

# 硬體裝修乙級-IC 控制原理介紹

文章撰寫:許展維

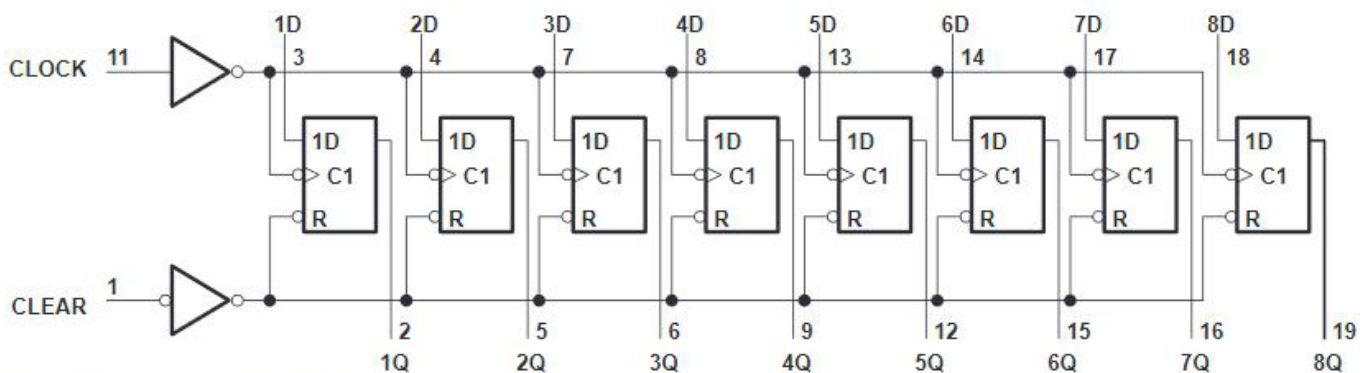
這是一篇簡易的介紹文章，有誤請指證。

## IC 簡易介紹:

### IC 74273 :控制紅色 LED

8 組 D 型正反器、1 個 Clock 與 1 個 Clear 端組合成的「並入/並出正反器」，能夠儲存 8 位元的資料 (正緣觸發)。

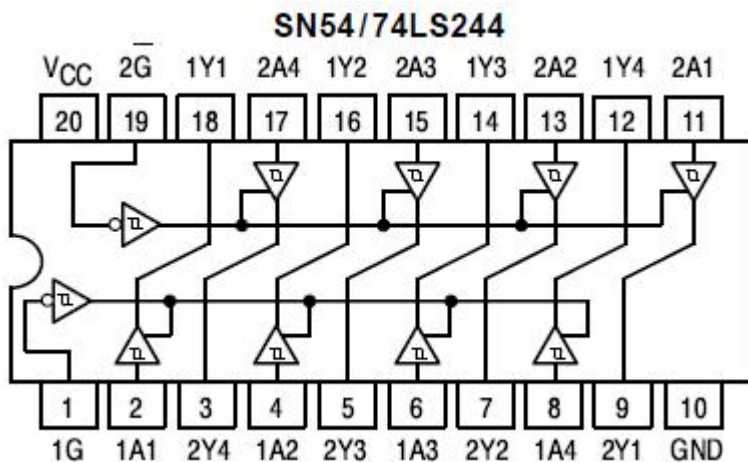
#### logic diagram (positive logic)



由於此暫存器為正緣觸發，只有在正緣觸發時才會將輸入端(1D~8D)的資料並出給輸出端(1Q~8Q)，電路中紅色 LED 就是接在輸出端的位置，所以要讓紅色 LED 的第 1 顆(1Q)與第 3 顆(3Q)亮燈時，必須在 1D~8D 送出  $1010\ 0000_{(2)}$  的資料，這時輸入端已經為  $1010\ 0000_{(2)}$  的資料了，接下來只要觸發一個正緣訊號即可將輸入端的資料送往輸出端，並暫時存下這組資料，讓紅色 LED 依照指令亮燈。

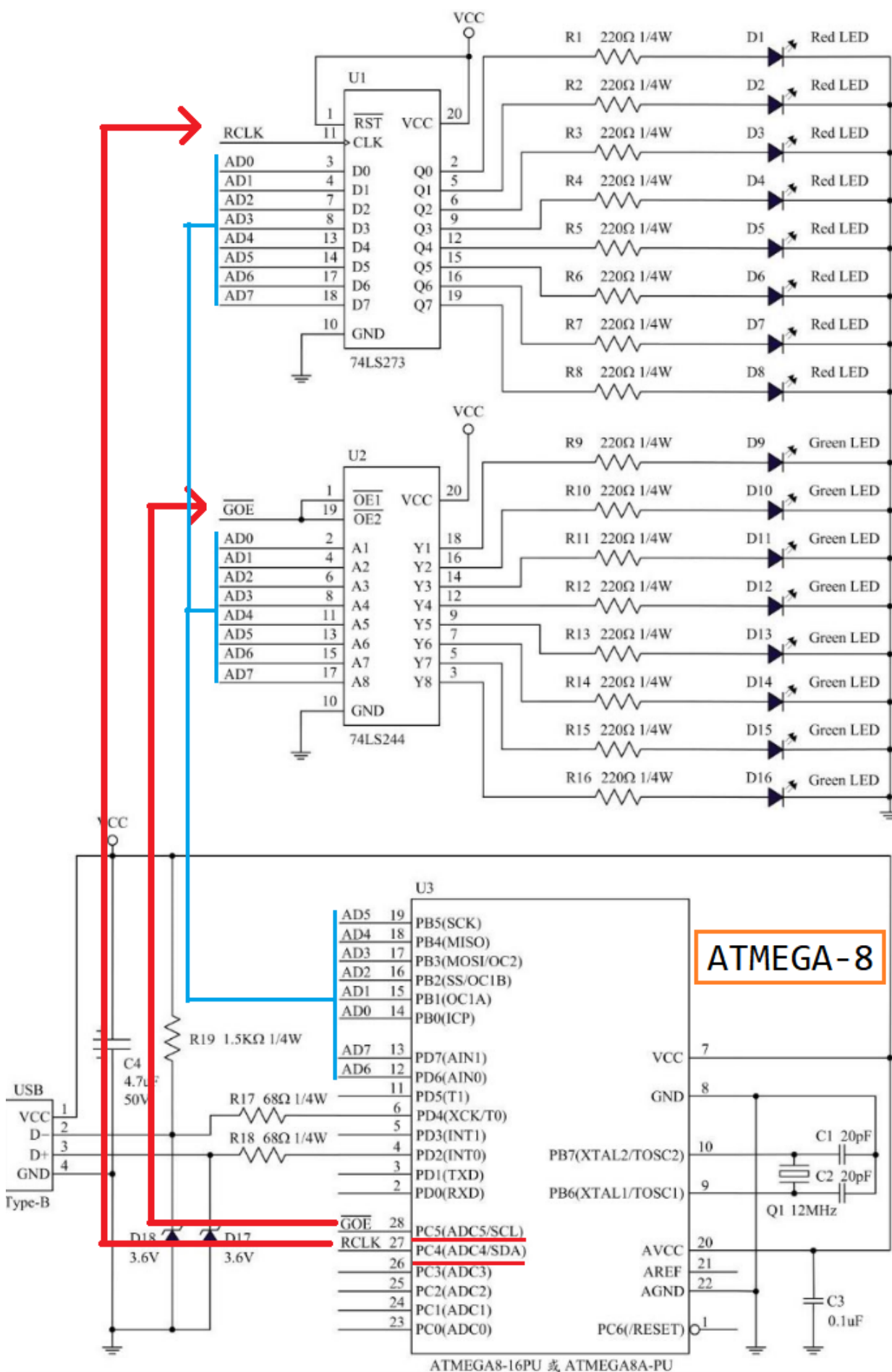
### IC 74244 :控制綠色 LED

8 個三態閘與 2 個致能端組合成的控制 IC。



雖然三態閘有 0、1 和高組態，但這邊都只是輸入直接送輸出，只是用致能腳控制要不要輸出這組資料而已，簡單來說就是用 EN 腳控制這組綠色 LED 要不要亮，當 1G、2G 都為 0 時啟用 IC，為 1 時停用 IC。

請先詳閱電路圖：



## 關於控制方式 OutDataCtrl(Data, Control) :

在這裡設定 **Data** 為 AD0~AD7 的輸出，**Control** 為 PC0~PC6 的輸出。

從上面電路圖可以看到，74273 和 74244 的輸入端都是接到 ATMEGA-8 的 AD0~AD7，而 74244 的 2 個致能腳(1G、2G)接到 ATMEGA-8 的 GOE(PC5)腳，74273 的 CLK 接到 ATMEGA-8 的 RCLK(PC4)。

由前面 IC 介紹可知 74244 的 EN 腳皆為 0 時啟用，也就是當 ATMEGA-8 的 GOE(PC5)腳輸出 0 時啟用 74244 來控制綠色 LED。

而 74273 是在正緣觸發時運作，也就是當 ATMEGA-8 的 RCLK(PC4)先輸出 0 製造低態，再輸出 1 觸發正緣來暫存輸入，以此來控制紅色 LED

### 第一個參數 Data :

寫 VB 時這個位置都是填入當前要輸出高態的資料位置，像是我要讓第 1、3、5、7 號燈亮，就要輸入 0101 0101<sub>(2)</sub> 的 10 進制或 16 進制「85 or &H55」，然後 ATMEGA-8 就會依照你的 **Data** 輸入讓 AD0~AD7 腳位有對應的輸出，並將資料傳給 74244 與 74273 這兩顆 IC (電路圖)。



0101 0101<sub>(2)</sub> = 85，執行 OutDataCtrl(85, 先不看)，ATMEGA-8 在 AD0~AD7 的實際輸出：

AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0
0	1	0	1	0	1	0	1

### 第二個參數 Control :

Control 是控制 PC0~PC6，而 74244(綠色 LED)的 EN 腳接在 ATMEGA-8 的 PC5 上，要讓 74244 啟用的話，就必須使 ATMEGA-8 的 PC5 腳位輸出 0，所以 Control 參數會傳入 0000 0000<sub>(2)</sub>，要停用 74244 的話只要在 PC5 的位置輸出 1，Control 傳入 0001 0000<sub>(2)</sub> = 32，即可停用 74244。

而 74273(紅色 LED)的 CLK 腳接在 ATMEGA-8 的 PC4 上，前面也有提到，要讓 74273 暫存資料的話必須使用正緣觸發來讓暫存器運作，所以讓 PC4 腳位先輸出 0 歸位回低態，Control 傳入 0000 0000<sub>(2)</sub>，之後再讓 PC4 輸出 1 來觸發正緣，Control 這時傳入 0001 0000<sub>(2)</sub>，這時暫存器就能順利的儲存我們要控制的 LED 資料 0101 0101<sub>(2)</sub> = 85，讓 1、3、5、7 號紅色 LED 亮燈

Control							
 							
x	PC6	PC5	PC4	PC3	PC2	PC1	PC0
0	0	0	0	0	0	0	0

## 使用方式：

前面都了解的話這段就可以跳過了

如果要讓綠燈全滅的話，要讓 ATMEGA-8 的 AD0~AD7 輸出 0，所以 Data 要傳入 0，要啟用綠燈的 IC 74244 就要讓 ATMEGA-8 的 PC5 輸出 0，所以執行 `OutDataCtrl(0, 0)`，這樣就成功清除綠燈了。

讓綠燈亮 1、3、5、7 號的話，Data 部分要傳入 85，啟用 74244 要讓 PC5 輸出 0，所以執行 `OutDataCtrl(85, 0)`，這樣綠燈就依照指令亮燈了。

讓紅燈全滅的方式跟上面一樣，Data 一樣傳入 0，但 74273 要透過正緣觸發才能暫存資料，所以要分成兩個步驟：

1. PC4 先傳入 0，為了不動到綠燈，讓 PC5 傳入 1 來停用 74244，Control 為 `0010 0000` (2) = 32，因為 2 顆 IC 都沒有啟用，所以 Data 就無效了，這時 Data 可以先隨便傳值=> `OutDataCtrl(X, 32)`。
2. 然後再讓 PC4 傳入 1 觸發暫存器，將 Data 傳入 74273，Control 為 `0011 0000` (2) = 48，`OutDataCtrl(0, 32)`，然後紅燈就全滅了。

讓紅燈亮 1、3、5、7 號的話，Data 一樣傳入 85，剩下步驟跟剛剛一樣：

1. `OutDataCtrl(X, 32)`
2. `OutDataCtrl(85, 48)`

## 結語：

乙檢電路板內容差不多就這樣了，範例程式中的 Display 只是把 10 進制轉 2 進制，在表單上顯示 LED 狀態而已，照著程式走一遍就會了，另一站系統那邊照著步驟安裝就躺著過吧。

至於為什麼 Data 參數可以控制 AD0~AD7、Control 參數可以控制 PC0~PC6，我只大概了解 ATMEGA-8 的 PortB、PortC、PortD 皆為可控的 I/O 腳位，能透過指令操控腳位狀態，而控制方式大概跟 USBIO 的操作內容和 ATMEGA-8 的運作原理有關，這邊我也不是很了解。

## 參考資料：

展維隨筆: <http://davidhsu666.com>

LS74273 Datasheet : <https://goo.gl/pbeKit>

LS74244 Datasheet : <https://goo.gl/rQAnMn>

ATmega8 Datasheet : <https://goo.gl/1ZcMjR>