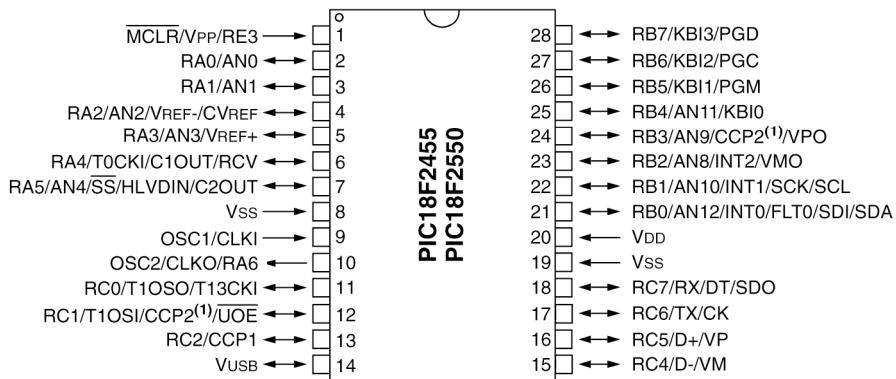
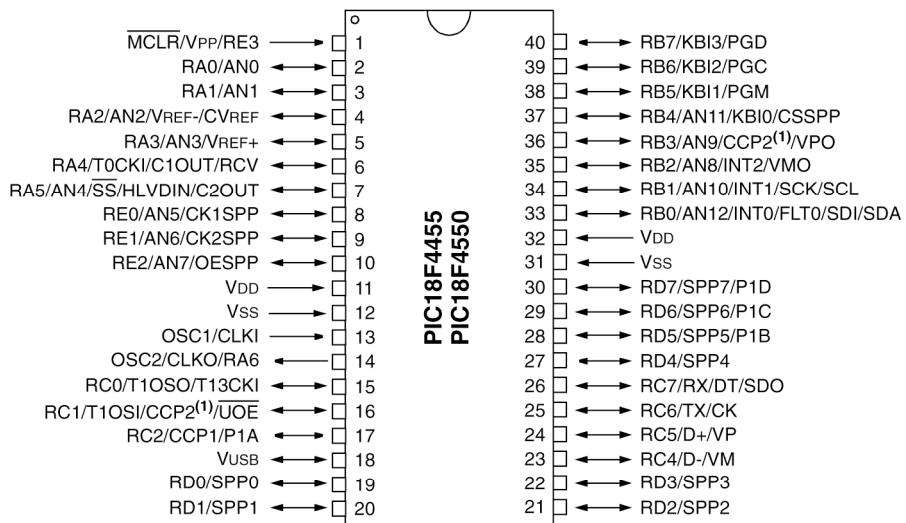


DIAGRAMA DE PINES

28-Pin PDIP, SOIC

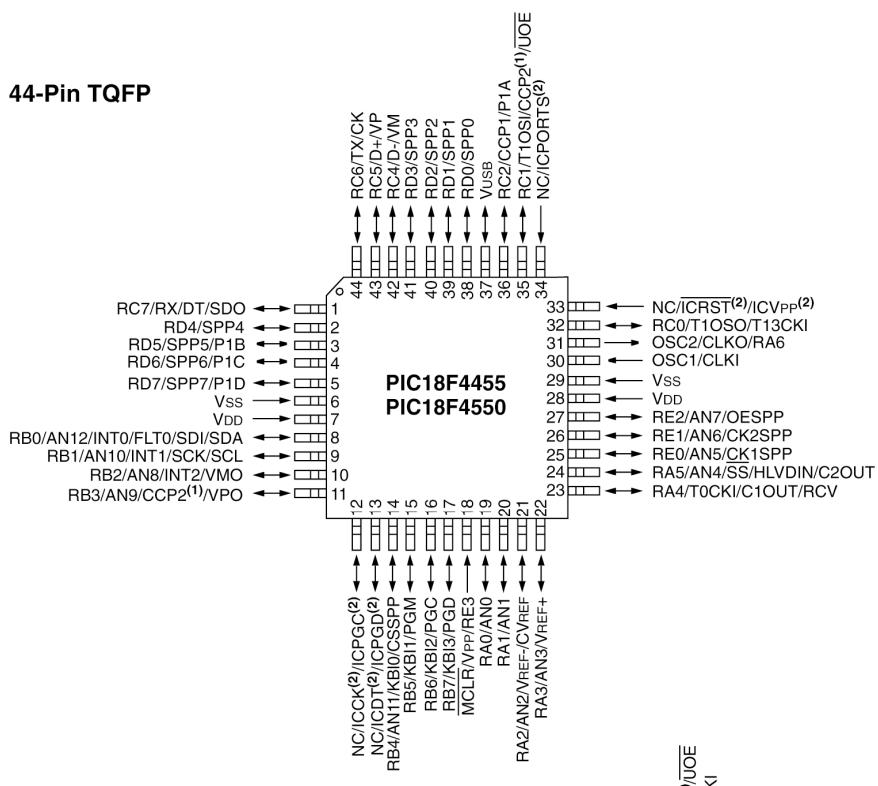


40-Pin PDIP

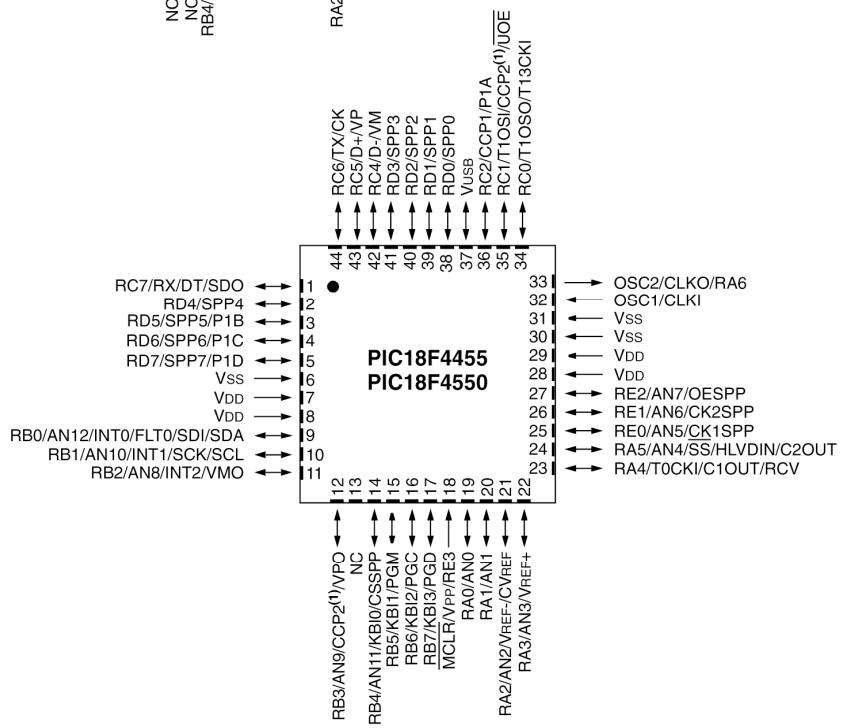


Note 1: RB3 is the alternate pin for CCP2 multiplexing.

44-Pin TQFP



44-Pin QFN



Note 1: RB3 is the alternate pin for CCP2 multiplexing.

2: Special ICPORTS features available in select circumstances. See **Section 25.9 “Special ICPORT Features (Designated Packages Only)”** for more information.

1.0 PERSPECTIVA GENERAL DEL DISPOSITIVO

TABLA 1-1: CARACTERÍSTICAS DEL DISPOSITIVO

Pin Name	Pin Number	Pin Type	Buffer Type	Description
	PDIP, SOIC			
RA0/AN0	2	I/O	TTL	PORTA is a bidirectional I/O port.
RA0		I	Analog	Digital I/O.
AN0				Analog input 0.
RA1/AN1	3	I/O	TTL	Digital I/O.
RA1		I	Analog	Analog input 1.
AN1				
RA2/AN2/VREF-/CVREF	4	I/O	TTL	Digital I/O.
RA2		I	Analog	Analog input 2.
AN2		I	Analog	A/D reference voltage (low) input.
VREF-		O	Analog	Analog comparator reference output.
CVREF				
RA3/AN3/VREF+	5	I/O	TTL	Digital I/O.
RA3		I	Analog	Analog input 3.
AN3		I	Analog	A/D reference voltage (high) input.
VREF+				
RA4/T0CKI/C1OUT/RCV	6	I/O	ST	Digital I/O.
RA4		I	ST	Timer0 external clock input.
T0CKI		O	—	Comparator 1 output.
C1OUT		I	TTL	External USB transceiver RCV input.
RCV				
RA5/AN4/SS/ HLVDIN/C2OUT	7	I/O	TTL	Digital I/O.
RA5		I	Analog	Analog input 4.
AN4		I	TTL	SPI slave select input.
SS		I	Analog	High/Low-Voltage Detect input.
HLVDIN		O	—	Comparator 2 output.
C2OUT				
RA6	—	—	—	See the OSC2/CLKO/RA6 pin.

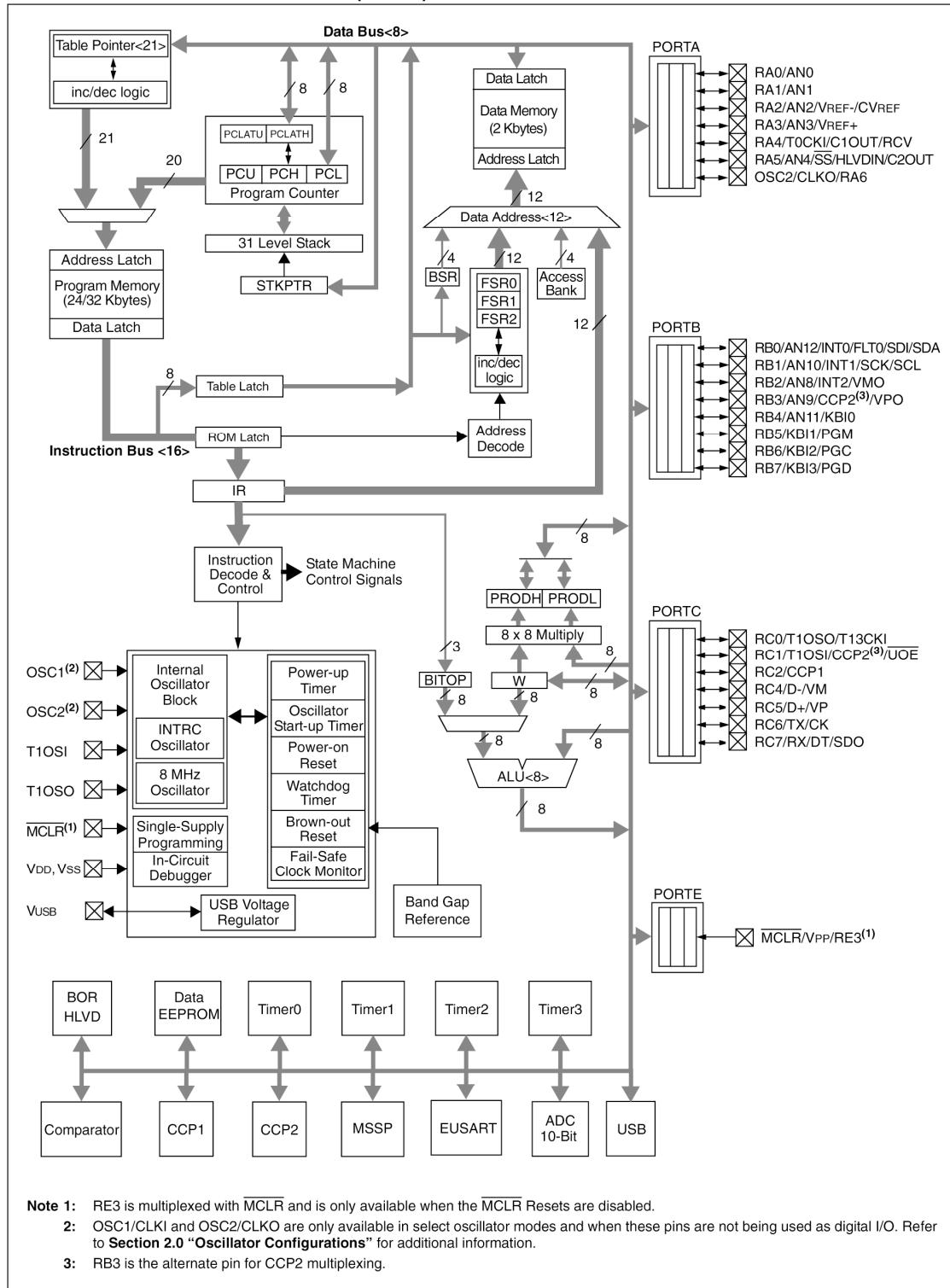
FIGURA 1-1: DIAGRAMA DE BLOQUES DEL PIC18F2455/2550 28PINES

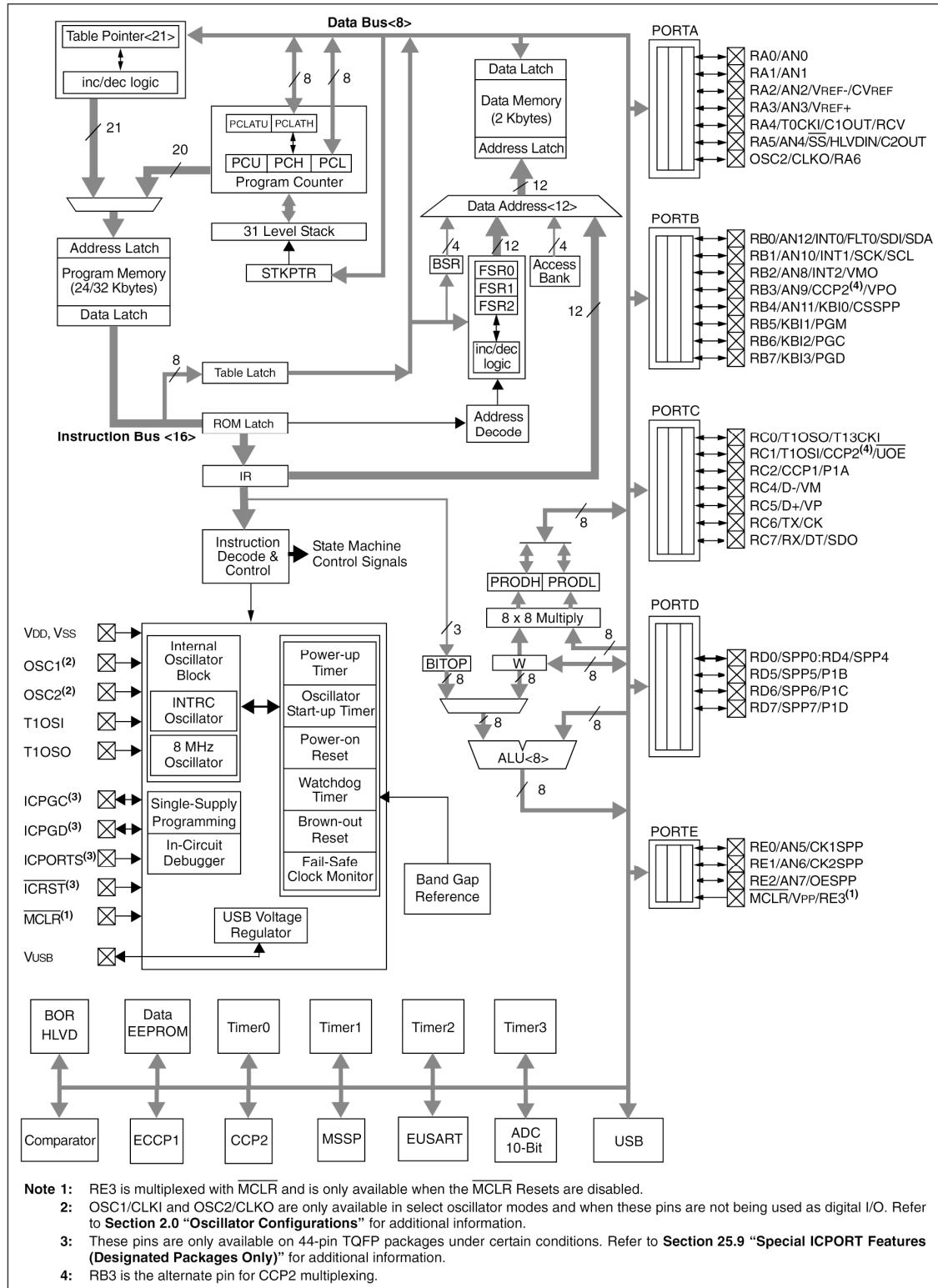
FIGURA 1-2: DIAGRAMA DE BLOQUES DEL PIC18F4455/4550 40/44PINES

TABLA 1-2: DESCRIPCIÓN DE LOS PINES E/S DEL PIC18F2455/2550

Pin Name	Pin Number	Pin Type	Buffer Type	Description
	PDIP, SOIC			
MCLR/VPP/RE3 MCLR	1	I	ST	Master Clear (input) or programming voltage (input). Master Clear (Reset) input. This pin is an active-low Reset to the device.
VPP RE3		P I	ST	Programming voltage input. Digital input.
OSC1/CLKI OSC1 CLKI	9	I I	Analog Analog	Oscillator crystal or external clock input. Oscillator crystal input or external clock source input. External clock source input. Always associated with pin function OSC1. (See OSC2/CLKO pin.)
OSC2/CLKO/RA6 OSC2 CLKO RA6	10	O O I/O	— — TTL	Oscillator crystal or clock output. Oscillator crystal output. Connects to crystal or resonator in Crystal Oscillator mode. In select modes, OSC2 pin outputs CLKO which has 1/4 the frequency of OSC1 and denotes the instruction cycle rate. General purpose I/O pin.
Pin Name	Pin Number	Pin Type	Buffer Type	Description
	PDIP, SOIC			
RA0/AN0 RA0 AN0	2	I/O I	TTL Analog	PORTA is a bidirectional I/O port. Digital I/O. Analog input 0.
RA1/AN1 RA1 AN1	3	I/O I	TTL Analog	Digital I/O. Analog input 1.
RA2/AN2/VREF-/CVREF RA2 AN2 VREF- CVREF	4	I/O I I O	TTL Analog Analog Analog	Digital I/O. Analog input 2. A/D reference voltage (low) input. Analog comparator reference output.
RA3/AN3/VREF+ RA3 AN3 VREF+	5	I/O I I	TTL Analog Analog	Digital I/O. Analog input 3. A/D reference voltage (high) input.
RA4/T0CKI/C1OUT/RCV RA4 T0CKI C1OUT RCV	6	I/O I O I	ST ST — TTL	Digital I/O. Timer0 external clock input. Comparator 1 output. External USB transceiver RCV input.
RA5/AN4/SS/ HLVDIN/C2OUT RA5 AN4 SS HLVDIN C2OUT	7	I/O I I I O	TTL Analog TTL Analog —	Digital I/O. Analog input 4. SPI slave select input. High/Low-Voltage Detect input. Comparator 2 output.
RA6	—	—	—	See the OSC2/CLKO/RA6 pin.

Pin Name	Pin Number	Pin Type	Buffer Type	Description
	PDIP, SOIC			
RB0/AN12/INT0/FLT0/ SDI/SDA	21	I/O	TTL	PORTB is a bidirectional I/O port. PORTB can be software programmed for internal weak pull-ups on all inputs.
RB0		I	Analog	Digital I/O.
AN12		I	ST	Analog input 12.
INT0		I	ST	External interrupt 0.
FLT0		I	ST	PWM Fault input (CCP1 module).
SDI		I	ST	SPI data in.
SDA		I/O	ST	I ² C™ data I/O.
RB1/AN10/INT1/SCK/ SCL	22	I/O	TTL	
RB1		I/O	TTL	Digital I/O.
AN10		I	Analog	Analog input 10.
INT1		I	ST	External interrupt 1.
SCK		I/O	ST	Synchronous serial clock input/output for SPI mode.
SCL		I/O	ST	Synchronous serial clock input/output for I ² C mode.
RB2/AN8/INT2/VMO	23	I/O	TTL	
RB2		I/O	TTL	Digital I/O.
AN8		I	Analog	Analog input 8.
INT2		I	ST	External interrupt 2.
VMO		O	—	External USB transceiver VMO output.
RB3/AN9/CCP2/VPO	24	I/O	TTL	
RB3		I/O	TTL	Digital I/O.
AN9		I	Analog	Analog input 9.
CCP2 ⁽¹⁾		I/O	ST	Capture 2 input/Compare 2 output/PWM 2 output.
VPO		O	—	External USB transceiver VPO output.
RB4/AN11/KBI0	25	I/O	TTL	
RB4		I/O	TTL	Digital I/O.
AN11		I	Analog	Analog input 11.
KBI0		I	TTL	Interrupt-on-change pin.
RB5/KBI1/PGM	26	I/O	TTL	
RB5		I/O	TTL	Digital I/O.
KBI1		I	TTL	Interrupt-on-change pin.
PGM		I/O	ST	Low-Voltage ICSP™ Programming enable pin.
RB6/KBI2/PGC	27	I/O	TTL	
RB6		I/O	TTL	Digital I/O.
KBI2		I	TTL	Interrupt-on-change pin.
PGC		I/O	ST	In-Circuit Debugger and ICSP programming clock pin.
RB7/KBI3/PGD	28	I/O	TTL	
RB7		I/O	TTL	Digital I/O.
KBI3		I	TTL	Interrupt-on-change pin.
PGD		I/O	ST	In-Circuit Debugger and ICSP programming data pin.

Pin Name	Pin Number	Pin Type	Buffer Type	Description
	PDIP, SOIC			
RC0/T1OSO/T13CKI RC0 T1OSO T13CKI	11	I/O O I	ST — ST	PORTC is a bidirectional I/O port. Digital I/O. Timer1 oscillator output. Timer1/Timer3 external clock input.
RC1/T1OSI/CCP2/UOE RC1 T1OSI CCP2 ⁽²⁾ UOE	12	I/O I I/O —	ST CMOS ST —	Digital I/O. Timer1 oscillator input. Capture 2 input/Compare 2 output/PWM 2 output. External USB transceiver OE output.
RC2/CCP1 RC2 CCP1	13	I/O I/O	ST ST	Digital I/O. Capture 1 input/Compare 1 output/PWM 1 output.
RC4/D-/VM RC4 D- VM	15	I I/O I	TTL — TTL	Digital input. USB differential minus line (input/output). External USB transceiver VM input.
RC5/D+/VP RC5 D+ VP	16	I I/O O	TTL — TTL	Digital input. USB differential plus line (input/output). External USB transceiver VP input.
RC6/TX/CK RC6 TX CK	17	I/O O I/O	ST — ST	Digital I/O. EUSART asynchronous transmit. EUSART synchronous clock (see RX/DT).
RC7/RX/DT/SDO RC7 RX DT SDO	18	I/O I I/O O	ST ST ST —	Digital I/O. EUSART asynchronous receive. EUSART synchronous data (see TX/CK). SPI data out.
RE3	—	—	—	See MCLR/VPP/RE3 pin.
VUSB	14	O	—	Internal USB 3.3V voltage regulator.
Vss	8, 19	P	—	Ground reference for logic and I/O pins.
Vdd	20	P	—	Positive supply for logic and I/O pins.

FIGURA 1-3: DESCRIPCIÓN DE LOS PINES DEL PIC18F4455/4550

Pin Name	Pin Number			Pin Type	Buffer Type	Description
	PDIP	QFN	TQFP			
MCLR/VPP/RE3 MCLR VPP RE3	1	18	18	I P I	ST Analog	Master Clear (input) or programming voltage (input). Master Clear (Reset) input. This pin is an active-low Reset to the device. Programming voltage input. Digital input.
OSC1/CLKI OSC1 CLKI	13	32	30	I I	Analog Analog	Oscillator crystal or external clock input. Oscillator crystal input or external clock source input. External clock source input. Always associated with pin function OSC1. (See OSC2/CLKO pin.)
OSC2/CLKO/RA6 OSC2 CLKO RA6	14	33	31	O O I/O	— — TTL	Oscillator crystal or clock output. Oscillator crystal output. Connects to crystal or resonator in Crystal Oscillator mode. In RC mode, OSC2 pin outputs CLKO which has 1/4 the frequency of OSC1 and denotes the instruction cycle rate. General purpose I/O pin.

Pin Name	Pin Number			Pin Type	Buffer Type	Description
	PDIP	QFN	TQFP			
RB0/AN12/INT0/ FLT0/SDI/SDA RB0 AN12 INT0 FLT0 SDI SDA	33	9	8	I/O I I I I I/O	TTL Analog ST ST ST ST	PORTB is a bidirectional I/O port. PORTB can be software programmed for internal weak pull-ups on all inputs. Digital I/O. Analog input 12. External interrupt 0. Enhanced PWM Fault input (ECCP1 module). SPI data in. I ² C™ data I/O.
RB1/AN10/INT1/SCK/ SCL RB1 AN10 INT1 SCK SCL	34	10	9	I/O I I I/O I/O	TTL Analog ST ST ST	Digital I/O. Analog input 10. External interrupt 1. Synchronous serial clock input/output for SPI mode. Synchronous serial clock input/output for I ² C mode.
RB2/AN8/INT2/VMO RB2 AN8 INT2 VMO	35	11	10	I/O I I O	TTL Analog ST —	Digital I/O. Analog input 8. External interrupt 2. External USB transceiver VMO output.
RB3/AN9/CCP2/VPO RB3 AN9 CCP2 ⁽¹⁾ VPO	36	12	11	I/O I I/O O	TTL Analog ST —	Digital I/O. Analog input 9. Capture 2 input/Compare 2 output/PWM 2 output. External USB transceiver VPO output.
RB4/AN11/KBI0/CSSPP RB4 AN11 KBI0 CSSPP	37	14	14	I/O I I O	TTL Analog TTL —	Digital I/O. Analog input 11. Interrupt-on-change pin. SPP chip select control output.
RB5/KBI1/PGM RB5 KBI1 PGM	38	15	15	I/O I I/O	TTL TTL ST	Digital I/O. Interrupt-on-change pin. Low-Voltage ICSP™ Programming enable pin.
RB6/KBI2/PGC RB6 KBI2 PGC	39	16	16	I/O I I/O	TTL TTL ST	Digital I/O. Interrupt-on-change pin. In-Circuit Debugger and ICSP programming clock pin.
RB7/KBI3/PGD RB7 KBI3 PGD	40	17	17	I/O I I/O	TTL TTL ST	Digital I/O. Interrupt-on-change pin. In-Circuit Debugger and ICSP programming data pin.

Pin Name	Pin Number			Pin Type	Buffer Type	Description
	PDIP	QFN	TQFP			
RC0/T1OSO/T13CKI RC0 T1OSO T13CKI	15	34	32	I/O O I	ST — ST	PORTC is a bidirectional I/O port. Digital I/O. Timer1 oscillator output. Timer1/Timer3 external clock input.
RC1/T1OSI/CCP2/ UOE RC1 T1OSI CCP2 ⁽²⁾ UOE	16	35	35	I/O I I/O O	ST CMOS ST —	Digital I/O. Timer1 oscillator input. Capture 2 input/Compare 2 output/PWM 2 output. External USB transceiver OE output.
RC2/CCP1/P1A RC2 CCP1 P1A	17	36	36	I/O I/O O	ST ST TTL	Digital I/O. Capture 1 input/Compare 1 output/PWM 1 output. Enhanced CCP1 PWM output, channel A.
RC4/D-/VM RC4 D- VM	23	42	42	I I/O I	TTL — TTL	Digital input. USB differential minus line (input/output). External USB transceiver VM input.
RC5/D+/VP RC5 D+ VP	24	43	43	I I/O I	TTL — TTL	Digital input. USB differential plus line (input/output). External USB transceiver VP input.
RC6/TX/CK RC6 TX CK	25	44	44	I/O O I/O	ST — ST	Digital I/O. EUSART asynchronous transmit. EUSART synchronous clock (see RX/DT).
RC7/RX/DT/SDO RC7 RX DT SDO	26	1	1	I/O I I/O O	ST ST ST —	Digital I/O. EUSART asynchronous receive. EUSART synchronous data (see TX/CK). SPI data out.

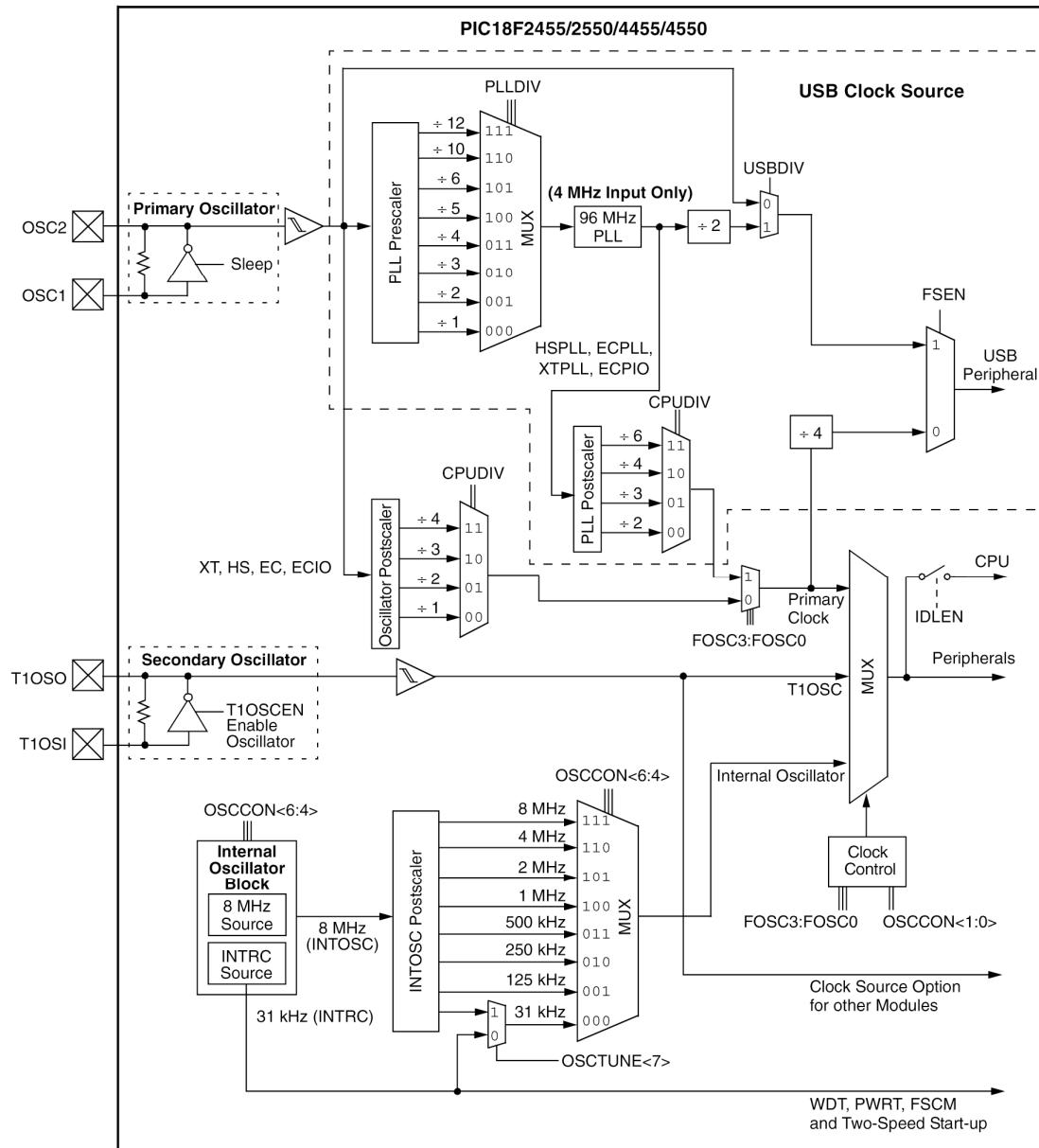
Pin Name	Pin Number			Pin Type	Buffer Type	Description
	PDIP	QFN	TQFP			
RD0/SPP0 RD0 SPP0	19	38	38	I/O I/O	ST TTL	PORTD is a bidirectional I/O port or a Streaming Parallel Port (SPP). These pins have TTL input buffers when the SPP module is enabled. Digital I/O. Streaming Parallel Port data.
RD1/SPP1 RD1 SPP1	20	39	39	I/O I/O	ST TTL	Digital I/O. Streaming Parallel Port data.
RD2/SPP2 RD2 SPP2	21	40	40	I/O I/O	ST TTL	Digital I/O. Streaming Parallel Port data.
RD3/SPP3 RD3 SPP3	22	41	41	I/O I/O	ST TTL	Digital I/O. Streaming Parallel Port data.
RD4/SPP4 RD4 SPP4	27	2	2	I/O I/O	ST TTL	Digital I/O. Streaming Parallel Port data.
RD5/SPP5/P1B RD5 SPP5 P1B	28	3	3	I/O I/O O	ST TTL —	Digital I/O. Streaming Parallel Port data. Enhanced CCP1 PWM output, channel B.
RD6/SPP6/P1C RD6 SPP6 P1C	29	4	4	I/O I/O O	ST TTL —	Digital I/O. Streaming Parallel Port data. Enhanced CCP1 PWM output, channel C.
RD7/SPP7/P1D RD7 SPP7 P1D	30	5	5	I/O I/O O	ST TTL —	Digital I/O. Streaming Parallel Port data. Enhanced CCP1 PWM output, channel D.

Pin Name	Pin Number			Pin Type	Buffer Type	Description
	PDIP	QFN	TQFP			
RE0/AN5/CK1SPP RE0 AN5 CK1SPP	8	25	25	I/O I O	ST Analog —	PORTE is a bidirectional I/O port. Digital I/O. Analog input 5. SPP clock 1 output.
RE1/AN6/CK2SPP RE1 AN6 CK2SPP	9	26	26	I/O I O	ST Analog —	Digital I/O. Analog input 6. SPP clock 2 output.
RE2/AN7/OESPP RE2 AN7 OESPP	10	27	27	I/O I O	ST Analog —	Digital I/O. Analog input 7. SPP output enable output.
RE3	—	—	—	—	—	See MCLR/VPP/RE3 pin.
Vss	12, 31	6, 30, 31	6, 29	P	—	Ground reference for logic and I/O pins.
VDD	11, 32	7, 8, 28, 29	7, 28	P	—	Positive supply for logic and I/O pins.
VUSB	18	37	37	O	—	Internal USB 3.3V voltage regulator output.
NC/ICCK/ICPGC ⁽³⁾ ICCK ICPGC	—	—	12	I/O I/O	ST ST	No Connect or dedicated ICD/ICSP™ port clock. In-Circuit Debugger clock. ICSP programming clock.
NC/ICDT/ICPGD ⁽³⁾ ICDT ICPGD	—	—	13	I/O I/O	ST ST	No Connect or dedicated ICD/ICSP port clock. In-Circuit Debugger data. ICSP programming data.
NC/ICRST/ICVPP ⁽³⁾ ICRST ICVPP	—	—	33	I P	— —	No Connect or dedicated ICD/ICSP port Reset. Master Clear (Reset) input. Programming voltage input.
NC/ICPORTS ⁽³⁾ ICPORTS	—	—	34	P	—	No Connect or 28-pin device emulation. Enable 28-pin device emulation when connected to Vss.
NC	—	13	—	—	—	No Connect.

2.0 CONFIGURACIÓN DEL OSCILADOR

El microcontrolador necesita un reloj estable para proporcionar una fuente de reloj separada.

FIGURA 2-1: DIAGRAMA DEL RELOJ



2.1 CONTROL DEL OSCILADOR

El oscilador se controla mediante dos registros de configuración y otros dos de control.

En CONFIG1L y CONFIG1H se elige el oscilador y las opciones de los escaladores del USB (prescaler y postscaler).

Los bits de configuración se seleccionan al programarlos y permanecerán hasta que se vuelvan a programar.

El registro OSCCON selecciona el modo activo de reloj.

El registro OSCTUNE se utiliza para recortar frecuencia de la fuente INTRC.

2.2 TIPOS DE OSCILADOR

Hay que programar los bits de configuración FOSC3:FOSC0 para seleccionar un modo de estos:

- **XT:** XTAL/ circuito resonante.
- **XTPPLL:** XTAL/ circuito resonante con PLL activo.
- **HS:** XTAL/ circuito resonante muy rápido.
- **HSPLL:** XTAL/ circuito resonante muy rápido con PLL activo.
- **EC:** Reloj externo con Fosc/4 como salida en RA6.
- **ECIO:** Reloj externo con RA6 como E/S.
- **ECPLL:** Reloj externo con Fosc/4 como salida en RA6 y PLL activo.
- **ECPIO:** Reloj externo con RA6 como E/S y PLL activo.
- **INTHS:** Oscilador interno usado como reloj del microcontrolador y HS como reloj del USB.
- **INTXT:** Oscilador interno usado como reloj del microcontrolador y XT como reloj del USB.
- **INTIO:** Oscilador interno usado como reloj del microcontrolador y EC como reloj del USB con RA6 como E/S.
- **INTCKO:** Oscilador interno usado como reloj del microcontrolador y EC como reloj del USB con Fosc/4 como salida en RA6.

2.2.1 MODOS DEL OSILADOR Y OPERACIONES CON EL USB

En la familia PIC18F2455/PIC18F2550/PIC18F4455/PIC18F4550 el oscilador primario forma parte del módulo USB y no se puede asociar a ninguna otra fuente de reloj.

El módulo USB tiene que ser controlado por la fuente primaria y los otros dispositivos o el microcontrolador se pueden controlar por fuentes secundarias u osciladores internos como en el resto de microcontroladores PIC.

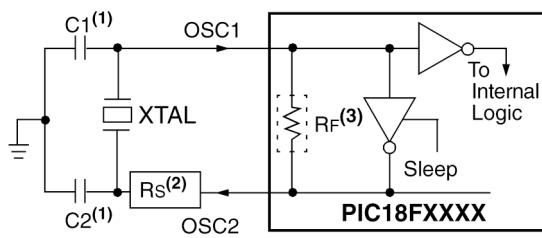
Al utilizar el USB necesitamos un reloj interno de 6MHz ó 48MHz, el resto del sistema puede funcionar con cualquier otro oscilador.

2.2.2 XTAL/RESONADORES CERÁMICOS

En HS, HSPLL, XT y XTPPLL el XTAL o el circuito resonador se conecta a OSC1 y OSC2.

Un postscaler interno permite elegir una frecuencia distinta a la del reloj de entrada. La división de la frecuencia se elige con el bit de configuración CPUDIV y se puede seleccionar a 1/2, 1/3 ó 1/4 de la frecuencia de entrada.

Se puede utilizar un reloj externo cuando el microcontrolador está en modo HS. En este caso el pin OSC2/CLK0 está abierto.

FIGURA 2-2: XTAL/RESONADOR CERÁMICO (CONFIGURACIÓN XT, HS O HSPLL)**Nota:**

1 Ver tablas 2-1 y 2-2 para conocer los valores iniciales de C1 y C2

2 Se puede necesitar una resistencia R_S con cristales de AT

3 R_F varía con el modo de oscilador seleccionado

TABLA 2-1: CONDENSADOR PARA OSCILADORES CERÁMICOS

Valores de los condensadores típicos:

Modo	Freq	OSC1	OSC2
XT	4.0MHz	33pF	33pF
HS	8.0MHz	27pF	27pF
	16.0MHz	22pF	22pF

El valor de los condensadores es sólo como guía en el diseño.

Estos condensadores se han probado con los osciladores de la lista de debajo en operaciones básicas de inicio. Estos valores no están optimizados.

Se puede necesitar otros condensadores de distinto valor para conseguir una oscilación aceptable en una determinada operación. El usuario tiene que probar la forma de la onda conseguida con una determinada V_{DD} y un rango de temperaturas en la cual trabaje la aplicación.

Osciladores utilizados:

- 4.0 MHz
- 8.0 MHz
- 16.0 MHz

TABLA 2-2: CONDENSADORES PARA LOS XTAL

Valores de los condensadores típicos:

Tipo de Osc	Xtal	Condensadores probados	
		Freq	C1
XT	4MHz	27pF	27pF
HS	4MHz	27pF	27pF
	8MHz	22pF	22pF
	20MHz	15pF	15pF

El valor de los condensadores es sólo como guía en el diseño.

Estos condensadores se han probado con los osciladores de la lista de debajo en operaciones básicas de inicio. Estos valores no están optimizados.

Se puede necesitar otros condensadores de distinto valor para conseguir una oscilación aceptable en una determinada operación. El usuario tiene que probar la forma de la onda conseguida con una determinada V_{DD} y un rango de temperaturas en la cual trabaje la aplicación.

Xtal utilizado:

- 4 MHz
- 8 MHz
- 20 MHz

2.2.3 SEÑAL DE RELOJ EXTERNA

Los modos EC, ECIO, ECPLL y ECPIO requieren una señal externa de reloj conectada al pin OCS1. No se necesita un oscilador de “puesta en marcha” después de un reset o al salir del modo sleep.

En los modos EC y ECPLL, se obtiene una señal en el pin OSC2 de la señal de reloj dividida por 4. Esta señal se puede usar para test o para sincronizar otros dispositivos.

En los modos ECIO y ECPIO el pin OSC2 funciona como un pin E/S, será el RA6.

El postscaler interno que se utiliza en los modos XT y HS se puede usar también en los modos EC y ECIO.

FIGURA 2-3: SEÑAL DE RELOJ EXTERNA (CONFIGURACIÓN HS)

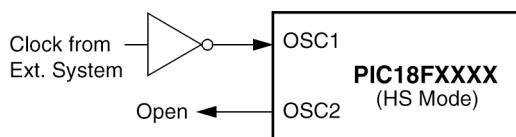


FIGURA 2-4: SEÑAL DE RELOJ EXTERNA (CONFIGURACIÓN EC Y ECPLL)

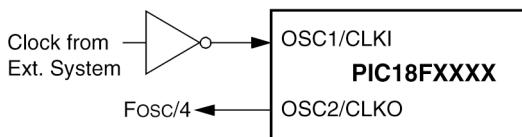
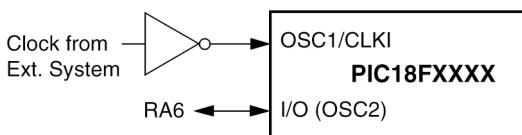


FIGURA 2-5: SEÑAL DE RELOJ EXTERNA (CONFIGURACIÓN ECIO Y ECPIO)



2.2.4 MULTIPLICADOR DE FRECUENCIA PLL

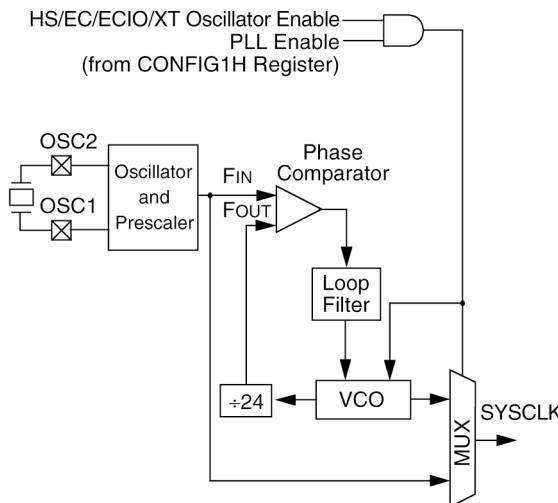
Esta familia de dispositivos incluye un circuito PLL (Paced Locked Loop). Se utiliza en aplicaciones USB con velocidad baja de oscilación y como fuente de reloj para el microcontrolador.

El PLL está activo en los modos HSPLL, XTPLL, ECPLL y ECPIO. Está diseñado para producir una señal de referencia de 96MHz de una señal de entrada de 4MHz. La salida se puede dividir y utilizar para el USB y el reloj del microcontrolador porque el PLL tiene una señal de entrada y salida fijas. Tiene ocho opciones de prescaler para elegir la frecuencia de entrada al PLL.

Hay un postscaler para elegir la velocidad del microcontrolador, para que el USB y el microcontrolador tengan una entrada de reloj y funcionen a velocidades diferentes. El postscaler tiene las opciones de 1/2, 1/4 y 1/6 de la salida del PLL.

Los modos HSPLL, ECPLL y ECPIO utilizan el modo del oscilador HS para frecuencias superiores a 48MHz. El prescaler divide la señal de entrada por 12 para producir 4MHz para el PLL. El modo XTPLL sólo utiliza la frecuencia de entrada de 4MHz que dirige el PLL directamente.

FIGURA 2-6: DIAGRAMA DEL BLOQUE PLL (CONFIGURACIÓN HS)



2.2.5 BLOQUE INTERNO DEL OSCILADOR

En la familia PIC18F2455/PIC18F2550/PIC18F4455/PIC18F4550 tienen un oscilador interno que genera dos señales diferentes; cualquiera se puede utilizar como reloj del microcontrolador. Si el USB no está utilizado, el oscilador interno puede eliminar el oscilador externo de los pines OSC1 y/o OSC2.

La salida principal (INTOSC) es una fuente de reloj de 8MHz que se puede utilizar para dirigir el reloj directamente. Esto también gobierna el postscaler de INTOSC, el cual puede proporcionar un rango de frecuencias de 31kHz a 4MHz. La salida INTOSC está activa cuando se selecciona una frecuencia de reloj de 125kHz a 8MHz.

La otra señal interna de reloj es el oscilador RC (INTRC) que da una salida nominal de 31kHz. INTRC está activo si se selecciona como fuente de reloj; se activa automáticamente cuando alguno de los siguientes está activo:

- Temporizador de encendido
- Monitor de reloj de seguridad
- Temporizador perro guardián
- Dos velocidades de inicio

La frecuencia se elige configurando el bit IRCF del registro OSCCON.

2.2.5.1 MODOS DEL OSCILADOR INTERNO

Cuando el oscilador interno funciona como reloj del microcontrolador, uno de los otros modos debe utilizarse como reloj del USB. La elección del USB se determina dependiendo del modo de funcionamiento interno.

Hay cuatro modos distintos:

- **INTHS**: Da la señal el oscilador en modo HS.
- **INTXT**: Da la señal el oscilador en modo XT.
- **INTCKO**: La señal la da un oscilador externo conectado al OSC1; en el OSC2 podemos obtener Fosc/4.
- **INTIO**: La señal la da un oscilador externo conectado al OSC1; en el OSC2 tenemos un pin E/S (RA6).

2.2.5.2 REGISTRO OSCTUNE

La salida interna de oscilación está calibrada de fábrica pero se puede ajustar a la aplicación del usuario. Se puede hacer escribiendo el registro OSCTUNE. La sensibilidad del ajuste es constante en todo el rango.

Cuando se modifica el OSCTUNE las frecuencias INTOSC e INTRC se cambiarán a la nueva frecuencia. El reloj INTRC alcanzará la nueva frecuencia en 8 ciclos de reloj mientras que el reloj INTOSC se estabilizará en 1ms. El programa se seguirá ejecutando durante este proceso. No hay ninguna indicación de que el suceso haya concluido.

El registro OSCTUNE contiene el bit INTSRC que permite seleccionar que oscilador interno da la señal de reloj cuando la frecuencia de 31kHz está seleccionada.

2.2.5.3 FRECUENCIA DEL OSCILADOR INTERNO Y CAMBIOS

El oscilador interno está calibrado para dar 8MHz. Sin embargo, esta frecuencia se puede modificar cuando V_{DD} o la temperatura cambian, puede afectar el control de una operación de muchas formas.

La baja frecuencia del oscilador INTRC funciona independientemente de la fuente INTOSC. Cualquier cambio en INTOSC por tensión o temperatura no tiene porqué reflejarse en INTRC y viceversa.

REGISTRO 2-1. OSCTUNE: Registro selector del oscilador

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
INTSRC	-	-	TUN4	TUN3	TUN2	TUN1	TUN0
BIT 7	INTSRC:	Bit selector de la baja frecuencia del oscilador interno.					
		1= 31,25kHz producido por los 8MHz de INTOSC (activa el divisor por 256)					
		0= 31kHz producido por el oscilador interno INTRC					
BIT 6-5		No implementados, se leen 0					
BIT 4-0	TUN4:TUN0:	Selectores de frecuencia: 01111: Máxima frecuencia 00001 00000: Frecuencia media. Funciona con la velocidad calibrada. 11111 10000: Mínima frecuencia.					

2.2.5.4 COMPENSACIÓN POR LOS CAMBIOS EN INTOSC

Es posible ajustar la frecuencia del INTOSC modificando el registro OSCTUNE. Esto no producirá ningún cambio en la frecuencia de INTRC.

El ajuste de INTOSC debe hacerse dependiendo de la aplicación. Por ejemplo, si necesitamos la EUSART, debemos ajustar la frecuencia para evitar fallos de recepción o de cuadrar los pulsos. Los errores por no cuadrar los pulsos indican que la frecuencia de reloj es muy alta, para ajustarla decrementamos el valor del OSCTUNE. Los errores en datos puedes ocurrir porque la velocidad del reloj es muy baja, para compensarlo aumentamos el OSCTUNE.

Es posible comparar la velocidad del reloj con un reloj de referencia. Se pueden usar dos temporizadores: un reloj externo, y el otro es un reloj de referencia como el oscilador Timer1. Ambos temporizadores se borran pero el reloj tomado como referencia genera interrupciones. Cuando ocurre una interrupción, se lee el reloj interno y los dos temporizadores borrados. Si el valor del temporizador interno es demasiado alto, el bloque del oscilador interno funciona muy deprisa. Para ajustarlo, hay que decrementar el registro OSCTUNE.

Se puede utilizar un módulo CCP libremente en el Timer1 o el Timer3, ajustado por el bloque oscilador interno y un evento externo con un periodo conocido (por ejemplo, una corriente alterna). El periodo del primer evento se captura en los registros CCPRxH:CCPRxL y se graba para poder utilizarlo posteriormente. Cuando ocurre un segundo evento, se resta al tiempo del segundo el del primero. Como se conoce el periodo externo, se puede calcular la diferencia entre los dos eventos.

Si la medida es mucho mayor que el tiempo calculado quiere decir que el bloque del oscilador interno está funcionando demasiado rápido; para compensarlo, hay que decrementar el registro OSCTUNE. Si la medida es mucho menor que la calculada, el bloque del oscilador interno funciona muy lento; para compensarlo, aumentar el registro OSCTUNE.

2.3 CONFIGURACIÓN DEL OSCILADOR PARA EL USB

Cuando el PIC18F4550 se utiliza en una conexión por USB, necesitamos un reloj de 6MHz ó 48MHz dependiendo de si usamos alta o baja velocidad. Debemos preverlo al seleccionar la frecuencia y al programar el microcontrolador.

TABLA 2-3: OPCIONES DE CONFIGURACIÓN PARA EL OSCILADOR CON USB

Frecuencia de entrada del Osc.	Divisor PILL	Modo del reloj	División del reloj	MCU	Frecuencia
48MHz	$\div N/A^{(1)}$	EC, ECIO	None(00) $\div 2(01)$ $\div 3(10)$ $\div 4(11)$	48MHz 24MHz 16MHz 12MHz	reloj del micro
48MHz	$\div 12(111)$	EC, ECIO	None(00) $\div 2(01)$ $\div 3(10)$ $\div 4(11)$ ECPLL, ECP PIO	48MHz 24MHz 16MHz 12MHz 48MHz 32MHz	(PLLDIV2:PLLDIVO)(FOSC3:FOSCO)(CPUDIV1:CPUDIVO)
40MHz	$\div 10(110)$	EC, ECIO	None(00) $\div 2(01)$ $\div 3(10)$ $\div 4(11)$ ECPLL, ECP PIO	40MHz 20MHz 13.33MHz 10MHz 48MHz	
24MHz	$\div 6(101)$	HS, EC, ECIO	None(00) $\div 2(01)$ $\div 3(10)$ $\div 4(11)$ HSPLL, ECPLL, ECP IO	12MHz 8MHz 6MHz 48MHz 32MHz	
20MHz	$\div 5(100)$	HS, EC, ECIO	None(00) $\div 2(01)$ $\div 3(10)$ $\div 4(11)$ HSPLL, ECPLL, ECP IO	20MHz 10MHz 6.67MHz 5MHz 48MHz	
16MHz	$\div 4(011)$	HS, EC, ECIO	None(00) $\div 2(01)$ $\div 3(10)$ $\div 4(11)$ HSPLL, ECPLL, ECP IO	16MHz 8MHz 5.33MHz 4MHz 48MHz	
12MHz	$\div 3(010)$	HS, EC, ECIO	None(00) $\div 2(01)$ $\div 3(10)$ $\div 4(11)$ HSPLL, ECPLL, ECP IO	12MHz 6MHz 4MHz 3MHz 48MHz	
8MHz	$\div 2(001)$	HS, EC, ECIO	None(00) $\div 2(01)$ $\div 3(10)$ $\div 4(11)$ HSPLL, ECPLL, ECP IO	8MHz 4MHz 2.67MHz 2MHz 48MHz	

4MHz	$\div 1(000)$	XT, HS, EC, ECIO	$\div 4(10)$	24MHz
			$\div 6(11)$	16MHz
			None(00)	4MHz
			$\div 2(01)$	2MHz
Frecuencia de del entrada del Osc.(PLLDIV2:PLLDIVO)(FOSC3:FOSCO)(CPUDIV1:CPUDIVO) reloj del micro	Divisor PILL	Modo del reloj	División del reloj	MCU Frecuencia
			$\div 3(10)$	1.33MHz
			$\div 4(11)$	1MHz
		HSPLL, ECPLL, XTPLL, ECPIO	$\div 2(00)$	48MHz
			$\div 3(01)$	32MHz
			$\div 4(10)$	24MHz
			$\div 6(11)$	16MHz

Nota: (1) Sólo es válido si el bit USBDIV está borrado.

Todas las frecuencias, excepto las de 24MHz, se utilizan en el USB 2.0. Las frecuencias de 24MHz se utilizan en el USB1.0.

2.3.1 OPERACIONES CON BAJA VELOCIDAD (USB 1.0)

El reloj del USB cuando funciona a baja velocidad se obtiene del oscilador primario y no del PLL. El oscilador se divide por 4 para obtener los 6MHz en el reloj. Por esto, el microcontrolador necesita una frecuencia de 24MHz cuando el módulo USB está activo y el controlador del reloj está en uno de los modos primarios del oscilador (XT, HS con o sin PLL).

Estas restricciones no se aplican si el reloj del microcontrolador es uno de los secundarios o del bloque interno.

2.3.2 FUNCIONAMIENTO DIFERENTE EL RELOJ DEL USB Y EL DEL MICROCONTROLADOR

El módulo USB puede funcionar de una forma asíncrona con respecto al reloj del microcontrolador y otros periféricos. Esto se puede realizar cuando el USB funciona con el oscilador primario y el microcontrolador con un reloj distinto con menor velocidad. Si necesitamos esto para ejecutar una aplicación entera con un solo reloj, las operaciones “a toda velocidad” proporcionan una gran variedad de frecuencias del microcontrolador.

2.4 FUENTES DE RELOJ Y CAMBIOS DE OSCILADOR

En la familia PIC18F2455/PIC18F2550/PIC18F4455/PIC18F4550 se incluye una opción para alternar la fuente de reloj con dos fuentes alternativas de menor frecuencia. Cuando activamos una fuente alternativa, tenemos disponibles varios modos de “control de energía”.

Hay tres fuentes de reloj en estos dispositivos:

- oscilador primario
- oscilador secundario
- bloque interno de oscilación

Los osciladores primarios incluyen los XTAL, los circuitos resonadores, los modos externos de reloj y el bloque interno de oscilación. El modo particular se elige con los bits de configuración FOSC3:FOSC0.

Los osciladores secundarios son las fuentes externas que no se conectan en los pines OSC1 u OSC2. Estas fuentes pueden seguir funcionando incluso cuando el microcontrolador se pone en modo “control de energía”.

Los micros PIC18F2455/PIC18F2550/PIC18F4455/PIC18F4550 ofrecen el Timer1 como oscilador secundario. Este oscilador es, a menudo, el temporizador base en las aplicaciones basadas en tiempo real en todos los modos de “control de energía”. Más comúnmente, se conecta un XTAL de 32,768kHz en los pines RC0/T1OSO/TI3CKI y RC1/T10SI/UOE* junto con un condensador e cada pin a masa.

El bloque interno de oscilación se puede usar como reloj en los modos “control de energía”. La fuente INTRC se utiliza en varias opciones especiales como el perro guardián o el monitor del reloj de seguridad.

2.4.1 REGISTROS DE CONTROL DEL OSCILADOR

El registro OSCCON maneja distintos aspectos de las operaciones del reloj en todos los modos de energía.

Los bits de selección de la fuente del reloj son SCS1:SCS0. Las fuentes de reloj disponibles son el reloj primario (definido por los bits FOSC3:FOSC0), el reloj secundario (oscilador Timer1) y el bloque del oscilador interno. La fuente de reloj cambia inmediatamente después de haber escrito el bit o los bits en un pequeño intervalo de reloj. Los bits SCS se borran en cualquier tipo de reset.

Los bits de selección de la frecuencia del oscilador interno son IRCF2:IRCF0, se elige la frecuencia de salida del bloque del oscilador interno. La elección se hace sobre la fuente INTRC, la fuente INTOSC o una de las frecuencias obtenidas de aplicar el postscaler. Si el bloque del oscilador interno aplica la señal de reloj, al cambiar los estados de estos bits obtendremos un cambio inmediato en la salida del oscilador. En un reset del dispositivo, la frecuencia del oscilador interno se configura como 1MHz.

Cuando elegimos una frecuencia de 31kHz (IRC2:IRC0=000), el usuario puede elegir que oscilador interno funciona como fuente. Esto se hace mediante el bit INTSRC del registro OSCTUNE. Activando este bit elegimos la fuente de reloj como INTOSC con una frecuencia de 31,25kHz activando el divisor por 256 en el postscaler del INTOSC. Desactivando este bit ponemos como fuente INTRC con una frecuencia nominal de 31kHz.

Esta opción permite a los usuarios seleccionar la fuente de reloj más precisa con INTOSC que mantiene el ahorro de energía con una baja velocidad de reloj.

INTRC siempre mantiene la señal de reloj en eventos como el perro guardián o el monitor de seguridad del reloj.

Los bits OSTS, IOFS y T1RUN indican cual es el reloj que proporciona la señal principal. El bit OSTS indica que el tiempo de arranque ha terminado y que el reloj primario proporciona el reloj en modo primario. El bit IOFS indica cuando se ha estabilizado el bloque del oscilador interno y que el reloj está en modo RC. El bit T1RUN indica cuando el Timer1 proporciona la señal secundaria de reloj. En los modos de “control de energía”, sólo se puede seleccionar uno de estos tres bits a la vez. Si no se han activado ninguno de estos tres bits, el reloj INTRC da la señal o el bloque interno del oscilador ha empezado y no está todavía listo.

El bit IDLEN determina si el dispositivo está en modo sleep o uno de los modos de ahorro, cuando se ha ejecutado la instrucción SLEEP.

2.4.2 TRANSICIONES DEL OSCILADOR

Esta familia de dispositivos tiene un circuito para prevenir “problemas” del reloj cuando cambian entre las distintas fuentes de reloj. Ocurre una pequeña pausa cuando cambia la señal de reloj. La longitud de esta pausa es la suma de dos ciclos del reloj antiguo mas de tres a cuatro ciclos del reloj nuevo. Esta formula indica cuando el reloj nuevo se hace estable.

REGISTRO 2-2. OSCCON: Registro de control del oscilador

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
IDLEN	IRCF2	IRCF1	IRCF0	OSTS	IOFS	SCS1	SCS0
BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
BIT 7	IDLEN:						
		Bit puntero del modo reposo					
		1= El dispositivo entra en modo reposo con la instrucción SLEEP					
		0= El dispositivo entra en modo sleep con la instrucción SLEEP					
BIT 6-4	IRCF2:IRCF0:	Selectores de la frecuencia interna del oscilador:					
		111=8MHz (INTOSC directamente)					
		110=4MHz					
		101=2MHz					
		100=1MHz (Por defecto en un reset)					
		011=500kHz					
		010=250kHz					
		001=125kHz					
		000=31kHz (INTOSC/256 o INTRC directamente. Se selecciona con el bit INTSRC del registro OSCTUNE)					
BIT 3	OSTS:	Bit de estado del desbordamiento de oscilador de inicio (depende del estado del bit de configuración IESO)					
		1= El tiempo de inicio ha terminado, el oscilador primario está activo.					
		0= El tiempo de inicio no ha acabado, el oscilador primario no está activado.					
BIT 2	IOFS:	Bit de frecuencia estable					
		1= La frecuencia de INTOSC es estable					
		0= La frecuencia de INTOSC no es estable					
BIT 1-0	SCS1:SCS0:	Bits de selección del reloj del sistema					
		1x= Oscilador interno					
		01= Oscilador Timer1					
		00= Oscilador primario					

2.5 EFECTO DE LOS MODOS DE CONTROL DE ENERGÍA EN VARAS FUENTES DE RELOJ

Cuando seleccionamos el modo PRI_IDLE el oscilador primario designado continúa sin interrupciones. En los demás modos de ahorro de energía, el oscilador que utilice el pin OSC1 se desactiva. A menos que el USB esté funcionando se desactivará el pin OSC1 (y el OSC2 si se utiliza) y parará la oscilación.

En el modo de reloj secundario (SEC_RUN y SEC_IDLE), el oscilador Timer1 funciona y proporciona la señal de reloj. El oscilador Timer1 o Timer3 pueden funcionar en todos los modos de ahorro de energía si se requiere como reloj.

En los modos de oscilador interno (RC_RUN y RC_IDLE), el oscilador interno proporciona la fuente de la señal de reloj. La salida de 31kHz de INTRC se puede utilizar directamente para producir la señal de reloj y varios de los modos de ahorro de energía especiales. La salida de INTOSC de 8MHz se puede utilizar para producir la señal de reloj del dispositivo directamente o por medio de algún postscaler. La salida del INTOSC se desactiva directamente si la señal de reloj la proporciona directamente la salida de INTRC.

El reloj del USB funciona sin verse afectado por el modo Run o el modo reposo. Si el dispositivo funciona con un XTAL o un circuito resonador, la oscilación

continuará en el reloj del USB. Éste módulo y los demás puede cambiarse a una nueva fuente de reloj.

Si se selecciona el modo sleep, se interrumpen todas las señales de reloj. Con todas las corrientes de polarización de los transistores obtenemos el menor consumo en el dispositivo.

No se debe activar el modo sleep cuando el módulo USB está activado y existen comunicaciones. La única excepción es cuando al dispositivo se le ha suspendido por medio de un comando del USB. Una vez que el módulo haya suspendido las operaciones y puesto en el estado de baja potencia, el microcontrolador se puede poner en modo sleep.

Activando cualquier proceso que funcione en el modo sleep se incrementará la corriente que necesita el circuito. El reloj INTRC se necesita para proporcionar las operaciones al WDT. El oscilador Timer1 se activará en aplicaciones en tiempo real. Hay otras operaciones que se pueden activar sin necesitar una señal de reloj.

2.6 RETRASO EN EL INICIO

Los retrasos en el inicio se controlan por medio de dos temporizadores por lo que, en la mayoría de las aplicaciones, no se necesitan reset externos. Los retrasos mantienen el dispositivo apagado hasta que la tensión es estable en circunstancias normales y el reloj primario es estable y está funcionando.

La primera vez es el Temporizador de inicio (PWRT), el cual proporciona un retraso fijo en el arranque. Se activa borrando en bit de configuración PWRTE*

El segundo temporizador es el oscilador-temporizador de arranque (OST), mantiene el chip apagado hasta que el oscilador es estable (en los modos XT y HS). El OST cuenta 1024 ciclos y después activa el micro.

Cuando elegimos el modo de oscilador HSPLL, el dispositivo permanece en reset 2ms más después del retraso del OST, para que el PLL pueda detectar la frecuencia de entrada.

Hay un intervalo de retraso T_{CSD} después del POR, cuando el controlador empieza a ejecutar una instrucción. Este retraso se ejecuta cuando están otros retrasos activos. Este es el único retraso que ocurre cuando se activan los modos de oscilador primario EC y el modo interno.

TABLA 2-4: ESTADO DE LOS PINES OSC1 Y OSC2 EN EL MODO SLEEP

Modo del Oscilador	OSC1 Pin	OSC2 Pin
INTCKO	Abierto, se conecta el reloj	Salida (reloj/4)
INTIO	Abierto, se conecta el reloj	Configurado como PORTA, bit 6
ECIO, ECHO	Abierto, se conecta el reloj	Configurado como PORTA, bit 6
EC	Abierto, se conecta el reloj	Salida (reloj/4)
XT and HS	Inactivo sin tensión	Inactivo sin tensión

3.0 MODOS DE CONTROL DE ENERGÍA

Los dispositivos PIC18F2455/2550/4455/4550 ofrecen un total de siete modos de funcionamiento para controlar la energía. Estos modos proporcionan una variedad de opciones para la conservación selectiva en las aplicaciones donde los recursos pueden ser limitados (es decir, los dispositivos con pilas).

Hay tres categorías de modos de control de energía:

- Modo ejecución.
- Modo reposo
- Modo sleep.

Estas categorías definen qué porciones del dispositivo se utilizan y a veces, qué velocidad. Los modos ejecución y reposo pueden utilizar cualquiera de los tres relojes disponibles (primario, secundario o bloque interno); el modo sleep no utiliza una fuente del reloj.

Los modos de control de energía incluyen varias características de ahorro de energía que se ofrecieron en los microcontroladores antiguos. Una es la característica de la conmutación del reloj, ofrecida en otros dispositivos PIC18, permitiendo que el regulador utilice el oscilador Timer1 en lugar del oscilador primario. También se incluye el modo sleep, ofrecido en todos los dispositivos PICmicro, donde se paran todos los relojes del dispositivo.

TABLA 3-1: MODOS DE CONTROL DE ENERGÍA

MODO IDLEN ⁽¹⁾	Bits OSCCON	CPU	Modulo que controla Periféricos	Fuente de reloj y oscilador disponible
Sleep	SCS1:SCS0 0	N/A	Off Off	Nada – los relojes están desactivados
PRI_RUN	N/A	00	Clocked Clocked	Primario-todos los modos del oscilador. Este es el modo de ejecución normal.
SEC-RUN	N/A	01	Clocked Clocked	Secundario-Como oscilador los Timer
RC RUN	N/A	1x	Clocked	Bloque interno del oscilador ⁽²⁾
PRI IDLE	1	00	Off Clocked	Primario- todos los modos de oscilador
SEC_IDLE	1	01	Off Clocked	Secundario-Como oscilador los Timer
RC IDLE	1	1x	Off Clocked	Bloque interno del oscilador ⁽²⁾

Nota: 1: IDLEN devuelve este valor cuando se ha ejecutado la instrucción SLEEP.

2: Incluye INTOSC y los postscaler INTOSC así como la fuente INTRC

3.1 SELECCIONAR LOS MODOS DE CONTROL DE ENERGÍA

Seleccionar un modo de control de energía requiere dos decisiones: si la CPU está controlada o no y la selección de una fuente del reloj. El bit IDLEN (OSCCON<7>) controla la CPU, mientras que los bits SCS1:SCS0 (OSCCON<1: 0>) seleccionan el reloj fuente.

3.1.1 FUENTES DEL RELOJ

Los bits SCS1:SCS0 permiten la selección de una de tres fuentes del reloj para los modos de control de energía. Son:

- El reloj primario, según lo definido en los bits de configuración FOSC3:FOSC0
- El reloj secundario (el oscilador Timer1)
- El bloque interno del oscilador (para los modos de RC)

3.1.2 COMIENZO DE LOS MODOS DE CONTROL DE ENERGÍA

El cambio de un modo de control de energía a otro comienza cargando el registro OSCCON. Los bits SCS1:SCS0 seleccionan la fuente del reloj y se determinan cuáles funcionan en los modos ejecución y reposo. Cambiar estos bits provoca un cambio inmediato al reloj seleccionado. El cambio puede estar sometido a retrasos en las transiciones del reloj.

La entrada a los modos de control de energía reposo o sleep se provoca por la ejecución de una instrucción SLEEP. El modo que se activa depende del estado del bit IDLEN.

Dependiendo del modo actual y del modo al cual vamos a cambiar, un cambio de un modo de control de energía no tiene por qué requerir seleccionar todos los bits. Muchas transiciones se pueden hacer cambiando los bits de selección del oscilador, o cambiando el bit IDLEN, antes de ejecutar una instrucción SLEEP. Si el bit IDLEN está configurado correctamente, puede ser necesario realizar una instrucción SLEEP para cambiar al modo deseado.

3.1.3 TRANSICIONES DE RELOJ Y PUNTERO DE ESTADO

La longitud de la transición entre las fuentes del reloj es la suma de dos ciclos de la fuente de reloj vieja y de tres a cuatro ciclos de la nueva fuente. Esta fórmula asume que la nueva fuente del reloj es estable. Tres bits indican la fuente actual del reloj y su estado. Son:

- OSTS (OSCCON<3>)
- IOFS (OSCCON<2>)
- T1RUN (T1CON<6>)

En general, sólo uno de estos bits se fijará en un modo de energía. Cuando el bit OSTS está activo, el reloj primario está proporcionando el reloj del dispositivo. Cuando se activa el bit IOFS, la salida de INTOSC está proporcionando una fuente de reloj de 8MHz estable a un divisor que conduce el reloj del dispositivo. Cuando se activa el bit de T1RUN, el oscilador Timer1 está proporcionando el reloj. Si no se activa ninguno de estos bits, entonces el reloj INTRC controla el dispositivo, o la fuente INTOSC no es estable todavía.

Si el bloque del oscilador interno se configura como el reloj primario por los bits de configuración FOSC3:FOSC0, entonces los bits OSTs y IOFS pueden activarse en los modos PRI_RUN y PRI_IDLE. Esto indica que es el reloj primario (salida INTOSC) que genera una señal estable de salida de 8MHz. Cambiar el modo de control de energía RC a otro de la misma frecuencia puede desactivar el bit OSTs.

Nota 1: Cuidado al modificar el bit IRCF. Si V_{DD} es menor de 3V, es posible seleccionar una velocidad de reloj más alta que la soportada por V_{DD}. Se puede provocar un error si se violan las especificaciones de V_{DD}/Fosc.

2: Ejecutar una instrucción SLEEP no pone necesariamente el dispositivo en el modo sleep. Actúa como disparador para colocar el controlador en el modo sleep, o en el modo reposo, dependiendo del ajuste del bit IDLEN.

3.1.4 MÚLTIPLES COMANDOS SLEEP

El modo control de energía que se invoca con la instrucción SLEEP se determina en el ajuste del bit IDLEN cuando se ejecuta la instrucción. Si se ejecuta otra instrucción del SLEEP, el dispositivo entrará en el modo de control de energía especificado por el bit IDLEN en ese momento. Si IDLEN ha cambiado, el dispositivo entrará en el nuevo modo de control de energía especificado por el nuevo ajuste.

3.2 MODOS DE EJECUCIÓN

En los modos de ejecución, están activos los relojes al núcleo y a los periféricos. La diferencia entre estos modos es la fuente de reloj.

3.2.1 MODO DE PRI_RUN

El modo de PRI_RUN es la ejecución de la potencia normal del microcontrolador. Éste también es el modo por defecto sobre un dispositivo reseteando a menos que el arranque a dos velocidades esté activo. En este modo, se activa el bit OSTS. El bit IOFS puede activarse si el bloque del oscilador interno es la fuente primaria de reloj.

3.2.2 MODO DE SEC_RUN

El modo de SEC_RUN es el modo compatible a “cambio de reloj” ofrecido en otros PIC18. En este modo, la CPU y los periféricos se controlan con el oscilador Timer1. Esto da a los usuarios la opción de un consumo de energía más bajo mientras que todavía se use una alta fuente estable de reloj.

Se pone el modo SEC_RUN fijando los bits SCS1:SCS0 con ‘01’. La fuente de reloj del dispositivo se cambia al oscilador Timer1, el oscilador primario se desactiva, el bit T1RUN (T1CON<6>) se activa y el bit OSTS borra.

Nota: El oscilador Timer1 debe estar funcionamiento antes de poner el modo SEC_RUN. Si el bit T1OSCEN no se activa cuando se ponen los bits SCS1:SCS0 a ‘01’, no se activará el modo SEC_RUN. Si El oscilador Timer1 está activado pero no funcionando, se provocará un retraso en los relojes del dispositivo hasta que el oscilador comience. En tal situación, la operación inicial del oscilador será inestable e imprevisible.

En transiciones entre los modos SEC_RUN a PRI_RUN, los periféricos y la CPU continúan siendo controlados por el oscilador Timer1 mientras que se enciende el reloj primario. Cuando el reloj primario está listo, se cambian los relojes. Cuando cambio de reloj se completa, el bit T1RUN se borra, y se activa el bit OSTS y el reloj primario está proporcionando la señal. A los bits IDLEN y SCS no les afecta el arranque; el oscilador Timer1 continúa funcionando.

FIGURA 3-1: TIEMPOS DE TRANSICIÓN PARA ENTRAR EN EL MODO SEC_RUN

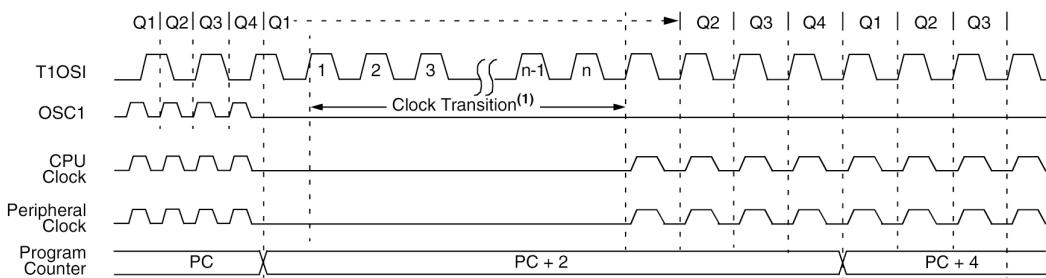
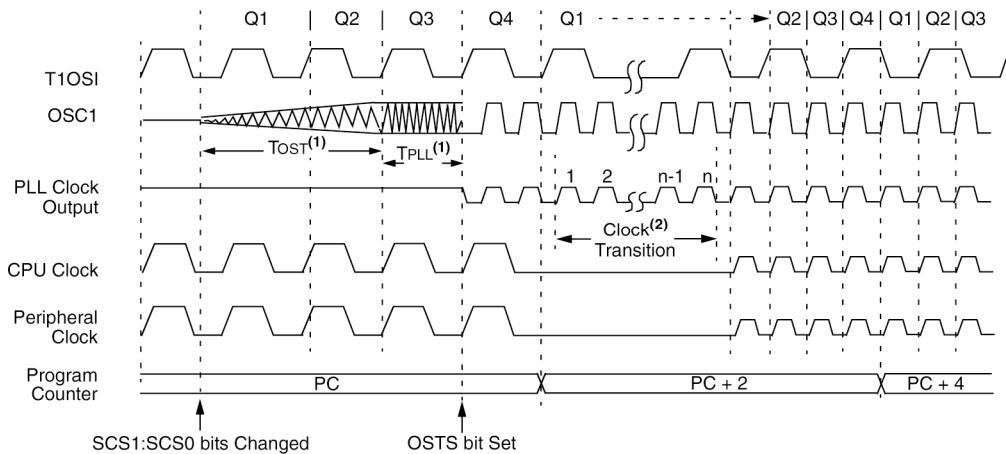


FIGURA 3-2: TIEMPOS DE TRANSICIÓN AL CAMBIAR DEL MODO SEC_RUN AL PRI_RUN (HSPLL)



3.2.3 MODO RC_RUN

En el modo RC_RUN, la CPU y los periféricos se están controlando por el bloque del oscilador interno usando el multiplexor INTOSC; se desactiva el reloj primario. Este modo proporciona la mejor conservación de energía de todos los modos cuando se ejecuta código. Trabaja bien cuando no se requiere medir el tiempo de una forma precisa ni alta velocidad. Si la fuente primaria del reloj es el bloque del oscilador interno (INTRC o INTOSC), no existen diferencias en la ejecución entre el PRI_RUN y RC_RUN. Sin embargo, un cambio en el reloj provocará un retraso durante la entrada y la salida del modo de RC_RUN. Por lo tanto, si la fuente primaria del reloj es el bloque interno del oscilador, no se recomienda el uso de RC_RUN.

Se activa este modo fijando SCS1 a '1'. Se recomienda que SCS0 también esté borrado; esto es para mantener la compatibilidad de software con dispositivos futuros. Cuando la fuente del reloj se cambia al multiplexor de INTOSC, el oscilador primario se desactiva y el bit OSTS se borra. Los bits IRCF se pueden modificar en cualquier momento y cambiar inmediatamente la velocidad de reloj.

Nota: Cuidado al modificar un solo bit de IRCF. Se puede provocar un error si se violan las especificaciones de V_{DD}/FOSC.

Si los bits IRCF e INTSRC están borrados, la salida de INTOSC está desactivada y el bit IOFS seguirá a 0; no habrá indicación de la fuente actual del reloj. La fuente INTRC será el reloj del dispositivo.

Si los bits IRCF se activan (permitiendo la salida INTOSC), o si se fija INTSRC, el bit IOFS se activará después de que la salida INTOSC sea estable. Las señales del dispositivo continúan mientras que la fuente de INTOSC se estabiliza después de un intervalo T_{IOBST}.

Si los bits IRCF estaban previamente en un valor diferente a cero o si INTSRC fue activado antes que SCS1 y la fuente INTOSC era estable, el bit IOFS seguirá activo.

En transiciones del modo RC_RUN al modo PRI_RUN, el dispositivo continúa siendo controlado por el multiplexor INTOSC mientras se activa el reloj primario. Cuando el reloj primario está listo, se produce un cambio a este. Cuando se completa el cambio de reloj, el bit IOFS se borra, se activa el bit OSTST y el reloj primario está proporcionando el reloj del dispositivo. A los bits IDLEN y de SCS no les afecta el cambio. La fuente de INTRC continuará funcionando si se permite el WDT o el monitor del reloj de seguridad.

FIGURA 3-3: TIEMPO DE TRANSICIÓN AL MODO RC_RUN

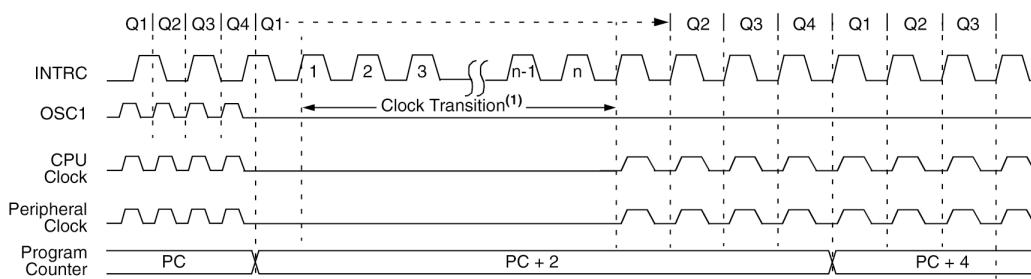
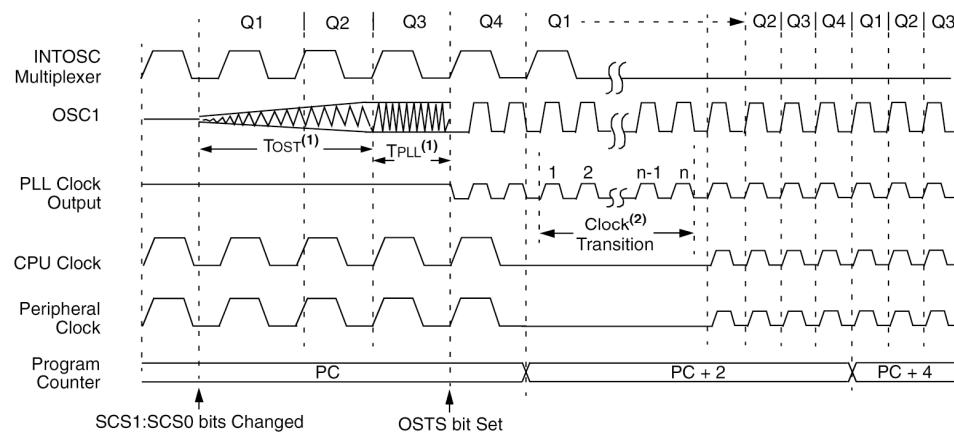


FIGURA 3-4: TIEMPO DE TRANSICIÓN EN EL CAMBIO DEL MODO RC_RUN AL PRI_RUN



3.3 MODO SLEEP

El modo de control de energía SLEEP en los dispositivos PIC18F2455/2550/4455/4550 es idéntico al modo SLEEP del resto de dispositivos de PICmicro. Se activa borrando el bit IDLEN (estado por defecto en el dispositivo) y

ejecutándose la instrucción SLEEP. Esto desactiva el oscilador. Se borran todos los bits de estado de la fuente del reloj.

Cambiar de un modo cualquiera al modo SLEEP no requiere un cambio de señal de reloj. Esto es porque no se necesitan relojes cuando el microcontrolador entra en el modo sleep. Si se activa el WDT, la fuente de INTRC empezará a funcionar. Si se permite el oscilador Timer1, también continuará funcionando.

Cuando ocurre un acontecimiento (una interrupción, un reset o el desbordamiento del WDT) en modo sleep, el dispositivo no funcionará hasta que la fuente del reloj seleccionada por los bits SCS1:SCS0 no esté lista, o se controlará por el bloque del oscilador interno si están activos el monitor del reloj de seguridad o el arranque con dos velocidades. En cualquier caso, se activa el bit OSTS cuando el reloj primario es el reloj del dispositivo. A los bits IDLEN y SCS no les afecta el arranque.

FIGURA 3-5: TIEMPO DE TRANSICIÓN PARA ENTRAR EN EL MODO SLEEP

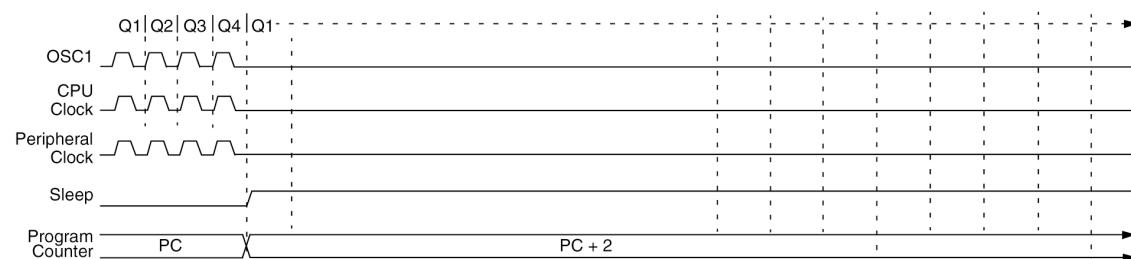
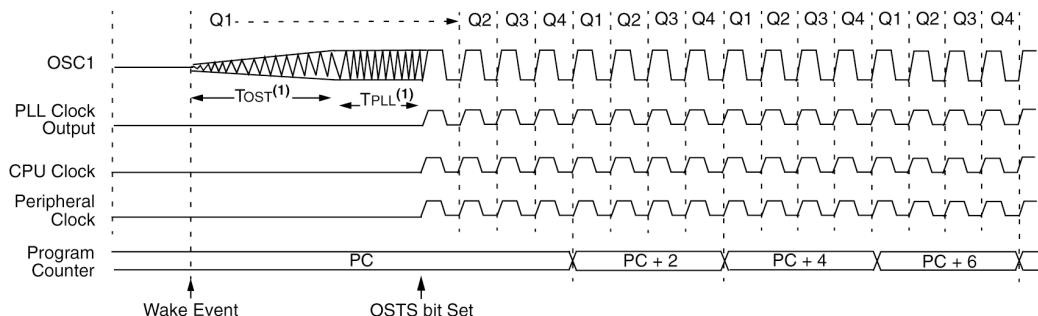


FIGURA 3-6: TIEMPO DE TRANSICIÓN DEL REINICIO DE UN SLEEP (HSPLL)



3.4 MODOS REPOSO

Los modos reposo permiten que la CPU del regulador esté apagada selectivamente mientras que los periféricos continúen funcionando. Seleccionar un modo particular permite a usuarios manejar más el consumo de energía.

Si el bit IDLEN se fija a ‘1’ cuando se ejecuta una instrucción SLEEP, los periféricos tendrán como fuente de reloj la seleccionada con los bits SCS1:SCS0; sin embargo, la CPU no tendrá reloj. A los bits de estado de la señal de reloj no les afecta. Activando el bit IDLEN y ejecutando una la instrucción SLEEP proporciona un método rápido de cambiar de un modo de funcionamiento dado a su modo de reposo correspondiente.

Si se selecciona el WDT, la fuente de INTRC continuará funcionando. Si se permite el oscilador Timer1, también continuará funcionando.

Puesto que la CPU no está ejecutando instrucciones, las únicas salidas de los modos de reposo son una interrupción, un desbordamiento del WDT descanso o un reset. Cuando ocurre un acontecimiento de este tipo, la CPU sufre un retraso del tipo T_{CSD} hasta que llega a ejecutar el código. Cuando la CPU comienza a ejecutar código, se acciona con la misma fuente del reloj que posea el Modo reposo. Por ejemplo, al despertar del modo RC_IDLE, el bloque del oscilador interno controlará la CPU y los periféricos (es decir, el modo RC_RUN). A los bits IDLEN y SCS no les afecta el reinicio.

Mientras que en cualquier modo reposo o modo sleep, un desbordamiento del WDT dará lugar a un reinicio al modo de funcionamiento especificado por los bits SCS1:SCS0.

3.4.1 MODO PRI_IDLE

Este modo es el único entre los tres modos de reposo de baja potencia que no inhabilita el reloj primario del dispositivo. Para aplicaciones con una temporización sensible, esto permite una reanudación más rápida del dispositivo, con una fuente primaria más exacta del reloj, puesto que la fuente del reloj no tiene que “calentar” o esperar la transición de otro oscilador.

El modo de PRI_IDLE entra en modo PRI_RUN activando el bit IDLEN y ejecutando una instrucción SLEEP. Si el dispositivo está en otro modo de funcionamiento, primero activa el bit IDLEN, después borrar los bits SCS y por último ejecutar una instrucción SLEEP. Aunque la CPU está desactivada, los periféricos continúan gobernados por la fuente primaria de reloj especificada por los bits de configuración FOSC3:FOSC0. El bit OSTS se activará.

Cuando ocurre un acontecimiento que provoque un reset, la CPU se controlará con la fuente primaria de reloj. Existe un retraso de tipo T_{CSD} entre el cambio de modo y el comienzo de la ejecución de código. Esto se provoca para permitir que la CPU esté lista para ejecutar instrucciones. Después del reinicio, el resto de bits OSTS se activan. A los bits IDLEN y SCS no les afecta el reset.

3.4.2 MODO SEC_IDLE

En modo de SEC_IDLE, la CPU se desactiva pero los periféricos continúan funcionando gracias al Timer1. Se cambia del modo SEC_RUN al SEC_IDLE activando el bit IDLEN y ejecutando una instrucción SLEEP. Si el dispositivo está en otro modo de funcionamiento, se activa primero el bit IDLEN, después se fijan los bits SCS1:SCS0 a ‘01’ y se ejecuta el comando SLEEP. Cuando la fuente del reloj se cambia al oscilador Timer1, se desactiva el oscilador primario, el bit OSTS se borra y se activa el bit T1RUN.

Cuando ocurre un acontecimiento de reinicio, los periféricos continúan funcionando con el oscilador Timer1. Después de un intervalo T_{CSD} que sigue a la

interrupción, la CPU comienza a ejecutar código gracias al oscilador Timer1. A los bits IDLEN y SCS no les afecta el reset; el oscilador Timer1 continúa funcionando.

Nota: El oscilador Timer1 debe estar en funcionamiento antes de entrar en el modo SEC_IDLE. Si el bit T1OSCEN no se activa cuando se ejecuta la instrucción SLEEP, se ignorará esta instrucción y no cambiara al modo SEC_IDLE. Si el oscilador Timer1 está activo pero no funciona todavía, se provocarán retrasos en los relojes de los periféricos hasta que el oscilador comience a contar. En tales situaciones, tenemos un oscilador imprevisible.

FIGURA 3-7: TIEMPO DE TRANSICIÓN PARA ENTRAR EN EL MODO IDLE

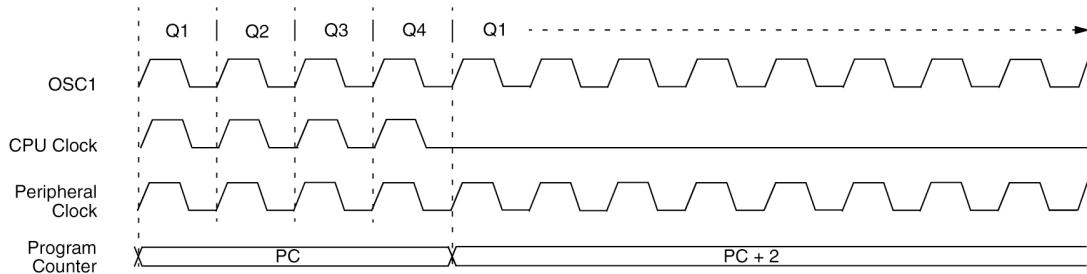
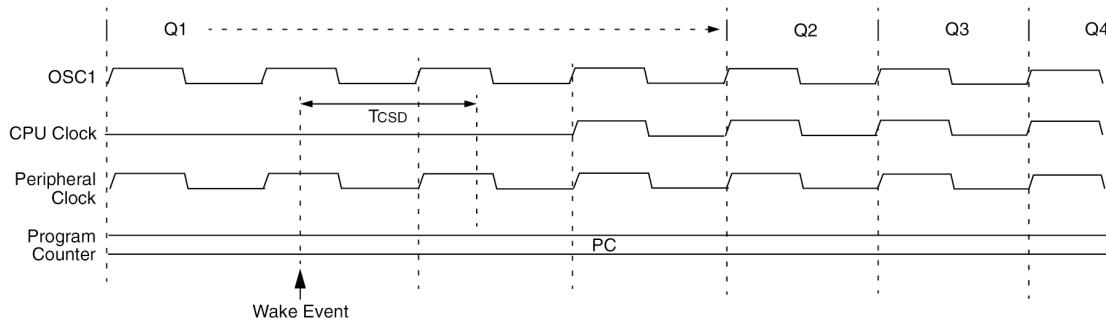


FIGURA 3-8: TIEMPO DE TRANSICIÓN EL REINICIO DEL MODO IDLE AL RUN



3.4.3 MODO RC_IDLE

En modo de RC_IDLE, la CPU se desactiva pero los periféricos continúan activos con el bloque del oscilador interno usando el multiplexor INTOSC. Este modo permite la conservación controlada de energía durante los períodos de reposo.

El paso del modo RC_RUN, a éste modo se realiza activando el bit IDLEN y ejecutando una instrucción SLEEP. Si el dispositivo está en otro modo de funcionamiento, primero se activa el bit IDLEN, después el bit SCS1 y por último, una instrucción SLEEP. Aunque no importa su valor, se recomienda que se borre el bit SCS0; esto es por mantener la compatibilidad de software con dispositivos futuros. El multiplexor INTOSC se puede utilizar para seleccionar una frecuencia de reloj más alta modificando los bits IRCF antes de ejecutar la instrucción SLEEP. Cuando la fuente del

reloj se cambia al multiplexor INTOSC, se desactiva el oscilador primario y el bit OSTS se borra.

Si los bits IRCF se ponen a cualquier valor diferente a cero, o se activa el bit INTSRC, se permite la salida INTOSC. El bit IOFS se activa cuando la salida INTOSC esté estable, después de un intervalo T_{IOBST} . Los relojes de los periféricos continúan hasta que la fuente de INTOSC se estabiliza. Si los bits IRCF estaban previamente en un valor diferente a cero, o INTSRC fue fijado antes de ejecutar la instrucción SLEEP y la fuente INTOSC es estable, el bit IOFS seguirá activo. Si los bits IRCF e INTSRC están todos borrados, la salida INTOSC no se permitirá, el bit IOFS seguirá borrado y no habrá indicación de la fuente actual de reloj.

Cuando ocurre un acontecimiento de reinicio, los periféricos continúan funcionando por el multiplexor INTOSC. Después del retraso T_{CSD} sigue la interrupción, la CPU comienza ejecutando el código que se controla con el multiplexor INTOSC. A los bits IDLEN y SCS no les afecta el reset. La fuente INTRC continuará funcionando si el WDT o el monitor de reloj de seguridad están activos.

3.5 SALIR DE LOS MODOS SLEEP Y REPOSO

La salida del modo sleep o de los modos de reposo se provoca accionando una interrupción, un reset o un desbordamiento del WDT. Esta sección discute los disparadores que causan las salidas de modos de ahorro de energía. El subsistema que registra las acciones se discuten en cada uno de los modos de ahorro de energía.

3.5.1 SALIDA POR UNA INTERRUPCIÓN

Cualquier interrupción puede provocar el cambio del modo de reposo o del modo sleep al modo ejecución. Para permitir esta funcionalidad, se debe activar una fuente de interrupción activando su bit en uno de los registros INTCON o PIE. Se inicia la secuencia de la salida cuando se activa el flag correspondiente a la interrupción.

En todas las salidas de modos reposo o sleep por una interrupción, el programa salta al vector interrupción si el bit GIE/GIEH (INTCON<7>) está activo. Si no, la ejecución de código continúa o se reanuda sin saltos.

Se necesita un retraso T_{CSD} después de la interrupción para que el sistema salga del modo reposo o sleep. Este retraso se provoca para que la CPU se prepare para la ejecución. La ejecución de la instrucción se reanuda en el primer ciclo de reloj que sigue a este retraso.

3.5.2 SALIDA POR DESBORDAMIENTO DEL WDT

La salida por el WDT causará diversas acciones dependiendo del modo de ahorro de energía que esté activo cuando se desborde.

Si el dispositivo no está ejecutando código (en cualquier modo reposo o en el modo sleep), el desbordamiento dará lugar a una salida del modo ahorro de energía. Si el dispositivo está ejecutando código (cualquiera de los modos de ejecución), el desbordamiento dará lugar a un reset del WDT.

El contador de tiempo y el postscaler del WDT se borran una instrucción SLEEP o CLRWD, la pérdida de la fuente de reloj seleccionada (si se permite el monitor de reloj de seguridad) y la modificación de los bit IRCF en el OSCCON si el bloque del oscilador interno es la fuente del reloj del dispositivo.

3.5.3 SALIDA POR RESET

Normalmente, el dispositivo se resetea con el contador de inicio (OST) hasta que el reloj primario esté listo. En ese momento, el bit OSTS se activa y el dispositivo comienza a ejecutar código. Si el bloque del oscilador interno es la nueva fuente del reloj, el bit IOFS se activa en lugar de otro.

El retraso entre el reset hasta que comienza la ejecución de código depende de las fuentes de reloj de antes y después del reinicio y del tipo de oscilador si la nueva fuente de reloj es el oscilador primario.

La ejecución del código puede comenzar antes de que el reloj primario este listo. Si el inicio con dos velocidades o el monitor de reloj de seguridad están activos, el dispositivo puede comenzar la ejecución tan pronto como la fuente del reset haya desaparecido. La ejecución se controla con el multiplexor INTOSC conducido por bloque del oscilador interno. La ejecución la controla por el bloque del interno oscilador hasta que el reloj primario esté listo o entre en un modo de ahorro de energía antes de que el reloj primario esté listo; el reloj primario se desactiva.

3.5.4 SALIDA SIN UN RETRASO DEL INICIO DEL OSCILADOR

Algunas salidas de los modos de ahorro de energía no llaman al OST. Hay dos casos:

- El modo PRI_IDLE donde la fuente primaria de reloj no se detiene
- La fuente primaria de reloj no es ni XT ni HS

En estas situaciones, la fuente primaria no necesita un retraso en el reinicio, porque está funcionando (en el modo PRI_IDLE) o porque normalmente no necesita un retraso en el inicio (el modo EC o cualquier modo del bloque interno). Sin embargo, se necesita un retraso T_{CSD} después de reiniciar el sistema para dejar el modo sleep o reposo para permitir que la CPU se prepare para ejecutar una instrucción. La ejecución de la instrucción se reanuda en el primer ciclo de reloj después del retraso.

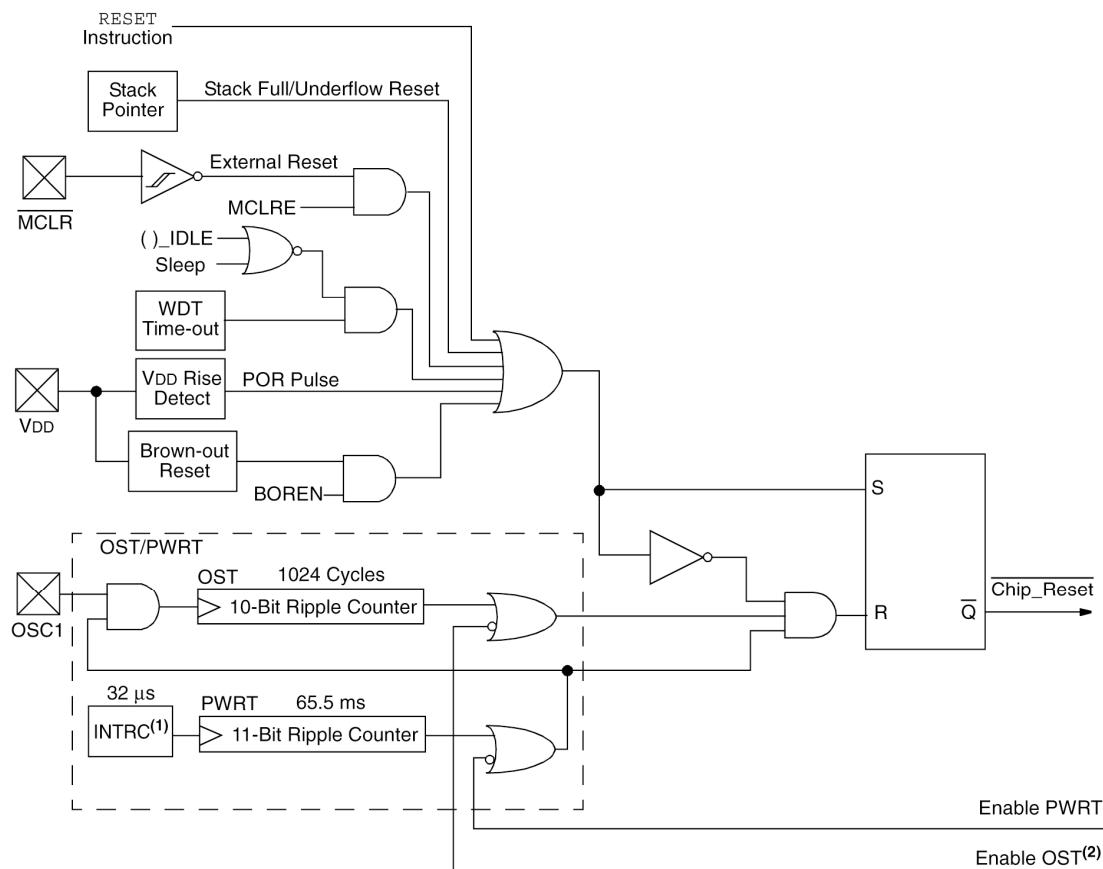
4.0 RESET

Los dispositivos PIC18F2455/2550/4455/4550 distinguen entre las distintas clases de reset:

- Reset por fallo de energía (POR)
- Reset por MCLR* durante la operación normal
- Reset de MCLR* durante modos ahorro de energía
- Reset por el perro guardián (WDT) (durante ejecución)
- Reset por cese de energía (BOR)
- Instrucción RESET
- Reset por desbordamiento la Pila
- Reset por vaciado de la Pila

Esta sección discute los resets generados por MCLR*, POR y BOR y cubre la operación de los varios contadores de tiempo inicio.

FIGURA 4-1: DIAGRAMA SIMPLIFICADO DEL CIRCUITO DE RESET DEL CHIP



4.1 REGISTRO RCON

Los resets del dispositivo se siguen con el registro RCON (registro 4-1). Los cinco bits más bajos del registro indican que ha ocurrido un reset. En la mayoría de los

casos, estos bits se borran con el reset y se deben activar por software después. El estado de estos flags se puede leer para indicar el tipo de reset que acaba de ocurrir.

El registro de RCON también tiene bits de control para elegir la prioridad de la interrupción (IPEN) y del software de control del BOR (SBOREN).

REGISTRO 4-1. RCON: Registro de control de reset

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
IPEN	SBOREN	-	RI*	TO*	PD*	POR*	BOR*
BIT 7	IPEN:						
BIT 6	SBOREN:						
BIT 4	RI*:						
BIT 3	TO*:						
BIT 2	PD*:						
BIT1	POR*:						
BIT0	BOR*:						

Nota 1: Se recomienda que el bit POR* esté fijado después de detectar el reset para poder detectar otros resets.

2: Normalmente, cuando ocurre un reset por cese de energía BOR* es ‘0’ y POR* ‘1’ (suponiendo que POR* se pone a ‘1’ después de un reset por subida de tensión).

4.2 MASTER CLEAR RESET (MCLR*)

El pin MCLR* proporciona un método para accionar un reset externo del dispositivo. El reset se genera poniendo el pin a '0'. Estos dispositivos tienen un filtro antirruidos en la trayectoria del reset de MCLR* que detecta y no hace caso de pulsos pequeños. El pin MCLR* no se pone a cero con ningún reset interno, incluyendo el

WDT. En los dispositivos PIC18F2455/2550/4455/4550, la entrada MCLR* se puede inhabilitar con la configuración del bit MCLRE. Cuando MCLR* está desactivado, el pin se convierte en una entrada digital.

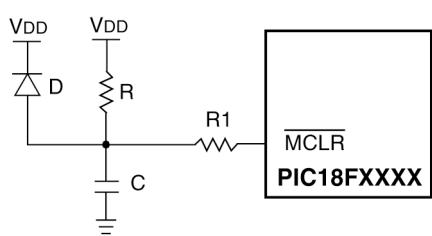
4.3 RESET POR SUBIDA DE TENSIÓN (POR)

Un reset por subida de tensión se genera en el chip cuando V_{DD} sube hasta un umbral. Esto provoca que el dispositivo se inicialice cuando V_{DD} tenga un valor adecuado.

Para aprovecharse del POR, conectar el pin MCLR* a V_{DD} a través de una resistencia ($1\text{k}\Omega$ a $10\text{k}\Omega$). Esto elimina los componentes RC externos que se necesitan para crear un retraso en el reset por subida de tensión.

Cuando el dispositivo comienza la operación normal (es decir, salir de la condición de reset), los parámetros de funcionamiento del dispositivo (tensión, frecuencia, temperatura, etc.) se deben conocer para asegurar el correcto funcionamiento del dispositivo. Si estas condiciones no se conocen, el dispositivo debe estar en reset hasta averiguarlas. Los reset POR se capturan con el bit POR (RCON<1>). El estado del bit se fija a ‘0’ siempre que ocurra un POR; no lo cambia ningún otro reset. POR no se ajusta a ‘1’ con ningún acontecimiento de hardware. Para capturar acontecimientos múltiples, el usuario pone manualmente el bit a ‘1’ por software después de cualquier POR.

FIGURA 4-2: CIRCUITO DE RESET EXTERNO POR SUBIDA DE TENSIÓN (PARA LENTA SUBIDA DE V_{DD})



- Nota:**
- 1:** Se necesita este circuito sólo si V_{DD} sube lentamente.
El diodo D se necesita para descargar C en las bajadas de V_{DD} .
 - 2:** $R < 40\Omega$ Recomendada para que la tensión no dañe el chip.
 - 3:** $R_1 \geq 1\text{k}\Omega$ limitará la corriente en MCLR*.

4.4 RESET POR CESE DE TENSIÓN (BOR)

Los dispositivos PIC18F2455/2550/4455/4550 tienen un circuito BOR en ejecución que provee al usuario un número de opciones de configuración y de ahorro de energía. El BOR se controla con los bits BORV1:BORV0 y BOREN1:BOREN0. Hay un total de cuatro configuraciones del BOR.

El umbral del BOR se fija con los bits BORV1:BORV0. Si se permite BOR (cualquier valor de BOREN1:BOREN0 excepto el ‘00’), cualquier tensión V_{DD} por debajo de V_{BOR} durante un tiempo mayor que T_{BOR} resetea el dispositivo. Un reset

puede o no puede ocurrir si V_{DD} baja por debajo de V_{BOR} durante un tiempo menor que T_{BOR} . El chip permanece en reset hasta que V_{DD} vale más que V_{BOR} .

Si se permite el contador de tiempo de inicio, se invocará después de que V_{DD} sea mayor que V_{BOR} ; entonces mantendrá el chip en reset durante un retraso adicional, T_{PWRT} . Si V_{DD} cae por debajo de V_{BOR} mientras que el contador de tiempo de inicio está funcionando, el chip volverá a un nuevo reset por cese de tensión y el contador de tiempo de inicio se inicializará. Una vez que V_{DD} se levante sobre V_{BOR} , El contador de tiempo de inicio ejecutará un retraso.

V_{BOR} y el contador de inicio (PWRT) se configuran independientemente. Permitir el reset BOR no permite automáticamente el PWRT.

4.4.1 EL SOFTWARE QUE ACTIVA EL BOR

Cuando BOREN1:BOREN0 = 01, el BOR puede activarse o inhabilitarse por el usuario mediante software. Esto se consigue con el bit del control, SBOREN (RCON<6>). Activar SBOREN permite al BOR funcionar como se ha descrito previamente. Borrar SBOREN inhabilita el BOR totalmente. El bit SBOREN funciona solamente en este modo; si no, se lee como ‘0’.

La colocación del BOR bajo control por software da al usuario la flexibilidad adicional de adaptar la aplicación sin tener que reprogramar el dispositivo para cambiar la configuración del BOR. También permite al usuario adaptar el consumo de energía del dispositivo mediante software eliminando la corriente incremental que consume el BOR. Mientras que la corriente de BOR es muy pequeña, puede tener cierto impacto en usos de baja potencia.

Nota: Aun cuando el BOR está bajo control por software, el nivel volátil del reset BOR todavía se fija con los bits BORV1:BORV0. No se puede cambiar por software.

4.4.2 DETECCIÓN DEL BOR

Cuando se permite el BOR, el bit BOR* se resetean en cualquier acontecimiento de BOR o de POR. Esto hace que sea difícil determinar si ha ocurrido un reset BOR por la lectura el estado del bit BOR*. Un método de mayor confianza es comprobar simultáneamente el estado de los bits POR* y BOR*. Esto asume que el bit POR* se pone a ‘1’ por software inmediatamente después de cualquier reset POR. SI BOR es ‘0’ mientras que POR es ‘1’, se puede asegurar que ha ocurrido un reset BOR.

4.4.3 DESACTIVAR BOR EN EL MODO SLEEP

Cuando BOREN1:BOREN0 = 10, los reset BOR están bajo el control del hardware y funciona como se ha descrito previamente. Siempre que el dispositivo entre en modo sleep el BOR se inhabilita automáticamente. Cuando el dispositivo vuelve a cualquier otro modo de funcionamiento, se vuelve a permitir el BOR automáticamente.

Este modo permite aplicaciones en las que el dispositivo se recupere de situaciones de cese de energía, mientras que ejecutan código, cuando el dispositivo requiere la protección BOR. Al mismo tiempo, ahorra energía adicional en modo sleep eliminando la corriente incremental de BOR.

TABLA 4-1: CONFIGURACIÓN DEL BOR

Configuración del BOR			Estado de	Operaciones del BOR
BOREN1	BOREN0	SBOREN (RCON<6>)		
0	0	No disponible	BOR desactivado; para activarlo reprogramar el bit de configuración	
0	1	Disponible	BOR activo en software; se controlan las operaciones por SBOREN.	
1	0	No disponible	BOR activo en hardware en los modos RUN e IDLE, desactivado en el modo SLEEP	
1	1	No disponible	BOR activo en hardware; para desactivarlo reprogramar el bit de configuración.	

4.5 CONTADORES DEL RESET

Los dispositivos PIC18F2455/2550/4455/4550 incorporan tres contadores de tiempo separados en el chip que ayudan a regular el proceso de puesta en marcha del reset. Su función principal es asegurarse de que el reloj del dispositivo es estable antes de que se ejecute el código. Estos contadores de tiempo son:

- Temporizador de inicio (PWRT)
- Oscilador temporizador de arranque (OST)
- Temporizador de control del PLL

4.5.1 TEMPORIZADOR DE INICIO (PWRT)

El temporizador de inicio (PWRT) de los dispositivos PIC18F2455/2550/4455/4550 es un contador de 11 bits que utiliza la fuente de INTRC como entrada de reloj. Este intervalo del tiempo de $2048 \times 32\mu s = 65,6ms$. Mientras que el PWRT está contando, el dispositivo está en reset.

El retraso depende del reloj INTRC y variará en el chip debido a la variación de la temperatura y del proceso.

El PWRT se permite borrando el bit PWRTE.

4.5.2 OSCILADOR TEMPORIZADOR DE ARRANQUE (OST)

El oscilador temporizador de arranque (OST) proporciona un ciclo de 1024 oscilaciones (de la entrada OSC1) se inicia después del retraso PWRT. Esto asegura que el oscilador XTAL o el resonador haya comenzado y se haya estabilizado.

El desbordamiento de OST ocurre solamente en los modos XT, HS y HSPLL y solamente en un inicio o al salir de un modo de ahorro de energía.

4.5.3 TEMPORIZADOR DEL CONTROL DE PLL

Con el PLL permitido en su modo PLL, el retraso que sigue al inicio es levemente diferente de otros modos del oscilador. Un contador de tiempo proporciona un retraso fijo que es suficiente para que el PLL detecte la frecuencia principal del oscilador. Este retraso (T_{PLL}) es típicamente de 2ms y sigue al retraso del inicio del oscilador.

4.5.4 SECUENCIA DE RETRASO

En ciclo inicial, la secuencia del retraso es:

1. Despues de que la condición de POR haya desaparecido, se invoca el retraso PWRT (si está permitido).
2. Entonces, se activa el OST.

El retraso total variará basado en la configuración del oscilador y el estado del PWRT. En los dispositivos en modo RC y con el PWRT inhabilitado, no habrá retraso.

Puesto que los retrasos ocurren por pulsos POR, si MCLR* permanece activo suficiente tiempo, en todos los retrasos se activará. El MCLR* comenzará a ejecutarse inmediatamente. Esto es útil para pruebas o a sincronizar más de un PIC18FXXXX funcionando en paralelo.

TABLA 4-2: RETRASO EN DISTINTAS SITUACIONES

Configuración del Oscilador	Aumento ⁽²⁾ y cese de tensión PWRTE=0	Aumento ⁽²⁾ y cese de tensión PWRTE=1	Salida del Modo de ahorro de energía
HS, XT	66ms ⁽¹⁾ + 1024T _{osc}	1024T _{osc}	1024T _{osc}
HSPLL, XTPLL	66ms ⁽¹⁾ +1024T _{osc} +2ms ⁽²⁾	1024T _{osc} +2ms ⁽²⁾	1024T _{osc} +2ms ⁽²⁾
EC, ECIO	66ms ⁽¹⁾	-	-
ECPLL, ECHO	66ms ⁽¹⁾ +2ms ⁽²⁾	2ms ⁽²⁾	2ms ⁽²⁾
INTIO, INTCKO	66ms ⁽¹⁾	-	-
INTHS, INTXT	66ms ⁽¹⁾ +1024T _{osc}	1024T _{osc}	1024T _{osc}

Nota: 1: 66ms (65,5ms) es el retraso nominal de reinicio (PWRT)

2: 2ms es el tiempo nominal requerido para que el PLL reconozca una frecuencia.

4.6 ESTADO DE LOS REGISTROS EN UN RESET

A la mayoría de los registros no les afectan los resets. Su estado se desconoce en un POR y sin cambios por cualquier otro reset. Los otros registros se fuerzan a un “estado de reset” dependiendo del tipo de reset ocurrido.

A la mayoría de los registros no les afecta un reinicio por WDT, puesto que esto se ve como la reanudación a la operación normal. Los bits estado RCON, RI*, TO*, PD*, POR* y BOR*, se activan o se borran diferentemente en los distintos resets. Estos bits se utilizan en software para determinar la naturaleza del reset.

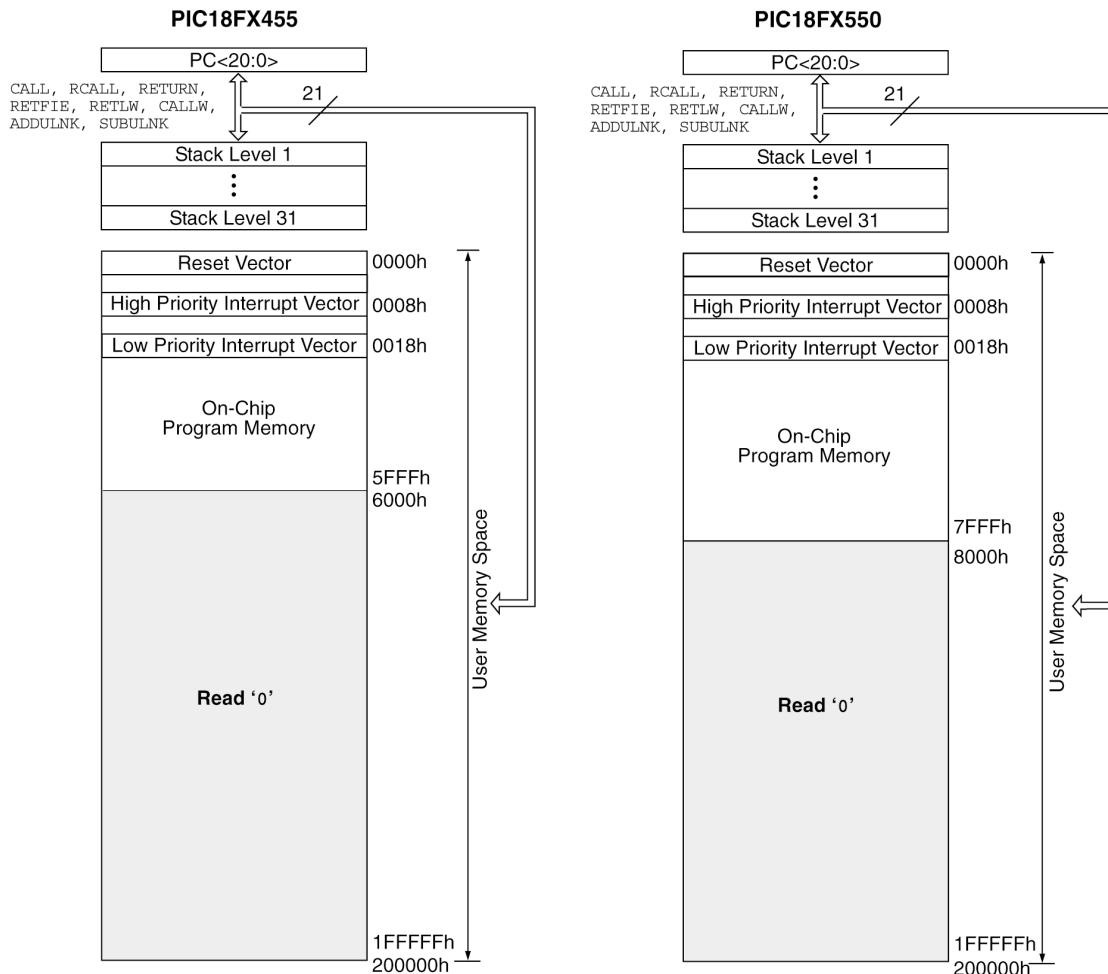
5.0 ORGANIZACIÓN DE LA MEMORIA

Hay tres tipos de memoria en los PIC18:

- Memoria de programa
- RAM de datos
- EEPROM de datos

Como dispositivos de arquitectura Harvard, los buses la memoria de datos y del programa están separados; esto permite el acceso a la vez en las dos memorias. La EEPROM de datos, en la práctica, se puede utilizar como un dispositivo periférico, puesto que se maneja a través de un sistema de registros de control.

FIGURA 5-1: MAPA DE MEMORIA DE PROGRAMA Y DE LA PILA



5.1 ORGANIZACIÓN DE LA MEMORIA DE PROGRAMA

Los microcontroladores PIC18 implementan un contador de programa de 21bits que es capaz de tratar 2Mb memoria de programa. Accediendo a una localización en

medio el límite superior de la memoria implementada físicamente y los 2Mb direccionables se leerá todo ‘0’ (una instrucción NOP).

Los PIC18F2455 y los PIC18F4455 cada uno tienen 24kb de memoria flash y pueden almacenar hasta 12.288 instrucciones de palabra única. El PIC18F2550 y PIC18F4550 cada uno tienen 32kb de la memoria flash y puede almacenar hasta 16.384 instrucciones de palabra única. Los dispositivos PIC18 tienen dos vectores de interrupción. En un reset, la dirección del vector está en 0000h y los vectores de interrupción están en las direcciones 0008h y 0018h.

5.1.1 CONTADOR DE PROGRAMA

El contador de programa (PC) especifica la dirección de la instrucción que se va a ejecutar. El PC es de 21bits de ancho separados en tres registros de 8bits. El byte bajo, conocido como registro PCL, es legible y escribible. El byte alto, o registro PCH, contiene los bits<15:8> del PC; no es directamente ni legible ni escribible. Las actualizaciones del registro PCH se realizan a través del registro de PCLATH. El byte superior se llama PCU. Este registro contiene los bits<20:16> del PC; no es directamente ni legible ni escribible. Las actualizaciones del registro PCU se realizan a través del registro PCLATU.

El contenido de PCLATH y de PCLATU se transfiere al contador de programa por cualquier operación que escriba el PCL. Semejantemente, los dos bytes superiores del contador de programa se transfieren al PCLATH y al PCLATU por cualquier operación que lea el PCL.

El PC trata bytes en la memoria del programa. Para evitar que el PC no se alinee bien con la instrucción, el bit menos significativo de PCL está fijado a un valor de ‘0’. El PC se incrementa de 2 en 2 para tratar instrucciones secuenciales en la memoria de programa.

Las rutinas de salto CALL, los RCALL y GOTO escriben en el contador de programa directamente. Con estas instrucciones, el contenido de PCLATH y PCLATU no se transfieren al contador de programa.

5.1.2 DEVOLVER LA DIRECCIÓN DE LA PILA

La pila almacena la dirección de hasta 31 saltos. El PC se posiciona sobre la pila cuando se ejecuta una instrucción CALL o RCALL o una llamada a una interrupción. El contenido de la pila se devuelve al CP con las instrucciones RETURN, RETLW o una instrucción RETFIE. A los registros PCLATU y PCLATH no les afectan las instrucciones CALL ni RETURN.

La pila funciona como 31 palabras por 21bits de RAM y 5bits de Stack Pointer, STKPTR. El espacio de la pila no es parte de la memoria de programa o de datos. La pila es legible y escribible y la última dirección de la pila es legible y escribible con Registros especiales de Top-of-Stack. Los datos se pueden guardar, o utilizarlos en la pila, usando éstos registros.

Un tipo instrucción que guarda una dirección en la pila es CALL. El puntero de la pila se incrementa y la localización que señalaba por el puntero de la pila se escribe con el contenido del PC (que señala ya a la instrucción que sigue a CALL). Con la instrucción RETURN se utiliza el contenido de la pila. El contenido de la localización señalado por el STKPTR se transfieren al PC y entonces el stack pointer decremente.

El stack pointer se inicializa con ‘00000’ después de cada reset. No hay RAM asociado a la localización que corresponde con un valor del stack pointer de ‘00000’; esto sólo es un valor de reset. Los bits de estado indican si la pila está llena, ha desbordado o tiene está vacía y tiene que devolver una dirección (underflow).

FIGURA 5-2: DEVOLUCIÓN DE LA DIRECCIÓN DE LA PILA Y LOS REGISTROS ASOCIADOS



5.1.2.1 ACCESO AL ÚLTIMO VALOR DE LA PILA

Solamente el último valor de la pila (TOS) es legible y escribible. Un sistema de tres registros, TOSU:TOSH:TOSL, llevan a cabo el contenido de la localización de la pila señalado por el registro STKPTR. Esto permite que los usuarios pongan una dirección en la pila mediante software en caso de necesidad. Después de que una CALL, un RCALL o una interrupción, el software puede guardar el valor en la pila escribiendo los registros TOSU:TOSH:TOSL. Una vez ejecutada la rutina, por software, se puede devolver los valores de TOSU:TOSH:TOSL y regresar al lugar del salto en el programa.

El usuario debe inhabilitar las interrupciones globalmente y permitir los bits que tengan acceso a la pila para prevenir errores incontrolados en la pila.

5.1.2.2 RETORNO DEL STACK POINTER (STKPTR)

El registro STKPTR (registro 5-1) contiene el valor del stack pointer, el bit de estado STKFUL (la pila está completa) y el bit de estado STKUNF (desbordamiento de capacidad inferior de la pila). El valor del stack pointer puede ser de 0 a 31. Se incrementa el stack pointer antes de que los valores se almacenen en pila y se decrementa después de que los valores de la pila se utilicen. En reset, el valor del stack pointer será cero. El usuario puede leer y escribir el valor del stack pointer. Esta característica se puede utilizar por un sistema operativo en tiempo real (RTOS) para controlar la pila.

Cuando el PC se almacene en la pila 31 veces (sin hacer utilizar ninguno de los valores de la pila), se activa el bit STKFUL. El bit STKFUL se borra por software o por un POR.

La acción que ocurre cuando se llene la pila depende por completo del estado del bit de configuración STVREN (permite el desbordamiento de la pila). Si se activa STVREN (defecto), el 31º almacenaje incrementará (PC + 2) el valor sobre de la pila, activará el bit STKFUL y reseteará el dispositivo. El bit STKFUL seguirá siendo '1' y el stack pointer se borrará.

Si STVREN es cero, el bit STKFUL se activará en el 31º almacenamiento y el stack pointer incrementará a 31. Ningún dato adicional sobrescribirá el dato 31º y el STKPTR permanecerá en 31.

Cuando la pila se ha liberado, el siguiente almacenaje devolverá un cero al PC y activa el bit STKUNF, mientras que el stack pointer permanece en cero. Seguirá activo el bit STKUNF hasta que se borre por software u ocurra un POR.

Nota: Devolver un valor de cero al PC en un underflow tiene el efecto de dirigir al programa al vector de reset, donde las condiciones de la pila se pueden verificar y se pueden tomar las acciones apropiadas. Esto es no igual que un reset, no les afecta al contenido de los SFRs.

5.1.2.3 INSTRUCCIONES PUSH Y POP

Puesto que el Top-of-Stack es legible y escribible, tiene la capacidad de almacenar valores sobre la pila y quitarlos, sin que afecte a la ejecución de programa normal, es una característica deseable. El sistema de instrucción PIC18 incluye dos instrucciones, PUSH y POP, que permiten manipular la TOS por software. TOSU, TOSH y TOSL se pueden modificar poniendo datos o devolviéndolos a la pila.

La instrucción PUSH pone el valor actual del PC en la pila. Este incrementa el stack pointer y carga el valor actual del PC sobre la pila.

La instrucción POP desecha la TOS actual decrementando el stack pointer. El valor que se almacenó anteriormente en la pila convierte en el valor de la TOS.

REGISTRO 5-1. STKPTR: Registro del stack pointer

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
STKFUL	STKUNF	-	SP4	SP3	SP2	SP1	SP0
BIT 7	STKFUL:		Bit señalizador del llenado de la pila: (Borrado por software)				
			1= La pila está llena o sobresaturada				
			0= La pila no está llena ni sobresaturada				
BIT 6	STKUNF:		Bit señalizador del underflow: (Borrado por software)				
			1= Ha ocurrido un undeflow				
			0= No ha ocurrido un underflow				
BIT 4-0	SP4:SP0:		Bits de localización del stack pointer				

5.1.2.4 RESETS POR LLENADO DE LA PILA Y POR UNDERFLOW

Los reset por desbordamiento y por underflow de la pila se activan activando el bit STVREN del registro de configuración 4L. Cuando se activa STVREN, un llenado o vaciado de la pila activará el bit apropiado STKFUL o STKUNF y después hará un reset en el dispositivo. Cuando STVREN está borrado, al llenar o vaciar la pila se activará el bit apropiado STKFUL o STKUNF pero no se reseteará el dispositivo. Los bits STKFUL o STKUNF se borran por software o en un reinicio.

5.1.3 PILA DE ALMACENACIN RPIDO

Un almacenamiento rápido en la pila se proporciona con los registros ESTADO, WREG y BSR para lograr una “vuelta rápida” al lugar del salto. Cada pila es de un nivel de profundidad y no es ni legible ni escribible. Se carga con el valor actual del registro correspondiente cuando se ejecuta una interrupción. Todas las fuentes de la interrupción almacenarán valores en los registros de la pila. Los valores cargan nuevamente dentro de sus registros se utiliza la instrucción RETFIE o FAST para volver de la interrupción.

Si se permiten las interrupciones bajas y prioritarias, los registros de la pila no se pueden utilizar fiablemente para volver de las interrupciones de prioridad baja. Si ocurre una interrupción prioritaria mientras que se mantiene una interrupción de baja prioridad, los valores del registro de la pila almacenados por la interrupción de prioridad baja serán eliminados y se almacenarán los de la alta. En estos casos, el usuario debe almacenar los datos de los registros en memoria.

Si las interrupciones de alta prioridad no se utilizan, todas las interrupciones pueden utilizar la pila almacenamiento rápido para volver de la interrupción. Si no se utilizan las interrupciones, la pila de almacenamiento rápido se puede usar para restaurar los registros ESTADO, WREG y BSR al final de una llamada de un subprograma. Para utilizar la pila de almacenamiento rápido en una llamada de un subprograma, hay que usar una instrucción CALL LABEL, FAST donde LABEL es el

nombre de la subrutina se debe utilizar para guardar los registros de ESTADO, WREG y BSR en la pila de almacenamiento rápido. Para restaurar estos registros hay que utilizar una instrucción RETURN, FAST.

EJEMPLO 5-1: CODIGO DE EJEMPLO DE ACCESO A LA PILA DE ALMACENAMIENTO RÁPIDO

CALL SUB1, FAST	;Se guardan en el registro de la pila de acceso rápido los ;registros ESTADO, WREG y BSR
.....	
.....	
SUB1	
.....	
RETURN, FAST	;Devolvemos los valores guardados en la pila de acceso ;rápido

5.1.4 OPERACIONES DE BÚSQUEDA EN LAS TABLAS DE MEMORIA DE PROGRAMA

Puede haber programas que en determinadas situaciones requieran la creación de estructuras de datos, o tablas con búsqueda, dentro de la memoria del programa. En los dispositivos PIC18, las operaciones de búsqueda en tablas se puede implementar de dos maneras:

- GOTO controlado
- Lectura de la tabla

5.1.4.1 GOTO CONTROLADO

Un GOTO controlado se logra agregando una compensación al contador de programa.

Una opción de búsqueda en la tabla se puede conseguir con una instrucción ADDWF PCL y un grupo de instrucciones RETLW nn. El registro W se carga con una compensación en la tabla antes de ejecutar la llamada a esa tabla. La primera instrucción de la rutina llamada es la instrucción ADDWF PCL. La instrucción siguiente ejecutada será una RETLW nn que devuelve “el valor” nn a la función que llama.

El valor compensado (en WREG) especifica el número de bytes que el contador de programa debe avanzar y deben ser los múltiplos de 2 (LSb = 0).

Con este método, solamente se puede almacenar un byte de datos en cada localización y sitio de cada instrucción, en la vuelta se requiere la dirección de pila.

EJEMPLO 5-2: GOTO CONTROLADO USANDO UN VALOR COMPENSADO

ORG	MOVF	OFFSET,W
TABLE	CALL	TABLE
	nn00h	
	ADDWF	PCL
	RETLW	nnh
	RETLW	nnh
	RETLW	nnh

....

....

5.1.4.2 LECTURA Y ESCRITURA EN LAS TABLAS

Un método mejor de almacenar datos en la memoria del programa permite que dos bytes de datos sean almacenados en cada localización de la instrucción.

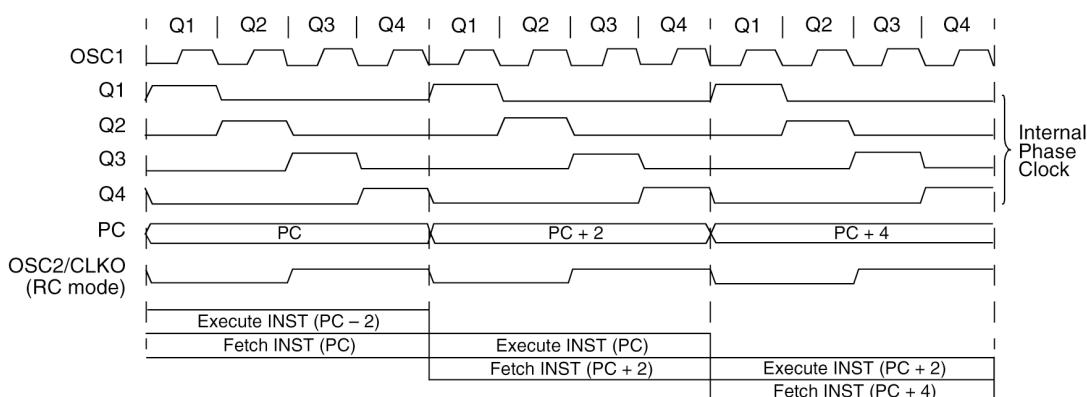
Las operaciones de búsqueda en tabla de datos puede almacenar dos bytes por palabra usando la lectura y la escritura en tabla. El registro puntero de la tabla (TBLPTR) especifica el byte de dirección y el registro de latch de la tabla (TABLAT) contiene los datos que se leen o se escriben en la memoria del programa. Los datos se transfieren a o desde la memoria del programa de byte en byte.

5.2 CICLOS INSTRUCCIÓN EN LOS PIC18

5.2.1 ESQUEMA DE RELOJ

La entrada de reloj del microcontrolador, si es una fuente interna o externa, se divide por cuatro internamente para generar cuatro relojes sin desfase (Q1, Q2, Q3 y Q4). Internamente, el contador de programa se incrementa en cada Q1; en Q4 se trae la instrucción de la memoria del programa y almacenado en el registro de la instrucción (IR). Se descifra la instrucción y se ejecuta durante los siguientes Q1 hasta Q4.

FIGURA 5-3: CICLO RELOJ/INSTRUCCIÓN



5.2.2 INSTRUCCIÓN FLOW/PIPELINING

Un “ciclo de instrucción” consiste en cuatro ciclos de Q: de Q1 hasta Q4. La instrucción trae y se ejecuta es controlada de manera que traer tome un ciclo de la instrucción, mientras que el descifrar y ejecutar tome los otros ciclos de la instrucción. Sin embargo, debido al pipelining, cada instrucción se ejecuta con eficacia en un ciclo. Si una instrucción provoca que el contador de programa cambie (Ej., GOTO), entonces se requieren dos ciclos para terminar la instrucción. El ciclo de traer comienza con el contador de programa (el PC) incremento en Q1. En el ciclo de la ejecución, la instrucción traída se guarda en el registro de la instrucción (IR) en el ciclo Q1. Esta

instrucción después se descifra y se ejecuta durante Ciclos Q2, Q3 y Q4. La memoria de los datos se lee durante Q2 (operando leído) y escrito durante Q4 (destinación escribir).

EJEMPLO 5-3: INSTRUCCIÓN PIPELINE FLOW

	Tcy0	Tcy1	Tcy2	Tcy3	Tcy4	Tcy5
1. MOVLW 55h	Fetch 1	Execute 1				
2. MOVWF PORTB		Fetch 2	Execute 2			
3. BRA SUB_1			Fetch 3	Execute 3		
4. BSF PORTA, BIT3 (Forced NOP)				Fetch 4	Flush (NOP)	
5. Instruction @ address SUB_1					Fetch SUB_1	Execute SUB_1

Nota: Todas las instrucciones son de un solo ciclo, excepto las de salto. Estas tienen dos ciclos desde que se busca la instrucción se utiliza hasta que la instrucción nueva se busca y se ejecuta.

5.2.3 INSTRUCCIONES EN LA MEMORIA DE PROGRAMA

A la memoria de programa se dirige en bytes. Las instrucciones se almacenan en dos o cuatro bytes en la memoria de programa. El byte menos significativo de una instrucción se almacena siempre en una posición de memoria del programa con una dirección ($LSb = 0$). Para mantener la alineación con límites de la instrucción, el PC se incrementa en múltiplos de 2 y el LSb siempre se leerá ‘0’.

Las instrucciones CALL y GOTO tienen la dirección de programa fija en la instrucción. Puesto que las instrucciones se almacenan siempre en direcciones de memoria, los datos contenidos en la instrucción son una dirección. La dirección de la instrucción se escribe en $PC<20:1>$, que tiene acceso a la dirección deseada del byte en la memoria del programa. Funciona de manera semejante las instrucciones de brinco del programa, que codifican la dirección. El valor almacenado en una instrucción de brinco representa el número de instrucciones de una palabra que el PC brincará.

FIGURA 5-4: INSTRUCCIONES EN LA MEMORIA DE PROGRAMA

Word Address ↓

Program Memory Byte Locations →		LSB = 1	LSB = 0	
				000000h
				000002h
				000004h
				000006h
Instruction 1:	MOVLW	055h	0Fh	55h
Instruction 2:	GOTO	0006h	EFh	03h
Instruction 3:	MOVFF	123h, 456h	F0h	00h
			C1h	23h
			F4h	56h
				000008h
				00000Ah
				00000Ch
				00000Eh
				000010h
				000012h
				000014h

5.2.4 INSTRUCCIONES DE DOS PALABRAS

El sistema estándar del PIC18 tiene cuatro instrucciones de dos palabras: CALL, MOVFF, GOTO y LSFR. En todos los casos, la segunda palabra de las instrucciones son siempre ‘1111’ como sus cuatro bits más significativos; los otros 12 bits son datos literales, generalmente una dirección de memoria de datos.

El uso del ‘1111’ en los 4 MSbs de una instrucción especifica una forma especial de NOP. Si la instrucción se ejecuta en la secuencia apropiada, inmediatamente después de la primera palabra, se conocen los datos de la segunda palabra y se utilizan como secuencia de la instrucción. Si se salta la primera palabra por alguna razón y la

segunda palabra se ejecuta por sí sola, se lee un NOP. Esto es necesario para los casos cuando la instrucción de dos palabras es precedida por una instrucción condicional que cambie el PC.

EJEMPLO 5-4: INSTRUCCIONES DE DOS PALABRAS

CASO 1:

Código máquina:	Código fuente:		
0110011000000000	TSTFSZ	REG1	;La RAM está en la localización 0?
1100000100100011	MOVFF	REG1,REG2	;No, salta esta instrucción
1111010001010110			;ejecuta esta otra como NOP
0010010000000000	ADDWF	REG3	;Continua el código

CASO 2:

0110011000000000	TSTSZ	REG1	;La RAM está en la localización 0?
1100000100100011	MOVFF	REG1,REG2	;Si, ejecuta la instrucción
1111010001010110			;2ª palabra de la instrucción
0010010000000000	ADDWF	REG3	;Continua el código

5.3 ORGANIZACIÓN DE LA MEMORIA DE DATOS

Nota: La operación de algunos aspectos de la memoria de datos se cambia cuando el PIC18 permite el sistema de instrucción extendida.

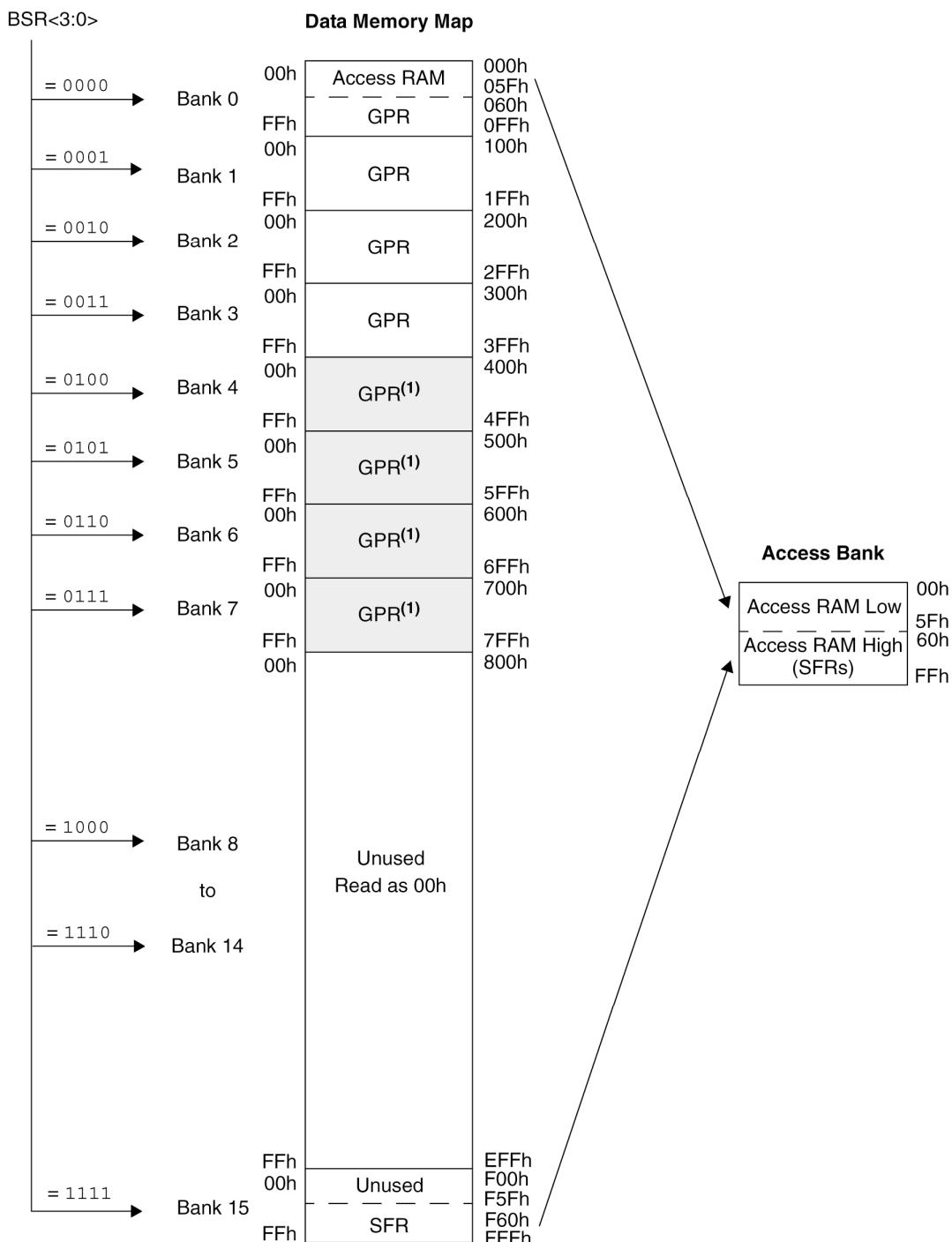
La memoria de datos en los dispositivos PIC18 se implementa como RAM estática. Cada registro en la memoria de datos tiene una dirección de 12 bits, permitiendo hasta 4096 bytes de memoria de los datos. La memoria se divide en 16 bancos que contienen 256 bytes cada uno. Los dispositivos PIC18F2455/2550/4455/4550 ponen ocho bancos completos en ejecución, para un total de 2048 bytes.

La memoria de datos contiene los registros de función especial (SFRs) y los registros de fines generales (GPRs). Los SFRs se utilizan para el control y el estado del regulador y de las funciones periféricas, mientras que los GPRs se utilizan para el almacenaje de datos y las operaciones para salvar datos del usuario. Cuando se lee una localización no implementada aparecerán '0's.

El sistema y la arquitectura de la instrucción permiten operaciones a través de todos los bancos. Se puede tener acceso a la memoria entera de datos por modos de direccionamiento directo, indirecto o puestos en un índice.

Para asegurarse de que los registros de uso general (SFRs y elegido GPRs) se puede alcanzar en un solo ciclo, los dispositivos PIC18 tienen un banco de acceso en ejecución. Éste es una memoria de 256 bytes que proporciona el acceso rápido a SFRs y la parte más baja del banco 0 de GPR sin usar BSR.

FIGURA 5-5: MAPA DE LA MEMORIA DE DATOS



Cuando a=0:

Se ignora el BSR y usamos el acceso al banco.

Los primeros 96bytes de la RAM son de propósito general (del banco 0)

Los 160bytes restantes son registros de funciones especiales (banco 15)

Cuando a=1:

El BSR especifica el banco usado en la instrucción.

5.3.1 RAM DEL USB

Los bancos 4 a 7 de la memoria de datos están mapeados a un puerto dual especial de RAM. Cuando el módulo USB está desactivado, los GPRs en estos bancos se utilizan como cualquier otro GPR en la memoria de datos.

Cuando se permite el módulo USB, la memoria en éstos bancos se asigna como RAM de almacenamiento intermedio para las operaciones del USB. Esta área se comparte entre el núcleo del microcontrolador y el motor de la interfaz en serie (SIE) del USB y se utiliza para transferir datos directamente entre los dos.

En teoría, se pueden utilizar las áreas de la RAM del USB que no se asignan como Buffers del USB para almacenar variables. En la práctica, la naturaleza dinámica de la asignación del almacenamiento intermedio hace que sea peligroso. Además, el banco 4 se utiliza para controlar el almacenamiento intermedio del USB cuando se permite el módulo y no se debe utilizar para otros propósitos durante ese tiempo.

5.3.2 REGISTRO SELECTOR DEL BANCO (BSR)

Las áreas grandes de la memoria de datos requieren un eficiente esquema de dirección para hacer el acceso rápido a cualquier dirección posible. Idealmente, esto significa que no se tiene que proporcionar una dirección entera para cada operación de lectura o escritura. En los dispositivos PIC18, esto se logra con un esquema de los bancos de la RAM. Esto divide la memoria en 16 bancos contiguos de 256 bytes. Dependiendo de la instrucción, cada localización puede tratar directamente una dirección completa de 12 bits, o una dirección de orden inferior de 8 bits y un puntero del banco de 4 bits.

La mayoría de las instrucciones en el sistema del PIC18 hacen uso del puntero de banco, conocido como el registro selector del banco (BSR). Este SFR controla los 4 bits más significativos de la dirección de localización; la instrucción incluye los ocho bits menos significativos. Solamente los cuatro bits más bajos del BSR están implementados (BSR3:BSR0). No se utilizan los cuatro bits altos; leerán ‘0’ y no se pueden escribir. El BSR se puede cargar directamente usando la instrucción MOVLB.

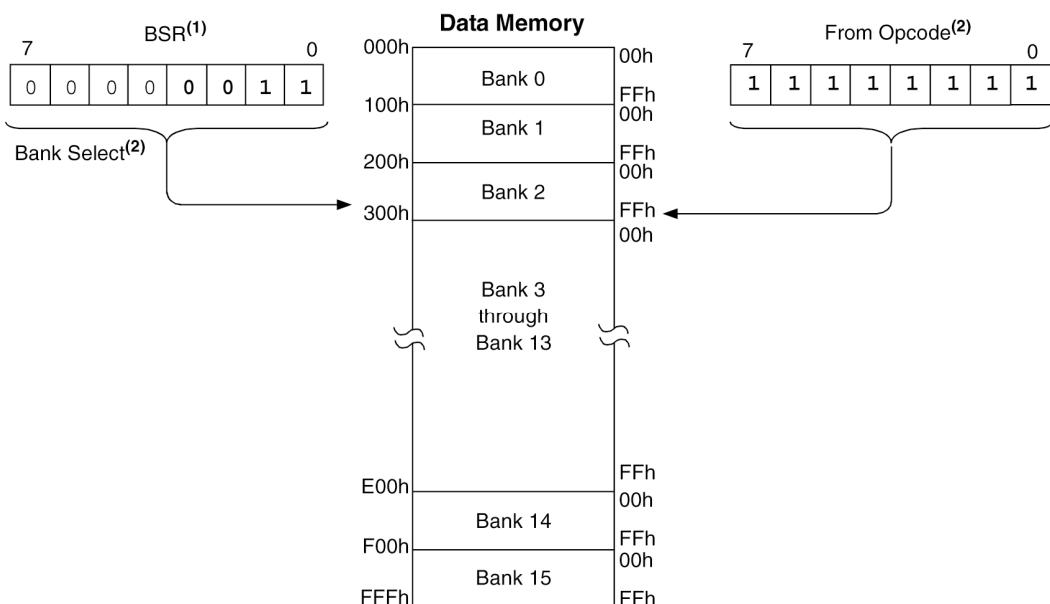
El valor del BSR indica el banco en la memoria de datos. Los ocho bits de la instrucción muestran la localización en el banco y se pueden conocer como el límite bajo del banco.

Como hasta dieciséis registros pueden compartir la misma dirección de orden inferior, el usuario debe tener cuidado de asegurarse de que está seleccionado el banco apropiado antes de leer o escribir datos. Por ejemplo, escribiendo que los datos del programa de una dirección de 8 bits son F9h, mientras que el BSR es 0Fh, reseteará el contador de programa.

Mientras que cualquier banco se puede seleccionar, sólo esos bancos que se ponen en ejecución se pueden leer o escribir. Cuando se escriba en los bancos no implementados no ocurrirá nada, mientras que cuando se lean se devolverán ‘0’s. Sin embargo, al registro ESTADO le afecta como si la operación fuera correcta.

En el núcleo del sistema de instrucciones de los PIC18, solamente la instrucción MOVFF especifica completamente la dirección de 12 bits de los registros de la fuente y del blanco. Con esta instrucción se ignora el BSR cuando se ejecuta. El resto de las instrucciones incluyen solamente la dirección de orden inferior como operando y deben utilizar el BSR o el banco del acceso para localizar sus registros del blanco.

FIGURA 5-6: USO DEL BANCO SELECCIONANDO EL REGISTRO (DIRECCIONAMIENTO DIRECTO)



Nota: 1: El bit de acceso a RAM de la instrucción se puede usar para invalidar el banco seleccionado (BSR<3:0>) del registro de acceso al banco.

2: La instrucción MOVFF inserta los 12bit de dirección en la instrucción.

5.3.3 ACCESO AL BANCO

Mientras que el uso del BSR, con una dirección de 8bits, permite que los usuarios traten la gama entera de memoria de datos, también significa que se debe asegurar el usuario siempre que esté seleccionado el banco correcto. Si no, los datos se pueden leer o escribir en una localización incorrecta. Esto puede ser desastroso si un GPR se va a modificar por una operación porque se escribe un SFR a en lugar de otro. Verificar y/o cambiar el BSR para cada lectura o escritura en la memoria de datos puede resultar ineficaz.

Para mejorar el acceso a las posiciones de memoria de datos de uso general, la memoria de datos se configura con un banco de acceso, que permite a usuarios el acceso a un bloque mapeado de la memoria sin especificar un BSR. El banco de acceso consiste en los primeros 96 bytes de memoria (00h-5Fh) en el banco 0 y los últimos 160 bytes de memoria (60h-FFh) en el bloque 15. Se conoce la mitad inferior como el “acceso RAM” y se compone de los GPRs. La mitad superior es donde los SFRs del dispositivo están mapeados. A estas dos áreas mapeadas en el banco de acceso se puede dirigir con una dirección de 8 bits.

El banco de acceso se utiliza por las instrucciones de núcleo de los PIC18 que incluyen el bit de acceso a RAM (el parámetro ‘a’ dentro de la instrucción). Cuando el ‘a’ es igual a ‘1’, la instrucción utiliza el BSR y la dirección de 8 bits incluidos en el código para la dirección de memoria de datos. Cuando el ‘a’ es ‘0’, sin embargo, se fuerza a utilizar el acceso con las direcciones mapeadas del banco a la instrucción; se ignora el valor actual del BSR.

El usar estas direcciones “forzadas” permite a la instrucción funcionar en una dirección de datos en un solo ciclo sin actualizar primero el bit BSR. Para direcciones de 8bits de 60h y mas, esto significa que los usuarios pueden evaluar y modificar los SFRs más eficientemente. Acceso la a RAM por debajo de 60h es bueno para almacenar los valores que el usuario pueda necesitar rápidamente, por ejemplo de cómputo inmediato resultados o variables comunes del programa. Tener acceso a la RAM también cuenta la rapidez y más ahorro del código y conmutación de variables.

El acceso al banco mapeado es levemente diferente cuando se permite el sistema de instrucciones extendidas (bit de configuración XINST=1).

5.3.4 REGISTRO DE ARCHIVOS DE PROPÓSITO GENERAL

En los dispositivos PIC18 se pudo depositar memoria en el área GPR. Éste es la RAM de datos disponible para el uso de todas las instrucciones. GPRs empieza en el fondo del banco 0 (dirección 000h) y crece hasta la última área de SFR. GPRs no se inicializa por un reinicio y no cambia en ningún otro reset.

5.3.5 REGISTROS DE FUNCIÓN ESPECÍFICA

Los registros de función específica (SFRs) son registros usados por la CPU y los módulos periféricos para controlar las operaciones deseadas del dispositivo. Estos registros se ponen en ejecución como RAM estática en la memoria de datos. SFRs empieza la última dirección de memoria de datos y extiende hacia abajo hasta ocupar el segmento superior del banco 15, de F60h a FFFh.

Los SFRs se pueden clasificar en dos sistemas: los asociados a la funcionalidad del “núcleo” del dispositivo (ALU, reajustes e interrupciones) y los relacionados con las funciones de los periféricos.

Los SFRs se distribuyen por las funciones los periféricos que controlan. Las localizaciones de SFR no utilizadas no están implementadas y se leen como ‘0’.

TABLA 5-1: MAPA DE LOS REGISTROS DE FUNCION ESPECÍFICA

Address	Name	Address	Name	Address	Name	Address	Name	Address	Name
FFFh	TOSU	FDFh	INDF2 ⁽¹⁾	FBFh	CCPR1H	F9Fh	IPR1	F7Fh	UEP15
FFEh	TOSH	FDEh	POSTINC2 ⁽¹⁾	FBEh	CCPR1L	F9Eh	PIR1	F7Eh	UEP14
FFDh	TOSL	FDDh	POSTDEC2 ⁽¹⁾	FBDh	CCP1CON	F9Dh	PIE1	F7Dh	UEP13
FFCh	STKPTR	FDCh	PREINC2 ⁽¹⁾	FBCh	CCPR2H	F9Ch	— ⁽²⁾	F7Ch	UEP12
FFBh	PCLATU	FDBh	PLUSW2 ⁽¹⁾	FBBh	CCPR2L	F9Bh	OSCTUNE	F7Bh	UEP11
FFAh	PCLATH	FDAh	FSR2H	FBAh	CCP2CON	F9Ah	— ⁽²⁾	F7Ah	UEP10
FF9h	PCL	FD9h	FSR2L	FB9h	— ⁽²⁾	F99h	— ⁽²⁾	F79h	UEP9
FF8h	TBLPTRU	FD8h	STATUS	FB8h	BAUDCON	F98h	— ⁽²⁾	F78h	UEP8
FF7h	TBLPTRH	FD7h	TMR0H	FB7h	ECCP1DEL	F97h	— ⁽²⁾	F77h	UEP7
FF6h	TBLPTRL	FD6h	TMR0L	FB6h	ECCP1AS	F96h	TRISE ⁽³⁾	F76h	UEP6
FF5h	TABLAT	FD5h	T0CON	FB5h	CVRCON	F95h	TRISD ⁽³⁾	F75h	UEP5
FF4h	PRODH	FD4h	— ⁽²⁾	FB4h	CMCN	F94h	TRISC	F74h	UEP4
FF3h	PRODL	FD3h	OSCCON	FB3h	TMR3H	F93h	TRISB	F73h	UEP3
FF2h	INTCON	FD2h	HLVDCON	FB2h	TMR3L	F92h	TRISA	F72h	UEP2
FF1h	INTCON2	FD1h	WDTCON	FB1h	T3CON	F91h	— ⁽²⁾	F71h	UEP1
FF0h	INTCON3	FD0h	RCON	FB0h	SPBRGH	F90h	— ⁽²⁾	F70h	UEP0
FEFh	INDF0 ⁽¹⁾	FCFh	TMR1H	FAFh	SPBRG	F8Fh	— ⁽²⁾	F6Fh	UCFG
FE Eh	POSTINC0 ⁽¹⁾	FCEh	TMR1L	FAEh	RCREG	F8Eh	— ⁽²⁾	F6Eh	UADDR
FEDh	POSTDEC0 ⁽¹⁾	FCDh	T1CON	FADh	TXREG	F8Dh	LATE ⁽³⁾	F6Dh	UCON
FECh	PREINC0 ⁽¹⁾	FCCh	TMR2	FACh	TXSTA	F8Ch	LATD ⁽³⁾	F6Ch	USTAT
FE Bh	PLUSW0 ⁽¹⁾	FCBh	PR2	FABh	RCSTA	F8Bh	LATC	F6Bh	UEIE
FEAh	FSR0H	FCAh	T2CON	FAAh	— ⁽²⁾	F8Ah	LATB	F6Ah	UEIR
FE9h	FSR0L	FC9h	SSPBUF	FA9h	EEADR	F89h	LATA	F69h	UIE
FE8h	WREG	FC8h	SSPADD	FA8h	EEDATA	F88h	— ⁽²⁾	F68h	UIR
FE7h	INDF1 ⁽¹⁾	FC7h	SSPSTAT	FA7h	EECON2 ⁽¹⁾	F87h	— ⁽²⁾	F67h	UFRMH
FE6h	POSTINC1 ⁽¹⁾	FC6h	SSPCON1	FA6h	EECON1	F86h	— ⁽²⁾	F66h	UFRML
FE5h	POSTDEC1 ⁽¹⁾	FC5h	SSPCON2	FA5h	— ⁽²⁾	F85h	— ⁽²⁾	F65h	SPPCON ⁽³⁾
FE4h	PREINC1 ⁽¹⁾	FC4h	ADRESH	FA4h	— ⁽²⁾	F84h	PORTE	F64h	SPPEPS ⁽³⁾
FE3h	PLUSW1 ⁽¹⁾	FC3h	ADRESL	FA3h	— ⁽²⁾	F83h	PORTD ⁽⁸⁾	F63h	SPPCFG ⁽³⁾
FE2h	FSR1H	FC2h	ADCON0	FA2h	IPR2	F82h	PORTC	F62h	SPPDATA ⁽³⁾
FE1h	FSR1L	FC1h	ADCON1	FA1h	PIR2	F81h	PORTB	F61h	— ⁽²⁾
FE0h	BSR	FC0h	ADCON2	FA0h	PIE2	F80h	PORTA	F60h	— ⁽²⁾

Nota: 1: No es un registro físico.

2: Registros no implementados se leen como ‘0’.

3: Estos registros se implementan en los micros de 40/44 pines.

5.3.6 REGISTRO ESTADO

El registro ESTADO, contiene el estado aritmético de la ALU. Como con cualquier otro SFR, se puede modificar con cualquier instrucción.

Si el registro ESTADO es el destino de una instrucción que afecte a los bits Z, C.C., C, OV o N, no escribirá el resultado de la instrucción, en vez de eso, el registro ESTADO se actualiza según la instrucción realizada. Por lo tanto, el resultado de una instrucción con el registro ESTADO como destino puede ser diferente que de lo previsto. Por ejemplo, CLRF STATUS activará el bit Z y el resto de los bits no cambian ('000u u1uu').

REGISTRO 5-2. STATUS: Registro estado

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
-	-	-	N	OV	Z	DC	C
U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
BIT 4	N:				Bit negativo. Este bit se utiliza en el complemento a 2. Indica cuando el resultado ha sido negativo. 1= Resultado negativo 0= Resultado no negativo		
BIT 3	OV:				Bit de desbordamiento. Este bit se utiliza en el complemento a 2. Indica que una operación de 7bits se ha desbordado. 1= Ha ocurrido un desbordamiento 0= No ha ocurrido un desbordamiento		
BIT 2	Z:				Bits de cero 1= El resultado de una operación aritmética es 0 0= El resultado de una operación aritmética no es 0		
BIT 1	DC:				Dígito de acarreo/Bit BORROW* En BORROW*, se invierte la polaridad. Una se ejecuta una substracción agregando el complemento a 2 del segundo operando. En las instrucciones de rotar (RRF, RLF), este bit se carga con el bit 4 ó 3 del registro fuente. Con las instrucciones ADDWF, ADDLW, SUBLW y SUBWF 1= Ocurrió un acarreo en los 4 bits de menos peso 0= No ocurrió un acarreo en los 4 bits de menos peso		
BIT0	C:				Bit de acarreo/Bit BORROW* En BORROW*, se invierte la polaridad. Una se ejecuta una substracción agregando el complemento a 2 del segundo operando. En las instrucciones de rotar (RRF, RLF), este bit se carga con el bit de menos peso u orden inferior registro fuente. Con las instrucciones ADDWF, ADDLW, SUBLW y SUBWF 1= Ocurrió un acarreo en los bits de mayor peso 0= No ocurrió un acarreo en los bits de mayor peso		

5.4 MODOS DE DIRECCIONAMIENTO DE LOS DATOS

Nota: La ejecución de algunas instrucciones en el núcleo del PIC18 cambia cuando se activan las instrucciones extendidas.

Mientras que la memoria de programa se puede tratar en una sola dirección (a través del contador de programa) la información en la memoria de datos se puede tratar de varias maneras. En la mayoría de las instrucciones, el modo de dirección es fijo. Otras instrucciones pueden utilizar hasta tres modos, dependiendo de cuales utilizan los operandos y si o no el sistema de instrucción extendida está permitido.

Los modos de direccionamiento son:

- Inherente
- Literal
- Directo
- Indirecto

Hay otro modo de direccionamiento, indexación del literal compensado, está disponible cuando se permiten las instrucciones extendidas (configuración del bit XINST=1).

5.4.1 DIRECCIONAMIENTO INHERENTE Y LITERAL

Muchas instrucciones de control del PIC18 no necesitan todos los argumentos; pueden realizar una operación global, que afecte al dispositivo o funcionar implícitamente en un registro. Se conoce este modo de dirección como Dirección inherente. Los ejemplos incluyen SLEEP, RESET y DAW.

Otras instrucciones trabajan de una manera similar pero requieren un argumento explícito adicional en el código máquina. Esto se conoce como modo de direccionamiento literal porque requieren un cierto valor literal como argumento. Los ejemplos incluyen ADDLW y MOVLW, que respectivamente, suman o mueven un valor literal al registro W. Los otros ejemplos incluyen CALL y GOTO, que incluyen 20bits para la dirección de memoria de programa.

5.4.2 DIRECCIONAMIENTO DIRECTO

El modo de direccionamiento directo especifica el todo o una parte de la fuente y/o de la dirección de destino de la operación dentro del código de programa. Las opciones son especificadas por los argumentos que acompañan la instrucción.

En las instrucciones bit-orientadas y byte-orientadas del sistema de instrucción del núcleo de los PIC18, utilizan una cierta versión de direccionamiento directo por defecto. Todas estas instrucciones incluyen una cierta dirección literal de 8 bits como su byte menos significativo. Esta dirección especifica una dirección en uno de los bancos de la RAM o una localización en el banco de acceso como el código de la instrucción.

El bit de acceso a RAM la ‘a’ determina cómo es la dirección interpretada. Cuando la ‘a’ es el ‘1’, el contenido del BSR se utiliza con la dirección para determinar los 12bit completos de la dirección del registro. Cuando la ‘a’ es ‘0’, la dirección se interpreta como si fuese un registro en el banco de acceso. El direccionamiento que utiliza el acceso a RAM se conoce como MODO DE DIRECCIONAMIENTO DIRECTO FORZOSO.

Algunas instrucciones, tales como MOVFF, incluyen la dirección de 12 bits (fuente o destinación) entera en su código máquina. En estos casos, el BSR se ignora completamente.

El destino de los resultados de la operación se determina por el bit de destino ‘d’. Cuando el ‘d’ es ‘1’, los resultados se almacenan en el registro fuente, sobrescribiendo su original contenido. Cuando el ‘d’ es ‘0’, los resultados se almacenan

dentro del registro W. Las instrucciones sin la ‘d’ tienen un destino implícito en la instrucción; su destino es cualquier registro del blanco que funciona o el registro W.

5.4.3 DIRECCIONAMIENTO INDIRECTO

El direccionamiento indirecto permite que el usuario tenga acceso a una localización en la memoria de datos sin dar una dirección fija en la instrucción. Esto se hace usando los registros selectores del archivo (FSRs) como punteros a las localizaciones que se leerán o se escribirán. Puesto que los FSRs están situados en la RAM como registros de función específica, sólo se pueden manipular directamente bajo control de programa. Esto hace que los FSRs sean muy útiles en estructuras de datos, tales como tablas y órdenes en memoria de datos.

Los registros para el direccionamiento indirecto están puestos con los operandos de archivo indirectos (INDFs) esa manipulación automática del permiso en ejecución del valor del puntero con auto-incremento, auto-decremento o compensando con otro valor. Esto permite un código eficiente, usando lazos.

EJEMPLO 5-5: COMO BORRAR RAM (BANCO 1) UTILIZANDO EL DIRECCIONAMIENTO INDIRECTO

```

LFSR FSR0,100h
NEXT CLRF      POSTINC0 ;Borrar el registro INDF e incrementar el puntero
    BTFSS      FSR0H,1   ;¿Todo el banco 1 en blanco?
    BRA       NEXT     ;No, borrar el siguiente
CONTINUE

```

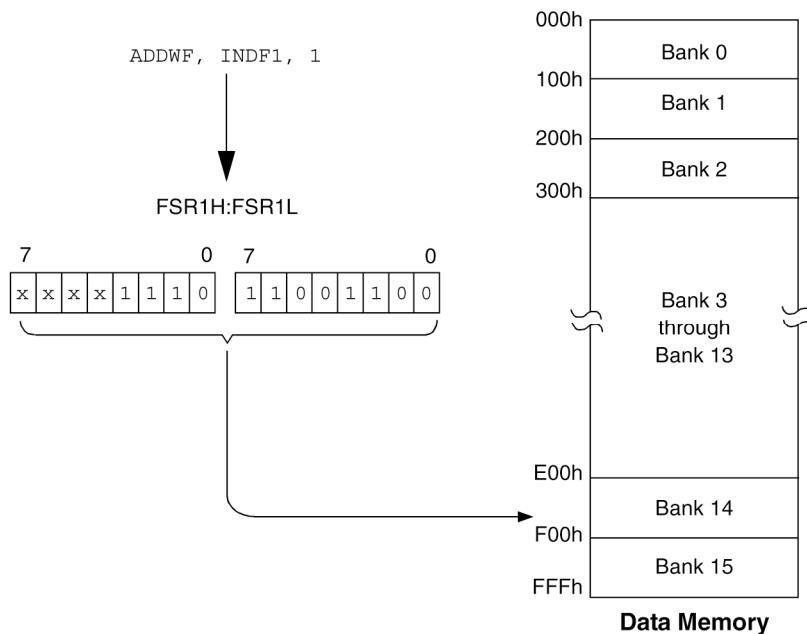
5.4.3.1 REGISTROS FSR Y EL OPERANDO INDF

En el núcleo del direccionamiento indirecto hay tres sistemas de registros: FSR0, FSR1 y FSR2. Cada uno representa un par de registros de 8 bit: FSRnH y FSRnL. Los cuatro bits superiores del registro FSRnH no se utilizan, así que cada par de FSR tiene un valor de 12 bits. Esto representa un valor que puede direccionar la gama entera de memoria de datos de una manera lineal. Los pares del registro FSR sirven como punteros de las posiciones de memoria de datos.

El direccionamiento indirecto se logra con un sistema de Operandos de archivo indirectos, INDF0 con INDF2. Éstos se pueden tomar como registros “virtuales”; están mapeados en el espacio de SFR pero no se ponen en ejecución físicamente. En la lectura o escritura de un registro INDF particular se accede a su par correspondiente. Leer en INDF1, por ejemplo, lee los datos de la dirección indicada por FSR1H:FSR1L. Las instrucciones utilizan los registros INDF mientras que los operandos utilizan el contenido del FSR correspondiente como puntero de la instrucción. El operando de INDF es una manera conveniente de usar los punteros.

Como el direccionamiento indirecto utiliza una dirección completa de 12 bits, las actividades bancarias de la RAM de datos no son necesarias. Así, el contenido actual del BSR y del bit de acceso a RAM no tiene efecto sobre la determinación de la dirección objetivo.

FIGURA 5-7: DIRECCIONAMIENTO INDIRECTO



Usando una instrucción de los registros de direccionamiento indirecto como operando...

...utiliza los 12bit de direccionamiento guardados en el par FSR asociado con este registro...

...para determinar la localización de la memoria de datos que se usará en esta operación.

En este caso, la pareja FSR1 contiene ECCh. Gracias a esto, el contenido de la localización ECCh se asocia al registro W y se guarda en ECh.

5.4.3.2 REGISTROS FSR Y POSTINC, POSTDEC, PREINC Y PLUSW

Además del operando INDF, cada par de registros FSR también tiene cuatro operandos indirectos adicionales. Como INDF, éstos son los registros “virtuales” a los cuales no se puede leer o escribir indirectamente. Al acceder a estos registros se accede realmente al par asociado del registro FSR, pero también realiza una acción específica que almacena valor. Son:

- **POSTDEC:** tiene acceso al valor de FSR, entonces automáticamente lo decrementa en ‘1’ después
- **POSTINC:** tiene acceso al valor de FSR, entonces automáticamente lo incrementa en ‘1’ después
- **PREINC:** incrementa el valor de FSR en ‘1’, después lo utilizan en la operación
- **PLUSW:** suma el valor del registro W (gama de -127 a 128) al del FSR y utiliza el nuevo valor en la operación.

En este contexto, al tener acceso a un registro INDF se utiliza el valor en el FSR sin cambiarlo. Semejantemente, tener acceso a un registro PLUSW da el valor de FSR compensado de este en el registro W; ninguno de los dos valores se cambian realmente en la operación. Tener acceso a los otros registros virtuales cambia el valor de los registros de FSR.

Operaciones en el FSRs con POSTDEC, POSTINC y PREINC afectan el par de registros; es decir, los resets del FSRnL se colocan, de FFh a 00h, lo transportan al registro FSRnH. Por otra parte, los resultados de estas operaciones no cambian el valor de los flags del registro ESTADO (Ej., Z, N, OV, etc.).

El registro PLUSW se puede utilizar para implementar una forma de direccionamiento indexado en la memoria de datos. Manipulando el valor del registro, los usuarios pueden conseguir direcciones fijando el puntero de direcciones. En algunos casos, esto se puede utilizar para poner un control estructurado del programa, por ejemplo pilas del software, dentro de la memoria de datos.

5.4.3.3 OPERACIONES DE FSRs EN FSRs

Las operaciones de direccionamiento indirecto que apuntan otro FSRs o registros virtuales representan casos especiales. Por ejemplo, usar un FSR para señalar uno de los registros virtuales no dará lugar a operaciones acertadas. Como caso específico, asumir que FSR0H:FSR0L contiene FE7h, la dirección de INDF1. Las tentativas de leer el valor de INDF1, usando INDF0 como operando, devolverán 00h. Tentativas de escribir a INDF1, usando INDF0 como operando, resultará en un NOP.

Por otra parte, usar los registros virtuales para escribir un par de FSR puede no ocurrir tal y como se ha planteado. En estos casos, el valor se escribirá al par de FSR pero sin el incremento o el decremento. Así, escribiendo en INDF2 o en POSTDEC2 escribirá el mismo valor a FSR2H:FSR2L.

Puesto que los FSRs son registros físicos mapeados en el espacio de SFR, se pueden manipular con todas las operaciones directas. Los usuarios deben trabajar proceder cautelosamente en estos registros, particularmente si su código utiliza el direccionamiento indirecto.

Semejantemente, las operaciones por el direccionamiento indirecto se permiten generalmente en el resto de los SFRs. Los usuarios deben tener la precaución de no cambiar inadvertidamente los ajustes que puedan afectar la operación del dispositivo.

5.5 MEMORIA DEL PROGRAMA Y EL SISTEMA DE INSTRUCCIONES EXTENDIDAS

A la operación de la memoria de programa no le afectada el uso del sistema de instrucciones extendidas.

Permitir el sistema de instrucciones extendida agrega ocho comandos de palabras dobles adicionales al sistema de instrucciones del PIC18: ADDFSR, ADDULNK, CALLW, MOVSF, MOVSS, PUSHL, SUBFSR y SUBULNK.

5.6 MEMORIA DE DATOS Y EL SISTEMA DE INSTRUCCIONES EXTENDIDAS

Permitir el sistema de instrucciones extendidas del PIC18 (configuración del bit XINST=1) cambia ciertos aspectos de la memoria de datos y de su dirección. Específicamente, el uso del banco de acceso para muchas de las instrucciones del núcleo de los PIC18 es diferente. Esto se debe a la introducción de un nuevo modo de dirección para la memoria de datos. Este modo también altera el comportamiento de FSR2 que usa direccionamiento indirecto y sus operandos asociados.

Lo que no cambia es lo importante. El tamaño de la memoria de datos no cambia, así como su direccionamiento lineal. El mapa de los SFRs sigue siendo igual. Las instrucciones del núcleo de los PIC18 pueden funcionar en ambos modo de direccionamiento, directo e indirecto; las instrucciones inherentes y literales no cambian. Sigue habiendo un direccionamiento indirecto de FSR0 y FSR1 sin cambios.

5.6.1 DIRECCIONAMIENTO INDEXADO CON LITERAL COMPENSADO

Al permitir el sistema de instrucciones extendidas de los PIC18 cambia el comportamiento de direccionamiento indirecto usando el par de registros FSR2 y sus

operandos de archivos asociados. Bajo las condiciones apropiadas, las instrucciones que utilizan el acceso a banco, -o sea, las instrucciones bit-orientadas y byte-orientadas-pueden invocar una forma de direccionamiento indexado usando una instrucción compensada específica. Este modo de direccionamiento especial se conoce como indexación del literal compensado o modo indexado por literal compensado.

Al usar el sistema de instrucciones extendidas, este modo de direccionamiento requiere lo siguiente:

- Se fuerza el uso del banco de acceso ('a'=0);
- El argumento de dirección es menor o igual a 5Fh.

Bajo estas condiciones, el direccionamiento de las instrucciones no se interpreta como el byte más bajo de una dirección (utilizando el BSR en direccionamiento directo), o como una dirección de 8 bits en el banco de acceso. En lugar, el valor es interpretado como un valor compensado a un puntero de la dirección especificada por FSR2. La compensación y el contenido de FSR2 se suman para obtener la dirección de objetivo de la operación.

5.6.2 INSTRUCCIONES AFECTADAS POR EL MODO INDEXADO POR LITERAL COMPENSADO

Algunas de las instrucciones del núcleo del PIC18 pueden utilizar el direccionamiento directo les afecta el modo indexado por literal compensado. Esto incluye todas las instrucciones byte-orientadas y bit-orientadas, o casi una mitad del sistema de instrucción estándar del PIC18. A las instrucciones que utilizan solamente los modos de direccionamiento inherente o literal no les afecta.

Además, las instrucciones byte-orientadas y bit-orientadas no les afecta si utilizan el banco de acceso (El bit de acceso a RAM es '1') o incluye una dirección del archivo de 60h para arriba. Las instrucciones que resuelven estos criterios continuarán ejecutándose como antes.

Los que desean utilizar las instrucciones byte-orientadas o bit-orientadas en el modo indexado por literal compensando deben observar los cambios de la sintaxis del ensamblador.

FIGURA 5-8: EL COMPARAR TRATANDO LAS OPCIONES PARA LAS INSTRUCCIONES BIT-ORIENTDAS Y BYTE-ORIENTADAS (SISTEMA DE INSTRUCCIONES EXTENDIDAS PERMITIDO)

INSTRUCCIÓN DEL EJEMPLO: ADDWF, f, d, a (Opcode: ffff del 0010 01da)

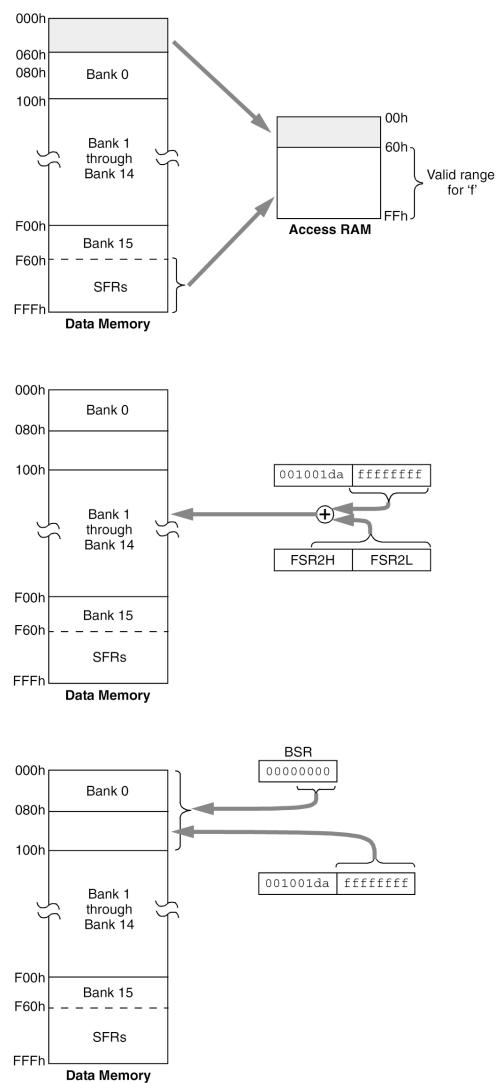
Cuando a=0 y f=60h: La instrucción se ejecuta en el modo directo forzado. Se interpreta ‘f’ como un localización en el acceso a RAM entre 060h y 0FFh. Éste es igual que el SFRs o las localizaciones F60h a 0FFh (banco 15) de la memoria datos.

Las localizaciones debajo de 60h no están disponibles en este modo de direccionamiento.

Cuando a=0 y f=5Fh: La instrucción se ejecuta en el Modo indexado por literal compensado. ‘f’ se interpreta como compensación del valor de dirección en FSR2. Los dos se suman para obtener la dirección objetivo para la instrucción. La dirección puede estar dondequiera dentro la memoria de datos.

Observar que en este modo, la sintaxis correcta es: ADDWF [k], d donde ‘k’ es lo mismo que ‘f’.

Cuando a = 1 (cualquier valor de f): La instrucción se ejecuta en el modo directo (también conocido como Modo largo directo). Se interpreta la ‘f’ como localización en uno de los 16 bancos de la memoria de datos. El banco se elige con el registro BSR. La dirección puede estar en cualquier banco implementado en la memoria de datos.



5.6.3 MAPEO DEL BANCO DE ACCESO EN EL MODO INDEXADO POR LITERAL COMPENSADO

El uso del modo indexado por literal compensado cambia el mapeado de la parte baja de la RAM accesible (00h a 5Fh). En vez de mantener el contenido de la mitad inferior del banco 0, este modo mapea el contenido del banco 0 y otro definido por el usuario “ventana” que se puede localizar en cualquier parte de la memoria de datos. El

valor de FSR2 establece bajar el límite de las direcciones mapeadas en la ventana, mientras que el límite superior se define por FSR2 más 95 (5Fh).

El remapeado del banco de acceso se aplica solamente a las operaciones usando el modo indexado por literal compensado. Operaciones utilizan el BSR (el bit de acceso a RAM es '1') continuarán utilizando el direccionamiento directo. Cualquier operación indirecta o indexada que utilice operandos indirectos (FSR2 incluido) seguirá trabajando con el direccionamiento indirecto estándar. Cualquier instrucción que utilice el banco de acceso, pero incluya una dirección del registro mayor que 05Fh, utilizará el direccionamiento directo y el mapa normal del banco de acceso.

FIGURA 5-9: REMAPEO DEL BANCO DE ACCESO EN EL MODO INDEXADO POR LITERAL COMPENSADO

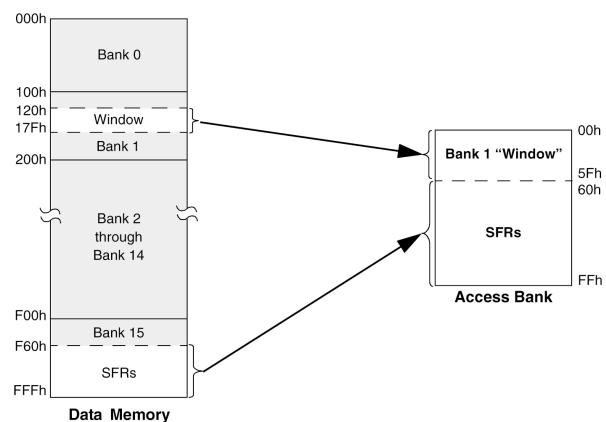
Ejemplo de situación:

ADDWF f,d,a
FSR2H:FSR2L=120h

Las posiciones de la región del puntero FSR2 (120h) a las del puntero mas (17Fh) están mapeadas con la parte de arriba de RAM (000h-05Fh).

Los registros de funciones especiales de F60h a FFFh están mapeadas de 60h a FFh como es normal.

Las direcciones del banco 0 por debajo de 5Fh no están disponibles en este modo. Se pueden direccionar usando el BSR



5.6.4 BSR EN EL MODO INDEXADO POR LITERAL COMPENSADO

Aunque el banco de acceso está remapeado cuando se permite el sistema de instrucciones extendidas, las operaciones del resto de BSR no cambian. El direccionamiento directo, utiliza el BSR para seleccionar el banco de memoria de datos, funciona en la misma manera según lo descrito previamente.

6.0 MEMORIA FLASH DE PROGRAMA

La memoria flash de programa es legible, escribible y borrable, durante la operación normal sobre la gama entera de V_{DD} .

La lectura en la memoria de programa se ejecuta de byte en byte. Al escribir en la memoria de programa se ejecuta en bloques de 32 bytes a la vez. La memoria de programa se borra en bloques de 64 bytes. No se pueden borrar grandes cantidades con el código del usuario.

Escribir o borrar memoria de programa parará las búsquedas de instrucciones hasta que se complete la operación. No se puede acceder a la memoria de programa al escribir o borrar, por lo tanto, código no se puede ejecutar. Un contador interno de la memoria de programa termina las operaciones de escribir y borrar.

No siempre se va a escribir una instrucción válida en la memoria de programa. Ejecutar una localización de la memoria de programa que no sea instrucción válida dará lugar a un NOP.

6.1 ESCRIBIR Y LEER EN TABLAS

Para leer y escribir en la memoria de programa, hay dos operaciones que permiten que el procesador mueva bytes entre la memoria de programa y la RAM de datos:

- Leer tabla (TBLRD)
- Escribir tabla (TBLWT)

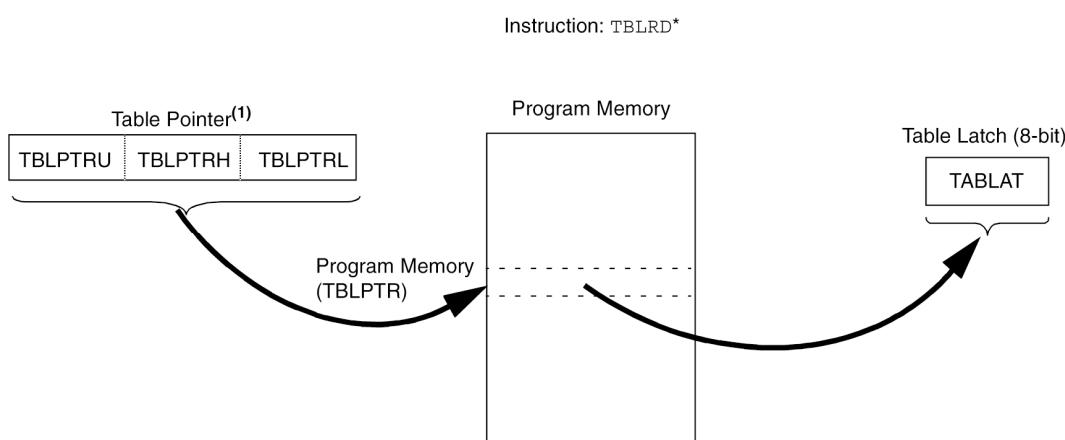
La memoria de programa es de 16 bits de ancho, mientras que el espacio de la RAM de datos es de 8 bits de ancho. La lectura y escritura de datos en tabla mueve entre estas dos memorias a través de un registro de 8 bits (TABLAT).

Las operaciones de lectura de la tabla recuperan datos de la memoria de programa y lo coloca en el espacio de la RAM de datos.

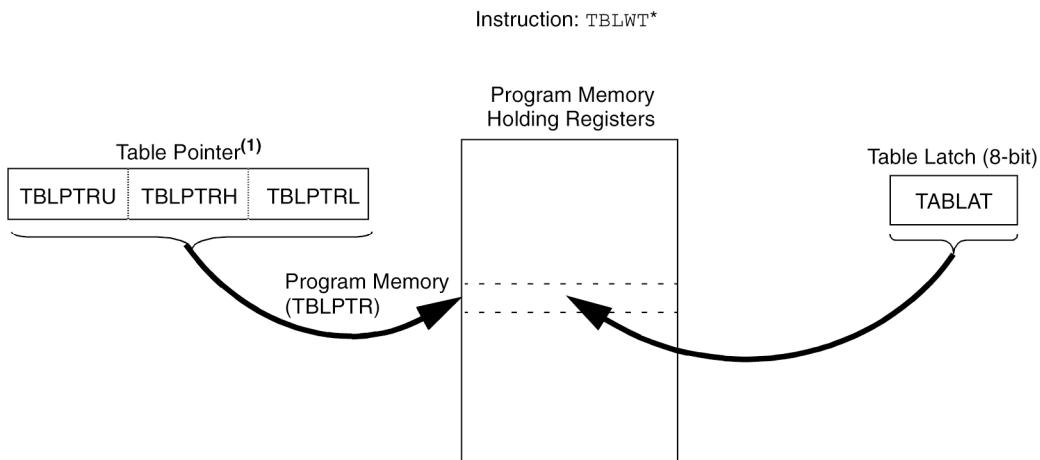
La escritura en tabla almacena los datos en la memoria de datos en los registros bloqueados de la memoria de programa.

Un bloque de la tabla contiene datos, más que instrucciones de programa, no se necesita ser una “palabra-alineada”. Por lo tanto, un bloque de la tabla puede comenzar y terminar en cualquier byte de dirección. Si en una tabla se escribe código ejecutable en la memoria de programa, las instrucciones de programa necesitarán ser palabra-alineadas.

FIGURA 6-1: OPERACIÓN DE LECTURA EN TABLA



Nota: 1: Registros de puntero de tabla a un byte de la memoria de programa.

FIGURA 6-2: OPERACIÓN DE ESCRITURA EN TABLA

Nota: 1: El puntero de tabla dirige uno de los 32 registros bloqueados, la dirección de cada uno determina con TBLPTRL<4:0>

6.2 REGISTROS DE CONTROL

Los registros de control que se utilizan conjuntamente con las instrucciones TBLRD y TBLWT. Son:

- Registro EECON1
- Registro EECON2
- Registro TABLAT
- Registros TBLPTR

6.2.1 REGISTROS EECON1 Y EECON2

El registro EECON1 es el registro de control para los accesos a la memoria. El registro EECON2 no es un registro físico; se utiliza exclusivamente en la memoria para escribir y borrar secuencias. Al leer EECON2 leerá todo ‘0’s.

El bit de control EEPGD determina si el acceso es un acceso de memoria de programa o de datos EEPROM. Al borrarlo, cualquier operación posterior funcionará en la memoria de datos EEPROM. Cuando se activa, las operaciones funcionarán sobre la memoria de programa.

El bit de control CFGS determina si el acceso es a los registros de la configuración o de calibración o a la memoria de programa o a la memoria de los datos EEPROM. Cuando está activo, las operaciones siguientes funcionarán seteando los registros de configuración sin importar EEPGD. Cuando está borrado, el acceso de selección de la memoria se determina por EEPGD.

El bit FREE, cuando se activa, permitirá borrar una instrucción de la memoria de programa. Cuando FREE está activo, la operación de borrado se ejecuta en el siguiente comando WR. Cuando FREE está borrado, sólo se permite escribir.

El bit WREN, cuando está activo, permitirá una operación de escritura. Al iniciar, el bit WREN está borrado. El bit WRERR se activa por hardware cuando se setea y se resetea al finalizar el contador de tiempo de programación interno y se completa la operación de escritura.

Nota: En una operación normal el bit WRERR se lee ‘1’. Esto puede indicar que la operación de lectura ha terminado demasiado pronto por un reset o que no se ha escrito bien.

El bit de control WR inicializa la operación de escritura. El bit no se puede borrar, sólo activar, por software; se borra por hardware al terminar la operación de escritura.

Nota: Se fija el flag de la interrupción EEIF (PIR2<4>) cuando se completa la escritura. Debe borrarse por software.

REGISTRO 6-1: EECON1: Registro 1 de control de los datos de la EEPROM

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
EEPGD	CFGs	-	FREE	WRERR ⁽¹⁾	WREN	WR	RD
R/W-x	r/w-x	U-0	R/W-0	R/W-x	R/W-0	R/S-0	R/S-0
BIT 7	EEPGD:				Bit selector de la memoria flash o de la EEPROM: 1= Memoria de programa flash 0= Memoria de datos EEPROM		
BIT 6	CFGs:				Bit selector de Flash/EEPROM o configuración: 1= Acceso a los registros de configuración 0= Acceso a la memoria flash o EEPROM		
BIT 4	FREE:				Bit de borrado serie de la flash 1= Borra la memoria de programa con direccionamiento serie con TBLPTR en el próximo comando WR. (Se borra al completar la operación de borrado) 0= Sólo escritura		
BIT 3	WRERR:				Flag de error de la memoria flash/EEPROM ⁽¹⁾ 1= Una operación de escritura ha terminado bruscamente 0= La operación de escritura se ha completado		
BIT 2	WREN:				Bit activador de la escritura en la flash/EEPROM 1= Permite ciclos de escritura en la flash/EEPROM 0= No permite ciclos de escritura		
BIT 1	WR:				Bit de control de la escritura: 1= Inicia un ciclo de borrado/escritura en la EEPROM o en la memoria de programa. (El bit se borra por hardware después del ciclo) 0= Ha completado el ciclo de escritura en la EEPROM		
BIT 0	RD:				Bit de control de lectura: 1= Inicia una lectura en la EEPROM. Se borra por hardware, no se puede activar cuando EEPGD=1 o CFGS=1 0= No se inicia una operación de lectura de la EEPROM		

Nota 1: Cuando ocurre un WRERR, los bits EEPGD y CFGS no se borran. Esto permite determinar la condición de error.

6.2.2 REGISTRO DE TABLA LATCH (TABLAT)

El TABLAT es un registro de 8 bits mapeados en el espacio de SFR. El registro de tabla latch se utiliza para almacenar 8 bits de datos durante las transferencias de datos entre la memoria de programa y la RAM de datos.

6.2.3 REGISTRO DEL PUNTERO DE TABLA (TBLPTR)

El registro del puntero de tabla (TBLPTR) direcciona un byte dentro de la memoria de programa. El TBLPTR abarca tres registros de SFR: Byte superior del puntero de tabla, byte alto del puntero de tabla y byte bajo del puntero de tabla (TBLPTRU:TBLPTRH:TBLPTRL). Estos tres registros se ensamblan para formar un puntero de 22 bits de ancho. Los 21 bits inferiores permiten que el dispositivo dirija hasta 2Mbytes de memoria de programa. El 22º bit permite el acceso a la identificación del dispositivo, a la identificación del usuario y a los bits de la configuración.

El puntero de la tabla, TBLPTR, se controla por las instrucciones TBLRD y TBLWT. Estas instrucciones pueden actualizar el TBLPTR en una de las cuatro maneras basadas en la operación de tabla. Estas operaciones en el TBLPTR afectan solamente los 21 bits de orden inferior.

TABLA 6-1: OPERACIONES DEL PUNTERO DE TABLA CON LAS INSTRUCCIONES TBLRD Y TBLWT

Ejemplo	Operaciones en el puntero de tabla
TBLRD* TBLWT*	No modifica TBLPTR
TBLRD* + TBLWT*+	Se incrementa TBLPTR después de la lectura/escritura
TBLRD* - TBLWT*-	Se decrementa TBLPTR después de la lectura escritura
TBLRD +* TBLWT+*	Se incrementa TBLPTR antes de la lectura/escritura

6.2.4 LÍMITES DEL PUNTERO DE TABLA

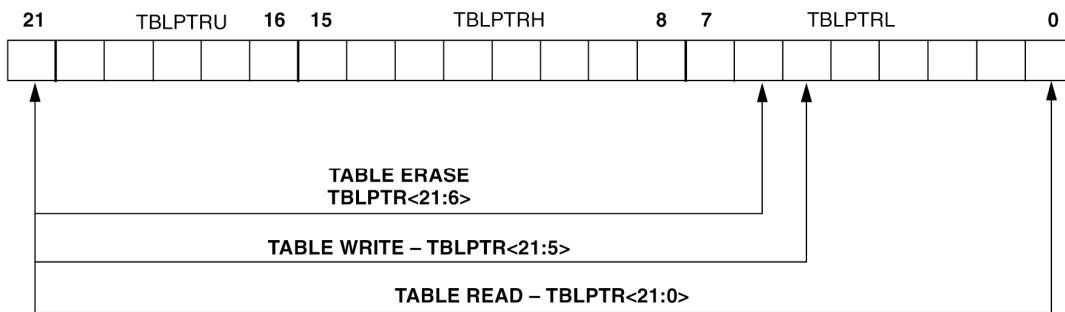
TBLPTR se utiliza en operaciones de lectura, escritura y borrado de la Memoria flash de programa.

Cuando se ejecuta un TBLRD, los 22 bits del TBLPTR determinan qué byte se lee de la memoria de programa a TABLAT.

Cuando se ejecuta un TBLWT, los cinco bits menos significativos del registro de puntero de la tabla (TBLPTR<4:0>) determinan cuáles de los 32 registros bloqueados de la memoria de programa se escriben. Cuando empieza el tiempo de escritura en la memoria de programa (por el bit WR), los 16bits más significativos del TBLPTR (TBLPTR<21: 6>) determinan qué bloque de la memoria de programa de 32 bytes se escribe.

Cuando se ejecuta una instrucción borrar en la memoria de programa, los 16 bits más significativos del registro de puntero de tabla (TBLPTR<21:6>) selecciona el bloque de 64 bytes que será borrado. Los Bits menos significativos (TBLPTR<5:0>) se ignoran.

FIGURA 6-3: LÍMITES DEL PUNTERO DE TABLA BASADO EN LAS OPERACIONES



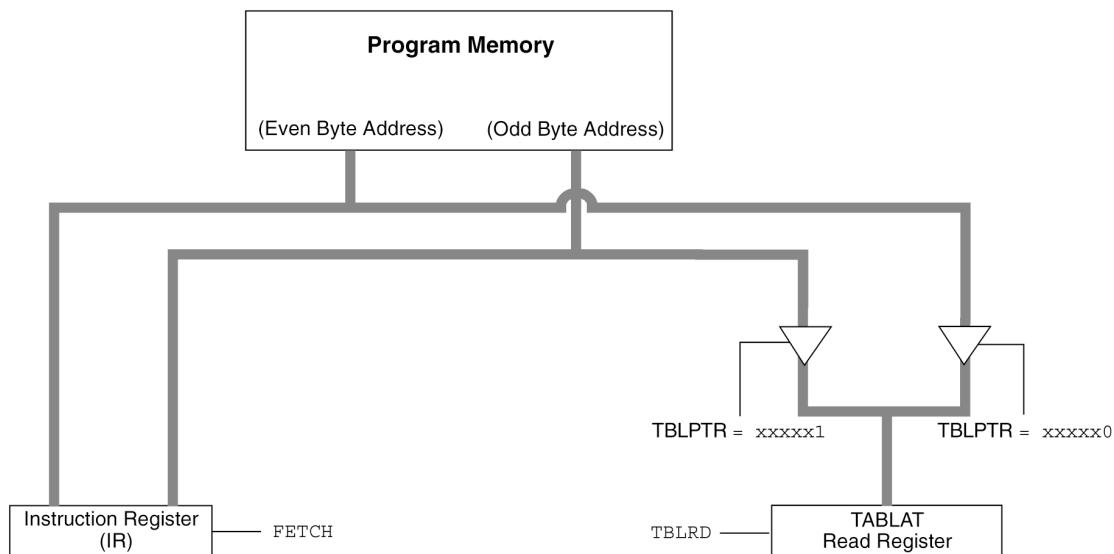
6.3 LECTURA DE LA MEMORIA DE PROGRAMA FLASH

Se utiliza la instrucción TBLRD para recuperar datos de la memoria de programa y los posiciona en la RAM de datos. Sólo se lee un bit cada vez en una operación de lectura de tabla.

TBLPTR señala un byte de dirección en el espacio del programa. Al ejecutar TBLRD se pone el byte señalado en TABLAT. Además, TBLPTR se puede modificar automáticamente en la siguiente operación de lectura de la tabla.

La memoria interna del programa se organiza en palabras. El bit menos significativo de la dirección selecciona entre los bytes altos y bajos de la palabra.

FIGURA 6-4: LECTURAS DE LA MEMORIA DE PROGRAMA



EJEMPLO 6-1: LECTURA DE UNA PALABRA DE LA MEMORIA DE PROGRAMA FLASH

MOVLW	CODE_ADDR_UPPER	
MOVWF	TBLPTRU	;Carga TBLPTR con la base ;dirección de la palabra
MOVLW	CODE_ADDR_HIGH	
MOVWF	TBLPTRH	
MOVLW	CODE_ADDR_LOW	
MOVWF	TBLPTRL	
READ_WORD		
TBLRD*+		
MOVF	TABLAT, W	;lee en TABLAT e incrementa
MOVWF	WORD_EVEN	;consigue los datos
TBLRD*+		
MOVF	TABLAT, W	;lee en TABLAT e incrementa
MOVF	WORD_ODD	;consigue los datos

6.4 BORRADO DE MEMORIA FLASH DE PROGRAMA

El mínimo bloque de borrado es de 32 palabras ó 64 bytes. Solamente con el uso de un programador externo, o con control ICSP, se podrán borrar bloques más grandes. No se pueden borrar palabras en la matriz flash.

Cuando el micro inicia una secuencia de borrado se borra un bloque de 64 bytes de memoria de programa. Se borran los 16 bits más significativos del TBLPTR<21:6> hasta el bloque. Se ignoran los bits TBLPTR<5:0>.

El registro EECON1 ordena la operación de borrado. El bit EEPGD se debe activar para señalar la memoria flash de programa. El bit WREN se debe activar para permitir las operaciones de escritura. El bit FREE se activa para seleccionar una operación de borrado.

Por protección, se debe utilizar la secuencia de inicio de escritura en EECON2.

Se necesita una escritura larga para borrar la flash interna. Se para la ejecución de instrucciones mientras que dure el ciclo de escritura. La escritura la finalizará el contador de tiempo de programación interno.

6.4.1 SECUENCIA DE BORRADO DE LA MEMORIA FLASH DE PROGRAMA

La secuencia de eventos para borrar un bloque de la memoria interna del programa es:

1. Carga el registro del puntero de tabla con la dirección de la fila que se va a borrar.
2. Activa el registro EECON1 para la operación de borrado:
 - Activa el bit EEPGD para dirigirse a la memoria de programa;
 - Borra el bit CFGS para tener acceso a memoria de programa;
 - Activa el bit WREN para permitir la escritura;
 - Activa el bit FREE para permitir el borrado.
1. Inhabilita las interrupciones.
2. Escribe 55h en EECON2.
3. Escribe 0AAh en EECON2.
4. Activa el bit WR. Esto comenzará el ciclo de borrado de la fila.
5. La CPU se parará durante el borrado (cerca de 2ms usando el contador de tiempo interno).
8. Permite las interrupciones.

EJEMPLO 6-2: BORRAR UNA FILA DE LA MEMORIA FLASH DE PROGRAMA

MOVLW	CODE_ADDR_UPPER	;carga TBLPTR con la base
MOVWF	TBLPTRU	;dirección del bloque de memoria
MOVLW	CODE_ADDR_HIGH	
MOVWF	TBLPTRH	
MOVLW	CODE_ADDR_LOW	
MOVWF	TBLPTRL	
ERASE_ROW		
BSF	EECON1, EEPGD	;selección de la memoria FLASH
BCF	EECON1, CFGS	;acceso a la memoria flash
BSF	EECON1, WREN	;permite escribir en la memoria
BSF	EECON1, FREE	;permite la operación de borrado
BCF	INTCON, GIE	;inhabilitar las interrupciones
MOVLW	55h	
MOVWF	EECON2	;escribir 55h
MOVLW	0AAh	
MOVWF	EECON2	;escribir 0AAh
BSF	EECON1, WR	;comienza a borrar
BSF	INTCON, GIE	;volver a permitir las interrupciones

6.5 ESCRIBIR EN LA MEMORIA FLASH DE PROGRAMA

El bloque de programación mínimo es de 16 palabras ó 32 bytes. No se puede programar una palabra o un byte.

Se utiliza internamente la escritura en tabla para cargar los registros necesarios para programar la memoria flash. Se utilizan 32 registros en la escritura en tabla para programar.

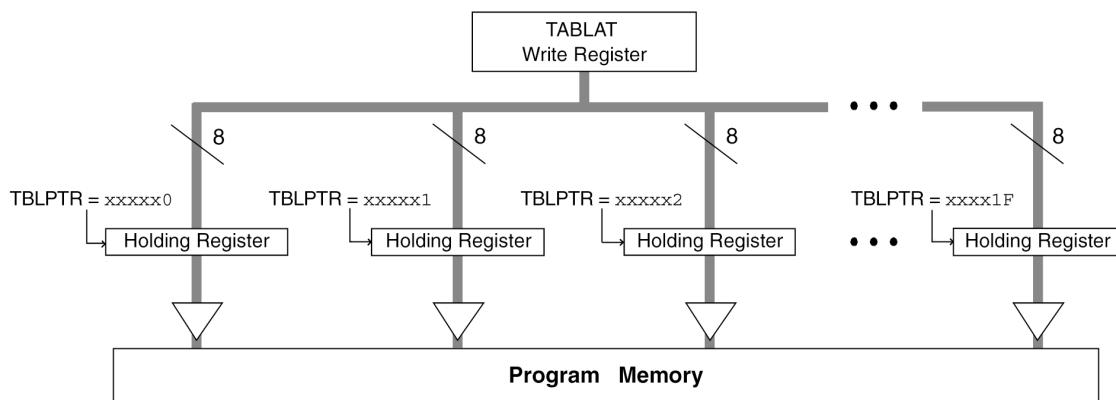
Puesto que el latch de tabla (TABLAT) es solamente un byte, la instrucción TBLWT se tiene que ejecutar 32 veces para cada operación de programación. Toda la tabla escribe operaciones cortas porque sólo se escriben los registros bloqueados. Al final de la actualización de los 32 registros, el registro EECON1 debe estar escrito para comenzar la operación de programación con una escritura larga.

Se necesita la escritura larga para programar la flash interna. Se para la ejecución de instrucciones mientras dure la escritura larga. La escritura larga finalizará con el contador interno de programación.

El contador de tiempo de la EEPROM controla el tiempo de escritura. Las tensiones de escritura/borrado las genera el chip con una bomba de carga, ajustada para funcionar sobre la gama de la tensión del dispositivo.

Nota: El valor prefijado de los registros bloqueados en un reset y antes de una operación de escritura es FFh. Al escribir FFh en un registro de este tipo, no modifica ese byte. Esto significa que los bytes individuales de memoria de programa se pueden modificar, siempre que el cambio no sea modificar un bit de '0' a '1'. Al modificar bytes individuales, no es necesario cargar los 32 registros antes de ejecutar una operación de escritura.

FIGURA 6-5: LA ESCRITURA DE TABLA EN LA MEMORIA FLASH DE PROGRAMA



6.5.1 SECUENCIA DE ESCRITURA EN LA MEMORIA FLASH DE PROGRAMA

La secuencia para programar una posición interna de memoria de programa es:

1. Leer 64 bytes en RAM.
 2. Actualizar los valores de los datos en RAM cuando sea necesario.
 3. Cargar el registro del puntero de tabla con la dirección que se borre.
 4. Ejecutar el procedimiento de borrar fila.
 5. Cargar el registro del puntero de tabla con la dirección del primer byte que se escriba.
 6. Escribir 32 bytes en los registros bloqueados con EL auto incremento.
 7. Activar el registro EECON1 para la operación de escritura:
 - Activar el bit EEPGD para que apunte a la memoria a programar;
 - Borrar el bit CFGS para tener acceso a memoria de programa;
 - Activar el bit WREN para permitir escribir.
 8. Inhabilitar las interrupciones.
 9. Escribir 55h en EECON2.
 10. Escribir 0AAh en EECON2.
 11. Activar el bit WR. Con esto comenzará el ciclo de escritura.
 12. La CPU se parará durante la escritura (cerca de 2 ms usando el contador de tiempo interno).
 13. Volver a permitir las interrupciones.
 14. Repetir los pasos 6 a 14 una vez más para escribir 64 bytes.
 15. Verificar la memoria (leer tabla).
- Este procedimiento requerirá cerca de 8ms para actualizar una fila de 64 bytes de memoria.

Nota: Antes de activar el bit WR, el puntero de tabla tiene que estar dentro del radio de acción de las direcciones previstas de 32 bytes dentro el registro bloqueado.

EJEMPLO 6-3: EL ESCRIBIR EN LA MEMORIA FLASH DEL PROGRAMA

	MOVLW D'64'	;nº de bytes en el bloque a
borrar		
MOVWF		COUNTER
MOVLW BUFFER_ADDR_HIGH	:puntero	al buffer
MOVWF		FSR0H
MOVLW		BUFFER_ADDR_LOW
MOVWF		FSR0L
MOVLW CODE_ADDR_UPPER	;Cargar TBLPTR con la dirección base	
MOVWF TBLPTRU	:del bloque de memoria	
MOVLW		CODE_ADDR_HIGH
MOVWF		TBLPTRH
MOVLW		CODE_ADDR_LOW
MOVWF		TBLPTRL
READ_BLOCK		
TBLRD*+		
MOVF TABLAT, W	;leer en TABLAT, e incrementar	
MOVWF POSTINC0	;coger datos	
DECFSZ COUNTER	;guardar datos	
BRA READ_BLOCK	;correcto?	
MODIFY_WORD		
MOVLW DATA_ADDR_HIGH	;Puntero al buffer	
MOVWF		FSR0H
MOVLW		DATA_ADDR_LOW
MOVWF		FSR0L
MOVLW NEW_DATA_LOW	;Actualizar buffer	
MOVLW		NEW_DATA_HIGH
MOVWF		INDFO
ERASE_BLOCK		
MOVLW CODE_ADDR_UPPER	;Cargar TBLPTR con la dirección base	
MOVWF TBLPTRU	:del bloque de memoria	
MOVLW		CODE_ADDR_HIGH
MOVWF		TBLPTRH
MOVLW		CODE_ADDR_LOW
MOVWF		TBLPTRL
BSF EECON1, EEPGD	;Puntero a la Flash	
BCF EECON1, CFGS	;acceso a la Flash	
BSF EECON1, WREN	;activar la escritura en memoria	
BSF EECON1, FREE	;Activar la operación de borrado de fila	
BCF INTCON, GIE	;desactiva interrupciones	
MOVLW		55h
MOVWF EECON2	;escribe	55h
MOVLW		0AAh
MOVWF EECON2	;escribe	0AAh
BSF EECON1, WR	;comienzo del borrado (CPU parada)	
BSF INTCON, GIE	;reactivae interrupciones	
TBLRD*-	;decremto después de leer	
MOVLW BUFFER_ADDR_HIGH	;puntero al buffer	
MOVWF		FSR0H
MOVLW		BUFFER_ADDR_LOW
MOVWF		FSR0L
MOVLW		D'2'
MOVWF		COUNTER1
WRITE_BUFFER_BACK		
MOVLW D'32'		
MOVWF COUNTER		
WRITE_BYTE_TO_HREGS		
MOVF POSTINC0, W	;coger el byte bajo del buffer	

MOVWF TABLAT	;llevar dato al lacht de tabla
TBLWT+*	;escribir dato (escritura corta)
DECFSZ COUNTER	;repetir hasta que el buffer se llene
BRA WRITE_WORD_TO_HREGS	
PROGRAM_MEMORY	
BSF EECON1, EEPGD	;Puntero a la memoria flash
BCF EECON1, CFGS	;Acceso a la memoria flash
BSF EECON1, WREN	;Activar escritura en memoria
BCF INTCON, GIE	;Desactivar interrupciones
MOVLW 55h	
MOVWF EECON2	;Escribir 55h
MOVLW 0AAh	
MOVWF EECON2	;Escribir 0AAh
BSF EECON1, WR	;Comienzo del programa
DECFSZ COUNTER1	
BRA WRITE_BUFFER_BACK	
BSF INTCON, GIE	;Activa las interrupciones
BCF EECON1, WREN	;Desactiva la escritura en memoria

6.5.2 VERIFICACIÓN DE LA ESCRITURA

Dependiendo del uso, en una buena programación se puede necesitar que el valor escrito en la memoria se verifique con el valor original. Esto se debe utilizar en usos donde excesivas escrituras pueden provocar que los bits estén cerca del límite de la especificación.

6.5.3 FINALIZACIÓN INESPERADA DE LA OPERACIÓN ESCRIBIR

Si una escritura termina por un acontecimiento imprevisto, tal y como la pérdida de energía o un reset inesperado, las localizaciones de la memoria programadas se deben verificar y reprogramar si se necesita. Si se interrumpe la operación de escritura por un reset MCLR* o un desbordamiento del WDT durante la operación normal, el usuario puede comprobar el bit WRERR y reescribir las localizaciones según se necesite.

TABLA 6-2: REGISTROS ASOCIADOS CON LA MEMORIA FLASH

Nombre	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TBLPTRU	-	-	bit 21 ⁽¹⁾	Byte superior del puntero de tabla de la memoria de				

				programa (TBLPTR<20:16>)				
TBLPTRH	Byte alto del puntero de tabla de la memoria de programa (TBLPTR<15:8>)							
TBLPTRL	Byte bajo del puntero de tabla de la memoria de programa (TBLPTR<7:0>)							
TABLAT	Match de tabla de la memoria de programa							
INTCON	GIE/GIEH	PEIE/GIEL	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF
EECON2	Registro de control 2 EEPROM (no es un registro físico)							
EECON1	EEPGD	CFGSS	-	FREE	WRERR	WREN	WR	RD
IPR2	OSCFIP	CMIP	USBIP	EEIP	BCLIP	HLVDIP	TMR3IP	CCP2IP
PIR2	OSCFIF	CMIF	USBIF	EEIF	BCLIF	HLVDIF	TMR3IF	CCP2IF
PIE2	OSCFIE	CMIE	USBIE	EEIE	BCLIE	HLVDIE	TMR3IE	CCP2IE

Leyenda: - = no implementado, se lee '0'. Las celdas sombreadas no se utilizan en el acceso a la Flash/EEPROM

Nota: 1: El bit 21 de TBLPTRU permite el acceso a los bits de configuración del dispositivo.

7.0 MEMORIA DE DATOS EEPROM

La EEPROM es una matriz de memoria permanente, separada de la RAM de datos y de la memoria de programa, se utiliza en almacenamientos de larga duración de los datos del programa. No está mapeada directamente en los registros de archivo o en la memoria del programa, sino que se trata indirectamente a través de los registros especiales de la función (SFRs). La EEPROM es legible y escribible durante la operación normal sobre la gama entera de V_{DD}.

Cuatro SFRs se utilizan para leer y para escribir los datos en la EEPROM así como la memoria del programa. Son:

- EECON1
- EECON2
- EEDATA
- EEADR

Los datos EEPROM permiten la lectura y escritura de bytes. Cuando se conecta al bloque de la memoria de datos, el EEDATA sostiene los 8bits de datos para la lectura/escritura y el registro EEADR lleva a cabo el direccionamiento de la localización de la EEPROM.

La memoria de datos EEPROM se clasifica como muy resistente a los ciclos de escritura/borrado. Un byte escribe y automáticamente borra la localización y escribe los datos nuevos (borrar-antes-escribir). El tiempo de escritura se controla por un contador de tiempo en el chip; variará con la tensión y la temperatura así como de chip a chip.

7.1 REGISTROS EECON1 Y EECON2

El acceso a los datos EEPROM se controla con dos registros: EECON1 y EECON2. Éstos son iguales que los del acceso de control a la memoria del programa y se utilizan de una manera similar para los datos EEPROM.

El registro EECON1 es el registro de control para el acceso a memoria de datos y de programa. Controlando el bit, EEPGD, se determina si el acceso es a la memoria de programa o a la memoria de datos EEPROM. Cuando está borrado, las operaciones se hacen a la memoria de datos EEPROM. Cuando está activado, se dirige a la memoria de programa. Controlando el bit, CFGS, se determina si el acceso es a los registros de configuración o a la memoria del programa/EEPROM. Cuando está activado, las operaciones siguientes tienen acceso a los registros de configuración. Cuando CFGS está borrado, el bit EEPGD selecciona la flash o la memoria EEPROM. El bit WREN, cuando está activado, permitirá una operación de escritura. En ciclo inicial, el bit WREN está borrado. El bit WRERR se activa por hardware cuando se activa y se borra el bit WREN cuando finaliza el contador de tiempo de programación interno y se completa la operación de escritura.

Nota: Durante la operación normal, el WRERR se lee como ‘1’. Esto puede indicar que ha finalizado una operación de escritura prematuramente cerca de un reset o una operación de escritura finaliza incorrectamente.

El bit de control WR inicia operaciones de escritura. El bit no se puede borrar, sólo está activo, en software; se borra en hardware al terminar la operación de escritura.

Nota: Se activa el flag de la interrupción EEIF (PIR2<4>) cuando la escritura se completa. Debe desactivarse por software.

Controlando los bits, RD y WR, comienzan las operaciones leer y borrar/escribir, respectivamente. Estos bits se activa por firmware y se desactiva por hardware al terminar la operación.

El bit RD no se puede activar al tener acceso a la memoria de programa (EEPGD=1).

El registro EECON2 no es un registro físico. Se utiliza exclusivamente en las secuencias de escribir y borrar en la memoria. La lectura EECON2 leerá todo ‘0’s.

7.2 LECTURA DE LOS DATOS DE LA MEMORIA EEPROM

Para leer una posición de memoria de datos, el usuario debe escribir la dirección en el registro EEADR, borrar el bit EEPGD (EECON1<7>) y después activar el bit RD (EECON1<0>). Los datos están disponibles en el siguiente ciclo de instrucción; por lo tanto, el registro EEDATA se puede leer con la instrucción siguiente. EEDATA mantendrá este valor hasta otra operación de lectura o hasta que lo escriba el usuario (durante una operación de escritura).

EJEMPLO 7-1: LECTURA DE LA MEMORIA EEPROM

```

MOVLW    DATA_EE_ADDR
MOVWF    EEDRA      ;Bits bajos de la dirección de memoria a leer
BCF     EECON1,EEPGD ;Puntero a la memoria de datos
BCF     EECON1,CFG5  ;Acceso a la memoria EEPROM
BSF     EECON1,RD    ;Lectura en EEPROM
MOVWF    EEDATA,W   ;W=EEDATA

```

7.3 ESCRITURA EN LA MEMORIA DE DATOS EEPROM

Para escribir una localización de la EEPROM, se tiene que escribir la dirección en el registro EEADR y los datos al registro EEDATA. La escritura no comenzará si no se ejecuta la secuencia: escribir 55h en EECON2, escribir 0AAh en EECON2, después activar el bit WR; para cada byte. Se recomienda que las interrupciones estén desactivadas durante esto segmento de código.

EJEMPLO 7-2: ESCRITURA DE DATOS EN LA EEPROM

```

MOVLW    DATA_EE_ADDR
MOVWF    EEDATA      ;Bits bajos de la dirección de memoria a escribir
MOVLW    DATA_EE_DATA
MOVWF    EEDATA      ;Valor de la memoria de datos a escribir
BCF     EECON1,EPGD  ;Puntero a la memoria de datos
BCF     EECON1,CFG5  ;Acceso a la EEPROM
BSF     EECON1,WREN  ;Activa escritura
BCF     INTCON,GIE   ;Desactivación de las interrupciones
MOVLW    55h
MOVWF    EECON2
MOVLW    0AAh
MOVWF    EECON2
BSF     EECON1,WR    ;Activa WR para empezar a escribir
BSF     INTCON,GIE   ;Activa las interrupciones
BCF     EECON1,WREN  ;Desactiva la escritura cuando se complete

```

7.4 VERIFICACIÓN DE LA ESCRITURA

Dependiendo de la aplicación, se puede necesitar que se verifiquen los datos después de escribirlos en la memoria comparándolos con el valor original. Se tiene que

utilizar esto en los casos en los que haya muchas operaciones de escritura y los bits trabajen cercanos a los límites de las especificaciones.

7.5 OPERACIÓN CON CÓDIGO DE PROTECCIÓN

La memoria de datos EEPROM tiene los bits de los códigos de protección en palabras de configuración. Las operaciones de lectura y escritura externas se desactivan si se permite el código de protección.

El microcontrolador puede leer y escribir en la EEPROM sin importar el estado del bit de configuración del código de protección.

7.6 PROTECCIÓN CONTRA ESCRITURAS FALSAS

Hay casos en los que el dispositivo no puede escribir en la memoria EEPROM. Para proteger la EEPROM contra estas falsas escrituras, se ponen varios mecanismos en ejecución. En un ciclo de inicio, el bit WREN está borrado. Además, se bloquea la escritura en la EEPROM durante el período de ciclo inicial del contador de tiempo. La secuencia de inicio de la escritura y el bit WREN ayudan a prevenir una escritura accidental durante un cese de tensión, fallo en la energía o malfuncionamiento del software.

7.7 USAR LOS DATOS EEPROM

Los datos EEPROM tienen una gran resistencia, una matriz direccionable de bytes que se ha optimizado para el almacenamiento de información que cambia con frecuencia (ej., variables de programa u otros datos que se actualizan a menudo). Los valores que cambian con frecuencia se actualizarán muy a menudo. Si éste no es el caso, se debe hacer un refresco de la matriz. Por esta razón, las variables que cambian infrecuentemente (por ejemplo constantes, las identificaciones, la calibración, etc.) se deben almacenar en la memoria flash del programa.

Nota: Si los datos EEPROM se utilizan solamente para almacenar constantes y/o datos que cambian raramente, no se necesita un refresco de la matriz.

EJEMPLO 7-3: RUTINA DE REFRESCO DE LA EEPROM

CLRF	EEADR	;Comienzo en la dirección 0
BCF	EECON1,CFG5	;Activa memoria
BCF	EECON1,EEPGD	;Activa memoria EEPROM

	BCF	INTCON,GIE	;Desactiva interrupciones
	BSF	EECON1,WREN	;Activa escritura
Loop	BSF	EECON1,RD	;Lee la dirección actual
	MOVLW	55h	
	MOVWF	EECON2	
	MOVLW	0AAh	
	MOVWF	EECON2	
	BSF	EECON1,WR	;Activa WR para empezar a escribir
	BTFSC	EECON1,WR	;Espera que se complete la escritura
	BRA	\$-2	
	INCFSZ	EEEDR,F	;Incrementa la dirección
	BRA	LOOP	;Si no es cero lo repite
	BCF	EECON1,WREN	;Desactiva la escritura
	BSF	INTCON,GIE	;Activa interrupciones

TABLA 7-1: REGISTROS ASOCIADOS CON LA MEMORIA EEPROM

Nombre	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTCON	GIE/GIEH	PEIE/GIEL	TMROIE	INT0IE	RBIE	TMROIF	INT0IF	RBIF
EEADR	Registro de direccionamiento de la EEPROM							
EEDATA	Registro de datos de la EEPROM							
EECON2	2º Registro de control de la EEPROM (no es un registro físico)							
EECON1	EEPGD	CFGs	-	FREE	WRERR	WREN	WR	RD
IPR2	OSCFIP	CMIP	USBIP	EEIP	BCLIP	HLVDIP	TMR3IP	CCP2IP
PIR2	OSCFIF	CMIF	USBIF	EEIF	BCLIF	HLVDIF	TMR3IF	CCP2IF
PIE2	OSCFIE	CMIE	USBIE	EEIE	BCLIE	HLVDIE	TMR3IE	CCP2IE

Leyenda:- = no implementado, se lee '0'. Las casillas sombreadas no se utilizan en el acceso a la Flash/EEPROM.

8.0 MULTIPLICADOR DE 8x8 POR HARDWARE

8.1 INTRODUCCIÓN

Todos los dispositivos PIC18 incluyen un multiplicador de 8x8 por hardware como parte de la ALU. El multiplicador realiza una operación sin signo y devuelve un resultado que se almacene en el par de registros del producto, PRODH:PRODL de 16bits. La operación del multiplicador no afecta ningún flag en el registro ESTADO.

La fabricación de la operación multiplicación por hardware permite que se termine en un solo ciclo de instrucción. Esto tiene ventajas, un rendimiento de cómputo más alto y un tamaño de código reducido para los algoritmos de multiplicación y permite que los dispositivos PIC18 sean utilizados en muchos usos reservados previamente para los procesadores de la señal numérica.

EJEMPLO 8-1: RUTINA DE MULTIPLICACIÓN SIN SIGNO DE 8x8

```
MOVF      ARG1,W  
MULWF    ARG2          ;ARG1*ARG2=PRODH:PRODL
```

EJEMPLO 8-2: RUTINA DE MULTIPLICACIÓN CON SIGNO DE 8x8

```
MOVF      ARG1,W  
MULWF    ARG2          ;ARG1*ARG2=PRODH:PRODL  
BTFSCL   ARG2,SB        ;Bit test del signo  
SUBWF    PRODH,F       ;PRODH=PRODH-ARG1  
MOVF      ARG2,W  
BTFSCL   ARG1,SB        ;Bit test del signo  
SUBWF    PRODH,F       ;PRODH=PRODH-ARG2
```

ECUACIÓN 8-1: ALGORITMO DE MULTIPLICACIÓN 16x16 SIN SIGNO

$$\begin{aligned} \text{RES3:RES0} &= \text{ARG1H:ARG1L*ARG2H:ARG2L} = \\ &= (\text{ARG1H} * \text{ARG2H} * 2^{16}) + (\text{ARG1H} * \text{ARG2L} * 2^8) + \\ &+ (\text{ARG1L} * \text{ARG2H} * 2^8) + (\text{ARG1L} * \text{ARG2L}) \end{aligned}$$

EJEMPLO 8-3: RUTINA DE MULTIPLICACIÓN DE 16x16 SIN SIGNO

MOVF	ARG1L,W
MULWF	ARG2L
MOVFF	PRODH,RES1
MOVFF	PRODL,RES0
MOVF	ARG1H,W
MULWF	ARG2H
MOVFF	PRODH,RES3
MOVFF	PRODH,RES2
MOVF	ARG1L,W
MULWF	ARG2H
MOVF	PRODL,W
ADDWF	RES1,F
MOVF	PRODH,W
ADDWFC	RES2,F
CLRF	WREG
ADDWFC	RES3,F
MOVF	ARG1H,W
ADDWF	RES1,F
MOVF	PRODH,W
ADDWFC	RES2,F
CLRF	WREG
ADDWFC	RES3,F

ECUACIÓN 8-2: ALGORITMO DE MULTIPLICACIÓN 16x16 CON SIGNO

$$\begin{aligned}
 \text{RES3:RES0} &= \text{ARG1H:ARG1L} * \text{ARG2H:ARG2L} = \\
 &= (\text{ARG1H} * \text{ARG2H} * 2^{16}) + (\text{ARG1H} * \text{ARG2L} * 2^8) + \\
 &+ (\text{ARG1L} * \text{ARG2H} * 2^8) + (\text{ARG1L} * \text{ARG2L}) + \\
 &+ (-1 * \text{ARG2H} <7> * \text{ARG1H:ARG1L} * 2^{16}) + \\
 &+ (-1 * \text{ARG1H} <7> * \text{ARG2H:ARG2L} * 2^{16})
 \end{aligned}$$

EJEMPLO 8-4: RUTINA DE MULTIPLICACIÓN 16x16 CON SIGNO

```

MOVF      ARG1L,W
MULWF    ARG2L
MOVFF    PRODH,RES1
MOVFF    PRODL,RES0
MOVF      ARG1H,W
MULWF    ARG2H
MOVFF    PRODH,RES3
MOVFF    PRODH,RES2
MOVF      ARG1L,W
MULWF    ARG2H
MOVF      PRODL,W
ADDWF    RES1,F
MOVF      PRODH,W
ADDWFC    RES2,F
CLRF      WREG
ADDWFC    RES3,F
MOVF      ARG1H,W
ADDWF    RES1,F
MOVF      PRODH,W
ADDWFC    RES2,F
CLRF      WREG
ADDWFC    RES3,F
BTFS S   ARG2H,7
BRA       SIGN_ARG1
MOVF      ARG1L,W
SUBWF    RES2
MOVF      ARG1H,W
SUBWFB    RES3
SIGN_ARG1
BTFS S   ARG1H,7
BRA       CONT_CODE
MOVF      ARG2L,W
SUBWF    RES2
MOVF      ARG2H,W
SUBWFB    RES3
CONT_CODE

```

9.0 INTERRUPCIONES

Los dispositivos PIC18F2455/2550/4455/4550 tienen fuentes múltiples de interrupción y una prioridad de interrupción que permite que cada fuente de interrupción se le asigne un nivel prioritario o un nivel bajo de prioridad. El puntero de alta prioridad está en 000008h y el puntero de interrupción de baja prioridad está en 000018h. Los acontecimientos prioritarios de interrupción pararán cualquier interrupción de prioridad baja que pueda estar en marcha.

Hay diez registros que se utilizan para controlar las operaciones de interrupción. Estos registros son:

- RCON
- INTCON
- INTCON2
- INTCON3
- PIR1, PIR2
- PIE1, PIE2
- IPR1, IPR2

Cada fuente de interrupción tiene tres bits para controlar su operación. Las funciones de estos bits son:

- Flag para indicar que ha ocurrido una interrupción
- El bit que permite la ejecución de programa en la dirección del puntero de interrupción cuando se activa el flag
- El bit de prioridad para seleccionar alta o baja prioridad

Las características de prioridad de las interrupciones se determinan activando el bit IPEN (RCON<7>). Cuando se permite la prioridad de las interrupciones, hay dos bits que permiten las interrupciones globalmente. Activando el bit GIEH (INTCON<7>) se permiten todas las interrupciones que tengan el bit de prioridad activo (prioritario). Seteando el bit GIEL (INTCON<6>) permite todas las interrupciones que tengan el bit de prioridad borrado (prioridad baja). Cuando el flag de interrupción, el bit de permiso y el bit de las interrupciones globales están activos, el programa saltará inmediatamente a la dirección 000008h ó 000018h, dependiendo del bit de prioridad. Las interrupciones individuales pueden desactivarse con su correspondiente bit de permiso.

Cuando el bit IPEN está borrado (estado por defecto), las características de prioridad de las interrupciones están desactivadas y las interrupciones son compatibles con los dispositivos de alcance medio de PICmicro®. En el modo de compatibilidad, los bits de prioridad de las interrupciones para cada fuente no tienen ningún efecto. INTCON<6> es el bit PEIE el cual permite/inhabilita todas las fuentes periféricas de interrupción. INTCON<7> es el bit GIE que permite/inhabilita todas las fuentes de interrupción. Todas las interrupciones saltan a la dirección 000008h en modo de compatibilidad.

Cuando se responde una interrupción, el bit de permiso de las interrupciones globales está borrado para inhabilitar las otras. Si el bit IPEN está reseteado, éste es el bit GIE. Si se utilizan los niveles de prioridad de interrupción, éste será el bit GIEH o GIEL. Las fuentes de interrupción prioritarias pueden interrumpir una interrupción de baja prioridad. Las interrupciones de prioridad bajas no se procesan mientras que las interrupciones prioritarias estén en marcha.

La dirección de retorno se pone en la pila y el PC se carga con la dirección del puntero de interrupción (000008h ó 000018h). Una vez en la rutina de la interrupción, las fuentes de interrupción se pueden determinar interrogando los flags de interrupción.

Los flags de interrupción se tienen que despejar por software antes de volver a permitir las interrupciones para evitar interrupciones recurrentes.

La “instrucción para volver de la interrupción”, RETFIE, devuelve de la rutina de interrupción y activa el bit GIE (GIEH o GIEL si se utilizan los niveles de prioridad) que vuelve a permitir las interrupciones.

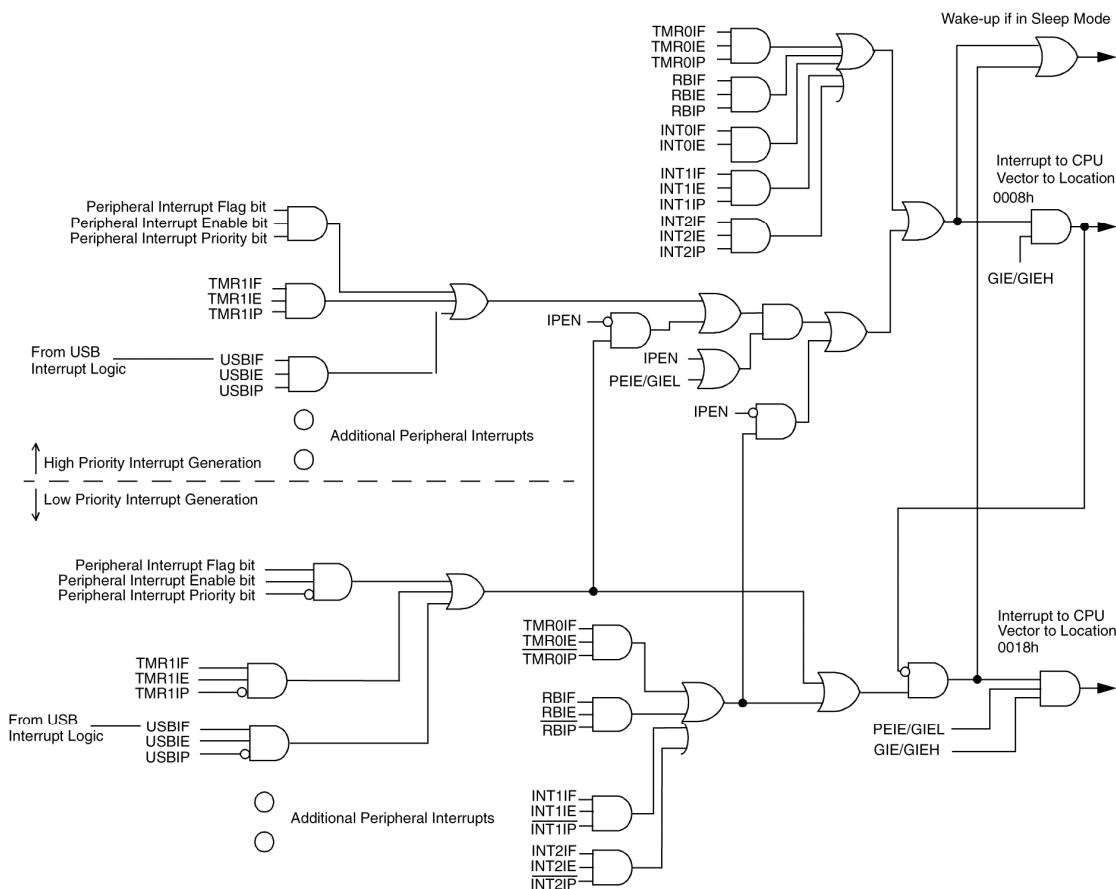
Para los acontecimientos externos de interrupción, tales como los pines INT o la interrupción por cambio de la entrada PORTB, el estado latente de la interrupción es de tres a cuatro ciclos de instrucción. El exacto el estado latente es igual para instrucciones de uno o dos ciclos. Los flags individuales de interrupción se activan sin importar el estado de su bit de permiso correspondiente o el bit GIE.

Nota: No utilizar la instrucción MOVFF para modificar registros de control de interrupción mientras que se permita cualquier interrupción. Al hacerlo puede provocar un comportamiento errático del microcontrolador.

9.1 INTERRUPCIONES DEL USB

Como otros periféricos, el módulo USB es capaz de generar una amplia gama de interrupciones para muchos tipos de acontecimientos. Éstos incluyen varios tipos de acontecimientos normales de comunicación y de estado y varios niveles de errores de acontecimientos.

Para manejar estos acontecimientos, el módulo USB se equipa con su propia lógica de interrupción. La lógica funciona de una forma similar a las interrupciones del microcontrolador, con cada fuente de interrupción tenemos un flag separado y los bits de permiso. Todos los acontecimientos se concentran en una sola interrupción del dispositivo, USBIF (PIR2<5>). A los acontecimientos individuales de interrupción del USB no se les puede asignar su propia prioridad. Esto se determinada en el nivel de prioridad de las interrupciones del dispositivo para todos los acontecimientos del USB con el bit USBIP.

FIGURA 9-1: LÓGICA DE INTERRUPCIÓN

9.2 REGISTROS INTCON

Los registros INTCON son legibles y escribibles y contienen varios bits de permiso, prioridad y flags.

Nota: Se fijan los flags cuando ocurre una condición de interrupción sin importar el estado del bit de permiso correspondiente o del bit de permiso global. El usuario debe asegurarse de que el flag de la interrupción esté a cero antes de habilitar dicha interrupción.

REGISTRO 9-1: INTCON: Registro de control de interrupción

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
GIE/GIEH	PEIE/GIEL	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF ⁽¹⁾
R/W-0	R/W-0	R/W-0	R/W-0	R/W-x	R/W-0	R/W-0	R/W-x0
BIT 7	GIE/GIEH:	Bit activador del permiso global de interrupción: Cuando IPEN=0 1= Activa interrupciones enmascarables 0= Desactiva todas las interrupciones Cuando IPEN=1 1= Activa interrupciones de prioridad alta 0= Desactiva interrupciones de prioridad alta					
BIT 6	PEIE/GIEL:	Bit activador de las interrupciones de periféricos: Cuando IPEN=0 1= Activa interrupciones enmascarables 0= Desactiva todas las interrupciones Cuando IPEN=1 1= Activa interrupciones de prioridad baja 0= Desactiva interrupciones de prioridad baja					
BIT 5	TMR0IE:	Bit de permiso de interrupción por desbordamiento del TMR0: 1= Activa la interrupción 0= Desactiva la interrupción					
BIT 4	INT0IE:	Bit de permiso de la interrupción externa INT0: 1= Activa la interrupción 0= Desactiva la interrupción					
BIT 3	RBIE:	Bit de permiso de la interrupción por cambio del PORTB: 1= Activa la interrupción 0= Desactiva la interrupción					
BIT 2	TMR0IF:	Flag de la interrupción por desbordamiento del TMR0: 1= Se ha activado la interrupción 0= No se ha activado la interrupción					
BIT 1	INT0IF:	Flag de la interrupción externa INT0: 1= Se ha activado la interrupción 0= No se ha activado la interrupción					
BIT 0	RBIF:	Flag de la interrupción del cambio de PORTB ⁽¹⁾ : 1= Uno de los pines RB7:RB4 ha cambiado (se tiene que borrar por software) 0= No ha cambiado ningún pin					

Nota 1: Una condición de unión mal hecha continuará activando este bit. Al leer PORTB terminará la condición de unión mal hecha y permitirá que el bit se borre.

REGISTRO 9-2: INTCON2: Registro de control de interrupción 2

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
RBPU*	INTEDG0	INTEDG1	INTEDG2	-	TMR0IP	-	RBIP
R/W-0	R/W-0	R/W-0	R/W-0	R/W-x	R/W-0	R/W-0	R/Wx0
BIT 7	RBPU*:	Bit activador de pull-up del PORTB: 1= Desactivado 0= Activado por valores individuales de latch					
BIT 6	INTEDG0:	Bit selector del flanco de la interrupción 0: 1= Flanco de subida 0= Flanco de bajada					
BIT 5	INTEDG1:	Bit selector del flanco de la INT1: 1= Flanco de subida 0= Flanco de bajada					
BIT 4	INTEDG2:	Bit selector del flanco de la INT2: 1= Flanco de subida 0= Flanco de bajada					
BIT 2	TMR0IP:	Bit de prioridad del desbordamiento del TMR0: 1= Alta prioridad 0= Baja prioridad					
BIT 0	RBIP:	Bit de prioridad del cambio de PORTB: 1= Alta prioridad 0= Baja prioridad					

REGISTRO 9-3: INTCON3: Registro de control de interrupción 3

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
INT2IP	INT1IP	-	INT2IE	INT1IE	-	INT2IF	INT1IF
R/W-0	R/W-0	R/W-0	R/W-0	R/W-x	R/W-0	R/W-0	R/Wx0
BIT 7	INT2IP:	Bit de prioridad de la INT2: 1= Alta prioridad 0= Baja prioridad					
BIT 6	INT1IP:	Bit de prioridad de la INT1: 1= Alta prioridad 0= Baja prioridad					
BIT 4	INT2IE:	Bit de permiso de la interrupción externa INT2: 1= Activa la interrupción 0= Desactiva la interrupción					
BIT 3	INT1IE:	Bit de permiso de la interrupción externa INT1: 1= Activa la interrupción 0= Desactiva la interrupción					
BIT 1	INT2IF:	Flag de la interrupción externa INT2: 1= Se ha activado la interrupción 0= No se ha activado la interrupción					
BIT 0	INT1IF:	Flag de la interrupción externa INT1: 1= Se ha activado la interrupción 0= No se ha activado la interrupción					

9.3 REGISTROS PIR

Los registros PIR contienen los flags individuales de las interrupciones periféricas. Debido al número de fuentes de interrupción periféricas hay dos registros de respuesta de las interrupciones periféricas (PIR1 y PIR2).

REGISTRO 9-4: PIR1: Registro de respuesta de las interrupciones periféricas 1

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
SPPIF ⁽¹⁾	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF
R/W-0	R/W-0	R/W-0	R/W-0	R/W-x	R/W-0	R/W-0	R/Wx0
BIT 7	SPPIF:			Flag de acceso al puerto paralelo lectura/escritura ⁽¹⁾ :			
				1= Ocurre una lectura/escritura (borrado por software)			
				0= No ocurre lectura/escritura			
BIT 6	ADIF:			Flag del conversor A/D:			
				1= Conversión A/D completa (borrado por software)			
				0= Conversión A/D no completa			
BIT 5	RCIF:			Flag de recepción por la EUSART:			
				1= Buffer de recepción lleno (borrado por software)			
				0= Buffer de recepción vacío			
BIT 4	TXIF:			Flag de transmisión por la EUSART:			
				1= Buffer de transmisión vacío (borrado por software)			
				0= Buffer de transmisión lleno			
BIT 3	SSPIF:			Flag del puerto serie síncrono:			
				1= Transmisión/recepción completa (borrado por software)			
				0= Esperando transmisión/recepción			
BIT 2	CCP1IF:			Flag del CCP1:			
				Modo captura:			
				1= Ha ocurrido una captura en el TMR1 (borrado por software)			
				0= No ha ocurrido una captura			
				Modo comparación:			
				1= Ha ocurrido una comparación en el TMR1 (borrado por software)			
				0= No ha ocurrido una comparación			
				Modo PWM:			
				No se utilizan			
BIT 1	TMR2IF:			Flag de interrupción al igualar TMR2 y PR2:			
				1= Se han igualado (borrado por software)			
				0= No se han igualado			
BIT 0	TMR1IF:			Flag de desbordamiento del TMR1:			
				1= Desbordamiento del TMR1 (borrado por software)			
				0= No se ha desbordado el TMR1			

Nota 1: Este pin está reservado en los micros de 28 pines; mantenerlo siempre borrado

REGISTRO 9-5: PIR2: Registro de respuesta de las interrupciones periféricas 2

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
OSCIF R/W-0	CMIIF R/W-0	USBIF R/W-0	EEIF R/W-0	BCLIF R/W-x	HLVIF R/W-0	TMR3IF R/W-0	CCP2IF R/Wx0
BIT 7	OSCIF:			Flag de fallo del oscilador: 1= Ha ocurrido un fallo (borrado por software) 0= No ha ocurrido ningún fallo			
BIT 6	CMIIF:			Flag del comparador: 1= La entrada del comparador ha cambiado (borrado por software) 0= La entrada no ha cambiado			
BIT 5	USBIF:			Flag del USB: 1= El USB pide una interrupción (borrado por software) 0= El USB no necesita interrupción			
BIT 4	EEIF:			Flag de escritura en la EEPROM/Flash: 1= Se ha completado la escritura (borrado por software) 0= No se ha completado la escritura			
BIT 3	BCLIF:			Flag del colisión en el bus: 1= Ha ocurrido una colisión (borrado por software) 0= No ha ocurrido colisión			
BIT 2	HLVIF:			Flag de detección de alta tensión: 1= Ha ocurrido una condición de alta tensión (borrado por software) 0= No ha ocurrido una condición de AT			
BIT 1	TMR3IF:			Flag de desbordamiento del TMR3: 1= Se ha desbordado el TMR3 (borrado por software) 0= No se ha desbordado			
BIT 0	CCP2IF:			Flag del CCP2: Modo captura: 1= Ha ocurrido una captura en TMR1 o TMR3 (borrado por software) 0= No ha ocurrido ninguna captura Modo comparación: 1= Ha ocurrido una comparación en el TMR1 o TMR3 (borrado por software) 0= No ha ocurrido una comparación Modo PWM: No se utiliza			

9.4 REGISTROS PIE

Los registros PIE contienen los bits de permiso individual para las interrupciones periféricas. Debido al número de fuentes de interrupción periféricas, se necesitan dos

registros de permiso (PIE1 y PIE2). Cuando IPEN = 0, el bit PEIE se debe activar para no permitir ninguna de estas interrupciones.

REGISTRO 9-6: PIE1: Registro de permiso de las interrupciones periféricas

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
SPPIE ⁽¹⁾	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE
R/W-0	R/W-0	R/W-0	R/W-0	R/W-x	R/W-0	R/W-0	R/Wx0
BIT 7	SPPIE:						
		Bit de permiso de interrupción de SPP ⁽¹⁾ :					
		1= Interrupción por una lectura/escritura permitida					
		0= No se permite la interrupción					
BIT 6	ADIE:						
		Bit de permiso de interrupción por el conversor A/D:					
		1= Interrupción activada					
		0= Interrupción desactivada					
BIT 5	RCIE:						
		Bit de permiso de interrupción por recepción					
		por la EUSART:					
		1= Interrupción activada					
		0= Interrupción desactivada					
BIT 4	TXIE:						
		Bit de permiso de la interrupción por transmisión					
		por la EUSART:					
		1= Interrupción activada					
		0= Interrupción desactivada					
BIT 3	SSPIE:						
		Bit de permiso de interrupción por el MSSP:					
		1= Interrupción activada					
		0= Interrupción desactivada					
BIT 2	CCP1IE:						
		Bit de permiso de interrupción por el CCP1:					
		1= Interrupción activada					
		0= Interrupción desactivada					
BIT 1	TMR2IE:						
		Bit de permiso de interrupción al igualar TMR2 y PR2:					
		1= Interrupción activada					
		0= Interrupción desactivada					
BIT 0	TMR1IE:						
		Bit de permiso de la interrupción por desbordamiento					
		del TMR1:					
		1= Interrupción activada					
		0= Interrupción desactivada					

Nota 1: Este pin está reservado en los micros de 28 pines; mantenerlo siempre borrado.

REGISTRO 9-7: PIE2: Registro de permiso de las interrupciones periféricas 2

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
OSCIE	CMIIE	USBIE	EEIE	BCLIE	HLVIE	TMR3IE	CCP2IE

R/W-0 BIT 7	R/W-0 OSCIE:	R/W-0 R/W-x	R/W-0 R/W-0	R/W-0 R/W-0	R/W-0 R/Wx0
		Bit de permiso de interrupción por fallo del oscilador: 1= Interrupción activada 0= Interrupción desactivada			
BIT 6	CMIE:	Bit de permiso de interrupción del comparador: 1= Interrupción activada 0= Interrupción desactivada			
BIT 5	USBIE:	Bit de permiso de interrupción del USB: 1= Interrupción activada 0= Interrupción desactivada			
BIT 4	EEIE:	Bit de permiso de interrupción por escribir en EEPROM/Flash: 1= Interrupción activada 0= Interrupción desactivada			
BIT 3	BCLIE:	Bit de permiso de interrupción por colisión en el bus: 1= Interrupción activada 0= Interrupción desactivada			
BIT 2	HLVIE:	Bit de permiso de interrupción por detección de alta tensión: 1= Interrupción activada 0= Interrupción desactivada			
BIT 1	TMR3IE:	Bit de permiso de interrupción por desbordamiento del TMR3: 1= Interrupción activada 0= Interrupción desactivada			
BIT 0	CCP2IE:	Bit de permiso de interrupción del CCP2: 1= Interrupción activada 0= Interrupción desactivada			

9.5 REGISTROS IPR

Los registros IPR contienen los bits de prioridad de las interrupciones periféricas. Debido al número de fuentes de interrupción periféricas, hay dos registros

(IPR1 e IPR2). Al usar los bits de la prioridad requieren que la prioridad de la interrupción esté permita (IPEN activado).

REGISTRO 9-8: IPR1: Registro de prioridad de las interrupciones periféricas

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
SPPIP ⁽¹⁾	ADIP	RCIP	TXIP	SSPIP	CCP1IP	TMR2IP	TMR1IP
R/W-0	R/W-0	R/W-0	R/W-0	R/W-x	R/W-0	R/W-0	R/Wx0
BIT 7	SPPIP:			Bit de prioridad de la interrupción de SPP ⁽¹⁾ :			
				1= Alta prioridad			
				0= Baja prioridad			
BIT 6	ADIP:			Bit de prioridad de la interrupción por el conversor A/D:			
				1= Alta prioridad			
				0= Baja prioridad			
BIT 5	RCIP:			Bit de prioridad de la interrupción por recepción			
				por la EUSART:			
				1= Alta prioridad			
				0= Baja prioridad			
BIT 4	TXIP:			Bit de prioridad de la interrupción por transmisión			
				por la EUSART:			
				1= Alta prioridad			
				0= Baja prioridad			
BIT 3	SSPIP:			Bit de prioridad de la interrupción por el MSSP:			
				1= Alta prioridad			
				0= Baja prioridad			
BIT 2	CCP1IP:			Bit de prioridad de la interrupción por el CCP1:			
				1= Alta prioridad			
				0= Baja prioridad			
BIT 1	TMR2IP:			Bit de prioridad de la interrupción al igualar			
				TMR2 y PR2:			
				1= Alta prioridad			
				0= Baja prioridad			
BIT 0	TMR1IP:			Bit de prioridad de la interrupción por desbordamiento			
				del TMR1:			
				1= Alta prioridad			
				0= Baja prioridad			

Nota 1: Este pin está reservado en los micros de 28 pines; mantenerlo siempre borrado

REGISTRO 9-9: IPR2: Registro de prioridad de las interrupciones periféricas 2

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
-------	-------	-------	-------	-------	-------	-------	-------

	OSCIP R/W-0	CMIP R/W-0	USBIP R/W-0	EEIP R/W-0	BCLIP R/W-x	HLVIP R/W-0	TMR3IP R/W-0	CCP2IP R/Wx0
BIT 7		OSCIP:			Bit de prioridad de la interrupción por fallo del oscilador: 1= Alta prioridad 0= Baja prioridad			
BIT 6		CMIP:			Bit de prioridad de la interrupción del comparador: 1= Alta prioridad 0= Baja prioridad			
BIT 5		USBIP:			Bit de prioridad de la interrupción del USB: 1= Alta prioridad 0= Baja prioridad			
BIT 4		EEIP:			Bit de prioridad de la interrupción por escribir en EEPROM/Flash: 1= Alta prioridad 0= Baja prioridad			
BIT 3		BCLIP:			Bit de prioridad de la interrupción por colisión en el bus: 1= Alta prioridad 0= Baja prioridad			
BIT 2		HLVIP:			Bit de prioridad de la interrupción por detección de alta tensión: 1= Alta prioridad 0= Baja prioridad			
BIT 1		TMR3IP:			Bit de prioridad de la interrupción por desbordamiento del TMR3: 1= Alta prioridad 0= Baja prioridad			
BIT 0		CCP2IP:			Bit de prioridad de la interrupción del CCP2: 1= Alta prioridad 0= Baja prioridad			

9.6 REGISTRO RCON

El registro RCON contiene los flags los cuales se utilizan para determinar la causa del reset o del reinicio que ha provocado la salida de los modos de reposo o

SLEEP. RCON también contiene el bit IPEN el cuál permite las prioridades de las interrupciones.

REGISTRO 9-10: RCON: Registro de control del Reset

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
IPEN	SBOREN	-	RI*	TO*	PD*	POR*	BOR*
R/W-0	R/W-1 ⁽¹⁾	U-0	R/W-0	R-1	R-1	R/W-0	R/W-0
BIT 7	IPEN:						
							Bit activador de las prioridades de las interrupciones: 1=Niveles de prioridad permitidos 0=Niveles de prioridad no permitidos
BIT 6	SBOREN:						Bit activador de BOR ⁽¹⁾ : Ver registro 4-1
BIT 4	RI*:						Flag de reset por instrucción: Ver registro 4-1
BIT 3	TO*:						Flag de desbordamiento del Watch-dog: Ver registro 4-1
BIT 2	PD*:						Flag de detección de la caída de tensión: Ver registro 4-1
BIT 1	POR*:						Bit de estado de reset por aumento de tensión ⁽²⁾ : Ver registro 4-1
BIT 0	CCP2IP:						Bit de estado de reset por cese de tensión: Ver registro 4-1

Nota 1: Si se permite SBOREN, su estado en el reset es ‘1’; si no, es ‘0’. Ver el registro 4-1 para la información adicional.

2: El valor real en el reset del POR se determina por el tipo de reset del dispositivo. Ver el registro 4-1 para adicional información.

9.7 PINES DE INTERRUPCIONES INTn

Las interrupciones externas de los pines RB0/AN12/INT0/FLT0/SDI/SDA, RB1/AN10/INT1/SCK/SCL y RB2/AN8/INT2/VMO son activas por flanco. Si el bit correspondiente INTEDGx en el registro INTCON2 se activa (= 1), la interrupción se acciona por un flanco de subida; si el bit está borrado, el disparo ocurre en el flanco de bajada. Cuando ocurre un flanco válido en el pin RBx/INTx, el flag el correspondiente se activa, INTxIF. Esta interrupción se puede inhabilitar borrando el bit INTxIE. El flag, INTxIF, debe borrarse por software antes de volver a permitir la interrupción.

Todas las interrupciones externas (INT0, INT1 e INT2) pueden despertar al procesador de los modos de ahorro de energía si se activa el bit, INTxIE antes de entrar en ellos. Si se permiten las interrupciones globales, se activa GIE, el procesador saltará al puntero de la interrupción después del reinicio.

La prioridad de la interrupción para INT1 e INT2 se determina con el valor contenido en los bits de prioridad de la interrupción, el INT1IP (INTCON3<6>) y el INT2IP (INTCON3<7>). No hay un bit de prioridad asociado a la interrupción INT0. Siempre es una fuente de interrupción prioritaria.

9.8 INTERRUPCIÓN TMR0

En el modo de 8 bits (que está por defecto), un desbordamiento en el registro TMR0 (FFh => 00h) activará el flag, TMR0IF. En modo de 16 bits, un desbordamiento en el par de registros TMR0H:TMR0L (FFFFh => 0000h) activará TMR0IF. La interrupción se puede permitir/inhabilitar activando/borrando el bit de permiso,

TMR0IE (INTCON<5>). La prioridad de la interrupción para Timer0 se determina por el valor contenido en el bit de prioridad de la interrupción, TMR0IP (INTCON2<2>).

9.9 INTERRUPCIÓN AL CAMBIAR PORTB

Un cambio en la entrada en PORTB<7:4> activa el flag, RBIF (INTCON<0>). La interrupción se puede permitir/inhabilitar activando/borrando el bit de permiso, RBIE (INTCON<3>). La prioridad de la interrupción para la interrupción se determinada con el valor contenido en el bit de la prioridad de la interrupción, RBIP (INTCON2<0>).

9.10 ALMACENAMIENTO DE DATOS DURANTE LA INTERRUPCIÓN

Durante las interrupciones, la dirección del PC de retorno se almacena en la pila. Además, los registros WREG, ESTADO y BSR se guardan en la pila de acceso rápido. Si no se utiliza un retorno rápido de la interrupción, el usuario puede necesitar guardar los registros WREG, ESTADO y BSR en entrada a la rutina de la interrupción. Dependiendo de la aplicación, también se puede necesitar guardar otros registros.

EJEMPLO 9-1: Almacenamiento de los registros ESTADO, WREG y BSR en RAM

MOVWF W_TEMP	;W_TEMP está en el banco virtual
MOVFF STATUS,STATUS_TEMP	;STATUS_TEMP direccionado en ;cualquier sitio
MOVFF BSR, BSR_TEMP	;BSR_TMEP direccionado en ;cualquier sitio
;	
;CÓDIGO DEL USUARIO	
;	
MOVFF BSR_TEMP, BSR	;Restaura BSR
MOVF W_TEMP, W	;Restaura WREG
MOVFF STATUS_TEMP, STATUS	;Restaura ESTADO

10.0 PUERTOS DE ENTRADA-SALIDA

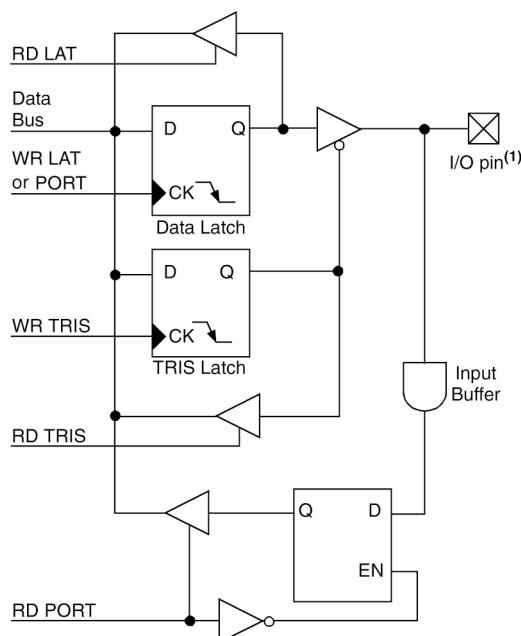
Dependiendo del dispositivo seleccionado y de las características permitidas, hay hasta cinco puertos disponibles. Algunos pines de los puertos de entrada-salida se multiplexan con una función alternativa de las características periféricas del dispositivo. Generalmente cuando se activa un periférico, ese pin no se puede utilizar como pin de entrada-salida.

Cada puerto tiene tres registros para operar. Estos registros son:

- Registro TRIS (registro de la dirección de los datos)
- Registro POR (lee los niveles en los pines del dispositivo)
- Registro LAT (latch de salida)

El registro del latch de datos (LATA) es útil para leer-modificar-escribir las operaciones en el valor de los pines de entrada-salida.

FIGURA 10-1: PUERTO GENÉRICO DE OPERACIONES DE ENTRADA-SALIDA



Nota 1: Los puertos de E/S tienen un diodo de protección para V_{DD} y V_{SS}.

10.1 REGISTROS PORTA, TRISA Y LATA

El PORTA es un puerto bidireccional de 8bits de ancho. El registro de la dirección de los datos es TRISA. Activar el bit TRISA (=1) hará que el pin correspondiente de PORTA sea una entrada (es decir, pone el conductor correspondiente de la salida en un modo alta impedancia). Borrar un bit de TRISA (=0) hace que el pin correspondiente de PORTA sea una salida (es decir, pone el contenido del latch de salida en el pin seleccionado).

Leyendo el registro PORTA se lee el estado de los pines; escribiendo en él, se graba el latch.

El registro del latch de datos (LATA) sólo es memoria mapeada. Las operaciones de lectura-modificación-escritura en el registro LATA lee y escribe el valor del latch de salida del PORTA.

El pin RA4 se multiplexa con la entrada de reloj del módulo Timer0 para hacerse el pin RA4/T0CKI. El pin RA6 se multiplexa con el pin principal del oscilador; se permite como un oscilador o pin de entrada-salida al seleccionar el oscilador

principal en el registro 1H de la configuración. Cuando no está utilizado como un puerto, RA6 y su TRIS asociado y los bits del LAT se leen como ‘0’.

El RA4 también se multiplexa con el módulo del USB; sirve como una entrada receptora de un transmisor-receptor externo del USB.

Varios pines de PORTA se multiplexan con las entradas analógicas, las entradas analógicas de VREF+ y de VREF- y la salida de referencia de la tensión del comparador. La operación de los pines RA5 y RA3:RA0 como entradas del convertidor A/D se seleccionan al borrar/activar los bits de control en el registro ADCON1 (registro de control A/D1).

Nota: En un reset por subida de tensión, RA5 y RA3:RA0 se configuran como entradas analógicas y se leen como ‘0’. RA4 se configura como entrada digital.

El resto de los pines de PORTA tienen niveles de introducción de datos de TTL y drivers de salida de CMOS.

El registro TRISA controla la dirección de los pines RA, aun cuando se están utilizando como entradas analógicas. El usuario debe asegurarse de que los bits del registro TRISA son ‘1’ al usarlos como entradas analógicas.

EJEMPLO 10-1: INCIALIZACIÓN PORTA

CLRF PORTA	;Inicializar PORTA borrando los latch de salida
CLRF LATA	;Otro método para borrar los latch
MOVLW 0Fh	;Configurar A/D
MOVWF ADCON1	;para entradas digitales
MOVLW 07h	;Configurar los comparadores
MOVWF CMCON	;para entrada digital
MOVLW 0CFh	;Valor usado para inicializar las direcciones de los datos
MOVWF TRISA	;Activar RA<3:0> como entradas y RA<5:4> como salidas

TABLA 10-1: RESUMEN DE ENTRADAS-SALIDAS DEL PORTA

Pin	Función	Configuración Del TRIS	E/S	Tipo de E/S	Descripción
RA0/AN0	RA0	0	salida	salida digital	LATA<0> salida de datos; no le afectan las entradas analógicas
		1	entrada	Buffer de entrada TTL	PORTA<0> entradas de datos; desactivada cuando se activen las entradas analógicas
	AN0	1	entrada	analógica	Canal 0 de entrada A/D y entrada del comparador C1-. Configuración por defecto en POR; no le afectan salidas digitales.
RA1/AN1	RA1	0	salida	salida digital	LATA<1> salida de datos; no le afectan las entradas analógicas
		1	entrada	Buffer de entrada TTL	PORTA<1> entrada de datos; se lee '0' en POR.
	AN1	1	entrada	analógica	Canal 1 de entrada A/D y entrada del comparador C2-. Configuración por defecto en POR; no le afectan salidas digitales.
RA2/AN2/ VREF-/CVREF	RA2	0	salida	salida digital	LATA<2> salida de datos; no le afectan las entradas analógicas. Desactivado cuando la salida CVREF esté activa.
		1	entrada	Buffer de entrada TTL	PORTA<2> entrada de datos. Desactivado cuando las funciones analógicas estén activas; desactivado cuando la salida CVREF esté activa.
	AN2	1	entrada	analógica	Canal 2 de entrada A/D y entrada del comparador C2+. Configuración por defecto en un POR; no le afectan las salidas analógicas.
	VREF-	1	entrada	analógica	Tensión baja de referencia de entrada para el A/D y el comparador.
	CVREF	x	salida	analógica	Tensión de referencia de salida. Activando esto se desactiva las E/S digitales.
RA3/AN3/ VREF+	RA3	0	salida	salida digital	LATA<3> salida de datos; no le afectan las entradas analógicas
		1	entrada	Buffer de entrada TTL	PORTA<3> entrada de datos; desactivado cuando las entradas analógicas estén activas.
	AN3	1	entrada	analógica	Canal 3 de entrada A/D y del comparador C1+. Configuración por defecto en un POR.
	VREF+	1	entrada	analógica	Tensión de referencia alta de entrada para el A/D y el comparador.
RA4/T0CKI/ C1OUT/RCV	RA4	0	salida	salida digital	LATA<4> salida de datos; no le afectan las entradas analógicas
		1	entrada	Buffer de entrada Schmitt	PORTA<4> entrada de datos; desactivado cuando las entradas analógicas estén activas.
	T0CKI	1	entrada	Buffer de entrada Schmitt	Entrada del reloj Timer0.
	C1OUT	0	salida	salida digital	Salida del comparador 1; tiene prioridad sobre el puerto de datos.
	RCV	x	entrada	Buffer de entrada TTL	Entrada externa del transmisor RCV del USB.
RA5/AN4/SS*/ HLVDIN/C2OUT	RA5	0	salida	salida digital	LATA<5> salida de datos; no le afectan las entradas analógicas
		1	entrada	Buffer de entrada TTL	PORTA<5> entrada de datos; desactivado cuando las entradas analógicas estén activas.
	AN4	1	entrada	analógica	Canal 4 de entrada A/D. Por defecto en un POR.
	SS*	1	entrada	Buffer de entrada TTL	Entrada selectora de esclavo para SSP (Módulo MSSP).
	HLVDIN	1	entrada	analógica	Detector de tensión Alta/Baja externa.
OSC2/CLKO/RA6	OSC2	x	salida	analógica	Conexión de regreso del oscilador (todos los modos XT y HS).
	CLKO	x	salida	salida digital	Salida del sistema del reloj (FOSC/4); disponible en los modos EC, ECPLL e INTCKO.
	RA6	0	salida	salida digital	LATA<6> salida de datos. Disponible en los modos ECIO, ECPIO e INTIO; en otro caso, se lee '0'.
		1	entrada	Buffer de entrada TTL	PORTA<6>entrada de datos. Disponible en los modos ECIO, ECPIO e INTIO; en otro caso se lee '0'.

TABLA 10-2: SUMARIO DE REGISTROS ASOCIADOS AL PORTA

Nombre	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PORTA	—	RA6 ⁽¹⁾	RA5	RA4	RA3	RA2	RA1	RA0
LATA	—	LATA6 ⁽¹⁾	LATA5	LATA4	LATA3	LATA2	LATA1	LATA0
TRISA	—	TRISA6 ⁽¹⁾	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0
ADCON1	—	—	VCFG1	VCFG0	PCFG3	PCFG2	PCFG1	PCFG0
CMCN	C2OUT	C1OUT	C2INV	C1INV	CIS	CM2	CM1	CM0
CVRCN	CVREN	CVROE	CVRR	CVRSS	CVR3	CVR2	CVR1	CVR0
UCON	—	PPBRST	SE0	PKTDIS	USBEN	RESUME	SUSPND	—

Leyenda: — = no implementado, se lee ‘0’. Las celdas sombreadas no las usa el PORTA.

Nota 1: El RA6 y sus bits de direcciones y latch asociados se activan como pines de E/S basándose en la configuración del oscilador; en otro caso, se leen como ‘0’.

10.2 REGISTROS PORTB, TRISB y LATB

El PORTB es un puerto bidireccional de 8bits de ancho. El registro de la dirección de los datos es TRISB. Activar el bit TRISB (=1) hará que el pin correspondiente de PORTB sea una entrada (es decir, pone el conductor correspondiente de la salida en un modo alta impedancia). Borrar un bit de TRISB (=0) hace que el pin correspondiente de PORTB sea una salida (es decir, pone el contenido del latch de salida en el pin seleccionado).

El registro del latch de datos (LATB) sólo es memoria mapeada. Las operaciones de lectura-modificación-escritura en el registro LATB lee y escribe el valor del latch de salida del PORTB.

Cada uno de los pines de PORTB tiene un pull-up interno. Un solo bit de control puede cambiar todo el pull-up. Esto se realiza borrando el bit, RBPU* (INTCON2<7>). El pull-up cambia automáticamente cuando el pin se configura como salida. Se inhabilita el pull-up en un reinicio.

Nota: En reinicio, RB4:RB0 se configuran como entradas analógicas por defecto y se leen como ‘0’; Se configuran los bits RB7:RB5 como entradas digitales.

Programando el bit de configuración, PBADEN (CONFIG3H<1>), RB4:RB0 se configuran alternativamente como entradas digitales en POR.

Cuatro de los pines de PORTB (RB7: RB4) tienen una interrupción “al cambiar”. Solamente los pines configurados como entradas pueden provocar esta interrupción. Cualquier pin RB7:RB4 configurado como salida se excluye de la interrupción “al cambiar”. Los pines se comparan con el último valor leído en el PORTB.

La interrupción se puede utilizar para despertar al dispositivo del sleep. El usuario, en la rutina del servicio de la interrupción, puede inhabilitar la interrupción de la manera siguiente:

- a) Una lectura o escritura en el PORTB (excepto con la instrucción MOVFF x, PORTB). Esto terminará la condición de error de conexión.
- b) Borrar el flag, RBIF.

Al leer el PORTB finalizará la condición de error de conexión y permitirá que el flag, RBIF, sea borrado.

La interrupción “por cambio” se recomienda para despertar el dispositivo cuando se presiona una tecla y operaciones en las que el PORTB se utiliza solamente

por su interrupción. No es recomendable interrogar el PORTB cuando estamos usando la interrupción “por cambio”.

Los pines, RB2 y RB3, se multiplexan con el USB y sirven como las salidas para un transmisor-receptor externo USB (configuración TRIS).

RB4 se multiplexa con CSSPP, el chip selecciona funcionar con puerto paralelo (SPP) ajustando el TRIS.

EJEMPLO 10-2: INCIALIZACIÓN DE PORTB

CLRF PORTB	;Inicializar PORTB borrando los latch de salida de datos
CLRF LATB	;Otra método para borrar los latch de salida
MOVLW 0Eh	;Activar RB<4:0> como pines digitales de entrada-salida
MOVWF ADCON1	; (requerido si el bit PBADEN está activo)
MOVLW 0CFh	;Valor usado para inicializar las direcciones de los datos
MOVWF TRISB	;Activa RB<3:0> como entradas ;RB<5:4> como salidas ;RB<7:6> como entradas

TABLA 10-3: RESUMEN DE ENTRADAS-SALIDAS DEL PORTB

Pin	Función	Configuración Del TRIS	E/S	Tipo De E/S	Descripción
RB0/AN12/ INT0/FLT0/ SDI/SDA	RB0	0	salida	salida digital	LATB<0> salida de datos; no le afectan las entradas analógicas
		1	entrada	Buffer de entrada TTL	PORTB<0> entrada de datos; cuando el bit RBPU se borre, poca resistencia. Desactivada cuando se activen las entradas analógicas ⁽¹⁾
	AN12	1	entrada	señal analógica	Canal de entrada A/D 12. ⁽¹⁾
	INT0	1	entrada	Buffer de entrada Schmitt	Entrada de interrupción externa 0
	FLT0	1	entrada	Buffer de entrada Schmitt	Entrada de PWM mejorado contra fallos (Módulo ECCP1); activo por software
	SDI	1	entrada	Buffer de entrada Schmitt	SPI entrada de datos (Módulo MSSP).
	SDA	1	salida	salida digital	Salida de datos I ² C (Módulo MSSP); tiene prioridad sobre el puerto de datos.
		1	entrada	I2C/SMB	Entrada de datos I ² C (Módulo MSSP); el tipo de entrada depende de las características seleccionadas en el módulo.
RB1/AN10/ INT1/SCK/ SCL	RB1	0	salida	salida digital	LATB<1> salida de datos; no le afectan las entradas analógicas
		1	entrada	Buffer de entrada TTL	PORTB<1> entrada de datos; cuando el bit RBPU se borre, poca resistencia. Desactivada cuando se activen las entradas analógicas ⁽¹⁾
	AN10	1	entrada	señal analógica	Canal de entrada A/D 10. ⁽¹⁾
	INT1	1	entrada	Buffer de entrada Schmitt	Entrada de interrupción externa 1
	SCK	0	salida	salida digital	Salida de reloj SPI (Módulo MSSP); tiene prioridad sobre el puerto de datos.
		1	entrada	Buffer de entrada Schmitt	Entrada de reloj SPI (Módulo MSSP).
	SCL	0	salida	salida digital	Salida de reloj I ² C (Módulo MSSP); tiene prioridad sobre el puerto de datos.
		1	entrada	I2C/SMB	Entrada de reloj I ² C (Módulo MSSP); el tipo de entrada depende de las características seleccionadas del módulo.
RB2/AN8/ INT2/VMO	RB2	0	salida	salida digital	LATB<2> salida de datos; no le afectan las entradas analógicas
		1	entrada	Buffer de entrada TTL	PORTB<2> entrada de datos; cuando el bit RBPU se borre, poca resistencia. Desactivada cuando se activen las entradas analógicas ⁽¹⁾
	AN8	1	entrada	señal analógica	Canal de entrada A/D 8. ⁽¹⁾
	INT2	1	entrada	Buffer de entrada Schmitt	Entrada de interrupción externa 2
	VMO	0	salida	salida digital	Salida de datos del transmisor externo USB VMO.

RB3/AN9/ CCP2/VPO	RB3	0	salida	salida digital	LATB<3> salida de datos; no le afectan las entradas analógicas
		1	entrada	Buffer de entrada TTL	PORTB<3> entrada de datos; cuando el bit RBPU se borre, poca resistencia. Desactivada cuando se activen las entradas analógicas ⁽¹⁾
	CCP2 ⁽²⁾	1	entrada	señal analógica	Canal de entrada A/D 9. ⁽¹⁾
		0	salida	salida digital	Salida del comparador CCP2 y del PWM.
	VPO	0	salida	Buffer de entrada Schmitt	Capturador de entrada CCP2.
					Salida de datos del transmisor externo USB VPO.
RB4/AN11/ KBI0/CSSPP	RB4	0	salida	salida digital	LATB<4> salida de datos; no le afectan las entradas analógicas
		1	entrada	Buffer de entrada TTL	PORTB<4> entrada de datos; cuando el bit RBPU se borre, poca resistencia. Desactivada cuando se activen las entradas analógicas ⁽¹⁾
	AN11	1	entrada	señal analógica	Canal de entrada A/D 11. ⁽¹⁾
	KBI0	1	entrada	Buffer de entrada TTL	Interrupción al cambiar.
	CSSPP ⁽⁴⁾	0	salida	salida digital	Salida de control del chip select de SPP.
RB5/KBI1/ PGM	RB5	0	salida	salida digital	LATB<5> salida de datos.
		1	entrada	Buffer de entrada TTL	PORTB<5> entrada de datos; cuando el bit RBPU se borre, poca resistencia.
	KBI1	1	entrada	Buffer de entrada TTL	Interrupción al cambiar.
	PGM	x	entrada	Buffer de entrada Schmitt	Modo de programación de suministro único (ICSP™). Activo por el bit LVP; todas las funciones de los demás pines desactivadas.
RB6/KBI2/ PGC	RB6	0	salida	salida digital	LATB<6> salida de datos.
		1	entrada	Buffer de entrada TTL	PORTB<6> entrada de datos; cuando el bit RBPU se borre, poca resistencia.
	KBI2	1	entrada	Buffer de entrada TTL	Interrupción al cambiar.
	PGC	x	entrada	Buffer de entrada Schmitt	Reloj de entrada serie para operaciones ICSP e ICD. ⁽³⁾
RB7/KBI3/ PGD	RB7	0	salida	salida digital	LATB<7> salida de datos.
		1	entrada	Buffer de entrada TTL	PORTB<7> entrada de datos; cuando el bit RBPU se borre, poca resistencia.
	KBI3	1	entrada	Buffer de entrada TTL	Interrupción al cambiar.
	PGD	x	salida	salida digital	Salida serie de datos para operaciones ICSP e ICD. ⁽³⁾
		x	entrada	Buffer de entrada Schmitt	Entrada serie de datos para operaciones ICSP e ICD. ⁽³⁾

Nota 1: Configuración en el POR determinada por el bit PBADEN. Los pines se configuran como entradas analógicas cuando el bit PBADAN está activo y entradas digitales cuando está borrado.

2: Selección de pin alternativo de CCP2 cuando CCP2MX = 0. La selección natural es RC1.

3: Todas las funciones de los pines están desactivadas cuando ICSP o ICD se activan.

4: Sólo en los dispositivos de 40/44-pines.

TABLA 10-4: EL RESUMEN DE REGISTROS ASOCIADOS AL PORTB

Nombre	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PORTB	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0
LATB	LATB7	LATB6	LATB5	LATB4	LATB3	LATB2	LATB1	LATB0
TRISB	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0
INTCON	GIE/GIEH	PEIE/GIEL	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF
INTCON2	RBU*	INTEDG0	INTEDG1	INTEDG2	—	TMR0IP	—	RBIP
INTCON3	INT2IP	INT1IP	—	INT2IE	INT1IE	—	INT2IF	INT1IF
ADCON1	—	—	VCFG1	VCFG0	PCFG3	PCFG2	PCFG1	PCFG0
SPPCON(1)	—	—	—	—	—	—	SPPOWN	SPPEN
SPPCFG(1)	CLKCFG1	CLKCFG0	CSEN	CLK1EN	WS3	WS2	WS1	WS0
UCON	—	PPBRST	SE0	PKTDIS	USBEN	RESUME	SUSPND	—

Leyenda: — = no implementado, se lee como ‘0’. Las celdas sombreadas no se utilizan para el PORTB.

Nota: Estos registros son solo para los dispositivos de 28 pins.

10.3 REGISTROS PORTC, TRISC Y LATC

El PORTC es un puerto bidireccional de 7bits de ancho. El registro de la dirección de los datos es TRISC. Activar el bit TRISC (=1) hará que el pin correspondiente de PORTC sea una entrada (es decir, pone el conductor correspondiente de la salida en un modo alta impedancia). Borrar un bit de TRISC (=0) hace que el pin correspondiente de PORTC sea una salida (es decir, pone el contenido del latch de salida en el pin seleccionado).

En los dispositivos PIC18F2455/2550/4455/4550, el pin RC3 no está implementado.

El registro del latch de datos (LATC) sólo es memoria mapeada. Las operaciones de lectura-modificación-escritura en el registro LATC lee y escribe el valor del latch de salida del PORTC.

PORTC se multiplexa sobre todo con los módulos de comunicación serie, incluyendo el EUSART, el módulo MSSP y el módulo USB. A excepción de RC4 y RC5, el PORTC utiliza buffer de disparo de entrada Schmitt.

Los pines RC4 y RC5 se multiplexan con el módulo USB. Dependiendo de la configuración del módulo, pueden servir como las líneas de datos diferenciales para el transmisor-receptor del USB del chip, o las entradas de datos del transmisor-receptor del USB externo. RC4 y RC5, el PORTC tiene buffer de entrada TTL en lugar de los buffer Schmitt de los otros pines.

RC4 y RC5 no tienen bits de TRISC asociados a ellos. Como puertos digitales, sólo pueden funcionar como entradas digitales. Cuando se configura para operaciones con USB, la dirección de los datos se determina por la configuración y el estado del módulo USB en ese momento. Si se utiliza un transmisor-receptor externo, siempre funcionarán RC4 y RC5 como entradas del transmisor-receptor. Si se utiliza el transmisor-receptor del chip, la dirección de los datos se determina por la operación que se realiza por el módulo en el momento.

Cuando se permite el transmisor-receptor externo, RC2 sirve como la salida de control del transmisor-receptor.

Al permitir funciones periféricas en los pines de PORTC con excepción de RC4 y de RC5, hay que tener cuidado al definir los bits de TRIS. Algunos periféricos eliminan el bit TRIS para configurar un pin como salida, mientras que otros periféricos activan el bit TRIS para configurar un pin como entrada.

Nota: En un reset por subida de tensión, estos pines, excepto RC4 y RC5, se configuran como entradas digitales. Para utilizar los pines RC4 y RC5 como entradas digital, el módulo USB debe desactivarse (`UCON<3>=0`) y en el chip el transmisor-receptor del USB debe desactivarse también (`UCFG<3>=1`).

El contenido del registro de TRISC afectado por el periférico se elimina. La lectura de TRISC siempre devuelve el contenido actual, aun cuando un dispositivo periférico puede eliminar uno o más pines.

EJEMPLO 10-3: INCIALIZACIÓN DE PORTC

CLRF PORTC	;Inicializar PORTC borrando los latch de salida de datos
CLRF LATC	;Otra método para borrar los latch de salida
MOVLW 07h	;Valor usado para inicializar las direcciones de los datos
MOVWF TRISC	;Activa RC<5:0> como salidas ;RC<7:6> como entradas

TABLA 10-5: RESUMEN DE ENTRADAS-SALIDAS DEL PORTC

Pin	Función	Ajustes del TRIS	E/S	Tipo de E/S	Descripción
RC0/T1OS O/ T13CKI	RC0	0	S	salida digital	LATC<0> salida de datos.
		1	E	Buffer de entrada Schmitt	PORTC<0> entrada de datos.
	T1OSO	x	S	señal analógica	Salida del oscilador del Timer1; activo cuando se active el oscilador del Timer1. Inhabilita las E/S digitales.
	T13CKI	1	E	Buffer de entrada Schmitt	Entrada del contador Timer1/Timer3.
RC1/T1OSI /CCP2/UOE *	RC1	0	S	salida digital	LATC<1> salida de datos.
		1	E	Buffer de entrada Schmitt	PORTC<1> entrada de datos.
	T1OSI	x	E	señal analógica	Entrada del oscilador Timer1; activo cuando se active el oscilador Timer1. Desactiva las E/S digitales.
	CCP2 ⁽¹⁾	0	S	salida digital	Salida del comparador CCP2 y del PWM output; tiene prioridad sobre el puerto de datos.
		1	E	Buffer de entrada Schmitt	Entrada de captura CCP2.
	UOE*	0	S	salida digital	Salida del transmisor OE externo del USB.
RC2/CCP1/ P1A	RC2	0	S	salida digital	LATC<2> salida de datos.
		1	E	Buffer de entrada Schmitt	PORTC<2> entrada de datos.
	CCP1	0	S	salida digital	Salida del comparador ECCP1 y del PWM output; tiene prioridad sobre el puerto de datos.
		1	E	Buffer de entrada Schmitt	Entrada de captura ECCP1.
	P1A ⁽³⁾	0	S	salida digital	Salida ECCP1 y PWM mejorado, canal A; tiene prioridad sobre el puerto de datos. Tiene que configurarse como tri-estado durante los eventos de caída del PWM mejorado.
RC4/D- /VM	RC4	— ⁽²⁾	E	Buffer de entrada TTL	PORTC<4> entrada de datos; desactivado cuando el modulo transmisor del USB del chip esté activo.
	D-	— ⁽²⁾	S	transmisor USB	Línea de salida del bus USB diferencial negativo USB (transmisor interno).
		— ⁽²⁾	E	transmisor USB	Línea de entrada del bus USB diferencial negativo USB (transmisor interno).
	VM	— ⁽²⁾	E	Buffer de entrada TTL	Entrada del USB VM externo.
RC5/D+/V P	RC5	— ⁽²⁾	E	Buffer de entrada TTL	PORTC<5> entrada de datos; desactivado cuando el modulo transmisor del USB del chip esté activo.
	D+	— ⁽²⁾	S	transmisor USB	Línea de salida del bus USB diferencial positivo USB (transmisor interno).
		— ⁽²⁾	E	transmisor USB	Línea de entrada del bus USB diferencial positivo USB (transmisor interno).
	VP	— ⁽²⁾	E	Buffer de entrada TTL	Entrada del USB VP externo.
	TX	0	S	salida digital	Salida de transmisión serie asíncrona (módulo EUSART); tiene prioridad sobre el puerto de datos. El usuario tiene que configurarla como salid.
RC6/TX/C K	RC6	0	S	salida digital	Salida de transmisión serie síncrona (módulo EUSART); tiene prioridad sobre el puerto de datos.
		1	E	Buffer de entrada Schmitt	Entrada del reloj serie síncrono (Módulo EUSART).
	CK	0	S	salida digital	Entrada del reloj serie síncrono (Módulo EUSART).
	RC7	0	S	salida digital	PORTC<7> salida de datos
		1	E	Buffer de entrada Schmitt	PORT<7> entrada de datos
RC7/RX/D T/SDO	RX	1	E	Buffer de entrada Schmitt	Entrada de datos del receptor serie asíncrono (módulo EUSART).
	DT	1	S	salida digital	Salida serie de datos síncrona (modulo EUSART); tiene prioridad sobre SPI y el puerto de datos.
		1	E	Buffer de entrada Schmitt	Entrada de datos serie síncrona (módulo EUSART). El usuario tiene que configurarla como entr.
	SDO	0	S	salida digital	Salida de datos SPI (módulo MSSP); tiene prioridad sobre el puerto de datos.

Nota 1: Asignación del pin por defecto. La asignación del pin alternativa es RB3 (cuando CCP2MX = 0).

2: RC4 y RC5 no tienen bits correspondientes de TRISC. En el modo de puertos estos pines son sólo entradas. La dirección de los datos del USB se determina por la configuración del USB.

3: Solamente dispositivos de 40/44pines.

TABLA 10-6: EL RESUMEN DE REGISTROS ASOCIADOS AL PORTC

Nombre	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PORTC	RC7	RC6	RC5 ⁽¹⁾	RC4 ⁽¹⁾	—	RC2	RC1	RC0
LATC	LATC7	LATC6	—	—	—	LATC2	LATC1	LATC0
TRISC	TRISC7	TRISC6	—	—	—	TRISC2	TRISC1	TRISC0
UCON	—	PPBRST	SE0	PKTDIS	USBEN	RESUME	SUSPND	—

Leyenda: — = no implementado, se lee como ‘0’. Las celdas sombreadas no se utilizan para el PORTC.

Nota 1: RC5 y RC4 están solamente disponibles como puertos cuando el módulo del USB está desactivado (UCON<3> = 0).

10.4 REGISTROS PORTD, TRISD Y LATD

Nota: El PORTD solamente está disponible en dispositivos de 40/44pines.

El PORTD es un puerto bidireccional de 8bits de ancho. El registro de la dirección de los datos es TRISD. Activar el bit TRISD (=1) hará que el pin correspondiente de PORTD sea una entrada (es decir, pone el conductor correspondiente de la salida en un modo alta impedancia). Borrar un bit de TRISD (=0) hace que el pin correspondiente de PORTD sea una salida (es decir, pone el contenido del latch de salida en el pin seleccionado).

El registro del latch de datos (LATD) sólo es memoria mapeada. Las operaciones de lectura-modificación-escritura en el registro LATD lee y escribe el valor del latch de salida del PORTD.

Todos los pines del PORTD están implementados con buffer de entrada de disparadores Shmitt. Cada pin se puede configurar independientemente como entrada o salida.

Cada uno de los pines de PORTD tiene un pull-up interno. Un solo bit de control puede cambiar todo el pull-up. Esto se realiza borrando el bit, RDPU (PORTE<7>). El pull-up cambia automáticamente cuando el pin se configura como salida. Se inhabilita el pull-up en un reinicio.

Tres de los pines de PORTD se multiplexan con salidas, P1B, P1C y P1D, del módulo CCP.

Nota: En un reset por subida de tensión estos pines se configuran como entrada de datos digital.

El PORTD también se puede configurar como un puerto paralelo (SPP) de 8bits de ancho. En este modo, los Buffers de entrada son TTL.

Nota: Cuando se utiliza el modo PWM mejorado con salidas dobles o cuádruples, las funciones del MSSP en el PORTD se desactivan automáticamente.

EJEMPLO 10-4: INCIALIZACIÓN DE PORTD

CLRF PORTD	;Inicializar PORTD borrando los latch de salida de datos
CLRF LATD	;Otra método para borrar los latch de salida
MOVLW 0C7h	;Valor usado para inicializar las direcciones de los datos
MOVWF TRISD	;Activa RD<3:0> como entradas ;RD<5:4> como salidas ;RD<7:6> como entradas

TABLA 10-7: RESUMEN DE ENTRADAS-SALIDAS DEL PORTD

Pin	Función	Ajustes del TRIS	E/S	Tipo de E/S	Descripción
RD0/SPP0	RD0	0	salida	Salida digital	LATD<0> salida de datos.
		1	entrada	Buffer de entrada Schmitt	PORTD<0> entrada de datos.
	SPP0	1	salida	Salida digital	SPP<0> salida de datos; tiene prioridad sobre el puerto de datos.
		1	entrada	Buffer de entrada TTL	SPP<0> entrada de datos.
RD1/SPP1	RD1	0	salida	Salida digital	LATD<1> salida de datos.
		1	entrada	Buffer de entrada Schmitt	PORTD<1> entrada de datos.
	SPP1	1	salida	Salida digital	SPP<1> salida de datos; tiene prioridad sobre el puerto de datos.
		1	entrada	Buffer de entrada TTL	SPP<1> entrada de datos.
RD2/SPP2	RD2	0	salida	Salida digital	LATD<2> salida de datos.
		1	entrada	Buffer de entrada Schmitt	PORTD<2> entrada de datos.
	SPP2	1	salida	Salida digital	SPP<2> salida de datos; tiene prioridad sobre el puerto de datos.
		1	entrada	Buffer de entrada TTL	SPP<2> entrada de datos.
RD3/SPP3	RD3	0	salida	Salida digital	LATD<3> salida de datos.
		1	entrada	Buffer de entrada Schmitt	PORTD<3> entrada de datos.
	SPP3	1	salida	Salida digital	SPP<3> salida de datos; tiene prioridad sobre el puerto de datos.
		1	entrada	Buffer de entrada TTL	SPP<3> entrada de datos.
RD4/SPP4	RD4	0	salida	Salida digital	LATD<4> salida de datos.
		1	entrada	Buffer de entrada Schmitt	PORTD<4> entrada de datos.
	SPP4	1	salida	Salida digital	SPP<4> salida de datos; tiene prioridad sobre el puerto de datos.
		1	entrada	Buffer de entrada TTL	SPP<4> entrada de datos.
RD5/SPP5/P1B	RD5	0	salida	Salida digital	LATD<5> salida de datos
		1	entrada	Buffer de entrada Schmitt	PORTD<5> entrada de datos
	SPP5	1	salida	Salida digital	SPP<5> salida de datos; tiene prioridad sobre el puerto de datos.
		1	entrada	Buffer de entrada TTL	SPP<5> entrada de datos.
	P1B	0	salida	Salida digital	Salida ECCP1 y PWM mejorado, canal B; tiene prioridad sobre el puerto de datos. ⁽¹⁾
RD6/SPP6/P1C	RD6	0	salida	Salida digital	LATD<6> salida de datos.
		1	entrada	Buffer de entrada Schmitt	PORTD<6> entrada de datos.
	SPP6	1	salida	Salida digital	SPP<6> salida de datos; tiene prioridad sobre el puerto de datos.
		1	entrada	Buffer de entrada TTL	SPP<6> entrada de datos.
	P1C	0	salida	Salida digital	Salida ECCP1 y PWM mejorado, canal C; tiene prioridad sobre el puerto de datos. ⁽¹⁾

RD7/SPP7/P1D	RD7	0	salida	Salida digital	LATD<7> salida de datos.
		1	entrada	Buffer de entrada Schmitt	PORTD<7> entrada de datos.
	SPP7	1	salida	Salida digital	SPP<7> salida de datos; tiene prioridad sobre el puerto de datos.
		1	entrada	Buffer de entrada TTL	SPP<7> entrada de datos.
	P1D	0	salida	Salida digital	Salida ECCP1 y PWM mejorado, canal D; tiene prioridad sobre el puerto de datos. ⁽¹⁾

Nota 1: Puede que configurarse como tri-estado en los flancos de bajada del PWM mejorado.

TABLA 10-8: RESUMEN DE REGISTROS ASOCIADOS AL PORTD

Nombre	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PORTD ⁽³⁾	RD7	RD6	RD5	RD4	RD3	RD2	RD1	RD0
LATD ⁽³⁾	LATD7	LATD6	LATD5	LATD4	LATD3	LATD2	LATD1	LATD0
TRISD ⁽³⁾	TRISD7	TRISD6	TRISD5	TRISD4	TRISD3	TRISD2	TRISD1	TRISD0
PORTE	RDPU ⁽³⁾	—	—	—	RE3 ^(1,2)	RE2 ⁽³⁾	RE1 ⁽³⁾	RE0 ⁽³⁾
CCP1CON	P1M1 ⁽³⁾	P1M0 ⁽³⁾	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0
SPPCON ⁽³⁾	—	—	—	—	—	—	SPPOWN	SPPEN

Leyenda: — = no implementado, se lee como ‘0’. Las celdas sombreadas no se utilizan para el PORTD.

Nota 1: Implementado solamente cuando la función Master Clean está desactivada (configuración del bit MCLRE=0).

2: RE3 es el único bit PORTE implementado en los dispositivos de 28 y 40/44pines. El resto de los bits se implementan cuando lo está el PORTE (es decir, en los dispositivos 40/44pines).

3: Estos registros y/o bits no están implementados en los dispositivos de 28pines.

10.5 REGISTROS PORTE, TRISE Y LATE

Dependiendo del PIC18F2455/2550/4455/4550 seleccionado, el PORTE se pone en ejecución en dos maneras diferentes.

Para los dispositivos 40/44pines, el PORTE es un puerto de 4bits de ancho. Tres pines (RE0/AN5/CK1SPP, RE1/AN6/CK2SPP y RE2/AN7/OESPP) se configuran individualmente como entradas o salidas. Estos pines tienen buffer de entrada tipo disparador Shmitt. Cuando se seleccionan como entrada analógica, éstos pines se leerán como ‘0’s.

El registro de dirección de los datos es TRISE. Activar el bit TRISE (=1) hará que el pin correspondiente de PORTE sea una entrada (es decir, pone el conductor correspondiente de la salida en un modo alta impedancia). Borrar un bit de TRISE (=0) hace que el pin correspondiente de PORTE sea una salida (es decir, pone el contenido del latch de salida en el pin seleccionado).

Además de los datos del puerto, el registro PORTE también contiene el bit de control RDPU (PORTE<7>); esto permite o inhabilita el pull-up del PORTD.

TRISE controla la dirección de los pines RE, incluso cuando se utilizan como entradas analógicas. El usuario debe cerciorarse de mantener los pines configurados como entradas cuando use las analógicas.

Nota: En un reset por subida de tensión, RE2:RE0 se configuran como entradas analógicas.

El registro del latch de datos (LATE) sólo es memoria mapeada. Las operaciones de lectura-modificación-escritura en el registro LATE lee y escribe el valor del latch de salida del PORTE.

El cuarto pin PORTE (MCLR*/VPP/RE3) sólo es un pin de entrada. Su operación se controlada con el bit de configuración MCLRE. Cuando se selecciona como puerto (MCLRE = 0), funciona como una entrada digital; como tal, no tiene bits TRIS o LAT asociados. Si no, funciona como entrada Master Clear del dispositivo. En cualquier configuración, RE3 también funciona como entrada de tensión de programación.

Nota: En un reset de subida de tensión, se activa RE3 como una entrada digital solamente si la función de Master Clear está desactivada.

EJEMPLO 10-5: INCIALIZACIÓN DE PORTE

CLRF PORTE	;Inicializar PORTE borrando los latch de salida de datos
CLRF LATE	;Otra método para borrar los latch de salida
MOVLW 0Ah	;Configura A/D como entradas digitales
MOVWF ADCON1	;
MOVLW 03h	;Valor usado para inicializar las direcciones de los datos
MOVLW 07h	;Desactiva los comparadores
MOVWF CMCON	;
MOVWF TRISE	;Activa RE<0> como entradas ;RE<1> como salidas ;RE<2> como entradas

10.5.1 PORTE EN LOS DISPOSITIVOS DE 28-PIN

En los dispositivos de 28pines, el PORTE sólo está disponible cuando el Master Clear está desactivado (MCLRE = 0). En éstos los casos, PORTE son de un solo bit, puerto de entrada de RE3 solamente. El pin funciona según lo descrito previamente.

REGISTRO 10-1: REGISTRO PORTE

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
RDPU ⁽³⁾	-	-	-	RE3 ^(1,2)	RE2 ⁽³⁾	RE1 ⁽³⁾	RE0 ⁽¹⁾
R/W-0	U-0	U-0	U-0	R/W-X	R/W-0	R/W-0	R/W-0
BIT 7	RDPU:			Bit activador del pull-up del PORD: 1= Pull-up activado 0= Pull-up desactivado			
BIT 3-0	RE3:RE0:			Bits de entrada de datos del PORTE ^(1,2,3)			

Nota 1: puesto en ejecución solamente cuando la funcionalidad clara principal es lisiada (configuración de MCLRE mordida = 0); si no, leer como '0'.

2: RE3 es el único bit de PORTE implementado todos los dispositivos. El resto de los bits sólo se implementan en los dispositivos de 40/44pines.

3: No implementado en los dispositivos de 28pines; se lee '0'.

TABLA 10-9: RESUMEN DE ENTRADAS-SALIDAS DEL PORTE

Pin	Función	Ajustes del TRIS	E/S	Tipo de E/S	Descripción
RE0/AN5/ CK1SPP	RE0	0	salida	Salida digital	LATE<0> salida de datos; no le afectan las entradas analógicas.
		1	entrada	Buffer de entrada Schmitt	PORTE<0> entrada de datos; desactivado cuando se activan las entradas analógicas.
	AN5	1	entrada	Señal analógica	Entrada A/D canal 5; configuración por defecto en un POR.
	CK1SPP	0	salida	Salida digital	Salida SPP reloj 1 (SPP activo).
RE1/AN6/ CK2SPP	RE1	0	salida	Salida digital	LATE<1> salida de datos; no le afectan las entradas analógicas.
		1	entrada	Buffer de entrada Schmitt	PORTE<1> entrada de datos; desactivado cuando se activan las entradas analógicas.
	AN6	1	entrada	Señal analógica	A/D input channel 6; configuración por defecto en un POR.
	CK2SPP	0	salida	Salida digital	Salida SPP reloj 2 (SPP activo).
RE2/AN7/ OESPP	RE2	0	salida	Salida digital	LATE<2> salida de datos; no le afectan las entradas analógicas.
		1	entrada	Buffer de entrada Schmitt	PORTE<2> entrada de datos; desactivado cuando se activan las entradas analógicas.
	AN7	1	entrada	Señal analógica	Entrada A/D canal 7; configuración por defecto en un POR.
	OESPP	0	salida	Salida digital	Salida active SPP (SPP activo).
MCLR*/VPP/ RE3	MCLR*	— ⁽¹⁾	entrada	Buffer de entrada Schmitt	Entrada externa Clear; activa cuando el bit MCLRE esté activo.
	VPP	— ⁽¹⁾	entrada	Señal analógica	Detección de alta tensión, utilizado en el modo de detección de entrada ICSP. Siempre disponible sin importar el modo del pin.
	RE3	— ⁽¹⁾	entrada	Buffer de entrada Schmitt	PORTE<3> entrada de datos; enabled when MCLRE Configuration bit is clear.

Nota 1: RE3 no tiene un bit correspondiente TRISE<3>. Este pin es siempre una entrada sin importar modo.

TABLA 10-10: EL RESUMEN DE REGISTROS ASOCIADOS AL PORTE

Nombre	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PORTE	RDPU ⁽³⁾	—	—	—	RE3 ^(1,2)	RE2 ⁽³⁾	RE1 ⁽³⁾	RE0 ⁽³⁾
LATE ⁽³⁾	—	—	—	—	—	LATE2	LATE1	LATE0
TRISE ⁽³⁾	—	—	—	—	—	TRISE2	TRISE1	TRISE0
ADCON1	—	—	VCFG1	VCFG0	PCFG3	PCFG2	PCFG1	PCFG0
CMCON	C2OUT	C1OUT	C2INV	C1INV	CIS	CM2	CM1	CM0
SPPCON ⁽³⁾	—	—	—	—	—	—	SPPOWN	SPPEN
SPPCFG ⁽³⁾	CLKCFG1	CLKCFG0	CSEN	CLK1EN	WS3	WS2	WS1	WS0

Leyenda: — = no implementado, se lee como ‘0’. Las celdas sombreadas no se utilizan para el PORTE.

Nota 1: Implementado sólo cuando la función Master Clear está desactivada (configuración del bit MCLRE=0).

2: RE3 es el único bit de PORTE implementado en todos los dispositivos. Se implementan el resto de los bits solamente cuando se pone en ejecución PORTE (es decir, los dispositivos 40/44-pin).

3: Estos registros o bits no están implementados en los dispositivos de 28 el pin.

11.0 MÓDULO TIMER0

El módulo Timer0 incorpora las siguientes características:

- Contador de 8bits ó 16bits seleccionable por software
- Registros legibles y esribibles
- 8 bits dedicados al prescaler programable por software
- Fuente de reloj (interno o externo) seleccionable
- Flanco seleccionable del reloj externo
- Interrupción por desbordamiento

El registro de T0CON controla todos aspectos de operación del módulo, incluyendo selección del prescaler. Es legible y esribible.

REGISTRO 11-1: T0CON: REGISTRO DE CONTROL TIMER0

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
TMR0ON	T08BIT	T0CS	T0SE	PSA	T0PS2	T0PS1	T0PS0
R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
BIT 7	TMROON:	Bit de control para activar el Timer0 1 = permite Timer0 0 = no permite Timer0					
BIT 6	T08BIT:	Bit de selección del Timer0 de 8-Bit/16-Bit 1 = Timer0 se configura como contador de 8 bits 0 = Timer0 se configura como contador de 16 bits					
BIT 5	T0CS:	Bit selector de la fuente del reloj Timer0 1 = reloj en el pin T0CKI 0 = reloj interno del ciclo de la instrucción (CLKO)					
BIT 4	T0SE:	Bit selector del flanco de la fuente Timer0 1 = incremento en el flanco de bajada del pin T0CKI 0 = incremento en el flanco de subida del pin T0CKI					
BIT 3	PSA:	Bit de la asignación del prescaler del Timer0 1 = el prescaler Timer0 no se asigna. 0 = se asigna el prescaler Timer0					
BIT 2-0	T0PS2:T0PS0:	Bits selectores del prescaler del Timer0 111 = valor de 1:256 110 = valor de 1:128 101 = valor de 1:64 100 = valor de 1:32 011 = valor de 1:16 010 = valor de 1:8 001 = valor de 1:4 000 = valor de 1:2					

11.1 OPERACIONES DEL TIMERO

Timer0 puede funcionar como un temporizador o contador; el modo se selecciona borrando el bit T0CS (T0CON<5>). En modo temporizador, el módulo se incrementa en cada pulso del reloj a menos que se seleccione un prescaler. Si el registro TMR0 se escribe, se inhibe el incremento en los dos ciclos siguientes de la instrucción. El usuario puede trabajar con esto escribiendo un valor al registro TMR0.

El modo contador se selecciona activando el bit TOCS (=1). En este modo, el Timer0 se incrementa en cualquier flanco de subida o de bajada del pin RA4/T0CKI. El flanco se determina por el pin TOSE (T0CON<4>); borrando este bit se selecciona el flanco de subida.

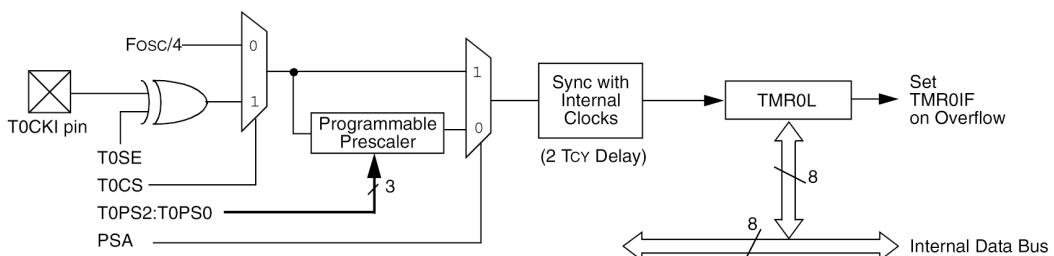
Se puede utilizar una fuente de reloj externa para controlar el Timer0; sin embargo, debe cumplir ciertos requisitos para asegurarse que el reloj externo se sincronice con reloj interno de fase (Tosc). Hay un retraso en la sincronización y el inicio del incremento del temporizador/contador.

11.2 TIMERO LECTURAS Y ESCRITURAS EN MODO DE 16BITS

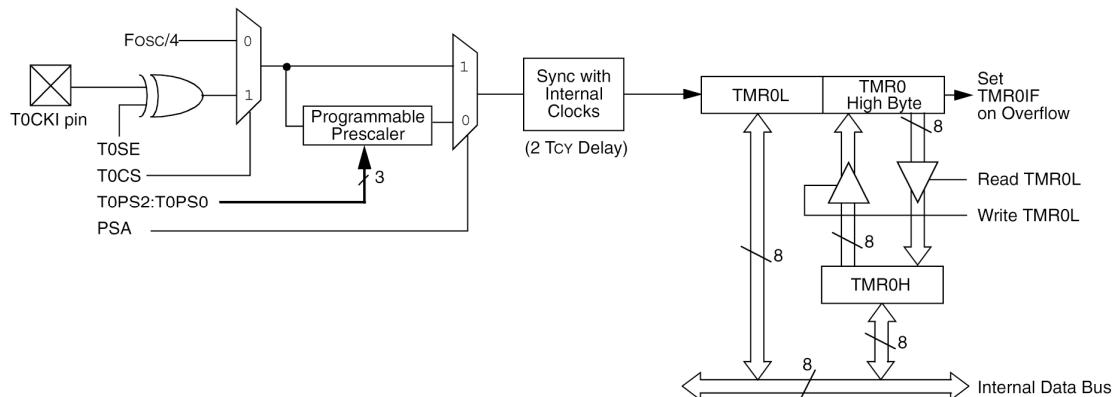
TMR0H no es el byte alto real del Timer0 en el modo de 16 bits. Realmente, es una versión protegida del verdadero byte alto del Timer0 que no se puede leer ni escribir directamente. TMR0H se actualiza con el contenido del byte alto de Timer0 durante la lectura del TMR0L. Esto proporciona la capacidad de leer los 16 bits de Timer0 sin tener que verificar que la lectura del byte alto y del bajo son correctas, debido a las lecturas sucesivas del byte bajo y del alto.

El byte alto del Timer0 se escribe a través del registro intermedio del TMR0H. El byte alto se actualiza con el contenido del TMR0H cuando se escribe el TMR0L. Esto permite que los 16 bits del Timer0 se actualicen inmediatamente.

FIGURA 11-1: DIAGRAMA DEL TIMERO (MODO 8-BIT)



Nota: En el reset, Timer0 se activa en modo de 8 bits con la entrada de reloj T0CKI y el prescaler al máximo.

FIGURA 11-2: DIAGRAMA DEL TIMER0 (MODO 16-BIT)

Nota: En el reset, Timer0 se activa en modo de 8 bits con la entrada de reloj T0CKI y el prescaler al máximo.

11.3 PRESCALER

Estos micros disponen de un contador de 8bit como prescaler para el módulo Timer0. El prescaler no se puede leer o escribir directamente; su valor se fija con los bits PSA y TOPS2:TOPS0 (T0CON<3: 0>), los cuáles determinan el divisor del prescaler y la asignación del prescaler.

Borrar el bit PSA asigna el prescaler al Módulo Timer0. Cuando se activa, los valores del prescaler van de 1:2 a 1:256, con incrementos de potencias de 2.

Cuando está activo el módulo Timer0, todas las instrucciones que escriben en él (ej., CLRF TMR0, MOVWF, TMR0, BSF TMR0, etc.) se borra el prescaler.

Nota: Al escribir el TMR0 cuando el prescaler está asignado al Timer0 lo despejará pero no cambiará su asignación.

11.3.1 CONMUTACIÓN EN LA ASIGNACIÓN DEL PRESCALER

La asignación del prescaler se controla completamente bajo software y se puede cambiar “en marcha” durante la ejecución del programa.

11.4 INTERRUPCIÓN TIMERO

Se genera la interrupción TMR0 cuando se desborda de FFh a 00h en modo de 8 bits, o de FFFFh a 0000h en modo de 16 bits. Este desbordamiento activa el flag TMR0IF. La interrupción se puede enmascarar al borrar el bit TMR0IE (INTCON<5>). Antes de volver a permitir la interrupción, el bit TMR0IF debe borrarse por software en la rutina de la interrupción.

Como el Timer0 se desconecta en el modo sleep, su interrupción no puede despertar el dispositivo.

TABLA 11-1: REGISTROS ASOCIADOS AL TIMER0

Nombre	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TMR0L	Timer0 Register Low Byte							
TMR0H	Timer0 Register High Byte							
INTCON	GIE/GIEH	PEIE/GIEL	TMR0IE	INT0IE	RBIE	TMROIF	INT0IF	RBIF
INTCON2	RBPU	INTEDG0	INTEDG1	INTEDG2	—	TMROIP	—	RBIP
TOCON	TMR0ON	T08BIT	T0CS	T0SE	PSA	T0PS2	T0PS1	T0PS0
TRISA	—	TRISA6 ⁽¹⁾	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0

Leyenda: - = las localizaciones no están implementadas. Las celdas sombreadas no se utilizan con Timer0.

Nota 1: Se configura RA6 como un puerto basado en varios modos primarios del oscilador. Cuando el puerto se desactiva, todos los bits asociados leen '0'.

12.0 MÓDULO TIMER1

El módulo temporizador/contador Timer1 incorpora las siguientes características:

- Temporizador/contador de 16bits seleccionable por software.
- Registros legibles y escribibles de 8 bits (TMR1H y TMR1L)
- Fuente seleccionable de reloj (interna o externa) con opciones del reloj o del oscilador interno Timer1 del dispositivo
- Interrupción por desbordamiento
- El módulo reset en un evento especial de disparo del CCP
- Flag de estado del reloj del dispositivo (T1RUN)

El módulo incorpora su propio oscilador de baja potencia para proporcionar una opción de registro adicional. El oscilador Timer1 se puede utilizar como fuente de baja potencia de reloj para el microcontrolador en las operaciones de ahorro de energía.

El Timer1 se puede utilizar para proporcionar la funcionalidad del reloj en tiempo real (RTC) en las aplicaciones con un mínimo de componentes externos y de código.

El Timer1 se controla con el registro T1CON. También contiene el bit de permiso del oscilador Timer1 (T1OSCEN). El Timer1 puede permitirse o inhabilitarse activando o borrando el bit de control, TMR1ON (T1CON<0>).

REGISTRO 12-1: T1CON: REGISTRO DE CONTROL DEL TIMER1

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
RD16	T1RUN	T1CKPS1	T1CKPS0	T1OSCEN	T1SYNC*	TMR1CS	TMR1ON
R/W-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
BIT 7	RD16:	Bit de permiso de la lectura/escritura en el modo de 16-Bit 1 = permite la lectura/escritura del Timer1 en una operación de 16bits 0 = permite lectura/escritura del Timer1 en dos operaciones de 8bits					
BIT 6	T1RUN:	Bit de estado del reloj del sistema Timer1 1 = reloj del dispositivo se deriva del oscilador Timer1 0 = reloj del dispositivo se deriva de otra fuente					
BIT 5-4	T1CKPS1: T1CKPS0:	Bits de selección del prescaler del Timer1 11 = valor de 1:8 10 = valor de 1:4 01 = valor de 1:2 00 = valor de 1:1					
BIT 3	T1OSCEN:	Bit de permiso del oscilador Timer1 1 = se permite el oscilador Timer1 0 = se desconecta el oscilador Timer1 El inversor del oscilador y la resistencia de regeneración se apagan para eliminar la energía de drenaje.					
BIT 2	T1SYNC*:	Bit de selección de la sincronización externa con el Timer1 Cuando TMR1CS = 1: 1 = no sincroniza la entrada de reloj externa 0 = sincroniza la entrada de reloj externa Cuando TMR1CS = 0: Se no hace caso este bit. Timer1 utiliza el reloj interno cuando TMR1CS = 0.					

BIT 1	TMR1CS:	Bit de selección de la fuente del reloj del Timer1 1 = reloj externo en el pin RC0/T1OSO/T13CKI (en el flanco de subida) 0 = reloj interno ($F_{osc}/4$)
BIT 0	TMR1ON:	Bit de activación del Timer1 1 = permite Timer1 0 = para el Timer1

12.1 OPERACIONES DEL TIMER1

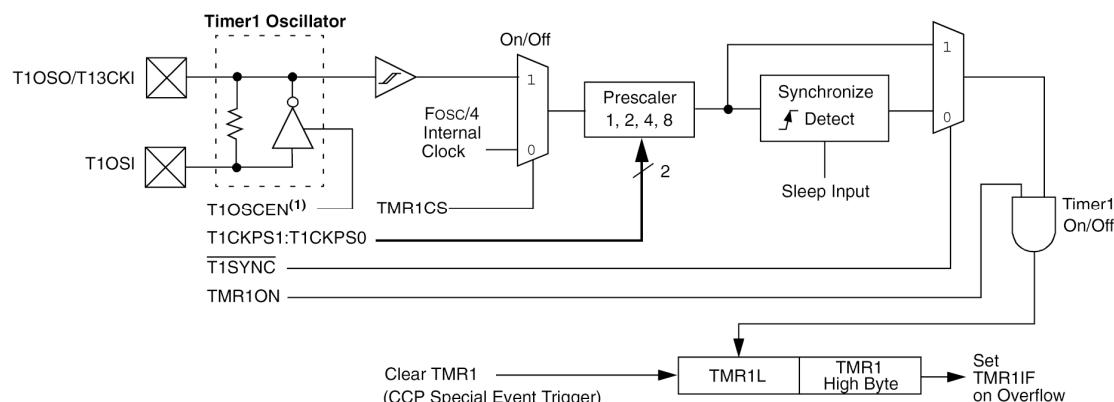
Timer1 puede funcionar en uno de estos modos:

- Temporizador
- Contador síncrono
- Contador asincrónico

El modo de funcionamiento se determina con el bit de selección TMR1CS (T1CON<1>). Cuando se borra TMR1CS (=0), incrementos del Timer1 en cada instrucción interna ($F_{osc}/4$). Cuando se activa el bit, los incrementos Timer1 en cada flanco de subida de la entrada del reloj externo Timer1 o del oscilador Timer1, si está activo.

Cuando se permite el Timer1, los pines RC1/T1OSI/UOE y RC0/T1OSO/T13CKI se convierten en entradas. Esto significa que se ignoran los valores de TRISC<1:0> y los pines se leen como ‘0’.

FIGURA 12-1: DIAGRAMA DEL TIMER1



Nota 1: Cuando el bit de permiso, T1OSCEN, se borra, el inversor del oscilador y la resistencia de regeneración se apagan para eliminar la energía de drenaje.

12.2 MODO DE LECTURA/ESCRITURA DE 16-BIT DEL TIMER1

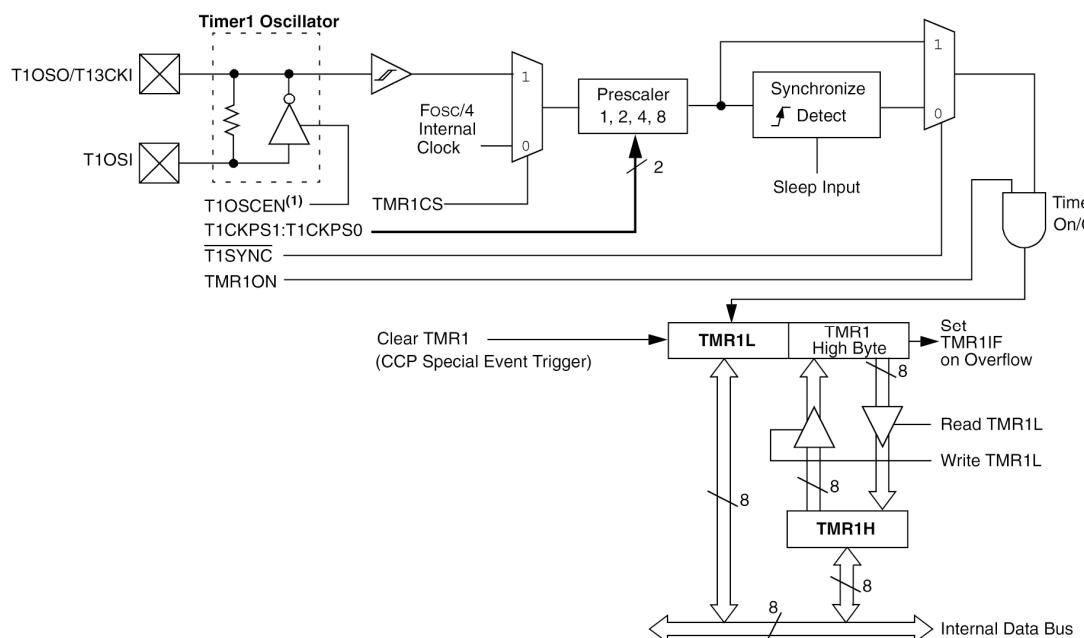
Timer1 se puede configurar para lecturas/escrituras de 16bits. Cuando el bit de control RD16 (T1CON<7>) se activa, la dirección del TMR1H se mapea con un

registro del almacenamiento intermedio del byte alto del Timer1. Al leer el TMR1L cargará el contenido del byte alto del Timer1 en el byte alto del Buffer del Timer1. Esto proporciona al usuario la capacidad de leer exactamente los 16 bits de Timer1 sin tener que determinar si lee el byte alto, seguido del byte bajo, se volverá inválido debido a las lecturas redundantes.

Al escribir el byte alto del Timer1 debe pasar a través del registro de almacenamiento intermedio de TMR1H. El byte alto del Timer1 se actualiza con el contenido de TMR1H cuando se escribe el TMR1L. Esto permite que el usuario escriba los 16 bits a los bytes altos y bajos del Timer1 inmediatamente.

No se puede leer o escribir el byte alto de Timer1 directamente en este modo. Toda lectura y escritura debe hacerse a través del registro de almacenamiento intermedio. Al escribir el TMR1H no se borra el prescaler del Timer1. El prescaler se borra solamente escribiendo el TMR1L.

FIGURA 12-2: DIAGRAMA DEL TIMER1 (MODO DE LECTURA/ESCRITURA DE 16-BIT)



Nota 1: Cuando el bit de permiso, T1OSCEN, se borra, el inversor del oscilador y la resistencia de regeneración se apagan para eliminar la energía de drenaje.

12.3 OSCILADOR TIMER1

Se incorpora un circuito oscilador de XTAL en el chip entre los pinos T1OSI (entrada) y T1OSO (amplificador de salida). Se permite el oscilador Timer1 activando el bit, T1OSCEN (T1CON<3>). El oscilador es un circuito de baja potencia para cristales de 32kHz. Continua funcionando en cualquier modo de ahorro energía.

El usuario debe proporcionar por software el retraso para asegurar el inicio apropiado del oscilador Timer1.

FIGURA 12-3: COMPONENTES EXTERNOS DEL OSCILADOR DE TIMER1

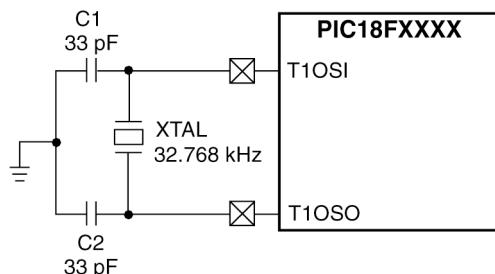


TABLA 12-1: SELECCIÓN DEL CONDENSADOR DEL OSCILADOR DEL TEMPORIZADOR

Osc	Freq	C1	C2
LP	32 kHz	27 pF	27 pF

12.3.1 USAR TIMER1 COMO FUENTE DE RELOJ

El oscilador Timer1 está también disponible como fuente del reloj en los modos de ahorro de energía. Poniendo los bits, SCS1:SCS0 (OSCCON<1:0>), a ‘01’, el dispositivo cambia al modo SEC_RUN. La CPU y los periférico se controlan con el oscilador Timer1. Si el bit IDLEN (OSCCON<7>) se borra y se ejecuta la instrucción SLEEP, el dispositivo entra en el modo SEC_IDLE.

Siempre que el oscilador Timer1 proporcione la fuente de reloj, el flag de estado del reloj del sistema Timer1, T1RUN (T1CON<6>), se activa. Esto se puede utilizar para determinar el modo que controla el dispositivo. Puede indicar también la fuente del reloj que utiliza actualmente el monitor a prueba de fallos. Si se permite el monitor del reloj y el oscilador del Timer1 falla mientras que proporciona la señal de reloj, preguntando al bit T1RUN sabremos si el reloj lo proporciona el oscilador Timer1 u otra fuente.

12.3.2 OPCIÓN DE BAJA POTENCIA TIMER1

El oscilador Timer1 puede funcionar en dos niveles distintos de consumo de energía basado en la configuración de dispositivo. Cuando se setea el bit de configuración LPT1OSC, el oscilador Timer1 funciona en un modo de baja potencia. Cuando LPT1OSC no se activa, el Timer1 funciona en un nivel de energía más alto. El consumo de energía para un modo particular es relativamente constante, sin importar el

modo del dispositivo. La configuración por defecto del Timer1 es el modo de energía más alto.

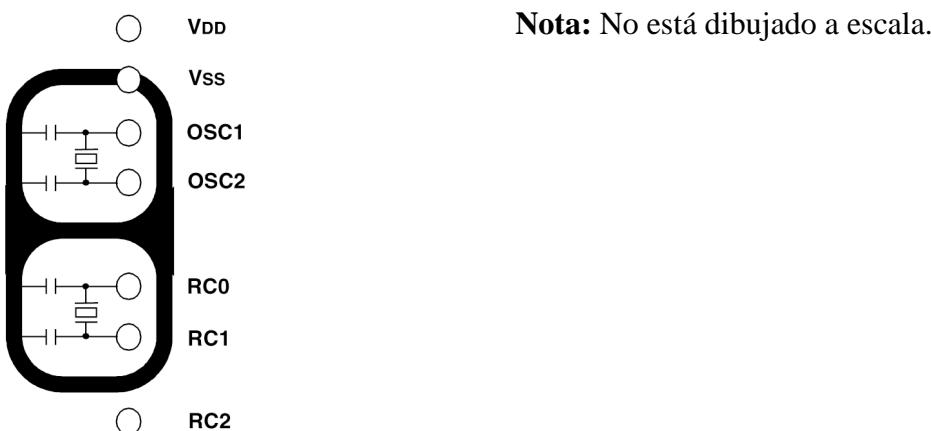
Como el modo de baja potencia Timer1 tiende a ser más sensible a las interferencias, los ambientes con mucho de ruido pueden causar una cierta inestabilidad del oscilador. La opción de baja potencia es, por tanto, más adecuada en aplicaciones de poco ruido donde la conservación de energía es una consideración de diseño importante.

12.3.3 CONSIDERACIONES DEL LAYOUT DEL OSCILADOR TIMER1

El circuito del oscilador Timer1 crea una energía muy pequeña durante su operación. Debido a la naturaleza de baja potencia del oscilador, puede provocarse interferencias rápidamente. El circuito del oscilador debe localizarse lo más cercano posible al microcontrolador. No puede haber circuitos que pasen por dentro de los límites del oscilador del circuito con excepción de V_{SS} o de V_{DD}.

Si hay que poner un circuito de alta velocidad cerca del oscilador (por ejemplo el pin CCP1, o el oscilador primario usando el pino OSC2), hay que poner un anillo de puesta a tierra protector alrededor del circuito del oscilador; puede ser útil cuando está utilizado en PWB de una cara o además de un plano de tierra.

FIGURA 12-4: CIRCUITO DEL OSCILADOR DE PUESTA A TIERRA



12.4 INTERRUPCIÓN TIMER1

El par de registros TMR1 (TMR1H:TMR1L) incrementan de 0000h a FFFFh y vuelve a 0000h. La interrupción Timer1, si está permitida, se genera en el desbordamiento que activa el flag de la interrupción, TMR1IF (PIR1<0>). Esta interrupción se puede permitir o inhabilitar activando o borrando el bit, TMR1IE (PIE1<0>).

12.5 RESET DEL TIMER1 USANDO EL DISPARADOR ESPECIAL DEL ACONTECIMIENTO CCP

Si cualquiera de los módulos CCP se configuran como un modo de comparación para generar un disparo especial (CCP1M3:CCP1M0 o CCP2M3:CCP2M0 = 1011), esta señal reseteará el Timer1. El disparo de CCP2 también comenzará una conversión A/D, si es el módulo A/D está permitido.

El módulo se debe configurar como un temporizador o un contador síncrono para aprovechar esta característica. Cuando se está utilizando de esta manera, el par de registros CCPRH:CCPRL se convierten en un registro del período del Timer1.

Si Timer1 está funcionando en modo contador asíncrono, esta operación de reset puede que no funcione.

En caso que al escribir el Timer1 coincida con un evento de disparo especial, la operación de escritura tomará precedencia.

Nota: Los acontecimientos de disparo especiales del módulo CCP2 no activarán el flag de interrupción TMR1IF (PIR1<0>).

12.6 USAR TIMER1 COMO RELOJ DE TIEMPO REAL (RTC)

Añadiendo un oscilador externo al Timer1 da a los usuarios la opción de incluir aplicaciones en tiempo real. Esto se logra con un cristal de reloj para proporcionar una frecuencia exacta de base y varias líneas de código de la aplicación para calcular el tiempo. Al funcionar en modo sleep y con batería o súper condensador como fuente de energía, puede eliminar la necesidad de un RTC separado y una batería de reserva.

La rutina del código, RTCisr, demuestra un método simple para incrementar un contador en intervalos de un segundo usando una rutina de interrupción. Incrementando el par de registros TMR1 para desbordarlo y provocar la llamada de la interrupción, que incrementa el segundo contador en uno. Los contadores adicionales para minutos y horas se incrementan al desbordar el contador anterior.

Puesto que el par del registro es de 16 bits de ancho, contando hasta que se desborde el registro directamente con un reloj de 32,768kHz tardaría 2s. Para forzar el desbordamiento en intervalos de un segundo, es necesario cargar el temporizador. El método más simple es fijar el MSb TMR1H con una instrucción BSF. Observar que el registro TMR1L nunca se carga o se altera; el hacerlo, puede introducir un error acumulativo sobre muchos ciclos.

Para que este método sea exacto, el Timer1 debe funcionar en modo asíncrono y la interrupción del desbordamiento Timer1 debe estar permitida (PIE1<0>=1) según las indicaciones de rutina, RTCinit. El oscilador Timer1 tiene que estar permitido y funcionando siempre.

EJEMPLO 12-1: PONER UN RELOJ EN EJECUCIÓN EN TIEMPO REAL USANDO LA INTERRUPCIÓN TIMER1

RTCinit

MOVLW	80h	;Carga del par de registros TMR1
MOVWF	TMR1H	;para desbordamiento en 1 segundo
CLRF	TMR1L	

```

MOVlw b'00001111' ;Configurar el reloj externo,
MOVwf T1osc ;Operación asíncrono, oscilador externo
CLRF Secs ;Inicializar los registros para la puntualidad
CLRF mins
MOVLW d'12'
MOVWF hours
BSF PIE1,TMR1IE;Permitir la interrupción Timer1
RETURN

RTCisr
BSF TMR1H,7 ;Carga para desbordamiento de 1s
BCF PIR1,TMR1IF;Flag de interrupción borrado
INCF secs,F ;Incremento de segundos
MOVLW d'59' ;¿Transcurrieron 60 segundos?
CPFGT secs
RETURN ;No
CLRF secs ;Si, Borrar segundos
INCF mins,F ;Incremento minutos
MOVLW d' 59' ;¿Transcurrieron 60 minutos?
CPFGT mins
RETURN ;No
CLRF mins ;Si, borrado de minutos
INCF hours,F ;Incremento de horas
MOVLW d' 23' ;¿Transcurrieron 24 horas?
CPFGT hours
RETURN ;No
MOVLW d' 01' ;Si, Reset de las horas a 1
MOVWF hours
RETURN ;Hecho

```

TABLA 12-2: LOS REGISTROS ASOCIADOS AL TIMER1 COMO TEMPORIZADOR/COTADOR

Nombre	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTCON	GIE/GIEH	PEIE/GIEL	TMROIE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF
PIR1	SPPIF ⁽¹⁾	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF
PIE1	SPPIE ⁽¹⁾	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE
IPR1	SPPIP ⁽¹⁾	ADIP	RCIP	TXIP	SSPIP	CCP1IP	TMR2IP	TMR1IP
TMR1L	Timer1 Register Low Byte							
TMR1H	Timer1 Register High Byte							
T1CON	RD16	T1RUN	T1CKPS1	T1CKPS0	T1OSCEN	T1SYNC*	TMR1CS	TMR1ON

Leyenda: - = las localizaciones no están implementadas. Las celdas sombreadas no se utilizan con Timer0.

Nota 1: Estos bits no están implementados en los dispositivos de 28pins; mantenerlos siempre borrados.

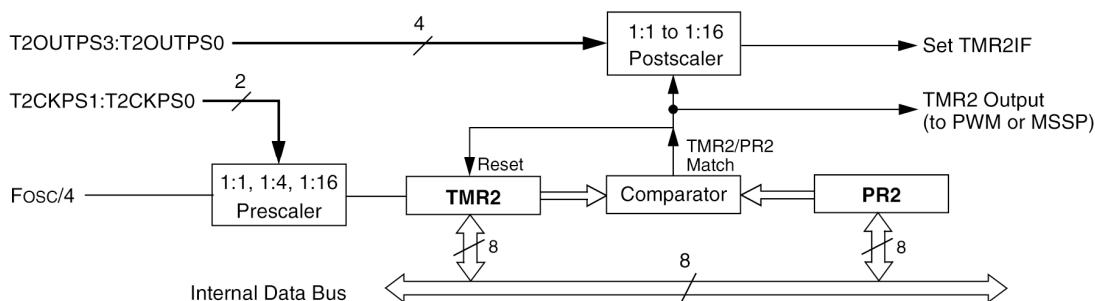
13.0 MÓDULO TIMER2

El temporizador Timer2 incorpora las siguientes características:

- Registros temporizador y período de 8bit (TMR2 y PR2, respectivamente)
- Legible y escribible (ambos registros)
- Prescaler programable por software (1: 1, 1:4 y 1:16)
- Postscaler programable por software (1: 1 a 1:16)
- Interrupción en TMR2 unida al PR2
- Uso opcional como reloj del cambio para el módulo MSSP

El módulo se controla a través del registro T2CON que permite o inhabilita el temporizador y configura el prescaler y el postscaler. El Timer2 se puede apagar borrando el bit de control, TMR2ON (T2CON<2>), para reducir al mínimo el consumo de energía.

FIGURA 13-1: DIAGRAMA DE BLOQUES DEL TIMER2



13.1 OPERACIONES DEL TIMER2

En una operación normal, TMR2 se incrementa de 00h en cada pulso del reloj (Fosc/4). Un contador/prescaler de 2bits en la entrada de reloj da la entrada directa, con las opciones de prescaler divisor por 4 y divisor por 16. Éstos se eligen con los bits, T2CKPS1:T2CKPS0 (T2CON<1:0>). El valor de TMR2 se compara al del registro del período, PR2, en cada ciclo de reloj. Cuando los dos valores son iguales, el comparador genera una señal de sincronismo como temporizador de salida. Esta señal también resetea el valor de TMR2 a 00h en el ciclo siguiente y controla el contador/postscaler de salida.

Los registros TMR2 y PR2 son directamente legibles y escriptibles. El registro TMR2 se borra en cualquier reset del dispositivo, mientras que el registro PR2 se inicializa en FFh. Los contadores del prescaler y del postscaler se borran con los siguientes acontecimientos:

- Al escribir en el registro TMR2
- Al escribir en el registro T2CON
- Cualquier reset del dispositivo

El TMR2 no se borra cuando se escribe el T2CON.

REGISTRO 13-1: T2CON: REGISTRO DE CONTROL TIMER2

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
-	T2OUTPS3	T2OUTPS2	T2OUTPS1	T2OUTPS0	TMR2ON	T2CKPS1	T2CKPS0
U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
BIT 6-3	T2OUTPS3:T2OUTPS0:	Bits selectores del Postscaler de la salida					
		0000 = 1:1 Postscaler					
		0001 = 1:2 Postscaler					
		.					
		1111 = 1:16 Postscaler					
BIT 2	TMR2ON:	Bit de activación del Timer2					
		1 = Timer2 está encendido					
		0 = Timer2 está apagado					
BIT 1-0	T2CKPS1:T2CKPS0:	Bits selectores del Prescaler del reloj					
		00 = Prescaler de 1					
		01 = Prescaler de 4					
		1x = Prescaler de 16					

13.2 INTERRUPCIÓN TIMER2

Timer2 puede generar una interrupción opcional en el dispositivo. La señal de salida Timer2 (TMR2 unida con PR2) proporciona la entrada para el contador/postscaler de salida de 4 bits. Este contador genera el flag de interrupción del TMR2 que está en TMR2IF (PIR1<1>). La interrupción se permite activando el bit de la interrupción, TMR2IE (PIE1<1>). Una gama de 16 postscaler (de 1:1 a 1:16 inclusivo) puede seleccionarse con los bits de control, T2OUTPS3:T2OUTPS0 (T2CON<6:3>).

13.3 SALIDA DEL TMR2

La salida sin escala del TMR2 está disponible sobre todo para los módulos de CCP, donde se utiliza como base de tiempos de operaciones en modo PWM.

El Timer2 se puede utilizar opcionalmente como fuente del reloj para el módulo MSSP que funciona en modo SPI.

TABLA 13-1: LOS REGISTROS ASOCIADOS AL TIMER2 TRABAJANDO COMO TEMPORIZADOR/CONTADOR

Nombre	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTCON	GIE/GIEH	PEIE/GIEL	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF
PIR1	SPPIF ⁽¹⁾	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF
PIE1	SPPIE ⁽¹⁾	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE
IPR1	SPPIP ⁽¹⁾	ADIP	RCIP	TXIP	SSPIP	CCP1IP	TMR2IP	TMR1IP
TMR2	Timer2 Register							
T2CON	—	T2OUTPS3	T2OUTPS2	T2OUTPS1	T2OUTPS0	TMR2ON	T2CKPS1	T2CKPS0
PR2	Timer2 Period Register							

Leyenda: - = no implementado, se lee como '0'. Las celdas sombreadas no las utiliza por módulo Timer2.

Nota 1: Estos bits no están implementados en los dispositivos de 28pines; mantenerlos siempre borrados.

14.0 MÓDULO TIMER3

El módulo Timer3 temporizador/contador incorpora las siguientes características:

- Temporizador de 16bits o contador seleccionable por software
- Registros legibles y escribibles de 8bits (TMR3H y TMR3L)
- Fuente seleccionable de reloj (interna o externa) con opciones internas del reloj o del oscilador Timer1
- Interrupción del dispositivo al desbordarse
- El módulo de reset en un disparo especial del CCP

El módulo Timer3 es controlado con el registro T3CON. También selecciona la fuente del reloj opciones para los módulos de CCP.

REGISTRO 14-1: T3CON: REGISTRO DE CONTROL DEL TIMER3

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
RD16	T3CCP2	T3CKPS1	T3CKPS0	T3CCP1	T3SYNC*	TMR3CS	TMR3ON
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
BIT 7	RD16:						
		Bit de permiso del modo de lectura/escritura de 16Bits					
		1 = permite la lectura/escritura del Timer3 en una					
		operación de 16 bits					
		0 = permite la lectura/escritura del Timer3 en dos					
		operaciones de 8 bits					
BIT 6,3	T3CCP2:T3CCP1:						
		Bits de permiso del Timer3 y Timer1 al CCPx					
		1x = Timer3 es la fuente del reloj para los módulos CCP					
		01 = Timer3 es la fuente del reloj para CCP2; y el					
		Timer1 para CCP1					
		00 = Timer1 es la fuente del reloj para ambos CCP					
BIT 5-4	T3CKPS1:T3CKPS0:						
		Bits selectores del Prescaler del reloj de entrada					
		11 = Prescaler de 1:8					
		10 = Prescaler de 1:4					
		01 = Prescaler de 1:2					
		00 = Prescaler de 1:1					
BIT 2	T3SYNC*:						
		Bit de control de la sincronización del reloj externo					
		(No se puede usar cuando el reloj del dispositivo					
		es Timer1/Timer3.)					
		Cuando TMR3CS = 1:					
		1 = no sincroniza la entrada de reloj externa					
		0 = sincroniza la entrada de reloj externa					
		Cuando TMR3CS = 0:					
		Se ignora este bit. Timer3 utiliza el reloj.					
BIT 1	TMR3CS:						
		Bit selector de la fuente de reloj del Timer3					
		1 = entrada de reloj externa del oscilador Timer1 o de					
		T13CKI (en el flanco de subida)					
		0 = reloj interno ($F_{OSC}/4$)					
BIT 0	TMR3ON:						
		Bit de encendido del Timer3					
		1 = permite Timer3					
		0 = parado Timer3					

14.1 OPERACIONES DEL TIMER3

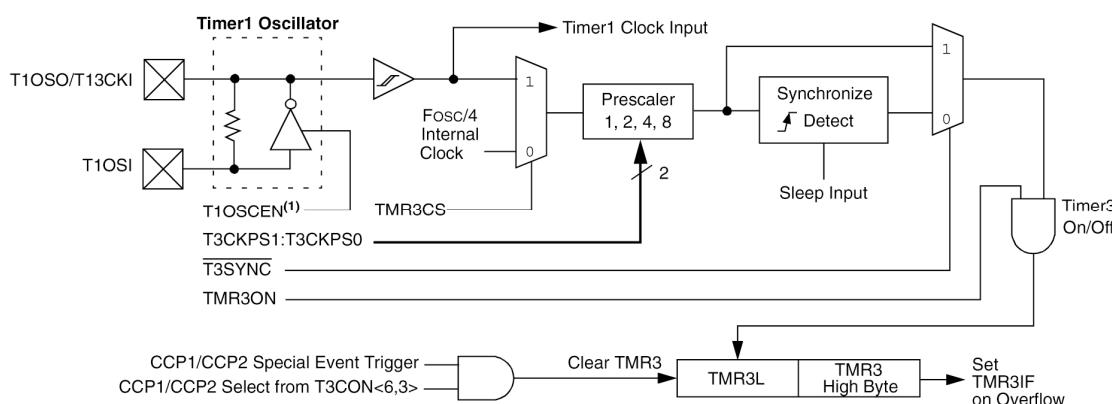
El Timer3 puede funcionar en uno de estos modos:

- Temporizador
- Contador síncrono
- Contador asincrónico

El modo de funcionamiento se determina con el bit, TMR3CS (T3CON<1>). Cuando TMR3CS está borrado (=0), el Timer3 incrementa en cada instrucción interna ciclo ($F_{osc}/4$). Cuando se activa el bit, los incrementos de Timer3 en cada flanco de subida de la entrada de reloj externa Timer1 o el oscilador Timer1, si está permitido.

Como el Timer1, los pines RC1/T1OSI/UOE RC0/T1OSO/T13CKI se convierten en entradas cuando el oscilador Timer1 se permite. Esto significa se ignoran los valores de TRISC<1:0> y los pines se leen '0'.

FIGURA 14-1: DIAGRAMA DE BLOQUES DEL TIMER3



Nota 1: Cuando el bit de permiso, T1OSCEN, se borra, el inversor y la resistencia de regeneración se apagan para eliminar la energía de drenaje.

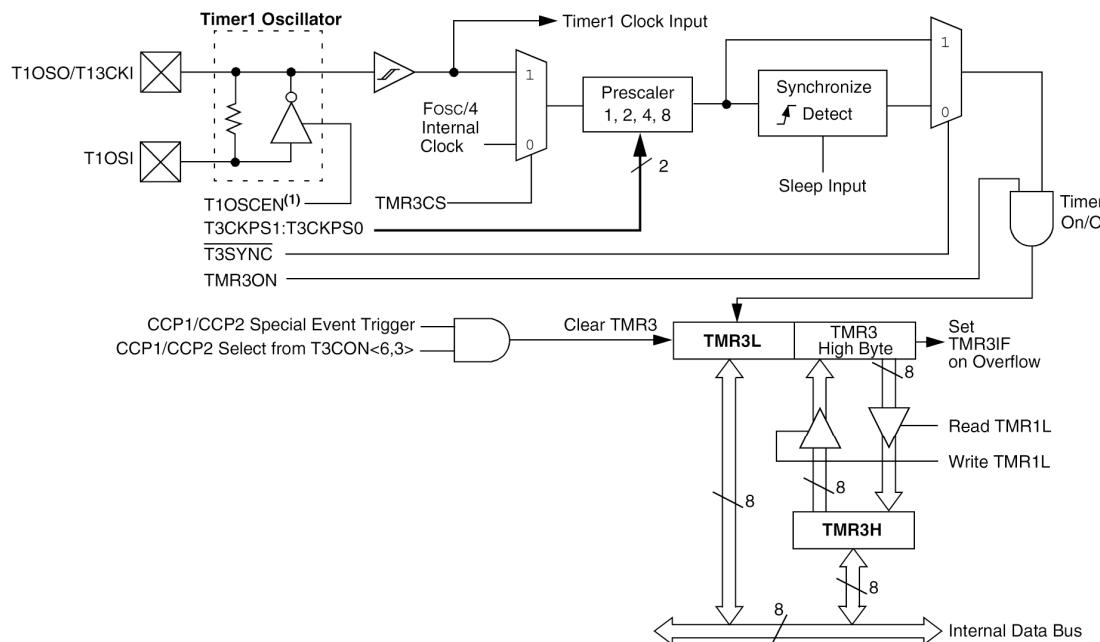
14.2 MODO DE LECTURA/GRABACIÓN DE 16BIT

Timer3 se puede configurar para que lea y escriba 16bits. Cuando el bit RD16 (T3CON<7>) se activa, la dirección del TMR3H se mapea con un registro de almacenamiento intermedio con el byte alto de Timer3. Al leer TMR3L cargará el contenido del byte alto de Timer3 en el byte alto del registro de almacenamiento intermedio. Esto proporciona al usuario la capacidad de leer los 16 bits sin tener que determinar si es el byte alto, seguido por el byte bajo, las lecturas redundantes provocan que sea inválido.

Al escribir el byte alto del Timer3 debe pasar a través del registro de almacenamiento intermedio de TMR3H. El byte alto del Timer3 se actualiza con el contenido de TMR3H cuando se escribe el TMR3L. Esto permite que el usuario escriba los 16 bits a los bytes altos y bajos de Timer3 inmediatamente.

No se puede leer o escribir el byte alto de Timer3 directamente en este modo. Toda lectura y escritura debe hacerse a través del registro de almacenamiento intermedio. Al escribir el TMR3H no se borra el prescaler del Timer3. El prescaler se borra solamente escribiendo el TMR3L.

FIGURA 12-2: DIAGRAMA DEL TIMER3 (MODO DE LECTURA/ESCRITURA DE 16-BIT)



Nota 1: Cuando el bit de permiso, T1OSCEN, se borra, el inversor y la resistencia de regeneración se apagan para eliminar la energía de drenaje.

14.3 USAR EL OSCILADOR TIMER1 COMO FUENTE DE RELOJ TIMER3

El oscilador interno Timer1 se puede utilizar como el reloj fuente del Timer3. El oscilador Timer1 se activa seteando el bit T1OSCEN (T1CON<3>). Para utilizarlo como fuente de reloj del Timer3, el bit TMR3CS debe estar activado. Esto configura el Timer3 para que incremente en cada flanco de subida de la fuente del oscilador.

14.4 INTERRUPCIÓN TIMER3

El par de registros TMR3 (TMR3H:TMR3L) incrementa de 0000h a FFFFh y se desborda a 0000h. La interrupción Timer3, si está permitida, se genera en el desbordamiento y activa en el flag de interrupción, TMR3IF (PIR2<1>). Esta interrupción puede permitirse o inhabilitarse activando o borrando el bit de interrupción del Timer3, TMR3IE (PIE2<1>).

14.5 RESET DEL TIMER3 USANDO EL DISPARADOR ESPECIAL DE ACONTECIMIENTOS DEL CCP

Si el módulo CCP2 se configura para generar el disparador especial de acontecimiento en el modo de comparación (CCP2M3:CCP2M0=1011), esta señal reseteará el Timer3. También comenzará una conversión A/D si el módulo A/D está activado.

El módulo se debe configurar como temporizador o contador síncrono para aprovecharse de esta característica. Cuando se está utilizando este modo, el par de registros CCPR2H:CCPR2L se convierten en un registro de período del Timer3.

Si Timer3 está funcionando en modo contador asíncrono, el reset puede que no funcione.

En caso que coincida una escritura en el Timer3 con el disparo especial de un módulo de CCP, la escritura tiene preferencia.

Nota: Los acontecimientos de disparo especiales del módulo CCP2 no activarán el flag TMR3IF (PIR2<1>).

TABLA 14-1: LOS REGISTROS ASOCIADOS AL TIMER3 TRABAJANDO COMO TEMPORIZADOR/CONTADOR

Nombre	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTCON	GIE/GIEH	PEIE/GIEL	TMROIE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF
PIR2	OSCFIF	CMIF	USBIF	EEIF	BCLIF	HLVDIF	TMR3IF	CCP2IF
PIE2	OSCFIE	CMIE	USBIE	EEIE	BCLIE	HLVDIE	TMR3IE	CCP2IE
IPR2	OSCFIP	CMIP	USBIP	EEIP	BCLIP	HLVDIP	TMR3IP	CCP2IP
TMR3L	Timer3 Register Low Byte							
TMR3H	Timer3 Register High Byte							
T1CON	RD16	T1RUN	T1CKPS1	T1CKPS0	T1OSCEN	T1SYNC*	TMR1CS	TMR1ON
T3CON	RD16	T3CCP2	T3CKPS1	T3CKPS0	T3CCP1	T3SYNC*	TMR3CS	TMR3ON

Leyenda: - = las localizaciones no están implementadas. Las celdas sombreadas no se utilizan con Timer3.

15.0 MÓDULOS CAPTURA/COMPARACIÓN/PWM (CCP)

Los dispositivos PIC18F2455/2550/4455/4550 tienen dos módulos CCP (Captura/Comparación/PWM). Cada módulo contiene un registro de 16 bits, que puede funcionar como registro de captura de 16bit, registro de comparación de 16bit o un registro PWM maestro/esclavo.

En dispositivos de 28pines, los dos módulos estándares CCP (CCP1 y CCP2) funcionan según lo descrito en este capítulo. En los dispositivos 40/44pines, CCP1 se implementa como módulo CCP mejorado, con los modos de captura y comparación estándar y PWM mejorado.

Las operaciones captura y comparación descritas en este capítulo se aplican a los módulos CCP estándares y al mejorado.

REGISTRO 15-1: CCPxCON: REGISTRO DE CONTROL ESTÁNDAR DE CCPx

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0						
-	-	DCxB1	DCxB0	CCPxM3	CCPxM2	CCPxM1	CCPxM0						
U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0						
BIT 5-4	DCxB1:DCxB0:	Bit 1 y 0 del Duty Cicle PWM del módulo CCPx											
Modo de captura: No utilizado.													
Modo comparación: No utilizado.													
Modo PWM:													
Estos bits son los dos LSbs (bit 1 y 0 bit) de los 10bits del PWM. Los ocho MSbs se encuentran en CCPR1L.													
BIT 3-0	CCPxM3:CCPxM0:	Bits selectores del modo del módulo CCPx											
0000= CCPX inhabilitado (CCPx en los resets)													
0001= reservado													
0010= modo comparación: salida activa al igualarse (se activa el bit CCPxIF)													
0011= reservado													
0100= Modo captura: cada flanco de bajada													
0101= Modo captura: cada flanco de subida													
0110= Modo captura: cada 4º flanco de subida													
0111= Modo captura: cada 16º flanco de subida													
1000= modo comparación: inicializa el pin bajo de CCPx; al igualar, se fuerza el pin alto de CCPx (Se activa el bit CCPxIF)													
1001= modo comparación: inicializa el pin alto de CCPx; al igualar, se fuerza el pin bajo de CCPx (Se activa el bit CCPxIF)													
1010= modo comparación: genera interrupción por software al igualar (se activa el bit CCPxIF, el pin CCPx refleja el estado de la entrada-salida)													
1011= modo comparación: accionar el acontecimiento especial, resetea el contador de tiempo, comienza la conversión A/D al igualar CCP2 (Se activa el bit de CCPxIF)													
11xx = modo PWM													

15.1 CONFIGURACIÓN DEL MÓDULO CCP

A cada módulo Captura/Comparación/PWM se le asocia un registro de control (genéricamente, CCPxCON) y registro de datos (CCPRx). El registro de datos, alternadamente, abarca dos registros de 8bit: CCPRxL (byte bajo) y CCPRxH (byte alto). Todos los registros son legibles y escribibles.

15.1.1 RECURSOS DE LOS MÓDULOS CCP Y DEL TEMPORIZADOR

Los módulos CCP utilizan los temporizadores 1, 2 ó 3, dependiendo del modo seleccionado. Timer1 y Timer3 están disponibles en los módulos de los modos captura o comparación, mientras que Timer2 está disponible para los módulos en modo PWM.

La asignación de un temporizador particular a un módulo se determina con el bit de permiso del temporizador a CCP del registro T3CON. Ambos módulos pueden activarse en cualquier momento y pueden compartir el mismo recurso del temporizador si se configuran para funcionar en el mismo modo (la captura/comparación o PWM) al mismo tiempo. La captura no estará disponible con el Timer1 en modo asíncrono.

TABLA 15-1: RECURSOS DE LOS MODOS CCP Y TEMPORIZADOR

Modo CCP/ECCP	Recurso temporizador
Captura	Timer1 o Timer3
Comparación	Timer1 o Timer3
PWM	Timer2

TABLA 15-2: INTERACCIONES ENTRE CCP1 Y CCP2 PARA LOS RECURSOS DEL TEMPORIZADOR

Modo CCP1	Modo CCP2	Interacción
Captura	Captura	Cada modulo puede usar TMR1 o TMR3 como base de tiempos. La base de tiempos puede ser distinta para cada CCP.
Captura	Comparación	CCP2 se puede configurar para resetear TMR1 o TMR3 en el disparo de un evento especial (dependiendo de la base de tiempos utilizada). Se pueden hacer conversiones A/D automáticas en el disparo. Puede afectar a las operaciones de CCP1 si tienen la misma base de tiempos.
Comparación	Captura	CCP1 se configura para resetear TMR1 o TMR3 en el disparo de un evento especial (dependiendo de la base de tiempos utilizada). Puede afectar a las operaciones de CCP2 si tienen la misma base de tiempos.
Comparación	Comparación	Cada modulo se puede configurar para que resetee la base de tiempos en un disparo de un evento especial. Se pueden utilizar las conversiones A/D automáticas en el CCP2. Puede haber conflictos si los dos módulos usan la misma base de tiempos.
Captura	PWM ⁽¹⁾	Ninguno
Comparación	PWM ⁽¹⁾	Ninguno
PWM ⁽¹⁾	Captura	Ninguno
PWM ⁽¹⁾	Comparación	Ninguno
PWM ⁽¹⁾	PWM	Ambos PWMs tendrán la misma frecuencia y ratio de actualización (interrupción TMR2).

Nota 1: Incluye las operaciones PWM estándar y mejorado.

15.1.2 CCP2 ASIGNACIÓN DE PIN

La asignación de pines para el CCP2 (la entrada de captura, salida de comparación y PWM) puede cambiar, basado en la configuración del dispositivo. El bit de configuración CCP2Mx determina qué pin CCP2 se multiplexa. Por defecto, está asignado a RC1 (CCP2Mx=1). Si el bit de configuración se borra, CCP2 se multiplexa con RB3. Cambiar la asignación del CCP2 no cambia automáticamente los requisitos del puerto del pin. Los usuarios deben verificar siempre que el registro TRIS se configura correctamente con la operación CCP2, sin importar donde se localiza.

15.2 MODO CAPTURA

En el modo captura, el par de registros CCPRxH:CCPRxL capturan el valor de 16bits del TMR1 o del TMR3 cuando ocurre un acontecimiento en el pin correspondiente CCPx. Un acontecimiento se define como:

- cada flanco de bajada
- cada flanco de subida
- cada 4º flanco de subida
- cada 16º flanco de subida

El acontecimiento se selecciona con los bits de selección del modo, CCPxM3:CCPxM0 (CCPxCON<3:0>). Cuando se hace una captura, el flag de petición de la interrupción, CCPxIF, se activa; se debe borrar por software. Si ocurre otra captura antes de que el valor en el registro CCPRx se lea, el valor anterior capturado se sobrescribe con el valor nuevo.

15.2.1 CONFIGURACIÓN DEL PIN CCP

En el modo captura, el pin apropiado CCPx debe configurarse como entrada activando el bit correspondiente del TRIS.

Nota: Si se configura RB3/CCP2 o RC1/CCP2 como salida, al escribir en el puerto puede causar una condición de captura.

15.2.2 SELECCIÓN DE MODO TIMER1/TIMER3

Los temporizadores que se deben utilizar con la característica de captura (Timer1 y/o Timer3) deben funcionar en modo temporizador o modo contador síncrono. En el modo contador asíncrono, la captura no funcionará. El temporizador que se utilice con cada módulo CCP se selecciona en el registro T3CON.

15.2.3 INTERRUPCIÓN POR SOFTWARE

Cuando se cambia el modo captura, se puede generar una interrupción falsa de captura. El usuario debe mantener el bit de permiso de interrupción CCPxIE borrado para evitar falsas interrupciones. El flag de interrupción, CCPxIF, tiene que borrarse después de cualquier modo de funcionamiento.

15.2.4 CCP PRESCALER

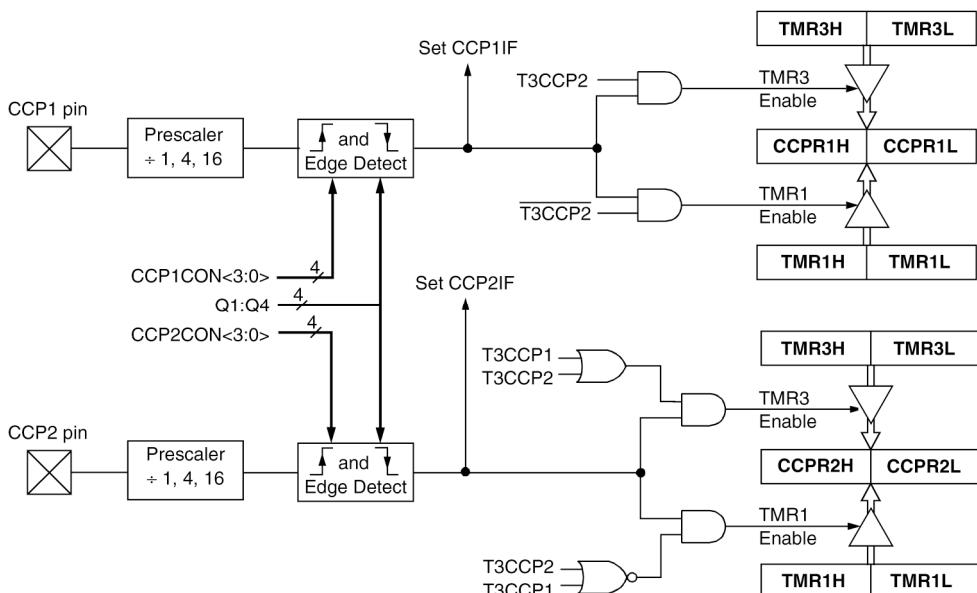
Hay cuatro ajustes del prescaler en el modo captura. Se especifican como parte del modo de funcionamiento seleccionado por los bits selectores del modo (CCPxM3:CCPxM0). Siempre que el módulo CCP se apague o el modo captura se desactiva, el contador del prescaler se borra. Esto significa que cualquier reset borrará el contador del prescaler.

El cambio de un prescaler de captura a otro puede generar una interrupción. También, el contador del prescaler no se borra, por lo tanto, la primera captura puede ser de un prescaler diferente a cero.

EJEMPLO 15-1: CAMBIO ENTRE PRESCALERS DE CAPTURA (MOSTRADO CCP2)

CLRF	CCP2CON	;Desconecta el CCP2
MOVLW	NEW_CAPT_PS	;Carga el WREG con el valor del nuevo ;prescaler y CCP encendido
MOVWF	CCP2CON	;Carga CCP2CON con este valor

FIGURA 15-1: DIAGRAMA DE BLOQUES DE LAS OPERACIONES DEL MODO CAPTURA



15.3 MODO COMPARACIÓN

En el modo comparación, el valor del registro CCPRx de 16 bits se compara constantemente con el valor del par de registros de TMR1 o de TMR3. Cuando se igualan, el pin de CCPx:

- Se pone en alto
- Se pone en bajo
- flip-flop (alto-a-bajo o bajo-a-alto)
- sin cambiar (es decir, refleja el estado del latch de la E/S)

La acción del pin se basa en el valor del modo seleccionado con los bits (CCPxM3:CCPxM0). Al mismo tiempo, se fija el flag de interrupción, CCPxIF.

15.3.1 TIPO DE CONEXIÓN CCP

El usuario debe configurar el pin CCPx como salida borrando el bit apropiado del TRIS.

Nota: Borrar el registro CCP2CON forzará el latch de comparación de salida de RB3 o RC1 (dependiendo de la configuración del dispositivo) a nivel bajo por defecto. Éste no es el PORTB o el latch de E/S de PORTC.

15.3.2 SELECCIÓN DE MODO TIMER1/TIMER3

Timer1 y/o Timer3 deben funcionar en modo temporizador, o modo contador síncrono, si el módulo CCP usa la característica de comparación. En modo contador asíncrono, no funciona la comparación.

15.3.3 MODO DE INTERRUPCIÓN POR SOFTWARE

Cuando se elige el modo de interrupción por software (CCPxM3:CCPxM0=1010), no varia el pin correspondiente al CCPx. Solamente se genera una interrupción CCP, si está permitida, y se activa el bit CCPxIE.

15.3.4 DISPARADOR DE ACONTECIMIENTO ESPECIAL

Ambos módulos CCP se equipan con un disparador de acontecimiento especial. Es una señal interna del hardware generada en el modo comparación para disparar acciones de otros módulos. El disparador de acontecimiento especial se activa seleccionando el modo del disparador de acontecimiento especial de comparación (CCPxM3:CCPxM0=1011).

Para cualquier módulo CCP, el disparador de acontecimiento especial resetea el par de registros del temporizador en cualquier recurso de temporizador asignado como base de tiempo. Esto permite que los registros CCPRx sirvan como un registro de período programable para cualquier temporizador.

El disparador de acontecimiento especial del CCP2 puede comenzar una conversión A/D. Para hacer esto, el convertidor A/D tiene que estar activado.

FIGURA 15-2: DIAGRAMA DE BLOQUES DE LAS OPERACIONES DEL MÓDULO COMPARACIÓN

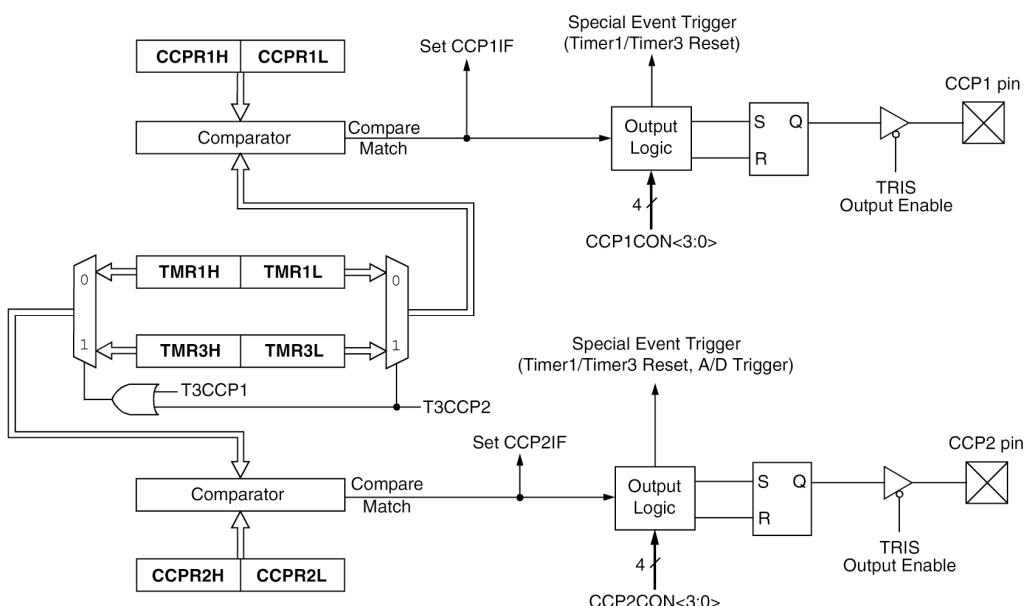


TABLA 15-3: REGISTROS ASOCIADOS A LA CAPTURA Y A LA COMPARACIÓN, TIMER1 Y TIMER3

Nombre	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTCON	GIE/GIEH	PEIE/GIEL	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF
RCON	IPEN	SBOREN ⁽¹⁾	—	RI*	TO*	PD*	POR*	BOR*
PIR1	SPPIF ⁽²⁾	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF
PIE1	SPPIE ⁽²⁾	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE
IPR1	SPPIP ⁽²⁾	ADIP	RCIP	TXIP	SSPIP	CCP1IP	TMR2IP	TMR1IP
PIR2	OSCFIF	CMIF	USBIF	EEIF	BCLIF	HLVDIF	TMR3IF	CCP2IF
PIE2	OSCFIE	CMIE	USBIE	EEIE	BCLIE	HLVDIE	TMR3IE	CCP2IE
IPR2	OSCFIP	CMIP	USBIP	EEIP	BCLIP	HLVDIP	TMR3IP	CCP2IP
TRISB	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0
TRISC	TRISC7	TRISC6	—	—	—	TRISC2	TRISC1	TRISCO
TMR1L	Timer1 Register Low Byte							
TMR1H	Timer1 Register High Byte							
T1CON	RD16	T1RUN	T1CKPS1	T1CKPS0	T1OSCEN	T1SYNC*	TMR1CS	TMR1ON
TMR3H	Timer3 Register High Byte							
TMR3L	Timer3 Register Low Byte							
T3CON	RD16	T3CCP2	T3CKPS1	T3CKPS0	T3CCP1	T3SYNC*	TMR3CS	TMR3ON
CCPR1L	Capture/Compare/PWM Register 1 Low Byte							
CCPR1H	Capture/Compare/PWM Register 1 High Byte							
CCP1CON	P1M1 ⁽²⁾	P1M0 ⁽²⁾	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0
CCPR2L	Capture/Compare/PWM Register 2 Low Byte							
CCPR2H	Capture/Compare/PWM Register 2 High Byte							
CCP2CON	—	—	DC2B1	DC2B0	CCP2M3	CCP2M2	CCP2M1	CCP2M0

Leyenda: - = las localizaciones no están implementadas. Las celdas sombreadas no se utilizan con Captura/Comparación, Timer1 o Timer3.

Nota 1: El bit SBOREN sólo está disponible cuando BOREN<1:0>=01; en otro caso, se lee ‘0’.

2: Este bit no se implementa en los dispositivos de 28pines; mantenerlo siempre a ‘0’.

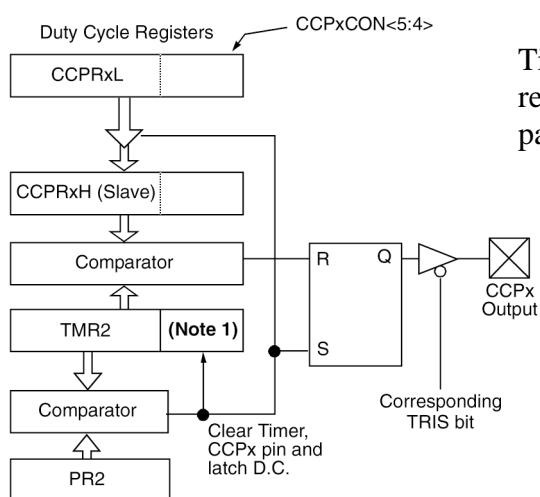
15.4 MODO PWM

En modo de modulación del ancho del pulso (PWM), el pin CCPx produce una salida PWM de hasta 10bits de resolución. Como el pin CCP2 se multiplexa con el PORTB o el latch del PORTC, el bit apropiado de TRIS tiene que borrarse para configurar el pin CCP2 como salida.

Nota: Borrar el registro CCP2CON forzará el latch de comparación de salida de RB3 o RC1 (dependiendo de la configuración del dispositivo) a nivel bajo por defecto. Éste no es el PORTB o el latch de E/S de PORTC.

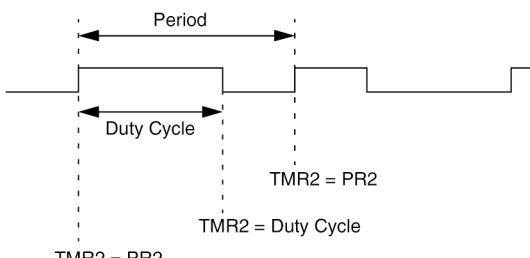
Una salida PWM tiene un tiempo base (periodo) y un tiempo en el que la salida permanece en nivel alto (Duty cicle). La frecuencia del PWM es la inversa del periodo.

FIGURA 15-3: DIAGRAMA DE BLOQUES SIMPLIFICADO DEL PWM



Nota 1: el valor de 8bits de Timer2 se concatena con los 2bits del reloj interno Q o 2bits del prescaler, para crear los 10bit del tiempo base

FIGURA 15-4: SALIDA PWM



15.4.1 PERIODO DE PWM

El periodo se especifica escribiendo el registro PR2. El periodo se calcula con la fórmula:

ECUACIÓN 15-1:

$$\text{Periodo_PWM} = [(PR2) + 1] \cdot 4 \cdot T_{osc} \cdot (\text{Valor_del_presaler_de_TMR2})$$

La frecuencia de PWM se define como $1/[\text{periodo de PWM}]$.

Cuando TMR2 es igual a PR2, ocurren los tres acontecimientos siguientes en el siguiente ciclo de incremento:

- TMR2 se borra
- Se activa el pin CCPx (excepción: si el duty cicle de PWM=0%, el pin CCPx no se activa)
- El duty cicle de PWM se asigna de CCPRxL a CCPRxH

Nota: Los postscalers Timer2 no se utilizan para calcular la frecuencia de PWM. El postscaler se podía utilizar para tener una frecuencia de salida diferente.

15.4.2 DUTY CICLE PWM

El ciclo trabajo de PWM se especifica escribiendo en el registro CCPRxL y los bits CCPxCON<5:4>. Tiene una resolución de 10bits. El CCPRxL contiene los ocho MSbs y los bits CCPxCON<5:4> los dos LSbs. Este valor de 10bits se representa con CCPRxL:CCPxCON<5:4>. La ecuación siguiente se utiliza para calcular el ciclo de trabajo:

ECUACIÓN 15-2:

$$\text{Duty_Cicle_PWM} = (\text{CCPRXL : CCPXCON } < 5 : 4 >) \bullet T_{OSC} \bullet (\text{Valor_del_prescaler_TMR2})$$

CCPRxL y CCPxCON<5:4> se pueden escribir en cualquier momento, pero el valor del duty cicle no se grabará en CCPRxH hasta después de que se igualen PR2 y TMR2 (es decir, se complete el período). En modo PWM, CCPRxH es un registro inalterable.

El registro CCPRxH y un latch interno de 2bits se utilizan como buffer doble del ciclo de trabajo de PWM. El buffer doble es esencial en los problemas técnicos del PWM.

Cuando se igualan los 2bits del latch de TMR2 y CCPRxH, se concatenan con un reloj interno de 2bits Q o 2bits del prescaler de TMR2, el pin de CCPx se borra.

La máxima resolución de PWM (bits) para una frecuencia PWM dada se calcula con la ecuación:

ECUACIÓN 15-3:

$$\text{Resolución_máxima_PWM} = \frac{\log(\frac{F_{OSC}}{F_{PWM}})}{\log(2)} \text{ bits}$$

Nota: Si el valor del ciclo de trabajo de PWM es mayor que el periodo, el pin CCPx no se borrará.

TABLA 15-4: EJEMPLO DE FRECUENCIAS Y RESOLUCIONES A 40MHz

Frecuencia del PWM	2,44kHz	9,77kHz	39,06kHz	156,25kHz	312,50kHz	416,67kHz
Timer Prescaler (1, 4, 16)	16	4	1	1	1	1
PR2 Valor	FFh	FFh	FFh	3Fh	1Fh	17h
Resolución máxima (bits)	10	10	10	8	7	6,58

15.4.3 PWM AUTO-APAGADO (CCP1 SOLAMENTE)

Las características del auto-apagado del PWM mejorado del módulo CCP está también disponible en el CCP1 de los dispositivos de 28pines.

Las características del auto-apagado no están disponibles en el CCP2.

15.4.4 CONFIGURACIÓN DEL PWM

Hay que seguir los siguientes pasos para configurar el PWM:

3. Fijar el período de PWM escribiendo el registro PR2.
4. Fijar el ciclo trabajo de PWM escribiendo el registro CCPRxL y los bits CCPxCON<5:4>.
5. Configurar el pin CCPx como salida borrando el bit del TRIS.
6. Fijar el valor del prescaler TMR2, después permitir el Timer2 escribiendo el T2CON.
7. Configurar el módulo CCPx para la operación PWM.

TABLA 15-5: REGISTROS ASOCIADOS AL PWM Y AL TIMER2

Nombre	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTCON	GIE/GIEH	PEIE/GIEL	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF
RCON	IPEN	SBOREN ⁽¹⁾	—	RI*	TO*	PD*	POR*	BOR*
PIR1	SPPIF ⁽²⁾	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF
PIE1	SPPIE ⁽²⁾	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE
IPR1	SPPPIF ⁽²⁾	ADIP	RCIP	TXIP	SSPIP	CCP1IP	TMR2IP	TMR1IP
TRISB	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0
TRISC	TRISC7	TRISC6	—	—	—	TRISC2	TRISC1	TRISC0
TMR2	Timer2 Register							
PR2	Timer2 Period Register							
T2CON	—	T2OUTPS3	T2OUTPS2	T2OUTPS1	T2OUTPS0	TMR2ON	T2CKPS1	T2CKPS0
CCPR1L	Capture/Compare/PWM Register 1 Low Byte							
CCPR1H	Capture/Compare/PWM Register 1 High Byte							
CCP1CON	P1M1 ⁽²⁾	P1M0 ⁽²⁾	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0
CCPR2L	Capture/Compare/PWM Register 2 Low Byte							
CCPR2H	Capture/Compare/PWM Register 2 High Byte							
CCP2CON	—	—	DC2B1	DC2B0	CCP2M3	CCP2M2	CCP2M1	CCP2M0
ECCP1AS	ECCPASE	ECCPAS2	ECCPAS1	ECCPAS0	PSSAC1	PSSAC0	PSSBD1 ⁽²⁾	PSSBD0 ⁽²⁾
ECCP1DEL	PRSEN	PDC6 ⁽²⁾	PDC5 ⁽²⁾	PDC4 ⁽²⁾	PDC3 ⁽²⁾	PDC2 ⁽²⁾	PDC1 ⁽²⁾	PDC0 ⁽²⁾

Leyenda: - = las localizaciones no están implementadas. Las celdas sombreadas no se utilizan con PWM o Timer2.

Nota 1: El bit SBOREN sólo está disponible cuando BOREN<1:0>=01; en otro caso, se lee ‘0’.

2: Este bit no se implementa en los dispositivos de 28pines; mantenerlo siempre a ‘0’.

16.0 MÓDULO MEJORADO DE CAPTURA/COMPARACIÓN/PWM (ECCP)

Nota: El módulo de ECCP se implementa en los dispositivos de 40/44 pines.

En los dispositivos PIC18F4455/4550, CCP1 se implementa como módulo estándar CCP con capacidades mejoradas de PWM. Incluyen la disposición para 2 ó 4 canales de salida, la polaridad seleccionable por el usuario, el control de banda-muerta y la parada automática y el reinicio. La captura, comparación y las funciones de salida simple de PWM del módulo ECCP son iguales al módulo estándar CCP.

El registro de control para el módulo mejorado diferencia de los registros CCPxCON de los dispositivos PIC18F2255/2550 en que los dos bits más significativos están implementados para controlar el PWM.

Además de la gama mejorada de los modos disponibles a través del registro de CCP1CON, el módulo ECCP tiene dos registros adicionales asociados a las características de las operaciones mejoradas y de la auto-parada de PWM. Son:

- ECCP1AS (configuración de la banda-muerta)
- ECCP1DEL (Configuración de la auto-parada)

16.1 SALIDAS Y CONFIGURACIÓN DE ECCP

El módulo mejorado de CCP puede tener hasta cuatro salidas PWM, dependiendo del modo de funcionamiento seleccionado. Estas salidas, señaladas P1A a P1D, se multiplexan con los pines de entrada-salida de PORTC y PORTD. Las salidas activas dependen del modo de funcionamiento del CCP seleccionado.

Para configurar los pines de entrada-salida como salidas PWM, se debe seleccionar el modo apropiado PWM activando los bits P1M1:P1M0 y CCP1M3:CCP1M0. Se tienen que activar los bits apropiados de TRISC y de TRISD para configurarlos como salidas.

16.1.1 RECURSOS DEL TEMPORIZADOR Y MÓDULO ECCP

Como los módulos estándar CCP, el módulo ECCP puede utilizar los temporizadores 1, 2 ó 3, dependiendo del modo seleccionado. Timer1 y Timer3 están disponibles para los modos captura o comparación, mientras que Timer2 está disponible para el modo PWM. Las interacciones entre el estándar y los módulos mejorados CCP son idénticas a las descritas para los módulos estándares CCP.

REGISTRO 16-1: CCP1CON: REGISTRO DE CONTROL DE ECCP

16.2 MODOS CAPTURA Y COMPARACIÓN

Los modos captura y comparación, a excepción de la operación del disparador de acontecimiento especial, del módulo ECCP son idénticos a los del CCP.

16.2.1 DISPARADOR DE ACONTECIMIENTO ESPECIAL

La salida del disparador del acontecimiento especial de ECCP resetea el par de registros TMR1 o TMR3, dependiendo de cual es la fuente de tiempo seleccionada. Los registros CCPR1H:CCPR1L permiten programar un periodo de 16bits fácilmente para Timer1 o Timer3.

16.3 EL MODO ESTÁNDAR DE PWM

Cuando está configurado en un solo modo de salida, el módulo ECCP funciona igual que el módulo estándar CCP en modo PWM. Esto también se llama modo “CCP compatible”.

TABLA 16-1: ASIGNACION DE PINES PARA VARIOS ECCP1

Modo ECCP	Configuración CCP1CON	RC2	RD5	RD6	RD7
dispositivos PIC18F4455/4550					
Compatible CCP	00xx 11xx	CCP1	RD5/SPP5	RD6/SPP6	RD7/SPP7
Dual PWM	10xx 11xx	P1A	P1B	RD6/SPP6	RD7/SPP7
Quad PWM	x1xx 11xx	P1A	P1B	P1C	P1D

Leyenda: x = no ocurre. Las celdas sombreadas no se utilizan con ECCP.

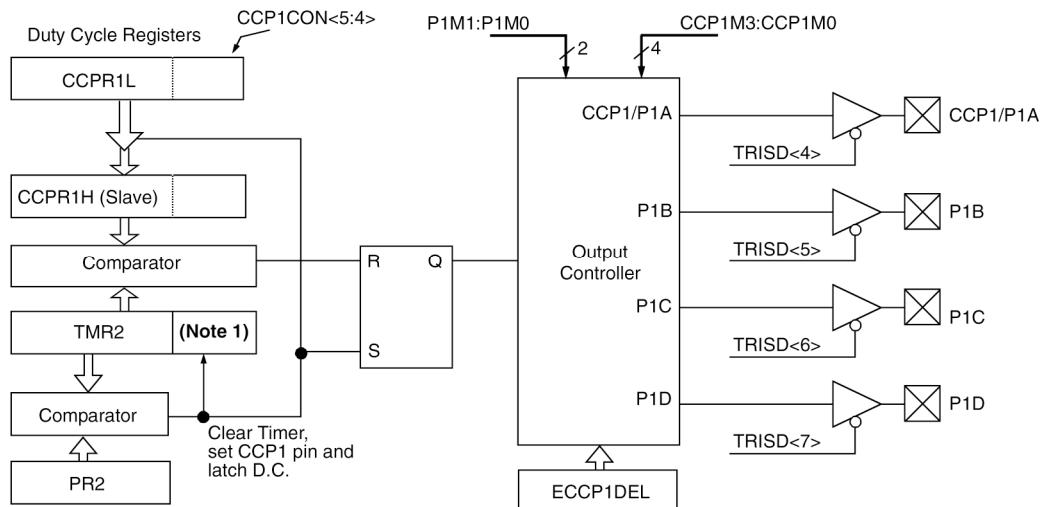
16.4 EL MODO PWM MEJORADO

El modo PWM mejorado proporciona opciones adicionales de salida PWM para ampliar las aplicaciones de control. El módulo es una versión compatible del módulo estándar CCP y ofrece hasta cuatro salidas, señaladas de P1A a P1D. Los usuarios pueden seleccionar la polaridad de la señal (activo-alto o activo-bajo). El modo y la polaridad de la salida del módulo se configuran determinando los bits P1M1:P1M0 y CCP1M3:CCP1M0 del registro CCP1CON.

Todos los registros de control tienen un doble buffer y se cargan al principio de un nuevo ciclo de PWM (el límite del período cuando se resetea Timer2) para prevenir interferencias en las salidas. La excepción es el registro de retraso “banda-muerta” PWM, ECCP1DEL, que se carga en el límite del duty cycle o en el límite del período (el que llegue primero). Debido al buffer, el módulo espera hasta los resets asignados del temporizador en vez de comenzar inmediatamente. Esto significa que las formas de onda de PWM mejorado no son exactamente las formas de onda estándares PWM, pero se compensa con un ciclo completo de la instrucción ($4T_{OSC}$).

Como antes, el usuario debe configurar manualmente los bits apropiados de TRIS como salida.

FIGURA 16-1: DIAGRAMA DE BLOQUES SIMPLIFICADO DEL MÓDULO PWM MEJORADO



Nota 1: el valor de 8bits de Timer2 se concatena con los 2bits del reloj interno Q o 2bits del prescaler, para crear los 10bit del tiempo base

16.4.1 PERIODO DE PWM

El periodo se especifica escribiendo el registro PR2. El periodo se calcula con la fórmula:

ECUACIÓN 16-1:

$$\text{Periodo_PWM} = [(PR2)+1] \cdot T_{OSC} \cdot (\text{Valor_del_presaler_de_TMR2})$$

La frecuencia de PWM se define como $1/\text{[periodo de PWM]}$.

Cuando TMR2 es igual a PR2, ocurren los tres acontecimientos siguientes en el siguiente ciclo de incremento:

- TMR2 se borra
- Se activa el pin CCP1 (excepción: si el duty cicle de PWM=0%, el pin CCP1 no se activa)
- El duty cicle de PWM se asigna de CCPR1L a CCPR1H

Nota: Los postscalers Timer2 no se utilizan para calcular la frecuencia de PWM. El postscaler se podía utilizar para tener una frecuencia de salida diferente.

16.4.2 DUTY CICLE PWM

El ciclo trabajo de PWM se especifica escribiendo en el registro CCPR1L y los bits CCP1CON<5:4>. Tiene una resolución de 10bits. El CCPR1L contiene los ocho MSbs y los bits CCP1CON<5:4> los dos LSbs. Este valor de 10 bits se representa con CCPR1L:CCP1CON<5:4>. La ecuación siguiente se utiliza para calcular el ciclo de trabajo:

ECUACIÓN 16-2:

$$\text{Duty_Cycle_PWM} = (CCPR1L : CCP1CON < 5 : 4 >) \cdot T_{OSC} \cdot (\text{Valor_del_presaler_TMR2})$$

CCPR1L y CCP1CON<5:4> se pueden escribir en cualquier momento, pero el valor del duty cycle no se grabará en CCPR1H hasta después de que se iguale PR2 y TMR2 (es decir, se complete el periodo). En modo PWM, CCPR1H es un registro inalterable.

El registro CCPR1H y un latch interno de 2 bits se utilizan como buffer doble del ciclo de trabajo de PWM. El buffer doble es esencial en los problemas técnicos del

PWM. Cuando se igualan los 2bits del latch de TMR2 y CCPR1H, se concatenan con un reloj interno de 2 bits Q o 2 bits del prescaler de TMR2, el pin de CCP1 se borra.

La máxima resolución de PWM (bits) para una frecuencia PWM dada se calcula con la ecuación:

ECUACIÓN 16-3:

$$\text{Resolución máxima PWM} = \frac{\log(\frac{F_{OSC}}{F_{PWM}})}{\log(2)} \text{ bits}$$

Nota: Si el valor del ciclo de trabajo de PWM es mayor que el periodo, el pin CCPx no se borrará.

TABLA 16-2: EJEMPLO DE FRECUENCIAS Y RESOLUCIONES A 40MHz

Frecuencia del PWM	2,44kHz	9,77kHz	39,06kHz	156,25kHz	312,50kHz	416,67kHz
Timer Prescaler (1, 4, 16)	16	4	1	1	1	1
PR2 Valor	FFh	FFh	FFh	3Fh	1Fh	17h
Resolución máxima (bits)	10	10	10	8	7	6,58

16.4.3 CONFIGURACIONES DE LA SALIDA DE PWM

Los bits P1M1:P1M0 del registro CCP1CON permiten una de cuatro configuraciones:

- Salida simple
- Salida half-bridge (medio-puente)
- Salida full-bridge (puente-completo); modo directo
- Salida full-bridge; modo inverso

FIGURA 16-2: RELACIONES DE LA SALIDA PWM (ESTADO ACTIVO-ALTO)

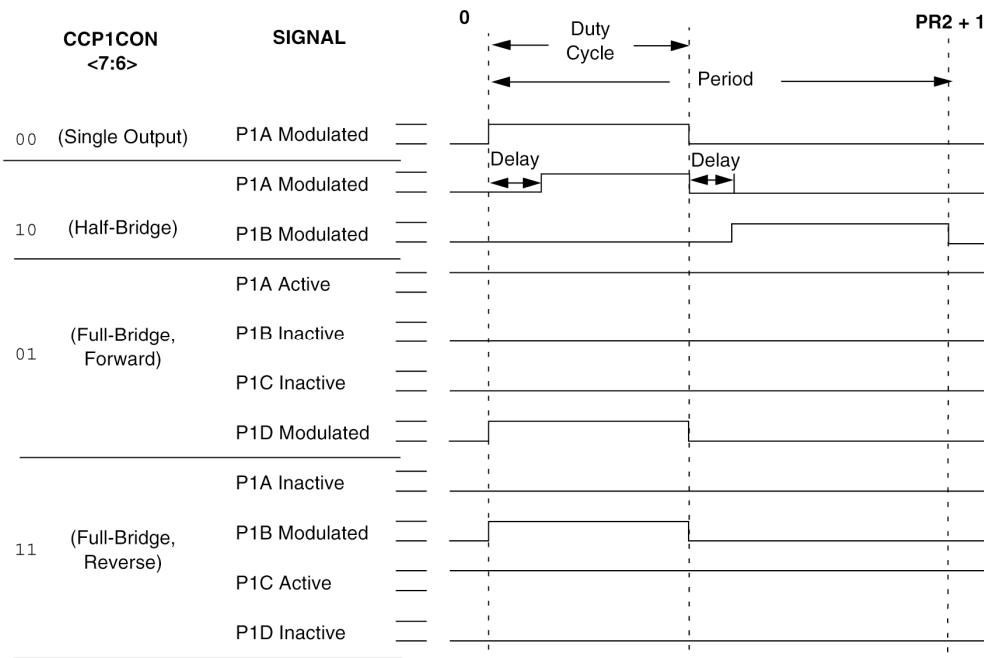
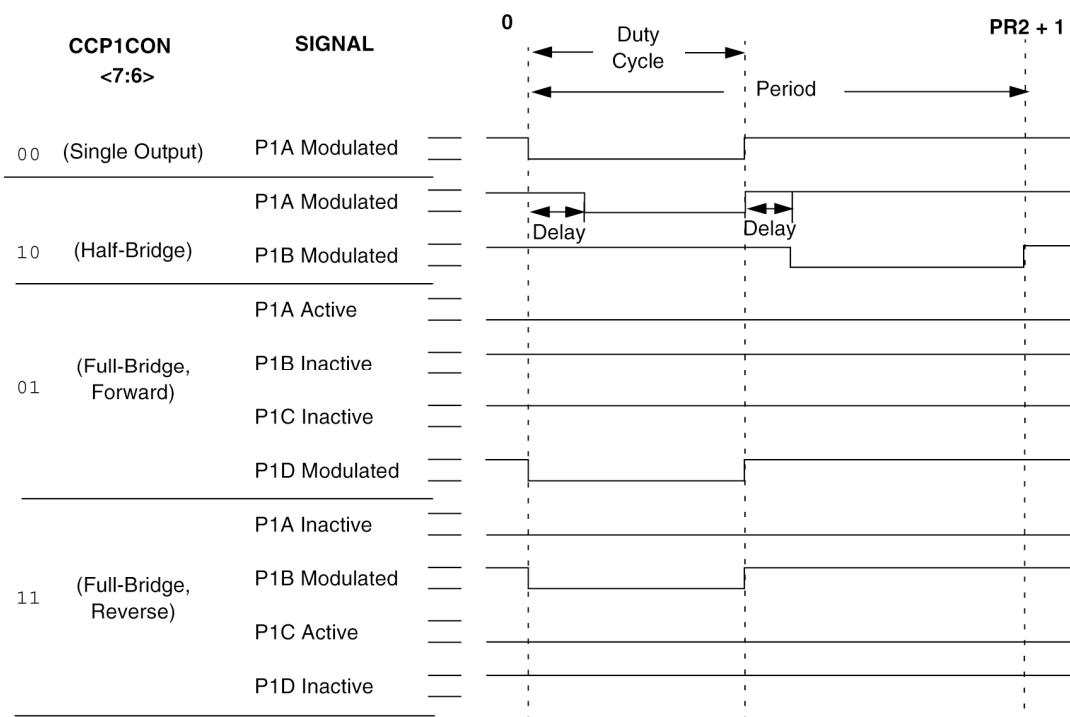


FIGURA 16-3: RELACIONES DE LA SALIDA PWM (ESTADO ACTIVO-BAJO)



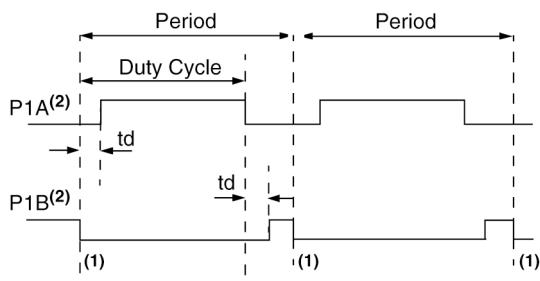
16.4.4 EL MODO HALF-BRIDGE

En el modo de salida half-bridge, dos pines se utilizan como salidas para controlar cargas en dos sentidos. La señal de salida PWM sale por el pin P1A, mientras que la señal de salida complementaria sale por el pin P1B. Este modo se puede utilizar para aplicaciones de half-bridge, o en aplicaciones full-bridge donde cuatro interruptores se modulan con dos señales PWM.

En modo de salida half-bridge, el retraso programable banda-muerta se puede utilizar para prevenir los picos de corriente. El valor de los bits PDC6:PDC0 determinan el número de ciclos de la instrucción antes de que la salida se active. Si el valor es mayor que el ciclo de trabajo, la salida correspondiente no se activa durante el ciclo entero.

Como las salidas P1A y P1B se multiplexan con los latch PORTC<2> y PORTD<5>, los bits TRISC<2> y TRISD<5> se tienen que borrar para configurar P1A y P1B como salidas.

FIGURA 16-4: SALIDAS DEL HALF-BRIDGE

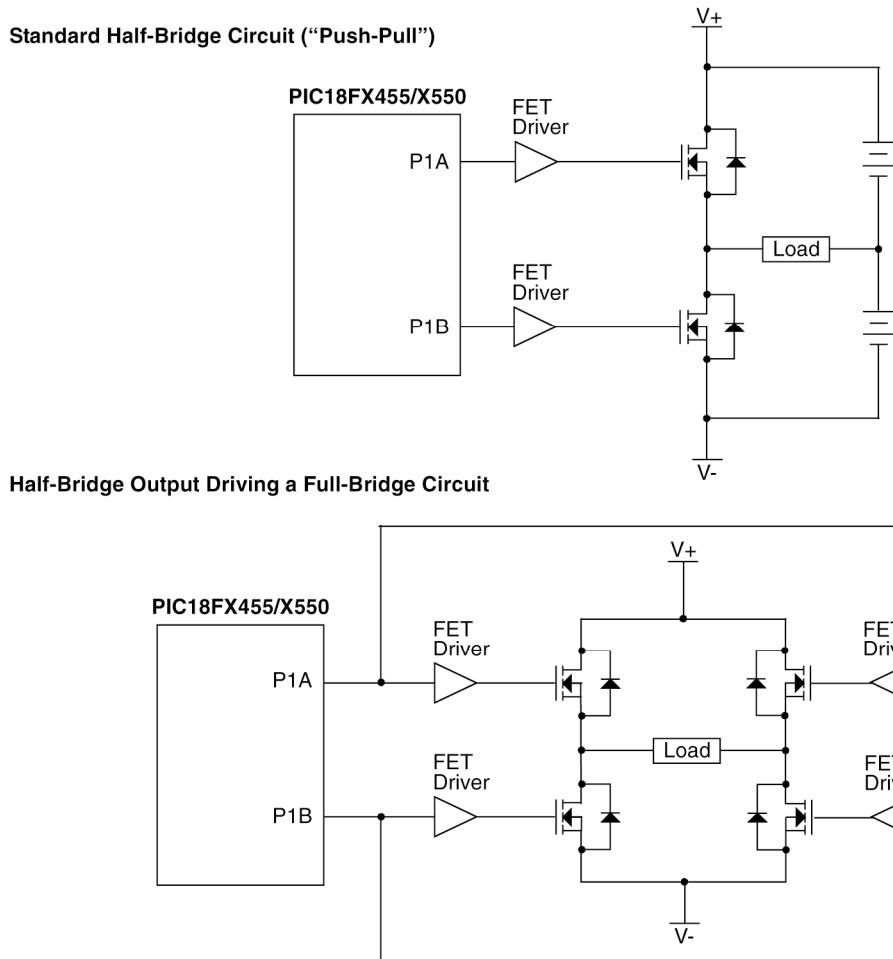


td = Retraso banda-muerta

Nota 1: En este momento el registro TMR2 es igual al registro PR2

2: Las salidas se muestran en activo-alto

FIGURA 16-5: EJEMPLOS DE LOS MODOS DE APLICACIÓN DEL HALF-BRIDGE



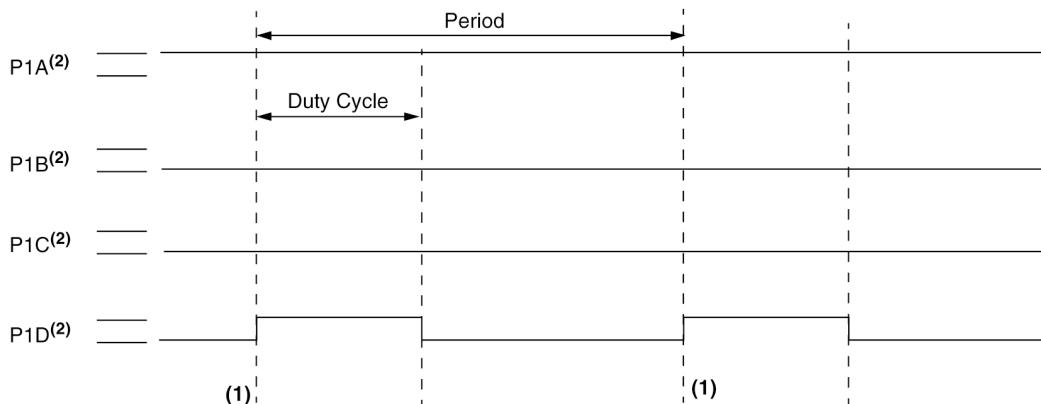
16.4.5 MODO FULL-BRIDGE

En la salida del modo Full-Bridge, los cuatro pines se utilizan como salidas; con dos entradas activas a la vez. En el modo directo, el pin P1A está activo y el pin P1D modula. En el modo inverso, el pin P1C está activo y el pin P1B modula.

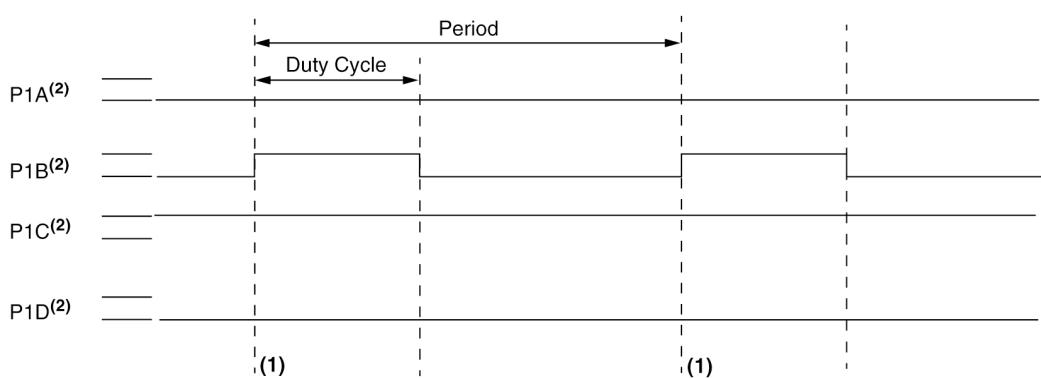
Las salidas P1A, P1B, P1C y P1D se multiplexan con los latchs de datos PORTC<2>, PORTD<5>, PORTD<6> y PORTD<7>. Los bits TRISC<2>, TRISD<5>, TRISD<6> y TRISD<7> deben borrarse para configurar los pines como salidas.

FIGURA 16-6: SALIDA FULL-BRIDGE

Forward Mode



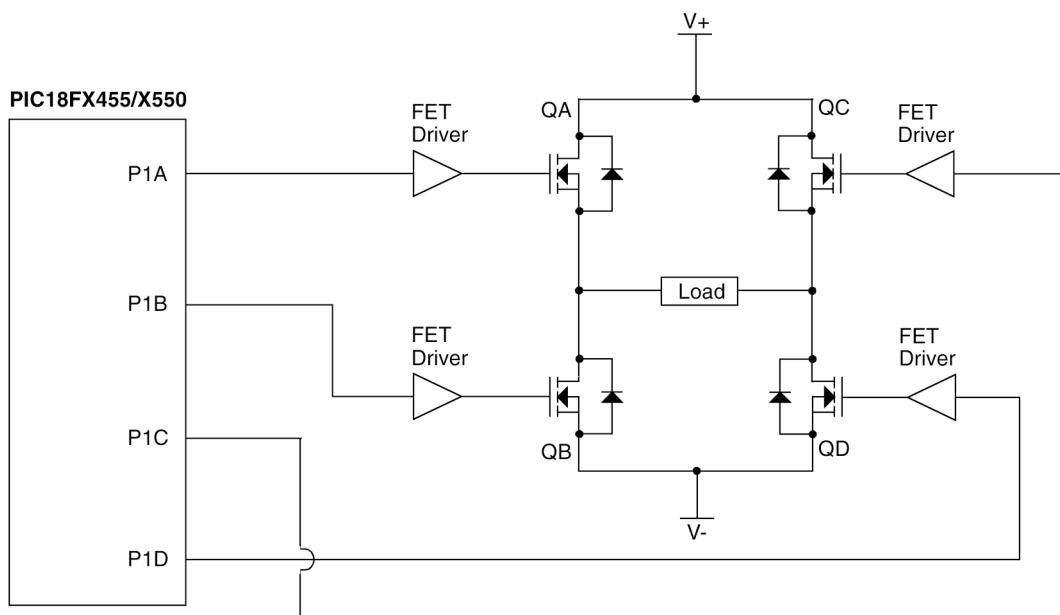
Reverse Mode



Nota 1: En este momento el registro TMR2 es igual al registro PR2

2: Las salidas se muestran en activo-alto

FIGURA 16-7: EJEMPLO DE APLICACIÓN FULL-BRIDGE



16.4.5.1 CAMBIO DE DIRECCIÓN EN EL MODO FULL-BRIDGE

En el modo Full-Bridge, el bit P1M1 del registro CCP1CON permite que el usuario controle la dirección directa/inversa. Cuando el firmware cambia este bit de control de dirección, el módulo asumirá la nueva dirección en el siguiente ciclo PWM.

Momentos antes del final del período actual de PWM, las salidas moduladas (P1B y P1D) se ponen en estado inactivo, mientras que las salidas no moduladas (P1A y P1C) se cambian a la dirección opuesta. Esto ocurre en un intervalo del tiempo de ($4T_{osc}^*$ (valor de Timer2 Prescaler)) antes de que comience el próximo período de PWM. El prescaler del Timer2 será 1, 4 ó 16, dependiendo del valor de los bits T2CKPS1:T2CKPS0 (T2CON<1:0>). Durante el intervalo del cambio de las salidas no moduladas al principio del siguiente período, las salidas moduladas (P1B y P1D) siguen estando inactivas.

Observar que en el modo de salida Full-Bridge, el módulo ECCP no proporciona ningún retraso “banda-muerta”. Generalmente, porque sólo modula una salida, y no se necesita el retraso. Sin embargo, hay una situación donde si se necesita. Esta situación ocurre cuando las condiciones siguientes son verdaderas:

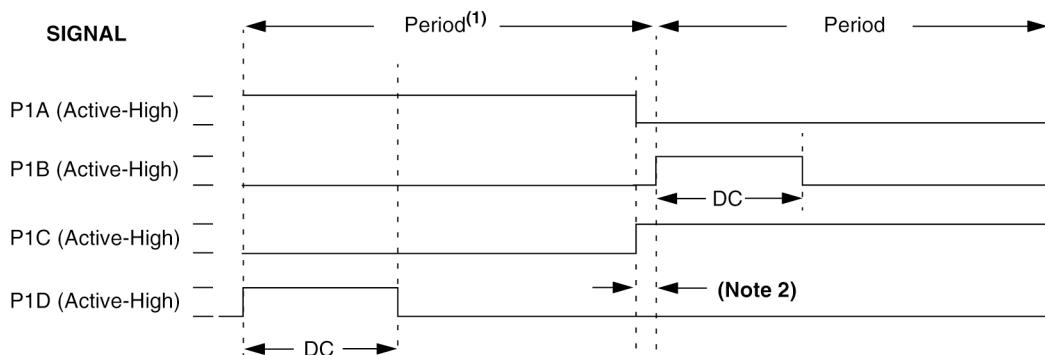
1. La dirección de la salida PWM cambia cuando el ciclo de trabajo de la salida está cerca o es el 100%.
2. El momento del apagado, incluyendo el circuito del dispositivo y del conductor de energía, es mayor que el tiempo de encendido.

Si hay que cambiar la dirección en una aplicación con el duty cicle alto, hay que cumplir una de estas condiciones:

1. Reducir el periodo de PWM antes de cambiar las direcciones.
2. Utilizar commutaciones más rápidas.

Pueden existir otras opciones para prevenir el pico de corriente.

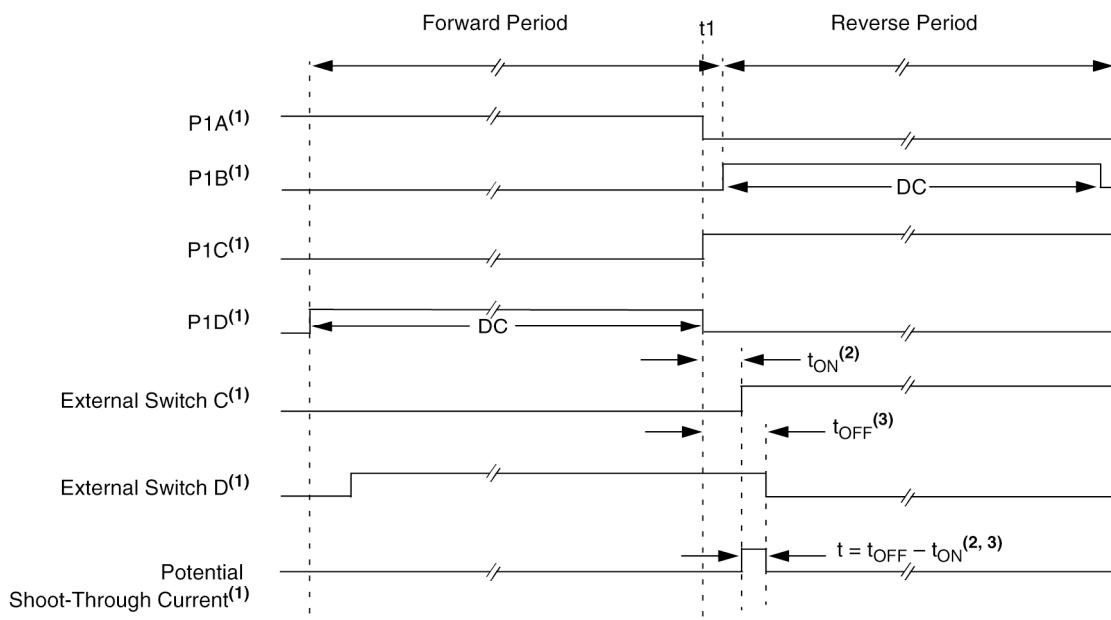
FIGURA 16-8: CAMBIO DE DIRECCIÓN



Nota 1: el bit de dirección del registro de control CCP1 (CCP1CON<7>) se escribe en cualquier momento del ciclo PWM.

2: Cuando se cambian las direcciones, las señales P1A y P1C cambian antes del final del ciclo PWM en intervalos de $4T_{osc}$, $16T_{osc}$ o $64T_{osc}$, dependiendo del valor del prescaler de Timer2. Las señales de modulación de P1B y P1D están inactivas.

FIGURA 16-9: CAMBIO DE DIRECCIÓN CON UN DUTY CICLE CERCA DEL 100%



Nota 1: Todas las señales se muestran como activa-alto

2: t_{ON} es el retraso del cambio de QC

3: t_{OFF} es el retraso del cambio de QD

16.4.6 RETRASO PROGRAMABLE “BANDA-MUERTA”

Nota: El retraso programable “banda-muerta” no se implementa en los dispositivos de 28pin con los módulos CCP estándares.

En las aplicaciones Half-Bridge donde todos los conmutadores se modulan con la frecuencia de PWM, los conmutadores, normalmente, necesitan más tiempo para apagarse. Si los conmutadores cambian al mismo tiempo (uno se activa y el otro se apaga), ambos pueden estar activos en un período de tiempo corto hasta que un conmutador se apague. Durante este breve intervalo, puede haber un pico muy grande de corriente en ambos conmutadores, poniendo en cortocircuito la fuente del puente. Para evitar este pico destructivo, primero se apaga el conmutador activo y, después de un retraso, se activa el otro.

En el modo de salida Half-Bridge, existe un retraso digital programable “banda-muerta” para evitar los picos de los cambios. El retraso ocurre en la transición de la señal del estado inactivo al estado activo. Los bits PDC6:PDC0 del registro ECCP1DEL determinan el período del retraso en términos de ciclos de instrucción del microcontrolador (T_{CY} o $4 T_{OSC}$). Estos bits no están disponibles en los dispositivos de 28pines, pues el módulo estándar CCP no soporta la operación Half-Bridge.

REGISTRO 16-2: ECCP1DEL: REGISTRO DEL RETRASO “BANDA-MUERTA”

BIT 7 PRSEN	BIT 6 PDC6 ⁽¹⁾	BIT 5 PDC5 ⁽¹⁾	BIT 4 PDC4 ⁽¹⁾	BIT 3 PDC3 ⁽¹⁾	BIT 2 PDC2 ⁽¹⁾	BIT 1 PDC1 ⁽¹⁾	BIT 0 PDC0 ⁽¹⁾
----------------	------------------------------	------------------------------	------------------------------	------------------------------	------------------------------	------------------------------	------------------------------

R/W-0 BIT 7	R/W-0 PRSEN:	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
		Bit de permiso de reinicio de PWM 1 = sobre el auto-apagado, los bits ECCPASE se borran automáticamente cuando se ejecuta el evento de apagado; el PWM reinicia automáticamente 0 = sobre el auto-apagado, los ECCPASE se tienen que borrar por software para reiniciar el PWM					
BIT 6-0	PDC6:PDC0:		Bits de cuenta del retraso ⁽¹⁾ Tiempo de retraso, en ciclos Fosc/4 (4*Tosc), entre la programación y el tiempo actual de una señal de PWM de la transición a activo.				

Nota 1: Reservado en los dispositivos de 28pines; mantener estos bits borrados.

16.4.7 PWM MEJORADO AUTO-APAGADO

Cuando se programa ECCP con un modo PWM mejorado, los pines activos de salida se pueden configurar para un auto-apagado. El auto-apagado coloca inmediatamente los pines de salida del PWM mejorados en un estado definido de apagado cuando ocurre un acontecimiento de apagado.

Un acontecimiento de apagado se puede provocar con los módulos del comparador, con un nivel bajo en el pin RB0/AN12/INT0/FLT0/SDI/SDA, o de cualquier combinación de estas tres fuentes. Los comparadores se pueden utilizar para supervisar una entrada de tensión proporcional a una corriente supervisada por el circuito del puente. Si la tensión excede un umbral, el comparador cambia de estado y acciona un apagado. Alternativamente, una señal numérica en el pin INT0 puede provocar un apagado. La característica del auto-apagado se puede inhabilitar al no seleccionar ninguna fuente de auto-apagado. Las fuentes de auto-apagado que se utilizarán se seleccionan usando los bits ECCPAS2:ECCPAS0 (bits<6:4> del registro ECCP1AS).

Cuando ocurre un apagado, los pines de salida se colocan asíncronamente en su estado de apagado, especificados por los bits PSSAC1:PSSAC0 y PSSBD1:PSSBD0 (ECCP1AS3:ECCP1AS0). Cada pareja de pines (P1A/P1C y P1B/P1D) se puede activar para conducir en alto, conducir en bajo o un tercer estado (no conduce).

El bit ECCPASE (ECCP1AS<7>) también se activa para llevar a cabo las salidas mejoradas de PWM a sus estados de parada.

El bit ECCPASE se activa por hardware cuando ocurre un acontecimiento de parada. Si el reinicio automático no está activo, el bit ECCPASE lo borra el firmware cuando cesa la causa de apagado. Si el reinicio automático está activo, el bit ECCPASE se borra automáticamente cuando cesa la causa de apagado.

Si se activa el bit ECCPASE cuando comienza un periodo de PWM, las salidas del PWM siguen estando en su estado de apagado durante el periodo entero de PWM. Cuando el bit ECCPASE está borrado, las salidas del PWM volverán al funcionamiento normal al principio del periodo siguiente de PWM.

Nota: El escribir el bit ECCPASE se desactiva cuando una condición de apagado está activa.

REGISTRO 16-3: ECCP1AS: REGISTRO DE CONTROL DEL AUTO-APAGADO DEL ECCP MEJORADO

BIT 7 ECCPASE R/W-0	BIT 6 ECCPAS2 R/W-0	BIT 5 ECCPAS1 R/W-0	BIT 4 ECCPAS0 R/W-0	BIT 3 PSSAC1 R/W-0	BIT 2 PSSAC0 R/W-0	BIT 1 PSSBD1 ⁽¹⁾ R/W-0	BIT 0 PSSBD0 ⁽¹⁾ R/W-0
---------------------------	---------------------------	---------------------------	---------------------------	--------------------------	--------------------------	-----------------------------------------	-----------------------------------------

BIT 7	ECCPASE:	Bit de estado del evento auto-apagado del ECCP 1= A ocurrido un acontecimiento de apagado; las salidas del ECCP están en estado de apagado 0= las salidas del ECCP están funcionando
BIT 6-4	ECCPAS2:ECCPAS0:	Bits selectores de la fuente del Auto-apagado del ECCP 111= FLT0 o comparador 1 o comparador 2 110= FLT0 o comparador 2 101= FLT0 o comparador 1 100= FLT0 011= cualquier comparador 1 ó 2 010= salida del comparador 2 001= salida del comparador 1 000= Auto-apagado desactivado
BIT 3-2	PSSAC1:PSSAC0:	Bits de control del estado de los pines A y C en un apagado 1x= los pines A y C en un tercer estado 01= los pines A y C a ‘1’ 00= los pines A y C a ‘0’
BIT 1-0	PSSBD1:PSSBD0:	Bits de control del estado de los pines B y D en un apagado ⁽¹⁾ 1x= pin B y D tercer estado 01= los pines B y D a ‘1’ 00= los pines B y D a ‘0’

Nota 1: Reservado en los dispositivos de 28pines; mantener estos bits borrados.

16.4.7.1 El Auto-apagado y el Auto-Reinicio

La característica del auto-apagado se puede configurar para permitir el reinicio automático del módulo después de un acontecimiento de apagado. Esto se permite activando el bit PRSEN del registro ECCP1DEL (ECCP1DEL<7>).

En modo del apagado con PRSEN=1, el bit ECCPASE seguirá activo mientras continúe la causa de apagado. Cuando la condición de apagado desaparece, el bit ECCP1ASE se borra. Si PRSEN=0, cuando ocurre la condición de parada, el bit ECCPASE sigue activo hasta que se borra por firmware. Cuando ECCPASE se borra, el PWM mejorado parará al principio del siguiente periodo PWM.

Nota: El escribir el bit ECCPASE se desactiva cuando la condición de apagado esté activa.

Independientemente del ajuste del bit PRSEN, si la fuente del auto-apagado es uno de los comparadores, la condición de la parada es un nivel. El bit ECCPASE no puede borrarse mientras dure la causa del apagado.

El modo auto-apagado puede迫使 escribiendo un ‘1’ en el bit ECCPASE.

FIGURA 16-10: AUTO-APAGADO (PRSEN=1, AUTO-REINICIO ACTIVO)

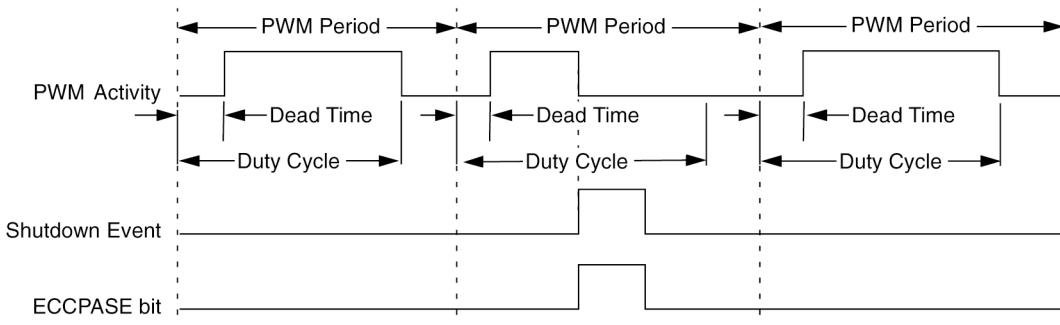
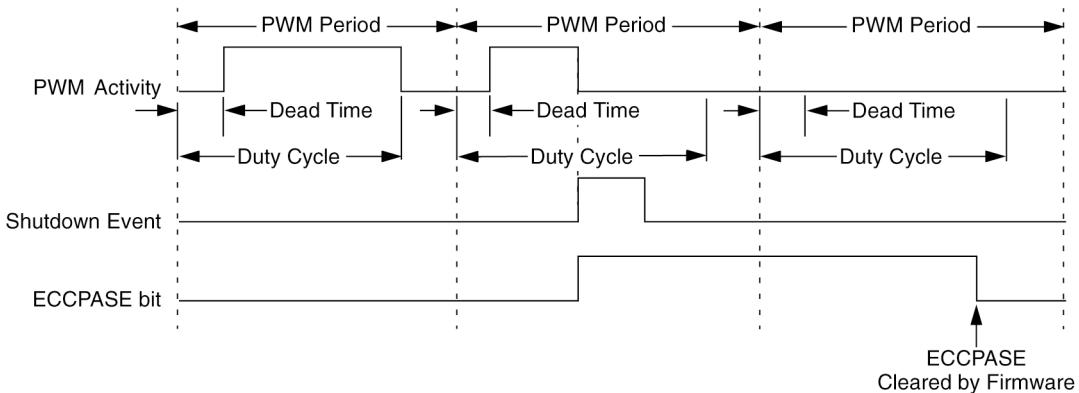


FIGURA 16-11: AUTO-APAGADO (PRSEN=0, AUTO-REINICIO DESACTIVADO)



16.4.8 CONSIDERACIONES DE REINICIO

Cuando el módulo CCP se utiliza en el modo PWM, el hardware de la aplicación debe utilizar las resistencias apropiadas pull-up y/o pull-down en los pines de salida del PWM. Cuando el microcontrolador se reinicia, todos los pines de E/S están en el estado de alta impedancia. Los circuitos externos deben mantener los componentes de potencia apagados hasta que el microcontrolador conduzca los pines de E/S con los niveles apropiados de señal o activar las salidas del PWM.

Los bits CCP1M1:CCP1M0 (CCP1CON<1:0>) permiten que el usuario elija si las señales de salida de PWM sean activa-alto o activa-bajo para cada par de pines de salida (P1A/P1C y P1B/P1D). Las polaridades de la salida de PWM deben seleccionarse antes de que los pines de PWM se configuren como salidas. Cambiar la configuración de la polaridad mientras que los pines de PWM están como salidas no se recomiendan, puesto que puede dar lugar a daño a los circuitos de la aplicación.

Los latch de salida P1A, P1B, P1C y P1D pueden no estar en el estado apropiado al iniciarse el módulo PWM. Activar los pines del PWM como salida al mismo tiempo que el módulo CCP puede causar daño al circuito de la aplicación. El módulo CCP debe permitirse en el modo de salida apropiado y terminar un ciclo completo antes de configurar los pines como salidas. El final de un ciclo completo se indica con el bit TMR2IF que permanece activo cuando el segundo periodo comienza.

16.4.9 PASOS PARA CONFIGURAR EL PWM

Hay que tomar los siguientes pasos para configurar el módulo CCP en el modo PWM:

1. Configurar los pines de PWM, P1A y P1B (y P1C y P1D, si se utilizan), como entradas activando los bits correspondientes de TRIS.
2. Fijar el período de PWM cargando el registro PR2.
3. Si se requiere el Auto-apagado:
 - Inhabilita el Auto-apagado (ECCPASE = 0)
 - Configurar la fuente (FLT0, comparador 1 o comparador 2)
 - Esperar una condición de no apagado
4. Configurar el módulo ECCP con el modo PWM deseado y configurar el registro CCP1CON cargando los valores apropiados:
 - Seleccionar una de las configuraciones y de las direcciones disponibles de salida con los bits P1M1:P1M0.
 - Seleccionar las polaridades de las señales de salida de PWM con los bits CCP1M3:CCP1M0.
5. Fijar el ciclo de trabajo de PWM cargando el registro CCPR1L y los bits CCP1CON<5:4>.
6. Para el modo de salida Half-Bridge, fijar el retraso “banda-muerta” cargando el valor apropiado en ECCP1DEL<6:0>.
7. Si se necesita el auto-apagado, cargar el registro ECCP1AS:
 - Seleccionar las fuentes del auto-apagado usando los bits ECCPAS2:ECCPAS0.
 - Selecciona los estados de parada de los pines de salida usando los bits PSSAC1:PSSAC0 y PSSBD1:PSSBD0.
 - Activar el bit ECCPASE (ECCP1AS<7>).
 - Configurar los comparadores con el registro CMCON.
 - Configurar las entradas del comparador como entradas analógicas.
8. Si se necesita el auto-reinicio, activar el bit PRSEN (ECCP1DEL<7>).
9. Configurar y comenzar TMR2:
 - Borrar el flag de interrupción de TMR2 borrando el bit TMR2IF (PIR1<1>).
 - Determinar el valor del prescaler de TMR2 cargando los bits T2CKPS (T2CON<1:0>).
 - Permitir el Timer2 activando el bit TMR2ON (T2CON<2>).
10. Permitir las salidas del PWM después de que un nuevo ciclo de PWM haya comenzado:
 - Esperar hasta que se desborda TMRn (bit TMRnIF permitido).
 - Activar los pines de salidas CCP1/P1A, P1B, P1C y/o P1D borrando los bits respectivos de TRIS.
 - Borrar el bit ECCPASE (ECCP1AS<7>).

16.4.10 OPERANDO EN LOS MODOS DE AHORRO DE ENERGÍA

En el modo sleep, todas las fuentes del reloj están desactivadas. El Timer2 no se incrementará y el estado del módulo no cambiará. Si el pin de ECCP está conduciendo un valor, continuará conduciendo ese valor. Cuando el dispositivo despierta, continuará en este estado. Si se permiten el reinicio a dos velocidades, la frecuencia inicial de reinicio del INTOSC y del postscaler puede que no sean estables inmediatamente.

En el modo PRI_IDLE, el reloj primario continuará registrando el módulo ECCP sin cambios. En el resto de los modos de ahorro de energía, el reloj del ahorro de

energía seleccionado gobernará el Timer2. Los relojes de otros modos de ahorro de energía tendrán una frecuencia diferente a la del reloj primario.

16.4.10.1 La operación con el monitor de reloj a prueba de fallos

Si el monitor de reloj a prueba de fallos está activo, un fallo del reloj forzará al dispositivo al modo RC_RUN y el bit OSCFIF (PIR2<7>) se activará. Al ECCP lo controlará la fuente del oscilador interno de reloj, que puede tener diversas frecuencias de reloj.

16.4.11 EFECTOS DE UN RESET

Los resets por aumento de energía y por resets sucesivos forzarán a los puertos al modo entrada y el CCP a su estado de reset.

Esto fuerza el módulo CCP mejorado a resetear a un estado compatible con el módulo CCP estándar.

TABLA 16-3: REGISTROS ASOCIADOS A LOS MÓDULOS ECCP, TIMER1 Y TIMER3

Nombre	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTCON	GIE/GIEH	PEIE/GIEL	TMROIE	INT0IE	RBIE	TMROIF	INT0IF	RBIF
RCON	IPEN	SBOREN ⁽¹⁾	—	RI*	TO*	PD*	POR*	BOR*
IPR1	SPPIP ⁽²⁾	ADIP	RCIP	TXIP	SSPIP	CCP1IP	TMR2IP	TMR1IP
PIR1	SPPIF ⁽²⁾	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF
PIE1	SPPIE ⁽²⁾	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE
IPR2	OSCFIP	CMIP	USBIP	EEIP	BCLIP	HLVDIP	TMR3IP	CCP2IP
PIR2	OSCFIF	CMIF	USBIF	EEIF	BCLIF	HLVDIF	TMR3IF	CCP2IF
PIE2	OSCFIE	CMIE	USBIE	EEIE	BCLIE	HLVDIE	TMR3IE	CCP2IE
TRISB	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0
TRISC	TRISC7	TRISC6	—	—	—	TRISC2	TRISC1	TRISC0
TRISD ⁽²⁾	TRISD7	TRISD6	TRISD5	TRISD4	TRISD3	TRISD2	TRISD1	TRISD0
TMR1L	Timer1 Register Low Byte							
TMR1H	Timer1 Register High Byte							
T1CON	RD16	T1RUN	T1CKPS1	T1CKPS0	T1OSCEN	T1SYNC*	TMR1CS	TMR1ON
TMR2	Timer2 Module Register							
T2CON	—	T2OUTPS3	T2OUTPS2	T2OUTPS1	T2OUTPS0	TMR2ON	T2CKPS1	T2CKPS0
PR2	Timer2 Period Register							
TMR3L	Timer3 Register Low Byte							
TMR3H	Timer3 Register High Byte							
T3CON	RD16	T3CCP2	T3CKPS1	T3CKPS0	T3CCP1	T3SYNC*	TMR3CS	TMR3ON
CCPR1L	Capture/Compare/PWM Register 1 (LSB)							
CCPR1H	Capture/Compare/PWM Register 1 (MSB)							
CCP1CON	P1M1 ⁽²⁾	P1M0 ⁽²⁾	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0
ECCP1AS	ECCPASE	ECCPAS2	ECCPAS1	ECCPAS0	PSSAC1	PSSAC0	PSSBD1 ⁽²⁾	PSSBD0 ⁽²⁾
ECCP1DEL	PRSEN	PDC6 ⁽²⁾	PDC5 ⁽²⁾	PDC4 ⁽²⁾	PDC3 ⁽²⁾	PDC2 ⁽²⁾	PDC1 ⁽²⁾	PDC0 ⁽²⁾

Leyenda: - = las localizaciones no están implementadas. Las celdas sombreadas no se utilizan con ECCP.

Nota 1: El bit SBOREN sólo está disponible cuando BOREN<1:0>=01

2: Este bit no se implementa en los dispositivos de 28pines.

17.0 BUS SERIE UNIVERSAL (USB)

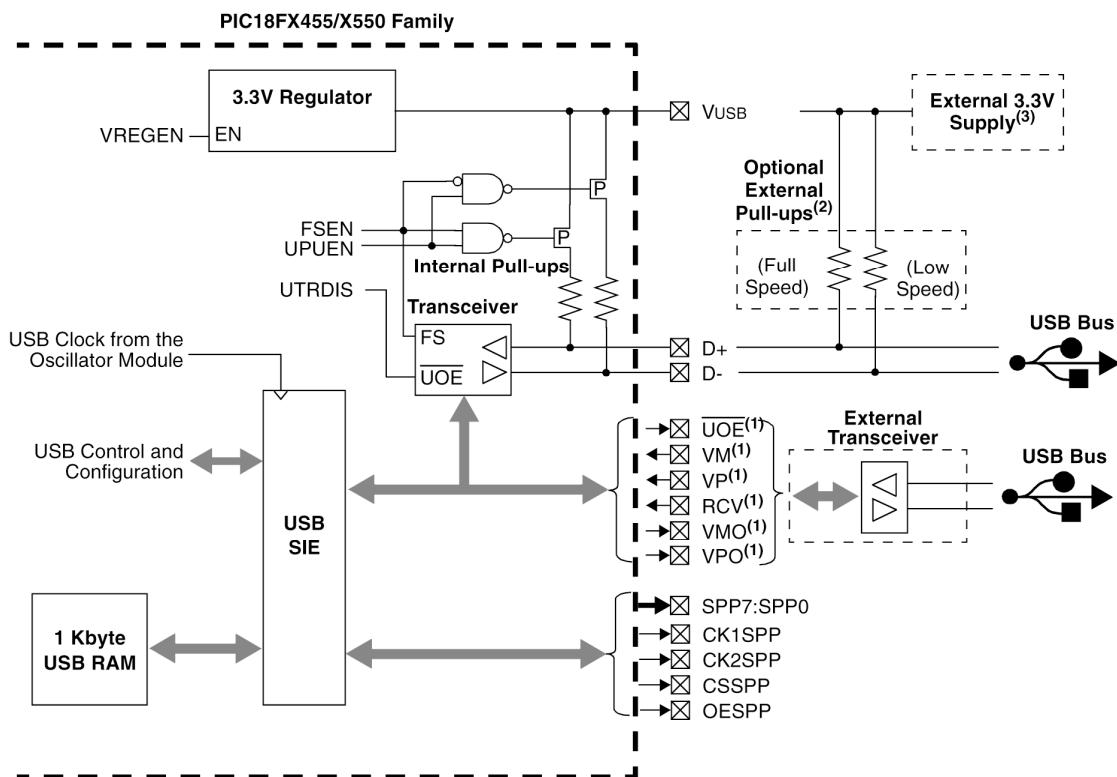
17.1 DESCRIPCIÓN DEL PERIFÉRICO USB

La familia de dispositivos PIC18FX455/X550 contiene una interfaz serie compatible con el SIE (serial interface engine, máquina con comunicación serie) USB “full-speed” (2.0) y “de poca velocidad” (1.0) que permite la comunicación rápida entre cualquier dispositivo USB y el microcontrolador PIC®.

El SIE puede interconectarse directamente al USB, utilizando el transmisor-receptor interno, o puede conectarse a través un transmisor-receptor externo. El PIC tiene un regulador interno de 3,3V para accionar el transmisor-receptor interno en aplicaciones de 5V.

Se han incluido algunas características especiales en el hardware para mejorar el funcionamiento. Se proporciona memoria de puerto dual en la memoria de datos del dispositivo (RAM del USB) para tener acceso directo a la memoria desde el núcleo del microcontrolador y desde el SIE. También se proporcionan unos buffer para que el programador elija libremente el final de la memoria dentro del espacio de la RAM del USB. Existe un puerto paralelo para transmitir datos grandes, por ejemplo datos al puerto paralelo, se ha proporcionado la ayuda de transferencia ininterrumpida de volúmenes de datos grandes, por ejemplo datos síncronos, a los buffer de memoria externos.

FIGURA 17-1: PERIFÉRICO Y OPCIONES DEL USB



Nota 1: Esta señal solo está disponible si el transmisor interno está desactivado (UTRDIS=1)

2: Las resistencias internas pull-up se tienen que desactivar (UPUEN=0)

3: No hay que activar el regulador interno cuando se utiliza una fuente de 3,3V externa.

17.2 ESTADO Y CONTROL DEL USB

Las operaciones del módulo USB se configuran y controlan con tres registros. En total hay 22 registros para manejar las transacciones del USB. Los registros son:

- Registro de control del USB (UCON)
- Registro de configuración del USB (UCFG)
- Registro de estado de la transferencia del USB (USTAT)
- Registro de dirección de dispositivo USB (UADDR)
- Registros del número del frame (UFRMH: UFRML)
- Registros activadores de los Endpoints de 0 a 15 (UEPn)

17.2.1 REGISTRO DE CONTROL DEL USB (UCON)

El registro de control del USB UCON contiene los bits necesarios para dirigir el comportamiento del módulo durante las transferencias. El registro contiene los bits que gobiernan lo siguiente:

- Permiso del periférico principal del USB
- Reset de los punteros tipo ping-pong
- Control del modo al suspender
- Desactivar la transferencia de paquetes

Además, el registro de control del USB contiene un bit de estado, SE0 (UCON<5>), que se utiliza para indicar el estado del bus, si sólo se manda un cero. Cuando se permite el módulo USB, este bit debe supervisarse para determinar si las líneas de datos han salido de una condición de single-ended cero (sólo se manda cero). Esto ayuda a distinguir el estado de ciclo inicial de la señal de reset del USB.

La operación total del módulo USB se controla con el bit USBEN (UCON<3>). Activar este bit setea el módulo y resetea todos los bits PPBI en el Buffer a '0'. Este bit también activa el regulador de tensión del chip y conecta las resistencias pull-up, si se permiten. Así, este bit puede utilizarse como una unión/separación al USB. Aunque se ignoran todos los estados y bits de control si este bit está borrado, el módulo necesita preconfigurarse antes de activar este bit.

El bit PPBRST (UCON<6>) controla el estado del reset cuando se utiliza el modo del Doble-Buffering (buffer ping-pong). Cuando se activa el bit PPBRST, todos los buffers ping-pong se fijan a los buffers intermedios. El PPBRST tiene que borrarse por firmware. Este bit se ignora en los modos de buffer que no usen el buffer ping-pong.

El bit PKTDIS (UCON<4>) es un flag que indica si el SIE ha inhabilitado la transmisión y la recepción de paquetes. Este bit activa el SIE cuando recibe un SETUP para permitir su procesamiento. Este bit no lo puede activar el microcontrolador, sólo borrar; al despejarlo el SIE continua la transmisión y/o recepción. Cualquier acontecimiento pendiente dentro de buffer intermedio todavía estará disponible, indicado dentro del registro USTAT en el buffer FIFO.

El bit RESUME (UCON<2>) permite al periférico realizar un reinicio ejecutando la señal resume. Para generar un reinicio válido, por firmware se debe activar el RESUME durante 10ms y entonces borrar el bit.

El bit SUSPND (UCON<1>) coloca el módulo y soporte del circuito (es decir, regulador de tensión) en un modo de baja potencia. El reloj de entrada al SIE se desactiva. Este bit debe activarse por software dentro de la respuesta a una interrupción IDLEIF. Debe borrarse por firmware después de observar una interrupción ACTVIF. Cuando este bit está activo, sigue estando el dispositivo unido al bus pero las salidas del transmisor-receptor permanecen en reposo. La tensión en el pin V_{USB} puede variar

dependiendo del valor de este bit. Activar este bit antes de un IDLEIF dará lugar a comportamiento imprevisible del bus.

Nota: Cuando esté en el modo de suspensión, el bus del dispositivo USB se limita al 500 μ A de corriente. Ésta es la corriente completa cedida por el dispositivo PIC y su circuito de soporte. Hay que tener cuidado de ceder la corriente mínima cuando el dispositivo entre en el modo de suspensión.

REGISTRO 17-1: UCON: REGISTRO DE CONTROL DEL USB

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
-	PPBRST	SE0	PKTDIS	USBEN	RESUME	SUSPND	-
U-0	R/W-0	R-x	R/C-0	R/W-0	R/W-0	R/W-0	U-0
BIT 6	PPBRST:				Bit de protección contra el ping-pong de datos: 1 = Reset todos los punteros del buffer ping-pong 0 = Buffer ping-pong no se resetean		
BIT 5	SE0:				Flag de Single-Ended cero: 1 = Single-ended cero activo en el bus USB 0 = No se ha detectado single-ended		
BIT 4	PKTDIS:				Bit de inhabilitación de transferencias de paquetes: 1 = SIE y procesamiento de paquetes desactivado, se activa automáticamente cuando recibe un SETUP 0 = SIE y procesamiento de paquetes permitido		
BIT 3	USBEN:				Bit de permiso del módulo USB 1 = módulo USB y soporte del circuito activados (dispositivo unido) 0 = módulo USB y soporte del circuito desactivados (dispositivo separado)		
BIT 2:	RESUME:				Bit de permiso de la señal RESUME 1 = Señal RESUME activada 0 = Señal RESUME desactivada		
BIT 1	SUSPND:				Bit para suspender el USB 1 = módulo del USB y el soporte del circuito están en el modo de conservación de energía, reloj SIE inactivo 0 = módulo USB y el soporte del circuito están operando normalmente, reloj SIE controlado con el ratio asignado		

17.2.2 REGISTRO DE CONFIGURACIÓN DEL USB (UCFG)

Antes de comunicarse con el USB, se tiene que configurar el módulo del hardware interno y/o externo. La mayor parte de la configuración se realiza con el registro UCFG. El regulador de tensión del USB se maneja con la configuración de los registros.

El registro UFCG contiene la mayor parte de los bits que dirigen el comportamiento del módulo USB. Éstos incluyen:

- Velocidad del bus (“velocidad completa” contra “poca velocidad”)
- Permiso de las resistencias pull-up del chip
- Permiso del transmisor del chip
- Uso del buffer ping-pong

El registro UCFG también contiene dos bits que ayudan a probar el módulo, eliminando errores y certificaciones del USB. La salida de control de estos bits permite el monitor de estado y generación de patrones “de ojo”.

Nota: La velocidad del USB, la transmisión y las pull-up deben configurarse solamente durante la fase de activación del módulo. No se recomienda cambiar estos ajustes mientras que el módulo esté funcionando.

REGISTRO 17-2: UCFG: REGISTRO DE CONFIGURACIÓN DEL USB

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
UTEYE	UOEMON ⁽¹⁾	-	UPUEN ^(2,3)	UTRDIS ⁽²⁾	FSEN ⁽²⁾	PPB1	PPB0
R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
BIT 7	UTEYE:						
			Bit de permiso del test patrón “de ojo” del USB				
			1 = prueba del patrón “de ojo” permitido				
			0 = prueba del patrón “de ojo” inhabilitado				
BIT 6	UOEMON:						
			Bit de permiso del monitor OE* del USB ⁽¹⁾				
			1 = señal del UOE* activa; indica los intervalos durante				
			los cuales las líneas D+/D- están conduciendo				
			0 = señales del UOE* inactivas				
BIT 4	UPUEN:						
			Bit de permiso del pull-up del chip ^(2,3)				
			1 = Pull-up del chip permitido (pull-up en D+ con				
			FSEN=1 o en D- con FSEN=0)				
			0 = Pull-up del chip desactivado				
BIT 3	UTRDIS:						
			Bit inhabilitador del transmisor del chip ⁽²⁾				
			1 = Transmisor del chip inhabilitado; la interfaz digital				
			del transmisor permitida				
			0 = transmisor del chip activo				
BIT 2	FSEN:						
			Bit de permiso del Full-Speed ⁽²⁾				
			1 = dispositivo Full-speed: controla los flancos del				
			transmisor; requiere un reloj de 48MHz				
			0 = dispositivo de poca velocidad: controla los flancos				
			del transmisor; requiere un reloj de 6MHz				
BIT 1-0	PPB1:PPB0:						
			Bits de configuración de los buffer ping-pong				
			11 = buffers ping-pong permitidos en los End-Points				
			1 a 15				
			10 = buffers ping-pong permitidos en todos los End-				
			Points				
			01 = buffer ping-pong permitido en los End-Points				
			de salida 0				
			00 = buffers ping-pong inhabilitados				

Nota 1: Si se activa UTRDIS, la señal UOE* estará activa independiente del ajuste del bit UOEMON.

2: Los bits UPUEN, UTRDIS y FSEN no deben cambiarse mientras el módulo USB esté permitido. Estos valores se deben preconfigurar antes de permitir el módulo.

3: Este bit solamente es válido cuando el transmisor del chip está activo (UTRDIS = 0); si no, se ignora.

17.2.2.1 Transmisor interno

El periférico USB tiene, USB 2.0, full-speed y un transmisor de poca velocidad, internamente conectado al SIE. Esta característica es útil en aplicaciones baratas con un chip. El bit UTRDIS (UCFG<3>) controla el transmisor; está permitido por defecto (UTRDIS = 0). El bit FSEN (UCFG<2>) controla la velocidad del transmisor; activar el bit permite las operaciones “full-speed”.

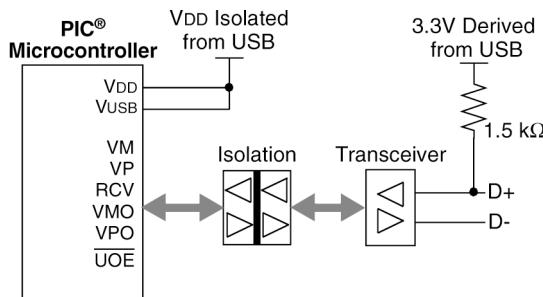
Las resistencias pull-up del chip del USB se controlan con el bit UPUEN (UCFG<4>). Pueden seleccionarse solamente cuando se permite el transmisor del chip.

Las especificaciones del USB requieren 3,3V en las comunicaciones; sin embargo, el resto del chip puede funcionar con una tensión más alta. Así, la fuente de energía del transmisor es una fuente separada, V_{USB}.

17.2.2.2 Transmisor externo

Este módulo proporciona ayuda para el uso de un transmisor fuera del chip. El transmisor externo se utiliza en aplicaciones donde las condiciones físicas dictan la localización del transmisor lejos del SIE. Por ejemplo, aplicaciones que requieren el aislamiento del USB podía utilizar un transmisor externo con algunos aislamientos en el SIE del microcontrolador. Las operaciones con un transmisor externo se permiten activando el bit UTRDIS.

FIGURA 17-2: TRANSMISOR EXTERNO CON AISLAMIENTOS



Hay 6 señales en el módulo para comunicar y controlar un transmisor externo:

- VM: Entrada de la línea single-ended D-
- VP: Entrada de la línea single-ended D+
- RCV: La entrada del receptor diferencial
- VMO: Salida a la línea diferencial del conductor
- VPO: Salida a la línea diferencial del conductor
- UOE: Salida permitida

Las señales VPO y VMO son salidas del SIE al transmisor externo. La señal del RCV es la salida del transmisor externo al SIE; representa las señales diferenciales del bus serie traducido en un tren de pulsos. Las señales VM y VP se utilizan para divulgar condiciones en el bus serie al SIE que no se puede capturar con la señal del RCV.

La señal UOE* indica el estado del transmisor externo. A esta línea la pone en nivel bajo el dispositivo para activar las transmisiones de datos del SIE con un dispositivo externo.

TABLA 17-1: SALIDAS DIFERENCIALES AL TRANSMISOR

VPO	VMO	Estado del bus
0	0	Single-Ended Cero
0	1	Diferencial '0'
1	0	Diferencial '1'
1	1	Condición Ilegal

TABLA 17-2: ENTRADAS SINGLE-ENDED DEL TRANSMISOR

VP	VM	Estado del Bus
0	0	Single-Ended Cero
0	1	Baja velocidad
1	0	Toda velocidad
1	1	Error

17.2.2.3 Resistencias internas pull-up

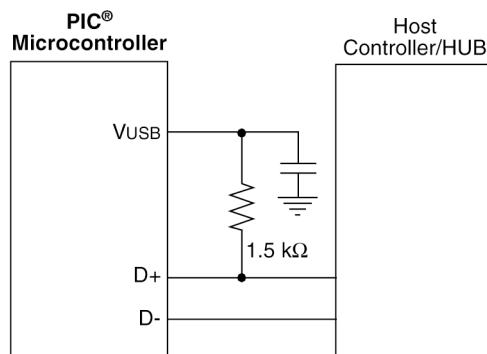
Los dispositivos PIC18FX455/X550 tienen unas resistencias pull-up internas diseñadas para conocer los requerimientos del USB “baja velocidad” y “toda velocidad”. El bit UPUEN (UCFG<4>) activa las pull-up internas.

17.2.2.4 Resistencias externas pull-up

Se pueden necesitar resistencias externas pull-up. El pin V_{USB} se puede utilizar para aumentar D+ o D-. La resistencia pull-up tiene que ser de 15kΩ.

FIGURA 17-3: CIRCUITO EXTERNO

El diseño muestra una aplicación con USB “a toda velocidad” utilizando el regulador interno y una resistencia pull-up externa.



17.2.2.5 Permiso de la salida del monitor del USB

El monitor del USB OE* proporciona información de la operación del SIE si está escuchando el bus o está conduciendo por el bus. Esto se permite por defecto al usar un transmisor externo o cuando UCFG<6>=1. La supervisión del USB OE* es útil para el eliminar errores del sistema inicial.

17.2.2.6 Permiso del test patrón “de ojo”

Se genera una prueba automática al activar el bit UCFG<7> del patrón “de ojo”. La salida del patrón “de ojo” dependerá de los ajustes del módulo, significando que el usuario es el primer responsable de configurar los ajustes del reloj SIE, las resistencias pull-up y el modo transmisor. Además, el módulo tiene que estar permitido.

Una vez que se active UTEYE, el módulo emula un cambio a una recepción para transmitir el estado y comenzará a transmitir una secuencia de bits J-K-J-K (K-J-K-J en “velocidad completa”). La secuencia se repetirá indefinidamente mientras que el test patrón “de ojo” esté permitido.

Observar que este bit no se debe activar mientras que el módulo está conectado con un sistema real USB. Este modo de prueba se utiliza para ayudar con las verificaciones internas de las pruebas de certificación USB. Se utiliza para depurar los fallos obtenidos por el ruido de las señales que pueden afectar los flancos del sistema, uniones de impedancias mal hechas y proximidad a otros componentes. No comprueba correctamente la transición de un estado de recepción a uno de transmisión. Aunque el patrón de ojo no significa que se sustituya la prueba más compleja de certificación del USB, pero ayuda durante el primer test para eliminar errores del sistema.

17.2.2.7 Regulador interno

Los dispositivos PIC18FX455/X550 tienen un regulador incorporado 3,3V para proporcionar energía al transmisor interno y proporcionar una fuente para el pull-up interno y externo. Para que el regulador sea estable se necesita un condensador externo de 220nF ($\pm 20\%$).

Nota: La corriente de V_{USB} es suficiente para controlar un pull-up externo y el transmisor interno.

El regulador se activa por defecto y puede inhabilitarse a través del bit de configuración VREGEN. Cuando está activo, la tensión es visible en el pin V_{USB} . Cuando el regulador está desactivado, se tiene que conectar una fuente de 3,3V al pin V_{USB} para alimentar el transmisor interno. Si el transmisor interno no se utiliza, V_{USB} tampoco se activa.

Nota 1: No permitir el regulador interno si el regulador externo está conectado a V_{USB} .

2: V_{DD} debe ser mayor que V_{USB} en cualquier momento, o igual con el regulador inhabilitado.

17.2.3 REGISTRO DE ESTADO DEL USB (USTAT)

El registro de estado del USB divulga el estado de las transacciones dentro del SIE. Cuando el SIE publica una interrupción de transferencia completa por el USB, hay que leer USTAT para determinar el estado de la transferencia. USTAT contiene el número del Endpoint de la transferencia, dirección y valor del puntero del buffer ping-pong (si está utilizado).

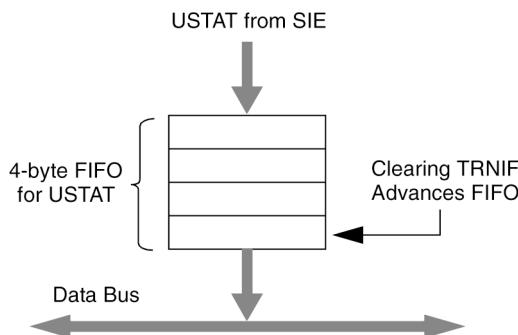
Nota: Los datos en el registro de estado del USB son válidos solamente cuando el flag de interrupción TRNIF está activo.

El registro USTAT es realmente una ventana legible de los cuatro bytes de estado FIFO mantenida por el SIE. Permite al microcontrolador procesar una transferencia mientras que SIE procesa los Endpoints adicionales. Cuando el SIE termina con un buffer de lectura o escritura de datos, actualiza el registro USTAT. Si se realiza otra transferencia USB antes de realizar una interrupción de transacción completa, el SIE almacenará el estado de la transferencia siguiente en el estado FIFO.

Despejando el flag de la transferencia completa, TRNIF, provoca al SIE que avance el FIFO. Si los datos siguientes en el registro FIFO son válidos, el SIE inmediatamente reafirma la interrupción. Si no hay datos adicionales presentes, TRNIF seguirá borrado; los datos de USTAT puede que no sean correctos.

Nota: Si se recibe una petición de Endpoint mientras que el USTAT FIFO este lleno, el SIE publica automáticamente un NAK de nuevo al anfitrión.

FIGURA 17-4: USAT FIFO



REGISTRO 17-3: USTAT: REGISTRO DE ESTADO DEL USB

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
-------	-------	-------	-------	-------	-------	-------	-------

	-	ENDP3	ENDP2	ENDP1	ENDP0	DIR	PPBI ⁽¹⁾	-
U-0	R-x	R-x	R-x	R-x	R-x	R-x	U-0	
BIT 6-3	ENDP3:ENDP0: Bits de actividad del número codificado del último Endpoint (representa el número del BDT actualizado por la última transferencia del USB)							
	1111 = Endpoint 15 1110 = Endpoint 14 0001 = Endpoint 1 0000 = Endpoint 0							
BIT 2	DIR: Bit puntero de la última dirección BD 1 = la última transacción era de entrada 0 = la última transacción era de salida o de SETUP							
BIT 1	PPBI: Bit puntero del puntero ping-pong BD ⁽¹⁾ 1 = la transacción pasada estaba al banco Odd de BD 0 = la transacción pasada estaba al banco Even de BD							

Nota 1: Este bit es solamente válido para los Endpoints con registros Even y Odd BD disponibles.

17.2.4 CONTROL DE LOS ENDPOINTS DEL USB

Cada uno de los 16 Endpoints bidireccionales posibles tiene un registro de control independiente, UEPn (donde ‘n’ representa el número del Endpoint). Cada registro tiene los mismos bits de control.

El bit EPHSHK (UEPn<4>) controla el protocolo de intercambio de los Endpoints; activar este bit permite el protocolo de intercambio del USB. Típicamente, este bit se setea siempre excepto al usar Endpoints síncronos.

El bit EPCONDIS (UEPn<3>) se utiliza para permitir o inhabilitar las operaciones de control del USB (SETUP) con Endpoint. Borrar este bit permite las transacciones SETUP. Observar que los bits EPINEN y EPOUTEN se deben activar para permitir las transacciones de entrada y de salida. Para el Endpoint 0, este bit debe estar siempre borrado por que las especificaciones del USB identifican el Endpoint 0 como el Endpoint de control por defecto.

El bit EPOUTEN (UEPn<2>) se utiliza para permitir o para inhabilitar las transacciones de salida del anfitrión. Activar este bit permite transacciones de salida. Semejantemente, el bit EPINEN (UEPn<1>) permite o inhabilita las transacciones de entrada al anfitrión.

El bit EPSTALL (UEPn<0>) se utiliza para indicar la condición STALL para el Endpoint. Si se ejecuta una STALL en un Endpoint particular, el bit EPSTALL para ese Endpoint lo activará el SIE. Este bit permanece hasta que se borre por software o hasta que ocurra un reset del SIE.

REGISTRO 17-4: UEPn: REGISTRO DE CONTROL DEL ENDPOINT n DEL USB (UEP0 HASTA UEP15)

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
-	-	-	EPSHK	EPCONDIS	EPOUTEN	EPINEN	EPSTALL ⁽¹⁾
U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
BIT 4	EPHSHK:		Bit de permiso del protocolo de intercambio 1= se permite el protocolo de intercambio del Endpoint 0= se inhabilita (utilizado con los Endpoints síncronos)				
BIT 3	EPCONDIS:		Bit bidireccional de control del Endpoint Si EPOUTEN=1 y EPINEN=1: 1= inhabilita las transferencias de control del Endpoint n; sólo permite las de entrada y las de salida 0= permite las transferencias de control (SETUP) y las de entrada y salida del Endpoint n				
BIT2	EPOUTEN:		Bit de permiso de la salida del Endpoint 1= la salida del Endpoint n permitida 0= la salida del Endpoint n inhabilitada				
BIT 1	EPINEN:		Bit de permiso de la entrada del Endpoint 1= la entrada del Endpoint n permitida 0= la entrada del Endpoint n inhabilitada				
BIT 0	EPSTALL:		Bit de permiso de la parada del Endpoint ⁽¹⁾ 1= se puede parar el Endpoint n 0= el Endpoint n no se puede parar				

Nota 1: Válido solamente si se permite el Endpoint n; si no, se ignora el bit.

17.2.5 REGISTRO DE DIRECCIÓN DEL USB (UADDR)

El registro de dirección del USB contiene la única dirección del USB que el periférico descifra cuando está activo. UADDR se pone a 00h cuando recibe un reset del USB, indicado por URSTIF, o con un reset al microcontrolador. La dirección del USB la tiene que escribir el microcontrolador durante la fase de setup del USB como parte del firmware del USB de la ayuda de microchip.

17.2.6 REGISTROS DEL NÚMERO DEL FRAME DEL USB (UFRMH: UFRML)

Los registros del número del frame contienen los 11bits del número del frame. El byte de orden inferior está en UFRML, mientras que los tres bits de orden superior permanecen en UFRMH. El par de registros se actualizan con el número de frame actual cuando recibe un SOF. Para el microcontrolador, estos registros son sólo de lectura. El registro del número del frame se utiliza sobre todo para transferencias síncronas.

17.3 RAM DEL USB

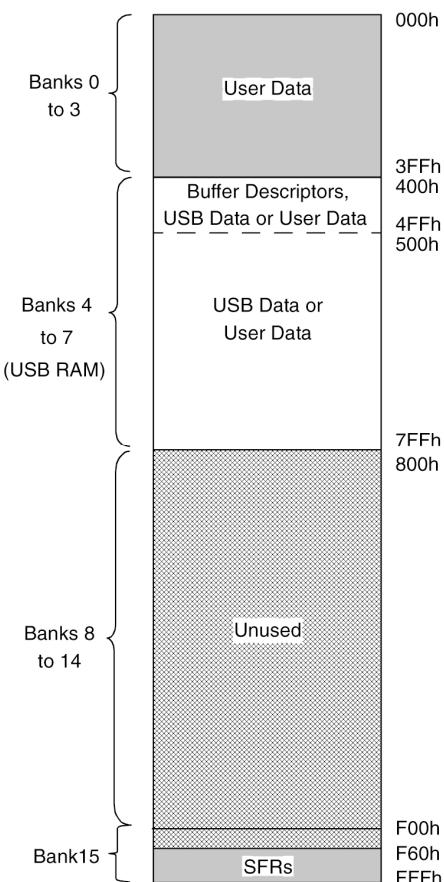
Los datos del USB se mueven entre el núcleo del microcontrolador y el SIE a través de una memoria

conocida como USB RAM. Ésta es una memoria de puerto dual especial que está mapeada en la memoria normal de datos en los bancos de 4 a 7 (400h a 7FFh) para un total de 1kbyte.

El banco 4 (400h con 4FFh) se utiliza específicamente en el control del buffer del Endpoint, mientras que los bancos de 5 a 7 están disponibles para los datos del USB. Dependiendo del tipo de buffer que se utilice, los 8 bytes del banco 4 pueden estar disponibles para utilizarlos como buffer del USB.

Aunque la RAM del USB está disponible en el microcontrolador como memoria de datos, las secciones que está modificando el SIE no las puede usar el microcontrolador. Se utiliza un mecanismo de semáforos para determinar el acceso a un buffer en un momento dado.

FIGURA 17-5: IMPLEMENTACIÓN DE LA RAM DEL USB EN LA ESPACIO DE MEMORIA DE DATOS



17.4 BUFFER DESCRIPTOR Y TABLA DEL BUFFER DESCRIPTOR

Los registros del banco 4 se utilizan específicamente para el control del buffer del Endpoint en una estructura conocida como Tabla del Buffer descriptor (BDT). Esto proporciona un método flexible a los usuarios para construir y para controlar los Buffers del Endpoint de varias longitudes y configuraciones.

Los BDT se componen de los Buffers descriptores (BD) que se utilizan para definir y controlar los Buffers reales del Espacio de la RAM del USB. Cada BD, alternadamente, consiste en cuatro registros, donde n representa uno de los 64 BDs posibles (gama de 0 a 63):

- BDnSTAT: Registro de estado de BD
- BDnCNT: Registro del byte de cuenta de BD
- BDnADRL: Registro bajo de la dirección de BD
- BDnADRH: Registro alto de la dirección de BD

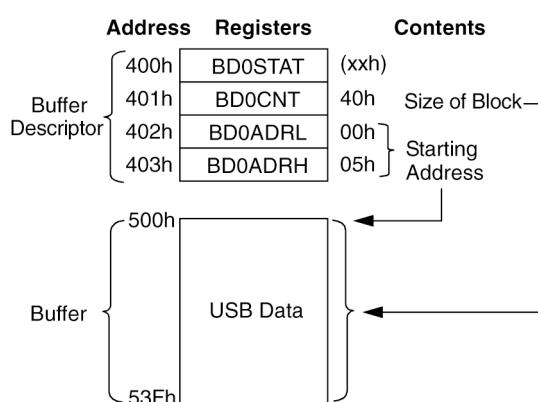
BDs ocurre siempre como bloque de cuatro bytes en la secuencia, BDnSTAT:BDnCNT:BDnADRL:BDnADRH. La dirección de BDnSTAT es siempre una compensación de (4n-1) (en hexadecimal) de 400h, con n como el número del Buffer descriptor.

Dependiendo de la configuración del buffering utilizada, hay 32, 33 ó 64 sistemas de Buffer descriptores. El BDT debe ser por lo menos 8 bytes de largo. Esto se debe a que la especificación del USB asigna que por mandato debe tener cada dispositivo el Endpoint0 configurado como entrada y salida en la disposición inicial. Dependiendo del Endpoint y de la configuración buffering, el BDT puede ser de 256 bytes de largo.

Aunque se puede creer que los buffers descriptores y los registros de dirección son registros de funciones especiales que no están mapeados en hardware, como lo están los SFRs convencionales en el banco 15 del microcontrolador. Si el Endpoint correspondiente a un BD particular no está permitido, sus registros no se utilizan. En vez de aparecer como direcciones no implementadas, aparecen como RAM disponible. Solamente cuando un Endpoint está permitido, activando el bit UEPn<1>, se consigue que la memoria de esas direcciones funcione como BD. Como los registros BD tienen cualquier dirección en la memoria de datos en un reset, también tienen un valor indeterminado.

Un sistema particular de los registros BD son solamente válidos si el Endpoint correspondiente está permitido usando el registro UEPn. Todos los registros BD están disponibles en la RAM del USB. El BD para cada Endpoint debe configurarse antes de permitirlo.

FIGURA 17-6: EJEMPLO DE UN BUFFER DESCRIPTOR



17.4.1 ESTADO Y CONFIGURACIÓN DE LOS BD

Los Buffer descriptores no sólo definen el tamaño de un Buffer Endpoint, sino también determina su configuración y control. La mayor parte de la configuración se hace con el registro de estado del BD, BDnSTAT. Cada BD tiene su propio registro correspondientemente numerado BDnSTAT.

No como otros registros de control, la configuración de los bits del registro BDnSTAT depende del contexto. Hay dos configuraciones distintas, dependiendo de si el microcontrolador o el módulo del USB está modificando el BD y Buffer en un momento dado. Solamente se comparten tres definiciones de bit entre los dos.

17.4.1.1 Propiedad del Buffer

Porque los Buffers y su BDs los comparten la CPU y el módulo del USB, se utiliza un semáforo para distinguir el BD y los Buffers asociados en memoria que se permiten actualizar.

Esto se logra con el bit UOWN (BDnSTAT<7>). UOWN es el único bit compartido entre las dos configuraciones BDnSTAT.

Cuando UOWN está borrado, la entrada de BD la dirige por el núcleo del microcontrolador. Cuando se activa el bit UOWN, la entrada del BD y la memoria del Buffer los controla por el periférico USB. El núcleo no debe modificar el BD o su

Buffer correspondiente en este momento. Observar que el núcleo del microcontrolador puede leer BDnSTAT mientras que el SIE controla el Buffer y viceversa.

Los Buffer descriptores tienen varios significados dependiendo de la fuente de actualización del registro. Antes de poner en sus manos el periférico del USB, el usuario puede configurar las operaciones básicas del periférico con los bits BDnSTAT. Durante este tiempo, el byte de control de la cuenta y los registros de direccionamiento del Buffer también pueden fijarse.

Cuando se activa UOWN, el usuario puede depender de los valores escritos en los BDs. El SIE actualiza el BDs cuanto es necesario, sobrescribiendo los valores originales del BD. El registro BDnSTAT lo actualiza el SIE con el PID; la cuenta de la transferencia, BDnCNT, se actualiza también.

El byte BDnSTAT del BDT debe ser el último byte que se actualice al prepararse para armar un Endpoint. El SIE borrará el bit UOWN cuando se termine una transacción. La única excepción es cuando KEN y/o BSTALL están permitidos.

No existe ningún mecanismo por hardware para bloquear el acceso cuando se setea el bit UOWN. Así, puede ocurrir un comportamiento inesperado si el microcontrolador intenta modificar la memoria cuando el SIE lo posee. Semejantemente, leyendo tal memoria se pueden obtener datos inexactos hasta que el periférico USB devuelve la propiedad al microcontrolador.

17.4.1.2 Registro BDnSTAT (modo CPU)

Cuando UOWN=0, el núcleo del microcontrolador posee BD. En este punto, los otros siete bits del registro toman las funciones de control.

El bit, KEN (BDnSTAT<5>), determina si un BD permanece activo. Si se setea el bit, una vez que el bit UOWN esté activo, seguirá controlado el SIE independiente de la actividad del Endpoint. Esto previene a la FIFO USTAT de actualizarse, así como activar la interrupción de transacción completa para el Endpoint. Esta característica se debe permitir solamente cuando el puerto paralelo se selecciona como canal de entrada-salida de datos en lugar de la RAM del USB.

El bit inhabilita el incremento de la dirección, INCDIS (BDnSTAT<4>), que controla el direccionamiento automático por incremento del SIE. Activar INCDIS inhabilita el auto incremento de la dirección del Buffer por el SIE para cada byte transmitido o recibido. Esta característica sólo se tiene que utilizar con el puerto paralelo, donde cada byte de datos se procesa a/desde la misma posición de memoria.

El bit de permiso de la sincronización de palabras, DTSEN (BDnSTAT<3>), se encarga de comprobar la paridad de los datos. Activar DTSEN permite la sincronización con el SIE. Cuando está permitido, comprueba la paridad del paquete de los datos contra el valor de DTS (BDnSTAT<6>). Si un paquete llega con una sincronización incorrecta, los datos se ignoran. No se escriben en la RAM del USB y el flag de interrupción de transferencia completa del USB no se activará. Sin embargo, el SIE enviará un ACK al anfitrión para reconocer el recibo.

El bit del buffer de parada, BSTALL (BDnSTAT<2>), proporciona ayuda en el control de las transferencias, generalmente una parada en el Endpoint 0. También proporciona ayuda con los comandos SET_FEATURE/CLEAR_FEATURE; típicamente, paradas continuas en cualquier Endpoint con excepción del Endpoint de control por defecto.

El bit BSTALL también permite las paradas del Buffer. Activar BSTALL hace que el SIE devuelva un STALL al anfitrión si el símbolo recibido utilizaría el BD en esa localización. Cuando se activa el bit EPSTALL en el registro de control correspondiente UEPn se genera una interrupción STALL cuando se manda la STALL al anfitrión. El

bit UOWN activado y el BDs no se cambia a menos que se reciba un SETUP. En este caso, la condición de la STALL se borra y la propiedad del BD se devuelve al núcleo del microcontrolador.

Los bits BD9:BD8 (BDnSTAT<1:0>) guardan los dos dígitos más significativos del byte de la cuenta del SIE; los 8 dígitos más bajos se almacenan en el registro correspondiente BDnCNT.

TABLA 17-3: EFECTO DEL BIT DTSEN EN LA RECEPCIÓN DE PAQUETES PARES/IMPARES (DATA0/DATA1)

Salida del paquete del Host	Configuración BDnSTAT		Respuesta del dispositivo después de recibir el paquete			
	DTSEN	DTS	Protocolo	UOWN	TRNIF	BDnSTAT y USTAT Estado
DATA0	1	0	ACK	0	1	Actualizado
DATA1	1	0	ACK	1	0	No Actualizado
DATA0	1	1	ACK	0	1	Actualizado
DATA1	1	1	ACK	1	0	No Actualizado
Otro	0	x	ACK	0	1	Actualizado
Otro, con error	x	x	NAK	1	0	No Actualizado

REGISTRO 17-5: BDnSTAT: REGISTRO DEL ESTADO DEL BUFFER DESCRIPTOR n (BD0STAT HASTA BD63STAT), MODO CPU (LOS DATOS SE ESCRIBEN AL LADO)

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
UOWN ⁽¹⁾	DTS ⁽²⁾	KEN	INCDIS	DTSEN	BSTALL	BC9	BC8
R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
BIT 7	UOWN:	Bit de posesión del USB ⁽¹⁾ 0 = el núcleo del microcontrolador posee el BD y su Buffer correspondiente					
BIT 6	DTS:	Bit de sincronización de los datos ⁽²⁾					

		1 = paquete de datos 1 0 = paquete de datos 0
BIT 5	KEN:	Bit de permiso de la subsistencia de BD 1 = USB guardará el BD indefinidamente una vez que UOWN se active (requerido en la configuración de los Endpoint del SPP) 0 = USB guardará el último símbolo procesado
BIT 4	INCDIS:	Bit de inhabilitación del incremento de la dirección 1 = el incremento de la dirección inhabilitado (requerido en la configuración de los Endpoint del SPP) 0 = incremento de la dirección permitido
BIT 3	DTSEN:	Bit de permiso de la sincronización 1 = sincronización de los datos permitida; los paquetes de los datos con valor incorrecto de la sinc. se ignoran excepto un SETUP, que se acepta 0 = ninguna sincronización de los datos
BIT 2	BSTALL:	Bit de permiso de paradas en el Buffer 1 = parada del Buffer permitida; el protocolo de la STALL publica si se recibe un símbolo que utilizaría el BD en la localización dada (UOWN se activa, el resto sin cambios) 0 = parada del Buffer inhabilitado
BIT 1-0	BC9:BC8:	Bits del byte de cuenta 9 y 8 Los bits de cuenta del byte representan el número de bytes que se transmitirán con un símbolo IN o recibidos durante un símbolo OUT. Junto con BC<7:0>, las cuentas de byte válidas son 0-1023.

Nota 1: Este bit debe inicializarlo el usuario con el valor deseado antes de permitir el módulo USB.

2: Se ignora este bit a menos que DTSEN = 1.

17.4.1.3 Registros BDnSTAT (modo SIE)

Cuando los BDs y su Buffer los gobierna el SIE, la mayoría de los bits de BDnSTAT toman distintos significados. Al activarse UOWN, cualquier dato o ajuste de control escritos por el usuario se sobrescriben con datos del SIE.

El registro BDnSTAT lo actualiza el SIE con el identificador del paquete (PID) se almacena en BDnSTAT<5:3>. Se actualiza la cuenta de la transferencia en el registro BDnCNT correspondiente. Los valores que desbordan el registro de 8 bits se transportan a los dos dígitos más significativos de la cuenta, almacenados en BDnSTAT<1:0>.

REGISTRO 17-6: BDnSTAT: REGISTRO DE ESTADO DEL BUFFER DESCRIPTOR n (BD0STAT A BD63STAT), MODO SIE (DATOS DEVUELtos POR EL LADO DEL MICROCONTROLADOR)

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
UOWN	-	PID3	PID2	PID1	PID0	BC9	BC8
R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
BIT 7	UOWN:		Bit de posesión del USB 1= el SIE gobierna el BD y el buffer correspondiente				
BIT 5-2	PID3:PID0:		Bits identificadores del paquete El valor recibido del símbolo PID de la última transferencia (sólo IN, OUT o SETUP)				
BIT 1-0	BC9:BC8:		Bits del byte de cuenta 9 y 8 Esto bits los actualiza el SIE para reflejar al número de bytes recibidos en una transferencia OUT y el número de bytes transmitidos en una IN.				

17.4.2 BYTE DE CUENTA DE BD

El byte de cuenta representa el número total de bytes que se transmitirán durante una IN. Después de la transferencia IN, el SIE devolverá el número de bytes enviados al anfitrión.

Para una transferencia OUT, el byte de cuenta representa número máximo de los bytes que se pueden recibir y almacenar en la RAM del USB. Después de una transferencia OUT, el SIE devolverá el número real de bytes recibidos. Si este número excede el byte de cuenta correspondiente, el paquete de datos se rechazará y se generará un protocolo de intercambio NAK. Cuando sucede esto, el byte de cuenta no se actualiza.

El byte de cuenta de 10 bits se distribuye sobre dos registros. Los 8 bits más bajos de la cuenta residen en el registro BDnCNT. Los dos altos en BDnSTAT<1:0>. Esto representa una gama válida para el byte de 0 a 1023.

17.4.3 VALIDACIÓN DE LA DIRECCIÓN DE BD

El par de registros de dirección de BD contiene la dirección de comienzo de la RAM para el Buffer del Endpoint correspondiente. Para que una localización que comienza en el Endpoint sea válida, debe estar en la gama de la RAM del USB, 400h a 7FFh. No hay ningún mecanismo por hardware para comprobar la dirección del BD.

Si el valor de la dirección de BD no señala a una dirección de la RAM del USB, o si señala a una dirección dentro del Buffer de otro Endpoint, es probable que se pierdan los datos o que se sobrescriban. Semejantemente, solapando un Buffer de recepción (Endpoint de salida) con una localización de BD en uso se obtienen resultados inesperados. Cuando se desarrollan aplicaciones USB, el usuario puede incluir software para validar las direcciones en el código.

17.4.4 BUFFERING PING-PONG

Un Endpoint se define para tener un Buffer ping-pong cuando tiene dos sistemas de entradas de BD: un sistema para una transferencia par y otro para una transferencia impar. Esto permite a la CPU procesar un BD mientras que el SIE procesa el otro BD. El doble buffering BD, permite un rendimiento de procesamiento máximo del/al USB.

El módulo USB apoya cuatro modos de operación:

- Ninguna ayuda del ping-pong
- Ayuda del Buffer del ping-pong del OUT Endpoint 0 solamente
- Ayuda del Buffer del ping-pong para todas los Endpoints
- Ayuda del Buffer del ping-pong para el resto de los Endpoints excepto el 0

Los ajustes del Buffer ping-pong se configuran con los bits PPB1:PPB0 en el registro UCFG.

El módulo USB no pierde de vista el puntero ping-pong de cada Endpoint. Todos los punteros están reseteados inicialmente al BD par cuando se activa el módulo. Al terminar una transacción (SIE borra UOWN), el puntero se une al BD impar. Al terminar la transacción siguiente, el puntero se une de nuevo al BD par y así sucesivamente.

El estado par/impar de la transacción realizada se almacena en el bit PBI del registro USTAT. El usuario puede resetear todos los punteros ping-pong al par con el bit PPBRST.

Cada BD tiene una relación fija con un Endpoint particular, dependiendo de la configuración del buffering. Esta relación significa también que pueden aparecer vacíos en las BDT si los Endpoints no se activan contiguamente. Esto significa en teoría, que los BDs de los Endpoints desactivados podían utilizarse como espacio de Buffer. En la práctica, los usuarios deben evitar usar tales espacios en el BDT a menos que el método de validar direcciones de BD esté en ejecución.

FIGURA 17-7: TABLAS DEL BUFFER DESCRIPTOR MAPEADAS PARA LOS MODOS DE LOS BUFFER

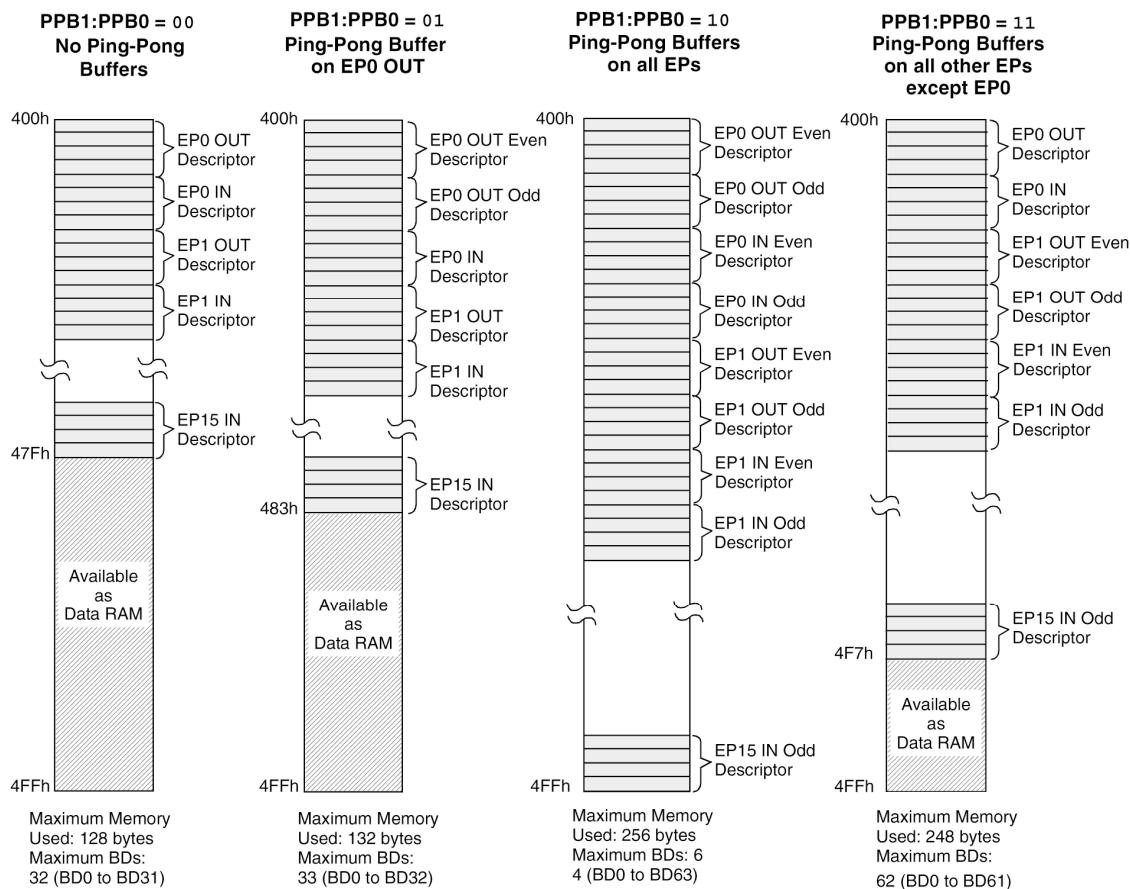


TABLA 17-4: ASIGNACIÓN DE LOS BUFFERS DESCRIPTORES A LOS DIFERENTES MODOS DE BUFFERING

Endpoint	BDs asignados a Endpoint							
	Modo 0(No Ping-Pong)		Modo 1(Ping-Pong en EP0 OUT)		Modo 2(Ping-Pong en todos EPs)		Modo 3(Ping-Pong en todos EPs, excepto EP0)	
	Out	In	Out	In	Out	In	Out	In
0	0	1	0(E), 1(O)	2	0(E), 1(O)	2(E), 3(O)	0	1
1	2	3	3	4	4(E), 5(O)	6(E), 7(O)	2(E), 3(O)	4(E), 5(O)
2	4	5	5	6	8(E), 9(O)	10(E), 11(O)	6(E), 7(O)	8(E), 9(O)
3	6	7	7	8	12(E), 13(O)	14(E), 15(O)	10(E), 11(O)	12(E), 13(O)
4	8	9	9	10	16(E), 17(O)	18(E), 19(O)	14(E), 15(O)	16(E), 17(O)
5	10	11	11	12	20(E), 21(O)	22(E), 23(O)	18(E), 19(O)	20(E), 21(O)
6	12	13	13	14	24(E), 25(O)	26(E), 27(O)	22(E), 23(O)	24(E), 25(O)
7	14	15	15	16	28(E), 29(O)	30(E), 31(O)	26(E), 27(O)	28(E), 29(O)
8	16	17	17	18	32(E), 33(O)	34(E), 35(O)	30(E), 31(O)	32(E), 33(O)
9	18	19	19	20	36(E), 37(O)	38(E), 39(O)	34(E), 35(O)	36(E), 37(O)
10	20	21	21	22	40(E), 41(O)	42(E), 43(O)	38(E), 39(O)	40(E), 41(O)
11	22	23	23	24	44(E), 45(O)	46(E), 47(O)	42(E), 43(O)	44(E), 45(O)
12	24	25	25	26	48(E), 49(O)	50(E), 51(O)	46(E), 47(O)	48(E), 49(O)
13	26	27	27	28	52(E), 53(O)	54(E), 55(O)	50(E), 51(O)	52(E), 53(O)
14	28	29	29	30	56(E), 57(O)	58(E), 59(O)	54(E), 55(O)	56(E), 57(O)
15	30	31	31	32	60(E), 61(O)	62(E), 63(O)	58(E), 59(O)	60(E), 61(O)

Leyenda: (E) =Buffer de transacción par, (O) = Buffer de transacción impar

TABLA 17-5: SUMARIO DE LOS REGISTROS DE LOS BDT DEL USB

Nombre	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
BDnSTAT ⁽¹⁾	UOWN	DTS ⁽⁴⁾	PID3 ⁽²⁾ KEN ⁽³⁾	PID2 ⁽²⁾ INCDIS ⁽³⁾	PID1 ⁽²⁾ DTSEN ⁽³⁾	PID0 ⁽²⁾ BSTALL ⁽³⁾	BC9	BC8
BDnCNT ⁽¹⁾	Byte cuenta							
BDnADRL ⁽¹⁾	Buffer dirección bajo							
BDnADRH ⁽¹⁾	Buffer dirección alto							

Nota 1: En los registros de los BD, la n es un valor de 0 a 63. Los 64 registros son semejantes. Todos tienen valores indeterminados en los reset.

2: Del bit 5 al 2 del BDnSTAT lo utiliza el SIE para devolver los valores PID3:PID0 una vez el registro haya cambiado al SIE (Activar el bit UOWN). Cuando los registros estén bajo el control del SIE, los valores de KEN, DTSEN, INCDIS y BSTALL no tienen validez.

3: Antes de activar el bit UOWN, los bits 5 al 2 del BDnSTAT configuran KEN, DTSEN, INCDIS y BSTALL.

4: Se ignora este bit a menos que DTSEN=1

17.5 INTERRUPCIONES DEL USB

El módulo USB puede generar condiciones de interrupción múltiples. Para acomodar todas estas fuentes de interrupción, el módulo proporciona su propia lógica de estructura de interrupción, similar a la del microcontrolador. Las interrupciones del USB se activan con un sistema de registros de control y registradas con un sistema separado de flags. Todas las fuentes se concentran en una sola petición de interrupción del USB, USBIF (PIR2<5>).

Hay dos capas de registros de interrupción en el módulo USB. El nivel superior consiste en todas las interrupciones de estado del USB; éstos se permiten y se señalan por medio de un flag en los registros UIE y UIR, respectivamente. El segundo nivel consiste en las condiciones de error del USB, se permiten y se señalan por medio de un flag en los registros UEIR y UEIE. Ninguna condición de interrupción en estos provoca la activación del flag de interrupción por error del USB (UERRIF) en el nivel superior.

Las interrupciones se pueden utilizar para detectar acontecimientos rutinarios en una transacción USB.

FIGURA 17-8: EMBUDO DE LA LÓGICA DE LA INTERRUPCIÓN DEL USB

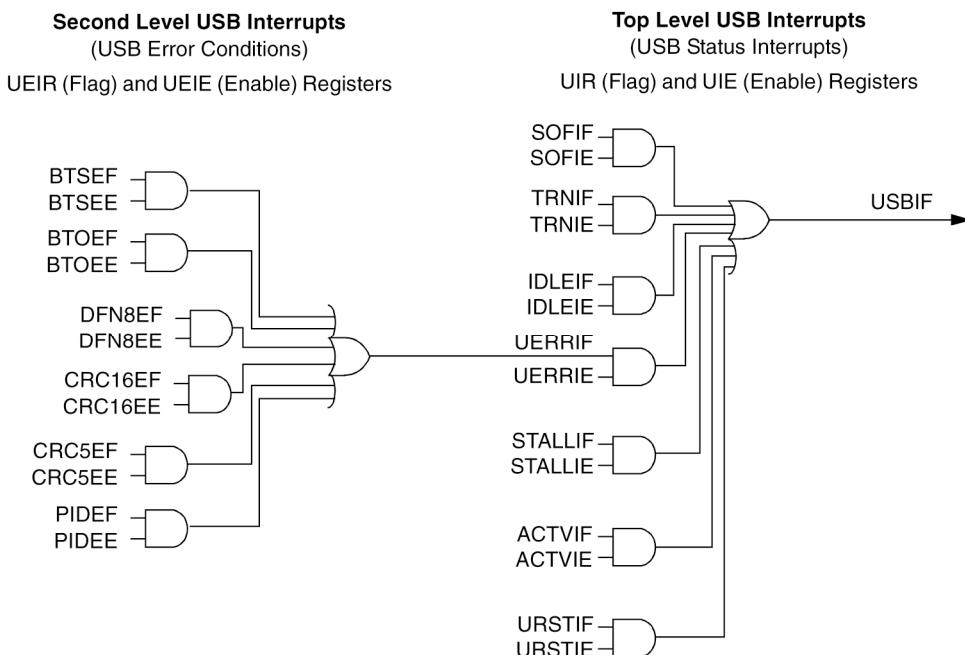
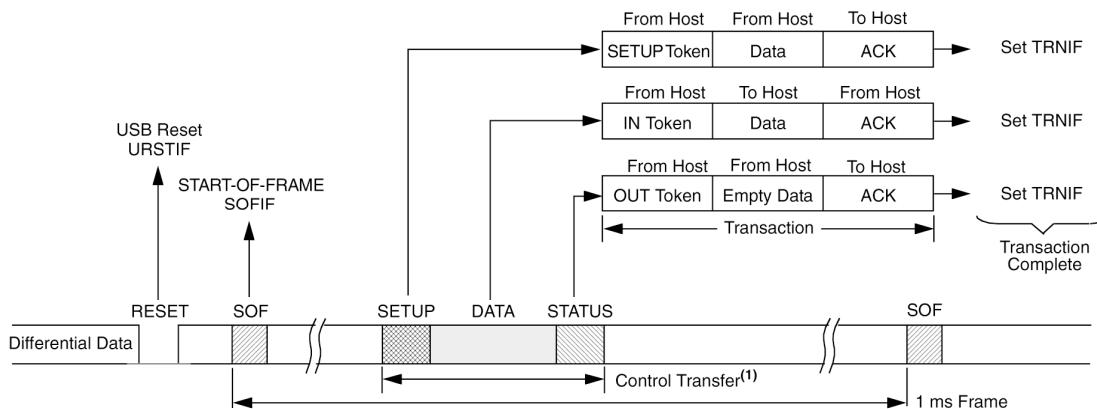


FIGURA 17-9: EJEMPLO DE LOS EVENTOS TRANSACCIÓN E INTERRUPCIÓN

Nota 1: La transferencia de control mostrada es sólo un ejemplo que muestra los eventos que ocurren en cada transacción. El control típico de la transferencia se puede extender a varios frames.

17.5.1 REGISTRO DE ESTADO DE LAS INTERRUPCIONES DEL USB (UIR)

El registro de estado de las interrupciones del USB contiene los flags para el estado de cada fuente de interrupción. Cada una de estas interrupciones tiene un bit de permiso en el registro UIE correspondiente. Todos los flags de estado del USB se suman para generar el flag de interrupción USBIF para el túnel de interrupción del micro.

Una vez que el SIE activa un bit de interrupción, se tiene que borrar por software escribiendo un '0'. Los flags se pueden activar por software para ayudar en la búsqueda de errores del firmware.

REGISTRO 17-7: UIR: REGISTRO DE ESTADO DE LAS INTERRUPCIONES DEL USB

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
-------	-------	-------	-------	-------	-------	-------	-------

		SOFIF	STALLIF	IDLEIF ⁽¹⁾	TRNIF ⁽²⁾	ACTVIF ⁽³⁾	UERRIF ⁽⁴⁾	URSTIF
	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
BIT 6		SOFIF:			Bit de interrupción del símbolo START-OF-FRAME 1= START-OF-FRAME recibido por el SIE 0= ningún START-OF-FRAME recibido por el SIE			
BIT 5		STALLIF:			Bit de interrupción del protocolo de STALL 1= protocolo de STALL enviado por el SIE 0= protocolo de intercambio STALL no se ha enviado			
BIT 4		IDLEIF:			Bit de interrupción detector de reposo ⁽¹⁾ 1= Reposo detectado (estado de reposo de 3ms o más) 0= ninguna condición de reposo detectada			
BIT 3		TRNIF:			Bit de interrupción de transacción completa ⁽²⁾ 1= transacción pendiente completa; leer el registro USTAT para información del Endpoint 0= transacción pendiente no completada o no hay			
BIT 2		ACTVIF:			Bit de interrupción de detección de la actividad del bus ⁽³⁾ 1= actividad detectada en las líneas D+/D- 0= ninguna actividad detectada en las líneas D+/D-			
BIT 1		UERRIF:			Bit de interrupción de la condición de error del USB ⁽⁴⁾ 1= ha ocurrido una condición de error desenmascarada 0= no ha ocurrido ninguna condición de error.			
BIT 0		URSTIF:			Bit de interrupción de reset del USB 1= Ha ocurrido un reset en el USB, 00h se carga en el registro UADDR 0= no ha ocurrido ningún reset del USB			

Nota 1: Una vez que se detecte un estado de reposo, el usuario puede colocar al módulo USB en este modo.

2: Borrar este bit hará avanzar la USTAT FIFO (válido solamente con los símbolos IN, OUT y SETUP).

3: Este bit es desenmascarado al detectar un acontecimiento de interrupción UIDLE.

4: Solamente las condiciones de error permitidas a través del registro UEIE activarán este bit. Este bit es bit de estado y no lo puede modificar el usuario.

17.5.1.1 Bit de interrupción de detección de la actividad del bus (ACTVIF)

El bit ACTVIF no se puede ser borrar inmediatamente después de despertar al módulo USB del modo reposo o cuando se suspende. Se necesita un retraso para sincronizar el estado interno del hardware antes de que el bit ACTVIF pueda borrarse por firmware. Borrar el bit ACTVIF antes de sincronizar el hardware puede que no cause efecto. Además, si el módulo USB utiliza una fuente de reloj de 96MHz PLL, después se borra el bit SUSPND, el módulo del USB puede ser operacional inmediatamente mientras que espera los 96MHz PLL.

EJEMPLO 17-1: BORRADO DEL BIT ACTVIF (UIR<2>)

Ensamblador:

BCF UCON, SUSPND

LOOP:

BTFSS	UIR, ACTVIF
BRA	DONE
BCF	UIR, ACTVIF
BRA	LOOP

DONE:

C:

```
UCONbits.SUSPND=0;
while (UIRbits.ACTVIF) {UIRbits.ACTVIF=0;}
```

17.5.2 REGISTRO DE PERMISO DE LAS INTERRUPCIONES (UIE)

El registro de permiso de la interrupción del USB contiene los bits de permiso del estado de las fuentes de interrupción USB. No fijar ninguno de estos bits permitirá la interrupción elegida en el registro UIR.

Los valores en este registro afectan solamente la propagación de una condición de interrupción a la lógica de interrupción del microcontrolador. Los flags todavía están activados por su condición de interrupción, permite que sean interrogados y se mantienen sin realmente la interrupción.

REGISTRO 17-8: UIE: REGISTRO DE LA INTERRUPCIÓN DEL USB

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
-	SOFIE	STALLIE	IDLEIE	TRNIE	ACTVIE	UERRIE	URSTIE
U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
BIT 6	SOFIE:						
		Bit de permiso de la interrupción del START-OF-FRAME					
		1 = interrupción permitida					
		0 = interrupción inhabilitada					
BIT 5	STALLIE:						
		Bit de permiso de la interrupción del protocolo STALL					
		1 = interrupción permitida					
		0 = interrupción inhabilitada					
BIT 4	IDLEIE:						
		Bit de permiso de la interrupción de reposo					
		1 = interrupción permitida					
		0 = interrupción inhabilitada					
BIT 3	TRNIE:						
		Bit de permiso de la interrupción de transacción completa					
		1 = interrupción permitida					
		0 = interrupción inhabilitada					
BIT 2	ACTVIE:						
		Bit de permiso de la interrupción de detección de la actividad del bus					
		1 = interrupción permitida					
		0 = interrupción inhabilitada					
BIT 1	UERRIE:						
		Bit de permiso de la interrupción de error del USB					
		1 = interrupción permitida					
		0 = interrupción inhabilitada					
BIT 0	URSTIE:						
		Bit de permiso de la interrupción del reset del USB					
		1 = interrupción permitida					
		0 = interrupción inhabilitada					

17.5.3 REGISTRO DE ESTADO DE LAS INTERRUPCIONES POR ERROR DEL USB (UEIR)

El registro de estado de las interrupciones de error del USB contiene los flags para cada fuente de error del periférico USB. Cada una de estas fuentes se controla con el bit de permiso de interrupción correspondiente del registro UEIE. Todas los flags de error de USB se suman para generar el flag de interrupción de error del USB (UERRIF) en el nivel superior de la lógica de interrupción.

Cada bit de error se activa cuando se detecta la condición de interrupción. Así, la interrupción normalmente no corresponde con el final de un símbolo que se acaba de procesar.

Una vez que un bit de interrupción haya activado el SIE, tiene que borrarse por software escribiendo un '0'.

REGISTRO 17-9: UEIR: REGISTRO DE ESTADO DE LAS INTERRUPCIONES DE ERROR DEL USB

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
BTSEF	-	-	BTOEF	DFN8EF	CRC16EF	CRC5EF	PIDEF
R/C-0	U-0	U-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0
BIT 7	BTSEF:			Flag de error del bit mercancía			
				1= Error detectado			
				0= No se ha detectado error			
BIT 4	BTOEF:			Flag de error de espera en el procesamiento del bus			
				1= error detectado (han llegado más de 16bits de reposo antes de un EOP)			
				0= error no detectado			
BIT 3	DFN8EF:			Flag de error del tamaño de los datos			
				1= error detectado (no era un número entero de bytes)			
				0= error no detectado			
BIT 2	CRC16EF:			Flag de fallo CRC16			
				1= error detectado			
				0= error no detectado			
BIT 1	CRC5EF:			Flag de error del anfitrión CRC5			
				1= error detectado (paquete simbólico rechazado)			
				0= error no detectado			
BIT 0	PIDEF:			Flag de prueba de fallo de PID			
				1= error detectado			
				0= error no detectado			

17.5.4 REGISTRO DE PERMISO DE LAS INTERRUPCIONES DE ERROR (UEIE)

El registro de permiso de las interrupciones de error (UEIE) contiene los bits de activación para cada fuente de interrupción de error del USB. Activando cualquiera de estos bits se activa la fuente de la interrupción respectiva.

Como el registro UIE, los bits activos sólo afectan la propagación de la condición de la interrupción. Los flags se activan cuando se cumplen sus condiciones.

REGISTRO 17-10: UEIE: REGISTRO DE PERMISO DE LA INTERRUPCIÓN DE ERROR DEL USB

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
BTSEE	-	-	BTOEE	DFN8EE	CRC16EE	CRC5EE	PIDEE
R/C-0	U-0	U-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0
BIT 7	BTSEF:		Bit de permiso de interrupción del error del bit mercancía 1= Interrupción permitida 0= Interrupción no permitida				
BIT 4	BTOEF:		Bit de permiso de interrupción del error de espera en el procesamiento del bus 1= Interrupción permitida 0= Interrupción no permitida				
BIT 3	DFN8EF:		Bit de permiso de interrupción del error del tamaño de los datos 1= Interrupción permitida 0= Interrupción no permitida				
BIT 2	CRC16EF:		Bit de permiso de interrupción del fallo CRC16 1= Interrupción permitida 0= Interrupción no permitida				
BIT 1	CRC5EF:		Bit de permiso de interrupción del error del anfitrión CRC5 1= Interrupción permitida 0= Interrupción no permitida				
BIT 0	PIDEF:		Bit de permiso de interrupción del prueba de fallo de PID 1= Interrupción permitida 0= Interrupción no permitida				

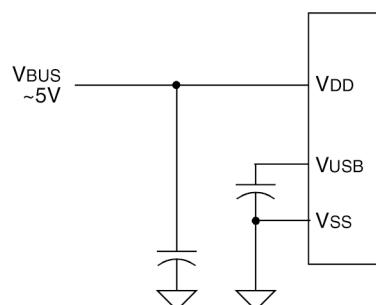
17.6 MODOS DE ENERGÍA DEL USB

Las aplicaciones USB tendrán diferentes requisitos y configuración de energía. Los casos más comunes son los presentados aquí.

17.6.1 SÓLO ENERGÍA EN EL BUS

En modo de sólo energía en el bus. Es el método más simple de energía para el dispositivo.

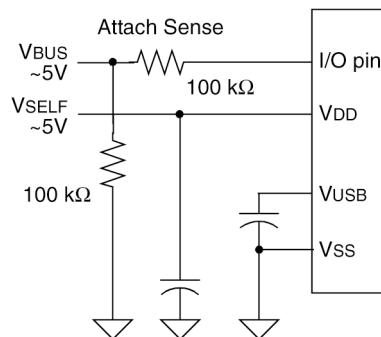
FIGURA 17-10: SÓLO ENERGÍA EN EL BUS



17.6.2 SÓLO SELF-POWER

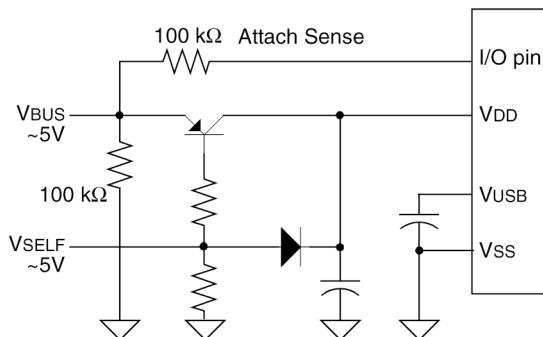
En modo sólo SELF-POWER, el uso del USB proporciona su propia energía, con la energía muy pequeña

cedida por el USB. Observar que indica cuando el USB ha estado conectado.

FIGURA 17-11: SÓLO SELF-POWER

17.6.3 ENERGÍA DUAL CON DOMINANCIA SELF-POWER

Algunas aplicaciones necesitan una opción con dos energías. La aplicación utiliza la fuente de energía interna como primaria, pero cambia a la energía del USB cuando no se dispone de una fuente lineal.

FIGURA 17-12: ENERGÍA DUAL CON DOMINANCIA SELF-POWER

Nota: Los usuarios deben tener presente los límites de energía del USB. Según la especificación del USB 2.0, no puede exceder 100mA en un dispositivo de baja potencia ó 500mA en uno de alta.

17.7 STREAMING PARALLEL PORT (SPP)

El puerto paralelo (SPP) es una ruta alternativa de los datos además de la RAM del USB. Usando el SPP, un Endpoint se puede configurar para enviar o para recibir datos directamente del hardware externo.

Este método presenta posibilidades de diseño donde el microcontrolador actúa como encargado de los datos, permitiendo al SPP pasar bloques grandes de datos sin que el micro regule lo que procesa realmente. Un ejemplo de aplicación puede incluir un sistema de adquisición de datos, donde los datos fluyen de una FIFO externa a través del USB al ordenador. En este caso, el control del Endpoint lo realiza el microcontrolador y los movimientos de datos en bruto se procesan externamente.

El SPP se permite como puerto de un Endpoint del USB a través del BD asociado al Endpoint. El Endpoint tiene que activarse de la siguiente manera:

1. Activar BDnADRL:BDnADRH direccionado a FFFFh.
2. Activar KEN (BDnSTAT<5>) para que el SIE controle el Buffer.
3. Activar INCDIS (BDnSTAT<4>) para inhabilitar el incremento de dirección automática.

Nota 1: Si un Endpoint se configura para utilizar el SPP, el módulo SPP debe configurarse para utilizar el módulo USB. Si no, puede ocurrir una operación inesperada.

2: Además, si un Endpoint se configura para utilizar el SPP, el tipo de transferencia de datos de ese Endpoint debe ser síncrona.

17.8 OSCILADOR

El módulo USB necesita una señal específica de reloj. En operaciones full-speed, el reloj tiene que ser de 48MHz. El microcontrolador y los periféricos no tienen porque tener la misma frecuencia de reloj o la misma fuente.

TABLA 17-6: REGISTROS ASOCIADOS A LAS OPERACIONES DEL MÓDULO USB⁽¹⁾

Nombre	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTCON	GIE/GIEH	PEIE/GIEL	TMROIE	INTOIE	RBIE	TMROIF	INT0IF	RBIF
IPR2	OSCFIP	CMIP	USBIP	EEIP	BCLIP	HLVDIP	TMR3IP	CCP2IP
PIR2	OSCFIF	CMIF	USBIF	EEIF	BCLIF	HLVDIF	TMR3IF	CCP2IF
PIE2	OSCFIE	CMIE	USBIE	EEIE	BCLIE	HLVDIE	TMR3IE	CCP2IE
UCON	—	PPBRST	SE0	PKTDIS	USBEN	RESUME	SUSPND	—
UCFG	UTEYE	UOEMON	—	UPUEN	UTRDIS	FSEN	PPB1	PPB0
USTAT	—	ENDP3	ENDP2	ENDP1	ENDP0	DIR	PPBI	—
UADDR	—	ADDR6	ADDR5	ADDR4	ADDR3	ADDR2	ADDR1	ADDR0
UFRML	FRM7	FRM6	FRM5	FRM4	FRM3	FRM2	FRM1	FRM0
UFRMH	—	—	—	—	—	FRM10	FRM9	FRM8
UIR	—	SOFIF	STALLIF	IDLEIF	TRNIF	ACTVIF	UERRIF	URSTIF
UIE	—	SOFIE	STALLIE	IDLEIE	TRNIE	ACTVIE	UERRIE	URSTIE
UEIR	BTSEF	—	—	BTOEF	DFN8EF	CRC16EF	CRC5EF	PIDEF
UEIE	BTSEE	—	—	BTOEE	DFN8EE	CRC16EE	CRC5EE	PIDEE
UEP0	—	—	—	EPHSHK	EPCONDIS	EPOUTEN	EPINEN	EPSTALL
UEP1	—	—	—	EPHSHK	EPCONDIS	EPOUTEN	EPINEN	EPSTALL
UEP2	—	—	—	EPHSHK	EPCONDIS	EPOUTEN	EPINEN	EPSTALL
UEP3	—	—	—	EPHSHK	EPCONDIS	EPOUTEN	EPINEN	EPSTALL
UEP4	—	—	—	EPHSHK	EPCONDIS	EPOUTEN	EPINEN	EPSTALL
UEP5	—	—	—	EPHSHK	EPCONDIS	EPOUTEN	EPINEN	EPSTALL
UEP6	—	—	—	EPHSHK	EPCONDIS	EPOUTEN	EPINEN	EPSTALL
UEP7	—	—	—	EPHSHK	EPCONDIS	EPOUTEN	EPINEN	EPSTALL
UEP8	—	—	—	EPHSHK	EPCONDIS	EPOUTEN	EPINEN	EPSTALL
UEP9	—	—	—	EPHSHK	EPCONDIS	EPOUTEN	EPINEN	EPSTALL
UEP10	—	—	—	EPHSHK	EPCONDIS	EPOUTEN	EPINEN	EPSTALL
UEP11	—	—	—	EPHSHK	EPCONDIS	EPOUTEN	EPINEN	EPSTALL
UEP12	—	—	—	EPHSHK	EPCONDIS	EPOUTEN	EPINEN	EPSTALL
UEP13	—	—	—	EPHSHK	EPCONDIS	EPOUTEN	EPINEN	EPSTALL
UEP14	—	—	—	EPHSHK	EPCONDIS	EPOUTEN	EPINEN	EPSTALL
UEP15	—	—	—	EPHSHK	EPCONDIS	EPOUTEN	EPINEN	EPSTALL

Leyenda: - = las localizaciones no están implementadas. Las celdas sombreadas no se utilizan con el módulo USB.

Nota 1: Esta tabla incluye solamente las localizaciones SFRs mapeadas en el banco 15 de la memoria de datos. Los registros de los BD, que están mapeados en el banco 4 y no son SFRs verdaderos, están en la tabla 17-5.

17.9 DESCRIPCIÓN DEL USB

Esta sección presenta algunos conceptos básicos del USB e información útil necesaria para diseñar un dispositivo USB. Así, se anima al lector que refiera a las especificaciones del USB para más información (www.usb.org). Si estás muy al corriente de los detalles de USB, entonces ésta sección sirve como recuerdo básico, de alto nivel del USB.

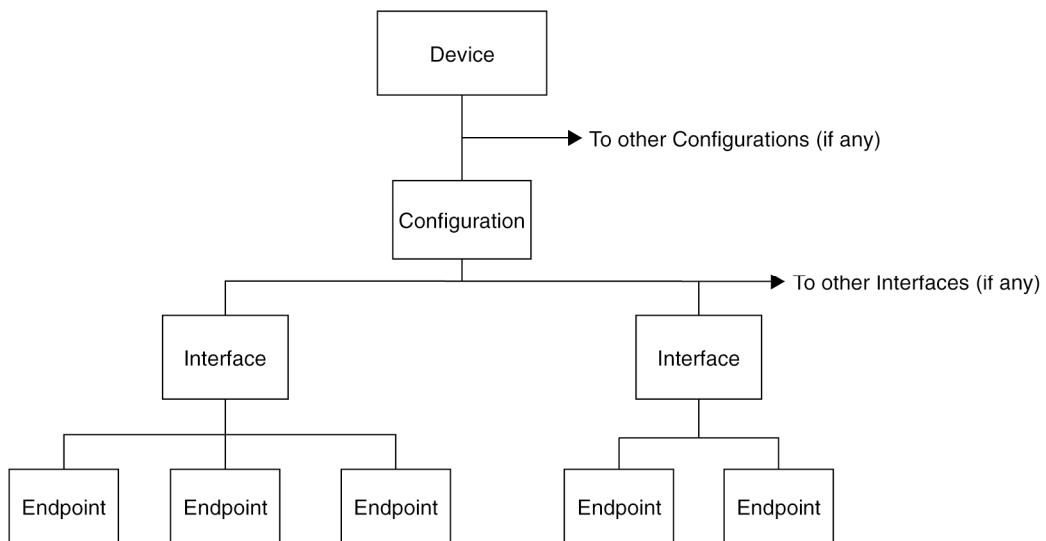
17.9.1 ESQUEMA DE CAPAS

La funcionalidad del dispositivo del USB se estructura en un esquema de capas. Cada nivel se asocia a un nivel funcional dentro del dispositivo. La capa más alta, con excepción del dispositivo, es la de configuración. Un dispositivo puede tener configuraciones múltiples. Por ejemplo, un dispositivo particular puede tener requisitos de energía múltiples basados en Self-Power o modos de energía sólo del bus.

Para cada configuración, puede haber múltiples interfaces. Cada interfaz podía apoyar un modo particular de esa configuración.

Debajo del interfaz están los Endpoints. Los datos se mueven directamente a este nivel. Puede haber 16 Endpoints bidireccionales. El Endpoint 0 es siempre el Endpoint de control por defecto; cuando el dispositivo está en el bus, el Endpoint 0 debe estar disponible para configurarlo.

FIGURA 17-13: CAPAS DEL USB



17.9.2 FRAMES

La información que se comunica en el bus se agrupa en ranuras de 1ms, designadas frames. Cada frame puede contener muchas transacciones a varios dispositivos y Endpoints.

17.9.3 TRANSFERENCIAS

Hay cuatro tipos de transferencias definidas en las especificaciones del USB.

- **Síncrona:** Este tipo proporciona un método de transferencia para cantidades de datos grandes (hasta 923 bytes) con la entrega puntual asegurada; sin embargo, la integridad de los datos no se asegura. Es bueno en aplicaciones donde los datos son pequeños y la pérdida no es crítica, por ejemplo audio.
- **Bulk:** Este método de transferencia permite grande cantidades de datos que se transferirán asegurando la integridad de los datos; sin embargo, la puntualidad de la entrega no se asegura.
- **Interrupción:** Este tipo de transferencia prevé la entrega puntual asegurada para bloques pequeños de datos, se asegura la integridad de los datos.
- **Control:** Este tipo prevé la disposición de dispositivo control.

Mientras que los dispositivos “full-speed” soportan todos los tipos de transferencia, los dispositivos “de poca velocidad” se limitan a las transferencias interrupción y control.

17.9.4 ENERGÍA

La energía se puede obtener del USB. Las especificaciones del USB definen los requisitos de energía del bus. Los dispositivos pueden ser self-powered o alimentados por bus. Los dispositivos Self-powered consiguen energía de una fuente externa, mientras que los dispositivos alimentados por bus utilizan la energía del bus.

La especificación del USB limita la energía tomada del bus. Cada dispositivo se asegura 90mA y 5V (una unidad de carga). La energía adicional puede solicitarse, hasta un máximo de 500mA. Observar que ceder más energía que la de una unidad de carga es una petición y el anfitrión o el hub no tiene que proporcionarla obligatoriamente. Así, un dispositivo capaz de consumir más de una unidad de carga debe ser capaz de mantener una configuración de baja potencia de una unidad carga o menos, en caso de necesidad.

La especificación del USB también define un modo de reposo. En esta situación, la corriente se debe limitar a $500\mu A$, sobre 1 segundo. Un dispositivo se debe incorporar al estado de reposo después de 3ms de inactividad (es decir, ningún símbolo SOF en 3ms). El entrar el dispositivo en el modo reposo debe caer la consumición actual en los 10ms siguientes. Asimismo, al señalar un reinicio, el dispositivo debe señalarlo en los 10ms cediendo corriente.

17.9.5 ENUMERACIÓN

Cuando el dispositivo se une al bus, el anfitrión crea un proceso de enumeración en un intento por identificar el dispositivo. Esencialmente, el anfitrión interroga al dispositivo, recopila información tal como el consumo de energía, tarifas y tamaños de

los datos, protocolo y otra información descriptiva, que contienen los descriptores. El proceso de enumeración sería:

1. Reset del USB: Resetear el dispositivo. Así, el dispositivo no se configura y no tiene una dirección (dirección 0).
2. Conseguir el descriptor del dispositivo: El anfitrión solicita una pequeña parte del descriptor del dispositivo.
3. Reset del USB: Resetear el dispositivo otra vez.
4. Fijar la dirección: El anfitrión asigna una dirección al dispositivo.
5. Conseguir el descriptor del dispositivo: El anfitrión recupera el descriptor del dispositivo, recolectando la información, por ejemplo fabricante, tipo de dispositivo, control máximo del tamaño de los paquetes.
6. Conseguir los descriptores de configuración.
7. Conseguir cualquier otro descriptor.
8. Fijar una configuración.

El proceso exacto de enumeración depende del anfitrión.

17.9.6 DESCRIPTORES

Hay ocho estándares del descriptor de cuál cinco son los más importantes para este dispositivo.

17.9.6.1 Descriptor del dispositivo

El descriptor del dispositivo proporciona la información de carácter general, por ejemplo fabricante, número del producto, número de serie, la clase del dispositivo y el número de configuraciones. Hay solamente un descriptor del dispositivo.

17.9.6.2 Descriptor de la configuración

El descriptor de la configuración proporciona la información de los requisitos de energía del dispositivo y cuántos interfaces diferentes soporta cuando está en esta configuración. Puede haber más de una configuración del dispositivo (es decir, configuraciones de baja potencia y de alta potencia).

17.9.6.3 Descriptor de la interfaz

El descriptor de la interfaz detalla el número de Endpoints utilizados en esta interfaz, así como la clase de la interfaz. Puede haber más de una interfaz en la configuración.

17.9.6.4 Descriptor del Endpoint

El descriptor del Endpoint identifica el tipo de transferencia y dirección, también algunos otros específicos del Endpoint. Puede haber muchos Endpoints en un dispositivo y los Endpoints pueden compartirse en diversas configuraciones.

17.9.6.5 Descriptor de la secuencia

Muchos de los descriptores anteriores se refieren a uno o más descriptores de la secuencia. Los descriptores de la secuencia proporcionan información legible por un humano sobre la capa que pueden describir. Estas secuencias muestran en el anfitrión una ayuda para identificar el dispositivo. Los descriptores de la secuencia son generalmente opcionales para ahorrar memoria y se codifican en formato unicode.

17.9.7 VELOCIDAD DEL BUS

Cada dispositivo USB debe indicar su presencia del bus y su velocidad al anfitrión. Esto se logra con una resistencia de $1,5\text{k}\Omega$ que se conecta al bus en el momento del acoplamiento.

Dependiendo de la velocidad del dispositivo, la resistencia une la línea D+ o D- a 3,3V. En dispositivos de poca velocidad, la resistencia se conecta a la línea D-. En los dispositivos full-speed, la resistencia se conecta a la línea D+.

17.9.8 ESPECIFICACIONES DE CLASE Y DRIVERS

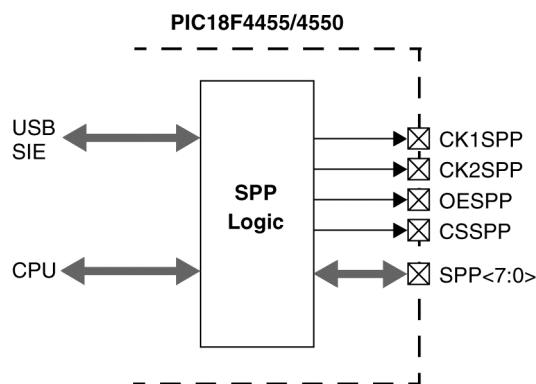
Las especificaciones del USB incluyen las especificaciones de clase que los vendedores del sistema operativo apoyan opcionalmente. Los ejemplos de clases incluyen el audio, memoria de masa, comunicaciones e interfaz humano (HID). En la mayoría de los casos, se requiere un driver en el lado del anfitrión para comunicarse con el dispositivo USB. En aplicaciones de encargo, se puede necesitar un driver convertido. Afortunadamente, los drivers están disponibles para la mayoría de los sistemas anfitriones comunes para las clases más comunes de los dispositivos. Así, estos drivers se pueden reutilizar.

18.0 PUERTO PARALELO STREAMING (SPP)

Nota: El puerto paralelo streaming está solamente disponible en los dispositivos de 40/44-pines.

Los dispositivos USB PIC18F4455/4550 proporcionan un Puerto Paralelo Streaming como interfaz de alta velocidad para mover datos a y desde un sistema externo. Este puerto paralelo funciona como puerto principal, completo con un chip select y salidas de reloj para controlar el movimiento de los datos a los dispositivos auxiliares. Los datos se pueden dirigir directamente al SIE del USB o al núcleo del microprocesador.

FIGURA 18-1: TRAYECTORIA DE DATOS DEL SPP



Además, el SPP pueden proporcionar tiempo multiplexado tratando la información junto con los datos usando segunda salida estroboscopia. Así, el número del Endpoint del USB se puede escribir conjuntamente con los datos en ese Endpoint.

18.1 CONFIGURACIÓN DEL SPP

La operación del SPP se controla con dos registros: SPPCON y SPPCFG. El registro SPPCON dirige la operación total del puerto paralelo y determina si funciona bajo el control del USB o del microcontrolador. El registro SPPCFG controla la configuración del tiempo y los pines de salida.

18.1.1 PERMITIR EL SPP

Para permitir el SPP, activar el bit SPEN (SPPCON<0>). Además, los bits del TRIS del SPP correspondientes los pines se deben configurar correctamente. El mínimo:

- Bits TRISD<7:0> se deben activar (= 1)
- Bits TRISE<2:1> se deben borrar (= 0)

Si se utiliza CK1SPP:

- El bit TRISE<0> se debe borrar (= 0)

Si se utiliza CSPP:

- El bit TRISB<4> se debe borrar (= 0)

REGISTRO 18-1: SPPCON: REGISTRO DE CONTROL DEL SPP

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
-	-	-	-	-	-	SPPOWN	SPPEN
U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
BIT2	EPOUTEN:	Bit de permiso de la salida del Endpoint 1= la salida del Endpoint n permitida					
BIT1	SPPOWN:	Bit de propiedad del SPP 1= El periférico USB controla el SPP 0= El microcontrolador controla directamente el SPP					
BIT0	SPPEN:	Bit de permiso del SPP 1 = se permite 0 = se desactiva					

REGISTRO 18-2: SPPCFG: REGISTRO DE LA CONFIGURACIÓN DE EL SPP

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
CLKCFG1	CLKCFG0	CSEN	CLK1EN	WS3	WS2	WS1	WS0
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
BIT 7-6	CLKCFG1:CLKCFG0:	Bits de configuración del reloj del SPP 1x= CLK1 cambia a lectura o escritura una dirección impar de Endpoint; CLK2 cambia a lectura o escritura una dirección par de Endpoint. 01= CLK1 cambia a escritura; CLK2 cambia a lectura 00= CLK1 cambia sólo en la dirección del Endpoint de escritura; CLK2 cambia en la lectura o escritura de datos					
BIT 5	CSEN:	Bit de permiso del pin selector del chip del SPP 1 = RB4 lo controla el módulo SPP y funciona como salida CS del SPP 0 = RB4 funciona como un puerto digital de E/S					
BIT 4	CLK1EN:	Bit de permiso del pin del SPP CLK1 1 = RE0 lo controla el módulo SPP y funciona como la salida del SPP CLK1 0 = RE0 funciona como un puerto digital de E/S					
BIT 3-0	WS3:WS0:	Bits de los estados espera del SPP 1111 = 30 estados de espera adicionales 1110 = 28 estados de espera adicionales 0001 = 2 estados de espera adicionales 0000 = 0 estados de espera adicionales					

18.1.2 REGISTRO DE DATOS

El SPP tiene cuatro salidas del control:

- Dos salidas de reloj separadas (CK1SPP y CK2SPP)
- Salida permitida (OESPP)
- Chip select (CSSPP)

Después, permiten diversas configuraciones para controlar el flujo de datos a los dispositivos auxiliares. Cuando se utilizan todas las salidas de control, las tres opciones principales son:

- CLK1 controla la información del Endpoint diseccionado cuando CLK2 controla los datos.
- CLK1 controla las operaciones de escritura mientras que CLK2 controla las de lectura.
- CLK1 controla los datos en las direcciones impares mientras que CLK2 controla las de los pares.

Las opciones de control adicionales derivan inhabilitando las salidas CK1SPP y CSSPP. Se permiten o inhabilitan con los bits CLK1EN y CSEN, respectivamente.

18.1.3 ESTADOS DE ESPERA

El SPP se diseña con la capacidad de agregar estados de espera para las operaciones de lectura y de escritura. Esto permite el acceso paralelo a los dispositivos que requieren más tiempo para acceder.

Registrar el estado de espera se basa en el reloj de la fuente de datos. Si el SPP se configura para funcionar como Endpoint del USB, entonces los estados de espera se basan en el reloj del USB. Asimismo, si el SPP se configura para funcionar desde el microcontrolador, entonces los estados de espera se basa en el ratio de instrucción ($F_{osc}/4$).

Los bits WS3:WS0 activaron los estados de espera utilizados por el SPP, con una gama de ningunos estado de espera a 30 estados de espera, en múltiplos de dos. Los estados de espera se agregan simétricamente en todas las transacciones, con una mitad añadida a cada uno de los dos ciclos de reloj requeridos para la transacción.

18.1.4 SPP PULL-UPS

Las líneas de datos del SPP ($SPP<7:0>$) se equipan con pull-ups internas para aplicaciones que puedan salir del puerto en condiciones e alta impedancia. Los pull-ups se permiten con el bit de control, RDPU (PORTE $<7>$).

FIGURA 18-2: SINCRONIZACIÓN PARA EL MICROCONTROLADOR ESCRIBE DIRECCIÓN, ESCRIBE DATOS Y LEE DATOS (NINGÚN ESTADO DE ESPERA)

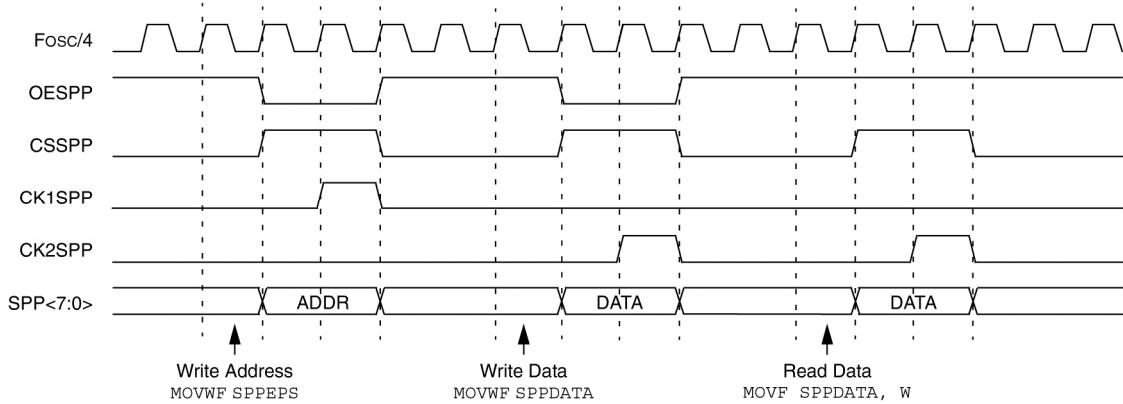


FIGURA 18-3: SINCRONIZACIÓN PARA EL USB ESCRIBE DIRECCIÓN Y DATOS (4 ESTADOS DE ESPERA)

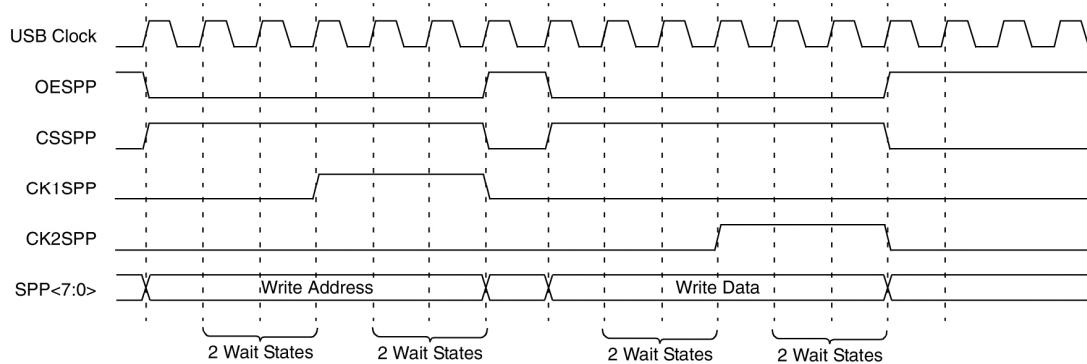
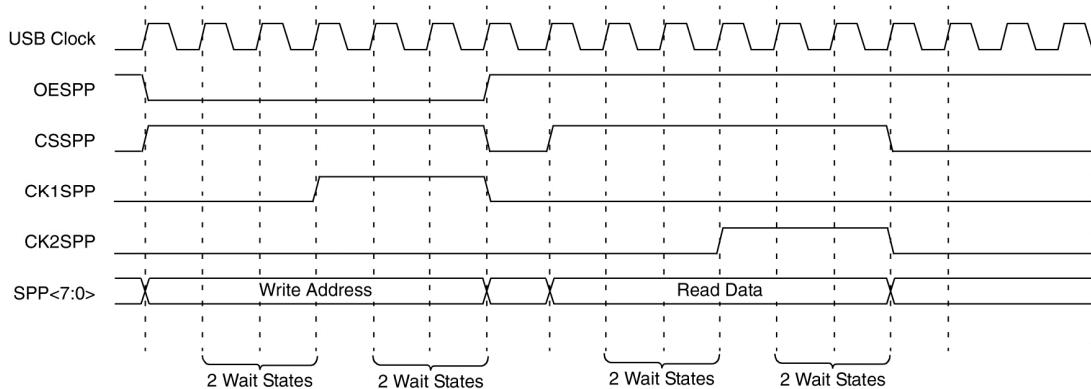


FIGURA 18-4: SINCRONIZACIÓN PARA EL USB ESCRIBE DIRECCIÓN Y LEE DATOS (4 ESTADOS DE ESPERA)



18.2 SISTEMA PARA EL CONTROL DEL USB

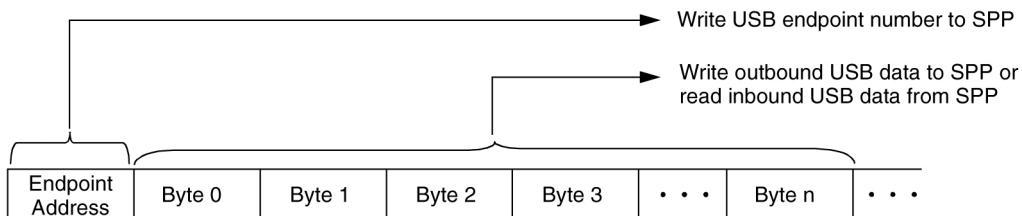
Cuando el SPP se configura para operar como USB, los datos se pueden controlar directamente a y desde el periférico USB sin la intervención del microcontrolador; así, no requiere tiempo de procesado. Los datos se manejan a o hacia fuera del SPP con la información del Endpoint (dirección) primero, seguido por uno o

más bytes de datos. Esto es ideal en aplicaciones que requieren transmisión síncrona, de gran capacidad. Los siguientes pasos se requieren para configurar el SPP para controlr el USB:

1. Configurar el SPP según lo deseado, incluyendo estados de espera y relojes.
2. Activar el bit SPPOWN para la propiedad del USB.
3. Activar el descriptor del buffer que comienza la dirección (BDnADRL:BDnADRH) a FFFFh.
4. Activar el bit KEN (BDnSTAT<5>) pero el buffer descriptor lo se guarda indefinidamente el SIE.
5. Activar el bit INCDIS (BDnSTAT<4>) para inhabilitar el incremento automático de la dirección del buffer.
6. Activar el bit SPPEPEN para permitir el módulo.

Nota: Si un Endpoint del USB se configura para utilizar el SPP, el tipo de la transferencia de datos de ese Endpoint debe ser síncrona.

FIGURA 18-5: TRANSFERENCIA DE DATOS ENTRE USB SIE Y SPP



18.3 SISTEMA PARA EL CONTROL DEL MICROCONTROLADOR

El SPP puede actuar como puerto paralelo del microcontrolador. En este modo, el registro SPPEPS proporciona el estado y la dirección de control de escritura. Los datos se escriben y se leen en el registro SPPDATA. Cuando el SPP lo gobierna el microcontrolador, el reloj del SPP lo controla el reloj de instrucción ($F_{osc}/4$). Los pasos siguientes se requieren para configurar el SPP para operación del microcontrolador:

1. Configurar el SPP según lo deseado, incluyendo estados de espera y relojes.
2. Borrar el bit SPPOWN.
3. Activar SPPEPEN para permitir el módulo.

18.3.1 INTERRUPCIONES SPP

Cuando lo controla el núcleo del microcontrolador, el control puede generar una interrupción para notificar la aplicación cuando termina cada operación de lectura y escritura. El flag de interrupción es SPPIF (PIR1<7>) y se permite con el bit SPPIE (PIE1<7>). Como el resto de prioridades de interrupciones del microcontrolador, se puede fijar a un nivel bajo o prioritario. Esto se hace con el bit SPPIP (IPR1<7>).

18.3.2 ESCRIBIR EL SPP

Una vez que esté configurado, la escritura del SPP se realiza escribiendo los registros SPPEPS y SPPDATA. Si el SPP se configura para registrar la dirección de los Endpoint OUT con los datos, escribiendo el registro SPPEPS inicia el ciclo de escritura en la dirección. Si no, la escritura comienza escribiendo los datos al registro SPPDATA. El bit SPPBUSY indica el estado de la dirección y los datos de los ciclos de escritura.

Lo que sigue es un ejemplo de secuencia de escritura:

1. Escribir la dirección de 4 bits al registro SPPEPS. El SPP comienza automáticamente escribiendo la dirección. Si la escritura de la dirección no se utiliza, vamos al paso 3.
2. Supervisar el bit SPPBUSY para determinar cuando se ha enviado la dirección. La duración depende en los estados de espera.
3. Escribir los datos al registro SPPDATA. El SPP comienza automáticamente a escribir los datos.
4. Supervisar el bit SPPBUSY para determinar cuando se han enviado los datos. La duración depende de los estados de espera.
5. Ir de nuevo a los pasos 1 ó 3 para escribir una nueva dirección o datos.

Nota: Se debe comprobar el bit SPPBUSY para asegurarse de que las sucesivas escrituras a los registros SPPEPS o SPPDATA no invadan el tiempo de espera debido al ajuste del estado de espera.

18.3.3 LECTURA DEL SPP

La lectura del SPP implica leer el registro SPPDATA. Leyendo el registro por primera vez inicia la operación de lectura. Cuando la lectura ha finalizado, se indica con el bit SPPBUSY, el SPPDATA se habrá cargado con los datos actuales.

Ejemplo de secuencia de lectura:

1. Escribir la dirección de 4bits al registro SPPEPS. El SPP comienza automáticamente escribiendo la dirección. Si la dirección escrita no se usa saltemos al paso 3.
2. Supervisar el bit SPPBUSY para determinar cuando se ha enviado la dirección. La duración depende de los estados de espera.
3. Leer los datos del registro SPPDATA; se devuelven los datos de la operación de lectura anterior. El SPP comienza automáticamente la lectura para el próximo ciclo de lectura.
4. Supervisar el bit SPPBUSY para determinar cuando se han leído los datos. La duración depende de los estados de espera.
5. Ir al paso 3 para leer el byte actual del SPP y empezar el siguiente ciclo de lectura.

REGISTRO 18-3: SPPEPS: DIRECCIÓN DEL ENDPOINT DEL SPP Y REGISTRO DE ESTADO

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
RDSPP	WRSP	-	SPPBUSY	ADDR3	ADDR2	ADDR1	ADDR0
R-0	R-0	U-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0
BIT 7	RDSPP:		Bit del estado de lectura del SPP (válido cuando SPPCON<SPPOWN>=1, USB)				

1 = la transacción anterior del SPP era de lectura

BIT 6	WRSPP:	0 = la transacción anterior del SPP no era de lectura Bit de estado de escritura del SPP (válido cuando SPPCON<SPPOWN>=1, USB) 1 = la transacción anterior del SPP era de escritura
BIT 4	SPPBUSY:	0 = la transacción anterior del SPP no era de escritura Bit de ocupación del bus del SPP 1 = El SPP está ocupado
		0 = El SPP está listo para aceptar otras peticiones de lectura y escritura
BIT 3-0	ADDR3:ADDR0:	Bits de dirección del Endpoint del SPP 1111 = Endpoint 15 ... 0001 0000 = Endpoint 0

TABLA 18-1: REGISTROS ASOCIADOS AL PUERTO PARALELO STREAMING

Nombre	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SPPCON ⁽³⁾	—	—	—	—	—	—	SPPOWN	SPPEN
SPPCFG ⁽³⁾	CLKCFG1	CLKCFG0	CSEN	CLK1EN	WS3	WS2	WS1	WS0
SPPEPS ⁽³⁾	RDSPP	WRSPP	—	SPPBUSY	ADDR3	ADDR2	ADDR1	ADDR0
SPPDATA ⁽³⁾	DATA7	DATA6	DATA5	DATA4	DATA3	DATA2	DATA1	DATA0
PIR1	SPPIF ⁽³⁾	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF
PIE1	SPPIE ⁽³⁾	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE
IPR1	SPPIP ⁽³⁾	ADIP	RCIP	TXIP	SSPIP	CCP1IP	TMR2IP	TMR1IP
PORTE	RDPUI ⁽³⁾	—	—	—	RE3 ^(1,2)	RE2 ⁽³⁾	RE1 ⁽³⁾	RE0 ⁽³⁾

Leyenda: --no implementado, se leen ‘0’. Las casillas sombreadas no se utilizan con el SPP.

Nota 1: Implementado cuando el Master Clean está desactivado (configuración del bit MCLRE=0).

2: RE3 es el único bit de PORTE implementado en dispositivos de 28 pines y de 40/44 pines. El resto de pines se implementan si existen el resto del PORTE (es decir, en los dispositivos de 40/44 pines).

3: Estos registros y/o bits no están implementados en los dispositivos de 28 pines.

21.0 MÓDULO CONVERTIDOR DE 10-BIT ANALÓGICO A DIGITAL (A/D)

El módulo conversor de analógico a digital (A/D) tiene 10 entradas con los dispositivos de 28 pines y 13 en los de 40/44 pines. Este módulo permite la conversión de una señal de entrada analógica a un número digital de 10bits.

El módulo tiene cinco registros:

- Registro alto del resultado A/D (ADRESH)
- Registro bajo del resultado A/D (ADRESL)
- Registro de control A/D 0 (ADCON0)
- Registro de control A/D 1 (ADCON1)
- Registro de control A/D 2 (ADCON2)

El registro ADCON0, gobierna las operaciones del módulo A/D. El registro ADCON1 configura las funciones de los puertos. El registro ADCON2 configura el reloj, programa el tiempo de adquisición y justificación.

REGISTRO 21-1: ADCON0: REGISTRO DE CONTROL A/D 0

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0						
-	-	CHS3	CHS2	CHS1	CHS0	GO/DONE*	ADON						
U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0						
BIT 5-2		Bits selectores del canal analógico											
		0000 = canal 0 (AN0)											
		0001 = canal 1 (AN1)											
		0010 = canal 2 (AN2)											
		0011 = canal 3 (AN3)											
		0100 = canal 4 (AN4)											
		0101 = canal 5 (AN5) ^(1,2)											
		0110 = canal 6 (AN6) ^(1,2)											
		0111 = canal 7 (AN7) ^(1,2)											
		1000 = canal 8 (AN8)											
		1001 = canal 9 (AN9)											
		1010 = canal 10 (AN10)											
		1011 = canal 11 (AN11)											
		1100 = canal 12 (AN12)											
		1101 = No implementado ⁽²⁾											
		1110 = No implementado ⁽²⁾											
		1111 = No implementado ⁽²⁾											
BIT 1	GO/DONE*: Bit A/D del estado de la conversión Cuando ADON=1: 1 = conversión A/D en marcha 0 = A/D en reposo												
BIT 0	ADON: Bit de permiso del A/D 1 = módulo A/D permitido 0 = módulo A/D no permitido												

Nota 1: Estos canales no se implementan en los dispositivos de 28pines.

2: La ejecución de una conversión en los canales no implementados devolverá una medida flotante de la entrada.

REGISTRO 21-2: ADCON1: REGISTRO DE CONTROL A/D 1

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
-	-	VCFG1	VCFG0	PCFG3	PCFG2	PCFG1	PCFG0
U-0	U-0	R/W-0	R/W-0	R/W-0 ⁽¹⁾	R/W ⁽¹⁾	R/W ⁽¹⁾	R/W ⁽¹⁾
BIT 5	VCFG1:	Bit de configuración de la tensión de referencia (fuente de VREF-) 1 = VREF- (AN2) 0 = VSS					
BIT 4	VCFG0:	Bit de configuración de la tensión de referencia (fuente de VREF+) 1 = VREF+ (AN3) 0 = VDD					

BIT 3-0 PCFG3:PCFG0: Bits de configuración de los puertos A/D:

PCFG3: PCFG0	AN12	AN11	AN10	AN9	AN8	AN7 ⁽²⁾	AN6 ⁽²⁾	AN5 ⁽²⁾	AN4	AN3	AN2	AN1	AN0
0000 ⁽¹⁾	A	A	A	A	A	A	A	A	A	A	A	A	A
0001	A	A	A	A	A	A	A	A	A	A	A	A	A
0010	A	A	A	A	A	A	A	A	A	A	A	A	A
0011	D	A	A	A	A	A	A	A	A	A	A	A	A
0100	D	D	A	A	A	A	A	A	A	A	A	A	A
0101	D	D	D	A	A	A	A	A	A	A	A	A	A
0110	D	D	D	D	A	A	A	A	A	A	A	A	A
0111 ⁽¹⁾	D	D	D	D	D	A	A	A	A	A	A	A	A
1000	D	D	D	D	D	D	A	A	A	A	A	A	A
1001	D	D	D	D	D	D	D	A	A	A	A	A	A
1010	D	D	D	D	D	D	D	D	A	A	A	A	A
1011	D	D	D	D	D	D	D	D	D	A	A	A	A
1100	D	D	D	D	D	D	D	D	D	D	A	A	A
1101	D	D	D	D	D	D	D	D	D	D	D	A	A
1110	D	D	D	D	D	D	D	D	D	D	D	D	A
1111	D	D	D	D	D	D	D	D	D	D	D	D	D

A= Entrada analógica

D= E/S digital

Nota 1: El valor de POR de los bits PCFG depende del valor del bit de configuración PBADEN. Cuando PBADEN=1, PCFG<3: 0>=0000; cuando PBADEN=0, PCFG<3: 0>=0111.

2: AN5 con AN7 están disponibles solamente en los dispositivos 40/44pines.

REGISTRO 21-3: ADCON2: REGISTRO DE CONTROL A/D 2

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
ADFM	-	ACQT2	ACQT1	ACQT0	ADCS2	ADCS1	ADCS0
R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
BIT 7	ADFM:		Bit selecto del formato A/D del resultado 1 = derecho justificado 0 = izquierdo justificado				
BIT 5-3	ACQT2:ACQT0:	Bits selectores del tiempo de adquisición 111 = 20 T _{AD} 110 = 16 T _{AD} 101 = 12 T _{AD} 100 = 8 T _{AD} 011 = 6 T _{AD} 010 = 4 T _{AD} 001 = 2 T _{AD} 000 = 0 T _{AD} ⁽¹⁾					
BIT 2-0	ADCS2:ADCS0:	Bits selectores del reloj de conversión 111= F _{RC} (reloj A/D derivado del oscilador RC) ⁽¹⁾ 110= F _{OSC} /64 101= F _{OSC} /16 100= F _{OSC} /4 011= F _{RC} (reloj A/D derivado del oscilador RC) ⁽¹⁾ 010= F _{OSC} /32 001= F _{OSC} /8 000= F _{OSC} /2					

Nota 1: Si se selecciona la fuente de reloj A/D F_{RC}, se añade un retraso de un T_{CY} (ciclo de instrucción) antes de que el reloj A/D comience. Esto permite que la instrucción SLEEP se ejecute antes de comenzar una conversión.

La tensión analógico de referencia es seleccionable por software cualquier fuente de tensión positivo y negativo del dispositivo (V_{DD} y V_{SS}) o el nivel de tensión en los pines RA3/AN3/VREF+ y RA2/AN2/VREF-/CVREF.

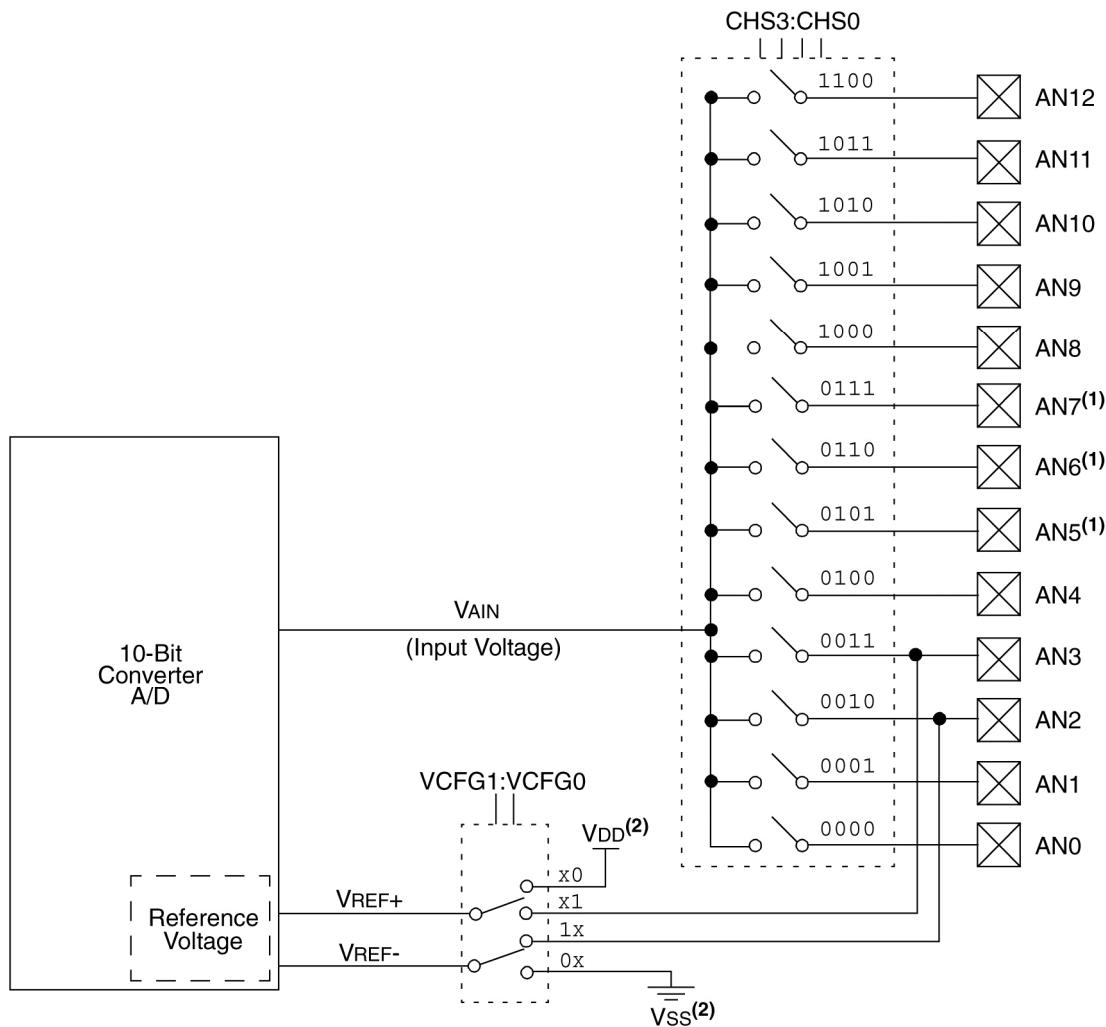
El convertidor A/D tiene la característica de poder funcionar mientras que el dispositivo está en modo sleep. Al funcionar en sleep, el reloj de conversión A/D debe derivar del oscilador interno RC A/D.

La salida de la muestra y del asimiento es la entrada en el convertidor, que genera el resultado vía aproximaciones sucesivas.

Un reset del dispositivo fuerza todos los registros a su estado de reset. Esto provoca que el módulo A/D se apague y la conversión en marcha se aborte.

Cada pin del puerto asociado al convertidor A/D puede configurarse como entrada analógica o como entrada-salida digital. Los registros ADRESH y ADRESL contienen el resultado de la conversión A/D. Cuando la conversión A/D se completa, el resultado se carga en el par de registros ADRESH:ADRESL, el bit GO/DONE* (registro ADCON0) se borra y se activa el flag de interrupción A/D, ADIF.

FIGURA 21-1: DIAGRAMA DE BLOQUES DEL A/D



Nota 1: Los canales de 5 a 7 no están disponibles en los dispositivos de 28 pines.

2: Los pines de E/S tienen un diodo de protección a V_{DD} y V_{SS} .

El valor en los registros ADRESH:ADRESL no se modifica con un POR. Los registros ADRESH:ADRESL contendrán datos desconocidos después de un POR.

Después de que el módulo A/D se haya configurado según lo deseado, el canal seleccionado debe adquirir antes de que comience la conversión. Los canales de entrada analógica deben tener sus bits TRIS correspondientes seleccionados como entrada. Después de esta adquisición el tiempo ha transcurrido, la conversión A/D puede comenzar. Se puede programar un periodo de adquisición para que ocurra entre activar el bit GO/DONE* y el comienzo real de la conversión.

Hay que seguir los siguientes pasos para realizar la conversión A/D:

1. Configurar el módulo A/D:

- Configura los pines analógicos, referencia de tensión y entrada-salida digital (ADC1CON)
- Seleccionar el canal A/D de entrada (ADC0CON)
- Seleccionar el tiempo de adquisición A/D (ADC2CON)
- Seleccionar el reloj e la conversión A/D (ADC2CON)
- Cambiar el módulo A/D (ADC0CON)

2. Configuran la interrupción A/D (si se desea):
 - Borrar el bit ADIF
 - Activar el bit ADIE
 - Activar el bit GIE
3. Esperar el tiempo requerido de adquisición (si procede).
4. Comienzo de la conversión:
 - Activar el bit GO/DONE* (registro ADCON0)
5. Esperar que termine la conversión A/D, por cualquiera:
 - La interrogación del bit GO/DONE* para ver si está borrado
 - Esperar la interrupción A/D
6. Leer los registros del resultado A/D (ADRESH:ADRESL); borrar el bit ADIF, si procede.
7. Para la conversión siguiente, ir al paso 1 o al paso 2, al que se necesite. El tiempo de la conversión A/D por bit es definido como T_{AD} . Se necesita esperar como mínimo $3T_{AD}$ antes de comenzar la siguiente adquisición.

FIGURA 21-2: FUNCIÓN DE TRANSFERENCIA A/D

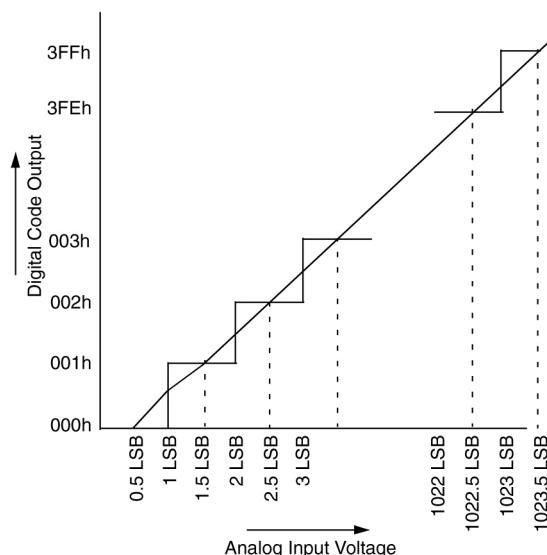
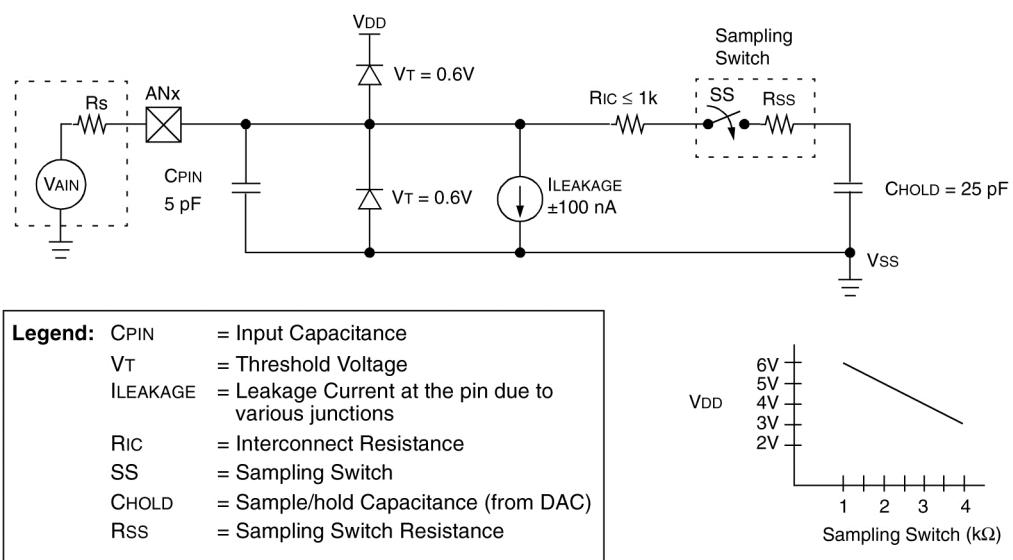


FIGURA 21-3: MODELO DE ENTRADA ANALÓGICA



21.1 REQUISITOS DE LA ADQUISICIÓN A/D

En el convertidor A/D, para satisfacer su exactitud especificada, la carga que sostiene el condensador (C_{HOLD}) debe permitirse que se cargue completamente para que entre el nivel de tensión del canal. La impedancia de la fuente (R_S) y la impedancia del cambio del muestreo interno (R_{SS}) afectan directamente al tiempo requerido para cargar el condensador C_{HOLD} . La impedancia del cambio del muestreo (R_{SS}) varía sobre la tensión del dispositivo (V_{DD}). La impedancia de la fuente afecta a la tensión compensando la entrada analógica (debido al pin de salida de corriente). **La impedancia recomendada máxima para las fuentes analógicas es de $2,5\text{k}\Omega$.** Después de que el canal de entrada analógico esté seleccionado (cambiado), el canal se debe muestrear por lo menos el tiempo mínimo de adquisición antes de comenzar una conversión.

Nota: Cuando comienza la conversión, el condensador se desconecta del pin de entrada.

Para calcular el tiempo mínimo de adquisición, se puede usar la ecuación 21-1. Esta ecuación asume que existe el error de $1/2\text{LSb}$ (1024 pasos para el A/D). El error de $1/2\text{LSb}$ es el error máximo permitido del A/D para conseguir su resolución específica.

El ejemplo 21-3 muestra el cálculo del tiempo mínimo de adquisición requerido T_{ACQ} . Este cálculo se base con los siguientes datos de aplicación:

$$C_{HOLD}=25\text{pF}$$

$$R_S=2,5\text{k}\Omega$$

$$\text{Error de la conversión} \leq 1/2 \text{ LSb}$$

$$V_{DD}=5\text{V}$$

$$R_{SS}=2\text{k}\Omega$$

$$\text{Temperatura}=85^\circ\text{C} \text{ (máximo del sistema)}$$

ECUACIÓN 21-1: ADQUISICIÓN TIEMPO

$$T_{ACQ} = \text{Tiempo Colocación Amplificador} + \text{Tiempo Carga} + C_{HOLD} + \text{Coeficiente Temperatura} = \\ = T_{AMP} + T_C + T_{COFF}$$

ECUACIÓN 21-2: TIEMPO MÍNIMO DE CARGA A/D

$$V_{HOLD} = (V_{REF} - (V_{REF} / 2048)) \bullet (1 - e^{(-T_C / C_{HOLD}(R_{IC} + R_{SS} + R_S))})$$

o

$$T_C = -(C_{HOLD})(R_{IC} + R_{SS} + R_S) \ln(1 / 2048)$$

ECUACIÓN 21-3: CÁLCULO DEL TIEMPO DE ADQUISICIÓN MÍNIMO

$$T_{ACQ} = T_{AMP} + T_C + T_{COFF}$$

$$T_{AMP} = 2\mu\text{s}$$

$$T_{COFF} = (Temp - 25^\circ\text{C})(0,02\mu\text{s} / ^\circ\text{C}) = (85^\circ\text{C} - 25^\circ\text{C})(0,02\mu\text{s} / ^\circ\text{C}) = 1,2\mu\text{s}$$

El coeficiente de temperatura sólo se necesita para temperaturas $> 25^\circ\text{C}$. Por debajo de 25°C , $T_{COFF}=0\text{ms}$.

$$T_C = -(C_{HOLD})(R_{IC} + R_{SS} + R_S) \ln(1 / 2048)\mu\text{s} = -(25\text{pF})(1\text{k}\Omega + 2\text{k}\Omega + 2,5\text{k}\Omega) \ln(0,0004883)\mu\text{s} = 1,05\mu\text{s}$$

$$T_{ACQ} = 0,2\mu\text{s} + 1,05\mu\text{s} + 1,2\mu\text{s} = 2,45\mu\text{s}$$

21.2 SELECCIONAR Y CONFIGURACIÓN EL TIEMPO DE ADQUISICIÓN TIEMPO

El registro ADCON2 permite que el usuario seleccione tiempo de adquisición que ocurre cada vez que se activa el bit GO/DONE*. También da a los usuarios la opción de utilizar un tiempo de adquisición automáticamente determinado. El tiempo de adquisición se puede determinar con los bits ACQT2:ACQT0 (ADCON2<5: 3>) que proporcionan una gama de 2 a 20 T_{AD} . Cuando se activa el bit GO/DONE*, el módulo A/D continúa muestreando la entrada con el tiempo de adquisición seleccionado, entonces comienza automáticamente una conversión. Desde que se programa el tiempo de adquisición, puede que no sea necesario esperar un periodo de adquisición entre la selección de un canal y activar el bit GO/DONE*.

Se selecciona la adquisición manual cuando ACQT2:ACQT0 = 000. Cuando se activa el bit GO/DONE*, se para el muestreo y comienza una conversión. El usuario es el responsable de asegurar que ha pasado el tiempo de adquisición requerido entre la selección de la entrada deseada y fijar el bit GO/DONE*. Esta opción es el estado por defecto en un reset de los bits ACQT2:ACQT0 y es compatible con los dispositivos que no ofrecen tiempos de adquisición programables.

En cualquier caso, cuando se termina la conversión, el bit GO/DONE* se borra, se activa el flag ADIF y A/D comienza a muestrear el canal actualmente seleccionado otra vez. Si se programa un periodo de adquisición, no hay nada que indique si el tiempo de adquisición ha terminado o si la conversión ha comenzado.

21.3 SELECCIONAR EL RELOJ DE CONVERSIÓN A/D

El tiempo A/D de conversión por bit se define como T_{AD} . La conversión A/D requiere 11 T_{AD} para la conversión de 10 bits. La fuente de reloj A/D de la conversión es seleccionable por software. Hay siete opciones posibles para el T_{AD} :

- 2 Tosc
- 4 Tosc
- 8 Tosc
- 16 Tosc
- 32 Tosc
- 64 Tosc
- Oscilador interno RC

Para las conversiones A/D correctas, el reloj de conversión A/D (T_{AD}) debe ser tan corto como sea posible pero mayor que un mínimo T_{AD} (para más información véase el parámetro 130 en la tabla 28-29).

TABLA 21-1: T_{AD} VS. FRECUENCIAS DE OPERACIÓN DE LOS DISPOSITIVOS

Fuente del reloj AD (T_{AD})		Máxima frecuencia del dispositivo	
Opción	ADCS2:ADCS0	PIC18FXXXX	PIC18LFXXXX ⁽⁴⁾
2 T_{OSC}	000	2.86MHz	1.43MHz
4 T_{OSC}	100	5.71MHz	2.86MHz
8 T_{OSC}	001	11.43MHz	5.72MHz
16 T_{OSC}	101	22.86MHz	11.43MHz
32 T_{OSC}	010	45.71MHz	22.86MHz
64 T_{OSC}	110	48.0MHz	45.71MHz
RC ⁽³⁾	x11	1.00MHz ⁽¹⁾	1.00MHz ⁽²⁾

Nota 1: La fuente RC tiene un T_{AD} de 4ms.

2: La fuente RC tiene un T_{AD} de 6ms.

3: Para dispositivos con frecuencias alrededor de 1MHz, el dispositivo tiene que estar en sleep durante la conversión o la exactitud del A/D está fuera de lo especificado.

4: Dispositivos de baja potencia

21.4 OPERACIÓN EN LOS MODOS DE AHORRO DE ENERGÍA

La selección del tiempo de adquisición automática y el reloj de conversión A/D se determinado en parte con la fuente de reloj y la frecuencia mientras se trabaje en un modo de ahorro de energía.

Si se espera que el A/D funcione mientras que el dispositivo esté dentro de un modo de ahorro de energía, los bits ACQT2:ACQT0 y ADCS2:ADCS0 en ADCON2 se deben actualizar con la fuente de reloj que se utilizará en este modo. Después de entrar en el modo, una adquisición o la conversión A/D puede comenzar. Una vez que haya comenzado, el dispositivo debe continuar controlado con la misma fuente de reloj hasta que la conversión termine.

Si se desea, el dispositivo se puede poner en modo reposo durante la conversión. Si la frecuencia de reloj del dispositivo es menos de 1MHz, se tiene que seleccionar la fuente de reloj RC A/D.

Operar en el modo sleep requiere que se seleccione el reloj A/D F_{RC} . Si los bits ACQT2:ACQT0 se fijan a '000' y comienza una conversión, la conversión tendrá un retraso de un ciclo de instrucción para permitir la ejecución de la instrucción SLEEP y entrada del modo sleep. El bit IDLEN (OSCCON<7>) tiene que estar borrado antes de comenzar la conversión.

21.5 CONFIGURACIÓN DE LOS PINES DE LOS PUERTOS ANALÓGICOS

Los registros ADCON1, TRISA, TRISB y TRISE configuran los pines de los puertos A/D. Los pines tienen que estar como entradas analógicas, deben tener sus bits correspondientes del TRIS activos (entrada). Si es el bit TRIS está borrado (salida), el nivel de salida digital (V_{OH} o V_{OL}) se convertirá.

La operación A/D es independiente del estado de los bits CHS3:CHS0 y los bits TRIS.

Nota 1: Al leer el registro PORT, todos los pines configurados como canales de entrada analógica se leerán como borrado (un nivel bajo). Los pines configurados

como entrada digital convertirán como entradas analógicas. Los niveles analógicos de la entrada digital estarán bien convertidos.

2: Los niveles analógicos en cualquier pin definido como entrada digital puede causar al buffer de entrada digital que consuma una corriente fuera de los límites de la especificación de dispositivo.

3: El bit PBADEN en el registro de configuración 3H configura los pines de PORTB a resetar como pines analógicos o digitales dependiendo de como estén los bits PCFG0 de ADCON1 después el reset.

21.6 CONVERSIONES A/D

La figura 21-4 muestra la operación del convertidor A/D después de que se haya activado el bit GO/DONE* y los bits ACQT2:ACQT0 se hayan borrado. Comienza una conversión después de que la siguiente instrucción permita la entrada en el modo sleep antes de comenzar la conversión.

La figura 21-5 muestra la operación del convertidor A/D después de que se hayan fijado los bits GO/DONE*, y ACQT2:ACQT0 a '010' y seleccionar el tiempo de adquisición a $4T_{AD}$ antes del comienzo de la conversión.

Despejar el bit GO/DONE* durante una conversión aborta la conversión actual. El par de registros del resultado A/D no se actualiza cuando termina una parte de la conversión A/D. Esto significa que los registros ADRESH:ADRESL continuarán conteniendo el valor de la conversión anterior (o el último valor escrito en los registros ADRESH:ADRESL).

Después de que termine o se aborte la conversión A/D, se necesita esperar $2T_{AD}$ antes de que la adquisición siguiente pueda comenzar. Después de esta espera, la adquisición en el canal seleccionado se activa automáticamente.

Nota: El bit de GO/DONE* no se debe activar dentro de la misma instrucción que activa el A/D.

21.7 DESCARGA

La fase de descarga se utiliza para inicializar el valor de la matriz del condensador. La matriz se descarga antes de cada muestra. Esta característica ayuda a optimizar la ganancia unitaria del amplificador como el circuito siempre carga la matriz del condensador, más bien que cargar/descarga se basa en la medida de valores anteriores.

FIGURA 21-4: CICLOS T_{AD} EN LA CONVERSIÓN A/D (ACQT<2:0>=000, T_{AQQT}=0)

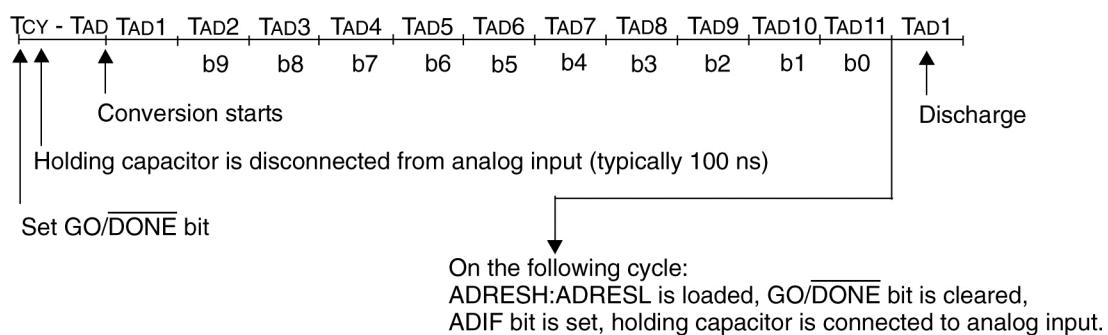
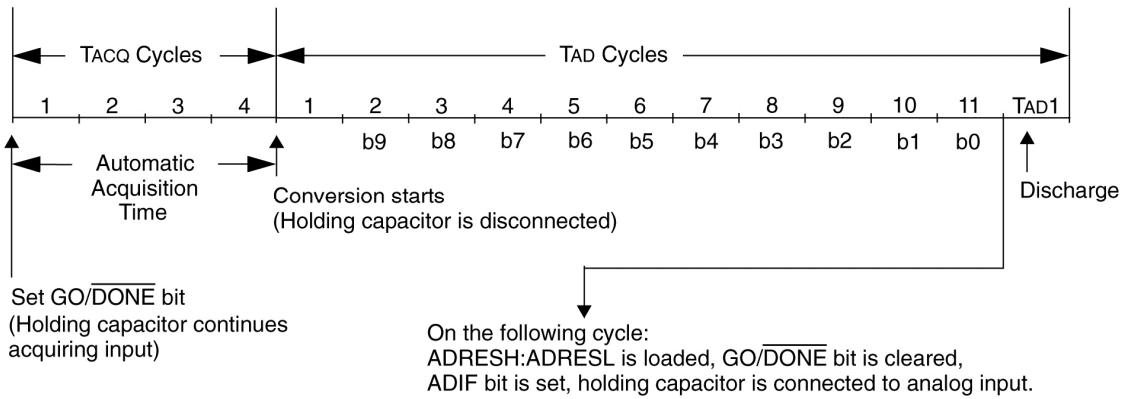


FIGURA 21-5: CICLOS T_{AD} EN LA CONVERSIÓN A/D (ACQT<2:0>=010, $T_{AQQT}=4T_{AD}$)



21.8 USO DEL DISPARADOS DEL CCP2

Una conversión A/D puede empezar con el disparo de un evento especial del módulo CCP2. Esto requiere que los bits CCP2M3:CCP2M0 (CCP2CON<3:0>) se programen como ‘1011’ y que el módulo A/D esté activo (ADON esté activo). Cuando ocurre un disparo, el bit GO/DONE* se activará, empezando la adquisición y conversión A/D y el Timer1 (o Timer3) se resetea para repetir automáticamente el periodo de adquisición A/D con el uso de un mínimo de software (mover ADRESH:ADRESL a las localizaciones deseadas). Se tiene que seleccionar el canal de entrada analógico apropiado y un tiempo de adquisición mínimo elegido por el usuario o un T_{ACQ} apropiado antes de que el disparo del evento especial active el bit GO/DONE* (comience la conversión).

Si el módulo A/D no está permitido (ADON está borrado), el disparo del evento especial lo ignora el módulo A/D pero reseteará el contador Timer1 (o Timer3).

TABLA 21-2: REGISTROS ASOCIADOS A LAS OPERACIONES A/D

Nombre	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTCON	GIE/GIEH	PEIE/GIEL	TMROIE	INT0IE	RBIE	TMROIF	INT0IF	RBIF
PIR1	SPPIF ⁽⁴⁾	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF
PIE1	SPPIE ⁽⁴⁾	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE
IPR1	SPPIP ⁽⁴⁾	ADIP	RCIP	TXIP	SSPIP	CCP1IP	TMR2IP	TMR1IP
PIR2	OSCFIF	CMIF	USBIF	EEIF	BCLIF	HLVDIF	TMR3IF	CCP2IF
PIE2	OSCFIE	CMIE	USBIE	EEIE	BCLIE	HLVDIE	TMR3IE	CCP2IE
IPR2	OSCFIP	CMIP	USBIP	EEIP	BCLIP	HLVDIP	TMR3IP	CCP2IP
ADRESH	A/D Result Register High Byte							
ADRESL	A/D Result Register Low Byte							
ADCON0	—	—	CHS3	CHS2	CHS1	CHS0	GO/DONE*	ADON
ADCON1	—	—	VCFG1	VCFG0	PCFG3	PCFG2	PCFG1	PCFG0
ADCON2	ADFM	—	ACQT2	ACQT1	ACQT0	ADCS2	ADCS1	ADCS0
PORTA	—	RA6 ⁽²⁾	RA5	RA4	RA3	RA2	RA1	RA0
TRISA	—	TRISA6 ⁽²⁾	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0
PORTB	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0
TRISB	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0
LATB	LATB7	LATB6	LATB5	LATB4	LATB3	LATB2	LATB1	LATB0
PORTE	RDPU ⁽⁴⁾	—	—	—	RE3 ^(1,3)	RE2 ⁽⁴⁾	RE1 ⁽⁴⁾	RE0 ⁽⁴⁾
TRISE ⁽⁴⁾	—	—	—	—	—	TRISE2	TRISE1	TRISE0
LATE ⁽⁴⁾	—	—	—	—	—	LATE2	LATE1	LATE0

Leyenda: -no implementado, se leen ‘0’. Las casillas sombreadas no se utilizan con el módulo A/D.

Nota 1: Implementado cuando el Master Clean está desactivado (configuración del bit MCLRE=0).

2: RA6 y su latch asociado y los bits de dirección se activan como puertos de E/S dependiendo de la configuración del oscilador; en otro caso, se leen ‘0’.

3: El puerto RE3 está disponible sólo como entrada cuando el bit de configuración MCLRE es ‘0’.

4: Estos registros y/o bits no están disponibles en los dispositivos de 28pines.

22.0 EI MÓDULO COMPARADOR

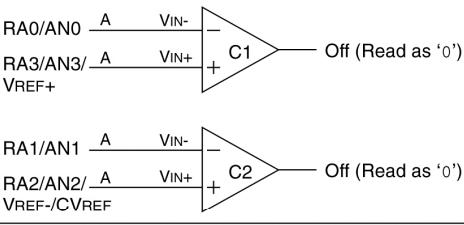
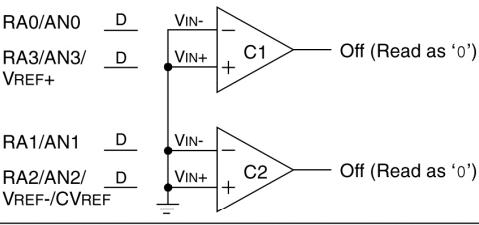
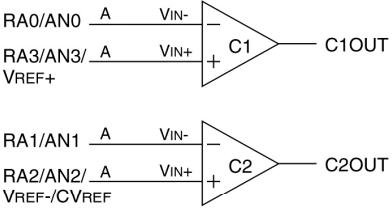
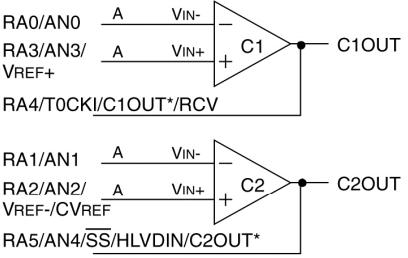
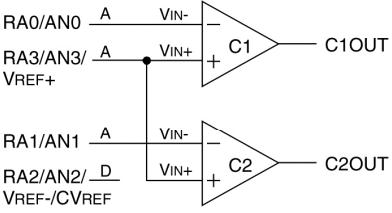
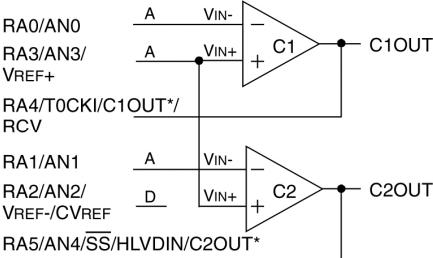
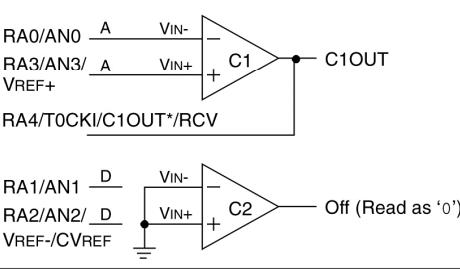
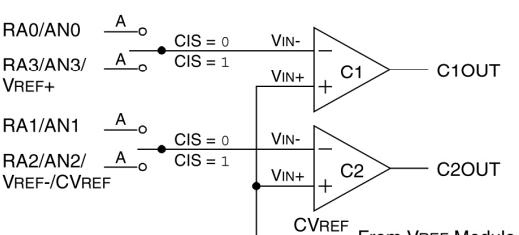
El módulo comparador analógico contiene dos comparadores que se pueden configurar de muchas maneras. Las entradas se pueden elegir de las entradas analógicas multiplexadas con los pines RA0 a RA5, como el módulo de tensión de referencia del chip. Las salidas digitales (normales o invertidas) están disponibles en los niveles de los pines o también se pueden leer a través del registro de control.

El registro CMCON selecciona la configuración de la entrada y de la salida del comparador.

REGISTRO 22-1: CMCON: REGISTRO DE CONTROL DEL COMPARADOR

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
C2OUT R-0	C1OUT R-0	C2INV R/W-0	C1INV R/W-0	CIS	CM2	CM1	CM0
BIT 7	C2OUT:	Bit de salida del comparador 2 Cuando C2INV=0: 1 = C2 V _{IN+} > C2 V _{IN-} 0 = C2 V _{IN+} < C2 V _{IN-} Cuando C2 _{IN} V=1: 1 = C2 V _{IN+} < C2 V _{IN-} 0 = C2 V _{IN+} > C2 V _{IN-}					
BIT 6	C1OUT:	Bit de salida del comparador 1 Cuando C1INV=0: 1 = C1 V _{IN+} > C1 V _{IN-} 0 = C1 V _{IN+} < C1 V _{IN-} Cuando C1INV=1: 1 = C1 V _{IN+} < C1 V _{IN-} 0 = C1 V _{IN+} > C1 V _{IN-}					
BIT 5	C2INV:	Bit inversor de la salida del comparador 2 1 = salida C2 invertida 0 = salida C2 no invertida					
BIT 4	C1INV:	Bit inversor de la salida del comparador 1 1 = salida C1 invertida 0 = salida C1 no invertida					
BIT 3	CIS:	Bit del cambio de la entrada del comprador Cuando CM2:CM0=110: 1 =C1 V _{IN-} se conecta con RA3/AN3/VREF+ El C2 V _{IN-} se conecta con RA2/AN2/VREF-/CVREF 0 =C1 V _{IN-} se conecta con RA0/AN0 El C2 V _{IN-} se conecta con RA1/AN1					
BIT 2-0	CM2:CM0:	Bits del modo del comparador La figura 22-1 muestra los modos del comparador y los ajustes de los bits CM2:CM0.					

FIGURA 22-1: MODOS DE OPERACIÓN DE LAS E/S DEL COMPARADOR

Comparators Reset CM2:CM0 = 000 	Comparators Off (POR Default Value) CM2:CM0 = 111 
Two Independent Comparators CM2:CM0 = 010 	Two Independent Comparators with Outputs CM2:CM0 = 011 
Two Common Reference Comparators CM2:CM0 = 100 	Two Common Reference Comparators with Outputs CM2:CM0 = 101 
One Independent Comparator with Output CM2:CM0 = 001 	Four Inputs Multiplexed to Two Comparators CM2:CM0 = 110 
A = Analog Input, port reads zeros always D = Digital Input * Setting the TRISA<5:4> bits will disable the comparator outputs by configuring the pins as inputs.	CIS (CMCON<3>) is the Comparator Input Switch * Setting the TRISA<5:4> bits will disable the comparator outputs by configuring the pins as inputs.

22.1 CONFIGURACIÓN DEL COMPARADOR

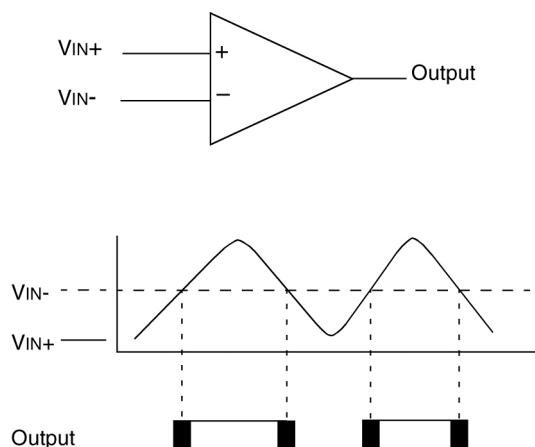
Hay ocho modos de operación para los comparadores. Los bits CM2:CM0 del registro CMCON se utilizan para seleccionar estos modos. El registro TRISA controla la dirección de los datos de los pines del comparador en cada modo. Si cambia el modo del comparador, el nivel de salida del comparador puede que no sea válido para el modo especificado de cambio de retraso.

Nota: Las interrupciones del comparador se tienen que desactivar durante el cambio de modo del comparador. En otro caso, puede que ocurra una falsa interrupción.

22.2 OPERACIONES DEL COMPARADOR

En la figura 22-2 se muestra un comparador simple, junto con la relación entre los niveles de entrada de datos analógicos y la salida digital. Cuando la entrada analógica en V_{IN+} es menor que la entrada analógica V_{IN-} , la salida del comparador es un nivel digital bajo. Cuando la entrada analógica V_{IN+} es mayor que la entrada analógica V_{IN-} , la salida del comparador es un nivel digital alto. Las áreas sombreadas de la salida del comparador en la figura 22-2 representan la incertidumbre, debido a las compensaciones de la entrada y al tiempo de reacción.

FIGURA 22-2: COMPARADOR SIMPLE



22.3 REFERENCIA DEL COMPARADOR

Dependiendo del modo de funcionamiento del comparador, se puede utilizar cualquier referencia externa o interna de tensión. La señal analógica presente en V_{IN-} se compara con la señal en V_{IN+} y la salida digital del comparador se ajusta consecuentemente.

22.3.1 SEÑAL DE REFERENCIA EXTERNA

Cuando son externas se utilizan las referencias de tensión, el módulo del comparador se puede configurar para tener los comparadores funcionar desde la misma u otra fuente de referencia. Sin embargo, las aplicaciones del detector del umbral pueden necesitar la misma referencia. La señal de referencia debe estar entre V_{SS} y V_{DD} y se puede aplicar a cualquier pin del/los comparador/es.

22.3.2 SEÑAL DEREFERENCIA INTERNA

El módulo del comparador también permite la selección de referencia interna de tensión interna generada por el módulo comparador de tensión de referencia.

La referencia interna está disponible solamente en el modo donde cuatro entradas se multiplexan a dos comparadores ($CM2:CM0=110$). En este modo, la tensión interna de referencia se aplica al pin $VIN+$ de ambos comparadores.

22.4 TIEMPO DE RESPUESTA DEL COMPARADOR

El tiempo de reacción es el tiempo mínimo, después de seleccionar una nueva tensión de referencia o fuente entrada, antes de que la salida del comparador tenga un nivel válido. Si la referencia interna cambia, se tiene que considerar el máximo retraso de la tensión interna de referencia al usar las salidas del comparador. Si no, se tiene que utilizar el máximo retraso de los comparadores.

22.5 SALIDAS DEL COMPARADOR

Las salidas del comparador se leen con el registro CMCON. Estos bits son inalterables. Las salidas del comparador se pueden poner directamente a los pines RA4 y RA5 de E/S. Cuando está permitido, los multiplexores en la trayectoria de salida de los pines RA4 y RA5 cambiará y la salida de cada pin no estará sincronizada con la del comparador. La incertidumbre de cada uno de los comparadores se relaciona con la tensión compensada de la entrada y el tiempo de reacción dado en las especificaciones.

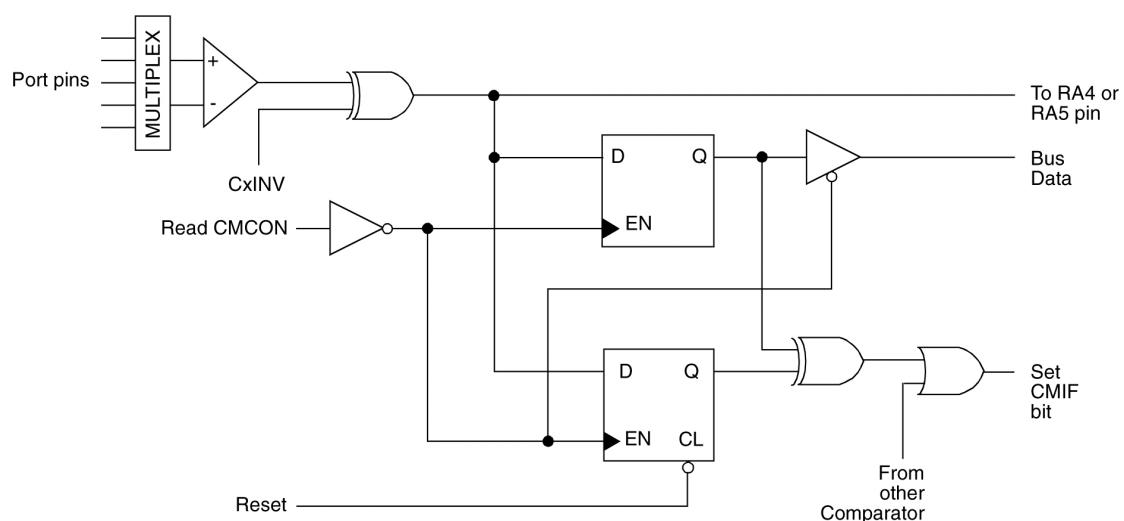
Los bits TRISA activarán/desactivarán la salida de los pines RA4 y RA5 mientras esté en este modo.

La polaridad de las salidas del comparador se puede cambiar usando los bits C2INV y C1INV (CMCON<5: 4>).

Nota 1: Al leer el registro PORT, todos los pines configurados como entradas analógicas se leerán ‘0’. Los pines configurados como entradas digitales convertir una entrada analógica según las especificaciones de entrada del disparador Schmitt.

2: Los niveles analógicos de cualquier pin definidos como entrada digital pueden hacer que el buffer de entrada consuma más corriente que la que se especifica.

FIGURA 22-3: DIAGRAMA DE BLOQUES DEL COMPARADOR DE SALIDA



22.6 INTERRUPCIONES DEL COMPARADOR

Se activa el flag de interrupción del comparador siempre que haya un cambio en el valor de la salida de cualquier comparador. El software necesitará mantener la información sobre estado de los bits de la salida, dependiendo de la lectura de CMCON<7:6>, para determinar el cambio real que ha ocurrido. El bit CMIF (PIR2<6>) es el flag de interrupción del comparador. El bit CMIF se tiene que resetear borrándolo. Puesto que está también posible escribir un ‘1’ en este registro, se puede iniciar una interrupción simulada.

El bit CMIE (PIE2<6>) y el bit PEIE (INTCON<6>) se tienen que activar para permitir la interrupción. Además, el bit GIE (INTCON<7>) tiene que estar activo. Si ninguno de estos bits están activos, no se permiten las interrupciones, aunque el bit CMIF todavía se active si ocurre una condición de interrupción.

Nota: Si un cambio en el registro de CMCON (C1OUT o C2OUT) ocurre cuando se está ejecutando la operación de lectura (comienzo del ciclo Q2), entonces el flag de interrupción CMIF (PIR2<6>) puede que no se active.

El usuario, en la rutina de interrupción, puede borrar la interrupción de la manera siguiente:

- a) Cualquier lectura o escritura de CMCON terminará con condición de error de unión.
- b) Borrar el flag CMIF.

Una condición de error de unión continuará activando el flag CMIF. La lectura de CMCON terminará con la condición de error de unión y permitirá que el flag CMIF se borre.

22.7 OPERACIONES DEL COMPARADOR DURANTE EL MODO SLEEP

Cuando un comparador está activo y el dispositivo se pone en modo sleep, el comparador sigue estando activo y la interrupción es funcional si está permitida. Esta interrupción despertará al dispositivo del modo sleep, cuando está permitida. Cada comparador operacional consumirá corriente adicional, según las indicaciones de las especificaciones del comparador. Para reducir al mínimo el consumo de energía en el modo sleep, se apagan los comparadores (CM2: CM0 = 111) antes de entrar en el modo sleep. Si el dispositivo despierta del sleep, no le afecta al contenido del registro CMCON.

22.8 EFECTOS DE UN RESET

Un reset del dispositivo fuerza al registro CMCON a su estado de reset, haciendo que los módulos del comparador se apaguen (CM2:CM0=111). Sin embargo, los pines de entrada (RA0 a RA3) se configuran como entradas analógicas por defecto. La configuración de E/S para éstos pines se determina con el ajuste de los bits PCFG3:PCFG0 (ADCON1<3:0>). Por lo tanto, la corriente del dispositivo se reduce al mínimo cuando las entradas analógicas están presentes en el tiempo de reset.

22.9 CONSIDERACIONES EN LA CONEXIÓN DE LAS INTRADAS ANALÓGICAS

Como los pines analógicos se conectan a una salida digital, hay diodos polarizados inversamente a V_{DD} y V_{SS}. La entrada analógica, tiene que estar entre la corriente de salida V_{SS} y V_{DD}. Si la tensión de entrada se desvía de este rango más de 0,6V en cualquier dirección, uno de los diodos se polariza directamente y ocurre una condición de latch-up. Se recomienda una impedancia máxima de la fuente de 10kΩ para fuentes analógicas. Cualquier componente externo conectado a un pin de entrada analógica, como un condensador o un diodo zener, debe tener muy poca pérdida de corriente.

FIGURA 22-4: MODELO DE ENTRADA DEL COMPARADOR ANALÓGICO

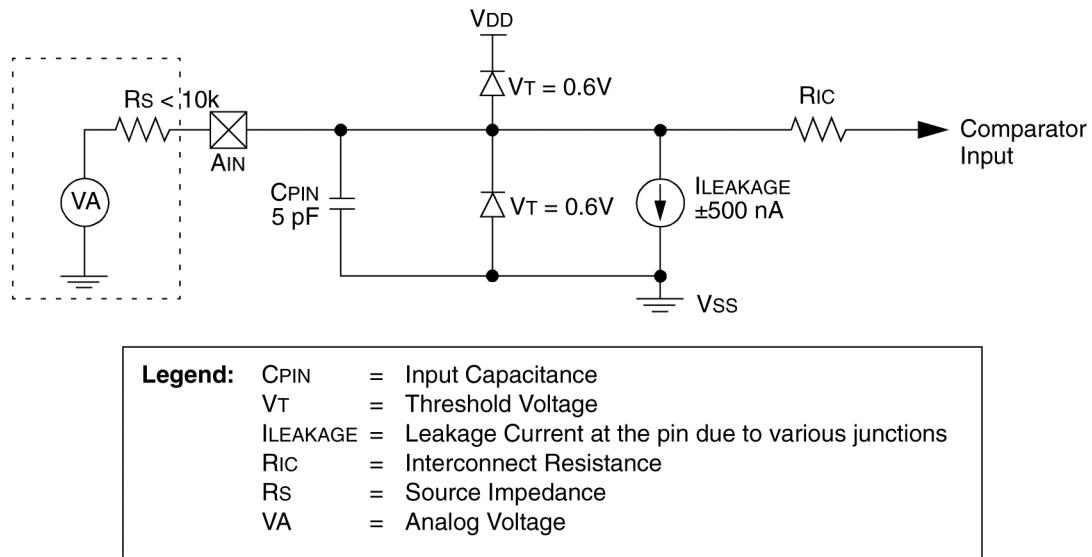


TABLA 22-1: REGISTROS ASOCIADOS AL MÓDULO COMPARADOR

Nombre	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CMCON	C2OUT	C1OUT	C2INV	C1INV	CIS	CM2	CM1	CM0
CVRCON	CVREN	CVROE	CVRR	CVRSS	CVR3	CVR2	CVR1	CVR0
INTCON	GIE/GIEH	PEIE/GIEL	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF
PIR2	OSCFIF	CMIF	USBIF	EEIF	BCLIF	HLVDIF	TMR3IF	CCP2IF
PIE2	OSCFIE	CMIE	USBIE	EEIE	BCLIE	HLVDIE	TMR3IE	CCP2IE
IPR2	OSCFIP	CMIP	USBIP	EEIP	BCLIP	HLVDIP	TMR3IP	CCP2IP
PORTA	—	RA6 ⁽¹⁾	RA5	RA4	RA3	RA2	RA1	RA0
LATA	—	LATA6 ⁽¹⁾	LATA5	LATA4	LATA3	LATA2	LATA1	LATA0
TRISA	—	TRISA6 ⁽¹⁾	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0

Leyenda: -no implementado, se leen '0'. Las casillas sombreadas no se utilizan con el módulo comparador.

Nota 1: PORTA<6> y su dirección y bit de latch se configuran individualmente como pin de puerto en varios modos de oscilador. Cuando está desactivado estos bits se leen '0'.

23.0 MÓDULO COMPARADOR DE TENSIÓN DE REFERENCIA

La tensión del comparador de referencia es una red de 16 resistencias que proporciona una referencia seleccionable de tensión. Aunque su propósito primario es proporcionar una referencia para los comparadores analógicos, se puede utilizar también independientemente de ellos.

La red de resistencias se divide en segmentos para proporcionar dos gamas de valores de CV_{REF} y tiene una función de caída de energía para conservar la energía cuando la referencia no se está utilizando. La fuente de referencia del módulo se puede obtener de dispositivo V_{DD}/V_{SS} o una referencia externa de tensión.

23.1 CONFIGURACIÓN DEL MÓDULO COMPARADOR DE TENSIÓN DE REFERENCIA

El módulo comparador de tensión de referencia se controla con el registro CVRCON. El comparador de tensión de referencia proporciona dos gamas de tensión de salida, cada uno con 16 niveles distintos. La gama que se utiliza se selecciona con el bit CVRR (CVRCON<5>). La diferencia primaria entre las gamas es el tamaño de los pasos seleccionados con los bits CVREF (CVR3:CVR0), con una gama de resolución más fina. Las ecuaciones utilizadas para calcular la salida de referencia de la tensión del comparador son:

Si $CVRR=1$:

$$CV_{REF}=((CVR3:CVR0)/24) \times CV_{RSRC}$$

Si $CVRR=0$:

$$CV_{REF}=(CVRSRC/4)+(((CVR3:CVR0)/32) \times CV_{RSRC})$$

La tensión de la fuente de referencia del comparador puede venir de V_{DD} y de V_{SS} , o de V_{REF+} externo y V_{REF-} que se multiplexan con RA2 y RA3. La fuente de tensión se selecciona con el bit CVRSS (CVRCON<4>).

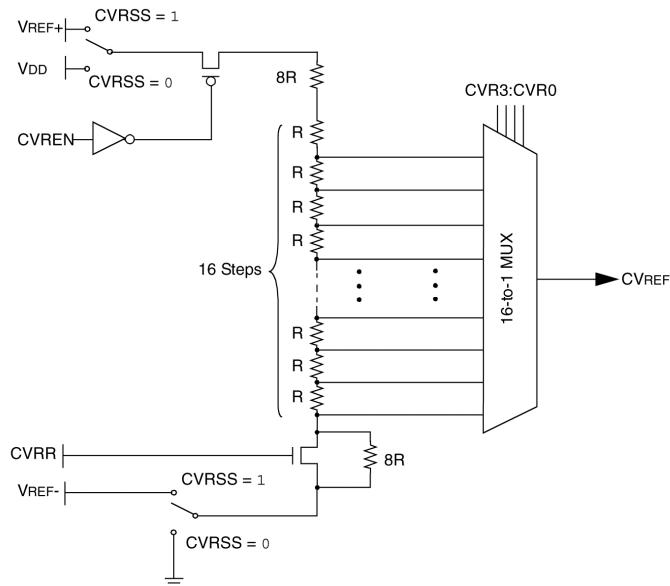
Se tiene que considerar el periodo de configuración de comparador de tensión de referencia al cambiar la salida CV_{REF} .

REGISTRO 23-1: CVRCON: REGISTRO DE CONTROL DE LA TENSIÓN DE REFERENCIA DEL COMPARADOR

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
CVREN R/W-0	CVROE ⁽¹⁾ R/W-0	CVRR R/W-0	CVRSS R/W-0	CVR3 R/W-0	CVR2 R/W-0	CVR1 R/W-0	CVR0 R/W-0
BIT 7	CVREN:		Bit de permiso de comparador de tensión de referencia 1= Circuito CV _{REF} encendido 0= Circuito CV _{REF} apagado				
BIT 6	CVROE:		Bit de permiso de la salida del comparador V _{REF} ⁽¹⁾ 1= El nivel de tensión CV _{REF} también sale por el pin RA2/AN2/V _{REF} /CV _{REF} 0= La tensión CV _{REF} se desconecta del pin RA2/AN2/V _{REF} /CV _{REF}				
BIT 5	CVRR:		Bit de selección del rango del comparador V _{REF} 1 = 0 a 0,667CV _{RSRC} , con el tamaño de paso CV _{RSRC} /24 (gama baja) 0 = 0,25CV _{RSRC} a 0,75CV _{RSRC} , con el tamaño de paso CV _{RSRC} /32 (alta gama)				
BIT 4	CVRSS:		Bit de selección de la fuente del comparador V _{REF} 1 = fuente de referencia del comparador, CV _{RSRC} =(V _{REF+})-(V _{REF-}) 0 = fuente de referencia del comparador, CV _{RSRC} =V _{DD} -V _{SS}				
BIT 3-0	CVR3:CVR0:		Bits de selección del valor del comparador V _{REF} (0<=(CVR3:CVR0)<=15) Cuando CVRR=1: CV _{REF} =((CVR3:CVR0)/24)•(CV _{RSRC}) Cuando CVRR=0: CV _{REF} =(CV _{RSRC} /4)+((CVR3:CVR0)/32)•(CV _{RSRC})				

Nota 1: CVROE elimina el ajuste del bit TRISA<2>.

FIGURA 23-1: DIAGRAMA DE BLOQUES DEL COMPARADOR DE TENSIÓN DE REFERENCIA



23.2 EXACTITUD/ERROR DE LA TENSIÓN DE REFERENCIA

La gama completa de la tensión de referencia no se puede observar debido a la construcción del módulo. Los transistores en la tapa y el fondo de la red de la escala de resistencias guardan CV_{REF} la referencia de la fuente. La tensión de referencia deriva de la fuente de referencia; por lo tanto, la salida CV_{REF} cambia con fluctuaciones en esa fuente.

23.3 OPERACIÓN EN MODO SLEEP

Cuando el dispositivo despierta del sleep con una interrupción o un desbordamiento del perro guardián, no le afecta al contenido del registro CVRCON. Para reducir al mínimo la consumición actual en el modo sleep, la tensión de referencia debe desactivarse.

23.4 EFECTOS DE UN RESET

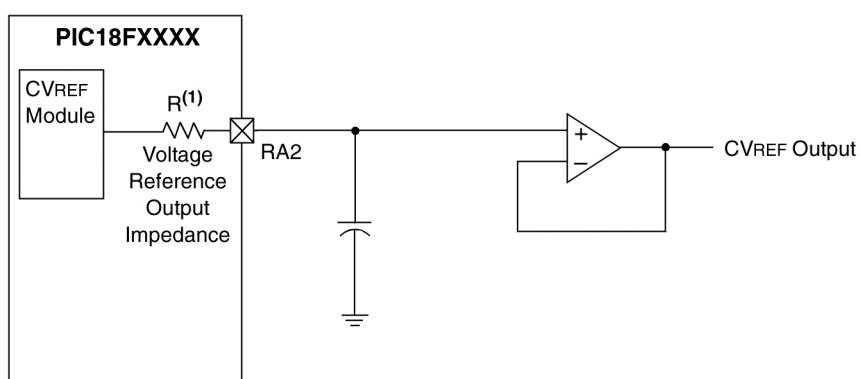
Un reset del dispositivo inhabilita la referencia de tensión borrando el bit, CVREN (CVRCON<7>). Este reset también desconecta la referencia del pin RA2 borrando el bit, CVROE (CVRCON<6>) y selecciona el rango de alta tensión borrando el bit, CVRR (CVRCON<5>). El valor de los bits selectores CVR también se borra.

23.5 CONSIDERACIONES DE LA CONEXIÓN

El módulo de tensión de referencia funciona independientemente del módulo comparador. La salida del generador de referencia se puede conectar al pin RA2 si el bit TRISA<2> y el bit CVROE están activos. Permitiendo a la tensión poner la salida en RA2 cuando se configura como una entrada digital aumentará la consumición de corriente. Conectando RA2 como salida digital con CVRSS permitido también aumentará el consumo de corriente.

El pin RA2 se puede utilizar como salida D/A simple con capacidad de control limitada. Debido a la capacidad limitada de corriente actual, se debe utilizar un buffer en la salida de tensión de referencia para las conexiones externas a VREF.

FIGURA 23-2: EJEMPLO DE BUFFER DE SALIDA DEL COMPARADOR DE TENSIÓN DE REFERENCIA



Nota 1: R depende de los bits de configuración de la tensión de referencia, CVRCON<5> y CVRCON<3:0>.

TABLA 23-1: REGISTROS ASOCIADOS AL COMPARADOR DE TENSIÓN DE REFERENCIA

Nombre	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CVRCON	CVREN	CVROE	CVRR	CVRSS	CVR3	CVR2	CVR1	CVR0
CMCON	C2OUT	C1OUT	C2INV	C1INV	CIS	CM2	CM1	CM0
TRISA	—	TRISA6(1)	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0

Leyenda: -no implementado, se leen ‘0’. Las casillas sombreadas no se utilizan con el módulo comparador de la tensión de referencia.

Nota 1: PORTA<6> y su dirección y bit de latch se configuran individualmente como pin de puerto en varios modos de oscilador. Cuando está desactivado estos bits se leen ‘0’.

24.0 DETECTOR DE ALTA/BAJA TENSIÓN (HLVD)

Los dispositivos PIC18F2455/2550/4455/4550 tienen un módulo detector de alta/baja tensión (HLVD). Éste es un circuito programable que permite al usuario especificar la tensión del cambio del dispositivo y la dirección del cambio en ese punto. Si el dispositivo experimenta un cambio mayor, se activa un flag de interrupción. Si se permite la interrupción, la ejecución del programa saltará a la dirección de vector de interrupción y el software responde a la interrupción.

El registro de control que detecta la alta/baja tensión controla totalmente la operación del Módulo HLVD. Esto permite que el circuito “lo apague” el usuario bajo control por software que reduce al mínimo la consumición actual del dispositivo.

REGISTRO 24-1: HLVDCON: REGISTRO DE CONTROL DEL DETECTOR DE ALTA/BAJA TENSIÓN

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
VDIRMAG	-	IRVST	HLVDEN	HLVDL3	HLVDL2	HLVDL1	HLVDL0
R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-1	R/W-0	R/W-1
BIT 7	VDIRMAG:		Bit selector de la magnitud de la dirección de tensión 1 = el acontecimiento ocurre cuando la tensión iguala o excede el punto (HLVDL3:HLVDL0) 0 = el acontecimiento ocurre cuando la tensión iguala o cae por debajo del punto (HLVDL3: HLVDL0)				
BIT 5	IRVST:		Flag de la tensión interna de referencia estable 1 = indica que la tensión detectada generará el flag de interrupción en la gama especificada 0 = indica que la tensión detectada no generará el flag de interrupción en la gama especificada y la interrupción HLVD no se debe permitir				
BIT 4	HLVDEN:		Bit de permiso del detector de la alta/baja tensión 1 = HLVD permitido 0 = HLVD inhabilitado				
BIT 3-0	HLVDL3:HLVDL0:		Bits del límite de detección de tensión 1111 = se utiliza la entrada analógica externa (la entrada es el pin HLVDIN) 1110 = ajuste máximo ... 0000 = ajuste mínimo				

El módulo se permite activando el bit HLVDEN. Cada vez que se permite el módulo HLVD, el circuito requiere un cierto tiempo para estabilizarse. El bit IRVST es un bit inalterable y se utiliza para indicar cuando el circuito es estable. El módulo puede generar una sola interrupción después de que el circuito sea estable y se activa IRVST.

El bit VDIRMAG determina la operación total del módulo. Cuando se borra VDIRMAG, el módulo supervisa las bajadas en V_{DD} por debajo de un punto predeterminado. Cuando se activa el bit, la supervisión del módulo se realiza en las subidas de V_{DD} sobre el punto de ajuste.

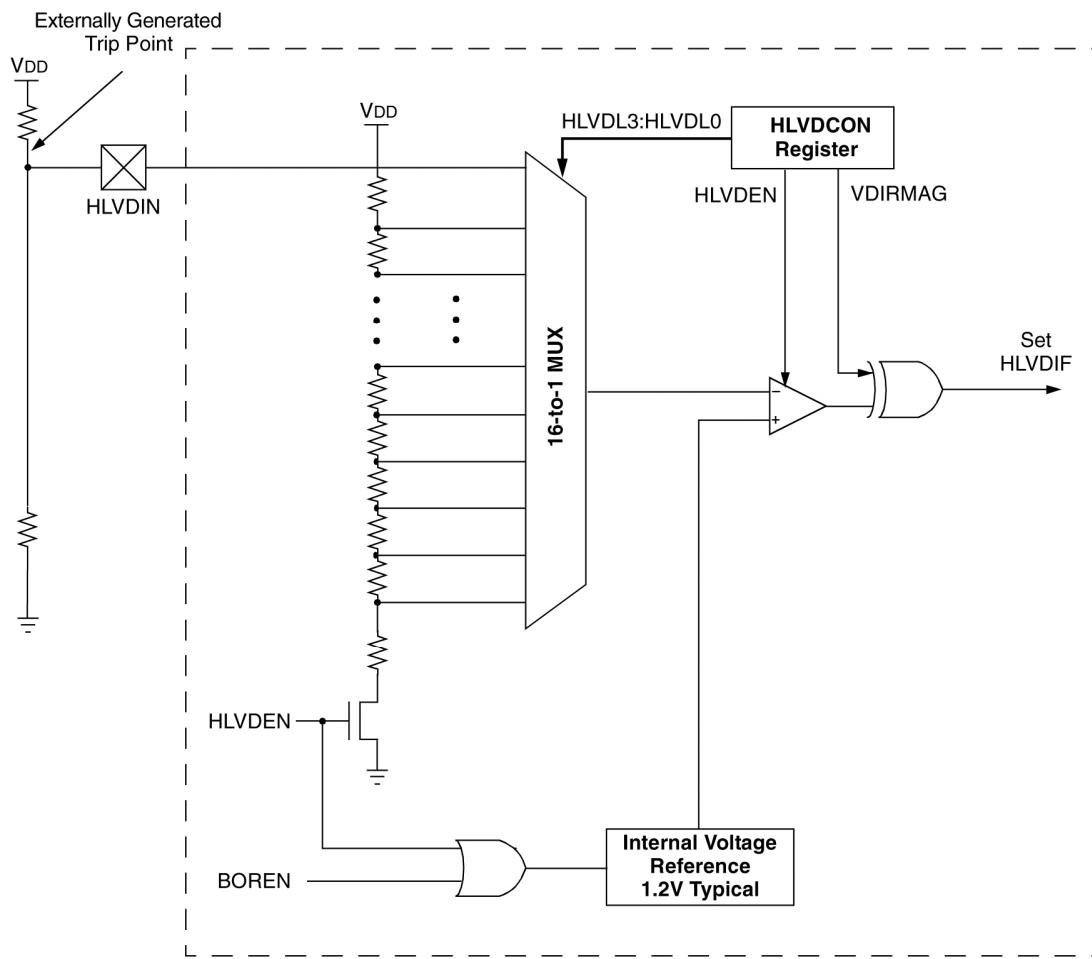
24.1 OPERACIÓN

Cuando se permite el módulo HLVD, un comparador usa una tensión de referencia interna generada como punto de ajuste. El punto de ajuste se compara con el punto de cambio, donde cada nodo en el divisor de la resistencia representa la tensión del punto de cambio. “La tensión del punto de cambio” es el nivel de tensión en el cual el dispositivo detecta un acontecimiento de subida o bajada de tensión, dependiendo de la configuración del módulo. Cuando la fuente de tensión es igual al punto de cambio, la tensión que llega a la matriz de resistencias es igual a la tensión de referencia interna generada por el módulo de tensión de referencia. El comparador entonces genera una señal de interrupción activando el bit HLVDIF.

La tensión del punto de cambio es programable por software a cualquiera de los 16 valores. El punto de cambio se selecciona programando los bits HLVDL3:HLVDL0 (HLVDCON<3: 0>).

El módulo HLVD tiene una característica adicional que permite al usuario proveer la tensión de cambio al módulo mediante una fuente externa. Se permite este modo cuando los bits, HLVDL3:HLVDL0, se fijan a ‘1111’. En este estado, la entrada del comparador se multiplexa con la entrada del pin externa, HLVDIN. Esto da a los usuarios flexibilidad porque permite que configuren la interrupción del detector de alta/baja tensión para que ocurra en la gama de tensión en funcionamiento.

FIGURA 24-1: DIAGRAMA DE BLOQUES DEL MÓDULO HLVD (CON ENTRADA EXTERNA)



24.2 CONFIGURACIÓN DEL HLVD

Se necesitan los siguientes pasos para instalar el módulo HLVD:

1. Inhabilitar el módulo despejando el bit HLVDEN (HLVDCON<4>).
2. Escribir el valor del punto de cambio en los bits HLVDL3:HLVDL0.
3. Activar el bit VDIRMAG para detectar la subida de tensión (VDIRMAG=1) o la bajada de tensión (VDIRMAG=0).
4. Permitir el módulo HLVD activando el bit HLVDEN.
5. Despejar el flag de interrupción HLVD, HLVDIF (PIR2<2>), que puede estar activo.
6. Permitir la interrupción HLVD, si se quieren las interrupciones, activando los bits HLVDIE y GIE/GIEH (PIE2<2> e INTCON<7>). Una interrupción se genera hasta que se activa el bit IRVST.

24.3 CONSUMICIÓN DE CORRIENTE

Cuando se permite el módulo, el comparador HLVD y el divisor de tensión consumirán parásitos atmosféricos.

Dependiendo de la aplicación, el módulo HLVD no necesita funcionar constantemente. Para disminuir los requisitos actuales, el circuito HLVD puede permitirse en periodos cortos donde se compruebe la tensión. Después de hacer la comprobación, el módulo HLVD puede desactivarse.

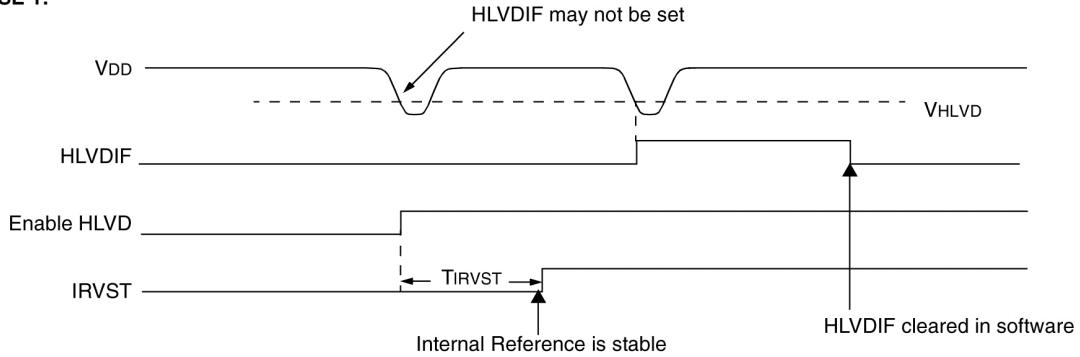
24.4 TIEMPO DE ARRANQUE DE HLVD

La tensión interna de referencia del módulo HLVD, la puede utilizar otro circuito interno, por ejemplo el reset programable por cese de tensión. Si el HLVD u otros circuitos que usan la tensión de referencia se inhabilitan para bajar la consumición actual del dispositivo, la tensión de referencia del circuito requerirá un tiempo para ser estable antes de que una condición de bajada o subida de tensión pueda detectarse fiablemente. Este tiempo de arranque, T_{IRVST} , es un intervalo que es independiente de la velocidad de reloj del dispositivo.

El flag de interrupción de HLVD permanece desactivado hasta que T_{IRVST} expire y se alcance una tensión estable de referencia. Por esta razón, los excesos leves del punto de cambio puede que no se detecten.

FIGURA 24-2: OPERACIÓN DE DETECCIÓN DE BAJADA DE TENSIÓN (VDIRMAG=0)

CASE 1:



CASE 2:

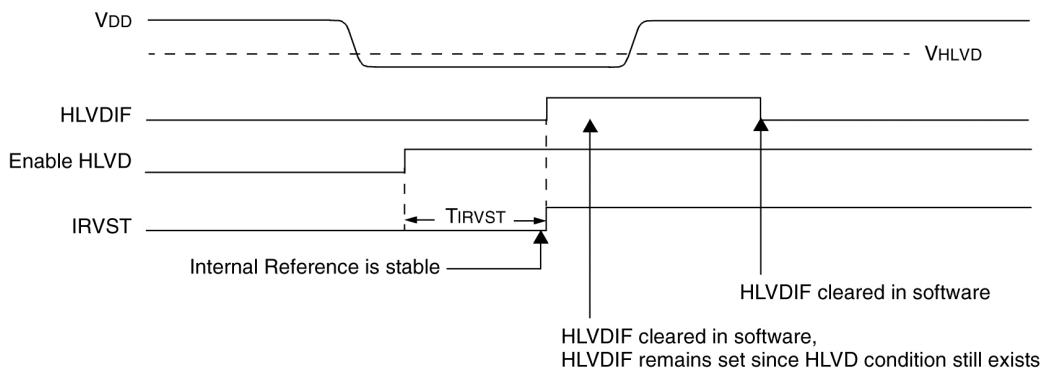
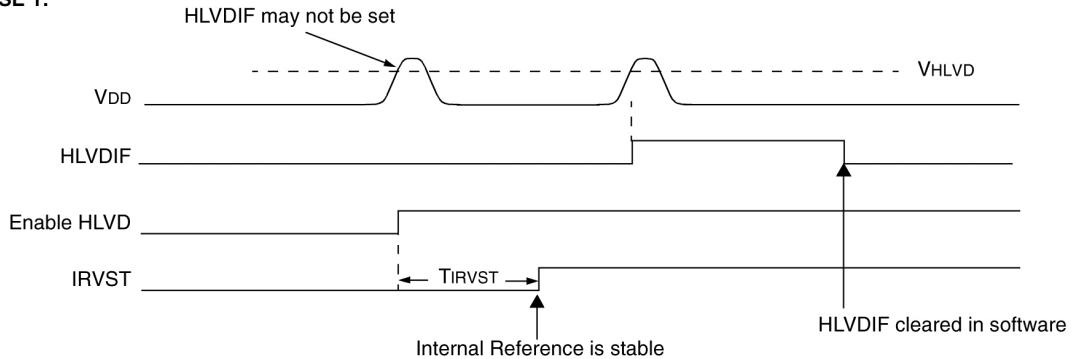
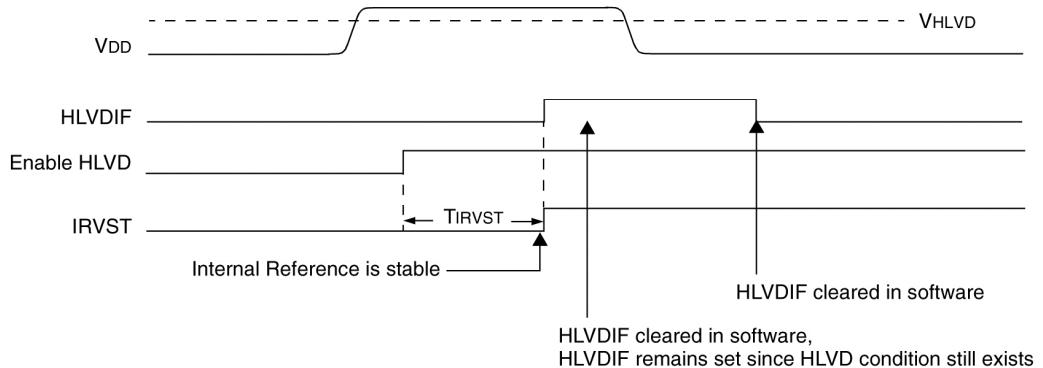


FIGURA 24-3: OPERACIÓN DE DETECCIÓN DE SUBIDA DE TENSIÓN (VDIRMAG=1)

CASE 1:



CASE 2:



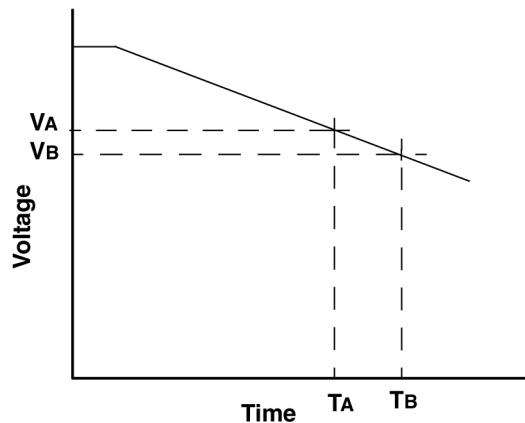
24.5 APPLICACIONES

En muchas aplicaciones, se desea la capacidad de detectar un pico bajo o la subida de un umbral particular. Para el ejemplo, el módulo HLVD podía estar configurado para detectar periódicamente la conexión o desconexión del Bus Serie Universal (USB). Esto asume que el dispositivo se acciona con una fuente de tensión más baja que cuando el USB está desconectado. Una activación indicaría que se ha detectado un nivel alto de tensión, por ejemplo, 3,3V a 5V (tensión en el USB) y viceversa al desconectar. Esta característica podía ahorrar el diseño de algún suplemento de componentes y una señal de activación (pin de entrada).

En aplicaciones generales con batería, el cuadro 24-4 muestra una curva posible de tensión. En un cierto tiempo, la tensión del dispositivo disminuye. Cuando la tensión del dispositivo alcanza la tensión V_A , la lógica del HLVD genera una interrupción en el tiempo, T_A . La interrupción podría causar la ejecución

de un ISR, el cuál permitiría que la paliación realizara “tareas de economización” y realizan una parada controlada antes de que la tensión salga del rango de tensión válida para el dispositivo en T_B . El HLVD, así, daría a la aplicación una ventana de tiempo, representada por la diferencia entre T_A y T_B , para salir con seguridad.

FIGURA 24-4: APPLICACIÓN TÍPICA DEL DETECTOR DE ALTA/BAJA TENSIÓN



Legend: V_A = HLVD trip point
 V_B = Minimum valid device operating voltage

24.6 OPERACIONES EN EL MODO SLEEP

Cuando se activa, el circuito HLVD continúa funcionando durante el modo sleep. Si la tensión del dispositivo cruza el punto de cambio, el bit HLVDIF se activará y el dispositivo se despertará del sleep. La ejecución del dispositivo continuará en el vector interrupción si están permitidas las interrupciones globalmente.

24.7 EFECTOS DE UN RESET

Un reset del dispositivo fuerza a los registros a su estado de reset. Esto provoca que el módulo HLVD se apague.

TABLA 24-1: REGISTROS ASOCIADOS AL MÓDULO DETECTOR DE ALTA/BAJA TENSIÓN

Nombre	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
HLVDCON	VDIRMAG	—	IRVST	HLVDEN	HLVDL3	HLVDL2	HLVDL1	HLVDL0
INTCON	GIE/GIEH	PEIE/GIEL	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF
PIR2	OSCFIF	CMIF	USBIF	EEIF	BCLIF	HLVDIF	TMR3IF	CCP2IF
PIE2	OSCFIE	CMIE	USBIE	EEIE	BCLIE	HLVDIE	TMR3IE	CCP2IE
IPR2	OSCFIP	CMIP	USBIP	EEIP	BCLIP	HLVDIP	TMR3IP	CCP2IP

Leyenda: -=no implementado, se leen ‘0’. Las casillas sombreadas no se utilizan con el módulo HLVD.

25.0 CARACTERÍSTICAS ESPECIALES DE LA CPU

Los dispositivos PIC18F2455/2550/4455/4550 incluyen varias características para maximizar la confiabilidad y reducir el coste al mínimo con la eliminación de componentes externos. Éstos son:

- Selección del oscilador
- Resets:
 - ✓ Reset por subida de tensión (POR)
 - ✓ Temporizador de inicio (PWRT)
 - ✓ Temporizador de inicio del oscilador (OST)
 - ✓ Reset por cese de tensión (BOR)
- Interrupciones
- Temporizador perro guardián (WDT)
- Monitor de reloj a prueba de fallos
- Arranque a dos velocidades
- Código de protección
- Localizaciones de identificación
- Programación serie “en Circuito”

El oscilador se puede configurar para cada aplicación dependiendo de la frecuencia, la energía, la exactitud y el coste.

Además de sus contadores de tiempo del arranque y del oscilador temporizador de inicio para los resets, los dispositivos PIC18F2455/2550/4455/4550 tienen un temporizador perro guardián, que se permite permanentemente con los bits de configuración o el software de control (si está inhabilitado).

La inclusión de un oscilador interno RC también proporciona ventajas adicionales de un monitor de reloj a prueba de fallos (FSCM) y arranque de dos velocidades. FSCM proporciona una supervisión oculta del reloj del periférico y un cambio automático en caso de fallo. El arranque a dos velocidades permite al código ejecutarse casi inmediatamente en el inicio, mientras que la fuente de reloj primario termina su retraso de inicio.

Todas estas características se permiten y se configuran activando los bits apropiados del registro de configuración.

25.1 BITS DE CONFIGURACIÓN

Los bits de configuración pueden programarse (leerse como ‘0’) o no programarse (leerse como ‘1’) para seleccionar varias configuraciones del dispositivo. Estos bits están mapeados en la posición de memoria del programa 300000h.

El usuario observará que la dirección 300000h está más allá de memoria del programa de usuario. De hecho, pertenece a memoria de la configuración (300000h-3FFFFFh), la cual puede alcanzarse solamente usando la lectura y escritura de tablas.

La programación de los registros de configuración se hace de manera similar a la programación de la memoria flash. El bit WR del registro EECON1 comienza una escritura auto-temporizada en el registro de configuración. En modo de operación normal, una instrucción TBLWT, con el TBLPTR señalando al registro de configuración, selecciona la dirección y los datos para escribir el registro de configuración. Al activar el bit WR comienza una “escritura larga” en el registro de configuración. En los registros de configuración se escribe un byte a la vez. Al escribir o borrar una célula de configuración, una instrucción TBLWT puede escribir un ‘1’ o un ‘0’ en la célula.

TABLA 25-1: BITS DE CONFIGURACIÓN E IDENTIFICADORES DEL DISPOSITIVO

Nombre del Registro		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Valor por defecto
300000h	CONFIG1L	—	—	USBDIV	CPUDIV1	CPUDIV0	PLLDIV2	PLLDIV1	PLLDIV0	--000000
300001h	CONFIG1H	IESO	FCMEN	—	—	FOSC3	FOSC2	FOSC1	FOSC0	00-0101
300002h	CONFIG2L	—	—	VREGEN	BORV1	BORV0	BOREN1	BOREN0	PWRTEN*	-011111
300003h	CONFIG2H	—	—	—	WDTPS3	WDTPS2	WDTPS1	WDTPS0	WDTEN	---1111
300005h	CONFIG3H	MCLRE	—	—	—	—	LPT1OSC	PBADEN	CCP2MX	1---011
300006h	CONFIG4L	DEBUG*	XINST	ICPRT ⁽³⁾	—	—	LVP	—	STVREN	100--1-1
300008h	CONFIG5L	—	—	—	—	CP3 ⁽¹⁾	CP2	CP1	CP0	----1111
300009h	CONFIG5H	CPD	CPB	—	—	—	—	—	—	11----
30000Ah	CONFIG6L	—	—	—	—	WRT3 ⁽¹⁾	WRT2	WRT1	WRT0	----1111
30000Bh	CONFIG6H	WRTD	WRTB	WRTC	—	—	—	—	—	111----
30000Ch	CONFIG7L	—	—	—	—	EBTR3 ⁽¹⁾	EBTR2	EBTR1	EBTR0	----1111
30000Dh	CONFIG7H	—	EBTRB	—	—	—	—	—	—	-1----
3FFFFEh	DEVID1	DEV2	DEV1	DEV0	REV4	REV3	REV2	REV1	REV0	xxxxxxxx ⁽²⁾
3FFFFFh	DEVID2	DEV10	DEV9	DEV8	DEV7	DEV6	DEV5	DEV4	DEV3	00010010 ⁽²⁾

Leyenda: x= desconocido; u=sin cambios; --no implementado, se leen '0'

Nota 1: No implementado en los dispositivos PIC18Fx455; mantener el bit activo

2: Ver los registros 25-13 y 25-14 para los valores de DEVID. Los registros DEVID sólo se pueden leer y no los puede programar el usuario.

3: Disponible en los PIC18F4455/4550 en encapsulado TQFP de 44pines. Mantener este bit borrado en el resto de dispositivos.

REGISTRO 25-1: CONFIG1L: REGISTRO 1 DE CONFIGURACIÓN BAJO (DIRECCIÓN DEL BYTE 300000h)

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
-	-	USBDIV	CPUDIV1	CPUDIV0	PLLDIV2	PLLDIV1	PPDIV0
U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
BIT 5	USBDIV:		Bit de selección del reloj del USB (usado sólo en el modo Full-Speed; UCFG: FSEN = 1)				
			1 = fuente del reloj del USB viene de los 96MHz PLL divididos por 2				
			0 = fuente del reloj del USB viene directamente del bloque del oscilador primario sin postscaler				
BIT 4-3	CPUDIV1:CPUDIV0:		Bits de selección del Postscaler del reloj del sistema				
			Para modos del oscilador XT, HS, EC y ECIO:				
			11 = oscilador primario se divide por 4 para obtener el reloj del sistema				
			10 = oscilador primario se divide por 3 para obtener el reloj del sistema				
			01 = oscilador primario se divide por 2 para obtener el reloj del sistema				
			00 = oscilador primario usado directamente en el reloj del sistema (ningún postscaler)				
			Para modos del oscilador XTPPLL, HSPLL, ECPLL y ECPIO:				
			11 = 96MHz PLL se divide por 6 para obtener el reloj del sistema				
			10 = 96MHz PLL se divide por 4 para obtener el reloj del sistema				
			01 = 96MHz PLL se divide por 3 para obtener el reloj del sistema				
			00 = 96MHz PLL se divide por 2 para obtener el reloj del sistema				
BIT 2-0	PLLDIV2:PLLDIV0:		Bits de selección del Prescaler del PLL				
			111 = se divide por 12 (la entrada del oscilador de 48MHz)				
			110 = se divide por 10 (la entrada del oscilador de 40MHz)				
			101 = se divide por 6 (la entrada del oscilador de 24MHz)				
			100 = se divide por 5 (la entrada del oscilador de 20MHz)				
			011 = se divide por 4 (la entrada del oscilador de 16MHz)				
			010 = se divide por 3 (la entrada del oscilador de 12MHz)				
			001 = se divide por 2 (la entrada del oscilador de 8MHz)				
			000 = ningún prescaler (conducen 4MHz de entrada del oscilador PLL directamente)				

REGISTRO 25-2: CONFIG1H: REGISTRO 1 DE CONFIGURACIÓN ALTO (DIRECCIÓN DEL BYTE 300001h)

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
IESO	FCMEN	-	-	FOSC3 ⁽¹⁾	FOSC2 ⁽¹⁾	FOSC1 ⁽¹⁾	FOSC0 ⁽¹⁾
R/P-0	R/P-0	U-0	U-0	R/P-0	R/P-1	R/P-0	R/P-1
BIT 7	IESO:			Bit del intercambio interno y externo del oscilador 1 = modo del intercambio del oscilador permitido 0 = modo del intercambio del oscilador inhabilitado			
BIT 6	FCMEN:			Bit de permiso del monitor del reloj a prueba de fallos 1 = monitor del reloj a prueba de fallos permitido 0 = monitor del reloj a prueba de fallos inhabilitado			
BIT 3-0	FOSC3:FOSC0:			Bits de selección del oscilador ⁽¹⁾ 111x= oscilador HS, PLL permitido (HSPLL) 110x= oscilador HS (HS) 1011= oscilador interno, oscilador HS utilizado por el USB (INTHS) 1010= oscilador interno, XT utilizado por USB el (INTXT) 1001= oscilador interno, CLKO en RA6, EC utilizado por el USB (INTCKO) 1000= oscilador interno, puerto en RA6, EC utilizado por el USB (INTIO) 0111= Oscilador EC, PLL permitido, CLKO en RA6 (ECPLL) 0110= Oscilador EC, PLL permitido, puerto en RA6 (ECPIO) 0101= Oscilador EC, CLKO en RA6 (EC) 0100 = Oscilador EC, puerto en RA6 (ECIO) 001x= oscilador XT, PLL permitido (XTPLL) 000x = oscilador de XT (XT)			

Nota 1: El microcontrolador y el módulo USB utilizan el oscilador seleccionado como su fuente del reloj en los modos XT, HS y EC. El módulo USB utiliza el oscilador indicado XT, HS o EC como su fuente del reloj siempre que el microcontrolador utilice el oscilador interno.

REGISTRO 25-3: CONFIG2L: REGISTRO 2 DE CONFIGURACIÓN BAJO (DIRECCIÓN DEL BYTE 300002h)

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
-------	-------	-------	-------	-------	-------	-------	-------

	-	VREGEN	BORV1	BIRV0	BOREN1 ⁽¹⁾	BOREN1 ⁽¹⁾	PWRTEN ^{*(1)}
BIT5	U-0	R/P-0-0	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
	VREGEN:	Bit de permiso del regulador interno de tensión del USB 1 = regulador de tensión del USB permitido 0 = regulador de tensión del USB inhabilitado					
BIT 4-3	BORV1:BORV0:	Bits de ajuste de la tensión de los resets por caída de tensión 11 = ajuste mínimo					
		.					
		.					
		.					
		00 = ajuste máximo					
BIT 2-1	BOREN1:BOREN0:	Bit de permiso del reset por caída de tensión 11= reset permitido en hardware solamente (SBOREN desactivado) 10= reset permitido en hardware solamente e inhabilitado en modo sleep (SBOREN desactivado) 01= reset permitido y controlado por software (SBOREN permitido) 00= reset desactivado en hardware y software					
BIT 0	PWRTEN:	Bit de permiso del temporizador de inicio (PWRT) ⁽¹⁾ 1 = PWRT inhabilitado 0 = PWRT permitido					

Nota 1: El temporizador de inicio se desempareja del reset de caída de tensión, permitiendo que estas características sean controladas independientemente.

REGISTRO 25-4: CONFIG2H: REGISTRO 2 DE CONFIGURACIÓN ALTO (DIRECCIÓN DEL BYTE 300003h)

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
-------	-------	-------	-------	-------	-------	-------	-------

			USBDIV	CPUDIV1	CPUDIV0	PLLDIV2	PLLDIV1	PPDIV0
U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
BIT 4-1	WDTPS3:WDTPS0: Bits selectores del Postscaler del temporizador del perro guardián							
	1111 = 1:32,768							
	1110 = 1:16,384							
	1101 = 1:8,192							
	1100 = 1:4,096							
	1011 = 1:2,048							
	1010 = 1:1,024							
	1001 = 1:512							
	1000 = 1:256							
	0111 = 1:128							
	0110 = 1:64							
	0101 = 1:32							
	0100 = 1:16							
	0011 = 1:8							
	0010 = 1:4							
	0001 = 1:2							
	0000 = 1:1							
BIT 0	WDTEN: Bit de permiso del temporizador del perro guardián 1 = WDT permitido 0 = WDT inhabilitado (el bit de control es SWDTEN)							

REGISTRO 25-5: CONFIG3H: REGISTRO 3 DE CONFIGURACIÓN ALTO (DIRECCIÓN DEL BYTE 300005h)

	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
MCLRE	-	-	-	-	-	LPT1OSC	PBADEN	CCP2MX
R/P-1	U-0	U-0	U-0	U-0	U-0	R/P-0	R/P-1	R/P-1
BIT 7	MCLRE: Bit de permiso del pin MCLR* 1 = pin MCLR permitido, pin RE3 inhabilitado 0 = pin RE3 permitido, pin MCLR inhabilitado							
BIT 2	LPT1OSC: Bit de permiso de la potencia del oscilador Timer1 1 = Timer1 configurado como baja potencia 0 = Timer1 configurado como alta potencia							
BIT 1	PBADEN: Bit de permiso del PORTB A/D (Afecta al estado del reset de ADCON1. ADCON1 controla la configuración del pin PORTB<4:0>.) 1 = los pines PORTB<4:0> se configuran como canales de la entrada analógica en el reset 0 = los pines PORTB<4:0> se configuran como E/S digital en el reset							
BIT 0	CCP2MX: Bit CCP2 MUX 1 = la entrada-salida CCP2 se multiplexa con RC1 0 = la entrada-salida CCP2 se multiplexa con RB3							

REGISTRO 25-6: CONFIG4L: REGISTRO 4 DE CONFIGURACIÓN BAJO (DIRECCIÓN DEL BYTE 300006h)

	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
DEBUG*	XINST	ICPRT ⁽¹⁾	-	-	-	LVP	-	STVREN

	R/P-1	R/P-0	R/P-0	U-0	U-0	R/P-1	U-0	R/P-1
BIT 7		DEBUG*:		Bit de permiso de la depuración oculta 1 = depuración oculta inhabilitada, RB6 y RB7 configurados como pines E/S generales 0 = depuración oculta permitida, RB6 y RB7 se dedica a eliminar errores “en Circuito”				
BIT 6		XINST:		Bit de permiso del sistema de instrucciones extendidas 1 = extensión del sistema de instrucción y modo de dirección indexado permitidos 0 = extensión del sistema de instrucción y modo de dirección indexado inhabilitados (modo herencia)				
BIT 5		ICPRT:		Bit de permiso del puerto de programación/depuración “en Circuito” (ICPORT) ⁽¹⁾ 1 = ICPORT permitido 0 = ICPORT inhabilitado				
BIT 2		LVP:		Bit de permiso de un Sólo-Proveedor ICSP™ 1 = Sólo-Proveedor ICSP permitido 0 = Sólo-Proveedor ICSP inhabilitado				
BIT 0		STVREN:		Bit de permiso de reset por desbordamiento superior/inferior de la pila 1 = Reset permitido 0 = Reset no permitido				

Nota 1: Disponible solamente en los dispositivos PIC18F4455/4550 de 44 pines con encapsulado TQFP. Borrar este bit en el resto de los dispositivos.

REGISTRO 25-7: CONFIG5L: REGISTRO 5 DE CONFIGURACIÓN BAJO (DIRECCIÓN DEL BYTE 300008h)

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
-	-	-	-	CP3 ⁽¹⁾	CP2	CP1	CP0
U-0	U-0	U-0	U-0	R/C-1	R/C-1	R/C-1	R/C-1

BIT 3	CP3:	Bit de protección del código ⁽¹⁾ 1 = el bloque 3 (006000-007FFFh) no es el código de protección 0 = el bloque 3 (006000-007FFFh) es el código de protección
BIT 2	CP2:	Bit de protección del código 1 = el bloque 2 (004000-005FFFh) no es el código de protección 0 = el bloque 2 (004000-005FFFh) es el código de protección
BIT 1	CP1:	Bit de protección del código 1 = el bloque 1 (002000-003FFFh) no es el código de protección 0 = el bloque 1 (002000-003FFFh) es el código de protección
BIT 0	CP0:	Bit de protección del código 1 = el bloque 0 (000800-001FFFh) no es el código de protección 0 = el bloque 0 (000800-001FFFh) es el código de protección

Nota 1: No implementado en los dispositivos PIC18FX455; mantener este bit activo.

REGISTRO 25-8: CONFIG5H: REGISTRO 5 DE CONFIGURACIÓN ALTO (DIRECCIÓN DEL BYTE 300009h)

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
CPD	CP8	-	-	-	-	-	-
R/C-1	R/C-1	U-0	U-0	U-0	U-0	U-0	U-0
BIT 7	CPD:	Bit de protección del código de los datos EEPROM 1 = la EEPROM no se protege 0 = la EEPROM se protege					
BIT 6	CPB:	Bit de protección del código del bloque de arranque 1 = el bloque de arranque (000000-0007FFh) no se protege 0 = el bloque de arranque (000000-0007FFh) se protege					

REGISTRO 25-9: CONFIG6L: REGISTRO 6 DE CONFIGURACIÓN BAJO (DIRECCIÓN DEL BYTE 30000Ah)

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
-	-	-	-	WRT3 ⁽¹⁾	WRT2	WRT1	WRT0
U-0	U-0	U-0	U-0	R/C-1	R/C-1	R/C-1	R/C-1
BIT 3	WRT3:	Bit de protección de la escritura ⁽¹⁾					

		1 = el bloque 3 (006000-007FFFh) no es la protección contra la escritura 0 = el bloque 3 (006000-007FFFh) es la protección contra la escritura
BIT 2	WRT2:	Bit de protección de la escritura 1 = el bloque 2 (004000-005FFFh) no es la protección contra la escritura 0 = el bloque 2 (004000-005FFFh) es la protección contra la escritura
BIT 1	WRT1:	Bit de protección de la escritura 1 = el bloque 1 (002000-003FFFh) no es la protección contra la escritura 0 = el bloque 1 (002000-003FFFh) es la protección contra la escritura
BIT 0	WRT0:	Bit de protección de la escritura 1 = el bloque 0 (000800-001FFFh) o (001000-001FFFh) no es la protección contra la escritura 0 = el bloque 0 (000800-001FFFh) o (001000-001FFFh) es la protección contra la escritura

Nota 1: No implementado en los dispositivos PIC18FX455; mantener este bit activo.

REGISTRO 25-10: CONFIG6H: REGISTRO 6 DE CONFIGURACIÓN ALTO (DIRECCIÓN DEL BYTE 30000Bh)

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
WRTD	WRTB	WRTC ⁽¹⁾	-	-	-	-	-
R/C-1	R/C-1	R/C-1	U-0	U-0	U-0	U-0	U-0
BIT 7	WRTD:		Bit de protección de la escritura de datos EEPROM 1 = los datos EEPROM no tienen protección de escritura 0 = los datos EEPROM tienen protección de escritura				
BIT 6	WRTB:		Bit de protección de la escritura en el bloque de arranque 1 = el bloque de arranque (000000-00007FFh) no tiene protección de escritura 0 = el bloque de arranque (000000-00007FFh) tiene protección de escritura				
BIT 5	WRTC:		Bit de protección de la escritura en el registro de configuración ⁽¹⁾ 1 = registros (300000-3000FFh) no están protegidos 0 = registros (300000-3000FFh) protegidos				

Nota 1: Este bit es inalterable en modo normal de ejecución; se puede escribir solamente en modo programación.

REGISTRO 25-11: CONFIG7L: REGISTRO 7 DE CONFIGURACIÓN BAJO (DIRECCIÓN DEL BYTE 3000Ch)

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3 ⁽¹⁾	BIT 2	BIT 1	BIT 0
-	-	-	-	EBTR3 ⁽¹⁾	EBTR2	EBTR1	EBTR0
U-0	U-0	U-0	U-0	R/C-1	R/C-1	R/C-1	R/C-1
BIT 3	EBTR3:	Bit de protección de lectura de tabla ⁽¹⁾					

		1 = el bloque 3 (006000-007FFFh) no protege contra la lectura en otros bloques 0 = el bloque 3 (006000-007FFFh) protege contra la lectura en otros bloques
BIT 2	EBTR2:	Bit de protección de lectura de tabla 1 = el bloque 2 (004000-005FFFh) no protege contra la lectura en otros bloques 0 = el bloque 2 (004000-005FFFh) protege contra la lectura en otros bloques
BIT 1	EBTR1:	Bit de protección de lectura de tabla 1 = el bloque 1 (002000-003FFFh) no protege contra la lectura en otros bloques 0 = el bloque 1 (002000-003FFFh) protege contra la lectura en otros bloques
BIT 0	EBTR0:	Bit de protección de lectura de tabla 1 = el bloque 0 (000800-001FFFh) no protege contra la lectura en otros bloques 0 = el bloque 0 (000800-001FFFh) protege contra la lectura en otros bloques

Nota 1: No implementado en los dispositivos PIC18FX455; mantener este bit activo.

REGISTRO 25-12: CONFIG7H: REGISTRO 7 DE CONFIGURACIÓN ALTO (DIRECCIÓN DEL BYTE 30000Dh)

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
-	EBTRB	-	-	-	-	-	-
U-0	R/C-1	U-0	U-0	U-0	U-0	U-0	U-0
BIT 6	EBTRB:	Bit de protección de lectura de tabla en el bloque de arranque					
		1 = el bloque de arranque (000000-0007FFh) no protege contra la lectura en otros bloques					
		0 = el bloque del cargador (000000-0007FFh) protege contra la lectura en otros bloques					

**REGISTRO 25-13: DEVID1: REGISTRO 1 DE IDENTIFICACIÓN DEL
DISPOSITIVO PARA LOS DISPOSITIVOS PIC18F2455/2550/4455/4550**

BIT 7 BIT 6 BIT 5 BIT 4 BIT 3 BIT 2 BIT 1 BIT 0
 DEV2 DEV1 DEV0 REV4 REV3 REV2 REV1 REV0
 R R R R R R R R
 BIT 7-5 DEV2:DEV0: Bits de identificación del dispositivo
 011 = PIC18F2455

010 = PIC18F2550

001 = PIC18F4455

000 = PIC18F4550

BIT 4-0 REV3:REV0: Bits de identificación de la revisión
Estos bits se utilizan para indicar la revisión del dispositivo.

REGISTRO 25-14: DEVID2: REGISTRO 2 DE IDENTIFICACIÓN DEL DISPOSITIVO PARA LOS DISPOSITIVOS PIC18F2455/2550/4455/4550

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
DEV10 ⁽¹⁾	DEV9 ⁽¹⁾	DEV8 ⁽¹⁾	DEV7 ⁽¹⁾	DEV6 ⁽¹⁾	DEV5 ⁽¹⁾	DEV4 ⁽¹⁾	DEV3 ⁽¹⁾

R	R	R	R	R	R	R	R
---	---	---	---	---	---	---	---

BIT 7-0 DEV10:DEV3: Bits de identificación del dispositivo⁽¹⁾
Estos bits se utilizan con los bits DEV2:DEV0 del DEVID1 para identificar el número de parte.
0001 0010 = dispositivos PIC18F2455/2550/4455/4550

Nota 1: Estos valores de DEV10:DEV3 se puede compartir con otros dispositivos. El dispositivo específico es identificado siempre usando la secuencia de bits DEV10:DEV0.

25.2 TEMPORIZADOR DEL PERRO GUARDIÁN (WDT)

En los dispositivos PIC18F2455/2550/4455/4550, el WDT se controla con la fuente INTRC. Cuando está permitido el WDT, la fuente del reloj también lo está. El período nominal del WDT es de 4ms y tiene la misma estabilidad que Oscilador INTRC.

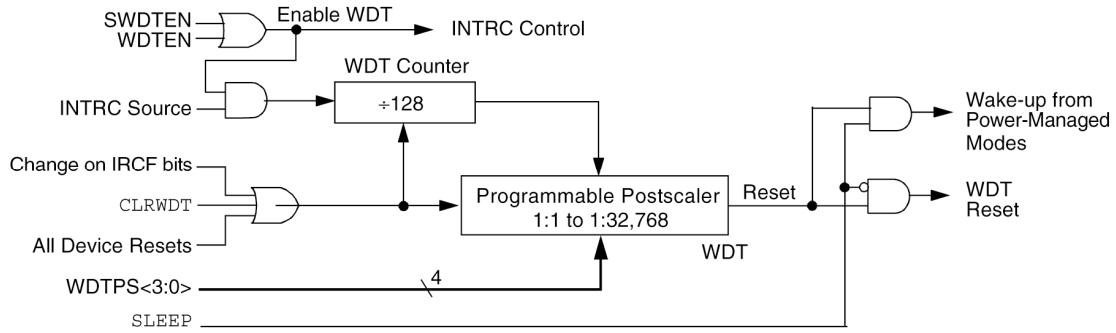
El período de 4ms del WDT se multiplica por un postscaler de 16bit. Cualquier salida del postscaler de WDT se selecciona con un multiplexor, controlado con los bits del Registro 2H. La gama de periodos disponible es a partir de 4ms a 131,072 segundos (2,18 minutos). El WDT y el postscaler se borra cuando ocurre uno de estos acontecimientos: se ejecuta una instrucción SLEEP o CLRWDT, se cambian los bits IRCF (OSCCON<6:4>) o falla el reloj.

Nota 1: Las instrucciones CLRWDT y SLEEP borran las cuentas del WDT y del postscaler.

2: Cambiar el ajuste de los bits IRCF (OSCCON<6:4>) borra las cuentas del WDT y del postscaler.

3: Cuando se ejecuta una instrucción de CLRWDT, borra la cuenta del postscaler.

FIGURA 25-1: DIAGRAMA DE BLOQUES DEL WDT



25.2.1 REGISTRO DE CONTROL

El registro WDTCON es un registro legible y escribible que contiene un bit de control que permite que el software elimine el bit de permiso del WDT, pero solamente si el bit de configuración está inhabilitando el WDT.

REGISTRO 25-15: WDTCON: REGISTRO DE CONTROL DEL WDT

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
-	-	-	-	-	-	-	SWDTEN ⁽¹⁾
U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
BIT 0	SWDTEN:	Software del control del bit de permiso del WDT ⁽¹⁾					
		1 = temporizador del perro guardián está encendido					
		0 = temporizador del perro guardián está apagado					

Nota 1: Este bit no tiene ningún efecto si se permite el bit de configuración, WDTEN.

TABLA 25-2: SUMARIO DE REGISTROS DEL WDT

Nombre	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
RCON	IPEN	SBOREN ⁽¹⁾	—	RI*	TO*	PD*	POR*	BOR*
WDTCON	—	—	—	—	—	—	—	SWDTEN

Leyenda: - = no implementado, se lee '0'; las celdas sombreadas no las utiliza el WDT.

Nota 1: El bit SBOREN sólo está disponible si BOREN<1:0>=01; en otro caso se lee '0'.

25.3 ARRANQUE A DOS VELOCIDADES

La característica de arranque de dos velocidades ayuda a reducir al mínimo período del estado latente del inicio del oscilador para la ejecución de código, permitiendo que el microcontrolador utilice el oscilador INTOSC como fuente de reloj hasta que la fuente de reloj primaria está disponible. Se permite activando el bit IESO.

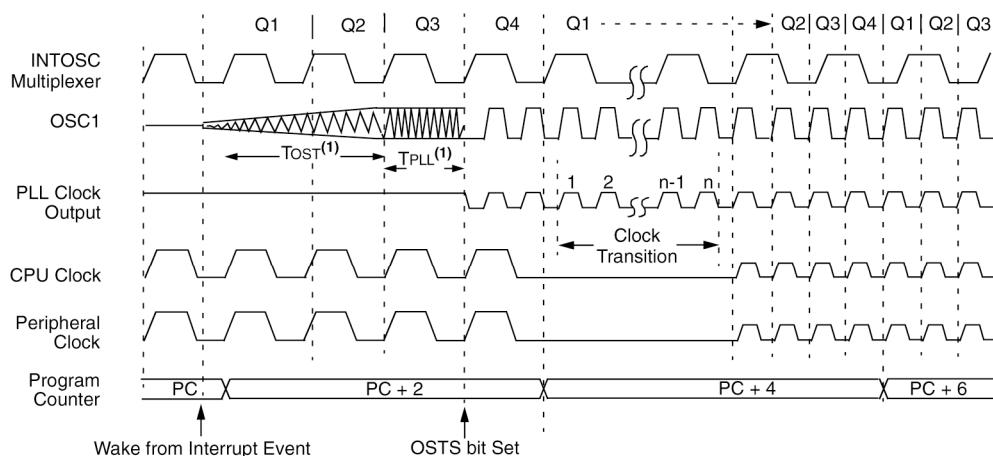
El arranque de dos velocidades sólo se puede permitir si el modo del oscilador primario es XT, HS, XTPLL o HSPLL (modos basados en XTAL). Las demás fuentes no requieren un retraso de inicio OST; para éstos, hay que inhabilitar el arranque de dos velocidades.

Cuando está permitido, los resets y reinicios del modo sleep hacen que el dispositivo se configure para que funcione el bloque interno como fuente de reloj, siguiente, el desbordamiento del temporizador de inicio después de activar un POR. Esto permite la ejecución de código casi de inmediato mientras que el oscilador primario comienza y OST está funcionando. Cuando se sobrepasan los tiempos OST, el dispositivo automáticamente cambia al modo PRI RUN.

Como el registro OSCCON se borra en los resets, la fuente de reloj INTOSC (o postscaler) no se disponen inicialmente después de un reset; el reloj INTRC se utiliza directamente en su baja frecuencia. Para utilizar un reloj con velocidad más alta en el inicio, se utilizan las fuentes de reloj INTOSC o postscaler fijando los bits, IRCF2:IRCF0, inmediatamente después del reset. Para los reinicios del sleep, las fuentes de reloj INTOSC o el postscaler se pueden seleccionar fijando IRCF2:IRCF0 antes de entrar en el modo sleep.

En el resto de los modos de ahorro de energía, el arranque a dos velocidades no se utiliza. El dispositivo se controlará con la fuente de reloj seleccionada hasta que la fuente primaria de reloj esté disponible. El ajuste del bit IESO se ignora.

FIGURA 25-2: TRANSICIONES DE TIEMPO PARA EL INICIO DE DOS VELOCIDADES (INTOSC A HSPLL)



Note 1: TOST = 1024 Tosc; TPLL = 2 ms (approx).

25.3.1 CONSIDERACIONES ESPECIALES PARA USAR ARRANQUE A DOS VELOCIDADES

Cuando se usa el oscilador INTRC en el arranque de a velocidades, el dispositivo obedece las secuencias normales del comando para entrar en los modos de ahorro de energía, incluyendo la instrucción SLEEP. En la práctica, esto significa que el usuario cambia los bits SCS1:SCS0 con la instrucción SLEEP antes de terminar los tiempos OST. Esto permite un uso brevemente del reinicio, realizar rutinas de “economía doméstica” devuelve del modo sleep antes de que el dispositivo empiece a operar con el oscilador primario.

El código del usuario puede comprobar si la fuente primaria de reloj está gobernando el dispositivo comprobando el estado del bit OSTS (OSCCON<3>). Si se activa el bit, el oscilador primario está controlando el reloj. Si no, el bloque interno del oscilador gobierna durante el reinicio del modo sleep o de un reset.

25.4 MONITOR DE RELOJ A PRUEBA DE FALLOS

El Monitor de reloj a prueba de fallos (FSCM) permite al microcontrolador continuar la operación en caso de fallo del oscilador externo automáticamente cambiando el reloj del dispositivo al bloque interno del oscilador. El la función FSCM se permite activando la configuración del bit FCMEN.

Cuando se permite el FSCM, el oscilador INTRC funciona todas las veces que supervise los relojes de los periféricos y proporcione un reloj de reserva en caso de un fallo del reloj. La supervisión del reloj se logra creando una muestra de la señal, que es la salida de INTRC dividido por 64. Esto da un plazo de tiempo amplio entre la muestra de FSCM y que ocurra un flanco del reloj del periférico. El reloj del dispositivo periférico y la muestra del reloj se presentan como entradas del latch del monitor de reloj (CM). El CM se activa en el flanco de bajada de la fuente de reloj del dispositivo, pero borrado en el flanco de subida del reloj de muestreo.

El fallo del reloj se prueba en el flanco de bajada del reloj de muestreo. Si ocurre un flanco de bajada del reloj de muestreo mientras que el CM todavía esté activo, se ha detectado un fallo del reloj. Esto provoca lo siguiente:

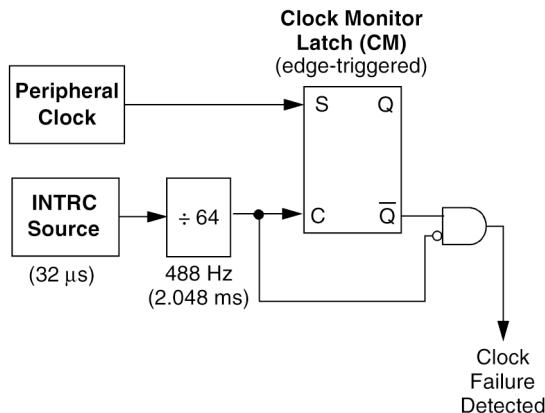
- el FSCM genera una interrupción de fallo del oscilador activando el bit, OSCFIF (PIR2<7>);
- la fuente del reloj del dispositivo se cambia al bloque interno del oscilador (OSCCON no se actualiza para mostrar la fuente actual del reloj - esta es la condición a prueba de fallos);
- se resetea el WDT.

Durante el intercambio, la frecuencia del postscaler del bloque interno del oscilador puede que no sea suficientemente estable para aplicaciones sensibles que miden el tiempo. En estos casos, se puede seleccionar otra configuración del reloj y entrar en un modo de ahorro de energía diferente. Esto se puede hacer para realizar una recuperación parcial o ejecutar una parada controlada.

Para utilizar una velocidad de reloj más alta en el reinicio, se pueden seleccionar el INTOSC o las fuentes del reloj del postscaler los bits IRCF2:IRCF0 inmediatamente después del reset. Para los reinicios del sleep, INTOSC o las fuentes del reloj del postscaler se pueden seleccionar fijando IRCF2:IRCF0 antes de entrar en el modo sleep.

El FSCM detectará los fallos de las fuentes de reloj primaria o secundaria solamente. Si el bloque del oscilador interno falla, no se detectaría, y no se puede realizar ninguna acción.

FIGURA 25-3: DIAGRAMA DE BLOQUES DEL FSCM



25.4.1 FSCM Y EL TEMPORIZADOR PERRO GUARDIÁN

Los FSCM y los WDT se controlan con el Oscilador INTRC. Puesto que el WDT funciona con el divisor separado y el contador, inhabilitar el WDT no tiene ningún efecto sobre el oscilador INTRC cuando se activa el FSCM.

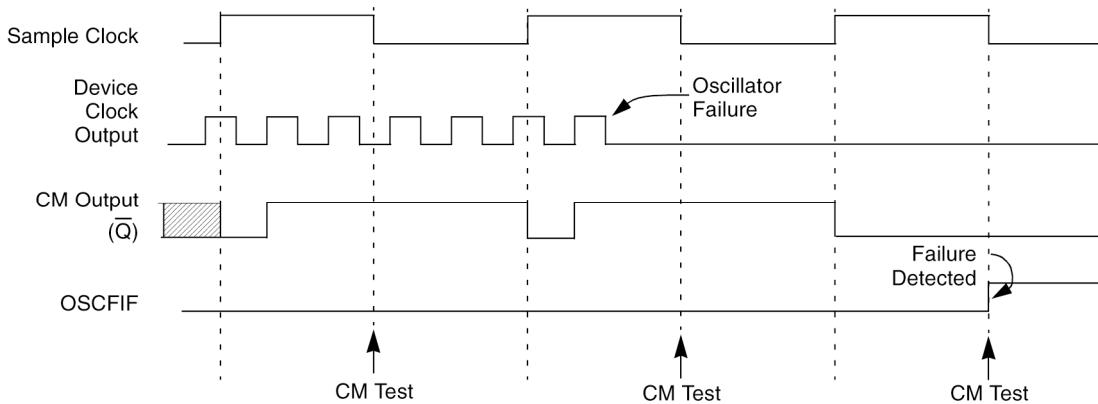
Según lo observado ya, la fuente de reloj se cambia al Reloj INTOSC cuando se detecta un fallo. Dependiendo de la frecuencia seleccionada por los bits IRCF2:IRCF0, puede significar un cambio substancial en la velocidad de ejecución del código. Si se permite el WDT con un valor pequeño de prescaler, disminuye la velocidad del reloj y permite que ocurra un desbordamiento del WDT y su correspondiente reset del dispositivo. Por esta razón, los acontecimientos del monitor de reloj a prueba de fallos también resetean el WDT y el postscaler, permitiendo comenzar la sincronización del cambio de la velocidad de ejecución y disminuyendo la probabilidad de un desbordamiento erróneo.

25.4.2 SALIR DE LA OPERACIÓN A PRUEBA DE FALLOS

La condición a prueba de fallos termina por cualquier reset o entrando en un modo de ahorro de energía. En el reset, el regulador comienza con la fuente primaria de reloj especificada en el registro de configuración 1H (con cualquier retraso de inicio que requiera el modo del oscilador, por ejemplo temporizador OST o PLL). El multiplexor INTOSC proporciona el reloj del dispositivo hasta que la fuente primaria de reloj esté lista (similar a un arranque de a velocidades). La fuente de reloj entonces se cambia al reloj primario (indicado por el bit OSTS en el registro OSCCON que se activa). El monitor de reloj a prueba de fallos entonces reanuda la supervisión del reloj del periférico.

La fuente primaria de reloj puede que no llegue a estar lista en el inicio. En este caso, la operación se controla con el multiplexor INTOSC. El registro OSCCON seguirá en reset hasta que entre en un modo de ahorro de energía.

FIGURA 25-4: DIAGRAMA DE TIEMPOS DEL FSCM



25.4.3 INTERRUPCIONES DE FSCM DENTRO DE LOS MODOS DE AHORRO DE ENERGÍA

Entrando en un modo de ahorro de energía, el reloj del multiplexor selecciona la fuente de reloj elegida en el registro OSCCON. La supervisión del reloj a prueba de fallos de los modos de ahorro de energía de la fuente de reloj se reanuda en un modo de ahorro de energía.

Si ocurre un fallo del oscilador durante el modo de ahorro de energía, los acontecimientos subsecuentes dependen de si la interrupción de fallo del oscilador está permitida o no. Si está permitida (OSCFIF=1), la ejecución del código la controlará el multiplexor INTOSC. No habrá un cambio automático al reloj que ha fallado.

Si la interrupción no está activa, las siguientes interrupciones mientras que esté en modo reposo harán que la CPU comience a ejecutar instrucciones cuando la fuente INTOSC siga controlando el dispositivo.

25.4.4 POR O REINICIO DEL SLEEP

El FSCM se ha diseñado para detectar fallos en el oscilador en cualquier punto después de salir el dispositivo de un reset por aumento de energía (POR) o del modo de baja potencia sleep. Cuando el reloj primario del dispositivo es EC o INTRC, la supervisión puede comenzar inmediatamente después de estos acontecimientos.

En los modos del oscilador que implican un cristal o un resonador (HS, HSPLL o XT), la situación es algo diferente. Puesto que el oscilador puede necesitar un retraso de inicio considerablemente más de largo que el tiempo de reloj de la muestra de FCSM, se puede detectar un fallo falso del reloj. Para prevenir esto, el bloque del oscilador interno se configura automáticamente como el reloj y las funciones del dispositivo hasta que el reloj primario es estable (los contadores de tiempo OST y PLL se han desbordado). Esto es igual al modo de inicio de dos velocidades. Una vez que el reloj primario es estable, el INTRC vuelve a ser la fuente de FSCM.

Nota: La misma lógica que previene el oscilador de falsas interrupciones por fallo en un POR o al despertar del sleep también prevendrá la detección del fallo del oscilador en todos los acontecimientos que siguen a estos. Esto se puede evitar supervisando el bit OSTS y usar la rutina de sincronización para determinar si el oscilador está tardando demasiado en comenzar. Sin embargo, la interrupción de no fallo del oscilador se señalará.

También es posible seleccionar otra configuración del reloj e incorporar un modo de ahorro de energía suplemento mientras que el reloj primario sea estable. Cuando el nuevo modo de ahorro de energía se selecciona, el reloj primario se desactiva.

25.5 VERIFICACIÓN DEL PROGRAMA Y CÓDIGO DE PROTECCIÓN

La estructura total del código de protección de los dispositivos PIC18F difiere significativamente de otros dispositivos PICmicro®.

La memoria de programa del usuario se divide en cinco bloques. Uno de estos es el bloque de arranque de 2kb. El resto de la memoria se divide en cuatro bloques en límites binarios.

Cada uno de estos cinco bloques tiene tres códigos de protección asociados a ellos. Son:

- Bit de protección del código (CPn)
- Bit de protección de la escritura (WRTn)
- Bit de lectura del bloque de tabla externa (EBTRn)

FIGURA 25-5: CÓDIGO DE PROTECCIÓN DE LA MEMORIA DE PROGRAMA PARA LOS PIC18F2455/2550/4455/4550

MEMORY SIZE/DEVICE		Address Range	Block Code Protection Controlled By:
24 Kbytes (PIC18F2455/2555)	32 Kbytes (PIC18F2550/4550)		
Boot Block	Boot Block	000000h 0007FFh	CPB, WRTB, EBTRB
Block 0	Block 0	000800h 001FFFh	CP0, WRT0, EBTR0
Block 1	Block 1	002000h 003FFFh	CP1, WRT1, EBTR1
Block 2	Block 2	004000h 005FFFh	CP2, WRT2, EBTR2
Unimplemented Read '0's	Block 3	006000h 007FFFh	CP3, WRT3, EBTR3
Unimplemented Read '0's	Unimplemented Read '0's	008000h 1FFFFFFh	(Unimplemented Memory Space)

TABLA 25-3: SUMARIO DE LOS REGISTROS DEL CÓDIGO DE PROTECCIÓN

Nombre	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
300008h CONFIG5L	—	—	—	—	CP3 ⁽¹⁾	CP2	CP1	CP0
300009h CONFIG5H	CPD	CPB	—	—	—	—	—	—
3000Ah CONFIG6L	—	—	—	—	WRT3 ⁽¹⁾	WRT2	WRT1	WRT0
3000Bh CONFIG6H	WRTD	WRTB	WRTC	—	—	—	—	—
3000Ch CONFIG7L	—	—	—	—	EBTR3 ⁽¹⁾	EBTR2	EBTR1	EBTR0
3000Dh CONFIG7H	—	EBTRB	—	—	—	—	—	—

Leyenda: las celdas sombreadas no están implementadas.

Nota 1: No implementado en los dispositivos PIC18Fx455; mantener este bit activo.

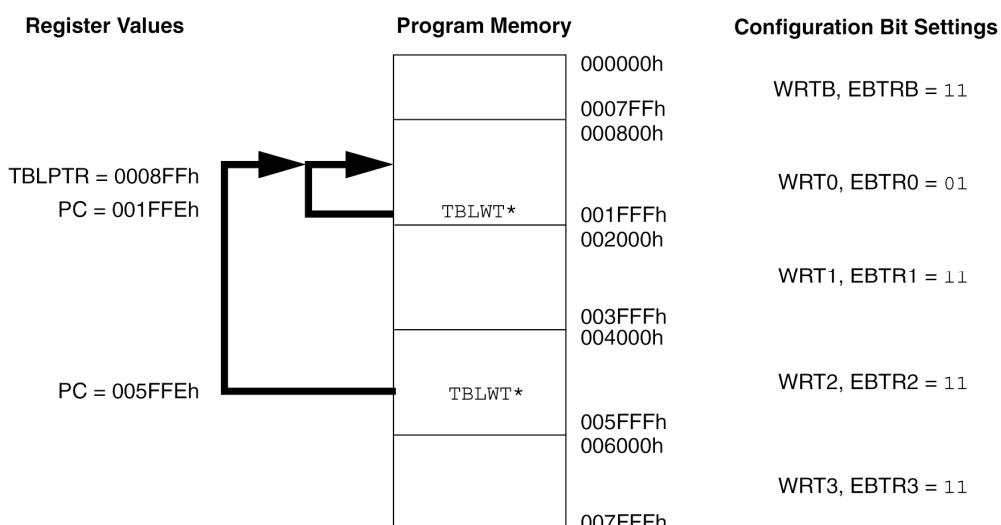
25.5.1 MEMORIA DE PROGRAMA CÓDIGO DE PROTECCIÓN

La memoria de programa se puede leer o escribir en cualquier localización usando las instrucciones de lectura y escritura en tabla. La identificación del dispositivo se puede leer con la lectura de tabla. Los registros de configuración se pueden leer y escribir con la lectura y escritura de tabla.

En el modo normal de ejecución, los bits CPn no tienen ningún efecto directo. Los bits CPn inhiben la lectura y escritura externa. El bloque de memoria del usuario se puede proteger contra la escritura de tabla si el bit de configuración WRTn está a '0'. Los bits EBTRn controlan la lectura de tabla. Para un bloque de memoria del usuario con el bit EBTRn a '0', se permite una instrucción de lectura de tabla ejecutada dentro de ese bloque. Una instrucción de lectura de tabla que se ejecuta en una localización exterior de ese bloque no se permite y da lugar a '0's.

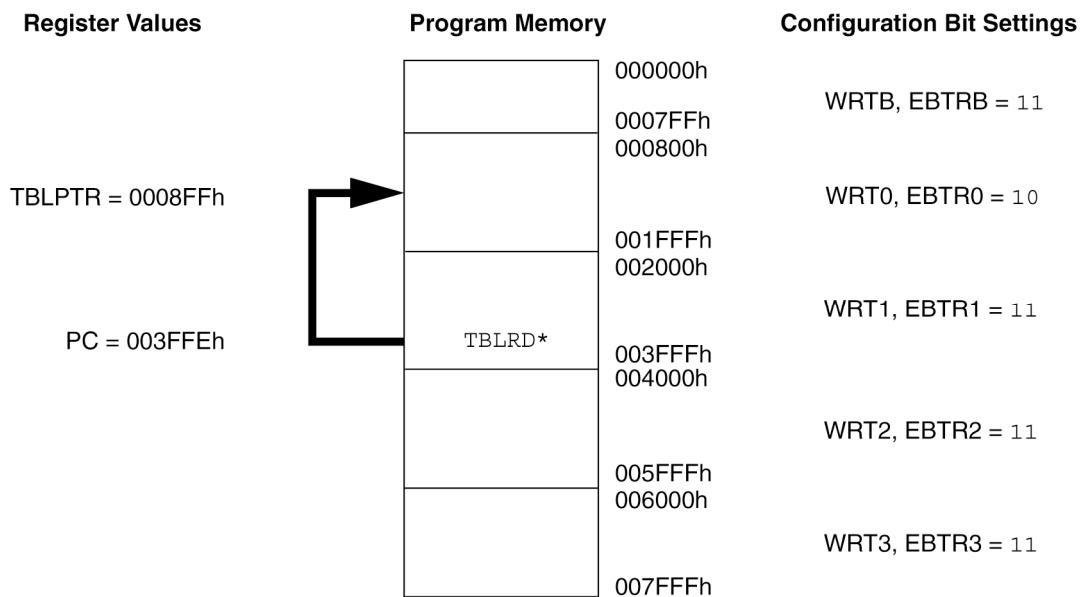
Nota: Los bits de protección del código se pueden escribir solamente de un '0' a un '1'. No es posible a escribir un '1' en un bit a '0'. Los bits del código de protección se activan solamente con un borrado completo del chip o al borrar el bloque función. El borrado completo del chip y el borrado del bloque función sólo se pueden realizar por medio de ICSP o de un programador externo.

FIGURA 25-6: ESCRITURA EN TABLA RECHAZADA (WRTn)



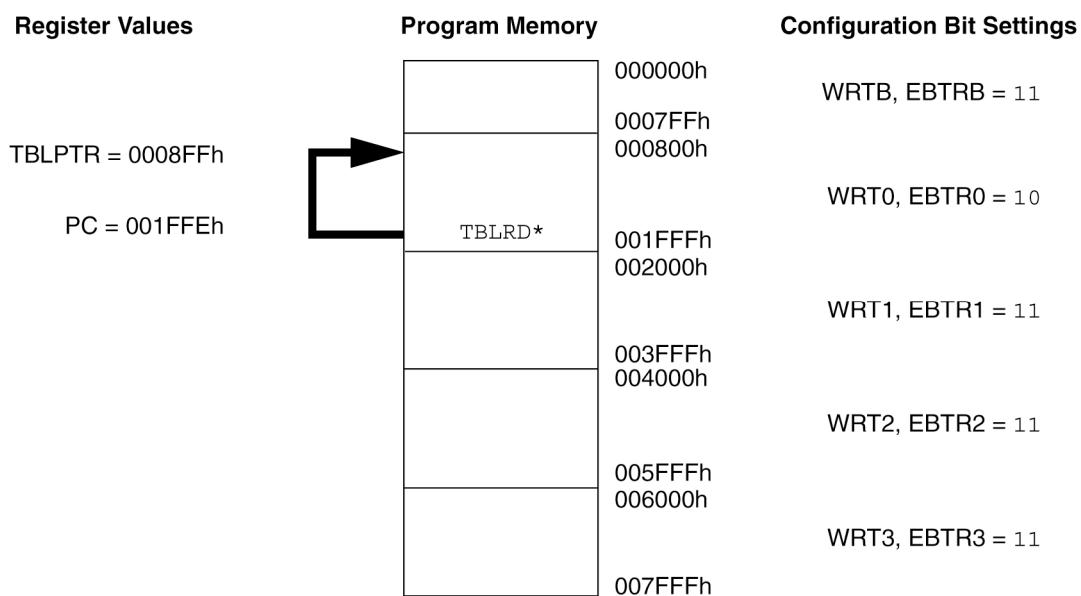
Resultado: Todas las escrituras en tabla desactivadas en el Blockn cuando WRTn=0.

FIGURA 25-7: LECTURA EXTERNA DE TABLA RECHAZADA (EBTRn)



Resultados: Todas las lecturas de los bloques externos al Blockn están desactivadas cuando EBTRn=0. El registro TABLAT devolverá ‘0’.

FIGURA 25-8: LECTURA EXTERNA DE TABLA PERMITIDA (EBTRn)



Resultados: Lecturas en tabla permitidas al Blockn, incluso cuando EBTRBn=0. El registro TABLAT devolverá el valor del dato en la localización TBLPTR.

25.5.2 DATOS EEPROM CÓDIGO DE PROTECCIÓN

Los datos EEPROM se protegen contra lecturas y escrituras externas con dos bits: CPD y WRTD. CPD inhibe lecturas y escrituras externas en los datos EEPROM. Y WRTD inhibe escrituras internas y externas en los datos EEPROM. La CPU puede continuar leyendo y escribiendo datos en la EEPROM sin importar el ajuste de los bits de protección.

25.5.3 CONFIGURACIÓN DEL REGISTRO DE PROTECCIÓN

Los registros de configuración pueden protegerse contra la escritura. El bit WRTC controla la protección de los registros de configuración. En modo de ejecución normal, el bit WRTC es solamente legible. WRTC sólo se puede escribir vía ICSP o con un programador externo.

25.6 LOCALIZACIONES DE IDENTIFICACIÓN

En ocho posiciones de memoria (200000h-200007h) se guardan como localizaciones de identificación, en donde el usuario puede almacenar versiones u otros números de identificación del código. Éstas localizaciones son legibles y escribibles durante la ejecución normal con las instrucciones TBLRD y TBLWT o durante programación/verificación. Las localizaciones de identificación se pueden leer cuando se protege el código del dispositivo.

25.7 PROGRAMACIÓN SERIE IN-CIRCUIT

Los microcontroladores PIC18F2455/2550/4455/4550 pueden programarse en serie en el circuito de la aplicación. Esto se hace simplemente con dos líneas para el reloj y los datos y tres otras líneas para la energía, tierra y tensión de programación. Esto permite que los clientes fabriquen las placas con los dispositivos sin programar y entonces programar el microcontrolador momentos antes de enviar el producto. Esto también permite los firmwares más recientes o firmwares de encargo.

25.8 DEPURACIÓN IN-CIRCUIT

Cuando el bit de configuración DEBUG* se programa con un ‘0’, la funcionalidad de la depuración “en Circuito” se permite. Esto permite funciones simples para eliminar errores cuando se utiliza MPLAB® IDE. Cuando el microcontrolador tiene esta característica permitida, algunos recursos no están disponibles en el uso general.

Para utilizar la función In-Circuit de la depuración del microcontrolador, el diseño debe implementar las conexiones de la programación In-Circuit: MCLR*/VPP/RE3, VDD, VSS, RB7 y RB6. Esto interconectará al módulo depuración In-Circuit disponible de microchip o de una de las compañías de herramientas de desarrollo.

TABLA 25-4: RECURSOS DE DEPURACIÓN

Pines E/S:	RB6, RB7
Pila:	2 niveles
Memoria de programa:	512 bytes
Memoria de datos:	10 bytes

25.9 CARACTERÍSTICAS ESPECIALES DE ICPORT (SEÑALANDO PAQUETES SOLAMENTE)

Bajo circunstancias específicas, los pines que no se conectan (NC) de los dispositivos PIC18F4455/4550 de 44pines en encapsulado TQFP pueden proporcionar funcionalidad adicional. Éstas características se controlan con los bits de configuración del dispositivo y están disponibles solamente en este tipo de encapsulado y número de pines.

25.9.1 PUERTO DEDICADO ICD/ICSP

Los dispositivos de 44pines con encapsulado TQFP pueden utilizar los pines NC para proporcionar un puerto alterno para depuración In-Circuit (ICD) y la programación serie In-Circuit (ICSP). Estos pines se conocen como el puerto dedicado ICSP/ICD, puesto que no comparten ninguna otra función del dispositivo.

Cuando se implementa, el puerto dedicado activa tres pines NC para proporcionar un reset del dispositivo, datos y puertos del reloj alternos. Ningunos de estos puertos se solapan con los pines estándares de E/S, poniendo los pines de E/S a disposición de la aplicación del usuario.

El puerto dedicado ICSP/ICD se permite activando el bit de configuración ICPRT. El puerto funciona de la misma manera que el puerto de herencia ICSP/ICD en RB6/RB7.

Aun cuando el puerto dedicado está permitido, el ICSP y las funciones de ICD siguen funcionando en el puerto de herencia. Cuando se detecta V_{IH} en el pin MCLR*/VPP/RE3, el estado de ICRST/ICVPP se ignora.

Nota 1: El bit de configuración ICPRT puede estar programado con puerto ICSP por defecto.

2: El bit de configuración ICPRT debe mantenerse borrado en los dispositivos de 28pines y 40pines; si no, pueden ocurrir operaciones inesperadas.

TABLA 25-5: EQUIVALENCIA DE PINES ENTRE LOS PUERTOS DEDICADOS Y HERENCIA ICD/ICSP

Nombre del pin		Tipo de pin	Función del pin
Puerto herencia	Puerto dedicado		
MCLR*/VPP/RE3	NC/ICRST*/ICVPP	Energía	Reset del dispositivo y permiso de programación
RB6/KBI2/PGC	NC/ICCK/ICPGC	Entrada	Reloj serie
RB7/KBI3/PGD	NC/ICDT/ICPGD	E/S	Datos serie

25.9.2 EMULACIÓN DE 28PINES

Los dispositivos PIC18F4455/4550 de 44 pines con encapsulado TQFP tienen la capacidad de cambiar su configuración bajo control externo para depurar errores. Esto permite que el dispositivo se comporte como si fuera un PIC18F2455/2550 de 28pines.

Este modo de configuración de 28pines se controla a través de un solo pin, NC/ICPORTS. Conectar este pin a V_{SS} fuerza al dispositivo para que funcione como un dispositivo de 28pines. Las características asociadas a los dispositivos de 40/44pines se desactivan junto a sus registros y bits de control correspondientes. Esto incluye PORTD y PORTE, los SPP y la funcionalidad PWM mejorada del CCP1. Por otra parte, conectar el pin a V_{DD} fuerza al dispositivo que funcione en su configuración por defecto.

La opción de configuración sólo está disponible cuando se permite la depuración oculta y el puerto dedicado ICD/ICSP (bit DEBUG* está borrado y se activa el bit ICPRT). Cuando está inhabilitado, NC/ICPORTS es un pin NC.

25.10 SUMINISTRO ÚNICO DE LA PROGRAMACIÓN ICSP

El bit de configuración LVP permite el suministro único de la programación ICSP (conocida antes como Programación ICSP de baja tensión o LVP). Cuando el suministro único de programación se permite, el microcontrolador puede programarse sin requerir la tensión alta que se aplica al pin MCLR*/VPP/RE3, pero el pin RB5/KBI1/PGM se dedica a controlar la entrada del modo de programa y no está disponible como pin de entrada-salida.

Cuando se programa usando el suministro único de programación, V_{DD} se aplica al pin MCLR*/VPP/RE3 como en el modo de ejecución normal. Para entrar en modo de programación, V_{DD} se aplica al pin PGM.

Nota 1: La programación de alta tensión siempre está disponible, sin importar el estado del bit LVP, aplicando V_{IHH} al pin MCLR*.

2: Mientras que en la programación de baja tensión del modo ICSP, el pin RB5 no puede utilizarse más como pin de entrada-salida y deber llevarse a un punto de tensión bajo durante la operación normal.

3: Al usar la programación de baja tensión ICSP (LVP) y se permiten los pull-ups del PORTB, el bit 5 en el registro TRISB debe borrarse para inhabilitar los pull-ups en RB5 y asegurar el funcionamiento correcto del dispositivo.

4: Si el Master Clear está desactivado, verificar que lo siguiente sea correcto para asegurar la entrada apropiada en el modo ICSP:

- a) inhabilita la programación de baja tensión (CONFIG4L<2>=0);
- b) RB5/KBI1/PGM está en nivel bajo durante la entrada de ICSP.

Si el modo de programación suministro único ICSP no se utiliza, el bit LVP puede borrarse. RB5/KBI1/PGM entonces esta disponible como pin de E/S digital, RB5. El bit LVP puede activarse o borrarse solamente al usar la programación estándar de alta tensión (V_{IHH} aplicado al pin MCLR*/V_{PP}/RE3). Una vez que se haya inhabilitado LVP, solamente la programación de alta tensión estándar está disponible y debe utilizarse para programar el dispositivo.

La memoria sin código de protección puede borrarse usando el borrado de un bloque, o borrar fila por fila, entonces escrita en cualquier V_{DD} especificada. Si se quiere borrar la memoria protegida, se necesita borrar un bloque. Si se realiza el borrado de un bloque al usar la programación de baja tensión, el dispositivo necesita una V_{DD} de 4,5V a 5,5V.