**微算機實驗報告格式**

**姓姓名：陳達軒**

**學號：0610837**

**上課時間：**

**2018 Fall 2EF**



Lab # 4

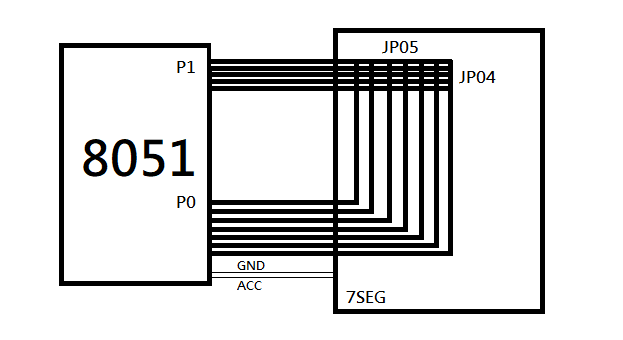
**ㄧ、實驗目的：**

但請以自己對本次實驗的認知來加以說明。

了解點矩陣控制方式，和實際熟悉點矩陣使用流程。

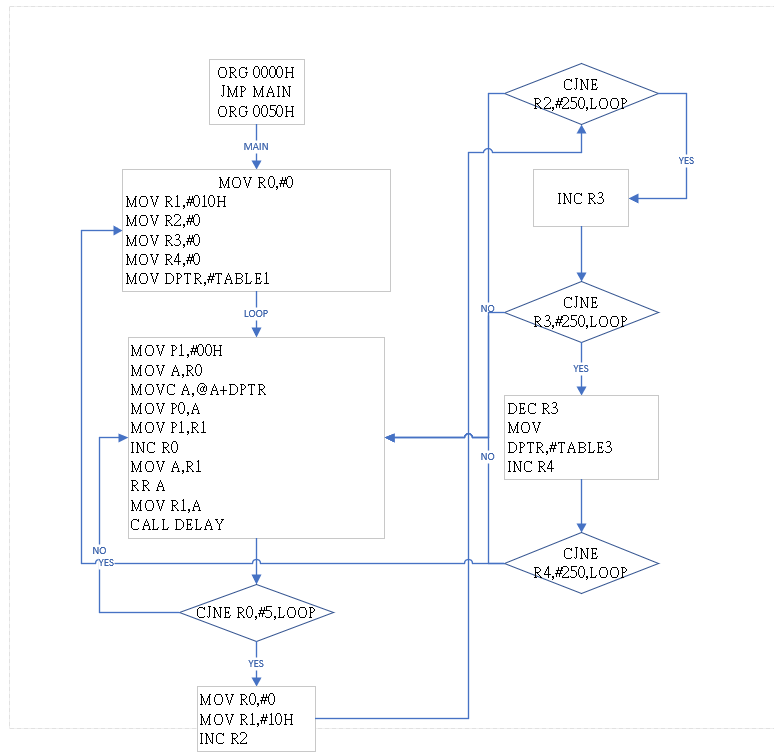
**二、硬體架構：**

請儘量自己畫圖不管是手繪或者是以電腦繪圖(Word or PoworPoint)。

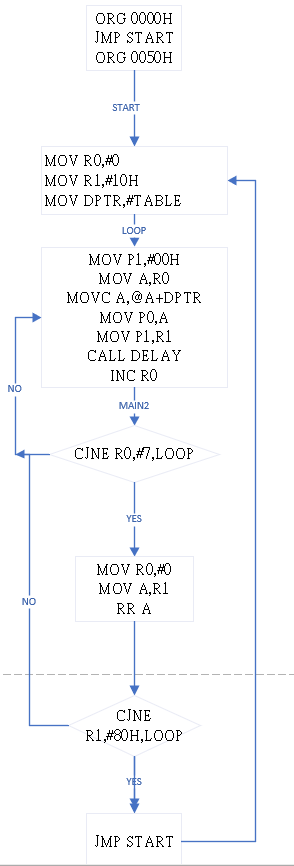


**三、程式流程圖：**

基礎題：



進階題：



**四、問題與討論：**

Note:

1.回答助教在實驗講義中所提的問題。

2.自己發現的疑問以及如何解決的方法。

如果直接給予JP04 10000，則會有一排全亮的狀況，此時因為JP05沒有訊號，所以全部為0，所有燈在JP04許可的狀況下都會亮起。如果delay time時間變長對於目前作用的單一燈來說會更亮一些，但是整體上來說這個時間點其他的燈會稍微暗一點，也可能因為太短而造成所有燈都亮起來的情形，或著過慢無法形成殘像，此時就會分散開來。

**五、心得：**

Note:

1.對於實驗內容的心得感想

2.對本課程的建議。

我覺得這次的程式和上次的四個七段顯示器概念很相似，同樣是用一部份的資料來控制輸出的目標，不過不同的是控制的方式和之前不同，利用一個一個bit去分別控制點矩陣的各個點和列，延遲的方式則和上次差不多沒有太大的差異，希望未來能在課程中學習到更多不同的方式來進行控制。這次這需要多注意0和1的關係，有時候0是亮有時候則相反。

**六、程式碼與註解：**

基礎題：

ORG 0000H //從00開始

JMP MAIN //跳到MAIN

ORG 0050H //從500開始

MAIN:

MOV R0,#0 //R0歸零

MOV R1,#010H //R1=10

MOV R2,#0 //R2歸零作為counter

MOV R3,#0 //R3歸零

MOV R4,#0 //R4歸零

MOV DPTR,#TABLE1 //將TABLE放置DPTR

LOOP:

MOV P1,#00H //00輸出P1

MOV A,R0 //A=R0

MOVC A,@A+DPTR //DPTR放到A

MOV P0,A //輸出到A+R0

MOV P1,R1 //輸出R1

INC R0 //R0++

MOV A,R1 //A=R1

RR A //R1輪轉

MOV R1,A

CALL DELAY //顯示延遲

CJNE R0,#5,LOOP //如果R0沒到5就重複上述步驟

MOV R0,#0 //R0歸零

MOV R1,#10H //R1=10

INC R2 //R2++

CJNE R2,#250,LOOP //如果R2!=250則重複LOOP步驟

DEC R2 //R2--

MOV DPTR,#TABLE2 //將TABLE2放到DPTR

INC R3 //R3++

CJNE R3,#250,LOOP //如果R3!=250則重複LOOP步驟

DEC R3 //R3--

MOV DPTR,#TABLE3 //將TABLE3放到DPTR

INC R4 //R4++

CJNE R4,#250,LOOP //如果R4!=250則重複LOOP步驟

JMP MAIN //跳到MAIN

TABLE1:

DB 00010010B

DB 00010100B

DB 01111000B

DB 00010100B

DB 00010010B

TABLE2:

DB 00011100B

DB 00010100B

DB 01111111B

DB 00010100B

DB 00011100B

TABLE3:

DB 00001000B

DB 00010010B

DB 01111110B

DB 00010000B

DB 00001000B

DELAY:

MOV R6,#50

DELAY1:

MOV R7,#50

DELAY2:

DJNZ R7,DELAY2

DJNZ R6,DELAY1

RET

END

進階題：

ORG 0000H //從00開始

JMP START //跳到START

ORG 0050H //從50開始

START:

MOV R0,#0 //R0清空

MOV R1,#10H //R1=10

MOV DPTR,#TABLE //將TABLE放到DPTR當陣列使用

LOOP:

MOV P1,#00H //P1清空

MOV A,R0 //A=R0

MOVC A,@A+DPTR //用R0的值去取DPTR的值

MOV P0,A //輸出到P0

MOV P1,R1 //R1移到P1

CALL DELAY //延遲顯示

INC R0 //R0++

CJNE R0,#7,LOOP //如果R0!=7則重新執行上述步驟

MOV R0,#0 //將R0歸零，以待下一行使用

MOV A,R1 //A=R1

RR A //A輪轉

MOV R1,A //將A移回R1

CJNE R1,#80H,LOOP //若是R1尚未執行完一輪5排，則回到LOOP

JMP START

TABLE:

DB 01000000B

DB 00100000B

DB 00010000B

DB 00001000B

DB 00000100B

DB 00000010B

DB 00000001B

DELAY:

MOV R6,#100

DELAY1:

MOV R7,#100

DELAY2:

MOV R5,#20

DELAY3:

DJNZ R5,DELAY3

DJNZ R7,DELAY2

DJNZ R6,DELAY1

RET

END