



案名：國產 IC 開發板種子講師合作 (北部)
廠商：智慧立方有限公司
項目：**DSI5188** 應用教學教案設計開發 2 件

DSI5188 水位偵測與補水器

設計者：Hartman Hsieh

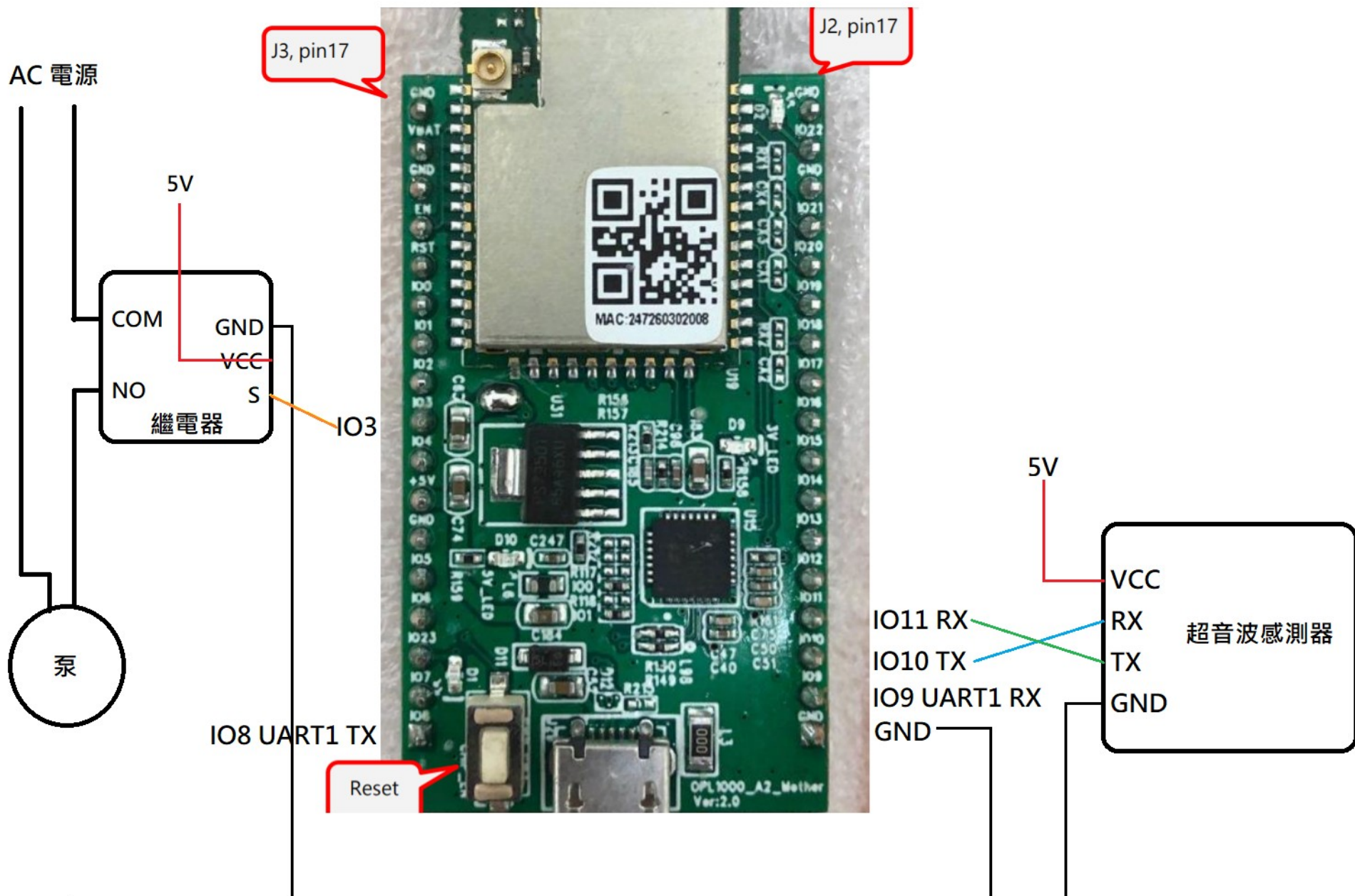
情境概述

利用超音波探測水位高度，若水位異常則發出警報到 IDEASChain 平台上。可用繼電器來控制水泵電源，水位過低，通電啟動水泵來補水至目標高度後自動停止。養殖漁業，畜牧業儲水槽等等 ...

所需材料

- DSI5188 (OPL1000)
- 超音波感測器
- 繼電器
- 水泵 (沉水馬達)
- 麵包板
- 數條不同顏色的單芯線

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 ， 連接超音波感測器。此款防水超音波感測器是 UART 介面溝通。
- 繼電器的控制腳接到 DSI5188 的 IO3 。
- 泵採用沉水馬達，電源使用家用的交流電 110V 。繼電器控制其中一條線達到開關泵的目的。

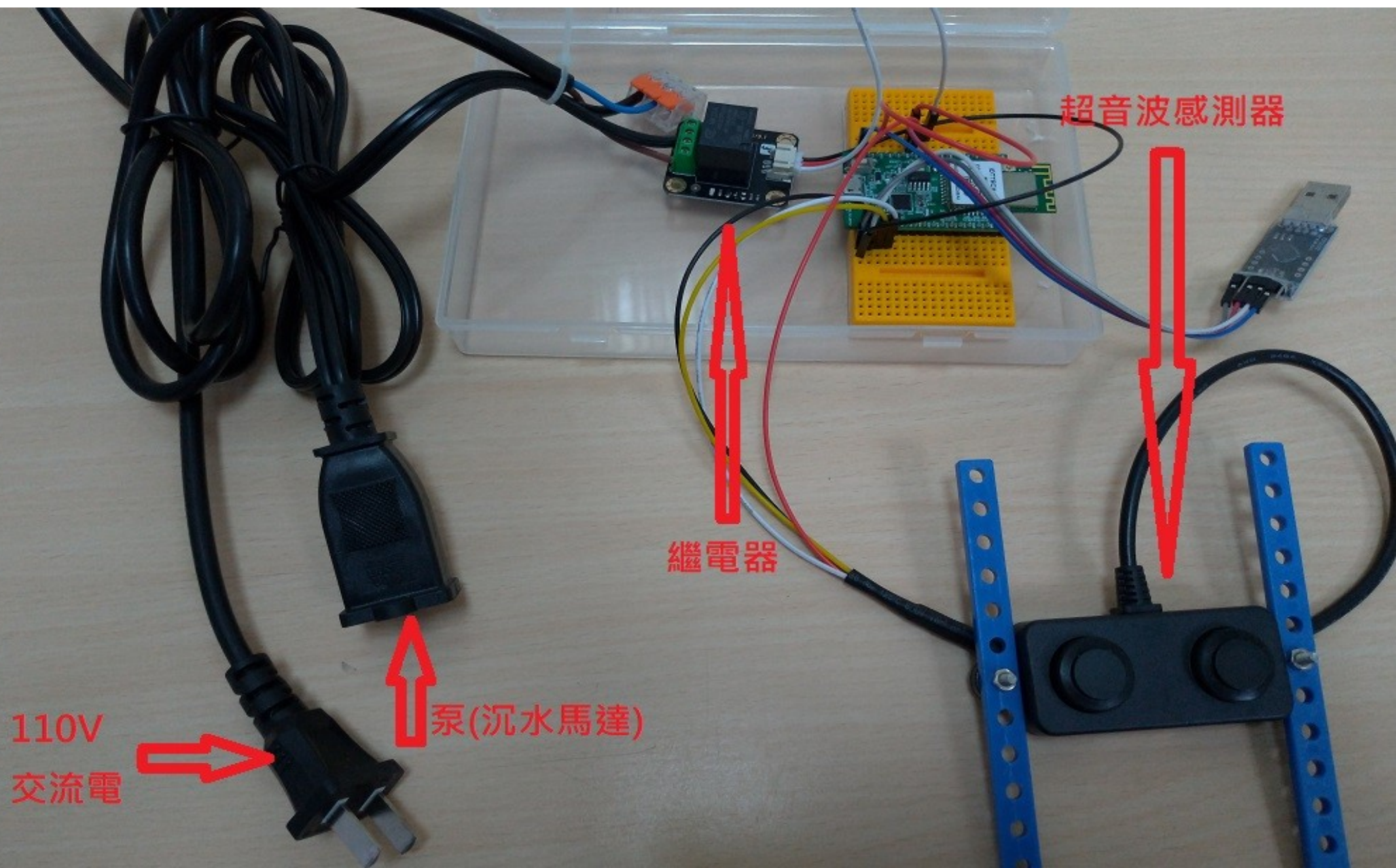
操作說明

超音波感測器朝下水平放置水面之上，儘量遠離其它物體，避免誤判水面距離。

水位低於目標水位，DSI5188 會打開繼電器啟動泵抽水來補水。等到水位升到目標水位後，繼電器斷開，停止補水。

DSI5188 每 2 秒會傳送一次水位高度到 IDEAS Chain 數據平台。低於設定水位則傳送警報訊息。

接線完成照片



操作照片



IDEAS Chain 儀表板

儀表板庫 > 水位偵測

論壇 應用案例

水位偵測

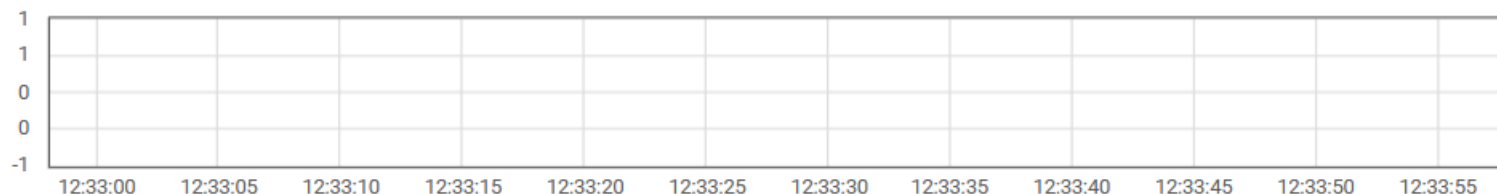
水位偵測

WaterLevel

14.4 CM

← 即時水位高度

Timeseries Bars - Flot



Alert

低水位時此圖表會出現立棒

平均值

程式介紹 - 初始化

```
//  
// 取得超音波 trigger GPIO 資料  
//  
gpio_ultra_sonic_tri_ptr = &OPL1000_periph.gpio[0];  
gpio_ultra_sonic_tri = Hal_Pinmux_GetIO(gpio_ultra_sonic_tri_ptr->pin);  
digitalWrite(gpio_ultra_sonic_tri, GPIO_LEVEL_HIGH);  
  
//  
// 取得控制繼電器 GPIO 資料  
//  
gpio_relay_ptr = &OPL1000_periph.gpio[1];  
gpio_relay = Hal_Pinmux_GetIO(gpio_relay_ptr->pin);  
digitalWrite(gpio_relay, GPIO_LEVEL_LOW);
```

程式介紹 - 觸發超音波感測器

```
//  
// 對超音波感測器發送一個低電平訊號，觸發它開始探測距離。  
//  
digitalWrite(uartTx, GPIO_LEVEL_LOW);  
  
delay(5);  
  
//  
// 回到高電平  
//  
digitalWrite(uartTx, GPIO_LEVEL_HIGH);
```

程式介紹 - 解析 UART 資料

```
//  
// 發現符合格式的數值  
//  
if(buff[0] == 0xff) {  
    //  
    // 計算 Check Sum  
    //  
    int sum = (buff[0] + buff[1] + buff[2]) & 0x00ff;  
    if(sum == buff[3]) { // Check Sum 正確  
        int distance = buff[1] * 256;  
        distance += buff[2];  
        //Serial.print(distance);  
        //Serial.println(" mm");  
        return distance; // 傳回距離值  
    }  
}
```

3.2.4 UART 輸出格式

幀數據	說明	字節
幀頭	固定為 0xFF	1 字節
Data_H	距離數據的高 8 位	1 字節
Data_L	距離數據的低 8 位	1 字節
SUM	通訊校驗和	1 字節

main_patch.c

程式介紹 - 計算水位高度

```
//  
// 超音波測距感測器至池底的距離 - 超音波測距感測器至水面的距離 = 水位高度  
//  
WaterLevel = SENSOR_TO_BOTTOM - ((float)Distance / 10); // 單位：公分  
printf("WaterLevel = %f CM\n", WaterLevel);  
osDelay(100);
```

程式介紹 - 根據水位控制泵

```
//  
// 水位低於目標水位 ( 低標 )  
//  
if (WaterLevel < TargetWaterLevelL) {  
    //  
    // 打開水泵注水  
    //  
    digitalWrite(gpio_relay, GPIO_LEVEL_HIGH);  
    printf("Water level is too Low\n");  
    FlagAlert = 5;  
}  
  
//  
// 水位低於目標水位 ( 高標 )  
//  
if (WaterLevel >= TargetWaterLevelH) {  
    if(digitalRead(gpio_relay) == GPIO_LEVEL_HIGH) {  
        printf("=== Stop Pump ===\n");  
    }  
  
    //  
    // 關閉水泵  
    //  
    digitalWrite(gpio_relay, GPIO_LEVEL_LOW);  
    FlagAlert = 0;  
}
```

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, "{\"WaterLevel\": \"%f\", \"Alert\": \"%d\"}", WaterLevel, 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) {  
    DBG_PRINTF("... socket send failed \r\n");  
    close(s);  
    osDelay(4000);  
    continue;  
}  
DBG_PRINTF("... socket send success \r\n");
```

程式介紹 - 發生低水位警報

```
//  
// 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 < 2; countdown++) {  
    //  
    // 水位低於目標水位 ( 低標 )，中止 delay，立即傳送訊息至 IDEAS Chain 的數據平台  
    //  
    if(FlagAlert > 0) {  
        DBG_PRINTF("==== Alert =====\n");  
        break;  
    }  
  
    osDelay(100);  
}
```

總結

善用 **DSI5188** 多執行緒的處理能力，分別讀取感測器與傳送資料，減少動作延遲，為使用者帶來良好體驗。完整豐富的 **GPIO** 接腳，與各式各樣的感測器協調溝通順暢，開發速度快。**DSI5188** 的極低功耗，使用電池便可以運作很久，可以降低佈線難度。實際測試下，長時間運作也沒出現異常，可靠度佳。