

## 8-Bit CMOS Microcontrollers with A/D Converter and Capture/Compare/PWM

### Devices included in this Data Sheet:

- PIC16C712 • PIC16C716

### Microcontroller Core Features:

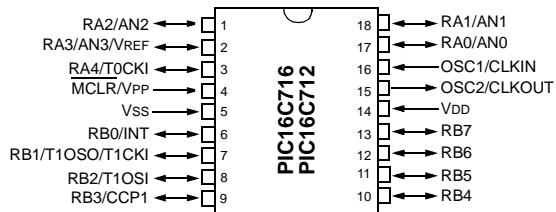
- High-performance RISC CPU
- Only 35 single-word instructions to learn
- All single-cycle instructions except for program branches which are two cycle
- Operating speed: DC – 20 MHz clock input  
DC – 200 ns instruction cycle

Device	Program Memory	Data Memory
PIC16C712	1K	128
PIC16C716	2K	128

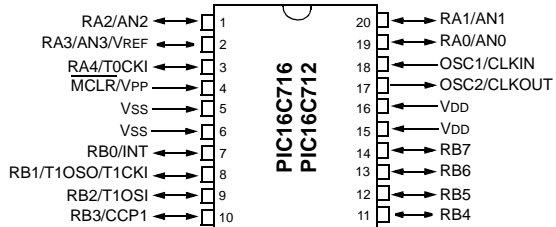
- Interrupt capability (up to 7 internal/external interrupt sources)
- Eight-level deep hardware stack
- Direct, Indirect and Relative Addressing modes
- Power-on Reset (POR)
- Power-up Timer (PWRT) and Oscillator Start-up Timer (OST)
- Watchdog Timer (WDT) with its own on-chip RC oscillator for reliable operation
- Brown-out detection circuitry for Brown-out Reset (BOR)
- Programmable code-protection
- Power-saving Sleep mode
- Selectable oscillator options
- Low-power, high-speed CMOS EPROM technology
- Fully static design
- In-Circuit Serial Programming™ (ICSP™)
- Wide operating voltage range: 2.5V to 5.5V
- High Sink/Source Current 25/25 mA
- Commercial, Industrial and Extended temperature ranges
- Low-power consumption:
  - < 2 mA @ 5V, 4 MHz
  - 22.5 µA, typical @ 3V, 32 kHz
  - < 1 µA, typical standby current

### Pin Diagrams

#### 18-pin PDIP, SOIC, Windowed Cerdip



#### 20-pin SSOP



### Peripheral Features:

- Timer0: 8-bit timer/counter with 8-bit prescaler
- Timer1: 16-bit timer/counter with prescaler can be incremented during Sleep via external crystal/clock
- Timer2: 8-bit timer/counter with 8-bit period register, prescaler and postscaler
- Capture, Compare, PWM module
- Capture is 16-bit, max. resolution is 12.5 ns, Compare is 16-bit, max. resolution is 200 ns, PWM maximum resolution is 10-bit
- 8-bit multi-channel Analog-to-Digital converter

# PIC16C712/716

Key Features PIC® Mid-Range Reference Manual (DS33023)	PIC16C712	PIC16C716
Operating Frequency	DC – 20 MHz	DC – 20 MHz
Resets (and Delays)	POR, BOR (PWRT, OST)	POR, BOR (PWRT, OST)
Program Memory (14-bit words)	1K	2K
Data Memory (bytes)	128	128
Interrupts	7	7
I/O Ports	Ports A,B	Ports A,B
Timers	3	3
Capture/Compare/PWM modules	1	1
8-bit Analog-to-Digital Module	4 input channels	4 input channels

## PIC16C7XX FAMILY OF DEVICES

		PIC16C710	PIC16C71	PIC16C711	PIC16C712	PIC16C715	PIC16C716	PIC16C72A	PIC16C73B
<b>Clock</b>	Maximum Frequency of Operation (MHz)	20	20	20	20	20	20	20	20
	<b>Memory</b>								
	EPROM Program Memory (x14 words)	512	1K	1K	1K	2K	2K	2K	4K
	Data Memory (bytes)	36	36	68	128	128	128	128	192
<b>Peripherals</b>	Timer Module(s)	TMR0	TMR0	TMR0	TMR0 TMR1 TMR2	TMR0	TMR0 TMR1 TMR2	TMR0 TMR1 TMR2	TMR0 TMR1 TMR2
	Capture/Compare/PWM Module(s)	—	—	—	1	—	1	1	2
	Serial Port(s) (SPI™/I <sup>2</sup> C™, USART)	—	—	—	—	—	—	SPI/I <sup>2</sup> C	SPI/I <sup>2</sup> C, USART
	A/D Converter (8-bit) Channels	4	4	4	4	4	4	5	5
<b>Features</b>	Interrupt Sources	4	4	4	7	4	7	8	11
	I/O Pins	13	13	13	13	13	13	22	22
	Voltage Range (Volts)	2.5-6.0	3.0-6.0	2.5-6.0	2.5-5.5	2.5-5.5	2.5-5.5	2.5-5.5	2.5-5.5
	In-Circuit Serial Programming™	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
	Brown-out Reset	Yes	—	Yes	Yes	Yes	Yes	Yes	Yes
Packages	18-pin DIP, SOIC; 20-pin SSOP	18-pin DIP, SOIC	18-pin DIP, SOIC; 20-pin SSOP	18-pin DIP, SOIC; 20-pin SSOP	18-pin DIP, SOIC; 20-pin SSOP	18-pin DIP, SOIC; 20-pin SSOP	18-pin DIP, SOIC; 20-pin SSOP	28-pin SDIP, SOIC, SSOP	28-pin SDIP, SOIC

## Table of Contents

1.0	Device Overview .....	5
2.0	Memory Organization .....	9
3.0	I/O Ports .....	21
4.0	Timer0 Module .....	29
5.0	Timer1 Module .....	31
6.0	Timer2 Module .....	36
7.0	Capture/Compare/PWM (CCP) Module(s) .....	39
8.0	Analog-to-Digital Converter (A/D) Module .....	45
9.0	Special Features of the CPU .....	51
10.0	Instruction Set Summary .....	67
11.0	Development Support .....	69
12.0	Electrical Characteristics .....	73
13.0	Packaging Information .....	89
	Revision History .....	95
	Conversion Considerations .....	95
	Migration from Base-line to Mid-Range Devices .....	95
	Index .....	97
	PIC16C712/716 Product Identification System .....	103

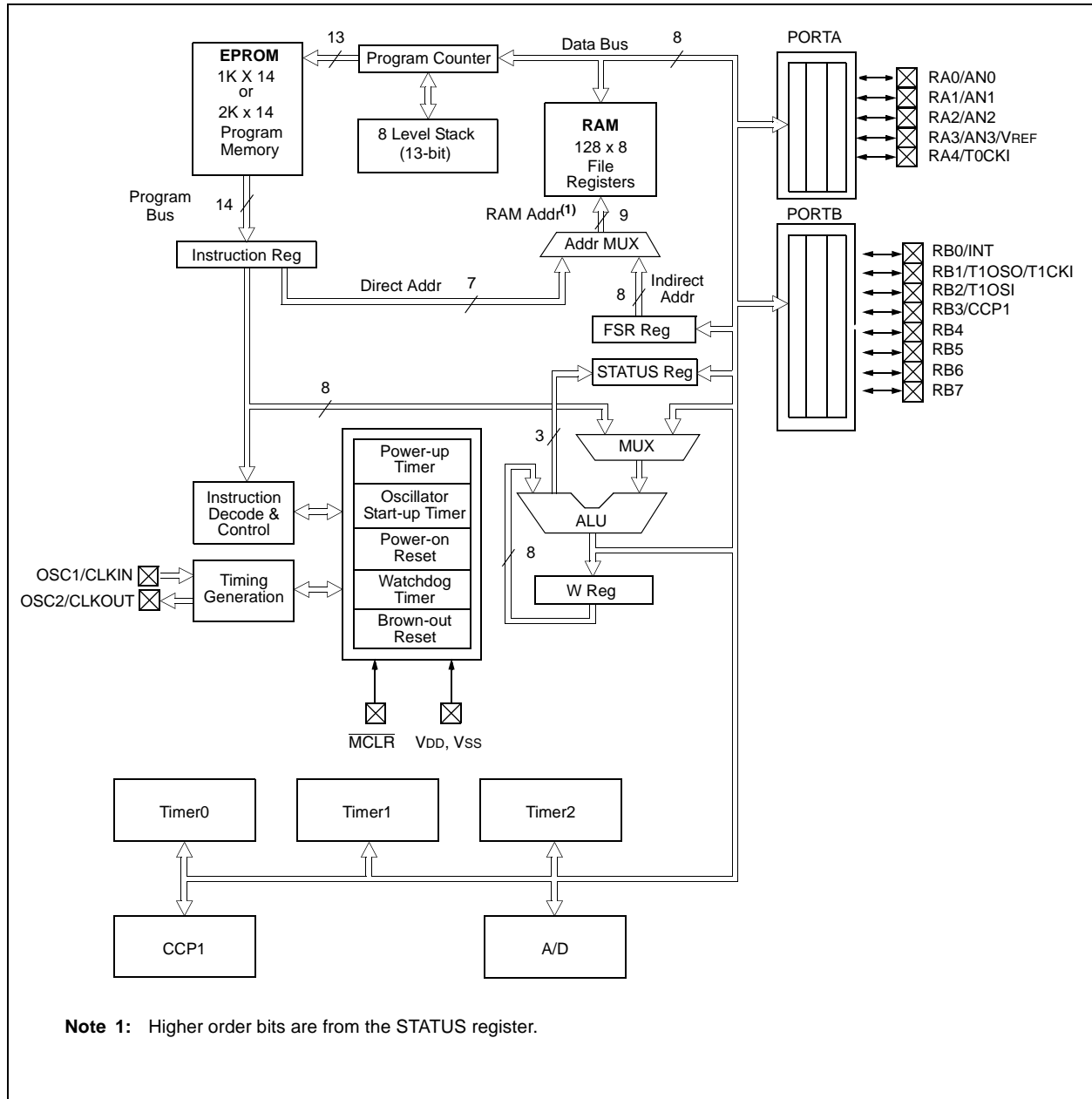
## 1.0 DEVICE OVERVIEW

This document contains device-specific information. Additional information may be found in the PIC® Mid-Range Reference Manual, (DS33023), which may be obtained from your local Microchip Sales Representative or downloaded from the Microchip web site. The Reference Manual should be considered a complementary document to this data sheet, and is highly recommended reading for a better understanding of the device architecture and operation of the peripheral modules.

There are two devices (PIC16C712, PIC16C716) covered by this data sheet.

Figure 1-1 is the block diagram for both devices. The pinouts are listed in Table 1-1.

**FIGURE 1-1: PIC16C712/716 BLOCK DIAGRAM**





# PIC16C712/716

**TABLE 1-1: PIC16C712/716 PINOUT DESCRIPTION (CONTINUED)**

Pin Name	PIC16C712/716		Pin Type	Buffer Type	Description
	DIP, SOIC	SSOP			
RB0/INT RB0 INT	6	7	I/O I	TTL ST	PORTB is a bidirectional I/O port. PORTB can be software programmed for internal weak pull-ups on all inputs.  Digital I/O External Interrupt
RB1/T1OSO/T1CKI RB1 T1OSO  T1CKI	7	8	I/O O  I	TTL —  ST	Digital I/O Timer1 oscillator output. Connects to crystal in oscillator mode. Timer1 external clock input.
RB2/T1OSI RB2 T1OSI	8	9	I/O I	TTL —	Digital I/O Timer1 oscillator input. Connects to crystal in oscillator mode.
RB3/CCP1 RB3 CCP1	9	10	I/O I/O	TTL ST	Digital I/O Capture1 input, Compare1 output, PWM1 output.
RB4	10	12	I/O	TTL	Digital I/O Interrupt on change pin.
RB5	11	12	I/O	TTL	Digital I/O Interrupt on change pin.
RB6	12	13	I/O  I	TTL  ST	Digital I/O Interrupt on change pin. ICSP programming clock.
RB7	13	14	I/O  I/O	TTL  ST	Digital I/O Interrupt on change pin. ICSP programming data.
VSS	5	5, 6	P	—	Ground reference for logic and I/O pins.
VDD	14	15, 16	P	—	Positive supply for logic and I/O pins.

**Legend:** TTL = TTL-compatible input    CMOS = CMOS compatible input or output  
 ST = Schmitt Trigger input with CMOS levels  
 OD = Open drain output  
 SM = SMBus compatible input. An external resistor is required if this pin is used as an output  
 NPU = N-channel pull-up    PU = Weak internal pull-up  
 No-P diode = No P-diode to VDD AN = Analog input or output  
 I = input    O = output  
 P = Power    L = LCD Driver

## 2.0 MEMORY ORGANIZATION

There are two memory blocks in each of these PIC<sup>®</sup> microcontroller devices. Each block (Program Memory and Data Memory) has its own bus so that concurrent access can occur.

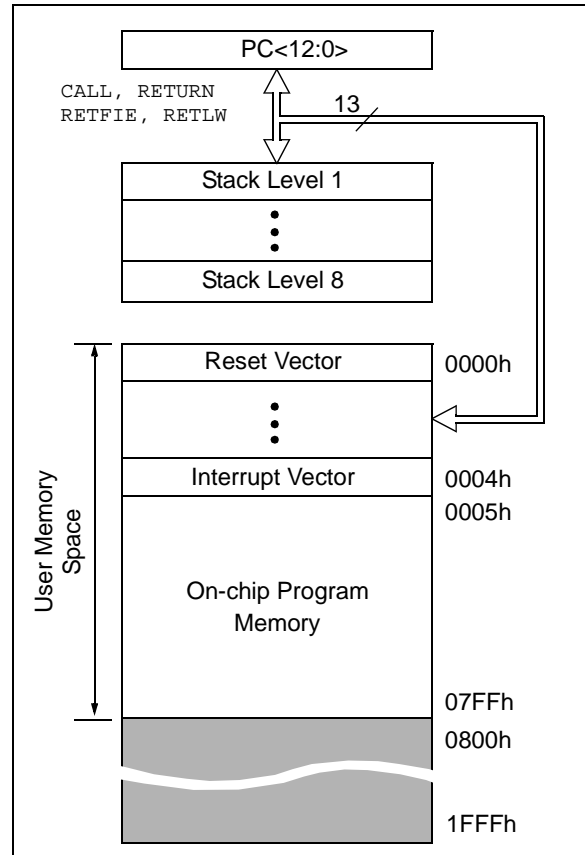
Additional information on device memory may be found in the PIC<sup>®</sup> Mid-Range Reference Manual, (DS33023).

### 2.1 Program Memory Organization

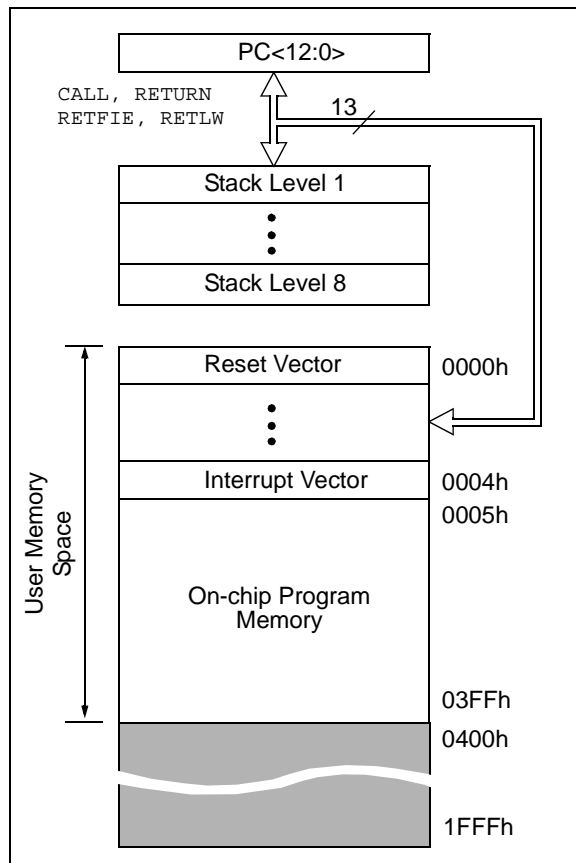
The PIC16C712/716 has a 13-bit Program Counter (PC) capable of addressing an 8K x 14 program memory space. PIC16C712 has 1K x 14 words of program memory and PIC16C716 has 2K x 14 words of program memory. Accessing a location above the physically implemented address will cause a wraparound.

The Reset vector is at 0000h and the interrupt vector is at 0004h.

**FIGURE 2-2: PROGRAM MEMORY MAP AND STACK OF PIC16C716**



**FIGURE 2-1: PROGRAM MEMORY MAP AND STACK OF THE PIC16C712**



# PIC16C712/716

## 2.2 Data Memory Organization

The data memory is partitioned into multiple banks which contain the General Purpose Registers and the Special Function Registers. Bits RP1 and RP0 are the bank select bits.

RP1 <sup>(1)</sup>	RP0
--------------------	-----

 (STATUS<6:5>)

- = 00 → Bank 0
- = 01 → Bank 1
- = 10 → Bank 2 (not implemented)
- = 11 → Bank 3 (not implemented)

**Note 1:** Maintain this bit clear to ensure upward compatibility with future products.

Each bank extends up to 7Fh (128 bytes). The lower locations of each bank are reserved for the Special Function Registers. Above the Special Function Registers are General Purpose Registers, implemented as static RAM. All implemented banks contain Special Function Registers. Some “high use” Special Function Registers from one bank may be mirrored in another bank for code reduction and quicker access.

### 2.2.1 GENERAL PURPOSE REGISTER FILE

The register file can be accessed either directly, or indirectly through the File Select Register FSR (see **Section 2.5 “Indirect Addressing, INDF and FSR Registers”**).

**FIGURE 2-3: REGISTER FILE MAP**

File Address			File Address
00h	INDF <sup>(1)</sup>	INDF <sup>(1)</sup>	80h
01h	TMR0	OPTION_REG	81h
02h	PCL	PCL	82h
03h	STATUS	STATUS	83h
04h	FSR	FSR	84h
05h	PORTA	TRISA	85h
06h	PORTB	TRISB	86h
07h	DATAACP	TRISCCP	87h
08h			88h
09h			89h
0Ah	PCLATH	PCLATH	8Ah
0Bh	INTCON	INTCON	8Bh
0Ch	PIR1	PIE1	8Ch
0Dh			8Dh
0Eh	TMR1L	PCON	8Eh
0Fh	TMR1H		8Fh
10h	T1CON		90h
11h	TMR2		91h
12h	T2CON	PR2	92h
13h			93h
14h			94h
15h	CCPR1L		95h
16h	CCPR1H		96h
17h	CCP1CON		97h
18h			98h
19h			99h
1Ah			9Ah
1Bh			9Bh
1Ch			9Ch
1Dh			9Dh
1Eh	ADRES		9Eh
1Fh	ADCON0	ADCON1	9Fh
20h	General Purpose Registers 96 Bytes	General Purpose Registers 32 Bytes	A0h
			BFh
			C0h
7Fh			FFh
	Bank 0	Bank 1	

Unimplemented data memory locations, read as '0'.

**Note 1:** Not a physical register.



# PIC16C712/716

## 2.2.2 SPECIAL FUNCTION REGISTERS

The Special Function Registers are registers used by the CPU and Peripheral Modules for controlling the desired operation of the device. These registers are implemented as static RAM. A list of these registers is given in Table 2-1.

The Special Function Registers can be classified into two sets; core (CPU) and peripheral. Those registers associated with the core functions are described in detail in this section. Those related to the operation of the peripheral features are described in detail in that peripheral feature section.

**TABLE 2-1: SPECIAL FUNCTION REGISTER SUMMARY**

Addr	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other Resets (4)	
<b>Bank 0</b>												
00h	INDF <sup>(1)</sup>	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	0000 0000	
01h	TMR0	Timer0 Module's Register								xxxx xxxx	uuuu uuuu	
02h	PCL <sup>(1)</sup>	Program Counter's (PC) Least Significant Byte								0000 0000	0000 0000	
03h	STATUS <sup>(1)</sup>	IRP <sup>(4)</sup>	RP1 <sup>(4)</sup>	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	rr01 lxxx	rr0q quuu	
04h	FSR <sup>(1)</sup>	Indirect Data Memory Address Pointer								xxxx xxxx	uuuu uuuu	
05h	PORTA <sup>(5,6)</sup>	—	—	— <sup>(7)</sup>	PORTA Data Latch when written: PORTA pins when read						--xx xxxx	--xu uuuu
06h	PORTB <sup>(5,6)</sup>	PORTB Data Latch when written: PORTB pins when read								xxxx xxxx	uuuu uuuu	
07h	DATAACP	— <sup>(7)</sup>	— <sup>(7)</sup>	— <sup>(7)</sup>	— <sup>(7)</sup>	— <sup>(7)</sup>	DCCP	— <sup>(7)</sup>	DT1CK	xxxx xxxx	xxxx xuxu	
08h-09h	—	Unimplemented								—	—	
0Ah	PCLATH <sup>(1,2)</sup>	—	—	—	Write Buffer for the upper 5 bits of the Program Counter						---0 0000	---0 0000
0Bh	INTCON <sup>(1)</sup>	GIE	PEIE	TOIE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u	
0Ch	PIR1	—	ADIF	—	—	—	CCP1IF	TMR2IF	TMR1IF	-0-- 0000	-0-- 0000	
0Dh	—	Unimplemented								—	—	
0Eh	TMR1L	Holding Register for the Least Significant Byte of the 16-bit TMR1 Register								xxxx xxxx	uuuu uuuu	
0Fh	TMR1H	Holding Register for the Most Significant Byte of the 16-bit TMR1 Register								xxxx xxxx	uuuu uuuu	
10h	T1CON	—	—	T1CKPS1	T1CKPS0	T1OSCEN	$\overline{T1SYNC}$	TMR1CS	TMR1ON	--00 0000	--uu uuuu	
11h	TMR2	Timer2 Module's Register								0000 0000	0000 0000	
12h	T2CON	—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	-000 0000	-000 0000	
13h-14h	—	Unimplemented								—	—	
15h	CCPR1L	Capture/Compare/PWM Register1 (LSB)								xxxx xxxx	uuuu uuuu	
16h	CCPR1H	Capture/Compare/PWM Register1 (MSB)								xxxx xxxx	uuuu uuuu	
17h	CCP1CON	—	—	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0	--00 0000	--00 0000	
18h-1Dh	—	Unimplemented								—	—	
1Eh	ADRES	A/D Result Register								xxxx xxxx	uuuu uuuu	
1Fh	ADCON0	ADCS1	ADCS0	CHS2	CHS1	CHS0	GO/DONE	—	ADON	0000 00-0	0000 00-0	

**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.

**Note 2:** The upper byte of the program counter is not directly accessible. PCLATH is a holding register for PC<12:8> whose contents are transferred to the upper byte of the program counter.

**Note 3:** Other (non Power-up) Resets include: external Reset through  $\overline{MCLR}$  and the Watchdog Timer Reset.

**Note 4:** The IRP and RP1 bits are reserved. Always maintain these bits clear.

**Note 5:** On any device Reset, these pins are configured as inputs.

**Note 6:** This is the value that will be in the port output latch.

**Note 7:** Reserved bits; Do Not Use.

# PIC16C712/716

**TABLE 2-1: SPECIAL FUNCTION REGISTER SUMMARY (CONTINUED)**

Addr	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other Resets (4)
<b>Bank 1</b>											
80h	INDF <sup>(1)</sup>	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	0000 0000
81h	OPTION_ REG	$\overline{\text{RBPU}}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
82h	PCL <sup>(1)</sup>	Program Counter's (PC) Least Significant Byte								0000 0000	0000 0000
83h	STATUS <sup>(1)</sup>	IRP <sup>(4)</sup>	RP1 <sup>(4)</sup>	RP0	$\overline{\text{TO}}$	$\overline{\text{PD}}$	Z	DC	C	rr01 1xxx	rr0q quuu
84h	FSR <sup>(1)</sup>	Indirect Data Memory Address Pointer								xxxx xxxx	uuuu uuuu
85h	TRISA	—	—	— <sup>(7)</sup>	PORTA Data Direction Register					--x1 1111	--x1 1111
86h	TRISB	PORTB Data Direction Register								1111 1111	1111 1111
87h	TRISCCP	— <sup>(7)</sup>	— <sup>(7)</sup>	— <sup>(7)</sup>	— <sup>(7)</sup>	— <sup>(7)</sup>	TCCP	— <sup>(7)</sup>	TT1CK	xxxx x1x1	xxxx x1x1
88h-89h	—	Unimplemented								—	—
8Ah	PCLATH <sup>(1,2)</sup>	—	—	—	Write Buffer for the upper 5 bits of the Program Counter					---0 0000	---0 0000
8Bh	INTCON <sup>(1)</sup>	GIE	PEIE	TOIE	INTE	RBIE	TOIF	INTF	RBIF	0000 000x	0000 000u
8Ch	PIE1	—	ADIE	—	—	—	CCP1IE	TMR2IE	TMR1IE	-0-- -000	-0-- -000
8Dh	—	Unimplemented								—	—
8Eh	PCON	—	—	—	—	—	—	$\overline{\text{POR}}$	$\overline{\text{BOR}}$	---- --pq	---- --uu
8Fh-91h	—	Unimplemented								—	—
92h	PR2	Timer2 Period Register								1111 1111	1111 1111
93h-9Eh	—	Unimplemented								—	—
9Fh	ADCON1	—	—	—	—	—	PCFG2	PCFG1	PCFG0	---- -000	---- -000

**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 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  $\overline{\text{MCLR}}$  and the Watchdog Timer Reset.

**4:** The IRP and RP1 bits are reserved. Always maintain these bits clear.

**5:** On any device Reset, these pins are configured as inputs.

**6:** This is the value that will be in the port output latch.

**7:** Reserved bits; Do Not Use.

## 2.2.2.1 Status Register

The STATUS register, shown in Figure 2-4, 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.

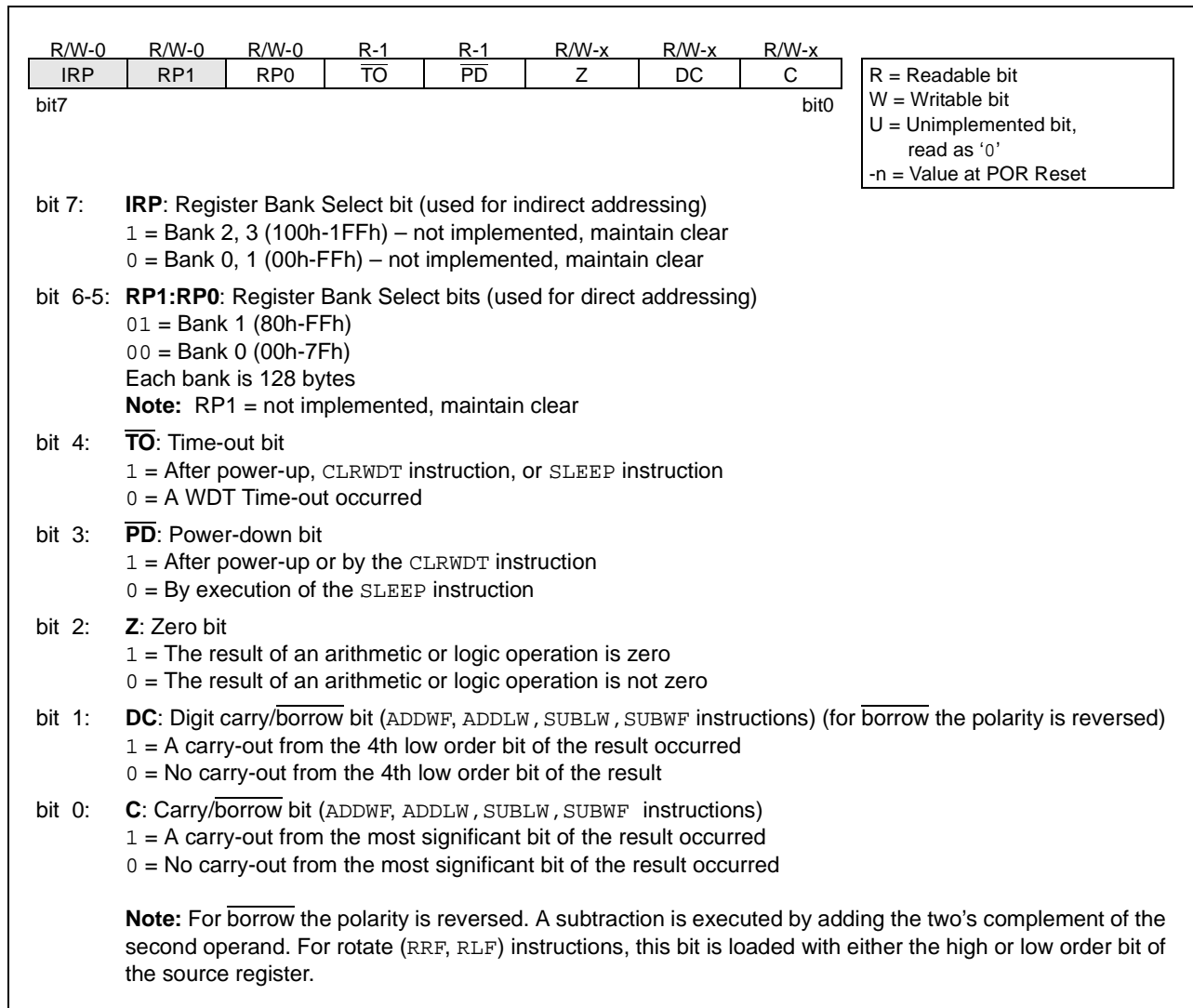
For example, `CLRF STATUS` will clear the upper-three bits and set the Z bit. This leaves the STATUS register as `000u u1uu` (where u = unchanged).

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:** These devices do not use bits IRP and RP1 (STATUS<7:6>). Maintain these bits clear to ensure upward compatibility with future products.

**2:** The C and DC bits operate as a borrow and digit borrow bit, respectively, in subtraction. See the `SUBLW` and `SUBWF` instructions for examples.

**FIGURE 2-4: STATUS REGISTER (ADDRESS 03h, 83h)**



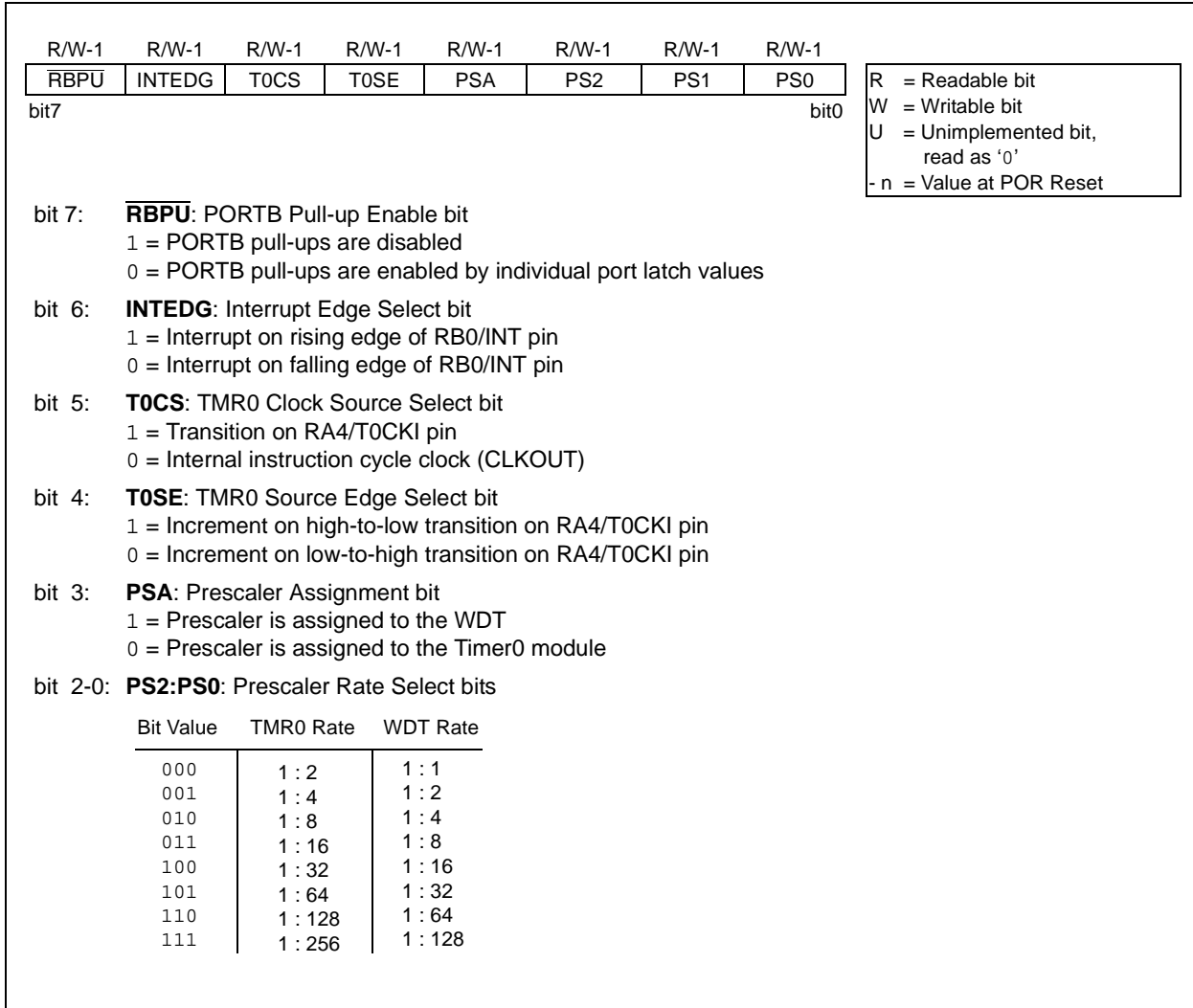
# PIC16C712/716

## 2.2.2.2 OPTION\_REG Register

The OPTION\_REG register is a readable and writable register, which contains various control bits to configure the TMR0 prescaler/WDT postscaler (single assignable register known also as the prescaler), the External INT Interrupt, TMR0 and the weak pull-ups on PORTB.

**Note:** To achieve a 1:1 prescaler assignment for the TMR0 register, assign the prescaler to the Watchdog Timer.

**FIGURE 2-5: OPTION\_REG REGISTER (ADDRESS 81h)**



## 2.2.2.3 INTCON Register

The INTCON Register is a readable and writable register which contains various enable and flag bits for the TMR0 register overflow, RB Port change and External RB0/INT pin interrupts.

**Note:** Interrupt flag bits get set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the global enable bit, GIE (INTCON<7>). User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.

**FIGURE 2-6: INTCON REGISTER (ADDRESS 0Bh, 8Bh)**

	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-x
bit7	GIE	PEIE	TOIE	INTE	RBIE	TOIF	INTF	RBIF
bit0								bit0

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

bit 7: **GIE:** Global Interrupt Enable bit  
 1 = Enables all unmasked interrupts  
 0 = Disables all interrupts

bit 6: **PEIE:** Peripheral Interrupt Enable bit  
 1 = Enables all unmasked peripheral interrupts  
 0 = Disables all peripheral interrupts

bit 5: **TOIE:** TMR0 Overflow Interrupt Enable bit  
 1 = Enables the TMR0 interrupt  
 0 = Disables the TMR0 interrupt

bit 4: **IINTE:** RB0/INT External Interrupt Enable bit  
 1 = Enables the RB0/INT external interrupt  
 0 = Disables the RB0/INT external interrupt

bit 3: **RBIE:** RB Port Change Interrupt Enable bit  
 1 = Enables the RB port change interrupt  
 0 = Disables the RB port change interrupt

bit 2: **TOIF:** TMR0 Overflow Interrupt Flag bit  
 1 = TMR0 register has overflowed (must be cleared in software)  
 0 = TMR0 register did not overflow

bit 1: **INTF:** RB0/INT External Interrupt Flag bit  
 1 = The RB0/INT external interrupt occurred (must be cleared in software)  
 0 = The RB0/INT external interrupt did not occur

bit 0: **RBIF:** RB Port Change Interrupt Flag bit  
 1 = At least one of the RB7:RB4 pins changed state (must be cleared in software)  
 0 = None of the RB7:RB4 pins have changed state

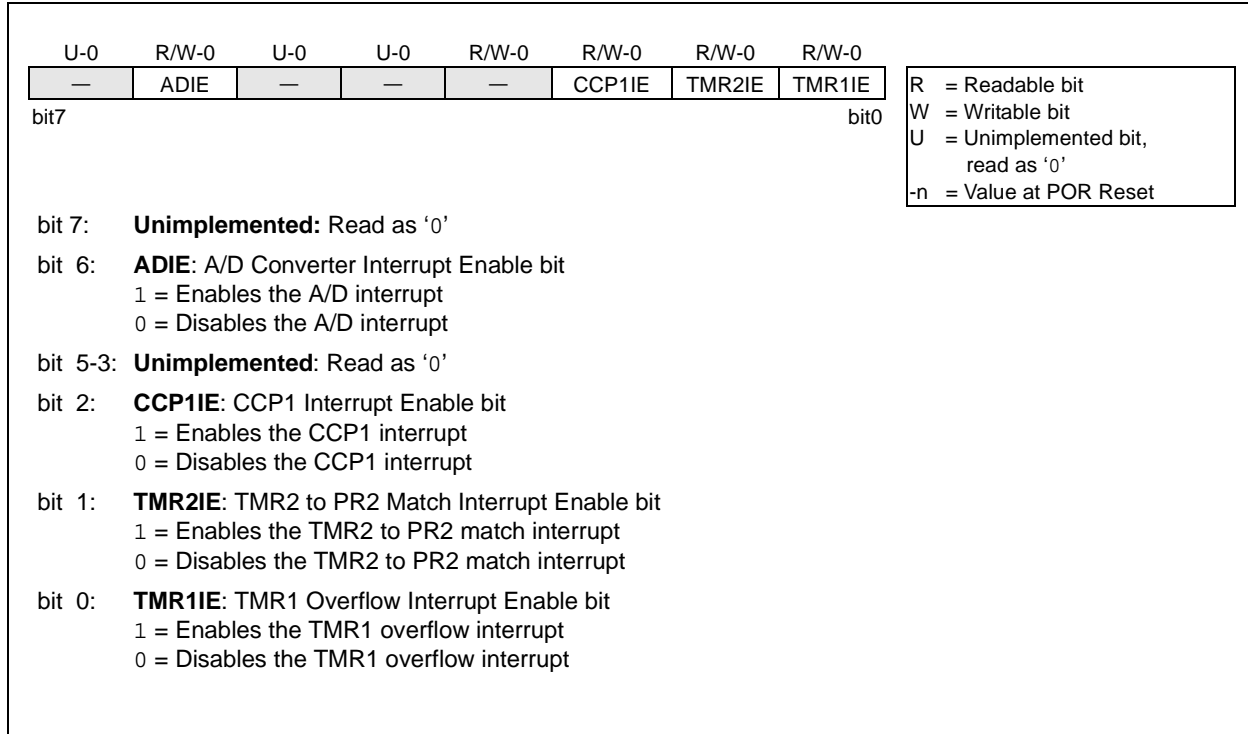
# PIC16C712/716

## 2.2.2.4 PIE1 Register

This register contains the individual enable bits for the peripheral interrupts.

**Note:** Bit PEIE (INTCON<6>) must be set to enable any peripheral interrupt.

**FIGURE 2-7: PIE1 REGISTER (ADDRESS 8Ch)**

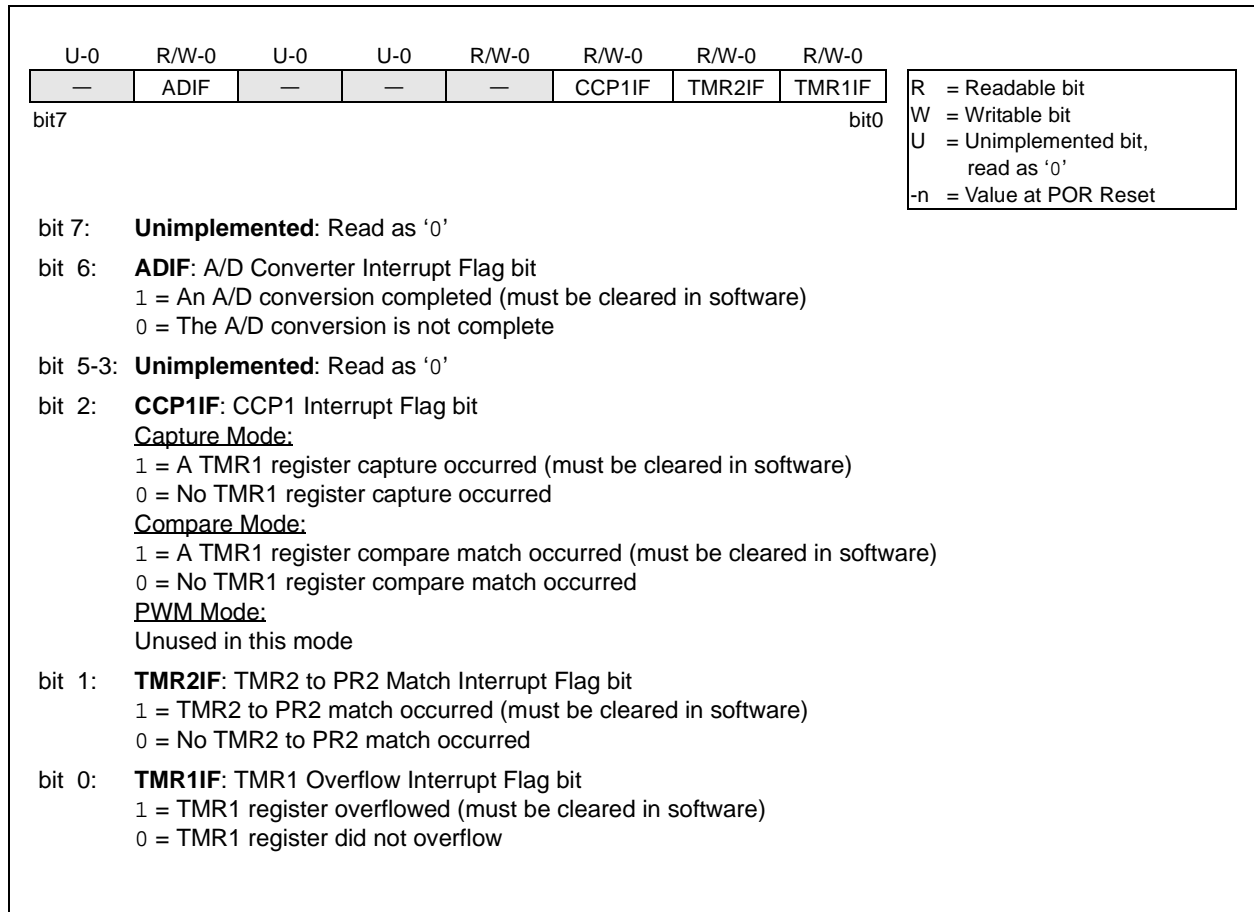


## 2.2.2.5 PIR1 Register

This register contains the individual flag bits for the peripheral interrupts.

**Note:** Interrupt flag bits get set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the global enable bit, GIE (INTCON<7>). User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.

**FIGURE 2-8: PIR1 REGISTER (ADDRESS 0Ch)**



# PIC16C712/716

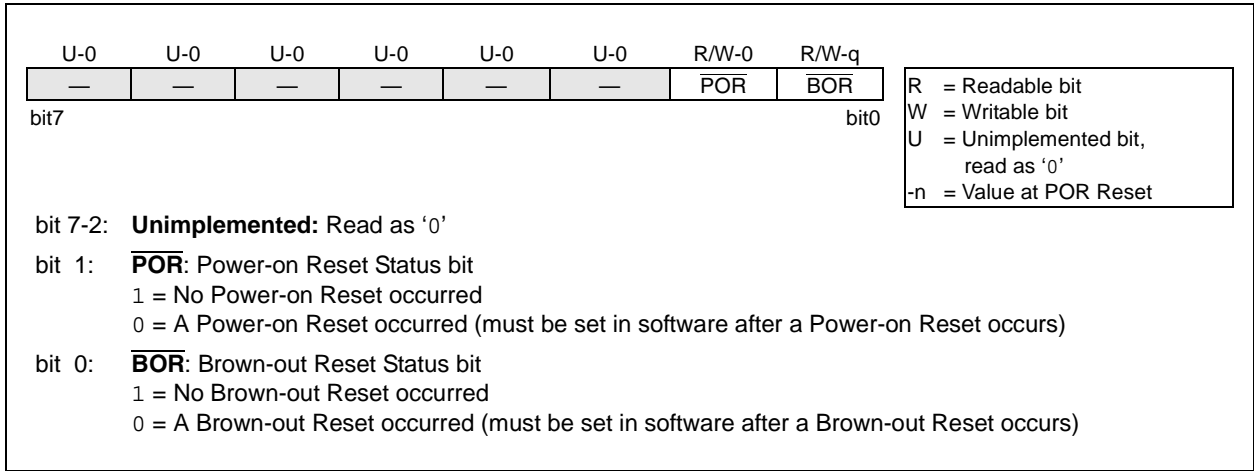
## 2.2.2.6 PCON Register

The Power Control (PCON) register contains a flag bit to allow differentiation between a Power-on Reset (POR) to an external MCLR Reset or WDT Reset. These devices contain an additional bit to differentiate a Brown-out Reset condition from a Power-on Reset condition.

**Note:** If the BODEN Configuration bit is set, BOR is '1' on Power-on Reset. If the BODEN Configuration bit is clear, BOR is unknown on Power-on Reset.

The  $\overline{\text{BOR}}$  Status bit is a "don't care" and is not necessarily predictable if the brown-out circuit is disabled (the BODEN Configuration bit is clear).  $\overline{\text{BOR}}$  must then be set by the user and checked on subsequent resets to see if it is clear, indicating a brown-out has occurred.

**FIGURE 2-9: PCON REGISTER (ADDRESS 8Eh)**





## 2.3 PCL and PCLATH

The Program Counter (PC) specifies the address of the instruction to fetch for execution. The PC is 13 bits wide. The low byte is called the PCL register. This register is readable and writable. The high byte is called the PCH register. This register contains the PC<12:8> bits and is not directly readable or writable. All updates to the PCH register go through the PCLATH register.

### 2.3.1 STACK

The stack allows a combination of up to 8 program calls and interrupts to occur. The stack contains the return address from this branch in program execution.

Mid-range devices have an 8-level deep x 13-bit wide hardware stack. The stack space is not part of either program or data space and the Stack Pointer is not readable or writable. The PC is PUSHed onto the stack when a CALL instruction is executed or an interrupt causes a branch. The stack is POPed in the event of a RETURN, RETLW or a RETFIE instruction execution. PCLATH is not modified when the stack is PUSHed or POPed.

After the stack has been PUSHed eight times, the ninth push overwrites the value that was stored from the first push. The tenth push overwrites the second push (and so on).

## 2.4 Program Memory Paging

The CALL and GOTO instructions provide 11 bits of address to allow branching within any 2K program memory page. When doing a CALL or GOTO instruction, the upper bit of the address is provided by PCLATH<3>. When doing a CALL or GOTO instruction, the user must ensure that the page select bit is programmed so that the desired program memory page is addressed. If a return from a CALL instruction (or interrupt) is executed, the entire 13-bit PC is pushed onto the stack. Therefore, manipulation of the PCLATH<3> bit is not required for the return instructions (which POPs the address from the stack).

# PIC16C712/716

## 2.5 Indirect Addressing, INDF and FSR Registers

The INDF register is not a physical register. Addressing INDF actually addresses the register whose address is contained in the FSR register (FSR is a *pointer*). This is indirect addressing.

### EXAMPLE 2-1: INDIRECT ADDRESSING

- Register file 05 contains the value 10h
- Register file 06 contains the value 0Ah
- Load the value 05 into the FSR register
- A read of the INDF register will return the value of 10h
- Increment the value of the FSR register by one (FSR = 06)
- A read of the INDF register now will return the value of 0Ah.

Reading INDF itself indirectly (FSR = 0) will produce 00h. Writing to the INDF register indirectly results in a no-operation (although Status bits may be affected).

A simple program to clear RAM locations 20h-2Fh using indirect addressing is shown in Example 2-2.

### EXAMPLE 2-2: HOW TO CLEAR RAM USING INDIRECT ADDRESSING

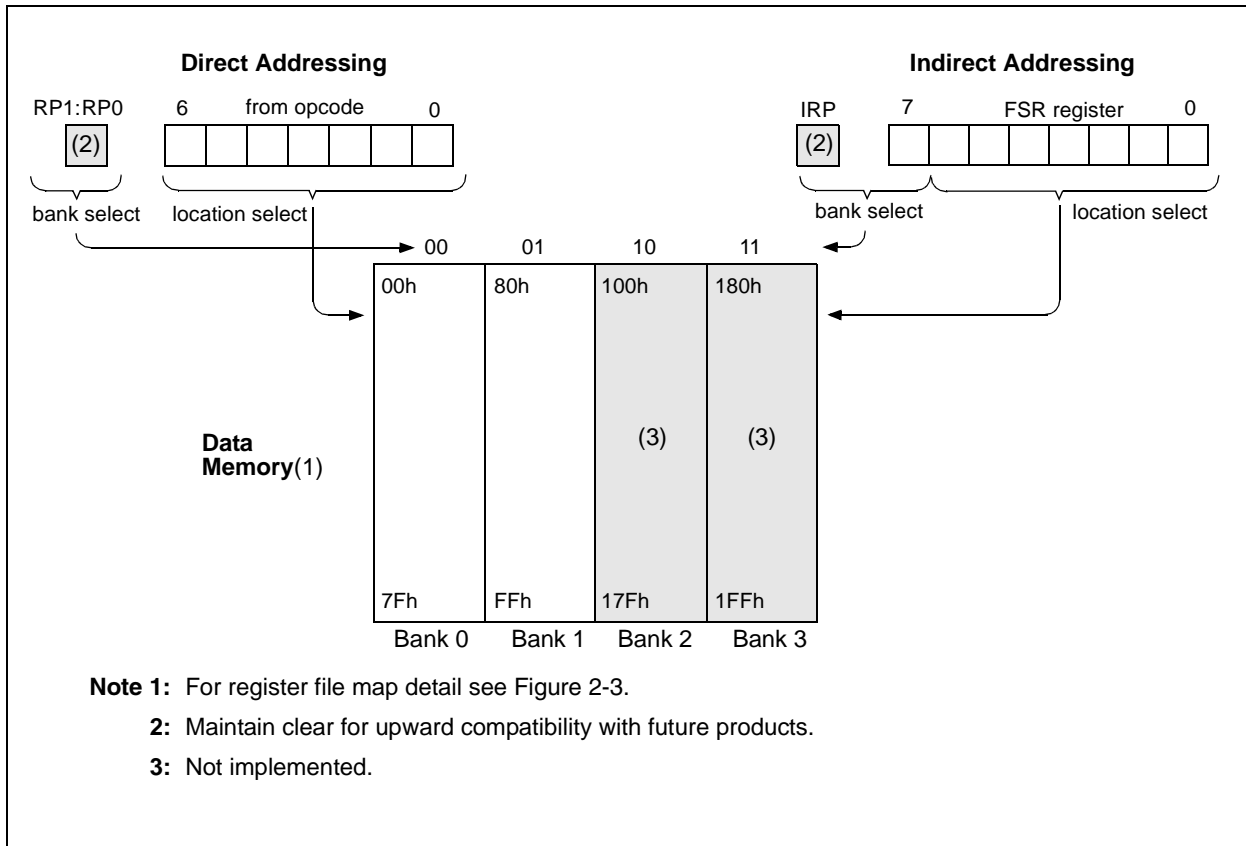
```

MOV LW 0x20 ;initialize pointer
MOV WF FSR ; to RAM
NEXT   CLR F INDF ;clear INDF register
      INC F FSR ;inc pointer
      BT FSS FSR,4 ;all done?
      GOTO NEXT ;NO, clear next

CONTINUE
      : ;YES, continue
    
```

An effective 9-bit address is obtained by concatenating the 8-bit FSR register and the IRP bit (STATUS<7>), as shown in Figure 2-10. However, IRP is not used in the PIC16C712/716.

FIGURE 2-10: DIRECT/INDIRECT ADDRESSING



## 3.0 I/O PORTS

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.

Additional information on I/O ports may be found in the PIC<sup>®</sup> Mid-Range Reference Manual, (DS33023).

### 3.1 PORTA and the TRISA Register

PORTA is a 5-bit wide bidirectional port. The corresponding data direction register is TRISA. Setting a TRISA bit (= 1) will make the corresponding PORTA pin an input, (i.e., put the corresponding output driver in a High-Impedance mode). Clearing a TRISA bit (= 0) will make the corresponding PORTA pin an output, (i.e., put the contents of the output latch on the selected pin).

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, the 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. 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.

PORTA pins, RA3:0, 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'.

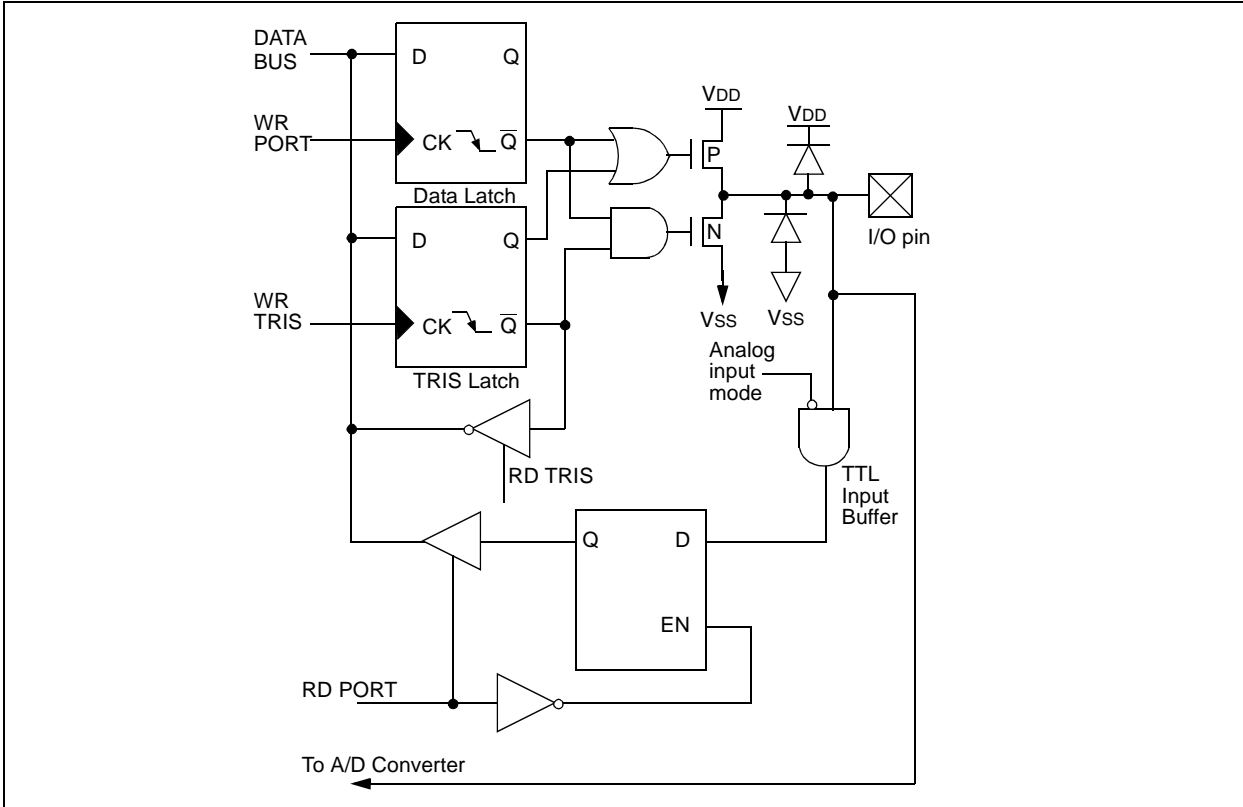
The TRISA register controls the direction of the RA pins, even when they are being used as analog inputs. The user must ensure the bits in the TRISA register are maintained set when using them as analog inputs.

#### EXAMPLE 3-1: INITIALIZING PORTA

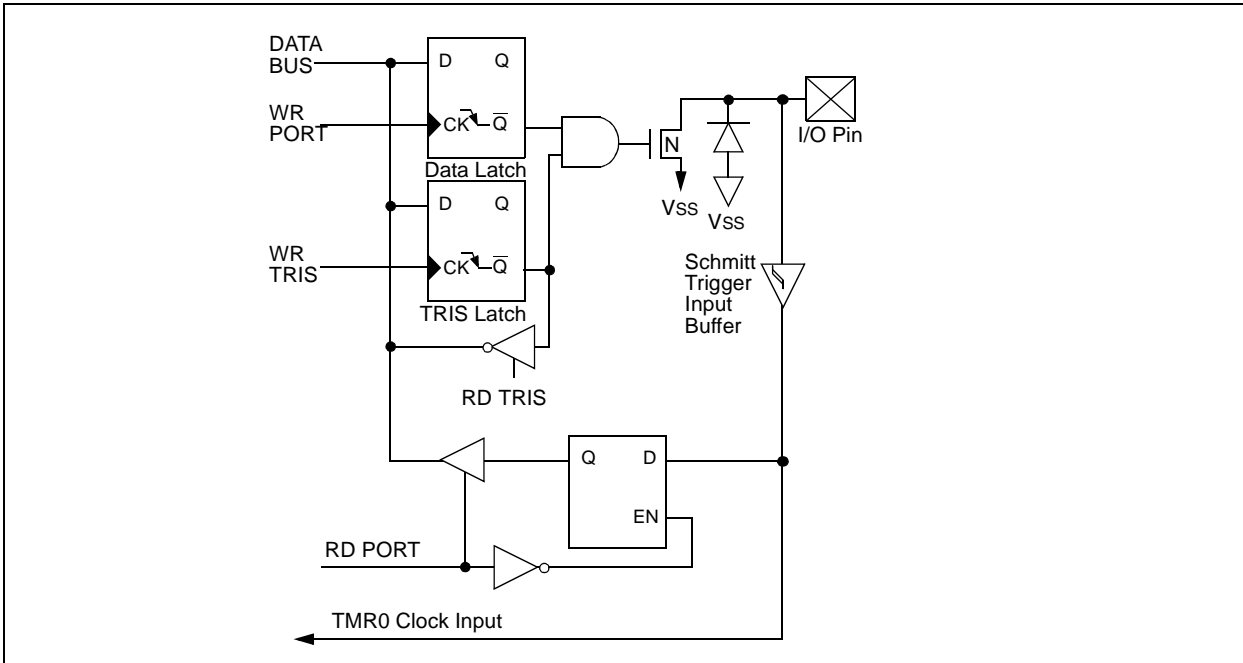
```
BCF    STATUS, RP0 ;
CLRF   PORTA       ; Initialize PORTA by
                ; clearing output
                ; data latches
BSF    STATUS, RP0 ; Select Bank 1
MOVLW  0xEF        ; Value used to
                ; initialize data
                ; direction
MOVWF  TRISA       ; Set RA<3:0> as inputs
                ; RA<4> as outputs
BCF    STATUS, RP0 ; Return to Bank 0
```

# PIC16C712/716

**FIGURE 3-1: BLOCK DIAGRAM OF RA3:RA0**



**FIGURE 3-2: BLOCK DIAGRAM OF RA4/T0CKI PIN**



**TABLE 3-1: PORTA FUNCTIONS**

Name	Bit#	Buffer	Function
RA0/AN0	bit 0	TTL	Input/output or analog input
RA1/AN1	bit 1	TTL	Input/output or analog input
RA2/AN2	bit 2	TTL	Input/output or analog input
RA3/AN3/VREF	bit 3	TTL	Input/output or analog input or VREF
RA4/T0CKI	bit 4	ST	Input/output or external clock input for Timer0 Output is open drain type

**Legend:** TTL = TTL input, ST = Schmitt Trigger input

**TABLE 3-2: SUMMARY OF REGISTERS ASSOCIATED WITH PORTA**

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
05h	PORTA	—	—	— <sup>(1)</sup>	RA4	RA3	RA2	RA1	RA0	--xx xxxx	--xu uuuu
85h	TRISA	—	—	— <sup>(1)</sup>	PORTA Data Direction Register					--11 1111	--11 1111
9Fh	ADCON1	—	—	—	—	—	PCFG2	PCFG1	PCFG0	---- -000	---- -000

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

**Note 1:** Reserved bits; Do Not Use.

# PIC16C712/716

## 3.2 PORTB and the TRISB Register

PORTB is an 8-bit wide bidirectional port. The corresponding data direction register is TRISB. Setting a TRISB bit (= 1) will make the corresponding PORTB pin an input, (i.e., put the corresponding output driver in a High-Impedance mode). Clearing a TRISB bit (= 0) will make the corresponding PORTB pin an output, (i.e., put the contents of the output latch on the selected pin).

Each of the PORTB pins has a weak internal pull-up. A single control bit can turn on all the pull-ups. This is performed by clearing bit  $\overline{\text{RBPU}}$  (OPTION\_REG<7>). The weak pull-up is automatically turned off when the port pin is configured as an output. The pull-ups are disabled on a Power-on Reset.

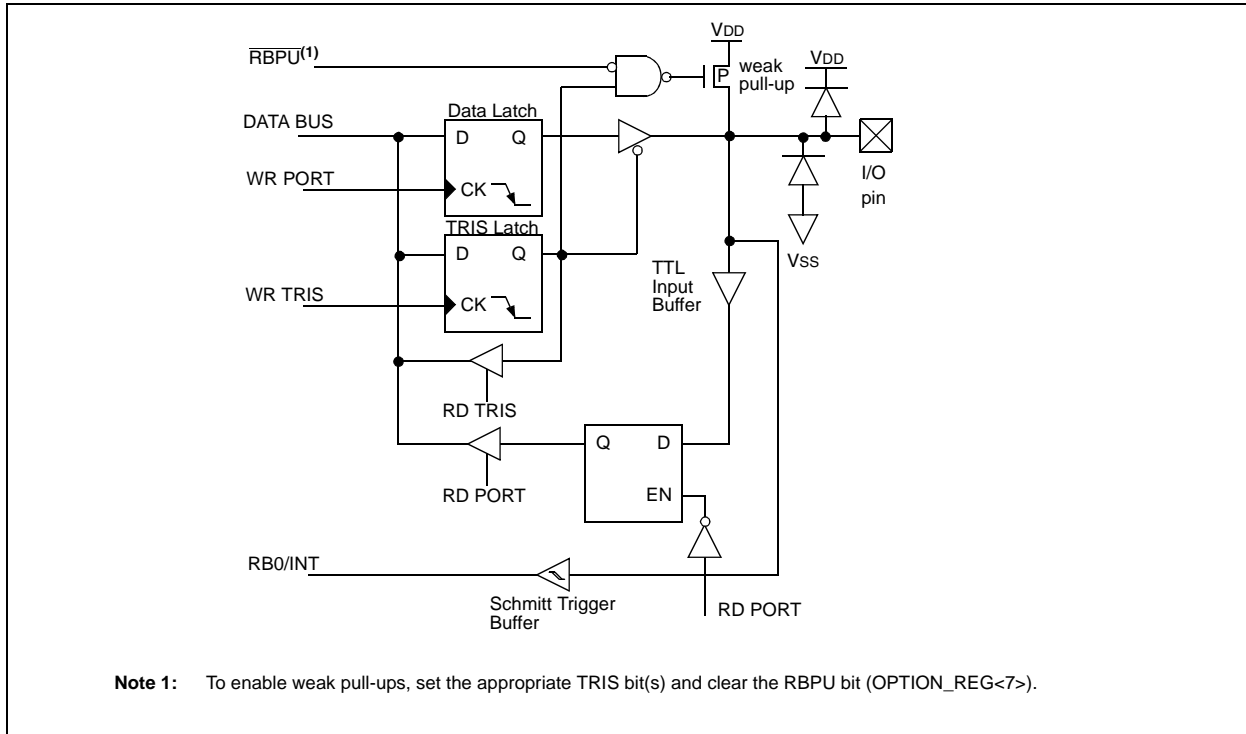
### EXAMPLE 3-2: INITIALIZING PORTB

```
BCF STATUS, RP0 ;
CLRF PORTB      ; Initialize PORTB by
                ; clearing output
                ; data latches

BSF STATUS, RP0 ; Select Bank 1
MOVLW 0xCF      ; Value used to
                ; initialize data
                ; direction

MOVWF TRISB     ; Set RB<3:0> as inputs
                ; RB<5:4> as outputs
                ; RB<7:6> as inputs
```

FIGURE 3-3: BLOCK DIAGRAM OF RB0 PIN



PORTB pins RB3:RB1 are multiplexed with several peripheral functions (Table 3-3). PORTB pins RB3:RB0 have Schmitt Trigger input buffers.

When enabling peripheral functions, care should be taken in defining TRIS bits for each PORTB pin. Some peripherals override the TRIS bit to make a pin an output, while other peripherals override the TRIS bit to make a pin an input. Since the TRIS bit override is in effect while the peripheral is enabled, read-modify-write instructions (*BSF*, *BCF*, *XORWF*) with TRISB as destination should be avoided. The user should refer to the corresponding peripheral section for the correct TRIS bit settings.

Four of PORTB's pins, RB7:RB4, have an interrupt-on-change feature. Only pins configured as inputs can cause this interrupt to occur (i.e., any RB7:RB4 pin configured as an output is excluded from the interrupt-on-change comparison). The input pins, RB7:RB4, are compared with the old value latched on the last read of

PORTB. The "mismatch" outputs of RB7:RB4 are OR'ed together to generate the RB Port Change Interrupt with flag bit RBIF (INTCON<0>).

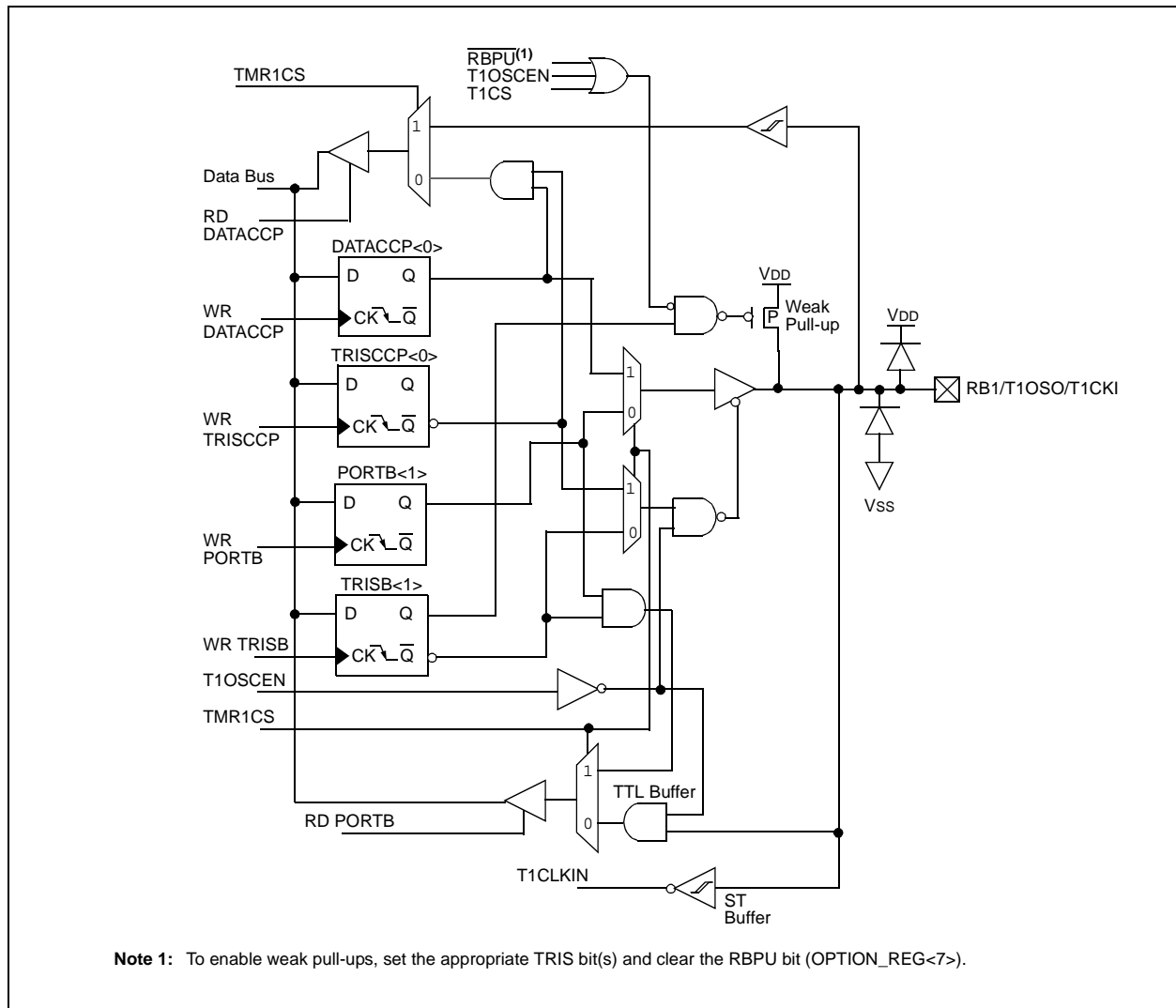
This interrupt can wake the device from Sleep. The user, in the Interrupt Service Routine, can clear the interrupt in the following manner:

- a) Any read or write of PORTB will end the mismatch condition.
- b) Clear flag bit RBIF.

A mismatch condition will continue to set flag bit RBIF. Reading PORTB will end the mismatch condition and allow flag bit RBIF to be cleared.

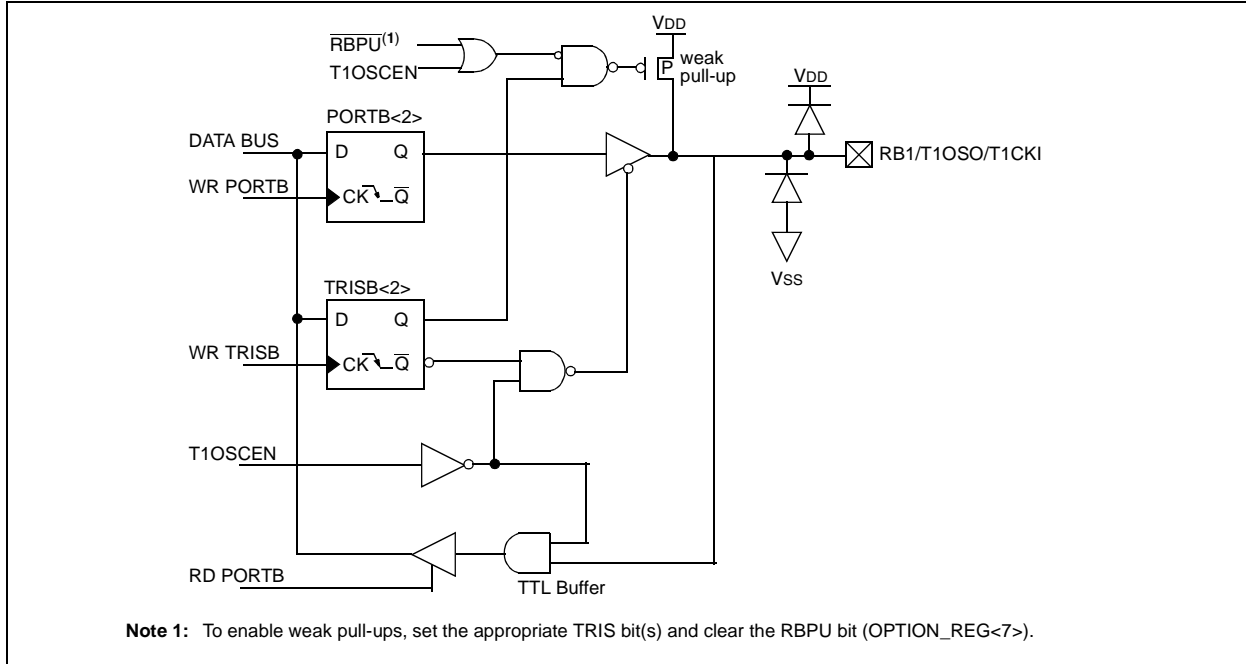
The interrupt-on-change feature is recommended for wake-up on key depression operation and operations where PORTB is only used for the interrupt-on-change feature. Polling of PORTB is not recommended while using the interrupt-on-change feature.

**FIGURE 3-4: BLOCK DIAGRAM OF RB1/T1OSO/T1CKI PIN**

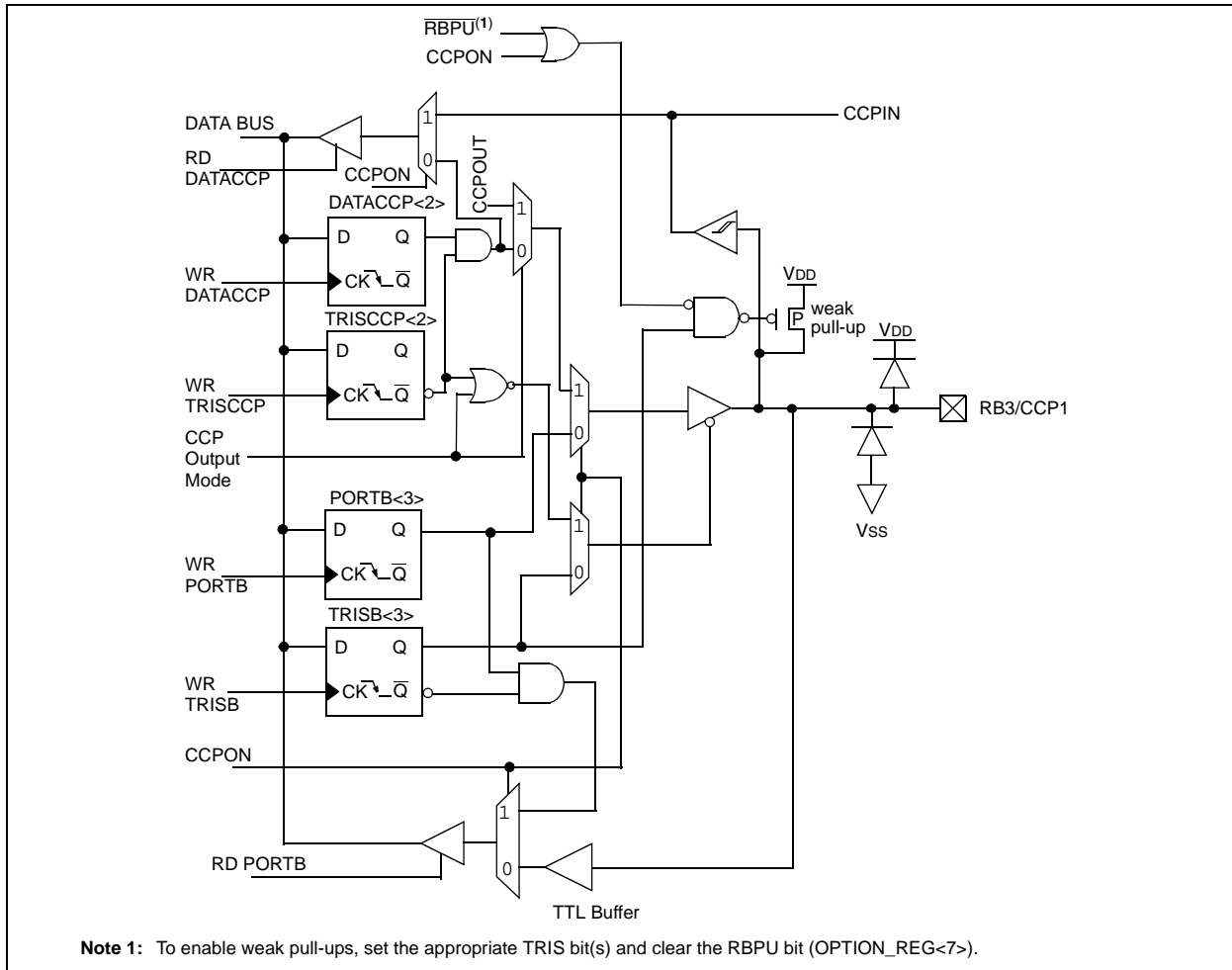


# PIC16C712/716

**FIGURE 3-5: BLOCK DIAGRAM OF RB2/T1OSI PIN**

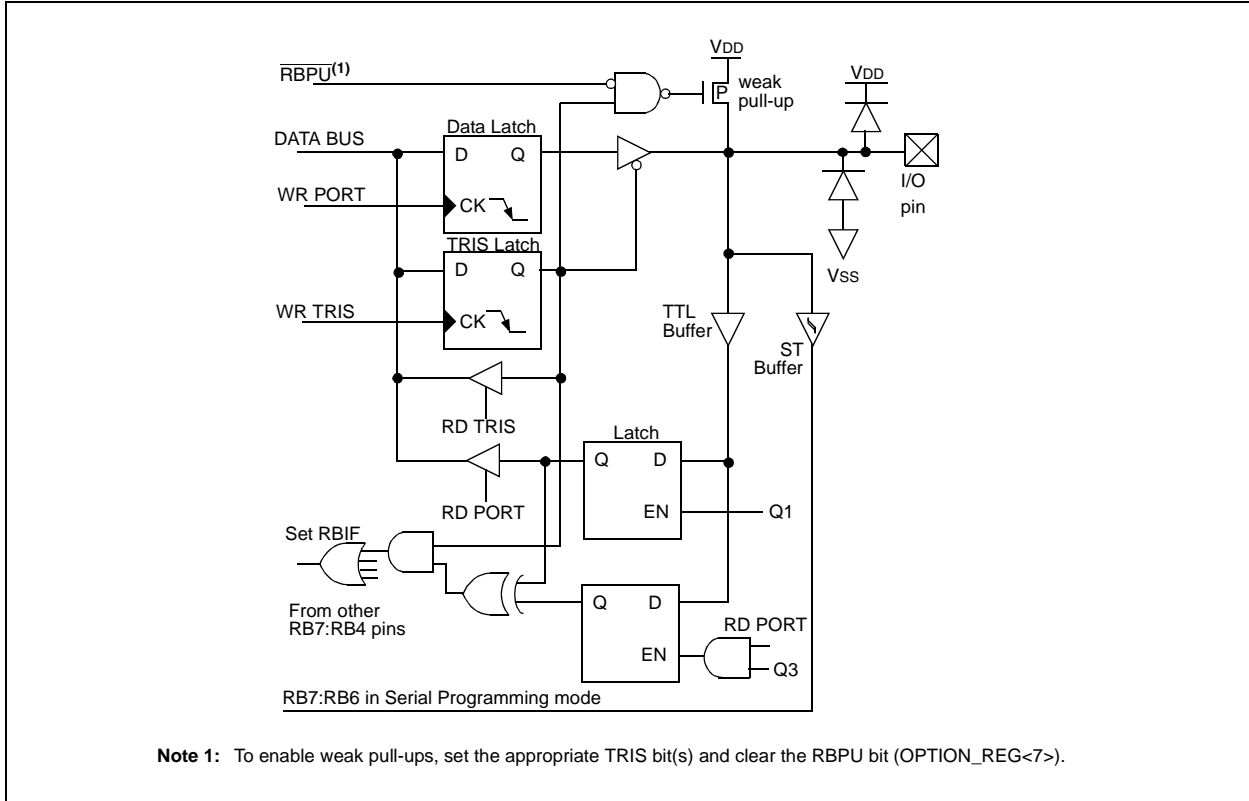


**FIGURE 3-6: BLOCK DIAGRAM OF RB3/CCP1 PIN**





**FIGURE 3-7: BLOCK DIAGRAM OF RB7:RB4 PINS**



**TABLE 3-3: PORTB FUNCTIONS**

Name	Bit#	Buffer	Function
RB0/INT	bit 0	TTL/ST <sup>(1)</sup>	Input/output pin or external interrupt input. Internal software programmable weak pull-up.
RB1/T1OS0/T1CKI	bit 1	TTL/ST <sup>(1)</sup>	Input/output pin or Timer1 oscillator output, or Timer1 clock input. Internal software programmable weak pull-up. See Timer1 section for detailed operation.
RB2/T1OSI	bit 2	TTL/ST <sup>(1)</sup>	Input/output pin or Timer1 oscillator input. Internal software programmable weak pull-up. See Timer1 section for detailed operation.
RB3/CCP1	bit 3	TTL/ST <sup>(1)</sup>	Input/output pin or Capture 1 input, or Compare 1 output, or PWM1 output. Internal software programmable weak pull-up. See CCP1 section for detailed operation.
RB4	bit 4	TTL	Input/output pin (with interrupt-on-change). Internal software programmable weak pull-up.
RB5	bit 5	TTL	Input/output pin (with interrupt-on-change). Internal software programmable weak pull-up.
RB6	bit 6	TTL/ST <sup>(2)</sup>	Input/output pin (with interrupt-on-change). Internal software programmable weak pull-up. Serial programming clock.
RB7	bit 7	TTL/ST <sup>(2)</sup>	Input/output pin (with interrupt-on-change). Internal software programmable weak pull-up. Serial programming data.

**Legend:** TTL = TTL input, ST = Schmitt Trigger input

**Note 1:** This buffer is a Schmitt Trigger input when configured as the external interrupt or peripheral input.

**2:** This buffer is a Schmitt Trigger input when used in Serial Programming mode.

# PIC16C712/716

**TABLE 3-4: SUMMARY OF REGISTERS ASSOCIATED WITH PORTB**

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other Resets
06h	PORTB	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	xxxx xxxx	uuuu uuuu
86h	TRISB	PORTB Data Direction Register								1111 1111	1111 1111
81h	OPTION_REG	$\overline{\text{RBPU}}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111

**Legend:** x = unknown, u = unchanged. Shaded cells are not used by PORTB.

## 4.0 TIMER0 MODULE

The Timer0 module timer/counter has the following features:

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

Figure 4-1 is a simplified block diagram of the Timer0 module.

Additional information on timer modules is available in the PIC<sup>®</sup> Mid-Range Reference Manual, (DS33023).

### 4.1 Timer0 Operation

Timer0 can operate as a timer or as a counter.

Timer mode is selected by clearing bit T0CS (OPTION\_REG<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. The user can work around this by writing an adjusted value to the TMR0 register.

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

When an external clock input is used for Timer0, it must meet certain requirements. The requirements ensure the external clock can be synchronized with the internal phase clock (Tosc). Also, there is a delay in the actual incrementing of Timer0 after synchronization.

Additional information on external clock requirements is available in the PIC<sup>®</sup> Mid-Range Reference Manual, (DS33023).

### 4.2 Prescaler

An 8-bit counter is available as a prescaler for the Timer0 module or as a postscaler for the Watchdog Timer, respectively (Figure 4-2). For simplicity, this counter is being referred to as “prescaler” throughout this data sheet. Note that there is only one prescaler available, which is mutually exclusively shared between the Timer0 module and the Watchdog Timer. Thus, a prescaler assignment for the Timer0 module means that there is no prescaler for the Watchdog Timer and vice-versa.

The prescaler is not readable or writable.

The PSA and PS2:PS0 bits (OPTION\_REG<3:0>) determine the prescaler assignment and prescale ratio.

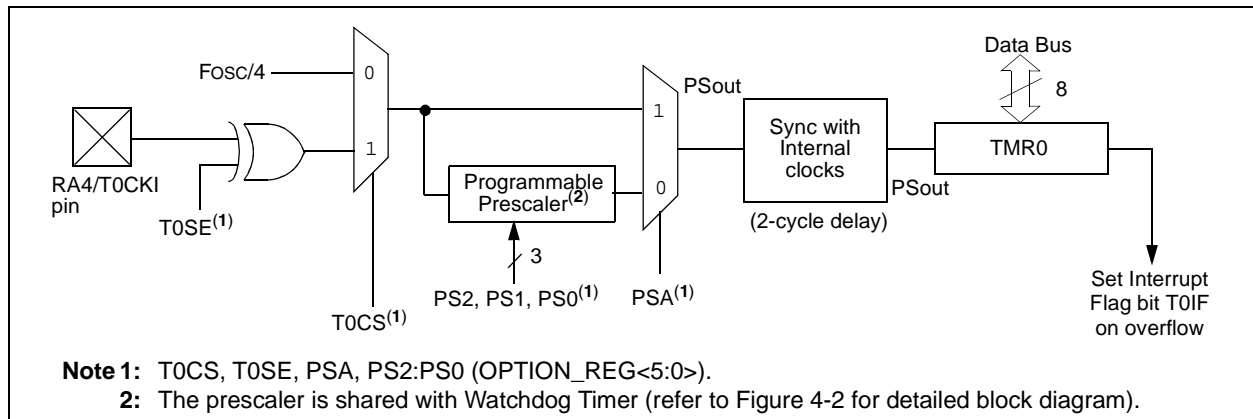
Clearing bit PSA will assign the prescaler to the Timer0 module. When the prescaler is assigned to the Timer0 module, prescale values of 1:2, 1:4, ..., 1:256 are selectable.

Setting bit PSA will assign the prescaler to the Watchdog Timer (WDT). When the prescaler is assigned to the WDT, prescale values of 1:1, 1:2, ..., 1:128 are selectable.

When assigned to the Timer0 module, all instructions writing to the TMR0 register (e.g., CLRWF 1, MOVWF 1, BSF 1,x,...etc.) will clear the prescaler. When assigned to WDT, a CLRWDT instruction will clear the prescaler along with the WDT.

**Note:** Writing to TMR0 when the prescaler is assigned to Timer0 will clear the prescaler count, but will not change the prescaler assignment.

**FIGURE 4-1: TIMER0 BLOCK DIAGRAM**



# PIC16C712/716

## 4.2.1 SWITCHING PRESCALER ASSIGNMENT

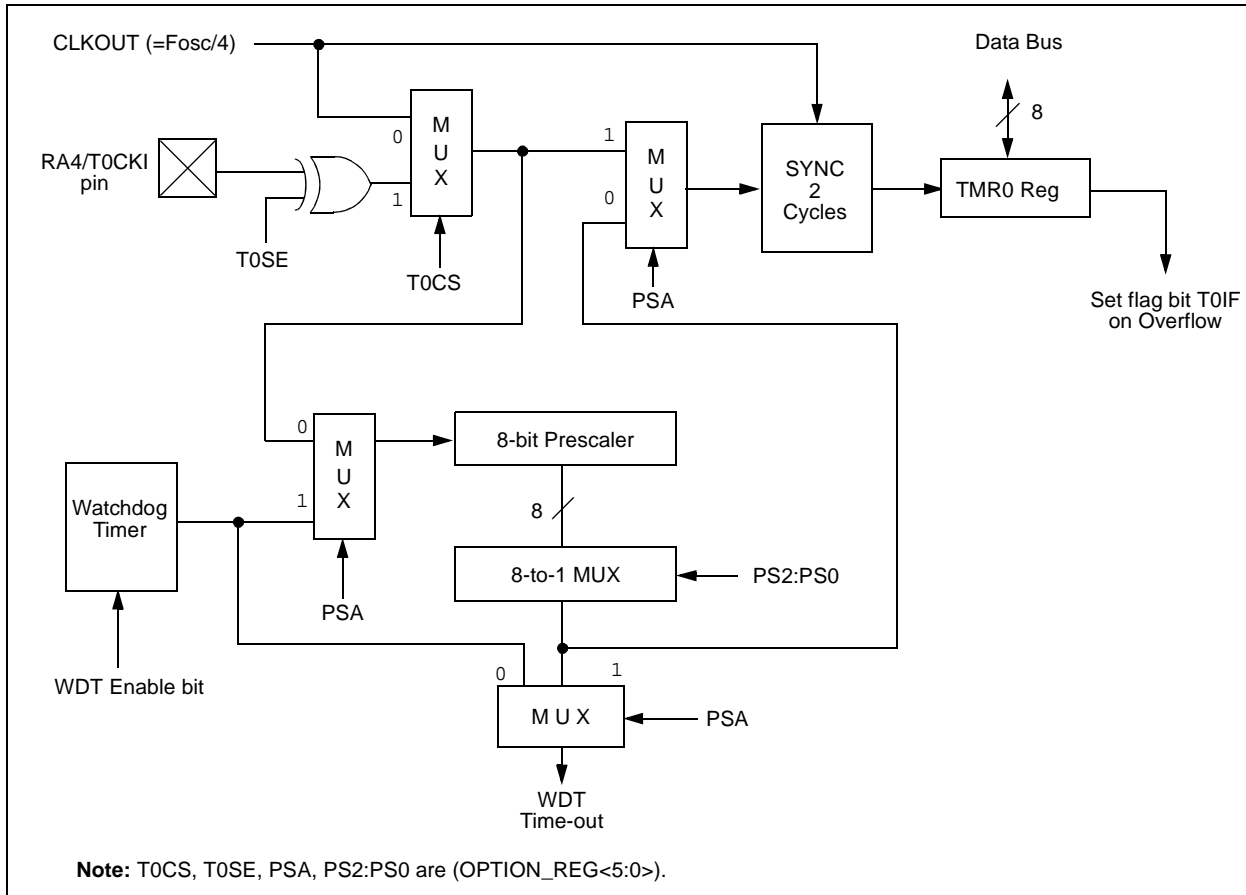
The prescaler assignment is fully under software control (i.e., it can be changed “on the fly” during program execution).

**Note:** To avoid an unintended device Reset, a specific instruction sequence (shown in the PIC® Mid-Range Reference Manual, DS33023) must be executed when changing the prescaler assignment from Timer0 to the WDT. This sequence must be followed even if the WDT is disabled.

## 4.3 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.

**FIGURE 4-2: BLOCK DIAGRAM OF THE TIMER0/WDT PRESCALER**



**TABLE 4-1: 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, BOR	Value on all other Resets
01h	TMR0	Timer0 Module's Register								xxxx xxxx	uuuu uuuu
0Bh,8Bh	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
81h	OPTION_REG	RBPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
85h	TRISA	—	—	— <sup>(1)</sup>	Bit 4	PORTA Data Direction Register			--11 1111	--11 1111	

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

**Note 1:** Reserved bit; Do Not Use.

## 5.0 TIMER1 MODULE

The Timer1 module timer/counter has the following features:

- 16-bit timer/counter  
(Two 8-bit registers; TMR1H and TMR1L)
- Readable and writable (Both registers)
- Internal or external clock select
- Interrupt on overflow from FFFFh to 0000h
- Reset from CCP module trigger

Timer1 has a control register, shown in Figure 5-1. Timer1 can be enabled/disabled by setting/clearing control bit TMR1ON (T1CON<0>).

Figure 5-2 is a simplified block diagram of the Timer1 module.

Additional information on timer modules is available in the PIC® Mid-Range Reference Manual, (DS33023).

## 5.1 Timer1 Operation

Timer1 can operate in one of these modes:

- As a timer
- As a synchronous counter
- As an asynchronous counter

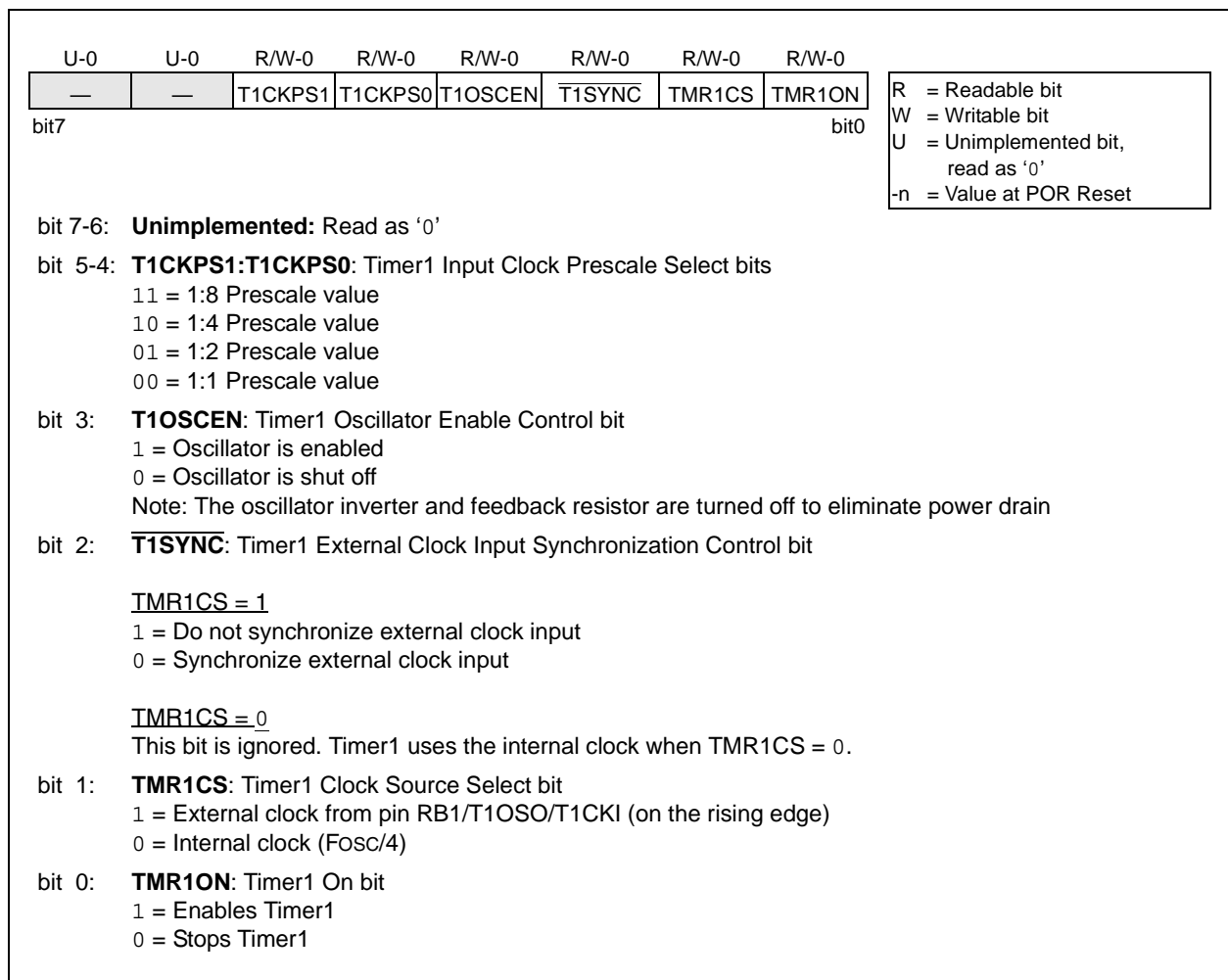
The operating mode is determined by the clock select bit, TMR1CS (T1CON<1>).

In timer mode, Timer1 increments every instruction cycle. In counter mode, it increments on every rising edge of the external clock input.

When the Timer1 oscillator is enabled (T1OSCEN is set), the RB2/T1OSI and RB1/T1OSO/T1CKI pins become inputs. That is, the TRISB<2:1> value is ignored.

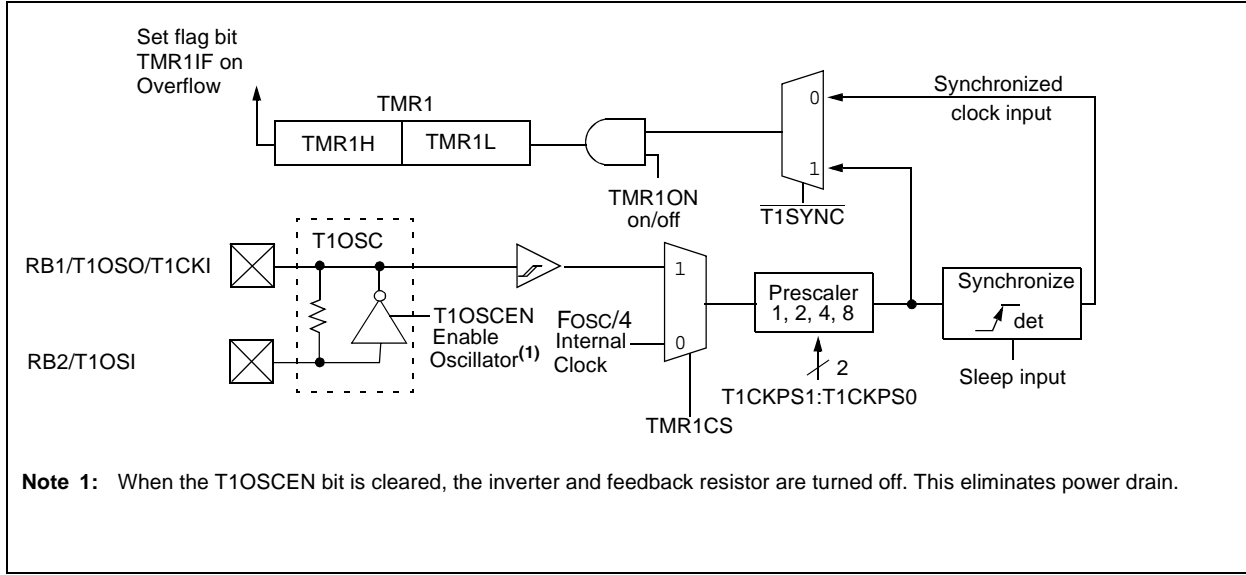
Timer1 also has an internal "Reset input". This Reset can be generated by the CCP module (see **Section 7.0 "Capture/Compare/PWM (CCP) Module(s)"**).

**FIGURE 5-1: T1CON: TIMER1 CONTROL REGISTER (ADDRESS 10h)**



# PIC16C712/716

**FIGURE 5-2: TIMER1 BLOCK DIAGRAM**



## 5.2 Timer1 Module and PORTB Operation

When Timer1 is configured as timer running from the main oscillator, PORTB<2:1> operate as normal I/O lines. When Timer1 is configured to function as a counter however, the clock source selection may affect the operation of PORTB<2:1>. Multiplexing details of the Timer1 clock selection on PORTB are shown in Figure 3-4 and Figure 3-5.

The clock source for Timer1 in the Counter mode can be from one of the following:

1. External circuit connected to the RB1/T1OSO/T1CKI pin
2. Firmware controlled DATA CCP<0> bit, DT1CKI
3. Timer1 oscillator

Table 5-1 shows the details of Timer1 mode selections, control bit settings, TMR1 and PORTB operations.

**TABLE 5-1: TMR1 MODULE AND PORTB OPERATION**

TMR1 Module Mode	Clock Source	Control Bits	TMR1 Module Operation	PORTB<2:1> Operation
Off	N/A	T1CON = --xx 0x00	Off	PORTB<2:1> function as normal I/O
Timer	Fosc/4	T1CON = --xx 0x01	TMR1 module uses the main oscillator as clock source. TMR1ON can turn on or turn off Timer1.	PORTB<2:1> function as normal I/O
Counter	External circuit	T1CON = --xx 0x11 TR1SCCP = ---- -x-1	TMR1 module uses the external signal on the RB1/T1OSO/T1CKI pin as a clock source. TMR1ON can turn on or turn off Timer1. DT1CK can read the signal on the RB1/T1OSO/T1CKI pin.	PORTB<2> functions as normal I/O. PORTB<1> always reads '0' when configured as input. If PORTB<1> is configured as output, reading PORTB<1> will read the data latch. Writing to PORTB<1> will always store the result in the data latch, but not to the RB1/T1OSO/T1CKI pin. If the TMR1CS bit is cleared (TMR1 reverts to the timer mode), then pin PORTB<1> will be driven with the value in the data latch.
	Firmware	T1CON = --xx 0x11 TR1SCCP = ---- -x-0	DATAACP<0> bit drives RB1/T1OSO/T1CKI and produces the TMR1 clock source. TMR1ON can turn on or turn off Timer1. The DATAACP<0> bit, DT1CK, can read and write to the RB1/T1OSO/T1CKI pin.	
	Timer1 oscillator	T1CON = --xx 1x11	RB1/T1OSO/T1CKI and RB2/T1OSI are configured as a 2 pin crystal oscillator. RB1/T1OSI/T1CKI is the clock input for TMR1. TMR1ON can turn on or turn off Timer1. DATAACP<1> bit, DT1CK, always reads '0' as input and can not write to the RB1/T1OSO/T1CK1 pin.	

# PIC16C712/716

## 5.3 Timer1 Oscillator

A crystal oscillator circuit is built in between pins T1OSI (input) and T1OSO (amplifier output). It is enabled by setting control bit T1OSCEN (T1CON<3>). The oscillator is a low-power oscillator rated up to 200 kHz. It will continue to run during Sleep. It is primarily intended for a 32 kHz crystal. Table 5-2 shows the capacitor selection for the Timer1 oscillator.

The Timer1 oscillator is identical to the LP oscillator. The user must provide a software time delay to ensure proper oscillator start-up.

**TABLE 5-2: CAPACITOR SELECTION FOR THE TIMER1 OSCILLATOR**

Osc Type	Freq.	C1	C2
LP	32 kHz	33 pF	33 pF
	100 kHz	15 pF	15 pF
	200 kHz	15 pF	15 pF
<b>These values are for design guidance only.</b>			
<b>Note 1:</b> Higher capacitance increases the stability of oscillator but also increases the start-up time.			
<b>2:</b> Since each resonator/crystal has its own characteristics, the user should consult the resonator/crystal manufacturer for appropriate values of external components.			

## 5.4 Timer1 Interrupt

The TMR1 Register pair (TMR1H:TMR1L) increments from 0000h to FFFFh and rolls over to 0000h. The TMR1 interrupt, if enabled, is generated on overflow which is latched in interrupt flag bit TMR1IF (PIR1<0>). This interrupt can be enabled/disabled by setting/clearing TMR1 interrupt enable bit TMR1IE (PIE1<0>).

## 5.5 Resetting Timer1 using a CCP Trigger Output

If the CCP module is configured in Compare mode to generate a “Special Event Trigger” (CCP1M3:CCP1M0 = 1011), this signal will reset Timer1 and start an A/D conversion (if the A/D module is enabled).

**Note:** The Special Event Triggers from the CCP1 module will not set interrupt flag bit TMR1IF (PIR1<0>).

Timer1 must be configured for either Timer or Synchronized Counter mode to take advantage of this feature. If Timer1 is running in Asynchronous Counter mode, this reset operation may not work.

In the event that a write to Timer1 coincides with a Special Event Trigger from CCP1, the write will take precedence.

In this mode of operation, the CCPR1H:CCPR1L registers pair effectively becomes the period register for Timer1.

**TABLE 5-3: REGISTERS ASSOCIATED WITH TIMER1 AS A TIMER/COUNTER**

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
0Bh,8Bh	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	—	ADIF	—	—	—	CCP1IF	TMR2IF	TMR1IF	-0-- -000	-0-- -000
8Ch	PIE1	—	ADIE	—	—	—	CCP1IE	TMR2IE	TMR1IE	-0-- -000	-0-- -000
0Eh	TMR1L	Holding Register for the Least Significant Byte of the 16-bit TMR1 Register								xxxx xxxx	uuuu uuuu
0Fh	TMR1H	Holding Register for the Most Significant Byte of the 16-bit TMR1 Register								xxxx xxxx	uuuu uuuu
10h	T1CON	—	—	T1CKPS1	T1CKPS0	T1OSCEN	T1SYN $\bar{C}$	TMR1CS	TMR1ON	--00 0000	--uu uuuu
07h	DATAACC P	—	—	—	—	—	DCCP	—	DT1CK	---- -x-x	---- -u-u
87h	TRISCCP	—	—	—	—	—	TCCP	—	TT1CK	---- -1-1	---- -1-1

**Legend:** x = unknown, u = unchanged, — = unimplemented read as ‘0’. Shaded cells are not used by the Timer1 module.



# PIC16C712/716

## 6.0 TIMER2 MODULE

The Timer2 module timer has the following features:

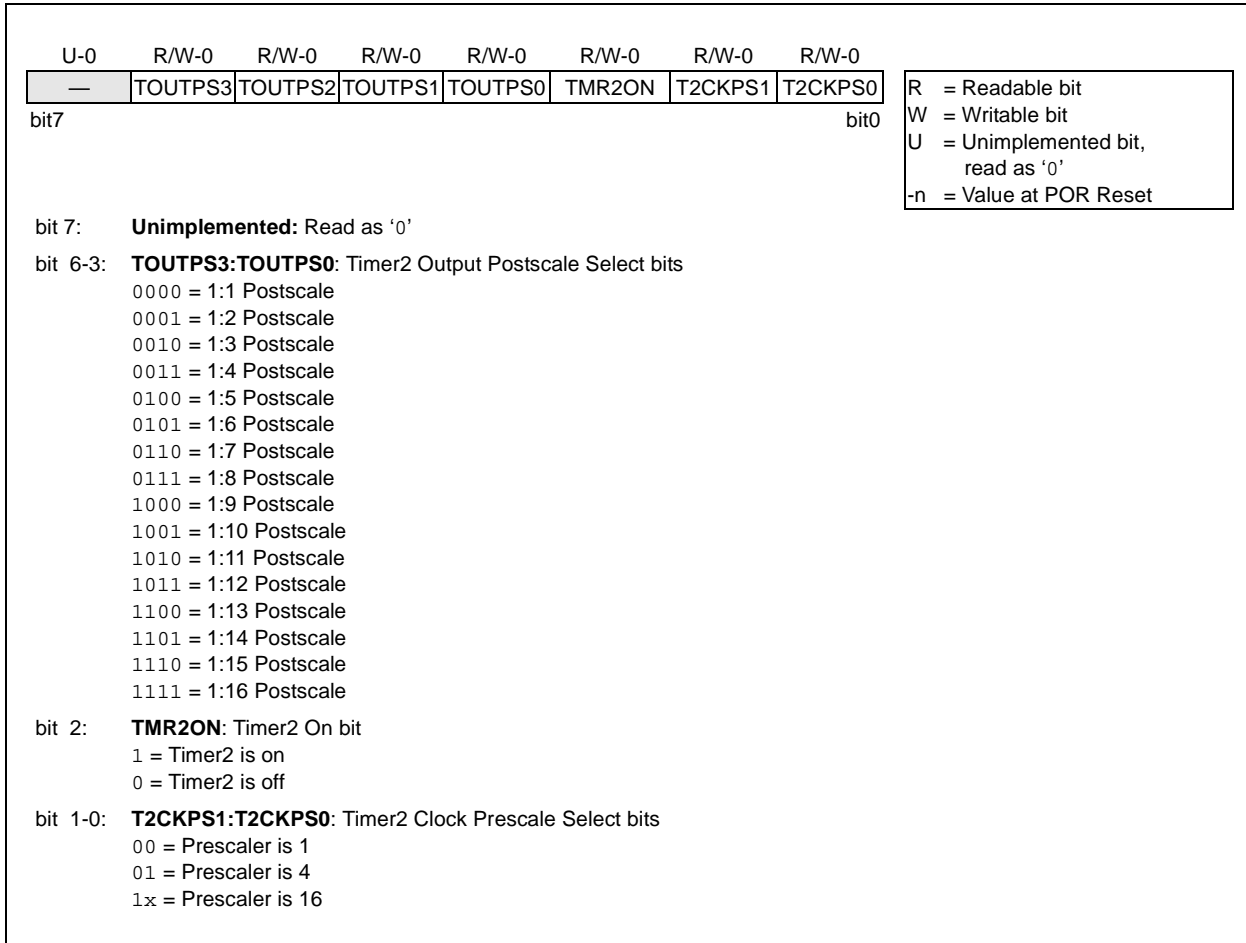
- 8-bit timer (TMR2 register)
- 8-bit period register (PR2)
- Readable and writable (both registers)
- Software programmable prescaler (1:1, 1:4, 1:16)
- Software programmable postscaler (1:1 to 1:16)
- Interrupt on TMR2 match of PR2

Timer2 has a control register, shown in Figure 6-1. Timer2 can be shut off by clearing control bit TMR2ON (T2CON<2>) to minimize power consumption.

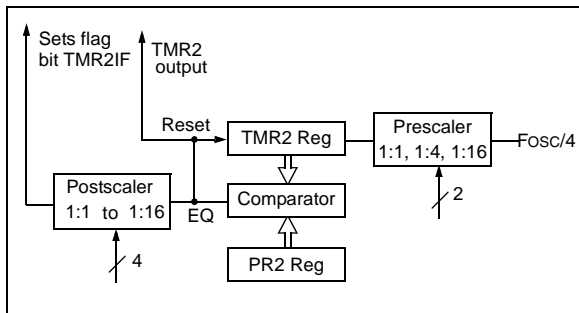
Figure 6-2 is a simplified block diagram of the Timer2 module.

Additional information on timer modules is available in the PIC<sup>®</sup> Mid-Range Reference Manual, (DS33023).

**FIGURE 6-1: T2CON: TIMER2 CONTROL REGISTER (ADDRESS 12h)**



**FIGURE 6-2: TIMER2 BLOCK DIAGRAM**



## 6.1 Timer2 Operation

Timer2 can be used as the PWM time base for PWM mode of the CCP module.

The TMR2 register is readable and writable, and is cleared on any device Reset.

The input clock ( $F_{osc}/4$ ) has a prescale option of 1:1, 1:4 or 1:16, selected by control bits T2CKPS1:T2CKPS0 (T2CON<1:0>).

The match output of TMR2 goes through a 4-bit postscaler (which gives a 1:1 to 1:16 scaling inclusive) to generate a TMR2 interrupt (latched in flag bit TMR2IF, (PIR1<1>)).

The prescaler and postscaler counters are cleared when any of the following occurs:

- a write to the TMR2 register
- a write to the T2CON register
- any device Reset (Power-on Reset,  $\overline{MCLR}$  Reset, Watchdog Timer Reset, or Brown-out Reset)

TMR2 is not cleared when T2CON is written.

## 6.2 Timer2 Interrupt

The Timer2 module has an 8-bit period register PR2. Timer2 increments from 00h until it matches PR2 and then resets to 00h on the next increment cycle. PR2 is a readable and writable register. The PR2 register is initialized to FFh upon Reset.

**TABLE 6-1: REGISTERS ASSOCIATED WITH TIMER2 AS A TIMER/COUNTER**

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
0Bh,8Bh	INTCON	GIE	PEIE	TOIE	INTE	RBIE	TOIF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	—	ADIF	—	—	—	CCP1IF	TMR2IF	TMR1IF	-00- -000	0000 -000
8Ch	PIE1	—	ADIE	—	—	—	CCP1IE	TMR2IE	TMR1IE	-0-- -000	0000 -000
11h	TMR2	Timer2 Module's Register								0000 0000	0000 0000
12h	T2CON	—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	-000 0000	-000 0000
92h	PR2	Timer2 Period Register								1111 1111	1111 1111

**Legend:** x = unknown, u = unchanged, — = unimplemented read as '0'. Shaded cells are not used by the Timer2 module.

## 7.0 CAPTURE/COMPARE/PWM (CCP) MODULE(S)

Each CCP (Capture/Compare/PWM) module contains a 16-bit register, which can operate as a 16-bit capture register, as a 16-bit compare register or as a PWM master/slave Duty Cycle register. Table 7-1 shows the timer resources of the CCP module modes.

Capture/Compare/PWM Register 1 (CCPR1) is comprised of two 8-bit registers: CCPR1L (low byte) and CCPR1H (high byte). The CCP1CON register controls the operation of CCP1. All are readable and writable.

Additional information on the CCP module is available in the PIC® Mid-Range Reference Manual, (DS33023).

**TABLE 7-1: CCP MODE – TIMER RESOURCE**

CCP Mode	Timer Resource
Capture	Timer1
Compare	Timer1
PWM	Timer2

**FIGURE 7-1: CCP1CON REGISTER (ADDRESS 17h)**

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	
—	—	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0	
bit7								bit0

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

bit 7-6: **Unimplemented:** Read as '0'

bit 5-4: **DC1B1:DC1B0:** PWM Least Significant bits  
 Capture Mode: Unused  
 Compare Mode: Unused  
 PWM Mode: These bits are the two LSBs of the PWM duty cycle. The eight MSBs are found in CCPR1L.

bit 3-0: **CCP1M3:CCP1M0:** CCP1 Mode Select bits  
 0000 = Capture/Compare/PWM off (resets CCP1 module)  
 0100 = Capture mode, every falling edge  
 0101 = Capture mode, every rising edge  
 0110 = Capture mode, every 4th rising edge  
 0111 = Capture mode, every 16th rising edge  
 1000 = Compare mode, set output on match (CCP1IF bit is set)  
 1001 = Compare mode, clear output on match (CCP1IF bit is set)  
 1010 = Compare mode, generate software interrupt on match (CCP1IF bit is set, CCP1 pin is unaffected)  
 1011 = Compare mode, trigger special event (CCP1IF bit is set; CCP1 resets TMR1 and starts an A/D conversion (if A/D module is enabled))  
 11xx = PWM mode

**FIGURE 7-2: TRISCCP REGISTER (ADDRESS 87H)**

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	—	—	—	TCCP	—	TT1CK
bit7						bit0	

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

bit 7-3: **Reserved bits; Do Not Use**

bit 2: **TCCP – Tri-state control bit for CCP**  
 0 = Output pin driven  
 1 = Output pin tristated

bit 1: **Reserved bit; Do Not Use**

bit 0: **TT1CK – Tri-state control bit for T1CKI pin**  
 0 = T1CKI pin is an output  
 1 = T1CKI pin is an input

# PIC16C712/716

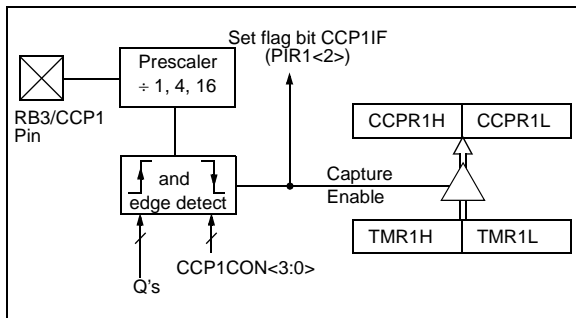
## 7.1 Capture Mode

In Capture mode, CCP1H:CCP1L captures the 16-bit value of the TMR1 register when an event occurs on pin RB3/CCP1. An event is defined as:

- every falling edge
- every rising edge
- every 4th rising edge
- every 16th rising edge

An event is selected by control bits CCP1M3:CCP1M0 (CCP1CON<3:0>). When a capture is made, the interrupt request flag bit CCP1IF (PIR1<2>) is set. It must be cleared in software. If another capture occurs before the value in register CCP1 is read, the old captured value will be lost.

**FIGURE 7-3: CAPTURE MODE OPERATION BLOCK DIAGRAM**



### 7.1.1 CCP PIN CONFIGURATION

In Capture mode, the CCP output must be disabled by setting the TRISCCP<2> bit.

**Note:** If the RB3/CCP1 is configured as an output by clearing the TRISCCP<2> bit, a write to the DCCP bit can cause a capture condition.

### 7.1.2 TIMER1 MODE SELECTION

Timer1 must be running in Timer mode or Synchronized Counter mode for the CCP module to use the capture feature. In Asynchronous Counter mode, the capture operation may not work.

### 7.1.3 SOFTWARE INTERRUPT

When the Capture mode is changed, a false capture interrupt may be generated. The user should keep bit CCP1IE (PIE1<2>) clear to avoid false interrupts and should clear the flag bit CCP1IF following any such change in Operating mode.

### 7.1.4 CCP PRESCALER

There are four prescaler settings, specified by bits CCP1M3:CCP1M0. Whenever the CCP module is turned off, or the CCP module is not in Capture mode, the prescaler counter is cleared. This means that any Reset will clear the prescaler counter.

Switching from one capture prescaler to another may generate an interrupt. Also, the prescaler counter will not be cleared, therefore the first capture may be from a non-zero prescaler. Example 7-1 shows the recommended method for switching between capture prescalers. This example also clears the prescaler counter and will not generate the "false" interrupt.

**EXAMPLE 7-1: CHANGING BETWEEN CAPTURE PRESCALERS**

```
CLRWF  CCP1CON      ;Turn CCP module off
MOVLW  NEW_CAPT_PS  ;Load the W reg with
                    ; the new prescaler
                    ; mode value and CCP ON
MOVWF  CCP1CON      ;Load CCP1CON with this
                    ; value
```

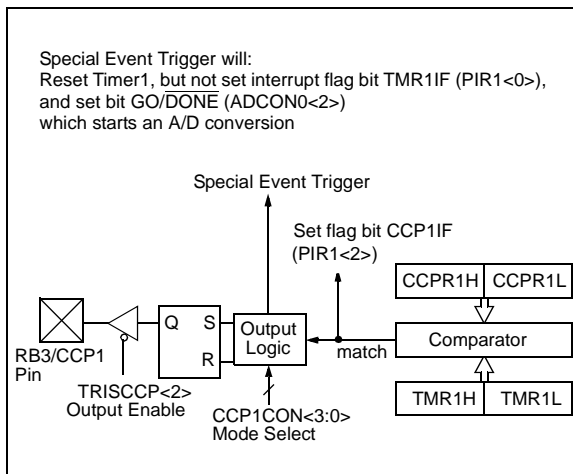
## 7.2 Compare Mode

In Compare mode, the 16-bit CCPR1 register value is constantly compared against the TMR1 register pair value. When a match occurs, the RB3/CCP1 pin is either:

- driven High
- driven Low
- remains Unchanged

The action on the pin is based on the value of control bits CCP1M3:CCP1M0 (CCP1CON<3:0>). At the same time, interrupt flag bit CCP1IF is set.

**FIGURE 7-4: COMPARE MODE OPERATION BLOCK DIAGRAM**



### 7.2.1 CCP PIN CONFIGURATION

The user must configure the RB3/CCP1 pin as the CCP output by clearing the TRISCCP<2> bit.

**Note:** Clearing the CCP1CON register will force the RB3/CCP1 compare output latch to the default low level. This is neither the PORTB I/O data latch nor the DATA CCP latch.

### 7.2.2 TIMER1 MODE SELECTION

Timer1 must be running in Timer mode or Synchronized Counter mode if the CCP module is using the compare feature. In Asynchronous Counter mode, the compare operation may not work.

### 7.2.3 SOFTWARE INTERRUPT MODE

When generate software interrupt is chosen the CCP1 pin is not affected. Only a CCP interrupt is generated (if enabled).

### 7.2.4 SPECIAL EVENT TRIGGER

In this mode, an internal hardware trigger is generated which may be used to initiate an action.

The Special Event Trigger output of CCP1 resets the TMR1 register pair. This allows the CCPR1 register to effectively be a 16-bit programmable period register for Timer1.

The Special Event Trigger output of CCP1 also starts an A/D conversion (if the A/D module is enabled).

**Note:** The Special Event Trigger from the CCP1 module will not set interrupt flag bit TMR1IF (PIR1<0>).

**TABLE 7-2: REGISTERS ASSOCIATED WITH CAPTURE, COMPARE, AND TIMER1**

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
07h	DATA CCP	—	—	—	—	—	DCCP	—	DT1CK	xxxx xxxx	xxxx xuxu
0Bh,8Bh	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	—	ADIF	—	—	—	CCP1IF	TMR2IF	TMR1IF	-0-- -000	-0-- -000
0Eh	TMR1L	Holding Register for the Least Significant Byte of the 16-bit TMR1 Register								xxxx xxxx	uuuu uuuu
0Fh	TMR1H	Holding Register for the Most Significant Byte of the 16-bit TMR1 Register								xxxx xxxx	uuuu uuuu
10h	T1CON	—	—	T1CKPS1	T1CKPS0	T1OSCEN	T1SYNĀ	TMR1CS	TMR1ON	--00 0000	--uu uuuu
15h	CCPR1L	Capture/Compare/PWM Register 1 (LSB)								xxxx xxxx	uuuu uuuu
16h	CCPR1H	Capture/Compare/PWM Register 1 (MSB)								xxxx xxxx	uuuu uuuu
17h	CCP1CON	—	—	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0	--00 0000	--00 0000
87h	TRISCCP	—	—	—	—	—	TCCP	—	TT1CK	xxxx x1x1	xxxx x1x1
8Ch	PIE1	—	ADIE	—	—	—	CCP1IE	TMR2IE	TMR1IE	-0-- -000	-0-- -000

**Legend:** x = unknown, u = unchanged, — = unimplemented read as '0'. Shaded cells are not used by Capture and Timer1.

# PIC16C712/716

## 7.3 PWM Mode

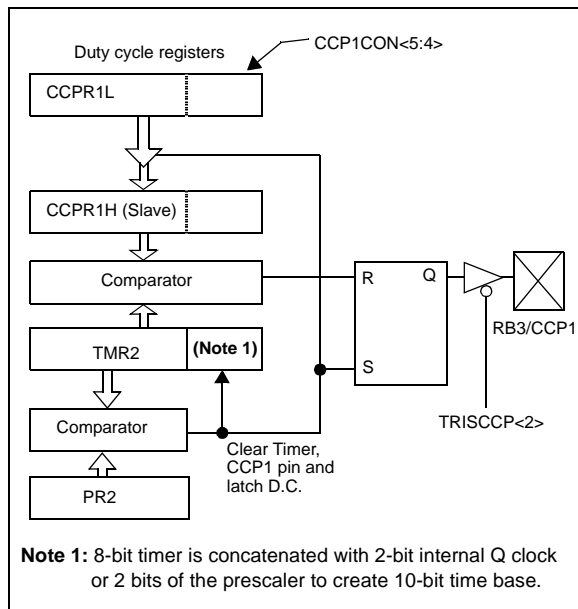
In Pulse Width Modulation (PWM) mode, the CCP1 pin produces up to a 10-bit resolution PWM output. Since the CCP1 pin is multiplexed with the PORTB data latch, the TRISCCP<2> bit must be cleared to make the CCP1 pin an output.

**Note:** Clearing the CCP1CON register will force the CCP1 PWM output latch to the default low level. This is neither the PORTB I/O data latch nor the DATAACP latch.

Figure 7-5 shows a simplified block diagram of the CCP module in PWM mode.

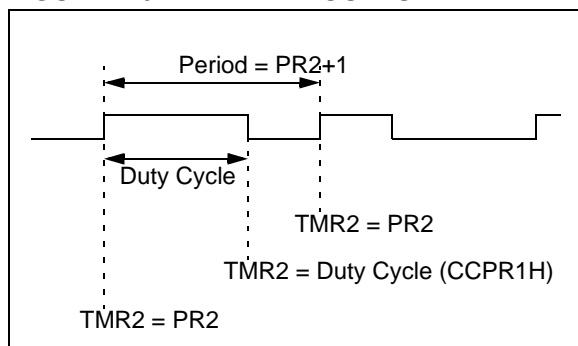
For a step by step procedure on how to set up the CCP module for PWM operation, see **Section 7.3.3 “Set-Up for PWM Operation”**.

**FIGURE 7-5: SIMPLIFIED PWM BLOCK DIAGRAM**



A PWM output (Figure 7-6) has a time base (period) and a time that the output stays high (duty cycle). The frequency of the PWM is the inverse of the period (1/period).

**FIGURE 7-6: PWM OUTPUT**



### 7.3.1 PWM PERIOD

The PWM period is specified by writing to the PR2 register. The PWM period can be calculated using the following formula:

$$PWM \text{ period} = [(PR2) + 1] \cdot 4 \cdot T_{osc} \cdot (TMR2 \text{ prescale value})$$

PWM frequency is defined as  $1 / [PWM \text{ period}]$ .

When TMR2 is equal to PR2, the following three events occur on the next increment cycle:

- TMR2 is cleared
- The CCP1 pin is set (exception: if PWM duty cycle = 0%, the CCP1 pin will not be set)
- The PWM duty cycle is latched from CCPR1L into CCPR1H

**Note:** The Timer2 postscaler (see **Section 6.0 “Timer2 Module”**) is not used in the determination of the PWM frequency. The postscaler could be used to have a servo update rate at a different frequency than the PWM output.

### 7.3.2 PWM DUTY CYCLE

The PWM duty cycle is specified by writing to the CCPR1L register and to the CCP1CON<5:4> bits. Up to 10-bit resolution is available. The CCPR1L contains the eight MSBs and the CCP1CON<5:4> contains the two LSBs. This 10-bit value is represented by CCPR1L:CCP1CON<5:4>. The following equation is used to calculate the PWM duty cycle in time:

$$PWM \text{ duty cycle} = (CCPR1L:CCP1CON<5:4>) \cdot T_{osc} \cdot (TMR2 \text{ prescale value})$$

CCPR1L and CCP1CON<5:4> can be written to at any time, but the duty cycle value is not latched into CCPR1H until after a match between PR2 and TMR2 occurs (i.e., the period is complete). In PWM mode, CCPR1H is a read-only register.

The CCPR1H register and a 2-bit internal latch are used to double buffer the PWM duty cycle. This double buffering is essential for glitchless PWM operation.

When the CCPR1H and 2-bit latch match TMR2 concatenated with an internal 2-bit Q clock or 2 bits of the TMR2 prescaler, the CCP1 pin is cleared.

Maximum PWM resolution (bits) for a given PWM frequency:

$$= \frac{\log\left(\frac{F_{osc}}{F_{PWM}}\right)}{\log(2)} \text{ bits}$$

**Note:** If the PWM duty cycle value is longer than the PWM period the CCP1 pin will not be cleared.

For an example PWM period and duty cycle calculation, see the PIC® Mid-Range Reference Manual, (DS33023).

## 7.3.3 SET-UP FOR PWM OPERATION

The following steps should be taken when configuring the CCP module for PWM operation:

1. Set the PWM period by writing to the PR2 register.
2. Set the PWM duty cycle by writing to the CCPR1L register and CCP1CON<5:4> bits.
3. Make the CCP1 pin an output by clearing the TRISCCP<2> bit.
4. Set the TMR2 prescale value and enable Timer2 by writing to T2CON.
5. Configure the CCP1 module for PWM operation.

**TABLE 7-3: EXAMPLE PWM FREQUENCIES AND RESOLUTIONS AT 20 MHz**

PWM Frequency	1.22 kHz	4.88 kHz	19.53 kHz	78.12 kHz	156.3 kHz	208.3 kHz
Timer Prescaler (1, 4, 16)	16	4	1	1	1	1
PR2 Value	0xFF	0xFF	0xFF	0x3F	0x1F	0x17
Maximum Resolution (bits)	10	10	10	8	7	5.5

**TABLE 7-4: REGISTERS ASSOCIATED WITH PWM AND TIMER2**

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
07h	DATA CCP	—	—	—	—	—	DCCP	—	DT1CK	xxxx xxxx	xxxx xuxu
0Bh,8Bh	INTCON	GIE	PEIE	TOIE	INTE	RBIE	TOIF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	—	ADIF	—	—	—	CCP1IF	TMR2IF	TMR1IF	-0-- -000	-0-- -000
11h	TMR2	Timer2 Module's Register								0000 0000	0000 0000
12h	T2CON	—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	-000 0000	-000 0000
15h	CCPR1L	Capture/Compare/PWM Register 1 (LSB)								xxxx xxxx	uuuu uuuu
16h	CCPR1H	Capture/Compare/PWM Register 1 (MSB)								xxxx xxxx	uuuu uuuu
17h	CCP1CON	—	—	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0	--00 0000	--00 0000
87h	TRISCCP	—	—	—	—	—	TCCP	—	TT1CK	xxxx x1x1	xxxx x1x1
8Ch	PIE1	—	ADIE	—	—	—	CCP1IE	TMR2IE	TMR1IE	-0-- -000	-0-- -000
92h	PR2	Timer2 Module's Period Register								1111 1111	1111 1111

**Legend:** x = unknown, u = unchanged, — = unimplemented read as '0'. Shaded cells are not used by PWM and Timer2.

# PIC16C712/716

## 7.4 CCP1 Module and PORTB Operation

When the CCP module is disabled, PORTB<3> operates as a normal I/O pin. When the CCP module is enabled, PORTB<3> operation is affected. Multiplexing details of the CCP1 module are shown on PORTB<3>, refer to Figure 3.6.

Table 7-5 below shows the effects of the CCP module operation on PORTB<3>

**TABLE 7-5: CCP1 MODULE AND PORTB OPERATION**

CCP1 Module Mode	Control Bits	CCP1 Module Operation	PORTB<3> Operation
Off	CCP1CON = --xx 0000	Off	PORTB<3> functions as normal I/O.
Capture	CCP1CON = --xx 01xx TRISCCP = ---- -1-x	The CCP1 module will capture an event on the RB3/CCP1 pin which is driven by an external circuit. The DCCP bit can read the signal on the RB3/CCP1 pin.	PORTB<3> always reads '0' when configured as input. If PORTB<3> is configured as output, reading PORTB<3> will read the data latch. Writing to PORTB<3> will always store the result in the data latch, but it does not drive the RB3/CCP1 pin.
	CCP1CON = --xx 01xx TRISCCP = ---- -0-x	The CCP1 module will capture an event on the RB3/CCP1 pin which is driven by the DCCP bit. The DCCP bit can read the signal on the RB3/CCP1 pin.	
Compare	CCP1CON = --xx 10xx TRISCCP = ---- -0-x	The CCP1 module produces an output on the RB3/CCP1 pin when a compare event occurs. The DCCP bit can read the signal on the RB3/CCP1 pin.	
PWM	CCP1CON = --xx 11xx TRISCCP = ---- -0-x	The CCP1 module produces the PWM signal on the RB3/CCP1 pin. The DCCP bit can read the signal on the RB3/CCP1 pin.	



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

The Analog-to-Digital (A/D) Converter module has four 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 RA3/AN3/VREF pin.

The A/D converter has a unique feature of being able to operate while the device is in Sleep mode. To operate in Sleep, the A/D conversion clock must be derived from the A/D's internal RC oscillator.

Additional information on the A/D module is available in the PIC<sup>®</sup> Mid-Range Reference Manual, (DS33023).

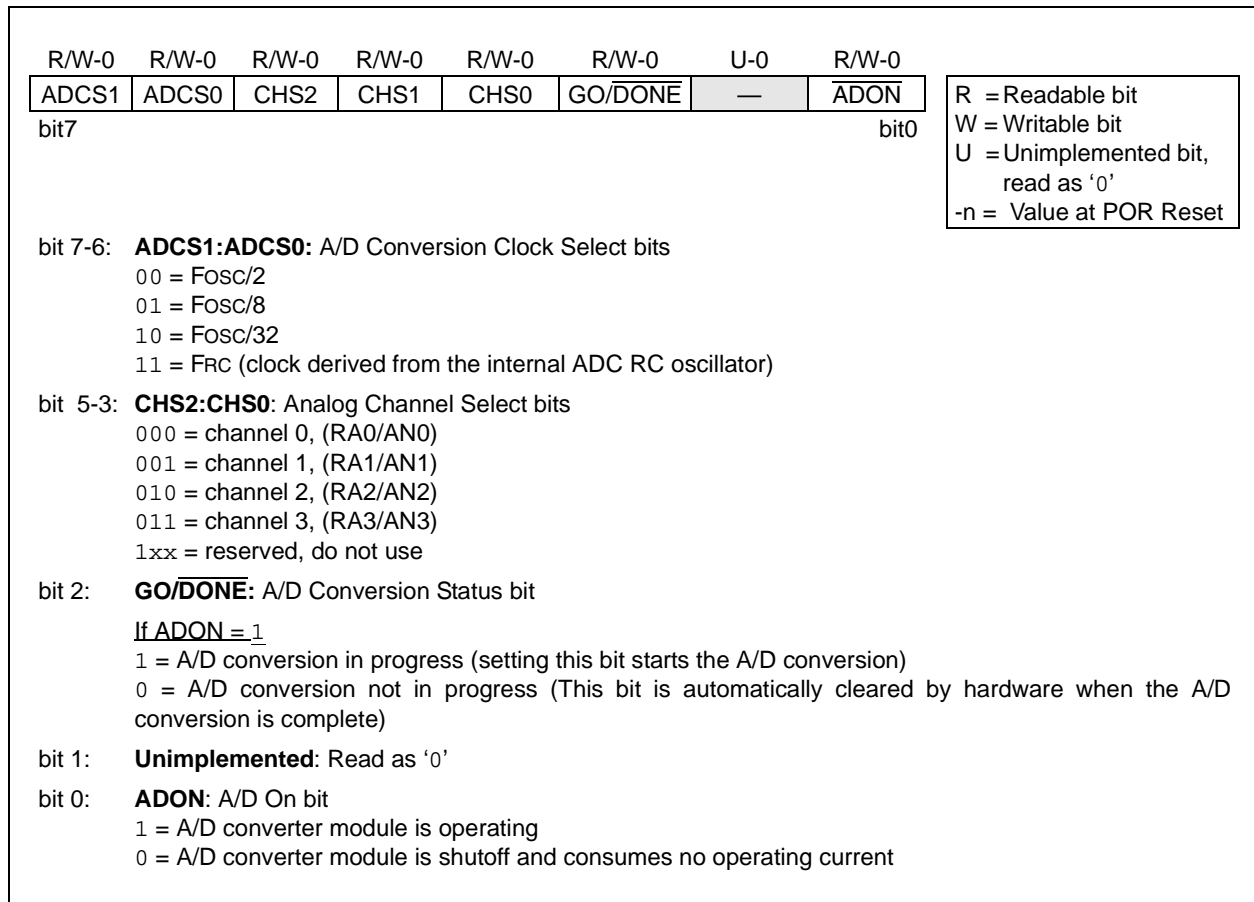
The A/D module has three registers. These registers are:

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

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.

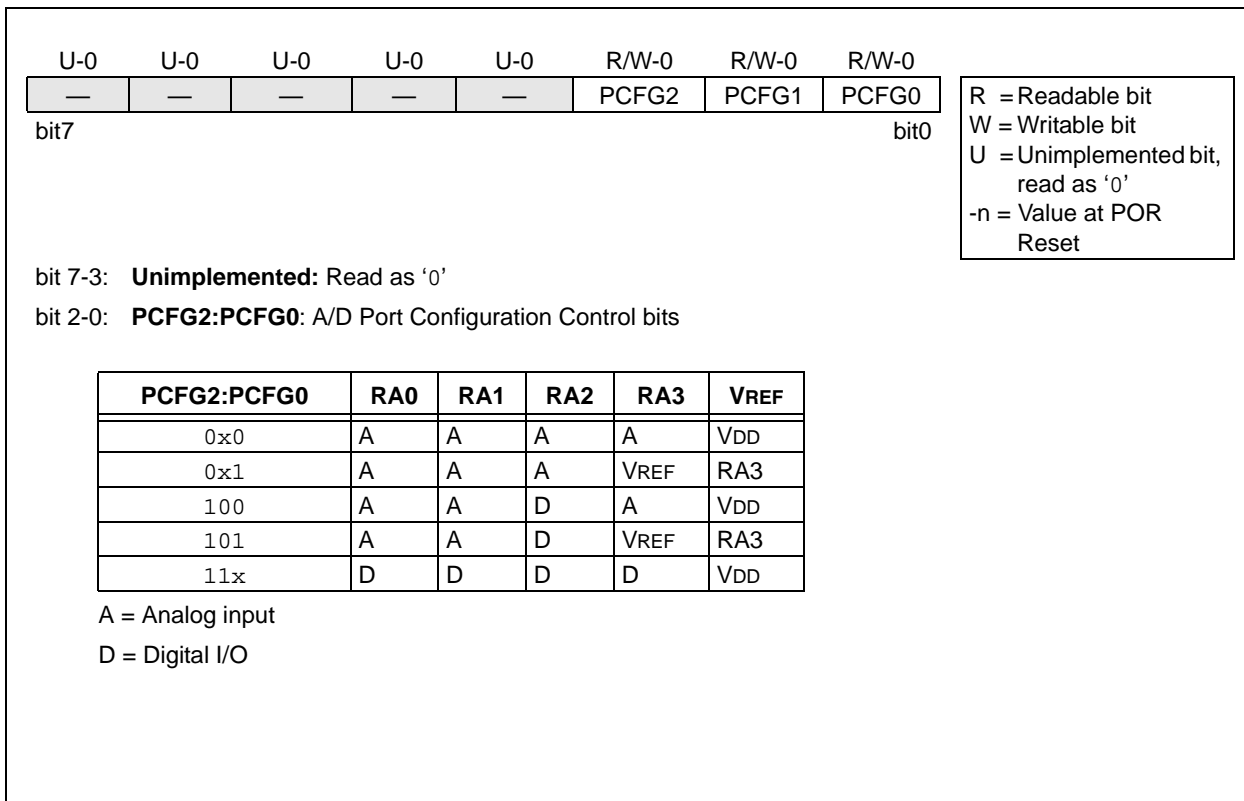
The ADCON0 register, shown in Figure 8-1, controls the operation of the A/D module. The ADCON1 register, shown in Figure 8-2, configures the functions of the port pins. The port pins can be configured as analog inputs (RA3 can also be a voltage reference) or as digital I/O.

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



# PIC16C712/716

**FIGURE 8-2: ADCON1 REGISTER (ADDRESS 9Fh)**



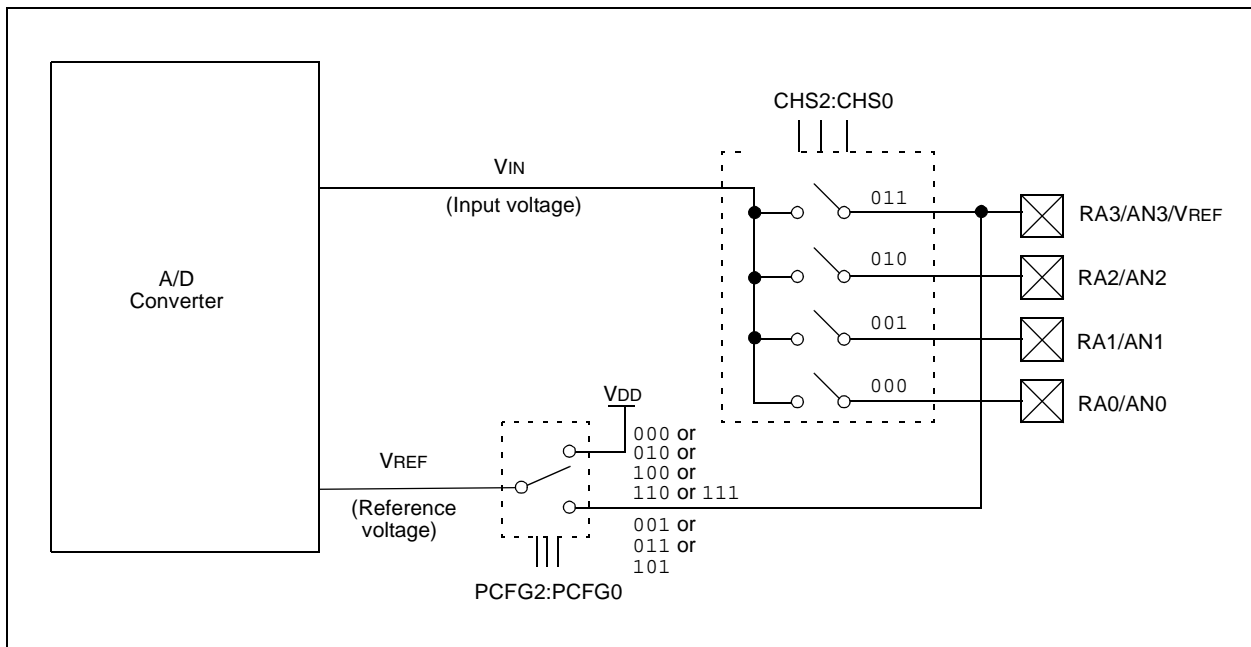
The ADRES register contains the result of the A/D conversion. When the A/D conversion is complete, the result is loaded into the ADRES register, the  $\overline{\text{GO/DONE}}$  bit (ADCON0<2>) is cleared and the A/D Interrupt Flag bit ADIF is set. The block diagram of the A/D module is shown in Figure 8-3.

The value that is in the ADRES register is not modified for a Power-on Reset. The ADRES register will contain unknown data after a Power-on Reset.

After the A/D module has been configured as desired, the selected channel must be acquired before the conversion is started. The analog input channels must have their corresponding TRIS bits selected as an input. To determine acquisition time, see **Section 8.1 “A/D Acquisition Requirements”**. After this acquisition time has elapsed, the A/D conversion can be started. The following steps should be followed for doing an A/D conversion:

1. Configure the A/D module:
  - Configure analog pins/voltage reference/ and digital I/O (ADCON1)
  - Select A/D input channel (ADCON0)
  - Select A/D conversion clock (ADCON0)
  - Turn on A/D module (ADCON0)
2. Configure A/D interrupt (if desired):
  - Clear ADIF bit
  - Set ADIE bit
  - Set GIE bit
3. Wait the required acquisition time.
4. Start conversion:
  - Set  $\overline{\text{GO/DONE}}$  bit (ADCON0)
5. Wait for A/D conversion to complete, by either:
  - Polling for the  $\overline{\text{GO/DONE}}$  bit to be cleared
  - OR
  - Waiting for the A/D interrupt
6. Read A/D Result register (ADRES), clear bit ADIF if required.
7. For the next conversion, go to step 1 or step 2 as required. The A/D conversion time per bit is defined as TAD. A minimum wait of 2TAD is required before next acquisition starts.

**FIGURE 8-3: A/D BLOCK DIAGRAM**



# PIC16C712/716

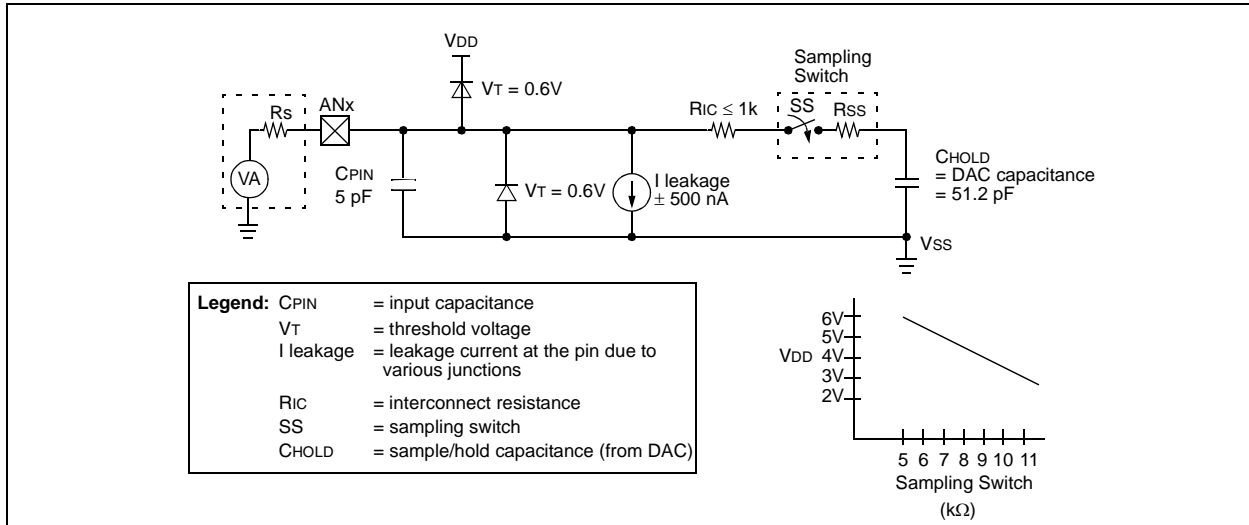
## 8.1 A/D Acquisition Requirements

For the A/D converter to meet its specified accuracy, the Charge Holding capacitor (CHOLD) must be allowed to fully charge to the input channel voltage level. The analog input model is shown in Figure 8-4. The source impedance ( $R_s$ ) and the internal sampling switch ( $R_{SS}$ ) impedance directly affect the time required to charge the capacitor CHOLD. The sampling switch ( $R_{SS}$ ) impedance varies over the device voltage ( $V_{DD}$ ). The source impedance affects the offset voltage at the analog input (due to pin leakage current). **The maximum recommended impedance for analog sources is 10 k $\Omega$ .** After the analog input channel is selected (changed) this acquisition must be done before the conversion can be started.

To calculate the minimum acquisition time,  $T_{ACQ}$ , see the PIC<sup>®</sup> Mid-Range Reference Manual, (DS33023). This equation calculates the acquisition time to within 1/2 LSB error (512 steps for the A/D). The 1/2 LSB error is the maximum error allowed for the A/D to meet its specified accuracy.

**Note:** When the conversion is started, the hold-ing capacitor is disconnected from the input pin.

**FIGURE 8-4: ANALOG INPUT MODEL**



## 8.2 Selecting the A/D Conversion Clock

The A/D conversion time per bit is defined as  $T_{AD}$ . The A/D conversion requires  $9.5T_{AD}$  per 8-bit conversion. The source of the A/D conversion clock is software selectable. The four possible options for  $T_{AD}$  are:

- $2T_{OSC}$
- $8T_{OSC}$
- $32T_{OSC}$
- Internal RC oscillator

For correct A/D conversions, the A/D conversion clock ( $T_{AD}$ ) must be selected to ensure a minimum  $T_{AD}$  time of  $1.6 \mu\text{s}$ .

Table 8-1 shows the resultant  $T_{AD}$  times derived from the device operating frequencies and the A/D clock source selected.

## 8.3 Configuring Analog Port Pins

The ADCON1 and TRISA registers control the operation of the A/D port pins. The port pins that are desired as analog inputs must have their corresponding TRIS bits set (input). If the TRIS bit is cleared (output), the digital output level ( $V_{OH}$  or  $V_{OL}$ ) will be converted.

The A/D operation is independent of the state of the CHS2:CHS0 bits and the TRIS bits.

**Note 1:** When reading the port register, all pins configured as analog input channels will read as cleared (a low level). Pins configured as digital inputs, will convert an analog input. Analog levels on a digitally configured input will not affect the conversion accuracy.

**2:** Analog levels on any pin that is defined as a digital input (including the AN3:AN0 pins), may cause the input buffer to consume current that is out of the devices specification.

**TABLE 8-1:  $T_{AD}$  vs. DEVICE OPERATING FREQUENCIES**

AD Clock Source ( $T_{AD}$ )		Device Frequency			
Operation	ADCS1:ADCS0	20 MHz	5 MHz	1.25 MHz	333.33 kHz
$2T_{OSC}$	00	100 ns <sup>(2)</sup>	400 ns <sup>(2)</sup>	1.6 $\mu\text{s}$	6 $\mu\text{s}$
$8T_{OSC}$	01	400 ns <sup>(2)</sup>	1.6 $\mu\text{s}$	6.4 $\mu\text{s}$	24 $\mu\text{s}$ <sup>(3)</sup>
$32T_{OSC}$	10	1.6 $\mu\text{s}$	6.4 $\mu\text{s}$	25.6 $\mu\text{s}$ <sup>(3)</sup>	96 $\mu\text{s}$ <sup>(3)</sup>
RC <sup>(5)</sup>	11	2-6 $\mu\text{s}$ <sup>(1,4)</sup>	2-6 $\mu\text{s}$ <sup>(1,4)</sup>	2-6 $\mu\text{s}$ <sup>(1,4)</sup>	2-6 $\mu\text{s}$ <sup>(1)</sup>

**Legend:** Shaded cells are outside of recommended range.

**Note 1:** The RC source has a typical  $T_{AD}$  time of 4  $\mu\text{s}$ .

**2:** These values violate the minimum required  $T_{AD}$  time.

**3:** For faster conversion times, the selection of another clock source is recommended.

**4:** When device frequency is greater than 1 MHz, the RC A/D conversion clock source is recommended for Sleep operation only.

**5:** For extended voltage devices (LC), please refer to Electrical Specifications section.

# PIC16C712/716

## 8.4 A/D Conversions

**Note:** The  $\overline{\text{GO/DONE}}$  bit should **NOT** be set in the same instruction that turns on the A/D.

## 8.5 Use of the CCP Trigger

An A/D conversion can be started by the “Special Event Trigger” of the CCP1 module. This requires that the CCP1M3:CCP1M0 bits (CCP1CON<3:0>) be programmed as 1011 and that the A/D module is enabled (ADON bit is set). When the trigger occurs, the

$\overline{\text{GO/DONE}}$  bit will be set, starting the A/D conversion, and the Timer1 counter will be reset to zero. Timer1 is reset to automatically repeat the A/D acquisition period with minimal software overhead (moving the ADRES to the desired location). The appropriate analog input channel must be selected and the minimum acquisition done before the “Special Event Trigger” sets the  $\overline{\text{GO/DONE}}$  bit (starts a conversion).

If the A/D module is not enabled (ADON is cleared), then the “Special Event Trigger” will be ignored by the A/D module, but will still reset the Timer1 counter.

**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 POR, BOR	Value on all other Resets		
05h	PORTA	—	—	— <sup>(1)</sup>	RA4	RA3	RA2	RA1	RA0	--xx xxxx	--xu uuuu		
0Bh,8Bh	INTCON	GIE	PEIE	TOIE	INTE	RBIE	TOIF	INTF	RBIF	0000 000x	0000 000u		
0Ch	PIR1	—	ADIF	—	—	—	CCP1IF	TMR2IF	TMR1IF	-0-- -000	-0-- -000		
1Eh	ADRES	A/D Result Register								xxxx xxxx	uuuu uuuu		
1Fh	ADCON0	ADCS1	ADCS0	CHS2	CHS1	CHS0	$\overline{\text{GO/DONE}}$	—	ADON	0000 00-0	0000 00-0		
85h	TRISA	—	—	— <sup>(1)</sup>	PORTA Data Direction Register					---	1111	---	1111
8Ch	PIE1	—	ADIE	—	—	—	CCP1IE	TMR2IE	TMR1IE	-0-- -000	-0-- 0000		
9Fh	ADCON1	—	—	—	—	—	PCFG2	PCFG1	PCFG0	---- -000	---- -000		

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

**Note 1:** Reserved bits; Do Not Use.

## 9.0 SPECIAL FEATURES OF THE CPU

The PIC16C712/716 devices have a host of features intended to maximize system reliability, minimize cost through elimination of external components, provide power-saving operating modes and offer code protection. These are:

- OSC Selection
- Reset:
  - Power-on Reset (POR)
  - Power-up Timer (PWRT)
  - Oscillator Start-up Timer (OST)
  - Brown-out Reset (BOR)
- Interrupts
- Watchdog Timer (WDT)
- Sleep
- Code protection
- ID locations
- In-Circuit Serial Programming™ (ICSP™)

These devices have a Watchdog Timer, which can be shut off only through Configuration bits. It runs off its own RC oscillator for added reliability. There are two timers that offer necessary delays on power-up. One is the Oscillator Start-up Timer (OST), intended to keep the chip in Reset until the crystal oscillator is stable. The other is the Power-up Timer (PWRT), which provides a fixed delay on power-up only and is designed to keep the part in Reset while the power supply stabilizes. With these two timers on-chip, most applications need no external Reset circuitry.

Sleep mode is designed to offer a very low-current Power-Down mode. The user can wake-up from Sleep through external Reset, Watchdog Timer Wake-up, or through an interrupt. Several oscillator options are also made available to allow the part to fit the application. The RC oscillator option saves system cost, while the LP crystal option saves power. A set of Configuration bits are used to select various options.

Additional information on special features is available in the PIC® Mid-Range Reference Manual, (DS33023).

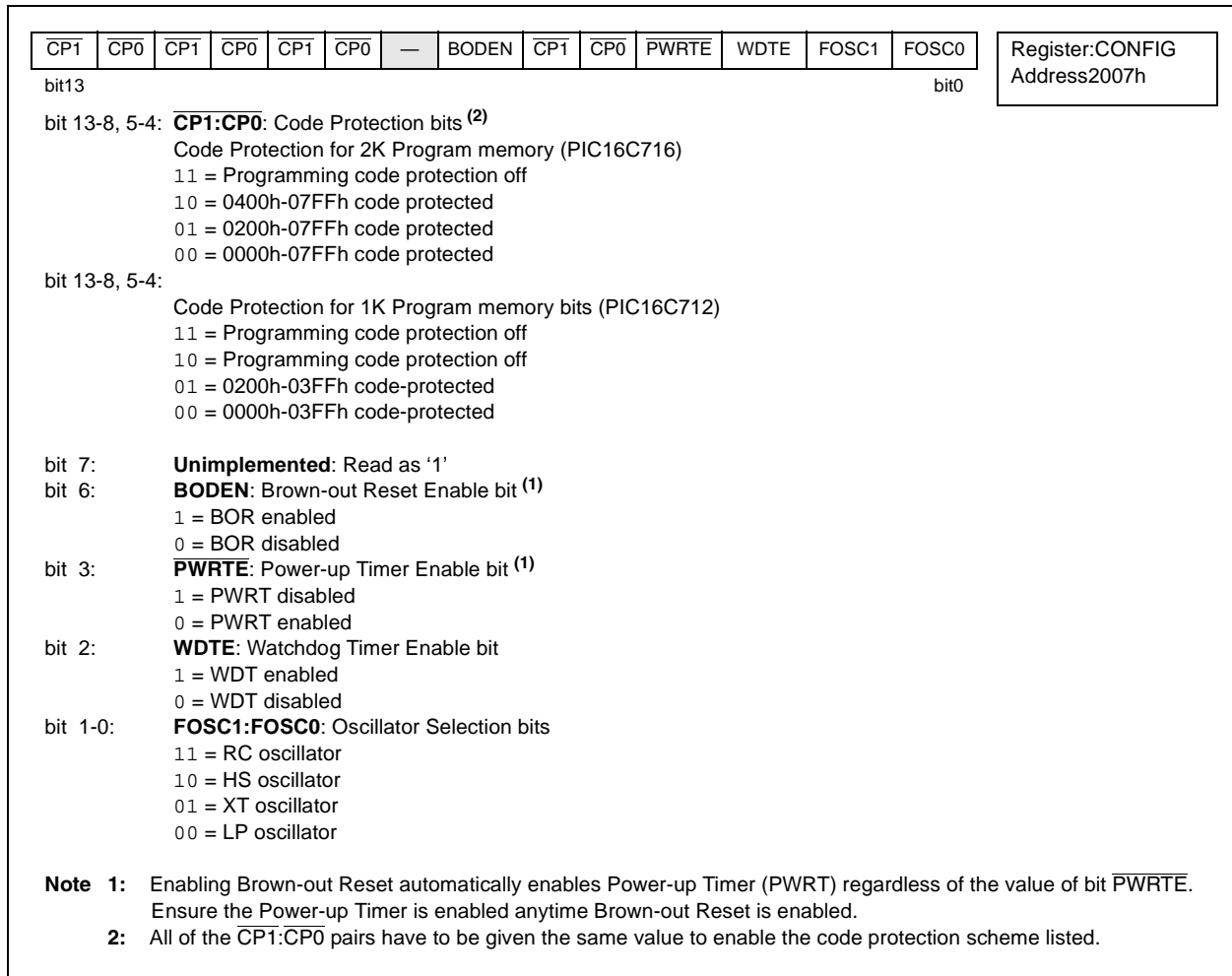
## 9.1 Configuration Bits

The Configuration bits can be programmed (read as '0') or left unprogrammed (read as '1') to select various device configurations. These bits are mapped in program memory location 2007h.

The user will note that address 2007h is beyond the user program memory space. In fact, it belongs to the special test/configuration memory space (2000h-3FFFh), which can be accessed only during programming.

# PIC16C712/716

**FIGURE 9-1: CONFIGURATION WORD**





## 9.2 Oscillator Configurations

### 9.2.1 OSCILLATOR TYPES

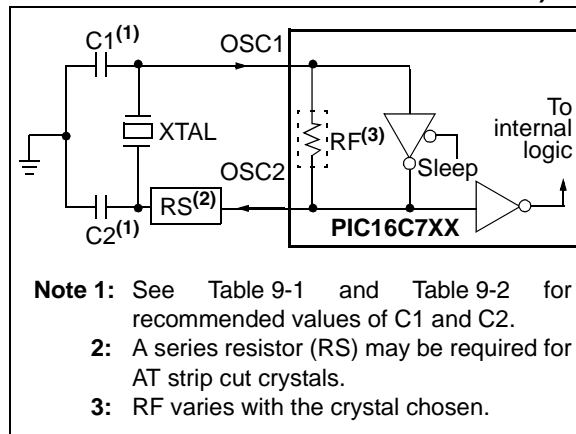
The PIC16CXXX can be operated in four different Oscillator modes. The user can program two Configuration bits (FOSC1 and FOSC0) to select one of these four modes:

- LP Low-Power Crystal
- XT Crystal/Resonator
- HS High-Speed Crystal/Resonator
- RC Resistor/Capacitor

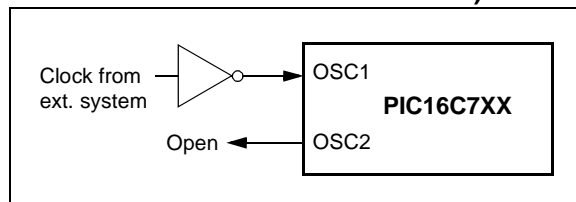
### 9.2.2 CRYSTAL OSCILLATOR/CERAMIC RESONATORS

In XT, LP or HS modes, a crystal or ceramic resonator is connected to the OSC1/CLKIN and OSC2/CLKOUT pins to establish oscillation (Figure 9-2). The PIC16CXXX 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, LP or HS modes, the device can have an external clock source to drive the OSC1/CLKIN pin (Figure 9-3).

**FIGURE 9-2: CRYSTAL/CERAMIC RESONATOR OPERATION (HS, XT OR LP OSC CONFIGURATION)**



**FIGURE 9-3: EXTERNAL CLOCK INPUT OPERATION (HS, XT OR LP OSC CONFIGURATION)**



**TABLE 9-1: CERAMIC RESONATORS**

Ranges Tested:			
Mode	Freq	OSC1	OSC2
XT	455 kHz	68-100 pF	68-100 pF
	2.0 MHz	15-68 pF	15-68 pF
	4.0 MHz	15-68 pF	15-68 pF
HS	8.0 MHz	10-68 pF	10-68 pF
	16.0 MHz	10-22 pF	10-22 pF

**These values are for design guidance only. See notes at bottom of page.**

**TABLE 9-2: CAPACITOR SELECTION FOR CRYSTAL OSCILLATOR**

Osc Type	Crystal Freq	Cap. Range C1	Cap. Range C2
LP	32 kHz	33 pF	33 pF
	200 kHz	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
HS	4 MHz	15 pF	15 pF
	8 MHz	15-33 pF	15-33 pF
	20 MHz	15-33 pF	15-33 pF

**These values are for design guidance only. See notes at bottom of page.**

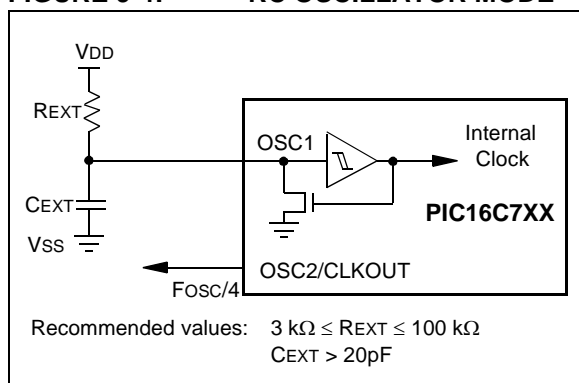
- Note 1:** Recommended values of C1 and C2 are identical to the ranges tested (Table 9-1).  
**2:** Higher capacitance increases the stability of the oscillator, but also increases the start-up time.  
**3:** Since each resonator/crystal has its own characteristics, the user should consult the resonator/crystal manufacturer for appropriate values of external components.  
**4:** Rs may be required in HS mode, as well as XT mode to avoid overdriving crystals with low drive level specification.

# PIC16C712/716

## 9.2.3 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 (R<sub>EXT</sub>) and capacitor (C<sub>EXT</sub>) 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 C<sub>EXT</sub> values. The user also needs to take into account variation due to tolerance of external R and C components used. Figure 9-4 shows how the R/C combination is connected to the PIC16CXXX.

**FIGURE 9-4: RC OSCILLATOR MODE**



## 9.3 Reset

The PIC16CXXX differentiates between various kinds of Reset:

- Power-on Reset (POR)
- $\overline{\text{MCLR}}$  Reset during normal operation
- $\overline{\text{MCLR}}$  Reset during Sleep
- WDT Reset (during normal operation)
- WDT Wake-up (during Sleep)
- Brown-out Reset (BOR)

Some registers are not affected in any Reset condition; their status is unknown on POR and unchanged in any other Reset. Most other registers are reset to a “Reset state” on Power-on Reset (POR), on the  $\overline{\text{MCLR}}$  and WDT Reset, on  $\overline{\text{MCLR}}$  Reset during Sleep and Brown-out Reset (BOR). They are not affected by a WDT Wake-up, which is viewed as the resumption of normal operation. The  $\overline{\text{TO}}$  and  $\overline{\text{PD}}$  bits are set or cleared differently in different Reset situations as indicated in Table 9-4. These bits are used in software to determine the nature of the Reset. See Table 9-6 for a full description of Reset states of all registers.

A simplified block diagram of the on-chip Reset circuit is shown in Figure 9-6.

The PIC microcontrollers have a  $\overline{\text{MCLR}}$  noise filter in the  $\overline{\text{MCLR}}$  Reset path. The filter will detect and ignore small pulses.

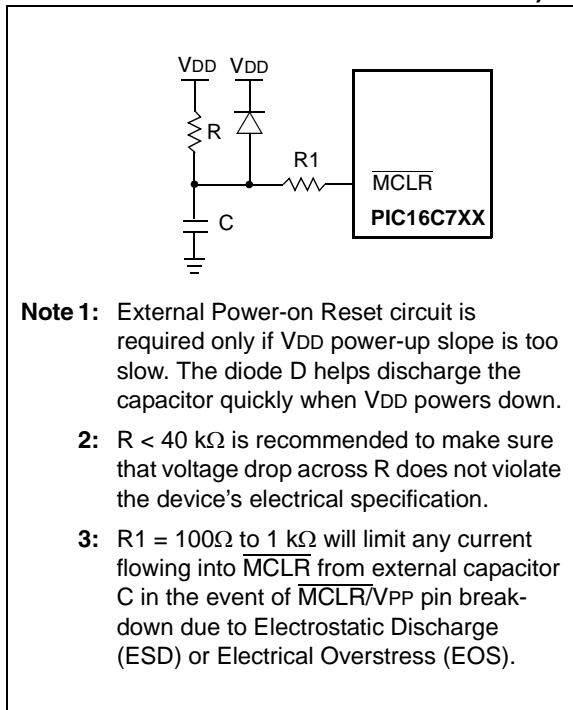
It should be noted that a WDT Reset does not drive  $\overline{\text{MCLR}}$  pin low.

## 9.4 Power-On Reset (POR)

A Power-on Reset pulse is generated on-chip when  $V_{DD}$  rise is detected (to a level of 1.5V-2.1V). To take advantage of the POR, just tie the  $\overline{MCLR}$  pin directly (or through a resistor) to  $V_{DD}$ . This will eliminate external RC components usually needed to create a Power-on Reset. A maximum rise time for  $V_{DD}$  is specified (parameter D004). For a slow rise time, see Figure 9-5.

When the device starts normal operation (exits the Reset condition), device operating parameters (voltage, frequency, temperature,...) must be met to ensure operation. If these conditions are not met, the device must be held in Reset until the operating conditions are met. Brown-out Reset may be used to meet the start-up conditions.

**FIGURE 9-5: EXTERNAL POWER-ON RESET CIRCUIT (FOR SLOW  $V_{DD}$  POWER-UP)**



## 9.5 Power-up Timer (PWRT)

The Power-up Timer provides a fixed nominal time-out (parameter #33), on power-up only, from the POR. The Power-up Timer operates on an internal RC oscillator. The chip is kept in Reset as long as the PWRT is active. The PWRT's time delay allows  $V_{DD}$  to rise to an acceptable level. A Configuration bit is provided to enable/disable the PWRT.

The power-up time delay will vary from chip to chip due to  $V_{DD}$ , temperature, and process variation. See DC parameters for details.

## 9.6 Oscillator Start-up Timer (OST)

The Oscillator Start-up Timer (OST) provides a 1024 oscillator cycle (from OSC1 input) delay after the PWRT delay is over (parameter #32). This ensures that the crystal oscillator or resonator has started and stabilized.

The OST time-out is invoked only for XT, LP and HS modes and only on Power-on Reset or wake-up from Sleep.

## 9.7 Brown-Out Reset (BOR)

The PIC16C712/716 members have on-chip Brown-out Reset circuitry. A Configuration bit, BODEN, can disable (if clear/programmed) or enable (if set) the Brown-out Reset circuitry. If  $V_{DD}$  falls below 4.0V, refer to  $V_{BOR}$  parameter D005( $V_{BOR}$ ) for a time greater than parameter (TBOR) in Table 12-6. The brown-out situation will reset the chip. A Reset is not guaranteed to occur if  $V_{DD}$  falls below 4.0V for less than parameter (TBOR).

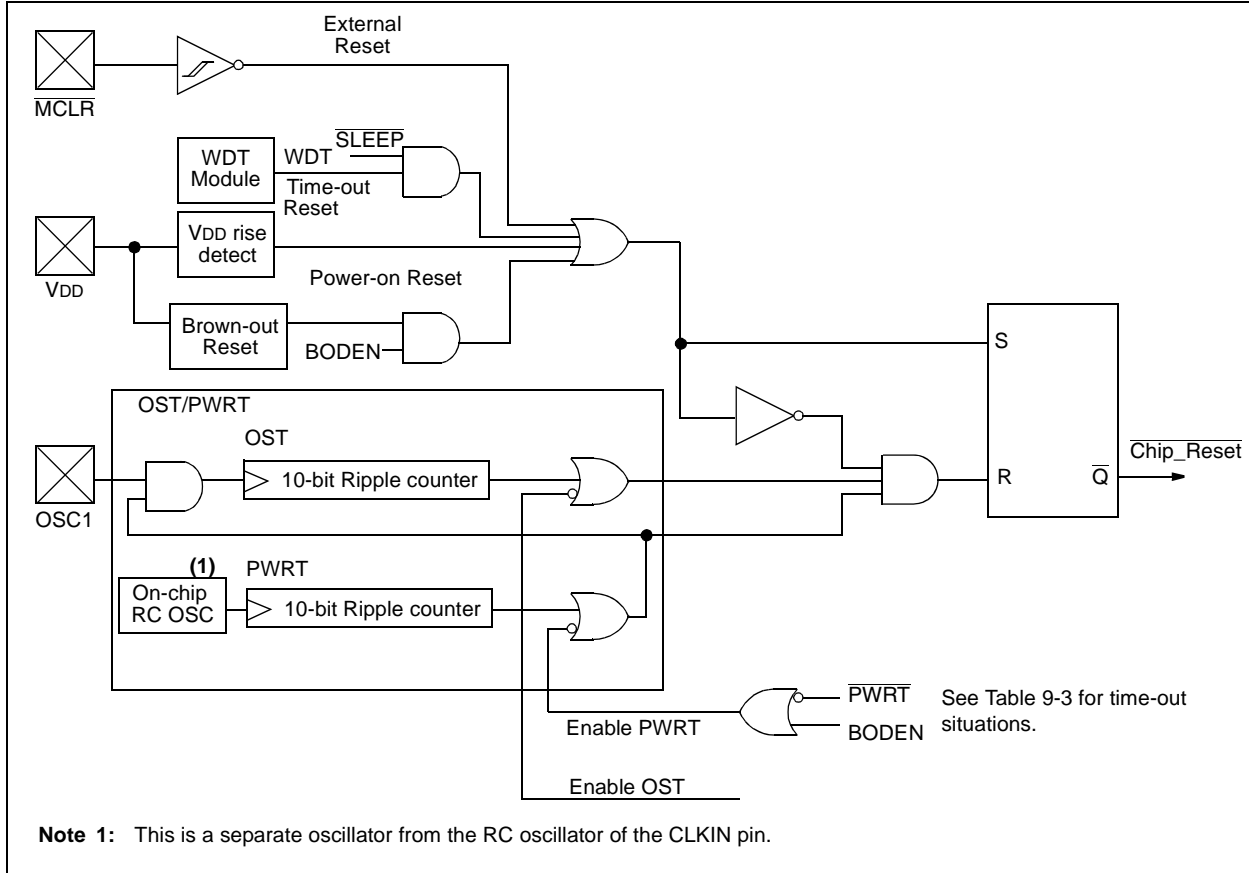
On any Reset (Power-on, Brown-out, Watchdog, etc.) the chip will remain in Reset until  $V_{DD}$  rises above  $V_{BOR}$ . The Power-up Timer will now be invoked and will keep the chip in Reset an additional 72 ms.

If  $V_{DD}$  drops below  $V_{BOR}$  while the Power-up Timer is running, the chip will go back into a Brown-out Reset and the Power-up Timer will be re-initialized. Once  $V_{DD}$  rises above  $V_{BOR}$ , the Power-Up Timer will execute a 72 ms Reset. The Power-up Timer should always be enabled when Brown-out Reset is enabled. Figure 9-7 shows typical Brown-out situations.

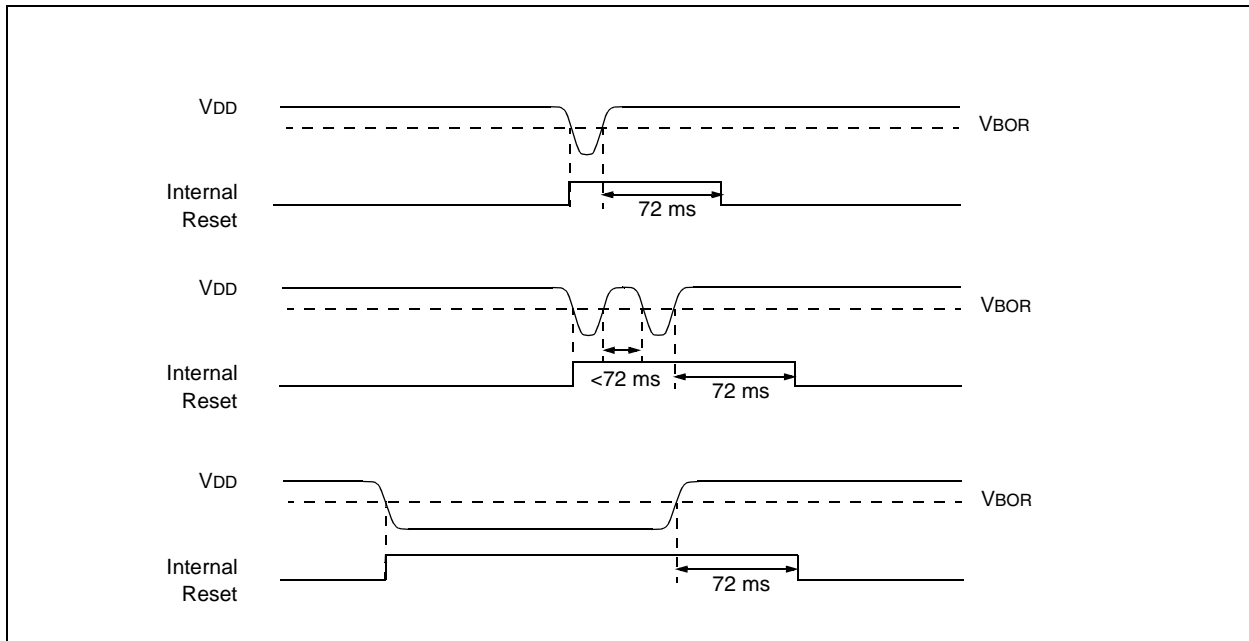
For operations where the desired brown-out voltage is other than 4V, an external brown-out circuit must be used. Figure 9-8, 9-9 and 9-10 show examples of external brown-out protection circuits.

# PIC16C712/716

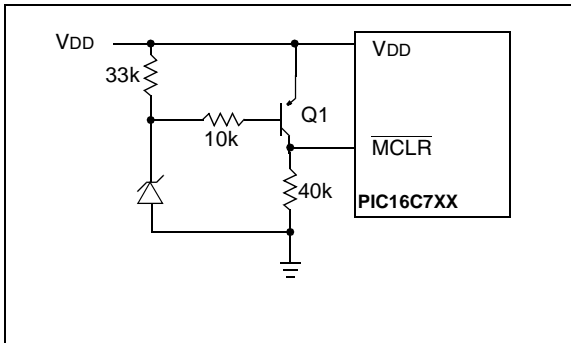
**FIGURE 9-6: SIMPLIFIED BLOCK DIAGRAM OF ON-CHIP RESET CIRCUIT**



**FIGURE 9-7: BROWN-OUT SITUATIONS**

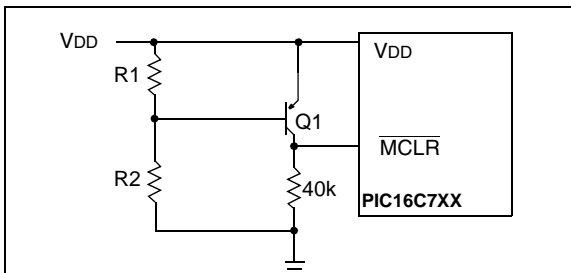


**FIGURE 9-8: EXTERNAL BROWN-OUT PROTECTION CIRCUIT 1**



- Note 1:** This circuit will activate Reset when VDD goes below (Vz + 0.7V) where Vz = Zener voltage.
- 2:** Internal Brown-out Reset circuitry should be disabled when using this circuit.

**FIGURE 9-9: EXTERNAL BROWN-OUT PROTECTION CIRCUIT 2**

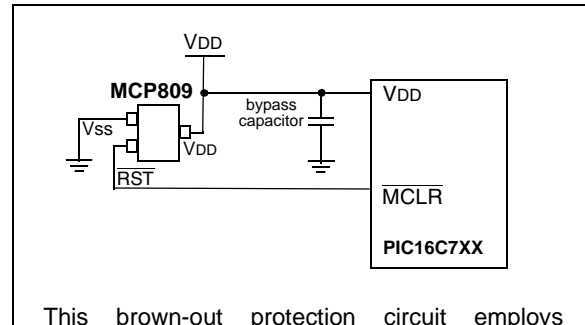


- Note 1:** This brown-out circuit is less expensive, albeit less accurate. Transistor Q1 turns off when VDD is below a certain level such that:

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

- 2:** Internal Brown-out Reset should be disabled when using this circuit.
- 3:** Resistors should be adjusted for the characteristics of the transistor.

**FIGURE 9-10: EXTERNAL BROWN-OUT PROTECTION CIRCUIT 3**



This brown-out protection circuit employs Microchip Technology's MCP809 microcontroller supervisor. The MCP8XX and MCP1XX families of supervisors provide push-pull and open collector outputs with both high and low active Reset pins. There are 7 different trip point selections to accommodate 5V and 3V systems

## 9.8 Time-out Sequence

On power-up the time-out sequence is as follows: First PWRT time-out is invoked after the POR time delay has expired. Then OST is activated. The total time-out will vary based on oscillator configuration and the status of the PWRT. For example, in RC mode with the PWRT disabled, there will be no time-out at all. Figure 9-11, Figure 9-12, and Figure 9-13 depict time-out sequences on power-up.

Since the time-outs occur from the POR pulse, if  $\overline{MCLR}$  is kept low long enough, the time-outs will expire. Then bringing  $\overline{MCLR}$  high will begin execution immediately (Figure 9-13). This is useful for testing purposes or to synchronize more than one PIC16CXXX device operating in parallel.

Table 9-5 shows the Reset conditions for some Special Function Registers, while Table 9-6 shows the Reset conditions for all the registers.

# PIC16C712/716

## 9.9 Power Control/Status Register (PCON)

The Power Control/Status Register, PCON has two bits.

Bit 0 is Brown-out Reset Status bit,  $\overline{\text{BOR}}$ . If the BODEN Configuration bit is set,  $\overline{\text{BOR}}$  is '1' on Power-on Reset. If the BODEN Configuration bit is clear,  $\overline{\text{BOR}}$  is unknown on Power-on Reset.

The  $\overline{\text{BOR}}$  Status bit is a "don't care" and is not necessarily predictable if the brown-out circuit is disabled (the BODEN Configuration bit is clear).  $\overline{\text{BOR}}$  must then be set by the user and checked on subsequent Resets to see if it is clear, indicating a brown-out has occurred.

Bit 1 is  $\overline{\text{POR}}$  (Power-on Reset Status bit). It is cleared on a Power-on Reset and unaffected otherwise. The user must set this bit following a Power-on Reset.

**TABLE 9-3: TIME-OUT IN VARIOUS SITUATIONS**

Oscillator Configuration	Power-up		Brown-out	Wake-up from Sleep
	$\overline{\text{PWRTE}} = 0$	$\overline{\text{PWRTE}} = 1$		
XT, HS, LP	72 ms + 1024TOSC	1024TOSC	72 ms + 1024TOSC	1024TOSC
RC	72 ms	—	72 ms	—

**TABLE 9-4: STATUS BITS AND THEIR SIGNIFICANCE**

$\overline{\text{POR}}$	$\overline{\text{BOR}}$	$\overline{\text{TO}}$	$\overline{\text{PD}}$	
0	x	1	1	Power-on Reset
0	x	0	x	Illegal, $\overline{\text{TO}}$ is set on $\overline{\text{POR}}$
0	x	x	0	Illegal, $\overline{\text{PD}}$ is set on $\overline{\text{POR}}$
1	0	1	1	Brown-out Reset
1	1	0	1	WDT Reset
1	1	0	0	WDT Wake-up
1	1	u	u	$\overline{\text{MCLR}}$ Reset during normal operation
1	1	1	0	$\overline{\text{MCLR}}$ Reset during Sleep or interrupt wake-up from Sleep

**TABLE 9-5: RESET CONDITION FOR SPECIAL REGISTERS**

Condition	Program Counter	STATUS Register	PCON Register
Power-on Reset	000h	0001 1xxx	---- --0x
$\overline{\text{MCLR}}$ Reset during normal operation	000h	000u uuuu	---- --uu
$\overline{\text{MCLR}}$ Reset during Sleep	000h	0001 0uuu	---- --uu
WDT Reset	000h	0000 1uuu	---- --uu
WDT Wake-up	PC + 1	uuu0 0uuu	---- --uu
Brown-out Reset	000h	0001 1uuu	---- --u0
Interrupt wake-up from Sleep	PC + 1 <sup>(1)</sup>	uuu1 0uuu	---- --uu

**Legend:** u = unchanged, x = unknown, – = unimplemented bit read as '0'.

**Note 1:** When the wake-up is due to an interrupt and the GIE bit is set, the PC is loaded with the interrupt vector (0004h).

**TABLE 9-6: INITIALIZATION CONDITIONS FOR ALL REGISTERS OF THE PIC16C712/716**

Register	Power-on Reset, Brown-out Reset	MCLR Resets WDT Reset	Wake-up via WDT or Interrupt
W	xxxx xxxx	uuuu uuuu	uuuu uuuu
INDF	N/A	N/A	N/A
TMR0	xxxx xxxx	uuuu uuuu	uuuu uuuu
PCL	0000h	0000h	PC + 1 <sup>(2)</sup>
STATUS	0001 1xxx	000q quuu <sup>(3)</sup>	uuuq quuu <sup>(3)</sup>
FSR	xxxx xxxx	uuuu uuuu	uuuu uuuu
PORTA <sup>(4)</sup>	--0x 0000	--xx xxxx	--xu uuuu
PORTB <sup>(5)</sup>	xxxx xxxx	uuuu uuuu	uuuu uuuu
DATAACP	---- -x-x	---- -u-u	---- -u-u
PCLATH	---0 0000	---0 0000	---u uuuu
INTCON	0000 -00x	0000 -00u	uuuu -uuu <sup>(1)</sup>
PIR1	---- 0000	---- 0000	---- uuuu <sup>(1)</sup>
	-0-- 0000	-0-- 0000	-u-- uuuu <sup>(1)</sup>
TMR1L	xxxx xxxx	uuuu uuuu	uuuu uuuu
TMR1H	xxxx xxxx	uuuu uuuu	uuuu uuuu
T1CON	--00 0000	--uu uuuu	--uu uuuu
TMR2	0000 0000	0000 0000	uuuu uuuu
T2CON	-000 0000	-000 0000	-uuu uuuu
CCPR1L	xxxx xxxx	uuuu uuuu	uuuu uuuu
CCPR1H	xxxx xxxx	uuuu uuuu	uuuu uuuu
CCP1CON	--00 0000	--00 0000	--uu uuuu
ADRES	xxxx xxxx	uuuu uuuu	uuuu uuuu
ADCON0	0000 00-0	0000 00-0	uuuu uu-u
OPTION_REG	1111 1111	1111 1111	uuuu uuuu
TRISA	--11 1111	--11 1111	--uu uuuu
TRISB	1111 1111	1111 1111	uuuu uuuu
TRISCCP	xxxx x1x1	xxxx x1x1	xxxx xuxu
PIE1	---- 0000	---- 0000	---- uuuu
	-0-- 0000	-0-- 0000	-u-- uuuu
PCON	---- --0q	---- --uq	---- --uq
PR2	1111 1111	1111 1111	1111 1111
ADCON1	---- -000	---- -000	---- -uuu

**Legend:** u = unchanged, x = unknown, - = unimplemented bit, read as '0', q = value depends on condition

**Note 1:** One or more bits in INTCON and/or PIR1 will be affected (to cause wake-up).

**2:** When the wake-up is due to an interrupt and the GIE bit is set, the PC is loaded with the interrupt vector (0004h).

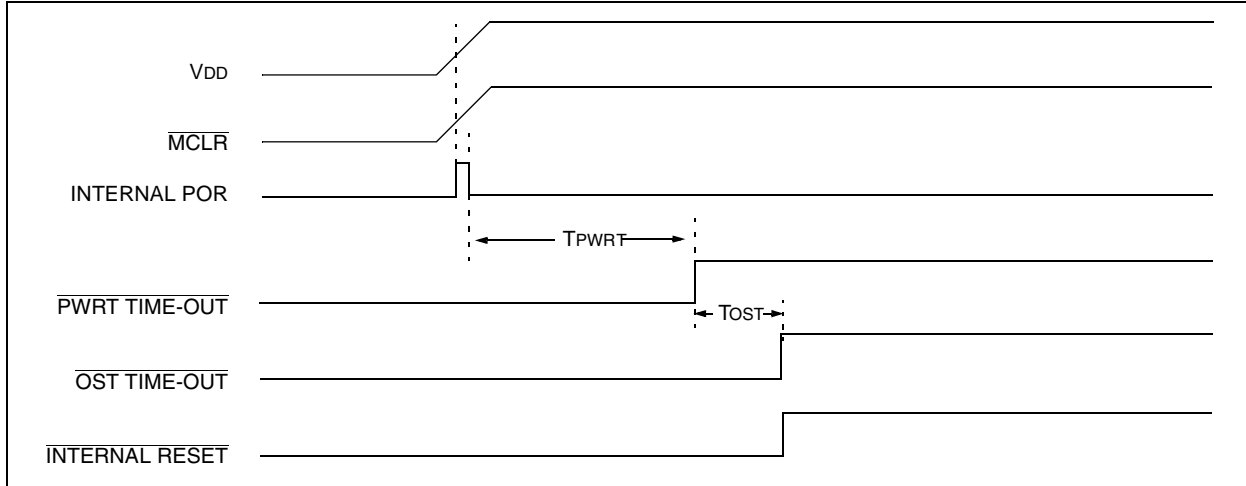
**3:** See Table 9-5 for Reset value for specific condition.

**4:** On any device Reset, these pins are configured as inputs.

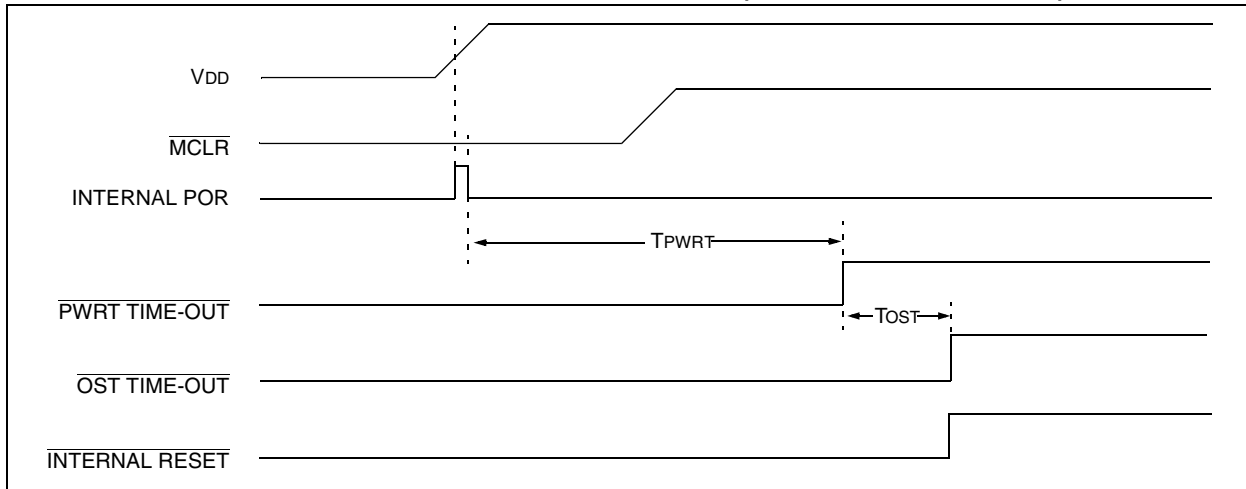
**5:** This is the value that will be in the port output latch.

# PIC16C712/716

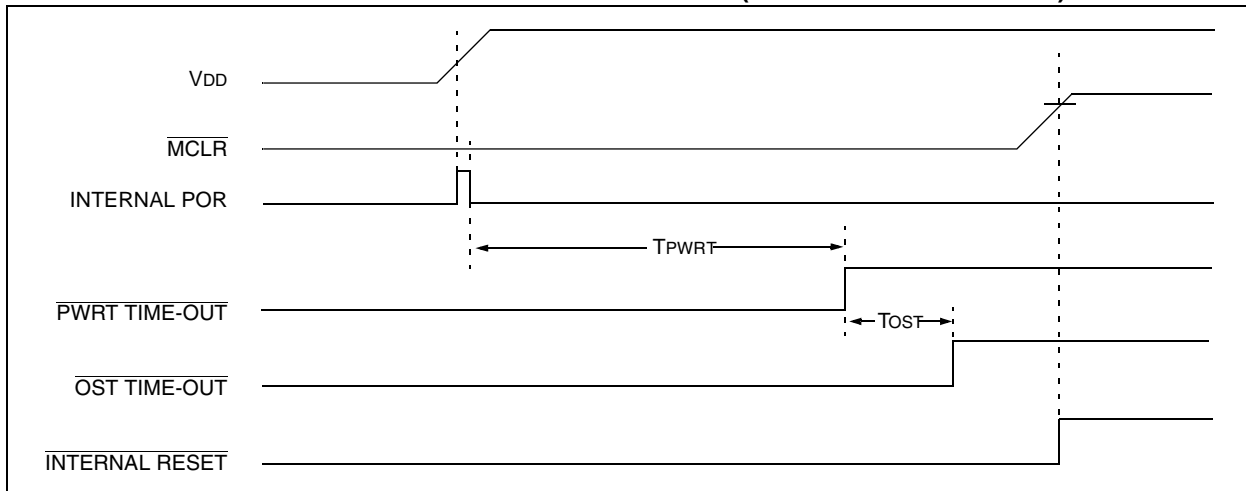
**FIGURE 9-11: TIME-OUT SEQUENCE ON POWER-UP ( $\overline{\text{MCLR}}$  TIED TO  $V_{DD}$ )**



**FIGURE 9-12: TIME-OUT SEQUENCE ON POWER-UP ( $\overline{\text{MCLR}}$  NOT TIED TO  $V_{DD}$ ): CASE 1**



**FIGURE 9-13: TIME-OUT SEQUENCE ON POWER-UP ( $\overline{\text{MCLR}}$  NOT TIED TO  $V_{DD}$ ): CASE 2**





## 9.10 Interrupts

The PIC16C712/716 devices have up to 7 sources of interrupt. The Interrupt Control Register (INTCON) records individual interrupt requests in flag bits. It also has individual and global interrupt enable bits.

**Note:** Individual interrupt flag bits are set regardless of the status of their corresponding mask bit or the GIE bit.

A Global Interrupt Enable bit, GIE (INTCON<7>) enables (if set) all unmasked interrupts or disables (if cleared) all interrupts. When bit GIE is enabled, and an interrupt's flag bit and mask bit are set, the interrupt will vector immediately. Individual interrupts can be disabled through their corresponding enable bits in various registers. Individual interrupt bits are set, regardless of the status of the GIE bit. The GIE bit is cleared on Reset.

The "return from interrupt" instruction, RETFIE, exits the interrupt routine, as well as sets the GIE bit, which re-enables interrupts.

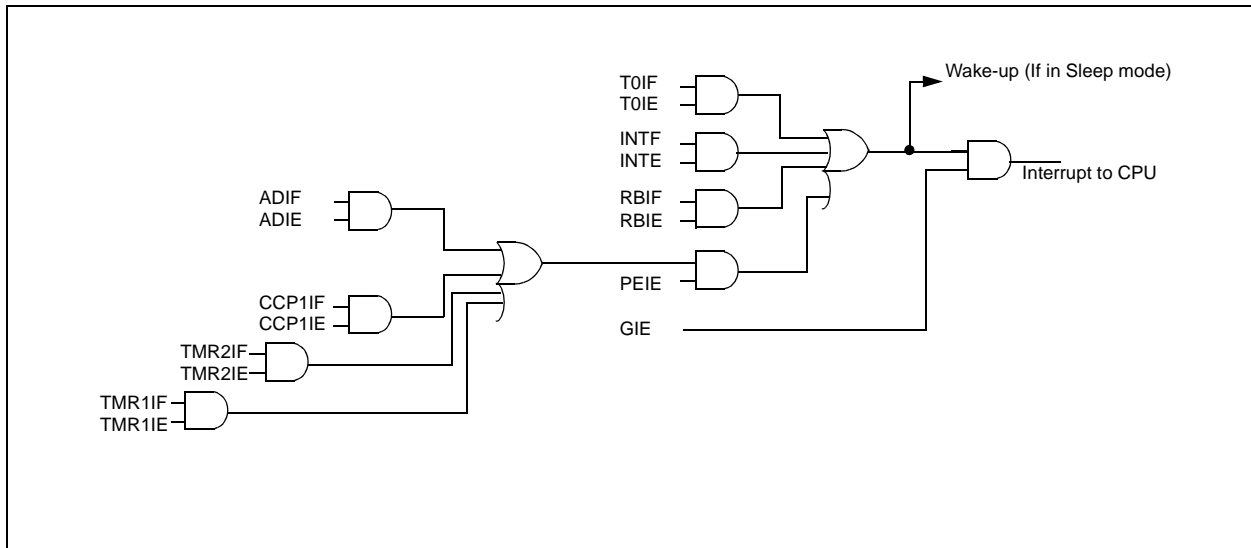
The RB0/INT pin interrupt, the RB port change interrupt and the TMR0 overflow interrupt flags are contained in the INTCON register.

The peripheral interrupt flags are contained in the Special Function Registers, PIR1 and PIR2. The corresponding interrupt enable bits are contained in Special Function Registers, PIE1 and PIE2, and the peripheral interrupt enable bit is contained in Special Function Register, INTCON.

When an interrupt is responded to, the GIE bit is cleared to disable any further interrupt, the return address is pushed onto the stack and the PC is loaded with 0004h. Once in the Interrupt Service Routine, the source(s) of the interrupt can be determined by polling the interrupt flag bits. The interrupt flag bit(s) must be cleared in software before re-enabling interrupts to avoid recursive interrupts.

For external interrupt events, such as the INT pin or PORTB change interrupt, the interrupt latency will be three or four instruction cycles. The exact latency depends when the interrupt event occurs. The latency is the same for one or two cycle instructions. Individual interrupt flag bits are set, regardless of the status of their corresponding mask bit or the GIE bit.

**FIGURE 9-14: INTERRUPT LOGIC**



# PIC16C712/716

## 9.10.1 INT INTERRUPT

External interrupt on RB0/INT pin is edge triggered, either rising if bit INTEDG (OPTION\_REG<6>) is set, or falling if the INTEDG bit is clear. When a valid edge appears on the RB0/INT pin, flag bit INTF (INTCON<1>) is set. This interrupt can be disabled by clearing enable bit INTE (INTCON<4>). Flag bit INTF must be cleared in software in the Interrupt Service Routine before re-enabling this interrupt. The INT interrupt can wake-up the processor from Sleep, if bit INTE was set prior to going into Sleep. The status of global interrupt enable bit GIE decides whether or not the processor branches to the interrupt vector following wake-up. See **Section 9.13 “Power-down Mode (Sleep)”** for details on Sleep mode.

## 9.10.2 TMR0 INTERRUPT

An overflow (FFh → 00h) in the TMR0 register will set flag bit T0IF (INTCON<2>). The interrupt can be enabled/disabled by setting/clearing enable bit T0IE (INTCON<5>). (**Section 4.0 “Timer0 Module”**)

## 9.10.3 PORTB INTCON CHANGE

An input change on PORTB<7:4> sets flag bit RBIF (INTCON<0>). The interrupt can be enabled/disabled by setting/clearing enable bit RBIE (INTCON<4>). (**Section 3.2 “PORTB and the TRISB Register”**)

## 9.11 Context Saving During Interrupts

During an interrupt, only the return PC value is saved on the stack. Typically, users may wish to save key registers during an interrupt, (i.e., W register and STATUS register). This will have to be implemented in software.

Example 9-1 stores and restores the W and STATUS registers. The register, W\_TEMP, must be defined in each bank and must be defined at the same offset from the bank base address (i.e., if W\_TEMP is defined at 0x20 in bank 0, it must also be defined at 0xA0 in bank 1).

The example:

- a) Stores the W register.
- b) Stores the STATUS register in bank 0.
- c) Stores the PCLATH register.
- d) Executes the Interrupt Service Routine code (User-generated).
- e) Restores the STATUS register (and bank select bit).
- f) Restores the W and PCLATH registers.

### EXAMPLE 9-1: SAVING STATUS, W, AND PCLATH REGISTERS IN RAM

```
MOVWF    W_TEMP           ;Copy W to TEMP register, could be bank one or zero
SWAPF    STATUS,W         ;Swap status to be saved into W
CLRF     STATUS           ;bank 0, regardless of current bank, Clears IRP,RP1,RP0
MOVWF    STATUS_TEMP      ;Save status to bank zero STATUS_TEMP register
MOVF     PCLATH, W        ;Only required if using pages 1, 2 and/or 3
MOVWF    PCLATH_TEMP     ;Save PCLATH into W
CLRF     PCLATH           ;Page zero, regardless of current page
BCF      STATUS, IRP      ;Return to Bank 0
MOVF     FSR, W           ;Copy FSR to W
MOVWF    FSR_TEMP        ;Copy FSR from W to FSR_TEMP
:
:(ISR)
:
MOVF     PCLATH_TEMP, W   ;Restore PCLATH
MOVWF    PCLATH           ;Move W into PCLATH
SWAPF    STATUS_TEMP,W    ;Swap STATUS_TEMP register into W
; (sets bank to original state)
MOVWF    STATUS           ;Move W into STATUS register
SWAPF    W_TEMP,F        ;Swap W_TEMP
SWAPF    W_TEMP,W        ;Swap W_TEMP into W
```

## 9.12 Watchdog Timer (WDT)

The Watchdog Timer is as a free running, on-chip, RC oscillator which does not require any external components. This RC oscillator is separate from the RC oscillator of the OSC1/CLKIN pin. That means that the WDT will run, even if the clock on the OSC1/CLKIN and OSC2/CLKOUT pins of the device have been stopped, for example, by execution of a `SLEEP` instruction.

During normal operation, a WDT Time-out generates a device Reset (Watchdog Timer Reset). If the device is in Sleep mode, a WDT Time-out causes the device to wake-up and continue with normal operation (Watchdog Timer Wake-up). The  $\overline{TO}$  bit in the STATUS register will be cleared upon a Watchdog Timer Time-out.

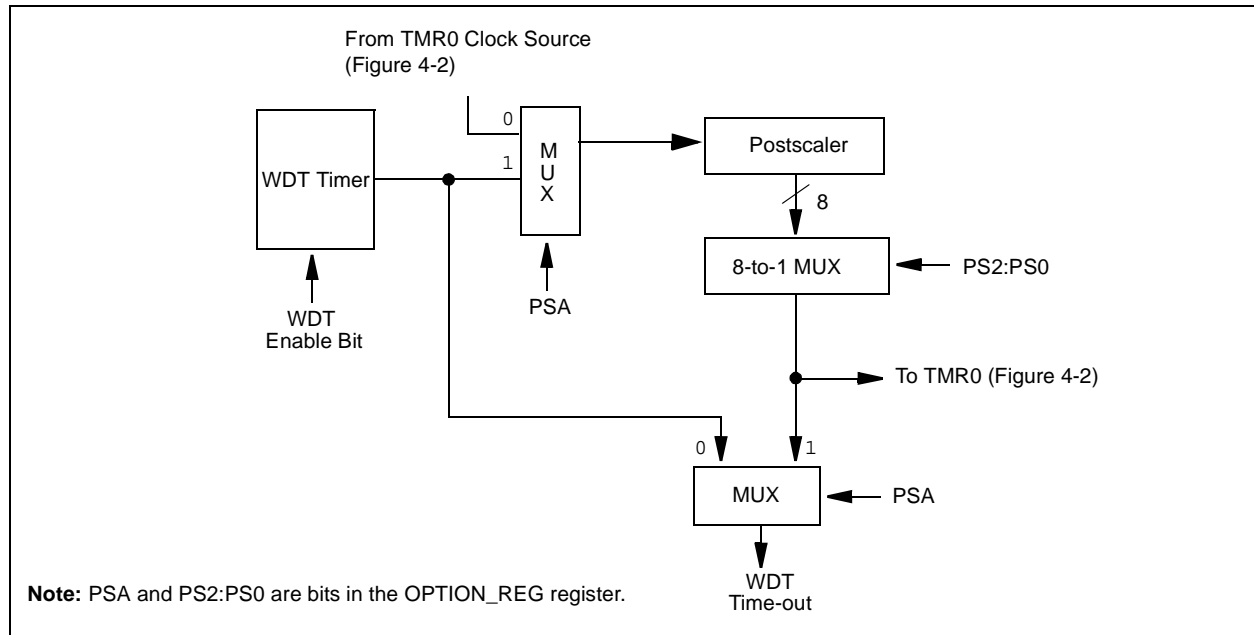
The WDT can be permanently disabled by clearing Configuration bit `WDTE` (Section 9.1 “Configuration Bits”).

WDT time-out period values may be found in the Electrical Specifications section under `TWDT` (parameter #31). Values for the WDT prescaler (actually a postscaler, but shared with the Timer0 prescaler) may be assigned using the `OPTION_REG` register.

**Note:** The `CLRWDT` and `SLEEP` instructions clear the WDT and the postscaler, if assigned to the WDT, and prevent it from timing out and generating a device Reset condition.

**Note:** When a `CLRWDT` instruction is executed and the prescaler is assigned to the WDT, the prescaler count will be cleared, but the prescaler assignment is not changed.

**FIGURE 9-15: WATCHDOG TIMER BLOCK DIAGRAM**



**FIGURE 9-16: SUMMARY OF WATCHDOG TIMER REGISTERS**

Address	Name	Bits 13:8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
2007h	Config. bits	(1)	—	BODEN <sup>(1)</sup>	$\overline{CP1}$	$\overline{CP0}$	PWRTE <sup>(1)</sup>	WDTE	FOSC1	FOSC0
81h	OPTION_REG	N/A	$\overline{RBPU}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0

**Legend:** Shaded cells are not used by the Watchdog Timer.

**Note 1:** See Figure 9-1 for operation of these bits.

## 9.13 Power-down Mode (Sleep)

Power-Down mode is entered by executing a `SLEEP` instruction.

If enabled, the Watchdog Timer will be cleared but keeps running, the `PD` bit (`STATUS<3>`) is cleared, the `TO` (`STATUS<4>`) bit is set, and the oscillator driver is turned off. The I/O ports maintain the status they had, before the `SLEEP` instruction was executed (driving high, low, or high-impedance).

For lowest current consumption in this mode, place all I/O pins at either `VDD` or `VSS`, ensure no external circuitry is drawing current from the I/O pin, power-down the `A/D` and the disable external clocks. Pull all I/O pins, that are high-impedance inputs, high or low externally to avoid switching currents caused by floating inputs. The `TOCKI` input should also be at `VDD` or `VSS` for lowest current consumption. The contribution from on-chip pull-ups on `PORTB` should be considered.

The `MCLR` pin must be at a logic high level (`VIHMC`).

### 9.13.1 WAKE-UP FROM SLEEP

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

1. External Reset input on `MCLR` pin.
2. Watchdog Timer Wake-up (if `WDT` was enabled).
3. Interrupt from `INT` pin, `RB` port change, or some peripheral interrupts.

External `MCLR` Reset will cause a device Reset. All other events are considered a continuation of program execution and cause a “wake-up”. The `TO` and `PD` bits in the `STATUS` register can be used to determine the cause of device Reset. The `PD` bit, which is set on power-up, is cleared when `SLEEP` is invoked. The `TO` bit is cleared if a `WDT` Time-out occurred (and caused wake-up).

The following peripheral interrupts can wake the device from Sleep:

1. `TMR1` interrupt. `Timer1` must be operating as an asynchronous counter.
2. `CCP` Capture mode interrupt.
3. Special Event Trigger (`Timer1` in Asynchronous mode using an external clock).

Other peripherals cannot generate interrupts, since during Sleep, no on-chip clocks are present.

When the `SLEEP` instruction is being executed, the next instruction (`PC + 1`) is pre-fetched. For the device to wake-up through an interrupt event, the corresponding interrupt enable bit must be set (enabled). Wake-up is regardless of the state of the `GIE` bit. If the `GIE` bit is clear (disabled), the device continues execution at the instruction after the `SLEEP` instruction. If the `GIE` bit is set (enabled), the device executes the instruction after the `SLEEP` instruction and then branches to the interrupt address (`0004h`). In cases where the execution of the instruction following `SLEEP` is not desirable, the user should have a `NOP` after the `SLEEP` instruction.

## 9.13.2 WAKE-UP USING INTERRUPTS

When global interrupts are disabled (`GIE` cleared) and any interrupt source has both its interrupt enable bit and interrupt flag bit set, one of the following will occur:

- If the interrupt occurs **before** the execution of a `SLEEP` instruction, the `SLEEP` instruction will complete as a `NOP`. Therefore, the `WDT` and `WDT`

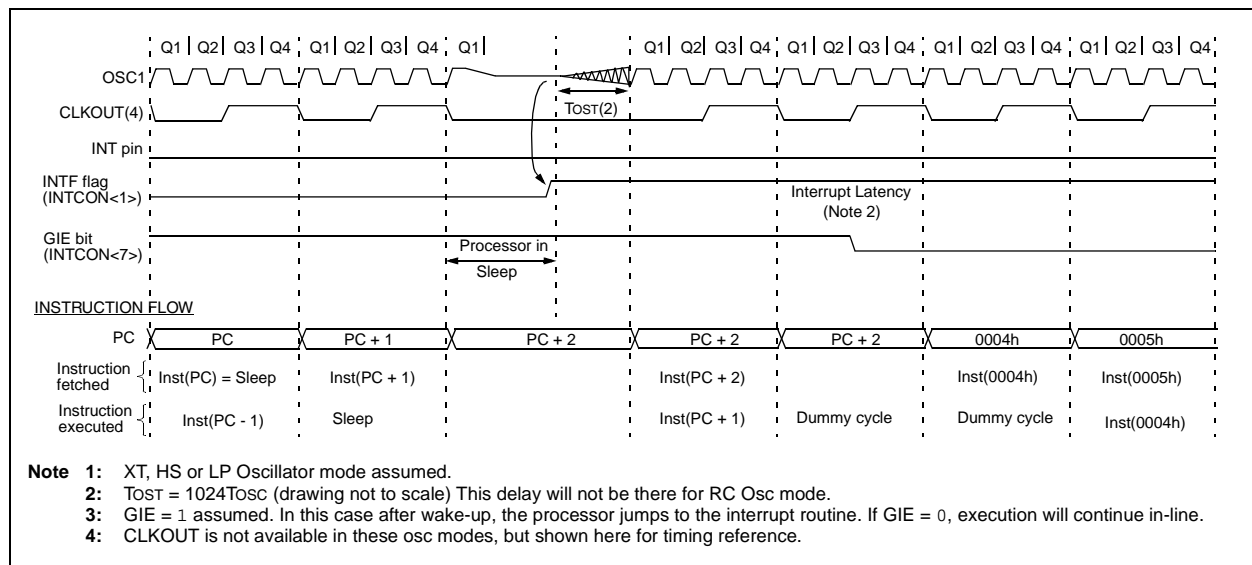
postscaler will not be cleared, the  $\overline{TO}$  bit will not be set and  $\overline{PD}$  bits will not be cleared.

- If the interrupt occurs **during or after** the execution of a `SLEEP` instruction, the device will immediately wake-up from Sleep. The `SLEEP` instruction will be completely executed before the wake-up. Therefore, the `WDT` and `WDT` postscaler will be cleared, the  $\overline{TO}$  bit will be set and the  $\overline{PD}$  bit will be cleared.

Even if the flag bits were checked before executing a `SLEEP` instruction, it may be possible for flag bits to become set before the `SLEEP` instruction completes. To determine whether a `SLEEP` instruction executed, test the  $\overline{PD}$  bit. If the  $\overline{PD}$  bit is set, the `SLEEP` instruction was executed as a `NOP`.

To ensure that the `WDT` is cleared, a `CLRWDT` instruction should be executed before a `SLEEP` instruction.

**FIGURE 9-17: WAKE-UP FROM SLEEP THROUGH INTERRUPT**



## 9.14 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.

## 9.15 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.

For ROM devices, these values are submitted along with the ROM code.

# PIC16C712/716

---

## 9.16 In-Circuit Serial Programming™

PIC16CXXX 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.

For complete details on serial programming, please refer to the In-Circuit Serial Programming™ (ICSP™) Guide, (DS30277).

## 10.0 INSTRUCTION SET SUMMARY

Each PIC16CXXX instruction is a 14-bit word divided into an OPCODE which specifies the instruction type and one or more operands which further specify the operation of the instruction. The PIC16CXXX instruction set summary in Table 10-2 lists **byte-oriented**, **bit-oriented**, and **literal and control** operations. Table 10-1 shows the opcode field descriptions.

For **byte-oriented** instructions, 'f' represents a file register designator and 'd' represents a destination designator. The file register designator specifies which file register is to be used by the instruction.

The destination designator specifies where the result of the operation is to be placed. If 'd' is zero, the result is placed in the W register. If 'd' is one, the result is placed in the file register specified in the instruction.

For **bit-oriented** instructions, 'b' represents a bit field designator which selects the number of the bit affected by the operation, while 'f' represents the number of the file in which the bit is located.

For **literal and control** operations, 'k' represents an eight or eleven bit constant or literal value.

**TABLE 10-1: OPCODE FIELD DESCRIPTIONS**

Field	Description
f	Register file address (0x00 to 0x7F)
W	Working register (accumulator)
b	Bit address within an 8-bit file register
k	Literal field, constant data or label
x	Don't care location (= 0 or 1) The assembler will generate code with x = 0. It is the recommended form of use for compatibility with all Microchip software tools.
d	Destination select; d = 0: store result in W, d = 1: store result in file register f. Default is d = 1
PC	Program Counter
TO	Time-out bit
PD	Power-down bit
Z	Zero bit
DC	Digit Carry bit
C	Carry bit

The instruction set is highly orthogonal and is grouped into three basic categories:

- **Byte-oriented** operations
- **Bit-oriented** operations
- **Literal and control** operations

All instructions are executed within one single instruction cycle, unless a conditional test is true or the program counter is changed as a result of an instruction. In this case, the execution takes two instruction cycles with the second cycle executed as a NOP. One instruction cycle consists of four oscillator periods. Thus, for an oscillator frequency of 4 MHz, the normal instruction execution time is 1  $\mu$ s. If a conditional test is true or the program counter is changed as a result of an instruction, the instruction execution time is 2  $\mu$ s.

Table 10-2 lists the instructions recognized by the MPASM assembler.

Figure 10-1 shows the general formats that the instructions can have.

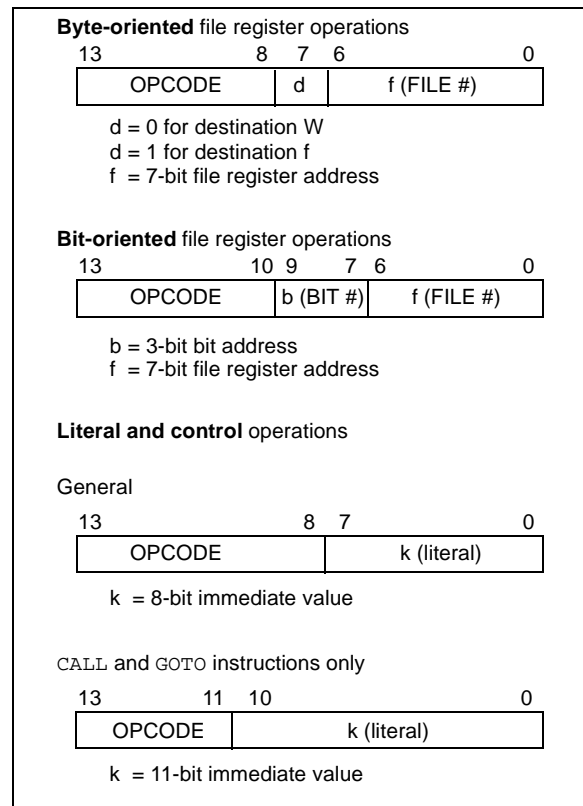
**Note:** To maintain upward compatibility with future PIC16CXXX products, do not use the OPTION and TRIS instructions.

All examples use the following format to represent a hexadecimal number:

0xhh

where h signifies a hexadecimal digit.

**FIGURE 10-1: GENERAL FORMAT FOR INSTRUCTIONS**



A description of each instruction is available in the PIC<sup>®</sup> Mid-Range Reference Manual, (DS33023).

# PIC16C712/716

**TABLE 10-2: PIC16CXXX INSTRUCTION SET**

Mnemonic, Operands	Description	Cycles	14-Bit Opcode				Status Affected	Notes	
			MSb	LSb					
<b>BYTE-ORIENTED FILE REGISTER OPERATIONS</b>									
<b>ADDWF</b>	f, d	Add W and f	1	00	0111	dfff	ffff	C,DC,Z	1,2
<b>ANDWF</b>	f, d	AND W with f	1	00	0101	dfff	ffff	Z	1,2
<b>CLRF</b>	f	Clear f	1	00	0001	1fff	ffff	Z	2
<b>CLRWF</b>	-	Clear W	1	00	0001	0000	0011	Z	
<b>COMF</b>	f, d	Complement f	1	00	1001	dfff	ffff	Z	1,2
<b>DECf</b>	f, d	Decrement f	1	00	0011	dfff	ffff	Z	1,2
<b>DECFSZ</b>	f, d	Decrement f, Skip if 0	1(2)	00	1011	dfff	ffff		1,2,3
<b>INCF</b>	f, d	Increment f	1	00	1010	dfff	ffff	Z	1,2
<b>INCFSZ</b>	f, d	Increment f, Skip if 0	1(2)	00	1111	dfff	ffff		1,2,3
<b>IORWF</b>	f, d	Inclusive OR W with f	1	00	0100	dfff	ffff	Z	1,2
<b>MOVF</b>	f, d	Move f	1	00	1000	dfff	ffff	Z	1,2
<b>MOVWF</b>	f	Move W to f	1	00	0000	1fff	ffff		
<b>NOP</b>	-	No Operation	1	00	0000	0xx0	0000		
<b>RLF</b>	f, d	Rotate Left f through Carry	1	00	1101	dfff	ffff	C	1,2
<b>RRF</b>	f, d	Rotate Right f through Carry	1	00	1100	dfff	ffff	C	1,2
<b>SUBWF</b>	f, d	Subtract W from f	1	00	0010	dfff	ffff	C,DC,Z	1,2
<b>SWAPF</b>	f, d	Swap nibbles in f	1	00	1110	dfff	ffff		1,2
<b>XORWF</b>	f, d	Exclusive OR W with f	1	00	0110	dfff	ffff	Z	1,2
<b>BIT-ORIENTED FILE REGISTER OPERATIONS</b>									
<b>BCF</b>	f, b	Bit Clear f	1	01	00bb	bfff	ffff		1,2
<b>BSF</b>	f, b	Bit Set f	1	01	01bb	bfff	ffff		1,2
<b>BTFSC</b>	f, b	Bit Test f, Skip if Clear	1 (2)	01	10bb	bfff	ffff		3
<b>BTFSS</b>	f, b	Bit Test f, Skip if Set	1 (2)	01	11bb	bfff	ffff		3
<b>LITERAL AND CONTROL OPERATIONS</b>									
<b>ADDLW</b>	k	Add literal and W	1	11	111x	kkkk	kkkk	C,DC,Z	
<b>ANDLW</b>	k	AND literal with W	1	11	1001	kkkk	kkkk	Z	
<b>CALL</b>	k	Call subroutine	2	10	0kkk	kkkk	kkkk		
<b>CLRWD<sub>T</sub></b>	-	Clear Watchdog Timer	1	00	0000	0110	0100	$\overline{TO}, \overline{PD}$	
<b>GOTO</b>	k	Go to address	2	10	1kkk	kkkk	kkkk		
<b>IORLW</b>	k	Inclusive OR literal with W	1	11	1000	kkkk	kkkk	Z	
<b>MOVLW</b>	k	Move literal to W	1	11	00xx	kkkk	kkkk		
<b>RETFIE</b>	-	Return from interrupt	2	00	0000	0000	1001		
<b>RETLW</b>	k	Return with literal in W	2	11	01xx	kkkk	kkkk		
<b>RETURN</b>	-	Return from Subroutine	2	00	0000	0000	1000		
<b>SLEEP</b>	-	Go into standby mode	1	00	0000	0110	0011	$\overline{TO}, \overline{PD}$	
<b>SUBLW</b>	k	Subtract W from literal	1	11	110x	kkkk	kkkk	C,DC,Z	
<b>XORLW</b>	k	Exclusive OR literal with W	1	11	1010	kkkk	kkkk	Z	

**Note 1:** When an I/O register is modified as a function of itself (e.g., `MOVF PORTB, 1`), the value used will be that value present on the pins themselves. For example, if the data latch is '1' for a pin configured as input and is driven low by an external device, the data will be written back with a '0'.

**2:** If this instruction is executed on the TMR0 register (and, where applicable, `d = 1`), the prescaler will be cleared if assigned to the Timer0 Module.

**3:** If Program Counter (PC) is modified or a conditional test is true, the instruction requires two cycles. The second cycle is executed as a `NOP`.



## 11.0 DEVELOPMENT SUPPORT

The PIC<sup>®</sup> microcontrollers are supported with a full range of hardware and software development tools:

- Integrated Development Environment
  - MPLAB<sup>®</sup> IDE Software
- Assemblers/Compilers/Linkers
  - MPASM<sup>™</sup> Assembler
  - MPLAB C18 and MPLAB C30 C Compilers
  - MPLINK<sup>™</sup> Object Linker/  
MPLIB<sup>™</sup> Object Librarian
  - MPLAB ASM30 Assembler/Linker/Library
- Simulators
  - MPLAB SIM Software Simulator
- Emulators
  - MPLAB ICE 2000 In-Circuit Emulator
  - MPLAB ICE 4000 In-Circuit Emulator
- In-Circuit Debugger
  - MPLAB ICD 2
- Device Programmers
  - PICSTART<sup>®</sup> Plus Development Programmer
  - MPLAB PM3 Device Programmer
- Low-Cost Demonstration and Development Boards and Evaluation Kits

## 11.1 MPLAB Integrated Development Environment Software

The MPLAB IDE software brings an ease of software development previously unseen in the 8/16-bit microcontroller market. The MPLAB IDE is a Windows<sup>®</sup> operating system-based application that contains:

- A single graphical interface to all debugging tools
  - Simulator
  - Programmer (sold separately)
  - Emulator (sold separately)
  - In-Circuit Debugger (sold separately)
- A full-featured editor with color-coded context
- A multiple project manager
- Customizable data windows with direct edit of contents
- High-level source code debugging
- Visual device initializer for easy register initialization
- Mouse over variable inspection
- Drag and drop variables from source to watch windows
- Extensive on-line help
- Integration of select third party tools, such as HI-TECH Software C Compilers and IAR C Compilers

The MPLAB IDE allows you to:

- Edit your source files (either assembly or C)
- One touch assemble (or compile) and download to PIC MCU emulator and simulator tools (automatically updates all project information)
- Debug using:
  - Source files (assembly or C)
  - Mixed assembly and C
  - Machine code

MPLAB IDE supports multiple debugging tools in a single development paradigm, from the cost-effective simulators, through low-cost in-circuit debuggers, to full-featured emulators. This eliminates the learning curve when upgrading to tools with increased flexibility and power.

## 11.2 MPASM Assembler

The MPASM Assembler is a full-featured, universal macro assembler for all PIC MCUs.

The MPASM Assembler generates relocatable object files for the MPLINK Object Linker, Intel® standard HEX files, MAP files to detail memory usage and symbol reference, absolute LST files that contain source lines and generated machine code and COFF files for debugging.

The MPASM Assembler features include:

- Integration into MPLAB IDE projects
- User-defined macros to streamline assembly code
- Conditional assembly for multi-purpose source files
- Directives that allow complete control over the assembly process

## 11.3 MPLAB C18 and MPLAB C30 C Compilers

The MPLAB C18 and MPLAB C30 Code Development Systems are complete ANSI C compilers for Microchip's PIC18 family of microcontrollers and dsPIC30F family of digital signal controllers. These compilers provide powerful integration capabilities, superior code optimization and ease of use not found with other compilers.

For easy source level debugging, the compilers provide symbol information that is optimized to the MPLAB IDE debugger.

## 11.4 MPLINK Object Linker/ MPLIB Object Librarian

The MPLINK Object Linker combines relocatable objects created by the MPASM Assembler and the MPLAB C18 C Compiler. It can link relocatable objects from precompiled libraries, using directives from a linker script.

The MPLIB Object Librarian manages the creation and modification of library files of precompiled code. When a routine from a library is called from a source file, only the modules that contain that routine will be linked in with the application. This allows large libraries to be used efficiently in many different applications.

The object linker/library features include:

- Efficient linking of single libraries instead of many smaller files
- Enhanced code maintainability by grouping related modules together
- Flexible creation of libraries with easy module listing, replacement, deletion and extraction

## 11.5 MPLAB ASM30 Assembler, Linker and Librarian

MPLAB ASM30 Assembler produces relocatable machine code from symbolic assembly language for dsPIC30F devices. MPLAB C30 C Compiler uses the assembler to produce its object file. The assembler generates relocatable object files that can then be archived or linked with other relocatable object files and archives to create an executable file. Notable features of the assembler include:

- Support for the entire dsPIC30F instruction set
- Support for fixed-point and floating-point data
- Command line interface
- Rich directive set
- Flexible macro language
- MPLAB IDE compatibility

## 11.6 MPLAB SIM Software Simulator

The MPLAB SIM Software Simulator allows code development in a PC-hosted environment by simulating the PIC MCUs and dsPIC® DSCs on an instruction level. On any given instruction, the data areas can be examined or modified and stimuli can be applied from a comprehensive stimulus controller. Registers can be logged to files for further run-time analysis. The trace buffer and logic analyzer display extend the power of the simulator to record and track program execution, actions on I/O, as well as internal registers.

The MPLAB SIM Software Simulator fully supports symbolic debugging using the MPLAB C18 and MPLAB C30 C Compilers, and the MPASM and MPLAB ASM30 Assemblers. The software simulator offers the flexibility to develop and debug code outside of the laboratory environment, making it an excellent, economical software development tool.

## 11.7 MPLAB ICE 2000 High-Performance In-Circuit Emulator

The MPLAB ICE 2000 In-Circuit Emulator is intended to provide the product development engineer with a complete microcontroller design tool set for PIC microcontrollers. Software control of the MPLAB ICE 2000 In-Circuit Emulator is advanced by the MPLAB Integrated Development Environment, which allows editing, building, downloading and source debugging from a single environment.

The MPLAB ICE 2000 is a full-featured emulator system with enhanced trace, trigger and data monitoring features. Interchangeable processor modules allow the system to be easily reconfigured for emulation of different processors. The architecture of the MPLAB ICE 2000 In-Circuit Emulator allows expansion to support new PIC microcontrollers.

The MPLAB ICE 2000 In-Circuit Emulator system has been designed as a real-time emulation system with advanced features that are typically found on more expensive development tools. The PC platform and Microsoft® Windows® 32-bit operating system were chosen to best make these features available in a simple, unified application.

## 11.8 MPLAB ICE 4000 High-Performance In-Circuit Emulator

The MPLAB ICE 4000 In-Circuit Emulator is intended to provide the product development engineer with a complete microcontroller design tool set for high-end PIC MCUs and dsPIC DSCs. Software control of the MPLAB ICE 4000 In-Circuit Emulator is provided by the MPLAB Integrated Development Environment, which allows editing, building, downloading and source debugging from a single environment.

The MPLAB ICE 4000 is a premium emulator system, providing the features of MPLAB ICE 2000, but with increased emulation memory and high-speed performance for dsPIC30F and PIC18XXXX devices. Its advanced emulator features include complex triggering and timing, and up to 2 Mb of emulation memory.

The MPLAB ICE 4000 In-Circuit Emulator system has been designed as a real-time emulation system with advanced features that are typically found on more expensive development tools. The PC platform and Microsoft Windows 32-bit operating system were chosen to best make these features available in a simple, unified application.

## 11.9 MPLAB ICD 2 In-Circuit Debugger

Microchip's In-Circuit Debugger, MPLAB ICD 2, is a powerful, low-cost, run-time development tool, connecting to the host PC via an RS-232 or high-speed USB interface. This tool is based on the Flash PIC MCUs and can be used to develop for these and other PIC MCUs and dsPIC DSCs. The MPLAB ICD 2 utilizes the in-circuit debugging capability built into the Flash devices. This feature, along with Microchip's In-Circuit Serial Programming™ (ICSP™) protocol, offers cost-effective, in-circuit Flash debugging from the graphical user interface of the MPLAB Integrated Development Environment. This enables a designer to develop and debug source code by setting breakpoints, single stepping and watching variables, and CPU status and peripheral registers. Running at full speed enables testing hardware and applications in real time. MPLAB ICD 2 also serves as a development programmer for selected PIC devices.

## 11.10 MPLAB PM3 Device Programmer

The MPLAB PM3 Device Programmer is a universal, CE compliant device programmer with programmable voltage verification at VDDMIN and VDDMAX for maximum reliability. It features a large LCD display (128 x 64) for menus and error messages and a modular, detachable socket assembly to support various package types. The ICSP™ cable assembly is included as a standard item. In Stand-Alone mode, the MPLAB PM3 Device Programmer can read, verify and program PIC devices without a PC connection. It can also set code protection in this mode. The MPLAB PM3 connects to the host PC via an RS-232 or USB cable. The MPLAB PM3 has high-speed communications and optimized algorithms for quick programming of large memory devices and incorporates an SD/MMC card for file storage and secure data applications.

## 11.11 PICSTART Plus Development Programmer

The PICSTART Plus Development Programmer is an easy-to-use, low-cost, prototype programmer. It connects to the PC via a COM (RS-232) port. MPLAB Integrated Development Environment software makes using the programmer simple and efficient. The PICSTART Plus Development Programmer supports most PIC devices in DIP packages up to 40 pins. Larger pin count devices, such as the PIC16C92X and PIC17C76X, may be supported with an adapter socket. The PICSTART Plus Development Programmer is CE compliant.

## 11.12 Demonstration, Development and Evaluation Boards

A wide variety of demonstration, development and evaluation boards for various PIC MCUs and dsPIC DSCs allows quick application development on fully functional systems. Most boards include prototyping areas for adding custom circuitry and provide application firmware and source code for examination and modification.

The boards support a variety of features, including LEDs, temperature sensors, switches, speakers, RS-232 interfaces, LCD displays, potentiometers and additional EEPROM memory.

The demonstration and development boards can be used in teaching environments, for prototyping custom circuits and for learning about various microcontroller applications.

In addition to the PICDEM™ and dsPICDEM™ demonstration/development board series of circuits, Microchip has a line of evaluation kits and demonstration software for analog filter design, KEELOQ® security ICs, CAN, IrDA®, PowerSmart® battery management, SEEVAL® evaluation system, Sigma-Delta ADC, flow rate sensing, plus many more.

Check the Microchip web page ([www.microchip.com](http://www.microchip.com)) and the latest *“Product Selector Guide”* (DS00148) for the complete list of demonstration, development and evaluation kits.

## 12.0 ELECTRICAL CHARACTERISTICS

### Absolute Maximum Ratings <sup>(†)</sup>

Ambient temperature under bias .....	-55°C to +125°C
Storage temperature .....	-65°C to +150°C
Voltage on any pin with respect to V <sub>SS</sub> (except V <sub>DD</sub> , $\overline{\text{MCLR}}$ , and RA4) .....	-0.3V to (V <sub>DD</sub> + 0.3V)
Voltage on V <sub>DD</sub> with respect to V <sub>SS</sub> .....	-0.3V to +7.5V
Voltage on $\overline{\text{MCLR}}$ with respect to V <sub>SS</sub> ( <b>Note 2</b> ) .....	0V to +13.25V
Voltage on RA4 with respect to V <sub>SS</sub> .....	0V to +8.5V
Total power dissipation ( <b>Note 1</b> ) (PDIP and SOIC) .....	1.0W
Total power dissipation ( <b>Note 1</b> ) (SSOP) .....	0.65W
Maximum current out of V <sub>SS</sub> pin .....	300 mA
Maximum current into V <sub>DD</sub> pin .....	250 mA
Input clamp current, I <sub>IK</sub> (V <sub>I</sub> < 0 or V <sub>I</sub> > V <sub>DD</sub> ) .....	±20 mA
Output clamp current, I <sub>OK</sub> (V <sub>O</sub> < 0 or V <sub>O</sub> > V <sub>DD</sub> ) .....	±20 mA
Maximum output current sunk by any I/O pin .....	25 mA
Maximum output current sourced by any I/O pin .....	25 mA
Maximum current sunk by PORTA and PORTB (combined) .....	200 mA
Maximum current sourced by PORTA and PORTB (combined) .....	200 mA

**Note 1:** Power dissipation is calculated as follows:  $P_{dis} = V_{DD} \times \{I_{DD} - \sum I_{OH}\} + \sum \{(V_{DD} - V_{OH}) \times I_{OH}\} + \sum (V_{OL} \times I_{OL})$

**2:** Voltage spikes below V<sub>SS</sub> at the  $\overline{\text{MCLR}}$ /V<sub>PP</sub> pin, inducing currents greater than 80 mA, may cause latch-up. Thus, a series resistor of 50-100Ω should be used when applying a “low” level to the  $\overline{\text{MCLR}}$ /V<sub>PP</sub> pin rather than pulling this pin directly to V<sub>SS</sub>.

† NOTICE: Stresses above those listed under “Absolute 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.

# PIC16C712/716

FIGURE 12-1: PIC16C712/716 VOLTAGE-FREQUENCY GRAPH,  $-40^{\circ}\text{C} \leq \text{TA} \leq +125^{\circ}\text{C}$

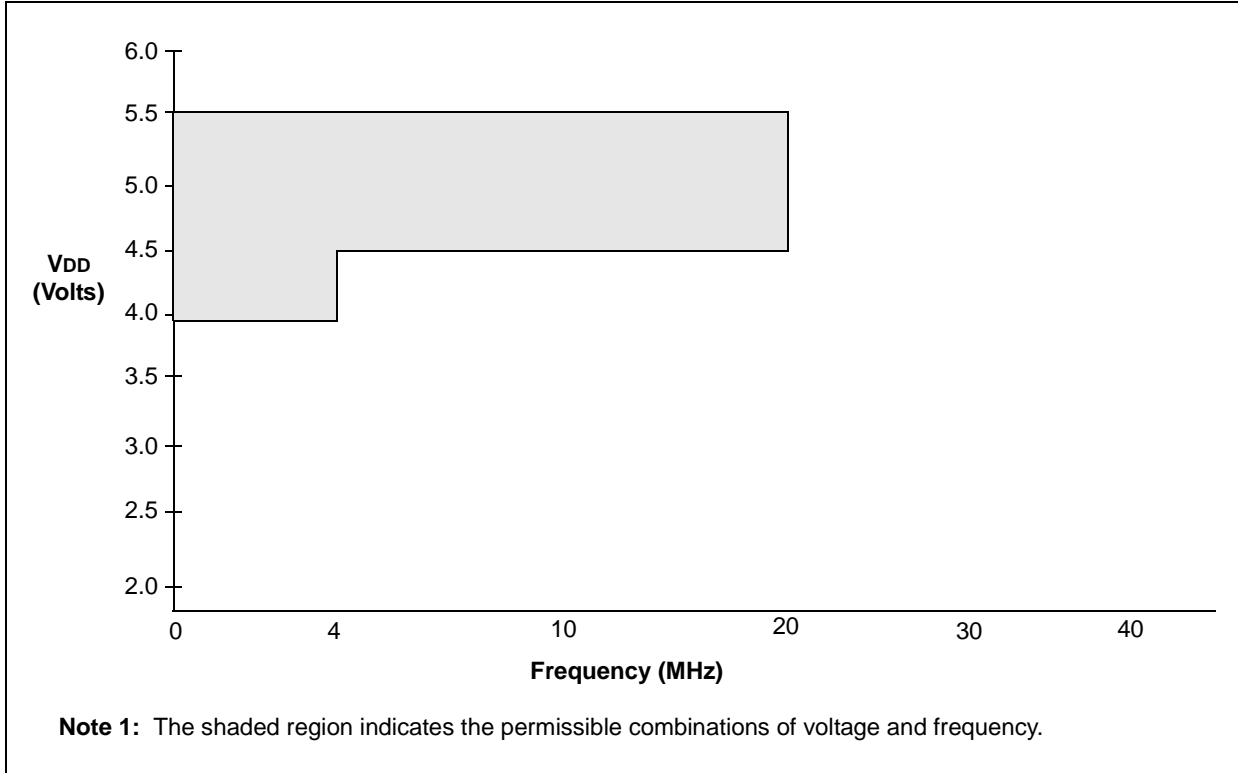
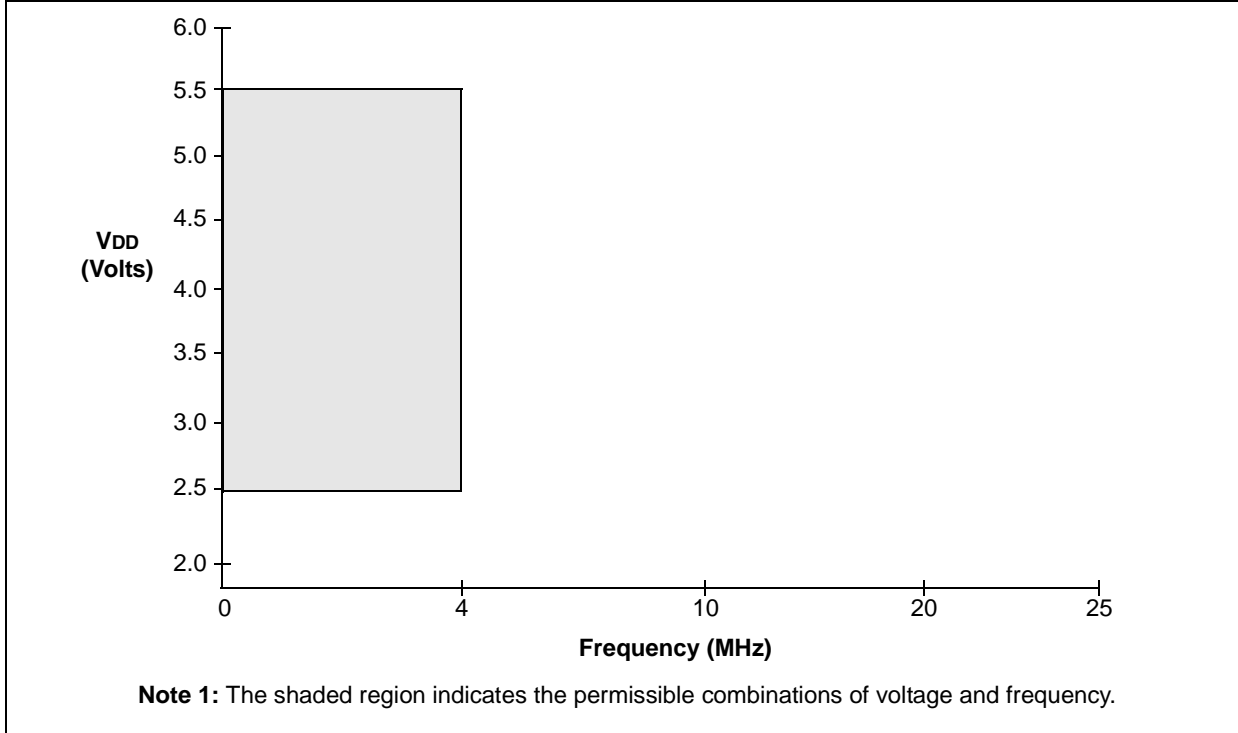


FIGURE 12-2: PIC16LC712/716 VOLTAGE-FREQUENCY GRAPH,  $0^{\circ}\text{C} \leq \text{TA} \leq +70^{\circ}\text{C}$



## 12.1 DC Characteristics: PIC16C712/716-04 (Commercial, Industrial, Extended) PIC16C712/716-20 (Commercial, Industrial, Extended)

Standard Operating Conditions (unless otherwise stated)							
DC CHARACTERISTICS							
Operating temperature							
0°C ≤ TA ≤ +70°C for commercial							
-40°C ≤ TA ≤ +85°C for industrial							
-40°C ≤ TA ≤ +125°C for extended							
Param No.	Sym.	Characteristic	Min.	Typ†	Max.	Units	Conditions
D001 D001A	VDD	Supply Voltage	4.0 4.5 VBOR*	— — —	5.5 5.5 5.5	V V V	XT, RC and LP osc mode HS osc mode BOR enabled <sup>(7)</sup>
D002*	VDR	RAM Data Retention Voltage <sup>(1)</sup>	—	1.5	—	V	
D003	VPOR	VDD Start Voltage to ensure internal Power-on Reset signal	—	VSS	—	V	See section on Power-on Reset for details
D004* D004A*	SVDD	VDD Rise Rate to ensure internal Power-on Reset signal	0.05 TBD	— —	— —	V/ms	PWRT enabled (PWRTÉ bit clear) PWRT disabled (PWRTÉ bit set) See section on Power-on Reset for details
D005	VBOR	Brown-out Reset voltage trip point	3.65	—	4.35	V	BODEN bit set
D010 D013	IDD	Supply Current <sup>(2,5)</sup>	— —	0.8 4.0	2.5 8.0	mA mA	FOSC = 4 MHz, VDD = 4.0V FOSC = 20 MHz, VDD = 4.0V
D020 D021 D021B	IPD	Power-down Current <sup>(3,5)</sup>	— — — —	10.5 1.5 1.5 2.5	42 16 19 19	μA μA μA μA	VDD = 4.0V, WDT enabled, -40°C to +85°C VDD = 4.0V, WDT disabled, 0°C to +70°C VDD = 4.0V, WDT disabled, -40°C to +85°C VDD = 4.0V, WDT disabled, -40°C to +125°C
D022* D022A*	ΔIWDT ΔIBOR	Module Differential Current <sup>(6)</sup> Watchdog Timer Brown-out Reset	— —	6.0 TBD	20 200	μA μA	WDTE bit set, VDD = 4.0V BODEN bit set, VDD = 5.0V
1A	FOSC	LP Oscillator Operating Frequency RC Oscillator Operating Frequency XT Oscillator Operating Frequency HS Oscillator Operating Frequency	0 0 0 0	— — — —	200 4 4 20	KHz MHz MHz MHz	All temperatures All temperatures All temperatures All temperatures

\* 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.

**Note1:** This is the limit to which VDD can be lowered 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 tri-stated, 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 high-impedance state and tied to VDD and VSS.

**4:** For RC Osc mode, current through REXT is not included. The current through the resistor can be estimated by the formula  $I_r = V_{DD}/2R_{EXT}$  (mA) with REXT in kOhm.

**5:** Timer1 oscillator (when enabled) adds approximately 20 μA to the specification. This value is from characterization and is for design guidance only. This is not tested.

**6:** 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.

**7:** This is the voltage where the device enters the Brown-out Reset. When BOR is enabled, the device will operate correctly to this trip point.

# PIC16C712/716

## 12.2 DC Characteristics: PIC16LC712/716-04 (Commercial, Industrial)

DC CHARACTERISTICS		Standard Operating Conditions (unless otherwise stated)					
		Operating temperature		0°C ≤ TA ≤ +70°C for commercial		-40°C ≤ TA ≤ +85°C for industrial	
Param No.	Sym.	Characteristic	Min.	Typ†	Max.	Units	Conditions
D001	VDD	<b>Supply Voltage</b>	2.5 VBOR*	—	5.5	V	BOR enabled ( <b>Note 7</b> )
D002*	VDR	<b>RAM Data Retention Voltage</b> <sup>(1)</sup>	—	1.5	—	V	
D003	VPOR	<b>VDD Start Voltage</b> to ensure internal Power-on Reset signal	—	VSS	—	V	See section on Power-on Reset for details
D004* D004A*	SVDD	<b>VDD Rise Rate</b> to ensure internal Power-on Reset signal	0.05 TBD	— —	— —	V/ms	PWRT enabled (PWRTÉ bit clear) PWRT disabled (PWRTÉ bit set) See section on Power-on Reset for details
D005	VBOR	<b>Brown-out Reset</b> voltage trip point	3.65	—	4.35	V	BODEN bit set
D010 D010A	IDD	<b>Supply Current</b> <sup>(2,5)</sup>	—	2.0	3.8	mA	XT, RC osc modes FOSC = 4 MHz, VDD = 3.0V ( <b>Note 4</b> )
			—	22.5	48	µA	LP osc mode FOSC = 32 kHz, VDD = 3.0V, WDT disabled
D020 D021 D021A	IPD	<b>Power-down Current</b> <sup>(3,5)</sup>	—	7.5	30	µA	VDD = 3.0V, WDT enabled, -40°C to +85°C
			—	0.9	5	µA	VDD = 3.0V, WDT disabled, 0°C to +70°C
			—	0.9	5	µA	VDD = 3.0V, WDT disabled, -40°C to +85°C
D022* D022A*	ΔIWDT ΔIBOR	<b>Module Differential Current</b> <sup>(6)</sup> Watchdog Timer Brown-out Reset	— —	6.0 TBD	20 200	µA µA	WDTE bit set, VDD = 4.0V BODEN bit set, VDD = 5.0V
1A	FOSC	LP Oscillator Operating Frequency RC Oscillator Operating Frequency XT Oscillator Operating Frequency HS Oscillator Operating Frequency	0 0 0 0	— — — —	200 4 4 20	KHz MHz MHz MHz	All temperatures All temperatures All temperatures All temperatures

\* 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.

**Note1:** This is the limit to which VDD can be lowered 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 tri-stated, 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 high-impedance state and tied to VDD and VSS.

**4:** For RC Osc mode, current through REXT is not included. The current through the resistor can be estimated by the formula  $I_r = V_{DD}/2R_{EXT}$  (mA) with REXT in kOhm.

**5:** Timer1 oscillator (when enabled) adds approximately 20 µA to the specification. This value is from characterization and is for design guidance only. This is not tested.

**6:** 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.

**7:** This is the voltage where the device enters the Brown-out Reset. When BOR is enabled, the device will operate correctly to this trip point.



## 12.3 DC Characteristics: PIC16C712/716-04 (Commercial, Industrial, Extended) PIC16C712/716-20 (Commercial, Industrial, Extended) PIC16LC712/716-04 (Commercial, Industrial)

Standard Operating Conditions (unless otherwise stated)							
Operating temperature $0^{\circ}\text{C} \leq T_A \leq +70^{\circ}\text{C}$ for commercial $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for industrial $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for extended Operating voltage $V_{DD}$ range as described in DC spec <b>Section 12.1</b> <b>“DC Characteristics: PIC16C712/716-04 (Commercial, Industrial, Extended) PIC16C712/716-20 (Commercial, Industrial, Extended)”</b> and <b>Section 12.2</b> <b>“DC Characteristics: PIC16LC712/716-04 (Commercial, Industrial)”</b>							
Param No.	Sym.	Characteristic	Min.	Typ†	Max.	Units	Conditions
D030 D030A D031 D032 D033	$V_{IL}$	<b>Input Low Voltage</b> I/O ports with TTL buffer with Schmitt Trigger buffer $\overline{MCLR}$ , OSC1 (in RC mode) OSC1 (in XT, HS and LP modes)	$V_{SS}$ $V_{SS}$ $V_{SS}$ $V_{SS}$ $V_{SS}$	— — — — —	0.8V 0.15V <sub>DD</sub> 0.2V <sub>DD</sub> 0.2V <sub>DD</sub> 0.3V <sub>DD</sub>	V V V V V	$4.5\text{V} \leq V_{DD} \leq 5.5\text{V}$ otherwise  <b>(Note 1)</b>
D040 D040A D041 D042 D042A D043	$V_{IH}$	<b>Input High Voltage</b> I/O ports with TTL buffer with Schmitt Trigger buffer $\overline{MCLR}$ OSC1 (XT, HS and LP modes) OSC1 (in RC mode)	2.0 0.25V <sub>DD</sub> + 0.8V 0.8V <sub>DD</sub> 0.8V <sub>DD</sub> 0.7V <sub>DD</sub> 0.9V <sub>DD</sub>	— — — — — — —	$V_{DD}$ $V_{DD}$ $V_{DD}$ $V_{DD}$ $V_{DD}$ $V_{DD}$ $V_{DD}$	V V V V V V V	$4.5\text{V} \leq V_{DD} \leq 5.5\text{V}$ otherwise  For entire $V_{DD}$ range <b>(Note 1)</b>
D060 D061 D063	$I_{IL}$	<b>Input Leakage Current (Notes 2, 3)</b> I/O ports $\overline{MCLR}$ , RA4/T0CKI OSC1	— — —	— — —	$\pm 1$ $\pm 5$ $\pm 5$	$\mu\text{A}$ $\mu\text{A}$ $\mu\text{A}$	$V_{SS} \leq V_{PIN} \leq V_{DD}$ , Pin at high-impedance $V_{SS} \leq V_{PIN} \leq V_{DD}$ $V_{SS} \leq V_{PIN} \leq V_{DD}$ , XT, HS and LP osc modes
D070	IPURB	<b>PORTB weak pull-up current</b>	50	250	400	$\mu\text{A}$	$V_{DD} = 5\text{V}$ , $V_{PIN} = V_{SS}$

\* 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:** In RC Oscillator mode, the OSC1/CLKIN pin is a Schmitt Trigger input. It is not recommended that the PIC MCU be driven with external clock in RC mode.
- 2:** The leakage current on the  $\overline{MCLR}/V_{PP}$  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 current sourced by the pin.

# PIC16C712/716

<b>Standard Operating Conditions (unless otherwise stated)</b> Operating temperature $0^{\circ}\text{C} \leq T_A \leq +70^{\circ}\text{C}$ for commercial $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for industrial $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for extended  Operating voltage $V_{DD}$ range as described in DC spec <b>Section 12.1</b> <b>“DC Characteristics: PIC16C712/716-04 (Commercial, Industrial, Extended) PIC16C712/716-20 (Commercial, Industrial, Extended)”</b> and <b>Section 12.2</b> <b>“DC Characteristics: PIC16LC712/716-04 (Commercial, Industrial)”</b>							
Param No.	Sym.	Characteristic	Min.	Typ†	Max.	Units	Conditions
D080	VOL	<b>Output Low Voltage</b> I/O ports	—	—	0.6	V	$I_{OL} = 8.5\text{ mA}$ , $V_{DD} = 4.5\text{V}$ , -40°C to +85°C
D083		OSC2/CLKOUT (RC Osc mode)	—	—	0.6	V	$I_{OL} = 7.0\text{ mA}$ , $V_{DD} = 4.5\text{V}$ , -40°C to +125°C
			—	—	0.6	V	$I_{OL} = 1.6\text{ mA}$ , $V_{DD} = 4.5\text{V}$ , -40°C to +85°C
			—	—	0.6	V	$I_{OL} = 1.2\text{ mA}$ , $V_{DD} = 4.5\text{V}$ , -40°C to +125°C
D090	VOH	<b>Output High Voltage</b> I/O ports ( <b>Note 3</b> )	$V_{DD}-0.7$	—	—	V	$I_{OH} = -3.0\text{ mA}$ , $V_{DD} = 4.5\text{V}$ , -40°C to +85°C
			$V_{DD}-0.7$	—	—	V	$I_{OH} = -2.5\text{ mA}$ , $V_{DD} = 4.5\text{V}$ , -40°C to +125°C
D092		OSC2/CLKOUT (RC Osc mode)	$V_{DD}-0.7$	—	—	V	$I_{OH} = -1.3\text{ mA}$ , $V_{DD} = 4.5\text{V}$ , -40°C to +85°C
			$V_{DD}-0.7$	—	—	V	$I_{OH} = -1.0\text{ mA}$ , $V_{DD} = 4.5\text{V}$ , -40°C to +125°C
D150*	VOD	<b>Open-Drain High Voltage</b>	—	—	8.5	V	RA4 pin
		<b>Capacitive Loading Specs on Output Pins</b>					
D100	Cosc2	OSC2 pin	—	—	15	pF	In XT, HS and LP modes when external clock is used to drive OSC1.
D101	Cio	All I/O pins and OSC2 (in RC mode)	—	—	50	pF	

\* 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:** In RC Oscillator mode, the OSC1/CLKIN pin is a Schmitt Trigger input. It is not recommended that the PIC MCU be driven with external clock in RC mode.
- 2:** The leakage current on the  $\overline{\text{MCLR}}/\text{VPP}$  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 current sourced by the pin.

## 12.4 AC (Timing) Characteristics

### 12.4.1 TIMING PARAMETER SYMBOLOGY

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

1. TppS2ppS
2. TppS

<b>T</b>			
F	Frequency	T	Time

Lowercase letters (pp) and their meanings:

<b>pp</b>			
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:

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

# PIC16C712/716

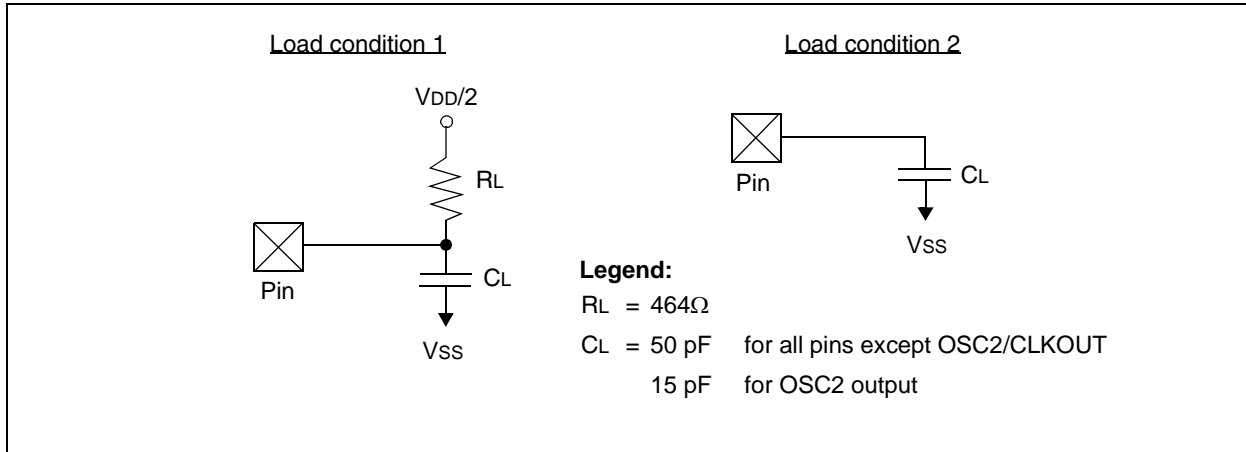
## 12.4.2 TIMING CONDITIONS

The temperature and voltages specified in Table 12-1 apply to all timing specifications, unless otherwise noted. Figure 12-3 specifies the load conditions for the timing specifications.

**TABLE 12-1: TEMPERATURE AND VOLTAGE SPECIFICATIONS – AC**

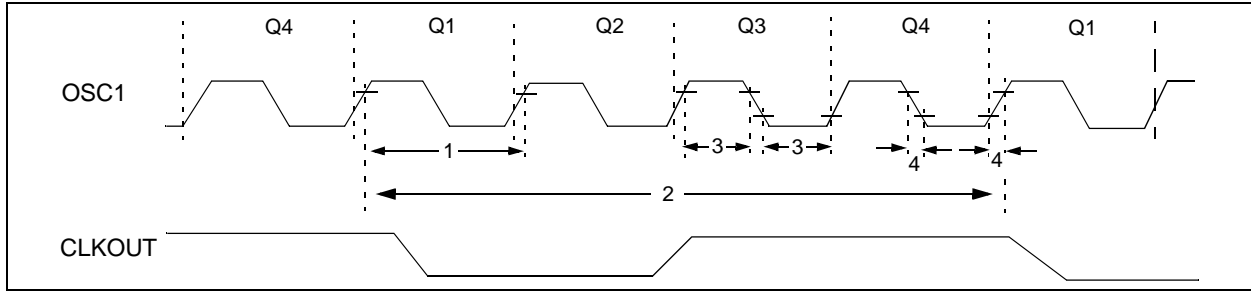
	<b>Standard Operating Conditions (unless otherwise stated)</b>
	Operating temperature $0^{\circ}\text{C} \leq T_A \leq +70^{\circ}\text{C}$ for commercial
	$-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for industrial
	$-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for extended
<b>AC CHARACTERISTICS</b>	Operating voltage $V_{DD}$ range as described in DC spec <b>Section 12.1 “DC Characteristics: PIC16C712/716-04 (Commercial, Industrial, Extended) PIC16C712/716-20 (Commercial, Industrial, Extended)”</b> and <b>Section 12.2 “DC Characteristics: PIC16C712/716-04 (Commercial, Industrial)”</b> .
	LC parts operate for commercial/industrial temp's only.

**FIGURE 12-3: LOAD CONDITIONS FOR DEVICE TIMING SPECIFICATIONS**



## 12.4.3 TIMING DIAGRAMS AND SPECIFICATIONS

**FIGURE 12-4: EXTERNAL CLOCK TIMING**



**TABLE 12-2: EXTERNAL CLOCK TIMING REQUIREMENTS**

Param No.	Sym.	Characteristic	Min.	Typ†	Max.	Units	Conditions
1A	Fosc	<b>External CLKIN Frequency (Note 1)</b>	DC	—	4	MHz	RC and XT osc modes
			DC	—	4	MHz	HS osc mode (-04)
			DC	—	20	MHz	HS osc mode (-20)
			DC	—	200	kHz	LP osc mode
		<b>Oscillator Frequency (Note 1)</b>	DC	—	4	MHz	RC osc mode
			0.1	—	4	MHz	XT osc mode
			4	—	20	MHz	HS osc mode
			5	—	200	kHz	LP osc mode
1	Tosc	<b>External CLKIN Period (Note 1)</b>	250	—	—	ns	RC and XT osc modes
			250	—	—	ns	HS osc mode (-04)
			50	—	—	ns	HS osc mode (-20)
			5	—	—	μs	LP osc mode
		<b>Oscillator Period (Note 1)</b>	250	—	—	ns	RC osc mode
			250	—	10,000	ns	XT osc mode
			250	—	250	ns	HS osc mode (-04)
			50	—	250	ns	HS osc mode (-20)
			5	—	—	μs	LP osc mode
2	Tcy	<b>Instruction Cycle Time (Note 1)</b>	200	—	DC	ns	Tcy = 4/Fosc
3*	TosL, TosH	<b>External Clock in (OSC1) High or Low Time</b>	100	—	—	ns	XT oscillator
			2.5	—	—	μs	LP oscillator
			15	—	—	ns	HS oscillator
4*	TosR, TosF	<b>External Clock in (OSC1) Rise or Fall Time</b>	—	—	25	ns	XT oscillator
			—	—	50	ns	LP oscillator
			—	—	15	ns	HS oscillator

\* These parameters are characterized but not tested.

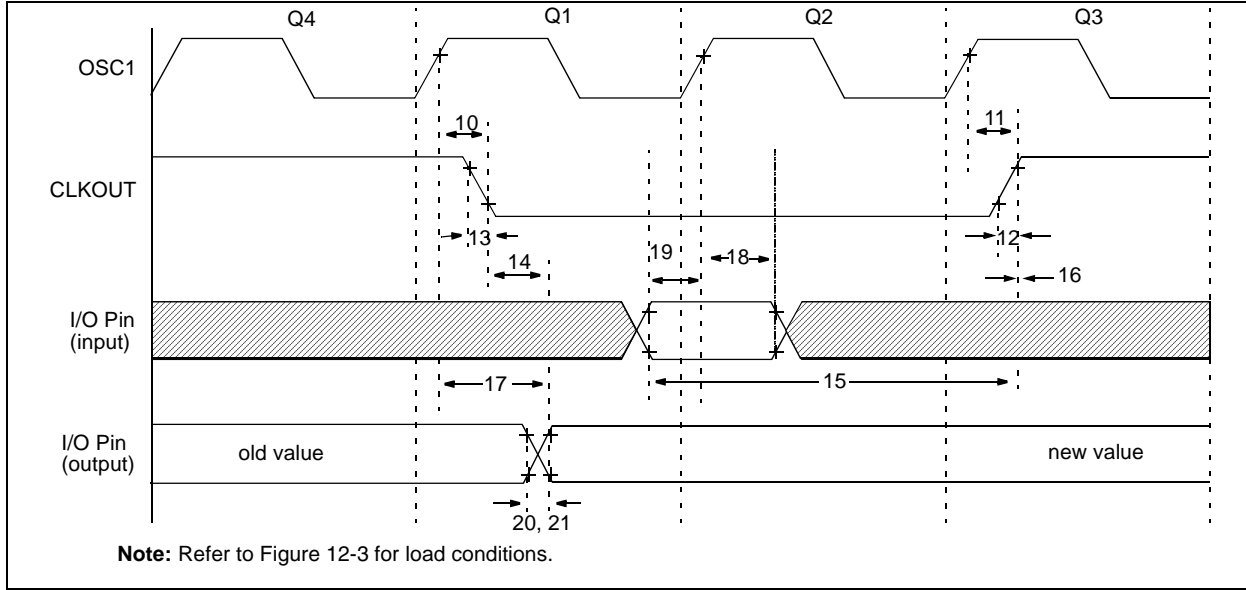
† Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

**Note1:** Instruction cycle period (Tcy) equals four times the input oscillator time base period. All specified values are based on characterization data for that particular oscillator type under standard operating conditions with the device executing code. Exceeding these specified limits may result in an unstable oscillator operation and/or higher than expected current consumption. All devices are tested to operate at "min." values with an external clock applied to the OSC1/CLKIN pin.

When an external clock input is used, the "Max." cycle time limit is "DC" (no clock) for all devices.

# PIC16C712/716

**FIGURE 12-5: CLKOUT AND I/O TIMING**



**TABLE 12-3: CLKOUT AND I/O TIMING REQUIREMENTS**

Param No.	Sym.	Characteristic	Min.	Typ†	Max.	Units	Conditions	
10*	TosH2ckL	OSC1↑ to CLKOUT↓	—	75	200	ns	<b>Note 1</b>	
11*	TosH2ckH	OSC1! to CLKOUT!	—	75	200	ns	<b>Note 1</b>	
12*	TckR	CLKOUT rise time	—	35	100	ns	<b>Note 1</b>	
13*	TckF	CLKOUT fall time	—	35	100	ns	<b>Note 1</b>	
14*	TckL2ioV	CLKOUT ∅ to Port out valid	—	—	0.5T <sub>CY</sub> + 20	ns	<b>Note 1</b>	
15*	TioV2ckH	Port in valid before CLKOUT !	Tosc + 200	—	—	ns	<b>Note 1</b>	
16*	TckH2iol	Port in hold after CLKOUT !	0	—	—	ns	<b>Note 1</b>	
17*	TosH2ioV	OSC1! (Q1 cycle) to Port out valid	—	50	150	ns		
18*	TosH2iol	OSC1! (Q2 cycle) to Port input invalid (I/O in hold time)	Standard	100	—	—	ns	
18A*			Extended (LC)	200	—	—	ns	
19*	TioV2osH	Port input valid to OSC1! (I/O in setup time)	0	—	—	ns		
20*	TioR	Port output rise time	Standard	—	10	40	ns	
20A*			Extended (LC)	—	—	80	ns	
21*	TioF	Port output fall time	Standard	—	10	40	ns	
21A*			Extended (LC)	—	—	80	ns	
22††	TINP	INT pin high or low time	T <sub>CY</sub>	—	—	ns		
23††	TRBP	RB7:RB4 change INT high or low time	T <sub>CY</sub>	—	—	ns		

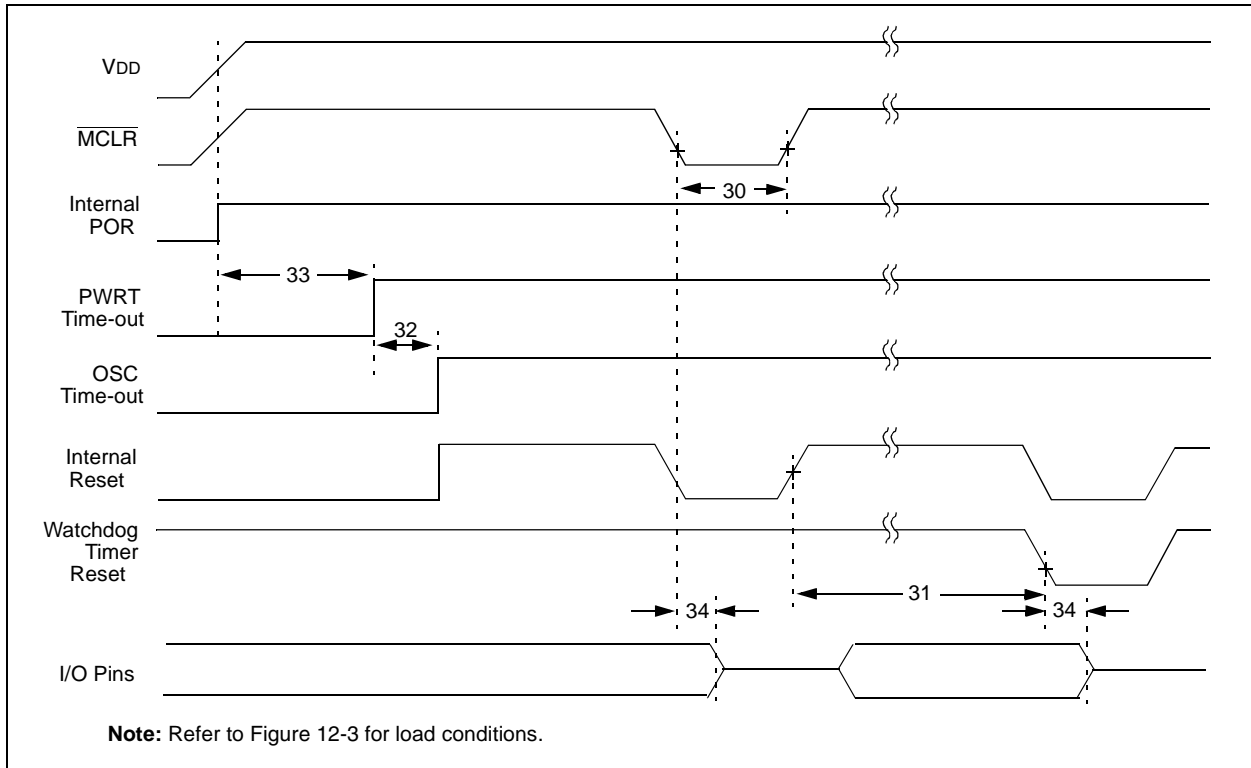
\* These parameters are characterized but not tested.

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

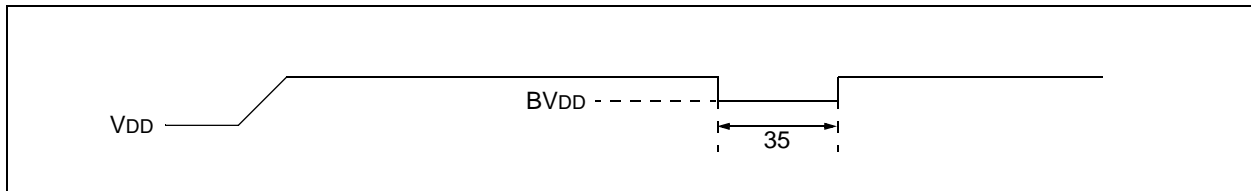
†† These parameters are asynchronous events not related to any internal clock edge.

**Note1:** Measurements are taken in RC mode where CLKOUT output is 4 x T<sub>osc</sub>.

**FIGURE 12-6: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER AND POWER-UP TIMER TIMING**



**FIGURE 12-7: BROWN-OUT RESET TIMING**



**TABLE 12-4: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER, POWER-UP TIMER, AND BROWN-OUT RESET REQUIREMENTS**

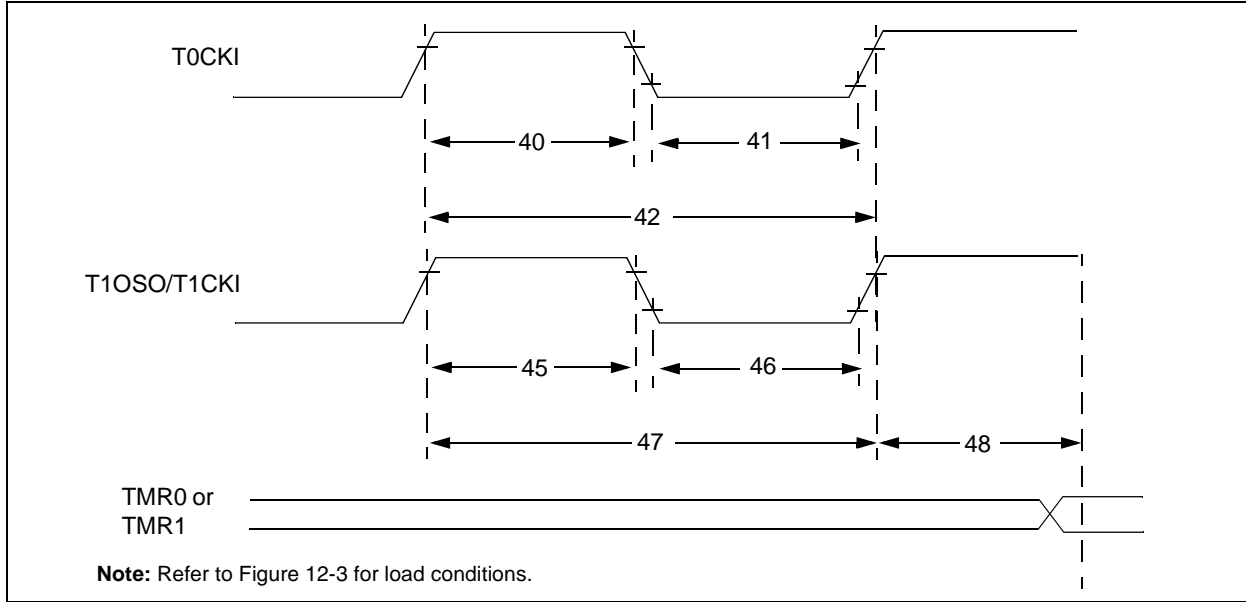
Parameter No.	Sym.	Characteristic	Min.	Typ†	Max.	Units	Conditions
30	TmCL	MCLR Pulse Width (low)	2	—	—	μs	VDD = 5V, -40°C to +125°C
31*	TWDT	Watchdog Timer Time-out Period (No Prescaler)	7	18	33	ms	VDD = 5V, -40°C to +125°C
32	TOST	Oscillation Start-up Timer Period	—	1024 TOSC	—	—	TOSC = OSC1 period
33*	TPWRT	Power-up Timer Period	28	72	132	ms	VDD = 5V, -40°C to +125°C
34	TIOZ	I/O High-impedance from MCLR Low or WDT Reset	—	—	2.1	μs	
35	TBOR	Brown-out Reset Pulse Width	100	—	—	μs	VDD ≤ BVDD (D005)

\* 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.

# PIC16C712/716

**FIGURE 12-8: TIMER0 AND TIMER1 EXTERNAL CLOCK TIMINGS**



**TABLE 12-5: TIMER0 AND TIMER1 EXTERNAL CLOCK REQUIREMENTS**

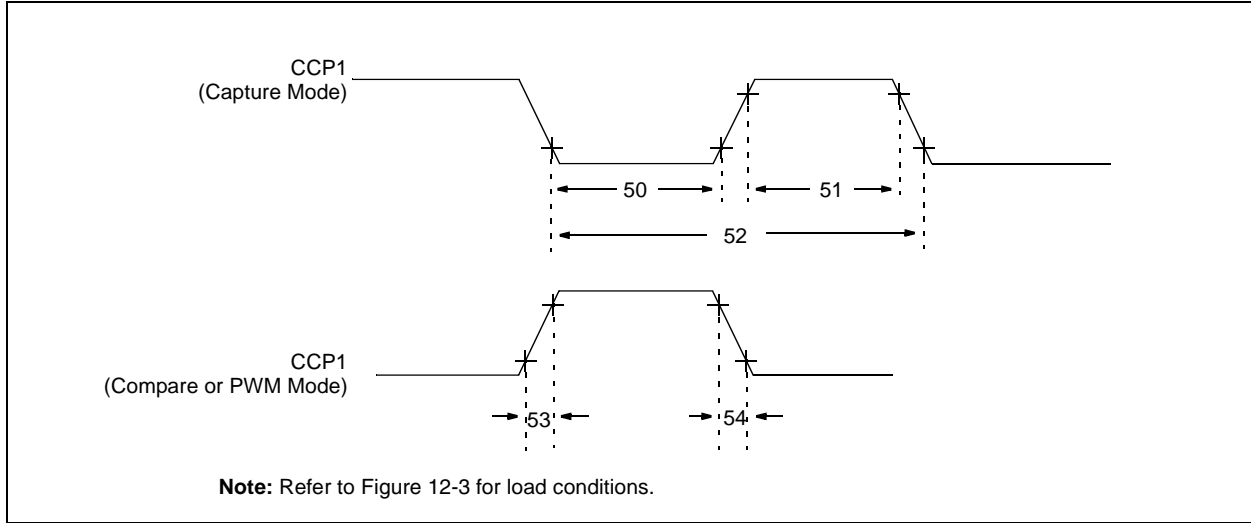
Param No.	Sym.	Characteristic	Min.	Typ†	Max.	Units	Conditions		
40*	Tt0H	T0CKI High Pulse Width	No Prescaler 0.5TCY + 20 With Prescaler 10	—	—	ns	Must also meet parameter 42		
41*	Tt0L	T0CKI Low Pulse Width	No Prescaler 0.5TCY + 20 With Prescaler 10	—	—	ns		Must also meet parameter 42	
42*	Tt0P	T0CKI Period	No Prescaler TCY + 40 With Prescaler Greater of: 20 or TCY + 40 N	—	—	ns	N = prescale value (2, 4, ..., 256)		
45*	Tt1H	T1CKI High Time	Synchronous, Prescaler = 1 0.5TCY + 20 Synchronous, Prescaler = 2,4,8 Standard 15 Extended (LC) 25 Asynchronous Standard 30 Extended (LC) 50	—	—	ns	Must also meet parameter 47		
46*	Tt1L	T1CKI Low Time	Synchronous, Prescaler = 1 0.5TCY + 20 Synchronous, Prescaler = 2,4,8 Standard 15 Extended (LC) 25 Asynchronous Standard 30 Extended (LC) 50	—	—	ns		Must also meet parameter 47	
47*	Tt1P	T1CKI input period	Synchronous Standard Greater of: 30 or TCY + 40 N Extended (LC) Greater of: 50 or TCY + 40 N Asynchronous Standard 60 Extended (LC) 100	—	—	ns			N = prescale value (1, 2, 4, 8)
	Ft1	Timer1 oscillator input frequency range (oscillator enabled by setting bit T1OSCEN)	DC	—	200	kHz			
48	TCKEZtmr1	Delay from external clock edge to timer increment	2Tosc	—	7Tosc	—			

\* 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.



**FIGURE 12-9: CAPTURE/COMPARE/PWM TIMINGS**



**TABLE 12-6: CAPTURE/COMPARE/PWM REQUIREMENTS**

Param No.	Sym.	Characteristic	Min	Typ†	Max	Units	Conditions	
50*	TccL	CCP1 input low time	No Prescaler		0.5Tcy + 20	—	—	ns
			With Prescaler	Standard	10	—	—	ns
				Extended (LC)	20	—	—	ns
51*	TccH	CCP1 input high time	No Prescaler		0.5Tcy + 20	—	—	ns
			With Prescaler	Standard	10	—	—	ns
				Extended (LC)	20	—	—	ns
52*	TccP	CCP1 input period	$\frac{3Tcy + 40}{N}$		—	—	ns	N = prescale value (1, 4, or 16)
53*	TccR	CCP1 output rise time	Standard	—	10	25	ns	
			Extended (LC)	—	25	45	ns	
54*	TccF	CCP1 output fall time	Standard	—	10	25	ns	
			Extended (LC)	—	25	45	ns	

\* 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.

# PIC16C712/716

**TABLE 12-7: A/D CONVERTER CHARACTERISTICS:  
PIC16C712/716-04 (COMMERCIAL, INDUSTRIAL, EXTENDED)  
PIC16C712/716-20 (COMMERCIAL, INDUSTRIAL, EXTENDED)  
PIC16LC712/716-04 (COMMERCIAL, INDUSTRIAL)**

Param No.	Sym.	Characteristic	Min.	Typ†	Max.	Units	Conditions	
A01	NR	Resolution	—	—	8-bits	bit	VREF = VDD = 5.12V, VSS £ VAIN £ VREF	
A02	EABS	Total Absolute error	—	—	< ± 1	LSb	VREF = VDD = 5.12V, VSS £ VAIN £ VREF	
A03	EIL	Integral linearity error	—	—	< ± 1	LSb	VREF = VDD = 5.12V, VSS £ VAIN £ VREF	
A04	EDL	Differential linearity error	—	—	< ± 1	LSb	VREF = VDD = 5.12V, VSS £ VAIN £ VREF	
A05	EFS	Full scale error	—	—	< ± 1	LSb	VREF = VDD = 5.12V, VSS £ VAIN £ VREF	
A06	E0FF	Offset error	—	—	< ± 1	LSb	VREF = VDD = 5.12V, VSS £ VAIN £ VREF	
A10	—	Monotonicity	—	guaranteed (Note 3)	—	—	VSS £ VAIN £ VREF	
A20	VREF	Reference voltage	2.5V	—	VDD + 0.3	V		
A25	VAIN	Analog input voltage	VSS - 0.3	—	VREF + 0.3	V		
A30	ZAIN	Recommended impedance of analog voltage source	—	—	10.0	kΩ		
A40	IAD	A/D conversion current (VDD)	Standard	—	180	—	μA	Average current consumption when A/D is on. (Note 1)
			Extended (LC)	—	90	—	μA	
A50	IREF	VREF input current (Note 2)	10	—	1000	—	μA	During VAIN acquisition. Based on differential of VHOLD to VAIN to charge CHOLD, see Section 9.1 "Configuration Bits".  During A/D Conversion cycle
			—	—	10	—	μA	

2: \* These parameters are characterized but not tested.

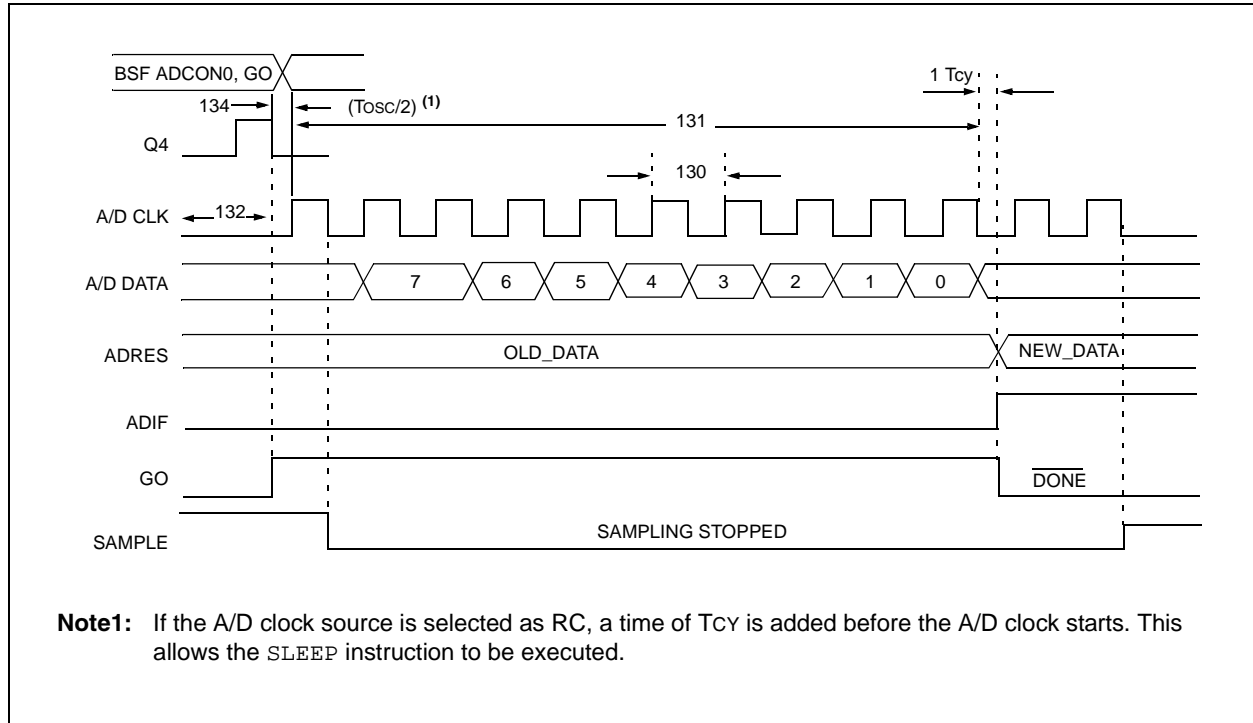
3: † Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

**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: The A/D conversion result never decreases with an increase in the Input Voltage, and has no missing codes.

**FIGURE 12-10: A/D CONVERSION TIMING**



**TABLE 12-8: A/D CONVERSION REQUIREMENTS**

Param No.	Sym.	Characteristic		Min.	Typ†	Max.	Units	Conditions
130	TAD	A/D clock period	Standard	1.6	—	—	$\mu\text{s}$	TOSC based, $V_{REF} \geq 3.0\text{V}$
			Extended (LC)	2.0	—	—	$\mu\text{s}$	TOSC based, $V_{REF}$ full range
			Standard	2.0	4.0	6.0	$\mu\text{s}$	A/D RC Mode
			Extended (LC)	3.0	6.0	9.0	$\mu\text{s}$	A/D RC Mode
131	Tcnv	Conversion time (not including S/H time) <b>(Note 1)</b>		11	—	11	TAD	
132	TACQ	Acquisition time		<b>(Note 2)</b>	20	—	$\mu\text{s}$	The minimum time is the amplifier settling time. This may be used if the “new” input voltage has not changed by more than 1 LSB (i.e., 20.0 mV @ 5.12V) from the last sampled voltage (as stated on CHOLD).
				5*	—	—	$\mu\text{s}$	
134	Tgo	Q4 to A/D clock start		—	$T_{osc}/2$ §	—	—	If the A/D clock source is selected as RC, a time of $T_{CY}$ is added before the A/D clock starts. This allows the <code>SLEEP</code> instruction to be executed.
135	Tswc	Switching from convert $\overline{AE}$ sample time		1.5 §	—	—	TAD	

: \* 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.

: § This specification ensured by design.

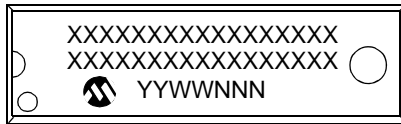
**Note 1:** ADRES register may be read on the following  $T_{CY}$  cycle.

**2:** See **Section 9.1 “Configuration Bits”** for min. conditions.

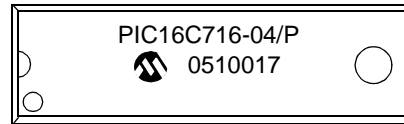
## 13.0 PACKAGING INFORMATION

### 13.1 Package Marking Information

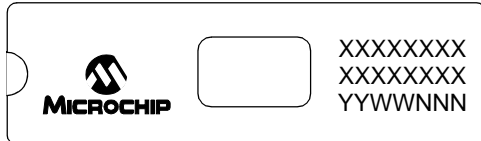
18-Lead PDIP



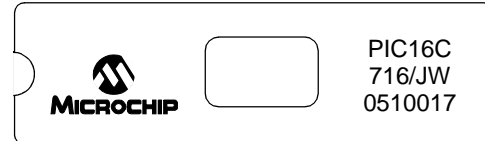
Example



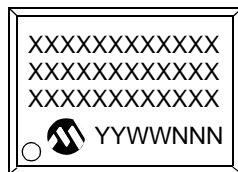
18-Lead CERDIP Windowed



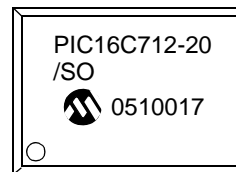
Example



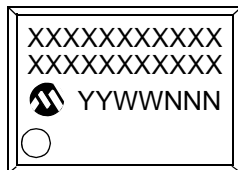
18-Lead SOIC (.300")



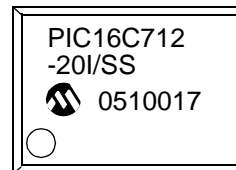
Example



20-Lead SSOP



Example



<b>Legend:</b>	XX...X	Customer-specific information
	Y	Year code (last digit of calendar year)
	YY	Year code (last 2 digits of calendar year)
	WW	Week code (week of January 1 is week '01')
	NNN	Alphanumeric traceability code
	(e3)	Pb-free JEDEC designator for Matte Tin (Sn)
	*	This package is Pb-free. The Pb-free JEDEC designator (e3) can be found on the outer packaging for this package.

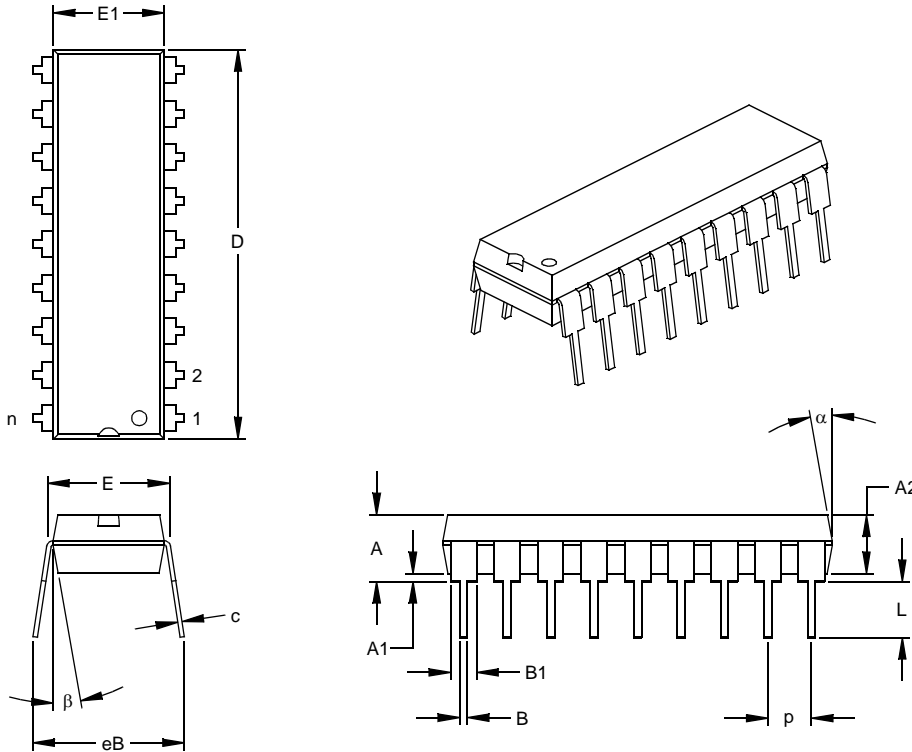
**Note:** In the event the full Microchip part number cannot be marked on one line, it will be carried over to the next line, thus limiting the number of available characters for customer-specific information.

# PIC16C712/716

## 13.2 Package Details

The following sections give the technical details of the packages.

### 18-Lead Plastic Dual In-line (P) – 300 mil (PDIP)



Dimension Limits	Units	INCHES*			MILLIMETERS		
	n	MIN	NOM	MAX	MIN	NOM	MAX
Number of Pins	n		18			18	
Pitch	p		.100			2.54	
Top to Seating Plane	A	.140	.155	.170	3.56	3.94	4.32
Molded Package Thickness	A2	.115	.130	.145	2.92	3.30	3.68
Base to Seating Plane	A1	.015			0.38		
Shoulder to Shoulder Width	E	.300	.313	.325	7.62	7.94	8.26
Molded Package Width	E1	.240	.250	.260	6.10	6.35	6.60
Overall Length	D	.890	.898	.905	22.61	22.80	22.99
Tip to Seating Plane	L	.125	.130	.135	3.18	3.30	3.43
Lead Thickness	c	.008	.012	.015	0.20	0.29	0.38
Upper Lead Width	B1	.045	.058	.070	1.14	1.46	1.78
Lower Lead Width	B	.014	.018	.022	0.36	0.46	0.56
Overall Row Spacing §	eB	.310	.370	.430	7.87	9.40	10.92
Mold Draft Angle Top	α	5	10	15	5	10	15
Mold Draft Angle Bottom	β	5	10	15	5	10	15

\* Controlling Parameter

§ Significant Characteristic

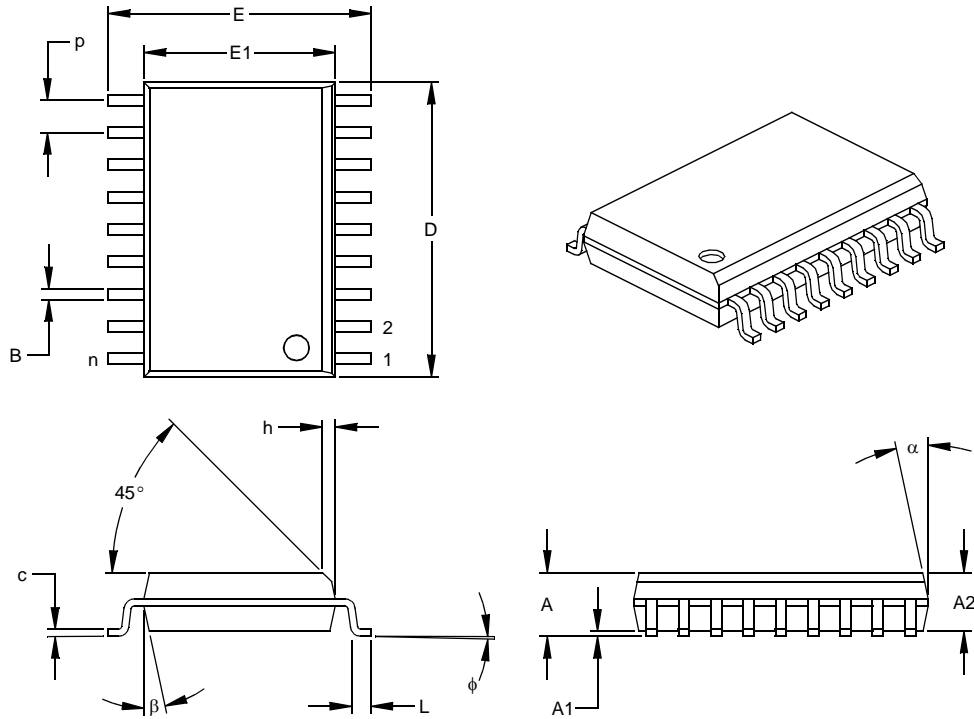
Notes:

Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010" (0.254mm) per side.

JEDEC Equivalent: MS-001

Drawing No. C04-007

## 18-Lead Plastic Small Outline (SO) – Wide, 300 mil (SOIC)



Units		INCHES*			MILLIMETERS		
Dimension	Limits	MIN	NOM	MAX	MIN	NOM	MAX
Number of Pins	n		18			18	
Pitch	P		.050			1.27	
Overall Height	A	.093	.099	.104	2.36	2.50	2.64
Molded Package Thickness	A2	.088	.091	.094	2.24	2.31	2.39
Standoff §	A1	.004	.008	.012	0.10	0.20	0.30
Overall Width	E	.394	.407	.420	10.01	10.34	10.67
Molded Package Width	E1	.291	.295	.299	7.39	7.49	7.59
Overall Length	D	.446	.454	.462	11.33	11.53	11.73
Chamfer Distance	h	.010	.020	.029	0.25	0.50	0.74
Foot Length	L	.016	.033	.050	0.41	0.84	1.27
Foot Angle	φ	0	4	8	0	4	8
Lead Thickness	c	.009	.011	.012	0.23	0.27	0.30
Lead Width	B	.014	.017	.020	0.36	0.42	0.51
Mold Draft Angle Top	α	0	12	15	0	12	15
Mold Draft Angle Bottom	β	0	12	15	0	12	15

\* Controlling Parameter  
 § Significant Characteristic

Notes:

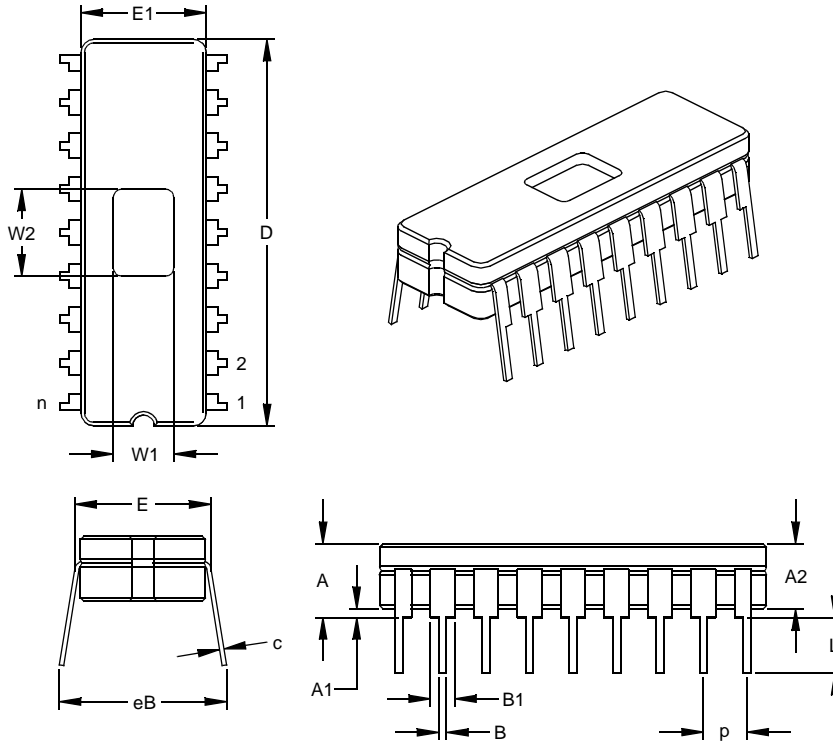
Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010" (0.254mm) per side.

JEDEC Equivalent: MS-013

Drawing No. C04-051

# PIC16C712/716

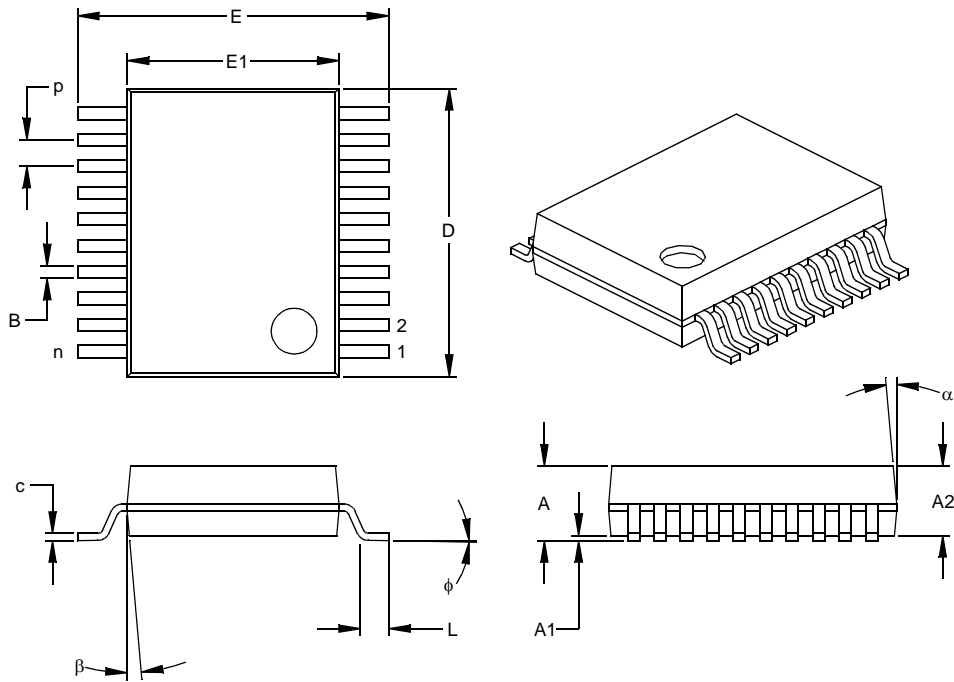
## 18-Lead Ceramic Dual In-line with Window (JW) – 300 mil (CERDIP)



Units		INCHES*			MILLIMETERS		
Dimension Limits		MIN	NOM	MAX	MIN	NOM	MAX
Number of Pins	n		18			18	
Pitch	p		.100			2.54	
Top to Seating Plane	A	.170	.183	.195	4.32	4.64	4.95
Ceramic Package Height	A2	.155	.160	.165	3.94	4.06	4.19
Standoff	A1	.015	.023	.030	0.38	0.57	0.76
Shoulder to Shoulder Width	E	.300	.313	.325	7.62	7.94	8.26
Ceramic Pkg. Width	E1	.285	.290	.295	7.24	7.37	7.49
Overall Length	D	.880	.900	.920	22.35	22.86	23.37
Tip to Seating Plane	L	.125	.138	.150	3.18	3.49	3.81
Lead Thickness	c	.008	.010	.012	0.20	0.25	0.30
Upper Lead Width	B1	.050	.055	.060	1.27	1.40	1.52
Lower Lead Width	B	.016	.019	.021	0.41	0.47	0.53
Overall Row Spacing	§ eB	.345	.385	.425	8.76	9.78	10.80
Window Width	W1	.130	.140	.150	3.30	3.56	3.81
Window Length	W2	.190	.200	.210	4.83	5.08	5.33

\* Controlling Parameter  
 § Significant Characteristic  
 JEDEC Equivalent: MO-036  
 Drawing No. C04-010

## 20-Lead Plastic Shrink Small Outline (SS) – 209 mil, 5.30 mm (SSOP)



Units		INCHES*			MILLIMETERS		
Dimension	Limits	MIN	NOM	MAX	MIN	NOM	MAX
Number of Pins	n		20			20	
Pitch	p		.026			0.65	
Overall Height	A	.068	.073	.078	1.73	1.85	1.98
Molded Package Thickness	A2	.064	.068	.072	1.63	1.73	1.83
Standoff §	A1	.002	.006	.010	0.05	0.15	0.25
Overall Width	E	.299	.309	.322	7.59	7.85	8.18
Molded Package Width	E1	.201	.207	.212	5.11	5.25	5.38
Overall Length	D	.278	.284	.289	7.06	7.20	7.34
Foot Length	L	.022	.030	.037	0.56	0.75	0.94
Lead Thickness	c	.004	.007	.010	0.10	0.18	0.25
Foot Angle	φ	0	4	8	0.00	101.60	203.20
Lead Width	B	.010	.013	.015	0.25	0.32	0.38
Mold Draft Angle Top	α	0	5	10	0	5	10
Mold Draft Angle Bottom	β	0	5	10	0	5	10

\* Controlling Parameter  
 § Significant Characteristic

**Notes:**

Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010" (0.254mm) per side.

JEDEC Equivalent: MO-150

Drawing No. C04-072



## APPENDIX A: REVISION HISTORY

Version	Date	Revision Description
A	2/99	This is a new data sheet. However, the devices described in this data sheet are the upgrades to the devices found in the <i>PIC16C6X Data Sheet</i> , DS30234, and the <i>PIC16C7X Data Sheet</i> , DS30390.
B	9/05	Removed Preliminary Status.
C	1/13	Added a note to each package outline drawing.

## APPENDIX B: CONVERSION CONSIDERATIONS

There are no previous versions of this device.

## APPENDIX C: MIGRATION FROM BASE-LINE TO MID-RANGE DEVICES

This section discusses how to migrate from a baseline device (i.e., PIC16C5X) to a mid-range device (i.e., PIC16CXXX).

The following are the list of modifications over the PIC16C5X microcontroller family:

- Instruction word length is increased to 14-bits. This allows larger page sizes both in program memory (2K now as opposed to 512 before) and register file (128 bytes now versus 32 bytes before).
- A PC high latch register (PCLATH) is added to handle program memory paging. Bits PA2, PA1, PA0 are removed from STATUS register.
- Data memory paging is redefined slightly. STATUS register is modified.
- Four new instructions have been added: RETURN, RETFIE, ADDLW, and SUBLW. Two instructions TRIS and OPTION are being phased out although they are kept for compatibility with PIC16C5X.
- OPTION\_REG and TRIS registers are made addressable.
- Interrupt capability is added. Interrupt vector is at 0004h.
- Stack size is increased to 8 deep.
- Reset vector is changed to 0000h.
- Reset of all registers is revisited. Five different Reset (and wake-up) types are recognized. Registers are reset differently.
- Wake-up from Sleep through interrupt is added.
- Two separate timers, Oscillator Start-up Timer (OST) and Power-up Timer (PWRT) are included for more reliable power-up. These timers are invoked selectively to avoid unnecessary delays on power-up and wake-up.
- PORTB has weak pull-ups and interrupt on change feature.
- T0CKI pin is also a port pin (RA4) now.
- FSR is made a full eight-bit register.
- “In-circuit serial programming” is made possible. The user can program PIC16CXX devices using only five pins: VDD, VSS, MCLR/VPP, RB6 (clock) and RB7 (data in/out).
- PCON STATUS register is added with a Power-on Reset Status bit (POR).
- Code protection scheme is enhanced such that portions of the program memory can be protected, while the remainder is unprotected.
- Brown-out protection circuitry has been added. Controlled by Configuration Word bit BODEN. Brown-out Reset ensures the device is placed in a Reset condition if VDD dips below a fixed setpoint.

To convert code written for PIC16C5X to PIC16CXXX, the user should take the following steps:

- Remove any program memory page select operations (PA2, PA1, PA0 bits) for CALL, GOTO.
- Revisit any computed jump operations (write to PC or add to PC, etc.) to make sure page bits are set properly under the new scheme.
- Eliminate any data memory page switching. Redefine data variables to reallocate them.
- Verify all writes to STATUS, OPTION, and FSR registers since these have changed.
- Change Reset vector to 0000h.

## INDEX

### A

A/D	45	Enable (CCP1IE Bit)	16
A/D Converter Enable (ADIE Bit)	16	Flag (CCP1IF Bit)	17
A/D Converter Flag (ADIF Bit)	17, 47	PWM Mode. See PWM	
A/D Converter Interrupt, Configuring	47	Timer Resources	39
ADCON0 Register	11, 45	Timing Diagram	85
ADCON1 Register	12, 45, 46	CCP1CON Register	39
ADRES Register	11, 45, 47	CCP1M3:CCP1M0 Bits	39
Analog Port Pins, Configuring	49	CCP1X:CCP1Y Bits	39
Block Diagram	47	Code Protection	51, 65
Block Diagram, Analog Input Model	48	CP1:CP0 Bits	52
Channel Select (CHS2:CHS0 Bits)	45	Compare (CCP Module)	41
Clock Select (ADCS1:ADCS0 Bits)	45	Block Diagram	41
Configuring the Module	47	CCP Pin Configuration	41
Conversion Clock (Tad)	49	CCPR1H:CCPR1L Registers	41
Conversion Status (GO/DONE Bit)	45, 47	Software Interrupt	41
Conversions	50	Special Event Trigger	34, 41, 50
Converter Characteristics	86	Timer1 Mode Selection	41
Module On/Off (ADON Bit)	45	Configuration Bits	51
Port Configuration Control (PCFG2:PCFG0 Bits)	46	Conversion Considerations	95
Sampling Requirements	48	Customer Change Notification Service	101
Special Event Trigger (CCP)	41, 50	Customer Notification Service	101
Timing Diagram	87	Customer Support	101
Absolute Maximum Ratings	73	<b>D</b>	
ADCON0 Register	11, 45	Data Memory	10
ADCS1:ADCS0 Bits	45	Bank Select (RP1:RP0 Bits)	10, 13
ADON Bit	45	General Purpose Registers	10
CHS2:CHS0 Bits	45	Register File Map	10
GO/DONE Bit	45, 47	Special Function Registers	11
ADCON1 Register	12, 45, 46	DC Characteristics	75, 77
PCFG2:PCFG0 Bits	46	Development Support	69
ADRES Register	11, 45, 47	Direct Addressing	20
Analog-to-Digital Converter. See A/D Architecture		<b>E</b>	
PIC16C712/716 Block Diagram	5	Electrical Characteristics	73
Assembler		Errata	3
MPASM Assembler	70	External Power-on Reset Circuit	55
<b>B</b>		<b>F</b>	
Banking, Data Memory	10, 13	Family of Devices	
BOR. See Brown-out Reset		PIC16C7XX	2
Brown-Out Reset (BOR)	55	Firmware Instructions	67
Brown-out Reset (BOR)	51, 54, 58, 59	<b>I</b>	
BOR Enable (BODEN Bit)	52	I/O Ports	21
BOR Status (BOR Bit)	18	ID Locations	51, 65
Timing Diagram	83	In-Circuit Serial Programming™ (ICSP™)	51, 65
<b>C</b>		Indirect Addressing	20
C Compilers		FSR Register	10, 11, 20
MPLAB C18	70	INDF Register	11
MPLAB C30	70	Instruction Format	67
Capture (CCP Module)	40	Instruction Set	67
Block Diagram	40	Summary Table	68
CCP Pin Configuration	40	INT Interrupt (RBO/INT). See Interrupt Sources	
CCPR1H:CCPR1L Registers	40	INTCON Register	11, 15
Changing Between Capture Prescalers	40	GIE Bit	15
Software Interrupt	40	INTE Bit	15
Timer1 Mode Selection	40	INTF Bit	15
Capture/Compare/PWM (CCP)	39	PEIE Bit	15
Capture Mode. See Capture		RBIE Bit	15
CCP1CON Register	11, 39	RBIF Bit	15, 24
CCPR1H Register	11, 39	TOIE Bit	15
CCPR1L Register	11, 39	TOIF Bit	15
Compare Mode. See Compare		Internet Address	101

# PIC16C712/716

Interrupt Sources.....	51, 61	XT .....	53, 58
A/D Conversion Complete .....	47	Oscillator, Timer1.....	31, 34
Block Diagram.....	61	Oscillator, WDT.....	63
Capture Complete (CCP).....	40	<b>P</b>	
Compare Complete (CCP).....	41	Packaging .....	89
Interrupt-on-Change (RB7:RB4 ) .....	24	Details.....	90
RB0/INT Pin, External.....	62	Paging, Program Memory.....	9, 19
TMR0 Overflow .....	30, 62	PCON Register.....	18, 58
TMR1 Overflow .....	31, 34	BOR Bit.....	18
TMR2 to PR2 Match .....	37	POR Bit.....	18
TMR2 to PR2 Match (PWM) .....	36, 42	PICSTART Plus Development Programmer.....	72
Interrupts, Context Saving During .....	62	PIE1 Register.....	12, 16
Interrupts, Enable Bits		ADIE Bit .....	16
A/D Converter Enable (ADIE Bit) .....	16	CCP1IE Bit .....	16
CCP1 Enable (CCP1IE Bit).....	16, 40	TMR1IE Bit .....	16
Global Interrupt Enable (GIE Bit) .....	15, 61	TMR2IE Bit .....	16
Interrupt-on-Change (RB7:RB4) Enable		Pin Functions	
(RBIE Bit).....	15, 62	MCLR/VPP .....	6
Peripheral Interrupt Enable (PEIE Bit) .....	15	RA0/AN0.....	6
RB0/INT Enable (INTE Bit) .....	15	RA1/AN1.....	6
TMR0 Overflow Enable (TOIE Bit).....	15	RA2/AN2.....	6
TMR1 Overflow Enable (TMR1IE Bit) .....	16	RA3/AN3/VREF.....	6
TMR2 to PR2 Match Enable (TMR2IE Bit) .....	16	RA4/T0CKI .....	6
Interrupts, Flag Bits		RB0/INT.....	7
A/D Converter Flag (ADIF Bit) .....	17, 47	RB1.....	7
CCP1 Flag (CCP1IF Bit).....	17, 40, 41	RB2.....	7
Interrupt-on-Change (RB7:RB4) Flag		RB3.....	7
(RBIF Bit).....	15, 24, 62	RB4.....	7
RB0/INT Flag (INTF Bit).....	15	RB5.....	7
TMR0 Overflow Flag (T0IF Bit) .....	15, 62	RB6.....	7
TMR1 Overflow Flag (TMR1IF Bit) .....	17	RB7.....	7
TMR2 to PR2 Match Flag (TMR2IF Bit).....	17	VDD .....	7
		Vss .....	7
<b>M</b>		Pinout Descriptions	
Master Clear (MCLR)		PIC16C712/716 Pinout Description .....	6
MCLR Reset, Normal Operation .....	54, 58, 59	PIR1 Register .....	11, 17
MCLR Reset, Sleep .....	59	ADIF Bit .....	17
MCLR Reset, Sleep .....	54, 58	CCP1IF Bit.....	17
Memory Organization		TMR1IF Bit.....	17
Data Memory .....	10	TMR2IF Bit .....	17
Program Memory .....	9	Pointer, FSR .....	20
Microchip Internet Web Site .....	101	POR. See Power-on Reset	
MPLAB ASM30 Assembler, Linker, Librarian .....	70	PORTA	
MPLAB ICD 2 In-Circuit Debugger.....	71	Initialization.....	21
MPLAB ICE 2000 High-Performance Universal		PORTA Register.....	11, 21
In-Circuit Emulator .....	71	RA3:RA0 Port Pins .....	21
MPLAB ICE 4000 High-Performance Universal		RA4/T0CKI Pin .....	22
In-Circuit Emulator .....	71	TRISA Register.....	12, 21
MPLAB Integrated Development Environment Software ...	69	PORTB	
MPLAB PM3 Device Programmer.....	71	Block Diagram of RB1/T1OSO/T1CKI Pin.....	24
MPLINK Object Linker/MPLIB Object Librarian .....	70	Block Diagram of RB2/T10SI Pin.....	25
		Block Diagram of RB3/CCP1 Pin .....	25
<b>O</b>		Initialization.....	23
OPCODE Field Descriptions .....	67	PORTB Register.....	11, 23
OPTION_REG Register .....	12, 14	Pull-up Enable (RBPU Bit).....	14
INTEDG Bit .....	14	RB0/INT Edge Select (INTEDG Bit) .....	14
PS2:PS0 Bits .....	14, 29	RB0/INT Pin, External.....	62
PSA Bit.....	14, 29	RB3:RB0 Port Pins .....	23
RBPU Bit.....	14	RB7:RB4 Interrupt-on-Change .....	62
T0CS Bit.....	14, 29	RB7:RB4 Interrupt-on-Change Enable (RBIE Bit) .....	15, 62
T0SE Bit.....	14, 29	RB7:RB4 Interrupt-on-Change Flag	
Oscillator Configuration.....	51, 53	(RBIF Bit).....	15, 24, 62
HS .....	53, 58	RB7:RB4 Port Pins .....	26
LP.....	53, 58	TRISB Register.....	12, 23
RC.....	53, 54, 58		
Selection (FOSC1:FOSC0 Bits).....	52		

PORTC			
TRISC Register.....	12		
Postscaler, Timer2			
Select (TOUTPS3:TOUTPS0 Bits) .....	36		
Postscaler, WDT .....	29		
Assignment (PSA Bit) .....	14, 29		
Block Diagram.....	30		
Rate Select (PS2:PS0 Bits) .....	14, 29		
Switching Between Timer0 and WDT .....	30		
Power-down Mode. <i>See</i> Sleep			
Power-on Reset (POR) .....	51, 54, 55, 58, 59		
Oscillator Start-up Timer (OST) .....	51, 55		
POR Status (POR Bit).....	18		
Power Control (PCON) Register.....	58		
Power-down (PD Bit) .....	13, 54		
Power-on Reset Circuit, External.....	55		
Power-up Timer (PWRT) .....	51, 55		
PWRT Enable (PWRTE Bit).....	52		
Time-out (TO Bit) .....	13, 54		
Time-out Sequence.....	57		
Time-out Sequence on Power-up .....	60		
Timing Diagram.....	83		
Prescaler, Capture .....	40		
Prescaler, Timer0.....	29		
Assignment (PSA Bit) .....	14, 29		
Block Diagram.....	30		
Rate Select (PS2:PS0 Bits) .....	14, 29		
Switching Between Timer0 and WDT .....	30		
Prescaler, Timer1.....	32		
Select (T1CKPS1:T1CKPS0 Bits).....	31		
Prescaler, Timer2.....	42		
Select (T2CKPS1:T2CKPS0 Bits).....	36		
Product Identification System .....	103		
Program Counter			
PCL Register.....	11, 19		
PCLATH Register .....	11, 19, 62		
Reset Conditions.....	58		
Program Memory .....	9		
Interrupt Vector .....	9		
Paging.....	9, 19		
Program Memory Map .....	9		
Reset Vector .....	9		
Program Verification .....	65		
Programming, Device Instructions .....	67		
PWM (CCP Module) .....	42		
Block Diagram.....	42		
CCPR1H:CCPR1L Registers.....	42		
Duty Cycle.....	42		
Example Frequencies/Resolutions .....	43		
Output Diagram.....	42		
Period.....	42		
Set-Up for PWM Operation .....	43		
TMR2 to PR2 Match .....	36, 42		
TMR2 to PR2 Match Enable (TMR2IE Bit) .....	16		
TMR2 to PR2 Match Flag (TMR2IF Bit) .....	17		
<b>Q</b>			
Q-Clock .....	42		
<b>R</b>			
RAM. <i>See</i> Data Memory			
Reader Response .....	104		
Register File.....	10		
Register File Map.....	10		
Reset .....	51, 54		
Block Diagram.....	56		
Brown-out Reset (BOR). <i>See</i> Brown-out Reset (BOR)			
MCLR Reset. <i>See</i> MCLR			
Power-on Reset (POR). <i>See</i> Power-on Reset (POR)			
Reset Conditions for All Registers.....	59		
Reset Conditions for PCON Register .....	58		
Reset Conditions for Program Counter .....	58		
Reset Conditions for STATUS Register .....	58		
Timing Diagram .....	83		
WDT Reset. <i>See</i> Watchdog Timer (WDT)			
Revision History .....	95		
<b>S</b>			
Sleep .....	64		
Sleep .....	51, 54		
Software Simulator (MPLAB SIM) .....	70		
Special Event Trigger. <i>See</i> Compare			
Special Features of the CPU .....	51		
Special Function Registers .....	11		
Speed, Operating .....	19		
Stack.....	1		
STATUS Register .....	11, 13, 62		
C Bit .....	13		
DC Bit .....	13		
IRP Bit .....	13		
PD Bit .....	13, 54		
RP1:RP0 Bits.....	13		
TO Bit .....	13, 54		
Z Bit .....	13		
<b>T</b>			
T1CON Register .....	11, 31		
T1CKPS1:T1CKPS0 Bits.....	31		
T1OSCEN Bit .....	31		
T1SYNC Bit .....	31		
TMR1CS Bit.....	31		
TMR1ON Bit .....	31		
T2CON Register .....	11, 36		
T2CKPS1:T2CKPS0 Bits.....	36		
TMR2ON Bit .....	36		
TOUTPS3:TOUTPS0 Bits .....	36		
Timer0 .....	29		
Block Diagram .....	29		
Clock Source Edge Select (T0SE Bit) .....	14, 29		
Clock Source Select (T0CS Bit) .....	14, 29		
Overflow Enable (T0IE Bit) .....	15		
Overflow Flag (T0IF Bit) .....	15, 62		
Overflow Interrupt .....	30, 62		
Prescaler. <i>See</i> Prescaler, Timer0			
Timing Diagram .....	84		
TMR0 Register .....	11		
Timer1 .....	31		
Block Diagram .....	32		
Capacitor Selection .....	34		
Clock Source Select (TMR1CS Bit).....	31		
External Clock Input Sync (T1SYNC Bit) .....	31		
Module On/Off (TMR1ON Bit) .....	31		
Oscillator.....	31, 34		
Oscillator Enable (T1OSCEN Bit) .....	31		
Overflow Enable (TMR1IE Bit) .....	16		
Overflow Flag (TMR1IF Bit).....	17		
Overflow Interrupt.....	31, 34		
Prescaler. <i>See</i> Prescaler, Timer1			
Special Event Trigger (CCP) .....	34, 41		
T1CON Register .....	11, 31		
Timing Diagram .....	84		
TMR1H Register.....	11, 31		

# PIC16C712/716

---

TMR1L Register .....	11, 31
Timer2	
Block Diagram .....	36
Postscaler. <i>See</i> Postscaler, Timer2	
PR2 Register .....	12, 36, 42
Prescaler. <i>See</i> Prescaler, Timer2	
T2CON Register .....	11, 36
TMR2 Register .....	11, 36
TMR2 to PR2 Match Enable (TMR2IE Bit) .....	16
TMR2 to PR2 Match Flag (TMR2IF Bit) .....	17
TMR2 to PR2 Match Interrupt .....	36, 37, 42
Timing Diagrams	
Time-out Sequence on Power-up .....	60
Wake-up from Sleep via Interrupt .....	65
Timing Diagrams and Specifications .....	81
A/D Conversion .....	87
Brown-out Reset (BOR) .....	83
Capture/Compare/PWM (CCP) .....	85
CLKOUT and I/O .....	82
External Clock .....	81
Oscillator Start-up Timer (OST) .....	83
Power-up Timer (PWRT) .....	83
Reset .....	83
Timer0 and Timer1 .....	84
Watchdog Timer (WDT) .....	83

## W

W Register .....	62
Wake-up from Sleep .....	51
Wake-up from Sleep .....	64
Interrupts .....	58, 59
MCLR Reset .....	59
Timing Diagram .....	65
WDT Reset .....	59
Watchdog Timer (WDT) .....	51, 63
Block Diagram .....	63
Enable (WDTE Bit) .....	52, 63
Postscaler. <i>See</i> Postscaler, WDT	
Programming Considerations .....	63
RC Oscillator .....	63
Time-out Period .....	63
Timing Diagram .....	83
WDT Reset, Normal Operation .....	54, 58, 59
WDT Reset, Sleep .....	54, 58, 59

## PIC16C712/716 PRODUCT IDENTIFICATION SYSTEM

To order or obtain information, e.g., on pricing or delivery, refer to the factory or the listed sales office.

PART NO.	-XX	X	/XX	XXX
Device	Frequency Range	Temperature Range	Package	Pattern
<p><b>Device:</b></p> <p>PIC16C712<sup>(1)</sup>, PIC16C712T<sup>(2)</sup>; VDD range 4.0V to 5.5V            PIC16LC712<sup>(1)</sup>, PIC16LC712T<sup>(2)</sup>; VDD range 2.5V to 5.5V            PIC16C716<sup>(1)</sup>, PIC16C716T<sup>(2)</sup>; VDD range 4.0V to 5.5V            PIC16LC716<sup>(1)</sup>, PIC16LC716T<sup>(2)</sup>; VDD range 2.5V to 5.5V</p> <p><b>Frequency Range:</b></p> <p>04 = 4 MHz            20 = 20 MHz</p> <p><b>Temperature Range:</b></p> <p>blank = 0°C to 70°C (Commercial)            I = -40°C to +85°C (Industrial)            E = -40°C to +125°C (Extended)</p> <p><b>Package:</b></p> <p>JW = Windowed CERDIP            SO = SOIC            P = PDIP            SS = SSOP</p> <p><b>Pattern:</b></p> <p>QTP, SQTP, Code or Special Requirements            (blank otherwise)</p>	<p><b>Examples:</b></p> <p>a) PIC16C716 – 04/P 301 = Commercial temp., PDIP package, 4 MHz, normal VDD limits, QTP pattern #301.</p> <p>b) PIC16LC712 – 04/SO = Industrial temp., SOIC package, 200 kHz, Extended VDD limits.</p> <p>c) PIC16C712 – 20I/P = Industrial temp., PDIP package, 20MHz, normal VDD limits.</p> <p><b>Note 1:</b> C = CMOS            LC = Low Power CMOS</p> <p><b>2:</b> T = in tape and reel – SOIC, SSOP packages only.</p> <p><b>3:</b> LC extended temperature device is not offered.</p> <p><b>4:</b> LC is not offered at 20 MHz</p>			

\* JW Devices are UV erasable and can be programmed to any device configuration. JW Devices meet the electrical requirement of each oscillator type (including LC devices).