

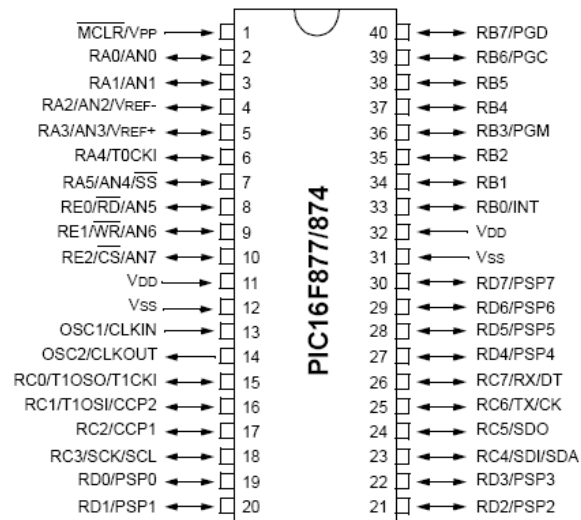
# MICROCONTROLADOR PIC DE MICROCHIP

## PIC16F877 - 28-PIN 8-BIT CMOS FLASH MICROCONTROLLER

### 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
- Up to 8K x 14 words of FLASH Program Memory,  
Up to 368 x 8 bytes of Data Memory (RAM)  
Up to 256 x 8 bytes of EEPROM Data Memory
- Pinout compatible to the PIC16C73B/74B/76/77
- Interrupt capability (up to 14 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
- Programmable code protection
- Power saving SLEEP mode
- Selectable oscillator options
- Low power, high speed CMOS FLASH/EEPROM  
technology
- Fully static design
- In-Circuit Serial Programming™ (ICSP) via two  
pins
- Single 5V In-Circuit Serial Programming capability
- In-Circuit Debugging via two pins
- Processor read/write access to program memory
- Wide operating voltage range: 2.0V to 5.5V
- High Sink/Source Current: 25 mA
- Commercial, Industrial and Extended temperature  
ranges
- Low-power consumption:
  - < 0.6 mA typical @ 3V, 4 MHz
  - 20 µA typical @ 3V, 32 kHz
  - < 1 µA typical standby current

### PDIP



### Peripheral Features:

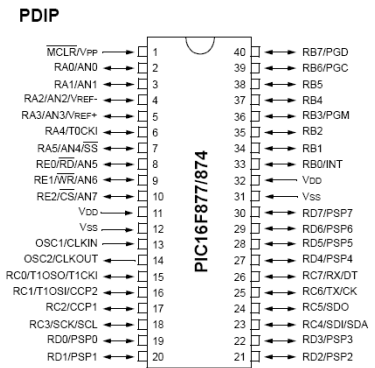
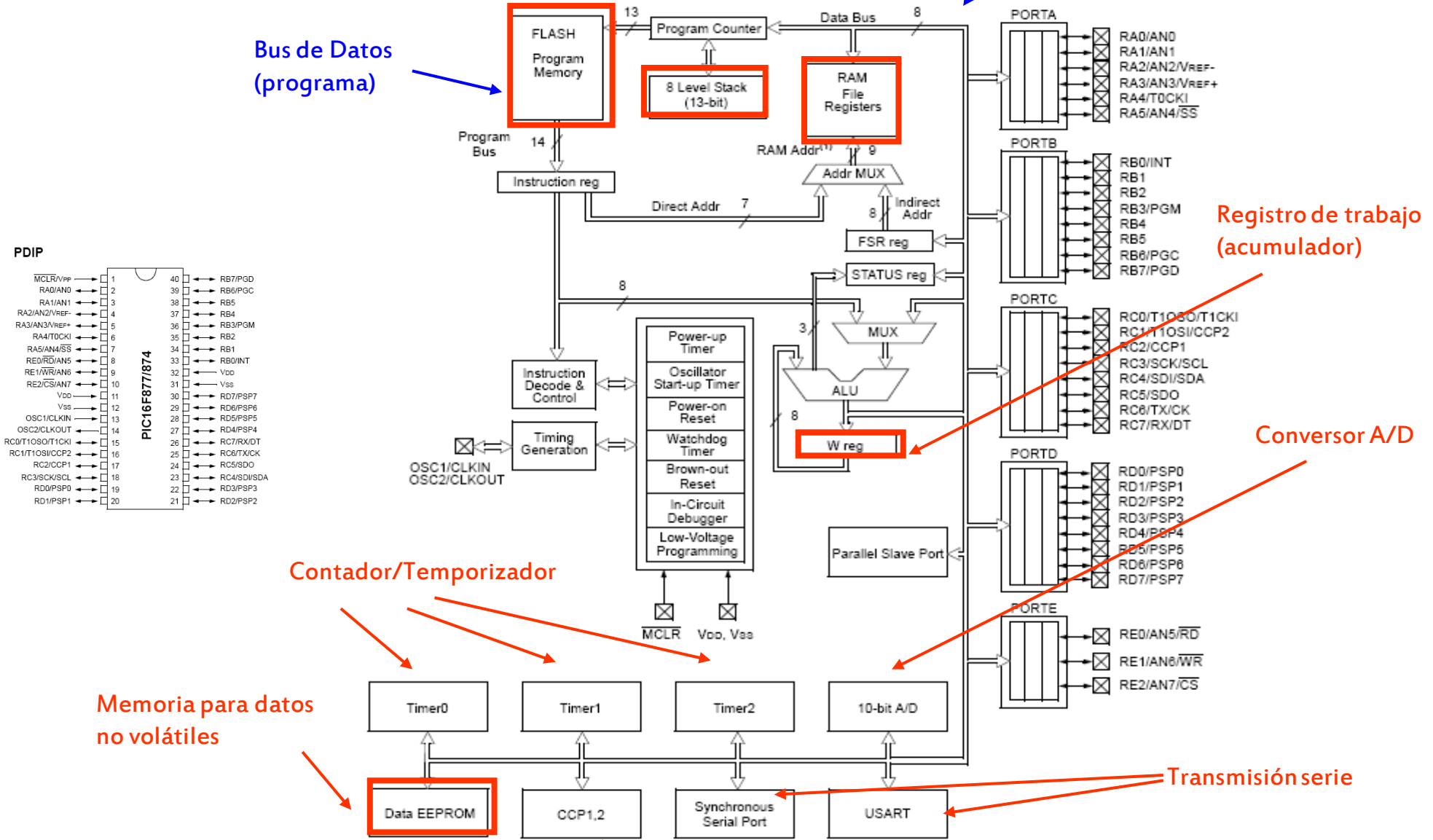
- Timer0: 8-bit timer/counter with 8-bit prescaler
- Timer1: 16-bit timer/counter with prescaler,  
can be incremented during SLEEP via external  
crystal/clock
- Timer2: 8-bit timer/counter with 8-bit period  
register, prescaler and postscaler
- Two Capture, Compare, PWM modules
  - Capture is 16-bit, max. resolution is 12.5 ns
  - Compare is 16-bit, max. resolution is 200 ns
  - PWM max. resolution is 10-bit
- 10-bit multi-channel Analog-to-Digital converter
- Synchronous Serial Port (SSP) with SPI™ (Master  
mode) and I<sup>2</sup>C™ (Master/Slave)
- Universal Synchronous Asynchronous Receiver  
Transmitter (USART/SCI) with 9-bit address  
detection
- Parallel Slave Port (PSP) 8-bits wide, with  
external RD, WR and CS controls (40/44-pin only)
- Brown-out detection circuitry for  
Brown-out Reset (BOR)

### PIC16CXXX/PIC16FXXX Family: 14-bit program word

With the introduction of new PIC16CXXX/PIC16FXXX family members, Microchip now provides the industry's highest performance Analog-to-Digital Converter capability at 12-bits for an MCU. The PIC16CXXX/PIC16FXXX family offers a wide-range of options, from 18- to 68-pin packages as well as low to highest levels of peripheral integration. This family has a 14-bit wide instruction set, interrupt handling capability and a deep, 8-level hardware stack. The PIC16CXXX/PIC16FXXX family provides the performance and versatility to meet the more demanding requirements of today's cost-sensitive marketplace for mid-range applications.

# PIC16F877: ESTRUCTURA INTERNA

Device	Program FLASH	Data Memory	Data EEPROM
PIC16F874	4K	192 Bytes	128 Bytes
PIC16F877	8K	368 Bytes	256 Bytes



## ***PERIFERICOS DENTRO DEL MICROCONTROLADOR***

- PUERTOS (PORT)
- TEMPORIZADORES/CONTADORES (TIMER/COUNTER)
- CONVERSORA/D
- LINEA SERIE (ASÍNCRONAS Y SÍNCRONAS)
- MEMORIA EEPROM DE DATOS

FIGURE 3-1: BLOCK DIAGRAM OF RA3:RA0 AND RA5 PINS

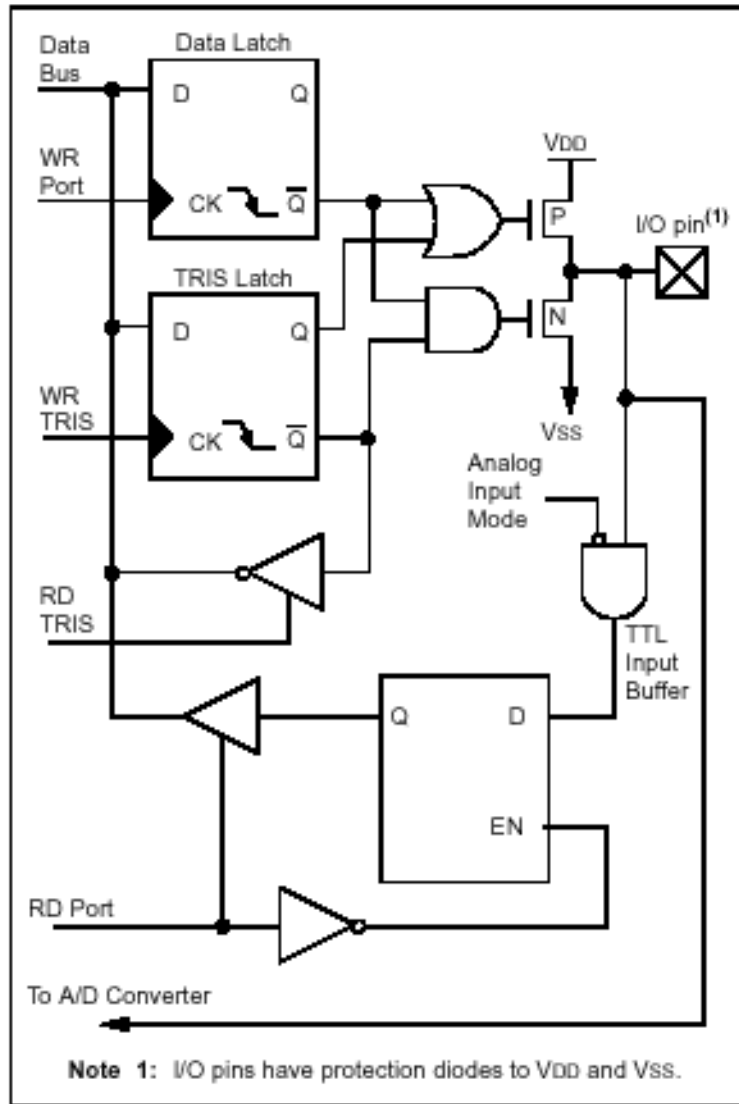


FIGURE 3-2: BLOCK DIAGRAM OF RA4/T0CKI PIN

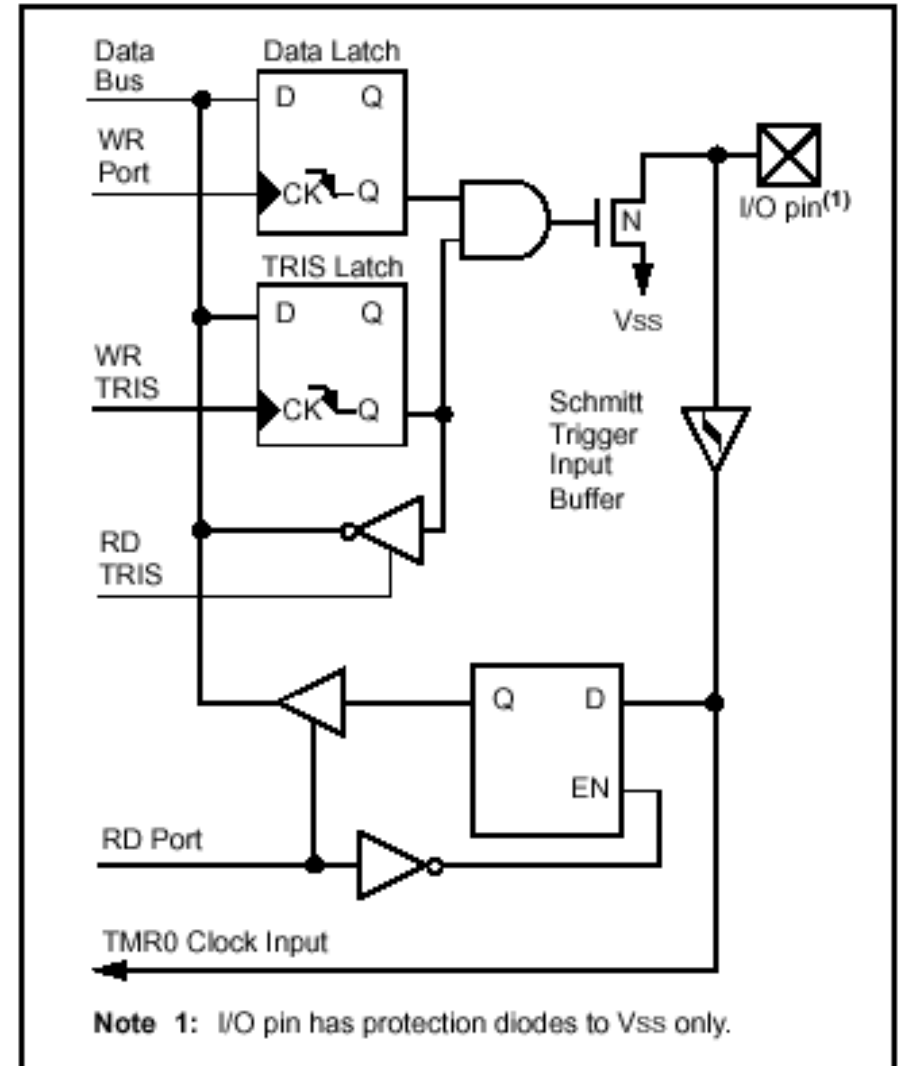


TABLE 3-1: PORTA FUNCTIONS

Name	Bit#	Buffer	Function
RA0/AN0	bit0	TTL	Input/output or analog input.
RA1/AN1	bit1	TTL	Input/output or analog input.
RA2/AN2	bit2	TTL	Input/output or analog input.
RA3/AN3/VREF	bit3	TTL	Input/output or analog input or VREF.
RA4/T0CKI	bit4	ST	Input/output or external clock input for Timer0. Output is open drain type.
RA5/SS/AN4	bit5	TTL	Input/output or slave select input for synchronous serial port or analog input.

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	—	—	RA5	RA4	RA3	RA2	RA1	RA0	--0x 0000	--0u 0000
85h	TRISA	—	—	PORTA Data Direction Register						--11 1111	--11 1111
9Fh	ADCON1	ADFM	—	—	—	PCFG3	PCFG2	PCFG1	PCFG0	--0- 0000	--0- 0000

Legend: x = unknown, u = unchanged, - = unimplemented locations read as '0'.

Shaded cells are not used by PORTA.

**REGISTRO TRISA**  
"1" ....ENTRADA  
"0" ....SALIDA

**NOTAIMPORTANTE:** Los bits del PORT A están configurados como entradas analógicas por defecto.  
Mirar el registro ADCON1 del conversor A/D.

REGISTER 11-2: **ADCON1 REGISTER (ADDRESS 9Fh)**

U-0	U-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
ADFM	—	—	—	PCFG3	PCFG2	PCFG1	PCFG0
bit 7				bit 0			

bit 7 **ADFM:** A/D Result Format Select bit  
 1 = Right justified. 6 Most Significant bits of ADRESH are read as '0'.  
 0 = Left justified. 6 Least Significant bits of ADRESL are read as '0'.

bit 6-4 **Unimplemented:** Read as '0'

bit 3-0 **PCFG3:PCFG0:** A/D Port Configuration Control bits:

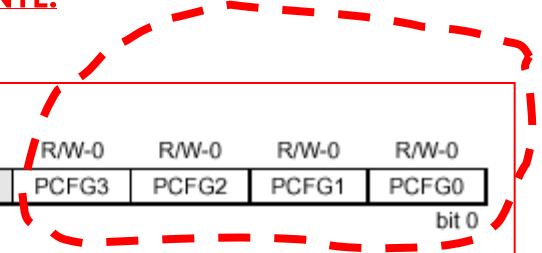
PCFG3: PCFG0	AN7 <sup>(1)</sup> RE2	AN6 <sup>(1)</sup> RE1	AN5 <sup>(1)</sup> RE0	AN4 RA5	AN3 RA3	AN2 RA2	AN1 RA1	AN0 RA0	VREF+	VREF-	CHAN/ Refs <sup>(2)</sup>
0000	A	A	A	A	A	A	A	A	VDD	VSS	8/0
0001	A	A	A	A	VREF+	A	A	A	RA3	VSS	7/1
0010	D	D	D	A	A	A	A	A	VDD	VSS	5/0
0011	D	D	D	A	VREF+	A	A	A	RA3	VSS	4/1
0100	D	D	D	D	A	D	A	A	VDD	VSS	3/0
0101	D	D	D	D	VREF+	D	A	A	RA3	VSS	2/1
011x	D	D	D	D	D	D	D	D	VDD	VSS	0/0
1000	A	A	A	A	VREF+	VREF-	A	A	RA3	RA2	6/2
1001	D	D	A	A	A	A	A	A	VDD	VSS	6/0
1010	D	D	A	A	VREF+	A	A	A	RA3	VSS	5/1
1011	D	D	A	A	VREF+	VREF-	A	A	RA3	RA2	4/2
1100	D	D	D	A	VREF+	VREF-	A	A	RA3	RA2	3/2
1101	D	D	D	D	VREF+	VREF-	A	A	RA3	RA2	2/2
1110	D	D	D	D	D	D	D	A	VDD	VSS	1/0
1111	D	D	D	D	VREF+	VREF-	D	A	RA3	RA2	1/2

A = Analog input D = Digital I/O

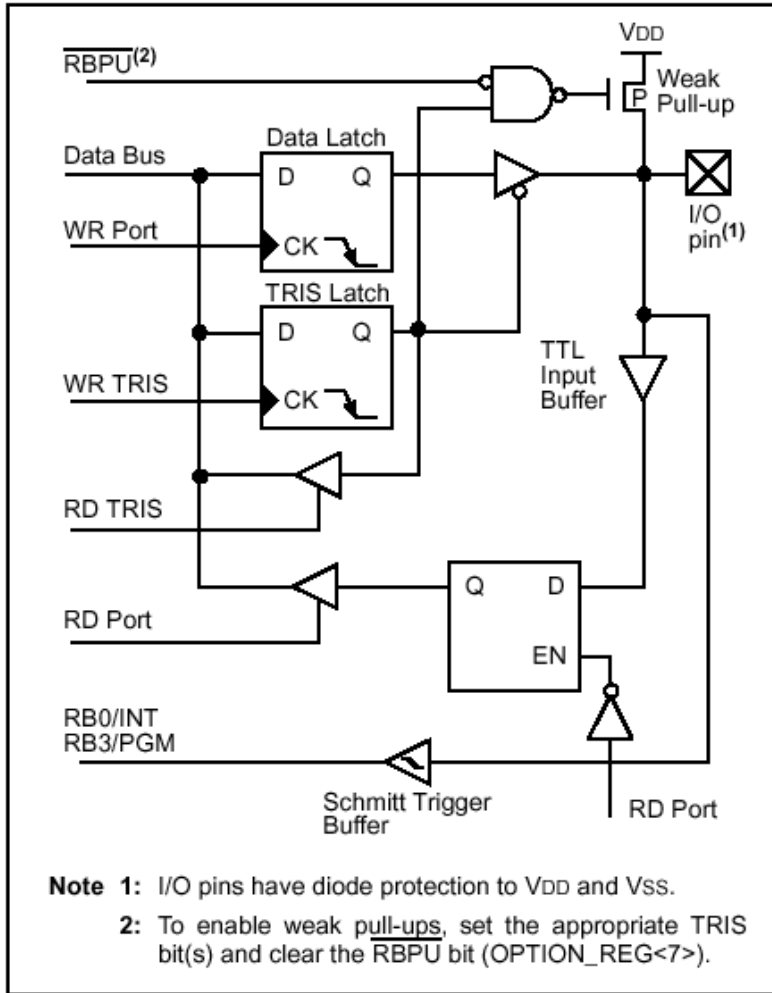
```
BSF STATUS,5
MVLW D'00000110'
MOVWF ADCON1
BCF STATUS,5
```

```
BSF STATUS,5
MVLW D'00001110'
MOVWF ADCON1
BCF STATUS,5
```

Por defecto



**FIGURE 3-3: BLOCK DIAGRAM OF RB3:RB0 PINS**



**FIGURE 3-4: BLOCK DIAGRAM OF RB7:RB4 PINS**

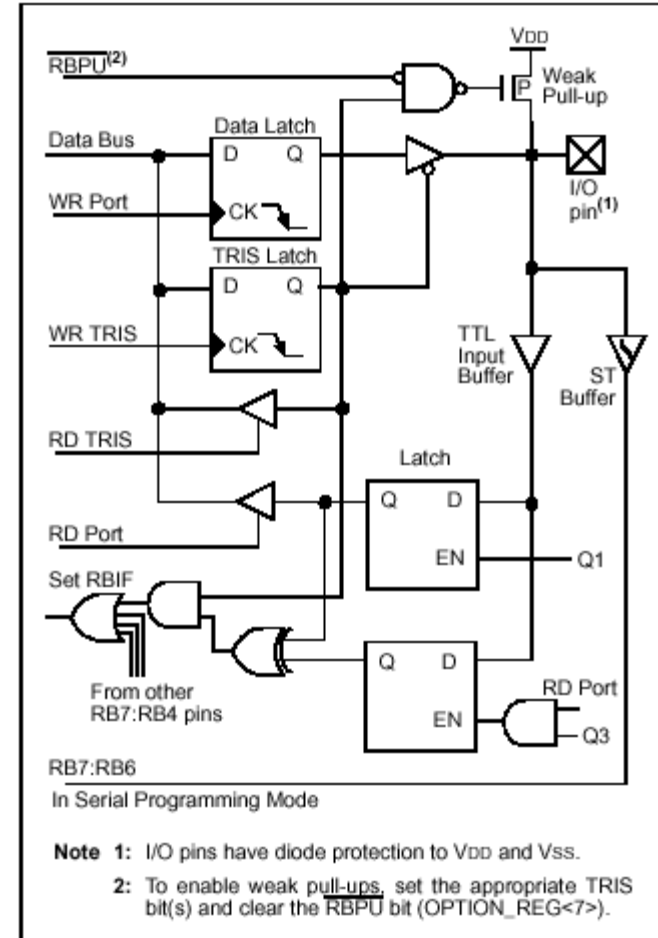


TABLE 3-3: PORTB FUNCTIONS

Name	Bit#	Buffer	Function
RB0/INT	bit0	TTL/ST <sup>(1)</sup>	Input/output pin or external interrupt input. Internal software programmable weak pull-up.
RB1	bit1	TTL	Input/output pin. Internal software programmable weak pull-up.
RB2	bit2	TTL	Input/output pin. Internal software programmable weak pull-up.
RB3/PGM <sup>(3)</sup>	bit3	TTL	Input/output pin or programming pin in LVP mode. Internal software programmable weak pull-up.
RB4	bit4	TTL	Input/output pin (with interrupt-on-change). Internal software programmable weak pull-up.
RB5	bit5	TTL	Input/output pin (with interrupt-on-change). Internal software programmable weak pull-up.
RB6/PGC	bit6	TTL/ST <sup>(2)</sup>	Input/output pin (with interrupt-on-change) or In-Circuit Debugger pin. Internal software programmable weak pull-up. Serial programming clock.
RB7/PGD	bit7	TTL/ST <sup>(2)</sup>	Input/output pin (with interrupt-on-change) or In-Circuit Debugger pin. 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.

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

**Note 3:** Low Voltage ICSP Programming (LVP) is enabled by default, which disables the RB3 I/O function. LVP must be disabled to enable RB3 as an I/O pin and allow maximum compatibility to the other 28-pin and 40-pin mid-range devices.



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, 106h	PORTB	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	xxxx xxxx	uuuu uuuu
86h, 186h	TRISB	PORTB Data Direction Register								1111 1111	1111 1111
81h, 181h	OPTION_REG	RBPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111

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

**REGISTRO TRISB**  
"1"....ENTRADA  
"0"....SALIDA

Permite poner o quitar las resistencias de PULL-UP (por defecto están quitadas)

**NOTA:**

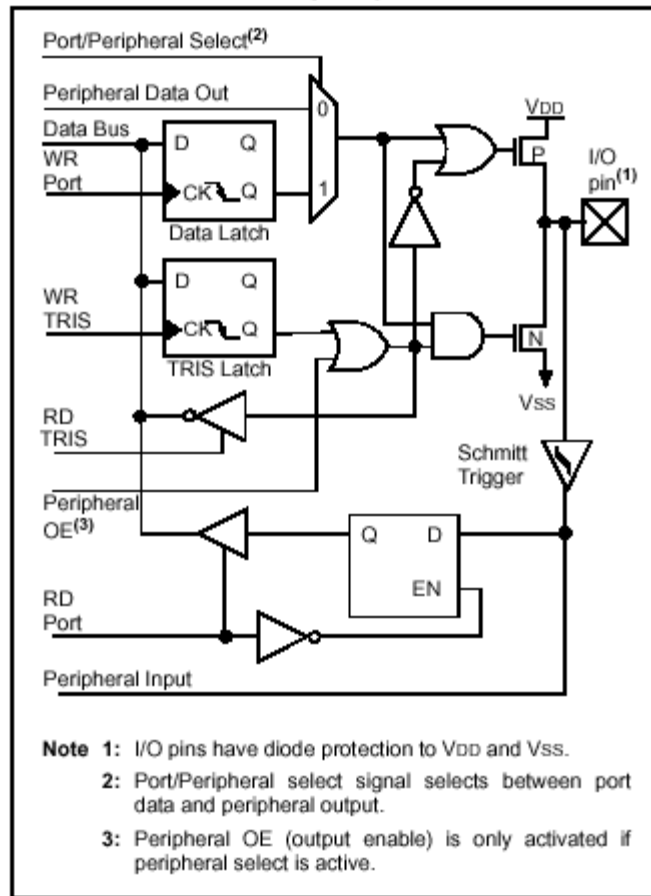
Los pines RB4 a RB7 pueden ser empleados para generar interrupciones.

Si el bit RBIE (bit 3 del registro INTCON) se encuentra a '1' (interrupciones del puerto B habilitadas), en el momento en que el valor en cualquiera de estos pines que se encuentre configurado como entrada difiera del último valor leído, se produce una interrupción y se activa el bit RBIF.

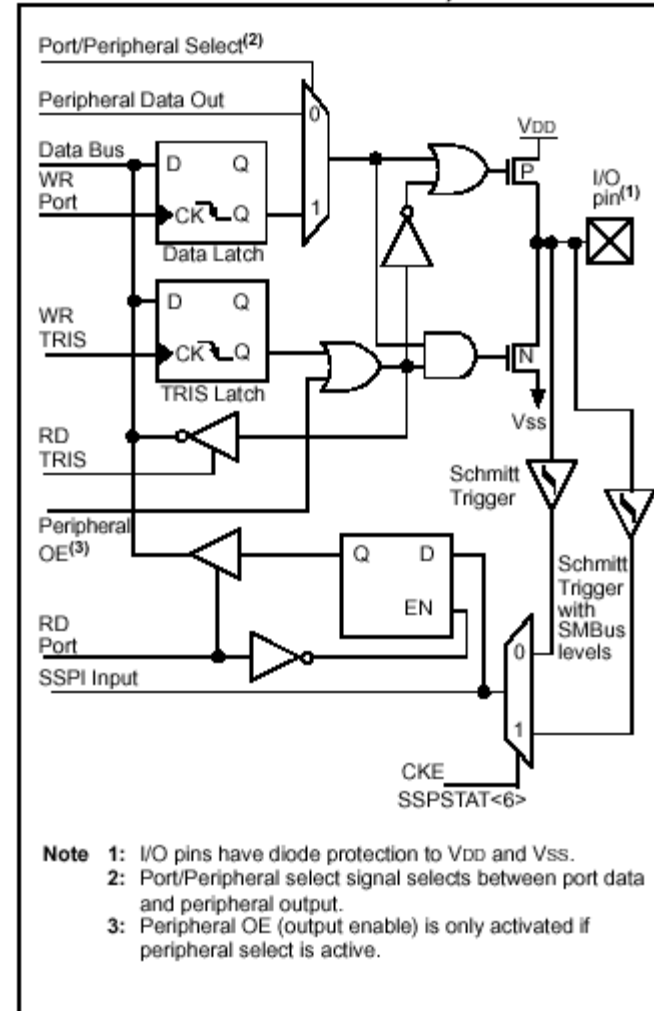
Para eliminar la condición de interrupciones necesario realizar una operación de lectura sobre PORTB.

**PIC16F876:  
PUERTO C**

**FIGURE 3-5: PORTC BLOCK DIAGRAM (PERIPHERAL OUTPUT OVERRIDE) RC<2:0>, RC<7:5>**



**FIGURE 3-6: PORTC BLOCK DIAGRAM (PERIPHERAL OUTPUT OVERRIDE) RC<4:3>**



**TABLE 3-5: PORTC FUNCTIONS**

Name	Bit#	Buffer Type	Function
RC0/T1OSO/T1CKI	bit0	ST	Input/output port pin or Timer1 oscillator output/Timer1 clock input.
RC1/T1OSI/CCP2	bit1	ST	Input/output port pin or Timer1 oscillator input or Capture2 input/Compare2 output/PWM2 output.
RC2/CCP1	bit2	ST	Input/output port pin or Capture1 input/Compare1 output/PWM1 output.
RC3/SCK/SCL	bit3	ST	RC3 can also be the synchronous serial clock for both SPI and I <sup>2</sup> C modes.
RC4/SDI/SDA	bit4	ST	RC4 can also be the SPI Data In (SPI mode) or data I/O (I <sup>2</sup> C mode).
RC5/SDO	bit5	ST	Input/output port pin or Synchronous Serial Port data output.
RC6/TX/CK	bit6	ST	Input/output port pin or USART Asynchronous Transmit or Synchronous Clock.
RC7/RX/DT	bit7	ST	Input/output port pin or USART Asynchronous Receive or Synchronous Data.

Legend: ST = Schmitt Trigger input

TABLE 3-6: SUMMARY OF REGISTERS ASSOCIATED WITH PORTC

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	PORTC	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0	xxxx xxxx	uuuu uuuu
87h	TRISC	PORTC Data Direction Register								1111 1111	1111 1111

Legend: x = unknown, u = unchanged

**REGISTRO TRISB**  
**“1”....ENTRADA**  
**“0”....SALIDA**

**NOTA:**

**Cada bit del port C puede tener varias funciones ligadas con la transmisión serie, el temporizado 1 o las funciones especiales capture, por defecto están desactivadas.**

# PIC16F876: REGISTROS DE PUERTOS

ACCESO Y CONTROL DE PUERTOS



FIGURE 2-3: PIC16F877/876 REGISTER FILE MAP

File Address	File Address	File Address	File Address
Indirect addr. <sup>(1)</sup> 00h	Indirect addr. <sup>(1)</sup> 80h	Indirect addr. <sup>(1)</sup> 100h	Indirect addr. <sup>(1)</sup> 180h
TMR0 01h	OPTION_REG 81h	TMR0 101h	OPTION_REG 181h
PCL 02h	PCL 82h	PCL 102h	PCL 182h
STATUS 03h	STATUS 83h	STATUS 103h	STATUS 183h
FSR 04h	FSR 84h	FSR 104h	FSR 184h
PORTA 05h	TRISA 85h		
PORTB 06h	TRISB 86h	PORTB 106h	TRISB 186h
PORTC 07h	TRISC 87h		
<del>PORTD 08h</del>	<del>TRISD 88h</del>		
<del>PORTEN 09h</del>	<del>TRISE 89h</del>		
PCLATH 0Ah	PCLATH 8Ah	PCLATH 10Ah	PCLATH 18Ah
INTCON 0Bh	INTCON 8Bh	INTCON 10Bh	INTCON 18Bh
PIR1 0Ch	PIE1 8Ch	EEDATA 10Ch	EECON1 18Ch
PIR2 0Dh	PIE2 8Dh	EEADR 10Dh	EECON2 18Dh
TMR1L 0Eh	PCON 8Eh	EEDATH 10Eh	Reserved <sup>(2)</sup> 18Eh
TMR1H 0Fh		EEADRH 10Fh	Reserved <sup>(2)</sup> 18Fh
T1CON 10h			
TMR2 11h	SSPCON2 91h		
T2CON 12h	PR2 92h		
SSPBUF 13h	SSPAD0 93h		
SSPCON 14h	SSPSTAT 94h		
CCPR1L 15h			
CCPR1H 16h			
CCP1CON 17h		General Purpose Register 16 Bytes 117h	General Purpose Register 16 Bytes 197h
RCSTA 18h	TXSTA 98h		
TXREG 19h	SPBRG 99h		
RCREG 1Ah			
CCPR2L 1Bh			
CCPR2H 1Ch			
CCP2CON 1Dh			
ADRESH 1Eh	ADRESL 9Eh		
ADCON0 1Fh	ADCON1 9Fh		
General Purpose Register 96 Bytes 20h	General Purpose Register 80 Bytes A0h	General Purpose Register 80 Bytes 120h	General Purpose Register 80 Bytes 1A0h
	accesses 70h-7Fh EFh	accesses 70h-7Fh 16Fh	accesses 70h-7Fh 1EFh
Bank 0 7Fh	Bank 1 FFh	Bank 2 17Fh	Bank 3 1FFh

■ Unimplemented data memory locations, read as '0'.  
\* Not a physical register.

Note 1: These registers are not implemented on the PIC16F876.  
2: These registers are reserved, maintain these registers clear.

## 15.0 ELECTRICAL CHARACTERISTICS

### Absolute Maximum Ratings †

Ambient temperature under bias .....	-55 to +125°C
Storage temperature .....	-65°C to +150°C
Voltage on any pin with respect to VSS (except VDD, $\overline{\text{MCLR}}$ , and RA4) .....	-0.3 V to (VDD + 0.3 V)
Voltage on VDD with respect to VSS .....	-0.3 to +7.5 V
Voltage on $\overline{\text{MCLR}}$ with respect to VSS (Note 2) .....	0 to +14 V
Voltage on RA4 with respect to VSS .....	0 to +8.5 V
Total power dissipation (Note 1) .....	1.0 W
Maximum current out of VSS pin .....	300 mA
Maximum current into VDD pin .....	250 mA
Input clamp current, I <sub>IK</sub> (V <sub>I</sub> < 0 or V <sub>I</sub> > VDD) .....	± 20 mA
Output clamp current, I <sub>OK</sub> (V <sub>O</sub> < 0 or V <sub>O</sub> > VDD) .....	± 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, PORTB, and PORTE (combined) (Note 3) .....	200 mA
Maximum current sourced by PORTA, PORTB, and PORTE (combined) (Note 3) .....	200 mA
Maximum current sunk by PORTC and PORTD (combined) (Note 3) .....	200 mA
Maximum current sourced by PORTC and PORTD (combined) (Note 3) .....	200 mA

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

2: Voltage spikes below VSS at the  $\overline{\text{MCLR}}$  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  $\overline{\text{MCLR}}$  pin, rather than pulling this pin directly to VSS.

3: PORTD and PORTE are not implemented on PIC16F873/876 devices.

† 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.

# TEMPORIZADORES

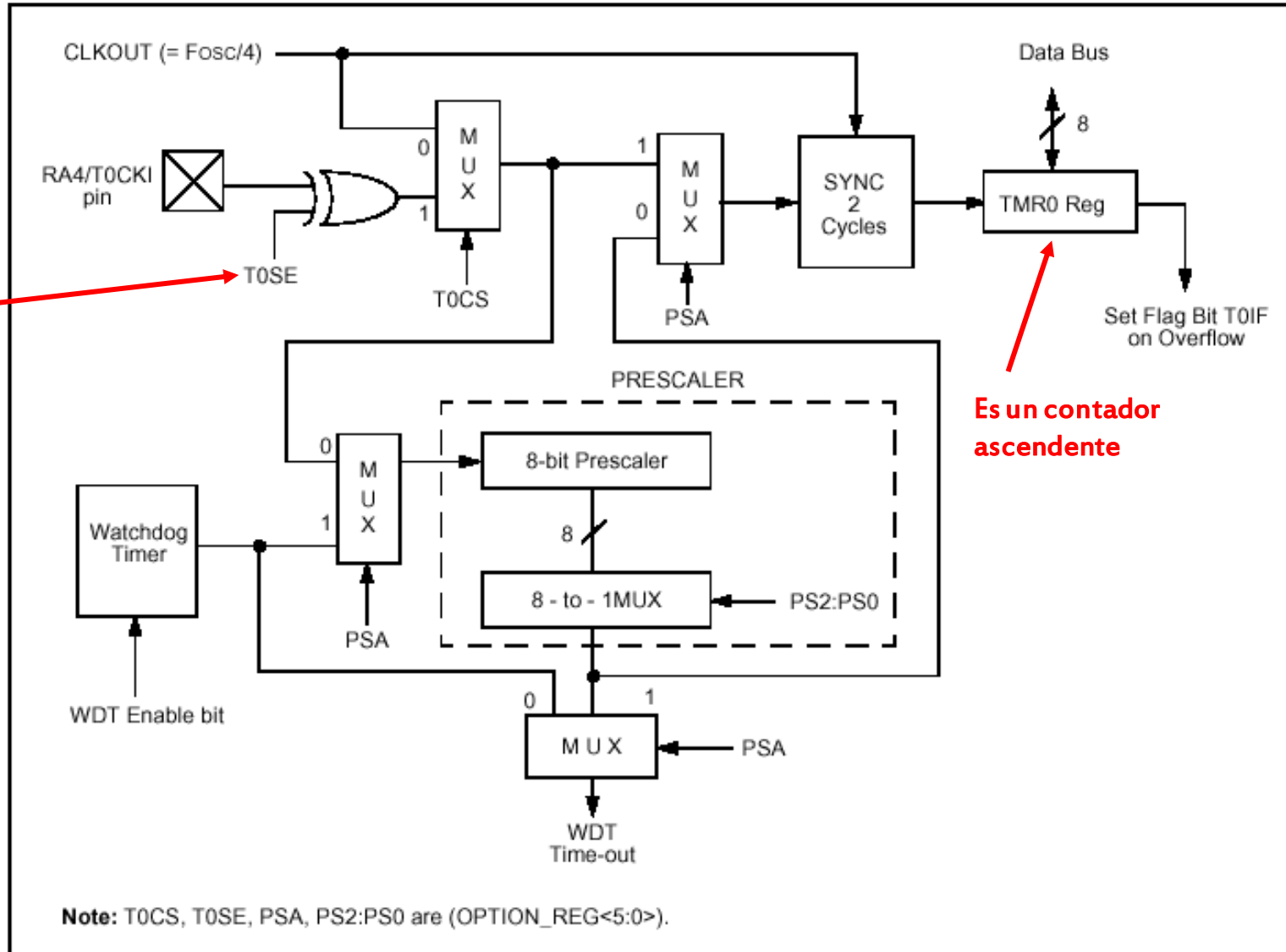
# PIC16F876: TEMPORIZADORES: TIMER0

Si  $F_{osc} = 4 \text{ MHz}$  ( $T = 250 \text{ nS}$ ) tenemos  $CLKOUT = 250 \text{ KHz}$  ( $1 \mu\text{S}$ )

The Timer0 module timer/counter has the following features:

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

FIGURE 5-1: BLOCK DIAGRAM OF THE TIMER0/WDT PRESCALER



TOSE=1



TOSE=0



Es un contador  
ascendente

NOTA:

Si TOSE = 1  
es un inversor

Si TOSE = 0  
no invierte

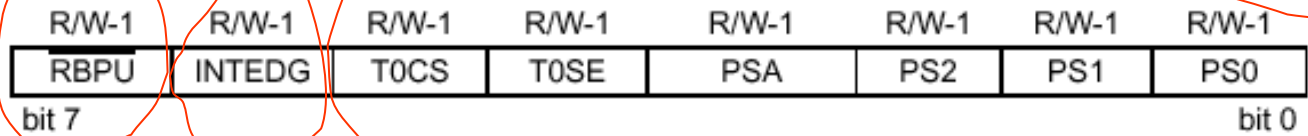


**PIC16F876:**  
**TEMPORIZADORES: TIMER0**

“ 6 bits para controlar el temporizador TMR0 y el temporizador de vigilancia WDT, 1 bit para configurar las interrupciones producidas por los flancos en el pin RB0/INT y 1 bit para habilitar las resistencias de “PULL-UP” en el puerto B”

**REGISTER 5-1: OPTION REG REGISTER**

Resistencias de PULL-UP del PORT B



- bit 7 **RBPU**
- bit 6 **INTEDG**
- bit 5 **T0CS**: TMR0 Clock Source Select bit  
 1 = Transition on T0CKI pin  
 0 = Internal instruction cycle clock (CLKOUT)
- bit 4 **T0SE**: TMR0 Source Edge Select bit  
 1 = Increment on high-to-low transition on T0CKI pin  
 0 = Increment on low-to-high transition on T0CKI pin
- bit 3 **PSA**: Prescaler Assignment bit  
 1 = Prescaler is assigned to the WDT  
 0 = Prescaler is assigned to the Timer0 module
- bit 2-0 **PS2:PS0**: Prescaler Rate Select bits

**Nota:**  
Lo máximo que se puede temporizar vale  $256 \times 256 \mu S = 65,5 \text{ mS}$

Bit Value	TMR0 Rate	WDT Rate
000	1 : 2	1 : 1
001	1 : 4	1 : 2
010	1 : 8	1 : 4
011	1 : 16	1 : 8
100	1 : 32	1 : 16
101	1 : 64	1 : 32
110	1 : 128	1 : 64
111	1 : 256	1 : 128

$T_{RELOJ} = 256 \mu S$



**PIC16F876:  
TEMPORIZADORES: TIMER0**

**Proceso para utilizar el TIMER0 como temporizador.**

- 1.- Configurar adecuadamente**
- 2.- Cargar TMR0 con el valor adecuado**
- 3.- Esperar a que se ponga a 1 el bit TOIF (bit 2 de INTCON)**
- 4.- Borrar este bit**

## Registro de interrupciones: INTCON

“Habilitar las distintas interrupciones y identificar la fuente que las ha producido”

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-x
GIE	PEIE	TOIE	INTE	RBIE	TOIF	INTF	RBIF
bit7							bit0
<p>bit 7: <b>GIE</b>:(<sup>1</sup>) Global Interrupt Enable bit 1 = Enables all un-masked interrupts 0 = Disables all interrupts</p> <p>bit 6: <b>PEIE</b>:(<sup>2</sup>) Peripheral Interrupt Enable bit 1 = Enables all un-masked peripheral interrupts 0 = Disables all peripheral interrupts</p> <p>bit 5: <b>TOIE</b>: TMR0 Overflow Interrupt Enable bit 1 = Enables the TMR0 overflow interrupt 0 = Disables the TMR0 overflow interrupt</p> <p>bit 4: <b>INTE</b>: RB0/INT External Interrupt Enable bit 1 = Enables the RB0/INT external interrupt 0 = Disables the RB0/INT external interrupt</p> <p>bit 3: <b>RBIE</b>: RB Port Change Interrupt Enable bit 1 = Enables the RB port change interrupt 0 = Disables the RB port change interrupt</p> <p>bit 2: <b>TOIF</b>: TMR0 Overflow Interrupt Flag bit 1 = TMR0 register overflowed (must be cleared in software) 0 = TMR0 register did not overflow</p> <p>bit 1: <b>INTF</b>: 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</p> <p>bit 0: <b>RBIF</b>: RB Port Change Interrupt Flag bit 1 = At least one of the RB7:RB4 pins changed state (see Section 5.2 to clear the interrupt) 0 = None of the RB7:RB4 pins have changed state</p> <p>Note 1: For the PIC16C61/62/64/65, if an interrupt occurs while the GIE bit is being cleared, the GIE bit may unintentionally be re-enabled by the <code>RETFIE</code> instruction in the user's Interrupt Service Routine. Refer to Section 13.5 for a detailed description.</p> <p>2: The PEIE bit (bit6) is unimplemented on the PIC16C61, read as '0'.</p>							
<p>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&lt;7&gt;). User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.</p>							

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

**PIC16F876:  
TEMPORIZADORES: TIMER0**

**TABLE 5-1: 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,101h	TMR0	Timer0 Module's Register								xxxx xxxx	uuuu uuuu
0Bh,8Bh, 10Bh,18Bh	INTCON	GIE	PEIE	TOIE	INTE	RBIE	TOIF	INTF	RBIF	0000 000x	0000 000u
81h,181h	OPTION_REG	RBPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111

# PIC16F876: TEMPORIZADORES: TIMER0

UNA PARTE DE INTCON Y DE OPTION  
ESTA LIGADA CON LA GESTIÓN DEL  
TEMPORIZADOR TIMER0

FIGURE 2-3: PIC16F877/876 REGISTER FILE MAP

File Address	File Address	File Address	File Address
Indirect addr. <sup>(1)</sup> 00h	Indirect addr. <sup>(1)</sup> 80h	Indirect addr. <sup>(1)</sup> 100h	Indirect addr. <sup>(1)</sup> 180h
TMR0 01h	OPTION_REG 81h	TMR0 101h	OPTION_REG 181h
PCL 02h	PCL 82h	PCL 102h	PCL 182h
STATUS 03h	STATUS 83h	STATUS 103h	STATUS 183h
FSR 04h	FSR 84h	FSR 104h	FSR 184h
PORTA 05h	TRISA 85h		
PORTB 06h	TRISB 86h	PORTB 106h	TRISB 186h
PORTC 07h	TRISC 87h		
<del>PORTD 08h</del>	<del>TRISD 88h</del>		
<del>PORTF 09h</del>	<del>TRISF 89h</del>		
PCLATH 0Ah	PCLATH 8Ah	PCLATH 10Ah	PCLATH 18Ah
INTCON 0Bh	INTCON 8Bh	INTCON 10Bh	INTCON 18Bh
PIR1 0Ch	PIE1 8Ch	EEDATA 10Ch	EECON1 18Ch
PIR2 0Dh	PIE2 8Dh	EEADR 10Dh	EECON2 18Dh
TMR1L 0Eh	PCON 8Eh	EEDATH 10Eh	Reserved <sup>(2)</sup> 18Eh
TMR1H 0Fh		EEADRH 10Fh	Reserved <sup>(2)</sup> 18Fh
T1CON 10h			
TMR2 11h	SSPCON2 91h		
T2CON 12h	PR2 92h		
SSPBUF 13h	SSPAD0 93h		
SSPCON 14h	SSPSTAT 94h		
CCPR1L 15h			
CCPR1H 16h			
CCP1CON 17h			
RCSTA 18h	TXSTA 98h	General Purpose Register 16 Bytes 117h	General Purpose Register 16 Bytes 197h
TXREG 19h	SPBRG 99h		
RCON 1Ah			
CCPR2L 1Bh			
CCPR2H 1Ch			
CCP2CON 1Dh			
ADRESH 1Eh	ADRESL 9Eh		
ADCON0 1Fh	ADCON1 9Fh		
General Purpose Register 96 Bytes 20h	General Purpose Register 80 Bytes A0h	General Purpose Register 80 Bytes 120h	General Purpose Register 80 Bytes 1A0h
	accesses 70h-7Fh EFh	accesses 70h-7Fh 16Fh	accesses 70h-7Fh 1EFh
Bank 0 7Fh	Bank 1 FFh	Bank 2 17Fh	Bank 3 1FFh

■ Unimplemented data memory locations, read as '0'.  
\* Not a physical register.

Note 1: These registers are not implemented on the PIC16F876.  
2: These registers are reserved, maintain these registers clear.

PIC16F876:  
TEMPORIZADORES: TIMER1

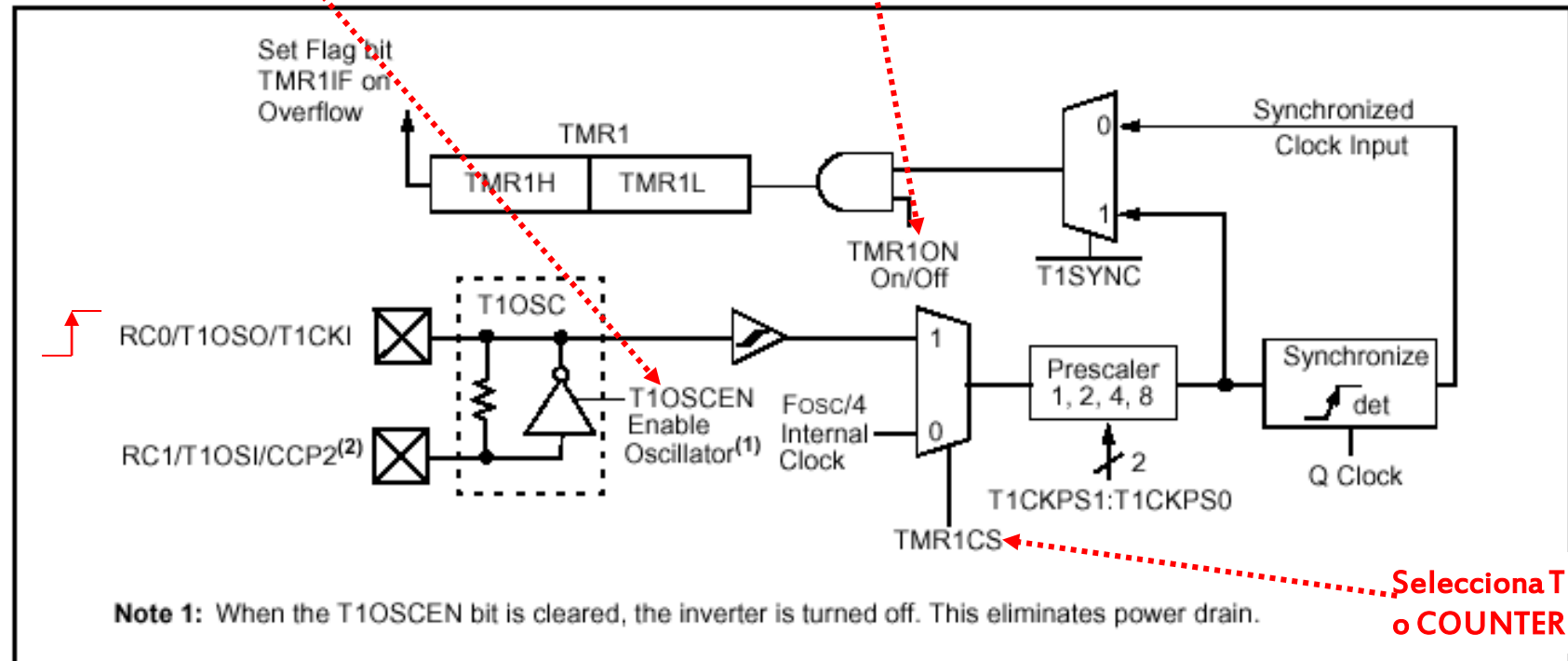
En modo TIMER incrementa cada ciclo de instrucción ( $F_{osc}/4$ ).

En modo COUNTER incrementa cada flanco de subida externo.

Activa oscilador externo (con 1)

Desactivación.  
Por defecto está desactivado

FIGURE 6-2: TIMER1 BLOCK DIAGRAM



Como TIMER: El máximo tiempo es:  $2^{16} \times 8 \times 1\mu S = 0.52 S$

**PIC16F876:  
TEMPORIZADORES: TIMER1**

**REGISTER 6-1: T1CON: TIMER1 CONTROL REGISTER (ADDRESS 10h)**

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	
—	—	T1CKPS1	T1CKPS0	T1OSCEN	T1SYNC	TMR1CS	TMR1ON	
bit 7								bit 0

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

bit 5-4 **T1CKPS1:T1CKPS0:** Timer1 Input Clock Prescale Select bits

- 11 = 1:8 Prescale value
- 10 = 1:4 Prescale value
- 01 = 1:2 Prescale value
- 00 = 1:1 Prescale value

bit 3 **T1OSCEN:** Timer1 Oscillator Enable Control bit

- 1 = Oscillator is enabled
- 0 = Oscillator is shut-off (the oscillator inverter is turned off to eliminate power drain)

bit 2 **T1SYNC:** Timer1 External Clock Input Synchronization Control bit

When TMR1CS = 1:

- 1 = Do not synchronize external clock input
- 0 = Synchronize external clock input

When TMR1CS = 0:

This bit is ignored. Timer1 uses the internal clock when TMR1CS = 0.

bit 1 **TMR1CS:** Timer1 Clock Source Select bit

- 1 = External clock from pin RC0/T1OSO/T1CKI (on the rising edge)
- 0 = Internal clock (FOSC/4)

bit 0 **TMR1ON:** Timer1 On bit

- 1 = Enables Timer1
- 0 = Stops Timer1

**PIC16F876:  
TEMPORIZADORES: TIMER1**

**TABLE 6-2: 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, 10Bh, 18Bh	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	PSPIF <sup>(1)</sup>	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
8Ch	PIE1	PSPIE <sup>(1)</sup>	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
0Eh	TMR1L	Holding Register for the Least Significant Byte of the 16-bit TMR1 Register								xxxxx xxxxx	uuuuu uuuuu
0Fh	TMR1H	Holding Register for the Most Significant Byte of the 16-bit TMR1 Register								xxxxx xxxxx	uuuuu uuuuu
10h	T1CON	—	—	T1CKPS1	T1CKPS0	T1OSCEN	T1SYNC	TMR1CS	TMR1ON	--00 0000	--uu uuuuu

Legend: x = unknown, u = unchanged, - = unimplemented, read as '0'. Shaded cells are not used by the Timer1 module.

**Note 1:** Bits PSPIE and PSPIF are reserved on the PIC16F873/876; always maintain these bits clear.



**PIC16F876:  
TEMPORIZADORES:TIMER1**

**UNA PARTE DE INTCON, PIR1 Y PIE1  
ESTA LIGADA CON LA GESTIÓN DEL  
TEMPORIZADOR TIMER1**

**FIGURE 2-3: PIC16F877/876 REGISTER FILE MAP**

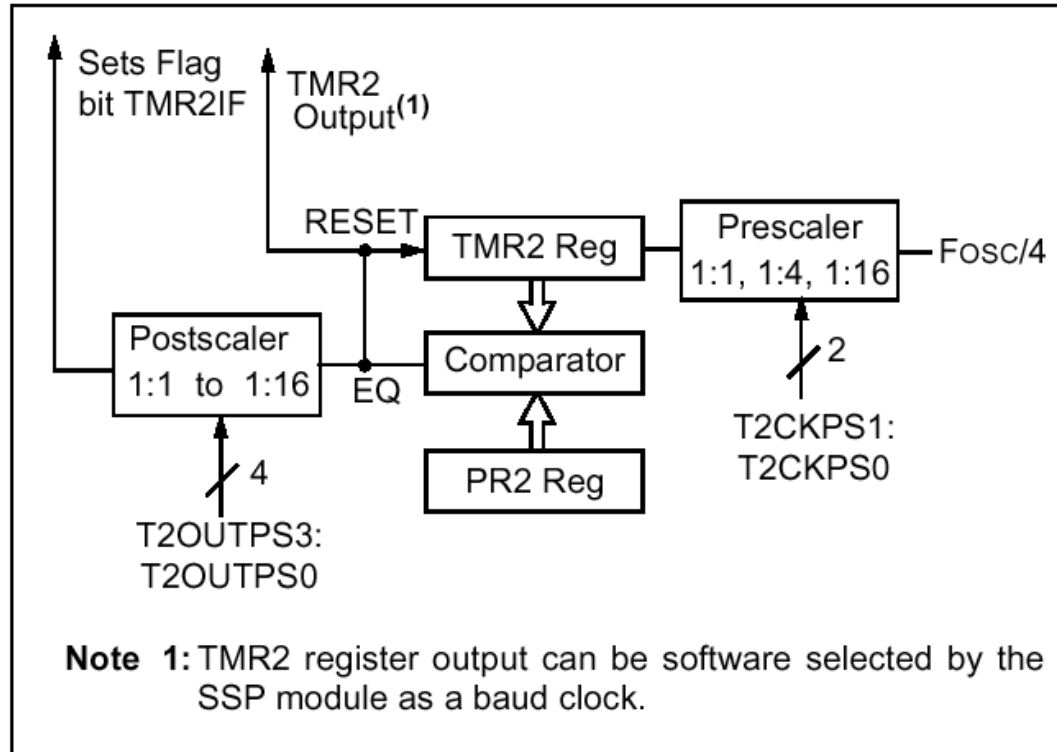
File Address	File Address	File Address	File Address
Indirect addr. <sup>(1)</sup> 00h	Indirect addr. <sup>(1)</sup> 80h	Indirect addr. <sup>(1)</sup> 100h	Indirect addr. <sup>(1)</sup> 180h
TMR0 01h	OPTION_REG 81h	TMR0 101h	OPTION_REG 181h
PCL 02h	PCL 82h	PCL 102h	PCL 182h
STATUS 03h	STATUS 83h	STATUS 103h	STATUS 183h
FSR 04h	FSR 84h	FSR 104h	FSR 184h
PORTA 05h	TRISA 85h		
PORTB 06h	TRISB 86h	PORTB 106h	TRISB 186h
PORTC 07h	TRISC 87h		
<del>PORTD 08h</del>	<del>TRISD 88h</del>		
<del>PORTF 09h</del>	<del>TRISF 89h</del>		
PCLATH 0Ah	PCLATH 8Ah	PCLATH 10Ah	PCLATH 18Ah
INTCON 0Bh	INTCON 8Bh	INTCON 10Bh	INTCON 18Bh
PIR1 0Ch	PIE1 8Ch	EEDATA 10Ch	EECON1 18Ch
PIR2 0Dh	PIE2 8Dh	EEADR 10Dh	EECON2 18Dh
TMR1L 0Eh	PCON 8Eh	EEDATH 10Eh	Reserved <sup>(2)</sup> 18Eh
TMR1H 0Fh		EEADRH 10Fh	Reserved <sup>(2)</sup> 18Fh
T1CON 10h			
TMR2 11h	SSPCON2 91h		
T2CON 12h	PR2 92h		
SSPBUF 13h	SSPAD0 93h		
SSPCON 14h	SSPSTAT 94h		
CCPR1L 15h			
CCPR1H 16h			
CCP1CON 17h		General Purpose Register 16 Bytes 117h	General Purpose Register 16 Bytes 197h
RCSTA 18h	TXSTA 98h		
TXREG 19h	SPBRG 99h		
RCREG 1Ah			
CCPR2L 1Bh			
CCPR2H 1Ch			
CCP2CON 1Dh			
ADRESH 1Eh	ADRESL 9Eh		
ADCON0 1Fh	ADCON1 9Fh		
General Purpose Register 96 Bytes 20h-7Fh	General Purpose Register 80 Bytes A0h-EFh	General Purpose Register 80 Bytes 120h-16Fh	General Purpose Register 80 Bytes 1A0h-1EFh
	accesses 70h-7Fh	accesses 70h-7Fh	accesses 70h-7Fh
Bank 0	Bank 1	Bank 2	Bank 3

Unimplemented data memory locations, read as '0'.  
 \* Not a physical register.

**Note 1:** These registers are not implemented on the PIC16F876.  
**Note 2:** These registers are reserved, maintain these registers clear.

PIC16F876:  
TEMPORIZADORES: TIMER2

FIGURE 7-1: TIMER2 BLOCK DIAGRAM



Cuenta hasta que iguala PR2 y luego se reinicia

El tiempo máximo es:

$$1 \mu\text{S} \times 16 \times 256 \times 16 = 65.5 \text{ mS}$$

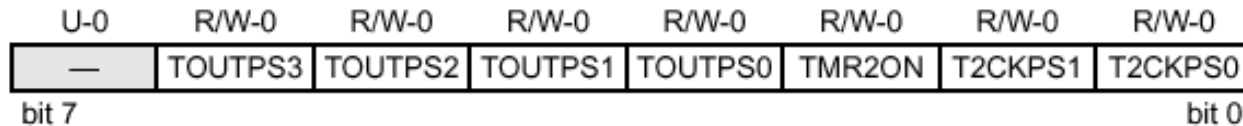
The prescaler and postscaler counters are cleared when any of the following occurs:

- a write to the TMR2 register
- a write to the T2CON register
- any device RESET (POR,  $\overline{\text{MCLR}}$  Reset, WDT Reset, or BOR)

TMR2 is not cleared when T2CON is written.

**PIC16F876:  
TEMPORIZADORES: TIMER2**

**REGISTER 7-1: T2CON: TIMER2 CONTROL REGISTER (ADDRESS 12h)**



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

bit 6-3 **TOUTPS3:TOUTPS0:** Timer2 Output Postscale Select bits

0000 = 1:1 Postscale

0001 = 1:2 Postscale

0010 = 1:3 Postscale

•

•

•

1111 = 1:16 Postscale

bit 2 **TMR2ON:** Timer2 On bit

1 = Timer2 is on

0 = Timer2 is off

bit 1-0 **T2CKPS1:T2CKPS0:** Timer2 Clock Prescale Select bits

00 = Prescaler is 1

01 = Prescaler is 4

1x = Prescaler is 16

**PIC16F876:**  
**TEMPORIZADORES: TIMER2**

**TABLE 7-1: 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, 10Bh,18Bh	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	PSPIF <sup>(1)</sup>	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
8Ch	PIE1	PSPIE <sup>(1)</sup>	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
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

**PIC16F876:  
TEMPORIZADORES:TIMER2**

**UNA PARTE DE INTCON, PIR1 Y PIE1  
ESTA LIGADA CON LA GESTIÓN DEL  
TEMPORIZADOR TIMER2**

**FIGURE 2-3: PIC16F877/876 REGISTER FILE MAP**

File Address	File Address	File Address	File Address
Indirect addr. <sup>(1)</sup> 00h	Indirect addr. <sup>(1)</sup> 80h	Indirect addr. <sup>(1)</sup> 100h	Indirect addr. <sup>(1)</sup> 180h
TMR0 01h	OPTION_REG 81h	TMR0 101h	OPTION_REG 181h
PCL 02h	PCL 82h	PCL 102h	PCL 182h
STATUS 03h	STATUS 83h	STATUS 103h	STATUS 183h
FSR 04h	FSR 84h	FSR 104h	FSR 184h
PORTA 05h	TRISA 85h		
PORTB 06h	TRISB 86h	PORTB 106h	TRISB 186h
PORTC 07h	TRISC 87h		
<del>PORTD 08h</del>	<del>TRISD 88h</del>		
<del>PORTE 09h</del>	<del>TRISE 89h</del>		
PCLATH 0Ah	PCLATH 8Ah	PCLATH 10Ah	PCLATH 18Ah
INTCON 0Bh	INTCON 8Bh	INTCON 10Bh	INTCON 18Bh
PIR1 0Ch	PIE1 8Ch	EEDATA 10Ch	EECON1 18Ch
PIR2 0Dh	PIE2 8Dh	EEADR 10Dh	EECON2 18Dh
TMR1L 0Eh	PCON 8Eh	EEDATH 10Eh	Reserved <sup>(2)</sup> 18Eh
TMR1H 0Fh		EEADRH 10Fh	Reserved <sup>(2)</sup> 18Fh
T1CON 10h			
TMR2 11h	SSPCON2 91h		
T2CON 12h	PR2 92h		
SSPBUF 13h	SSPAD0 93h		
SSPCON 14h	SSPSTAT 94h		
CCPR1L 15h			
CCPR1H 16h			
CCP1CON 17h			
RCSTA 18h	TXSTA 98h	General Purpose Register 16 Bytes	General Purpose Register 16 Bytes
TXREG 19h	SPBRG 99h		
RCREG 1Ah			
CCPR2L 1Bh			
CCPR2H 1Ch			
CCP2CON 1Dh			
ADRESH 1Eh	ADRESL 9Eh		
ADCON0 1Fh	ADCON1 9Fh		
General Purpose Register 96 Bytes	General Purpose Register 80 Bytes	General Purpose Register 80 Bytes	General Purpose Register 80 Bytes
	accesses 70h-7Fh	accesses 70h-7Fh	accesses 70h-7Fh
Bank 0 7Fh	Bank 1 FFh	Bank 2 17Fh	Bank 3 1FFh

■ Unimplemented data memory locations, read as '0'.  
\* Not a physical register.

**Note 1:** These registers are not implemented on the PIC16F876.  
**Note 2:** These registers are reserved, maintain these registers clear.



### WATCHDOG o "PERRO GUARDIAN"

• El temporizador Watchdog es un temporizador existente en el microcontrolador **basado en un oscilador RC interno, independiente del oscilador del microcontrolador** y que no requiere ningún componente externo. El WATCHDOG contará incluso si el reloj conectado a OSC1/CLKI y/o OSC2/CLKO está parado, por ejemplo, por la ejecución de una instrucción SLEEP ó por un defecto del cristal oscilador.

• Este oscilador RC interno no tiene nada que ver con un posible oscilador RC externo conectado a la patilla OSC1/CLKI.



• El **funcionamiento o no del Watchdog** se debe seleccionar en la **palabra de configuración** a la hora de grabar el microcontrolador: bit WDTE de la palabra de configuración a 1 -> activo (por defecto tras el borrado del microcontrolador); si está a 0 -> inactivo

• Si está activo, durante el funcionamiento normal del microcontrolador, **un desbordamiento** (ó time-out) del Watchdog **provoca un Reset del microcontrolador** (Watchdog Timer Reset). Para que no se desborde, cada cierto tiempo y antes de que llegue al límite, **se debe ejecutar una instrucción CLRWDT** que "limpia" el Watchdog y le hace comenzar una nueva cuenta desde cero.

• Si el dispositivo **está en modo dormido**, un desbordamiento del watchdog provoca que el micro **despierte y continúe con el funcionamiento normal** (Watchdog Timer Wake-Up) con la instrucción que sigue a SLEEP (la que lo mandó a dormir)

• El bit  $\overline{TO}$  del registro STATUS **se pone a cero tras un desbordamiento del Watchdog** y nos permite conocer tal circunstancia de desbordamiento.



WATCHDOG o "PERRO GUARDIAN" (II)

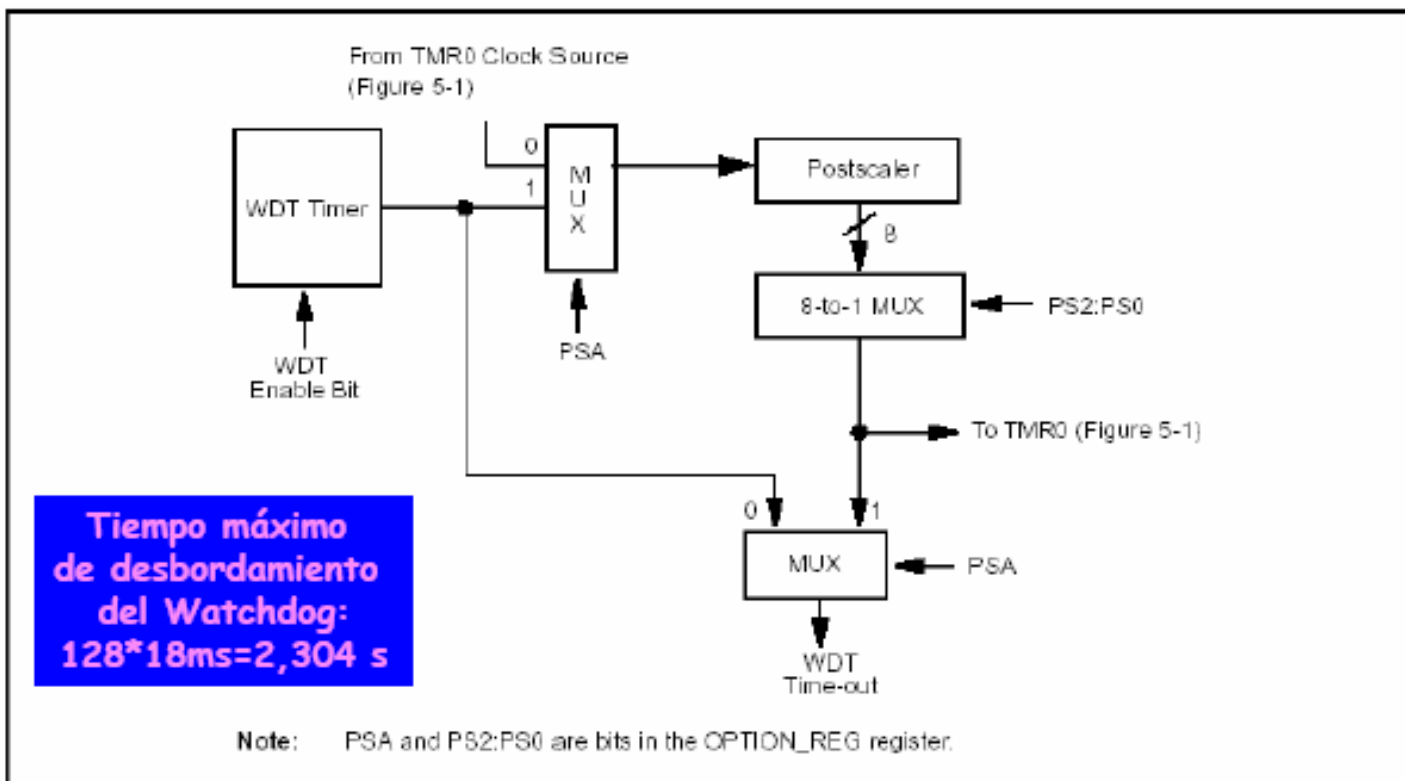
- El time-out mínimo, tiempo que tarda en desbordar (sin postscaler) para el WATCHDOG viene dado por el siguiente parámetro:

		Mínimo	Típico	Máximo		
31*	TWDT	Watchdog Timer Time-out Period (no prescaler)	7	18	33	ms VDD = 5V, -40°C to +85°C

- Ese tiempo, como se puede comprobar, *es muy variable al depender de una red RC*
- Un postscaler es un *divisor de frecuencia* que puede hacer que se cuente antes el número de desbordamientos del WDT y hacer así que el tiempo que tarda en resetear al microcontrolador sea más largo. El inconveniente es que ese divisor de frecuencia está compartido con el TMRO y por tanto, *si se usa para el TMRO no se puede usar para el WATCHDOG y viceversa.*
- El divisor de frecuencia del WATCHDOG viene definido por unos bits del registro OPTION:  
*PSA*: a quién se le asigna el divisor  
*PS2-PS1-PS0*: cuál es el factor de división de la frecuencia:

<u>000</u> : 1:2 para TMRO / 1:1 para WDT.	<u>100</u> : 1:32 para TMRO / 1:16 para WDT.
<u>001</u> : 1:4 para TMRO / 1:2 para WDT.	<u>101</u> : 1:64 para TMRO / 1:32 para WDT.
<u>010</u> : 1:8 para TMRO / 1:4 para WDT.	<u>110</u> : 1:128 para TMRO / 1:64 para WDT.
<u>011</u> : 1:16 para TMRO / 1:8 para WDT.	<u>111</u> : 1:256 para TMRO / 1:128 para WDT.

## DIAGRAMA DE BLOQUES DEL WATCHDOG



## REGISTROS ASOCIADOS AL FUNCIONAMIENTO DEL WATCHDOG

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
2007h	Config. bits	(1)	BODEN <sup>(1)</sup>	CP1	CP0	PWRTE <sup>(1)</sup>	WDTE	Fosc1	Fosc0
81h, 181h	OPTION_REG	RBPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0



## TEMPORIZADOR WATCHDOG (VIGILANTE)

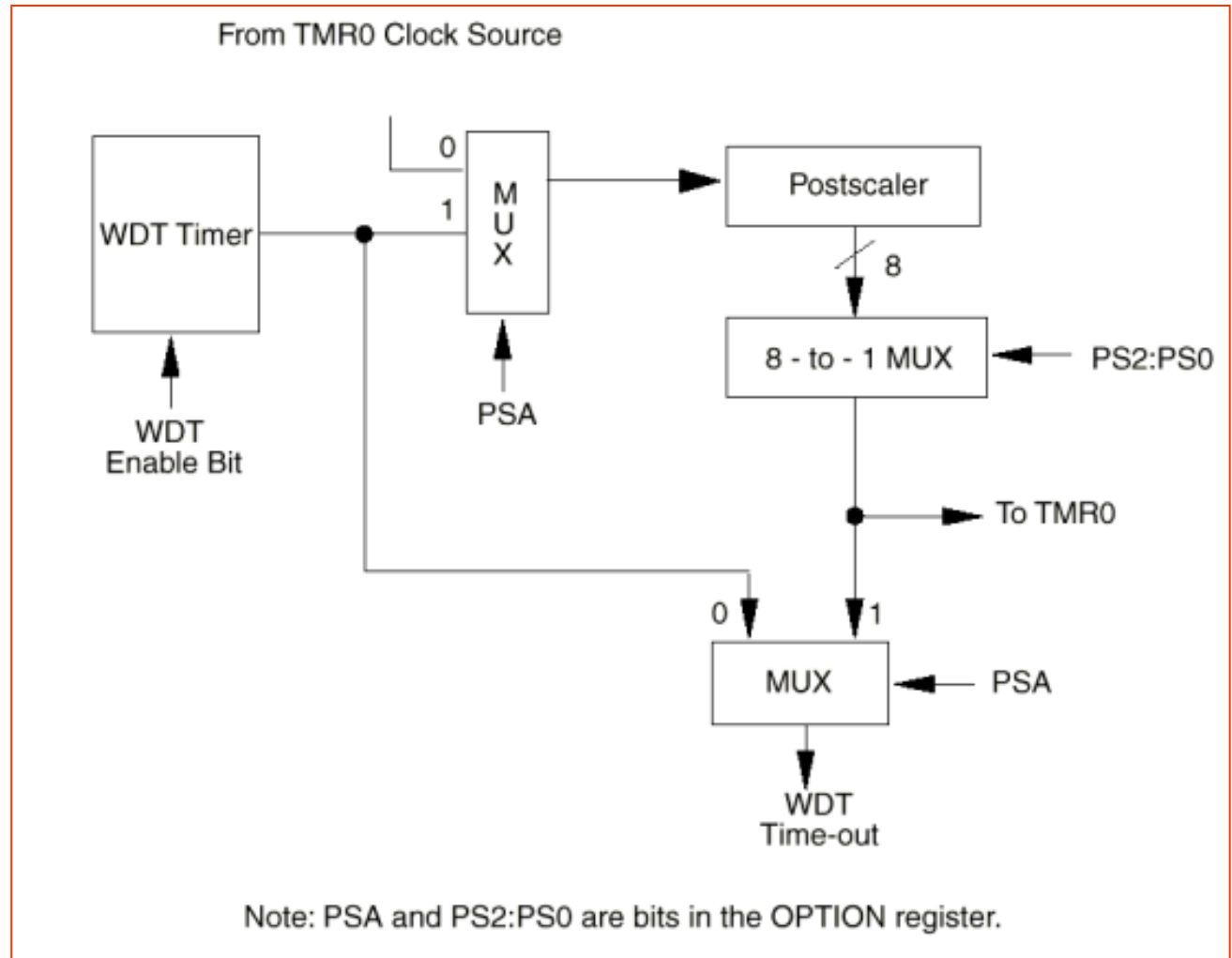
Funciona en modo SLEEP.  
Oscilador con RC independiente

Sin PRESCALER el tiempo que tarda en llegar a cero es de 18 mS.

Con el PRESCALER máximo ( $\div 128$ ) llegamos hasta 2.3 S

Instrucción CLRWDT

Provoca un RESET si no se borra periodicamente



## CONVERSOR A/D

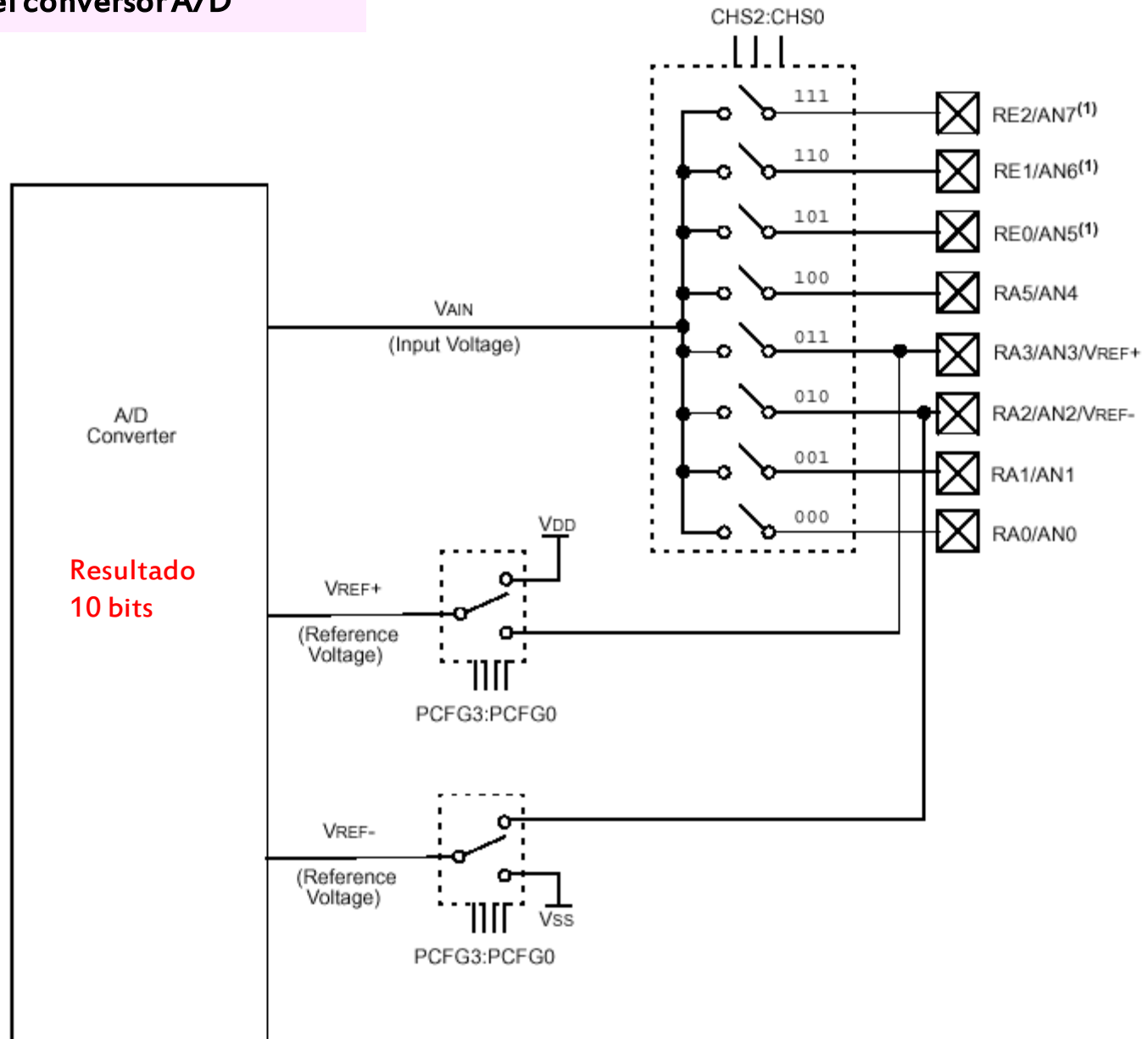
The Analog-to-Digital (A/D) Converter module has five inputs for the 28-pin devices and eight for the other devices.

The analog input charges a sample and hold capacitor. The output of the sample and hold capacitor is the input into the converter. The converter then generates a digital result of this analog level via successive approximation. The A/D conversion of the analog input signal results in a corresponding 10-bit digital number. The A/D module has high and low voltage reference input that is software selectable to some combination of VDD, VSS, RA2, or RA3.

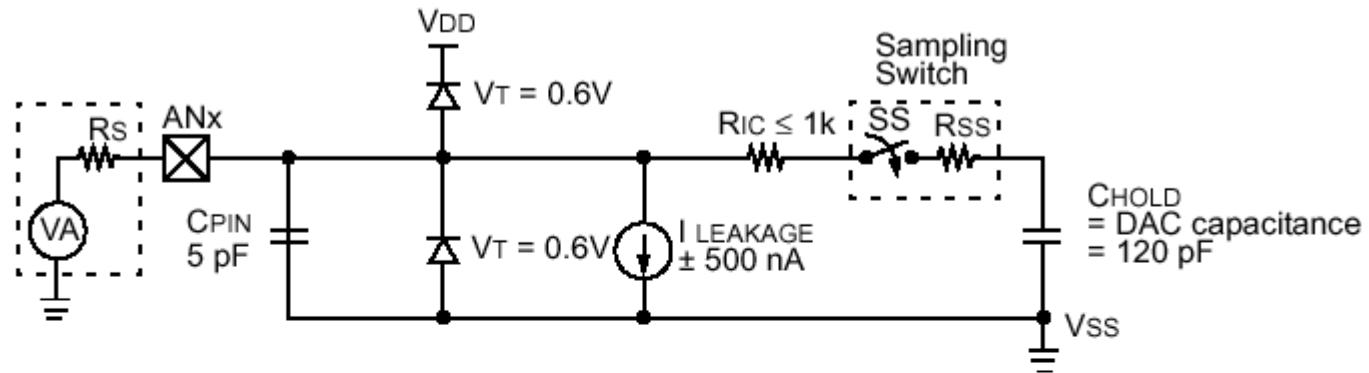
The A/D module has four registers. These registers are:

- A/D Result High Register (ADRESH)
- A/D Result Low Register (ADRESL)
- A/D Control Register0 (ADCON0)
- A/D Control Register1 (ADCON1)

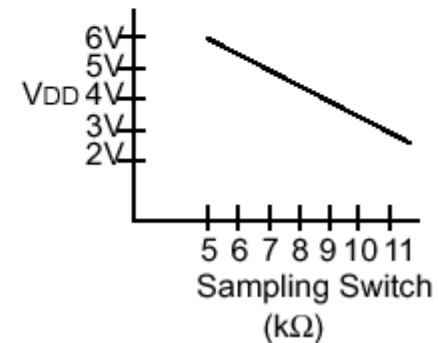
# Estructura del conversor A/D



## ESTRUCTURA DE ENTRADA: CIRCUITO DE CAPTURAY MANTENIMIENTO



Legend	CPIN	= input capacitance
	VT	= threshold voltage
	I LEAKAGE	= leakage current at the pin due to various junctions
	RIC	= interconnect resistance
	SS	= sampling switch
	CHOLD	= sample/hold capacitance (from DAC)



**Fijarse en el tiempo de adquisición: (con  $R_{ss} = 8K$  y  $R_s = 1K$ )**

**Constante de tiempo ( $\tau$ )  $\approx (R_{IC} + R_{ss} + R_s) \cdot C \approx 120 \cdot 10^{-12} \cdot 10 \cdot 10^3 = 1.2 \mu S$**

**$5\tau \approx 6 \mu S$  (Con 4 MHz son 6 instrucciones p.e. NOP)**

## Proceso de conversión

The A/D conversion time per bit is defined as  $T_{AD}$ . The A/D conversion requires a minimum  $12T_{AD}$  per 10-bit conversion. The source of the A/D conversion clock is software selected. The four possible options for  $T_{AD}$  are:

- $2T_{OSC}$
- $8T_{OSC}$
- $32T_{OSC}$
- Internal A/D module RC oscillator (2-6  $\mu s$ )

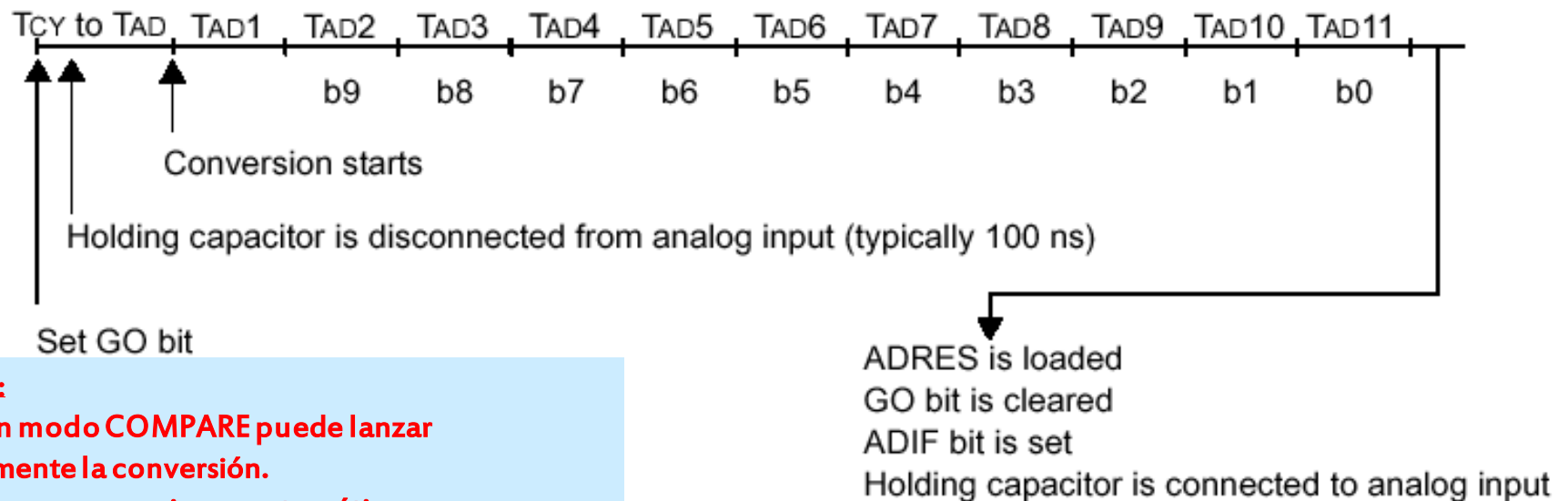
Debe asegurarse un  $T_{AD} > 1.6 \mu s$

Si relojes de 4 MHz ( $T_{OSC} = 0.25 \mu s$ )

Es buena opción tomar  $8T_{OSC}$

AD Clock Source ( $T_{AD}$ )		Maximum Device Frequency
Operation	ADCS1:ADCS0	Max.
$2T_{OSC}$	00	1.25 MHz
$8T_{OSC}$	01	5 MHz
$32T_{OSC}$	10	20 MHz
RC <sup>(1, 2, 3)</sup>	11	(Note 1)

Note 1: The RC source has a typical  $T_{AD}$  time of 4  $\mu s$ , but can vary between 2-6  $\mu s$ .



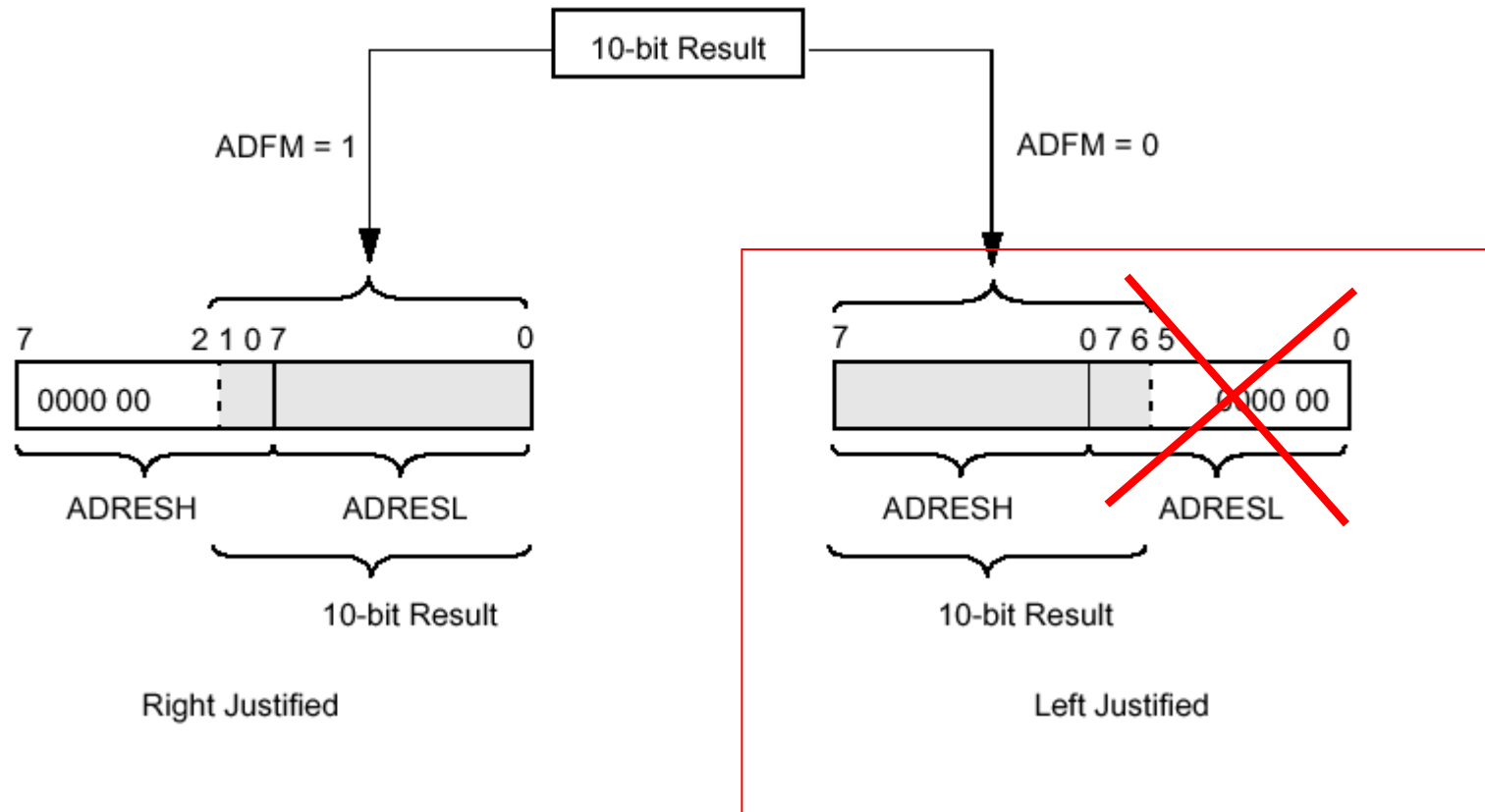
### Comentario:

El TIMER1 en modo COMPARE puede lanzar automáticamente la conversión.

Interesante para conversiones automáticas.

Se debe esperar  $2T_{AD}$  antes de empezar una nueva conversión

## Registros ADRESH y ADRESL donde se guarda la conversión.



Nos interesa esta opción y despreciamos los bits bajos (Es como dividir por 4 el resultado de la conversión)

**REGISTER 11-1: ADCON0 REGISTER (ADDRESS: 1Fh)**

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0
ADCS1	ADCS0	CHS2	CHS1	CHS0	GO/DONE	—	ADON
bit 7					bit 0		

bit 7-6 **ADCS1:ADCS0:** A/D Conversion Clock Select bits  
 00 = FOSC/2  
 → 01 = FOSC/8  
 10 = FOSC/32  
 11 = FRC (clock derived from the internal A/D module RC oscillator)

B'01 000 0 0 1'  
 H'41'

bit 5-3 **CHS2:CHS0:** Analog Channel Select bits  
 → 000 = channel 0, (RA0/AN0)  
 001 = channel 1, (RA1/AN1)  
 010 = channel 2, (RA2/AN2)  
 011 = channel 3, (RA3/AN3)  
 100 = channel 4, (RA5/AN4)  
 101 = channel 5, (RE0/AN5)<sup>(1)</sup>  
 110 = channel 6, (RE1/AN6)<sup>(1)</sup>  
 111 = channel 7, (RE2/AN7)<sup>(1)</sup>

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

bit 1 **Unimplemented:** Read as '0'

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

**REGISTER 11-2: ADCON1 REGISTER (ADDRESS 9Fh)**

U-0	U-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
ADFM	—	—	—	PCFG3	PCFG2	PCFG1	PCFG0
bit 7							bit 0

- bit 7     **ADFM:** A/D Result Format Select bit  
 1 = Right justified. 6 Most Significant bits of ADRESH are read as '0'.  
 0 = Left justified. 6 Least Significant bits of ADRESL are read as '0'.
- bit 6-4   **Unimplemented:** Read as '0'
- bit 3-0   **PCFG3:PCFG0:** A/D Port Configuration Control bits:

PCFG3: PCFG0	AN7 <sup>(1)</sup> RE2	AN6 <sup>(1)</sup> RE1	AN5 <sup>(1)</sup> RE0	AN4 RA5	AN3 RA3	AN2 RA2	AN1 RA1	AN0 RA0	VREF+	VREF-	CHAN/ Refs <sup>(2)</sup>
0000	A	A	A	A	A	A	A	A	VDD	VSS	8/0
0001	A	A	A	A	VREF+	A	A	A	RA3	VSS	7/1
0010	D	D	D	A	A	A	A	A	VDD	VSS	5/0
0011	D	D	D	A	VREF+	A	A	A	RA3	VSS	4/1
0100	D	D	D	D	A	D	A	A	VDD	VSS	3/0
0101	D	D	D	D	VREF+	D	A	A	RA3	VSS	2/1
011x	D	D	D	D	D	D	D	D	VDD	VSS	0/0
1000	A	A	A	A	VREF+	VREF-	A	A	RA3	RA2	6/2
1001	D	D	A	A	A	A	A	A	VDD	VSS	6/0
1010	D	D	A	A	VREF+	A	A	A	RA3	VSS	5/1
1011	D	D	A	A	VREF+	VREF-	A	A	RA3	RA2	4/2
1100	D	D	D	A	VREF+	VREF-	A	A	RA3	RA2	3/2
1101	D	D	D	D	VREF+	VREF-	A	A	RA3	RA2	2/2
1110	D	D	D	D	D	D	D	A	VDD	VSS	1/0
1111	D	D	D	D	VREF+	VREF-	D	A	RA3	RA2	1/2

A = Analog input    D = Digital I/O



These 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 PEIE 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 (with interrupts enabled); OR
  - Waiting for the A/D interrupt
6. Read A/D result register pair (ADRESH:ADRESL), 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  $T_{AD}$ . A minimum wait of  $2T_{AD}$  is required before the next acquisition starts.

**Note:** For the A/D module to operate in SLEEP, the A/D clock source must be set to RC (ADCS1:ADCS0 = 11). To allow the conversion to occur during SLEEP, ensure the SLEEP instruction immediately follows the instruction that sets the  $\overline{\text{GO/DONE}}$  bit.

**TABLE 11-2: REGISTERS/BITS ASSOCIATED WITH A/D**

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on MCLR, WDT
0Bh,8Bh, 10Bh,18Bh	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	PSPIF <sup>(1)</sup>	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
8Ch	PIE1	PSPIE <sup>(1)</sup>	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
1Eh	ADRESH	A/D Result Register High Byte								xxxx xxxx	uuuu uuuu
9Eh	ADRESL	A/D Result Register Low Byte								xxxx xxxx	uuuu uuuu
1Fh	ADCON0	ADCS1	ADCS0	CHS2	CHS1	CHS0	GO/DONE	—	ADON	0000 00-0	0000 00-0
9Fh	ADCON1	ADFM	—	—	—	PCFG3	PCFG2	PCFG1	PCFG0	--0- 0000	--0- 0000
85h	TRISA	—	—	PORTA Data Direction Register						--11 1111	--11 1111
05h	PORTA	—	—	PORTA Data Latch when written: PORTA pins when read						--0x 0000	--0u 0000
89h <sup>(1)</sup>	TRISE	IBF	OBF	IBOV	PSPMODE	—	PORTE Data Direction bits			0000 -111	0000 -111
09h <sup>(1)</sup>	PORTE	—	—	—	—	—	RE2	RE1	RE0	---- -xxx	---- -uuu

## Ejemplo de rutina para hacer una conversión A/D (Sin interrupciones)

```
B'01 000 0 0 1'  
H'41'
```

```
; Configurar y seleccionar canal  
AD:      MOVLW   H'41'  
         MOVWF  ADCON0  
; Esperar a que se establezca la señal analógica  
         CALL   DELAY  
; Inicio de la conversión  
         BSF    ADCON0,2  
; Esperar a que acabe la conversión  
ESPERA:  BTFSC   ADCON0,2  
         GOTO   ESPERA  
; Leer el resultado  
         MOVF   ADRESH,0  
; Volver  
         RETURN
```

```
; Se debe configurar correctamente el conversor  
; Al inicio del programa  
BSF      STATUS,5           ; cambiar al banco 1  
MOVLW    B'0000 1110'      ; A0 es analógico  
MOVWF    ADCON1            ; justificado IZQ  
BCF      STATUS,5           ; volver al banco 0
```

```
; La llamada a la rutina será  
CALLAD  
; En W tenemos el dato digitalizado en 8 bits
```

## Otro ejemplo de rutina para hacer una conversión A/D (con interrupciones)

```

;
; Este es un programa ejemplo de uso del conversor A/D
; en un PIC16F877, donde se usa un solo canal (CH0)
; y se usan interrupciones
; El conversor A/D se configura como sigue:
;   Vref = +5V interna.
;   A/D Osc. = RC interna
;   Canal A/D = CH0
; Se puede usar como Hardware para probar este ejemplo
; la tarjeta PICDEM II. El programa convierte el valor del
; potenciómetro conectado a RA0 en 10 bits, de los que los
; 8 bits más significativos se muestran en los leds conectados
; al PORTB.
;
;           #include <P16F877.INC>
;
TEMP equ 20h      ; Variable de almacenamiento temporal
;
;           ORG 0x00      ; Vector de Reset
;
;           goto start
;
;           org 0x04      ; Vector de interrupción
;           goto service_int
;
;
;           org 0x10
start            movlw OFFh      ; PORTB = 1111111b
                movwf PORTB
                bsf STATUS,RPO    ; Banco 1
                movwf TRISA        ; PORTA son entradas
                clrf TRISB         ; PORTB son salida
                bcf STATUS,RPO     ; Banco 0
                call InitializeAD
                call SetupDelay     ; Delay para Tad
                bsf ADCON0,GO      ; Inicia conversión A/D
loop            goto loop
;

```

```

; Rutina de interrupción A/D:
; muestra valor en los leds del PORTB

service_int     btfss PIR1,ADIF    ; ¿Interrupción del modulo A/D?
                retfie             ; Si no retornamos
                movf ADRESH,W      ; Cojo los 8 bits altos de la conversión
                movwf PORTB        ; los muestro en los LEDS del PORTB
                bcf PIR1,ADIF      ; Reseteo el flag
                call SetupDelay     ; Delay de adquisición
                call SetupDelay     ; mayor de 20 us
                bsf ADCON0,GO      ; lanzo una nueva conversión
                retfie             ; retorno, habilito GIE

; InitializeAD, inicializa el modulo A/D.
; Selecciona CH0 a CH3 como entradas analógicas, reloj RC y lee el CH0.
;
InitializeAD    bsf STATUS,RPO    ; Banco 1
                movlw B'00000100' ; RA0,RA1,RA3 entradas analogicas
                movwf ADCON1       ; Justificado a la izquierda
                ; 8 bits mas significativos en ADRESH
                bsf PIE1,ADIE      ; Habilitamos interrupciones A/D
                bcf STATUS,RPO     ; Banco 0
                movlw 0C1h         ; Oscilador RC, Entrada analógica CH0
                movwf ADCON0       ; Modulo A/D en funcionamiento
                bcf PIR1,ADIF      ; Limpio flag interrupción
                bsf INTCON,PEIE    ; Habilito interrupciones de perifericos
                bsf INTCON,GIE     ; Habilito interrupciones globales
                return

; Esta rutina es un retardo software de más de 10us si
; se usa un oscilador de 4MHz que se usa para asegurar
; un tiempo de adquisición de más de 20 us antes de
; lanzar una nueva conversión.

SetupDelay     movlw 3             ; Carga Temp con 3
                movwf TEMP
SD              decfsz TEMP,F      ; Bucle de retardo
                goto SD
                return

END

```



Características especiales: ARRANQUE Y REINICIOS



Los PIC16F87X tienen 6 posibles fuentes de RESET del MCU:

- Power-on Reset (POR) -> Reset de Alimentación del Microcontrolador
- MCLR Reset durante funcionamiento normal -> Activación del pin de Reset en modo normal
- MCLR Reset durante SLEEP -> Activación del pin de Reset en modo de bajo consumo
- WDT Reset (durante funcionamiento normal) -> Desbordamiento del Watchdog en modo normal
- WDT Wake-up (durante SLEEP) -> Desbordamiento del Watchdog en modo de bajo consumo
- Brown-out Reset (BOR) -> Reset por caída temporal de la alimentación

La mayoría de los registros del mapa de memoria de datos no se ven afectados por ningún tipo de RESET, su estado ó valor puede ser desconocido si se produce un POR o permanecer inalterado respecto a su último valor con otro tipo de RESET.

No obstante, hay muchos otros registros que son "reseteados" a un valor determinado si se produce un POR, un MCLR Reset ó WDT Reset durante funcionamiento normal, un MCLR Reset durante SLEEP ó un BOR. Estos registros no suelen ver afectado su valor si se produce un WDT Wake-Up, que realmente es una vuelta al funcionamiento normal desde el punto de programa donde se hubiera ejecutado el SLEEP.

Los bits  $\overline{TO}$  y  $\overline{PD}$  del registro STATUS y los bits  $\overline{POR}$  y  $\overline{BOR}$  del registro PCON dan información de cuál fue el motivo del último RESET. v pueden leerse v utilizarse luego en el programa.

TABLE 12-4: STATUS BITS AND THEIR SIGNIFICANCE

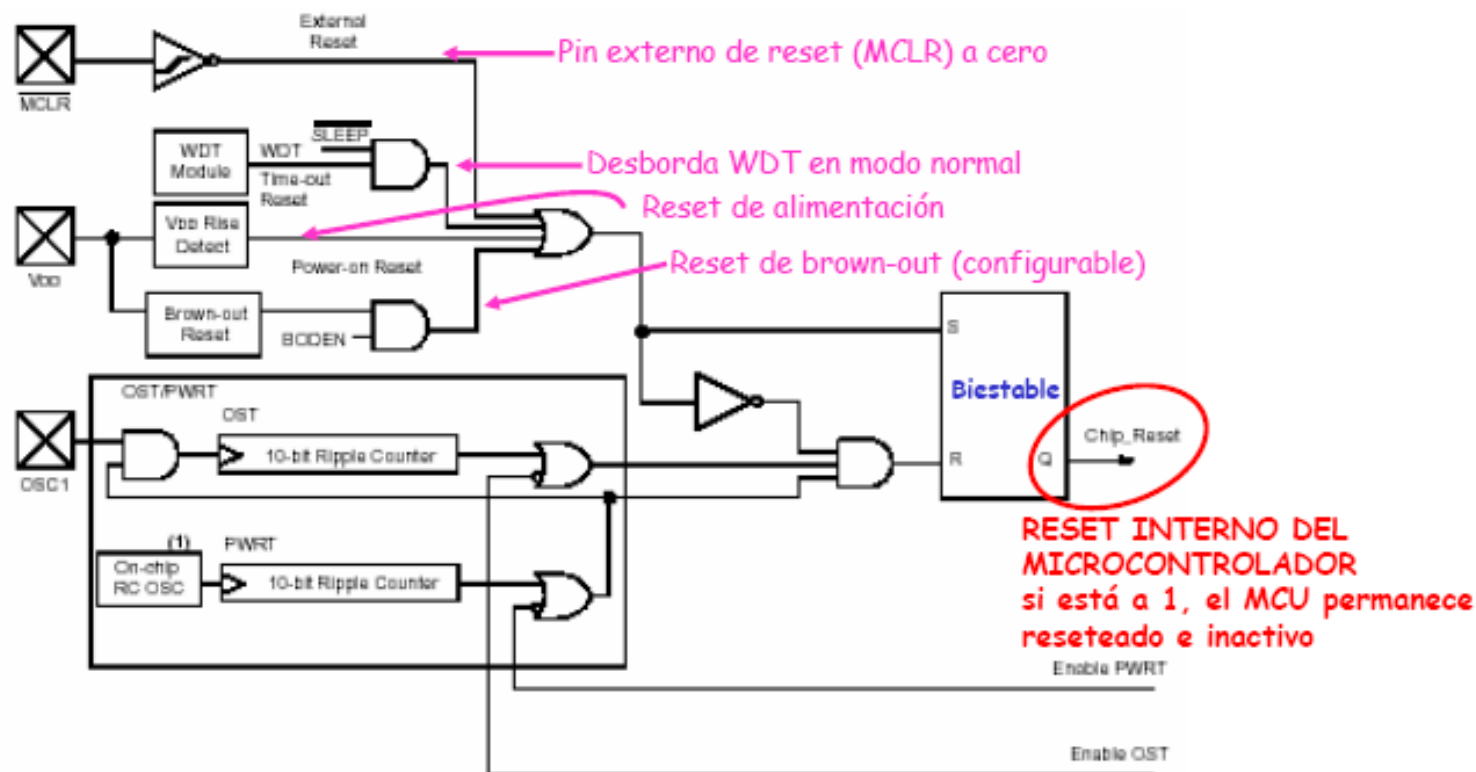
POR	BOR	TO	PD	
0	x	1	1	Power-on Reset
0	x	0	x	Illegal TO is set on POR
0	x	x	1	Illegal PD is set on POR
1	0	1	1	Brown-out Reset
1	1	0	1	WDT Reset
1	1	0	0	WDT Wake up
1	1	u	1	MCLR Reset during normal operation
1	1	1	0	MCLR Reset during SLEEP or interrupt wake-up from SLEEP

Legend: x = don't care, u = unchanged



**RESET DEL MCU Y TEMPORIZACIONES EN EL ARRANQUE**

La figura muestra el diagrama de bloques simplificado de la lógica interna del RESET del microcontrolador



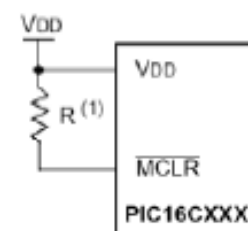
Note 1: This is a separate oscillator from the RC oscillator of the CLKIN pin.

Los PIC16F87X tienen un filtro en MCLR que detecta e ignora pequeños pulsos que se puedan producir en el pin.



En el micro, se genera internamente un pulso de reset cada vez que se alimenta el sistema. Este pulso **se produce al subir la tensión en el pin VDD y alcanzar éste un valor en el rango de 1,2V a 1,7V**. Esta característica permite eliminar el uso de redes RC externas al microcontrolador para generar un RESET en la puesta bajo tensión. Lo normal es unir la patilla  $\overline{MCLR}$  a VDD a través de una resistencia

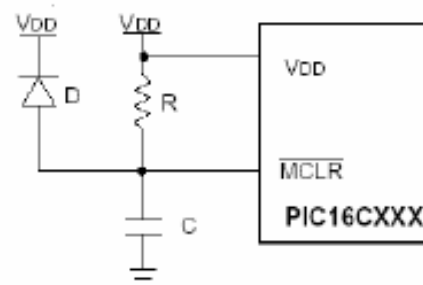
### POWER-ON RESET (POR)



Cuando el dispositivo **sale de la condición de RESET** y comienza el funcionamiento normal, debemos asegurar que los parámetros de funcionamiento (tensión, frecuencia, temperatura, etc) estén dentro de los márgenes permitidos, si no el micro podría no funcionar correctamente. Existen parámetros relativos a las características que debe tener **la pendiente de subida** de la tensión de alimentación para asegurar un pulso de POR.

D004	SVDD	VDD Rise Rate to ensure internal Power-on Reset signal	0.05	—	—	V/ms	See section on Power-on Reset for details
------	------	--	------	---	---	------	---

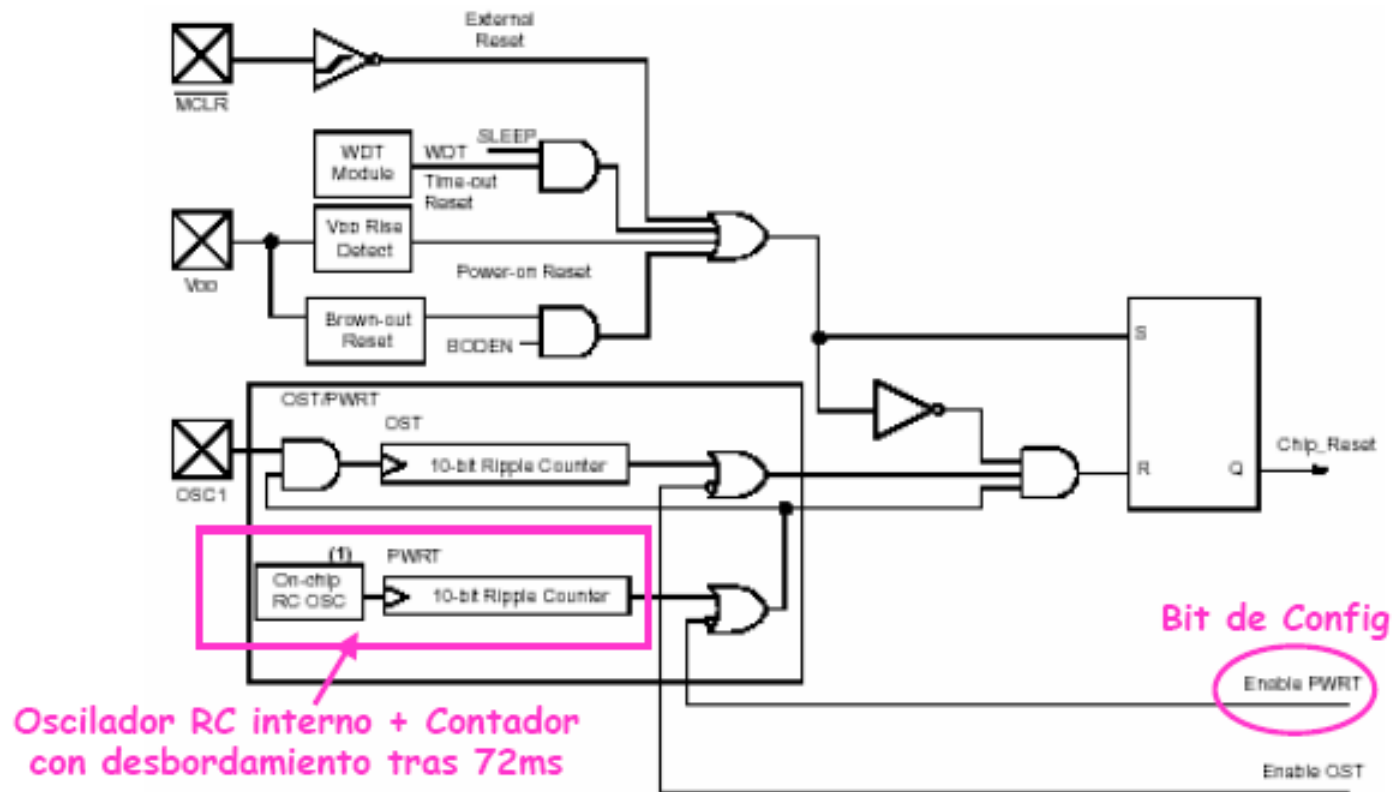
Si la pendiente de subida de la tensión VDD es excesivamente lenta suele colocarse una red RC en la patilla MCLR para asegurar un RESET correcto (el **micro se mantiene en reset** más tiempo hasta que el **nivel en el pin MCLR "sube"**). El diodo D únicamente ayuda a la descarga del condensador cuando se apaga la alimentación





POWER-UP TIMER (PWRT)

- De manera opcional (configurable) se puede esperar un tiempo (aprox. 72ms) antes de liberar el estado de reset del MCU, desde que finaliza el pulso interno de POR



Note 1: This is a separate oscillator from the RC oscillator of the CLKIN pin.



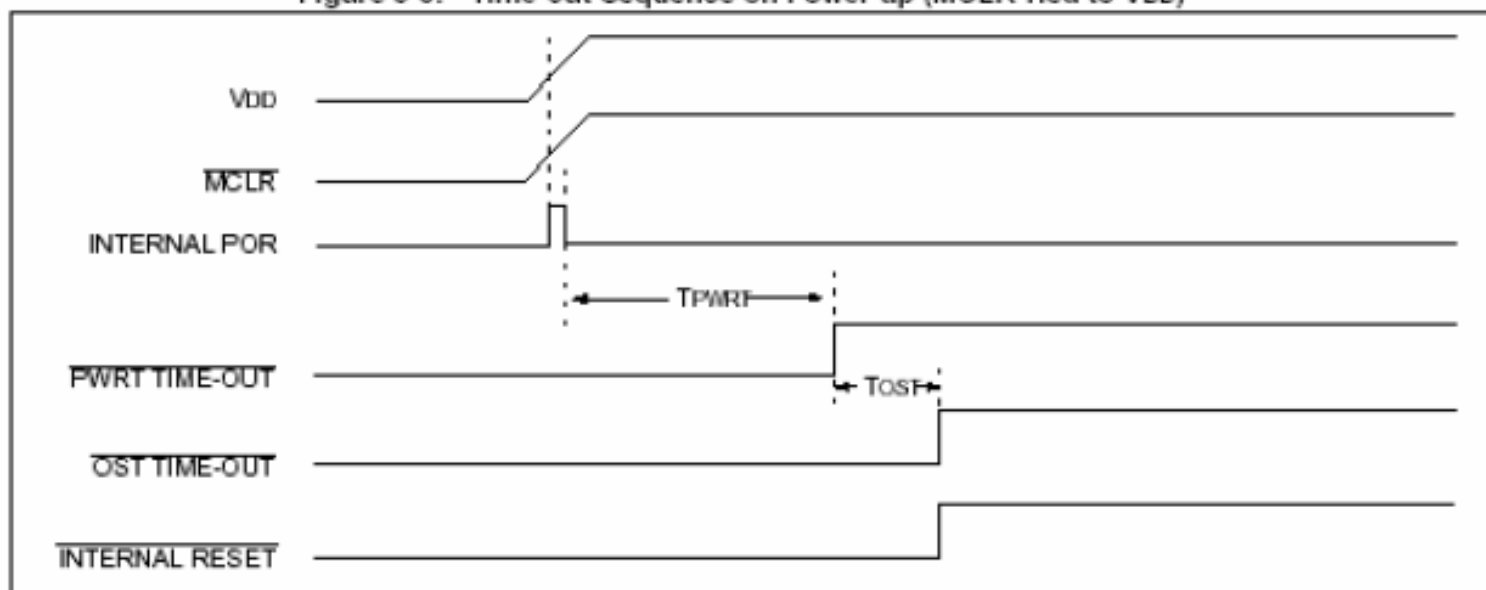


POWER-UP TIMER (PWRT)

El temporizador de Power-up proporciona un retardo fijo de unos 72ms desde que se produce el pulso de POR. Mientras que dura esta temporización el micro se mantiene en un estado de RESET. Este retardo PWRT permite a la tensión VDD crecer hasta un valor aceptable de alimentación para el propio micro y para el resto de circuitería que exista en la tarjeta y que se alimenta desde la misma fuente de alimentación.

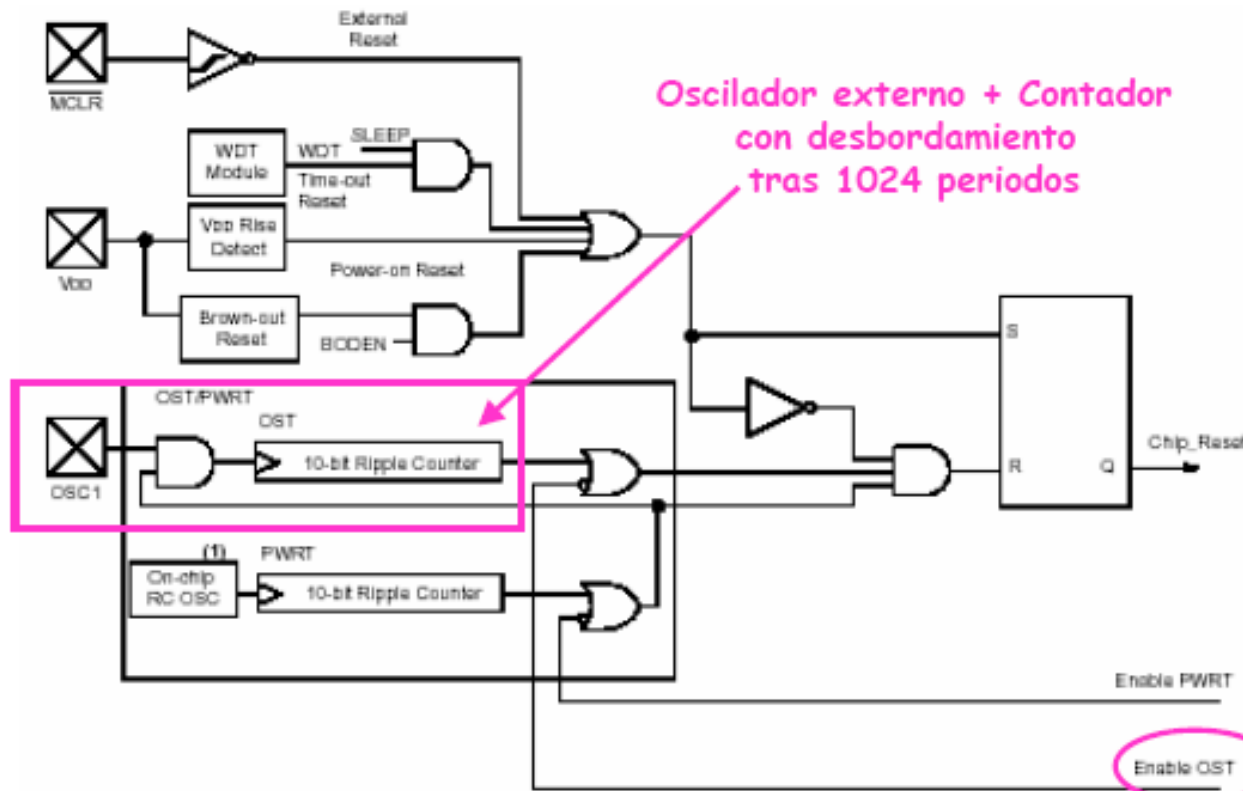
Existe un bit de configuración (PWRTEN) que permite habilitar/deshabilitar esta temporización.

Figure 3-5: Time-out Sequence on Power-up (MCLR Tied to VDD)



## OSCILLATOR START-UP TIMER (OST)

Aunque **no sea configurable**, se puede esperar después de la temporización de alimentación (PWRT) un **tiempo adicional** antes de liberar el estado de reset del MCU, todo ello para asegurar que el **oscilador ha alcanzado su frecuencia estable de oscilación**.

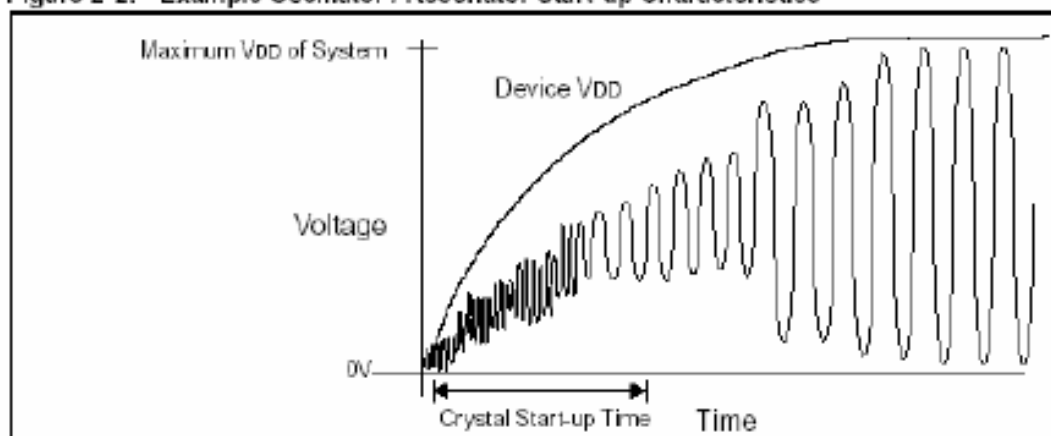


Note 1: This is a separate oscillator from the RC oscillator of the CLKIN pin.

## OSCILLATOR START-UP TIMER (OST)

La temporización de START-UP TIMER (OST) temporiza un retardo de **1024 ciclos del oscilador** una vez que finaliza el retardo debido a la temporización PWRT. Esto asegura que el oscilador de cristal o resonador cerámico ha empezado a oscilar y su **frecuencia es estable** cuando comienza el funcionamiento del programa.

Figure 2-2: Example Oscillator / Resonator Start-up Characteristics



La **temporización OST se activa** si el microcontrolador está programado como modo de **oscilador XT, LP ó HS**. Además, únicamente se produce en algunos tipos de RESET: POR, BOR y Wake-up desde SLEEP.

La cuenta de ciclos solo comienza cuando la amplitud de la oscilación alcanza los valores umbrales de oscilación (0.3 VDD y 0.7VDD).

TABLE 12-3: TIME-OUT IN VARIOUS SITUATIONS

Oscillator Configuration	Power-up	
	$\overline{\text{PWRT}} = 0$	$\overline{\text{PWRT}} = 1$
XT, HS, LP	72 ms + 1024T <sub>OSC</sub>	1024T <sub>OSC</sub>
RC	72 ms	—



**POWER-ON RESET (POR) EN DIVERSAS CIRCUNSTANCIAS**

FIGURE 12-5: TIME-OUT SEQUENCE ON POWER-UP (MCLR TIED TO VDD)

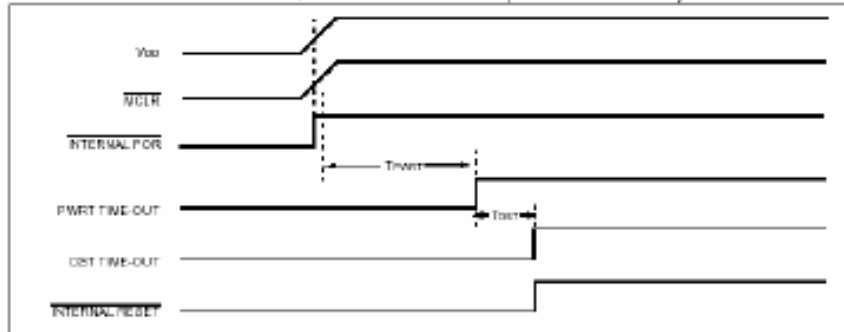


FIGURE 12-6: TIME-OUT SEQUENCE ON POWER-UP (MCLR NOT TIED TO VDD): CASE 1

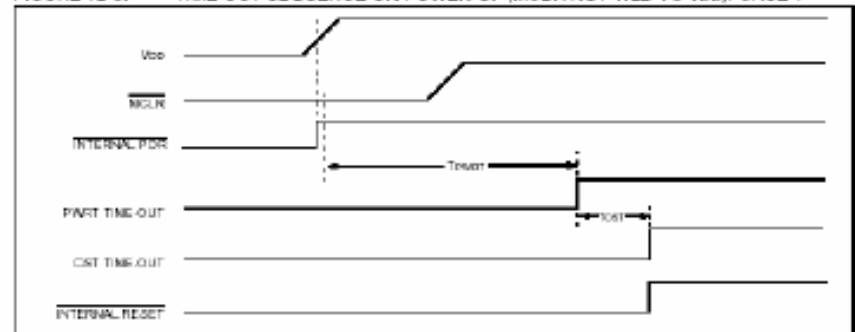


FIGURE 12-7: TIME-OUT SEQUENCE ON POWER-UP (MCLR NOT TIED TO VDD): CASE 2

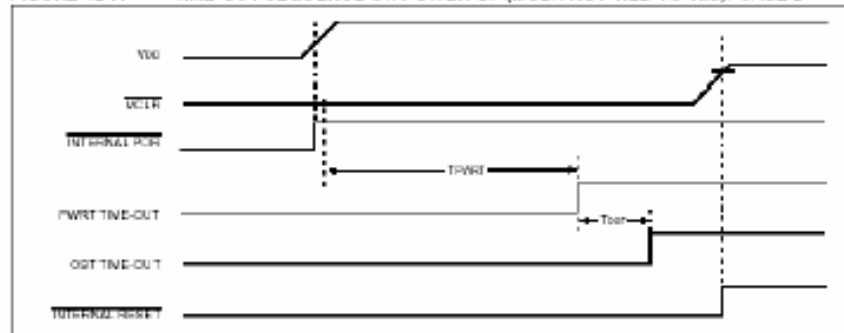
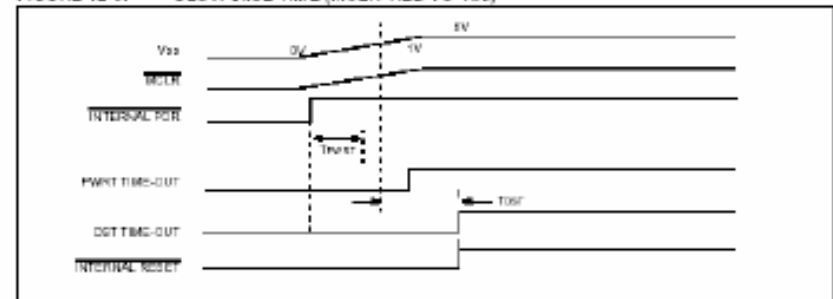


FIGURE 12-8: SLOW RISE TIME (MCLR TIED TO VDD)





### BROWN-OUT RESET

La circuitería de Brown-out disponible en el propio chip **detecta si la tensión de alimentación cae por debajo de un determinado valor** (BVDD) provocando en ese caso un RESET del dispositivo. Esto asegura que el microcontrolador no continúa con la ejecución del programa si la alimentación se sale del rango de funcionamiento válido. Brown-out RESET se utiliza principalmente en aplicaciones con alimentación desde red o desde batería donde se conmutan grandes cargas y puede suceder que la tensión de alimentación cae temporalmente por debajo de la tensión mínima de alimentación permitida. Como hemos visto **en la palabra de configuración existe un bit BODEN** (o varios en otros micros) que permite **habilitar (1) o deshabilitar (0) esta función**. Si el BROWN-OUT está habilitado, el POWER-UP Timer también debe estarlo.

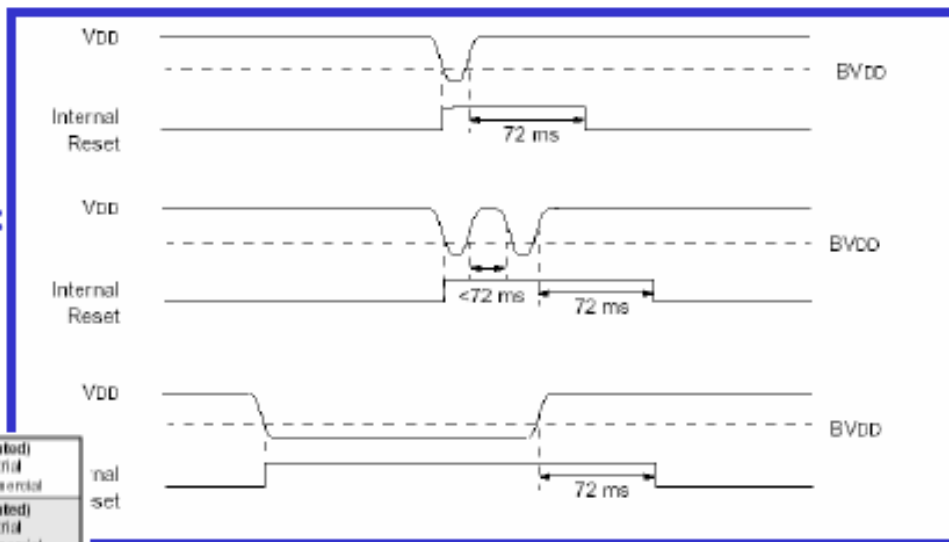
El parámetro eléctrico D005 (típicamente 4V) es la tensión mínima permitida en la alimentación. Si la tensión de alimentación desciende por debajo de este valor durante un tiempo mayor al fijado por un parámetro (el 35 que son unos 100µs) se producirá un RESET del microcontrolador. El RESET no está garantizado si la tensión de alimentación cae por debajo del valor D005 durante un tiempo menor del fijado por el parámetro 35.

El chip permanecerá en **RESET hasta que la tensión de alimentación supere BVDD**. En ese instante se inicia la temporización de POWER-UP (72 mseg) durante la que el chip se mantendrá reseteado.

Si durante esta temporización se vuelve a producir una caída de la tensión de alimentación por debajo de BVDD, el chip volverá al estado de RESET y la **temporización de Power-up volverá a arrancar desde cero** cuando la tensión de alimentación vuelva a recuperarse por encima de BVDD.

**BROWN-OUT RESET**

Ejemplo de BOR y secuencia:



PIC16F873/874/876/877-04 (Commercial, Industrial)		Standard Operating Conditions (unless otherwise stated) Operating temperature: $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for industrial $0^{\circ}\text{C} \leq T_A \leq +70^{\circ}\text{C}$ for commercial					
PIC16F873/874/876/877-04 PIC16F873/874/876/877-20 (Commercial, Industrial)		Standard Operating Conditions (unless otherwise stated) Operating temperature: $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for industrial $0^{\circ}\text{C} \leq T_A \leq +70^{\circ}\text{C}$ for commercial					
Param No.	Symbol	Characteristic/ Device	Min	Typ†	Max	Units	Conditions
Supply Voltage							
D001	VDD	16F87X	2.0	—	5.5	V	LP, XT, RC osc config; (DC to 4 MHz)
D001		16F87X	4.0	—	5.5	V	LP, XT, RC osc config;
D001A			4.5	—	5.5	V	HS osc configuration
		VBOR	3.7	4.0	4.35	V	BODEN bit in configuration word enabled
D002	VDR	RAM Data Retention Voltage <sup>(1)</sup>	—	1.5	—	V	
D003	VDR	VDD Start Voltage to ensure Internal Power-on Reset signal	—	VDD	—	V	See section on Power-on Reset for details
D004	SDD	VDD Rise Rate to ensure Internal Power-on Reset signal	0.05	—	—	V/ms	See section on Power-on Reset for details

FIGURE 15-9: BROWN-OUT RESET TIMING

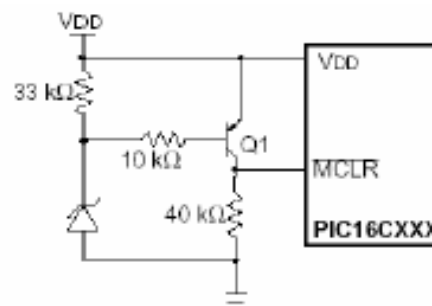


35	TBOR	Brown-out Reset pulse width	100	—	—	μs	VDD ≤ VDR (D003)
----	------	-----------------------------	-----	---	---	----	------------------

Parámetros de las características eléctricas

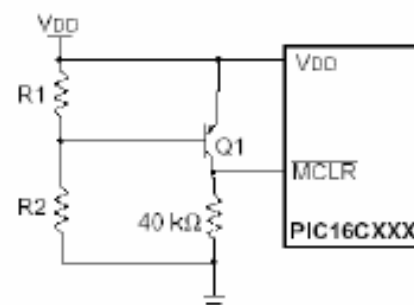
## BROWN-OUT RESET

Algunos dispositivos de la familia PIC16 **no disponen de circuitería de BROWN-OUT**. En estos dispositivos o en aquellos en los que las características de BROWN-OUT se queden cortas (es necesario asegurar una tensión de alimentación mínima por encima de la fijada por el parámetro D005) puede resultar necesario utilizar circuitos externos al chip como los que se indican en las siguientes figuras.



This circuit will activate reset when VDD goes below  $(V_z + 0.7V)$  where  $V_z$  = Zener voltage.

- Note 1: Internal Brown-out Reset circuitry should be disabled when using this circuit.  
 2: Resistors should be adjusted for the characteristics of the transistor.

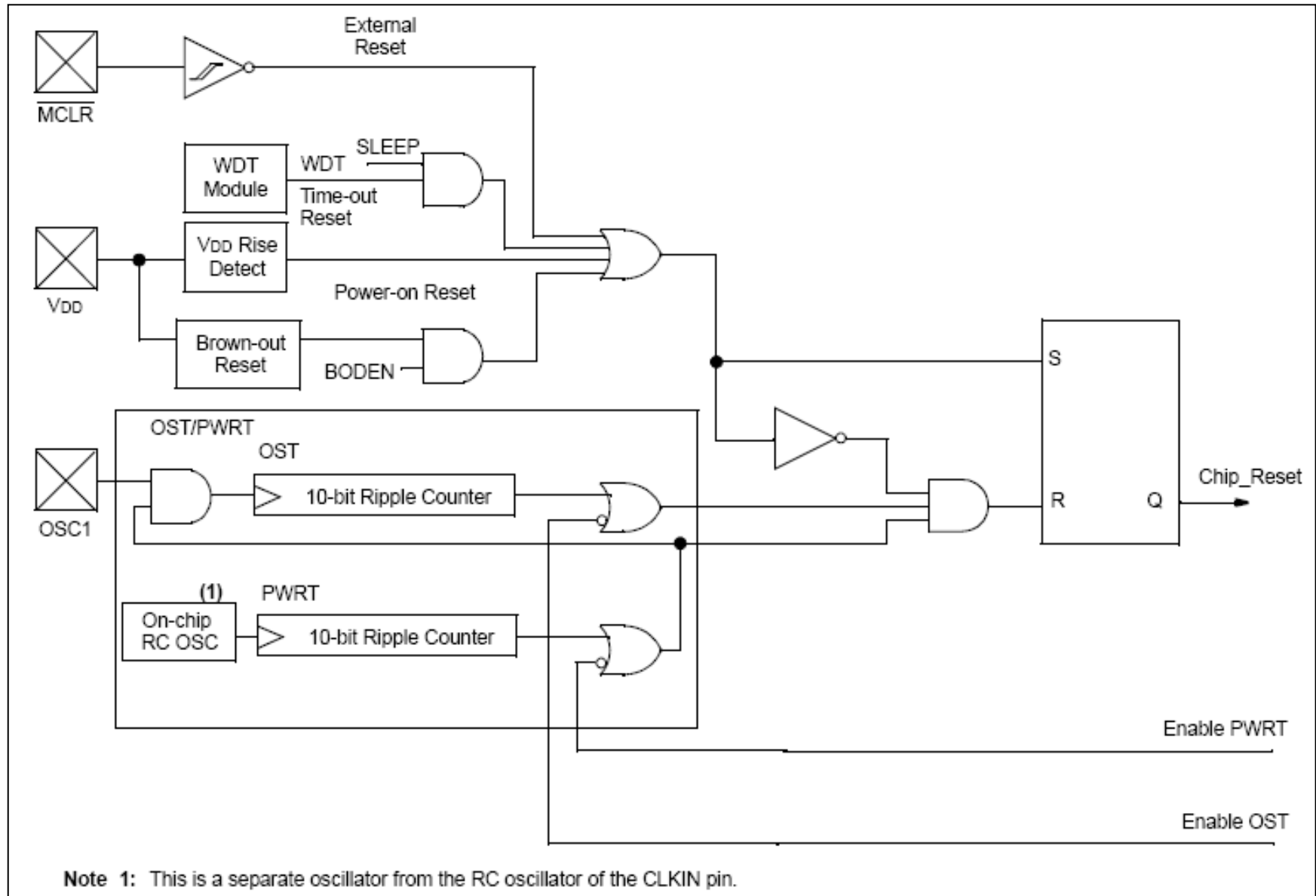


Note 1: This brown-out circuit is less expensive, albeit less accurate. Transistor Q1 turns off when VDD is below a certain level such that:

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

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

## TIPOS DE RESET





# MÓDULOS CCP1 Y CCP2

Each Capture/Compare/PWM (CCP) module contains a 16-bit register which can operate as a:

- 16-bit Capture register
- 16-bit Compare register
- PWM Master/Slave Duty Cycle register

## CCP1 Module:

Capture/Compare/PWM Register1 (CCPR1) is comprised of two 8-bit registers: CCPR1L (low byte) and CCPR1H (high byte). The CCP1CON register controls the operation of CCP1. The special event trigger is generated by a compare match and will reset Timer1.

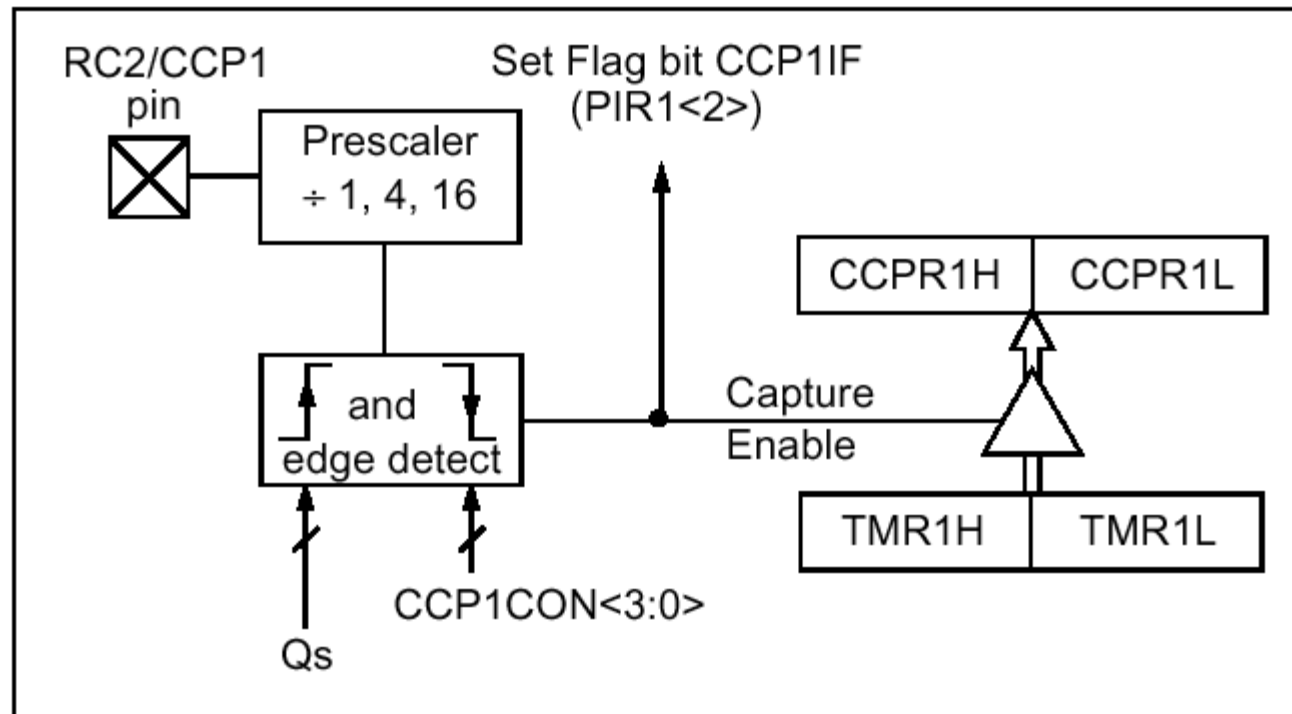
## CCP2 Module:

Capture/Compare/PWM Register2 (CCPR2) is comprised of two 8-bit registers: CCPR2L (low byte) and CCPR2H (high byte). The CCP2CON register controls the operation of CCP2. The special event trigger is generated by a compare match and will reset Timer1 and start an A/D conversion (if the A/D module is enabled).

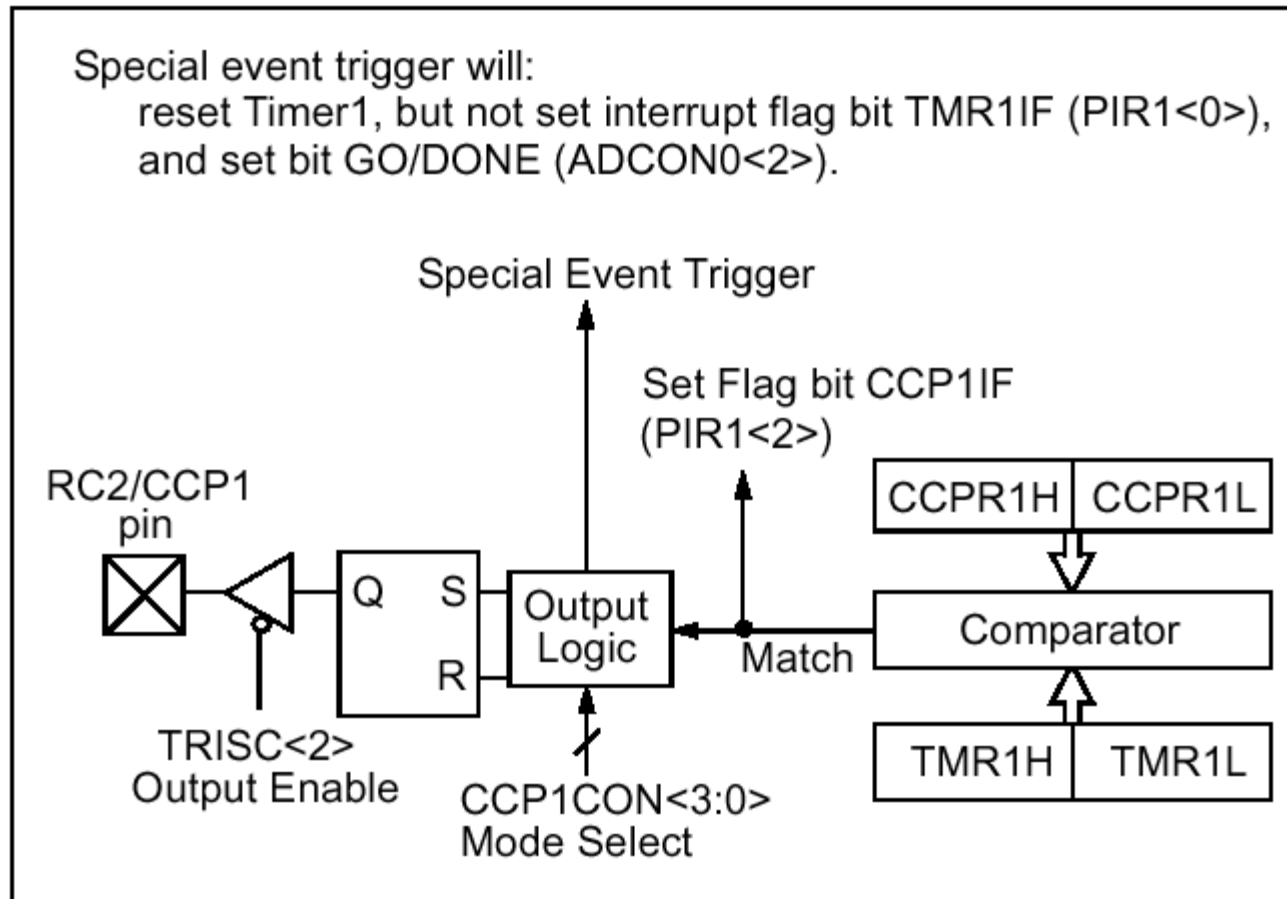
**TABLE 8-1: CCP MODE - TIMER RESOURCES REQUIRED**

CCP Mode	Timer Resource
Capture	Timer1
Compare	Timer1
PWM	Timer2

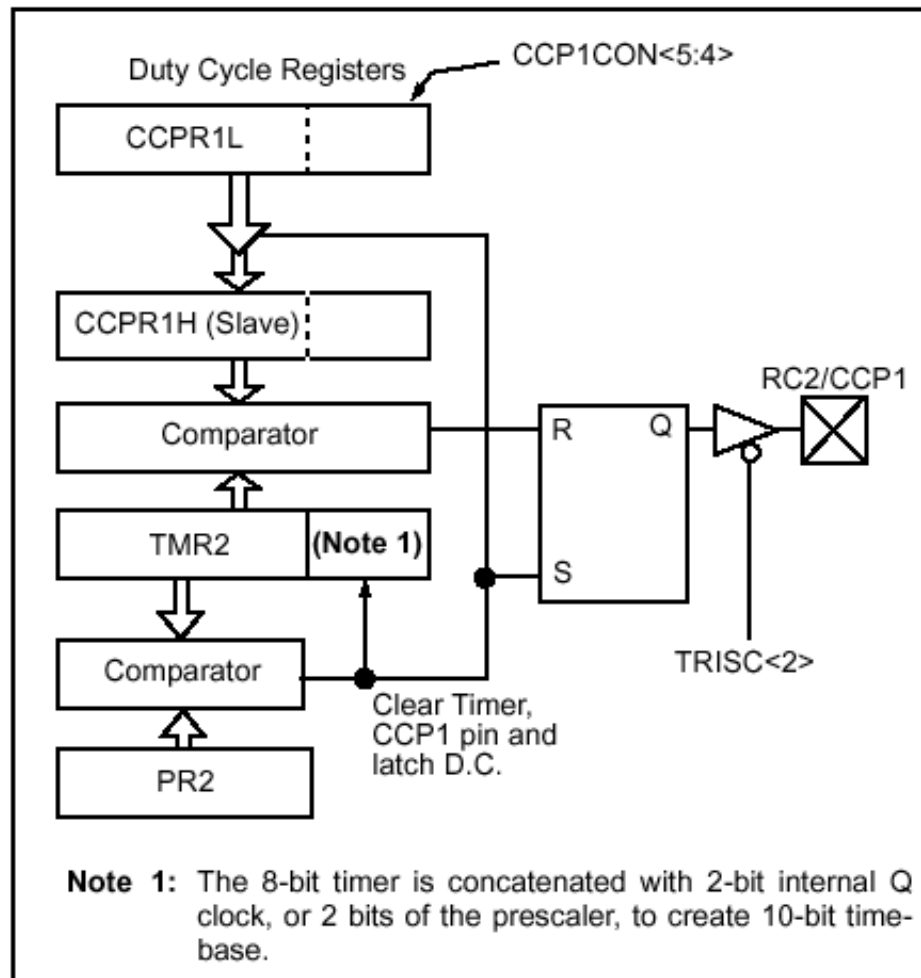
**FIGURE 8-1: CAPTURE MODE OPERATION BLOCK DIAGRAM**



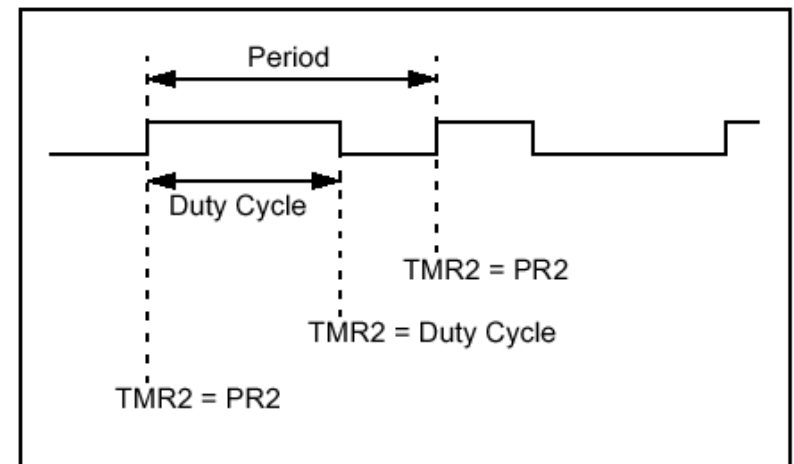
## FIGURE 8-2: COMPARE MODE OPERATION BLOCK DIAGRAM



**FIGURE 8-3: SIMPLIFIED PWM BLOCK DIAGRAM**



**FIGURE 8-4: PWM OUTPUT**



**PIC16F876:**  
**MÓDULO CAPTURE/COMPARE/PWM**

**TABLE 8-4: 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
0Bh,8Bh, 10Bh, 18Bh	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	PSPIF <sup>(1)</sup>	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
0Dh	PIR2	—	—	—	—	—	—	—	CCP2IF	---- --0	---- --0
8Ch	PIE1	PSPIE <sup>(1)</sup>	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
8Dh	PIE2	—	—	—	—	—	—	—	CCP2IE	---- --0	---- --0
87h	TRISC	PORTC Data Direction Register								1111 1111	1111 1111
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	T1SYNC	TMR1CS	TMR1ON	--00 0000	--uu uuuu
15h	CCPR1L	Capture/Compare/PWM Register1 (LSB)								xxxx xxxx	uuuu uuuu
16h	CCPR1H	Capture/Compare/PWM Register1 (MSB)								xxxx xxxx	uuuu uuuu
17h	CCP1CON	—	—	CCP1X	CCP1Y	CCP1M3	CCP1M2	CCP1M1	CCP1M0	--00 0000	--00 0000
1Bh	CCPR2L	Capture/Compare/PWM Register2 (LSB)								xxxx xxxx	uuuu uuuu
1Ch	CCPR2H	Capture/Compare/PWM Register2 (MSB)								xxxx xxxx	uuuu uuuu
1Dh	CCP2CON	—	—	CCP2X	CCP2Y	CCP2M3	CCP2M2	CCP2M1	CCP2M0	--00 0000	--00 0000

**PIC16F876:**  
**MÓDULO CAPTURE/COMPARE/PWM**

**TABLE 8-5: 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
0Bh,8Bh, 10Bh, 18Bh	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	PSPIF <sup>(1)</sup>	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
0Dh	PIR2	—	—	—	—	—	—	—	CCP2IF	---- --0	---- --0
8Ch	PIE1	PSPIE <sup>(1)</sup>	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
8Dh	PIE2	—	—	—	—	—	—	—	CCP2IE	---- --0	---- --0
87h	TRISC	PORTC Data Direction Register								1111 1111	1111 1111
11h	TMR2	Timer2 Module's Register								0000 0000	0000 0000
92h	PR2	Timer2 Module's Period Register								1111 1111	1111 1111
12h	T2CON	—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	-000 0000	-000 0000
15h	CCPR1L	Capture/Compare/PWM Register1 (LSB)								xxxx xxxx	uuuu uuuu
16h	CCPR1H	Capture/Compare/PWM Register1 (MSB)								xxxx xxxx	uuuu uuuu
17h	CCP1CON	—	—	CCP1X	CCP1Y	CCP1M3	CCP1M2	CCP1M1	CCP1M0	--00 0000	--00 0000
1Bh	CCPR2L	Capture/Compare/PWM Register2 (LSB)								xxxx xxxx	uuuu uuuu
1Ch	CCPR2H	Capture/Compare/PWM Register2 (MSB)								xxxx xxxx	uuuu uuuu
1Dh	CCP2CON	—	—	CCP2X	CCP2Y	CCP2M3	CCP2M2	CCP2M1	CCP2M0	--00 0000	--00 0000

## Registro PCON

PERMITE DIFERENCIAR ENTRE:  
POR (POWER ON RESET), BOR (BROWN-OUT RESET)

### REGISTER 2-8: PCON REGISTER (ADDRESS 8Eh)



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

bit 1 **POR:** Power-on Reset Status bit

1 = No Power-on Reset occurred

0 = A Power-on Reset occurred (must be set in software after a Power-on Reset occurs)

bit 0 **BOR:** Brown-out Reset Status bit

1 = No Brown-out Reset occurred

0 = A Brown-out Reset occurred (must be set in software after a Brown-out Reset occurs)

**PIC16F876:  
REGISTROS ESPECIALES  
DENTRO DE LA  
MEMORIA DE DATOS QUE  
AFECTAN A LA CPU**

**FIGURE 2-3: PIC16F877/876 REGISTER FILE MAP**

File Address	File Address	File Address	File Address
Indirect addr. <sup>(1)</sup> 00h	Indirect addr. <sup>(1)</sup> 80h	Indirect addr. <sup>(1)</sup> 100h	Indirect addr. <sup>(1)</sup> 180h
TMR0 01h	OPTION_REG 81h	TMR0 101h	OPTION_REG 181h
PCL 02h	PCL 82h	PCL 102h	PCL 182h
STATUS 03h	STATUS 83h	STATUS 103h	STATUS 183h
FSR 04h	FSR 84h	FSR 104h	FSR 184h
PORTA 05h	TRISA 85h		
PORTB 06h	TRISB 86h	PORTB 106h	TRISB 186h
PORTC 07h	TRISC 87h		
PORTD <sup>(1)</sup> 08h	TRISD <sup>(1)</sup> 88h		
PORTE <sup>(1)</sup> 09h	TRISE <sup>(1)</sup> 89h		
PCLATH 0Ah	PCLATH 8Ah	PCLATH 10Ah	PCLATH 18Ah
INTCON 0Bh	INTCON 8Bh	INTCON 10Bh	INTCON 18Bh
PIR1 0Ch	PIE1 8Ch	EEDATA 10Ch	ECON1 18Ch
PIR2 0Dh	PIE2 8Dh	EEADR 10Dh	ECON2 18Dh
TMR1L 0Eh	PCON 8Eh	EEDATH 10Eh	Reserved <sup>(2)</sup> 18Eh
TMR1H 0Fh		EEADRH 10Fh	Reserved <sup>(2)</sup> 18Fh
T1CON 10h			
TMR2 11h	SSPCON2 91h		
T2CON 12h	PR2 92h		
SSPBUF 13h	SSPAD0 93h		
SSPCON 14h	SSPSTAT 94h		
CCPR1L 15h			
CCPR1H 16h			
CCP1CON 17h			
RCSTA 18h	TXSTA 98h	General Purpose Register 16 Bytes	General Purpose Register 16 Bytes
TXREG 19h	SPBRG 99h		
RCREG 1Ah			
CCPR2L 1Bh			
CCPR2H 1Ch			
CCP2CON 1Dh			
ADRESH 1Eh	ADRESL 9Eh		
ADCON0 1Fh	ADCON1 9Fh		
General Purpose Register 96 Bytes	General Purpose Register 80 Bytes	General Purpose Register 80 Bytes	General Purpose Register 80 Bytes
	accesses 70h-7Fh	accesses 70h-7Fh	accesses 70h-7Fh
Bank 0	Bank 1	Bank 2	Bank 3

**RAM PARA DATOS  
368 BYTES**



■ Unimplemented data memory locations, read as '0'.  
\* Not a physical register.

**Note** 1: These registers are not implemented on the PIC16F876.  
2: These registers are reserved, maintain these registers clear.



## PIC16F876: MEMORIA EEPROM

DISPONE DE 256 BYTES DE MEMORIA EPROM.

SE PUEDE LEER Y ESCRIBIR EN ELLA DURANTE LA OPERACIÓN DEL MICRO (MEMORIA DE DATOS).

EL ACCESO DE LECTURA ESCRITURA A LA EEPROM SE HACE A TRAVÉS DE LOS REGISTROS ESPECIALES:

EEDATA  
EEDATH  
EEDR  
EEDRH  
EECON1  
EECON2

**ACCESO EEPROM**

FIGURE 2-3: PIC16F877/876 REGISTER FILE MAP

File Address	File Address	File Address	File Address
Indirect addr. <sup>(1)</sup> 00h	Indirect addr. <sup>(1)</sup> 80h	Indirect addr. <sup>(1)</sup> 100h	Indirect addr. <sup>(1)</sup> 180h
TMR0 01h	OPTION_REG 81h	TMR0 101h	OPTION_REG 181h
PCL 02h	PCL 82h	PCL 102h	PCL 182h
STATUS 03h	STATUS 83h	STATUS 103h	STATUS 183h
FSR 04h	FSR 84h	FSR 104h	FSR 184h
PORTA 05h	TRISA 85h		
PORTB 06h	TRISB 86h	PORTB 106h	TRISB 186h
PORTC 07h	TRISC 87h		
PORTD <sup>(1)</sup> 08h	TRISD <sup>(1)</sup> 88h		
PORTE <sup>(1)</sup> 09h	TRISE <sup>(1)</sup> 89h		
PCLATH 0Ah	PCLATH 8Ah	PCLATH 10Ah	PCLATH 18Ah
INTCON 0Bh	INTCON 8Bh	INTCON 10Bh	INTCON 18Bh
PIR1 0Ch	PIE1 8Ch	EEDATA 10Ch	EECON1 18Ch
PIR2 0Dh	PIE2 8Dh	EEADR 10Dh	EECON2 18Dh
TMR1L 0Eh		EEDATH 10Eh	Reserved <sup>(2)</sup> 18Eh
TMR1H 0Fh		EEDR 10Fh	Reserved <sup>(2)</sup> 18Fh
TMR2 10h			
TMR2 11h	SSPCON2 91h		
T2CON 12h	PR2 92h		
SSPBUF 13h	SSPAD0 93h		
SSPCON 14h	SSPSTAT 94h		
CCPR1L 15h			
CCPR1H 16h			
CCP1CON 17h			
RCSTA 18h	TXSTA 98h	General Purpose Register 16 Bytes 117h-119h	General Purpose Register 16 Bytes 197h-199h
TXREG 19h	SPBRG 99h		
RCON 1Ah			
CCPR2L 1Bh			
CCPR2H 1Ch			
CCP2CON 1Dh			
ADRESH 1Eh	ADRESL 9Eh		
ADCON0 1Fh	ADCON1 9Fh		
General Purpose Register 96 Bytes 20h-7Fh	General Purpose Register 80 Bytes A0h-EFh	General Purpose Register 80 Bytes 120h-16Fh	General Purpose Register 80 Bytes 1A0h-1EFh
	accesses 70h-7Fh	accesses 70h-7Fh	accesses 70h-7Fh
Bank 0	Bank 1	Bank 2	Bank 3

Unimplemented data memory locations, read as '0'.  
 \* Not a physical register.

**Note 1:** These registers are not implemented on the PIC16F876.  
**Note 2:** These registers are reserved, maintain these registers clear.

## PIC16F876: MEMORIA EEPROM

EL REGISTRO EECON1 PERMITE CONFIGURAR E INICIALIZAR EL ACCESO A LA MEMORIA EEPROM

**REGISTER 4-1: EECON1 REGISTER (ADDRESS 18Ch)**

R/W-x	U-0	U-0	U-0	R/W-x	R/W-0	R/S-0	R/S-0
EEPGD	—	—	—	WRERR	WREN	WR	RD
bit 7							bit 0

- bit 7 **EEPGD:** Program/Data EEPROM Select bit  
 1 = Accesses program memory  
 0 = Accesses data memory  
 (This bit cannot be changed while a read or write operation is in progress)
- bit 6-4 **Unimplemented:** Read as '0'
- bit 3 **WRERR:** EEPROM Error Flag bit  
 1 = A write operation is prematurely terminated  
 (any MCLR Reset or any WDT Reset during normal operation)  
 0 = The write operation completed
- bit 2 **WREN:** EEPROM Write Enable bit  
 1 = Allows write cycles  
 0 = Inhibits write to the EEPROM
- bit 1 **WR:** Write Control bit  
 1 = Initiates a write cycle. (The bit is cleared by hardware once write is complete. The WR bit can only be set (not cleared) in software.)  
 0 = Write cycle to the EEPROM is complete
- bit 0 **RD:** Read Control bit  
 1 = Initiates an EEPROM read. (RD is cleared in hardware. The RD bit can only be set (not cleared) in software.)  
 0 = Does not initiate an EEPROM read

**PIC16F876:  
MEMORIA EEPROM. PROCESO  
DE LECTURA**

The steps to reading the EEPROM data memory are:

1. Write the address to EEDATA. Make sure that the address is not larger than the memory size of the PIC16F87X device.
2. Clear the EEPGD bit to point to EEPROM data memory.
3. Set the RD bit to start the read operation.
4. Read the data from the EEDATA register.

**EXAMPLE 4-1: EEPROM DATA READ**

```
BSF    STATUS, RP1    ;  
BCF    STATUS, RP0    ;Bank 2  
MOVF   ADDR, W        ;Write address  
MOVWF  EEDATA         ;to read from  
BSF    STATUS, RP0    ;Bank 3  
BCF    EECON1, EEPGD  ;Point to Data memory  
BSF    EECON1, RD     ;Start read operation  
BCF    STATUS, RP0    ;Bank 2  
  
MOVF   EEDATA, W      ;W = EEDATA
```

## PIC16F876: MEMORIA EEPROM. PROCESO DE ESCRITURA

The steps to write to EEPROM data memory are:

1. If step 10 is not implemented, check the WR bit to see if a write is in progress.
2. Write the address to EEADR. Make sure that the address is not larger than the memory size of the PIC16F87X device.
3. Write the 8-bit data value to be programmed in the EEDATA register.
4. Clear the EEPGD bit to point to EEPROM data memory.
5. Set the WREN bit to enable program operations.
6. Disable interrupts (if enabled).
7. Execute the special five instruction sequence:
  - Write 55h to EECON2 in two steps (first to W, then to EECON2)
  - Write AAh to EECON2 in two steps (first to W, then to EECON2)
  - Set the WR bit
8. Enable interrupts (if using interrupts).
9. Clear the WREN bit to disable program operations.
10. At the completion of the write cycle, the WR bit is cleared and the EEIF interrupt flag bit is set. (EEIF must be cleared by firmware.) If step 1 is not implemented, then firmware should check for EEIF to be set, or WR to clear, to indicate the end of the program cycle.

### EXAMPLE 4-2: EEPROM DATA WRITE

```
BSF    STATUS, RP1    ;
BSF    STATUS, RP0    ;Bank 3
BTFSC  EECON1, WR     ;Wait for
GOTO   $-1            ;write to finish
BCF    STATUS, RP0    ;Bank 2
MOVF   ADDR, W        ;Address to
MOVWF  EEADR          ;write to
MOVF   VALUE, W       ;Data to
MOVWF  EEDATA         ;write
BSF    STATUS, RP0    ;Bank 3
BCF    EECON1, EEPGD  ;Point to Data memory
BSF    EECON1, WREN   ;Enable writes
                        ;Only disable interrupts
BCF    INTCON, GIE    ;if already enabled,
                        ;otherwise discard
MOVLW  0x55           ;Write 55h to
MOVWF  EECON2         ;EECON2
MOVLW  0xAA           ;Write AAh to
MOVWF  EECON2         ;EECON2
BSF    EECON1, WR     ;Start write operation
                        ;Only enable interrupts
BSF    INTCON, GIE    ;if using interrupts,
                        ;otherwise discard
BCF    EECON1, WREN   ;Disable writes
```

**PIC16F876:  
LECTURA DE LA MEMORIA FLASH  
CMOS (PROGRAMA)**

The steps to reading the FLASH program memory are:

1. Write the address to EEADRH:EEADR. Make sure that the address is not larger than the memory size of the PIC16F87X device.
2. Set the EEPGD bit to point to FLASH program memory.
3. Set the RD bit to start the read operation.
4. Execute two NOP instructions to allow the microcontroller to read out of program memory.
5. Read the data from the EEDATH:EEDATA registers.

**EXAMPLE 4-3: FLASH PROGRAM READ**

```
BSF    STATUS, RP1    ;
BCF    STATUS, RP0    ;Bank 2
MOVF   ADDR, W        ;Write the
MOVWF  EEADR          ;address bytes
MOVF   ADDR, W        ;for the desired
MOVWF  EEADRH        ;address to read
BSF    STATUS, RP0    ;Bank 3
BSF    EECON1, EEPGD ;Point to Program memory
BSF    EECON1, RD     ;Start read operation
NOP    ;Required two NOPs
NOP    ;
BCF    STATUS, RP0    ;Bank 2
MOVF   EEDATA, W      ;DATAH = EEDATA
MOVWF  DATAH        ;
MOVF   EEDATH, W     ;DATAH = EEDATH
MOVWF  DATAH        ;
```

## PIC16F876: ESCRITURA EN LA MEMORIA FLASH CMOS (PROGRAMA)

### EXAMPLE 4-4: FLASH PROGRAM WRITE

```
BSF    STATUS, RP1    ;
BCF    STATUS, RP0    ;Bank 2
MOVF   ADDR1, W       ;Write address
MOVWF  EEADR          ;of desired
MOVF   ADDRH, W       ;program memory
MOVWF  EEADRH         ;location
MOVF   VALUEL, W      ;Write value to
MOVWF  EEDATA         ;program at
MOVF   VALUEH, W      ;desired memory
MOVWF  EEDATH         ;location
BSF    STATUS, RP0    ;Bank 3
BSF    EECON1, EEPGD  ;Point to Program memory
BSF    EECON1, WREN   ;Enable writes
                          ;Only disable interrupts
BCF    INTCON, GIE    ;if already enabled,
                          ;otherwise discard
MOVLW  0x55           ;Write 55h to
MOVWF  EECON2         ;EECON2
MOVLW  0xAA           ;Write AAh to
MOVWF  EECON2         ;EECON2
BSF    EECON1, WR     ;Start write operation
NOP    ;Two NOPs to allow micro
NOP    ;to setup for write
                          ;Only enable interrupts
BSF    INTCON, GIE    ;if using interrupts,
                          ;otherwise discard
BCF    EECON1, WREN   ;Disable writes
```

The steps to write to program memory are:

1. Write the address to EEADRH:EEADR. Make sure that the address is not larger than the memory size of the PIC16F87X device.
2. Write the 14-bit data value to be programmed in the EEDATH:EEDATA registers.
3. Set the EEPGD bit to point to FLASH program memory.
4. Set the WREN bit to enable program operations.
5. Disable interrupts (if enabled).
6. Execute the special five instruction sequence:
  - Write 55h to EECON2 in two steps (first to W, then to EECON2)
  - Write AAh to EECON2 in two steps (first to W, then to EECON2)
  - Set the WR bit
7. Execute two NOP instructions to allow the microcontroller to setup for write operation.
8. Enable interrupts (if using interrupts).
9. Clear the WREN bit to disable program operations.

At the completion of the write cycle, the WR bit is cleared and the EEIF interrupt flag bit is set. (EEIF must be cleared by firmware.) Since the microcontroller does not execute instructions during the write cycle, the firmware does not necessarily have to check either EEIF, or WR, to determine if the write had finished.

**PIC16F876:  
PROTECCIÓN DE LA MEMORIA  
FLASH CMOS (PROGRAMA)**

**TABLE 4-1: READ/WRITE STATE OF INTERNAL FLASH PROGRAM MEMORY**

Configuration Bits			Memory Location	Internal Read	Internal Write	ICSP Read	ICSP Write
CP1	CP0	WRT					
0	0	x	All program memory	Yes	No	No	No
0	1	0	Unprotected areas	Yes	No	Yes	No
0	1	0	Protected areas	Yes	No	No	No
0	1	1	Unprotected areas	Yes	Yes	Yes	No
0	1	1	Protected areas	Yes	No	No	No
1	0	0	Unprotected areas	Yes	No	Yes	No
1	0	0	Protected areas	Yes	No	No	No
1	0	1	Unprotected areas	Yes	Yes	Yes	No
1	0	1	Protected areas	Yes	No	No	No
1	1	0	All program memory	Yes	No	Yes	Yes
1	1	1	All program memory	Yes	Yes	Yes	Yes

### Protección de la Memoria



- Tanto la memoria de programa como la EEPROM de datos del microcontrolador pueden ser protegidas mediante hardware para que no puedan ser leídas externamente
- Estas protecciones las selecciona el usuario en la palabra de configuración por tanto **no serán modificables en tiempo de ejecución**.
- La protección puede ser **parcial** o **total** y depende del tipo de microcontrolador

Palabra de configuración en PIC16F87x (CONFIG : 0x2007)



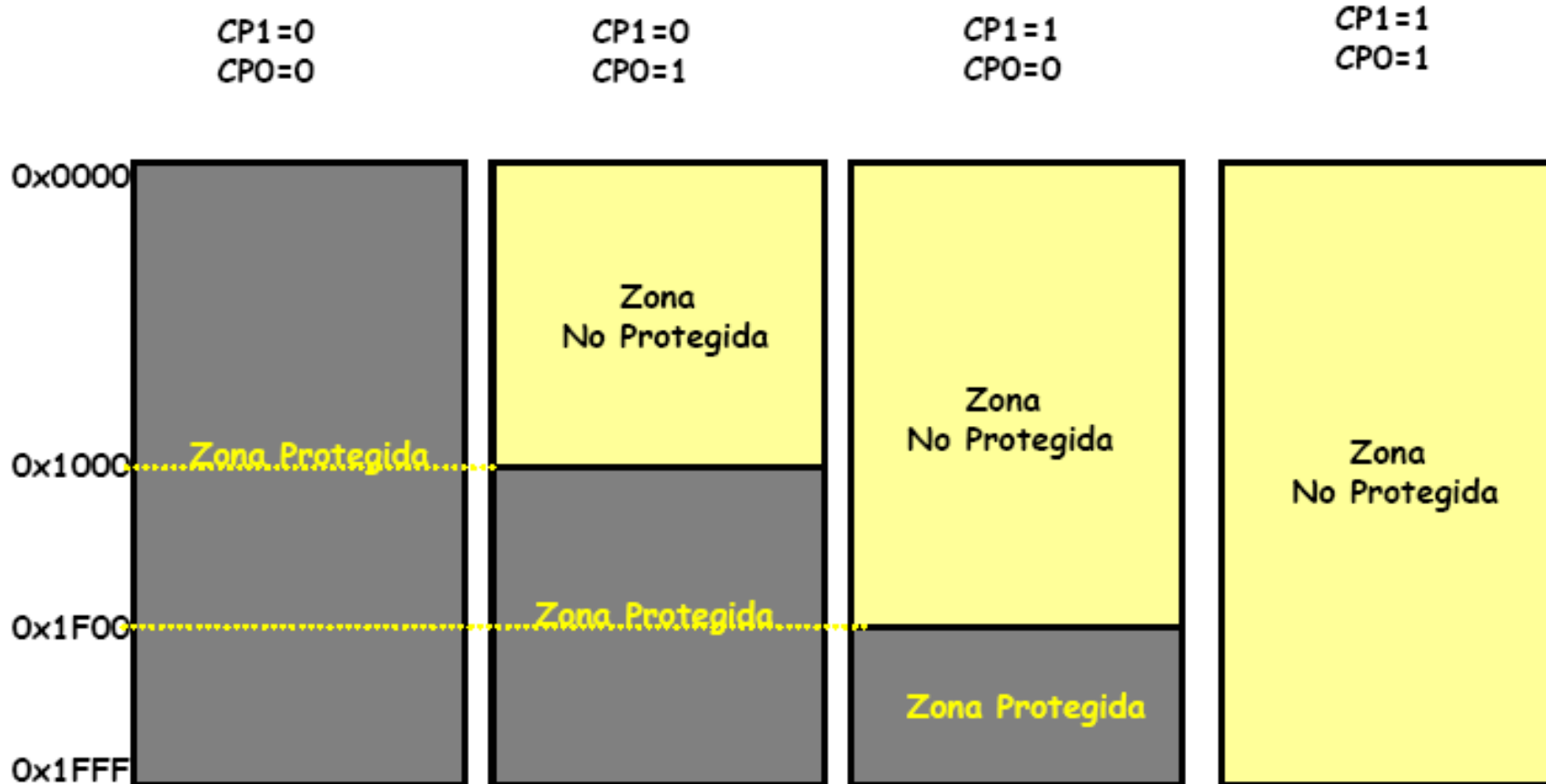
Protección de código  
Habilitar escritura interna en memoria de programa (desde programa)  
Protección de EEPROM de datos ante lectura externa

Hay dos "pares" de bits CP1-CP0, hay que grabar los dos pares por igual para lograr la protección del código de una manera segura, si se hace desde el entorno MPLAB con la ventana de Configuración, la grabación de la palabra CONFIG se hará cargando los dos pares de bits





**Configuración de la Protección de Código en PIC16F877  
en función de la carga de CP1 y CPO**





## Protección de la Memoria

- Cuando se está grabando el código, resulta posible **una primera lectura** (verificación) antes de hacer efectiva la protección. Primero se graba la memoria de programa, luego se lee para verificar la correcta escritura y finalmente se graban los bits de protección. Si esos bits activaran la protección, ya no serían posible posteriores verificaciones.
- Una vez que se ha activado la protección, **no resulta posible desproteger el código modificando exclusivamente los bits de CONFIG:**
  - En los microcontroladores **con memoria FLASH se debería borrar totalmente la memoria de programa** para poder modificar de nuevo CONFIG, pero si se ha borrado toda la memoria... ¿qué queremos proteger?
  - En los microcontroladores con EPROM y ventana para borrado por luz UV, la palabra de configuración se puede borrar igual que el resto de la memoria de programa, **salvo los bits de protección de código**, por ese motivo **se recomienda no proteger la memoria de los dispositivos EPROM salvo en su versión definitiva**
- La memoria EEPROM de programa de algunos dispositivos (p.e. PIC16F877) se puede **leer** y/o **escribir** desde el propio programa (desde dentro) durante el tiempo normal de ejecución y en todo el margen de la tensión de alimentación.
- La **lectura interna desde programa no se ve afectada por la protección** configurada en CONFIG, **siempre es posible leer desde programa**. La **escritura dependerá** de que esa zona esté o no **protegida** y de cómo se haya configurado el bit WRT (escritura interna de memoria de programa)



**Estado de Lectura/Escritura de la memoria de programa en PIC16F877**

CP1	CPO	WRT	Zona de Memoria	Lectura Interna	Escritura Interna	Lectura ICSP	Escritura ICSP
0	0	x	Toda la memoria protegida	SÍ	NO	NO	NO
Zonas protegida y zona no protegida CP1 - CPO=10 ó 01		0	Zonas protegidas	SÍ	NO	NO	NO
		1	Zonas protegidas	SÍ	NO	NO	NO
		0	Zonas no prot.	SÍ	NO	SÍ	NO
		1	Zonas no prot.	SÍ	SÍ	SÍ	NO
1	1	0	Toda la memoria desprotegida	SÍ	NO	SÍ	SÍ
1	1	1	Toda la memoria desprotegida	SÍ	SÍ	SÍ	SÍ

En las zonas protegidas no es posible la escritura interna aunque WRT=1

**PIC16F876:  
RESUMEN DE REGISTROS  
LIGADOS CON LA MEMORIA  
EEPROM Y FLASH CMOS**

**TABLE 4-2: REGISTERS ASSOCIATED WITH DATA EEPROM/PROGRAM FLASH**

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, 10Bh, 18Bh	INTCON	GIE	PEIE	TOIE	INTE	RBIE	TOIF	INTF	RBIF	0000 000x	0000 000u
10Dh	EEADR	EEPROM Address Register, Low Byte								xxxx xxxx	uuuu uuuu
10Fh	EEADRH	—	—	—	EEPROM Address, High Byte					xxxx xxxx	uuuu uuuu
10Ch	EEDATA	EEPROM Data Register, Low Byte								xxxx xxxx	uuuu uuuu
10Eh	EEDATH	—	—	EEPROM Data Register, High Byte					xxxx xxxx	uuuu uuuu	
18Ch	EECON1	EEPGD	—	—	—	WRERR	WREN	WR	RD	x--- x000	x--- u000
18Dh	EECON2	EEPROM Control Register2 (not a physical register)								—	—
8Dh	PIE2	—	(1)	—	EEIE	BCLIE	—	—	CCP2IE	-r-0 0--0	-r-0 0--0
0Dh	PIR2	—	(1)	—	EEIF	BCLIF	—	—	CCP2IF	-r-0 0--0	-r-0 0--0

**PIC16F876:  
RESUMEN DE REGISTROS  
LIGADOS CON LA MEMORIA  
EEPROM Y FLASH CMOS**

**UNA PARTE DE INTCON, PIE2 Y  
PIR2 ESTÁN LIGADOS CON LA  
GESTIÓN DE LA MEMORIA EEPROM  
Y FLASH CMOS**

**FIGURE 2-3: PIC16F877/876 REGISTER FILE MAP**

File Address	File Address	File Address	File Address
Indirect addr. <sup>(1)</sup> 00h	Indirect addr. <sup>(1)</sup> 80h	Indirect addr. <sup>(1)</sup> 100h	Indirect addr. <sup>(1)</sup> 180h
TMR0 01h	OPTION_REG 81h	TMR0 101h	OPTION_REG 181h
PCL 02h	PCL 82h	PCL 102h	PCL 182h
STATUS 03h	STATUS 83h	STATUS 103h	STATUS 183h
FSR 04h	FSR 84h	FSR 104h	FSR 184h
PORTA 05h	TRISA 85h		
PORTB 06h	TRISB 86h	PORTB 106h	TRISB 186h
PORTC 07h	TRISC 87h		
PORTD <sup>(1)</sup> 08h	TRISD <sup>(1)</sup> 88h		
PORTE <sup>(1)</sup> 09h	TRISE <sup>(1)</sup> 89h		
PCLATH 0Ah	PCLATH 8Ah	PCLATH 10Ah	PCLATH 18Ah
INTCON 0Bh	INTCON 8Bh	INTCON 10Bh	INTCON 18Bh
PIR1 0Ch	PIE1 8Ch	EEDATA 10Ch	EECON1 18Ch
PIR2 0Dh	PIE2 8Dh	EEADR 10Dh	EECON2 18Dh
TMR1L 0Eh	PCON 8Eh	EEDATH 10Eh	Reserved <sup>(2)</sup> 18Eh
TMR1H 0Fh		EEDARH 10Fh	Reserved <sup>(2)</sup> 18Fh
T1CON 10h			
TMR2 11h	SSPCON2 90h		
T2CON 12h	PR2 91h		
SSPBUF 13h	SSPAD0 92h		
SSPCON 14h	SSPSTAT 93h		
CCPR1L 15h			
CCPR1H 16h			
CCP1CON 17h			
RCSTA 18h	TXSTA 96h	General Purpose Register 16 Bytes 117h	General Purpose Register 16 Bytes 197h
TXREG 19h	SPBRG 97h		
RCREG 1Ah			
CCPR2L 1Bh			
CCPR2H 1Ch			
CCP2CON 1Dh			
ADRESH 1Eh	ADRESL 98h		
ADCON0 1Fh	ADCON1 99h		
General Purpose Register 96 Bytes 20h-7Fh	General Purpose Register 80 Bytes A0h-EFh	General Purpose Register 80 Bytes 120h-16Fh	General Purpose Register 80 Bytes 1A0h-1EFh
	accesses 70h-7Fh	accesses 70h-7Fh	accesses 70h-7Fh
Bank 0	Bank 1	Bank 2	Bank 3

■ Unimplemented data memory locations, read as '0'.  
\* Not a physical register.

**Note 1:** These registers are not implemented on the PIC16F876.  
**Note 2:** These registers are reserved, maintain these registers clear.

**PIC16F876:  
COMUNICACIÓN SERIE SINCRONA (MSSP)**

**TABLE 9-1: REGISTERS ASSOCIATED WITH SPI OPERATION**

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on: MCLR, WDT
0Bh, 8Bh, 10Bh, 18Bh	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	PSPIF <sup>(1)</sup>	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
8Ch	PIE1	PSPIE <sup>(1)</sup>	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
13h	SSPBUF	Synchronous Serial Port Receive Buffer/Transmit Register								xxxx xxxx	uuuu uuuu
14h	SSPCON	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	0000 0000
94h	SSPSTAT	SMP	CKE	D/ $\bar{A}$	P	S	R/ $\bar{W}$	UA	BF	0000 0000	0000 0000

**PIC16F876:**  
**COMUNICACIÓN SERIE SINCRONA (MSSP)**

**TABLE 9-3: REGISTERS ASSOCIATED WITH I<sup>2</sup>C OPERATION**

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on: MCLR, WDT
0Bh, 8Bh, 10Bh, 18Bh	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	PSPIF <sup>(1)</sup>	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
8Ch	PIE1	PSPIE <sup>(1)</sup>	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
0Dh	PIR2	—	(2)	—	EEIF	BCLIF	—	—	CCP2IF	-r-0 0--0	-r-0 0--0
8Dh	PIE2	—	(2)	—	EEIE	BCLIE	—	—	CCP2IE	-r-0 0--0	-r-0 0--0
13h	SSPBUF	Synchronous Serial Port Receive Buffer/Transmit Register								xxxx xxxx	uuuu uuuu
14h	SSPCON	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	0000 0000
91h	SSPCON2	GCEN	ACKSTAT	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	0000 0000	0000 0000
93h	SSPAD	I <sup>2</sup> C Slave Address/Master Baud Rate Register								0000 0000	0000 0000
94h	SSPSTAT	SMP	CKE	D/ $\bar{A}$	P	S	R/ $\bar{W}$	UA	BF	0000 0000	0000 0000

**PIC16F876:  
COMUNICACIÓN SERIE ASINCRONA (USART)**

**TABLE 10-2: REGISTERS ASSOCIATED WITH BAUD RATE GENERATOR**

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	
98h	TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010	
18h	RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	0000 000x	
99h	SPBRG	Baud Rate Generator Register									0000 0000	0000 0000

**TABLE 10-5: REGISTERS ASSOCIATED WITH ASYNCHRONOUS TRANSMISSION**

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, 10Bh, 18Bh	INTCON	GIE	PEIE	TOIE	INTE	RBIE	TOIF	INTF	ROIF	0000 000x	0000 000u	
0Ch	PIR1	PSPIF <sup>(1)</sup>	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000	
18h	RCSTA	SPEN	RX9	SREN	CREN	—	FERR	OERR	RX9D	0000 -00x	0000 -00x	
19h	TXREG	USART Transmit Register									0000 0000	0000 0000
8Ch	PIE1	PSPIE <sup>(1)</sup>	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000	
98h	TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010	
99h	SPBRG	Baud Rate Generator Register									0000 0000	0000 0000



**PIC16F876:  
COMUNICACIÓN SERIE ASINCRONA (USART)**

**TABLE 10-6: REGISTERS ASSOCIATED WITH ASYNCHRONOUS RECEPTION**

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, 10Bh, 18Bh	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	R0IF	0000 000x	0000 000u
0Ch	PIR1	PSPIF <sup>(1)</sup>	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
18h	RCSTA	SPEN	RX9	SREN	CREN	—	FERR	OERR	RX9D	0000 -00x	0000 -00x
1Ah	RCREG	USART Receive Register								0000 0000	0000 0000
8Ch	PIE1	PSPIE <sup>(1)</sup>	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
98h	TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010
99h	SPBRG	Baud Rate Generator Register								0000 0000	0000 0000

**PIC16F876:  
COMUNICACIÓN SERIE ASINCRONA (USART)**

**TABLE 10-7: REGISTERS ASSOCIATED WITH ASYNCHRONOUS RECEPTION**

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, 10Bh, 18Bh	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	R0IF	0000 000x	0000 000u
0Ch	PIR1	PSPIF <sup>(1)</sup>	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
18h	RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	0000 000x
1Ah	RCREG	USART Receive Register								0000 0000	0000 0000
8Ch	PIE1	PSPIE <sup>(1)</sup>	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
98h	TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010
99h	SPBRG	Baud Rate Generator Register								0000 0000	0000 0000

**PIC16F876:  
COMUNICACIÓN SERIE ASINCRONA (USART)**

**TABLE 10-8: REGISTERS ASSOCIATED WITH SYNCHRONOUS MASTER TRANSMISSION**

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, 10Bh, 18Bh	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	R0IF	0000 000x	0000 000u
0Ch	PIR1	PSPIF <sup>(1)</sup>	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
18h	RCSTA	SPEN	RX9	SREN	CREN	—	FERR	OERR	RX9D	0000 -00x	0000 -00x
19h	TXREG	USART Transmit Register								0000 0000	0000 0000
8Ch	PIE1	PSPIE <sup>(1)</sup>	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
98h	TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010
99h	SPBRG	Baud Rate Generator Register								0000 0000	0000 0000

**PIC16F876:  
COMUNICACIÓN SERIE ASINCRONA (USART)**

**TABLE 10-9: REGISTERS ASSOCIATED WITH SYNCHRONOUS MASTER RECEPTION**

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, 10Bh, 18Bh	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	R0IF	0000 000x	0000 000u
0Ch	PIR1	PSPIF <sup>(1)</sup>	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
18h	RCSTA	SPEN	RX9	SREN	CREN	—	FERR	OERR	RX9D	0000 -00x	0000 -00x
1Ah	RCREG	USART Receive Register								0000 0000	0000 0000
8Ch	PIE1	PSPIE <sup>(1)</sup>	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
98h	TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010
99h	SPBRG	Baud Rate Generator Register								0000 0000	0000 0000

**PIC16F876:  
COMUNICACIÓN SERIE ASINCRONA (USART)**

**TABLE 10-10: REGISTERS ASSOCIATED WITH SYNCHRONOUS SLAVE TRANSMISSION**

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, 10Bh, 18Bh	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	R0IF	0000 000x	0000 000u
0Ch	PIR1	PSP1IF <sup>(1)</sup>	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
18h	RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	0000 000x
19h	TXREG	USART Transmit Register								0000 0000	0000 0000
8Ch	PIE1	PSPIE <sup>(1)</sup>	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
98h	TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010
99h	SPBRG	Baud Rate Generator Register								0000 0000	0000 0000

**PIC16F876:  
COMUNICACIÓN SERIE ASINCRONA (USART)**

**TABLE 10-11: REGISTERS ASSOCIATED WITH SYNCHRONOUS SLAVE RECEPTION**

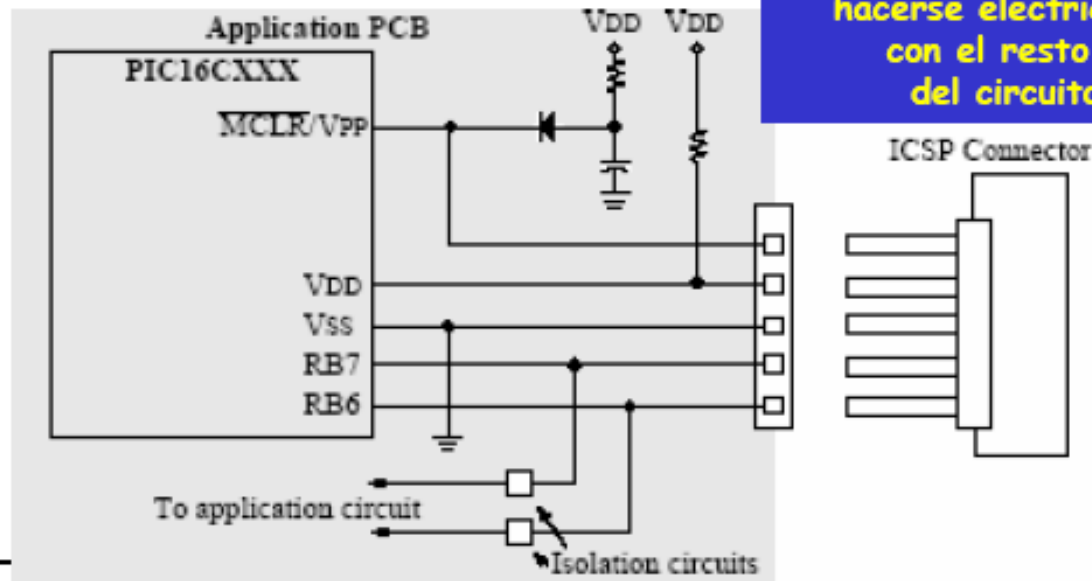
Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other RESETS
0Bh, 8Bh, 10Bh, 18Bh	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	R0IF	0000 000x	0000 000u
0Ch	PIR1	PSPIF <sup>(1)</sup>	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
18h	RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	0000 000x
1Ah	RCREG	USART Receive Register								0000 0000	0000 0000
8Ch	PIE1	PSPIE <sup>(1)</sup>	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
98h	TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010
99h	SPBRG	Baud Rate Generator Register								0000 0000	0000 0000



## Programación Serie En el Circuito (ICSP In Circuit Serial Programming)

- Los microcontroladores que disponen de esta capacidad (p.e. PIC16F877) pueden ser programados vía serie una vez insertados en la aplicación final. Esto permite montar los equipos con los microcontroladores "limpios" y luego grabarlos, también permitiría actualizar el firmware de manera simple.
- Para conseguir esa programación, se necesitan **dos líneas de comunicación serie**, una para **datos** (RB7) y otra para **reloj** (RB6) y otras tres para **alimentación** (VDD), **masa** (VSS) y **tensión de programación** (VPP).

Los pines usados para ICSP externa deben hacerse eléctricamente compatibles con el resto de las conexiones del circuito de la aplicación

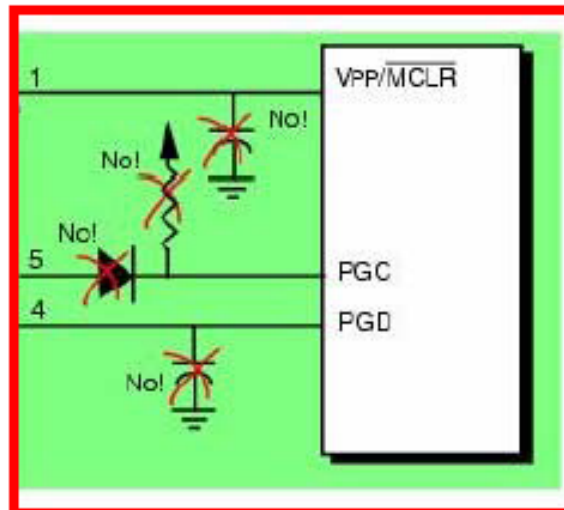




## Programación Serie En el Circuito (ICSP In Circuit Serial Programming)

- Se entra en el modo de Programación/Verificación **manteniendo RB6 y RB7 en estado bajo** mientras **sube la tensión del pin VPP/MCLR de  $V_{IL}$  a  $V_{IHH}$**  teniendo en el terminal VDD la **tensión de programación**.
- A partir de ahí, **los datos a grabar** entran vía serie y de manera síncrona: en RB6 aparece la señal de reloj y por RB7 los datos que se van cargando
- Para permitir la entrada en ese modo de funcionamiento, se necesita que **el resto de los componentes de la tarjeta de la aplicación** permitan ese tipo de transiciones

**$V_{IHH}$  es un parámetro eléctrico que depende del microcontrolador, en el caso de los PIC16F87x debe estar entre  $V_{DD}+3,5V$  y  $13,5V$  para entrar en modo programación**



Este modo es el que utiliza el MPLAB-ICD2 para transferir los programas al microcontrolador, tanto en modo "debugger" como en modo "programmer"





## Programación Serie En el Circuito a Baja Tensión (LV-ICSP Low Voltage In Circuit Serial Programming)

- Este modo, permite la programación en serie (ICSP) del microcontrolador, **empleando una única tensión continua VDD** y en todo el rango de funcionamiento del microcontrolador. **VPP no es necesario que suba hasta el valor  $V_{IHH}$**  (era 3,5v por encima de VDD como mínimo y 13,5V como máximo), puede mantenerse en el nivel normal de operación.
- Este modo **debe habilitarse mediante uno de los bits de configuración (LVP)** que debe estar a "uno" y que así se encuentra por defecto tras el borrado.
- En este modo, el pin **RB3/PGM deja de ser un pin de entrada/salida del puerto B** y se utiliza para funciones de programación. Se entra en modo de programación si se sitúa la tensión VDD en ese terminal RB3/PGM, esa misma tensión debe estar presente en el terminal MCLR
- Aunque esté activado el modo LV-ICSP, **el modo ICSP "normal"** o de "alta tensión" también **estará operativo** y se podrá entrar en ese modo si se producen las transiciones definidas en su momento.
- Si se configura el bit LVP=1, se **debe evitar que el pin RB3/PGM** quede a una tensión "flotante" porque **podría provocar que se entrara en ese modo de programación** durante la ejecución normal de un programa en el microcontrolador y que la memoria del microcontrolador se reprogramara.



MODO DORMIDO ("SLEEP")

- Los microcontroladores PIC pueden trabajar en dos modos distintos:
  - ❖ Modo Normal: ejecutando las instrucciones
  - ❖ Modo **Dormido** o de **bajo consumo**: se suspende la ejecución
- El **consumo de un microcontrolador** depende de su **frecuencia de trabajo**, a más frecuencia más consumo (por carga y descarga de capacidades internas y externas)
- El **modo dormido** supone un ahorro de consumo porque el **oscilador del microcontrolador deja de oscilar**, por tanto **no se ejecutan instrucciones**.
- Puede ser interesante su uso en **aplicaciones portátiles** (alimentadas desde **baterías** o **paneles solares**) si el microcontrolador no va hacer nada durante un periodo de tiempo dado y **en espera de que pase algo** que lo "despierte".
- En este modo "dormido" **se entra por software cuando se ejecuta la instrucción SLEEP**.



### MODO DORMIDO ("SLEEP")

- Al entrar en modo dormido, el bit  $\overline{PD}$  (STATUS<3>) se pone a 0 y el bit  $\overline{TO}$  (STATUS<4>) se pone a 1, estos bits indican que se entró en ese modo para conocimiento posterior en tiempo de ejecución.
- A continuación el oscilador deja de oscilar. Los pines asociados a Puertos de Entrada/Salida mantienen el valor previo a la ejecución de la instrucción SLEEP.
- Si está habilitado el WATCHDOG (en la palabra de configuración), su temporizador se pondrá a cero al ejecutar la instrucción SLEEP, pero se mantendrá "corriendo" y podría desbordar ya que el Watchdog tiene un oscilador RC independiente del propio del microcontrolador.
- Para asegurar un consumo mínimo de corriente en este modo, se aconseja colocar los pines en los estados 1 ó 0 de forma que ningún circuito externo al microcontrolador tenga consumo de la alimentación (si es posible), apagar el convertor A/D, deshabilitar todos los relojes (ponerlos a 1 ó a 0) y deshabilitar las resistencias de pull-up del PORTB
- El pin  $\overline{MCLR}$  debe estar a nivel alto para evitar un Reset externo.

## ¿COMO SE SALE DEL MODO DORMIDO?

El microcontrolador puede salir del modo de bajo consumo por alguno de los siguientes motivos:

1. Un RESET externo provocado en el pin  $\overline{\text{MCLR}}$ .
2. Desbordamiento del WATCHDOG.
3. Interrupción o "cuasi-interrupción" provocada por algún evento de los periféricos que pueden generarlos sin la presencia del oscilador.



El  $\overline{\text{MCLR}}$  RESET causará un RESET del dispositivo (salto a 0x0000), pero las otras dos formas de sacar al microcontrolador del modo dormido únicamente provocan un "despertar" y la continuación del programa con la instrucción que sigue a SLEEP.

En el supuesto de que se haya despertado por una "cuasi-interrupción" y la máscara global de interrupciones esté a 1, tras ejecutar la instrucción que sigue a SLEEP, se continuará con la ejecución de la primera instrucción de la rutina de interrupción (posición de memoria 0x0004).



## ¿COMO SE SALE DEL MODO DORMIDO (2)?

• Mientras que la instrucción SLEEP está siendo ejecutada, la siguiente instrucción (PC+1) ya se coloca en el registro de instrucciones.

• Para "despertar" al microcontrolador a través de un evento que puede provocar una interrupción, el bit de habilitación de la interrupción correspondiente debe estar a 1 (cuasi-interrupción).

• El "despertar" del microcontrolador se produce independientemente del estado de la máscara GIE.

• Colocar una instrucción NOP después de la instrucción SLEEP suele ser habitual si no se desea hacer nada tras un SLEEP y se va a saltar a la rutina de interrupción (0x0004).

