Lab08 CCP Module

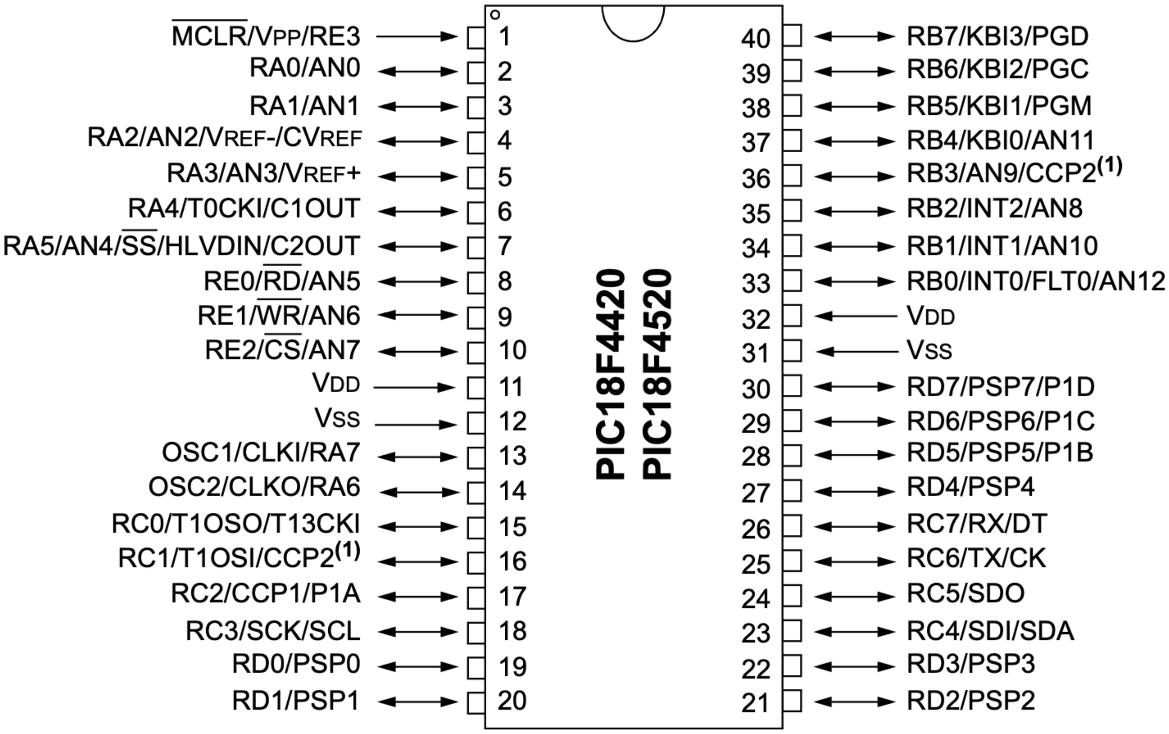
Reference: PIC18F4520 Data Sheet (https://ww1.microchip.com/downloads/aemDocuments/documents/OTH/ProductDocuments/DataSheets/PIC18F2420-2520-4420-4520-28-40-44-Pin-Microcontrollers-with-XLP-Technology-30009613F.pdf)

> Each Capture/Compare/PWM module is associated with a control register (generically, `CCPxCON`) and a data register (`CCPRx`).
> The data register, in turn, is comprised of two 8-bit registers: `CCPRxL` (low byte) and `CCPRxH` (high byte). All registers are both readable and writable.

CCP 為 Capture/Compare/PWM 的縮寫，每個模組都會對應到：

- Control Register (`CCPxCON`)：用來設定模組的運作模式（Capture、Compare 或 PWM）。
- Data Register (`CCPRx`)：用來存放比較值或捕獲到的時間值。
  其中，`CCPRx` 是由兩個 8-bit Register 組成：
  - `CCPRxL`：Low Byte
  - `CCPRxH`：High Byte

這些 Register 都能被讀取與寫入。



Timer

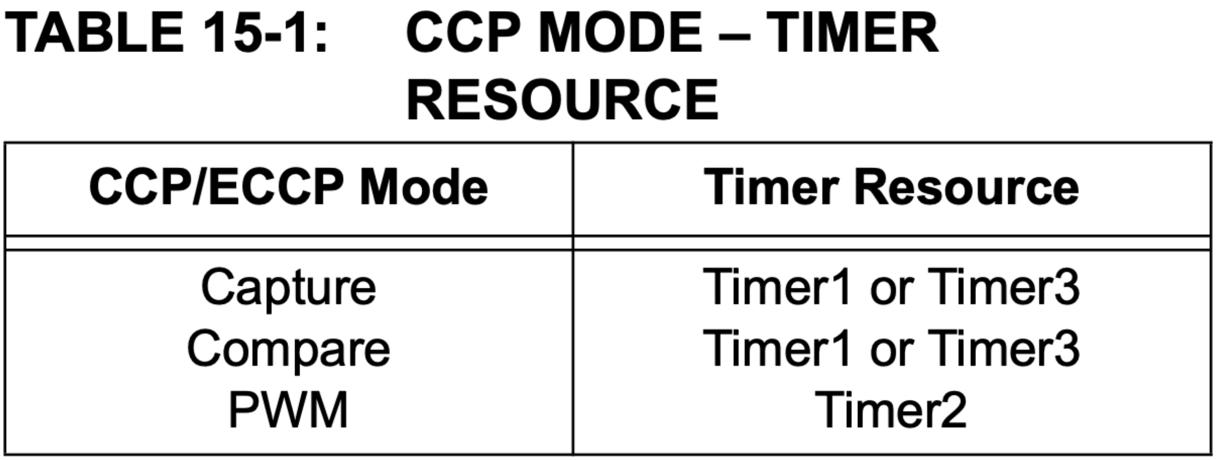
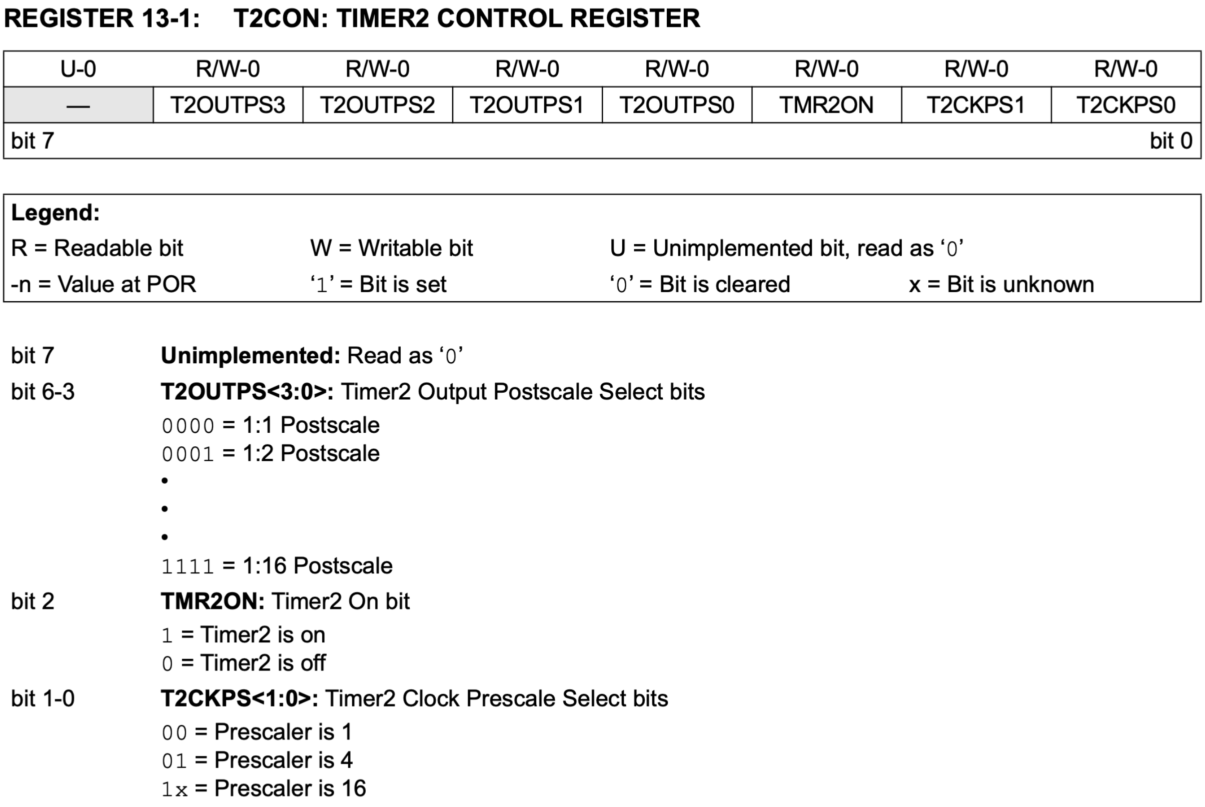


TABLE 15-1: CCP MODE – TIMER RESOURCE

CCP/ECCP Mode	Timer Resource
Capture	Timer1 or Timer3
Compare	Timer1 or Timer3
PWM	Timer2

- **PWM - T2CON**



REGISTER 13-1: T2CON: TIMER2 CONTROL REGISTER

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	T2OUTPS3	T2OUTPS2	T2OUTPS1	T2OUTPS0	TMR2ON	T2CKPS1	T2CKPS0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

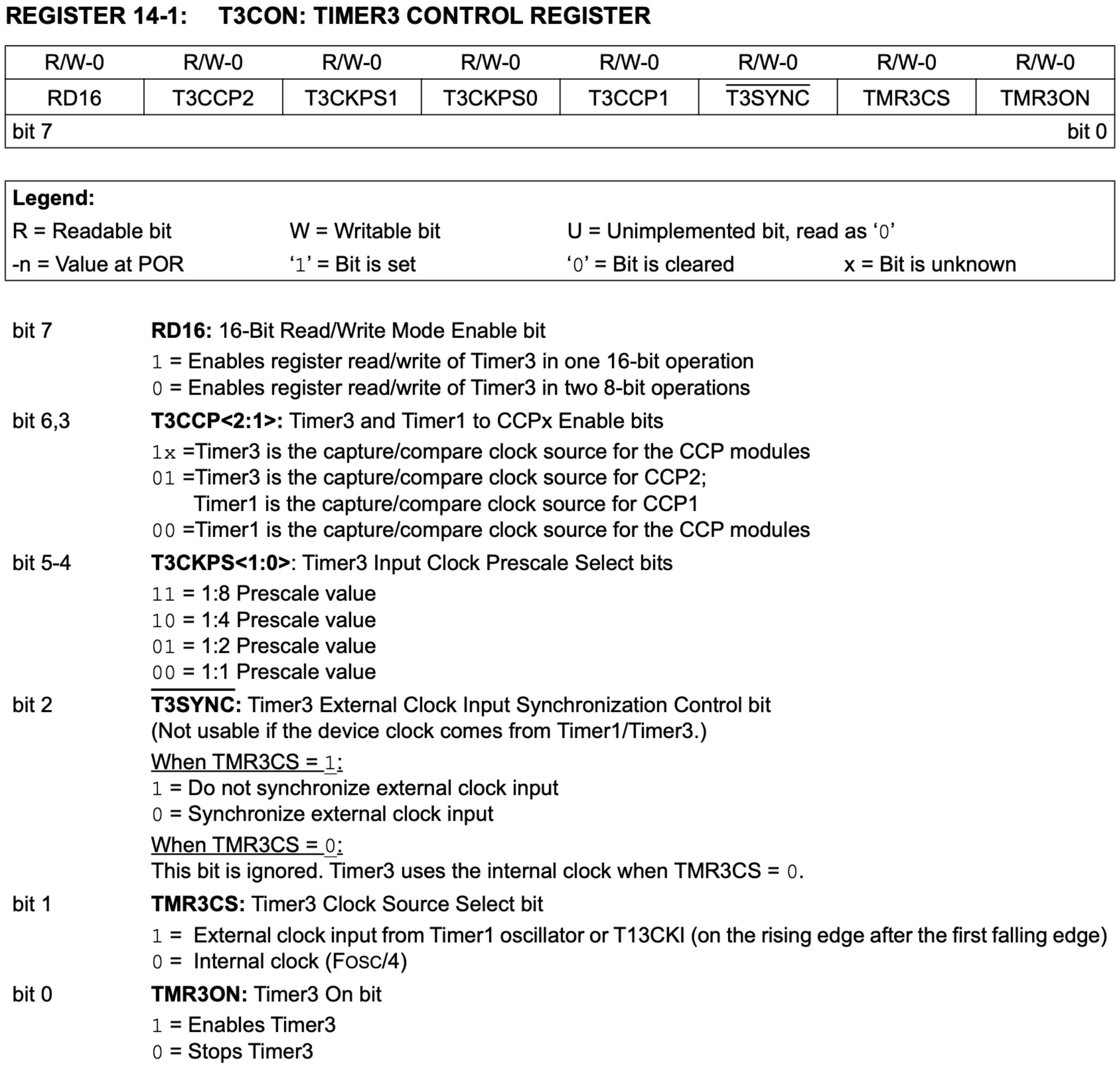
bit 7 **Unimplemented:** Read as ‘0’

bit 6-3 **T2OUTPS<3:0>:** Timer2 Output Postscale Select bits
0000 = 1:1 Postscale
0001 = 1:2 Postscale
•
•
•
1111 = 1:16 Postscale

bit 2 **TMR2ON:** Timer2 On bit
1 = Timer2 is on
0 = Timer2 is off

bit 1-0 **T2CKPS<1:0>:** Timer2 Clock Prescale Select bits
00 = Prescaler is 1
01 = Prescaler is 4
1x = Prescaler is 16

- ## Capture、Compare - T3CON



REGISTER 14-1: T3CON: TIMER3 CONTROL REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
RD16	T3CCP2	T3CKPS1	T3CKPS0	T3CCP1	$\overline{\text{T3SYNC}}$	TMR3CS	TMR3ON
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 7 **RD16:** 16-Bit Read/Write Mode Enable bit
1 = Enables register read/write of Timer3 in one 16-bit operation
0 = Enables register read/write of Timer3 in two 8-bit operations

bit 6,3 **T3CCP<2:1>:** Timer3 and Timer1 to CCPx Enable bits
1x =Timer3 is the capture/compare clock source for the CCP modules
01 =Timer3 is the capture/compare clock source for CCP2;
Timer1 is the capture/compare clock source for CCP1
00 =Timer1 is the capture/compare clock source for the CCP modules

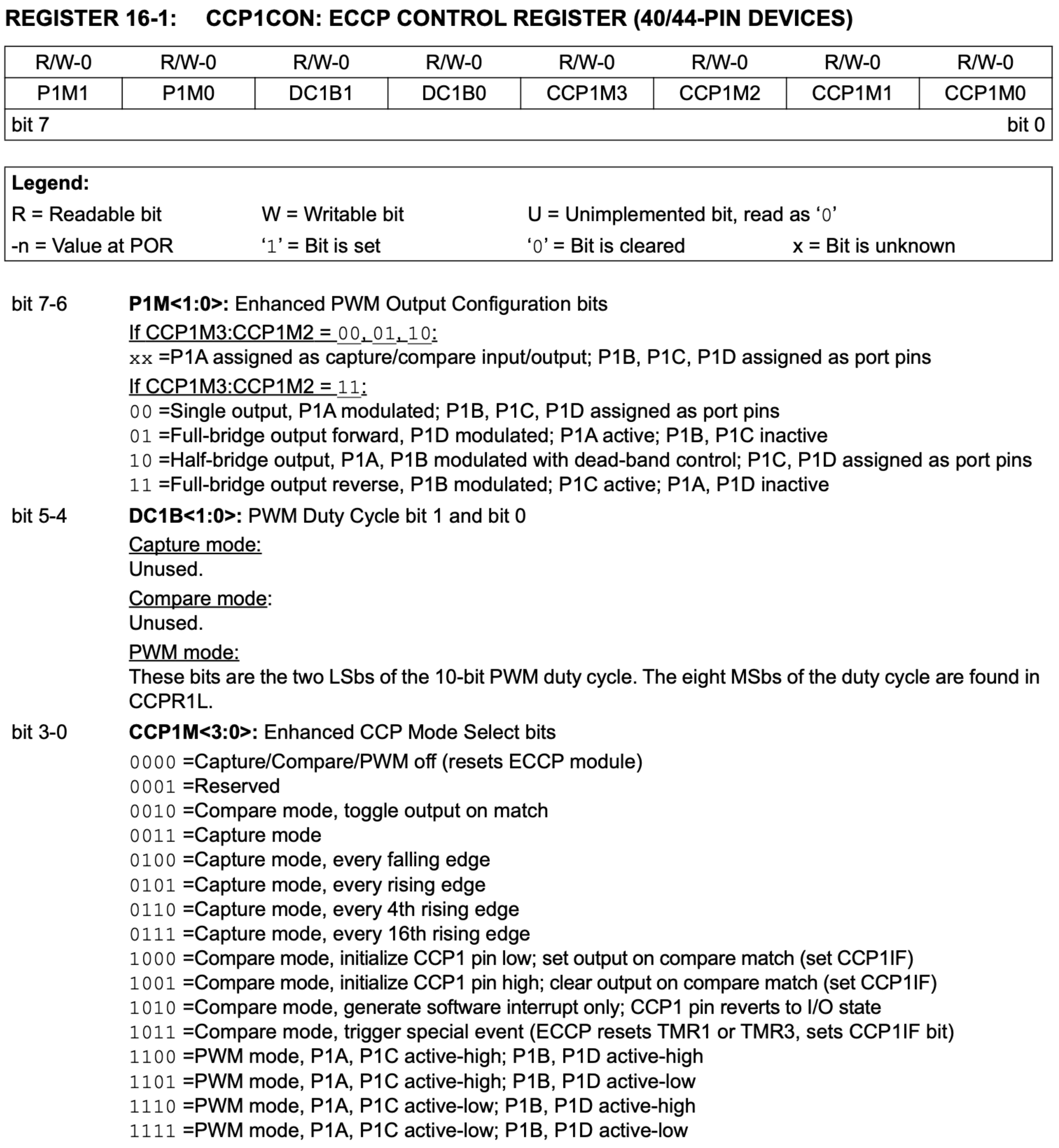
bit 5-4 **T3CKPS<1:0>**: Timer3 Input Clock Prescale Select bits
11 = 1:8 Prescale value
10 = 1:4 Prescale value
01 = 1:2 Prescale value
00 = 1:1 Prescale value

bit 2 **$\overline{\text{T3SYNC}}$:** Timer3 External Clock Input Synchronization Control bit
(Not usable if the device clock comes from Timer1/Timer3.)
When TMR3CS = 1:
1 = Do not synchronize external clock input
0 = Synchronize external clock input
When TMR3CS = 0:
This bit is ignored. Timer3 uses the internal clock when TMR3CS = 0.

bit 1 **TMR3CS:** Timer3 Clock Source Select bit
1 = External clock input from Timer1 oscillator or T13CKI (on the rising edge after the first falling edge)
0 = Internal clock (FOSC/4)

bit 0 **TMR3ON:** Timer3 On bit
1 = Enables Timer3
0 = Stops Timer3

Mode Selection



REGISTER 16-1: CCP1CON: ECCP CONTROL REGISTER (40/44-PIN DEVICES)

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
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 7-6 **P1M<1:0>:** Enhanced PWM Output Configuration bits

If CCP1M3:CCP1M2 = 00, 01, 10:

xx =P1A assigned as capture/compare input/output; P1B, P1C, P1D assigned as port pins

If CCP1M3:CCP1M2 = 11:

00 =Single output, P1A modulated; P1B, P1C, P1D assigned as port pins
01 =Full-bridge output forward, P1D modulated; P1A active; P1B, P1C inactive
10 =Half-bridge output, P1A, P1B modulated with dead-band control; P1C, P1D assigned as port pins
11 =Full-bridge output reverse, P1B modulated; P1C active; P1A, P1D inactive

bit 5-4 **DC1B<1:0>:** PWM Duty Cycle bit 1 and bit 0

Capture mode:
Unused.

Compare mode:
Unused.

PWM mode:
These bits are the two LSbs of the 10-bit PWM duty cycle. The eight MSbs of the duty cycle are found in CCPR1L.

bit 3-0 **CCP1M<3:0>:** Enhanced CCP Mode Select bits

0000 =Capture/Compare/PWM off (resets ECCP module)
0001 =Reserved
0010 =Compare mode, toggle output on match
0011 =Capture mode
0100 =Capture mode, every falling edge
0101 =Capture mode, every rising edge
0110 =Capture mode, every 4th rising edge
0111 =Capture mode, every 16th rising edge
1000 =Compare mode, initialize CCP1 pin low; set output on compare match (set CCP1IF)
1001 =Compare mode, initialize CCP1 pin high; clear output on compare match (set CCP1IF)
1010 =Compare mode, generate software interrupt only; CCP1 pin reverts to I/O state
1011 =Compare mode, trigger special event (ECCP resets TMR1 or TMR3, sets CCP1IF bit)
1100 =PWM mode, P1A, P1C active-high; P1B, P1D active-high
1101 =PWM mode, P1A, P1C active-high; P1B, P1D active-low
1110 =PWM mode, P1A, P1C active-low; P1B, P1D active-high
1111 =PWM mode, P1A, P1C active-low; P1B, P1D active-low

Capture mode

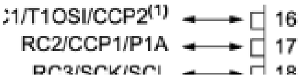
在 Capture mode 下，當指定的輸入事件發生時，Timer 的值會被複製並保存到 Capture Register 中。也就是說，我們可以藉此量測外部事件的到達時間。

CCP Pin Configuration

> In Capture mode, the appropriate `CCPx` pin needs to be configured as an **input** by setting the corresponding `TRIS` direction bit.

在 Capture mode 下，對應的 `CCPx` 腳位必須透過 **SET** 相應的 `TRIS` direction bit 來設定為 **Input**。

EX:　以CCP1為例，就要將RC2設為input



Event definition

> In Capture mode, the `CCPRxH:CCPRxL` register pair captures the 16-bit value of the `TMR1` or `TMR3` register when an event occurs on the corresponding `CCPx` pin.

在 Capture mode 下，當對應的 `CCPx` 腳位偵測到事件時， `CCPRxH:CCPRxL` 會擷取 `TMR1` 或 `TMR3` 的 16-bit value。

其中，「事件」被定義為下列其中一種：

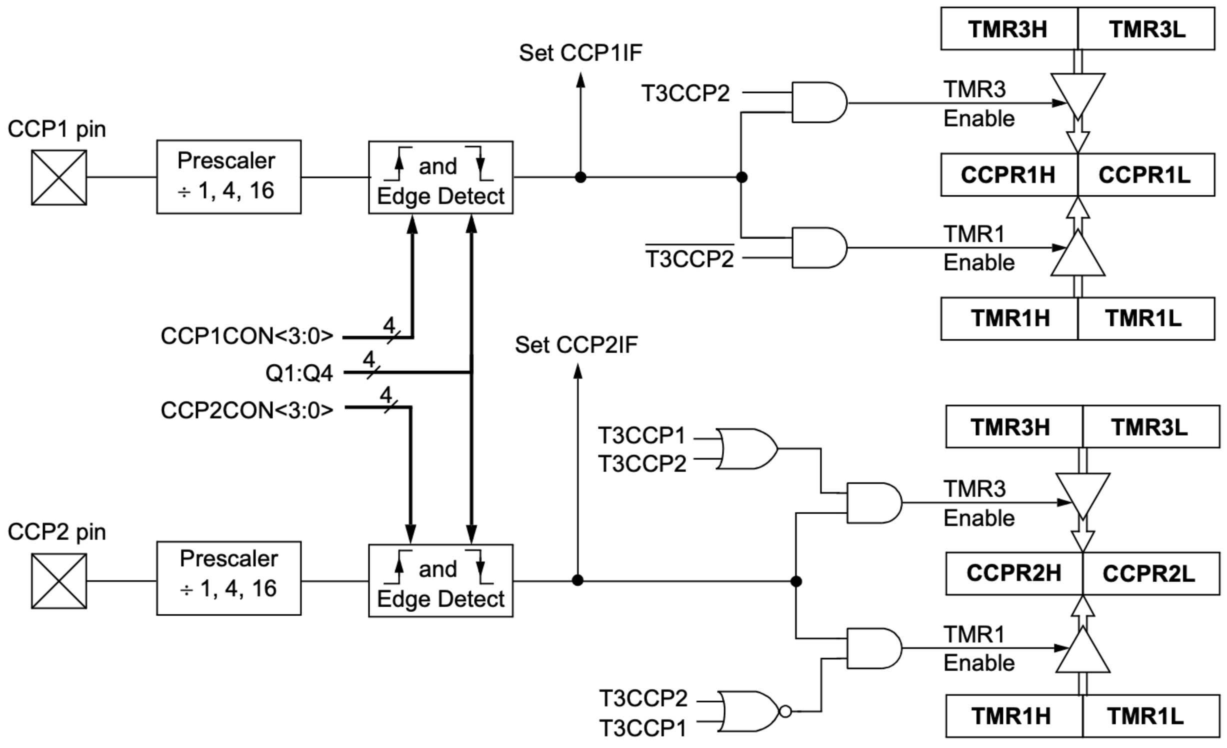
- 每一次 Falling Edge
- 每一次 Rising Edge
- 每第 4 次 Rising Edge
- 每第 16 次 Rising Edge

Interrupt

> When a capture is made, the interrupt request flag bit, `CCPxIF`, is set; it must be cleared in software. If another capture occurs before the value in register `CCPRx` is read, the old captured value is overwritten by the new captured value.

當發生一次 Capture 時，Interrupt Request Flag Bit `CCPxIF` 會被設為 1；這個 Flag Bit 必須在程式碼中清除。若在讀取 `CCPRx` 的值之前又有新的 Capture 發生，先前捕獲的值將被新值覆寫。

Capture Mode Operation Block Diagram



- `T3CON` 用來設定 `TMR1` 或 `TMR3` ，以作為 CCP modules 的 Time Base。
- 當在 `CCPx` 腳位偵測到 Rising 或 Falling Edge 時， `CCPxIF` 會被設為 1。
- 當 Interrupt 發生時，目前 Timer 的值會被複製到 `CCPRx` （ `CCPRxH:CCPRxL` ）中。

Compare mode

在 Compare mode 下，當 Timer 的值與 `CCPRx` 中設定的比較值相等時，會觸發一個事件。依照設定，這個事件可以讓 `CCPx` 腳位改變狀態（例如設為高電位、低電位或翻轉），或產生一個 Interrupt。也就是說，我們可以利用 Compare mode 在特定時間點產生控制訊號或中斷。

CCP Pin Configuration

> The user must configure the `CCPx` pin as an **output** by clearing the appropriate `TRIS` bit.

在 Compare mode 下，對應的 `CCPx` 腳位必須透過 **CLEAR** 相應的 `TRIS` direction bit 來設定為 **Output**。

Compare Mode Operation

> In Compare mode, the 16-bit `CCPRx` register value is constantly compared against either the `TMR1` or `TMR3` register pair value.

在 Compare mode 下，16-bit 的 `CCPRx` 內容會不斷與 `TMR1` 或 `TMR3` 的 Register Pair 值進行比較。

當發生 match 時， `CCPx` 腳位可以被設定為：

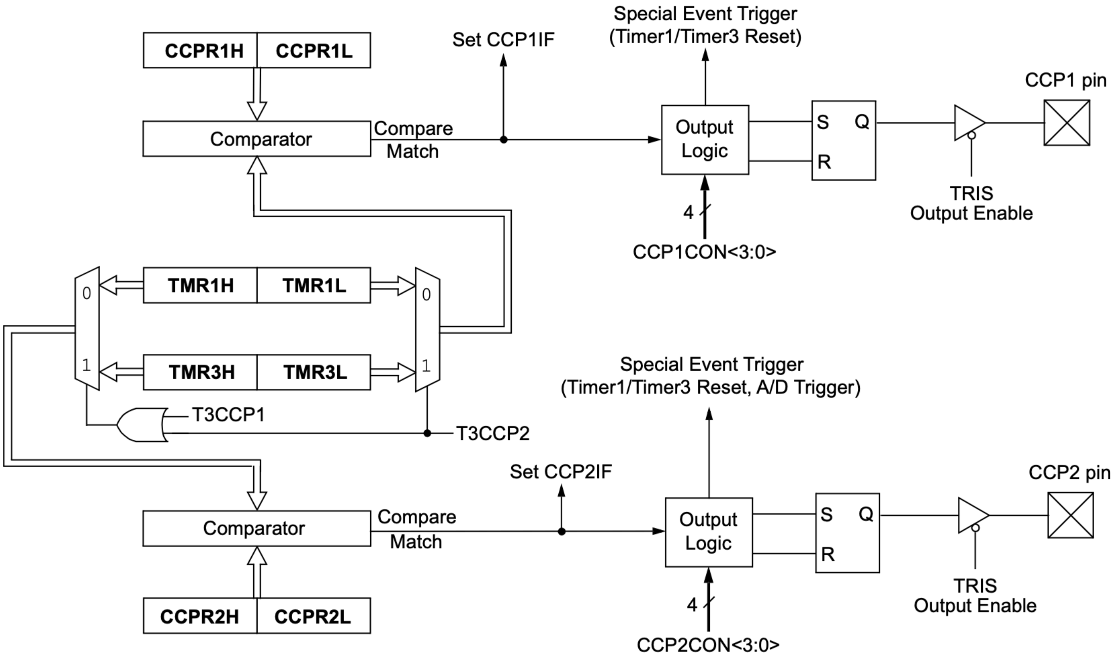
- Driven High（輸出高電位）
- Driven Low（輸出低電位）
- Toggled（在 High 與 Low 之間切換）
- Remain Unchanged（保持不變，也就是維持 I/O latch 的狀態）

Interrupt

> The action on the pin is based on the value of the mode select bits (`CCPxM<3:0>`). At the same time, the interrupt flag bit, `CCPxIF` , is set.

腳位的動作取決於 Mode Select Bits `CCPxM<3:0>` 的設定。於此同時，Interrupt Flag Bit `CCPxIF` 會被設為 1；這個 Flag Bit 必須在程式碼中清除。

Compare Mode Operation Block Diagram

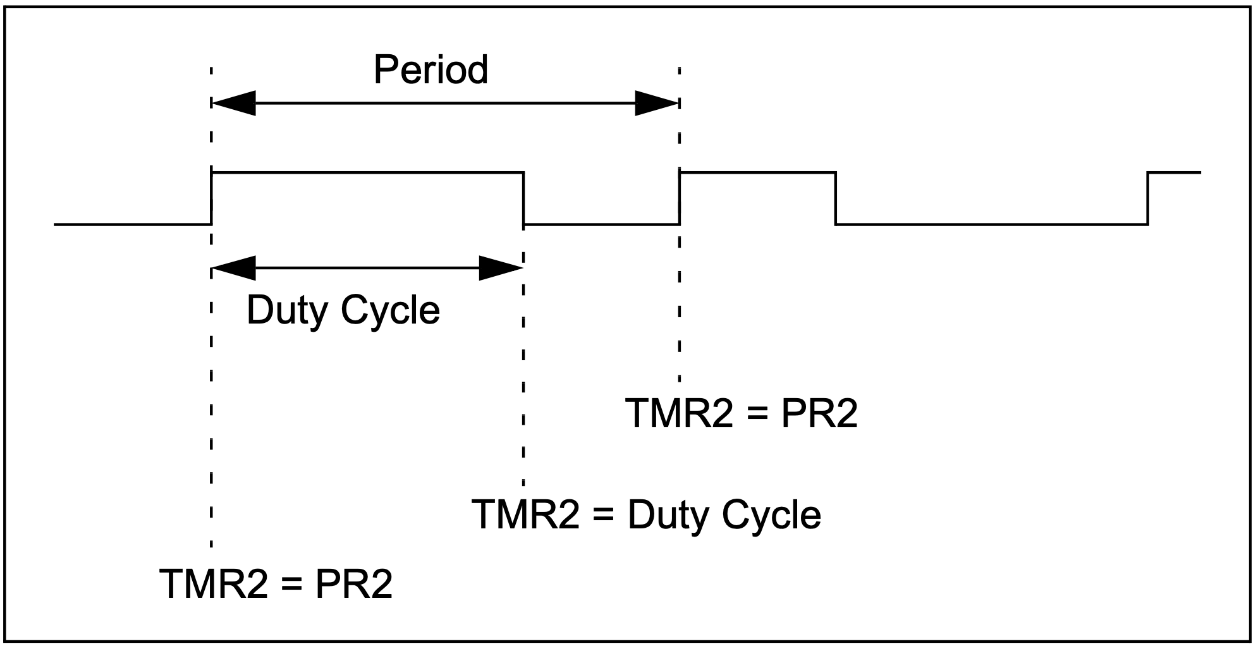


- `T3CON` 用來設定 `TMR1` 或 `TMR3` ，以作為 CCP modules 的 Time Base。
- `CCPRx` 的值會持續與 Timer 的值進行比較。
- 當兩者相等時， `CCPxIF` 會被設為 1，並且 `CCPx` 腳位的輸出狀態會依照設定而改變。

PWM mode

PWM（Pulse-Width Modulation）是一種透過改變 Duty Cycle 來控制輸出平均電壓的方式。輸出的方波訊號，其平均電壓由 Duty Cycle 與週期的比例決定。PWM 廣泛應用於各種領域，例如呼吸燈、伺服馬達控制等。

FIGURE 15-4: PWM OUTPUT



PWM Period

PWM 週期（PWM Period）是透過寫入 `PR2` Register 來設定的。其計算公式如下：

$$\text{PWM Period} = \left[(\text{PR2})+1\right] \times 4 \times T_{osc} \times (\text{TMR2 Prescale Value})$$

PWM 頻率（PWM Frequency）的定義為：

$$\text{PWM Frequency} = \frac{1}{[\,\text{PWM Period}\,]}$$

當 `TMR2` 的值等於 `PR2` 時，在下一個 Cycle 會發生以下三件事：

- `TMR2` 被清除
- `CCPx` 腳位被設為 High
  例外：如果 PWM duty cycle = 0%， `CCPx` 腳位不會被設為 High
- PWM Duty Cycle 的值會從 `CCPRxL` latch 到 `CCPRxH`

NOTE:

> The Timer2 postscalers are not used in the determination of the PWM frequency. The postscaler could be used to have a servo update rate at a different frequency than the PWM output.

Timer2 的 Postscalers 不會影響 PWM Frequency 的計算。
不過，Postscaler 可以用來讓伺服馬達的更新速率與 PWM 輸出頻率不同。

PWM Duty Cycle

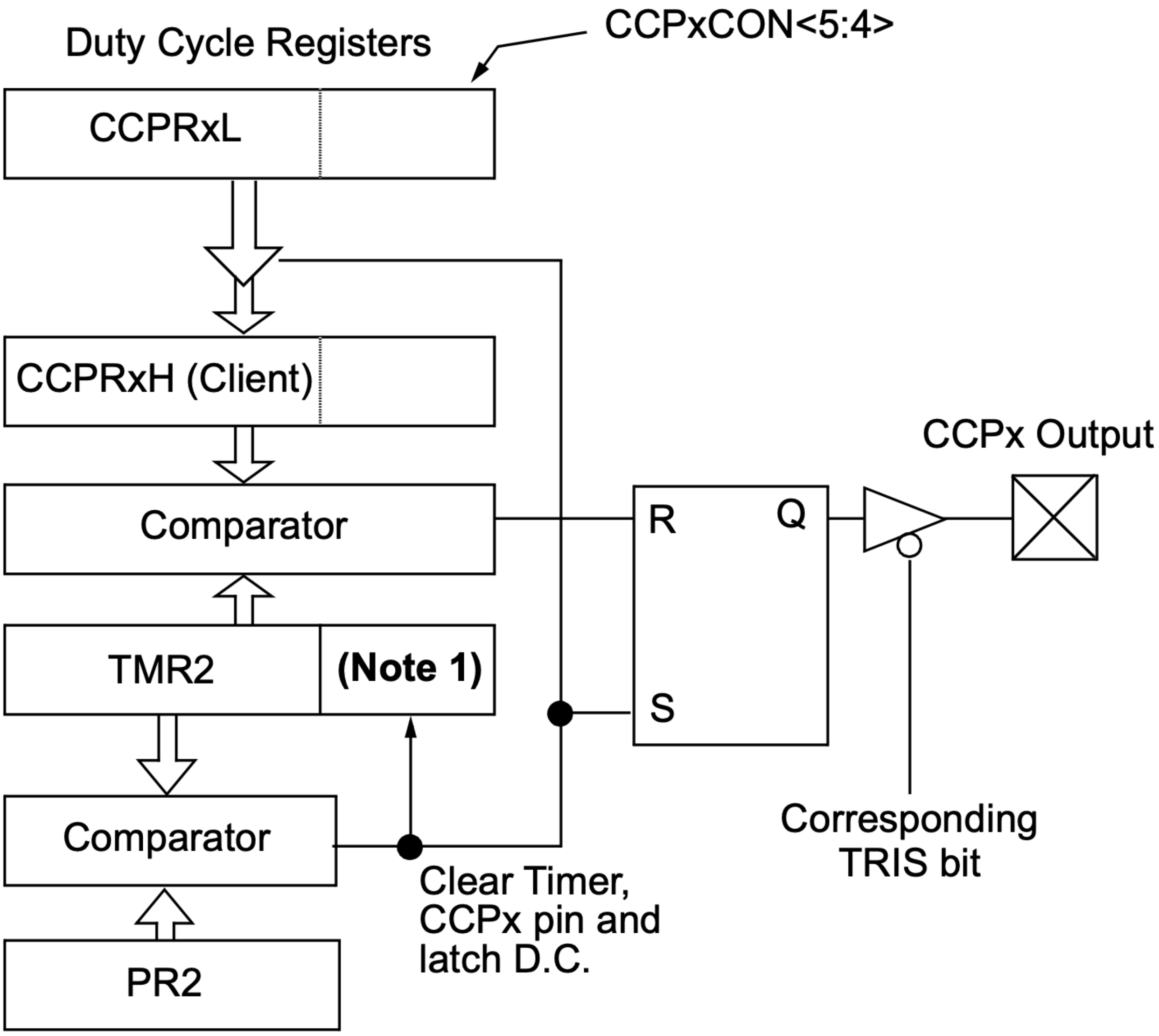
PWM Duty Cycle 是透過寫入 `CCPRxL` Register 以及 `CCPxCON<5:4>` bits 來設定的，最高可達 10-bit Resolution。其中， `CCPRxL` 儲存高 8 個 MSB，而 `CCPxCON<5:4>` bits 則儲存低 2 個 LSB。
這個 10-bit 值以 `CCPRxL:CCPxCON<5:4>` 的形式表示。

計算 PWM Duty Cycle 的公式如下：

$$\text{PWM Duty Cycle} = (\text{CCPRxL} : \text{CCPxCON} < 5:4 >) \times T_{osc} \times (\text{TMR2 Prescale Value})$$

`CCPRxL` 和 `CCPxCON<5:4>` 可以在任何時間寫入，但 Duty Cycle 的值不會立即生效，必須等到 `PR2` 與 `TMR2` 發生 match（也就是一個週期結束）後，才會被 latch 到 `CCPRxH` 。
在 PWM mode 下， `CCPRxH` 是一個 Read-Only Register。

Simplified PWM Block Diagram



Note 1: The 8-bit TMR2 value is concatenated with the 2-bit internal Q clock, or 2 bits of the prescaler, to create the 10-bit time base.

1. **Initialization**

- Load the period value (in `PR2`) and duty cycle (in `CCPRxL` and `CCPxCON<5:4>`).
- Initiate the `CCPx` pin as an **Output**.
- Configure `T2CON` and `TMR2`, set `TMR2` to 0, and start Timer2.
- Set the output to **High**.
  Exception: if PWM duty cycle = 0%, the `CCPx` pin will not be set.

2. **During the PWM cycle**

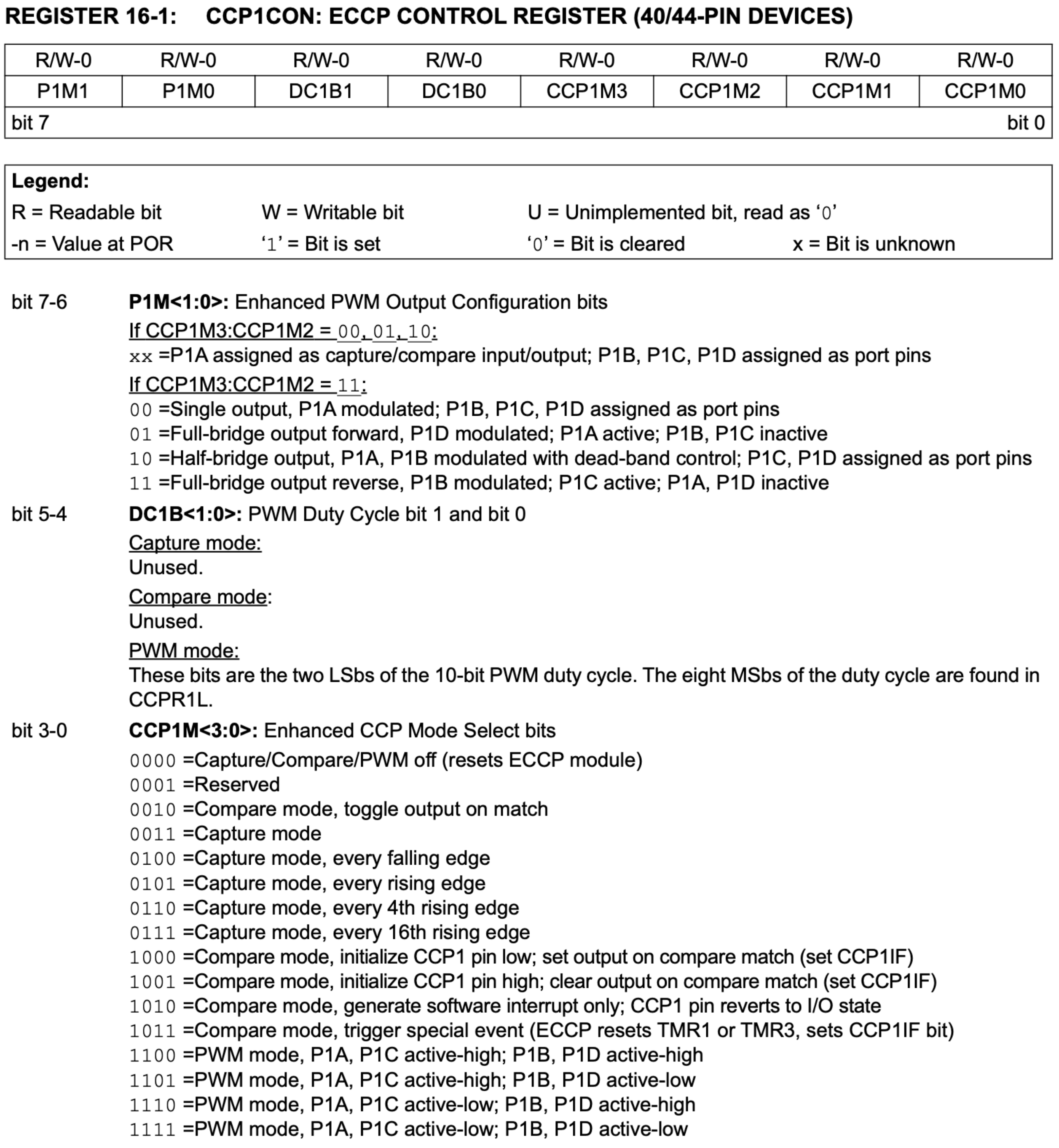
- `TMR2` is compared with `CCPRxH`.
- When they are the same, the output pin goes **Low**.

3. **End of the PWM cycle**
   When `TMR2 = PR2`:
   - `TMR2` is cleared.
   - The duty cycle value from `CCPRxL` and `CCPxCON<5:4>` is **latched into `CCPRxH`**.
   - The `CCPx` pin is set **High** (if duty cycle ≠ 0%).
   - A new PWM cycle begins.

Setup for PWM Operation

> The following steps need to be taken when configuring the CCP module for PWM operation:
>
> 1. Set the PWM period by writing to the `PR2` register.
> 2. Set the PWM duty cycle by writing to the `CCPRxL` register and `CCPxCON<5:4>` bits.
> 3. Make the CCPx pin an output by clearing the appropriate `TRIS` bit.
> 4. Set the `TMR2` prescale value, then enable Timer2 by writing to `T2CON`.
> 5. Configure the `CCPx` module for PWM operation.

1. 透過寫入 `PR2` Register 來設定 PWM Period。
2. 透過寫入 `CCPRxL` Register 與 `CCPxCON<5:4>` bits 來設定 PWM Duty Cycle。
3. 將 `CCPx` 腳位設為 Output，方法是清除對應的 `TRIS` bit。
4. 設定 `TMR2` 的 Prescale Value，然後透過寫入 `T2CON` 來啟用 Timer2。
5. 將 `CCPx` Module 設定為 PWM Operation。



REGISTER 16-1: CCP1CON: ECCP CONTROL REGISTER (40/44-PIN DEVICES)

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
bit 7							bit 0

Legend:

R = Readable bit W = Writable bit U = Unimplemented bit, read as ‘0’

-n = Value at POR ‘1’ = Bit is set ‘0’ = Bit is cleared x = Bit is unknown

bit 7-6 **P1M<1:0>:** Enhanced PWM Output Configuration bits

If CCP1M3:CCP1M2 = 00, 01, 10:
xx =P1A assigned as capture/compare input/output; P1B, P1C, P1D assigned as port pins

If CCP1M3:CCP1M2 = 11:
00 =Single output, P1A modulated; P1B, P1C, P1D assigned as port pins
01 =Full-bridge output forward, P1D modulated; P1A active; P1B, P1C inactive
10 =Half-bridge output, P1A, P1B modulated with dead-band control; P1C, P1D assigned as port pins
11 =Full-bridge output reverse, P1B modulated; P1C active; P1A, P1D inactive

bit 5-4 **DC1B<1:0>:** PWM Duty Cycle bit 1 and bit 0

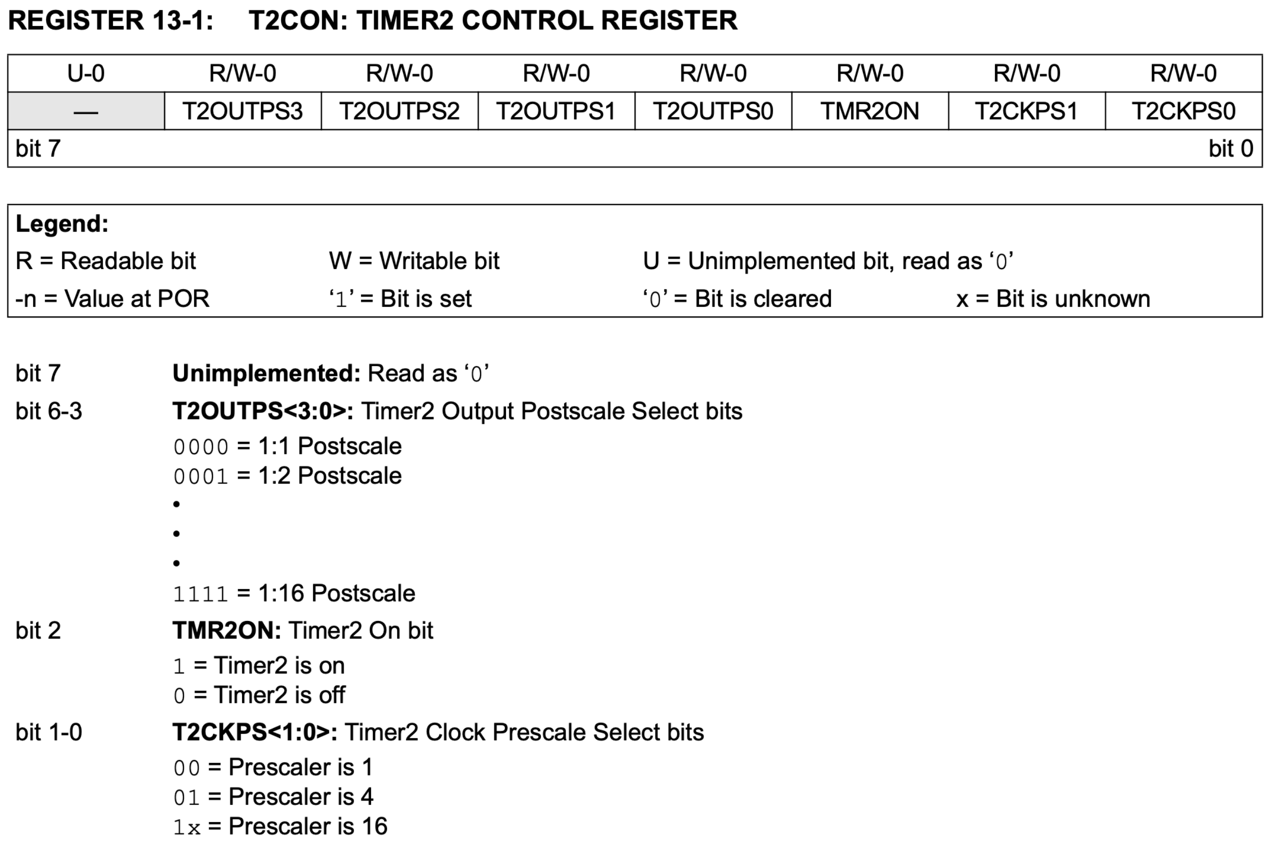
Capture mode:
Unused.

Compare mode:
Unused.

PWM mode:
These bits are the two LSbs of the 10-bit PWM duty cycle. The eight MSbs of the duty cycle are found in CCPR1L.

bit 3-0 **CCP1M<3:0>:** Enhanced CCP Mode Select bits

0000 =Capture/Compare/PWM off (resets ECCP module)
0001 =Reserved
0010 =Compare mode, toggle output on match
0011 =Capture mode
0100 =Capture mode, every falling edge
0101 =Capture mode, every rising edge
0110 =Capture mode, every 4th rising edge
0111 =Capture mode, every 16th rising edge
1000 =Compare mode, initialize CCP1 pin low; set output on compare match (set CCP1IF)
1001 =Compare mode, initialize CCP1 pin high; clear output on compare match (set CCP1IF)
1010 =Compare mode, generate software interrupt only; CCP1 pin reverts to I/O state
1011 =Compare mode, trigger special event (ECCP resets TMR1 or TMR3, sets CCP1IF bit)
1100 =PWM mode, P1A, P1C active-high; P1B, P1D active-high
1101 =PWM mode, P1A, P1C active-high; P1B, P1D active-low
1110 =PWM mode, P1A, P1C active-low; P1B, P1D active-high
1111 =PWM mode, P1A, P1C active-low; P1B, P1D active-low



REGISTER 13-1: T2CON: TIMER2 CONTROL REGISTER

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	T2OUTPS3	T2OUTPS2	T2OUTPS1	T2OUTPS0	TMR2ON	T2CKPS1	T2CKPS0
bit 7							bit 0

Legend:

R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 7 **Unimplemented:** Read as ‘0’

bit 6-3 **T2OUTPS<3:0>:** Timer2 Output Postscale Select bits
0000 = 1:1 Postscale
0001 = 1:2 Postscale
•
•
•
1111 = 1:16 Postscale

bit 2 **TMR2ON:** Timer2 On bit
1 = Timer2 is on
0 = Timer2 is off

bit 1-0 **T2CKPS<1:0>:** Timer2 Clock Prescale Select bits
00 = Prescaler is 1
01 = Prescaler is 4
1x = Prescaler is 16

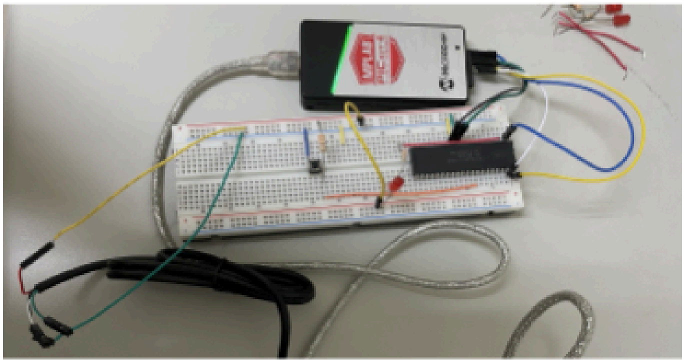
Tower Pro SG90 specs

Specification	Value
Modulation	Analog
Torque	4.8 V
Speed	0.12 sec/60°
Motor Type	3-pole
Gear Type	Plastic
Rotation/Support	Bushing
Pulse Width	500 ~ 2400 μs (-90° ~ 90°, 1450 μs → 0°)
Connector Type	JR

注意：馬達很脆弱，請小心使用！請參考下面說明

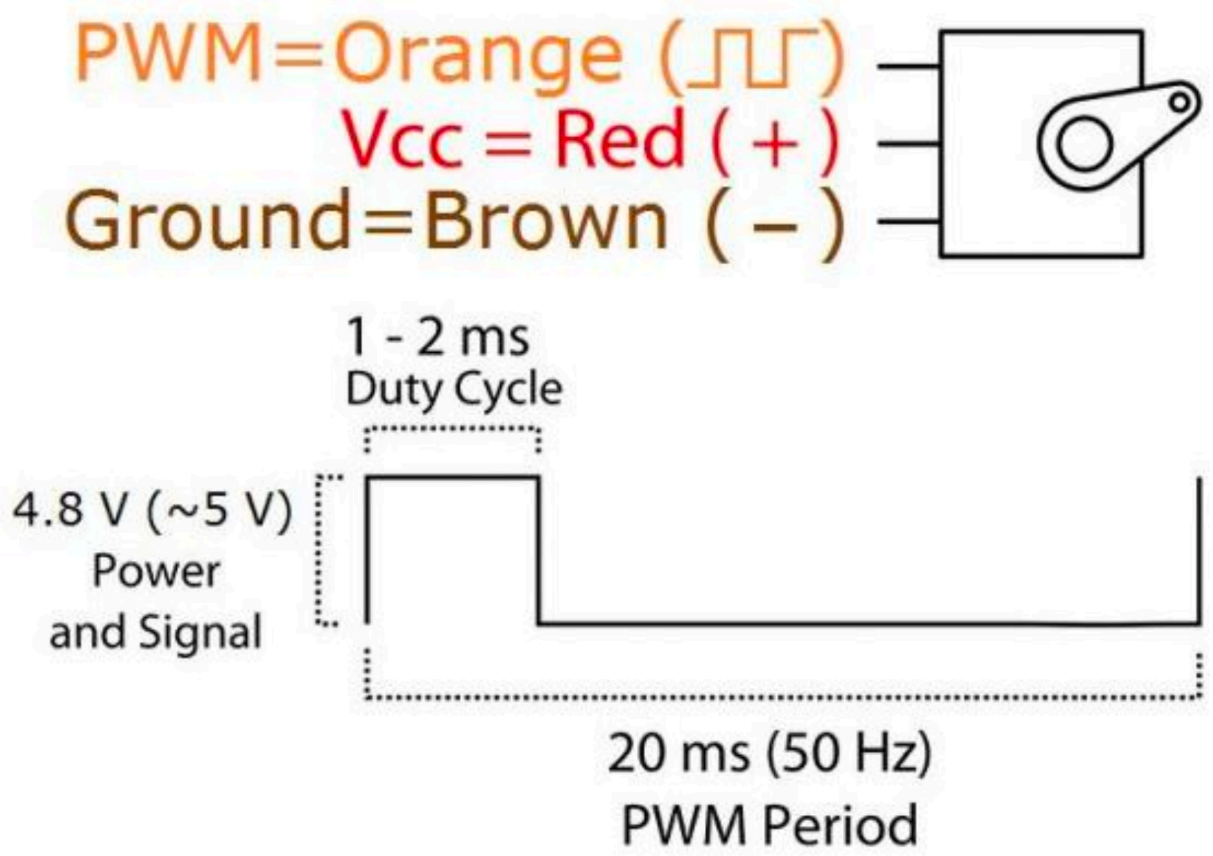
馬達

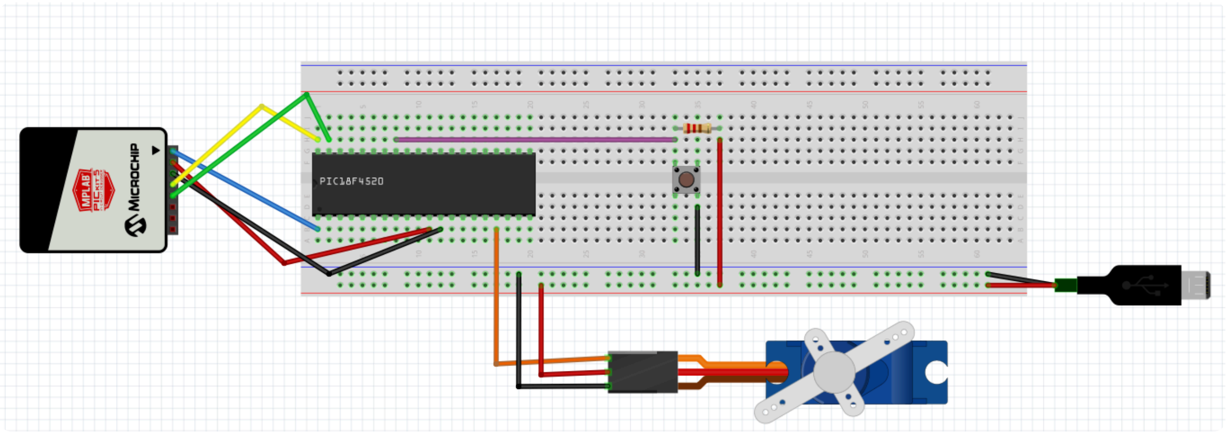
- 不要用手扳動轉軸，SG90很容易被轉壞，轉壞就是直接重買一個QQ
- 請確認接線沒問題再接電源，馬達非常容易燒壞
- 如果還是不行
  - 嘗試利用TTL線供電，記得要把properties的5V設定關掉



- 若仍無法解決代表馬達可能已經燒壞，下列幾個型號供參考
  - SG90： 最便宜、但齒輪是塑膠容易燒壞（蝦皮買大約40~50元、電子材料行大約70~80元）
  - MG90S：價格較SG90貴，但齒輪是金屬的較耐用（蝦皮買大約70~80元、電子材料行大約140~150元）
  - MG995：扭力很大、最耐用，但價格最貴、所需電壓略高，可能不太適合筆電有供電不穩問題的同學，程式碼在delay長度上可能與SG90不太一樣（蝦皮買大約140~150元、電子材料行大約200多元）
  - 價格非絕對，只是提供給同學一個參考，站前有很多電子材料行可供急用的同學購買，務必買180度可控制角度的伺服馬達，360度的程式碼在設計上會跟180度的不同
  - 不同馬達在角度上設置可能會有些微差異，若設定不對可能會卡住（例如SG90在90度的設置是0x14，但MG90S會是0x13），再請大家自行嘗試
- 上機考建議帶兩個馬達以備不時之需，因為如果壞掉你就沒分數了

- **接線圖**





Sample Code

> The following code will rotate the servomotor into 0°

```
// PIC18F4520 Configuration Bit Settings
// 'C' source line config statements

#pragma config OSC = INTIO67    // Oscillator Selection bits (Internal osci
#pragma config PWRT = OFF       // Power-up Timer Enable bit (PWRT disablec
#pragma config BOREN = ON       // Brown-out Reset Enable bits (Brown-out F
#pragma config WDT = OFF        // Watchdog Timer Enable bit (WDT disabled
#pragma config PBADEN = OFF     // PORTB A/D Enable bit (PORTB<4:0> pins ar
#pragma config LVP = OFF        // Single-Supply ICSP Enable bit (Single-Su
#pragma config CPD = OFF        // Data EEPROM Code Protection bit (Data EE

// #pragma config statements should precede project file includes.
// Use project enums instead of #define for ON and OFF.

#include <xc.h>
#include <pic18f4520.h>

void main(void){
    // Timer2 -> On, prescaler -> 4
    T2CONbits.TMR2ON = 0b1;
    T2CONbits.T2CKPS = 0b01;

    // Internal Oscillator Frequency, Fosc = 125 kHz, Tosc = 8 µs
    OSCCONbits.IRCF = 0b001;

    // PWM mode, P1A, P1C active-high; P1B, P1D active-high
    CCP1CONbits.CCP1M = 0b1100;

    // CCP1/RC2 -> Output
    TRISC = 0;
    LATC = 0;

    // Set up PR2, CCP to decide PWM period and Duty Cycle
    /*
     * PWM period
     * = (PR2 + 1) * 4 * Tosc * (TMR2 prescaler)
     * = (0x9B + 1) * 4 * 8µs * 4
     * = 0.019968s ~= 20ms
     */
    PR2 = 0x9B;

    /*
     * Duty cycle
     * = (CCPR1L:CCP1CON<5:4>) * Tosc * (TMR2 prescaler)
     * = (0x0B*4 + 0b01) * 8µs * 4
     * = 0.00144s ~= 1450µs
     */
    CCPR1L = 0x0B;
    CCP1CONbits.DC1B = 0b01;

    while(1);
    return;
}
```

> For additional implementation details,
> please refer to the MPLAB XC8 C Compiler User’s Guide for PIC MCU (https://ww1.microchip.com/downloads/aemDocuments/documents/DEV/ProductDocuments/UserGuides/MPLAB-XC8-C-Compiler-Users-Guide-for-PIC-DS50002737.pdf).
>
> In particular, see:
> <5.9> Interrupts
> <9.2> `<xc.h>` Device-specific Functions
>
> Reading these sections will provide you with a great deal of useful information and help you work smoothly during the lab sessions.