

ATH15溫濕度感測器 簡易藍芽傳輸



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



• ESP 32 藍牙功能介紹

- 全部程式碼
- 程式講解
- 最終執行成果





ESP32藍芽功能介紹

- 1. ESP32 內建藍牙,不需再接線或設定AT COMMAND,也不 用宣告TXRX 相關的通訊協定。
- 2. ESP32 可採用低功率藍牙(BLE: Bluetooth Low Energy) 耗電量較少,傳輸距離理論可達300公尺遠。
- 3. 可模擬ibeacon 進行室內定位等服務,功能強大且使用簡單。

注意事項:

由於bluetooth 函式庫iOS 手機不相容的原因,後續都將使用Android 裝置進行解說,請準備一台Android 6.0以上手機或平板,以利後續操作。









```
#include <Wire.h>
#include <BluetoothSerial.h> // 匯入藍牙序列通訊的函式庫
```

```
#define AHT15_ADDRESS 0x38
#define AHT15_RESET 0xBA
#define AHT15_INIT 0xBEa
#define AHT15_START_MEASUREMENT ØxAC
BluetoothSerial BT; // 宣告藍牙物件, 名稱為BT
```

```
void setup() {
 Wire.begin();
 Serial.begin(115200);
 BT.begin("Denny0628"); // 請改名例如英文+ 生日
```

```
// Reset sensor
Wire.beginTransmission(AHT15_ADDRESS);
Wire.write(AHT15_RESET);
Wire.endTransmission();
delay(20);
```





全部程式碼如下:

```
// Initialize sensor
  Wire.beginTransmission(AHT15_ADDRESS)
  Wire.write(AHT15_INIT);
  Wire.write(0x08);
 Wire.write(0x00);
  Wire.endTransmission();
  delay(10);
}
void loop() {
 // Start measurement
  Wire.beginTransmission(AHT15_ADDRESS);
  Wire.write(AHT15_START_MEASUREMENT);
 Wire.write(0x33);
  Wire.write(0x00);
  Wire.endTransmission();
  delay(80);
  // Read data
  Wire.requestFrom(AHT15_ADDRESS, 6);
  uint8_t data[6];
  for (int i = 0; i < 6; i++) {</pre>
    data[i] = Wire.read();
  }
```





全部程式碼如下:

```
// Calculate humidity and temperature
uint32_t rawHumidity = ((uint32_t)data[1] << 12) | ((uint32_t)data[2] << 4) | (data[3] >> 4);
float humidity = rawHumidity * 100.0 / (1 << 20);</pre>
```

```
uint32_t rawTemperature = (((uint32_t)data[3] & 0xF) << 16) | ((uint32_t)data[4] << 8) | data[5];</pre>
float temperature = ((200.0 * rawTemperature) / (1 << 20)) - 50;</pre>
```

```
// Print results in Serial Monitor
Serial.print("Humidity: ");
Serial.print(humidity);
Serial.print("% , ");
Serial.print("Temperature: ");
Serial.print(temperature);
Serial.println("°C");
```

```
// 在藍牙上打印溫度和濕度
BT.print(temperature); // 將溫度值透過藍牙傳輸
BT.print("*C, ");
BT.print(humidity); // 將濕度值透過藍牙傳輸
BT.println(" H");
delay(2500);
```





程式講解

關於ESP32藍牙功能的程式碼,首先我們須先匯入藍牙序列通訊的函式庫, 然後再宣告藍牙物件名稱,這裡舉例為BT。









請在setup加入這行程式碼BT.begin("Your name");(名稱可使用中英文數字) 這行程式碼是在設置你的藍牙廣播名稱。

10	<pre>void setup() {</pre>
11	Wire.begin();
12	Serial.begin(115200);
13	
14	





程式講解

請在loop加入這段程式碼 這段程式碼是用意是將感測器所讀到的溫濕度值透過藍芽輸出至手機。

61	// 在藍牙上打印溫度和濕度
62	BT.print(temperature); // 將溫度值透過藍牙傳輸
63	BT.print("*C, ");
64	BT.print(humidity); // 將濕度值透過藍牙傳輸
65	BT.println(" H");
66	delay(2500);





完成之後,我們來處理手機的部分: 請將Android 手機開啟App store,並搜尋 "Arduino Bluetooth Control "(如下圖), 若無法找到該APP,可掃描下方QR code下載。











手機安裝好APP 後,啟動藍牙連線設定,如果未在可用裝置中出現請點選「不常用類型裝置」, 您所設定的ESP32 藍牙名稱就會出現於此。 完成配對後,在手機開啟剛剛安裝的Arduino Bluetooth Control APP



- 1. 點選上方的圖示。
- 2. 選擇要本次要連線的ESP32 裝置。
- 3. 連線成功時,出現「Connected to 裝置名稱」。 4. 再點選右上角的序列通訊的功能。
- 即可看到ESP32傳送的訊息。
- (若沒出現,可回第一部重新連線,理論上即可看到訊息。)







03:45:25 Denny0628 > 22.55*C, 48.42 H 03:45:27 Denny0628 > 22.55*C, 48.39 H 03:45:30 Denny0628 > 22.56*C, 48.37 H 03:45:32 Denny0628 > 22.55*C, 48.38 H 03:45:35 Denny0628 > 22.55*C, 48.46 H

左圖為Arduino Bluetooth Control APP接收到訊息的輸出畫面。

下圖為電腦序列埠監視窗輸出畫面。

15:45:24.130	\rightarrow	Humidity:	48.44
15:45:26.698	->	Humidity:	48.42
15:45:29.275	->	Humidity:	48.39
15:45:31.880	->	Humidity:	48.37
15:45:34.443	\rightarrow	Humidity:	48.38

Type in data to send







O

就碳晶晶

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