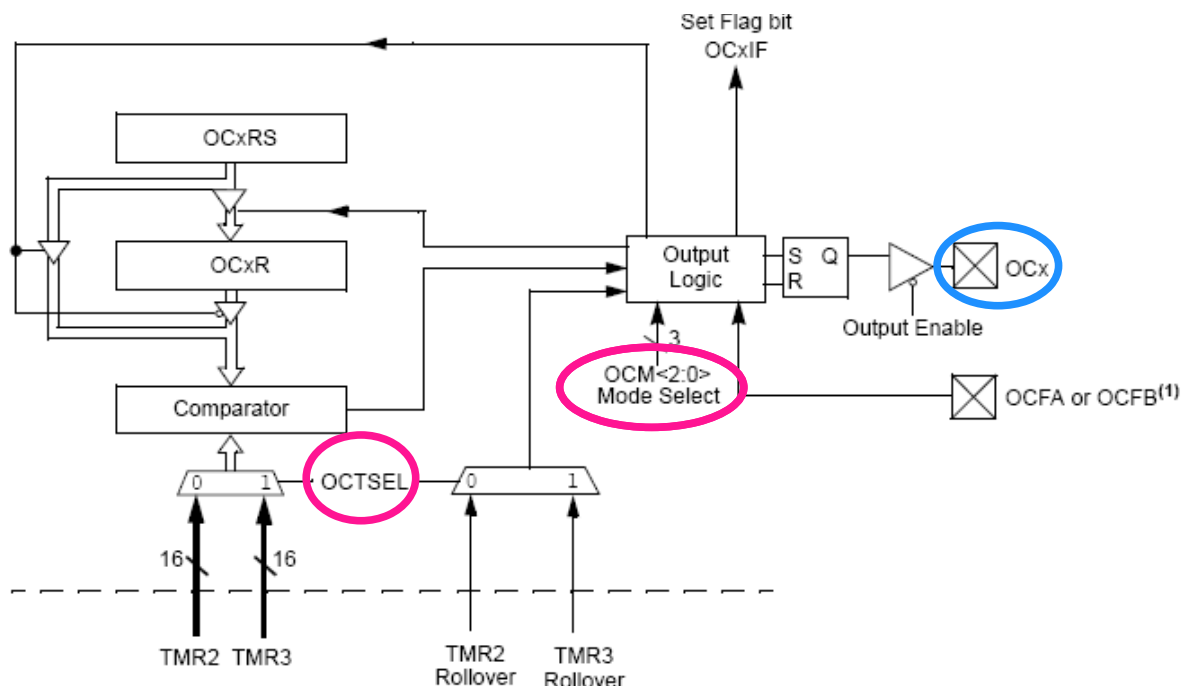


## MÓDULO OC (Output Compare) (section 13)

El módulo OC compara el valor del registro de cuenta (TMR) de un temporizador (T2 o T3, dependiendo de cómo se programe) con el valor de un registro del propio módulo OC (registro OCxR), y cuando son iguales genera un pulso o una secuencia de pulsos en el pin OCx, que es el pin de salida asociado al módulo OCx (véase figura 1). Tal y como se verá más adelante, el pulso o secuencia de pulsos depende de la programación del módulo OCx y del temporizador que se utilice con el mismo. Además, el módulo puede generar una petición de interrupción cuando ambos registros son iguales.

El esquema de la figura 1 muestra el esquema del módulo (la letra ‘x’ hace referencia al número de OC, ya que el PIC24H cuenta con 8 módulos OC: OC1, OC2,... OC8).



**Figura 1: esquema del módulo OCx**

Tal y como puede apreciarse, no se trata de un esquema complejo.

Un módulo OC cuenta con los siguientes registros:

- OCxR: registro de comparación (*Output Compare Register*)
- OCxRS: registro de comparación secundario (*Secondary Output Compare Register*)
- OCxCON: registro de control (*Output Compare Control Register*)

Los dos primeros se ven en la figura 1. El registro de control no se aprecia en el esquema, pero sí dos de los campos que guarda este registro: OCTSEL, para seleccionar la información del temporizador T2 o T3, y OCM, para determinar el modo de funcionamiento, tal y como se verá más adelante.

Cuando un módulo OC está activado, es él mismo el que controla que la dirección del pin OCx correspondiente es de salida (controla el TRIS de esa pata).

Los pines OCFA y OCFB son pines de entrada para protección de fallos y se asocian respectivamente a los módulos OC1-OC4 y OC5-OC8.

En el pin de salida OCx OC se pueden conseguir 7 modos de funcionamiento diferentes que se enumeran a continuación y que están gráficamente ilustrados en la figura 2. Que funcione en un modo u otro depende del valor del campo OCM (3 bits):

*Active Low One-Shot mode:* genera un solo pulso, activado en baja (es decir, se entiende la salida .L o en lógica negativa) cuya duración se controla mediante el registro OcxC. Puede volver a activarse el pulso escribiendo en el registro de control.

*Active High One-Shot mode:* Igual que el anterior pero generando el pulso el alta (.H o lógica positiva).

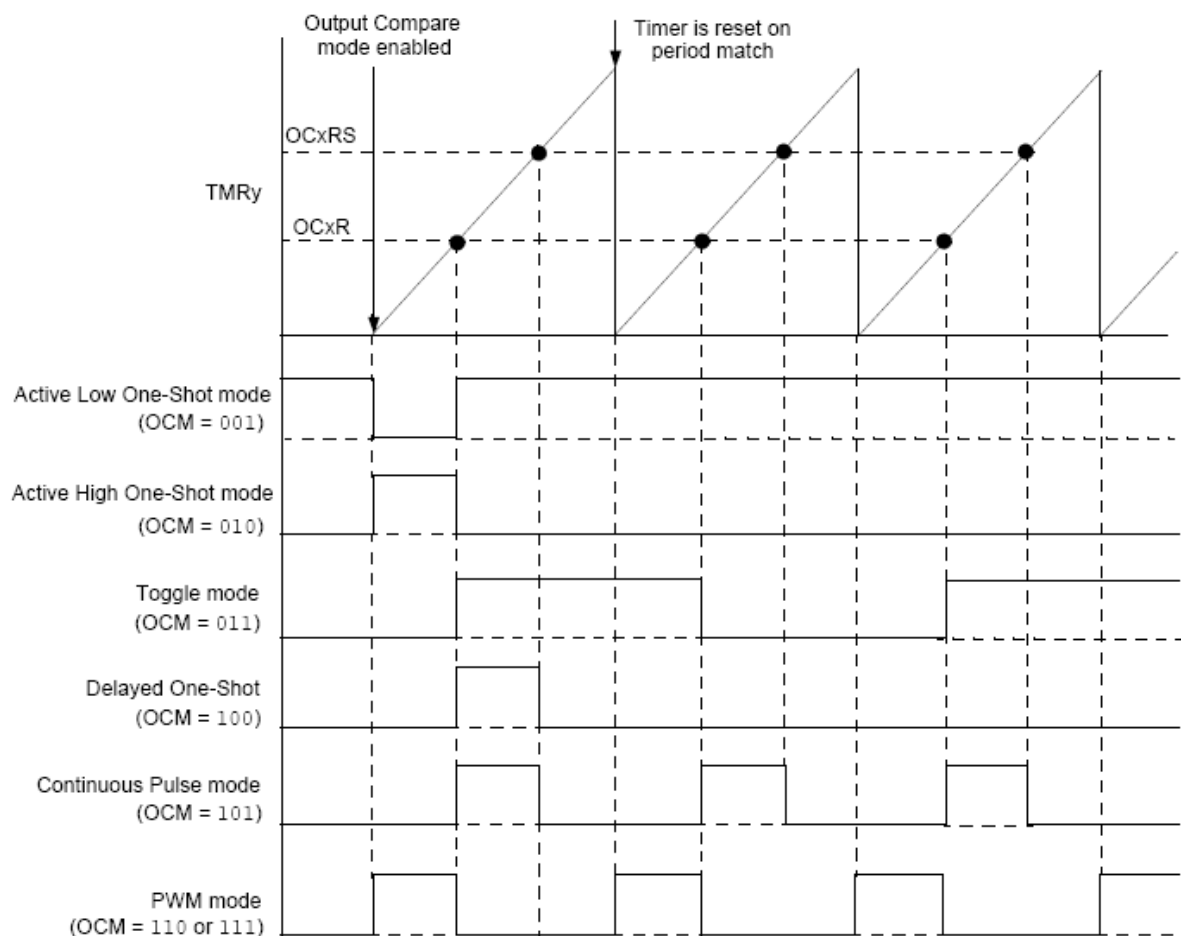
*Toggle mode:* invierte el valor del pin de salida cuando ocurre la igualdad entre TMRy y OCxC. Con este modo es fácil generar señales cuadradas periódicas.

*Delayed One-Shot mode:* genera un único pulso pero retardado. Se puede volver a activar escribiendo en el registro de control.

*Continuous Pulse mode:* genera pulsos continuos.

PWM mode (*Pulse Width Modulation*), sin protección ante fallos.

PWM mode (*Pulse Width Modulation*), con protección ante fallos.



**Figura 2: Modos de funcionamiento del módulo OCx**

## OCxCON: OUTPUT COMPARE x CONTROL REGISTER (x = 1 – 8)

U-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0	R-0,HC	R/W-0	R/W-0	R/W-0	R/W-0
-	-	OCSIDL	-	-	-	-	-	-	-	-	OCFLT	OCTSEL	OCM<2:0>		
Bit15								Bit8	Bit7						bit0

**OCSIDL:** Controla el estado del módulo en modo *Idle*

1 = módulo parado en modo *Idle* / 0 = módulo operativo en modo *Idle*

**OCFLT:** Bit de estado de condición de fallo de PWM. Solo es válido con OCM<2:0> = 111

1 = Se ha producido fallo en PWM

0 = No hay fallo en PWM

**OCTSEL:** Selecciona el temporizador

**1 = Temporizador 3**

**0 = Temporizador 2**

**OCM<2:0>:** Selecciona el modo de funcionamiento (véase figura 2):

000 = Módulo Off, no activado

001 = Inicializa la pata OCx a baja y cuando se detecta la igualdad con OCxR se pone a alta. Interrupción en la subida.

010 = Inicializa la pata OCx a alta y cuando se detecta la igualdad con OCxR se pone a baja. Interrupción en la bajada.

011 = Cada vez que se detecta la igualdad con OCxR se invierte la tensión en OCx, y se genera interrupción tanto en el flanco de subida como en el de bajada.

100 = Inicializa la pata OCx en baja y al detectarse la igualdad con OCxR la pasa a alta; permanece así hasta que se detecte la igualdad con OCxRS (registro secundario), momento en el que volverá a pasar a baja. Interrupción en la bajada.

101 = Igual que el modo anterior pero generando pulsos consecutivamente en todas las igualdades. Interrupción en las bajadas.

**110 = General señal PWM en OCx, con pata de fallo deshabilitada. No genera interrupción. Este es el modo que vamos a utilizar.**

111 = PWM en OCx con pata de fallo habilitada. Interrumpe si detecta flanco de bajada en OCFA u OCFB.

El modo de funcionamiento que nos interesa para las prácticas es el modo PWM, ya que con un pulso de ese tipo se puede controlar un servomotor. En ese modo se genera una señal PWM cuyo ancho de pulso (*duty cycle*) se puede variar mediante el registro OCxRS, que es el que establece el ancho del pulso. Tal y como se ve en la figura 3, el valor de OCxRS se carga automáticamente en el OCxR al comienzo del periodo del temporizador (*Tx Rollover*).

Para conseguir la señal PWM deseada, los pasos son los siguientes:

1. Ajustar el ancho del pulso PWM (*duty cycle*) escribiendo en OCxRS el valor adecuado (el ancho del primer pulso será el que determine OCxR).
2. Configurar el modo de funcionamiento del módulo OC mediante el campo OCM (campo de 3 bits en el registro de control, OCxCON <2:0>).
3. Establecer el periodo de la señal PWM escribiendo en el registro PR del temporizador correspondiente el valor adecuado.
4. Establecer el valor del prescaler adecuado, inicializar TMRY y activar el temporizador mediante el bit TON (TyCON <15>= 1).

Notas:

- Los registros OCxRS y OCxR se deben inicializar antes de habilitar el módulo.
- El OCxR se convierte en un registro de sólo lectura cuando funciona en el modo PWM.
- El valor que demos a OCxR se convertirá en el ciclo de PWM para el primer periodo.
- Podemos escribir (modificar) OCxRS en cualquier momento, pero el contenido del registro OCxRS, no se transfiere a OCxR hasta que termine el periodo en curso (*Rollover*).



## Ejemplo de código para activar el módulo OC1 en modo PWM (utilizando el temporizador T2)

En este ejemplo el registro se carga con el valor 100 (inicialmente, OC1R también). Por lo tanto, con la frecuencia que estamos utilizando, el ancho del pulso será  $100 \cdot T_{cy}$ . El temporizador T2 establece un periodo de  $500 \cdot T_{cy}$ .

```
// Inicialización módulo Output Compare en modo PWM
OC1CON = 0;           // Deshabilita módulo Output Compare
//OC1CONbits.OCTSEL = 0; // 0 -> Selecciona T2; 1 -> T3
OC1R = 100;           // duty cycle para el primer pulso PWM
OC1RS = 100;           // duty cycle para el siguiente pulso PWM
OC1CONbits.OCM = 0b110; // Selecciona el modo PWM sin protección

// Inicializa T2
T2CONbits.TON = 0;     // Deshabilita T2
T2CONbits.TCS = 0;     // Selecciona Clock interno (Fosc / 2)
T2CONbits.TGATE = 0;   // Deshabilita modo Gate
T2CONbits.TCKPS = 0b00; // Selecciona Prescaler 1:1
TMR2 = 0;              // Inicializa contador T2
PR2 = 500-1;           // Carga valor del periodo
IFS0bits.T2IF = 0;     // Borra flag interrupción
IEC0bits.T2IE = 0;     // Deshabilita Interrupción T2
// Con PWM no es necesario utilizar interrupciones
// Se puede modificar OCxRS en cualquier momento.
T2CONbits.TON = 1;     // Activa Timer
```