



MICROCHIP

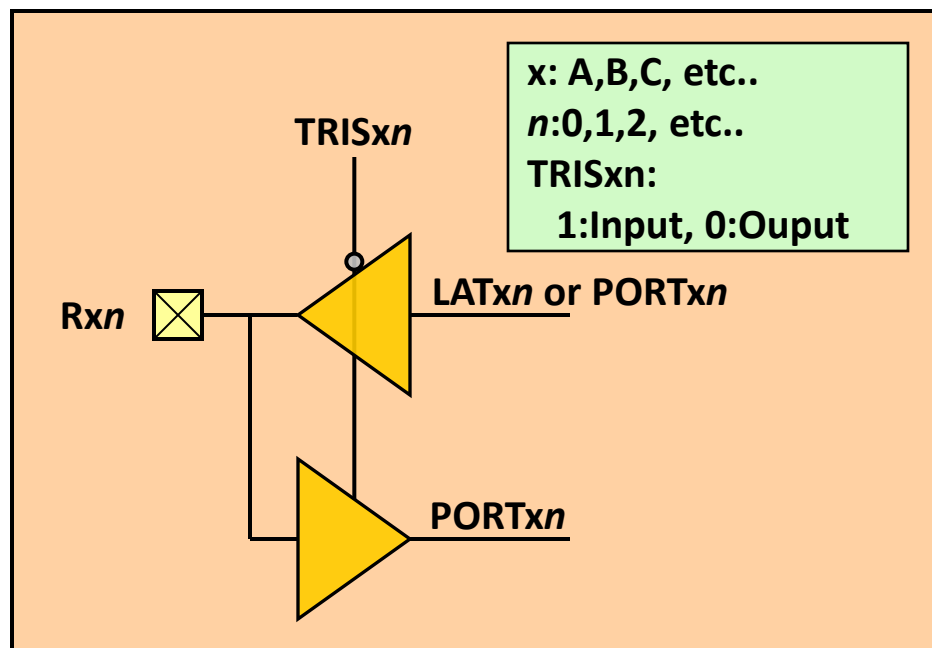
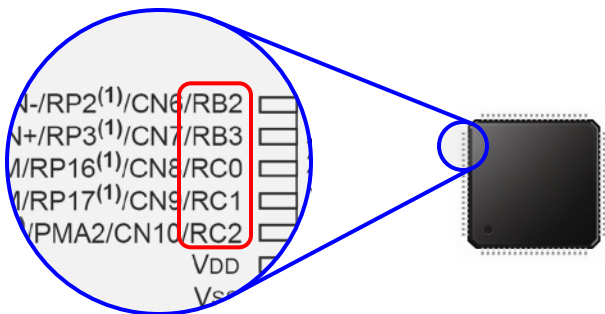
Regional Training Centers

Section 4

IO Port Architecture

I/O Port Block Diagram

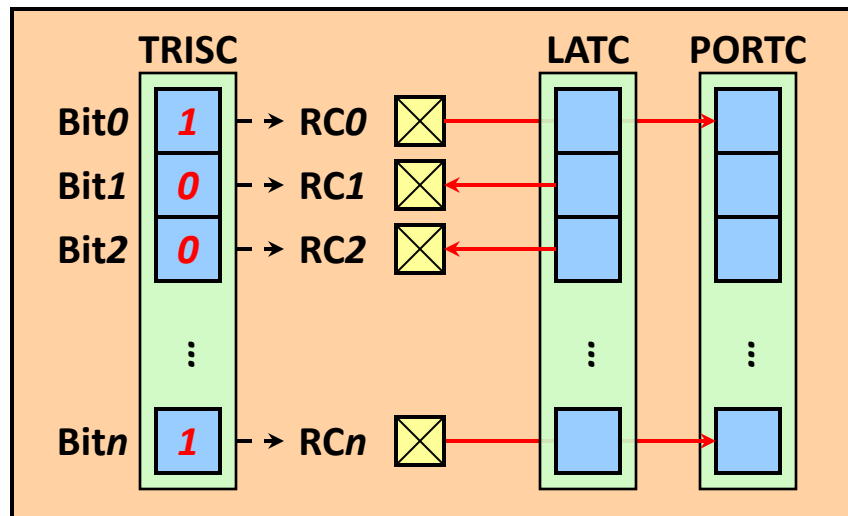
- 16-Bits MCU的IO Port示意圖, 如圖所示。
- 所有的IO Port都有TRIS, LAT跟PORT特殊功能暫存器。TRIS用來設定IO的方向, 要輸出時(Ex: 控制LED) $TRIS_{xn}$ 必須設為"0"。要輸入(Ex: 讀取按鍵狀態)時 $TRIS_{xn}$ 必須設為"1"。
- 設為輸出時, 要輸出的狀態填入 LAT_{xn} 或 $PORT_{xn}$, 對應的接腳 R_{xn} 就會有輸出。
- 設為輸入時, 可以讀取 $PORT_{xn}$ 取得外部的實際狀態。



I/O Port 的操作

- TRIS, LAT跟PORT特殊功能暫存器都是一個16-Bits的暫存器,暫存器中每個Bit都控制著對應的IO接腳。
- 以IO Port C為例,TRISC, LATC, PORTC的Bit 0,控制RC0接腳。
- 要用RC0控制LED時,必須先把TRISC的Bit 0設為0(輸出),然後把要輸出的狀態,填入LATC的Bit 0。
- 要用RC0讀取按鍵狀態時,則必須把TRISC的Bit 0設為1(輸入),然後就可以從PORTC的Bit 0取得外部的狀態。

- 為何要分成LAT跟PORT ?



PORT 與 LAT

- PORT 與 LAT 使用上的不同
設為輸出時,PORT 與LAT 的功能完全相同。
設為輸入時,PORT 為外部接腳的狀態,LAT則為內部D-Type 正反器的狀態。 (參考I/O Port Block Diagram)
- 輸出時,如果使用PORT來讀取狀態,避免對相同的PORT做連續讀寫的動作,以避免 Read-Modify-Write 的問題發生,(通常可以在兩指令之間插入NOP指令,讓IO有足夠時間轉態)。
- 但強烈建議, 輸出使用時, 以LAT來取代PORT的操作！

Lab2 - BasicIO

- 在Lab1的程式基礎上,嘗試加入IO Port的控制程式。
- 請嘗試控制TRIS xn , LAT xn , PORT xn ,來讓RD0接腳可以不斷的轉態(Toggle, 1->0->1->...)。每次Toggle的時間間隔設為500 mS。
- 記住前面所提過的,要操作SFR要先含入(include)MCU的標頭檔。
Ex:#include <p24fxxxx.h>
- 操作SFR時,可以直接操作整個暫存器,
Ex:TRISD &= 0xfffe; // Port D Bit0 設為輸出,
LATD |= 0x0001; // Port D Bit0 輸出High Level。
- 或使用MCU的標頭檔已經定義好的結構,來存取單一Bit,
Ex:TRISDbits.TRISD0 = 0;
LATDbits.LATD0 = 1;

Lab2 – Basic IO Step1

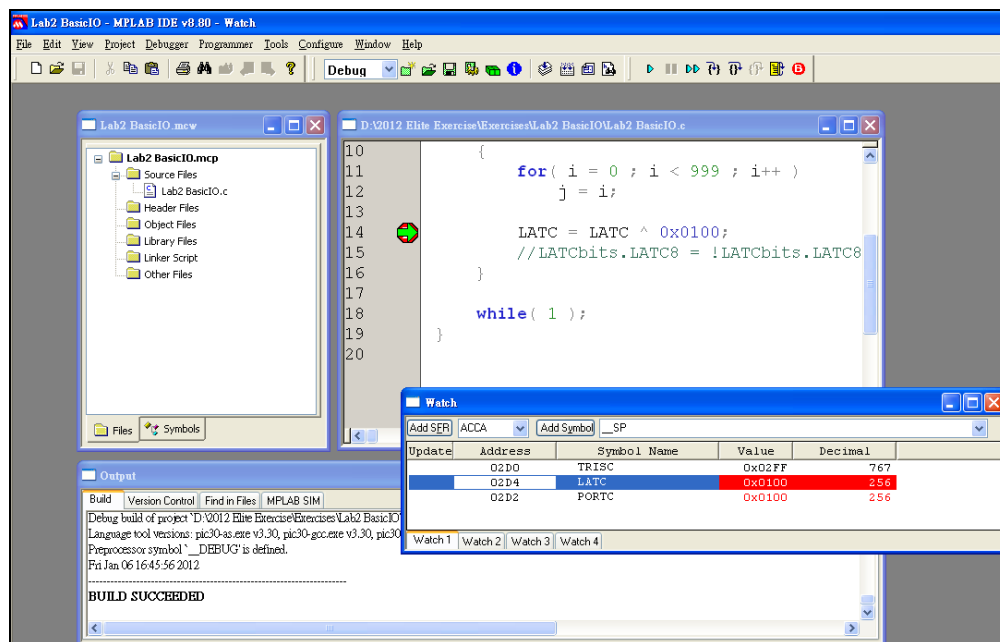
- 程式要如何設計才能達到Toggle的功能？

LATD = LATD ^ 0x0001; // 利用XOR

LATDbits.LATD0 = !LATDbits.LATD0; // 利用NOT

- 如何得知IO的狀態變化？

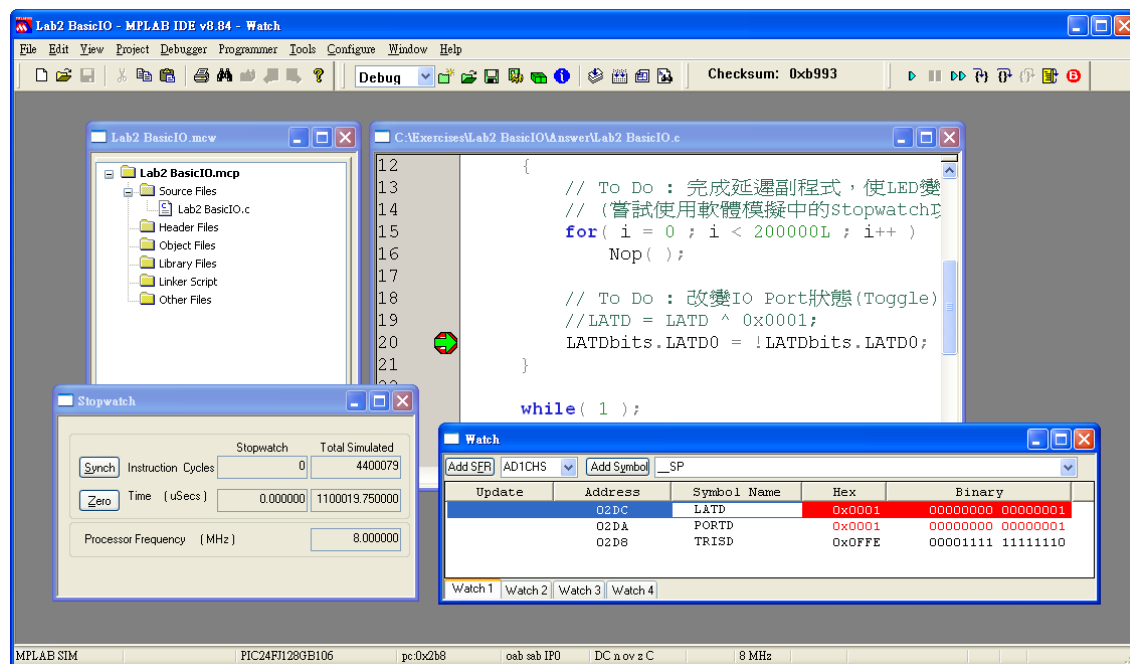
善用MPLAB SIM與Watch
Windows觀察程式的變化。



Lab2 – Basic IO Step2

- 如何控制IO每次Toggle的間隔時間？

控制for迴圈來決定Delay的時間。可以利用Stopwatch來測量迴圈的時間,計算前必須先設定預計的系統時脈。此處先設定為(Fosc)8MHz。



Analog or Digital Mode

- 有些 IO接腳(RX n)的功能跟類比輸入(AN n)共用的。但MCU在 Reset/Power On後的預設值為Analog mode,無法做為數位輸出入使用。所以如果碰到這類接腳,在使用時必須先改為Digital Mode。
- PIC24FJ256GB106透過ADxPCFGL/ADxPCFGH來設定。設1為 Digital mode,0為Analog mode。

