIC12C5XX Programming Specifications to determine how to access the configuration word.

TABLE 8-5: DRT (DEVICE RESET TIMER PERIOD)

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

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 VIL to VIHH (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



BSF	Bit Set f	BTFSS	Bit Test f, Skip if Set
Syntax:	[<i>label</i>] BSF f,b	Syntax:	[label] BTFSS f,b
Operands:	$0 \le f \le 31$ $0 \le b \le 7$	Operands:	0 ≤ f ≤ 31 0 ≤ b < 7
Operation:	$1 \rightarrow (f < b >)$	Operation:	skip if (f) = 1
Status Affected:	None	Status Affected:	None
Encoding:	0101 bbbf fff	Encoding:	0111 bbbf ffff
Description:	Bit 'b' in register 'f' is set.	Description:	If bit 'b' in register 'f' is '1' then the next
Words:	1		instruction is skipped.
Cycles:	1		fetched during the current instruction
Example:	BSF FLAG_REG, 7		execution, is discarded and an NOP is
Before Instru	uction		executed instead, making this a 2 cycle instruction.
$FLAG_REG = 0x0A$		Words:	1
After Instruction FLAG_REG = 0x8A		Cycles:	1(2)
		Example:	HERE BTFSS FLAG,1
			FALSE GOTO PROCESS_CODE
BTFSC	Bit Test f, Skip if Clear		TRUE •
Syntax:	[label] BTFSC f,b		•
Operands:	$0 \le f \le 31$	Before Instru	uction
	$0 \le b \le 7$	PC	= address (HERE)
Operation:	skip if $(f < b >) = 0$	After Instruc	tion
Status Affected:	None	If FLAG< PC	<1> = 0, = address (FALSE):
Encoding:	0110 bbbf ffff	if FLAG<	<1> = 1,
Description:	If bit 'b' in register 'f' is 0 then the next instruction is skipped.	PC	= address (TRUE)
	If bit 'b' is 0 then the next instruction fetched during the current instruction execution is discarded, and an NOP is		

executed instead, making this a 2 cycle

BTFSC FLAG,1

address (HERE)

address (TRUE);

address(FALSE)

PROCESS_CODE

GOTO

٠ •

0, =

1, =

instruction.

1

1(2)

HERE

TRUE

Before Instruction PC

After Instruction if FLAG<1>

if FLAG<1>

PC

PC

FALSE

=

=

=

Words:

Cycles:

Example:

SLEEP	Enter SL	EEP Mo	de		S		
Syntax:	[<i>label</i>]	SLEEP			S		
Operands:	None				C		
Operation:	$\begin{array}{l} 00h \rightarrow W \\ 0 \rightarrow WD \\ 1 \rightarrow \overline{TO}; \\ 0 \rightarrow \overline{PD} \end{array}$	VDT; T prescal	er;		C S		
Status Affected:	TO, PD,	GPWUF			с г		
Encoding:	0000	0000	0011	Ī	L		
Description:	Time-out status bit $\overline{(TO)}$ is set. The power down status bit $\overline{(PD)}$ is cleared.						
	GPWUF i	s unaffecte	ed.		V		
	The WDT cleared.	and its pre	escaler are	9	C		
	The processor is put into SLEEP mode with the oscillator stopped. See sec- tion on SLEEP for more details.						
Words:	1						
Cycles:	1						
Example:	SLEEP						

SUBWF	Su	btrac	t W from	f	
Syntax:	[lai	bel]	SUBWF	f,d	
Operands:	0 ≤ d ∈	≦f≤3 [0,1]	1]		
Operation:	(f)	– (W)	\rightarrow (dest)		
Status Affected:	C,	DC, Z	2		
Encoding:	0	000	10df	ffff	
Description:	Sul W r res 1 th	otract registe ult is s ne res	(2's comple er from regis stored in the ult is stored	ement meth ster 'f'. If 'd e W registe I back in re	hod) the ' is 0 the r. If 'd' is gister 'f
Words:	1				
Cycles:	1				
Example 1:	SUI	BWF	REG1, 1		
Before Instr	uctio	n			
REG1	=	3			
W	=	2			
Aftor Instruc	= tion	ł			
REG1	=	1			
W	=	2			
С	=	1	; result is	positive	
Example 2:					
Before Instr	uctio	n			
REG1	=	2			
W	=	2			
	=	ſ			
Arter Instruc		0			
W	_	2			
С	=	1	; result is	zero	
Example 3:					
Before Instr	uctio	n			
REG1	=	1			
W	=	2			
	=	?			
After Instruc	tion	EE			
W	=	2			
C	=	0	; result is	negative	

10.6 <u>SIMICE Entry-Level Hardware</u> <u>Simulator</u>

SIMICE is an entry-level hardware development system designed to operate in a PC-based environment with Microchip's simulator MPLAB[™]-SIM. Both SIM-ICE and MPLAB-SIM run under Microchip Technology's MPLAB Integrated Development Environment (IDE) software. Specifically, SIMICE provides hardware simulation for Microchip's PIC12C5XX, PIC12CE5XX, and PIC16C5X families of PICmicro® 8-bit microcontrollers. SIMICE works in conjunction with MPLAB-SIM to provide non-real-time I/O port emulation. SIMICE enables a developer to run simulator code for driving the target system. In addition, the target system can provide input to the simulator code. This capability allows for simple and interactive debugging without having to manually generate MPLAB-SIM stimulus files. SIMICE is a valuable debugging tool for entrylevel system development.

10.7 <u>PICDEM-1 Low-Cost PICmicro®</u> <u>Demonstration Board</u>

The PICDEM-1 is a simple board which demonstrates the capabilities of several of Microchip's microcontrollers. The microcontrollers supported are: PIC16C5X (PIC16C54 to PIC16C58A), PIC16C61, PIC16C62X, PIC16C71, PIC16C8X, PIC17C42, PIC17C43 and PIC17C44. All necessary hardware and software is included to run basic demo programs. The users can program the sample microcontrollers provided with the PICDEM-1 board, on a PRO MATE II or PICSTART-Plus programmer, and easily test firmware. The user can also connect the PICDEM-1 board to the MPLAB-ICE emulator and download the firmware to the emulator for testing. Additional prototype area is available for the user to build some additional hardware and connect it to the microcontroller socket(s). Some of the features include an RS-232 interface, a potentiometer for simulated analog input, push-button switches and eight LEDs connected to PORTB.

10.8 PICDEM-2 Low-Cost PIC16CXX Demonstration Board

The PICDEM-2 is a simple demonstration board that supports the PIC16C62, PIC16C64, PIC16C65, PIC16C73 and PIC16C74 microcontrollers. All the necessary hardware and software is included to run the basic demonstration programs. The user can program the sample microcontrollers provided with the PICDEM-2 board, on a PRO MATE II programmer or PICSTART-Plus, and easily test firmware. The MPLAB-ICE emulator may also be used with the PICDEM-2 board to test firmware. Additional prototype area has been provided to the user for adding additional hardware and connecting it to the microcontroller socket(s). Some of the features include a RS-232 interface, push-button switches, a potentiometer for simulated analog input, a Serial EEPROM to demonstrate usage of the I²C bus and separate headers for connection to an LCD module and a keypad.

10.9 PICDEM-3 Low-Cost PIC16CXXX Demonstration Board

The PICDEM-3 is a simple demonstration board that supports the PIC16C923 and PIC16C924 in the PLCC package. It will also support future 44-pin PLCC microcontrollers with a LCD Module. All the necessary hardware and software is included to run the basic demonstration programs. The user can program the sample microcontrollers provided with the PICDEM-3 board, on a PRO MATE II programmer or PICSTART Plus with an adapter socket, and easily test firmware. The MPLAB-ICE emulator may also be used with the PICDEM-3 board to test firmware. Additional prototype area has been provided to the user for adding hardware and connecting it to the microcontroller socket(s). Some of the features include an RS-232 interface, push-button switches, a potentiometer for simulated analog input, a thermistor and separate headers for connection to an external LCD module and a keypad. Also provided on the PICDEM-3 board is an LCD panel, with 4 commons and 12 seqments, that is capable of displaying time, temperature and day of the week. The PICDEM-3 provides an additional RS-232 interface and Windows 3.1 software for showing the demultiplexed LCD signals on a PC. A simple serial interface allows the user to construct a hardware demultiplexer for the LCD signals.

10.10 <u>MPLAB Integrated Development</u> <u>Environment Software</u>

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 PChosted 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 <u>Fuzzy Logic Development System</u> (fuzzyTECH-MP)

*fuzzy*TECH-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, *fuzzy*TECH-MP, Edition for implementing more complex systems.

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

10.15 <u>SEEVAL® Evaluation and</u> <u>Programming System</u>

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.

VDD (Volts)	Temperature (°C)	Min	Тур	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			

TABLE 11-1: PULL-UP RESISTOR RANGES - PIC12C508/C509

* These parameters are characterized but not tested.

11.4 Timing Diagrams and Specifications





AC Characteristics		$\begin{array}{ll} \mbox{Standard Operating Conditions (unless otherwise specified)} \\ \mbox{Operating Temperature} & 0^{\circ}C \leq TA \leq +70^{\circ}C \mbox{ (commercial)}, \\ -40^{\circ}C \leq TA \leq +85^{\circ}C \mbox{ (industrial)}, \\ -40^{\circ}C \leq TA \leq +125^{\circ}C \mbox{ (extended)} \\ \mbox{Operating Voltage VDD range is described in Section 11.1} \end{array}$						
Parameter No.	Sym	Characteristic	Min	Тур ⁽¹⁾	Мах	Units	Conditions	
	Fosc	External CLKIN Frequency ⁽²⁾						
			DC	—	4	MHz	XT osc mode	
			DC	—	200	kHz	LP osc mode	
		Oscillator Frequency ⁽²⁾						
			0.1	—	4	MHz	XT osc mode	
			DC	—	200	kHz	LP osc mode	
1	Tosc	External CLKIN Period ⁽²⁾	250	—	—	ns	EXTRC osc mode	
			250	—	—	ns	XT osc mode	
			5	—	—	ms	LP osc mode	
		Oscillator Period ⁽²⁾	250	—	—	ns	EXTRC osc mode	
			250	—	10,000	ns	XT osc mode	
			5	—	—	ms	LP osc mode	
2	Тсу	Instruction Cycle Time ⁽³⁾	—	4/Fosc	—	_		
3	TosL, TosH	Clock in (OSC1) Low or High Time	50*	—	—	ns	XT oscillator	
			2*	—	—	ms	LP oscillator	
4	TosR, TosF	Clock in (OSC1) Rise or Fall Time	—	—	25*	ns	XT oscillator	
			—	—	50*	ns	LP oscillator	

* 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.

Instruction cycle period (Tcy) equals four times the input oscillator time base period.

NOTES:

13.0 ELECTRICAL CHARACTERISTICS - PIC12C508A/PIC12C509A/ PIC12LC508A/PIC12LC509A/PIC12CR509A/PIC12CE518/PIC12CE519/ PIC12LCE518/PIC12LCE519/PIC12LCR509A

Absolute Maximum Ratings†

Ambient Temperature under bias	40°C to +125°C
Storage Temperature	65°C to +150°C
Voltage on VDD with respect to Vss	0 to +7.0 V
Voltage on MCLR with respect to Vss	0 to +14 V
Voltage on all other pins with respect to Vss	–0.3 V to (VDD + 0.3 V)
Total Power Dissipation ⁽¹⁾	700 mW
Max. Current out of Vss pin	200 mA
Max. Current into Vod pin	150 mA
Input Clamp Current, Iк (VI < 0 or VI > VDD)	±20 mA
Output Clamp Current, Iок (Vo < 0 or Vo > Vod)	±20 mA
Max. Output Current sunk by any I/O pin	25 mA
Max. Output Current sourced by any I/O pin	25 mA
Max. Output Current sourced by I/O port (GPIO)	100 mA
Max. Output Current sunk by I/O port (GPIO)	100 mA
Note 1: Power Dissipation is calculated as follows: PDIS = VDD x {IDD - Σ IOH} + Σ {(VDD-VDD) + Σ {(VDD-VD) + Σ {(VDD-VD) + Σ {(VDD-VD) + Σ {(VDD-VD) + Σ {(VDD) + Σ {(VD) + $\Sigma} {(VD) + \Sigma} {(VD) + \Sigma} {(VD) + {\Sigma} {(VD) + \Sigma} {(VD) + {\Sigma} {(V$	VOH) X IOH} + Σ (VOL X IOL)

[†]NOTICE: Stresses above those listed under "Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at those or any other conditions above those indicated in the operation listings of this specification is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.

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

VDD (Volts)	Temperature (°C)	Min	Тур	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.

NOTES:

Note the following details of the code protection feature on PICmicro[®] MCUs.

- The PICmicro family meets the specifications contained in the Microchip Data Sheet.
- Microchip believes that its family of PICmicro microcontrollers is one of the most secure products 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 PICmicro microcontroller in a manner outside the operating specifications contained in the data sheet. The person doing so may be 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 product.

If you have any further questions about this matter, please contact the local sales office nearest to you.

Information contained in this publication regarding device applications and the like is intended through suggestion only and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. No representation or warranty is given and no liability is assumed by Microchip Technology Incorporated with respect to the accuracy or use of such information, or infringement of patents or other intellectual property rights arising from such use or otherwise. Use of Microchip's products as critical components in life support systems is not authorized except with express written approval by Microchip. No licenses are conveyed, implicitly or otherwise, under any intellectual property rights.

Trademarks

The Microchip name and logo, the Microchip logo, FilterLab, KEELOQ, microID, MPLAB, PIC, PICmicro, PICMASTER, PICSTART, PRO MATE, SEEVAL and The Embedded Control Solutions Company are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

dsPIC, ECONOMONITOR, FanSense, FlexROM, fuzzyLAB, In-Circuit Serial Programming, ICSP, ICEPIC, microPort, Migratable Memory, MPASM, MPLIB, MPLINK, MPSIM, MXDEV, PICC, PICDEM, PICDEM.net, rfPIC, Select Mode and Total Endurance are trademarks of Microchip Technology Incorporated in the U.S.A.

Serialized Quick Turn Programming (SQTP) is a service mark of Microchip Technology Incorporated in the U.S.A.

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

© 2002, Microchip Technology Incorporated, Printed in the U.S.A., All Rights Reserved.





Microchip received QS-9000 quality system certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona in July 1999. The Company's quality system processes and procedures are QS-9000 compliant for its PICmicro® 8-bit MCUs, KEELoq® code hopping devices, Serial EEPROMs and microperipheral products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001 certified.