



微算機實驗報告

Lab # 5

姓名：仇健安

系級：電機系

學號：111511239

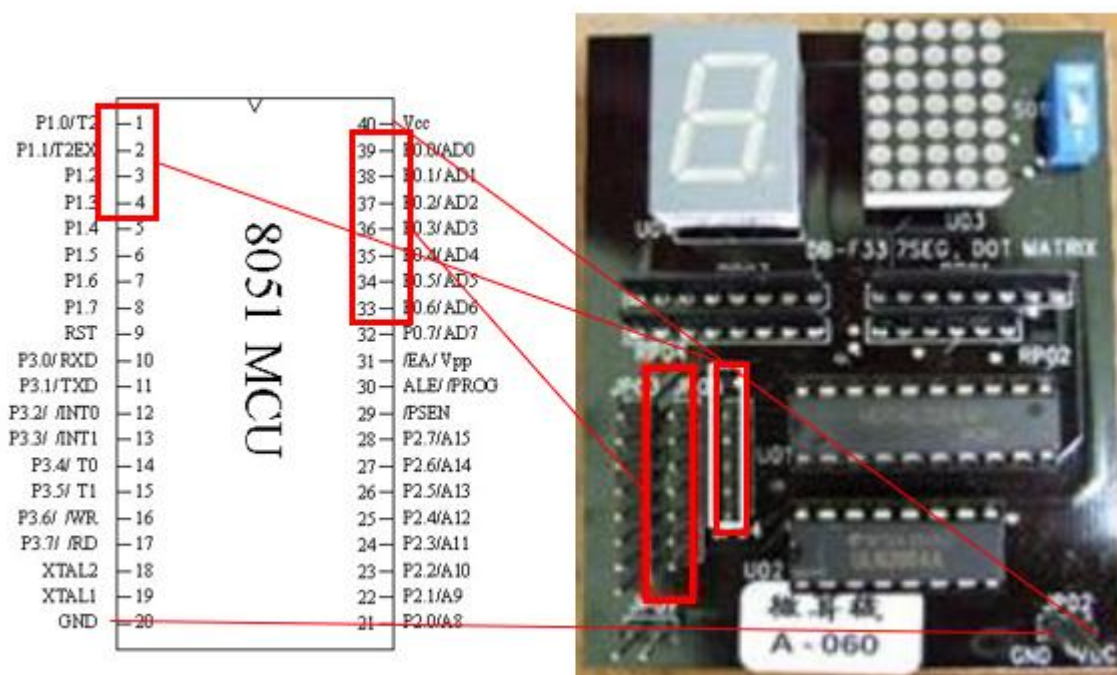
上課時間：2025/03/11

一、實驗目的：

本次實驗目的是了解 5*7 點矩陣 LED 的電路結構與其驅動原理，並實作不同圖樣的顯示，包含「大」、「中」、「小」字的切換動畫，以及依照指定順序掃描點矩陣 LED 的跑馬燈效果。透過本實驗可深入了解 74LS244 與 ULN2004 的角色，及其在共陽極與共陰極控制方式下的運作機制。

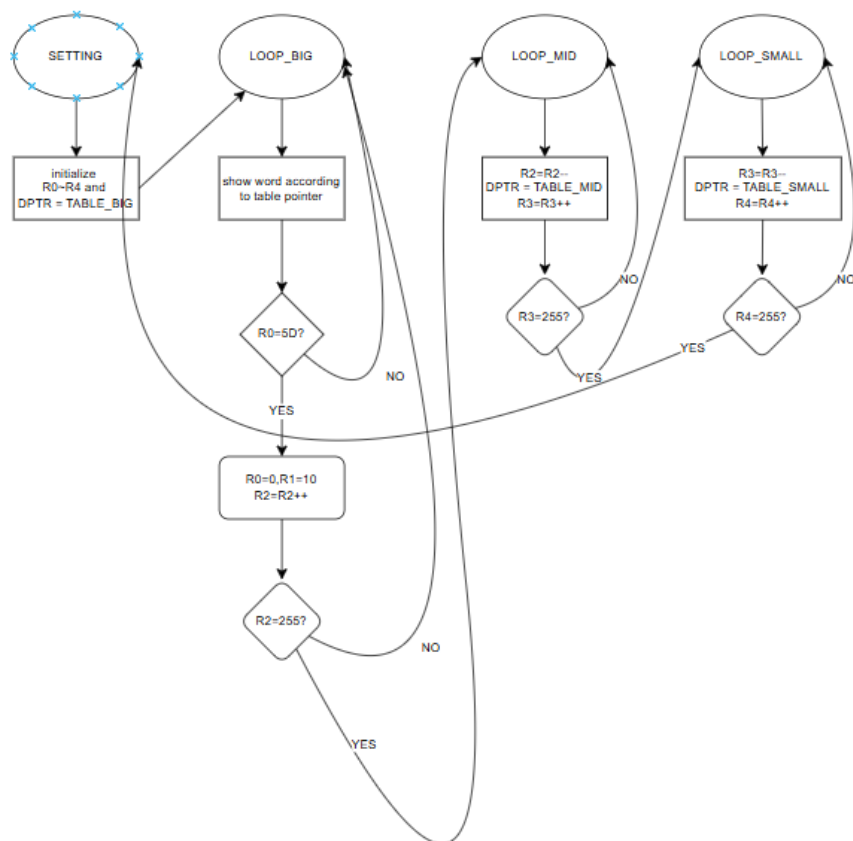
二、硬體架構：

本實驗採用七段顯示器與 5*7 點矩陣 LED 模組，矩陣的列由 74LS244 控制提供陽極電壓，行由 ULN2004 控制接地。程式利用 JP05 控制列(JP05=陽極=點亮列)，JP04 控制行(JP04=陰極=接地行)。當某列與某行同時為 1 時，該交點的 LED 便會亮起。

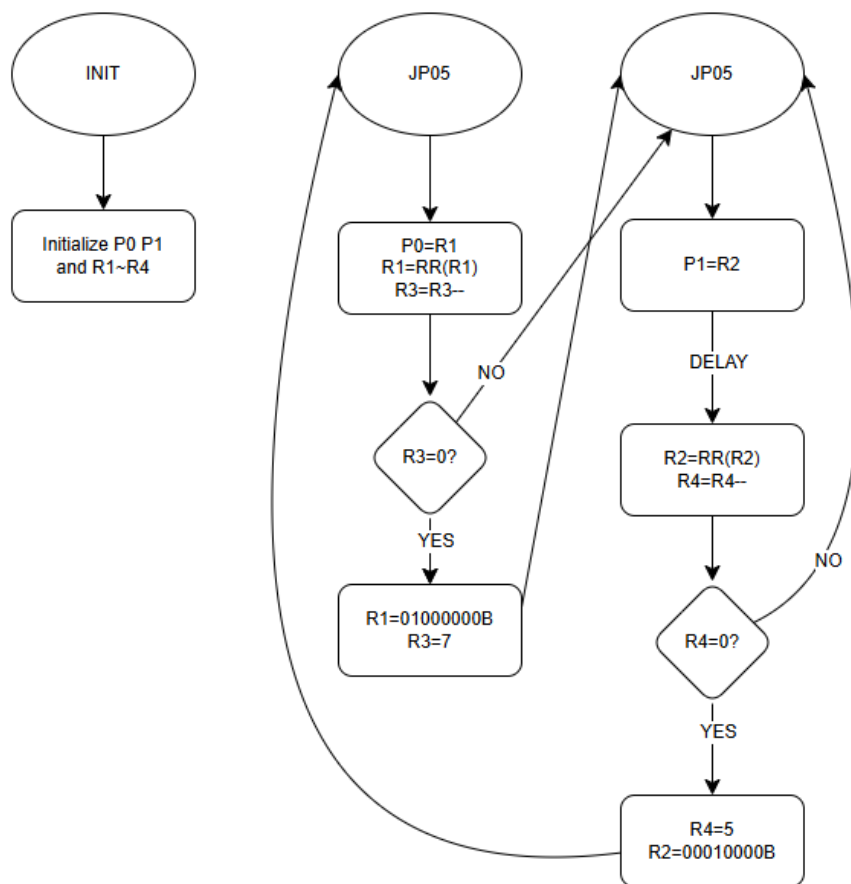


三、程式流程圖：

基本題：



進階題：



四、問題與討論：

(1) 程式中自行設計的 delay time，時間增加或減少會有什麼影響？

delay time 決定了單一 LED 切換閃爍的速度。若 delay 設太短，文字或跑馬燈動畫會變得太快，難以辨識；若設太長，可能會閃爍不連貫導致文字無法正確顯現，因為超過視覺暫留時間。

(2) 請舉例生活中，與 LED 點矩陣具有相同驅動方式的產品。

如電子看板（公車站牌、商業廣告牆）、老式電子時鐘、交通燈或運動場地大型的 LED 顯示屏，它們也使用點矩陣方式掃描顯示字元與圖形，並透過高速切換與視覺暫留達成持續發亮的錯覺。

五、程式碼與註解：

基本題：

ORG 0000H

JMP SETTING ; 程式啟動時跳到 SETTING，開始初始化與主迴圈

ORG 0050H ; 主程式區段開始

SETTING:

MOV R0, #0 ; R0 為目前顯示的資料索引（字元掃描）
MOV R1, #010H ; R1 為行掃描初始值（控制哪一行亮）
MOV R2, #0 ; R2 為切換到下一階段的計數器（字型輪替用）
MOV R3, #0 ; R3 為中等字的顯示次數計數
MOV R4, #0 ; R4 為小字的顯示次數計數
MOV DPTR, #TABLE_BIG ; 初始顯示 TABLE_BIG（大字）

LOOP_BIG:

MOV P1, #00H ; 清除 P1 行控制（避免殘留）
MOV A, R0 ; 把目前行數放進 A
MOVC A, @A+DPTR ; 取出該行要顯示的資料（從表中）
MOV P0, A ; 將資料輸出到 P0（列控制）
MOV P1, R1 ; 選擇要顯示的行
INC R0 ; 下一行
MOV A, R1
RR A ; 循環右移行掃描位置（每次掃一行）
MOV R1, A ; 更新行控制碼
CALL DELAY ; 延遲讓LED點亮時間足夠
CJNE R0, #5D, LOOP_BIG ; 顯示完所有字元前持續顯示（共 5D 次）

MOV R0, #0 ; 顯示完後歸0重新開始
MOV R1, #10H ; 行選擇初始回復
INC R2 ; 顯示次數+1
CJNE R2, #255, LOOP_BIG ; 若未滿255次，繼續顯示文字

LOOP_MID:

```
DEC R2                ; 清掉R2 (避免干擾)，進入中字循環
MOV DPTR, #TABLE_MID; 切換字型為 TABLE_MID (中)
INC R3                ; 中字顯示次數+1
CJNE R3, #255, LOOP_BIG; 顯示 255 次後再切換
```

LOOP_SMALL:

```
DEC R3                ; 清掉R3，進入小字循環
MOV DPTR, #TABLE_SMALL; 切換字型為 TABLE_SMALL (小)
INC R4                ; 小字顯示次數+1
CJNE R4, #255, LOOP_BIG; 顯示 255 次後再重頭開始
```

```
JMP SETTING          ; 所有顯示結束後重新設定再循環
```

; 字型資料表：共5行，用於逐行掃描顯示

TABLE_BIG:

```
DB 00010010B          ; 用二進位定義每一列顯示的點
DB 00010100B
DB 01111000B
DB 00010100B
DB 00010010B
```

TABLE_MID:

```
DB 00111110B
DB 00100010B
DB 01111111B
DB 00100010B
DB 00111110B
```

TABLE_SMALL:

```
DB 00011010B
DB 00000001B
DB 01111110B
DB 00000000B
DB 00011000B
```

; 延遲子程式：控制顯示停留時間

DELAY:

```
MOV R6, #20D          ; 外層延遲次數
```

DELAY1:

```
MOV R7, #50D          ; 內層延遲次數
```

```

DELAY2:
    DJNZ R7, DELAY2      ; 等 R7 倒數為0
    DJNZ R6, DELAY1      ; 等 R6 倒數為0
    RET                  ; 返回主程式
END

```

進階題:

```

ORG 0000H
JMP INIT          ; 程式起始，跳轉到 INIT 開始初始化與主程式邏輯

```

```

ORG 0050H

```

```

INIT:                ; 初始化區

```

```

    MOV P0,#01000000B    ; 設定 P0 初值，控制列輸出 (JP05，控制點矩陣 LED 的列)
    MOV P1,#00010000B    ; 設定 P1 初值，控制行輸出 (JP04，控制點矩陣 LED 的行)

```

```

    MOV R1,#01000000B    ; R1 控制列，初始為最左邊的列 (bit6為1)
    MOV R2,#00010000B    ; R2 控制行，初始為第3行 (bit4為1)

```

```

    MOV R3,#7            ; 計數器 R3 控制掃描 7 行 (7 次)
    MOV R4,#5            ; 計數器 R4 控制掃描 5 列 (5 次)

```

; 進入 JP05 掃描列 (P0控制)

```

JP05:

```

```

    MOV P0, R1           ; 將目前的列控制碼送出到 P0 (點亮該列的陽極)
    MOV A, R1
    RR A                 ; 將 R1 的位元右移一位 (移到下一列)
    MOV R1, A
    DEC R3               ; 行數 -1 (這裡控制 JP04 顯示幾行)
    CJNE R3, #0, JP04    ; 如果 R3 ≠ 0，繼續執行 JP04 (換行)

```

; 如果 R3 == 0 (行掃描完)，重設列與行

```

    MOV R1, #01000000B    ; 重設列控制碼為第一列
    MOV R3, #7            ; 重設行數為7行 (重新掃描所有行)

```

; 進入 JP04 掃描行 (P1控制)

```

JP04:

```

```

    MOV P1, R2           ; 將目前的行控制碼送出到 P1 (點亮該行的陰極)
    CALL DELAY           ; 等待一段時間，讓該點顯示穩定

```

```

    MOV A, R2

```

```

RR A                ; 將 R2 的位元右移一位 (移到下一行)
MOV R2, A
DEC R4              ; 列數 -1 (每列會跑5行)
CJNE R4, #0, JP04    ; 若 R4 ≠ 0, 繼續跑 JP04 換行掃描

MOV R4, #5          ; 若 R4 == 0, 表示該列5行掃完, 重設為5行
MOV R2, #00010000B  ; 重設行控制碼為初始行
AJMP JP05           ; 跳回 JP05 換下一列掃描

```

; 延遲子程式, 讓LED點亮時間夠久以達到視覺暫留

DELAY:

```
MOV R5, #0FFH      ; 外層延遲計數器
```

DELAY1:

```
MOV R6, #055H      ; 中層延遲計數器
```

DELAY2:

```
MOV R7, #10H       ; 內層延遲計數器
```

DELAY3:

```
DJNZ R7, DELAY3     ; 最內層延遲迴圈
```

```
DJNZ R6, DELAY2     ; 中層
```

```
DJNZ R5, DELAY1     ; 外層
```

```
RET                ; 延遲結束, 返回呼叫點
```

END ; 程式結束

六、心得：

上課心得：

老師在課堂中清楚說明點矩陣的控制邏輯與硬體設計, 尤其是 ULN2004 與 74LS244 如何控制行列的概念, 讓我快速理解程式的作用與電路互動方式。

除此之外老師還介紹了 8051 8052bank 的設計, 讓使用者可以使用更多暫存的位置。

實驗心得：

親手完成字型動畫與掃描控制, 讓我對點矩陣 LED 的驅動原理有更深體會。能夠控制顯示不同字與圖案, 並實作出動畫效果, 讓我很有成就感。

Notes:

1. 內容字體大小為 12, 間距為單行間距
2. 中文字字體為標楷體
3. 英文字和阿拉伯數字為 Times New Roman
4. 嚴禁抄襲, 抄襲者以 0 分計算
5. 請於報告左上角附上照片
6. 每次實驗課繳交上次實驗結報