

# 其他元件 - AI2 藍牙

作者：劉映辰

## 本章介紹

本章介紹 AI2 的藍牙通訊與 Arduino 連結，透過手機藍牙傳送資料到 Arduino，再將結果回傳至手機。我們這次是先以農業相關的角度出發，用上述功能來做一個 Arduino 的溫溼度感測器，透過藍牙將測到的溫溼度回傳至手機，讓農民可以第一時間知道溫溼度的資訊。

## 藍牙

Bluetooth 一種無線通訊技術標準，用來讓固定裝置與行動裝置，在短距離間交換資料，形成個人區域網路 ( PAN )。藍牙用於在不同的裝置之間進行無線連接，例如：連接電腦和外圍裝置如印表機、鍵盤等，又或者讓個人數位助理 ( PDA ) 與其它附近的 PDA 或電腦進行通訊。藍牙技術已經應用到 82 億件產品之中，例如：汽車、遊戲機、滑鼠、鍵盤、手機、印表機....等，而具備藍牙技術的手機可以連接到電腦、PDA 甚至連接到免持聽筒。

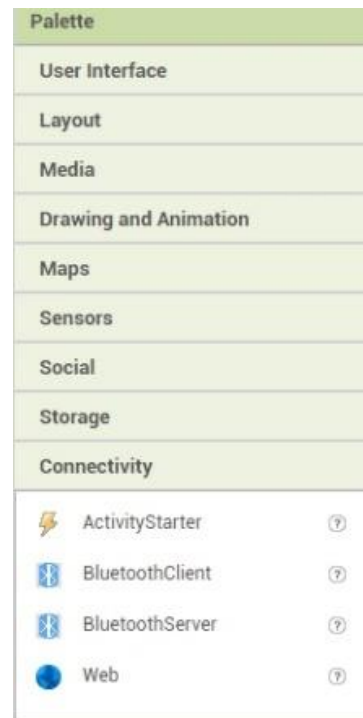
藍牙的規格從第一代到第三代其實是差不多的，只有傳輸速率不同，而第四代最大的不同就是降低功耗較為省電，第五代則是目前傳輸距離最遠且傳輸速度最快。但藍牙也有缺點，在 2.4GHz 的電波干擾問題一直為大家所詬病，特別是和無線區域網路 ( WLAN ) 間互相干擾的問題，當干擾發生時，就需要重新傳送封包的方法來解決干擾。

## 藍牙元件介紹

首先介紹藍牙元件的呼叫方法。

### 步驟一.

- 先將旁邊的 connectivity 點開。



### 步驟二.

- 將 BluetoothClient 拖曳至畫面中，下面就會出現藍牙圖示。



### 步驟三.

- 到程式編排畫面，點開旁邊的藍牙圖示，就會看藍牙元件了。



藍牙元件常用介紹：

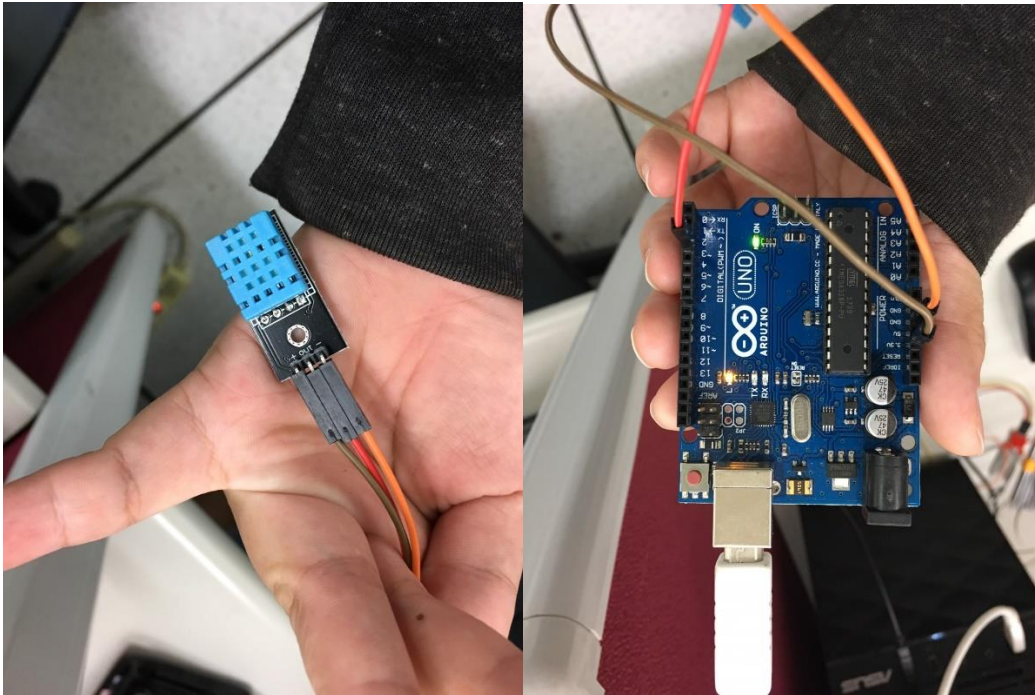
屬性	功能
	是否使用 SSP(簡易安全配對), 支援藍牙 V2.1 或更高版本的設備。
	取得已配對的藍牙裝置名稱及位址清單, 這邊如果沒有配對的裝置, 清單則會是空的。
	檢查當下的裝置是否可使用藍牙。
	檢查藍牙用戶端線在使否可用藍牙, 這邊值要為真, 藍牙才能使用。
	檢查是否建立藍牙連線。
方法	功能
	回傳可接收的位元組數, 例如如果後面設置大於 0, 那能接收的位元組數, 就會都是大於 0。
	與指定位址的藍牙裝置進行連線, 如果連線成功, 則會回傳 true。
	中斷藍牙連線。
	檢查指定位址的藍牙裝置是否配對。
	從所連結的藍牙裝置接收一個字串。注意如果值小於 0, 將持續讀取至直到收到一界定值為止。

## 範例 - Arduino 溫溼度感測器

這次我們的範例是要透過手機藍牙與 Arduino 裝置進行連結，做出一個簡易的溫溼度感測器。

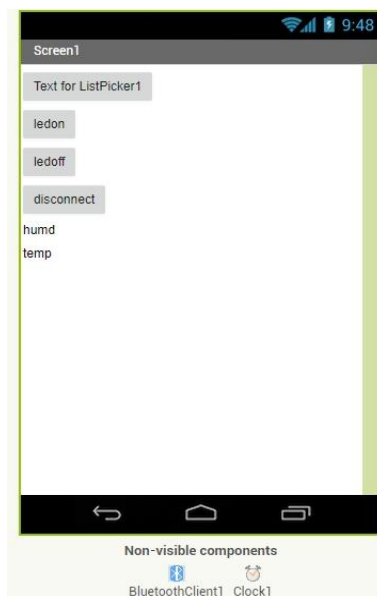
### 步驟一.

- 將溫溼度感測器的正極接到 Arduino 的 5v，負極接到 GND，最後將 out 接到 3 的位置(如下方圖所示)。



### 步驟二.

- 拉一個清單選項，3個按鈕分別是LED的開啟與關閉還有中斷藍牙連線，與 2 個文字顯示的地方顯示他的濕度與溫度。



### 步驟三.

- 當剛進到畫面時，會將溫溼度感測器與手機 app 的介面重置，這樣才不會當使用者點了 app 之後，畫面卻還是之前的畫面，沒有進行重置。

```

when Screen1.Initialize
do
  set ListPicker1.Enabled to true
  set Button1.Enabled to false
  set Button2.Enabled to false
  set Button3.Enabled to false
  set Clock1.TimerEnabled to false
  
```

### 步驟四.

- 紅框的部分為當使用者剛進到畫面重置完，會跑一個程式準備去選擇使用者要去連線的藍牙的位址和名稱，選擇完成後，便會開始執行下面的程式。

紅框下面的程式為呼叫剛剛所選的藍牙進行連接，並將之前重置過後的選項都打開，能看到重置前是 true 的變成了 false 而 false 的都變成了 true。

```

when ListPicker1.BeforePicking
do
  set ListPicker1.Elements to BluetoothClient1.AddressesAndNames

when ListPicker1.AfterPicking
do
  if call BluetoothClient1.Connect
    address ListPicker1.Selection
  then
    set ListPicker1.Enabled to false
    set Button1.Enabled to true
    set Button2.Enabled to true
    set Button3.Enabled to true
    set Clock1.TimerEnabled to true
  
```



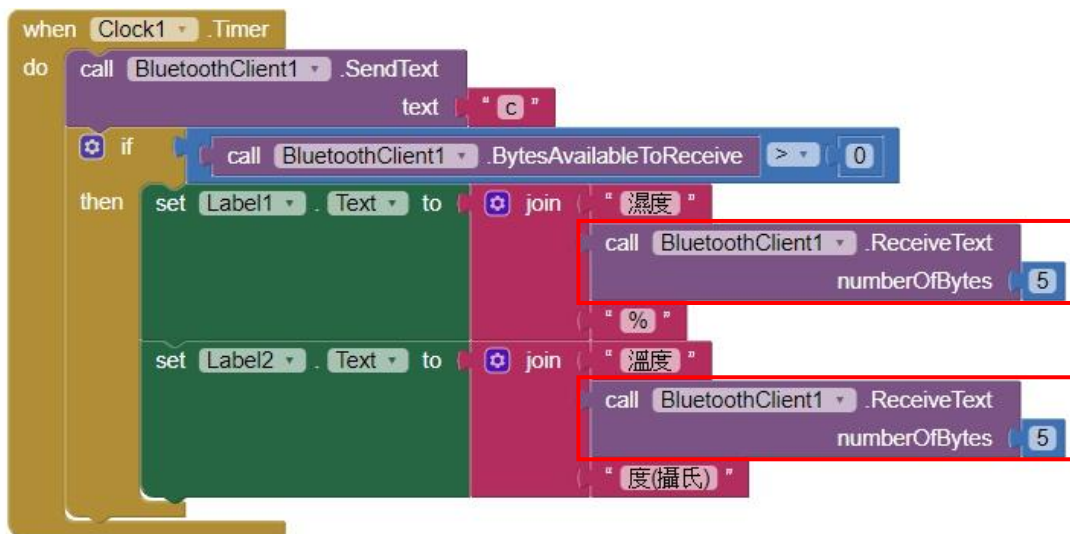
### 步驟五.

- 當按鈕 1 與按鈕 2 被點選時，傳送 a 與 b 的資料給藍牙。



### 步驟六.

- 當藍牙連結時，Clock 的選項便會變成 true，會傳送文字 c 給藍牙，而藍牙在接收到 c 的訊息時，便會回傳溼度跟溫度到 label1 與 label2。紅框的地方，是接收到的字元組數，例如：溫度不會只顯示到 37 度，而是會顯示到 37.123 度，讓數字更加精準。



## 步驟七.

- 第一個紅框的部分是對照上面 app 呼叫 a 跟 b 與 c 的地方。當呼叫 a 的時候 LED 便會亮起;呼叫 b 時便會關閉;而呼叫 c 時則會顯示濕度與溫度。第二個紅框則是讓這個迴圈結束。

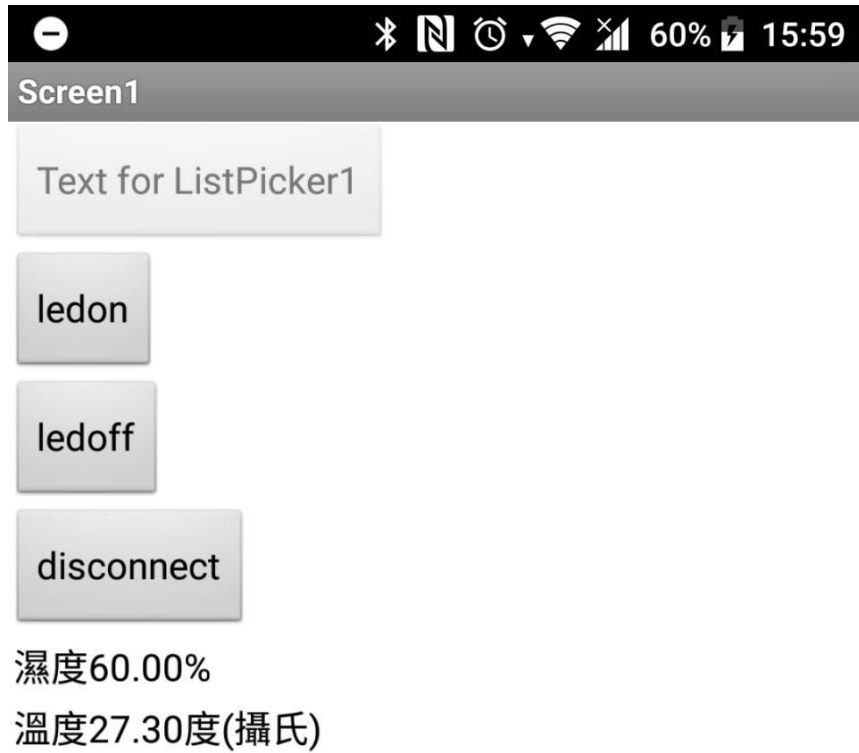
```

sketch apr9e  ReadMe.adoc
1 #include <SoftwareSerial.h>
2 #include "DHT.h"
3 #define DHTPIN 3
4 #define DHTTYPE DHT11
5
6 DHT dht(DHTPIN, DHTTYPE);
7
8 #include <Wire.h>
9
10 int LED = 2;
11 SoftwareSerial BT(10, 11);
12
13 void setup() {
14   BT.begin(9600);
15   pinMode(LED, OUTPUT);
16   Serial.begin(9600);
17   dht.begin();
18 }
19
20 void loop() {
21   byte cmd[20];
22   int strsize;
23   while (true) {
24     if ((strsize = (BT.available())) > 0) {
25       for (int i = 0; i < strsize; i++) {
26         cmd[i] = char(BT.read());
27       }
28       Serial.println(cmd[0]);
29     }
30     if (cmd[0] == 'a') {
31       digitalWrite(LED, HIGH);
32     } else if (cmd[0] == 'b') {
33       digitalWrite(LED, LOW);
34     } else if (cmd[0] == 'c') {
35       float h = dht.readHumidity();
36       float t = dht.readTemperature();
37       char hum_buf[10], tem_buf[10];
38       if (!isnan(h) && !isnan(t)) {
39         dtostrf(h, 3, 2, hum_buf);
40         dtostrf(t, 3, 2, tem_buf);
41         for (int i = 0; i < 5; i++) {
42           BT.write(hum_buf[i]);
43         }
44         for (int i = 0; i < 5; i++) {
45           BT.write(tem_buf[i]);
46         }
47       }
48     }
49     cmd[0] = 'd';
50   }
51 }
52

```

步驟八.

- 執行結果畫面。



### 活 動

利用上面的範例去修改，將 LED 燈的地方改成 RGB 燈，使 RGB 燈能亮起。