

# HUB5168+應用教學

## DAY1

講師: 楊穎煜 (Malo)



# 基礎介紹說明

# HUB5168+

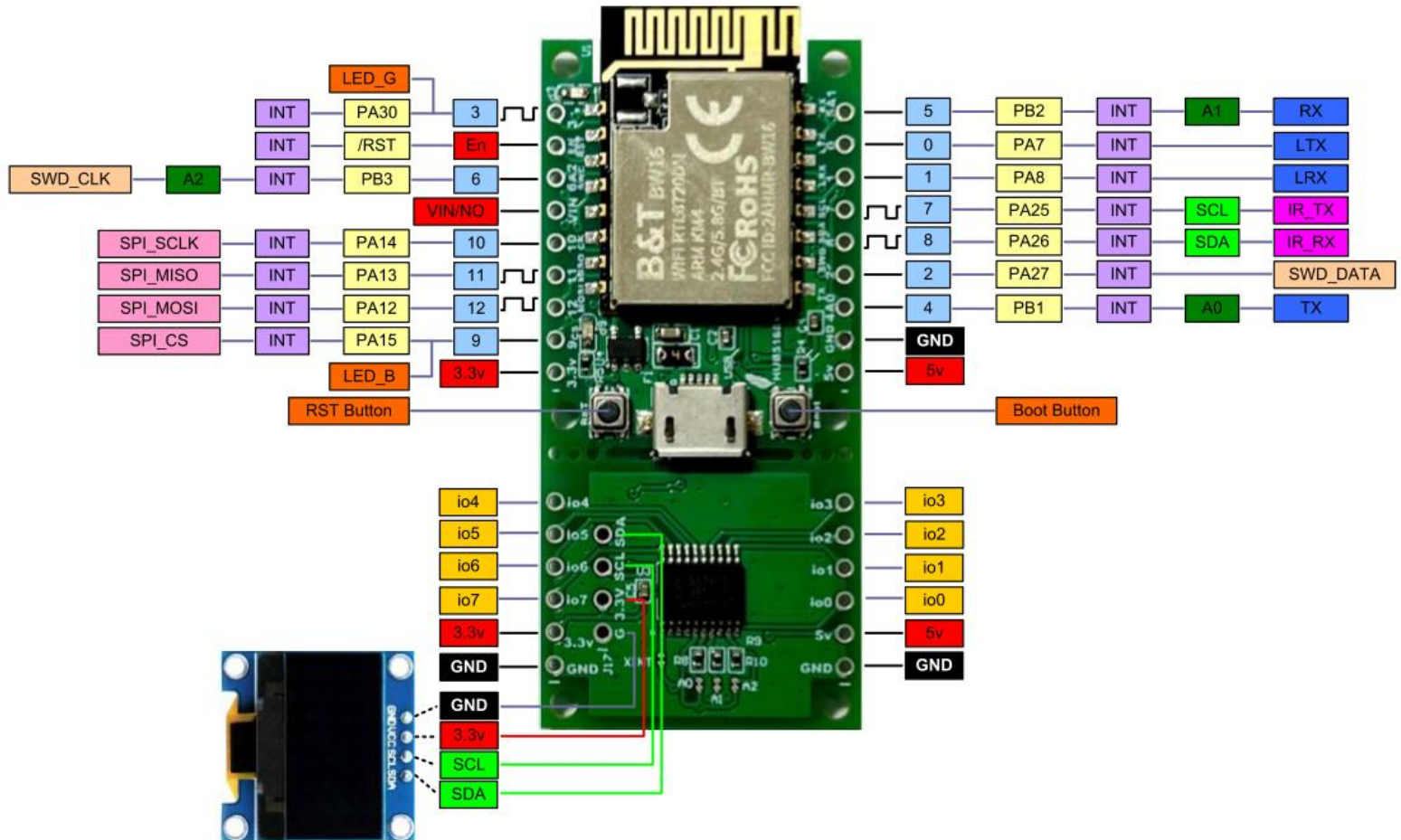
- 延續了 DSI5168 輕薄短小
- 支援 2.4G, 5G 雙頻 Wi-Fi 和低功耗 BLE5.0 及雙核心處理器
- 擴充模組可直上 OLED
- 為國產晶片於物聯網產品開發的好工具。

# HUB5168+

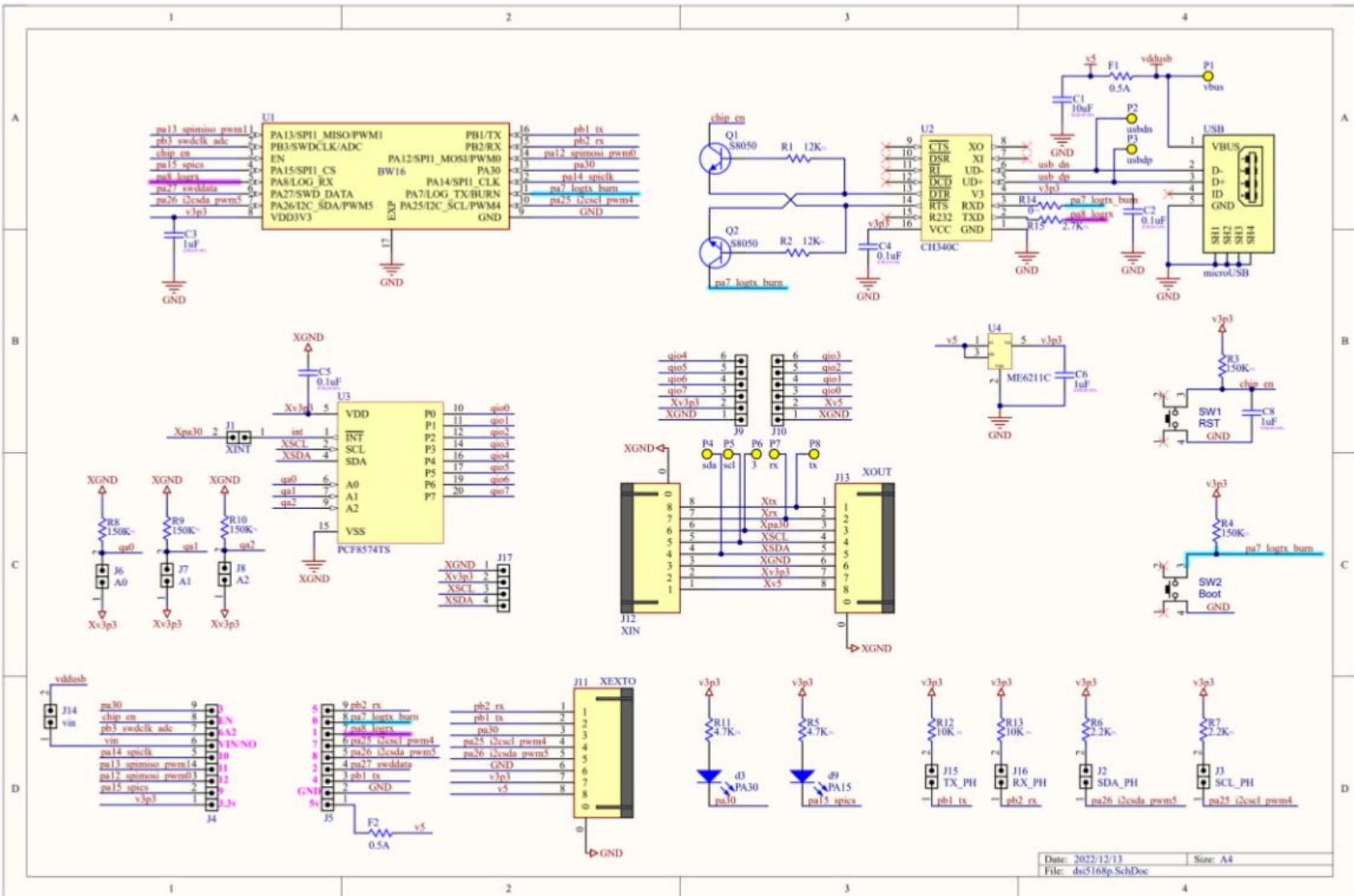
硬體功能	HUB 5168+
Wireless LAN	2.4G, 5G 雙頻 Wi-Fi
BLE BLE	低功耗 BLE5.0
Chipset	RTL8720DN
MCU	KM4 Arm Cortex-M4 core @ 200 MHz KM0 Arm Cortex-M0 core @ 20 MHz
Length	34.7mm
Width	25.4mm
Flash memory	16Mbit
HS_SRAM	512KB
LP_SRAM	64KB
Clock Speed	200 MHz
I/O	21
ADC	3
SPI	1
UART	2
I2C	1

# HUB5168+

LEGEND
POWER
GROUND
PIN NAME
Analog ADC
INTERRUPT
SPI
I2C
UART
IR
QIO
BOARD HARDWARE
SWD debug
PWM pin



# HUB5168+



# 接線圖



# Arduino安裝說明

- 盡量使用安裝版
- 使用1.8.19以上版本(講師使用1.8.19)

Legacy IDE (1.8.X)



## Arduino IDE 1.8.19

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. This software can be used with any Arduino board.

Refer to the [Arduino IDE 1.x documentation](#) for installation instructions.

### SOURCE CODE

Active development of the Arduino software is [hosted by GitHub](#). See the instructions for [building the code](#). Latest release source code archives are available [here](#). The archives are PGP-signed so they can be verified using [this](#) gpg key.

### DOWNLOAD OPTIONS

**Windows** Win 7 and newer

**Windows** ZIP file

**Windows app** Win 8.1 or 10



**Linux** 32 bits

**Linux** 64 bits

**Linux** ARM 32 bits

**Linux** ARM 64 bits

**Mac OS X** 10.10 or newer

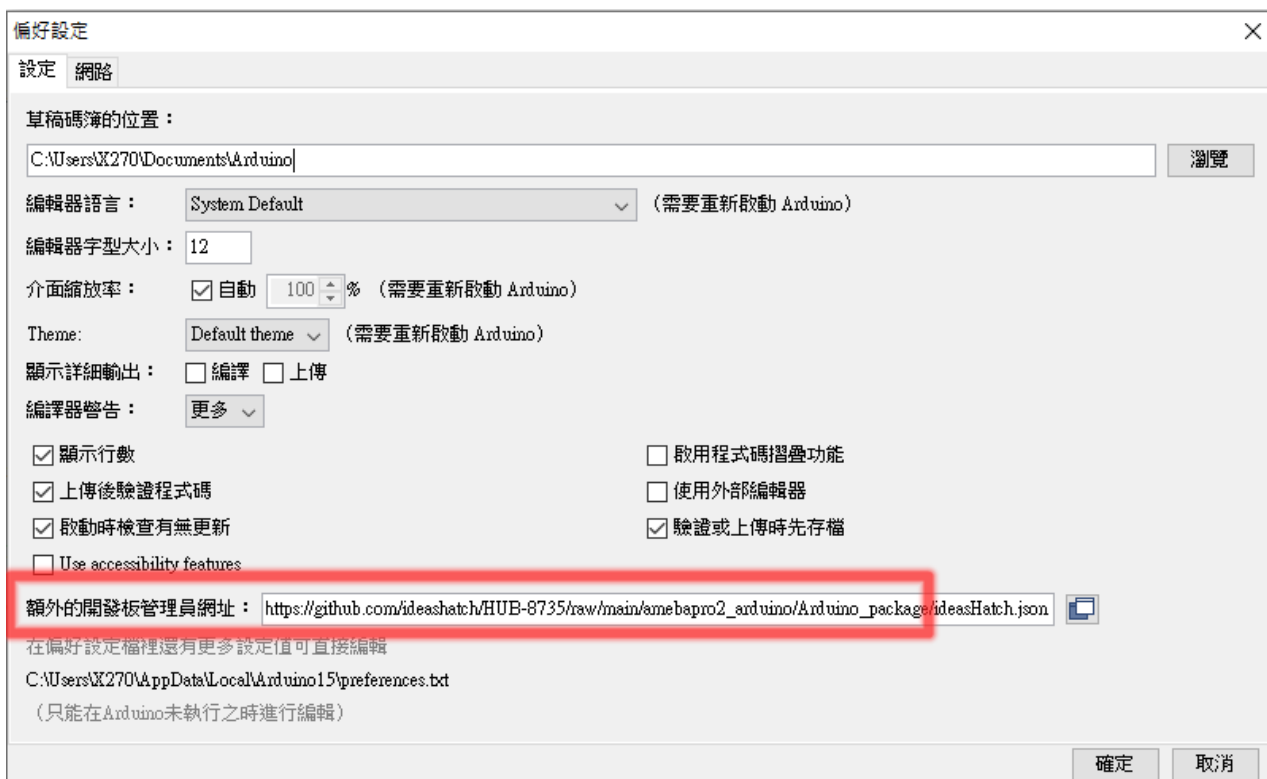
[Release Notes](#)

[Checksums \(sha512\)](#)



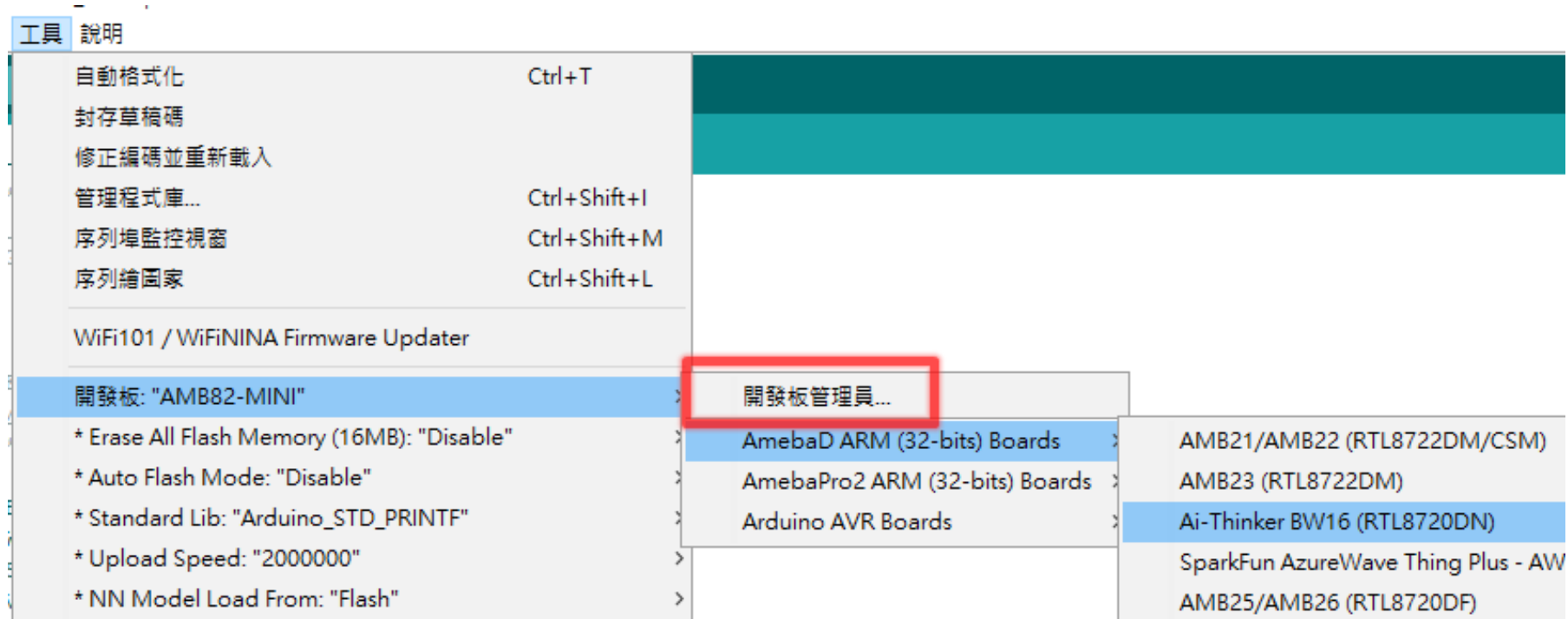
# Arduino 安裝說明

- 開啟Arduino IDE，打開「檔案」/「偏好設定」
- 在「開發板管理員網址」填入網址：  
[https://github.com/ambiot/ambd\\_arduino/raw/master/Arduino\\_package/package\\_realtek.com\\_amebad\\_index.json](https://github.com/ambiot/ambd_arduino/raw/master/Arduino_package/package_realtek.com_amebad_index.json)



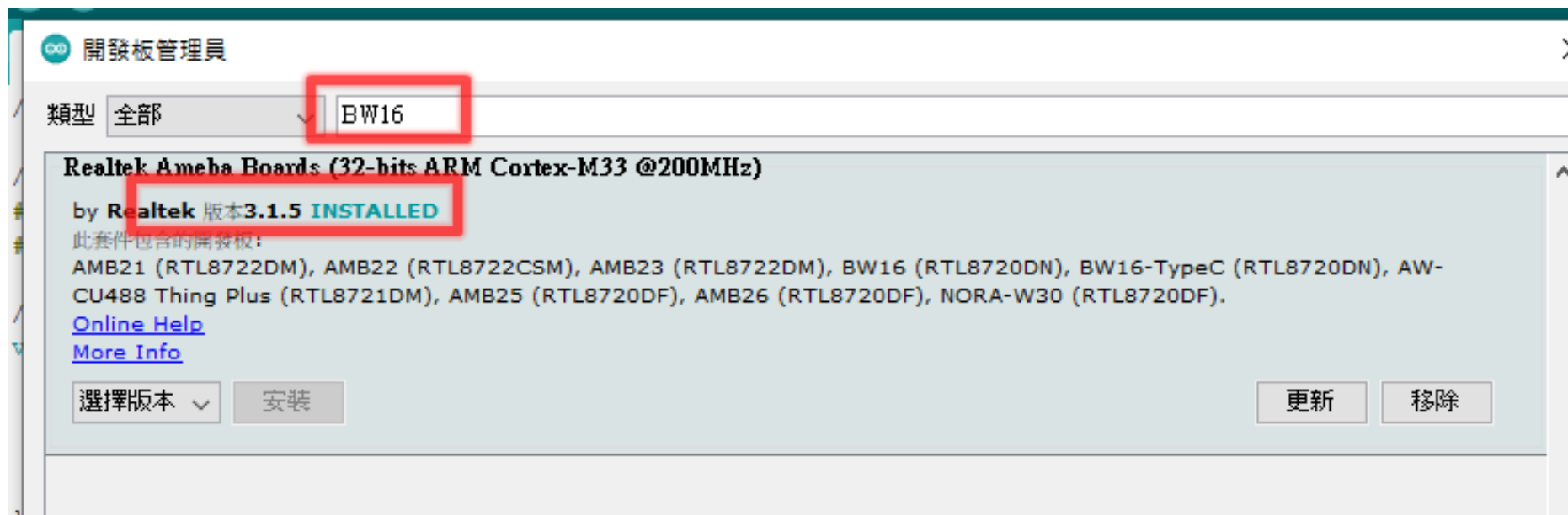
# Arduino安裝說明

- 選擇「工具」/「開發板管理員」



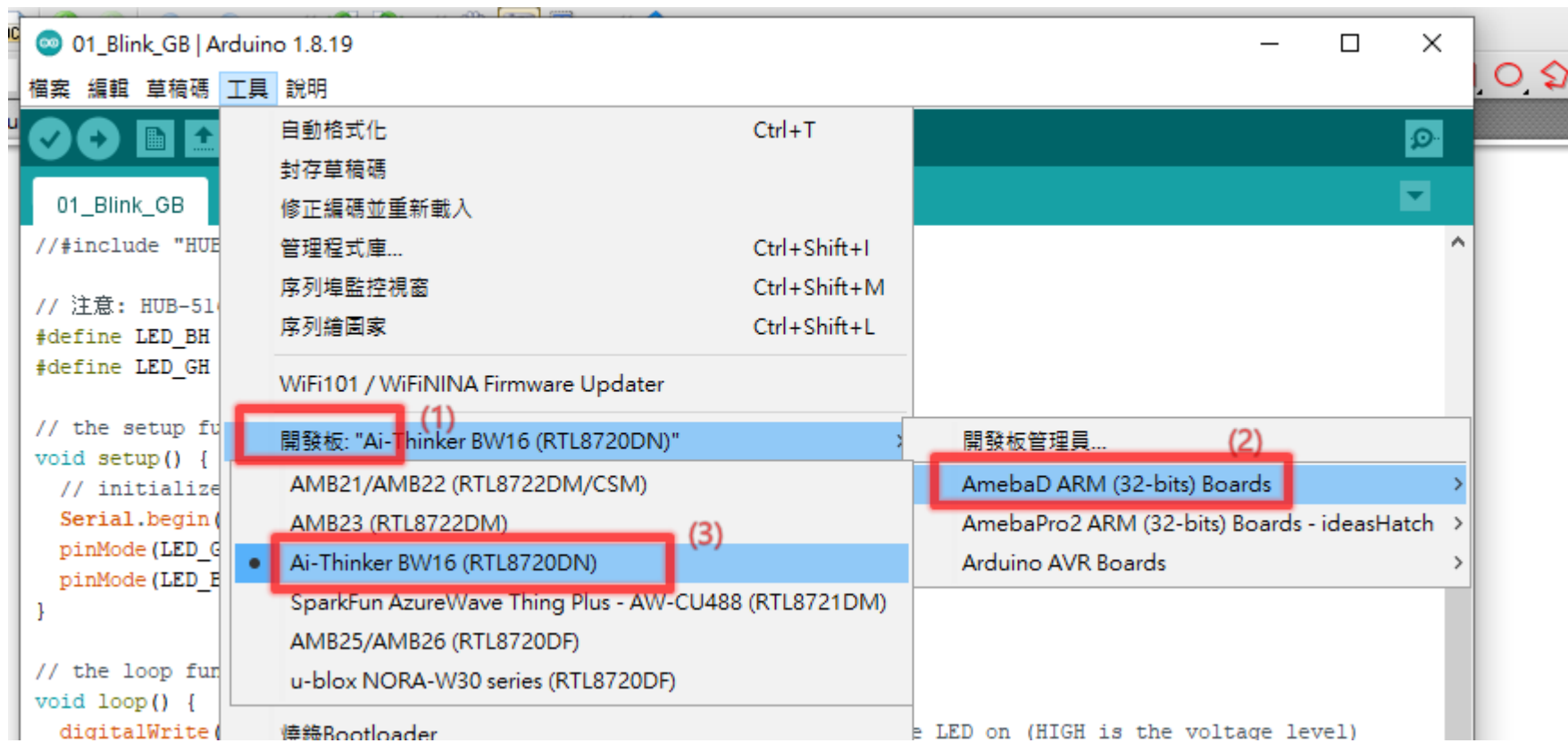
# Arduino安裝說明

- 填入BW16，以下為講師安裝的版本



# Arduino安裝說明

- 安裝好開發板套件後，再選擇RTL8720DN (此為realtek晶片的名稱)



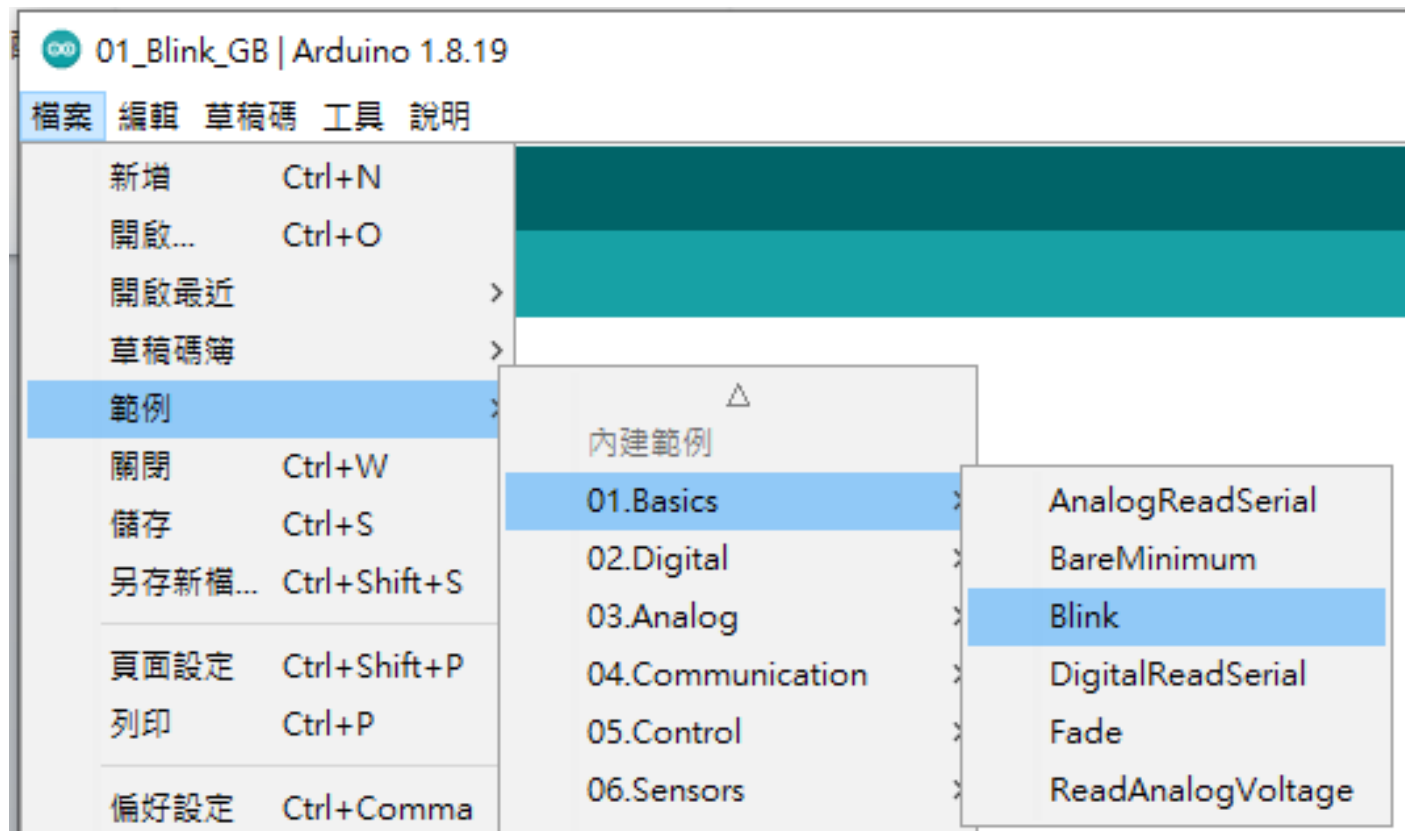
# Arduino安裝說明

- 此為講師的設定值
- 這一片模組有支援auto upload模式



# Arduino安裝說明

- 環境設定完成後，先載入第一個Arduino的demo程式，用Led說Hello



# 測試範例: Blink

- 燒錄測試最基本的程式!你會發現...

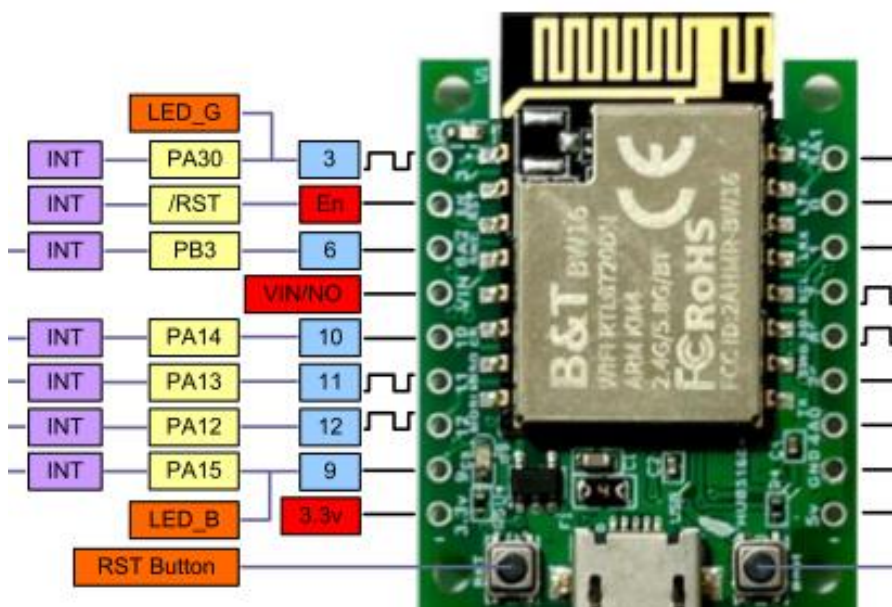
# 測試範例: Blink

- ❑ 失敗...沒有Led在閃爍!!



# 測試範例: Blink

□ 原來!! 要看電路圖



```
01_Blink | Arduino 1.8.19
檔案 編輯 草稿碼 工具 說明

01_Blink
^/

// 注意: HUB-5168的LED要反向控制
#define LED_B 9

// the setup function runs once when you press reset or po
void setup() {
  // initialize digital pin LED_BUILTIN as an output.
  Serial.begin(115200);
  pinMode(LED_B, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
  digitalWrite(LED_B, HIGH); // turn the LED on (HIGH is
                             // wait for a second
  delay(200);
  digitalWrite(LED_B, LOW); // turn the LED off by maki
                             // wait for a second
  delay(200);

  Serial.println("hub5168+");
}
```

# 測試範例: Blink

- 重新燒錄後  
可以看到  
藍色Led快閃



The screenshot shows the Arduino IDE interface. At the top, it says "01\_Blink | Arduino 1.8.19". Below that are tabs for "檔案", "編輯", "草稿碼", "工具", and "說明". The "草稿碼" tab is active, showing the Blink example code. The code is in Chinese and includes comments in Chinese. The code is as follows:

```
This example code is in the public domain.

https://www.arduino.cc/en/Tutorial/BuiltInExamples/Blink
*/

// 注意: HUB-5168的LED要反向控制
#define LED_B 9

// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin LED_BUILTIN as an output.
  Serial.begin(115200);
  pinMode(LED_B, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
  digitalWrite(LED_B, HIGH);    // turn the LED on (HIGH is the voltage level)
  delay(200);                  // wait for a second
  digitalWrite(LED_B, LOW);     // turn the LED off by making the voltage LOW
  delay(200);                  // wait for a second
}
```

Below the code, it says "上傳完畢。" (Upload complete.). Then, it shows the upload progress bar and the message "Upload Image done. All images are sent successfully!". The message "Upload Image done. All images are sent successfully!" is highlighted with a red box.

39

At-Thinker BW16 (RTL8720DN), Dis

# GPIO基本說明-練習題

Blink

# 測試範例: Blink

- 請修改剛剛的程式，並研究電路圖上的GPIO腳位，控制綠燈



The screenshot shows the Arduino IDE interface with the '01\_Blink' example open. The title bar reads '01\_Blink | Arduino 1.8.19'. Below the title bar is a menu bar with '檔案', '編輯', '草稿碼', '工具', and '說明'. A toolbar with icons for checkmark, run, file, upload, and download is visible. The code editor shows the following content:

```
01_Blink

This example code is in the public domain.

https://www.arduino.cc/en/Tutorial/BuiltInExamples/Blink
*/

// 注意: HUB-5168的LED要反向控制
#define LED_B 9

// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin LED_BUILTIN as an output.
  Serial.begin(115200);
  pinMode(LED_B, OUTPUT);
}

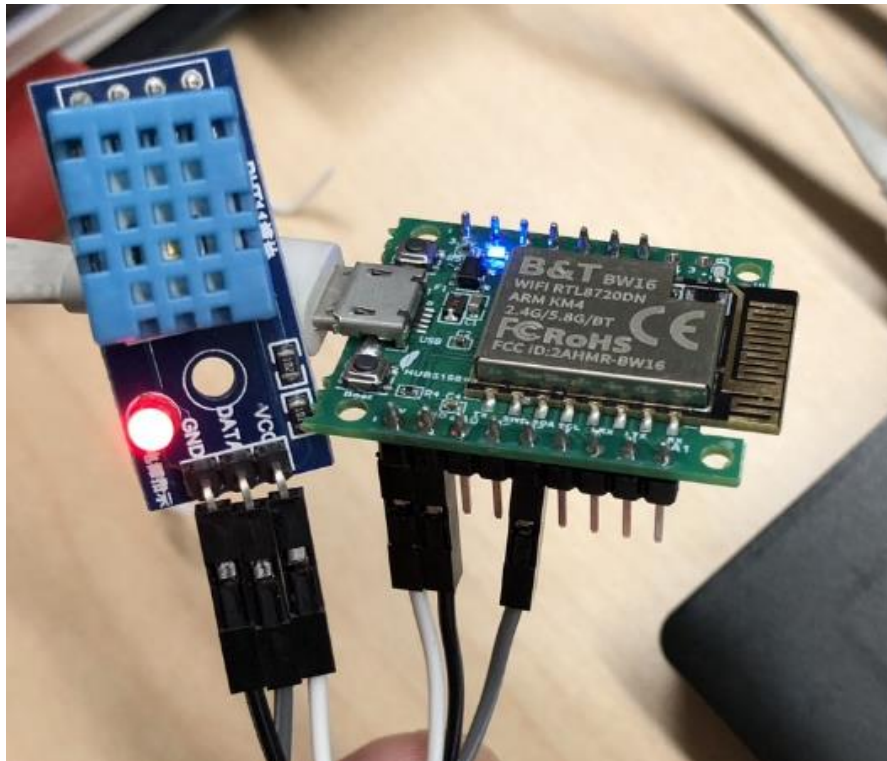
// the loop function runs over and over again forever
void loop() {
  digitalWrite(LED_B, HIGH);  // turn the LED on (HIGH is the voltage level)
  delay(200);                 // wait for a second
  digitalWrite(LED_B, LOW);   // turn the LED off by making the voltage LOW
  delay(200);                 // wait for a second
```

# 温度量測

DHT-11

# 測試範例: 02\_dht11

- 接線如下圖，使用PIN8讀取DHT-11的溫溼度數值



# 測試範例: 02\_dht11

- 打開com port可以看到溫溼度數值

```
#calibration_ok:[2:19:11]
DHTxx test!
Humidity: 40.00%  Temperature: 29.50°C 85.10°F  Heat index: 29.13°C 84.43°F
Humidity: 40.00%  Temperature: 29.40°C 84.92°F  Heat index: 29.02°C 84.24°F
Humidity: 40.00%  Temperature: 29.40°C 84.92°F  Heat index: 29.02°C 84.24°F
Humidity: 39.00%  Temperature: 29.40°C 84.92°F  Heat index: 28.92°C 84.07°F
Humidity: 39.00%  Temperature: 29.40°C 84.92°F  Heat index: 28.92°C 84.07°F
Humidity: 39.00%  Temperature: 29.40°C 84.92°F  Heat index: 28.92°C 84.07°F
Humidity: 39.00%  Temperature: 29.40°C 84.92°F  Heat index: 28.92°C 84.07°F
```

```
#calibration_ok:[2:19:11]
DHTxx test!
Humidity: 44.00%  Temperature: 26.80°C 80.24°F  Heat index: 26.91°C 80.45°F
Humidity: 43.00%  Temperature: 26.90°C 80.42°F  Heat index: 26.94°C 80.49°F
Humidity: 43.00%  Temperature: 26.90°C 80.42°F  Heat index: 26.94°C 80.49°F
Humidity: 43.00%  Temperature: 26.90°C 80.42°F  Heat index: 26.94°C 80.49°F
Humidity: 43.00%  Temperature: 26.90°C 80.42°F  Heat index: 26.94°C 80.49°F
Humidity: 43.00%  Temperature: 26.80°C 80.24°F  Heat index: 26.86°C 80.36°F
Failed to read from DHT sensor!
Failed to read from DHT sensor!
```

沒有插對的pin腳時

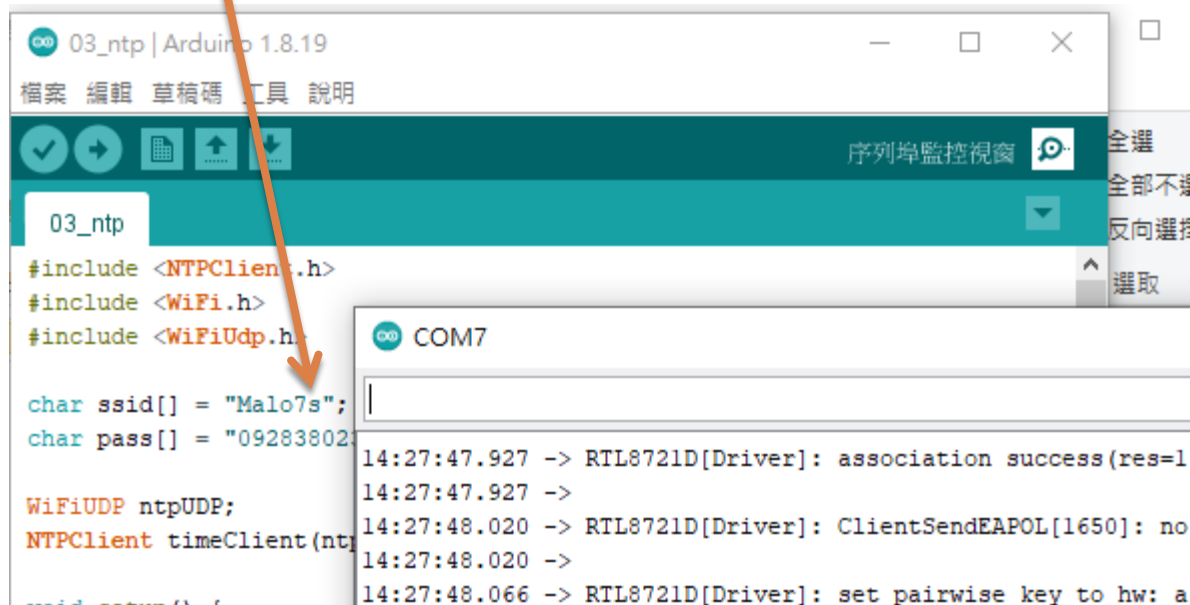
# WiFi使用

NTP



# 測試範例: 03\_ntp

- 修改ssid, pass , 再燒錄



The screenshot shows the Arduino IDE interface. The main window displays the sketch '03\_ntp' with the following code:

```
#include <NTPClient.h>
#include <WiFi.h>
#include <WiFiUdp.h>

char ssid[] = "Malo7s";
char pass[] = "09283802";

WiFiUDP ntpUDP;
NTPClient timeClient(ntpUDP, "pool.ntp.org", 1629494400, 60000);
```

An orange arrow points from the text '修改ssid, pass' in the list above to the variable declarations in the code. The serial monitor window, titled 'COM7', shows the following output:

```
14:27:47.927 -> RTL8721D[Driver]: association success(res=1
14:27:47.927 ->
14:27:48.020 -> RTL8721D[Driver]: ClientSendEAPOL[1650]: no
14:27:48.020 ->
14:27:48.066 -> RTL8721D[Driver]: set pairwise key to hw: a
```

# 測試範例: 03\_ntp

- 輸出如下，我們可以得到校時後的時間

```
#include <WiFiUdp.h>

char ssid[] = "Malo7s";
char pass[] = "09283802";

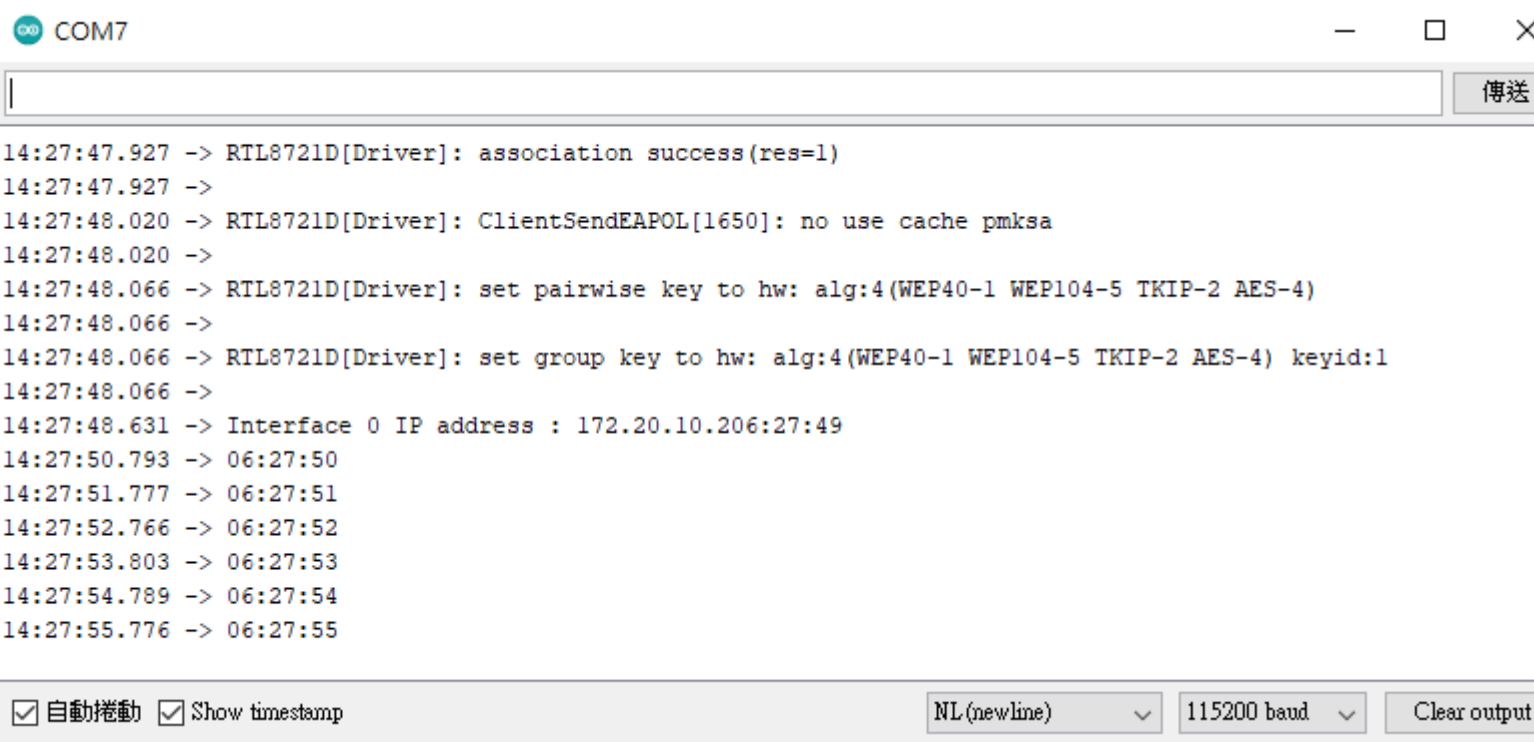
WiFiUDP ntpUDP;
NTPClient timeClient(ntpUDP, "pool.ntp.org", 1629436800, 60);

void setup() {
  Serial.begin(115200);

  WiFi.begin(ssid, pass);

  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }

  timeClient.begin();
}
```



```
COM7

14:27:47.927 -> RTL8721D[Driver]: association success(res=1)
14:27:47.927 ->
14:27:48.020 -> RTL8721D[Driver]: ClientSendEAPOL[1650]: no use cache pmksa
14:27:48.020 ->
14:27:48.066 -> RTL8721D[Driver]: set pairwise key to hw: alg:4(WEP40-1 WEP104-5 TKIP-2 AES-4)
14:27:48.066 ->
14:27:48.066 -> RTL8721D[Driver]: set group key to hw: alg:4(WEP40-1 WEP104-5 TKIP-2 AES-4) keyid:1
14:27:48.066 ->
14:27:48.631 -> Interface 0 IP address : 172.20.10.206:27:49
14:27:50.793 -> 06:27:50
14:27:51.777 -> 06:27:51
14:27:52.766 -> 06:27:52
14:27:53.803 -> 06:27:53
14:27:54.789 -> 06:27:54
14:27:55.776 -> 06:27:55
```

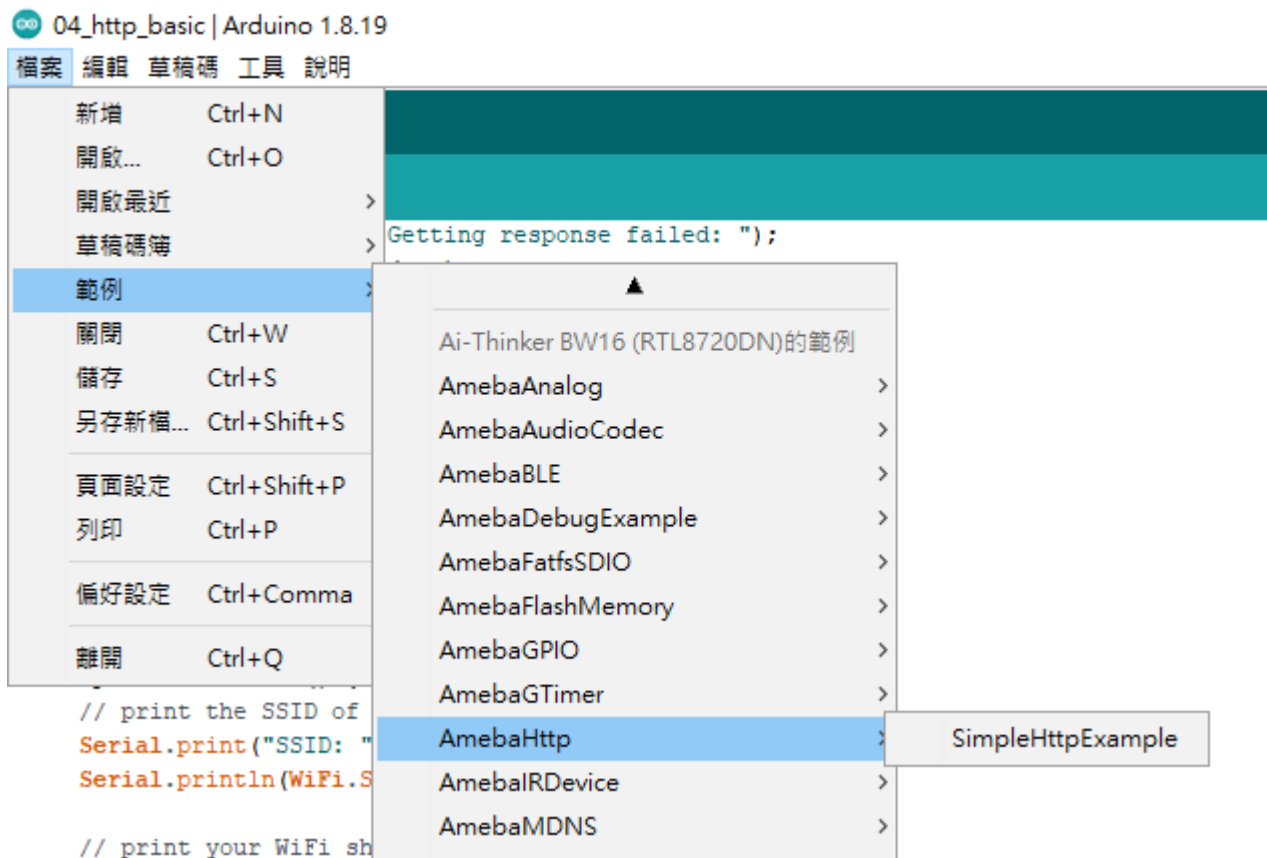
☒ 自動捲動 ☒ Show timestamp NL(newline) 115200 baud Clear output

# http範例

04\_http

# http應用

## □ 打開http的範例程式



# http應用

## □ 修改ssid, pass

```
// This example downloads the URL "http://www.amebaiot.com"

char ssid[] = "your ssid"; // your network SSID (name)
char pass[] = "your password"; // your network password (use
int keyIndex = 0; // your network key Index number (r

// Name of the server we want to connect to
const char kHostname[] = "www.amebaiot.com";

const char kPath[] = "/";
// Number of milliseconds to wait without receiving any data before
const int kNetworkTimeout = 30 * 1000;
// Number of milliseconds to wait if no data is available before
const int kNetworkDelay = 1000;
int status = WL_IDLE_STATUS;
```

# http應用

## □ 可以看到這樣的輸出

```
RTL8721D[Driver]: set group key to hw: alg:4(WEP40-1 WEP104-5 TKIP-2 AES-4) keyid:1

Interface 0 IP address : 172.20.10.2
Connected to wifi
SSID: Malo7s
IP Address: 172.20.10.2
signal strength (RSSI):-51 dBm

[INFO]server_drv.cpp: start_client
[INFO] Create socket successfully

[INFO] Connect to Server successfully!
startedRequest ok
Got status code: 301
Content length is: 315

Body returned follows:
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
<title>301 Moved Permanently</title>
</head><body>
<h1>Moved Permanently</h1>
<p>The document has moved <a href="https://www.amebaiot.com/">here</a>.</p>
<hr>
<address>Apache/2.4.29 (Ubuntu) Server at www.amebaiot.com Port 80</address>
</body></html>
```

# http應用

## □ 讓我們來解析一下輸出

```
RTL8721D[Driver]: set group key to hw: alg:4(WEP40-1 WEP104-5 TKIP-2 AES-4) keyid:1
```

```
Interface 0 IP address : 172.20.10.2
```

```
Connected to wifi
```

```
SSID: Malo7s
```

```
IP Address: 172.20.10.2
```

```
signal strength (RSSI):-51 dBm
```

```
[INFO]server_drv.cpp: start_client
```

```
[INFO] Create socket successfully
```

```
[INFO] Connect to Server successfully!
```

```
startedRequest ok
```

```
Got status code: 301
```

```
Content length is: 315
```

```
Body returned follows:
```

```
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
```

```
<html><head>
```

```
<title>301 Moved Permanently</title>
```

```
</head><body>
```

```
<h1>Moved Permanently</h1>
```

```
<p>The document has moved <a href="https://www.amebaiot.com/">here</a>.</p>
```

```
<hr>
```

```
<address>Apache/2.4.29 (Ubuntu) Server at www.amebaiot.com Port 80</address>
```

```
</body></html>
```

# 要怎麼上傳資料到Server?

## □ 讓我們來解析一下輸出

```
RTL8721D[Driver]: set group key to hw: alg:4(WEP40-1 WEP104-5 TKIP-2 AES-4) keyid:1
```

```
Interface 0 IP address : 172.20.10.2
```

```
Connected to wifi
```

```
SSID: Malo7s
```

```
IP Address: 172.20.10.2
```

```
signal strength (RSSI):-51 dBm
```

```
[INFO]server_drv.cpp: start_client
```

```
[INFO] Create socket successfully
```

```
[INFO] Connect to Server successfully!
```

```
startedRequest ok
```

```
Got status code: 301
```

```
Content length is: 315
```

```
Body returned follows:
```

```
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
```

```
<html><head>
```

```
<title>301 Moved Permanently</title>
```

```
</head><body>
```

```
<h1>Moved Permanently</h1>
```

```
<p>The document has moved <a href="https://www.amebaiot.com/">here</a>.</p>
```

```
<hr>
```

```
<address>Apache/2.4.29 (Ubuntu) Server at www.amebaiot.com Port 80</address>
```

```
</body></html>
```



# 要怎麼上傳資料到Server?

## □ 利用http get操作

```
int keyIndex = 0;           // your network key

// Name of the server we want to connect to
const char kHostname[] = "api.thingspeak.com";

char kPath[128];
float t=30.5;
float h=80;
```

```
WiFiClient c;
HttpClient http(c);

sprintf(kPath, "/update?api_key=5IRM6UNIDXLCAPM1&field1=%.2f&field2=%.2f", t, h);
sprintf(str1, "%s%s", kHostname, kPath);
Serial.println(str1);

err = http.get(kHostname, kPath);
if (err == 0) {
```

# 申請ThingSpeaker平台帳號

- 先申請ThingSpeaker帳號

# 申請ThingSpeak平台帳號

- 進入此分頁，右下角有許多API格式的範例

Private View

Public View

Channel Settings

Sharing

API Keys

Data Import / Export

Write API Key

## API Requests

### Write a Channel Feed

```
GET https://api.thingspeak.com/update?api_key=5IRM6UNIXLCAPM1&field1
```

### Read a Channel Feed

```
GET https://api.thingspeak.com/channels/617643/feeds.json?api_key=EU1
```

### Read a Channel Field

```
GET https://api.thingspeak.com/channels/617643/fields/1.json?api_key=
```

### Read Channel Status Updates

```
GET https://api.thingspeak.com/channels/617643/status.json?api_key=EU
```

# 申請ThingSpeaker平台帳號

- 依照格式可以測試一下讀取功能



The screenshot shows a web browser window with the address bar displaying the URL: `api.thingspeak.com/channels/617643/feeds.json?api_key=EU1JSB28NB8MMM2M&results=2`. The browser's developer tools are open, showing the JSON response from the API. The JSON structure is as follows:

```
{
  "channel": {
    "id": 617643,
    "name": "test",
    "latitude": "0.0",
    "longitude": "0.0",
    "field1": "溫度",
    "field2": "濕度",
    "created_at": "2018-11-03T04:09:10Z",
    "updated_at": "2023-10-20T03:47:27Z",
    "last_entry_id": 281
  },
  "feeds": [
    {
      "created_at": "2023-10-20T03:51:49Z",
      "entry_id": 280,
      "field1": "25",
      "field2": "50"
    },
    {
      "created_at": "2023-10-20T03:52:14Z",
      "entry_id": 281,
      "field1": "28",
      "field2": "81"
    }
  ]
}
```

# 申請ThingSpeaker平台帳號

- 接著測試一下上傳資料的功能
- 以我的API來說，長這個樣子
- [https://api.thingspeak.com/update?api\\_key=5IRM6UNIDXLCAPM1&field1=30&field2=70](https://api.thingspeak.com/update?api_key=5IRM6UNIDXLCAPM1&field1=30&field2=70)

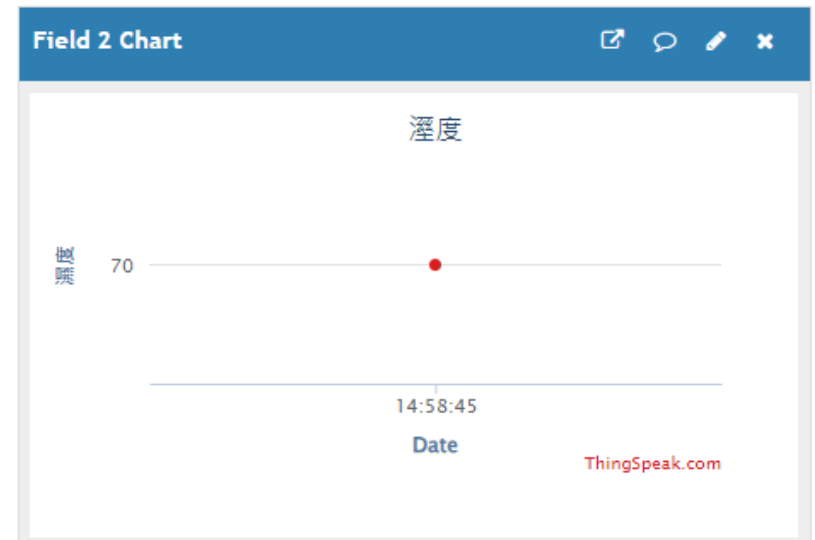
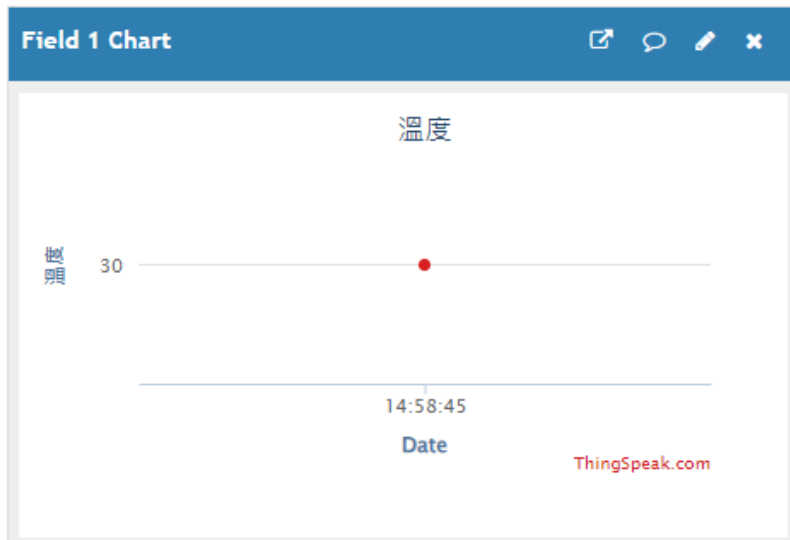
# 申請ThingSpeaker平台帳號

- 我們可以得到這樣的視覺化效果

Created: 5.years.ago

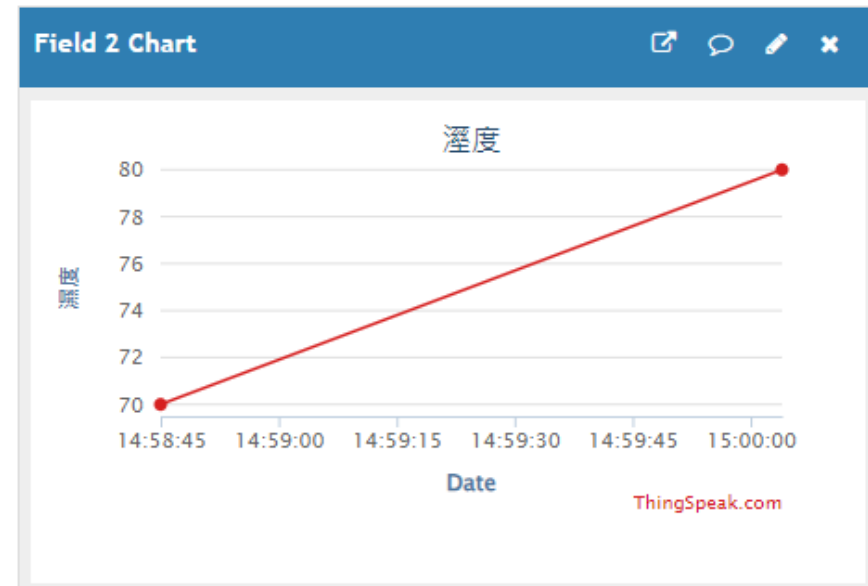
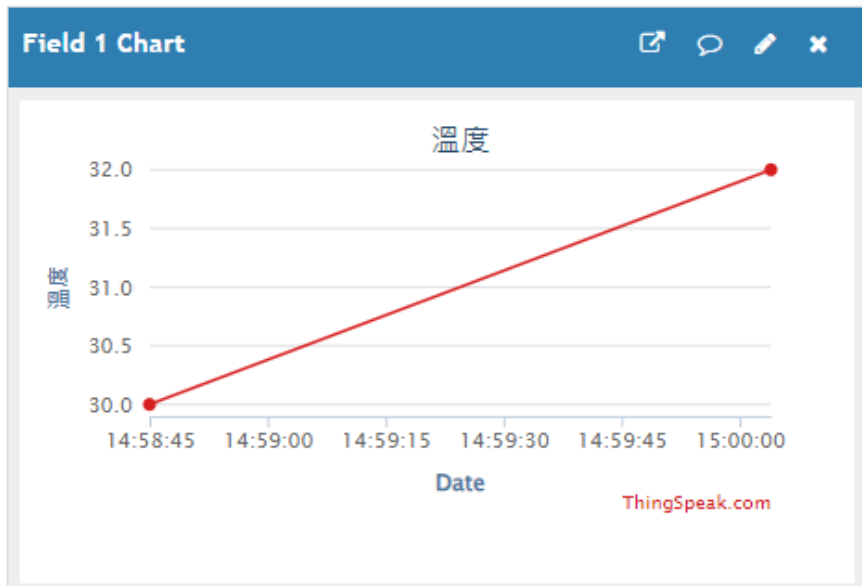
Last entry: less.than.a.minute.ago

Entries: 282



# 申請ThingSpeak平台帳號

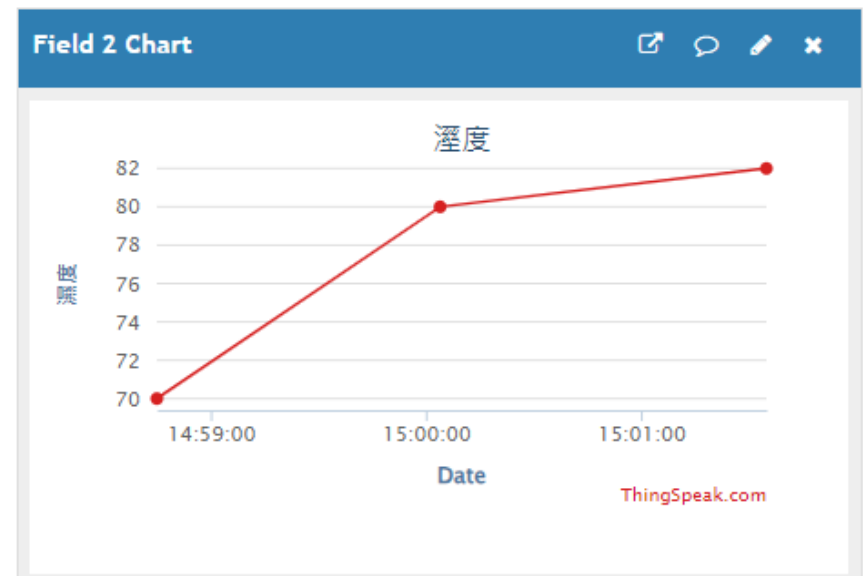
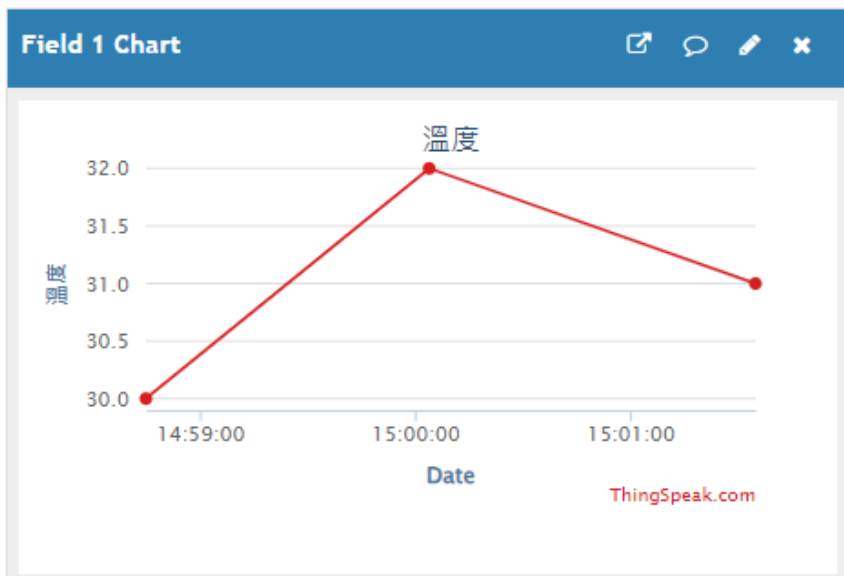
- 但是https對我們來說不好使用，所以調整成:([http://api.thingspeak.com/update?api\\_key=5IRM6UNIDXLCAPM1&field1=30&field2=70](http://api.thingspeak.com/update?api_key=5IRM6UNIDXLCAPM1&field1=30&field2=70))



# 申請ThingSpeak平台帳號

□ 再接著試試

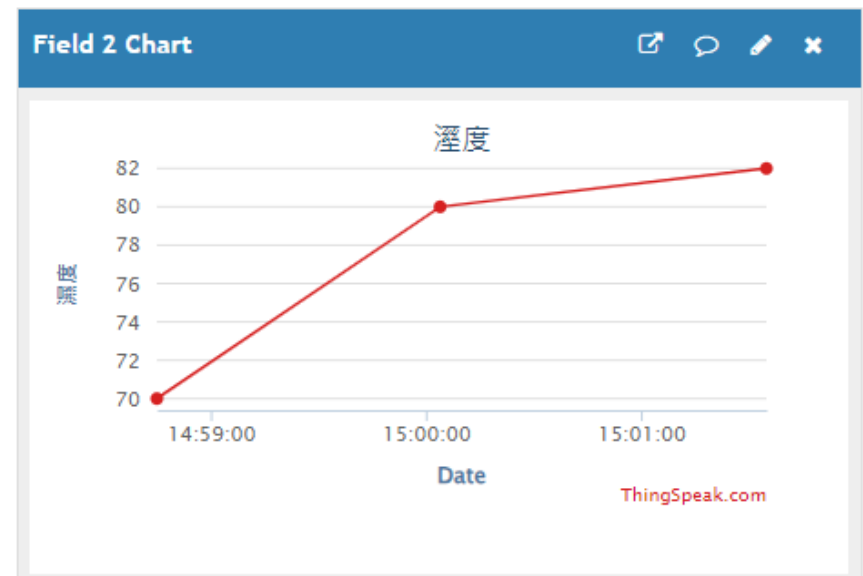
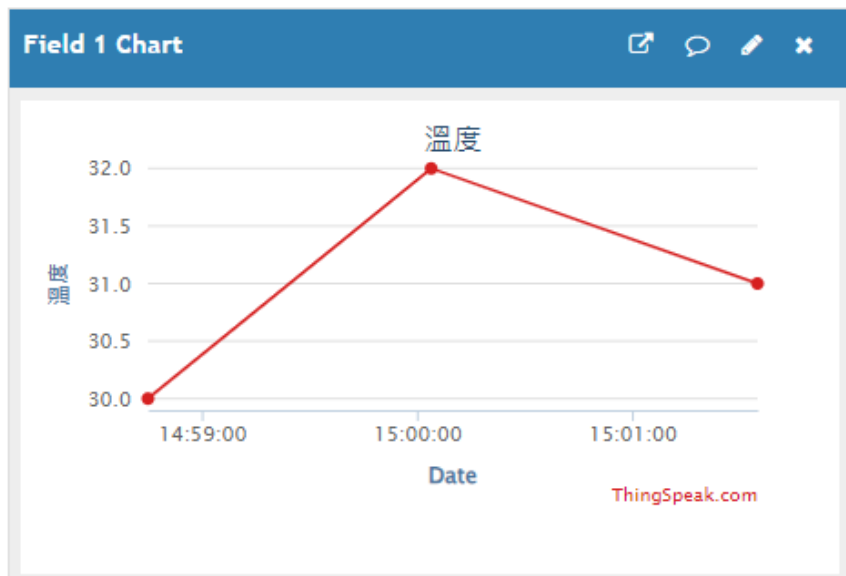
看:([http://api.thingspeak.com/update?api\\_key=5IRM6UNIDXLCAPM1&field1=31&field2=82](http://api.thingspeak.com/update?api_key=5IRM6UNIDXLCAPM1&field1=31&field2=82))





# 整合ThingSpeaker和DHT11

- 以此類推，我們先把http的程式改成可以上傳溫溼度資訊



# 整合ThingSpeaker和DHT11

## □ 上傳DHT11的資料到ThingSpeaker!

```
Interface 0 IP address : 172.20.10.2
Connected to wifi
SSID: Malo7s
IP Address: 172.20.10.2
signal strength (RSSI):-40 dBm
--> Temp: 27.00, Humi: 40.00
```

```
api.thingspeak.com/update?api_key=5IRM6UNIDXLCAPM1&field1=27.00&field2=40.00
```

```
[INFO]server_drv.cpp: start_client
[INFO] Create socket successfully
```

```
[INFO] Connect to Server
startedRequest ok
Got status code: 200
Content length is: 1
```

```
Interface 0 IP address : 172.20.10.2
Connected to wifi
SSID: Malo7s
IP Address: 172.20.10.2
signal strength (RSSI):-33 dBm
--> Temp: 27.10, Humi: 40.00
```

```
api.thingspeak.com/update?api_key=5IRM6UNIDXLCAPM1&field1=27.10&field2=40.00
```

```
[INFO]server_drv.cpp: start_client
[INFO] Create socket successfully

[INFO] Connect to Server successfully!
startedRequest ok
Got status code: 200
Content length is: 1
```

```
Body returned follows:
```

```
2
```

# 整合ThingSpeaker和I

- ❑ 測試前可以先清空原先的測試資料
- ❑ 清空的頁面如右圖，在Chanel Settings中

[Private View](#) [Public View](#) [Channel Settings](#) [Sharing](#)

## Channel Settings

Percentage Complete	30%
Channel ID	617643
Name	<input type="text" value="test"/>
Description	<input type="text"/>
Video URL	<input type="text" value="http://"/>
Show Status	<input type="checkbox"/>

Save Channel

Want to clear all feed data from this Channel?

Clear Channel

Want to delete this Channel?

Delete Channel



# NeoPixel / WS2812B

ws2812b

# NeoPixel (WS2812B)

---

□ API手冊參考([link](#))

# NeoPixel (WS2812B)

- WS2812B class
- fill(index, r, g, b)
- setPixelColor(r, g, b, index, num)
- show()

# 測試範例05:

- 05\_NP\_01.ino: RGB顏色切換

# 測試範例05:

- 05\_NP\_02.ino : 呼吸燈



# 測試範例05:

- 請試著改寫程式，切換不同的顏色閃動呼吸燈
- 請利用目前學到的應用，配合燈罩設計專屬於你的氣氛燈

# 原廠範例

- 05\_WS2812B\_Basics
- 05\_WS2812B\_Patterns