

# Cartilla para Microcontrolador PIC16F877

## 28/40 Pin 8-Bit CMOS FLASH.

### Características principales

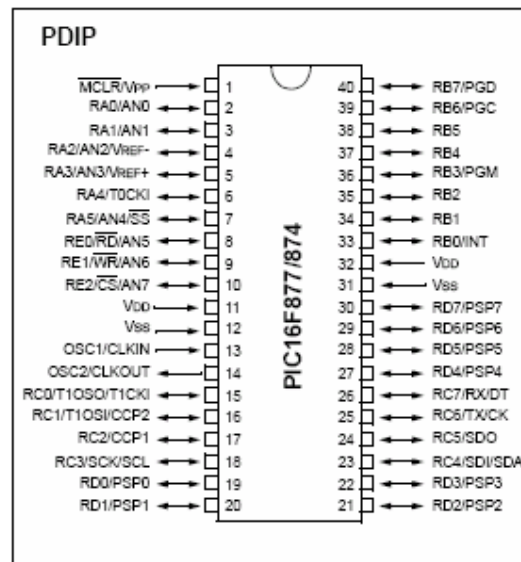
- CPU RISC de alta performance
- Set de 35 instrucciones
- Todas las instrucciones son de un ciclo salvo aquellas que incluyen saltos que son de 2 ciclos.
- Velocidad de Trabajo:
  - DC - 20 MHz clock input
  - DC - 200 ns ciclo de instrucción
- Hasta 8K x 14 words de FLASH Program Memory,
- Hasta 368 x 8 bytes de Data Memory (RAM)
- Hasta 256 x 8 bytes de EEPROM Data Memory
- Manejo de Interrupciones (hasta 14 fuentes)
- Stack de hardware de 8 niveles
- Modo de direccionamiento directo, indirecto y relativo.
- Power-on Reset (POR)
- Power-up Timer (PWRT) y Oscillator Start-up Timer (OST)
- Watchdog Timer (WDT) con el reloj RC interno para mejor seguridad.
- Protección de código programable.
- Programación serial via 2 pines, In-Circuit Serial Programming<sup>®</sup> (ICSP)
- In-Circuit Debugging via 2 pines
- Amplio rango de voltaje de trabajo: 2.0V a 5.5V

### Características periféricas

- Timer0: 8-bit timer/counter con 8-bit prescaler
- Timer1: 16-bit timer/counter con prescaler, que puede ser incrementado durante el modo SLEEP via reloj externo.
- Timer2: 8-bit timer/counter con registro de período de 8-bit, prescaler y postscaler
- Dos módulos Capture, Compare, PWM
  - Capture es de 16-bit, max. resolución: 12.5 ns
  - Compare es de 16-bit, max. resolución: 200 ns
  - PWM max. resolución: 10-bit
- Convertidor Analógico a Digital de 10-bit multi-channel
- Synchronous Serial Port (SSP) con SPI (Master mode) e I2C (Master/Slave)
- Universal Synchronous Asynchronous Receiver Transmitter (USART/SCI) con detección de direcciones de 9-bit
- Parallel Slave Port (PSP) de 8-bits de ancho, con controles externos de RD, WR y CS (solo 40/44-pin)
- Brown-out detection circuitry para Brown-out Reset (BOR)

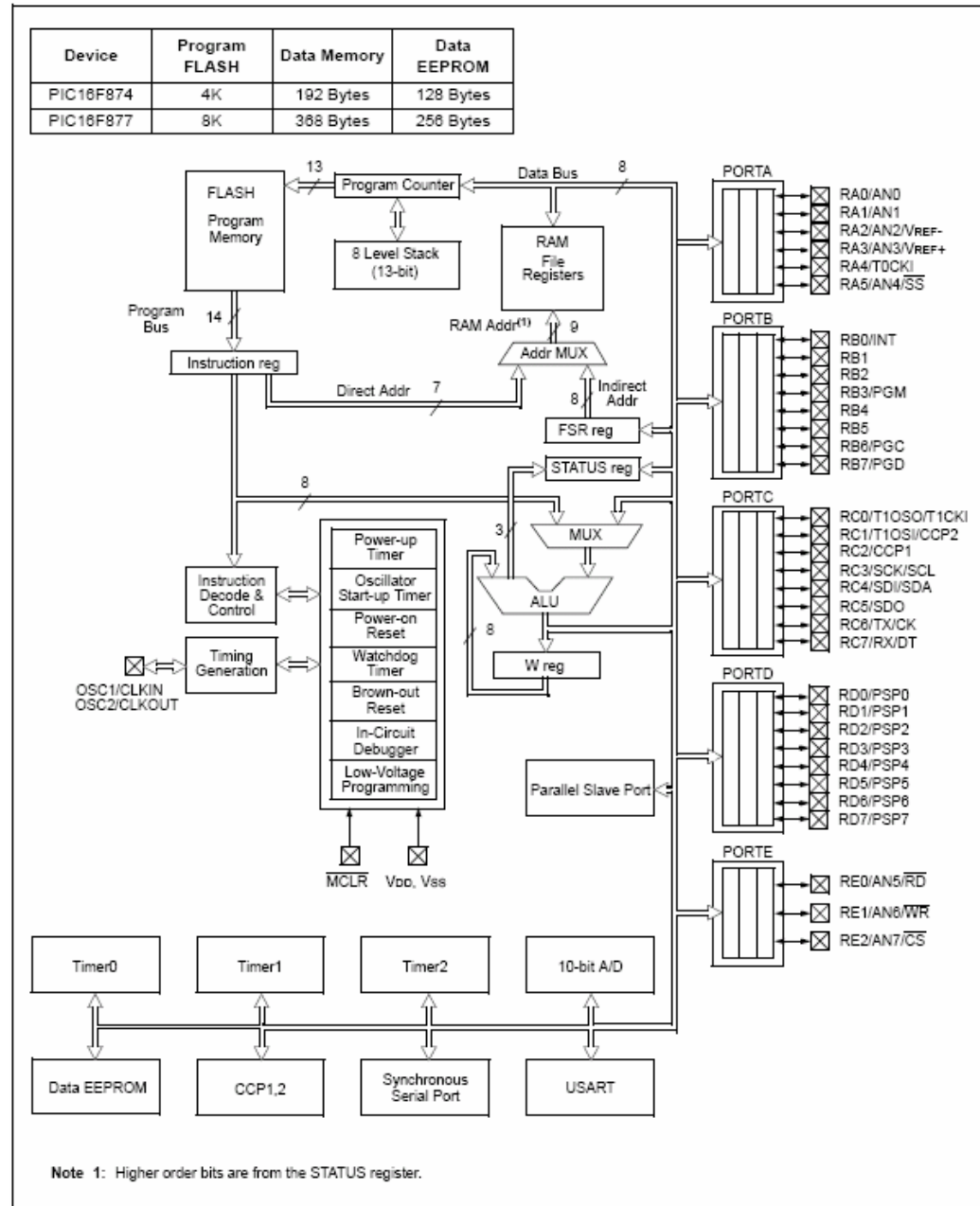
### Diagrama de Pins

40-Lead Plastic Dual In-line (P) – 600 mil (PDIP)



## Arquitectura

**Figura 1: Diagrama de bloque del PIC16F874 y PIC16F877**



## Organización de la memoria

Existen tres bloques de memoria dentro de un PIC16F87X. La memoria de programa y la memoria de datos tienen buses separados por lo que es posible el acceso a las mismas en forma concurrente. El tercer bloque de memoria es la memoria de datos EEPROM.

### Organización de la memoria de programa

El dispositivo PIC16F87X tiene un program counter de 13-bit capaz de direccionar 8K x 14 direcciones de memoria. Los dispositivos PIC16F877/876 tienen 8K x 14 palabras de FLASH program memory y los dispositivos PIC16F873/874 tienen 4K x 14.

El vector de RESET (por donde comienza a ejecutar el PIC) está en la dirección 0000h y el vector de interrupciones se encuentra en la dirección 0004h.

### Organización de la memoria de datos

La memoria de datos está particionada en múltiples bancos que contienen los registros de propósito general (General Purpose Registers) y los registros de funciones especiales (Special Function Registers).

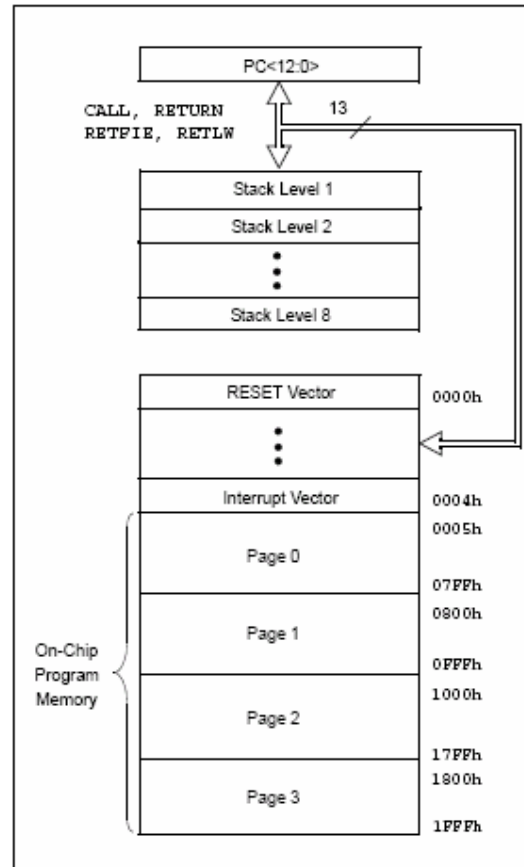
RP1:RP0	Bank
00	0
01	1
10	2
11	3

El bit RP1 (STATUS <6>) y RP0 (STATUS <5>) son los bits de selección de banco. Cada banco se extiende hasta 7Fh (128 bytes). Los lugares más bajos de cada banco están reservados para registros de funciones especiales. Debajo de estos se encuentran los registros de propósito general, implementados como RAM estática. Todos los bancos implementados contienen registros de funciones especiales. Algunos de los registros de funciones especiales que son accedidos frecuentemente desde un banco pueden ser espejados en otro banco para reducir el código y lograr un acceso más rápido a los mismos.

### Registros de funciones especiales

Los registros de funciones especiales son usados por la CPU y los módulos periféricos para controlar las operaciones del dispositivo. Los registros de funciones especiales pueden ser clasificados en dos conjuntos: core (CPU) y periféricos.

**Figura 2: Mapa de la memoria de programa y STACK del PIC16F877/76**



**Figura 3: Mapa del archivo de registros PIC16F877/876**

File Address		File Address		File Address		File Address	
Indirect addr. <sup>(*)</sup>	00h	Indirect addr. <sup>(*)</sup>	80h	Indirect addr. <sup>(*)</sup>	100h	Indirect addr. <sup>(*)</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		105h		185h
PORTB	06h	TRISB	86h	PORTB	106h	TRISB	186h
PORTC	07h	TRISC	87h		107h		187h
PORTD <sup>(*)</sup>	08h	TRISD <sup>(*)</sup>	88h		108h		188h
PORTE <sup>(*)</sup>	09h	TRISE <sup>(*)</sup>	89h		109h		189h
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		8Fh	EEADRH	10Fh	Reserved <sup>(2)</sup>	18Fh
T1CON	10h		90h	General Purpose Register 16 Bytes	110h	General Purpose Register 16 Bytes	190h
TMR2	11h	SSPCON2	91h		111h		191h
T2CON	12h	PR2	92h		112h		192h
SSPBUF	13h	SSPADDD	93h		113h		193h
SSPCON	14h	SSPSTAT	94h		114h		194h
CCPR1L	15h		95h		115h		195h
CCPR1H	16h		96h		116h		196h
CCP1CON	17h		97h		117h		197h
RCSTA	18h	TXSTA	98h		118h		198h
TXREG	19h	SPBRG	99h		119h		199h
RCREG	1Ah		9Ah		11Ah		19Ah
CCPR2L	1Bh		9Bh		11Bh		19Bh
CCPR2H	1Ch		9Ch		11Ch		19Ch
CCP2CON	1Dh		9Dh		11Dh		19Dh
ADRESH	1Eh	ADRESL	9Eh		11Eh		19Eh
ADCON0	1Fh	ADCON1	9Fh		11Fh		19Fh
General Purpose Register 96 Bytes	20h	General Purpose Register 80 Bytes	A0h	General Purpose Register 80 Bytes	120h	General Purpose Register 80 Bytes	1A0h
	7Fh	accesses 70h-7Fh	EFh F0h FFh	accesses 70h-7Fh	16Fh 170h 17Fh	accesses 70h - 7Fh	1EFh 1F0h 1FFh
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.

Tabla 1: Resumen de los registros de funciones especiales

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Details on page:		
Bank 0													
00h <sup>(R)</sup>	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	27		
01h	TMR0	Timer0 Module Register								xxxxxx xxxxxx	47		
02h <sup>(R)</sup>	PCL	Program Counter (PC) Least Significant Byte								0000 0000	26		
03h <sup>(R)</sup>	STATUS	IRP	RP1	RP0	T0	PD	Z	DC	C	0001 1xxxxx	18		
04h <sup>(R)</sup>	FSR	Indirect Data Memory Address Pointer								xxxxxx xxxxxx	27		
05h	PORTA	—		—		PORTA Data Latch when written: PORTA pins when read						--0x 0000	29
06h	PORTB	PORTB Data Latch when written: PORTB pins when read								xxxxxx xxxxxx	31		
07h	PORTC	PORTC Data Latch when written: PORTC pins when read								xxxxxx xxxxxx	33		
08h <sup>(R)</sup>	PORTD	PORTD Data Latch when written: PORTD pins when read								xxxxxx xxxxxx	35		
09h <sup>(R)</sup>	PORTE	—	—	—	—	—	RE2	RE1	RE0	---- -xxxxx	36		
0Ah <sup>(1,3)</sup>	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter						---0 0000	26	
0Bh <sup>(R)</sup>	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	20		
0Ch	PIR1	PSPIF <sup>(R)</sup>	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	22		
0Dh	PIR2	—	(5)	—	EEIF	BCLIF	—	—	CCP2IF	-x-0 0---0	24		
0Eh	TMR1L	Holding register for the Least Significant Byte of the 16-bit TMR1 Register								xxxxxx xxxxxx	52		
0Fh	TMR1H	Holding register for the Most Significant Byte of the 16-bit TMR1 Register								xxxxxx xxxxxx	52		
10h	T1CON	—	—	T1CKPS1	T1CKPS0	T1OSCEN	T1SYNC	TMR1CS	TMR1ON	--00 0000	51		
11h	TMR2	Timer2 Module Register								0000 0000	55		
12h	T2CON	—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	-000 0000	55		
13h	SSPBUF	Synchronous Serial Port Receive Buffer/Transmit Register								xxxxxx xxxxxx	70, 73		
14h	SSPCON	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	67		
15h	CCPR1L	Capture/Compare/PWM Register1 (LSB)								xxxxxx xxxxxx	57		
16h	CCPR1H	Capture/Compare/PWM Register1 (MSB)								xxxxxx xxxxxx	57		
17h	CCP1CON	—	—	CCP1X	CCP1Y	CCP1M3	CCP1M2	CCP1M1	CCP1M0	--00 0000	58		
18h	RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OCERR	RX9D	0000 000x	96		
18h	TXREG	USART Transmit Data Register								0000 0000	99		
1Ah	RCREG	USART Receive Data Register								0000 0000	101		
1Bh	CCPR2L	Capture/Compare/PWM Register2 (LSB)								xxxxxx xxxxxx	57		
1Ch	CCPR2H	Capture/Compare/PWM Register2 (MSB)								xxxxxx xxxxxx	57		
1Dh	CCP2CON	—	—	CCP2X	CCP2Y	CCP2M3	CCP2M2	CCP2M1	CCP2M0	--00 0000	58		
1Eh	ADRESH	A/D Result Register High Byte								xxxxxx xxxxxx	116		
1Fh	ADCON0	ADCS1	ADCS0	CHS2	CHS1	CHS0	GO/DONE	—	ADON	0000 00-0	111		
Bank 1													
80h <sup>(R)</sup>	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	27		
81h	OPTION_REG	RBPV	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	19		
82h <sup>(R)</sup>	PCL	Program Counter (PC) Least Significant Byte								0000 0000	26		
83h <sup>(R)</sup>	STATUS	IRP	RP1	RP0	T0	PD	Z	DC	C	0001 1xxxxx	18		
84h <sup>(R)</sup>	FSR	Indirect Data Memory Address Pointer								xxxxxx xxxxxx	27		
85h	TRISA	—	—	PORTA Data Direction Register								--11 1111	29
86h	TRISB	PORTB Data Direction Register								1111 1111	31		
87h	TRISC	PORTC Data Direction Register								1111 1111	33		
88h <sup>(R)</sup>	TRISD	PORTD Data Direction Register								1111 1111	35		
89h <sup>(R)</sup>	TRISE	IBF	OBF	IBOV	PSPMODE	—	PORTE Data Direction Bits			0000 -111	37		
8Ah <sup>(1,3)</sup>	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter						---0 0000	26	
8Bh <sup>(R)</sup>	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	20		
8Ch	PIE1	PSPIE <sup>(R)</sup>	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	21		
8Dh	PIE2	—	(5)	—	EEIE	BCLIE	—	—	CCP2IE	-x-0 0--0	23		
8Eh	PCON	—	—	—	—	—	—	POR	BOR	---- --gg	25		
8Fh	—	Unimplemented								—	—		
90h	—	Unimplemented								—	—		
91h	SSPCON2	GCEN	ACKSTAT	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	0000 0000	68		
92h	PR2	Timer2 Period Register								1111 1111	55		
93h	SSPAD	Synchronous Serial Port (I <sup>2</sup> C mode) Address Register								0000 0000	73, 74		
94h	SSPSTAT	SMP	CKE	D/A	P	S	R/W	UA	BF	0000 0000	66		
95h	—	Unimplemented								—	—		
96h	—	Unimplemented								—	—		
97h	—	Unimplemented								—	—		
98h	TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	95		
99h	SPBRG	Baud Rate Generator Register								0000 0000	97		
9Ah	—	Unimplemented								—	—		
9Bh	—	Unimplemented								—	—		
9Ch	—	Unimplemented								—	—		
9Dh	—	Unimplemented								—	—		
9Eh	ADRESL	A/D Result Register Low Byte								xxxxxx xxxxxx	116		
9Fh	ADCON1	ADFM	—	—	—	PCFG3	PCFG2	PCFG1	PCFG0	0--- 0000	112		



**Tabla 2: Resumen de los registros de funciones especiales (Continuación)**

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Details on page:
Bank 2											
100h <sup>(a)</sup>	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	27
101h	TMR0	Timer0 Module Register								xxxxx xxxxx	47
102h <sup>(a)</sup>	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	26
103h <sup>(a)</sup>	STATUS	IRP	RP1	RP0	TO	PD	Z	DC	C	0001 1xxxx	18
104h <sup>(a)</sup>	FSR	Indirect Data Memory Address Pointer								xxxxx xxxxx	27
105h	—	Unimplemented								—	—
106h	PORTB	PORTB Data Latch when written: PORTB pins when read								xxxxx xxxxx	31
107h	—	Unimplemented								—	—
108h	—	Unimplemented								—	—
109h	—	Unimplemented								—	—
10Ah <sup>(1,3)</sup>	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter					---0 0000	26
10Bh <sup>(a)</sup>	INTCON	GIE	PEIE	TOIE	INTE	RBIE	TOIF	INTF	RBIF	0000 000x	20
10Ch	EEDATA	EEPROM Data Register Low Byte								xxxxx xxxxx	41
10Dh	EEADR	EEPROM Address Register Low Byte								xxxxx xxxxx	41
10Eh	EEDATH	—	—	EEPROM Data Register High Byte					xxxxx xxxxx	41	
10Fh	EEADRH	—	—	—	EEPROM Address Register High Byte					xxxxx xxxxx	41
Bank 3											
180h <sup>(a)</sup>	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	27
181h	OPTION_REG	RBP1	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	19
182h <sup>(a)</sup>	PCL	Program Counter (PC) Least Significant Byte								0000 0000	26
183h <sup>(a)</sup>	STATUS	IRP	RP1	RP0	TO	PD	Z	DC	C	0001 1xxxx	18
184h <sup>(a)</sup>	FSR	Indirect Data Memory Address Pointer								xxxxx xxxxx	27
185h	—	Unimplemented								—	—
186h	TRISB	PORTB Data Direction Register								1111 1111	31
187h	—	Unimplemented								—	—
188h	—	Unimplemented								—	—
189h	—	Unimplemented								—	—
18Ah <sup>(1,3)</sup>	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter					---0 0000	26
18Bh <sup>(a)</sup>	INTCON	GIE	PEIE	TOIE	INTE	RBIE	TOIF	INTF	RBIF	0000 000x	20
18Ch	EECON1	EEPGD	—	—	—	WRERR	WREN	WR	RD	x--- x000	41, 42
18Dh	EECON2	EEPROM Control Register2 (not a physical register)								---- ----	41
18Eh	—	Reserved maintain clear								0000 0000	—
18Fh	—	Reserved maintain clear								0000 0000	—

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

Note 1: The upper byte of the program counter is not directly accessible. PCLATH is a holding register for the PC<12:8> whose contents are transferred to the upper byte of the program counter.  
2: Bits PSPIE and PSPIF are reserved on PIC16F873/876 devices; always maintain these bits clear.  
3: These registers can be addressed from any bank.  
4: PORTD, PORTE, TRISD, and TRISE are not physically implemented on PIC16F873/876 devices; read as '0'.  
5: PIR2<6> and PIE2<6> are reserved on these devices; always maintain these bits clear.

**Registro: STATUS**

El registro STATUS contiene el estado de la aritmética de la ALU, es estado del RESET y el banco elegido para los datos de la memoria (SRAM).

Puede ser el destino de cualquier instrucción como cualquier otro registro. Si el registro de estado es destino de una instrucción los bits Z, DC o C son afectados.

Los bits TO y PD son sólo de lectura.

**STATUS REGISTER (ADDRESS 03h, 83h, 103h, 183h)**

R/W-0	R/W-0	R/W-0	R-1	R-1	R/W-x	R/W-x	R/W-x
IRP	RP1	RP0	TO	PD	Z	DC	C
bit 7							bit 0

bit 7 IRP: Registro de selección de Banco.(usado para direccionamiento indirecto)

1 = Bank 2, 3 (100h - 1FFh)

0 = Bank 0, 1 (00h - FFh)

bit 6-5 RP1:RP0: Registro de selección de banco (usado para direccionamiento directo)

00 = Bank 0 (00h - 7Fh)

01 = Bank 1 (80h - FFh)

10 = Bank 2 (100h - 17Fh)

11 = Bank 3 (180h - 1FFh)

bit 4 TO: bit de Timeout

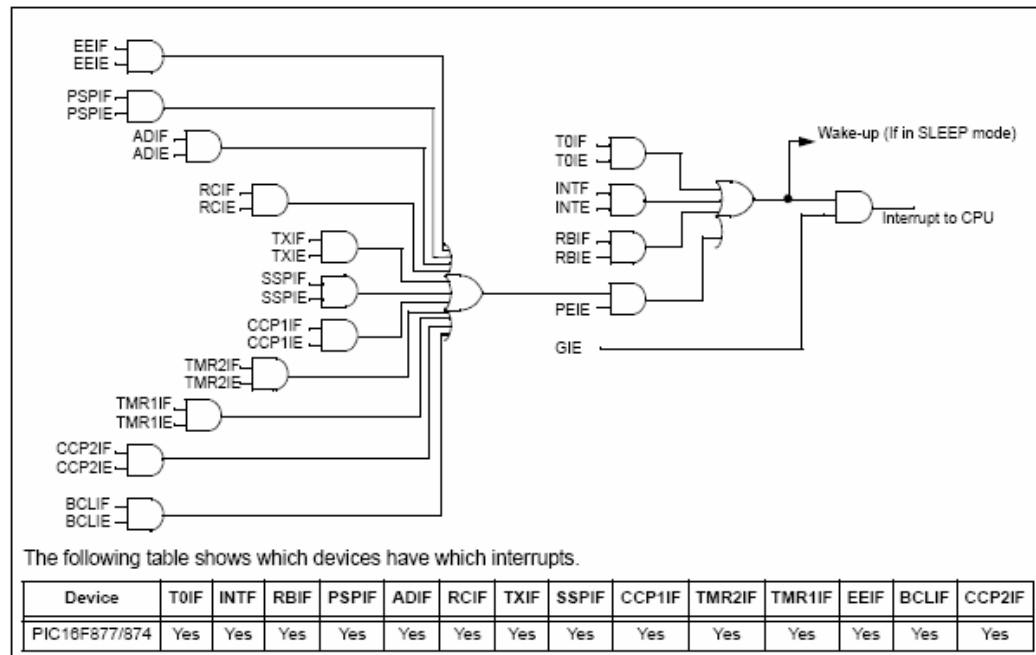
- 1 = Después de encender, una instrucción CLRWD T, o una instrucción SLEEP  
 0 = ocurrió un timeout de WDT
- bit 3 PD: bit de Apagado  
 1 = Después de encendido o por una instrucción CLRWD T  
 0 = Por una ejecución de la instrucción SLEEP
- bit 2 Z: Zero bit  
 1 = El resultado de una operación aritmética fue 0.  
 0 = El resultado de una operación aritmética no fue 0.
- bit 1 DC: Dígito de acarreo (para instrucciones ADDWF, ADDLW, SUBLW, SUBWF instructions)  
 1 = Hubo acarreo del cuarto bit de orden bajo en el resultado.  
 0 = No hubo acarreo del cuarto bit de orden bajo en el resultado.
- bit 0 C: bit de acarreo o préstamo (instrucciones ADDWF, ADDLW, SUBLW, SUBWF)  
 1 = Ocurrió acarreo en el bit más significativo del resultado.  
 0 = No ocurrió acarreo en el bit más significativo del resultado.

### Registro: INTCON

El registro INTCON es un registro legible y escribible que contiene las llaves para habilitar las fuentes de interrupción.

Nota: las banderas de interrupción se setearán independientemente de la forma en que esté seteado el INTCON, pero si el GIE no está habilitado la interrupción no se provocará.

**Figura 4: Lógica de Interrupción**



### INTCON REGISTER (ADDRESS 0Bh, 8Bh, 10Bh, 18Bh)

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	TBIE	TOIF	INTF	TBIF
bit 7							bit 0

bit 7 GIE: bit de interrupciones globales

1 = Habilita las interrupciones no enmascaradas.

0 = Deshabilita las interrupciones no enmascaradas.

bit 6 PEIE: bit de Interrupciones periféricas

1 = Habilita las interrupciones periféricas no enmascaradas.

0 = Deshabilita las interrupciones periféricas no enmascaradas.

bit 5 TOIE: bit de habilitación de interrupciones por overflow del timer 0.

- 1 = Habilita interrupciones por TMR0
- 0 = Deshabilita interrupciones por TMR0
- bit 4 INTE: bit de habilitación de la interrupción externa en RB0
  - 1 = Habilita interrupciones externas en RB0
  - 0 = Deshabilita interrupciones externas en RB0
- bit 3 RBIE: bit de habilitación de interrupciones por cambios en el puerto B.
  - 1 = Habilita interrupciones por cambio en puertoB
  - 0 = Deshabilita interrupciones por cambio en puertoB
- bit 2 T0IF: bit bandera de interrupción por TMR0
  - 1 = en el TMR0 ha ocurrido overflow (esta bandera debe volverse a 0 en software)
  - 0 = en el TMR0 no ha ocurrido overflow.
- bit 1 INTF: bit bandera de interrupción externa en RB0
  - 1 = ha ocurrido una interrupción externa por RB0 (esta bandera debe volverse a 0 en software)
  - 0 = no ha ocurrido una interrupción externa por RB0
- bit 0 RBIF: bit bandera de interrupción por cambio en el puerto B
  - 1 = Al menos una de las patas entre RB7:RB4 a cambiado de valor (esta bandera debe volverse a 0 en software)
  - 0 = Ninguna de las patas entre RB7:RB4 ha cambiado de valor

### CONFIGURATION WORD (ADDRESS 2007h)

CP1	CP0	CP1	CP0	-	CPD	LVP	BODEN	MCLR	FOSC2	PWRTEN	WDTE	FOSC1	FOSC0
bit 13													bit 0

- bit 13-10: CP1:CP0: bits de Protección del Código
  - Protección del código para memoria de programa de 2K
    - 11 = Protección de código desactivada.
    - 10 = 0400h-07FFh código protegido
    - 01 = 0200h-07FFh código protegido
    - 00 = 0000h-07FFh código protegido
  - Protección de código para memorias de programa de 1K
    - 11 = Protección de código desactivada.
    - 10 = Protección de código desactivada.
    - 01 = 0200h-03FFh código protegido
    - 00 = 0000h-03FFh código protegido
- bit 9: Sin implementar: Se lee como '0'
- bit 8: CPD: bits de Protección de datos
  - 1 = memoria de datos sin protección
  - 0 = memoria de datos protegida.
- bit 7: LVP: Habilita la programación de bajo voltaje
  - 1 = la pata RB4/PGM tiene la función PGM habilitada.
  - 0 = la pata RB4/PGM es digital I/O, HV en MCLR debe ser usado para la programación
- bit 6: BODEN: bit de habilitación de Reinicio por detección de Brown-out.
  - 1 = Reseteo por BOD habilitado.
  - 0 = Reseteo por BOD inhabilitado.
- bit 5: MCLR: pin de selección de función RA5/MCLR.
  - 1 = RA5/MCLR es master clear.
  - 0 = RA5/MCLR es entrada digital (internamente el master clear se pone a VCC).
- bit 3: PWRTEN: bit de habilitación de Timer de Power-up
  - 1 = PWRT inhabilitado
  - 0 = PWRT habilitado
- bit 2: WDTE: bit de habilitación del Watchdog
  - 1 = WDT habilitado
  - 0 = WDT deshabilitado
- bit 4, 1-0: FOSC2:FOSC0: Selección del oscilador
  - 111 = oscilador ER: función CLKOUT en la pata RA6/OSC2/CLKOUT , Resistor en RA7/OSC1/CLKIN
  - 110 = oscilador ER : función I/O en la pata RA6/OSC2/CLKOUT, Resistor en RA7/OSC1/CLKIN



101 = oscilador INTRC: función CLKOUT en la pata RA6/OSC2/CLKOUT, función I/O en RA7/OSC1/CLKIN  
 100 = oscilador INTRC: función I/O en la pata RA6/OSC2/CLKOUT, función I/O en RA7/OSC1/CLKIN  
 011 = oscilador EC: función I/O en la pata RA6/OSC2/CLKOUT, CLKIN en RA7/OSC1/CLKIN  
 010 = oscilador HS: Cristal de alta velocidad/resonador en RA6/OSC2/CLKOUT y RA7/OSC1/CLKIN  
 001 = oscilador XT: cristal/resonador en RA6/OSC2/CLKOUT y RA7/OSC1/CLKIN  
 000 = oscilador LP: cristal Low power en RA6/OSC2/CLKOUT y RA7/OSC1/CLKIN

## Resumen del Set de instrucciones

Cada instrucción del PIC16F87X es una palabra de 14-bit, dividida en un OPCODE quien especifica la instrucción y uno o más operandos. El set de instrucciones es altamente ortogonal y se encuentra agrupado en tres categorías básicas:

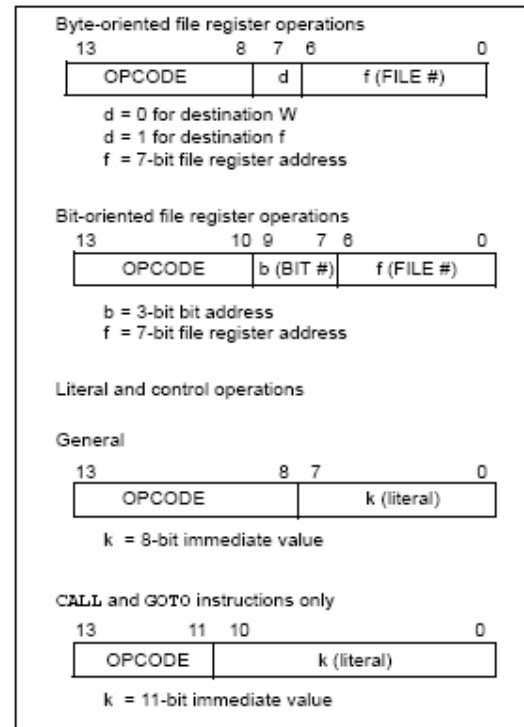
- Operaciones orientadas a Byte (Byte-oriented)
- Operaciones orientadas a bit (Bit-oriented)
- Operaciones Literales y control.

Para las instrucciones Byte-oriented, 'f' representa un designador de registro de archivo y 'd' representa el designador de destino. El designador de archivo especifica que registro debe ser usado por la instrucción. El designador de destino especifica donde colocar el resultado de la operación. Si 'd' es cero, el resultado se coloca en el registro de trabajo W. Si 'd' es uno, el resultado se coloca el registro de archivo especificado en la instrucción.

Para las instrucciones Bit-oriented, 'b' representa un designador de bit quien selecciona el número de bit afectado por la operación, mientras 'f' representa la dirección del archivo en donde se encuentra el bit.

Para las operaciones literales y control, 'k' representa un valor constante o literal de ocho o nueve bit.

**Figura 5: Formato de instrucción**



**Tabla 3: Descripción del campo OPCODE**

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

La figura 5 muestra el formato general que pueden tener las instrucciones.

Todas las instrucciones son ejecutadas dentro de un ciclo de instrucción salvo que la evaluación de una condición sea verdadera o se cambie el program counter como resultado de una instrucción. En este caso la instrucción toma dos ciclos de instrucción donde el segundo ciclo se ejecuta como un NOP. Un ciclo de instrucción consiste en cuatro períodos del oscilador.

La tabla 4 muestra las instrucciones reconocidas por el MPASM<sub>TM</sub> assembler.

**Tabla 4: Set de instrucciones para PIC16F87X**

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

- Note** 1: When an I/O register is modified as a function of itself ( e.g., MOVF PORTB, 1), the value used will be that value present on the pins themselves. For example, if the data latch is '1' for a pin configured as input and is driven low by an external device, the data will be written back with a '0'.
- 2: If this instruction is executed on the TMR0 register (and, where applicable, d = 1), the prescaler will be cleared if assigned to the Timer0 module.
- 3: If Program Counter (PC) is modified, or a conditional test is true, the instruction requires two cycles. The second cycle is executed as a NOP.