

# DSI5188 傾斜偵測器

設計者：Hartman Hsieh

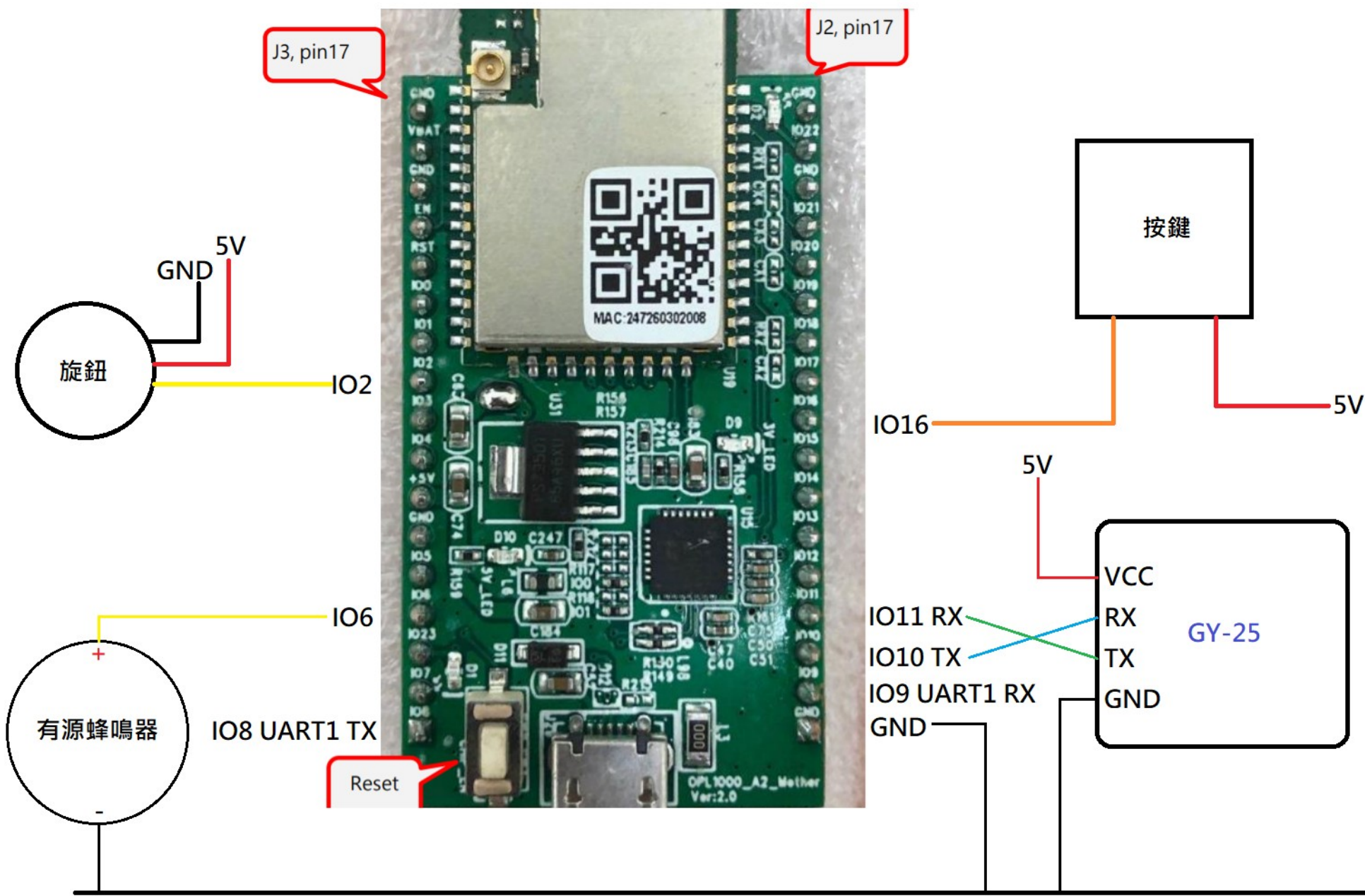
# 情境概述

可以測量物體表面 X 軸與 Y 軸傾斜的角度。有蜂鳴器，沒有水平時會發出嗶嗶聲。也可以當警報器，設定傾斜時透過網路發出警報。適用產業：裝潢業、水平平台安裝 ...

# 所需材料

- DSI5188 (OPL1000)
- 有源蜂鳴器
- 六軸感測器 MPU-6050 (GY-25)
- 可變電阻旋鈕模組
- 按鍵模組
- 麵包板
- 數條不同顏色的單芯線
- 18650 電池

# 接線圖



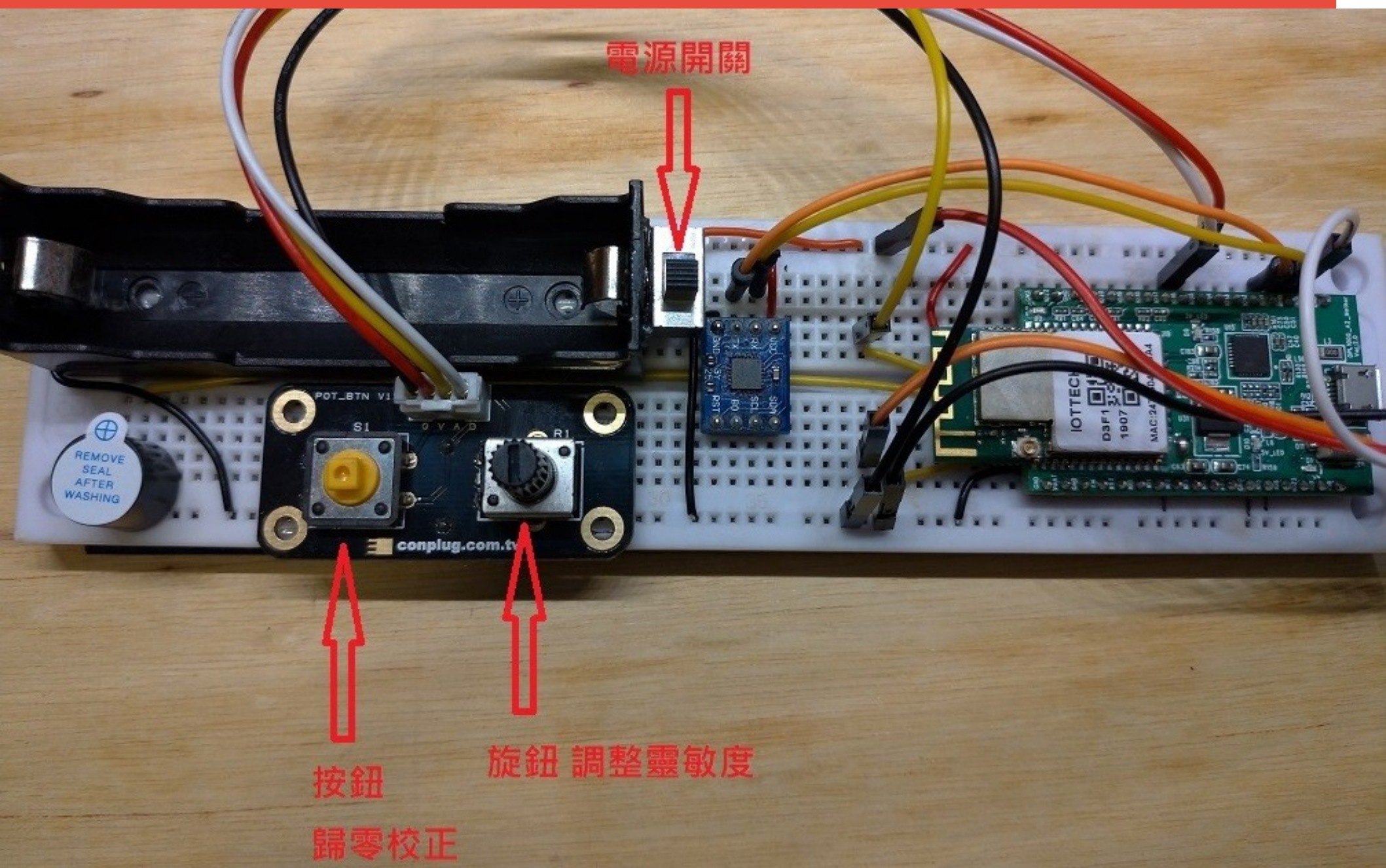
# DSI5188 腳位與元件介紹

IO8(TX), IO9(RX) 是 UART1，DSI5188 的除錯訊息會從這邊丟出來。

- IO10(TX), IO11(RX) 是 UART\_IDX\_0，連接 GY-25。
- 按鍵有下拉電阻，按下是高電位觸發中斷。
- 有源蜂鳴器只要一通電就會發出聲音。
- 旋鈕需接在類比輸入腳位 IO0 ~ IO15
- GY-25 上面有六軸感測器 MPU-6050，經過 MCU 計算出 3 軸角度 (Yaw, Roll, Pitch) 後直接輸出數值到串口。



# 接線完成照片



# 操作說明

放置水平桌面，把 USB 線插上 DSI5188 開發板上面 Micro USB，或是打開電池旁的開關（若有安裝 18650 鋰電池的話）。靜待約 8 秒，等待 GY-25 做校正歸零。

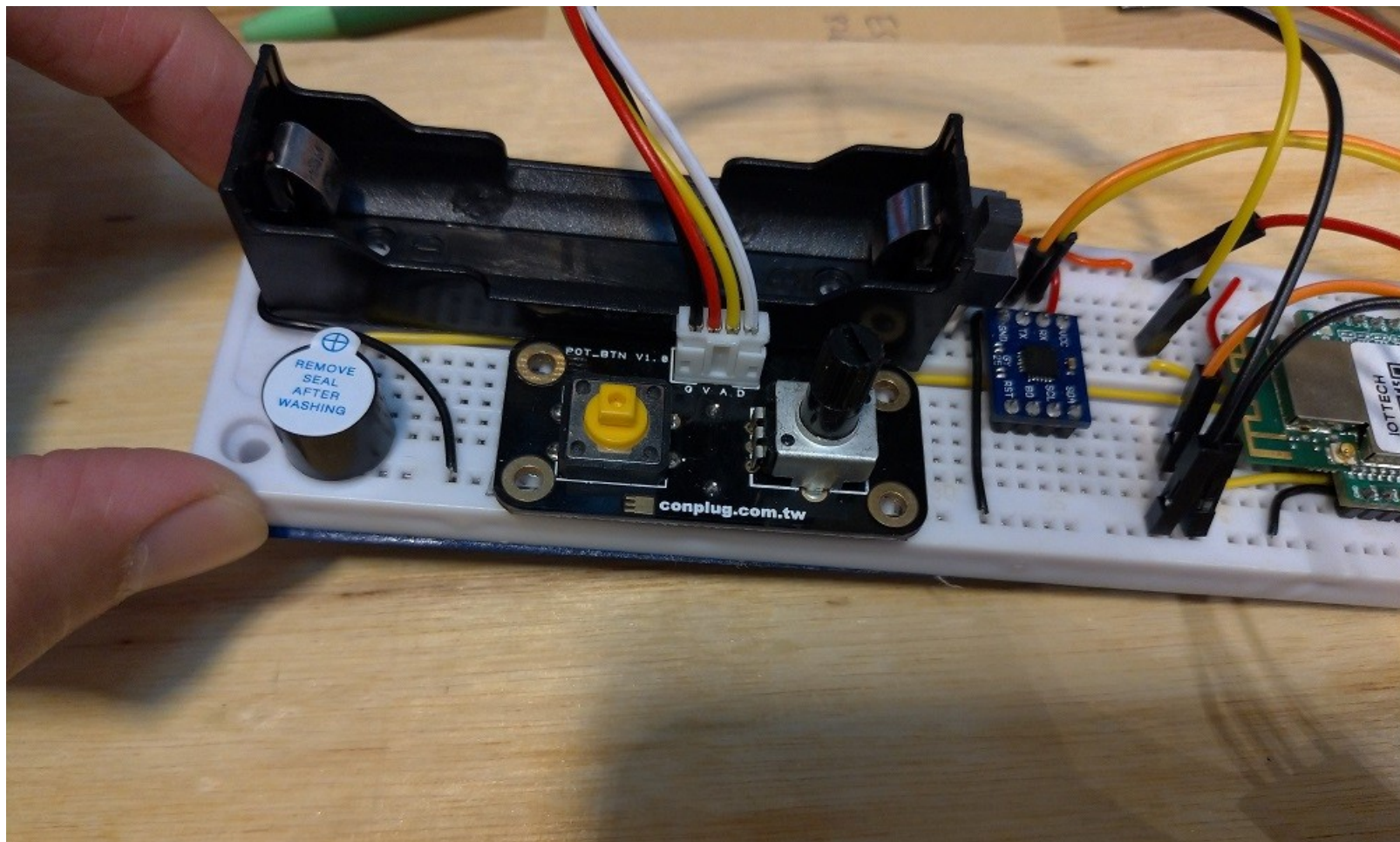
旋鈕可以調整靈敏度，逆時針轉越靈敏，順時針轉越不靈敏，順時針轉到底關閉蜂鳴器，但仍會傳數值到 IDEAS Chain 數據平台。一般使用可以調整至最靈敏。

按鈕按下會做校正歸零，按下後請靜待 8 秒。

蜂鳴器發出聲音代表有警報，會立刻傳送警報至 IDEAS Chain 數據平台。



# 操作照片



拉高麵包板一側，蜂鳴器會發出聲音，請調整旋鈕至適當靈敏度。



# IDEAS Chain 儀表板



YAW

10°

ROLL

0°

PITCH

0°

傾斜警報



— ALERT

平均值

0.04

# 程式介紹 - 初始化

```
//  
// ADC 初始化  
//  
Hal_Aux_Init();  
  
//  
// 等待 GY-25 初始化  
//  
osDelay(3500);  
  
//  
// 緩衝區清零  
//  
memset(&(buff[0]), 0, MAX_BUFFER_SIZE);  
buff_index = 0;  
  
//  
// 發送命令 - 進入校正模式  
//  
printf("make zero\n");  
gy25MakeZero(UART_IDX_0);  
  
osDelay(1000);
```

# 程式介紹 - GY-25 進入查詢模式

```
//  
// 發送命令給 GY-25 - 進入查詢模式  
//  
gy25RequestMode(UART_IDX_0);  
  
osDelay(30);  
  
//  
// 從 UART 讀取資料。  
//  
while(Hal_Uart_DataRcv(UART_IDX_0, &data32) == 0) {
```

# 程式介紹 - 讀取旋鈕數值

```
//  
// 間隔 0.6 秒讀取旋鈕數值  
//  
if((millis() - read_adc_time) > 600) {  
    //  
    // 讀取旋鈕數值 0 ~ 3.x  
    //  
    adc_voltage = analogRead(GPIO_IDX_ADC);  
  
    //  
    // 最低限 : 0.1  
    //  
    adc_voltage = adc_voltage + 0.1;  
  
    read_adc_time = millis();  
}
```



# 程式介紹 - 解析 GY-25 的資料

```
//  
// 解析 GY-25 傳來的資料。  
//  
Gy25Data = parseGy25(&(buff[buff_index-7]));  
  
//  
// 傾斜角度絕對值超過旋鈕數值則蜂鳴器發出聲音。  
//  
if((Gy25Data.ROLL > adc_voltage) || (Gy25Data.ROLL < (-adc_voltage))) {  
    if(adc_voltage < 3.2) { // 旋鈕數值大於等於 3.2 的話，關閉蜂鳴器 (靜音模式)。  
        digitalWrite(gpio_idx_beeper, GPIO_LEVEL_HIGH);  
    }  
  
    //  
    // 4 - 代表有警報，此警報數值會一併傳送到 IDEAS Chain 的數據平台。  
    //  
    FlagAlert = 4;  
}  
else {  
    digitalWrite(gpio_idx_beeper, GPIO_LEVEL_LOW);  
  
    //  
    // 0 - 代表無警報，此警報數值會一併傳送到 IDEAS Chain 的數據平台。  
    //  
    FlagAlert = 0;  
}
```

# 程式介紹 - 按鈕中斷 callback

```
//  
// 使用者按下按鍵  
//  
if(ButtonPressed == 1) {  
    //  
    // 發送命令 - 進入校正模式  
    //  
    printf("make zero\n");  
    gy25MakeZero(UART_IDX_0);  
  
    ButtonPressed = 0;  
  
    osDelay(1000);  
}
```

```
//  
// 按鈕中斷 callback  
//  
static void gpio_int_callback(E_Gpioldx_t tGpioldx)  
{  
    ButtonPressed = 1;  
}
```

# 程式介紹 - 連接 WIFI AP

## http\_request.h

```
#define WIFI_SSID "MY_SSID"  
#define WIFI_PASSWORD "MY_PASSWORD"
```

## http\_request.c

```
/* Initialize wifi stack and register wifi init complete event handler */  
wifi_init(&int_cfg, NULL);  
  
/* Set user's configuration */  
strcpy((char *)wifi_config.sta_config.ssid, WIFI_SSID);  
strcpy((char *)wifi_config.sta_config.password, WIFI_PASSWORD);  
wifi_config.sta_config.ssid_length = strlen(WIFI_SSID);  
wifi_config.sta_config.password_length = strlen(WIFI_PASSWORD);  
memcpy(wifi_config.sta_config.bssid, bssid, WIFI_MAC_ADDRESS_LENGTH);  
  
wifi_set_config(WIFI_MODE_STA, &wifi_config);  
  
/* Wi-Fi operation start */  
wifi_start();
```

# 程式介紹 - 建立執行緒處理 HTTP

建立一個執行緒專門處理傳送資料到 IDEAS Chain 數據平台的動作，不影響感測器讀取。

```
task_def.name = "user_app";  
task_def.stacksize = OS_TASK_STACK_SIZE_APP;  
task_def.tpriority = OS_TASK_PRIORITY_APP;  
task_def.pthread = thread_http_request;  
app_task_id = osThreadCreate(&task_def, (void*)NULL);
```



# 程式介紹 - 開始傳送資料

```
//  
// 產生 json 字串，準備傳送至 IDEAS Chain 的數據平台  
//  
sprintf(http_body, "{\"YAW\":\"%f\",\"PITCH\":\"%f\",\"ROLL\":\"%f\",\"ALERT\":\"%d\"}", Gy25Data.YAW, Gy25Data.PITCH,  
Gy25Data.ROLL, FlagAlert);  
sprintf(buff, POST_REQUEST, strlen(http_body), http_body);  
printf("%s\n", http_body);  
  
//  
// 開始傳送資料  
//  
if (write(s, buff, strlen(buff)) < 0) {  
    printf("... socket send failed \r\n");  
    close(s);  
    osDelay(4000);  
    continue;  
}
```

# 程式介紹 - 發生傾斜警報

```
//  
// 若無警報，約 2 秒傳送一次訊息至 IDEAS Chain 的數據平台。  
//  
for(int countdown = 0; countdown < 20; countdown++) {  
    //  
    // 發生傾斜警報，中止 delay，立即傳送訊息至 IDEAS Chain 的數據平台  
    //  
    if(flagAlert == 1) {  
        printf("FlagAlert == 1\n");  
        break;  
    }  
  
    osDelay(100);  
}
```

# 總結

DSI5188 具有完整的 **GPIO**，且擁有多執行緒的處理能力，可以把讀取感測器與傳送資料的動作分別放在不同執行緒，互不影響各自的動作，動作更即時，使用上不會有延遲的感覺。