



南臺科技大學
Southern Taiwan University of Science and Technology

減碳羅漢

古都土城仔綠電創能與智動養殖
之跨界整合永續淨零發展計畫

溶氧感測器

序列埠監控視窗



目錄



材料

撰寫程式步驟

接線說明

寫入程式步驟

校正方法

查看資訊



減碳資源

材料



ESP32

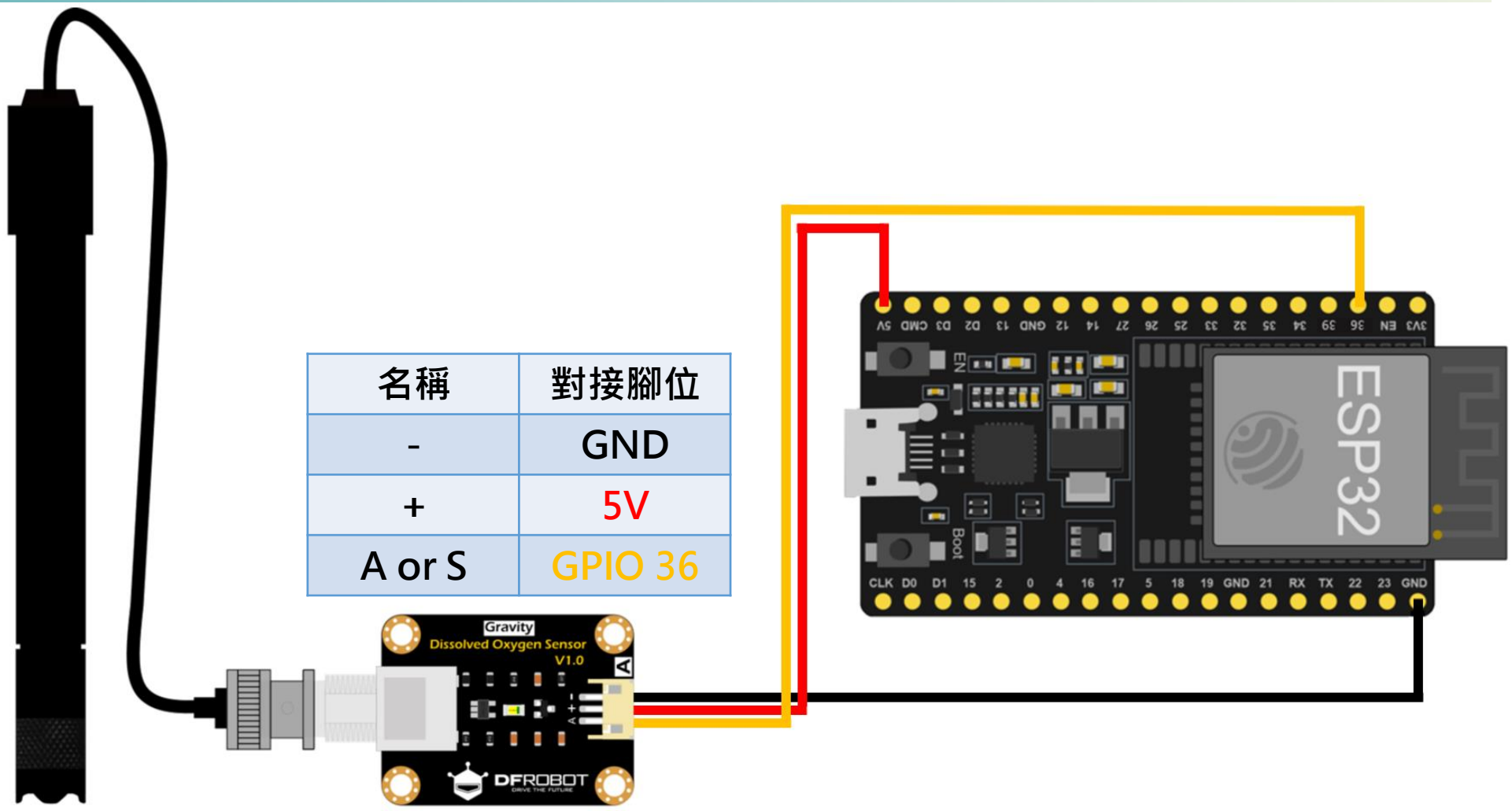


數據傳輸線 (MicroUSB)



溶解氧傳感器套件

接線說明



名稱	對接腳位
-	GND
+	5V
A or S	GPIO 36

校正方法

- 請參考簡報「[1.溶氧感測器 校正方法.pptx](#)」

撰寫程式步驟

- 開啟記事本
「範例程式 溶氧感測器.txt」
- 複製內容並貼上Arduino視窗中

撰寫程式步驟

```
#define DO_PIN 36 //腳位GPIO36
```

```
#define VREF 5000 //VREF (mv)
```

```
#define ADC_RES 4096 //ADC Resolution
```

```
//Single-point calibration Mode=0
```

```
//Two-point calibration Mode=1
```

```
#define TWO_POINT_CALIBRATION 0
```

使用單點校正法輸入「0」

使用兩點校正法輸入「1」

```
#define READ_TEMP (25) //修改量測溶氧水的溫度
```

撰寫程式步驟

```
//Single point calibration needs to be filled CAL1_V and CAL1_T
#define CAL1_V (1600) //mv
#define CAL1_T (25) //°C
//Two-point calibration needs to be filled CAL2_V and CAL2_T
//CAL1 High temperature point, CAL2 Low temperature point
#define CAL2_V (1300) //mv
#define CAL2_T (15) //°C
```

使用單點校正法請更改CAL1_V(校正電壓)、CAL1_T(校正的溫度)
CAL2_V及CAL2_T不須理會

使用兩點校正法請更改CAL1_V(第一杯水校正電壓)、CAL1_T(第一杯水的水溫)
更改CAL2_V(第二杯水校正電壓)、CAL2_T(第二杯水的水溫)

```
const uint16_t DO_Table[41] = {
    14460, 14220, 13820, 13440, 13090, 12740, 12420, 12110, 11810, 11530,
    11260, 11010, 10770, 10530, 10300, 10080, 9860, 9660, 9460, 9270,
    9080, 8900, 8730, 8570, 8410, 8250, 8110, 7960, 7820, 7690,
    7560, 7430, 7300, 7180, 7070, 6950, 6840, 6730, 6630, 6530, 6410};
```


撰寫程式步驟

```
uint8_t Temperaturet;  
uint16_t ADC_Raw;  
uint16_t ADC_Voltage;  
uint16_t DO;  
  
int16_t readDO(uint32_t voltage_mv, uint8_t temperature_c)  
{  
#if TWO_POINT_CALIBRATION == 0  
    uint16_t V_saturation = (uint32_t)CAL1_V + (uint32_t)35 * temperature_c - (uint32_t)CAL1_T * 35;  
    return (voltage_mv * DO_Table[temperature_c] / V_saturation);  
#else  
    uint16_t V_saturation = (int16_t)((int8_t)temperature_c - CAL2_T) * ((uint16_t)CAL1_V - CAL2_V) /  
    ((uint8_t)CAL1_T - CAL2_T) + CAL2_V;  
    return (voltage_mv * DO_Table[temperature_c] / V_saturation);  
#endif  
}  
double DOSensor;
```

撰寫程式步驟

```
void setup()  
{  
  Serial.begin(115200);  
}
```

```
void loop()  
{  
  Temperaturet = (uint8_t)READ_TEMP;  
  ADC_Raw = analogRead(DO_PIN);  
  ADC_Voltage = uint32_t(VREF) * ADC_Raw / ADC_RES;  
  DOSensor = ((double)(readDO(ADC_Voltage, Temperaturet))/1000);  
  Serial.print("Temperaturet:\t" + String(Temperaturet) + "\t");  
  Serial.print("ADC RAW:\t" + String(ADC_Raw) + "\t");  
  Serial.print("ADC Voltage:\t" + String(ADC_Voltage) + "\t");  
  Serial.println("DO:\t" + String(readDO(ADC_Voltage, Temperaturet)) + "\t");  
  Serial.print("目前溶氧：");  
  Serial.println( String(DOSensor) + " mg/L"); //序列埠視窗顯示溶氧值  
  
  delay(5000); //五秒更新一次  
}
```

寫入程式步驟

- 1. 確定工具欄位下的選項有正確選擇
- 2. 確認後點擊上傳

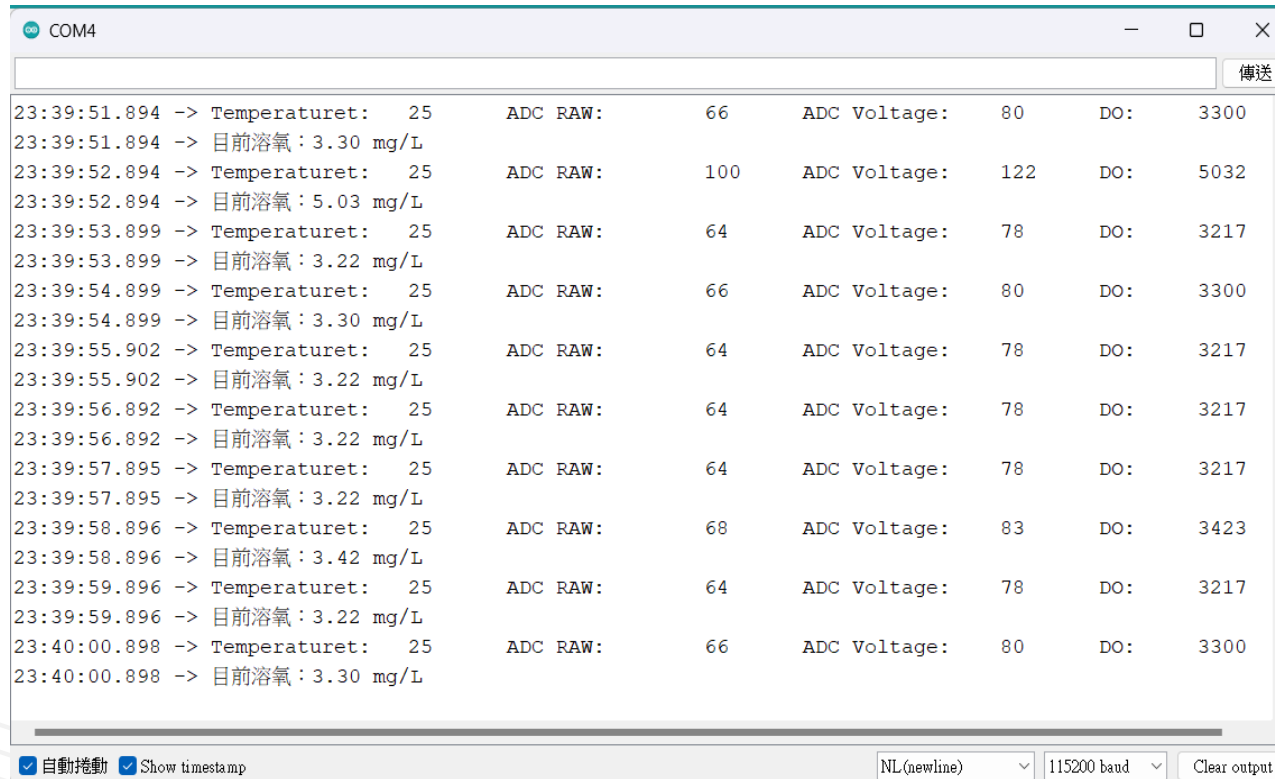


- 3. 等待底下出現此字串即成功

```
Leaving...
Hard resetting via RTS pin...
```

查看資訊

➤ 開啟右上角序列埠監控視窗



Temperature : 溫度
 ADC RAW : 類比訊號原始值
 ADC Voltage : 電壓值
 DO : 溶氧量
 目前溶氧量 : DO數值÷1000
 (單位mg/L)



藏碳蘊漁

古都土城仔綠電創能與智動養殖
之跨界整合永續淨零發展計畫

感謝聆聽
給予指導

