

中華電信研究院

**CHT IoT 平台與 Ameba 雙向通訊應用實  
例說明文件**

智慧聯網所編著

kemin

2017/3/6

## 中華電信智慧聯網平台- CHT IoT 平台與 Ameba 雙向通訊應用實例

中華電信智慧聯網平台(簡稱 CHT IoT 平台)具備 Connectivity Enabling 功能，可遠端控制感測裝置及將蒐集的感測資料回報至平台。以下透過 Ameba 裝置連接 2 個感測器(LED 及溫度計)，進行遠端控制與溫度資料蒐集之應用案例說明。

### [材料準備]

Ameba x 1

LED x 1

Grove Temperature Sensor V1.2 x 1

CHT IoT 平台：<https://iot.cht.com.tw/iot>

下載範例程式：chtiot\_client [←超連結到範例程式](#)


### [CHT IoT 平台準備]

- 申請帳號：目前 IoT 服務平台的帳號僅能透過電子郵件申請，如需申請帳號，煩請透過電子郵件，信箱位置為 [hsunwei@cht.com.tw](mailto:hsunwei@cht.com.tw) (數據分公司智聯處一科 沈宣璋)，並於信件內容中填寫姓名、帳號名稱、連絡電話、公司名稱等資訊，我們將有專人協助建置帳號並與您聯繫。

- 專案建置與設備配置：

每個使用者可以建立多個 IoT 專案，裡面可以納管多個設備與感測器，如下圖。



建立 Ameba Test 專案後，點選  進入專案編輯，於專案管理的權限資料內，可取得 **ApiKey** 的資料(18 碼英數混合)作為後續 Ameba 設備與平台驗證之用，如下圖。

專案管理

基本資料

權限資料

類型	內容	權限
ApiKey	PKAKK33 	admin <span>刪除</span>

新增權限：


完整權限

+

取消 儲存

於 Ameba 專案中點選  進入顯示專案，新增 2 個感測器。


- (1) LED:識別編號 ID 設定為 LED1，類型選擇 switch。
- (2) 溫度計: 識別編號 ID 設定為 TMP1，類型選擇 gauge。


 智慧聯網平台  
IOT PLATFORM


開發者中心 專案管理 產品登錄 帳號資訊 登出


專案名稱：Ameba Test + 感測器 + 設備 ← 返回


設備列表 快速搜尋:




設備編號 

設備名稱 

設備描述 

設備類型 

功能 

86141	Ameba1	Add LED and Temp sensor for controlling LED and monitoring temperature	general			
-------	--------	--	---------	---	---	---

顯示第 1 至 1 項結果，共 1 項

上頁 1 下頁

## 感測器管理



識別編號(ID) : LED1

顯示名稱 : LED1

描述 : LED

類型 : ☐ gauge ☐ counter ☒ switch ☐ snapshot ☐ text

屬性 : Key Value +

單位 : 單位 ...

取消

儲存

## 感測器管理



識別編號(ID) : TMP1

顯示名稱 : TMP1

描述 : Temperature sensor1

類型 : ☒ gauge ☐ counter ☐ switch ☐ snapshot ☐ text

屬性 : Key Value +

單位 : 單位 ...

取消

儲存

完成新增 2 個感測器後，專案內容會顯示所納管的所有感測器以及其最新的感測器狀態或蒐集的資料資訊；另外，**設備編號共 9 碼(由數字所組成)** 作為後續 **Ameba 設備 publish 及 subscribe MQTT Topic 之用**，如下圖



智慧聯網平台  
IOT PLATFORM

開發者中心 專案管理 產品登錄 帳號資訊 登出

專案名稱：Ameba Test

設備列表

設備編號 設備名稱 設備描述 設備類型 功能

86141	Ameba1	Add LED and Temp sensor for controlling LED and monitoring temperature	general	
-------	--------	--	---------	--

顯示第 1 至 1 項結果，共 1 項

設備編號：86141 感應器列表

LED1

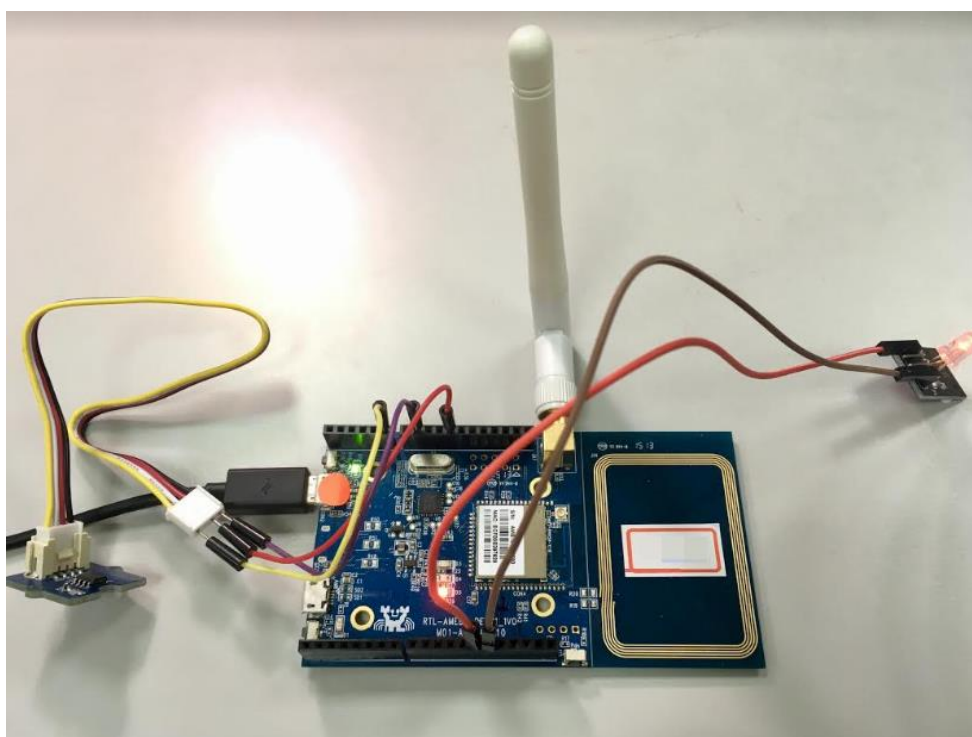
TMP1

22.14

## [Ameba 設備及感測器安裝]

將 LED 及溫度計與 Ameba 板子連接好，連接的 Pin 腳如下說明。

- (1) LED:VCC 連接 D13 及 GND 連接 Ameba GND。
- (2) 溫度計:VCC 連接 3.3V，GND 連接 Ameba GND，SIG 連接到 A0。

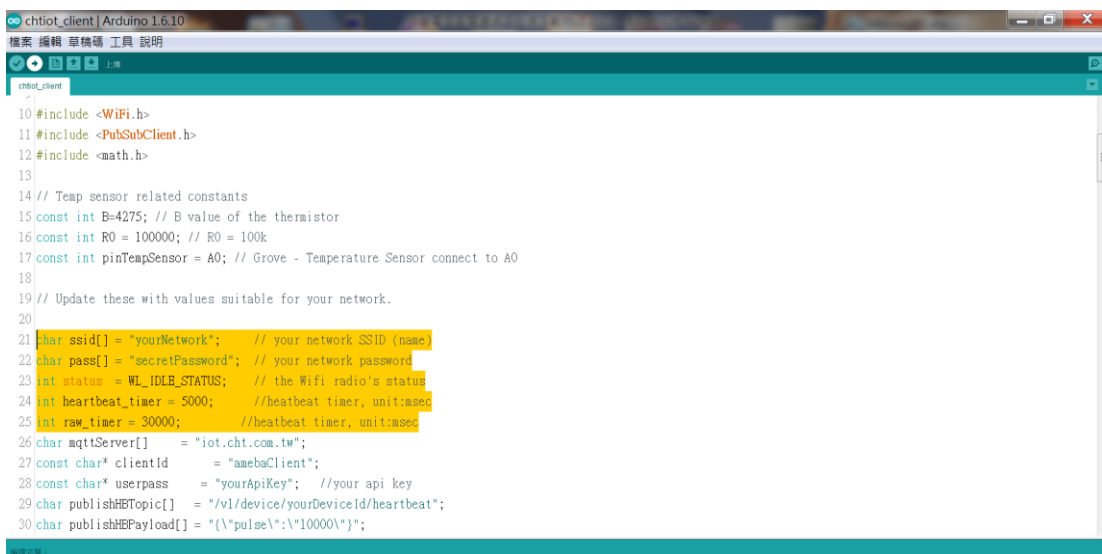


## [Arduino 範例程式說明]

這個範例裡，我們使用 Ameba 的 WiFi 連接上 AP，並可與 CHT IoT 平台進行雙向通訊應用，從 IoT 平台控制 LED 亮滅及將蒐集的溫度資料定時回報至平台。

(1)WiFi 及 Timer 相關資訊:

- WiFi 設定:請填入您使用 AP 的 SSID 及 PASSWORD 至 ssid 及 pass 變數中。
- Timer 設定:設定 heartbeat timer 為 5 秒(即 5000ms)，raw timer 為 30 秒(即 30000ms)。



```
10#include <WiFi.h>
11#include <PubSubClient.h>
12#include <math.h>
13
14// Temp sensor related constants
15const int B=4275; // B value of the thermistor
16const int R0 = 100000; // R0 = 100k
17const int pinTempSensor = A0; // Grove - Temperature Sensor connect to A0
18
19// Update these with values suitable for your network.
20
21char ssid[] = "yourNetwork"; // your network SSID (name)
22char pass[] = "secretPassword"; // your network password
23int status = WL_IDLE_STATUS; // the Wifi radio's status
24int heartbeat_timer = 5000; //heartbeat timer, unit:msec
25int raw_timer = 30000; //heartbeat timer, unit:msec
26char mqttServer[] = "iot.cht.com.tw";
27const char* clientId = "amebaClient";
28const char* userpass = "yourApiKey"; //your api key
29char publishHBTopic[] = "/v1/device/yourDeviceId/heartbeat";
30char publishHBPayload[] = "{\"pulse\":\"10000\"}";
```

(2)CHT IoT 平台相關資訊:

- MQTT Server: iot.cht.com.tw Port:1883
- clientId:可填入任意字串，此範例為"amebaClient"
- userpass:需填入建立專案時所產生的 ApiKey，由 18 碼英數所混合組成



```
16const int R0 = 100000; // R0 = 100k
17const int pinTempSensor = A0; // Grove - Temperature Sensor connect to A0
18
19// Update these with values suitable for your network.
20
21char ssid[] = "yourNetwork"; // your network SSID (name)
22char pass[] = "secretPassword"; // your network password
23int status = WL_IDLE_STATUS; // the Wifi radio's status
24int heartbeat_timer = 5000; //heartbeat timer, unit:msec
25int raw_timer = 30000; //heartbeat timer, unit:msec
26char mqttServer[] = "iot.cht.com.tw";
27const char* clientId = "amebaClient";
28const char* userpass = "yourApiKey"; //your api key
29char publishHBTopic[] = "/v1/device/yourDeviceId/heartbeat";
30char publishHBPayload[] = "{\"pulse\":\"10000\"}";
31char publishRawTopic[] = "/v1/device/yourDeviceId/rawdata";
32char publishRawPayload[] = "{\"id\":\"TMP1\",\"value\":{\"99.99\"}}";
33char subscribeLEDTopic[] = "/v1/device/yourDeviceId/sensor/LED/csv";
34long previousHBTime = 0; // previous HB previous time
35long previousRawTime = 0; // previous Raw previous time
36int led_value = 0; // led value(1:high,0:off)
```

(3)MQTT Publish 及 Subscribe 相關內容:參考 CHT IoT 平台之 MQTT 協定介紹，  
<https://iot.cht.com.tw/iot/doc/mqtt>

- **HeartBeat:** 每 5 秒將 publishHBPayload 訊息 publish 至 publishHBTopic 之 Topic，可讓平台確認設備仍正常運作，且可確保 MQTT 連線不會 Timeout(預設 15 秒)。
- **發佈感測資料:** 每 30 秒蒐集的溫度資料透過 publishRawPayload 訊息 publish 至 publishRawTopic 之 Topic，可讓平台收集到最新的溫度資料。
- **yourDeviceId:**請填入您的專案中的設備編號共 9 碼(由數字所組成)。

```

1 // Connect to the network
2 #include <WiFi.h>
3 #include <OneWire.h>
4 #include <DallasTemperature.h>
5 #include <MQTT.h>
6 #include <Arduino.h>
7
8 // OneWire setup
9 #define ONE_WIRE_ADDRESS 0x1B5687C8946A0A
10 OneWire oneWire(ONE_WIRE_ADDRESS);
11 DallasTemperature sensor(&oneWire);
12
13 // WiFi setup
14 const char* ssid = "yourNetwork";
15 const char* pass = "secretPassword";
16
17 // MQTT setup
18 const char* mqttServer = "iot.cht.com.tw";
19 const int mqttPort = 1883;
20 const char* clientId = "amebaClient";
21 const char* userpass = "yourApiKey";
22
23 // MQTT topics
24 const char* hbTopic = "/v1/device/yourDeviceId/heartbeat";
25 const char* rawTopic = "/v1/device/yourDeviceId/rawdata";
26 const char* ledTopic = "/v1/device/yourDeviceId/sensor/LED/csv";
27
28 // MQTT callbacks
29 void mqttCallback(char* topic, byte* payload, unsigned int length);
30 void subscribeLEDTopic();
31
32 // Global variables
33 long previousHBTime = 0;
34 long previousRawTime = 0;
35 int led_value = 0;
36
37 // Setup function
38 void setup() {
39   Serial.begin(115200);
40   while (!Serial) {
41     ; // wait for serial port to connect
42   }
43   Serial.println("Starting...");
44
45   // WiFi
46   WiFi.begin(ssid, pass);
47   while (WiFi.status() != WL_CONNECTED) {
48     delay(5000);
49     Serial.print("Connecting to WiFi... ");
50   }
51   Serial.println("Connected");
52
53   // MQTT
54   MQTTClient = new MQTTClient(mqttServer, mqttPort, clientId, userpass);
55   MQTTClient->connect();
56   subscribeHBTopic();
57   subscribeLEDTopic();
58 }
59
60 // Loop function
61 void loop() {
62   // Read temperature
63   float tempC = sensor.readTemperature();
64   float tempF = tempC * 9.0 / 5.0 + 32.0;
65   Serial.print("Temperature: ");
66   Serial.print(tempC);
67   Serial.print("C ");
68   Serial.print(tempF);
69   Serial.println("F");
70
71   // Send heartbeat
72   if (millis() - previousHBTime > 10000) {
73     String hbPayload = "{\"pulse\":" + String(tempC) + "\"}";
74     MQTTClient->publish(hbTopic, hbPayload.c_str(), hbPayload.length());
75     previousHBTime = millis();
76   }
77
78   // Send raw data
79   if (millis() - previousRawTime > 10000) {
80     String rawPayload = "{\"id\":\"TMP1\", \"value\":{\"99,99\"}}";
81     MQTTClient->publish(rawTopic, rawPayload.c_str(), rawPayload.length());
82     previousRawTime = millis();
83   }
84
85   // Control LED
86   digitalWrite(LED_BUILTIN, led_value);
87   delay(1000);
88   led_value = !led_value;
89 }
90
91 // MQTT callback functions
92 void mqttCallback(char* topic, byte* payload, unsigned int length) {
93   Serial.print("MQTT Callback: ");
94   Serial.print(topic);
95   Serial.print(" - ");
96   for (int i = 0; i < length; i++) {
97     Serial.print((char)payload[i]);
98   }
99   Serial.println();
100 }
101
102 void subscribeLEDTopic() {
103   MQTTClient->subscribe(ledTopic, mqttCallback);
104 }
105
106 // Main function
107 int main() {
108   setup();
109   while (1) {
110     loop();
111   }
112 }

```

(4)callback 函式:

- 訂閱感測資料變更通知(CSV): 處理 IoT 平台所傳送的訂閱 LED 感測資料變更通知。根據收到變動通知 CSV 內容中最後一欄數值來決定要點亮或熄滅 LED。藉由此方式達到 IoT 平台可遠端控制 LED 之亮滅。

(ex1) 點亮 LED :2017-03-03T10:36:12.892Z,861413392,LED1,1

(ex2) 熄滅 LED:2017-03-03T02:46:38.928Z,861413392,LED1,0

```

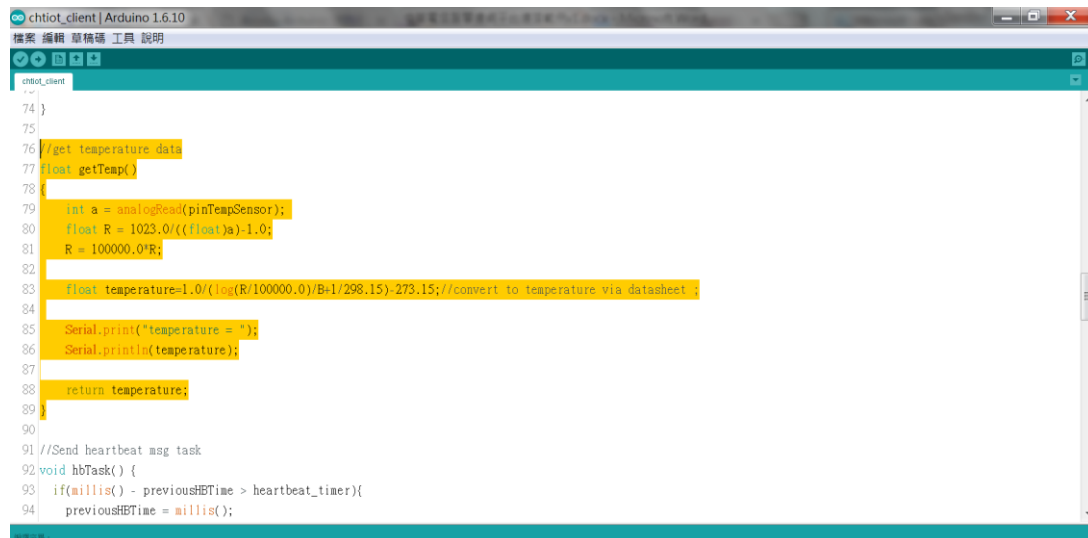
53 Serial.println();
54
55 //check LED topic or not
56 if (strcmp(topic, subscribeLEDTopic) == 0){
57     while ((str = strtok_r(p, " ", &p)) != NULL){ // delimiter is the semicolon
58         element_cnt++;
59
60         //check LED command is 0 or 1
61         if (element_cnt == 4){
62             if (str[0] == '0'){
63                 Serial.println("Turn off LED");
64                 digitalWrite(13, LOW); // turn the LED off by making the voltage LOW
65             }else if (str[0] == '1'){
66                 Serial.println("Turn on LED");
67                 digitalWrite(13, HIGH); // turn the LED on (HIGH is the voltage level)
68             }
69         }
70     }
71 }
72
73

```

### (5) getTemp 函式

- 透過類比溫度計之溫度與電阻值變化之公式取得溫度值，詳細說明可參考 Grove - Temperature Sensor 網站

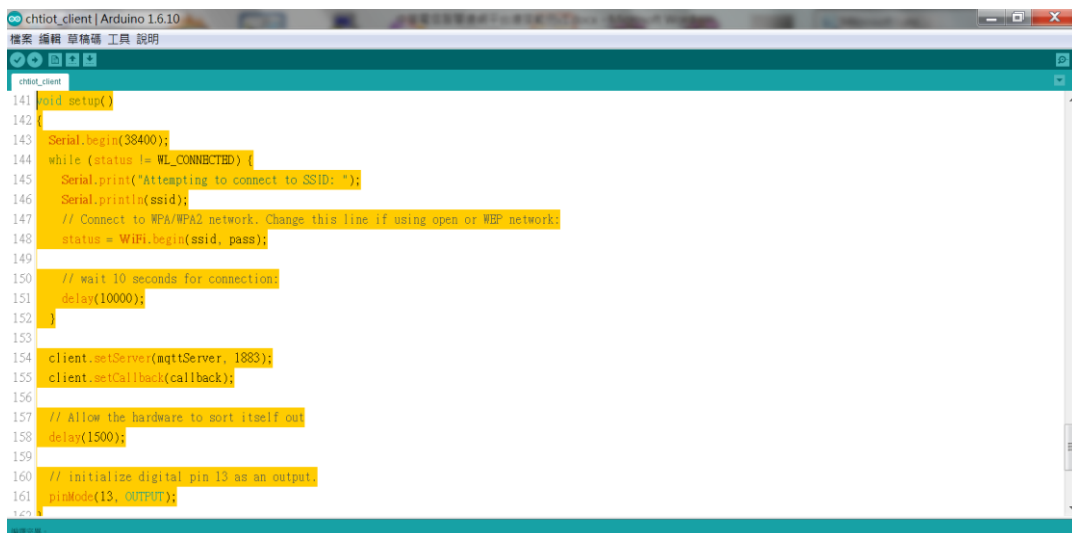
[http://wiki.seeed.cc/Grove-Temperature\\_Sensor\\_V1.2/](http://wiki.seeed.cc/Grove-Temperature_Sensor_V1.2/)



```
74 }
75
76 //get temperature data
77 float getTemp()
78 {
79     int a = analogRead(pinTempSensor);
80     float R = 1023.0/((float)a)-1.0;
81     R = 100000.0/R;
82
83     float temperature=1.0/(log(R/100000.0)/B-1/298.15)-273.15;//convert to temperature via datasheet
84
85     Serial.print("temperature = ");
86     Serial.println(temperature);
87
88     return temperature;
89 }
90
91 //Send heartbeat msg task
92 void hbTask() {
93     if(millis() - previousHBTime > heartbeat_timer){
94         previousHBTime = millis();
```

### (6) setup 函式:

- 初始化序列埠:將序列埠速度設為 38.4kbps 作為 debug 訊息輸出之用途。
- WiFi 連線:連線至 AP，若連線失敗會於 10 秒後重新連線。
- MQTT 設定:將 MQTT Server IP/Port/Callback 函式設定完成。
- LED 設定:設定 D13 腳位為 output 以利控制 LED 亮滅。

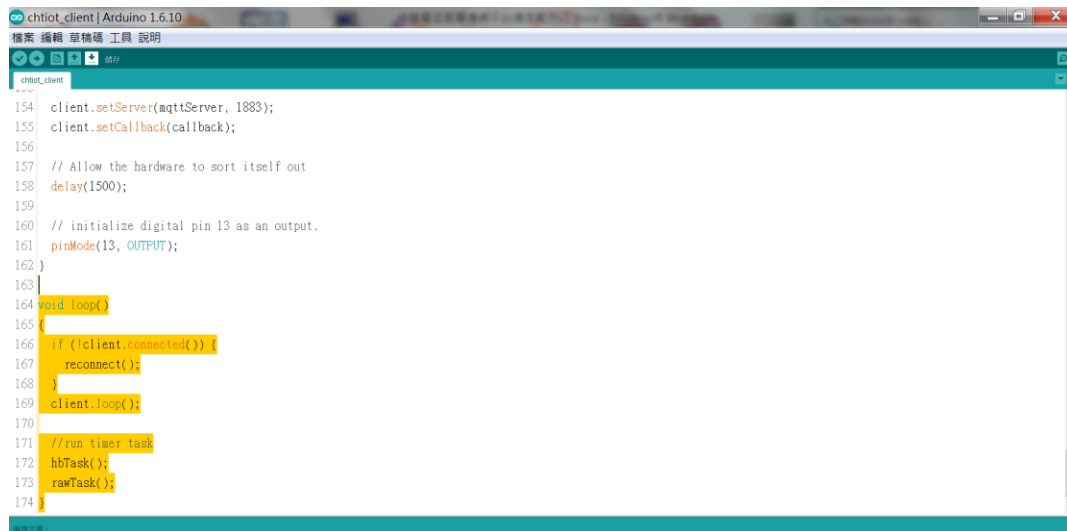


```
141 void setup()
142 {
143     Serial.begin(38400);
144     while (status != WL_CONNECTED) {
145         Serial.print("Attempting to connect to SSID: ");
146         Serial.println(ssid);
147         // Connect to WPA/WPA2 network. Change this line if using open or WEP network
148         status = WiFi.begin(ssid, pass);
149
150         // wait 10 seconds for connection
151         delay(10000);
152     }
153
154     client.setServer(mqttServer, 1883);
155     client.setCallback(callback);
156
157     // Allow the hardware to sort itself out
158     delay(1500);
159
160     // initialize digital pin 13 as an output
161     pinMode(13, OUTPUT);
162 }
```



## (7)loop 函式

- reconnect():檢查 client 連線若中斷時則進行 MQTT 重新連接之工作。
- client.loop():檢查是否有收到 MQTT 訂閱的資料送達。
- hbTask():每 5 秒傳送一筆 Heartbeat 訊息至 IoT 平台。
- rawTask():每 30 秒傳送一筆所收集的溫度訊息至 IoT 平台。



```
154 client.setServer(mqttServer, 1883);
155 client.setCallback(callback);
156
157 // Allow the hardware to sort itself out
158 delay(1500);
159
160 // initialize digital pin 13 as an output.
161 pinMode(13, OUTPUT);
162 }
163
164 void loop()
165 {
166   if (!client.connected()) {
167     reconnect();
168   }
169   client.loop();
170
171   //run timer task
172   hbTask();
173   rawTask();
174 }
```

## [遠端控制 LED 亮滅]

- 控制 LED 亮滅: 於 LED1 將 Switch 切換成 ☒ ON ☐ OFF，觀察 Ameba 板子的 LED 是否點亮；反之，於 LED1 將 Switch 切換成 ☐ OFF，觀察 Ameba 板子的 LED 是否熄滅。



IoT 智慧聯網平台

開發者中心 專案管理 產品登錄 帳號資訊 登出

86141 Ameba1 Add LED and Temp sensor for controlling LED and monitoring temperature general

顯示第 1 至 1 項結果，共 1 項

設備編號: 86141 感應器列表

LED1	TMP1
	

## [溫度資料收集結果]

- 顯示感測器歷史資料:於 TMP1 點選  後，可指定時間區間(採 UTC 時間)取得感測資料的歷史紀錄，也可以繪製成趨勢變化圖，如下圖。

