



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

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

溶氧感測器

校正方法



目錄



材料

校正流程-單點校準

接線說明

校正流程-兩點校準

製作探頭填充液

校正資料填入程式說明

校正流程

完整程式

校正程式

溶氧量上限對照表



藏碳源

材料



ESP32



數據傳輸線 (MicroUSB)



溶解氧傳感器套件

材料



片鹼(氫氧化鈉)

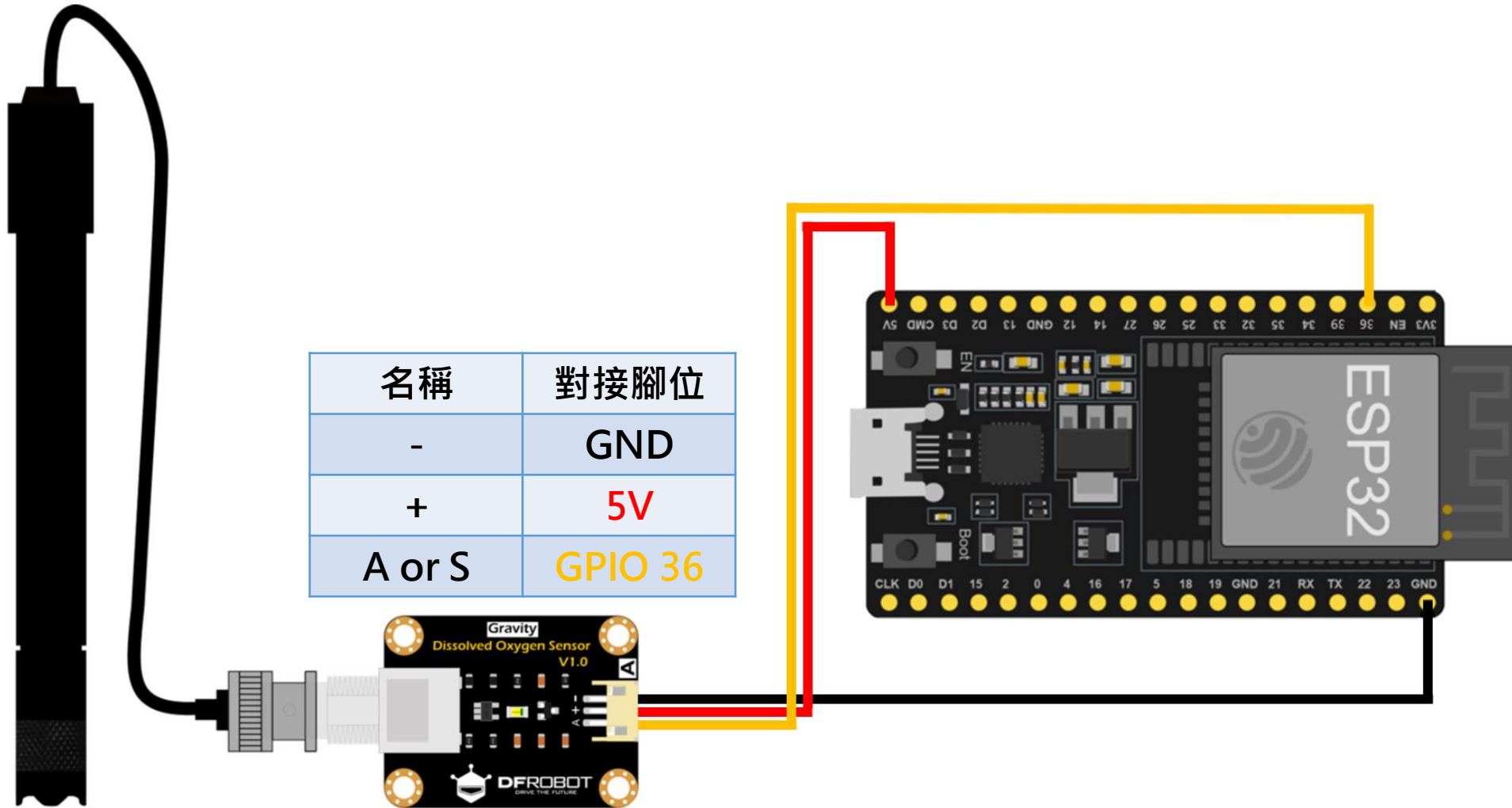


飽和溶氧水



水下溫度計

接線說明



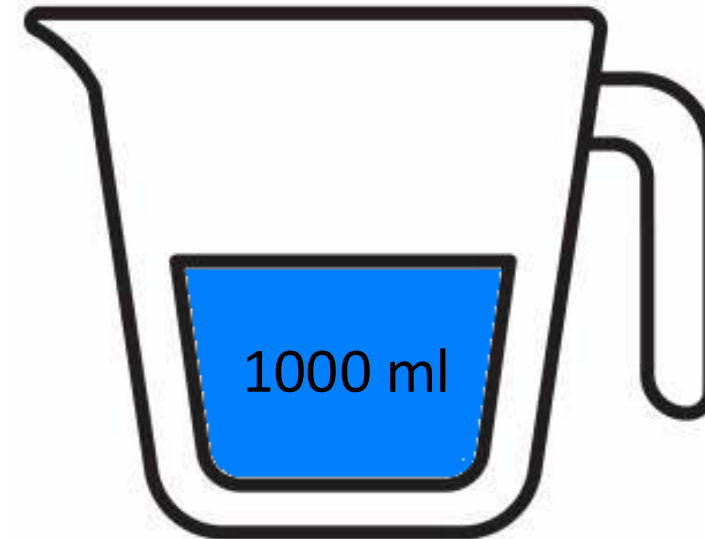
製作探頭填充液

➤ 首先要製作0.5mol/L的氫氧化鈉(NaOH)溶液

準備20g氫氧化鈉放入1000 ml的水中，將其攪拌均勻



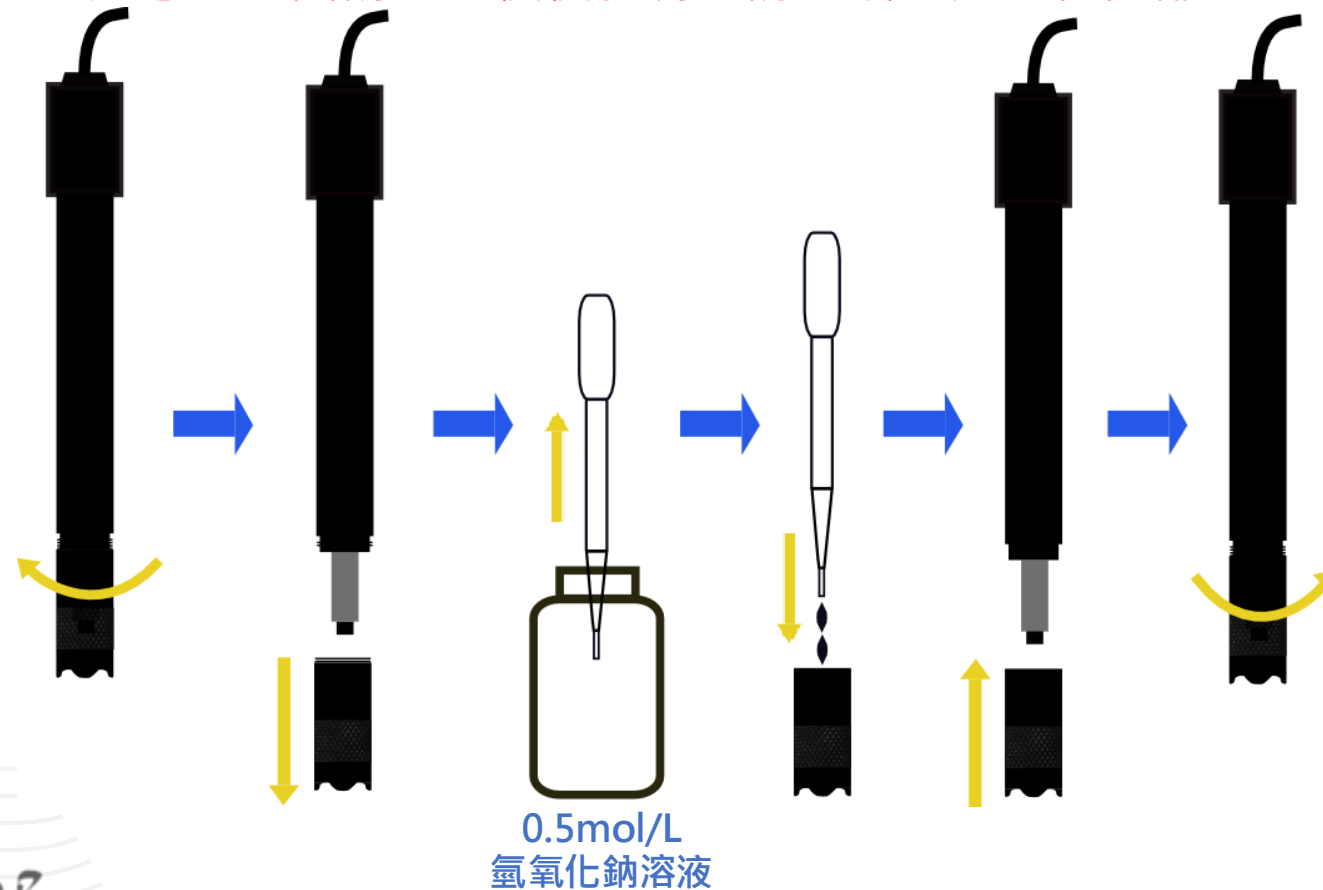
20 g
NaOH



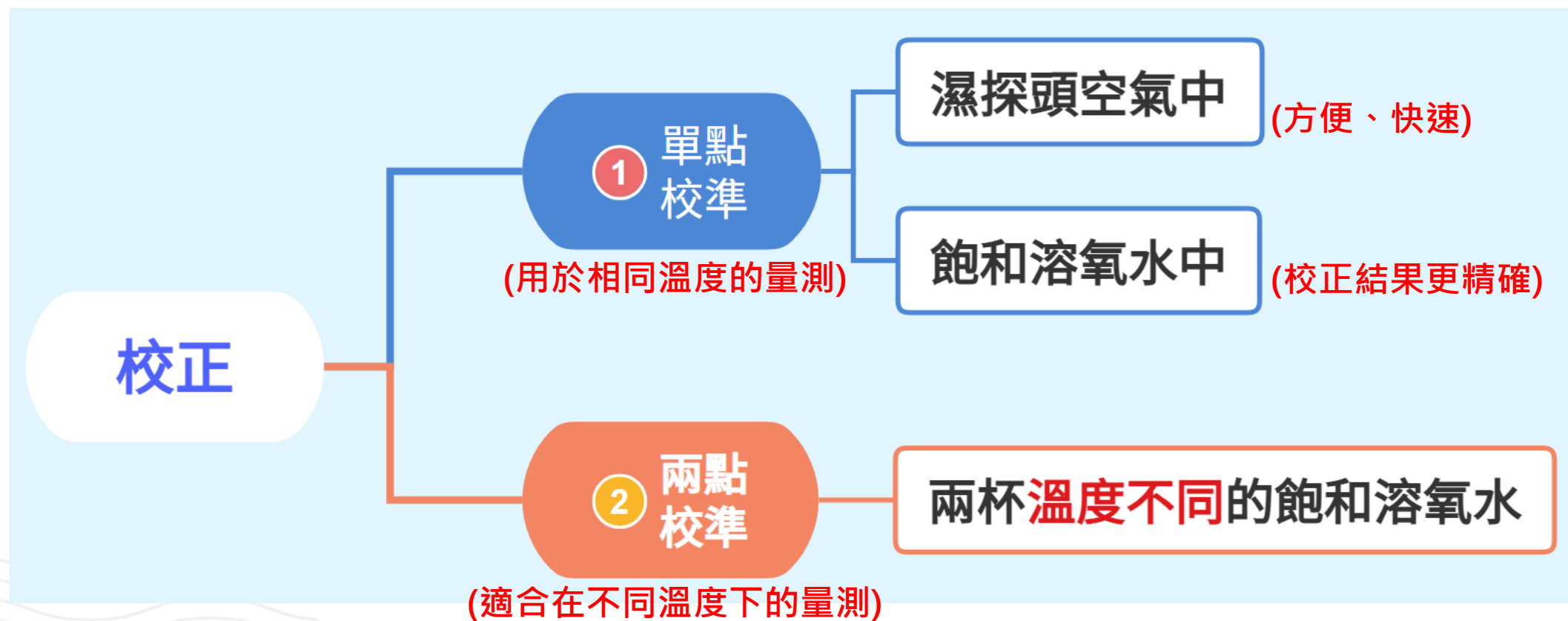
製作探頭填充液

➤ 擰下膜帽，以滴管將剛製作的氫氧化鈉(NaOH)溶液加入膜帽中，再將其鎖回

注意：鎖回請盡量慢慢旋轉，防止氣泡產生在內部



校正流程



校正流程-校正程式

```
#include <Arduino.h>

#define VREF 5000//VREF(mv)
#define ADC_RES 4096//ADC Resolution

uint32_t raw;

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

void loop()
{
    raw=analogRead(36);
    Serial.println("raw:\t"+String(raw)+"\tVoltage(mv)"+String(raw*VREF/ADC_RES));
    delay(1000);
}
```

校正流程-單點校準 (方法1)

➤ 1.濕探頭在空氣中校正法

探頭膜帽填充並鎖緊後，將探頭放入純水中，靜置1分鐘後拿起，放在空氣中等待校正程式的數值穩定，記錄當下溫度及Voltage(mv)值資訊。
(測量環境溫度)



(請勿將探頭放置在電風扇出風口校正，會導致數值不準確)

校正流程-單點校準 (方法1)

開啟Arduino IDE，並寫入校正程式，開啟序列埠視窗監控，等待數值穩定，記錄當下的電壓值(Voltage (mv))及水溫。

(使用單點校準方法1，請紀錄室溫)

COM4

微浮動時，取得平均值並記錄下來

傳送

```
23:27:59.570 -> raw: 200 Voltage (mv) 244
23:28:00.570 -> raw: 200 Voltage (mv) 244
23:28:01.572 -> raw: 176 Voltage (mv) 214
23:28:02.575 -> raw: 196 Voltage (mv) 239
23:28:03.581 -> raw: 200 Voltage (mv) 244
23:28:04.587 -> raw: 200 Voltage (mv) 244
23:28:05.544 -> raw: 200 Voltage (mv) 244
23:28:06.549 -> raw: 232 Voltage (mv) 283
23:28:07.553 -> raw: 200 Voltage (mv) 244
23:28:08.560 -> raw: 194 Voltage (mv) 236
23:28:09.562 -> raw: 200 Voltage (mv) 244
23:28:10.564 -> raw: 197 Voltage (mv) 240
23:28:11.570 -> raw: 198 Voltage (mv) 241
23:28:12.574 -> raw: 200 Voltage (mv) 244
```

例：244

☒ 自動捲動 ☒ Show timestamp NL(newline) 115200 baud Clear output

校正流程-單點校準 (方法2)

➤ 2.探頭在飽和溶氧水中校正法



[1] 準備一杯純水製作飽和溶氧水，透過以下其中一種方法：

- (1) 使用攪拌器或打蛋器連續攪拌10分鐘，使溶解氧飽和。
- (2) 用氣泵連續充氣10分鐘，使溶解氧飽和。

[2] 飽和後，等待水中氣泡消除，再將探頭放入水中

[3] 放入探頭後持續慢慢攪拌，但也要避免氣泡的產生

[4] 待數值穩定，記錄當下溫度及Voltage(mv)值資訊
(同簡報第10頁) (測量水中溫度)

校正流程-兩點校準

➤ 不同溫度下的飽和溶液校正

[1] 準備兩杯純水，一杯放入冰箱、一杯做加熱 (加熱請勿超過40°C，以免損壞探頭)

[2] 將這兩杯純水製作成飽和溶氧水，透過以下其中一種方法：

- (1) 使用攪拌器或打蛋器連續攪拌10分鐘，使溶解氧飽和。
- (2) 用氣泵連續充氣10分鐘，使溶解氧飽和。

[3] 飽和後，等待水中氣泡消除，再將探頭放入其中一杯水中

[4] 放入探頭後持續慢慢攪拌，但也要避免氣泡的產生

[5] 待數值穩定，記錄當下溫度及Voltage(mv)值資訊 (同簡報第10頁)
(測量水中溫度)

[6] 重複步驟3~5，將另一杯水做相同動作

校正資料填入程式說明

```
1 #define DO_PIN 36 //連接GPIO36
2
3 #define VREF 5000 //VREF (mv)
4 #define ADC_RES 4096 //ADC Resolution
5
6 //Single-point calibration Mode=0
7 //Two-point calibration Mode=1
8 #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」

校正資料填入程式說明

```
9  
10 #define READ_TEMP (25) //Current water temperature °C, Or temperature sensor function  
11
```

更改當下水溫
(如有溫度感測器，可導入溫度感測器程式，將框中數值替換成即時溫度)

校正資料填入程式說明

```
12 //Single point calibration needs to be filled CAL1_V and CAL1_T
13 #define CAL1_V (1600) //mv
14 #define CAL1_T (25) //°C
15 //Two-point calibration needs to be filled CAL2_V and CAL2_T
16 //CAL1 High temperature point, CAL2 Low temperature point
17 #define CAL2_V (1300) //mv
18 #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)

完整程式

```
#define DO_PIN 36

#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

#define READ_TEMP (25) //Current water temperature °C, Or temperature sensor function

//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

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};

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
}

float DOSensor;

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 = ((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");

    delay(5000);
}
```



溶氧量上限對照表

T ℃	DO mg/L	T ℃	DO mg/L	T ℃	DO mg/L
0	14.60	16	9.86	32	7.30
1	14.22	17	9.64	33	7.17
2	13.80	18	9.47	34	7.06
3	13.44	19	9.27	35	6.94
4	13.08	20	9.09	36	6.84
5	12.76	21	8.91	37	6.72
6	12.44	22	8.74	38	6.60
7	12.11	23	8.57	39	6.52
8	11.83	24	8.41	40	6.40
9	11.56	25	8.25	41	6.33
10	11.29	26	8.11	42	6.23
11	11.04	27	7.96	43	6.13
12	10.76	28	7.83	44	6.06
13	10.54	29	7.68	45	5.97
14	10.31	30	7.56	46	5.88
15	10.06	31	7.43	47	5.79

- 在不同溫度下，水的溶解氧濃度也不同，表格為各溫度水的溶氧量上限值

水溫越高，溶氧量越低



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

