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.
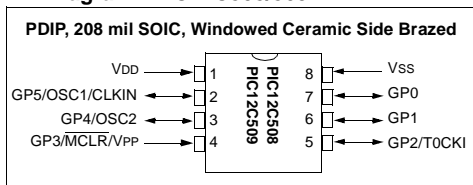
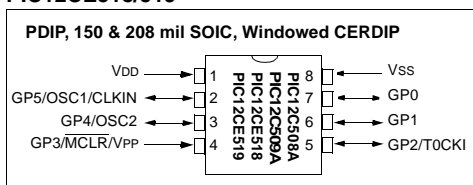**Applications of "Embedded - Microcontrollers"**

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	0°C ~ 70°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-04-sm

PIC12C5XX

Pin Diagram - PIC12C508/509

PDIP, 208 mil SOIC, Windowed Ceramic Side Brazed

```
               VDD  →  □ 1   8 □  ←  VSS
   GP5/OSC1/CLKIN  ↔  □ 2   7 □  ↔  GP0
        GP4/OSC2  ↔  □ 3   6 □  ↔  GP1
    GP3/MCLR/VPP  →  □ 4   5 □  ↔  GP2/T0CKI
```
(PIC12C508 / PIC12C509)

Pin Diagram - PIC12C508A/509A, PIC12CE518/519

PDIP, 150 & 208 mil SOIC, Windowed CERDIP

```
               VDD  →  □ 1   8 □  ←  VSS
   GP5/OSC1/CLKIN  ↔  □ 2   7 □  ↔  GP0
        GP4/OSC2  ↔  □ 3   6 □  ↔  GP1
    GP3/MCLR/VPP  →  □ 4   5 □  ↔  GP2/T0CKI
```
(PIC12C508A / PIC12C509A / PIC12CE518 / PIC12CE519)

Pin Diagram - PIC12CR509A

PDIP, 150 & 208 mil SOIC

```
               VDD  →  □ 1   8 □  ←  VSS
   GP5/OSC1/CLKIN  ↔  □ 2   7 □  ↔  GP0
        GP4/OSC2  ↔  □ 3   6 □  ↔  GP1
    GP3/MCLR/VPP  →  □ 4   5 □  ↔  GP2/T0CKI
```
(PIC12CR509A)
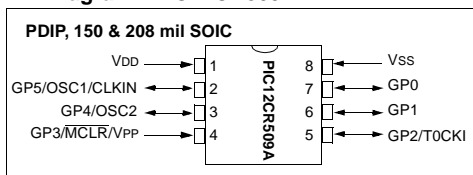
Device Differences

Device	Voltage Range	Oscillator	Oscillator Calibration[2] (Bits)	Process Technology (Microns)
PIC12C508A	3.0-5.5	See Note 1	6	0.7
PIC12LC508A	2.5-5.5	See Note 1	6	0.7
PIC12C508	2.5-5.5	See Note 1	4	0.9
PIC12C509A	3.0-5.5	See Note 1	6	0.7
PIC12LC509A	2.5-5.5	See Note 1	6	0.7
PIC12C509	2.5-5.5	See Note 1	4	0.9
PIC12CR509A	2.5-5.5	See Note 1	6	0.7
PIC12CE518	3.0-5.5	-	6	0.7
PIC12LCE518	2.5-5.5	-	6	0.7
PIC12CE519	3.0-5.5	-	6	0.7
PIC12LCE519	2.5-5.5	-	6	0.7

Note 1: If you change from the PIC12C50X to the PIC12C50XA or to the PIC12CR50XA, please verify oscillator characteristics in your application.

Note 2: See Section 7.2.5 for OSCCAL implementation differences.

3.0   ARCHITECTURAL OVERVIEW

The high performance of the PIC12C5XX family can be attributed to a number of architectural features commonly found in RISC microprocessors. To begin with, the PIC12C5XX uses a Harvard architecture in which program and data are accessed on separate buses. This improves bandwidth over traditional von Neumann architecture where program and data are fetched on the same bus. Separating program and data memory further allows instructions to be sized differently than the 8-bit wide data word. Instruction opcodes are 12-bits wide making it possible to have all single word instructions. A 12-bit wide program memory access bus fetches a 12-bit instruction in a single cycle. A two-stage pipeline overlaps fetch and execution of instructions. Consequently, all instructions (33) execute in a single cycle (1µs @ 4MHz) except for program branches.

The table below lists program memory (EPROM), data memory (RAM), ROM memory, and non-volatile (EEPROM) for each device.

Device	Memory			
	EPROM Program	ROM Program	RAM Data	EEPROM Data
PIC12C508	512 x 12		25	
PIC12C509	1024 x 12		41	
PIC12C508A	512 x 12		25	
PIC12C509A	1024 x 12		41	
PIC12CR509A		1024 x 12	41	
PIC12CE518	512 x 12		25 x 8	16 x 8
PIC12CE519	1024 x 12		41 x 8	16 x 8

The PIC12C5XX can directly or indirectly address its register files and data memory. All special function registers including the program counter are mapped in the data memory. The PIC12C5XX has a highly orthogonal (symmetrical) instruction set that makes it possible to carry out any operation on any register using any addressing mode. This symmetrical nature and lack of 'special optimal situations' make programming with the PIC12C5XX simple yet efficient. In addition, the learning curve is reduced significantly.

The PIC12C5XX device contains an 8-bit ALU and working register. The ALU is a general purpose arithmetic unit. It performs arithmetic and Boolean functions between data in the working register and any register file.

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

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

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

A simplified block diagram is shown in Figure 3-1, with the corresponding device pins described in Table 3-1.

TABLE 3-1: PIC12C5XX PINOUT DESCRIPTION

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/$\overline{\text{MCLR}}$/VPP	4	4	I	TTL/ST	Input port/master clear (reset) input/programming voltage input. When configured as $\overline{\text{MCLR}}$, this pin is an active low reset to the device. Voltage on $\overline{\text{MCLR}}$/VPP must not exceed VDD during normal device operation 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 $\overline{\text{MCLR}}$. ST when in $\overline{\text{MCLR}}$ mode.
GP4/OSC2	3	3	I/O	TTL	Bi-directional I/O port/oscillator crystal output. Connections to crystal or resonator in crystal oscillator mode (XT and LP modes only, 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	P	—	Positive supply for logic and I/O pins
VSS	8	8	P	—	Ground reference for logic and I/O pins

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

PIC12C5XX

3.1 Clocking Scheme/Instruction Cycle

The clock input (OSC1/CLKIN pin) is internally divided by four to generate four non-overlapping quadrature clocks namely Q1, Q2, Q3 and Q4. Internally, the program counter is incremented every Q1, and the instruction is fetched from program memory and latched into instruction register in Q4. It is decoded and executed during the following Q1 through Q4. The clocks and instruction execution flow is shown in Figure 3-2 and Example 3-1.
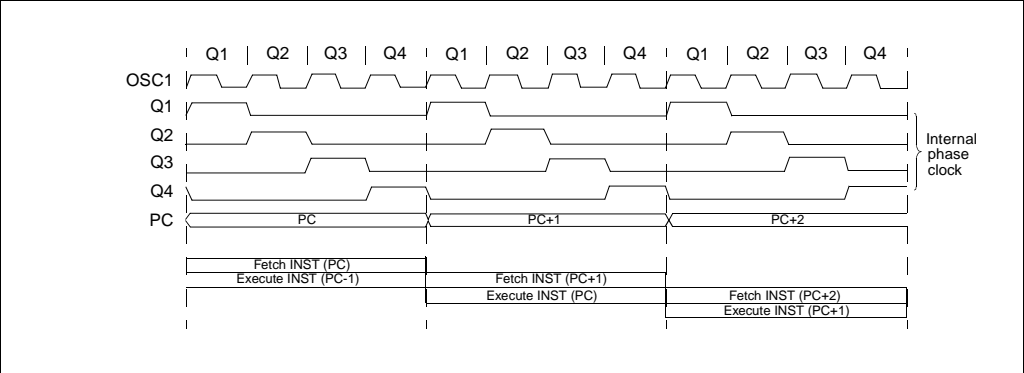
3.2 Instruction Flow/Pipelining

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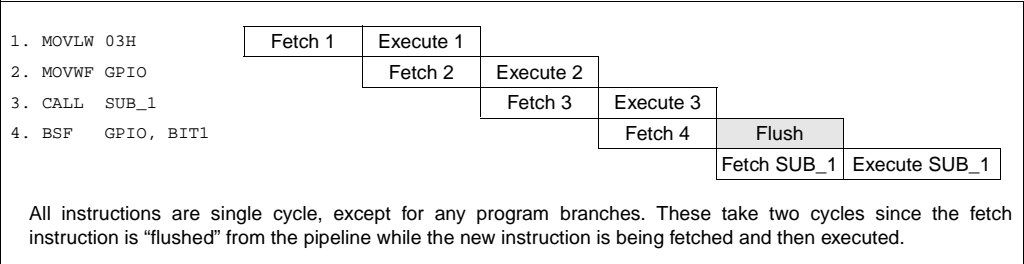
A fetch cycle begins with the program counter (PC) incrementing in Q1.

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

FIGURE 3-2: CLOCK/INSTRUCTION CYCLE

EXAMPLE 3-1: INSTRUCTION PIPELINE FLOW

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

© 1999 Microchip Technology Inc.

4.2.2 SPECIAL FUNCTION REGISTERS

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

The special registers can be classified into two sets. The special function registers associated with the "core" functions are described in this section. Those related to the operation of the peripheral features are described in the section for each peripheral feature.

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

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on Power-On Reset	Value on All Other Resets[2]
N/A	TRIS	—	—							--11 1111	--11 1111
N/A	OPTION	Contains control bits to configure Timer0, Timer0/WDT prescaler, wake-up on change, and weak pull-ups								1111 1111	1111 1111
00h	INDF	Uses contents of FSR to address data memory (not a physical register)								xxxx xxxx	uuuu uuuu
01h	TMR0	8-bit real-time clock/counter								xxxx xxxx	uuuu uuuu
02h[1]	PCL	Low order 8 bits of PC								1111 1111	1111 1111
03h	STATUS	GPWUF	—	PA0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	q00q quuu[3]
04h	FSR (PIC12C508/ PIC12C508A/ PIC12C518)	Indirect data memory address pointer								111x xxxx	111u uuuu
04h	FSR (PIC12C509/ PIC12C509A/ PIC12CR509A/ PIC12CE519)	Indirect data memory address pointer								110x xxxx	11uu uuuu
05h	OSCCAL (PIC12C508/ PIC12C509)	CAL3	CAL2	CAL1	CAL0	—	—	—	—	0111 ----	uuuu ----
05h	OSCCAL (PIC12C508A/ PIC12C509A/ PIC12CE518/ PIC12CE519/ PIC12CR509A)	CAL5	CAL4	CAL3	CAL2	CAL1	CAL0	—	—	1000 00--	uuuu uu--
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 boxes = unimplemented or unused, — = unimplemented, read as '0' (if applicable)
x = unknown, u = unchanged, q = see the tables in Section 8.7 for possible values.

Note 1: The upper byte of the Program Counter is not directly accessible. See Section 4.6 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.

PIC12C5XX

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 $\overline{\text{TO}}$ and $\overline{\text{PD}}$ 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 uluu` (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	$\overline{\text{TO}}$	$\overline{\text{PD}}$	Z	DC	C
bit7	6	5	4	3	2	1	bit0

R = Readable bit
W = Writable bit
- n = Value at POR reset

bit 7: **GPWUF**: GPIO reset bit
1 = Reset due to wake-up from SLEEP on pin change
0 = After power up or other reset

bit 6: **Unimplemented**

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.

bit 4: **$\overline{\text{TO}}$**: Time-out bit
1 = After power-up, `CLRWDT` instruction, or `SLEEP` instruction
0 = A WDT time-out occurred

bit 3: **$\overline{\text{PD}}$**: Power-down bit
1 = After power-up or by the `CLRWDT` instruction
0 = By execution of the `SLEEP` instruction

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

bit 1: **DC**: Digit carry/$\overline{\text{borrow}}$ bit (for `ADDWF` and `SUBWF` 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

bit 0: **C**: Carry/$\overline{\text{borrow}}$ bit (for `ADDWF`, `SUBWF` and `RRF`, `RLF` instructions)
ADDWF	**SUBWF**	**RRF or RLF**
1 = A carry occurred	1 = A borrow did not occur	Load bit with LSB or MSB, respectively
0 = A carry did not occur	0 = A borrow occurred	

5.0   I/O PORT

As with any other register, the I/O register can be written and read under program control. However, read instructions (e.g., MOVF GPIO,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 hi-impedance) since the I/O control registers are all set. See Section 7.0 for SCL and SDA description for PIC12CE5XX.

5.1   GPIO

GPIO is an 8-bit I/O register. Only the low order 6 bits are used (GP5:GP0). Bits 7 and 6 are unimplemented and read as '0's. Please note that 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 port read. Pins GP0, GP1, and GP3 can be configured with weak pull-ups and also with wake-up on change. The wake-up on change and weak pull-up functions are not pin selectable. If pin 4 is configured as $\overline{MCLR}$, weak pull-up is always on and wake-up on change for this pin is not enabled.

5.2   TRIS Register

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 hi-impedance mode. A '0' puts the contents of the output data latch on the selected pins, enabling the output buffer. The exceptions are GP3 which is input only and GP2 which may be controlled by the option register, see Figure 4-5.

> **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.

5.3   I/O Interfacing

The equivalent circuit for an I/O port pin is shown in Figure 5-1. All port pins, except 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 GPIO,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 GP3) can be programmed individually as input or output.

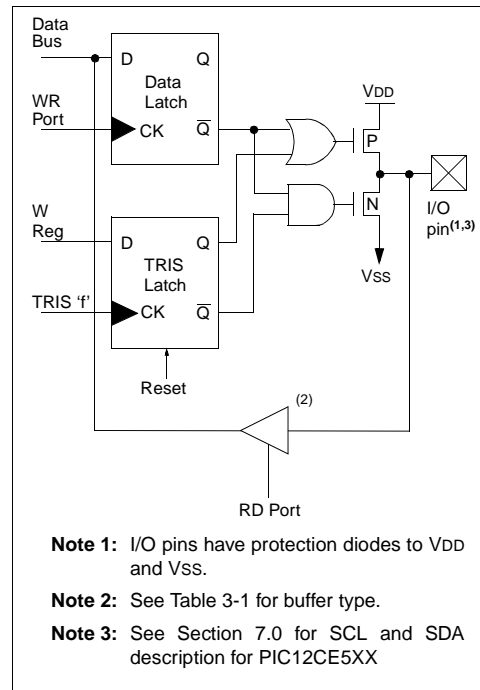**FIGURE 5-1:   EQUIVALENT CIRCUIT FOR A SINGLE I/O PIN**

Note 1: I/O pins have protection diodes to VDD and VSS.

Note 2: See Table 3-1 for buffer type.

Note 3: See Section 7.0 for SCL and SDA description for PIC12CE5XX

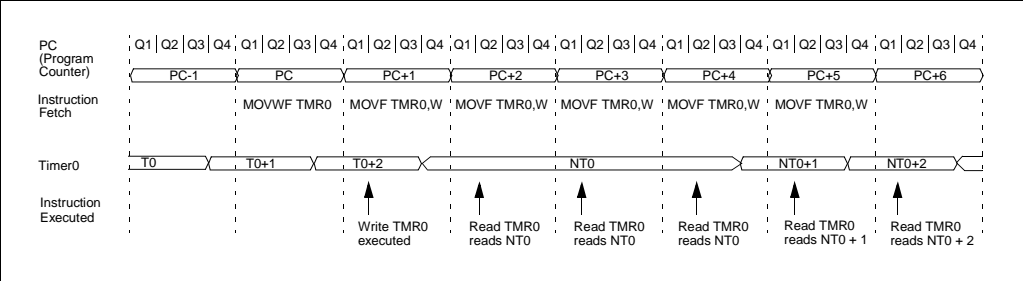**FIGURE 6-2: TIMER0 TIMING: INTERNAL CLOCK/NO PRESCALE**

FIGURE 6-3: TIMER0 TIMING: INTERNAL CLOCK/PRESCALE 1:2
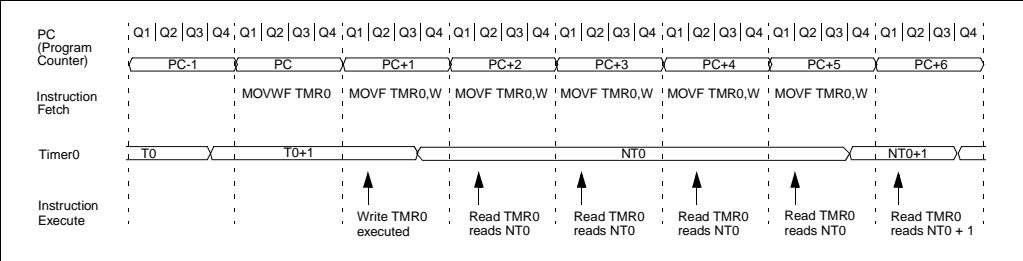
TABLE 6-1: REGISTERS ASSOCIATED WITH TIMER0

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,

7.3 WRITE OPERATIONS

7.3.1 BYTE WRITE

Following the start signal from the master, the device code (4 bits), the don't care bits (3 bits), and the R/$\overline{W}$ bit (which is a logic low) are placed onto the bus by the master transmitter. This indicates to the addressed slave receiver that a byte with a word address will follow after it has generated an acknowledge bit during the ninth clock cycle. Therefore, the next byte transmitted by the master is the word address and will be written into the address pointer. Only the lower four address bits are used by the device, and the upper four bits are don't cares. The address byte is acknowledgeable and the master device will then transmit the data word to be written into the addressed memory location. The memory acknowledges again and the master generates a stop condition. This initiates the internal write cycle, and during this time will not generate acknowledge signals (Figure 7-7). After a byte write command, the internal address counter will not be incremented and will point to the same address location that was just written. If a stop bit is transmitted to the device at any point in the write command sequence before the entire sequence is complete, then the command will abort and no data will be written. If more than 8 data bits are transmitted before the stop bit is sent, then the device will clear the previously loaded byte and begin loading the data buffer again. If more than one data byte is transmitted to the device and a stop bit is sent before a full eight data bits have been transmitted, then the write command will abort and no data will be written. The EEPROM memory employs a VCC threshold detector circuit which disables the internal erase/write logic if the VCC is below minimum VDD.

Byte write operations must be preceded and immediately followed by a bus not busy bus cycle where both SDA and SCL are held high.

7.4 ACKNOWLEDGE POLLING

Since the device will not acknowledge during a write cycle, this can be used to determine when the cycle is complete (this feature can be used to maximize bus throughput). Once the stop condition for a write command has been issued from the master, the device initiates the internally timed write cycle. ACK polling can be initiated immediately. This involves the master sending a start condition followed by the control byte for a write command (R/$\overline{W}$ = 0). If the device is still busy with the write cycle, then no ACK will be returned. If no ACK is returned, then the start bit and control byte must be re-sent. If the cycle is complete, then the device will return the ACK and the master can then proceed with the next read or write command. See Figure 7-6 for flow diagram.

FIGURE 7-6: ACKNOWLEDGE POLLING FLOW
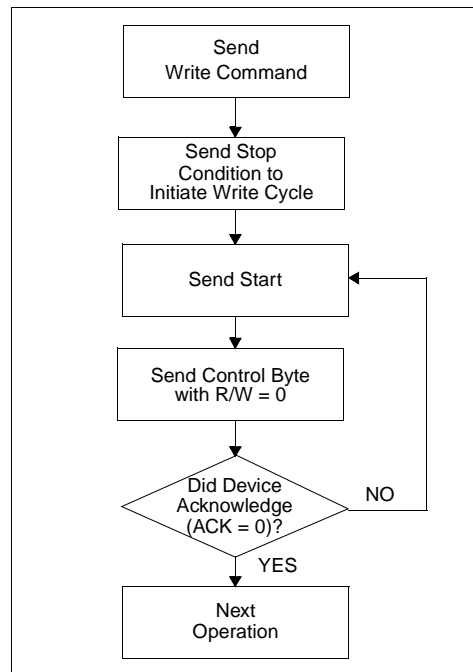
FIGURE 7-7: BYTE WRITE
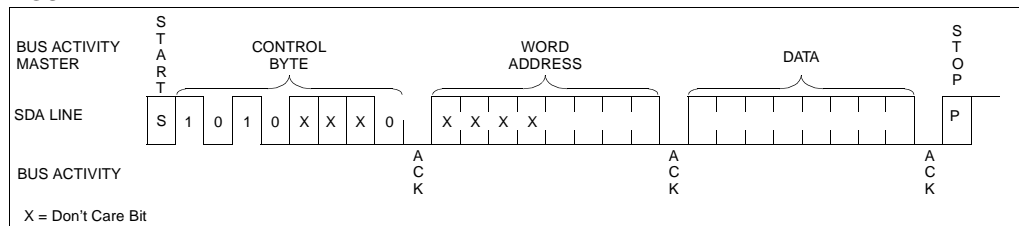
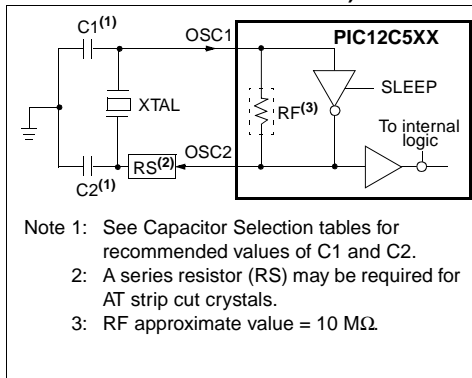## 8.2 Oscillator Configurations

8.2.1 OSCILLATOR TYPES

The PIC12C5XX can be operated in four different oscillator modes. The user can program two configuration bits (FOSC1:FOSC0) to select one of these four modes:

- LP: Low Power Crystal
- XT: Crystal/Resonator
- INTRC: Internal 4 MHz Oscillator
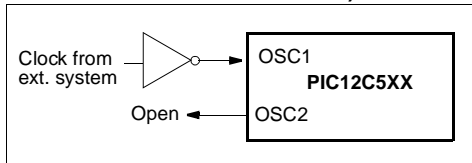- EXTRC: External Resistor/Capacitor

8.2.2 CRYSTAL OSCILLATOR / CERAMIC RESONATORS

In XT or LP modes, a crystal or ceramic resonator is connected to the GP5/OSC1/CLKIN and GP4/OSC2 pins to establish oscillation (Figure 8-2). The PIC12C5XX oscillator design requires the use of a parallel cut crystal. Use of a series cut crystal may give a frequency out of the crystal manufacturers specifications. When in XT or LP modes, the device can have an external clock source drive the GP5/OSC1/CLKIN pin (Figure 8-3).

FIGURE 8-2: CRYSTAL OPERATION (OR CERAMIC RESONATOR) (XT OR LP OSC CONFIGURATION)

Note 1: See Capacitor Selection tables for recommended values of C1 and C2.
2: A series resistor (RS) may be required for AT strip cut crystals.
3: RF approximate value = 10 MΩ.

FIGURE 8-3: EXTERNAL CLOCK INPUT OPERATION (XT OR LP OSC CONFIGURATION)

TABLE 8-1: CAPACITOR SELECTION FOR CERAMIC RESONATORS - PIC12C5XX

Osc Type	Resonator Freq	Cap. Range C1	Cap. Range C2
XT	4.0 MHz	30 pF	30 pF

These values are for design guidance only. Since each resonator has its own characteristics, the user should consult the resonator manufacturer for appropriate values of external components.

TABLE 8-2: CAPACITOR SELECTION FOR CRYSTAL OSCILLATOR - PIC12C5XX

Osc Type	Resonator Freq	Cap.Range C1	Cap. Range C2
LP	32 kHz[1]	15 pF	15 pF
XT	200 kHz	47-68 pF	47-68 pF
	1 MHz	15 pF	15 pF
	4 MHz	15 pF	15 pF

Note 1: For V_{DD} > 4.5V, C1 = C2 $\approx$ 30 pF is recommended.

These values are for design guidance only. Rs may be required to avoid overdriving crystals with low drive level specification. Since each crystal has its own characteristics, the user should consult the crystal manufacturer for appropriate values of external components.

8.2.3   EXTERNAL CRYSTAL OSCILLATOR CIRCUIT

Either a prepackaged oscillator or a simple oscillator circuit with TTL gates can be used as an external crystal oscillator circuit. Prepackaged oscillators provide a wide operating range and better stability. A well-designed crystal oscillator will provide good performance with TTL gates. Two types of crystal oscillator circuits can be used: one with parallel resonance, or one with series resonance.

Figure 8-4 shows implementation of a parallel resonant oscillator circuit. The circuit is designed to use the fundamental frequency of the crystal. The 74AS04 inverter performs the 180-degree phase shift that a parallel oscillator requires. The 4.7 kΩ resistor provides the negative feedback for stability. The 10 kΩ potentiometers bias the 74AS04 in the linear region. This circuit could be used for external oscillator designs.

FIGURE 8-4:   EXTERNAL PARALLEL RESONANT CRYSTAL OSCILLATOR CIRCUIT
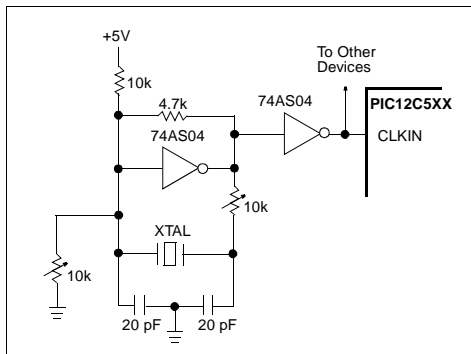
Figure 8-5 shows a series resonant oscillator circuit. This circuit is also designed to use the fundamental frequency of the crystal. The inverter performs a 180-degree phase shift in a series resonant oscillator circuit. The 330 Ω resistors provide the negative feedback to bias the inverters in their linear region.

FIGURE 8-5:   EXTERNAL SERIES RESONANT CRYSTAL OSCILLATOR CIRCUIT
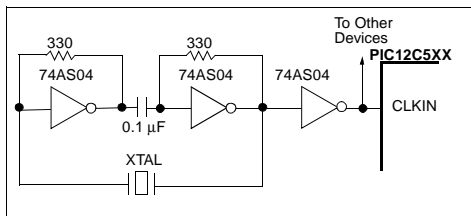
8.2.4   EXTERNAL RC OSCILLATOR

For timing insensitive applications, the RC device option offers additional cost savings. The RC oscillator frequency is a function of the supply voltage, the resistor (Rext) and capacitor (Cext) values, and the operating temperature. In addition to this, the oscillator frequency will vary from unit to unit due to normal process parameter variation. Furthermore, the difference in lead frame capacitance between package types will also affect the oscillation frequency, especially for low Cext values. The user also needs to take into account variation due to tolerance of external R and C components used.

Figure 8-6 shows how the R/C combination is connected to the PIC12C5XX. For Rext values below 2.2 kΩ, the oscillator operation may become unstable, or stop completely. For very high Rext values (e.g., 1 MΩ) the oscillator becomes sensitive to noise, humidity and leakage. Thus, we recommend keeping Rext between 3 kΩ and 100 kΩ.

Although the oscillator will operate with no external capacitor (Cext = 0 pF), we recommend using values above 20 pF for noise and stability reasons. With no or small external capacitance, the oscillation frequency can vary dramatically due to changes in external capacitances, such as PCB trace capacitance or package lead frame capacitance.

The Electrical Specifications sections show RC frequency variation from part to part due to normal process variation. The variation is larger for larger R (since leakage current variation will affect RC frequency more for large R) and for smaller C (since variation of input capacitance will affect RC frequency more).

Also, see the Electrical Specifications sections for variation of oscillator frequency due to VDD for given Rext/Cext values as well as frequency variation due to operating temperature for given R, C, and VDD values.

FIGURE 8-6:   EXTERNAL RC OSCILLATOR MODE
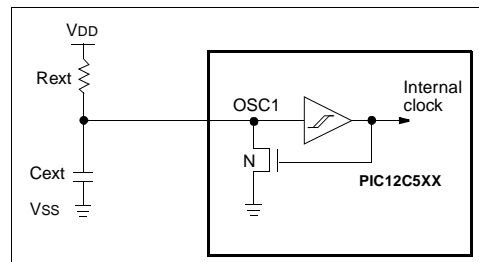
MOVF	**Move f**
Syntax:	[*label*] MOVF f,d
Operands:	$0 \leq f \leq 31$ $d \in [0,1]$
Operation:	(f) → (dest)
Status Affected:	Z
Encoding:	`0010` `00df` `ffff`
Description:	The contents of register 'f' is moved to destination 'd'. If 'd' is 0, destination is the W register. If 'd' is 1, the destination is file register 'f'. 'd' is 1 is useful to test a file register since status flag Z is affected.
Words:	1
Cycles:	1
Example:	`MOVF    FSR,    0`

   After Instruction
     W   =   value in FSR register

MOVLW	**Move Literal to W**
Syntax:	[*label*] MOVLW k
Operands:	$0 \leq k \leq 255$
Operation:	k → (W)
Status Affected:	None
Encoding:	`1100` `kkkk` `kkkk`
Description:	The eight bit literal 'k' is loaded into the W register. The don't cares will assemble as 0s.
Words:	1
Cycles:	1
Example:	`MOVLW   0x5A`

   After Instruction
     W   =   0x5A

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

   Before Instruction
     TEMP_REG   =   0xFF
     W   =   0x4F
   After Instruction
     TEMP_REG   =   0x4F
     W   =   0x4F

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

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

Before Instruction
    W   =   0x07

After Instruction
    OPTION =   0x07

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

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

Before Instruction
    W   =   0x07

After Instruction
    W   =   value of k8

RLF	Rotate Left f through Carry
Syntax:	[*label*] RLF f,d
Operands:	0 ≤ f ≤ 31 d ∈ [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'.
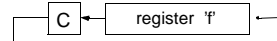
Words:	1
Cycles:	1
Example:	`RLF    REG1,0`

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 ≤ f ≤ 31 d ∈ [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'.
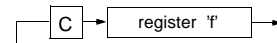
Words:	1
Cycles:	1
Example:	`RRF    REG1,0`

Before Instruction
    REG1   =   1110 0110
    C   =   0

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

11.0 ELECTRICAL CHARACTERISTICS - PIC12C508/PIC12C509

Absolute Maximum Ratings†

Ambient Temperature under bias .... −40°C to +125°C

Storage Temperature .... −65°C to +150°C

Voltage on V_{DD} with respect to V_{SS} .... 0 to +7.5 V

Voltage on $\overline{MCLR}$ with respect to V_{SS}.. 0 to +14 V

Voltage on all other pins with respect to V_{SS} .... −0.6 V to (V_{DD} + 0.6 V)

Total Power Dissipation[1] .... 700 mW

Max. Current out of V_{SS} pin .... 200 mA

Max. Current into V_{DD} pin .... 150 mA

Input Clamp Current, I_{IK} (V_I < 0 or V_I > V_{DD}) .... ±20 mA

Output Clamp Current, I_{OK} (V_O < 0 or V_O > V_{DD}).. ±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: $P_{DIS} = V_{DD} \times \{I_{DD} - \Sigma I_{OH}\} + \Sigma \{(V_{DD}-V_{OH}) \times I_{OH}\} + \Sigma(V_{OL} \times I_{OL})$

> †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.

PIC12C5XX

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

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 11-4:   TIMING REQUIREMENTS - PIC12C508/C509

AC Characteristics		Standard Operating Conditions (unless otherwise specified) Operating Temperature $0°C \leq T_A \leq +70°C$ (commercial) $-40°C \leq T_A \leq +85°C$ (industrial) $-40°C \leq T_A \leq +125°C$ (extended) Operating Voltage VDD range is described in Section 11.1					
Parameter No.	Sym	Characteristic	Min	Typ[1]	Max	Units
17	TosH2ioV	OSC1↑ (Q1 cycle) to Port out valid[3]	—	—	100*	ns
18	TosH2ioI	OSC1↑ (Q2 cycle) to Port input invalid (I/O in hold time)	TBD	—	—	ns
19	TioV2osH	Port input valid to OSC1↑ (I/O in setup time)	TBD	—	—	ns
20	TioR	Port output rise time[2, 3]	—	10	25**	ns
21	TioF	Port output fall time[2, 3]	—	10	25**	ns

*   These parameters are characterized but not tested.
**   These parameters are design targets and are not tested. No characterization data available at this time.
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:   Measurements are taken in EXTRC mode.
   3:   See Figure 11-1 for loading conditions.

FIGURE 11-4:   RESET, WATCHDOG TIMER, AND DEVICE RESET TIMER TIMING - PIC12C508/C509

Note 1: I/O pins must be taken out of hi-impedance mode by enabling the output drivers in software.
   2: Runs in MCLR or WDT reset only in XT and LP modes.
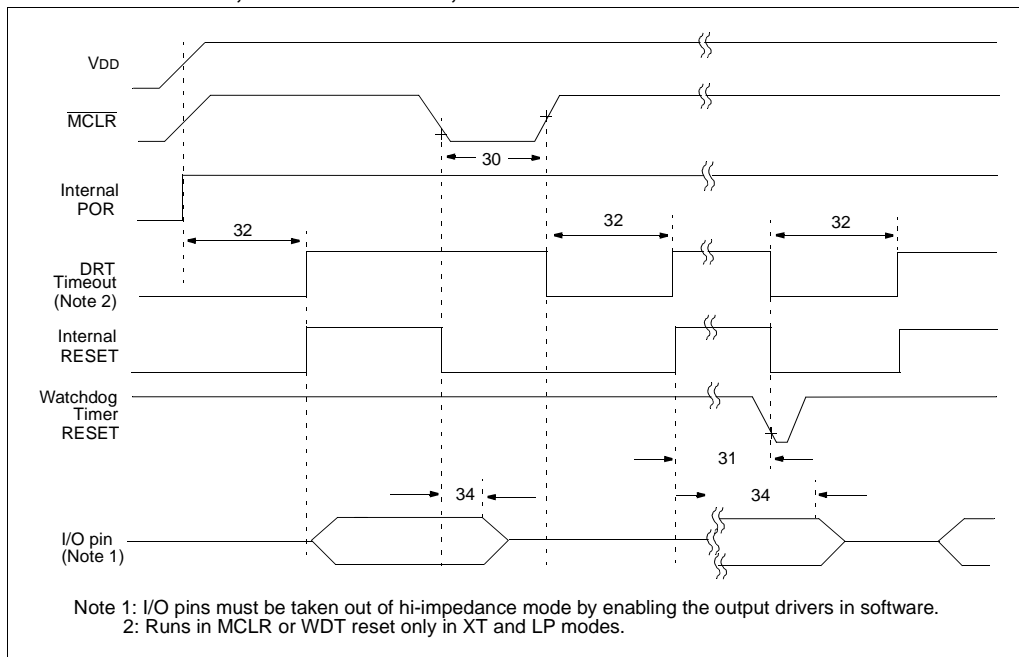
FIGURE 12-5: IOH vs. VOH, VDD = 2.5 V

FIGURE 12-6: IOH vs. VOH, VDD = 5.5 V

FIGURE 12-7: IOL vs. VOL, VDD = 2.5 V
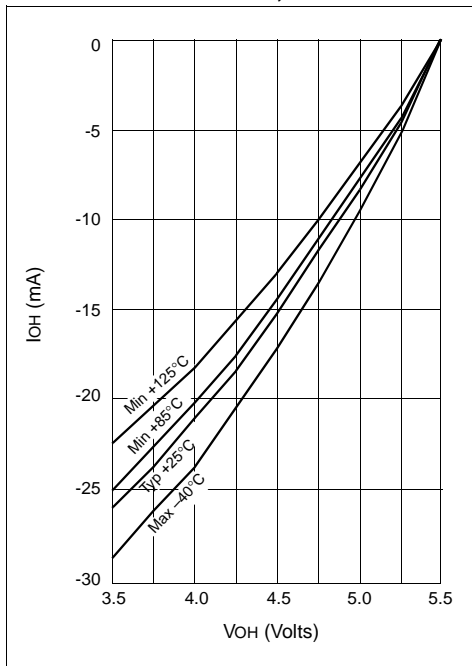
FIGURE 12-8: IOL vs. VOL, VDD = 5.5 V
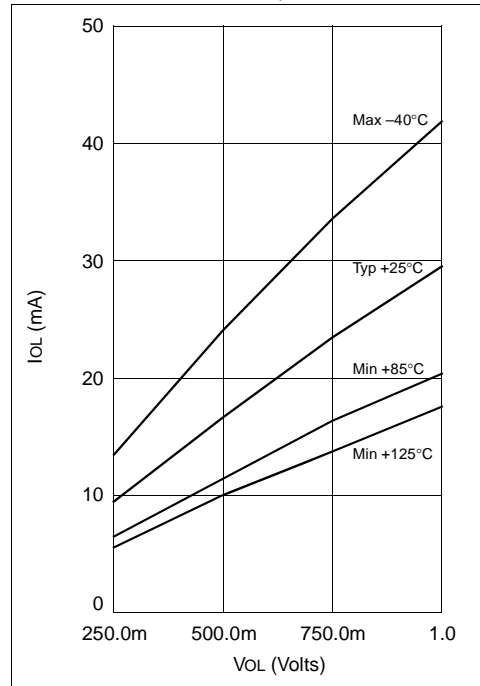
13.1 DC CHARACTERISTICS: PIC12C508A/509A (Commercial, Industrial, Extended)
PIC12CE518/519 (Commercial, Industrial, Extended)
PIC12CR509A (Commercial, Industrial, Extended)

DC Characteristics Power Supply Pins					Standard Operating Conditions (unless otherwise specified) Operating Temperature 0°C ≤ TA ≤ +70°C (commercial) −40°C ≤ TA ≤ +85°C (industrial) −40°C ≤ TA ≤ +125°C (extended)			
Parm No.	Characteristic	Sym	Min	Typ[1]	Max	Units	Conditions
D001	**Supply Voltage**	VDD	3.0		5.5	V	FOSC = DC to 4 MHz (Commercial/ Industrial, Extended)
D002	**RAM Data Retention Voltage**[2]	VDR		1.5*		V	Device in SLEEP mode
D003	**VDD Start Voltage to ensure Power-on Reset**	VPOR		VSS		V	See section on Power-on Reset for details
D004	**VDD Rise Rate to ensure Power-on Reset**	SVDD	0.05*			V/ms	See section on Power-on Reset for details
D010	**Supply Current**[3]	IDD	—	0.8	1.4	mA	XT and EXTRC options (Note 4) FOSC = 4 MHz, VDD = 5.5V
D010C			—	0.8	1.4	mA	INTRC Option FOSC = 4 MHz, VDD = 5.5V
D010A			—	19	27	µA	LP OPTION, Commercial Temperature FOSC = 32 kHz, VDD = 3.0V, WDT disabled
			—	19	35	µA	LP OPTION, Industrial Temperature FOSC = 32 kHz, VDD = 3.0V, WDT disabled
			—	30	55	µA	LP OPTION, Extended Temperature FOSC = 32 kHz, VDD = 3.0V, WDT disabled
D020 D021 D021B	**Power-Down Current**[5]	IPD	— — —	0.25 0.25 2	4 5 12	µA µA µA	VDD = 3.0V, Commercial WDT disabled VDD = 3.0V, Industrial WDT disabled VDD = 3.0V, Extended WDT disabled
D022	**Power-Down Current**	ΔIWDT	— — —	2.2 2.2 4	5 6 11	µA µA µA	VDD = 3.0V, Commercial VDD = 3.0V, Industrial VDD = 3.0V, Extended
	Supply Current[3] **During read/write to EEPROM peripheral**	ΔIEE	—	0.1	0.2	mA	FOSC = 4 MHz, Vdd = 5.5V, SCL = 400kHz

* These parameters are characterized but not tested.

Note 1: Data in the Typical ("Typ") column is based on characterization results at 25°C. This data is for design guidance only and is not tested.

2: This is the limit to which VDD can be lowered in SLEEP mode without losing RAM data.

3: The supply current is mainly a function of the operating voltage and frequency. Other factors such as bus loading, oscillator type, bus rate, internal code execution pattern, and temperature also have an impact on the current consumption.

   a) The test conditions for all IDD measurements in active operation mode are:
     OSC1 = external square wave, from rail-to-rail; all I/O pins tristated, pulled to
     VSS, T0CKI = VDD, $\overline{MCLR}$ = VDD; WDT enabled/disabled as specified.
   b) For standby current measurements, the conditions are the same, except that
     the device is in SLEEP mode.

4: Does not include current through Rext. The current through the resistor can be estimated by the formula: IR = VDD/2Rext (mA) with Rext in kOhm.

5: The power down current in SLEEP mode does not depend on the oscillator type. Power down current is measured with the part in SLEEP mode, with all I/O pins in hi-impedance state and tied to VDD or VSS.

13.3 DC CHARACTERISTICS: PIC12C508A/509A (Commercial, Industrial, Extended)
PIC12C518/519 (Commercial, Industrial, Extended)
PIC12CR509A (Commercial, Industrial, Extended)

DC CHARACTERISTICS	Standard Operating Conditions (unless otherwise specified) Operating temperature 0°C ≤ TA ≤ +70°C (commercial) −40°C ≤ TA ≤ +85°C (industrial) −40°C ≤ TA ≤ +125°C (extended) Operating voltage VDD range as described in DC spec Section 13.1 and Section 13.2.					
Param No.	**Characteristic**	**Sym**	**Min**	**Typ†**	**Max**	**Units**	**Conditions**

Param No.	Characteristic	Sym	Min	Typ†	Max	Units	Conditions
	Input Low Voltage						
	I/O ports	VIL					
D030	with TTL buffer		Vss	-	0.8V	V	For 4.5V ≤ VDD ≤ 5.5V
			Vss	-	0.15VDD	V	otherwise
D031	with Schmitt Trigger buffer		Vss	-	0.2VDD	V	
D032	MCLR, GP2/T0CKI (in EXTRC mode)		Vss	-	0.2VDD	V	
D033	OSC1 (in EXTRC mode)		Vss	-	0.2VDD		Note 1
D033	OSC1 (in XT and LP)		Vss	-	0.3VDD	V	Note 1
	Input High Voltage						
	I/O ports	VIH	-				
D040	with TTL buffer		0.25VDD + 0.8V	-	VDD	V	4.5V ≤ VDD ≤ 5.5V
D040A			2.0V	-	VDD	V	otherwise
D041	with Schmitt Trigger buffer		0.8VDD	-	VDD	V	For entire VDD range
D042	MCLR, GP2/T0CKI		0.8VDD	-	VDD	V	
D042A	OSC1 (XT and LP)		0.7VDD	-	VDD	V	Note 1
D043	OSC1 (in EXTRC mode)		0.9VDD	-	VDD	V	
D070	GPIO weak pull-up current (Note 4)	IPUR	30	250	400	µA	VDD = 5V, VPIN = VSS
	MCLR pull-up current	-	-	-	30	µA	VDD = 5V, VPIN = VSS
	Input Leakage Current (Notes 2, 3)						
D060	I/O ports	IIL	-	-	±1	µA	Vss ≤ VPIN ≤ VDD, Pin at hi-impedance
D061	T0CKI		-	-	±5	µA	Vss ≤ VPIN ≤ VDD
D063	OSC1		-	-	±5	µA	Vss ≤ VPIN ≤ VDD, XT and LP osc configuration
	Output Low Voltage						
D080	I/O ports	VOL	-	-	0.6	V	IOL = 8.5 mA, VDD = 4.5V, −40°C to +85°C
D080A			-	-	0.6	V	IOL = 7.0 mA, VDD = 4.5V, −40°C to +125°C
	Output High Voltage						
D090	I/O ports (Note 3)	VOH	VDD - 0.7	-	-	V	IOH = -3.0 mA, VDD = 4.5V, −40°C to +85°C
D090A			VDD - 0.7	-	-	V	IOH = -2.5 mA, VDD = 4.5V, −40°C to +125°C
	Capacitive Loading Specs on Output Pins						
D100	OSC2 pin	COSC2	-	-	15	pF	In XT and LP modes when external clock is used to drive OSC1.
D101	All I/O pins	CIO	-	-	50	pF	

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

Note 1: In EXTRC oscillator configuration, the OSC1/CLKIN pin is a Schmitt Trigger input. It is not recommended that the PIC12C5XX be driven with external clock in RC mode.

2: The leakage current on the MCLR pin is strongly dependent on the applied voltage level. The specified levels represent normal operating conditions. Higher leakage current may be measured at different input voltages.

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

4: This spec. applies when GP3/MCLR is configured as MCLR. The leakage current of the MCLR circuit is higher than the standard I/O logic.

INDEX