Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

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

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	PIC
Core Size	8-Bit
Speed	4MHz
Connectivity	-
Peripherals	POR, WDT
Number of I/O	5
Program Memory Size	1.75KB (1K x 14)
Program Memory Type	OTP
EEPROM Size	16 x 8
RAM Size	128 x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 5.5V
Data Converters	A/D 4x8b
Oscillator Type	Internal
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Through Hole
Package / Case	8-DIP (0.300", 7.62mm)
Supplier Device Package	8-PDIP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic12ce673-04-p

NOTES:

TABLE 3-1:   PIC12C67X PINOUT DESCRIPTION

Name	DIP Pin #	I/O/P Type	Buffer Type	Description
GP0/AN0	7	I/O	TTL/ST	Bi-directional I/O port/serial programming data/analog input 0. Can be software programmed for internal weak pull-up and interrupt-on-pin change. This buffer is a Schmitt Trigger input when used in serial programming mode.
GP1/AN1/V$_{REF}$	6	I/O	TTL/ST	Bi-directional I/O port/serial programming clock/analog input 1/ voltage reference. Can be software programmed for internal weak pull-up and interrupt-on-pin change. This buffer is a Schmitt Trigger input when used in serial programming mode.
GP2/T0CKI/AN2/INT	5	I/O	ST	Bi-directional I/O port/analog input 2. Can be configured as T0CKI or external interrupt.
GP3/$\overline{\text{MCLR}}$/V$_{PP}$	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 $\overline{\text{MCLR}}$/V$_{PP}$ must not exceed V$_{DD}$ during normal device operation. Can be software programmed for internal weak pull-up and interrupt-on-pin change. Weak pull-up always on if configured as $\overline{\text{MCLR}}$. This buffer is Schmitt Trigger when in $\overline{\text{MCLR}}$ mode.
GP4/OSC2/AN3/CLKOUT	3	I/O	TTL	Bi-directional I/O port/oscillator crystal output/analog input 3. Connections to crystal or resonator in crystal oscillator mode (HS, XT and LP modes only, GPIO in other modes). In EXTRC and INTRC modes, the pin output can be configured to CLK-OUT, which has 1/4 the frequency of OSC1 and denotes the instruction cycle rate.
GP5/OSC1/CLKIN	2	I/O	TTL/ST	Bi-directional IO port/oscillator crystal input/external clock source input (GPIO in INTRC mode only, OSC1 in all other oscillator modes). Schmitt trigger input for EXTRC oscillator mode.
V$_{DD}$	1	P	—	Positive supply for logic and I/O pins.
V$_{SS}$	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.

PIC12C67X

3.1 Clocking Scheme/Instruction Cycle

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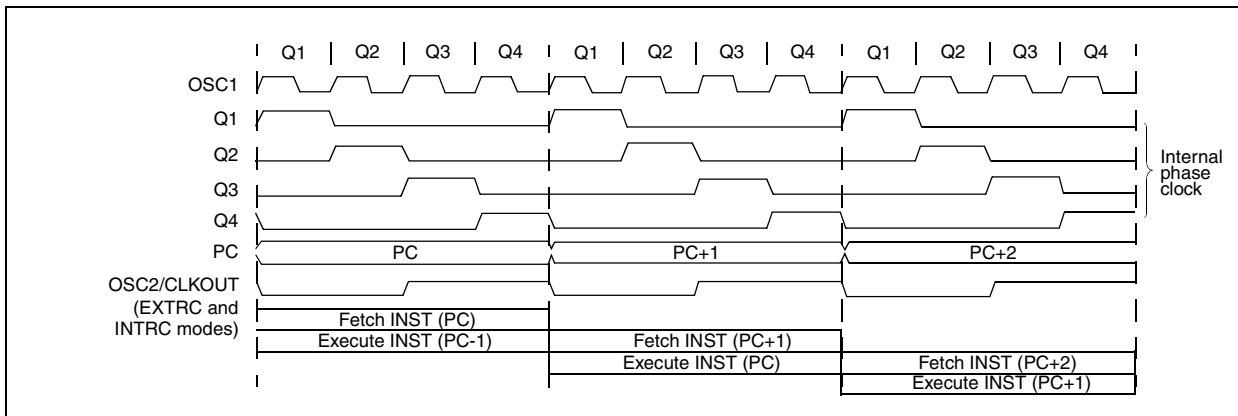
3.2 Instruction Flow/Pipelining

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 (i.e., GOTO), then two cycles are required to complete the instruction (Example 3-1).
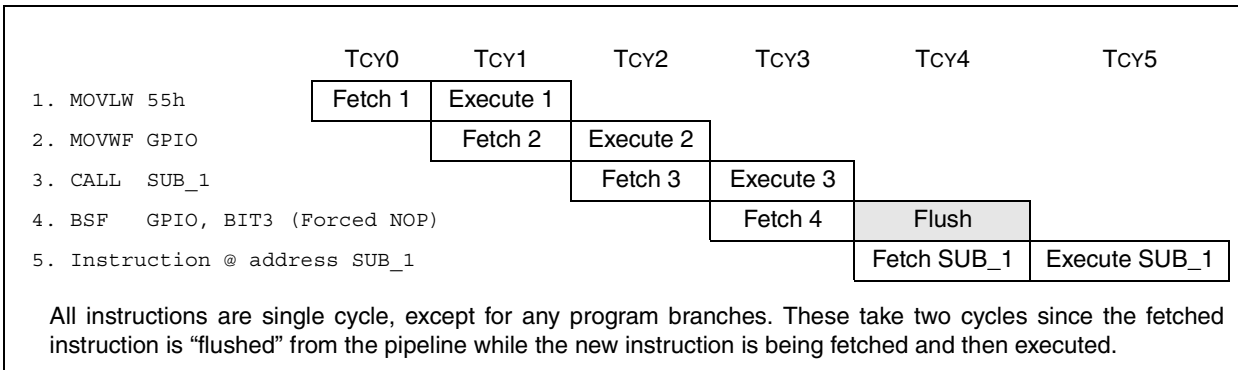
A fetch cycle begins with the program counter (PC) incrementing in Q1.

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

	TCY0	TCY1	TCY2	TCY3	TCY4	TCY5
1. MOVLW 55h	Fetch 1	Execute 1				
2. MOVWF GPIO		Fetch 2	Execute 2			
3. CALL SUB_1			Fetch 3	Execute 3		
4. BSF GPIO, BIT3 (Forced NOP)				Fetch 4	Flush	
5. Instruction @ address SUB_1					Fetch SUB_1	Execute SUB_1

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

PIC12C67X

4.2.2.6   PCON REGISTER

The Power Control (PCON) Register contains a flag bit
to allow differentiation between a Power-on Reset
(POR), an external $\overline{\text{MCLR}}$ Reset and a WDT Reset.

REGISTER 4-6:   PCON REGISTER (ADDRESS 8Eh)

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	U-0
—	—	—	—	—	—	$\overline{\text{POR}}$	—
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit,
      read as '0'
- n = Value at POR reset

bit 7-2:   **Unimplemented:** Read as '0'

bit 1:   **$\overline{\text{POR}}$:** Power-on Reset Status bit
           1 = No Power-on Reset occurred
           0 = A Power-on Reset occurred (must be set in software after a Power-on Reset occurs)

bit 0:   **Unimplemented:** Read as '0'

FIGURE 5-5: BLOCK DIAGRAM OF GP5/OSC1/CLKIN PIN
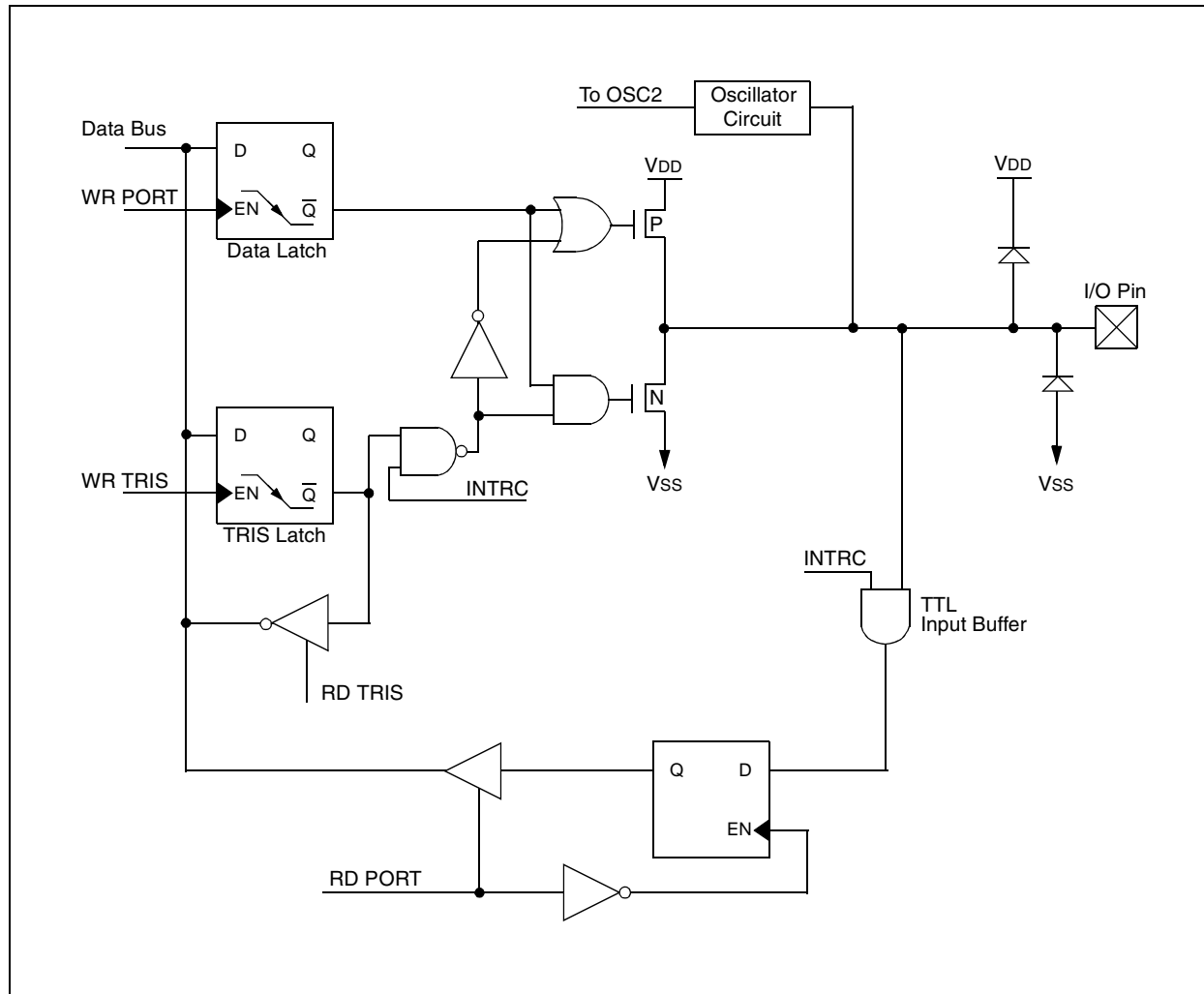
NOTES:

7.3.1 SWITCHING PRESCALER ASSIGNMENT

The prescaler assignment is fully under software control, (i.e., it can be changed "on-the-fly" during program execution).

Note:	To avoid an unintended device RESET, the following instruction sequence (shown in Example 7-1) must be executed when changing the prescaler assignment from Timer0 to the WDT. This sequence must be followed even if the WDT is disabled.

EXAMPLE 7-1: CHANGING PRESCALER (TIMER0→WDT)

```
BCF     STATUS, RP0  ;Bank 0
CLRF    TMR0         ;Clear TMR0 & Prescaler
BSF     STATUS, RP0  ;Bank 1
CLRWDT               ;Clears WDT
MOVLW   b'xxxx1xxx'  ;Select new prescale
MOVWF   OPTION_REG   ;value & WDT
BCF     STATUS, RP0  ;Bank 0
```

To change prescaler from the WDT to the Timer0 module, use the sequence shown in Example 7-2.

EXAMPLE 7-2: CHANGING PRESCALER (WDT→TIMER0)

```
CLRWDT               ;Clear WDT and
                     ;prescaler
BSF     STATUS, RP0  ;Bank 1
MOVLW   b'xxxx0xxx'  ;Select TMR0, new
                     ;prescale value and
MOVWF   OPTION_REG   ;clock source
BCF     STATUS, RP0  ;Bank 0
```

TABLE 7-1: REGISTERS ASSOCIATED WITH TIMER0

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR	Value on all other Resets
01h	TMR0	Timer0 module's register								xxxx xxxx	uuuu uuuu
0Bh/8Bh	INTCON	GIE	PEIE	T0IE	INTE	GPIE	T0IF	INTF	GPIF	0000 000x	0000 000u
81h	OPTION	GPPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
85h	TRIS	—	—	TRIS5	TRIS4	TRIS3	TRIS2	TRIS1	TRIS0	--11 1111	--11 1111

Legend: x = unknown, u = unchanged, - = unimplemented locations read as '0'. Shaded cells are not used by Timer0.

8.0   ANALOG-TO-DIGITAL CONVERTER (A/D) MODULE

The Analog-To-Digital (A/D) converter module has four analog inputs.

The A/D allows conversion of an analog input signal to a corresponding 8-bit digital number (refer to Application Note AN546 for use of A/D Converter). The output of the sample and hold is the input into the converter, which generates the result via successive approximation. The analog reference voltage is software selectable to either the device's positive supply voltage (V_{DD}) or the voltage level on the GP1/AN1/V_{REF} pin. The A/D converter has a unique feature of being able to operate while the device is in SLEEP mode.

The A/D module has three registers. These registers are:

- A/D Result Register (ADRES)
- A/D Control Register 0 (ADCON0)
- A/D Control Register 1 (ADCON1)

The ADCON0 Register, shown in Figure 8-1, controls the operation of the A/D module. The ADCON1 Register, shown in Figure 8-2, configures the functions of the port pins. The port pins can be configured as analog inputs (GP1 can also be a voltage reference) or as digital I/O.

> **Note 1:** If the port pins are configured as analog inputs (reset condition), reading the port (MOVF GPIO,W) results in reading '0's.
>
> **2:** Changing ADCON1 Register can cause the GPIF and INTF flags to be set in the INTCON Register. These interrupts should be disabled prior to modifying ADCON1.

REGISTER 8-1:   ADCON0 REGISTER (ADDRESS 1Fh)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ADCS1	ADCS0	reserved	CHS1	CHS0	GO/$\overline{DONE}$	reserved	ADON

bit7   bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as '0'
- n = Value at POR reset

bit 7-6: **ADCS<1:0>:** A/D Conversion Clock Select bits
00 = Fosc/2
01 = Fosc/8
10 = Fosc/32
11 = Frc (clock derived from an RC oscillation)

bit 5: **Reserved**

bit 4-3: **CHS<1:0>:** Analog Channel Select bits
00 = channel 0, (GP0/AN0)
01 = channel 1, (GP1/AN1)
10 = channel 2, (GP2/AN2)
11 = channel 3, (GP4/AN3)

bit 2: **GO/$\overline{DONE}$:** A/D Conversion Status bit
If ADON = 1
1 = A/D conversion in progress (setting this bit starts the A/D conversion)
0 = A/D conversion not in progress (this bit is automatically cleared by hardware when the A/D conversion is complete)

bit 1: **Reserved**

bit 0: **ADON:** A/D on bit
1 = A/D converter module is operating
0 = A/D converter module is shut off and consumes no operating current

8.5 A/D Operation During Sleep

The A/D module can operate during SLEEP mode. This requires that the A/D clock source be set to RC (ADCS<1:0> = 11). When the RC clock source is selected, the A/D module waits one instruction cycle before starting the conversion. This allows the SLEEP instruction to be executed, which eliminates all digital switching noise from the conversion. When the conversion is completed, the GO/$\overline{\text{DONE}}$ bit will be cleared, and the result loaded into the ADRES Register. If the A/D interrupt is enabled, the device will wake-up from SLEEP. If the A/D interrupt is not enabled, the A/D module will then be turned off, although the ADON bit will remain set.

When the A/D clock source is another clock option (not RC), a SLEEP instruction will cause the present conversion to be aborted and the A/D module to be turned off, though the ADON bit will remain set.

Turning off the A/D places the A/D module in its lowest current consumption state.

> **Note:** For the A/D module to operate in SLEEP, the A/D clock source must be set to RC (ADCS<1:0> = 11). To perform an A/D conversion in SLEEP, the GO/$\overline{\text{DONE}}$ bit must be set, followed by the SLEEP instruction.

8.6 A/D Accuracy/Error

The overall accuracy of the A/D is less than $\pm$ 1 LSb for VDD = 5V $\pm$ 10% and the analog VREF = VDD. This overall accuracy includes offset error, full scale error, and integral error. The A/D converter is monotonic over the full VDD range. The resolution and accuracy may be less when either the analog reference (VDD) is less than 5.0V or when the analog reference (VREF) is less than VDD.

The maximum pin leakage current is specified in the Device Data Sheet electrical specification, parameter #D060.

In systems where the device frequency is low, use of the A/D RC clock is preferred. At moderate to high frequencies, TAD should be derived from the device oscillator. TAD must not violate the minimum and should be $\leq$8 μs for preferred operation. This is because TAD, when derived from TOSC, is kept away from on-chip phase clock transitions. This reduces, to a large extent, the effects of digital switching noise. This is not possible with the RC derived clock. The loss of accuracy due to digital switching noise can be significant if many I/O pins are active.

In systems where the device will enter SLEEP mode after the start of the A/D conversion, the RC clock source selection is required. In this mode, the digital noise from the modules in SLEEP are stopped. This method gives high accuracy.

8.7 Effects of a Reset

A device reset forces all registers to their reset state. This forces the A/D module to be turned off, and any conversion is aborted. The value that is in the ADRES register is not modified for a Reset. The ADRES register will contain unknown data after a Power-on Reset.

8.8 Connection Considerations

If the input voltage exceeds the rail values (VSS or VDD) by greater than 0.2V, then the accuracy of the conversion is out of specification.

> **Note:** For the PIC12C67X, care must be taken when using the GP4 pin in A/D conversions due to its proximity to the OSC1 pin.

An external RC filter is sometimes added for anti-aliasing of the input signal. The R component should be selected to ensure that the total source impedance is kept under the 10 kΩ recommended specification. Any external components connected (via hi-impedance) to an analog input pin (capacitor, zener diode, etc.) should have very little leakage current at the pin.

8.9 Transfer Function

The ideal transfer function of the A/D converter is as follows: the first transition occurs when the analog input voltage (VAIN) is 1 LSb (or Analog VREF / 256) (Figure 8-3).

FIGURE 8-3: A/D TRANSFER FUNCTION
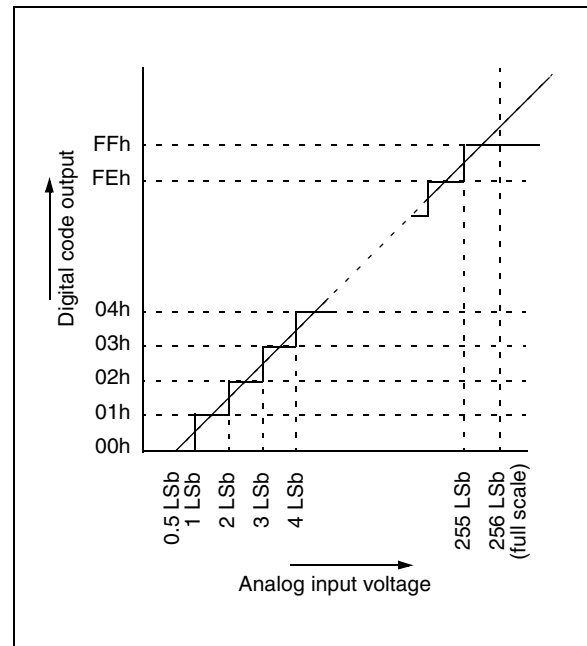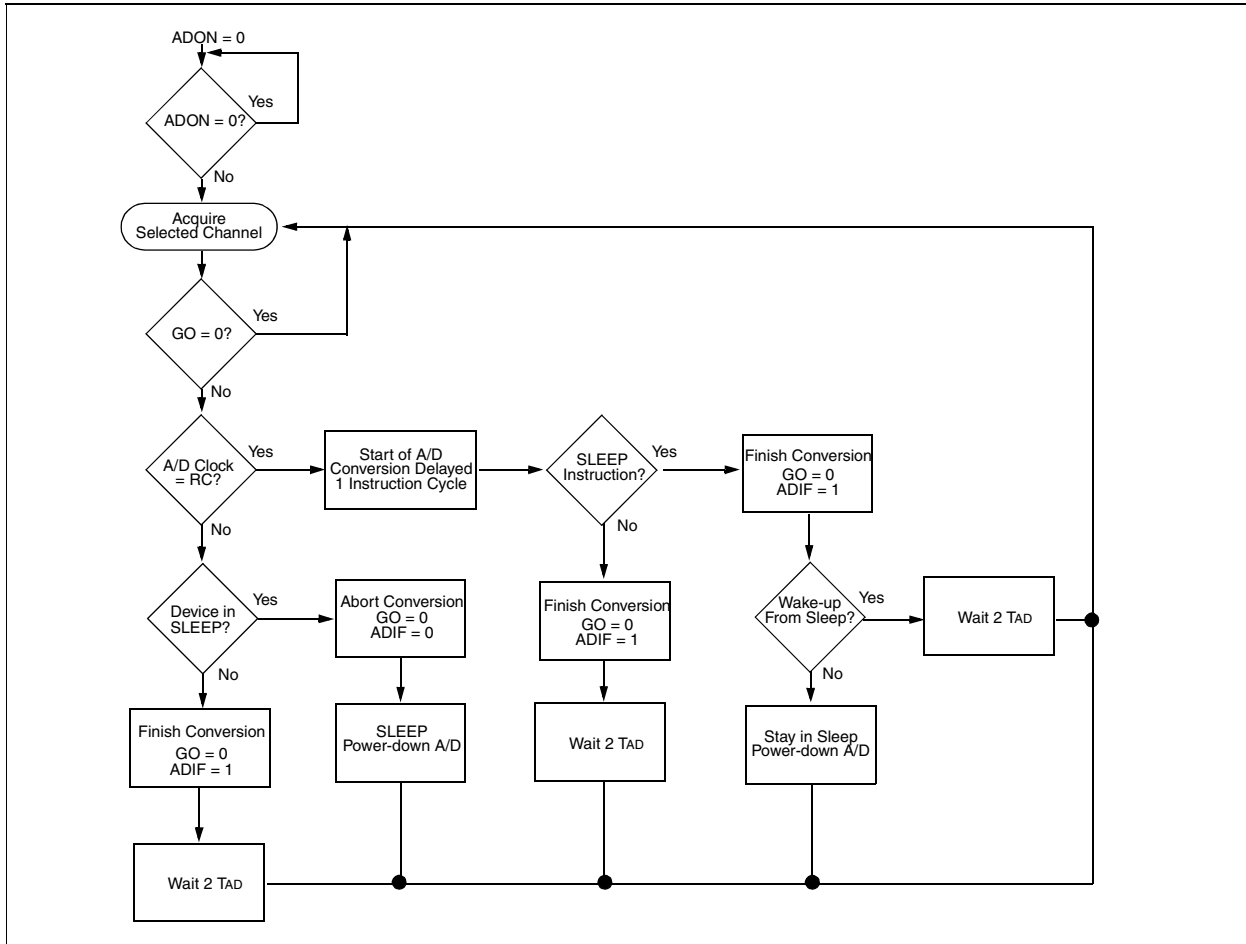
FIGURE 8-4: **FLOWCHART OF A/D OPERATION**

TABLE 8-2: **SUMMARY OF A/D 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
0Bh/8Bh	INTCON[1]	GIE	PEIE	T0IE	INTE	GPIE	T0IF	INTF	GPIF	0000 000x	0000 000u
0Ch	PIR1	—	ADIF	—	—	—	—	—	—	-0-- ----	-0-- ----
8Ch	PIE1	—	ADIE	—	—	—	—	—	—	-0-- ----	-0-- ----
1Eh	ADRES	A/D Result Register								xxxx xxxx	uuuu uuuu
1Fh	ADCON0	ADCS1	ADCS0	reserved	CHS1	CHS0	GO/$\overline{\text{DONE}}$	reserved	ADON	0000 0000	0000 0000
9Fh	ADCON1	—	—	—	—	—	PCFG2	PCFG1	PCFG0	---- -000	---- -000
05h	GPIO	SCL[2]	SDA[2]	GP5	GP4	GP3	GP2	GP1	GP0	11xx xxxx	11uu uuuu
85h	TRIS	—	—	TRIS5	TRIS4	TRIS3	TRIS2	TRIS1	TRIS0	--11 1111	--11 1111

Legend: x = unknown, u = unchanged, - = unimplemented read as '0'. Shaded cells are not used for A/D conversion.

Note 1: These registers can be addressed from either bank.

2: The SCL (GP7) and SDA (GP6) bits are unimplemented on the PIC12C671/672 and read as '0'.

FIGURE 9-10: EXTERNAL POWER-ON RESET CIRCUIT (FOR SLOW V_{DD} POWER-UP)

Note 1: External Power-on Reset circuit is required only if V_{DD} power-up slope is too slow. The diode D helps discharge the capacitor quickly when V_{DD} powers down.

2: R < 40 kΩ is recommended to make sure that voltage drop across R does not violate the device's electrical specification.

3: R1 = 100Ω to 1 kΩ will limit any current flowing into $\overline{\text{MCLR}}$ from external capacitor C, in the event of $\overline{\text{MCLR}}$/V_{PP} pin breakdown due to Electrostatic Discharge (ESD) or Electrical Overstress (EOS).

FIGURE 9-11: EXTERNAL BROWN-OUT PROTECTION CIRCUIT 1

Note 1: This circuit will activate reset when V_{DD} goes below (Vz + 0.7V), where Vz = Zener voltage.

2: Resistors should be adjusted for the characteristics of the transistor.
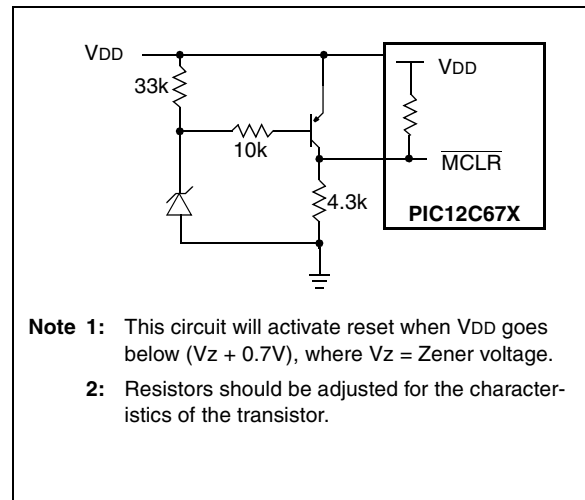
FIGURE 9-12: EXTERNAL BROWN-OUT PROTECTION CIRCUIT 2

Note 1: This brown-out circuit is less expensive, albeit less accurate. Transistor Q1 turns off when V_{DD} is below a certain level such that:

$$V_{DD} \cdot \frac{R1}{R1 + R2} = 0.7V$$

2: Resistors should be adjusted for the characteristics of the transistor.
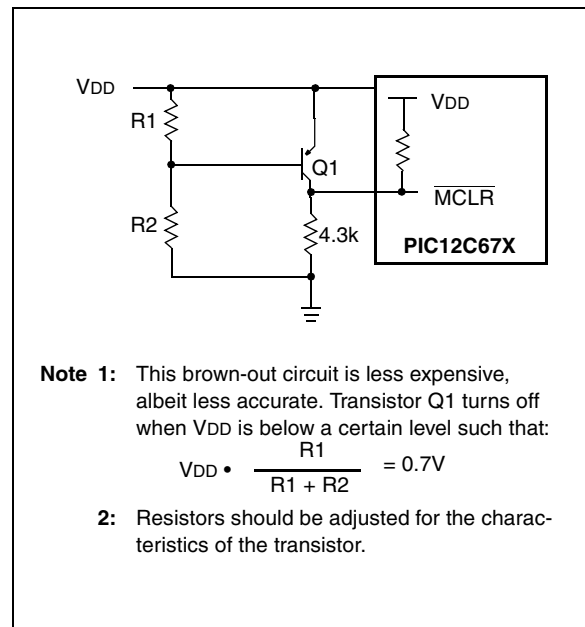
FIGURE 9-16: WAKE-UP FROM SLEEP THROUGH INTERRUPT

Note 1: XT, HS or LP oscillator mode assumed.

2: TOST = 1024TOSC (drawing not to scale) This delay will not be there for INTRC and EXTRC osc mode.

3: GIE = '1' assumed. In this case after wake- up, the processor jumps to the interrupt routine. If GIE = '0', execution will continue in-line.

4: CLKOUT is not available in XT, HS or LP osc modes, but shown here for timing reference.
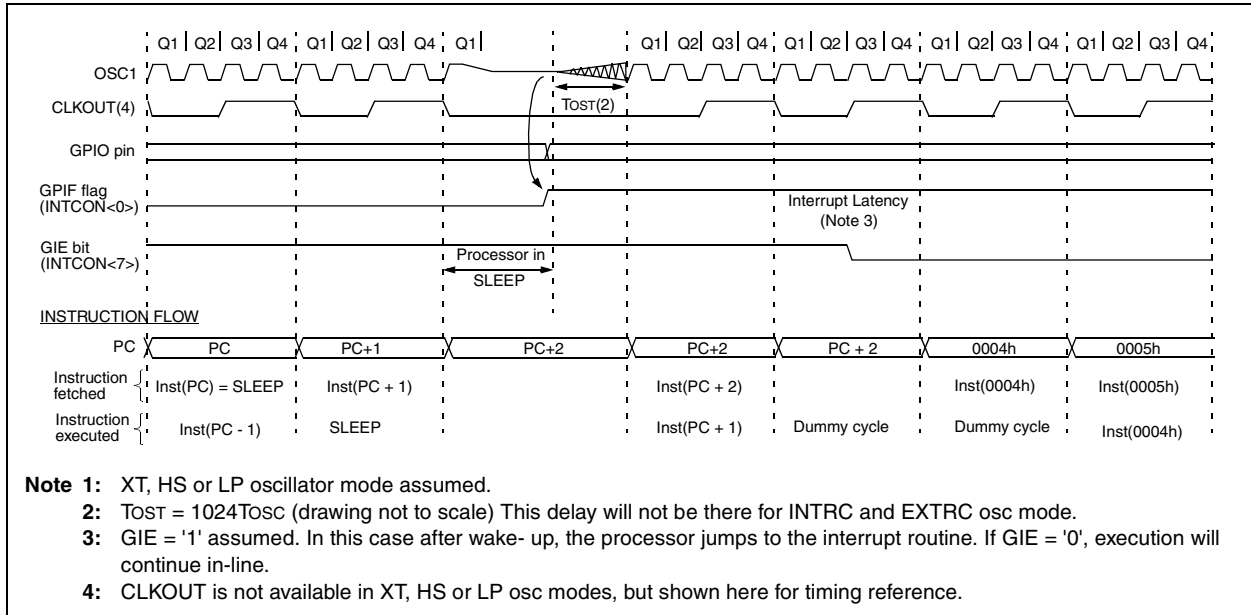
9.9   Program Verification/Code Protection

If the code protection bit(s) have not been programmed, the on-chip program memory can be read out for verification purposes.

> **Note:** Microchip does not recommend code protecting windowed devices.

9.10   ID Locations

Four memory locations (2000h - 2003h) 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. It is recommended that only the 4 least significant bits of the ID location are used.

9.11   In-Circuit Serial Programming

PIC12C67X microcontrollers 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 (clock) becomes the programming clock and GP0 (data) becomes the programming data. Both GP0 and GP1 are Schmitt Trigger inputs in this mode.

After reset, and if the device is placed into programming/verify mode, the program counter (PC) is at location 00h. 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 PIC12C67X Programming Specifications.

FIGURE 9-17: TYPICAL IN-CIRCUIT SERIAL PROGRAMMING CONNECTION
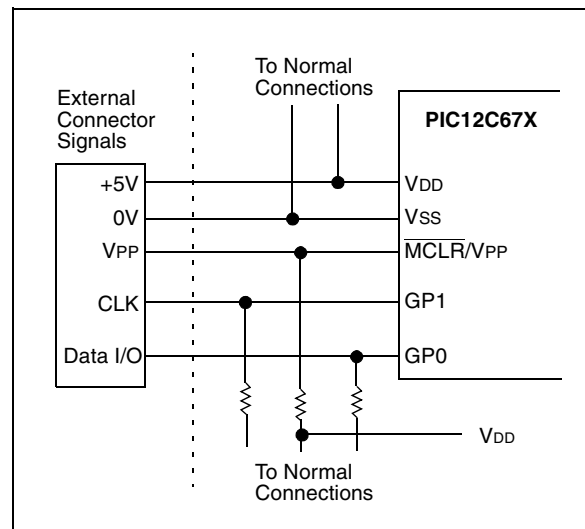
10.1 Special Function Registers as Source/Destination

The PIC12C67X's orthogonal instruction set allows read and write of all file registers, including special function registers. There are some special situations the user should be aware of:

10.1.1 STATUS AS DESTINATION

If an instruction writes to STATUS, the Z, C and DC bits may be set or cleared as a result of the instruction and overwrite the original data bits written. For example, executing `CLRF STATUS` will clear register STATUS, and then set the Z bit leaving `0000 0100b` in the register.

10.1.2 TRIS AS DESTINATION

Bit 3 of the TRIS register always reads as a '1' since GP3 is an input only pin. This fact can affect some read-modify-write operations on the TRIS register.

10.1.3 PCL AS SOURCE OR DESTINATION

Read, write or read-modify-write on PCL may have the following results:

Read PC: $PCL \rightarrow dest$

Write PCL: $PCLATH \rightarrow PCH$;
8-bit destination value $\rightarrow PCL$

Read-Modify-Write: $PCL \rightarrow ALU$ operand
$PCLATH \rightarrow PCH$;
8-bit result $\rightarrow PCL$

Where PCH = program counter high byte (not an addressable register), PCLATH = Program counter high holding latch, dest = destination, WREG or f.

10.1.4 BIT MANIPULATION

All bit manipulation instructions are done by first reading the entire register, operating on the selected bit and writing the result back (read-modify-write). The user should keep this in mind when operating on special function registers, such as ports.

PIC12C67X

10.2   Instruction Descriptions

ADDLW	Add Literal and W
Syntax:	[*label*] ADDLW k
Operands:	$0 \le k \le 255$
Operation:	$(W) + k \rightarrow (W)$
Status Affected:	C, DC, Z

Encoding:

11	111x	kkkk	kkkk

Description: The contents of the W register are added to the eight bit literal 'k' and the result is placed in the W register.

Words: 1

Cycles: 1

Example   ADDLW   0x15

Before Instruction
          W   =   0x10
After Instruction
          W   =   0x25

ADDWF	Add W and f
Syntax:	[*label*] ADDWF f,d
Operands:	$0 \le f \le 127$ $d \in [0,1]$
Operation:	$(W) + (f) \rightarrow (dest)$
Status Affected:	C, DC, Z

Encoding:

00	0111	dfff	ffff

Description: Add the contents of the W register with register 'f'. If 'd' is 0, the result is stored in the W register. If 'd' is 1, the result is stored back in register 'f'.

Words: 1

Cycles: 1

Example   ADDWF   FSR, 0

Before Instruction
          W   =   0x17
          FSR =   0xC2
After Instruction
          W   =   0xD9
          FSR =   0xC2

ANDLW	And Literal with W
Syntax:	[*label*] ANDLW k
Operands:	$0 \le k \le 255$
Operation:	$(W) .AND. (k) \rightarrow (W)$
Status Affected:	Z

Encoding:

11	1001	kkkk	kkkk

Description: The contents of W register are AND'ed with the eight bit literal 'k'. The result is placed in the W register.

Words: 1

Cycles: 1

Example   ANDLW   0x5F

Before Instruction
          W   =   0xA3
After Instruction
          W   =   0x03

ANDWF	AND W with f
Syntax:	[*label*] ANDWF f,d
Operands:	$0 \le f \le 127$ $d \in [0,1]$
Operation:	$(W) .AND. (f) \rightarrow (dest)$
Status Affected:	Z

Encoding:

00	0101	dfff	ffff

Description: AND the W register with register 'f'. If 'd' is 0, the result is stored in the W register. If 'd' is 1, the result is stored back in register 'f'.

Words: 1

Cycles: 1

Example   ANDWF   FSR, 1

Before Instruction
          W   =   0x17
          FSR =   0xC2
After Instruction
          W   =   0x17
          FSR =   0x02

PIC12C67X

BTFSS	Bit Test f, Skip if Set
Syntax:	[*label*] BTFSS f,b
Operands:	0 ≤ f ≤ 127 0 ≤ b < 7
Operation:	skip if (f<b>) = 1
Status Affected:	None

Encoding:

01	11bb	bfff	ffff

Description: If bit 'b' in register 'f' is '1', then the next instruction is skipped.
If bit 'b' is '1', then the next instruction fetched during the current instruction execution, is discarded and a NOP is executed instead, making this a 2 cycle instruction.

Words: 1

Cycles: 1(2)

Example

```
HERE    BTFSS  FLAG,1
FALSE   GOTO   PROCESS_CODE
TRUE    •
        •
        •
```

Before Instruction
    PC = address HERE
After Instruction
    if FLAG<1> = 0,
    PC =   address FALSE
    if FLAG<1> = 1,
    PC =   address TRUE

CALL	Call Subroutine
Syntax:	[*label*] CALL k
Operands:	0 ≤ k ≤ 2047
Operation:	(PC)+ 1→ TOS, k → PC<10:0>, (PCLATH<4:3>) → PC<12:11>
Status Affected:	None

Encoding:

10	0kkk	kkkk	kkkk

Description: Call Subroutine. First, return address (PC+1) is pushed onto the stack. The eleven bit immediate address is loaded into PC bits <10:0>. The upper bits of the PC are loaded from PCLATH. CALL is a two cycle instruction.

Words: 1

Cycles: 2

Example

```
HERE    CALL   THERE
```

Before Instruction
    PC = Address HERE
After Instruction
    PC = Address THERE
    TOS = Address HERE+1

CLRF	Clear f
Syntax:	[*label*] CLRF f
Operands:	0 ≤ f ≤ 127
Operation:	00h → (f) 1 → Z
Status Affected:	Z

Encoding:

00	0001	1fff	ffff

Description: The contents of register 'f' are cleared and the Z bit is set.

Words: 1

Cycles: 1

Example   CLRF   FLAG_REG

Before Instruction
    FLAG_REG = 0x5A
After Instruction
    FLAG_REG = 0x00
    Z   = 1

CLRW	Clear W
Syntax:	[*label*] CLRW
Operands:	None
Operation:	00h → (W) 1 → Z
Status Affected:	Z

Encoding:

00	0001	0000	0011

Description: W register is cleared. Zero bit (Z) is set.

Words: 1

Cycles: 1

Example   CLRW

Before Instruction
    W = 0x5A
After Instruction
    W = 0x00
    Z = 1

TABLE 11-1: DEVELOPMENT TOOLS FROM MICROCHIP

Category	Tool	PIC12CXXX	PIC14000	PIC16C5X	PIC16C6X	PIC16CXXX	PIC16F62X	PIC16C7X	PIC16C7XX	PIC16C8X	PIC16F8XX	PIC16C9XX	PIC17C4X	PIC17C7XX	PIC18CXX2	24CXX/25CXX/93CXX	HCSXXX	MCRFXXX	MCP2510
Software Tools	MPLAB® Integrated Development Environment	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓				
	MPLAB® C17 Compiler												✓	✓					
	MPLAB® C18 Compiler														✓				
	MPASM/MPLINK	✓	✓	✓	✓	✓	✓**	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		
Emulators	MPLAB®-ICE	✓	✓	✓	✓	✓	✓**	✓	✓	✓	✓	✓	✓	✓	✓				
	PICMASTER/PICMASTER-CE	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓	✓	✓					
	ICEPIC™ Low-Cost In-Circuit Emulator	✓		✓	✓	✓		✓	✓	✓		✓							
Debugger	MPLAB®-ICD In-Circuit Debugger				✓*			✓*			✓								
Programmers	PICSTART® Plus Low-Cost Universal Dev. Kit	✓	✓	✓	✓	✓	✓**	✓	✓	✓	✓	✓	✓	✓	✓				
	PRO MATE® II Universal Programmer	✓	✓	✓	✓	✓	✓**	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		
Demo Boards and Eval Kits	SIMICE	✓		✓															
	PICDEM-1			✓		✓		✓†		✓									
	PICDEM-2				✓†			✓†					✓		✓				
	PICDEM-3											✓							
	PICDEM-14A		✓																
	PICDEM-17													✓					
	KEELOQ® Evaluation Kit																✓		
	KEELOQ Transponder Kit																✓		
	microID™ Programmer's Kit																	✓	
	125 kHz microID Developer's Kit																	✓	
	125 kHz Anticollision microID Developer's Kit																	✓	
	13.56 MHz Anticollision microID Developer's Kit																	✓	
	MCP2510 CAN Developer's Kit																		✓

* Contact the Microchip Technology Inc. web site at www.microchip.com for information on how to use the MPLAB®-ICD In-Circuit Debugger (DV164001) with PIC16C62, 63, 64, 65, 72, 73, 74, 76, 77

** Contact Microchip Technology Inc. for availability date.

† Development tool is available on select devices.

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)
Parm No.	Characteristic	Sym	Min	Typ[1]	Max	Units	Conditions
	LP Oscillator Operating Frequency	FOSC	0		200	kHz	All temperatures
	INTRC/EXTRC Oscillator Operating Frequency		—		4[6]	MHz	All temperatures
	XT Oscillator Operating Frequency		0		4	MHz	All temperatures
	HS Oscillator Operating Frequency		0		10	MHz	All temperatures

*   These parameters are characterized but not tested.

Note 1: Data in 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 disabled.

   b) For standby current measurements, the conditions are the same, except that the device is in SLEEP mode.

4: For EXTRC osc configuration, current through REXT is not included. 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.

6: INTRC calibration value is for 4MHz nominal at 5V, 25°C.

TABLE 12-9: EEPROM MEMORY BUS TIMING REQUIREMENTS - PIC12CE673/674 ONLY.

AC Characteristics	Standard Operating Conditions (unless otherwise specified)
	Operating Temperature 0°C ≤ TA ≤ +70°C, Vcc = 3.0V to 5.5V (commercial)
	–40°C ≤ TA ≤ +85°C, Vcc = 3.0V to 5.5V (industrial)
	–40°C ≤ TA ≤ +125°C, Vcc = 4.5V to 5.5V (extended)
	Operating Voltage VDD range is described in Section 12.1

Parameter	Symbol	Min	Max	Units	Conditions
Clock frequency	FCLK	—	100	kHz	4.5V ≤ Vcc ≤ 5.5V (E Temp range)
		—	100		3.0V ≤ Vcc ≤ 4.5V
		—	400		4.5V ≤ Vcc ≤ 5.5V
Clock high time	THIGH	4000	—	ns	4.5V ≤ Vcc ≤ 5.5V (E Temp range)
		4000	—		3.0V ≤ Vcc ≤ 4.5V
		600	—		4.5V ≤ Vcc ≤ 5.5V
Clock low time	TLOW	4700	—	ns	4.5V ≤ Vcc ≤ 5.5V (E Temp range)
		4700	—		3.0V ≤ Vcc ≤ 4.5V
		1300	—		4.5V ≤ Vcc ≤ 5.5V
SDA and SCL rise time (Note 1)	TR	—	1000	ns	4.5V ≤ Vcc ≤ 5.5V (E Temp range)
		—	1000		3.0V ≤ Vcc ≤ 4.5V
		—	300		4.5V ≤ Vcc ≤ 5.5V
SDA and SCL fall time	TF	—	300	ns	(Note 1)
START condition hold time	THD:STA	4000	—	ns	4.5V ≤ Vcc ≤ 5.5V (E Temp range)
		4000	—		3.0V ≤ Vcc ≤ 4.5V
		600	—		4.5V ≤ Vcc ≤ 5.5V
START condition setup time	TSU:STA	4700	—	ns	4.5V ≤ Vcc ≤ 5.5V (E Temp range)
		4700	—		3.0V ≤ Vcc ≤ 4.5V
		600	—		4.5V ≤ Vcc ≤ 5.5V
Data input hold time	THD:DAT	0	—	ns	(Note 2)
Data input setup time	TSU:DAT	250	—	ns	4.5V ≤ Vcc ≤ 5.5V (E Temp range)
		250	—		3.0V ≤ Vcc ≤ 4.5V
		100	—		4.5V ≤ Vcc ≤ 5.5V
STOP condition setup time	TSU:STO	4000	—	ns	4.5V ≤ Vcc ≤ 5.5V (E Temp range)
		4000	—		3.0V ≤ Vcc ≤ 4.5V
		600	—		4.5V ≤ Vcc ≤ 5.5V
Output valid from clock (Note 2)	TAA	—	3500	ns	4.5V ≤ Vcc ≤ 5.5V (E Temp range)
		—	3500		3.0V ≤ Vcc ≤ 4.5V
		—	900		4.5V ≤ Vcc ≤ 5.5V
Bus free time: Time the bus must be free before a new transmission can start	TBUF	4700	—	ns	4.5V ≤ Vcc ≤ 5.5V (E Temp range)
		4700	—		3.0V ≤ Vcc ≤ 4.5V
		1300	—		4.5V ≤ Vcc ≤ 5.5V
Output fall time from VIH minimum to VIL maximum	TOF	20+0.1 CB	250	ns	(Note 1), CB ≤ 100 pF
Input filter spike suppression (SDA and SCL pins)	TSP	—	50	ns	(Notes 1, 3)
Write cycle time	TWC	—	4	ms	
Endurance		1M	—	cycles	25°C, VCC = 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 and avoid unintended generation of START or STOP conditions.
  3: The combined TSP and VHYS 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 ensured by characterization. For endurance estimates in a specific application, please consult the Total Endurance Model which can be obtained on Microchip's website.

Note the following details of the code protection feature on Microchip devices:

• Microchip products meet the specification contained in their particular Microchip Data Sheet.

• Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.

• There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.

• Microchip is willing to work with the customer who is concerned about the integrity of their code.

• Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as "unbreakable."

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

Trademarks

Microchip received ISO/TS-16949:2009 certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona; Gresham, Oregon and design centers in California and India. The Company's quality system processes and procedures are for its PIC® MCUs and dsPIC® DSCs, KEELOQ® code hopping devices, Serial EEPROMs, microperipherals, nonvolatile memory and analog products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001:2000 certified.

QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
═ ISO/TS 16949 ═