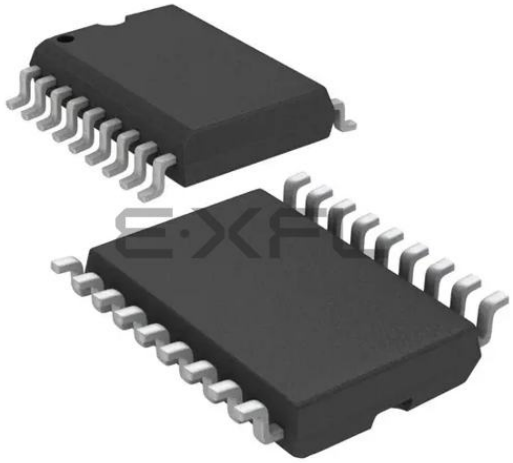


Welcome to E-XFL.COM

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	Brown-out Detect/Reset, POR, WDT
Number of I/O	13
Program Memory Size	3.5KB (2K x 14)
Program Memory Type	OTP
EEPROM Size	-
RAM Size	128 x 8
Voltage - Supply (Vcc/Vdd)	4V ~ 5.5V
Data Converters	A/D 4x8b
Oscillator Type	External
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Surface Mount
Package / Case	18-SOIC (0.295", 7.50mm Width)
Supplier Device Package	18-SOIC
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16c715t-04-so

Table of Contents

To Our Valued Customers

We constantly strive to improve the quality of all our products and documentation. We have spent an exceptional amount of time to ensure that these documents are correct. However, we realize that we may have missed a few things. If you find any information that is missing or appears in error, please use the reader response form in the back of this data sheet to inform us. We appreciate your assistance in making this a better document.

NOTES:

3.0 ARCHITECTURAL OVERVIEW

The high performance of the PIC16CXX family can be attributed to a number of architectural features commonly found in RISC microprocessors. To begin with, the PIC16CXX uses a Harvard architecture, in which, program and data are accessed from separate memories using separate buses. This improves bandwidth over traditional von Neumann architecture in which program and data are fetched from the same memory using the same bus. Separating program and data buses further allows instructions to be sized differently than the 8-bit wide data word. Instruction opcodes are 14-bits wide making it possible to have all single word instructions. A 14-bit wide program memory access bus fetches a 14-bit instruction in a single cycle. A two-stage pipeline overlaps fetch and execution of instructions (Example 3-1). Consequently, all instructions (35) execute in a single cycle (200 ns @ 20 MHz) except for program branches.

The table below lists program memory (EPROM) and data memory (RAM) for each PIC16C71X device.

Device	Program Memory	Data Memory
PIC16C710	512 x 14	36 x 8
PIC16C71	1K x 14	36 x 8
PIC16C711	1K x 14	68 x 8
PIC16C715	2K x 14	128 x 8

The PIC16CXX can directly or indirectly address its register files or data memory. All special function registers, including the program counter, are mapped in the data memory. The PIC16CXX has an 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 PIC16CXX simple yet efficient. In addition, the learning curve is reduced significantly.

PIC16CXX devices contain an 8-bit ALU and working register. The ALU is a general purpose arithmetic unit. It performs arithmetic and Boolean functions between the 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 working register (W register). The other operand is 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 $\overline{\text{borrow}}$ bit and a $\overline{\text{digit}}$ $\overline{\text{borrow}}$ out bit, respectively, in subtraction. See the SUBLW and SUBWF instructions for examples.

TABLE 4-2: PIC16C715 SPECIAL FUNCTION REGISTER SUMMARY (Cont.'d)

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR, PER	Value on all other resets (3)
Bank 1											
80h(1)	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	0000 0000
81h	OPTION	$\overline{RBPU}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
82h(1)	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	0000 0000
83h(1)	STATUS	IRP(4)	RP1(4)	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	000q quuu
84h(1)	FSR	Indirect data memory address pointer								xxxx xxxx	uuuu uuuu
85h	TRISA	—	—	PORTA Data Direction Register						--11 1111	--11 1111
86h	TRISB	PORTB Data Direction Register								1111 1111	1111 1111
87h	—	Unimplemented								—	—
88h	—	Unimplemented								—	—
89h	—	Unimplemented								—	—
8Ah(1,2)	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the PC					---0 0000	---0 0000
8Bh(1)	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
8Ch	PIE1	—	ADIE	—	—	—	—	—	—	-0-- ----	-0-- ----
8Dh	—	Unimplemented								—	—
8Eh	PCON	MPEEN	—	—	—	—	$\overline{PER}$	$\overline{POR}$	$\overline{BOR}$	u--- -1qq	u--- -1uu
8Fh	—	Unimplemented								—	—
90h	—	Unimplemented								—	—
91h	—	Unimplemented								—	—
92h	—	Unimplemented								—	—
93h	—	Unimplemented								—	—
94h	—	Unimplemented								—	—
95h	—	Unimplemented								—	—
96h	—	Unimplemented								—	—
97h	—	Unimplemented								—	—
98h	—	Unimplemented								—	—
99h	—	Unimplemented								—	—
9Ah	—	Unimplemented								—	—
9Bh	—	Unimplemented								—	—
9Ch	—	Unimplemented								—	—
9Dh	—	Unimplemented								—	—
9Eh	—	Unimplemented								—	—
9Fh	ADCON1	—	—	—	—	—	—	PCFG1	PCFG0	---- --00	---- --00

Legend: x = unknown, u = unchanged, q = value depends on condition, - = unimplemented read as '0'.
Shaded locations are unimplemented, read as '0'.

Note 1: These registers can be addressed from either bank.
2: The upper byte of the program counter is not directly accessible. PCLATH is a holding register for the PC<12:8> whose contents are transferred to the upper byte of the program counter.
3: Other (non power-up) resets include external reset through $\overline{MCLR}$ and Watchdog Timer Reset.
4: The IRP and RP1 bits are reserved on the PIC16C715, always maintain these bits clear.

4.2.2.1 STATUS REGISTER

Applicable Devices	710	71	711	715

The STATUS register, shown in Figure 4-7, contains the arithmetic status of the ALU, the RESET status and the bank select bits for data memory.

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

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

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

Note 1: For those devices that do not use bits IRP and RP1 (STATUS<7:6>), maintain these bits clear to ensure upward compatibility with future products.

Note 2: The C and DC bits operate as a $\overline{borrow}$ and $\overline{digit\ borrow}$ bit, respectively, in subtraction. See the SUBLW and SUBWF instructions for examples.

FIGURE 4-7: STATUS REGISTER (ADDRESS 03h, 83h)

R/W-0	R/W-0	R/W-0	R-1	R-1	R/W-x	R/W-x	R/W-x
IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C
bit7							bit0

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

bit 7: **IRP:** Register Bank Select bit (used for indirect addressing)
1 = Bank 2, 3 (100h - 1FFh)
0 = Bank 0, 1 (00h - FFh)

bit 6-5: **RP1:RP0:** Register Bank Select bits (used for direct addressing)
11 = Bank 3 (180h - 1FFh)
10 = Bank 2 (100h - 17Fh)
01 = Bank 1 (80h - FFh)
00 = Bank 0 (00h - 7Fh)
Each bank is 128 bytes

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

bit 3: **$\overline{PD}$:** 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/$\overline{borrow}$ bit (ADDWF, ADDLW, SUBLW, SUBWF instructions)(for $\overline{borrow}$ the polarity is reversed)
1 = A carry-out from the 4th low order bit of the result occurred
0 = No carry-out from the 4th low order bit of the result

bit 0: **C:** Carry/$\overline{borrow}$ bit (ADDWF, ADDLW, SUBLW, SUBWF instructions)
1 = A carry-out from the most significant bit of the result occurred
0 = No carry-out from the most significant bit of the result occurred
Note: For $\overline{borrow}$ the polarity is reversed. A subtraction is executed by adding the two's complement of the second operand. For rotate (RRF, RLF) instructions, this bit is loaded with either the high or low order bit of the source register.

5.0 I/O PORTS

Applicable Devices	710	71	711	715

Some pins for these I/O ports are multiplexed with an alternate function for the peripheral features on the device. In general, when a peripheral is enabled, that pin may not be used as a general purpose I/O pin.

5.1 PORTA and TRISA Registers

PORTA is a 5-bit latch.

The RA4/T0CKI pin is a Schmitt Trigger input and an open drain output. All other RA port pins have TTL input levels and full CMOS output drivers. All pins have data direction bits (TRIS registers) which can configure these pins as output or input.

Setting a TRISA register bit puts the corresponding output driver in a hi-impedance mode. Clearing a bit in the TRISA register puts the contents of the output latch on the selected pin(s).

Reading the PORTA register reads the status of the pins whereas writing to it will write to the port latch. All write operations are read-modify-write operations. Therefore a write to a port implies that the port pins are read, this value is modified, and then written to the port data latch.

Pin RA4 is multiplexed with the Timer0 module clock input to become the RA4/T0CKI pin.

Other PORTA pins are multiplexed with analog inputs and analog VREF input. The operation of each pin is selected by clearing/setting the control bits in the ADCON1 register (A/D Control Register1).

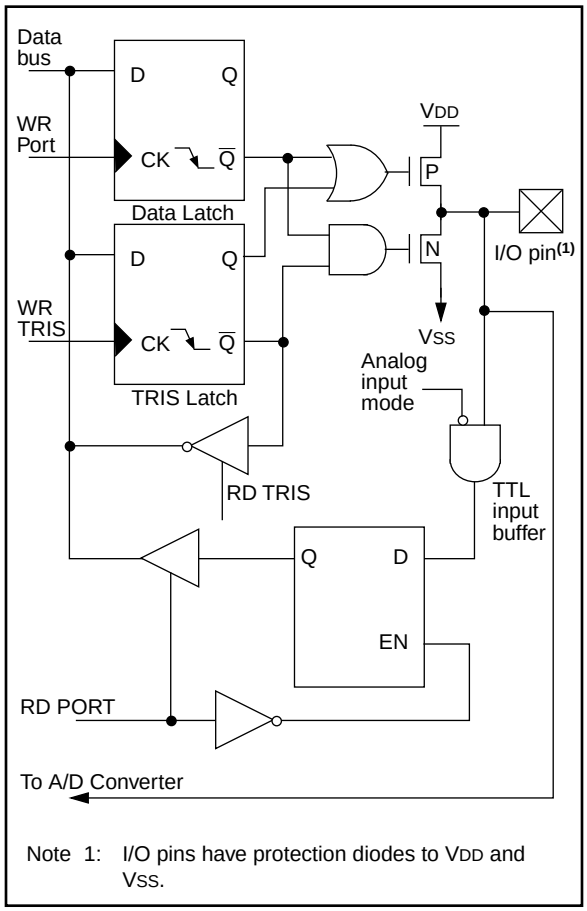
Note:	On a Power-on Reset, these pins are configured as analog inputs and read as '0'.

The TRISA register controls the direction of the RA pins, even when they are being used as analog inputs. The user must ensure the bits in the TRISA register are maintained set when using them as analog inputs.

EXAMPLE 5-1: INITIALIZING PORTA

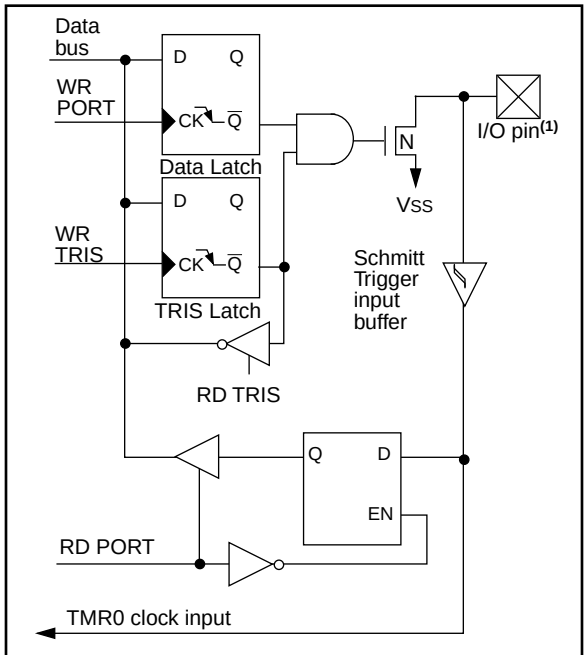
```
BCF     STATUS, RP0  ;
CLRF    PORTA        ; Initialize PORTA by
                     ; clearing output
                     ; data latches
BSF     STATUS, RP0  ; Select Bank 1
MOVLW   0xCF         ; Value used to
                     ; initialize data
                     ; direction
MOVWF   TRISA        ; Set RA<3:0> as inputs
                     ; RA<4> as outputs
                     ; TRISA<7:5> are always
                     ; read as '0'.
```

FIGURE 5-1: BLOCK DIAGRAM OF RA3:RA0 PINS



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

FIGURE 5-2: BLOCK DIAGRAM OF RA4/T0CKI PIN



Note 1: I/O pin has protection diodes to VSS only.

5.3 I/O Programming Considerations

5.3.1 BI-DIRECTIONAL I/O PORTS

Any instruction which writes, operates internally as a read followed by a write operation. The BCF and BSF instructions, for example, read the register into the CPU, execute the bit operation and write the result back to the register. Caution must be used when these instructions are applied to a port with both inputs and outputs defined. For example, a BSF operation on bit5 of PORTB will cause all eight bits of PORTB to be read into the CPU. Then the BSF operation takes place on bit5 and PORTB is written to the output latches. If another bit of PORTB is used as a bi-directional I/O pin (e.g., 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 to an output, the content of the data latch may now be unknown.

Reading the port register, reads the values of the port pins. Writing to the port register writes the value to the port latch. When using read-modify-write instructions (ex. BCF, BSF, etc.) on a port, the value of the port pins is read, the desired operation is done to this value, and this value is then written to the port latch.

Example 5-3 shows the effect of two sequential read-modify-write instructions on an I/O port.

EXAMPLE 5-3: READ-MODIFY-WRITE INSTRUCTIONS ON AN I/O PORT

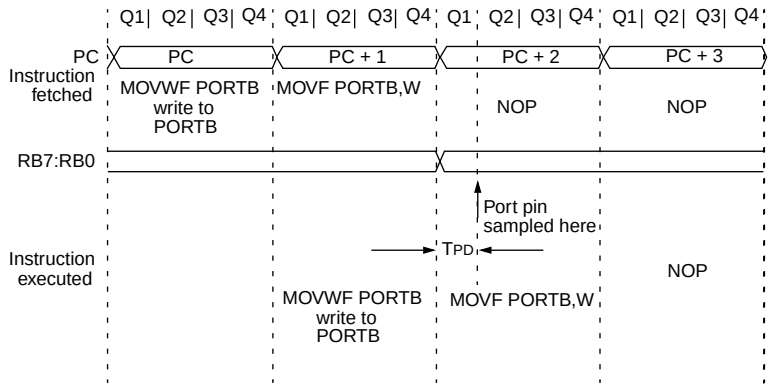
```
;Initial PORT settings: PORTB<7:4> Inputs
;                       PORTB<3:0> Outputs
;PORTB<7:6> have external pull-ups and are
;not connected to other circuitry
;
;                    PORT latch  PORT pins
;                    ----------  ---------
  BCF PORTB, 7      ; 01pp pppp    11pp pppp
  BCF PORTB, 6      ; 10pp pppp    11pp pppp
  BSF STATUS, RP0   ;
  BCF TRISB, 7      ; 10pp pppp    11pp pppp
  BCF TRISB, 6      ; 10pp pppp    10pp pppp
;
;Note that the user may have expected the
;pin values to be 00pp ppp. The 2nd BCF
;caused RB7 to be latched as the pin value
;(high).
```

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

5.3.2 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 5-6). 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 be such to 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.

FIGURE 5-6: SUCCESSIVE I/O OPERATION



Note:

This example shows a write to PORTB followed by a read from PORTB.

Note that:

data setup time = ($0.25T_{CY} - T_{PD}$)

where T_{CY} = instruction cycle
T_{PD} = propagation delay

Therefore, at higher clock frequencies, a write followed by a read may be problematic.

6.0 TIMER0 MODULE

Applicable Devices	710	71	711	715

The Timer0 module timer/counter has the following features:

- 8-bit timer/counter
- Readable and writable
- 8-bit software programmable prescaler
- Internal or external clock select
- Interrupt on overflow from FFh to 00h
- Edge select for external clock

Figure 6-1 is a simplified block diagram of the Timer0 module.

Timer mode is selected by clearing bit T0CS (OPTION<5>). In timer mode, the Timer0 module will increment every instruction cycle (without prescaler). If the TMR0 register is written, the increment is inhibited for the following two instruction cycles (Figure 6-2 and Figure 6-3). The user can work around this by writing an adjusted value to the TMR0 register.

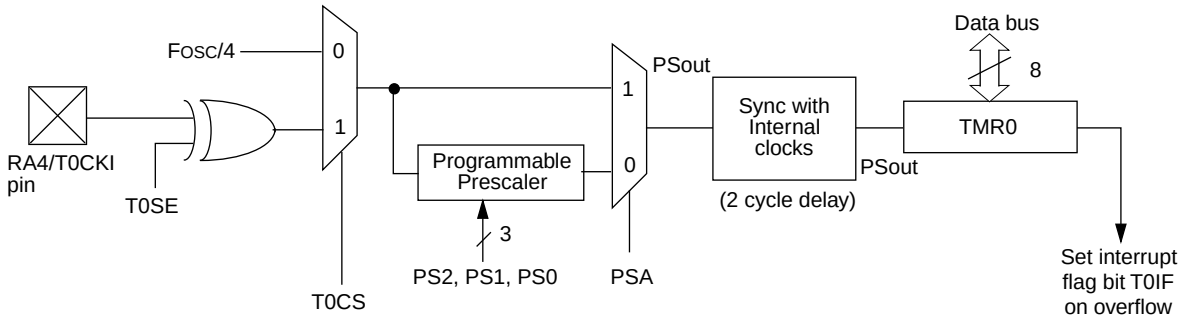
Counter mode is selected by setting bit T0CS (OPTION<5>). In counter mode, Timer0 will increment either on every rising or falling edge of pin RA4/T0CKI. The incrementing edge is determined by the Timer0 Source Edge Select bit T0SE (OPTION<4>). Clearing bit T0SE selects the rising edge. Restrictions on the external clock input are discussed in detail in Section 6.2.

The prescaler is mutually exclusively shared between the Timer0 module and the Watchdog Timer. The prescaler assignment is controlled in software by control bit PSA (OPTION<3>). Clearing bit PSA will assign the prescaler to the Timer0 module. The prescaler is not readable or writable. When the prescaler is assigned to the Timer0 module, prescale values of 1:2, 1:4, ..., 1:256 are selectable. Section 6.3 details the operation of the prescaler.

6.1 Timer0 Interrupt

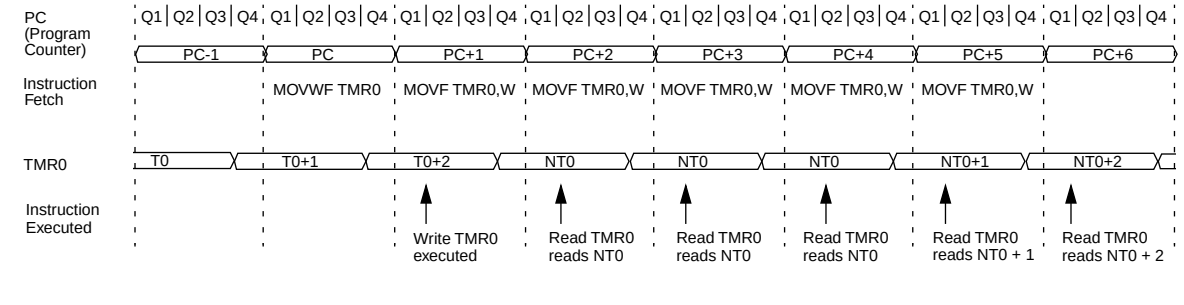
The TMR0 interrupt is generated when the TMR0 register overflows from FFh to 00h. This overflow sets bit T0IF (INTCON<2>). The interrupt can be masked by clearing bit T0IE (INTCON<5>). Bit T0IF must be cleared in software by the Timer0 module interrupt service routine before re-enabling this interrupt. The TMR0 interrupt cannot awaken the processor from SLEEP since the timer is shut off during SLEEP. See Figure 6-4 for Timer0 interrupt timing.

FIGURE 6-1: TIMER0 BLOCK DIAGRAM



Note 1: T0CS, T0SE, PSA, PS2:PS0 (OPTION<5:0>).
2: The prescaler is shared with Watchdog Timer (refer to Figure 6-6 for detailed block diagram).

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



7.4.1 FASTER CONVERSION - LOWER RESOLUTION TRADE-OFF

Not all applications require a result with 8-bits of resolution, but may instead require a faster conversion time. The A/D module allows users to make the trade-off of conversion speed to resolution. Regardless of the resolution required, the acquisition time is the same. To speed up the conversion, the clock source of the A/D module may be switched so that the TAD time violates the minimum specified time (see the applicable electrical specification). Once the TAD time violates the minimum specified time, all the following A/D result bits are not valid (see A/D Conversion Timing in the Electrical Specifications section.) The clock sources may only be switched between the three oscillator versions (cannot be switched from/to RC). The equation to determine the time before the oscillator can be switched is as follows:

Conversion time = $2T_{AD} + N \bullet T_{AD} + (8 - N)(2T_{OSC})$
Where: N = number of bits of resolution required.

Since the TAD is based from the device oscillator, the user must use some method (a timer, software loop, etc.) to determine when the A/D oscillator may be changed. Example 7-3 shows a comparison of time required for a conversion with 4-bits of resolution, versus the 8-bit resolution conversion. The example is for devices operating at 20 MHz and 16 MHz (The A/D clock is programmed for $32T_{OSC}$), and assumes that immediately after $6T_{AD}$, the A/D clock is programmed for $2T_{OSC}$.

The $2T_{OSC}$ violates the minimum TAD time since the last 4-bits will not be converted to correct values.

EXAMPLE 7-3: 4-BIT vs. 8-BIT CONVERSION TIMES

	Freq. (MHz)(1)	Resolution	
		4-bit	8-bit
TAD	20	1.6 µs	1.6 µs
	16	2.0 µs	2.0 µs
TOSC	20	50 ns	50 ns
	16	62.5 ns	62.5 ns
$2T_{AD} + N \bullet T_{AD} + (8 - N)(2T_{OSC})$	20	10 µs	16 µs
	16	12.5 µs	20 µs

Note 1: The PIC16C71 has a minimum TAD time of 2.0 µs.
All other PIC16C71X devices have a minimum TAD time of 1.6 µs.

GOTO **Unconditional Branch**

Syntax: [*label*] GOTO k

Operands: $0 \le k \le 2047$

Operation: $k \rightarrow PC<10:0>$
$PCLATH<4:3> \rightarrow PC<12:11>$

Status Affected: None

Encoding:

10	1kkk	kkkk	kkkk

Description: GOTO is an unconditional branch. The eleven bit immediate value is loaded into PC bits <10:0>. The upper bits of PC are loaded from PCLATH<4:3>. GOTO is a two cycle instruction.

Words: 1

Cycles: 2

Q Cycle Activity:

	Q1	Q2	Q3	Q4
1st Cycle	Decode	Read literal 'k'	Process data	Write to PC
2nd Cycle	NOP	NOP	NOP	NOP

Example

```
GOTO THERE
```

After Instruction
PC = Address THERE

INCF **Increment f**

Syntax: [*label*] INCF f,d

Operands: $0 \le f \le 127$
$d \in [0,1]$

Operation: (f) + 1 → (dest)

Status Affected: Z

Encoding:

00	1010	dfff	ffff

Description: The contents of register 'f' are incremented. If 'd' is 0 the result is placed in the W register. If 'd' is 1 the result is placed back in register 'f'.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register 'f'	Process data	Write to dest

Example

```
INCF    CNT, 1
```

Before Instruction
CNT = 0xFF
Z = 0
After Instruction
CNT = 0x00
Z = 1

NOP **No Operation**

Syntax: [*label*] NOP

Operands: None

Operation: No operation

Status Affected: None

Encoding:

00	0000	0xx0	0000

Description: No operation.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	NOP	NOP	NOP

Example

```
NOP
```

OPTION **Load Option Register**

Syntax: [*label*] OPTION

Operands: None

Operation: (W) → OPTION

Status Affected: None

Encoding:

00	0000	0110	0010

Description: The contents of the W register are loaded in the OPTION register. This instruction is supported for code compatibility with PIC16C5X products. Since OPTION is a readable/writable register, the user can directly address it.

Words: 1

Cycles: 1

Example

To maintain upward compatibility with future PIC16CXX products, do not use this instruction.

RETFIE **Return from Interrupt**

Syntax: [*label*] RETFIE

Operands: None

Operation: TOS → PC,
1 → GIE

Status Affected: None

Encoding:

00	0000	0000	1001

Description: Return from Interrupt. Stack is POPed and Top of Stack (TOS) is loaded in the PC. Interrupts are enabled by setting Global Interrupt Enable bit, GIE (INTCON<7>). This is a two cycle instruction.

Words: 1

Cycles: 2

Q Cycle Activity:

	Q1	Q2	Q3	Q4
1st Cycle	Decode	NOP	Set the GIE bit	Pop from the Stack
2nd Cycle	NOP	NOP	NOP	NOP

Example

```
RETFIE
```

After Interrupt
PC = TOS
GIE = 1

TABLE 10-1: DEVELOPMENT TOOLS FROM MICROCHIP

		PIC12C5XX	PIC14000	PIC16C5X	PIC16CXXX	PIC16C6X	PIC16C7XX	PIC16C8X	PIC16C9XX	PIC17C4X	PIC17C75X	24CXX 25CXX 93CXX	HCS200 HCS300 HCS301
Emulator Products	PICMASTER®/ PICMASTER-CE In-Circuit Emulator	✓	✓	✓	✓	✓	✓	✓	✓	✓	Available 3Q97		
	ICEPIC Low-Cost In-Circuit Emulator	✓		✓	✓	✓	✓	✓					
Software Tools	MPLAB™ Integrated Development Environment	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		
	MPLAB™ C Compiler	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		
	*fuzzy*TECH®-MP Explorer/Edition Fuzzy Logic Dev. Tool	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		
	MP-DriveWay™ Applications Code Generator			✓	✓	✓	✓	✓		✓			
	Total Endurance™ Software Model											✓	
Programmers	PICSTART® Lite Ultra Low-Cost Dev. Kit			✓		✓	✓	✓					
	PICSTART® Plus Low-Cost Universal Dev. Kit	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		
	PRO MATE® II Universal Programmer	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	KEELOQ® Programmer												✓
Demo Boards	SEEVAL® Designers Kit											✓	
	PICDEM-1			✓	✓			✓		✓			
	PICDEM-2					✓	✓						
	PICDEM-3								✓				
	KEELOQ® Evaluation Kit												✓

Applicable Devices	710	71	711	715

13.2 DC Characteristics: PIC16LC715-04 (Commercial, Industrial)

DC CHARACTERISTICS			Standard Operating Conditions (unless otherwise stated) Operating temperature 0°C ≤ TA ≤ +70°C (commercial) -40°C ≤ TA ≤ +85°C (industrial)				
Param No.	**Characteristic**	**Sym**	**Min**	**Typ†**	**Max**	**Units**	**Conditions**
D001	Supply Voltage	VDD	2.5	-	5.5	V	LP, XT, RC osc configuration (DC - 4 MHz)
D002*	RAM Data Retention Voltage (Note 1)	VDR	-	1.5	-	V	Device in SLEEP mode
D003	VDD start voltage to ensure internal Power-on Reset signal	VPOR	-	VSS	-	V	See section on Power-on Reset for details
D004*	VDD rise rate to ensure internal Power-on Reset signal	SVDD	0.05	-	-	V/ms	See section on Power-on Reset for details
D005	Brown-out Reset Voltage	BVDD	3.7	4.0	4.3	V	BODEN configuration bit is enabled
D010	Supply Current (Note 2)	IDD	-	2.0	3.8	mA	XT, RC osc configuration FOSC = 4 MHz, VDD = 3.0V (Note 4)
D010A			-	22.5	48	µA	LP osc configuration FOSC = 32 kHz, VDD = 3.0V, WDT disabled
D015	Brown-out Reset Current (Note 5)	ΔIBOR	-	300*	500	µA	BOR enabled VDD = 5.0V
D020	Power-down Current (Note 3)	IPD	-	7.5	30	µA	VDD = 3.0V, WDT enabled, -40°C to +85°C
D021			-	0.9	5	µA	VDD = 3.0V, WDT disabled, 0°C to +70°C
D021A			-	0.9	5	µA	VDD = 3.0V, WDT disabled, -40°C to +85°C
D023	Brown-out Reset Current (Note 5)	ΔIBOR	-	300*	500	µA	BOR enabled VDD = 5.0V

* These parameters are characterized but not tested.

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

2: The supply current is mainly a function of the operating voltage and frequency. Other factors such as I/O pin loading and switching rate, oscillator type, internal code execution pattern, and temperature also have an impact on the current consumption.
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 VDD
MCLR = VDD; WDT enabled/disabled as specified.

3: 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 and VSS.

4: For RC 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 Δ current is the additional current consumed when this peripheral is enabled. This current should be added to the base IDD or IPD measurement.

Applicable Devices	710	71	711	715

13.4 Timing Parameter Symbology

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

1. TppS2ppS
2. TppS

T			
F	Frequency	T	Time

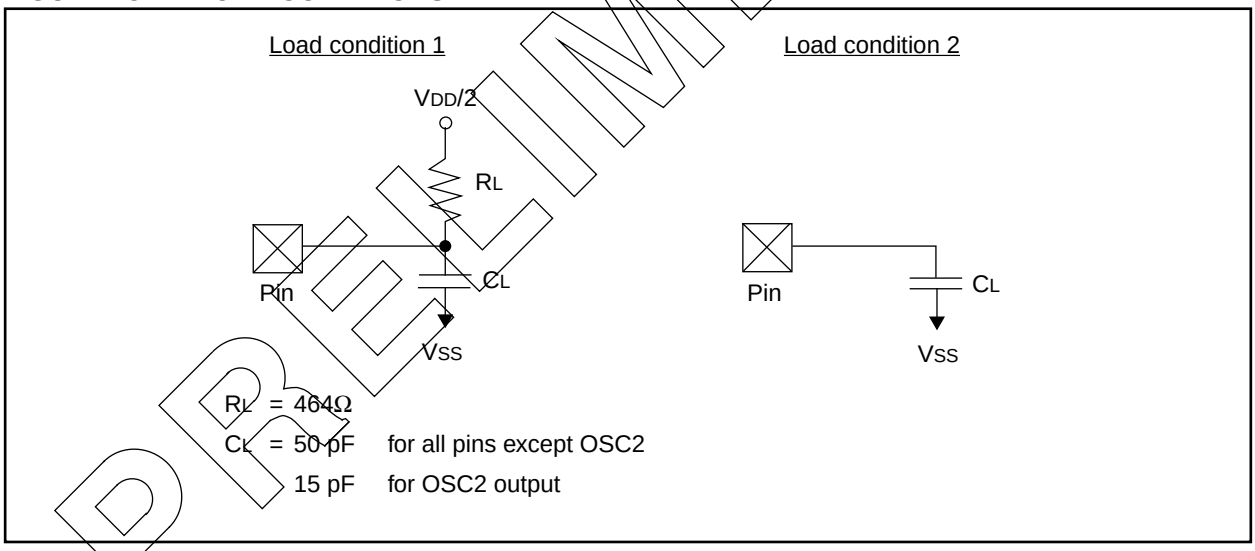
Lowercase letters (pp) and their meanings:

pp			
cc	CCP1	osc	OSC1
ck	CLKOUT	rd	$\overline{RD}$
cs	$\overline{CS}$	rw	$\overline{RD}$ or $\overline{WR}$
di	SDI	sc	SCK
do	SDO	ss	$\overline{SS}$
dt	Data in	t0	T0CKI
io	I/O port	t1	T1CKI
mc	$\overline{MCLR}$	wr	$\overline{WR}$

Uppercase letters and their meanings:

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

FIGURE 13-1: LOAD CONDITIONS



Applicable Devices	710	71	711	715

FIGURE 13-4: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER, AND POWER-UP TIMER TIMING

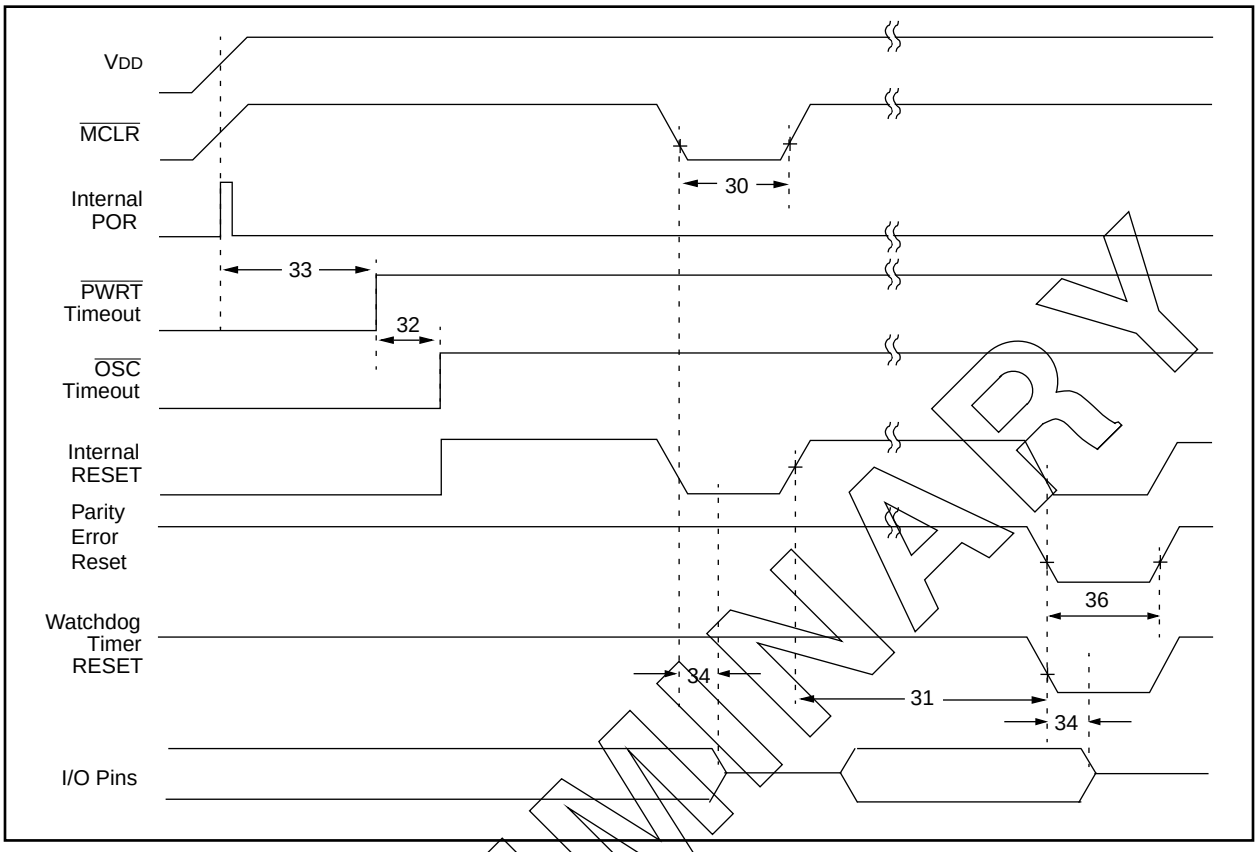


FIGURE 13-5: BROWN-OUT RESET TIMING

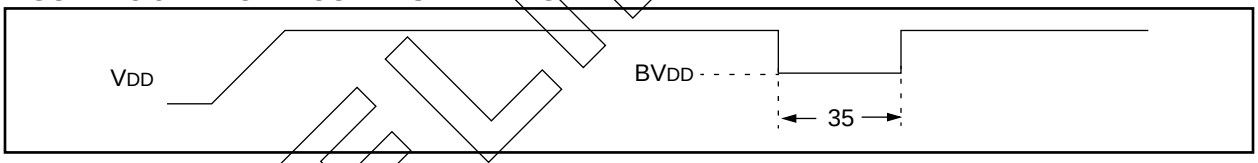


TABLE 13-4: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER, POWER-UP TIMER, AND BROWN-OUT RESET REQUIREMENTS

Parameter No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
30	TmcL	MCLR Pulse Width (low)	2	—	—	µs	VDD = 5V, -40˚C to +125˚C
31*	Twdt	Watchdog Timer Time-out Period (No Prescaler)	7	18	33	ms	VDD = 5V, -40˚C to +125˚C
32	Tost	Oscillation Start-up Timer Period	—	1024TOSC	—	—	TOSC = OSC1 period
33*	Tpwrt	Power up Timer Period	28	72	132	ms	VDD = 5V, -40˚C to +125˚C
34	TIOZ	I/O Hi-impedance from MCLR Low or Watchdog Timer Reset	—	—	2.1	µs	
35	TBOR	Brown-out Reset pulse width	100	—	—	µs	VDD ≤ BVDD (D005)
36	TPER	Parity Error Reset	—	TBD	—	µs	

* These parameters are characterized but not tested.

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

Applicable Devices	710	71	711	715

TABLE 15-6: A/D CONVERTER CHARACTERISTICS

Param No.	Sym	Characteristic		Min	Typ†	Max	Units	Conditions
A01	NR	Resolution		—	—	8 bits	bits	VREF = VDD = 5.12V, VSS ≤ VAIN ≤ VREF
A02	EABS	Absolute error	PIC16**C**71	—	—	< ±1	LSb	VREF = VDD = 5.12V, VSS ≤ VAIN ≤ VREF
			PIC16**LC**71	—	—	< ±2	LSb	VREF = VDD = 3.0V (Note 3)
A03	EIL	Integral linearity error	PIC16**C**71	—	—	< ±1	LSb	VREF = VDD = 5.12V, VSS ≤ VAIN ≤ VREF
			PIC16**LC**71	—	—	< ±2	LSb	VREF = VDD = 3.0V (Note 3)
A04	EDL	Differential linearity error	PIC16**C**71	—	—	< ±1	LSb	VREF = VDD = 5.12V, VSS ≤ VAIN ≤ VREF
			PIC16**LC**71	—	—	< ±2	LSb	VREF = VDD = 3.0V (Note 3)
A05	EFS	Full scale error	PIC16**C**71	—	—	< ±1	LSb	VREF = VDD = 5.12V, VSS ≤ VAIN ≤ VREF
			PIC16**LC**71	—	—	< ±2	LSb	VREF = VDD = 3.0V (Note 3)
A06	EOFF	Offset error	PIC16**C**71	—	—	< ±1	LSb	VREF = VDD = 5.12V, VSS ≤ VAIN ≤ VREF
			PIC16**LC**71	—	—	< ±2	LSb	VREF = VDD = 3.0V (Note 3)
A10	—	Monotonicity		—	guaranteed	—	—	VSS ≤ VAIN ≤ VREF
A20	VREF	Reference voltage		3.0V	—	VDD + 0.3	V	
A25	VAIN	Analog input voltage		VSS - 0.3	—	VREF	V	
A30	ZAIN	Recommended impedance of analog voltage source		—	—	10.0	kΩ	
A40	IAD	A/D conversion current (VDD)		—	180	—	μA	Average current consumption when A/D is on. (Note 1)
A50	IREF	VREF input current (Note 2)	PIC16**C**71	10	—	1000	μA	During VAIN acquisition. Based on differential of VHOLD to VAIN. To charge CHOLD see Section 7.1.
				—	—	40	μA	During A/D Conversion cycle
			PIC16**LC**71	—	—	1	mA	During VAIN acquisition. Based on differential of VHOLD to VAIN. To charge CHOLD see Section 7.1.
				—	—	10	μA	During A/D Conversion cycle

* These parameters are characterized but not tested.

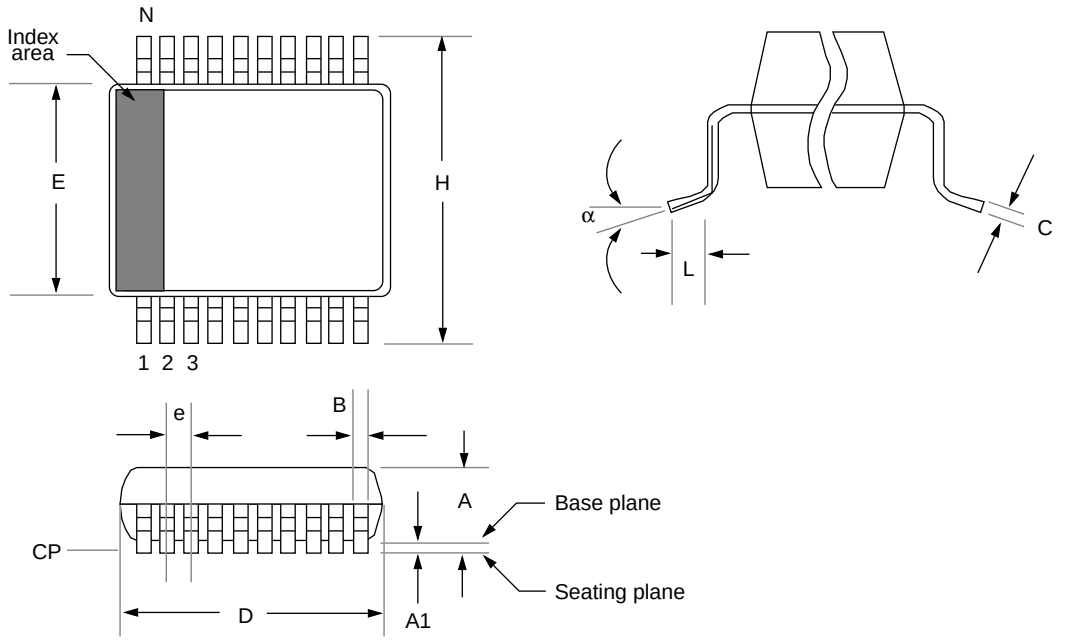
† Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: When A/D is off, it will not consume any current other than minor leakage current. The power-down current spec includes any such leakage from the A/D module.

2: VREF current is from RA3 pin or VDD pin, whichever is selected as reference input.

3: These specifications apply if VREF = 3.0V and if VDD ≥ 3.0V. VAIN must be between VSS and VREF.

17.4 20-Lead Plastic Surface Mount (SSOP - 209 mil Body 5.30 mm) (SS)



Package Group: Plastic SSOP						
	Millimeters			Inches		
Symbol	Min	Max	Notes	Min	Max	Notes
α	0°	8°		0°	8°	
A	1.730	1.990		0.068	0.078	
A1	0.050	0.210		0.002	0.008	
B	0.250	0.380		0.010	0.015	
C	0.130	0.220		0.005	0.009	
D	7.070	7.330		0.278	0.289	
E	5.200	5.380		0.205	0.212	
e	0.650	0.650	**Reference**	0.026	0.026	**Reference**
H	7.650	7.900		0.301	0.311	
L	0.550	0.950		0.022	0.037	
N	20	20		20	20	
CP	-	0.102		-	0.004	

Note 1: Dimensions D1 and E1 do not include mold protrusion. Allowable mold protrusion is 0.25m/m (0.010”) per side. D1 and E1 dimensions including mold mismatch.

2: Dimension “b” does not include Dambar protrusion, allowable Dambar protrusion shall be 0.08m/m (0.003”)max.

3: This outline conforms to JEDEC MS-026.

ON-LINE SUPPORT

Microchip provides two methods of on-line support. These are the Microchip BBS and the Microchip World Wide Web (WWW) site.

Use Microchip's Bulletin Board Service (BBS) to get current information and help about Microchip products. Microchip provides the BBS communication channel for you to use in extending your technical staff with microcontroller and memory experts.

To provide you with the most responsive service possible, the Microchip systems team monitors the BBS, posts the latest component data and software tool updates, provides technical help and embedded systems insights, and discusses how Microchip products provide project solutions.

The web site, like the BBS, is used by Microchip as a means to make files and information easily available to customers. To view the site, the user must have access to the Internet and a web browser, such as Netscape or Microsoft Explorer. Files are also available for FTP download from our FTP site.

Connecting to the Microchip Internet Web Site

The Microchip web site is available by using your favorite Internet browser to attach to:

www.microchip.com

The file transfer site is available by using an FTP service to connect to:

ftp://ftp.futureone.com/pub/microchip

The web site and file transfer site provide a variety of services. Users may download files for the latest Development Tools, Data Sheets, Application Notes, User's Guides, Articles and Sample Programs. A variety of Microchip specific business information is also available, including listings of Microchip sales offices, distributors and factory representatives. Other data available for consideration is:

- Latest Microchip Press Releases
- Technical Support Section with Frequently Asked Questions
- Design Tips
- Device Errata
- Job Postings
- Microchip Consultant Program Member Listing
- Links to other useful web sites related to Microchip Products

Connecting to the Microchip BBS

Connect worldwide to the Microchip BBS using either the Internet or the CompuServe® communications network.

Internet:

You can telnet or ftp to the Microchip BBS at the address: **mchipbbs.microchip.com**

CompuServe Communications Network:

When using the BBS via the Compuserve Network, in most cases, a local call is your only expense. The Microchip BBS connection does not use CompuServe membership services, therefore you do not need CompuServe membership to join Microchip's BBS. There is no charge for connecting to the Microchip BBS.

The procedure to connect will vary slightly from country to country. Please check with your local CompuServe agent for details if you have a problem. CompuServe service allow multiple users various baud rates depending on the local point of access.

The following connect procedure applies in most locations.

1. Set your modem to 8-bit, No parity, and One stop (8N1). This is not the normal CompuServe setting which is 7E1.
2. Dial your local CompuServe access number.
3. Depress the **<Enter>** key and a garbage string will appear because CompuServe is expecting a 7E1 setting.
4. Type +, depress the **<Enter>** key and "`Host Name:`" will appear.
5. Type MCHIPBBS, depress the **<Enter>** key and you will be connected to the Microchip BBS.

In the United States, to find the CompuServe phone number closest to you, set your modem to 7E1 and dial (800) 848-4480 for 300-2400 baud or (800) 331-7166 for 9600-14400 baud connection. After the system responds with "`Host Name:`", type `NETWORK`, depress the **<Enter>** key and follow CompuServe's directions.

For voice information (or calling from overseas), you may call (614) 723-1550 for your local CompuServe number.

Microchip regularly uses the Microchip BBS to distribute technical information, application notes, source code, errata sheets, bug reports, and interim patches for Microchip systems software products. For each SIG, a moderator monitors, scans, and approves or disapproves files submitted to the SIG. No executable files are accepted from the user community in general to limit the spread of computer viruses.

Systems Information and Upgrade Hot Line

The Systems Information and Upgrade Line provides system users a listing of the latest versions of all of Microchip's development systems software products. Plus, this line provides information on how customers can receive any currently available upgrade kits.The Hot Line Numbers are:

1-800-755-2345 for U.S. and most of Canada, and

1-602-786-7302 for the rest of the world.

970301

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.

DNV Certification, Inc. USA — ANSI-RAB QMS ACCREDITED — DNV MSC The Netherlands Accredited by the RvA — DNV — ISO 9001 / QS-9000 REGISTERED FIRM

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.