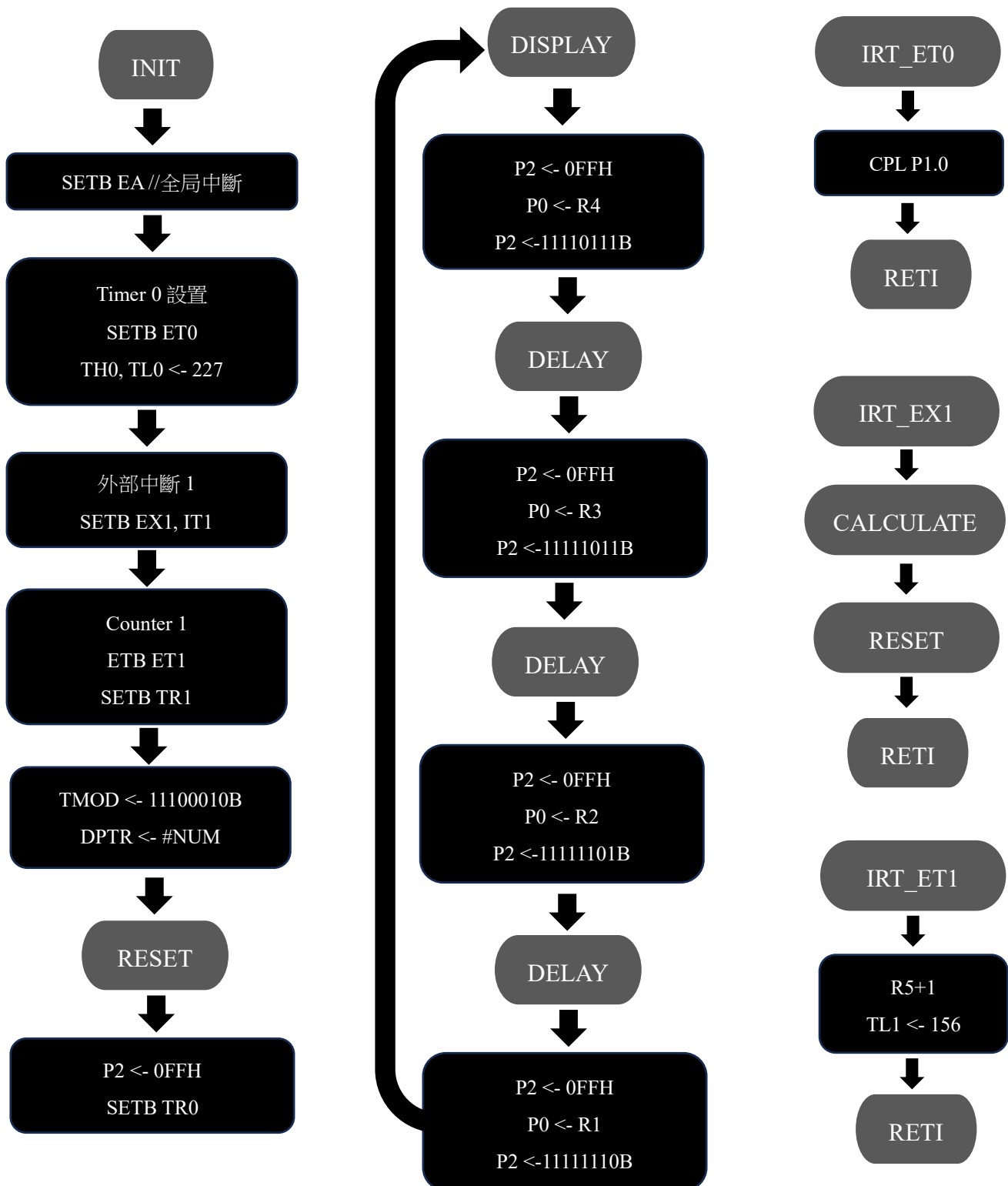
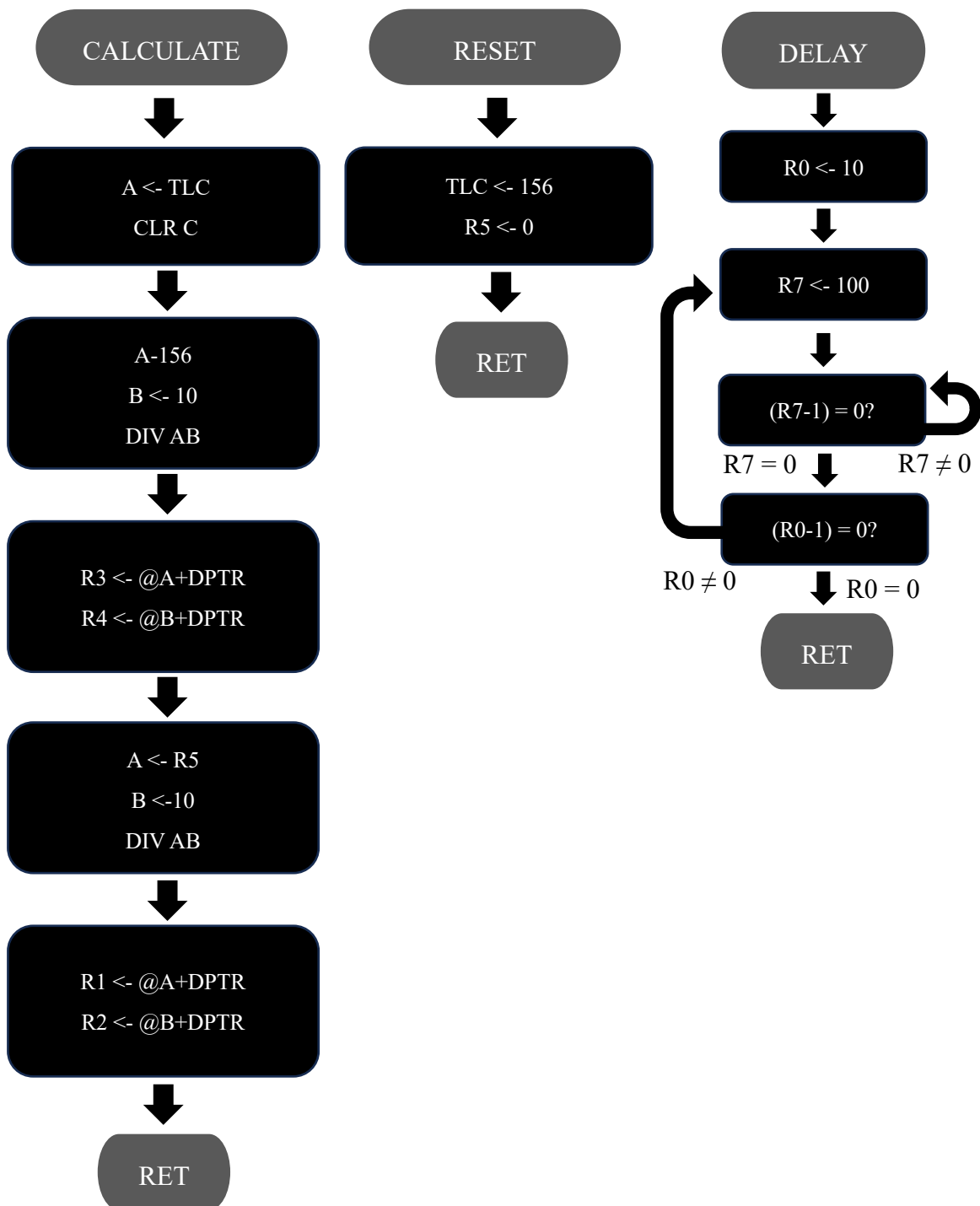




### 三、程式流程圖：





#### 四、問題與討論：

(1) 有哪些因素會造成頻率計算結果之誤差？

<i> Timer 以整數重設，導致時間上的小誤差

<ii> Timer 輸出方波和清除旗標的指令執行時延遲

<iii> 系統石英震盪器的固有誤差 (ex.製造缺陷和溫度影)

<iv> Counter 可能在兩次 Timer 觸發間結束計數

(小於一個 Timer 周期的誤差)

(2) 如果要增加頻率計算的範圍或是計算結果的解析度可以怎樣來設計？

<i> 減少 Timer 週期長度以降低 ECHO 結束時間點的誤差

(但會增加頻率計算的下限)

<ii> 增加 Counter 的紀錄長度，例如使用一個額外的暫存器記錄 TH

overflow 次數 (這樣的話資料記錄長度就會變成 24bits)，也可以對最

高 8bits 的暫存器紀錄溢位次數，進一步增加記錄範圍。

## 五、程式碼與註解：

```
1  ORG    0000H    // 程式起始位置
2  LJMP   INIT     // 跳轉至初始化子程序
3  ORG    000BH    // 定位至 Timer 0 中斷服務程序
4  LJMP   ITR_ET0   // 跳轉至 Timer 0 中斷服務程序 (MODE 2)
5  ORG    0013H    // 定位至外部中斷 1 中斷服務程序
6  LJMP   ITR_EX1   // 跳轉至外部中斷 1 中斷服務程序
7  ORG    001BH    // 定位至計數器 1 中斷服務程序
8  LJMP   ITR_ET1   // 跳轉至計數器 1 中斷服務程序 (MODE 2)
9  ORG    0050H    // 其他程式碼存放位置
10
11  INIT:
12      //全局中斷
13      SETB  EA      // 啟用
14
15      //Timer0中斷設置
16      SETB  ET0      // 啟用
17      MOV   TH0, #227 // 高位計數器初始值=58ms
18      MOV   TL0, #227 // 低位計數器初始值=58ms
19
20      //外部中斷1(回音信號中斷處理)
21      SETB  EX1      // 啟用
22      SETB  DSP 1     // 下降沿觸發
23
24      //計數器1中斷
25      ETB    ET1      // 啟用
26      SETB  TR1      // 啟動
27
28      MOV   TMOD, #11100010B // 配置計時器/計數器模式
29
30      MOV   DPTR, #NUM // 設定數碼管顯示表的資料指針
31      LCALL RESET     // 調用重置子程序
32      MOV   P2, #0FFH // 初始化 P2(高電位)
33      SETB  TR0      // 啟動 Timer 0
34
35  /*顯示循環，逐一更新四個數碼管顯示
36      (R4->D1, R3->D2, R2->D3, R1->D4)
37      Step1: 先將P2設定為高電位
38      Step2: 將要調用的暫存器數值輸出到P0
39      Step3: 顯示在要顯示的位置(D1-D4)
40      Step4: 延遲*/
41  DISPLAY:
42      //R4->D1
43      MOV   P2, #0FFH // P2 高電位
44      MOV   P0, R4
45      MOV   P2, #11110111B
46      LCALL DELAY
```

```

48      //R3->D2
49      MOV     P2, #0FFH    // P2 高電位
50      MOV     P0, R3
51      MOV     P2, #11111011B
52      LCALL   DELAY
53
54      //R2->D3
55      MOV     P2, #0FFH    // P2 高電位
56      MOV     P0, R2
57      MOV     P2, #11111101B
58      LCALL   DELAY
59
60      //R1->D4
61      MOV     P2, #0FFH    // P2 高電位
62      MOV     P0, R1
63      MOV     P2, #11111110B
64      LCALL   DELAY
65
66      LJMP    DISPLAY      // 跳回至 DISPLAY
67
68 ITR_ET0:
69      CPL     P1.0         // 反轉 P1.0 的狀態
70      RETI              // 返回中斷
71
72 ITR_EX1:
73      LCALL   CALCULATE    // 調用計算子程序
74      LCALL   RESET        // 調用重置子程序
75      RETI              // 返回中斷
76
77 ITR_ET1:
78      INC     R5           // R5+1
79      MOV     TL1, #156    // 設定計數器1的閾值
80      RETI              // 返回中斷

```

```

82      /*利用DIV更新A,B的數值後再輸入暫存器
83      |  DIV AB -> A/B = (new)A ... (new)B*/
84      CALCULATE:
85      MOV     A, TL1
86      CLR     C             // 清除進位標誌
87
88      //A=A-156, B=10計算(A/B)的商和餘數
89      SUBB    A, #156
90      MOV     B, #10
91      DIV     AB
92
93      /*把計算出來的數值對應到Table的相對位置取出
94      |  商 -> 放入R3, 餘數 -> 放入R4*/
95      MOVC    A, @A+DPTR
96      MOV     R3, A
97      MOV     A, B
98      MOVC    A, @A+DPTR
99      MOV     R4, A

```

```

101      //A=R5, B=10重新計算(A/B)的商和餘數
102      MOV     A, R5
103      MOV     B, #10
104      DIV     AB
105
106      /*把計算出來的數值對應到Table的相對位置取出
107      | 商 -> 放入R1, 餘數 -> 放入R2*/
108      MOVC     A, @A+DPTR
109      MOV     R1, A
110      MOV     A, B
111      MOVC     A, @A+DPTR
112      MOV     R2, A
113      RET                      // 返回
114
115      //重新設置
116      RESET:
117      MOV     TL1, #156      // 重置計數器1的閾值
118      MOV     R5, #0         // R5歸0
119      RET                      // 返回
120
121      //延遲
122      DELAY:
123      MOV     R0, #10
124      DELAY2:
125      MOV     R7, #100
126      DELAY1:
127      DJNZ    R7, DELAY1
128      DJNZ    R0, DELAY2
129      RET
130
131      NUM:
132      DB      0C0H, 0F9H, 0A4H, 0B0H, 099H, 092H, 082H, 0F8H, 080H, 090H //0-9
133      END

```

## 六、心得：

### 1. 對於上課內容的心得感想：

在看到 Spec 的時候覺得超音波模組很熟悉，之前在電子實驗的時候有接觸過，但當初只是配合電路板使用，並不需要加入程式碼去計算相較精確的數值，因此也沒有想過只是單存的距離測量卻牽扯到 `pluse` 數量的計算與花費時間。

### 2. 對於實驗內容的心得感想：

這次的實驗相比之前的我覺得有很大程度的提升，除了對於 `Timer` 跟 `Counter` 兩者分別的紀錄數據及功能要有明確的認知外，還要知道兩者之間的相互作用，如此一來才能夠順利地計算出距離。在計算完距離後又需要面臨另外一個難題，也就是數字的顯示，需要進行拆分才能夠送入七段顯示器輸出數值，是一次很考驗邏輯的實驗。