

# 秀林國中技藝班講義

114年第一學期

## 講師資訊

講師

郭德潤

上課網址

www.myiot.idv.tw

上課地點、時間

花工資訊科, 星期五下午

## 期望與目標

期望可以使用人工智慧輔助完成：

- Arduino UNO程式撰寫
- 透過processing 達成電腦與Arduino 板子的連線

## 課程教材

必備教材：<https://www.myiot.idv.tw>

國中技藝班-秀林國中

選用教材：**Gemini**

工作任務：

任務 主題

閱讀

完成打勾

1 讓AI了解硬體連接

講義範例1

2

3

操作技巧：使用開始處點選、到要複製的最後shift+左鍵，可以一次選擇多行。

ctrl+a：全選。ctrl+c：複製。ctrl+v：貼上。ctrl+z：回複上一步。

技藝班座號：

姓名：

第 1 頁

## 任務1:

複製下列指示，貼到gemini讓它分析：使用點選、shift+左鍵，ctrl+c

【

請解析以下的程式：

```
const byte LED[] = {13,12,11,10};

#define VarResistor A5

#define BUTTON1 A1

#define BUTTON2 A2

#define BUTTON3 A3

int PotValue;

/* Define shift register pins used for seven segment display */

#define LATCH_DIO 4

#define CLK_DIO 7

#define DATA_DIO 8

/* Segment byte maps for numbers 0 to 9 */

const byte SEGMENT_MAP[] = {0xC0,0xF9,0xA4,0xB0,0x99,0x92,0x82,0xF8,0X80,0X90};

/* Byte maps to select digit 1 to 4 */

const byte SEGMENT_SELECT[] = {0xF1,0xF2,0xF4,0xF8};

void setup ()

{

// initialize the digital pin as an output.

/* Set each pin to outputs */

pinMode(LED[0], OUTPUT);

pinMode(LED[1], OUTPUT);

pinMode(LED[2], OUTPUT);

pinMode(LED[3], OUTPUT);

/* Set DIO pins to outputs */

pinMode(LATCH_DIO,OUTPUT);

pinMode(CLK_DIO,OUTPUT);
```

```
pinMode(DATA_DIO,OUTPUT);
Serial.begin(9600);
}
/* Main program */
void loop()
{
  if(!digitalRead(BUTTON1))
  {
    digitalWrite(LED[0], HIGH);
    digitalWrite(LED[1], HIGH);
    digitalWrite(LED[2], HIGH);
    digitalWrite(LED[3], HIGH);
  }
  if(!digitalRead(BUTTON2))
  {
    digitalWrite(LED[0], LOW);
    digitalWrite(LED[1], LOW);
    digitalWrite(LED[2], LOW);
    digitalWrite(LED[3], LOW);
  }
  PotValue = analogRead(VarResistor);
  Serial.print("Potentiometer reading: ");
  Serial.println(PotValue);
  /* Update the display with the current counter value */
  WriteNumberToSegment(0 , PotValue / 1000);
  WriteNumberToSegment(1 , (PotValue / 100) % 10);
  WriteNumberToSegment(2 , (PotValue / 10) % 10);
  WriteNumberToSegment(3 , PotValue % 10);
  /* Update the display with the current counter value */
```

```

// WriteNumberToSegment(0 , 0);
// WriteNumberToSegment(1 , 1);
// WriteNumberToSegment(2 , 2);
// WriteNumberToSegment(3 , 3);
/* Wait 0.5 seconds before reading again */
}
/* Write a decimal number between 0 and 9 to one of the 4 digits of the display */
void WriteNumberToSegment(byte Segment, byte Value)
{
    digitalWrite(LATCH_DIO,LOW);
    shiftOut(DATA_DIO, CLK_DIO, MSBFIRST, SEGMENT_MAP[Value]);
    shiftOut(DATA_DIO, CLK_DIO, MSBFIRST, SEGMENT_SELECT[Segment] );
    digitalWrite(LATCH_DIO,HIGH);
}

```

Gemini分析結果：

1. 硬體定義與初始化：

LED 與按鈕：定義了\_\_\_\_\_ 顆 LED (接腳 \_\_\_\_\_ - \_\_\_\_\_) 與 \_\_\_\_\_ 個按鈕 (A1-A3)。

七段顯示器 (透過 \_\_\_\_\_ 74HC595 控制)：

LATCH\_DIO (接腳 \_\_\_\_\_), CLK\_DIO (接腳 \_\_\_\_\_), DATA\_DIO (接腳 \_\_\_\_\_) 是控制移位暫存器的接腳。

SEGMENT\_MAP：存儲了數字 0-9 對應的共陽極 (Common Anode) 編碼 (例如 0xC0 為數字 0)。

SEGMENT\_SELECT：選擇要點亮哪一個位數 (Digit 1-4)。

潛在問題與改進建議

顯示閃爍問題：在 loop 中連續呼叫四次 WriteNumberToSegment, 但沒有適當的 delay 或「顯示掃描機制」。在真實硬體上, 這會導致四個位數看起來像是在同時顯示, 但亮度可能 \_\_\_\_\_, 或是只有 \_\_\_\_\_ 最亮。通常會建議在每一位顯示後加入極短的延遲 (如 delay(2);)。

任務2: 解決顯示的問題

【幫我改寫為更有效率的「多工掃描 (Multiplexing)」架構】

任務3: 增加讀取類比數值, 並理解人體感應電場。

【請增加讀取A5的數值並顯示在七段顯示器上】。

任務4: 啟動processing

【使用processing 寫一個人機介面, 具備Slider Control。拉動Slider Control值從0-1023, 在螢幕上顯示出數值】

任務5: 透過processing完成資料傳送功能。

【將processing 的Slider Control值透過USB傳遞到arduino 並顯示在七段顯示器上】

注意事項: // 如果 Serial.list()[0] 不對, 請手動改成 "COM3" 或改成Serial.list()[1]等

任務6: 解決processing值停止變化時, 顯示0000的問題

【當Slider Control值停止變化後, arduino的值會變成0000】

任務7: 完成Arduino與電腦的雙向資料傳送

【增加將Arduino A5讀到的數值透過USB傳到processing並在螢幕上畫出曲線圖】