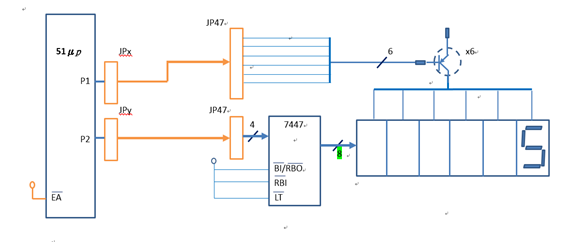
**微處理機 Lab4 結報**

0413335 郭逸琳 0416039 李佳燕

實驗主題與訓練目的

此次實驗主要是先透過接線了解七段式LED的運作方式，再進一步實驗能不能只用線路不用程式就能接出想要的結果。

實驗電路



實驗過程

先在uvision裡貼上下列的code：

org 0

mov SP, #50H

start: mov R7, #6

next1:

mov R5, #250

next11:

mov R6, #6

mov R1, #0FEH

mov R2, #0

next12:

mov A, R1

mov P1, A

RL A

mov R1, A

mov A, R2

inc R2

mov P2, A

call delay1 ; ===KKK===

djnz R6, next12

djnz R5, next11

mov P1, #0FFH

call delay2 ; ===LLL===

djnz R7, next1

mov R7, #6

next2:

mov P1, #0F8H

mov P2, #5

call delay2 ; ===III===

mov P1, #0C7H

mov P2, #1

call delay2 ; ===JJJ===

djnz R7, next2

jmp start

delay1:

push 1

mov R1, #200

djnz R1, $

pop 1

ret

delay2: ; appx. 0.5sec delay, why?

push 1

push 2

push 3

mov R1, #100

dd22: mov R2, #250

dd21: mov R3, #10

djnz R3, $

djnz R2, dd21

djnz R1, dd22

pop 3

pop 2

pop 1

ret

end

然後再利用以下方式接線：

P1接8bit的線(我們那組板子沒有只有7bit的線, 但其他組好像有)，線的另外一頭接到七段式LED的下排，P2接4bit的線，線的另外一頭接到七段式LED的上排。

最後將程式碼燒到板子裡再執行就可看到結果。

解釋程式碼

P1會是LED亮的位置, P2則是要亮的數字

先給P1 0FEh(第一個LED亮其他暗)，P2給0(亮數字0)，P1做left rotate的同時P2+1(亮的數字+1且改成隔壁的LED亮)，然後每次做完後要call一次delay，重複此行為六次後把P1改為FFH(全暗)並call delay(這次delay的時間比較長)。這一大段的行為總共會重複6次。然後進入第二階段，先給P1 0F8H(亮右半邊)，P2給5(亮數字5)，亮一次並call delay後將P1改成0C7H(亮左半邊)，將P2改成1(亮數字1)，亮完後call delay並重複此行為6次。

實驗結果

這次作業有兩個pattern：

第一個pattern是一次只亮一個跑6\*250次(從第一個LED到最後一個LED再重頭)再全暗，整個行為跑六次。但因為跑很快且全暗的delay比較久, 所以整體看起來不會是一次亮一個的, 會是全部都亮的，而且會閃爍。

第二個pattern是右半邊先亮數字5, 接下來左半邊亮數字1, 一直輪流交替。

實驗延伸

Manually wire-up the circuit so that all 7-seg digits display “0”, without code-driving. Rewire the circuit so that “0” appears on digit-0,

and again without code-driving. Explain the difference in the display intensity perceived in the two cases.

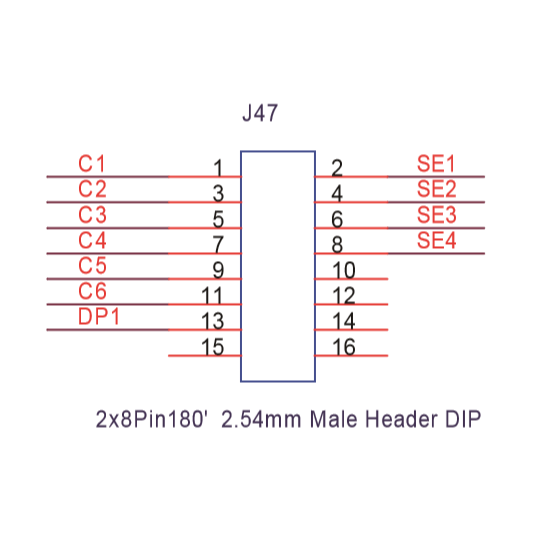
不更改Code, 僅僅用接線的方式, 達到上述的要求

7段顯示器上排用了4個bit(控制顯示的數字), 下排用7個bit(6個小塊+1個點)(控制顯示的位置)，所以下排要接在控制開關的旁邊, 就是DIP SW1~8, 這樣就可以透過開關來選擇要讓哪個亮，而上排就要接0000給它(可以透過P1或P2中一直是0或是其他pin預設是0的位置(J28的LOW)接給它)。

透過實驗我們發現只有在開關那排右邊的其中一排的pin(上面有寫L的)預設是0其他所有pin預設都是1，包括七段式LED上的pin，所以如果LED中的四個bit中要設為1的bit可以不用接線，只要把要設為0的bit接線就可以了。例如要顯示3(00000011)的話, 因為只取後面4個位數, 所以只看0011, 這時候1的訊號不用接, 因為LED的預設就是1, 但0一定要接, 不然會出錯。

實驗模組

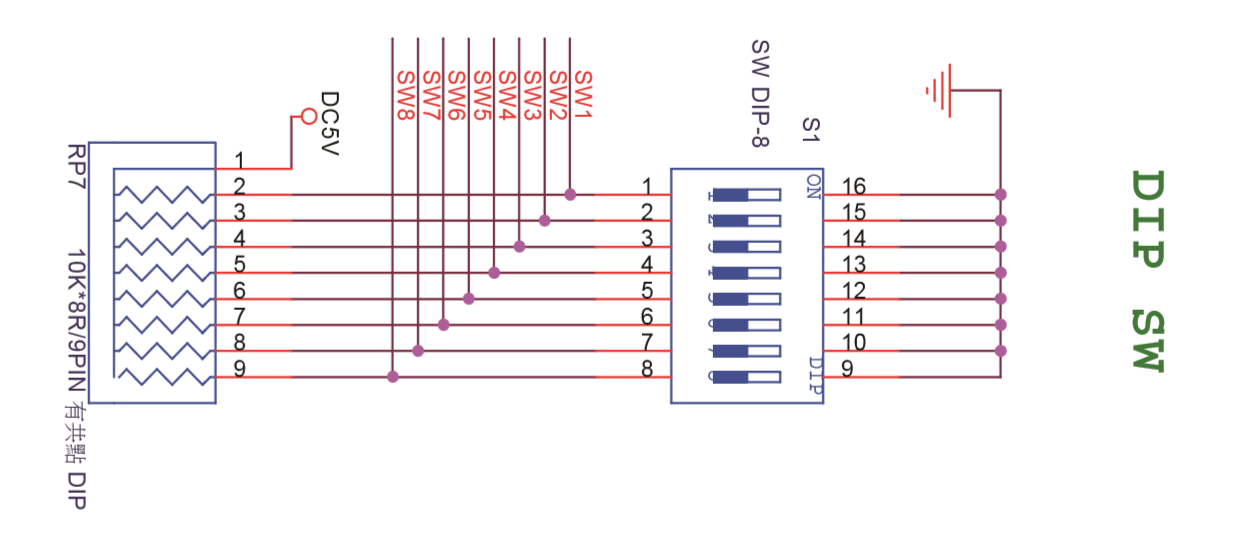
J47 :



左邊的就是板子上 下排的7個bit (C1~6控制要亮的區塊; DP1控制要不要亮小數點)，bit為0表示要亮，bit為1表示不要亮。

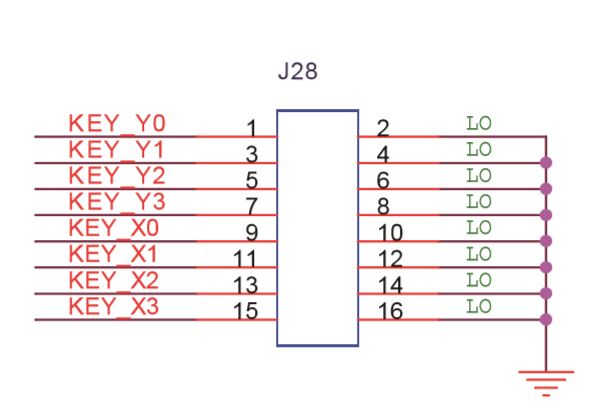
右邊的就是板子上 上排的4個bit，控制要顯示的數字。

DIP SW 1~8 :



透過開關可以控制要亮哪一個(6個區塊或是1個點，第八個開關不會用到)

J28的LOW :



透過名字可以看出來, 他的預設值是0，可用來控制要顯示的數字。

實驗心得

原來單純靠用板子接線, 就可以讓這個7-segment LED顯示出想要的數字, 以及要亮哪個(共有6個小塊, 加上一個點)。

然後我們還發現, 因為7段LED的下排接在開關那邊(DIP SW1~8), 我們發現如果讓越多個開關打開, 平均的亮度就會減低，因為同樣的電流會分散出去, 同理，只有一個開關打開的時候是最亮的。

透過這次實驗，我們比之前更加了解板子及各個pin的運作方式，更能體會到微處理機的樂趣所在。