實驗十一 步進馬達

1. 學習重點

- 了解步進馬達的運作原理
- 了解介面電路設計的觀念

2. 材料清單

表 11-1、材料清單

器材名稱		數量
AT89S51		1
12MHz 石英震盪器		1
LED 二極體		4
按壓開關		1
1N4001 二極體		4
電晶體 TIP102		4
反向器 SN74LS04N		1
步進馬達 28BYJ-48 5V		1
電阻	1kΩ	4
	100Ω	4
	10 kΩ	1
電容	20pF	2
	10uF	1

3. 元件原理

步進馬達 Step Motor

步進馬達是常見於各類工業產品上的元件，例如 3D 列印機的噴頭即是利用步進馬達控制的，因步進馬達的控制原理，使得它相比於傳統直流馬達，更能做出精密的控制，以下為步進馬達的原理介紹。

步進馬達的結構主要由兩部分磁鐵構成，負責轉動的永久磁鐵，轉子；以

及負責驅動轉子、固定不動，僅改變極性的電磁鐵，定子。如下圖 11-1 所示。控制馬達的方式便是透過輸入訊號給定子，使其改變極性，利用磁力控制轉子轉動，由此方式控制轉子能使每次旋轉固定的角度，故名為步進馬達。

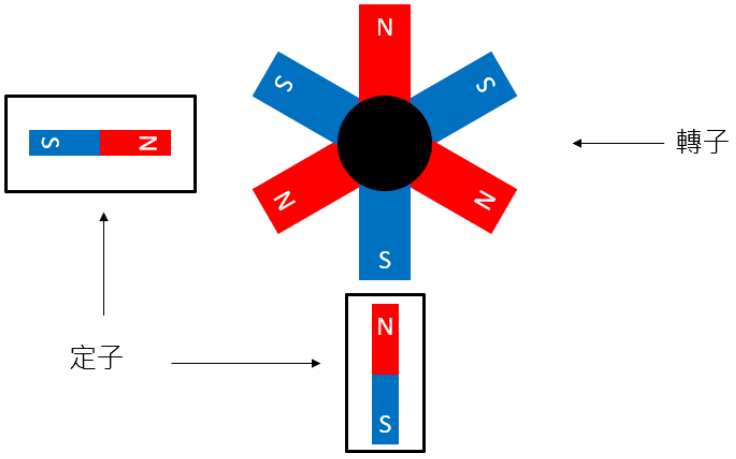

圖 11-1、步進馬達內部結構

若是使用不同的順序輸入訊號給馬達，會使馬達以不同的角度與力矩旋轉，根據不同馬達的定子數量，輸入的方式也會有所不同。以本次實驗使用的馬達為例，此型馬達有兩個定子，每個定子有兩條訊號線用以控制，定子 A 的控制線分別為 A 與 $\overline{A}$，定子 B 的為 B 與 $\overline{B}$，以及一條控制馬達開關的 COM 控制線，故為五線二相步進馬達，此類馬達驅動方式便有 1 相、2 相與 1.5 相驅動法。

1 相驅動顧名思義，便是一次僅極化一個定子，並每次改變極化的定子與磁性，也就是依序給予控制線 A、B、$\overline{A}$、$\overline{B}$ 高電位訊號，而 8051 輸出的訊號則需要視 8051 與馬達的連接狀況調整，若要改變旋轉方向，僅需將訊號的輸出的順序顛倒即可。

2 相驅動則是一次極化兩個定子，並每次改變一個極化定子的磁性，也就是依序給予控制線 AB、B$\overline{A}$、$\overline{A}\overline{B}$、$\overline{B}$A 高電位訊號，同樣 8051 輸出的訊號則需要視 8051 與馬達的連接狀況調整，若要改變旋轉方向，僅需將訊號的輸出的順序顛倒即可。2 相驅動相比於 1 相驅動有著更強的力矩，相對的也有著較大個電能需求，因此應視需求決定驅動方式。

最後的 1.5 相驅動則是將 1 相與 2 相驅動的訊號交錯輸出，也就是依序給予控制線 A、AB、B、B$\overline{A}$、$\overline{A}$、$\overline{A}\overline{B}$、$\overline{B}$、$\overline{B}$A 高電位訊號。此種驅動方式的特色為旋轉角度僅有其他兩種驅動方式的一半，故能做到更精細的控制。

4. 實驗內容

利用 8051 的 GPIO 輸出數位訊號，驅動步進馬達，並利用軟體控制馬達的驅動方式與方向。

5. 實驗電路圖

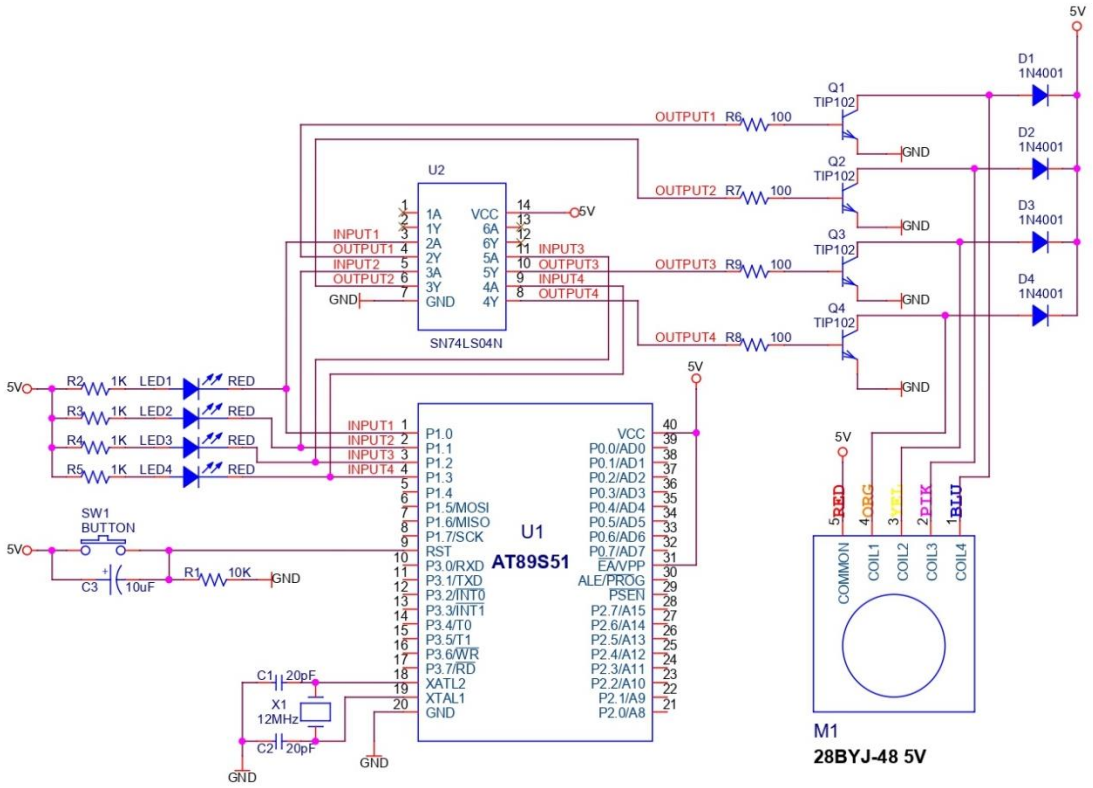

圖 11-2、實驗十一基礎題參考電路圖

此電路中所使用的 NOT 閘用途是作為緩衝器使用，便藉此控制電晶體的狀態，因 8051 GPIO 輸出的電流遠遠不足以驅動馬達，因此需要使用類似於實驗二的方式來驅動馬達。而與馬達連接的二極體用處為防止馬達停止時自然產生的脈衝回流，導致電晶體損壞。

6. 軟體流程圖

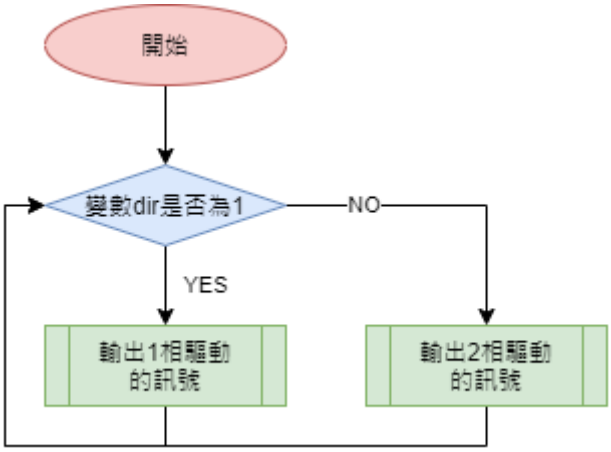

圖 11-3、實驗十一基礎題參考軟體流程圖

7. 範例程式碼

```
#include <regx51.h>
void delay(int);
void turn( );
code char one_phase[ ] = {0x01,0x02,0x04,0x08};
code char two_phase[ ] = {0x0c,0x06,0x03,0x09};
int dir , delay_time;

void main( )
{
   delay_time = 3000;
   dir = 1;
   while (1)
   {
      turn( );
   }
}

void turn( )
{
   int x;
   if (dir == 1) //one phase, turn right
   {
      for (x = 0; x < 4; x++)
      {
         P1 = one_phase[x];
         delay(delay_time);
      }
   }
   else //two phase, turn left
```

```
30      {
31          for(x = 0; x < 4; x++)
32          {
33              P1 = two_phase[x];
34              delay(delay_time);
35          }
36      }
37  }
38
39  void delay(int t)
40  {
41      while(t--);
42  }
```

8. 整理的題目，選擇/是非題