**電通二甲微處理器實驗 實驗結報**

|  |  |  |  |
| --- | --- | --- | --- |
| **實驗名稱** | 四合一七段顯示器 | | |
| **組別** | **25** | **組員** | 04050033梁艾蓉/04050475 胡皓雯 |

1. **實驗目的**

了解七段顯示器之顯示原理

1. 共陰極與共陽極七段顯示器之不同?

2. 如何讓七段顯示器顯示 0 – F?

3. 如何控制四合一七段顯示器?

4. 如何讓四合一七段顯示器顯示四位數字?

1. **實驗步驟**

實驗步驟1

\* Arduino 接好四合一七段顯示器之電路

\* 四合一七段顯示器每隔 0.3 秒鐘會跳一個數字, 從 0000 -> 0059 -> 0100 -> 跳至 2359 後歸零

實驗步驟2

\* 於 Pin 10 接一個按鍵, 按下後歸零顯示

\* 於 Pin 11 接一個按鍵, 按下後倒數計時

實驗步驟3

合併歸零及倒數計時

1. **程式碼**

四合一七段顯示器每隔 0.3 秒鐘會跳一個數字, 從 0000 -> 0059 -> 0100 -> 跳至 2359

sevseg.setNumber(Hour \* 100 + Min, 0);

Min = Min + 1;

If Min == 60 { Min = 0, If (Hour == 23) Hour=0 else Hour++ } sevseg.refreshDisplay();

sleep 0.3 second

1. **實驗結果及分析**
2. **會做出一個24小時制的時鐘。並加入2個按鈕，一個會將時間歸零，另一個會將時間做倒數。**
3. **將github上的壓縮檔加入程式中，要用Include Library。**
4. **供陰極和供陽極電路的接線腳位不同。**
5. **七段顯示器的3、8腳位負責接地。**
6. **心得討論**

**在這次的實驗中，我們學會了如何在Arduino中加入Github網站上的壓縮檔。**

**還有學會七段顯示器1-12腳位分別是在哪邊，如何讓七段顯示器顯示 0 – F?**

**，還有按鈕該如何接到類比腳位。**

1. **修正程式碼**
2. **時鐘**

**#include "SevSeg.h"**

**SevSeg sevseg;**

**int Min = 0 , Hour = 0 , t = 0;**

**void setup( )**

**{**

**Serial.begin(9600);**

**byte numDigits = 4;**

**byte digitPins[ ] = {5 , 4 , 3 , 2};**

**byte segmentPins[ ] = {6 , 7 , 8 , 9 , 10 , 11 , 12 , 13};**

**sevseg.begin(COMMON\_CATHODE , numDigits , digitPins , segmentPins);**

**}**

**void loop( )**

**{**

**sevseg.setNumber(Hour\*100 + Min , 4);**

**for(int t=0;t<10000;t++)**

**{**

**sevseg.refreshDisplay( );**

**}**

**Min = Min + 1;**

**if (Min == 60)**

**{**

**Min = 0;**

**if(Hour == 23)**

**Hour = 0;**

**else**

**Hour = Hour+1;**

**}**

**}**

1. [**時鐘合併歸零與倒數**](https://github.com/Ivy000000/Lab04/commit/6bfedde4df895173158d12fd0d2dfeae59124a66)

**#include "SevSeg.h"**

**SevSeg sevseg;**

**int Hour=0,Min=0,Step=1;**

**int Pin10,Pin11;**

**void setup( )**

**{**

**pinMode(A4,INPUT\_PULLUP);**

**pinMode(A5,INPUT\_PULLUP);**

**byte numDigits = 4;**

**byte digitPins[] = {5,4,3,2};**

**byte segmentPins[] = {6, 7, 8, 9, 10, 11, 12, 13};**

**sevseg.begin(COMMON\_CATHODE, numDigits, digitPins,segmentPins);**

**}**

**void loop( )**

**{**

**Pin10= digitalRead(A4);**

**Pin11= digitalRead(A5);**

**if(Pin10 == 0)**

**Min = Hour =0;**

**if(Pin11==0)**

**Step = -1;**

**else**

**Step =1;**

**sevseg.setNumber( Hour\*100+Min, 4);**

**Min = Min + Step;**

**if(Min == 60)**

**{**

**Min = 0;**

**Hour = Hour+1;**

**if(Hour == 24)**

**Hour=0;**

**}**

**if(Min == -1)**

**{**

**Min=59;**

**Hour = Hour-1;**

**if(Hour == -1)**

**Hour=23;**

**}**

**for(int i=0;i<30000;i++)**

**{**

**sevseg.refreshDisplay();**

**}**

**}**

**\*影片網址：**

**https://github.com/Ivy000000/Lab04/blob/master/IMG\_3518.MOV?raw=true**

**電路圖**

1. **程式碼**

於 Pin 10 接一個按鍵, 按下後歸零顯示

Pin10= DigitalRead (D10) // Reset

If (Pin10 == LOW)

Min = Hour = 0;

sevseg.setNumber(Hour \* 100 + Min, 0)

Min = Min + 1;

If Min == 60 { Min = 0, If (Hour == 23) Hour=0 else Hour++ } sevseg.refreshDisplay();

sleep 0.3 second

於 Pin 11 接一個按鍵, 按下後倒數計時

Pin10= DigitalRead (D10) // Reset

Pin11 = DigitalRead (D11) // Reverse

If (Pin10 == LOW)

Min = Hour = 0;

If (Pin11 ==LOW)

step = -1 // Reverse

else step = 1

sevseg.setNumber(Hour \* 100 + Min, 0)

Min = Min + step;

If Min == 60 {…} // overflow

If Min == -1 { Min = 59, If (Hour == -1) Hour = 23 else Hour-- } // underflow

sevseg.refreshDisplay();

sleep 0.3 second