**電通二甲微處理器實驗 實驗結報**

|  |  |  |  |
| --- | --- | --- | --- |
| **實驗名稱** | 跑馬燈 | | |
| **組別** | 27 | **組員** | 04050015陳姿華 04052502許書瑜 |

1. **實驗目的**

Arduino UNO D1 –D8 分別接到LED，實作跑馬燈展示。

1. **實驗步驟**

LED 向左及向右執行花色展示。

執行自定花色展示: 所有LED亮滅兩次-> 右移一次 -> 所有LED亮滅兩次

->左移一次。

1. Arduino接上一個開關

2. 開關OFF -> LED 向左及向右執行花色展示

3. 開關ON -> 執行自定花色展示

\* 注意事項: 需完成防彈跳程式碼

1. **程式碼**

**LED 向左及向右執行花色展示(課堂PDF提供的程式碼)**

// Lab02\_1 // 使用for迴圈的跑馬燈程式 // // 每個LED 正極接到Pin 0 -7, 負極接地

constbyte startPin= 1;

constbyte endPin= 8;

void setup()

{

// 將每個Pin 設定成輸出

for (byte i= startPin; i<= endPin; i++)

{

pinMode(i, OUTPUT);

}

}

void loop()

{

byte i; byte lightPin= startPin;

// 所有LED OFF

for (i=startPin; i<= endPin; i++)

{

digitalWrite(i,LOW);

}

// 設定要點亮的LED 位置

// 如果點亮的腳位編號小於結束腳位, 則腳位編號+1 // 否則從頭開始

if (lightPin< endPin)

{

lightPin++;

}

else

{

lightPin= startPin;

}

// 點亮LED

digitalWrite(lightPin, HIGH);

1. **實驗結果及分析**

要形成跑馬燈必須要先讓所有的LED先關閉，再由左而右或由右而左

讓LED燈依序一顆一顆亮過去。

在這次實驗當中最困難的就是解決開關的接線、開關造成的彈跳訊號所導致的延遲問題。

1. **心得討論**

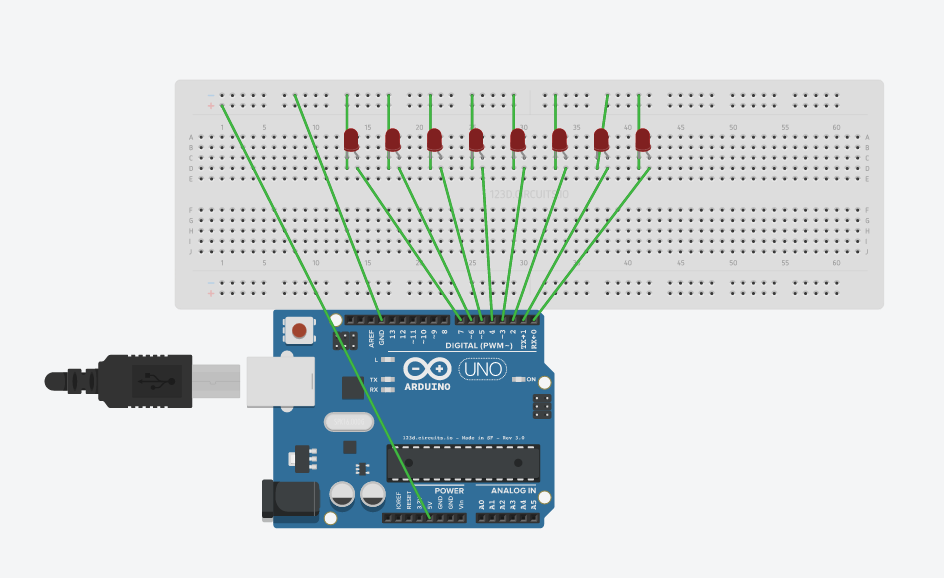
這一周的微處理機實驗比起上次的實驗明顯難了不少，開關去控制跑馬燈的程式和電路接線讓我們思考了很長一時間，尤其是開關彈跳訊號所導致的延遲，真的不太容易解決。

1. **修正電路圖**

**1. LED 向左及向右執行花色展示**

**2. 執行自定花色展示: 所有LED亮滅兩次-> 右移一次 -> 所有LED亮滅兩次**

**->左移一次。**

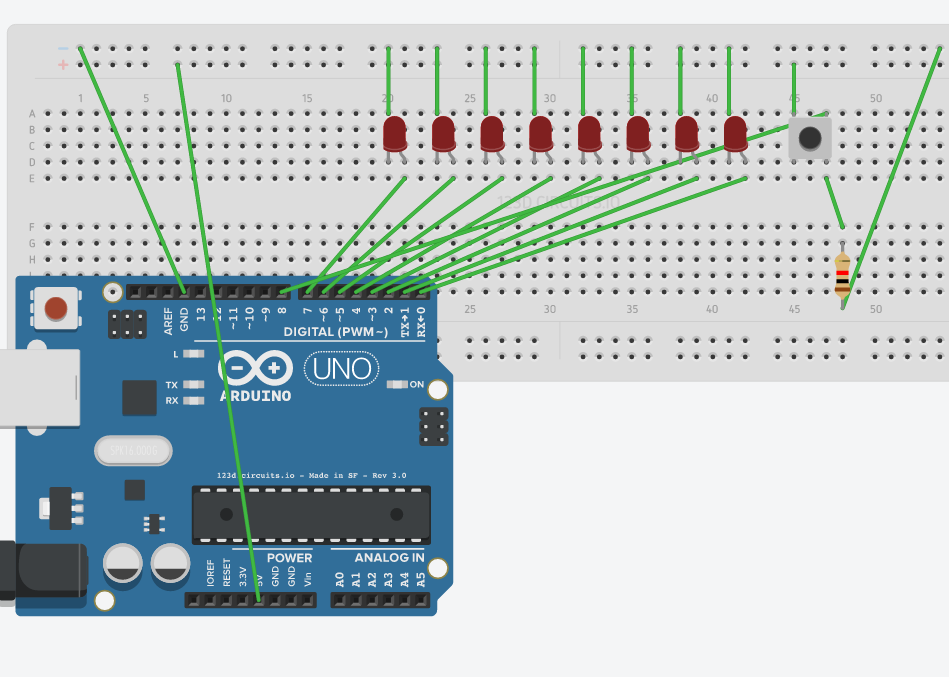


\* 麵包版的正接5V，負接地

\* LED燈的正端分別接在Arduino的D0~D7

負則接麵包版的負

**開關OFF -> LED 向左及向右執行花色展示 開關ON -> 執行自定花色展示**



**開關控制**

**7.修正程式碼**

**1. LED 向左及向右執行花色展示**

// Lab02\_1

// 使用for迴圈的跑馬燈程式

//

// 每個LED 正極接到Pin 0 - 7, 負極接地

const byte startPin = 0;

const byte endPin = 7;

void setup()

{

// 將每個Pin 設定成輸出

for (byte i = startPin; i<= endPin; i++)

{

pinMode(i, OUTPUT);

}

}

void loop()

{

byte j;

byte i;

byte lightPin = endPin;

// 所有LED OFF

for (i=startPin; i<= endPin; i++)

{

digitalWrite(i,LOW);

}

for (i=endPin; i>= startPin; i--)

{

// 設定要點亮的LED 位置

// 如果點亮的腳位編號小於結束腳位,

//則腳位編號+1

// 否則從頭開始

digitalWrite(lightPin, HIGH);

delay(100);

digitalWrite(lightPin, LOW);

delay(100);

if (lightPin > startPin)

{

lightPin --;

}

else

{

break;

}

}

lightPin = startPin;

for (i=startPin;i<=endPin; i++)

{

digitalWrite(lightPin, LOW);

}

for (i=startPin; i<= endPin; i++)

{

// 設定要點亮的LED 位置

// 如果點亮的腳位編號小於結束腳位,

//則腳位編號+1

// 否則從頭開始

digitalWrite(lightPin, HIGH);

delay(100);

digitalWrite(lightPin, LOW);

delay(100);

if (lightPin < endPin)

{

lightPin ++;

}

else

{

break;

}

}

}

**2. 執行自訂花色展示**

const byte startPin = 0;

const byte endPin = 7;

void setup()

{

// 將每個Pin 設定成輸出

for (byte i = startPin; i<= endPin; i++)

{

pinMode(i, OUTPUT);

}

}

void loop()

{

byte j;

byte i;

byte lightPin = endPin;

// 所有LED OFF

for(byte j=1;j<=2;j++)

{

for (i=startPin; i<= endPin; i++){

digitalWrite(i,HIGH);

}delay(100);

for (i=startPin; i<= endPin; i++){

digitalWrite(i,LOW);

}delay(100);

}

for (i=endPin; i>= startPin; i--)

{

// 設定要點亮的LED 位置

// 如果點亮的腳位編號小於結束腳位,

//則腳位編號+1

// 否則從頭開始

digitalWrite(lightPin, HIGH);

delay(100);

digitalWrite(lightPin, LOW);

delay(100);

if (lightPin > startPin)

{

lightPin --;

}

else

{

break;

}

}

for(j=1;j<=2;j++)

{

for (i=startPin; i<= endPin; i++)

{

digitalWrite(i,HIGH);

} delay(100);

for (i=startPin; i<= endPin; i++)

{

digitalWrite(i,LOW);

}delay(100);

}

lightPin = startPin;

for (i=startPin;i<=endPin; i++)

{

digitalWrite(lightPin, LOW);

}

for (i=startPin; i<= endPin; i++)

{

// 設定要點亮的LED 位置

// 如果點亮的腳位編號小於結束腳位,

//則腳位編號+1

// 否則從頭開始

digitalWrite(lightPin, HIGH);

delay(100);

digitalWrite(lightPin, LOW);

delay(100);

if (lightPin < endPin)

{

lightPin ++;

}

else

{

break;

}

}

}

**3. 開關OFF -> LED 向左及向右執行花色展示，開關ON -> 執行自定花色展示。**

// Lab03

// 使用for迴圈的跑馬燈程式

//

// 每個LED 正極接到Pin 0 - 7, 負極接地

const int sw=8;

const int debounceDelay=20;

const byte startPin = 0;

const byte endPin = 7;

int val;

int i;

int direct=0;

void setup()

{

pinMode(sw,INPUT);

digitalWrite(sw,HIGH);

// 將每個Pin 設定成輸出

for (byte i = startPin; i<= endPin; i++)

{

pinMode(i, OUTPUT);

}

}

void loop()

{

byte j;

for (i=startPin; i<= endPin; i++){

digitalWrite(i,LOW);

}

val=digitalRead(sw);

if(val==LOW)

{

delay(debounceDelay);

while(digitalRead(sw)==LOW);

direct=!direct;

}

byte lightPin = endPin;

// 所有LED OFF

for(byte j=1;j<=2;j++)

{

for (i=startPin; i<= endPin; i++){

digitalWrite(i,HIGH);

}delay(100);

for (i=startPin; i<= endPin; i++){

digitalWrite(i,LOW);

}

delay(100);

}

{

for (i=endPin; i>= startPin; i--)

{

// 設定要點亮的LED 位置

// 如果點亮的腳位編號小於結束腳位,

//則腳位編號+1

// 否則從頭開始

digitalWrite(lightPin, HIGH);

delay(100);

digitalWrite(lightPin, LOW);

delay(100);

if (lightPin > startPin)

{

lightPin --;

}

else

{

break;

}

}

for(j=1;j<=2;j++)

{

for (i=startPin; i<= endPin; i++)

{

digitalWrite(i,HIGH);

} delay(100);

for (i=startPin; i<= endPin; i++)

{

digitalWrite(i,LOW);

}delay(100);

}

lightPin = startPin;

for (i=startPin;i<=endPin; i++)

{

digitalWrite(lightPin, LOW);

}

for (i=startPin; i<= endPin; i++)

{

// 設定要點亮的LED 位置

// 如果點亮的腳位編號小於結束腳位,

//則腳位編號+1

// 否則從頭開始

digitalWrite(lightPin, HIGH);

delay(100);

digitalWrite(lightPin, LOW);

delay(100);

if (lightPin < endPin)

{

lightPin ++;

}

else

{

break;

}

}

}

}