



微算機實驗報告

Lab #09-2

姓名：洪巧芸

系級：資工 114

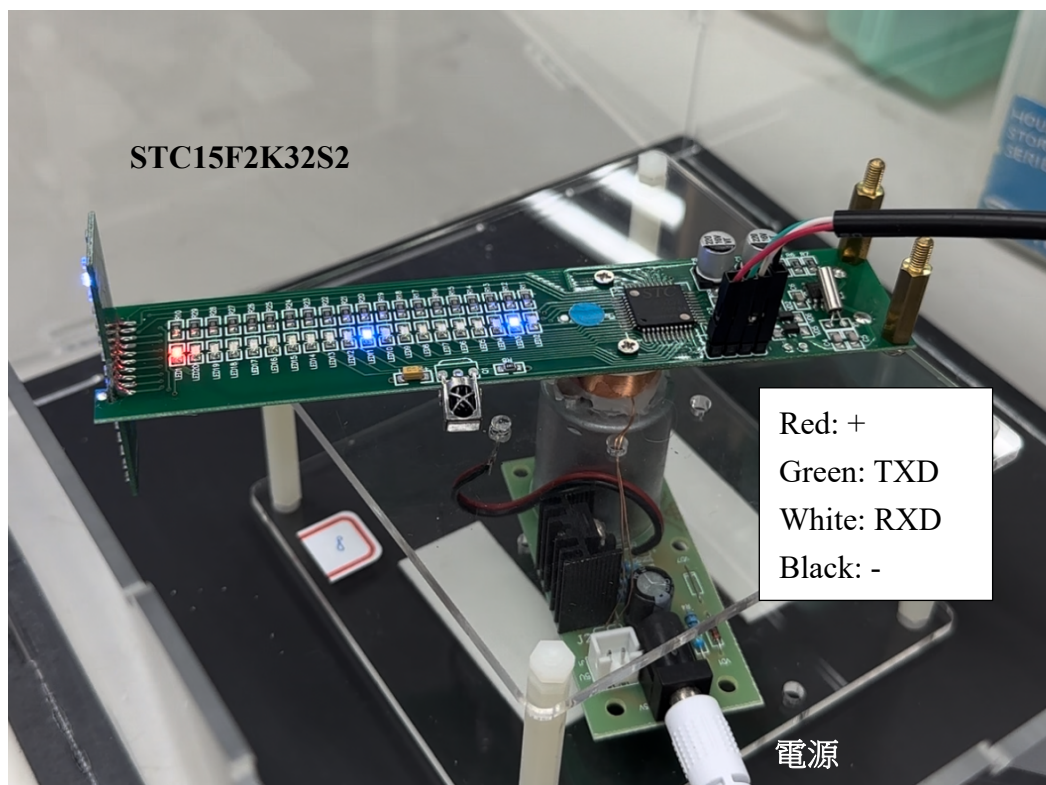
學號：110550143

上課時間：2023/12/5

一、實驗目的：

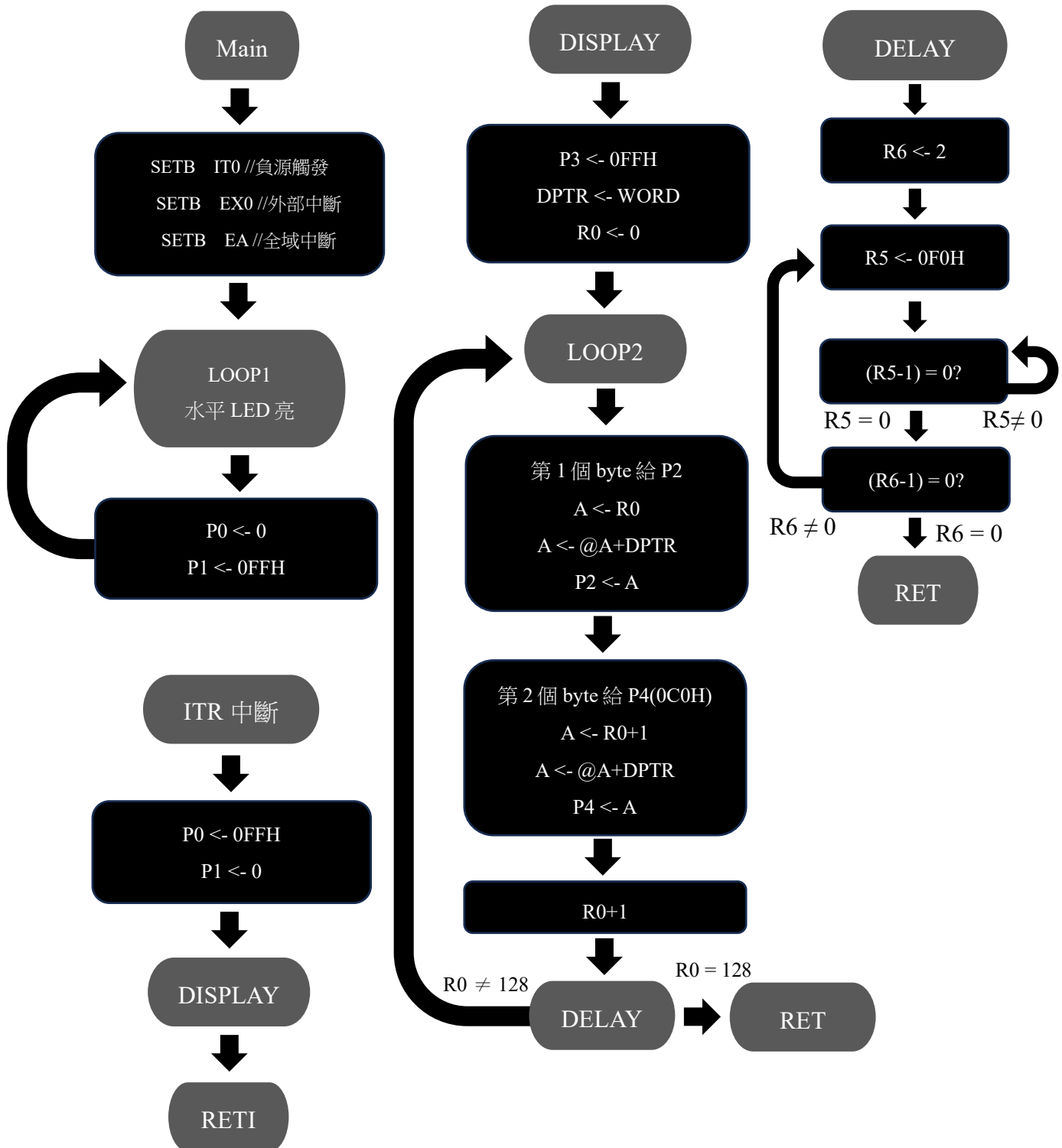
了解外部中斷原理與大型 TABLE 之讀取，並基於外部中斷準確控制 LED 畫面顯示。學習如何分開控制平面與側面的 LED 與負源觸發及資料讀取與輸出。

二、硬體架構：

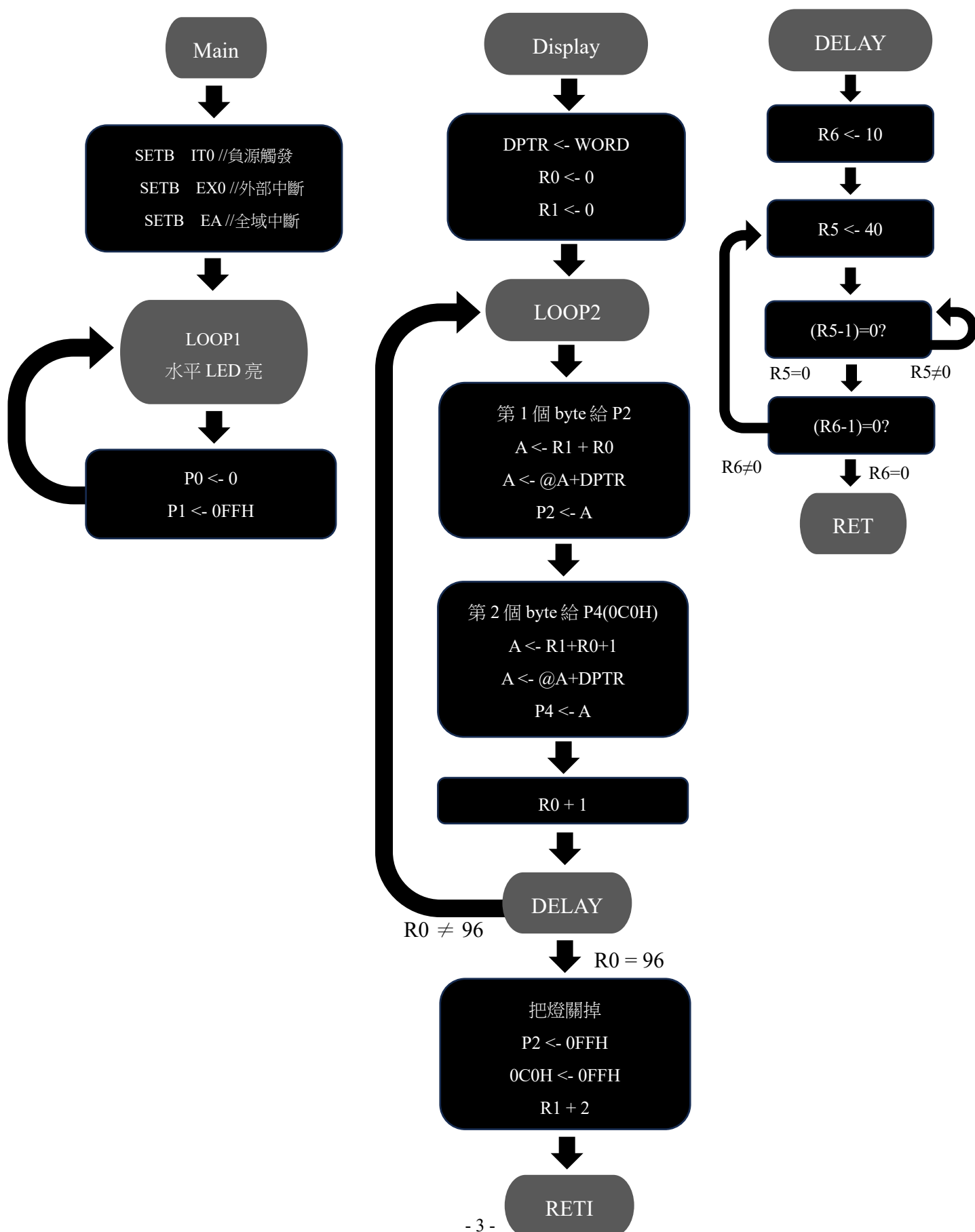


三、程式流程圖：

A. 基本題



B. 進階題



四、問題與討論：

(1) 詳述如何使旋轉 LED 進入外部中斷。

首先要先將中斷的設定開啟，設定中斷的優先順序、中斷允許位元、中斷觸發方式之後，在主程式的前面，使用 `ORG` 設定中斷向量的起始位址，然後使用 `AJMP` 指令跳轉到中斷後要執行的程式。在 `AJMP` 指令後，定義中斷觸發時要執行的程式碼。而這次實驗的中斷訊號來自 P3.2，也就是微控器的紅外線 SENSOR，需要確定中斷的觸發方式（電位或負源觸發）並設定計時器 0（`TMOD`、`TH0`、`TL0`）的相關參數，以確保中斷在正確的時間觸發。

五、程式碼與註解：

A、基本題

```
1  ORG    0000H
2  AJMP   MAIN
3  ORG    0003H
4  AJMP   ITR //中斷副程式的入口
5  ORG    0050H //新的起始位址
6
7  //外部中斷的設定
8  MAIN:
9      SETB    IT0 //負源觸發 0的時候燈亮
10     SETB    EX0 //外部中斷
11     SETB    EA //全域中斷
12
13     //水平LED亮
14     LOOP1:
15         MOV    P0, #0
16         MOV    P1, #0FFH
17         AJMP   LOOP1
18
19     //中斷：內圈暗外圈亮
20     ITR :
21         MOV    P0, #0FFH
22         MOV    P1, #0
23         JMP    DISPLAY //顯示文字
24         //RETI //中斷的RETURN
25
26     //顯示文字
27     DISPLAY:
28         MOV    DPTR, #WORD //把要顯示的Table移到DPTR的位址
29         MOV    R1, #0 // 從第一個數值開始顯示
30         MOV    R0, #0 //從第一個數值開始讀取
31
32     //側邊LED亮
33     LOOP2:
34         //第一個byte輸出給P2
35         MOV    A, R1
36         ADD    A, R0
37         MOVC   A, @A+DPTR
38         MOV    P2, A //顯示第一個數值
39
40         //第二個byte輸出給P4
41         INC    R0 //R0+1 換下一個數值
42         MOV    A, R1
43         ADD    A, R0
44         MOVC   A, @A+DPTR
45         MOV    0C0H, A // 0C0H = P4
46
47         INC    R0
48         ACALL  DELAY
49         CJNE   R0, #96, LOOP2 //Table內容數值共有 8*4*3 = 96
50
51         //把燈關掉
52         MOV    P2, #0FFH
53         MOV    0C0H, #0FFH
54
55         //下次顯示下一行，又因為兩個byte為一組，所以R1要+2
56         INC    R1
57         INC    R1
58         RETI
```

B、進階題

```
1  ORG    0000H
2  AJMP   MAIN
3  ORG    0003H
4  AJMP   ITR //中斷副程式的入口
5  ORG    0050H //新的起始位址
6
7  //外部中斷的設定
8  MAIN:
9      SETB    IT0 //負源觸發 0的時候燈亮
10     SETB    EX0 //外部中斷
11     SETB    EA //全域中斷
12
13     //水平LED亮
14     LOOP1:
15         MOV    P0, #0
16         MOV    P1, #0FFH
17         AJMP   LOOP1
18
19     //中斷：內圈暗外圈亮
20     ITR :
21         MOV    P0, #0FFH
22         MOV    P1, #0
23         JMP    DISPLAY //顯示文字
24
25
26     //顯示文字
27     DISPLAY:
28         MOV    DPTR, #WORD //把要顯示的Table移到DPTR的位址
29         MOV    R1, #0 // 從第一個數值開始顯示
30         MOV    R0, #0 //從第一個數值開始讀取
31
32     //側邊LED亮
33     LOOP2:
34         //第一個byte輸出給P2
35         MOV    A, R1
36         ADD    A, R0
37         MOVC    A, @A+DPTR
38         MOV    P2, A //顯示第一個數值
39
40         //第二個byte輸出給P4
41         INC    R0 //R0+1 換下一個數值
42         MOV    A, R1
43         ADD    A, R0
44         MOVC    A, @A+DPTR
45         MOV    0C0H, A // 0C0H = P4
46
47         INC    R0
48         ACALL  DELAY
49         CJNE   R0, #96, LOOP2 //Table內容數值共有 8*4*3 = 96
50
51         //把燈關掉
52         MOV    P2, #0FFH
53         MOV    0C0H, #0FFH
54
55         //下次顯示下一行，又因為兩個byte為一組，所以R1要+2
56         INC    R1
57         INC    R1
58         RETI
```

```

60 //延遲
61 DELAY:
62     MOV     R6,#10
63 DLAY2:
64     MOV     R5,#40
65 DLAY1:
66     DJNZ    R5,DLAY1
67     DJNZ    R6,DLAY2
68     RET

```

六、心得：

1. 對於上課內容的心得感想：

一開始看到實驗內容的時候覺得和上次的差不多，沒有很清楚實驗的重點，經過老師講解之後才了解到中斷的用處以及基本概念。

2. 對於實驗內容的心得感想：

上次的實驗沒有用到中斷，且在 LED 顯示的部分只有用到側邊顯示，但這次加入了中斷程式及水平 LED 顯示，剛開始一直不知道到底要怎麼讓水平 LED 在有字的部分亮外圈，沒字的部分亮內圈，後來把它們移到中斷的部分對調內外圈的亮暗才對了。