

# LOS PIC16F88X :Los Módulos CCP

IES Juan de la Cierva

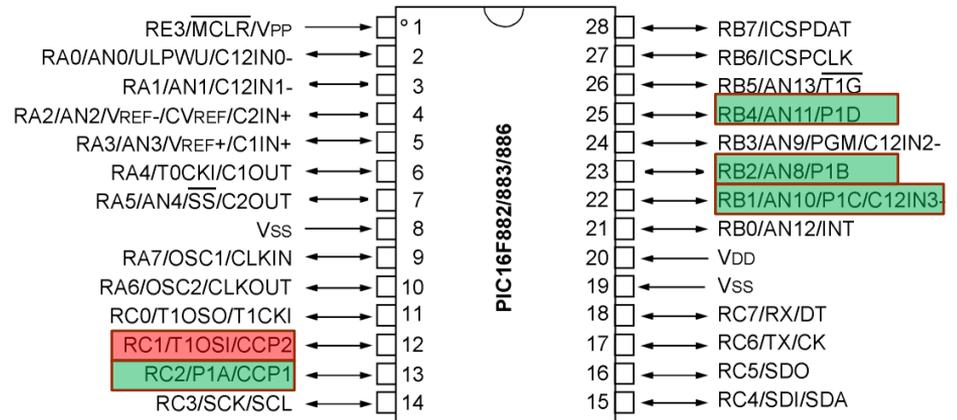
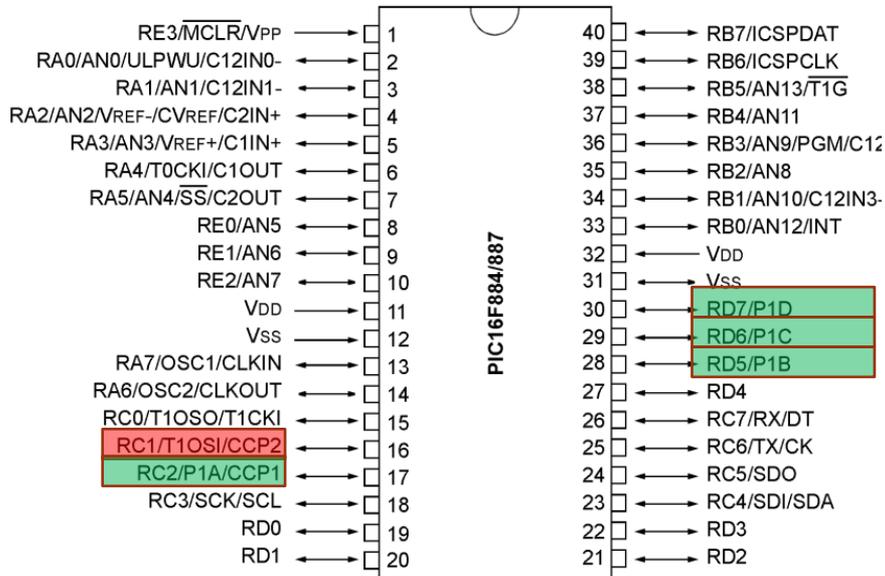


Aprendizaje de la Electrónica a través de la Robótica

# Los Módulos CCP

- Los PIC16F8xX integran 2 módulos CCP prácticamente iguales e independientes entre sí CCP1 y CCP2 que pueden trabajar en tres modos diferente.
- **Modo Captura:** Consiste en capturar el valor del TMR1 cuando se detecta un flanco ascendente o descendente de una señal.
- **Modo Comparación:** Cuando el Timer1 alcanza un valor prefijado se activa o no (“1” o “0”) una señal digital de salida. También se puede borrar el Timer1 o iniciar una conversión A/D.
- **Modo PWM (Pulse-Width Modulation):** Se genera una señal digital de salida, cuya anchura de impulso se puede variar o modular. Se entiende por anchura o “*duty cycle*”, el tiempo en que la señal se mantiene a “1”. También se puede modificar la anchura del impulso.

# Patillaje de los PIC16F88X



# Módulo ECCP1

- Tiene una notable mejora frente a los dispositivos anteriores a los PIC16F88X.
- Las patillas asociadas al ECCP1 cambian para los dispositivos de 28 y de 40 pines.
  - Para los de 28 pines: RC2/P1A/CCP1, RB2/P1B, RB1/P1C y RB4/P1D
  - Para los de 40 pines: RC2/P1A/CCP1, RD5/P1B, RD6/P1C y RD7/P1D.

# Registro CCP1CON

## dirección 17h

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
P1M1	P1M0	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0
Bit7							bit0

P1M<1:0>: Configuración de la salida PWM

Si los bits CCP1M<3:2> = 00, 01 o 10

XX = RC2/P1A/CCP1 es E/S en el modo captura/comparación, P1B, P1C y P1D son E/S.

Si los bits CCP1M<3:2> = 11

00 = RC2/P1A/CCP1 es **salida de la señal PWM**. Las líneas P1B, P1C y P1D son E/S

01 = **Puente en H directo**; RC2/P1A activa, P1B, P1C inactivas y P1D salida PWM.

10 = **Semipuente en H** RC2/P1A y P1B salidas PWM, P1C y P1D son E/S

11 = **Puente en H inverso** RC2/P1A y P1D inactivas y P1B salida PWM y P1C inactiva.

# Registro CCP1CON

## dirección 17h

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
P1M1	P1M0	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0
Bit7							bit0

DC1B<1:0>: Representan los dos bits de menor peso del ciclo útil de la señal PWM

Modo Captura

No se usan

Modo Comparación

No se usan

Modo PWM

Son los bits de menor peso del ciclo útil de la señal PWM y se concatenan con los bits de CCPR1L

# Registro CCP1CON

## dirección 17h

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
P1M1	P1M0	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0
Bit7							bit0

CCP1M<3:0>: Selección del modo de trabajo del módulo ECCP1

0000 = **Desconecta el módulo ECCP1**

0001 = No se usa

0010 = **Modo de comparación**. La salida RC2/P1A/CCP1 cambia de estado al coincidir (CCP1IF=1).

0011 = No se usa

0100 = **Modo Captura a cada flanco descendente** de entrada RC2/P1A/CCP1.

0101 = **Modo Captura a cada flanco ascendente** en la entrada RC2/P1A/CCP1.

0110 = **Modo Captura a cada 4 flancos ascendentes** de entrada RC2/P1A/CCP1.

0111 = **Modo Captura a cada 16 flancos ascendentes** de entrada RC2/P1A/CCP1.

1000= **Modo Comparación** La salida RC2/P1A/CCP1 se activa al coincidir (CCP1IF=1).

1001 = **Modo Comparación**: La salida RC2/P1A/CCP1 se desactiva al coincidir (CCP1IF=1).

1010 = **Modo Comparación** RC2/P1A/CCP1 no cambia al coincidir, simplemente (CCP1IF=1).

1011= **Modo Comparación Disparo de eventos**: al coincidir se borra el Timer1 o el 2. CCP1IF=1.

1100 = **Modo PWM**. RC2/P1A y P1C activos a "1" P1B y P1D activos a nivel alto

1101 = **Modo PWM**. RC2/P1A y P1C activos a "1" P1B y P1D activos a nivel "0"

1110 = **Modo PWM**. RC2/P1A y P1C activos a "0" P1B y P1D activos a nivel "1"

1111= **Modo PWM**. RC2/P1A y P1C activos a "0" P1B y P1D activos a nivel "0"

# Registro CCP2CON

## dirección 1Dh

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
---	---	DC2B1	DC2B0	CCP2M3	CCP2M2	CCP2M1	CCP2M0
Bit7							bit0

DC2B<1:0>: Representan los dos bits de menor peso del ciclo útil de la señal PWM

Modo Captura

No se usan

Modo Comparación

No se usan

Modo PWM

Son los bits de menor peso del ciclo útil de la señal PWM y se concatenan con los bits de CCPR1L

# Registro CCP2CON

## dirección 1Dh

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
---	---	DC2B1	DC2B0	CCP2M3	CCP2M2	CCP2M1	CCP2M0
Bit7							bit0

CCP1M<3:0>: Selección del modo de trabajo del módulo ECCEP1

0000 = **Desconecta el módulo ECCEP1**

0001 = No se usa

0010 = No se usa

0011 = No se usa

0100 = **Modo Captura a cada flanco descendente** de entrada RC1/CCP2.

0101 = **Modo Captura a cada flanco ascendente** en la entrada RC1/CCP2.

0110 = **Modo Captura a cada 4 flancos ascendentes** de entrada RC1/CCP2.

0111 = **Modo Captura a cada 16 flancos ascendentes** de entrada RC1/P1A/CCP2.

1000= **Modo Comparación** La salida RC1/CCP2 se activa al coincidir (CCP12F=1).

1001 = **Modo Comparación**: La salida RC1/CCP2 se desactiva al coincidir (CCP2IF=1).

1010 = **Modo Comparación** RC1/CCP2. no cambia al coincidir, simplemente (CCP2IF=1).

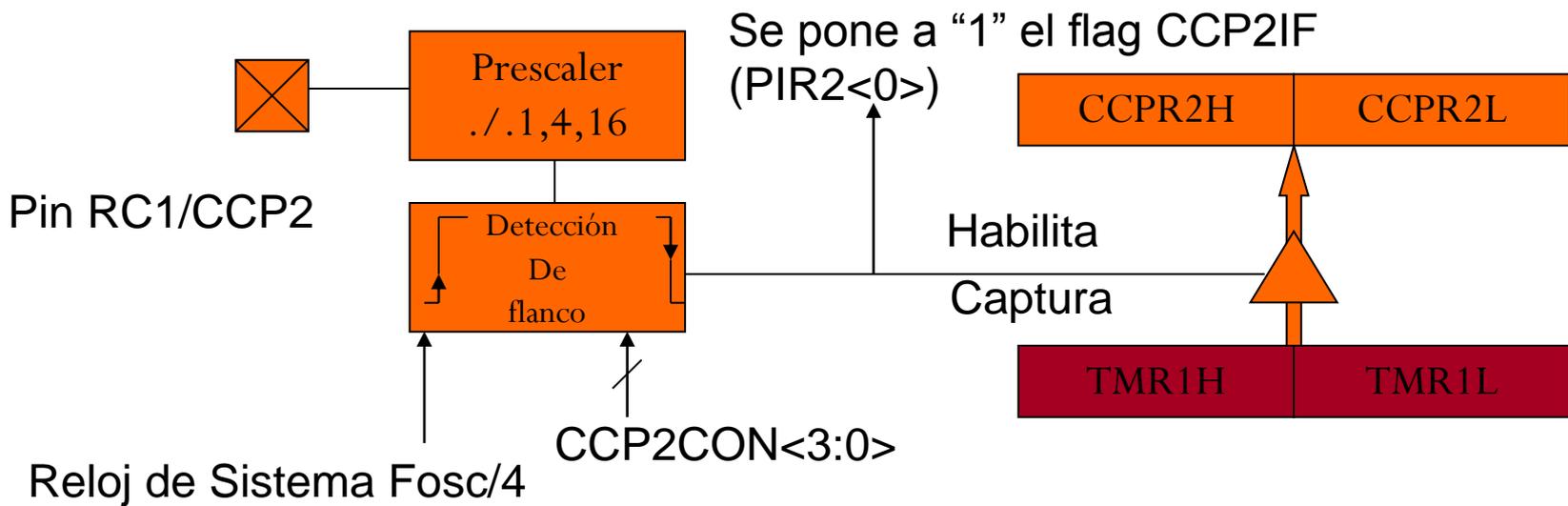
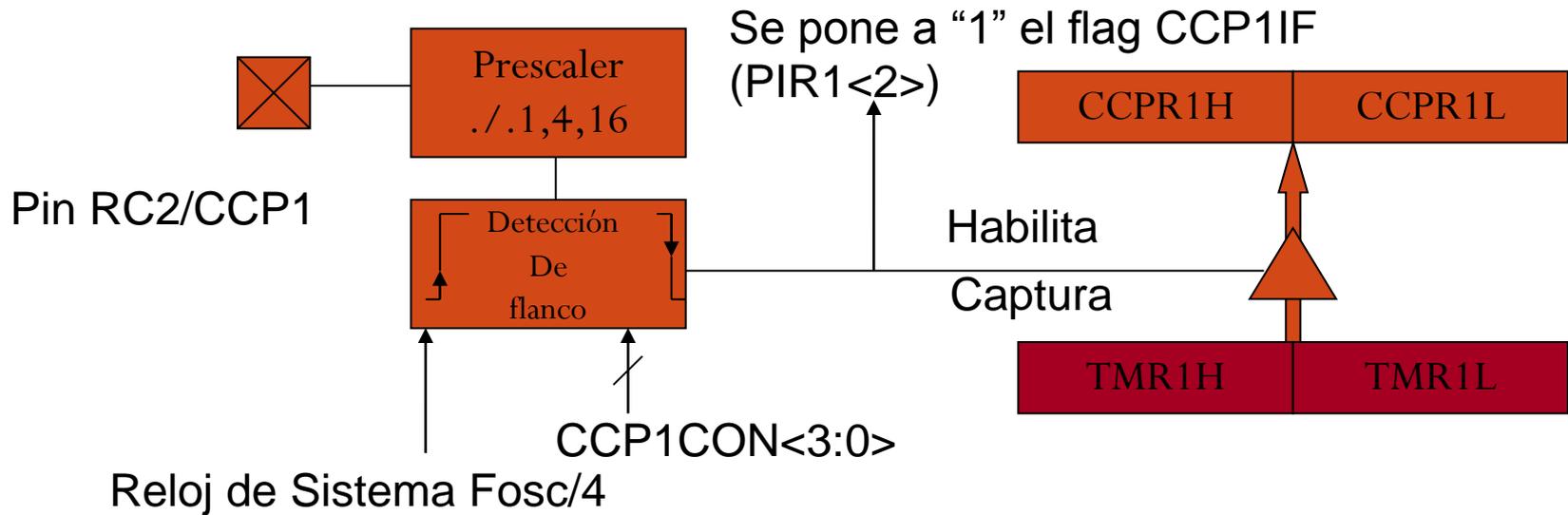
1011= **Modo Comparación Disparo de eventos**: al coincidir se borra el Timer1 y se inicia una nueva conversión A/D (si está habilitado). CCP1F=1. RC1/CCP2 no cambia de estado

11XX = **Modo PWM**. RC1/CCP2 salida de señal PWM.

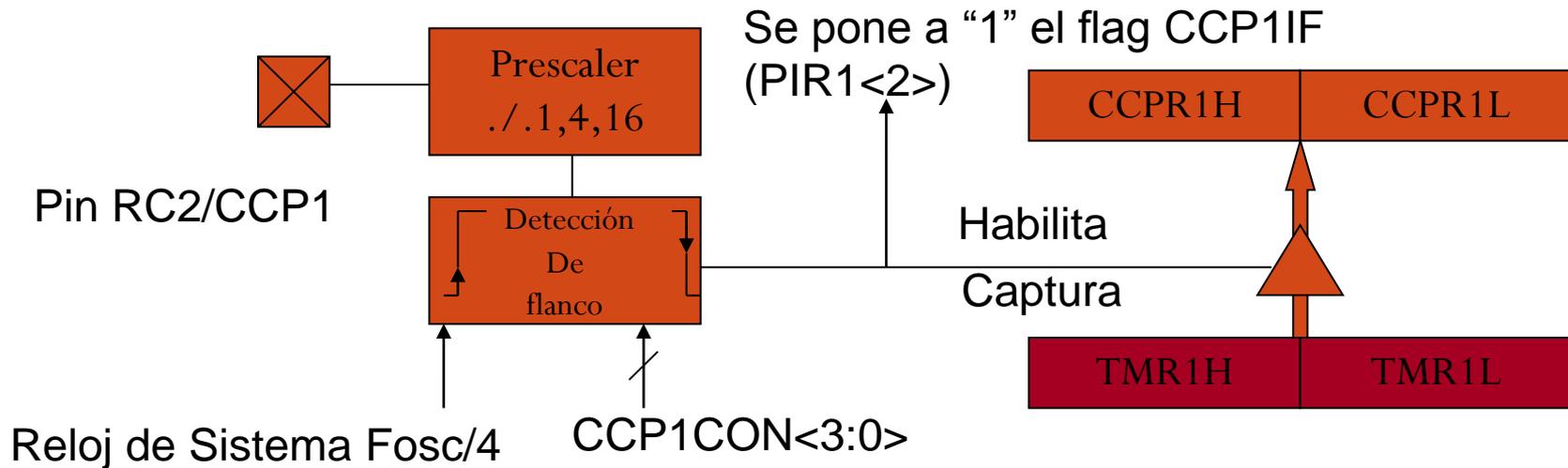
# Modo Captura

- Su funcionamiento es común a ambos módulos CCP1 y CCP2.
- Cada módulo consta de un registro de 16 bits formado por la pareja de registros CCPRxH:CCPRxL.
- Cuando sucede un evento por la patilla RCx/CCPx, el valor actual del TIMER1 se copia automáticamente sobre la pareja de registros CCPRxH:CCPRxL. Conocido el cada cuanto tiempo se incrementa el Timer1 es fácil medir el tiempo transcurrido entre un evento y otro.
- Los eventos por los que se captura el Timer1 pueden ser:
  - Captura cada 1 flanco descendente aplicado en RCx/CCPx
  - Captura cada 1 flanco ascendente aplicado en RCx/CCPx
  - Captura cada 4 flancos ascendentes aplicados en RCx/CCPx
  - Captura cada 16 flancos ascendentes aplicados en RCx/CCPx
- Las patillas RCx/CCPx deberán configurarse como entradas
- El módulo TMR1 debe de estar configurado en modo contador de impulsos externos o temporizador sincronizado.

# Modo de Captura

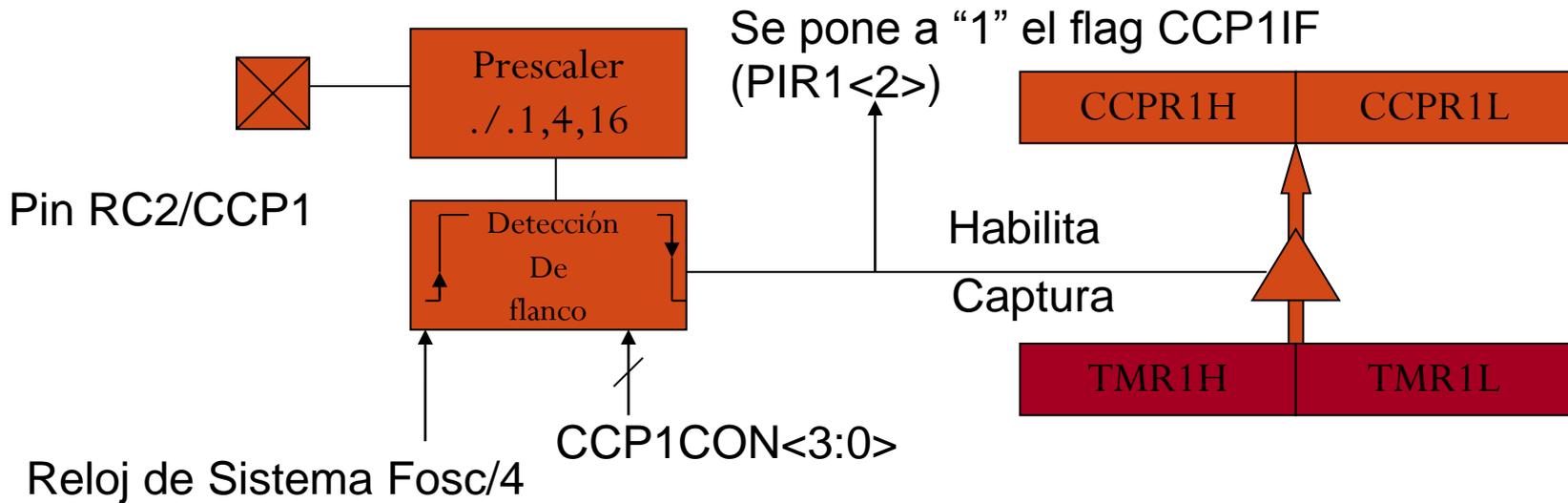


# Modo de Captura



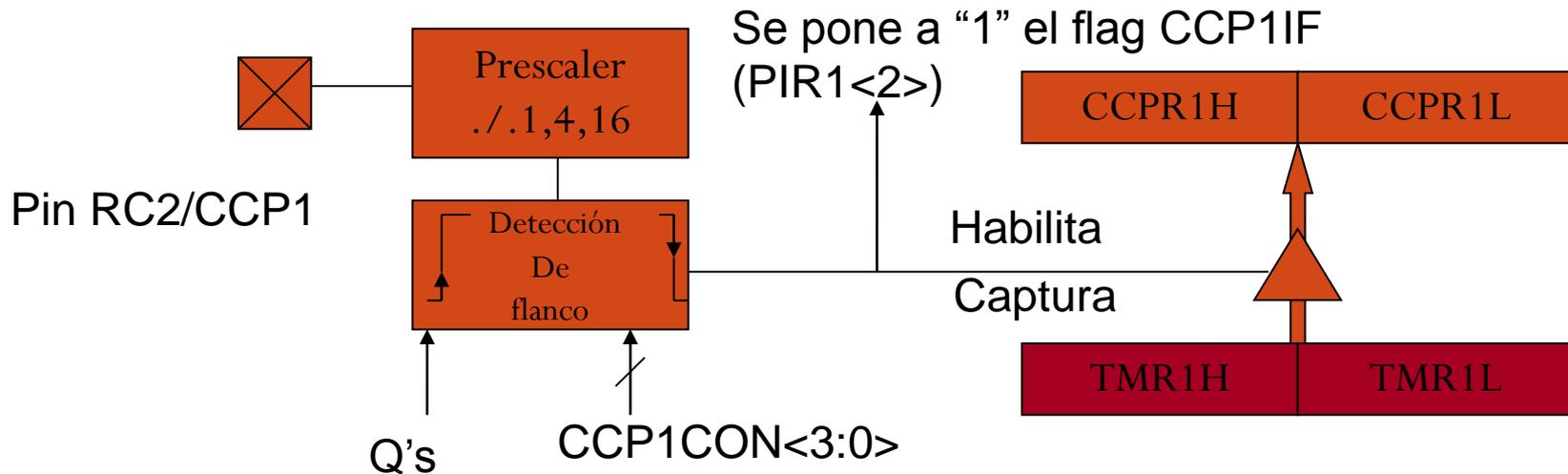
- Los registros CCPRx capturan el valor del TMR1 cuando sucede un evento en el pin CCPx. El evento se configura con los bits CCPxM3-0 del registro CCPxCON.
- Con la captura se activa el flag CCPIF, el cual debe ser borrado por software
- La interrupción se habilita con el bit CCPxIE

# Modo de Captura



- Se debe configurar el CCPx como entrada a través del registro TRISC
- El Timer1 debe estar en modo temporizador o modo contador sincronizado
- Al cambiar el modo de captura hay que tener inhabilitadas las interrupciones y borrar el flag para evitar interrupciones espúreas.

# Modo de Captura



- Cuando el TMR1 trabaja en modo captura con el CCPx, debe estar configurado para trabajar como temporizador o como contador síncrono, nunca en modo asíncrono.
- Si el pin RCx/CCPy se configura con resistencia de pull-up, una escritura en el puerto, puede causar un condición de captura.
- Es recomendable desactivar el módulo CCP poniendo a "0" el bit CCP1IE (PIE1<2>) cuando se van a cambiar las condiciones de funcionamiento en el modo captura, para evitar que se produzcan falsas interrupcion durante la operación.

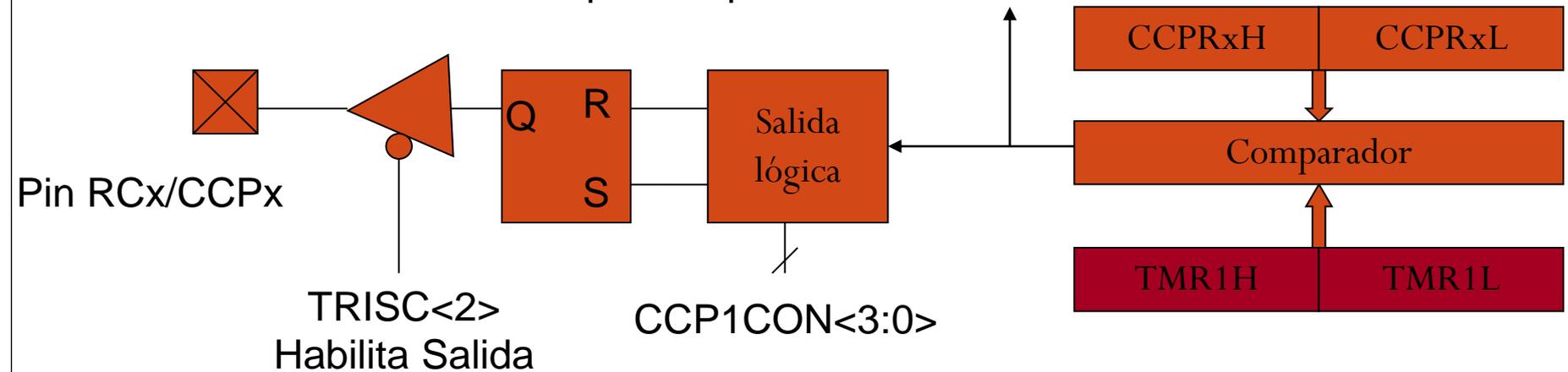
# Modo Comparación

- Consiste en comparar constantemente el valor del Timer1 con el valor previamente almacenado en la pareja de registros CCPR<sub>x</sub>\_H:CCPR<sub>x</sub>\_L. Cuando ambos valores coinciden el bit de estado CCP<sub>x</sub>IF correspondiente se activa y se puede producir una de las siguientes acciones:
  - La patilla de salida RC<sub>x</sub>/CCP<sub>x</sub> cambia de estado
  - La patilla de salida RC<sub>x</sub>/CCP<sub>x</sub> se pone a nivel “1”
  - La patilla de salida RC<sub>x</sub>/CCP<sub>x</sub> se pone a nivel “0”
  - Genera interrupción (si se habilitan los bits CCP<sub>x</sub>IE correspondientes así como los bit PEIE y GIE)
  - Dispara eventos internos:
    - En el caso del módulo CCP1 se puede borrar automáticamente tanto el Timer1 como el Timer2.
    - En el caso de CCP2 se borra el Timer1 y , si está activado, se inicia una nueva conversión A/D
  - El pin RC<sub>x</sub>/CCP<sub>x</sub> debe estar configurado como salida y el TMR1 debe de estar configurado como contador de eventos externos o temporizador sincronizado.

# Modo Comparación

Disparo Especial

Se pone a "1" el flag CCPxIF

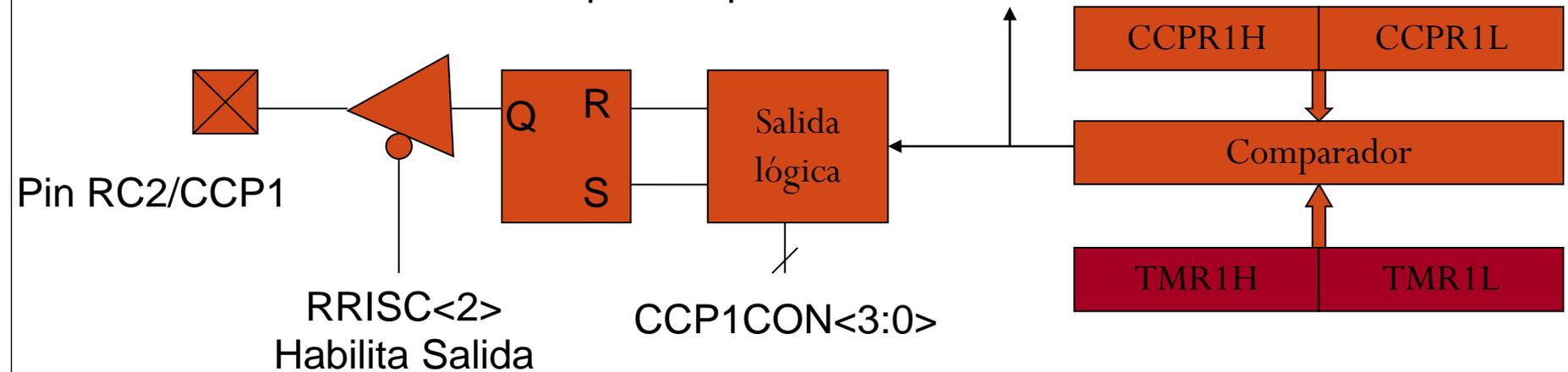


- Se compara el registro CCPRx con el valor del registro TMR1. Cuando se produce igualdad se realiza una acción sobre el pin CCPx, la cual viene dada por el modo establecido en CCPxM3:0.
- Se activa el flag CCPxIF
- Se genera una interrupción si está habilitada con CCPxIE

# Modo Comparación

Disparo Especial

Se pone a "1" el flag CCP1IF

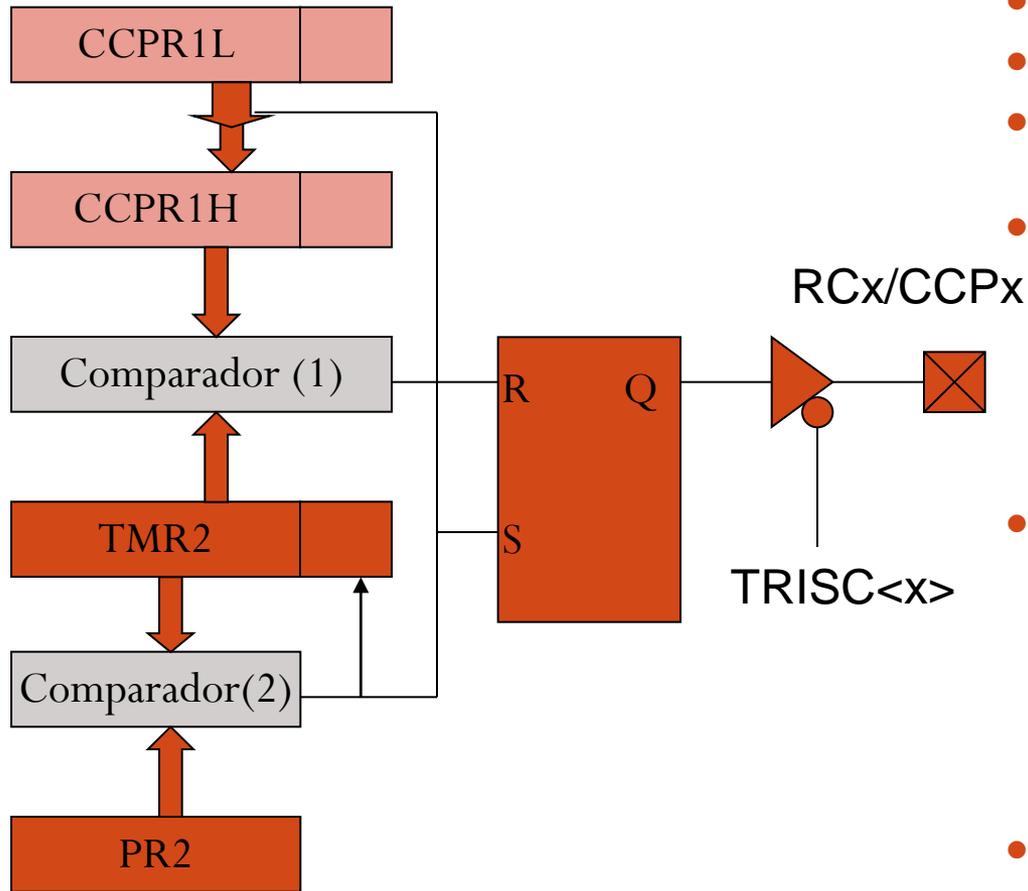


- En el modo “interrupción software” no se realiza ninguna acción en el pin CCPx.
- El modo “Special Event Trigger” :
  - En el caso del módulo CCP1 se puede borrar automáticamente tanto el Timer1 como el Timer2.
  - En el caso de CCP2 se borra el Timer1 y , si está activado, se inicia una nueva conversión A/D
- Se debe configurar el pin CCPx como salida, poniendo a “0” el TRISC. El Timer1 debe estar en modo Temporizador o Contador sincronizado.

# Modo PWM (Pulse-Width Modulation)

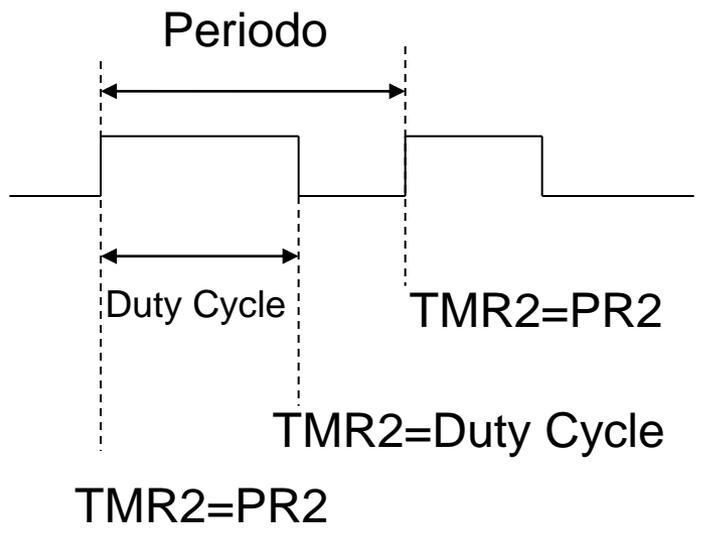
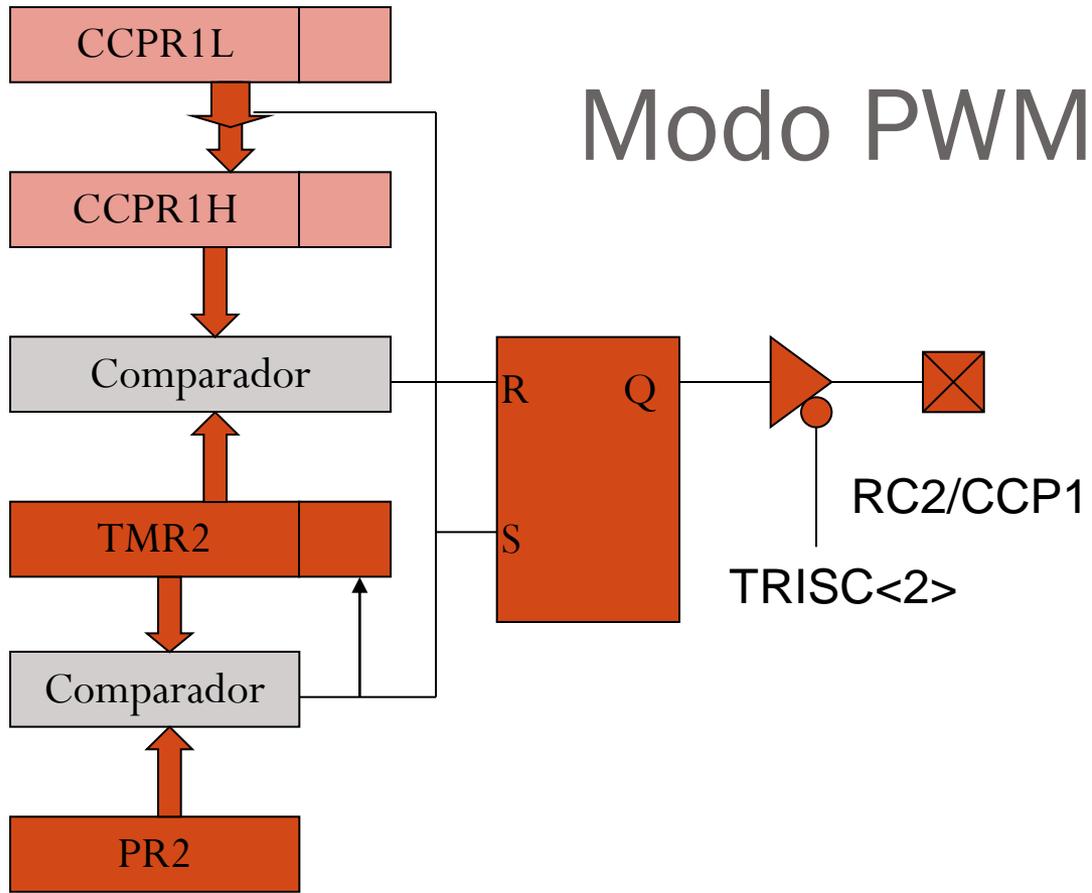
- Se emplea para generar una señal de salida modulada en anchura de impulsos por el pin RC<sub>x</sub>/CCP<sub>x</sub>.
- El módulo CCP1 tiene algunas mejoras sobre el CCP2
- El periodo de la señal se ajusta mediante el registro de periodos PR2. La anchura del pulso (*duty cycle*) se ajusta mediante los 8 bits del registro CCPR<sub>x</sub>L, que se conectan con los bit <5:4> del registro CCP<sub>x</sub>CON, obteniéndose una resolución de 10 bits.

# Modo PWM



- Tiene 10 bits de resolución
- Los pines CCPx son la salida PWM
- La base de tiempos es el Timer2 más dos bits adicionales para tener los 10
- El periodo se determina con el registro PR2, ya que cuando  $TMR2=PR2$ :
  - Se borra el TMR2
  - Se pone a "1" el pin CCPx
  - Se pasa el valor del CCPRxL al CCPRxH (para evitar *glitch*)
- Seguidamente el Timer2, mediante el comparador (2) se compara con el registro CCPRxH. Cuando coinciden se pone a "0" finalizando el ciclo útil. El Timer2 sigue avanzando hasta que nuevamente coincide con PR2, repitiéndose el ciclo indefinidamente.
- El ciclo de trabajo se determina con el contenido del CCPRxL y los dos bits de CCPxCON (CCPxX y CCPxY)

# Modo PWM



$$T_{PWM} = (PR2+1) \times 4 \times T_{OSC} \times TMR2 \times \text{valor del prescaler}$$

El ciclo útil, es decir el tiempo que la señal está a nivel alto es:

$$T_{CP} = (CCPRxL:CCPxX:CCPxY) \times T_{OSC} \times TMR2 \times \text{valor del preescaler}$$

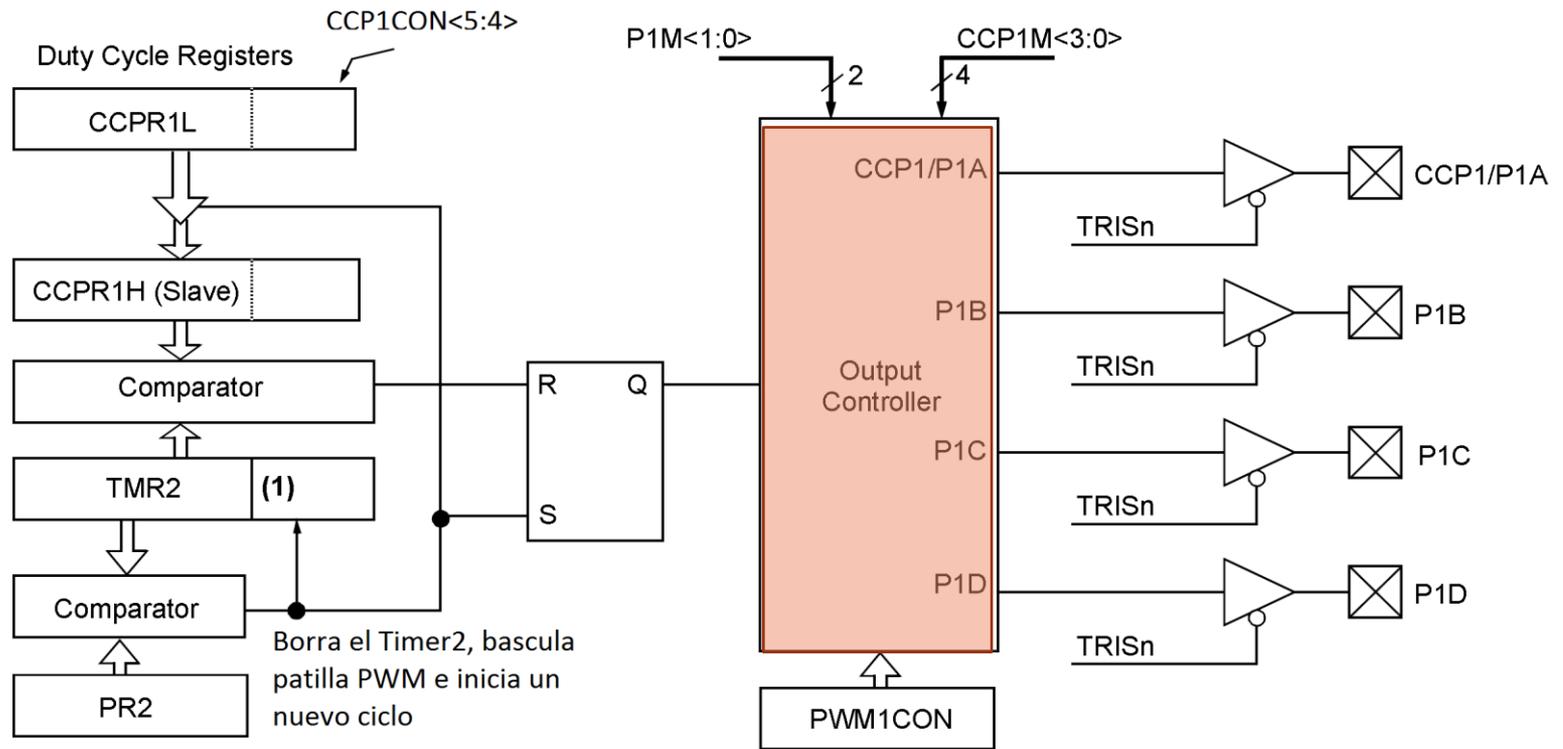
# Secuencia para la habilitación del PWM

- Para generar una señal PWM por cualquiera de los dos módulos se deben seguir los siguientes pasos.
  1. Desactivar la patilla CCPx deseada, configurando como entrada mediante el correspondiente registro TRIS.
  2. Ajustar el período de la señal PWM cargando el registro PR2 con el valor apropiado.
  3. Configurar el módulo CCPx en el modo PWM cargando el registro CCPxCON con su correspondiente valor.
  4. Ajustar el ciclo útil de la señal PWM con la anchura deseada. Para ello cargamos el valor apropiado de 10 bits en los registros CCPxL:CCPxCOM<5:4>.
  5. Configurar y activar el Timer2
    - Borrar el bit TMR2IF del registro PIR1
    - Seleccionar el preescaler del Timer2 mediante los bits T2CKPS del registro T2COM
    - Activar el TMR2 mediante el bit TMR"ON del registro T2CON
  6. Activar la Patilla de salida PWM cuando empiece un nuevo ciclo:
    - Esperar el desbordamiento del TMR2, cuando el bit TMR2IF del registro PIR1 se active
    - Configurar la patilla CCPx como salida mediante su correspondiente registro TRIS

# Modo PWM Mejorado

- El modo mejorado que no está presente en los PIC16F87X dispone de 5 tipos de PWM diferentes:
  - Salida PWM sencilla (la que ya hemos visto) con redireccionamiento de salida (“*Pulse Steering*”).
  - Salida PWM para el control de un semi-puente en H (*Half Bridge*)
  - Salida PWM para control de un puente en H (“*Full Bridge*”) en modo directo
  - Salida PWM para control de un puente en H (“*Full Bridge*”) en modo inverso.
  - Autodesconexión de las salidas (“*auto-shutdown*”)

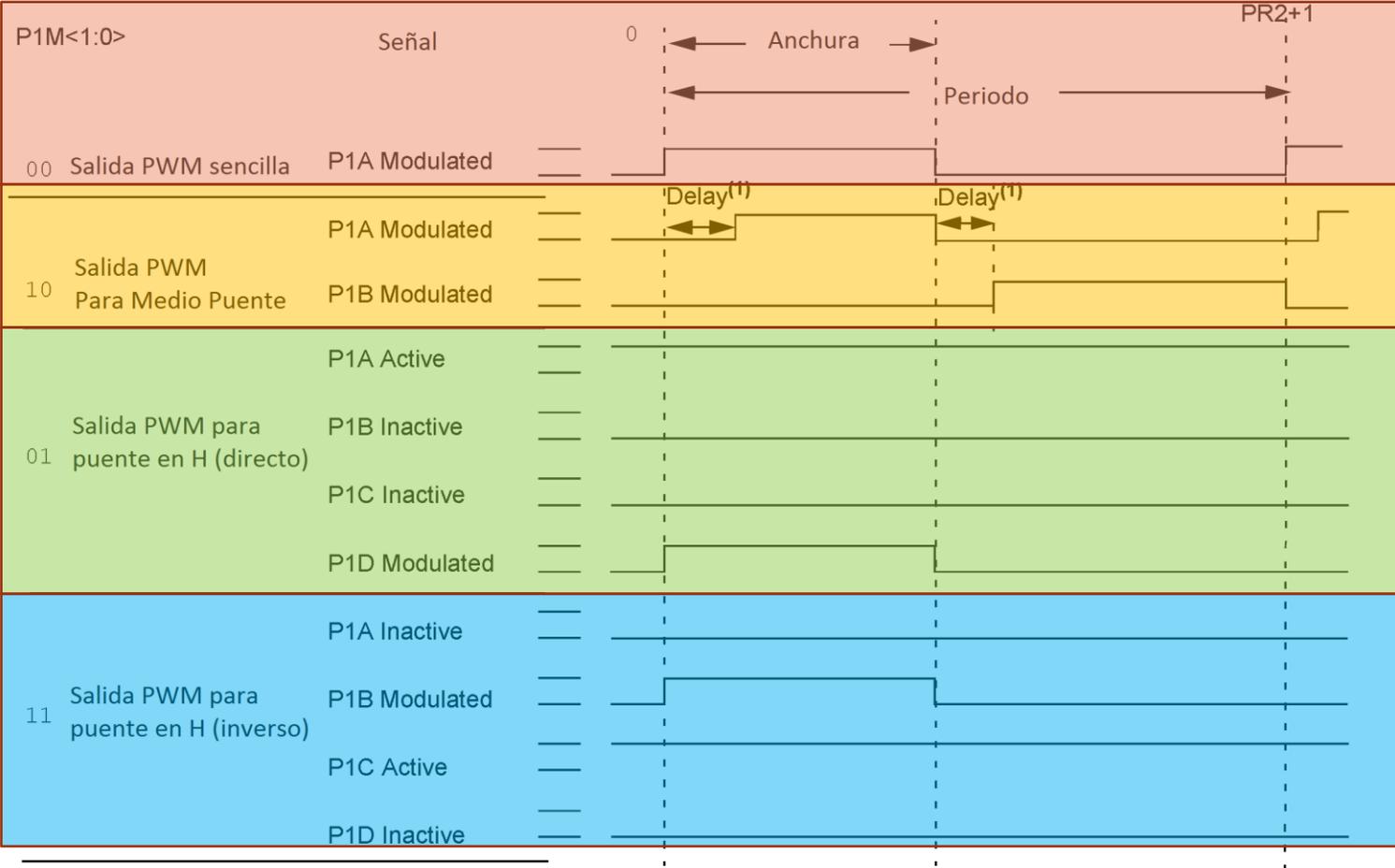
# El módulo CCP1



# Modo PWM Mejorado

- El circuito de modulación es común a ambos moduladores ECCP1 y CCP2. En este módulo ECCP1 se ha añadido un circuito controlador de salidas que se encarga de distribuir la señal de salida PWM sobre cuatro patillas: RC2/CCP1/P1A, P1B, P1C y P1D. Mediante los bits P1M<1:0> del registro CCP1CON cual de los cuatro tipos de salida PWM posible vamos a sacar por esas patillas y mediante los bits CCP1M<3:0>, la polaridad de las mismas.

# Salidas PWM



# Salida Sencilla PWM

- Este modo se selecciona cuando los bits CCP1M<3:2> swl registro CCP1CON están a “11”Y los bits P1M<1:0> A “00”. Es la forma más sencilla de utilizar el PWM cuyo periodo lo fija **PR2** y la anchura del impulso se fija mediante **CCPR1L**. Igual que el módulo CCPR2.
- Los PIC16F88X nos permiten que la señal esté presente en una, dos, tres o las cuatro salidas **RC2/CCP1/P1A**, **P1B**, **P1C** y **P1D**. Esto es lo que se conoce como redireccionamiento del pulso o “Pulse Steering”.
- El registro **PSTRCON** permite determinar sobre que patillas se quiere poner la misma señal PWM. Por defecto después de un Reset queda selecciona la patilla RC2/CCP1/P1A.

# Registro PSRCON

## dirección 9Dh

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-1
---	---	---	<b>STRSYNC</b>	<b>STRD</b>	<b>STRC</b>	<b>STRB</b>	<b>STRA</b>
Bit7							bit0

**STRSYNC:** Sincronismo de la señal PWM de salida

1= La salida de la señal se sincroniza con el siguiente periodo de la señal PWM

0= La salida no se sincroniza.

**STRD:** Habilitación de P1D

1= La señal PWM sale por la patilla P1D. La polaridad se establece mediante CCP1M<1:0>

0 = La patilla P1D se comporta como línea de E/S de propósito general

**STRC:** Habilitación de P1C

1= La señal PWM sale por la patilla P1C. La polaridad se establece mediante CCP1M<1:0>

0 = La patilla P1C se comporta como línea de E/S de propósito general

**STRB:** Habilitación de P1B

1= La señal PWM sale por la patilla P1B. La polaridad se establece mediante CCP1M<1:0>

0 = La patilla P1B se comporta como línea de E/S de propósito general

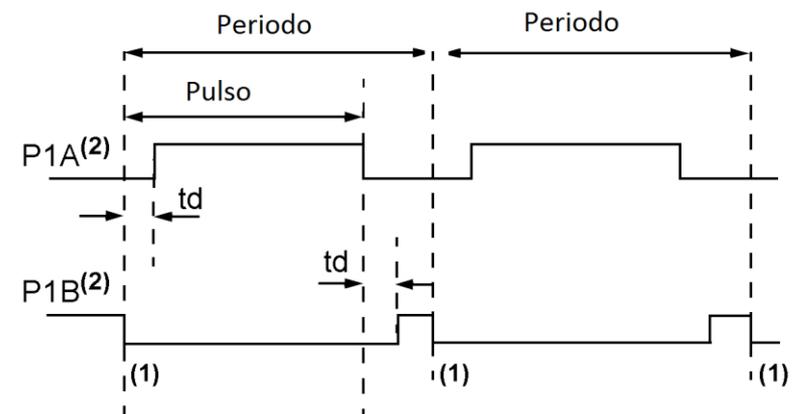
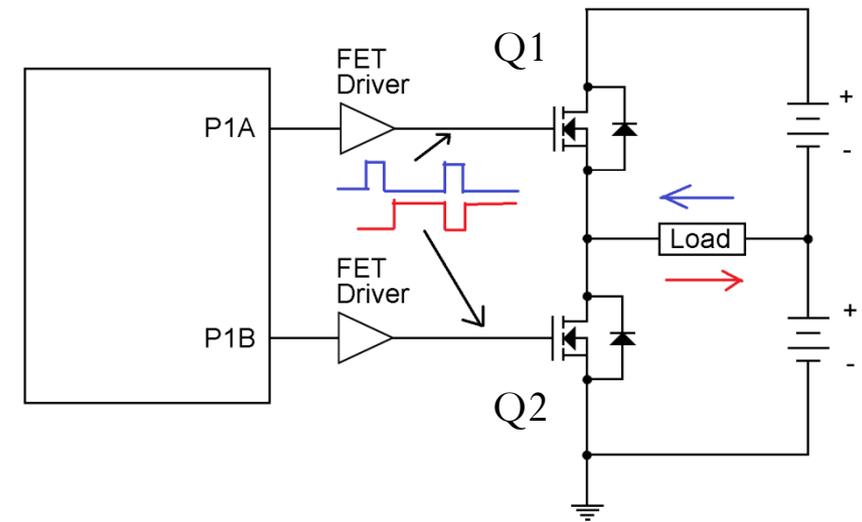
**STRA:** Habilitación de P1A

1= La señal PWM sale por la patilla P1A. La polaridad se establece mediante CCP1M<1:0>

0 = La patilla P1A se comporta como línea de E/S de propósito general

# Salida PWM para el control de un semipuente en H

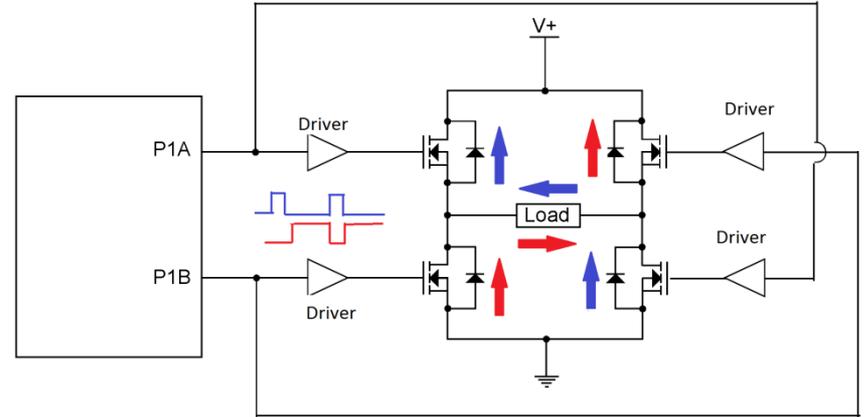
- En este modo se utilizan dos patillas de salida. La señal PWM se obtiene por la patilla **RC2/CCP1/P1A** y su complementaria **P1B**
- Entre ambas señales desfasadas  $180^\circ$ , se intercala el retardo “**td**” ajustable por el usuario que retrasa la transición de los flancos de una señal respecto a la otra.
- Este tipo de salidas PWM complementarias nos permiten controlar sistemas “push-pull” empleados en aplicaciones típicas de conmutación.
- En el circuito de la figura suponiendo que el nivel activo seleccionado para las señales sea el “1”, cuando por RC2/CCP1/P1A la señal PWM está a “1” se activa Q1 y la corriente fluye por la carga en un sentido.
- La señal PWM en P1B está a “0” bloqueando Q2. Cuando ambas señales cambian de estado el proceso se invierte y ahora es Q2 quien hace que la corriente fluya en sentido contrario



(1) En este instante PR2=TMR2 comienza un nuevo ciclo

# Control de puente en H

- Para el control de un puente en H como el de la figura se emplean dos parejas de transistores para cada sentido Q1:Q2 y Q3:Q4.
- En este tipo de salida PWM se dispone de un sistema de temporización que retarda ambas señales PWM entre sí. Recibe el nombre de “retardo de banda muerta” (Dead-Band Delay), que se utiliza para prevenir cortocircuitos en el semipunto o puente en H. Efectivamente solo puede estar conduciendo un transistor (o pareja de transistores) cada vez. Si por casualidad ambos transistores dejan pasar corriente a la vez, simultáneamente, la carga podría quedar en cortocircuito. Esta situación se puede dar aunque las señales sean complementarias, debido al tiempo de retardo de los transistores al conmutar



# Retardo de banda muerta

- El retardo de banda muerta se puede ajustar entre 0 y 127, mediante los 7 bits del registro PWM1CON. EL valor binario que se introduzca, representa el número de ciclos máquina ( $4T_{osc}$ ) de la temporización que se va a aplicar, en las transiciones de una señal PWM respecto a la otra. Si este valor es mayor que el valor del ciclo útil de la señal, las salidas permanecerán inactivas.

# Registro PWM1CON

## dirección 9Bh

R/W-0	R/W-1						
PRSEN	PDC6	PDC5	PDC4	PDC3	PDC2	PDC1	PDC0
Bit7							bit0

**PRSEN:** Bit de reinicio del PWM

1= Después de una autodesconexión, el bit **ECCPASE** se borra y la señal PWM se restaura automáticamente al desaparecer la causa que la provocó.

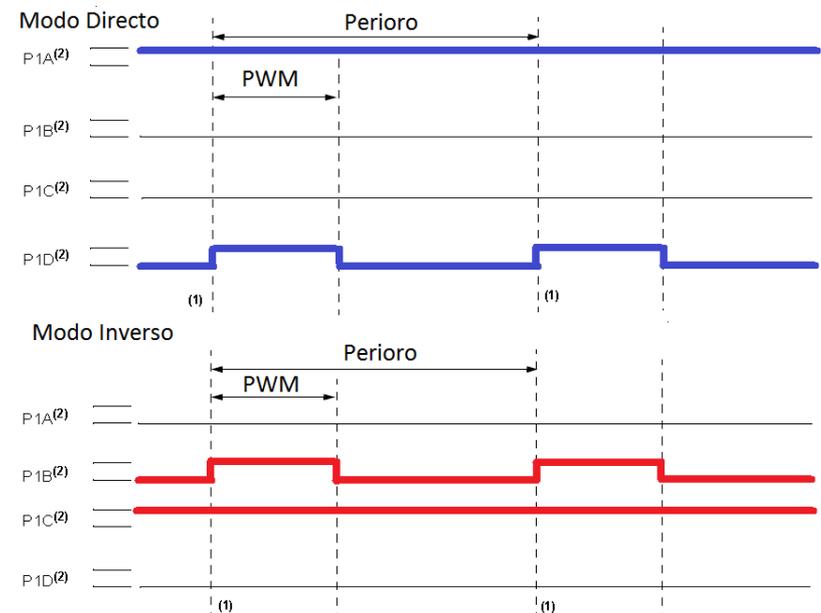
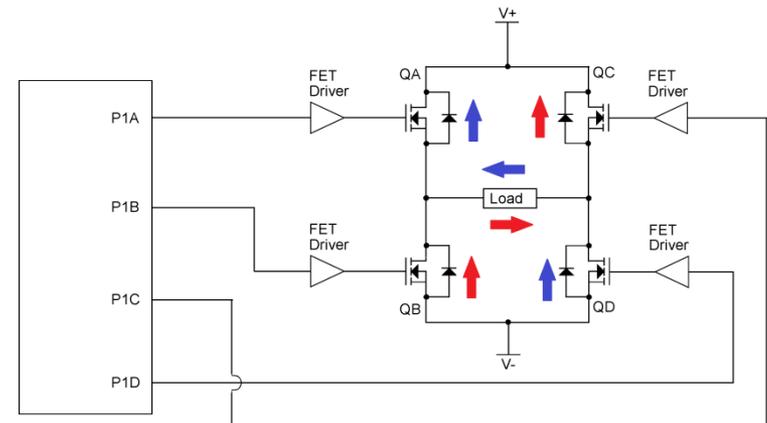
0= Después de una autodesconexión, el bit **ECCPASE** y la señal PWM se deben restaurar por software.

**PDC<6:0>:** Valor de retardo de banda muerta

Representa en binario, el número de ciclos máquina (84Tosc) que hay que temporizar en la transición de una señal PWM respecto a la otra.

# Salida PWM para el control de un puente en H (Full-Bridge) en modo directo-inverso

- En este modo PWM se emplean cuatro patillas de salida para controlar un puente en H (Full-Bridge).
- En el **modo directo** la patilla RC2/CCP1/P1A pasa al estado activo, activando a su vez al transistor QA. Por la patilla P1D aparece la señal PWM propiamente dicho, haciendo que el transistor QD module la potencia aplicada a la carga por donde la intensidad fluye en un sentido. Las patillas P1B y P1C se mantienen inactivas, por lo que la pareja de transistores QB:QC quedan bloqueados.
- En el modo inverso sucede justo lo contrario, La patilla P1C queda en estado activo y a su vez el transistor QC. Por la patilla P1B obtenemos la señal PWM gracias a la cual el transistor QB modula la potencia de la carga, por donde la intensidad fluye ahora en sentido contrario. Las señales P1A y P1D permanecen inactivas, bloqueando así a sus respectivos transistores QA y QD.



# Modo de autodesconexión

- La autodesconexión o *Auto-Shutdown* consiste en desconectar las salidas PWM, poniéndolas en un estado lógico determinado, cuando se produzca un evento o suceso externo. Podemos evitar así daños eléctricos en el circuito cuando se den determinadas situaciones.
- Las posibles causas que pueden provocar una autodesconexión pueden ser las siguientes y se definen en el registro para control de autodesconexión (ECCPAS):
  - Un nivel lógico “0” aplicado por la patilla RB0/INT
  - La salida del comparador analógico C1
  - La salida del comparador analógico C2
  - Activando, por software, el bit ECCPASE del registro ECCPAS

# Registro ECCPAS (Registro para control de la autodesconexión dirección 9Ch

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ECCPASE	ECCPAS2	ECCPAS1	ECCPS0	PSSAC1	PSSAC0	PSSBD1	PSSBD0
Bit7							bit0

**ECCPASE:** Bit de estado de autodesconexión

1= Se ha producido una autodesconexión. La salida PWM del módulo ECCP1 están deshabilitadas.

0= La salidas PWM del módulo ECCP1 están operativas

**ECCPAS<2:0>** Selección del origen para la autodesconexión

000 = Modo autodesconexión desconectado

001 = Cuando la salida del comparador C1 = "1"

010 = Cuando la salida del comparador C2 = "1"

011 = Cualquiera de las salidas de los dos comparadores están a "1"

100 = La patilla RB0/INT está a "0"

101 = La patilla RB0/INT está a "0" y la salida del comparador C1 está a "1".

110 = La patilla RB0/INT está a "0" y la salida del comparador C2 está a "1".

111 = La patilla RB0/INT está a "0" y cualquiera de las salidas de los dos comparadores están a "1"

# Registro ECCPAS (Registro para control de la autodesconexión dirección 9Ch

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ECCPASE	ECCPAS2	ECCPAS1	ECCPAS0	PSSAC1	PSSAC0	PSSBD1	PSSBD0
Bit7							bit0

**PSSAC<1:0>**: Estado de las patillas RC2/CCP1 y P1C cuando se produce la autodesconexión

00 = Las patillas RC2/CCP1/P1A y P1C se ponen a nivel "0"

01 = Las patillas RC2/CCP1/P1A y P1C se ponen a nivel "1"

1X= Las patillas RC2/CCP1/P1A y P1C quedan en estado de alta impedancia (desconectadas)

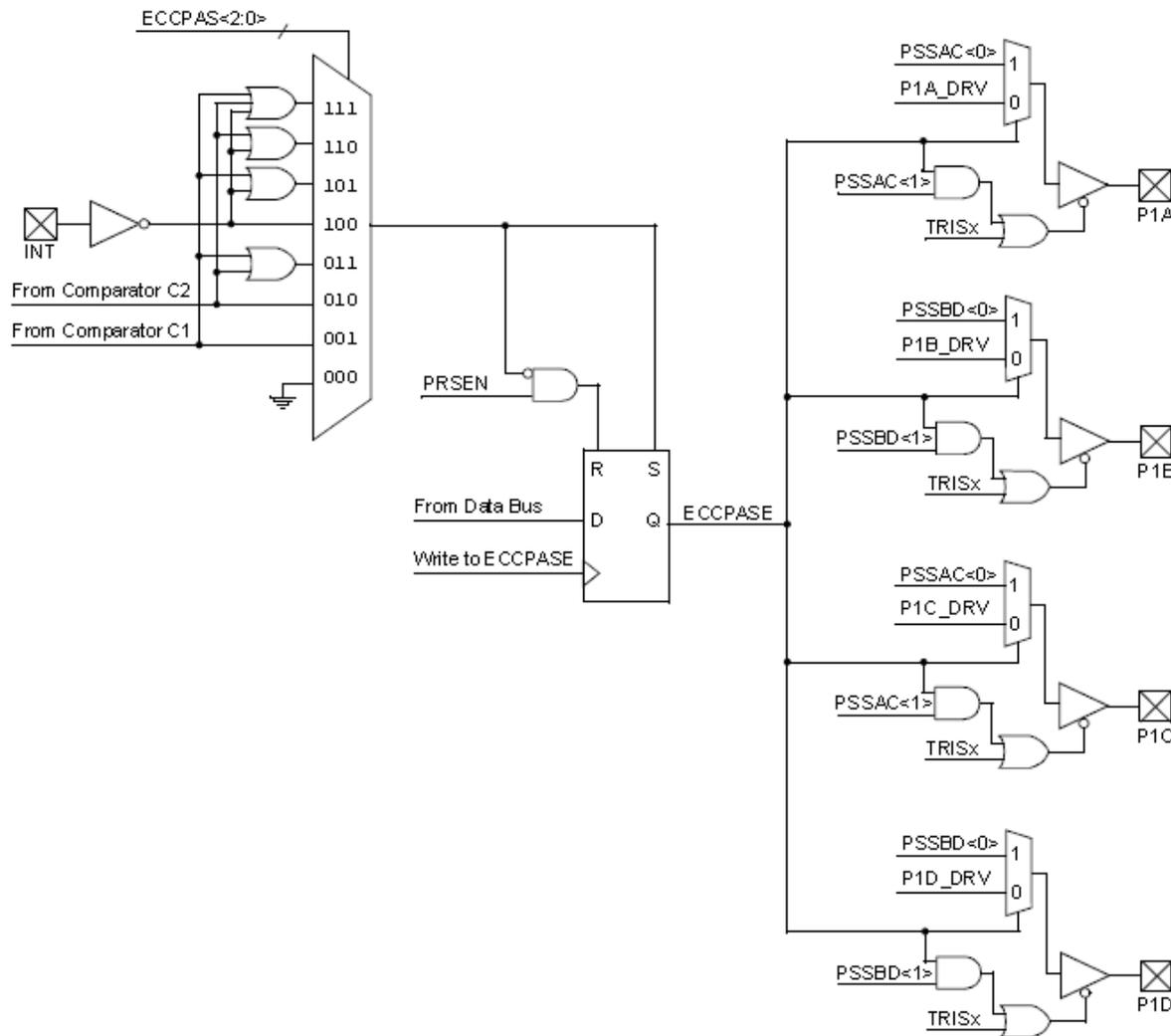
**PSSBD<10>** : Estado de las patillas P1B y P1D cuando se produce la autodesconexión

00 = Las patillas P1B y P1D se ponen a nivel "0"

01 = Las patillas P1B y P1D se ponen a nivel "1"

1X= Las patillas P1B y P1D quedan en estado de alta impedancia (desconectadas)

# Motivos de la autodesconexión



- El multiplexor permite seleccionar el motivo de la autodesconexión, evaluando las posibles causas: La patilla RB0/INT, comparadores C1 y C2 o escritura “1” en el bit ECCPASE.
- En condiciones normales de trabajo, los multiplexores de salida llevan a sus correspondientes patillas las señales PWM: P1A, P1B, P1C y P1D. Cuando se produce la autodesconexión los mismos multiplexores llevan a sus correspondientes patillas el estado lógico que hayamos preestalecido.

# Centros participantes en el proyecto: “Aprendizaje de la Electrónica a través de la Robótica” 2009-2011



- IES Politécnico Jesús Marín (Málaga)
- IES Juan de la Cierva (Madrid)
- IES Luis de Lucena (Guadalajara)
- IES María Moliner (Segovia)
- IES Joan Miró (San Sebastián de los Reyes. Madrid)
- IES Virgen de las Nieves (Granada)
- IES Torreón del Alcázar ( Ciudad Real)