



# 微算機實驗報告

## Lab #4

姓名：楊哲睿

系級：電機 10

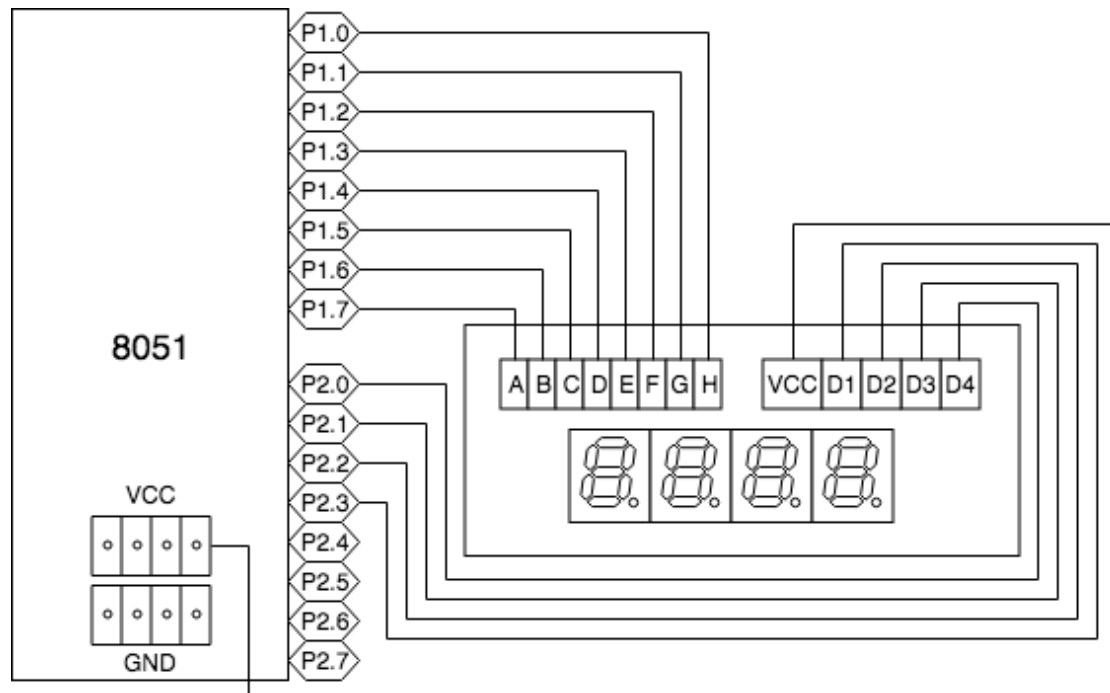
學號：0610780

上課時間：4EF、4IJ

### 一、實驗目的：

- 瞭解四顆七段顯示器的電路結構以及相關的控制方法。

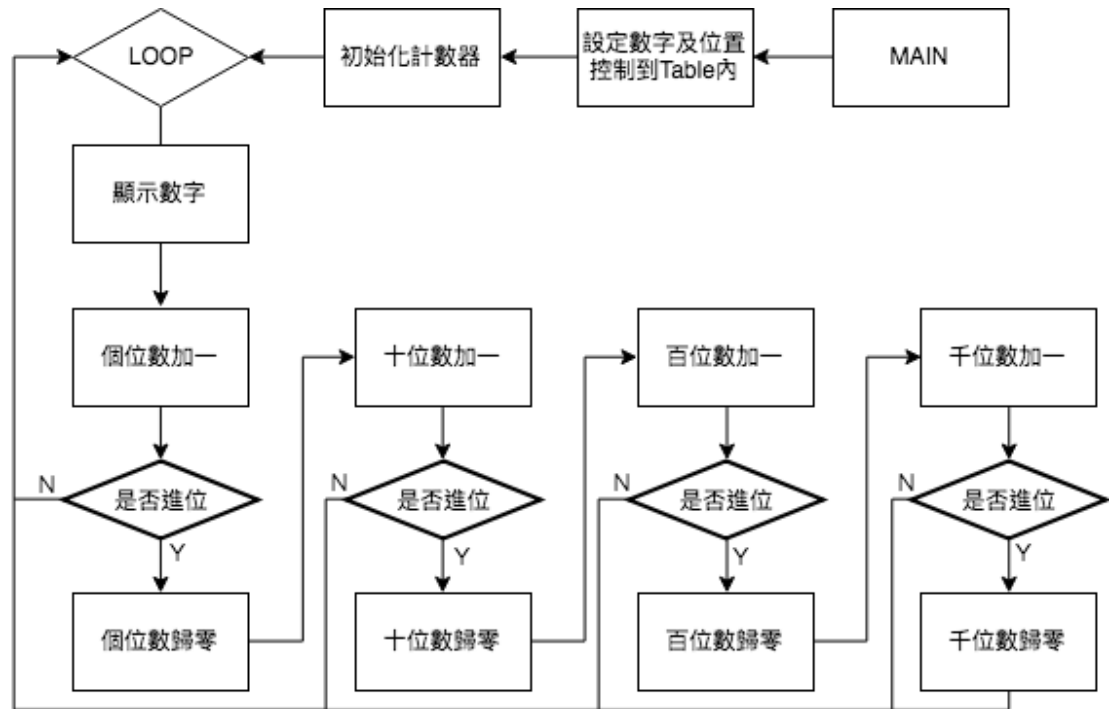
### 二、硬體架構：



### 三、程式流程圖：

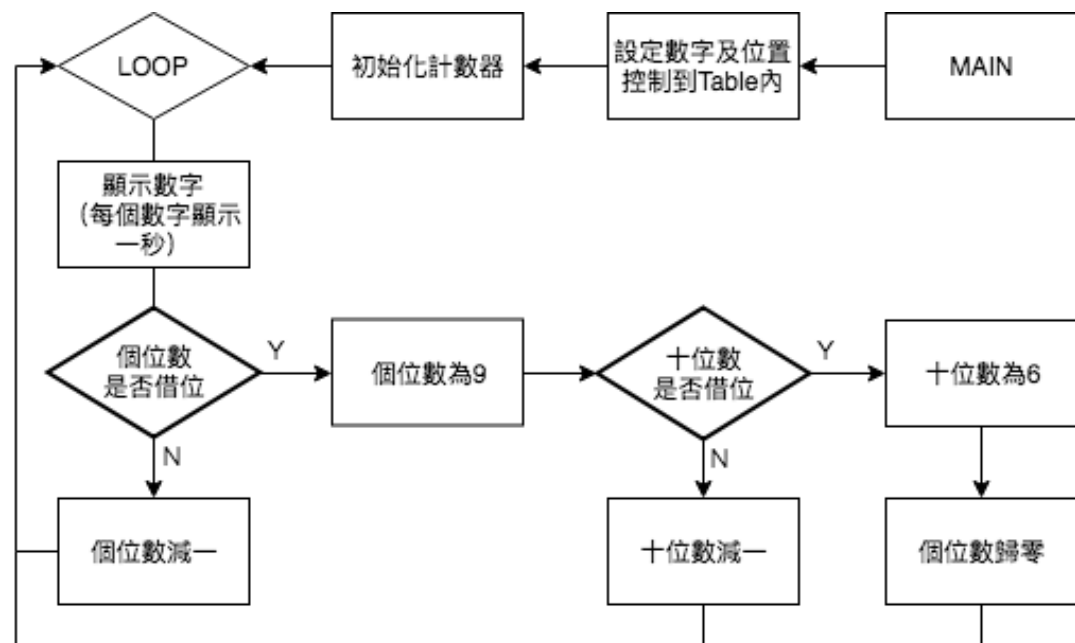
#### ● 基本題

設計一個上數計數器，由 9990→9991→...→9999→0000→0001→...依序顯示於 4 個七段顯示器上面，間隔時間自己設定，基本上不要快到看不出來，也不要慢到等了整堂課都沒有 run 完就可以。



#### ● 進階題

設計一個倒數碼錶，由 5→4→...→0→60→59→...依序顯示於最右邊兩個七段顯示器，間隔時間請盡可能接近 1 秒。



#### 四、問題與討論：

- 若要使用一個開關來控制計數器 run 或 stop，要如何設計？

先用 I/O 把開關的訊號吃進來，然後在程式的最前面設計一個迴圈，當開關等於某個特定值時（看要用 0 還是 1），就讓程式困在一個無窮迴圈（大概是用指令 CJNE 做判斷然後用指令 AJMP 形成迴圈），不要跑到計數器的部分，直到切換開關為止。

#### 五、程式碼與註解：

- 基本題

設計一個上數計數器，由 9990→9991→...→9999→0000→0001→...依序顯示於 4 個七段顯示器上面，間隔時間自己設定，基本上不要快到看不出來，也不要慢到等了整堂課都沒有 run 完就可以。

```
ORG 0000H
AJMP MAIN
ORG 0050H
MAIN:
    MOV R0, #0H        ;個位數
    MOV R1, #9H        ;十位數
    MOV R2, #9H        ;百位數
    MOV R3, #9H        ;千位數

LOOP1:
    ACALL SHOW

    INC R0
    CJNE R0, #0AH, LOOP1
    MOV R0, #0H
;個位數進位
    INC R1
    CJNE R1, #0AH, LOOP1
    MOV R1, #0H
;十位數進位
    INC R2
    CJNE R2, #0AH, LOOP1
    MOV R2, #0H
;百位數進位
    INC R3
```

```

    CJNE R3, #0AH, LOOP1
    MOV R3, #0H
    AJMP LOOP1
SHOW:
    MOV R4, #0H           ;作為計數器讓數字重複顯示
    MOV R5, #0H           ;形成視覺暫留
SHOWLOOP:
;First digjt
    MOV DPTR, #POS         ;先選位數
    MOV A, #0H
    MOVC A, @A+DPTR
    MOV P2, A

    MOV DPTR, #NUMTABLE    ;再選數字
    MOV A, R0
    MOVC A, @A+DPTR
    MOV P1, A

    ACALL DELAY
    MOV P1, #0FFH          ;避免干擾下一位數的顯示
;Second digit
    MOV DPTR, #POS         ;先選位數
    MOV A, #1H
    MOVC A, @A+DPTR
    MOV P2, A

    MOV DPTR, #NUMTABLE    ;再選數字
    MOV A, R1
    MOVC A, @A+DPTR
    MOV P1, A

    ACALL DELAY
    MOV P1, #0FFH          ;避免干擾下一位數的顯示
;Third digit
    MOV DPTR, #POS         ;先選位數
    MOV A, #2H
    MOVC A, @A+DPTR
    MOV P2, A

```

```

MOV DPTR, #NUMTABLE ;再選數字
MOV A, R2
MOVC A, @A+DPTR
MOV P1, A

ACALL DELAY
MOV P1, #0FFH ;避免干擾下一位數的顯示
;Forth digit
MOV DPTR, #POS ;先選位數
MOV A, #3H
MOVC A, @A+DPTR
MOV P2, A

MOV DPTR, #NUMTABLE ;再選數字
MOV A, R3
MOVC A, @A+DPTR
MOV P1, A

ACALL DELAY
MOV P1, #0FFH ;避免干擾下一位數的顯示

INC R4
CJNE R4, #0FFH, SHOWLOOP
MOV R4, #0
INC R5
CJNE R5, #02H, SHOWLOOP

RET
;=====
DELAY:
MOV R6, #0FFH
DELAY1:
MOV R7, #02H
DELAY2:
DJNZ R7, DELAY2
DJNZ R6, DELAY1
RET

```

```
;=====
```

```
NUMTABLE:
```

```
    DB 0C0H ;0
```

```
    DB 0F9H ;1
```

```
    DB 0A4H ;2
```

```
    DB 0B0H ;3
```

```
    DB 099H ;4
```

```
    DB 092H ;5
```

```
    DB 082H ;6
```

```
    DB 0F8H ;7
```

```
    DB 080H ;8
```

```
    DB 090H ;9
```

```
;=====
```

```
POS:
```

```
    DB 0F7H, 0FBH, 0FDH, 0FEH
```

● 進階題

設計一個倒數碼錶，由 5→4→...→0→60→59→...依序顯示於最右邊兩個七段顯示器，間隔時間請盡可能接近 1 秒。

```
ORG 0000H
```

```
AJMP MAIN
```

```
ORG 0050H
```

```
MAIN:
```

```
    MOV R0, #5H           ;個位數
```

```
    MOV R1, #0H           ;十位數
```

```
LOOP1:
```

```
    ACALL SHOW
```

```
    CJNE R0, #0H, DEC1
```

```
    MOV R0, #9H           ;借位
```

```
    AJMP LOOP2
```

```
DEC1:
```

```
    DEC R0
```

```
    AJMP LOOP1
```

```
LOOP2:
```

```
    CJNE R1, #0, DEC2
```

```
    MOV R1, #6H           ;下數到 00 時，重設至 60
```

```

    MOV R0, #0H
    AJMP LOOP1
DEC2:
    DEC R1
    AJMP LOOP1

SHOW:
    MOV R4, #0H           ;作為計數器讓數字重複顯示
    MOV R5, #0H           ;形成視覺暫留
SHOWLOOP:
;First digjt
    MOV DPTR, #POS        ;先選位數
    MOV A, #0H
    MOVC A, @A+DPTR
    MOV P2, A

    MOV DPTR, #NUMTABLE   ;再選數字
    MOV A, R0
    MOVC A, @A+DPTR
    MOV P1, A

    ACALL DELAY
    MOV P1, #0FFH         ;避免干擾下一位數的顯示
;Second digit
    MOV DPTR, #POS        ;先選位數
    MOV A, #1H
    MOVC A, @A+DPTR
    MOV P2, A

    MOV DPTR, #NUMTABLE   ;再選數字
    MOV A, R1
    MOVC A, @A+DPTR
    MOV P1, A

    ACALL DELAY
    MOV P1, #0FFH         ;避免干擾下一位數的顯示

    INC R4

```

```

    CJNE R4, #0B0H, SHOWLOOP
    MOV R4, #0
    INC R5
    CJNE R5, #03H, SHOWLOOP

    RET

;=====
DELAY:
    MOV R6, #0FAH
DELAY1:
    MOV R7, #0AH
DELAY2:
    DJNZ R7, DELAY2
    DJNZ R6, DELAY1
    RET

;=====
NUMTABLE:
    DB 0C0H ;0
    DB 0F9H ;1
    DB 0A4H ;2
    DB 0B0H ;3
    DB 099H ;4
    DB 092H ;5
    DB 082H ;6
    DB 0F8H ;7
    DB 080H ;8
    DB 090H ;9

;=====
POS:
    DB 0F7H, 0FBH, 0FDH, 0FEH

```

因為我的程式與範例所給的有些許不同，根據 MPC82G516A 的 Datasheet，我每顯示一組數字（兩位數）所需要的時間是：

$$Delay\ time(\mu s) = \frac{\left(\left((32 + (4 + 4R_7) \times R_6) \times 2 + 7\right) \times R_4 + 7\right) \times R_5}{12}$$

我選擇  $R_4$  為 180 (0xB5)、 $R_5$  為 3 (0x03)、 $R_6$  為 250 (0xFA)、 $R_7$  為 10



(0x0A)，這樣 Delay time 約為 0.9932 秒，再加上前面一些計算借位的程式，經過測試，計時一分鐘誤差在一秒以內（對照組是手機的計時器）。

#### 六、心得：

這次實驗使用的四個七段顯示器的板子，實際並無法同時顯示四個數字，所以要利用視覺暫留的效果讓四個顯示器「看起來」像是同時顯示。為了達到這個效果，需要增加很多類似迴圈的結構（每組數字需要重複使用數次），也必須要考慮到位置控制訊號切換時，如何不讓前一個位數去影響到下一個位數，整個過程實在是有點複雜有繁瑣，但成功完成後，也對這塊四顆七段顯示器的控制有更深的了解。

另外，進階題中的下數也對我是很大的挑戰，因為我不知道怎麼在 51 裡面使用負數，導致我在判斷是否要借位時發生了許多困難。總地來說，這次實驗相較於前幾次可說是難度三級跳。