

Android & Arduino Wemos D1 mini

Google™



段維瀚 老師



ESP8266

低功耗、高集成度的 Wi-Fi 芯片

仅需 7 个外围元器件

超宽工作温度范围：-40°C 至 +125°C

ESP8285 – ESP8266 内封 8 Mbit Flash



ESP32

2.4 GHz Wi-Fi 加蓝牙双模芯片
采用 TSMC 低功耗 40nm 技术
功耗性能和射频性能最佳
安全可靠，易于扩展至各种应用



ESP8266 基本介紹

- ESP8266，樂鑫信息科技公司（Espressif Systems）推出的WiFi晶片，價格超低，受到國外人士關注，並且把資料文件從中文翻成英文；後來釋出開發工具SDK，開始湧出各種韌體。社群網站在此，資訊豐富。
- 基本規格如下，詳情可看維基百科ESP8266：
 - CPU，32-bit RISC，Tensilica Xtensa L106，運作頻率80 MHz。
 - 64 KB指令RAM，96 KB資料RAM。
 - 外部Flash：0.5 MB到4 MB（最高可達16 MB）。
 - WiFi：2.4 GHz，IEEE 802.11 b/g/n、WEP or WPA/WPA2，可做為站台、AP存取點、或雙模。
 - 網路協定：TCP/IP、DNS。
 - 16個GPIO腳位，其他還有SPI、I2C、I2S、UART、10-bit ADC。
 - 工作電壓：3.3 V。
 - 功耗：待機小於200uA，正常狀態12mA（平均）到200mA（峰值），持續傳送狀態70mA（平均）到200mA（峰值）。
 - 傳送速率：110到921600 bps。



ESP8266 模組



ESP-01



ESP-02



ESP-03



ESP-04



ESP-05



ESP-06



ESP-07



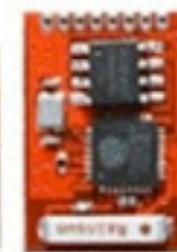
ESP-08



ESP-09



ESP-10



ESP-11



ESP-12



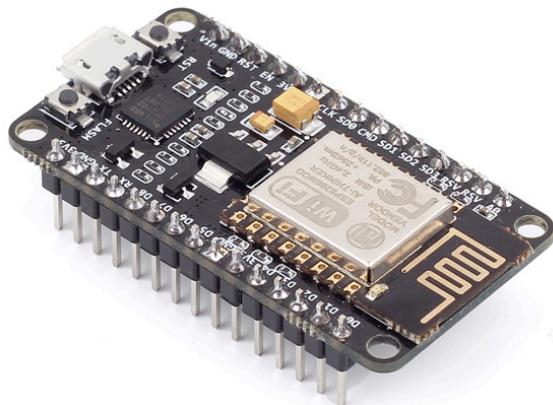
ESP-13



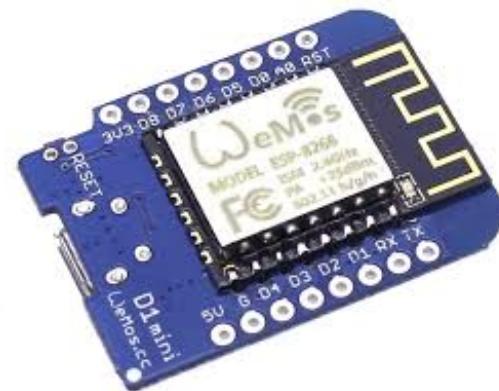
ESP-14



Based on ESP8266



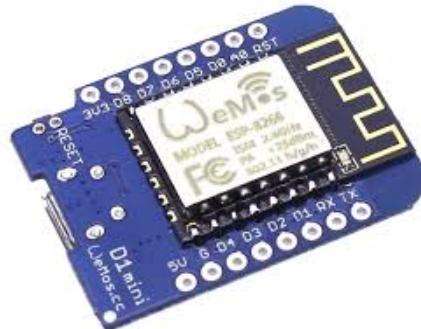
NodeMcu



Wemos D1 mini



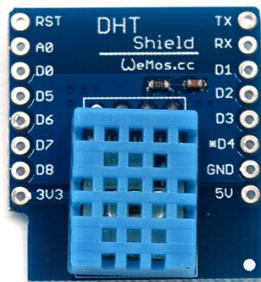
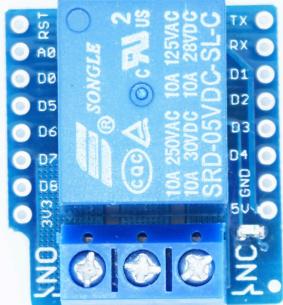
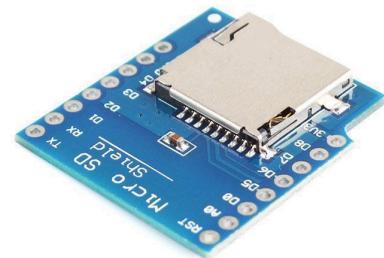
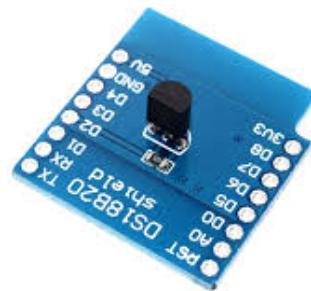
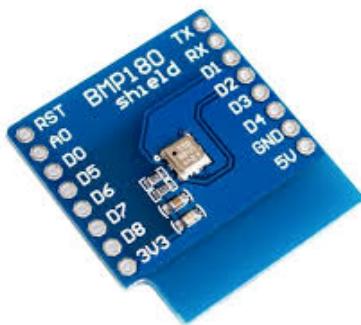
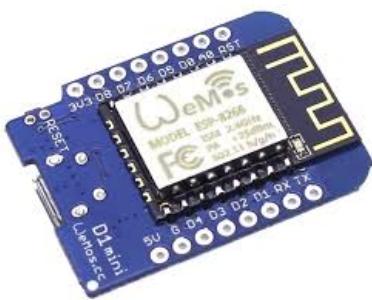
WeMos D1 mini



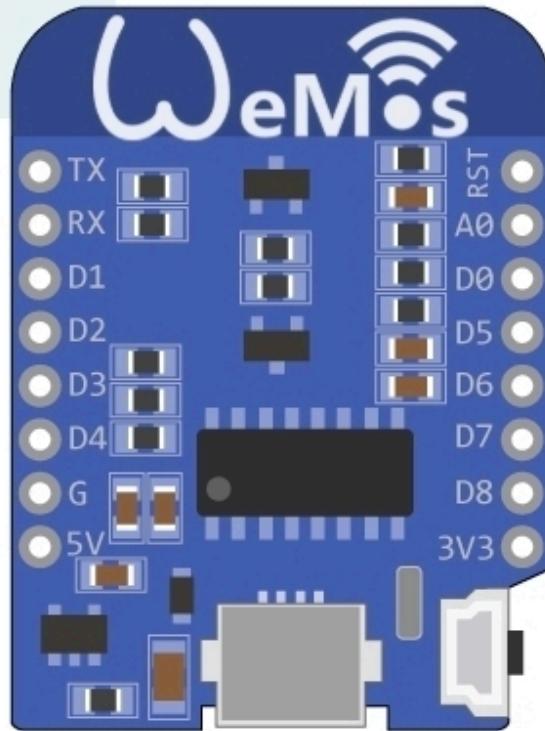
Technical specs

Microcontroller	ESP-8266EX
Operating Voltage	3.3V
Digital I/O Pins	11
Analog Input Pins	1 (Max input: 3.2V)
Clock Speed	80MHz/160MHz
Flash	4M bytes
Length	34.2mm
Width	25.6mm
Weight	10g

WeMos D1 mini 套件



mini



Power
GND
Control

Physical Pin
Port Pin
Pin function
NodeMCU

PWM/I2C/1-Wire
Serial Pin
Analog Pin
Arduino

D1 mini Pin

Pin	Function	ESP-8266 Pin
TX	TXD	TXD
RX	RXD	RXD
A0	Analog input, max 3.3V input	A0
D0	IO	GPIO16
D1	IO, SCL	GPIO5
D2	IO, SDA	GPIO4
D3	IO, 10k Pull-up	GPIO0
D4	IO, 10k Pull-up, BUILTIN_LED	GPIO2
D5	IO, SCK	GPIO14
D6	IO, MISO	GPIO12
D7	IO, MOSI	GPIO13
D8	IO, 10k Pull-down, SS	GPIO15
G	Ground	GND
5V	5V	-
3V3	3.3V	3.3V
RST	Reset	RST



Arduino 開發工具下載

<https://www.arduino.cc/en/Main/Software>

Download the Arduino IDE



ARDUINO 1.8.2

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. It runs on Windows, Mac OS X, and Linux. The environment is written in Java and based on Processing and other open-source software.

This software can be used with any Arduino board. Refer to the [Getting Started](#) page for Installation instructions.

[Windows Installer](#)

[Windows ZIP file for non admin install](#)

[Windows app](#)

[Mac OS X](#) 10.7 Lion or newer

[Linux 32 bits](#)

[Linux 64 bits](#)

[Linux ARM](#)

[Release Notes](#)

[Source Code](#)

[Checksums \(sha512\)](#)



Di mini

- Driver 安裝
 - CH340G USB to UART driver.(for D1 mini or D1)
 - <https://www.wemos.cc/tutorial/get-started-arduino.html>

Di mini

- **ESP 8266 套件板子安裝**

- <https://github.com/esp8266/Arduino>

- 開啟 Arduino IDE > Preferences 視窗
 - 在「額外的板子管理員網址」輸入

http://arduino.esp8266.com/stable/package_esp8266com_index.json

- 在工具 Tools > 板子 (Board menu) > 板子管理員
安裝 esp8266 套件
 - 搜尋：esp8266 關鍵字。



D1 mini

工具	說明
自動格式化	⌘T
封存草稿碼	
修正編碼並重新載入	
序列埠監控視窗	⇧⌘M
Serial Plotter	⇧⌘L
板子: "WeMos D1 R2 & mini"	▶
CPU Frequency: "80 MHz"	▶
Flash Size: "4M (3M SPIFFS)"	▶
Upload Speed: "115200"	▶
序列埠	▶



匯入 *Library*

- 請複製 \libraries 到 Arduino 安裝
路徑 \libraries



D1 mini

- Sample 範例執行：

```
void setup() {  
    pinMode(LED_BUILTIN, OUTPUT); // D1 mini 預設 LED  
}  
  
void loop() {  
    digitalWrite(LED_BUILTIN, LOW);  
    delay(1000);  
    digitalWrite(LED_BUILTIN, HIGH);  
    delay(2000);  
}
```



範例：*D1mini_Relay*

Description: Relay Shield, use relay to control other equipment.

Features:

- NO: 5A(250VAC/30VDC), 10A(125VAC), MAX:1250VA/150W
- NC: 3A(250VAC/30VDC), MAX:750VA/90W



規格：<https://www.wemos.cc/product/relay-shield.html>

範例：*D1mini_RGB*

Description: WS2812B RGB Shield, a intelligent control RGB (16777216 colors) LED light source.

Features:

- LED SIZE: 5050
- colors: 16777216
- Voltage: 5V



規格：<https://www.wemos.cc/product/ws2812b-rgb-shield.html>

D1 mini + RGB

```
#include <Adafruit_NeoPixel.h>

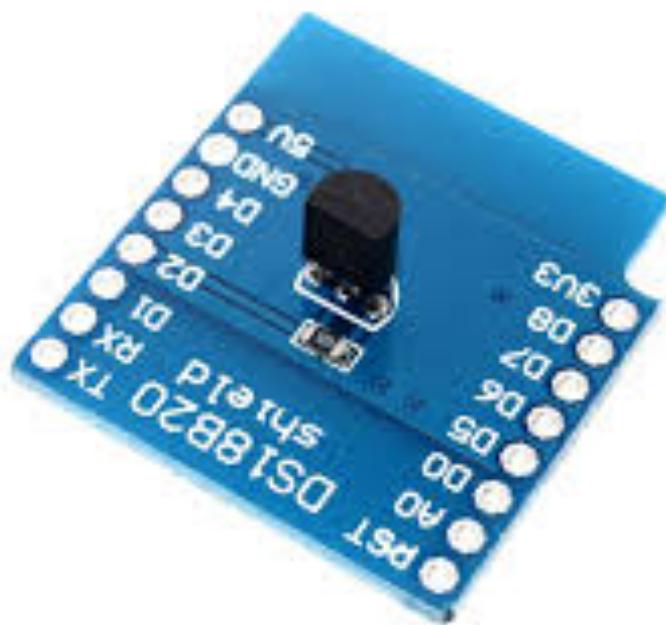
#define PIN D2
Adafruit_NeoPixel pixels = Adafruit_NeoPixel(1, PIN, NEO_GRB + NEO_KHZ800);

void setup() {
  pixels.begin();
}

void loop() {
  int r = random(0, 255);
  int g = 0;
  int b = 0;
  pixels.setPixelColor(0, pixels.Color(r, g, b));
  pixels.show();
  delay(200);
}
```

範例：*D1mini_DS18B20*

溫度範圍 $-55^{\circ}\text{C} \sim +125^{\circ}\text{C}$ ，在 $-10 \sim +85^{\circ}\text{C}$ 時精度為 $\pm 0.5^{\circ}\text{C}$

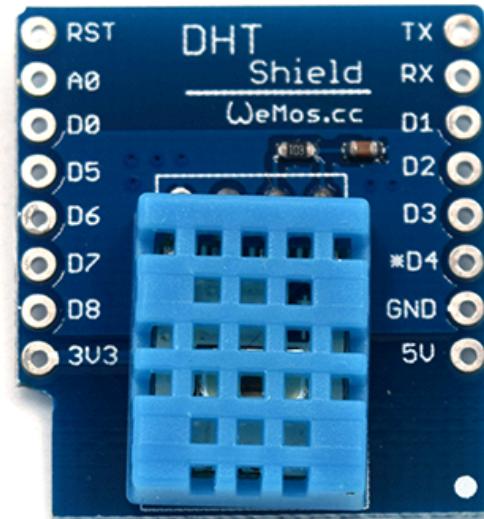


範例：*D1mini_DHT11*

DHT-11 是一個結合濕度計和測溫元件量測週遭空氣環境，並與一個高性能8位元單晶片相連接，將所量測到的溫、濕度資料拆解成為數位訊號，再由 **data pin** 腳將資料送出。

規格如下：

- 1、濕度測量範圍：20~90%RH;
- 2、濕度測量精度： $\pm 5\%$ RH;
- 3、溫度測量範圍：0~50°C
- 4、溫度測量精度： $\pm 2^\circ\text{C}$
- 5、電源供應範圍：3~5V
- 6、頻率不可超過：0.5Hz (每2秒一次)



範例：*D1mini_BMP180*

Pressure sensing range: 300-1100 hPa (9000m to -500m above sea level)

Up to 0.03hPa / 0.25m resolution

-40 to +85°C operational range, +/-2°C temperature accuracy

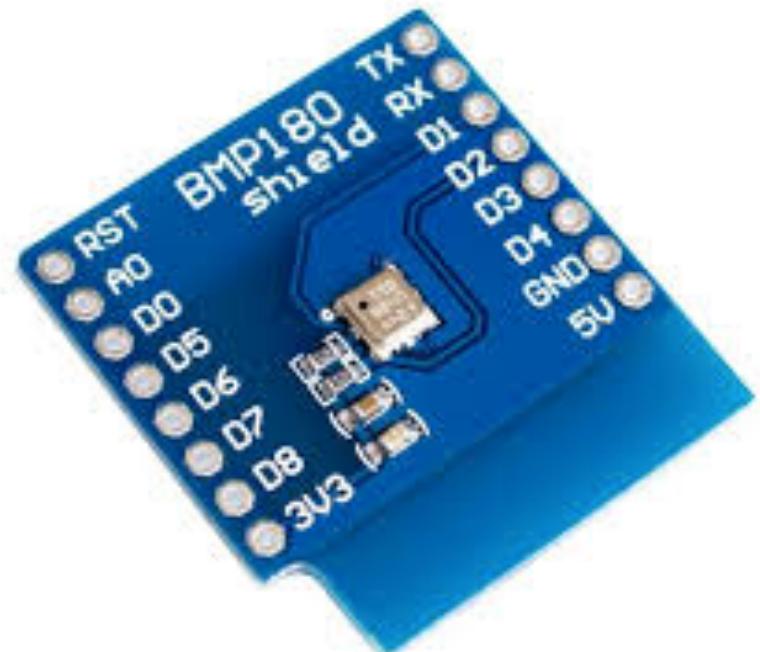
1百帕 (hPa)

= 100 Pa = 1 mbar(毫巴)

101325 pa 標準

每上升10公尺氣壓下降100Pa

海拔 (m) = (101325 - **bmp.readPressure()**) / 10;



範例：*D1mini_OLED*

Description: 64x48 pixels (0.66" Across) OLED Shield with I2C

Features:

- Screen Size: 64x48 pixels (0.66" Across)
- Operating Voltage: 3.3V
- Driver IC: SSD1306
- Interface: IIC(I2C)
- IIC Address: 0x3C or 0x3D



規格：<https://www.wemos.cc/product/oled-shield.html>

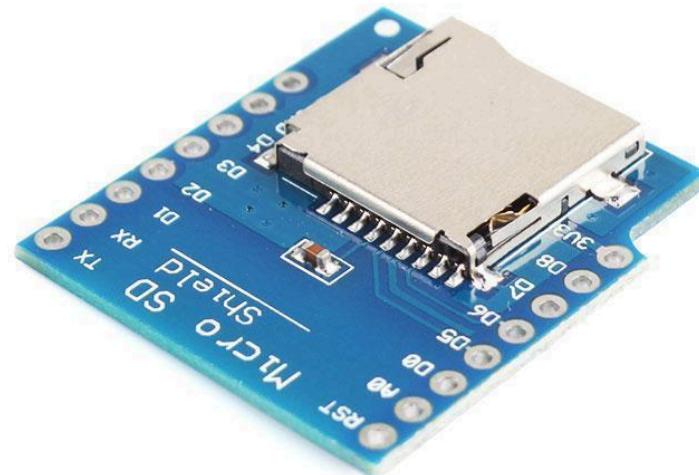
範例：

D1mini_SD_CardInfo

D1mini_MicroSD_RW

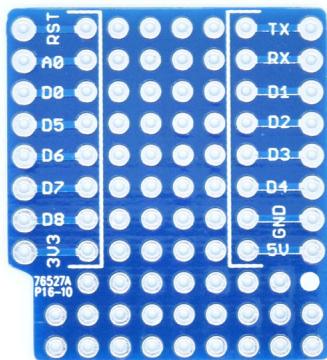
Description: Micro SD Shield is a shield to use Micro SD(TF) card for D1 mini.

Format : FAT16/32

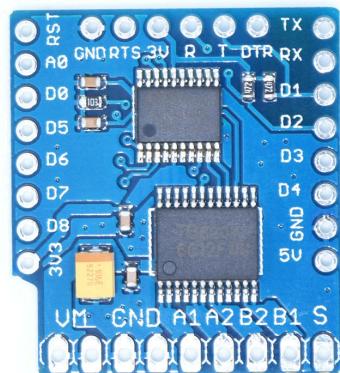


規格：<https://www.wemos.cc/product/micro-sd-card-shield.html>

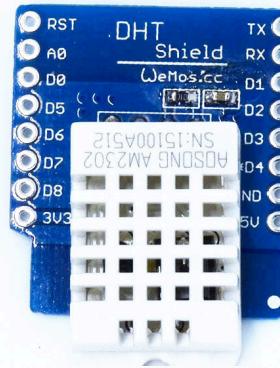
其他有趣的 Shield



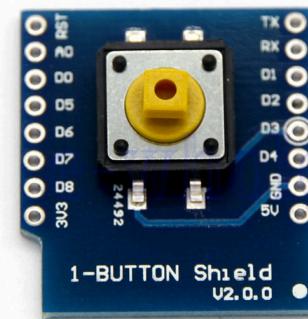
ProtoBoard



Motor Shield



DHT 22(Pro)



1 Button

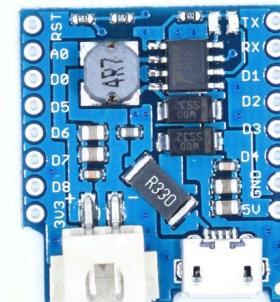
I2C Interface
digital
temperature
and humidity
sensor shield
based SHT30



SHT30

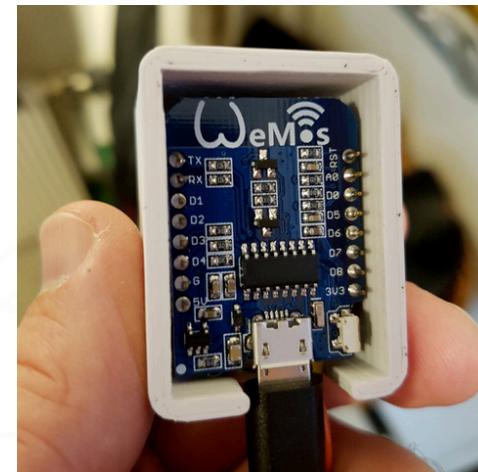


DC Power



Li battery

1 Button Shield



D1 mini + Firebase

- 利用Firebase即時資料庫控制家裡的物聯設備
 - 更新/檢視資料
 - 發送推播訊息
- 申請 Google Firebase
 - Firebase Console
 - <https://console.firebaseio.google.com>



D1 mini + Firebase

- ESP8266 for Firebase 範例集
 - <https://github.com/firebase/firebase-arduino>
 - 下載後匯入 Library
- 開啟 FirebaseArduino 範例
 - 開啟 Arduino IDE
 - 檔案 > 範例 > FirebaseArduino > FirebaseDemo_ESP8266
 - 程式修改請參考下頁...



D1 mini + Firebase

- 修改程式 43 行 (FirebaseDemo_ESP8266)
 - 原：
 - `Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH);`
 - 更正：
 - `Firebase.begin(FIREBASE_HOST);`

Firebase with Android



A screenshot of an Android Studio interface. The top navigation bar shows 'Tools', 'VCS', 'Window', 'Help', and other system status like battery level (86%) and connectivity. The main window has a title bar '[Firebase_Camera]'. On the left, a 'Tools' dropdown menu is open, showing options like 'Tasks & Contexts', 'Generate JavaDoc...', 'New Scratch File...', 'IDE Scripting Console', 'Create Command-line Launcher...', 'Firebase' (which is selected and highlighted in blue), 'App Links Assistant', and 'Android'. Below the menu, there's some Java code in a code editor:

```
setError error) {  
  
    requestCode, resultCode, Intent data) {  
  
();
```

On the right, the 'Assistant' panel is open, featuring the 'Firebase' logo and a brief introduction: 'Firebase gives you the tools and infrastructure from Google to help you develop, grow and earn money from your app.' It lists four services with icons: 'Analytics' (measure user activity), 'Cloud Messaging' (deliver messages), 'Authentication' (sign in users), and 'Realtime Database' (store sync data). Each service has a 'More info' link.

Android App即時『訊息』推播-使用Firebase雲端資料庫



<https://www.e-sharer.com/tw/course/index.aspx?kind1=56>

Android App即時『影像』推播-使用Firebase雲端資料庫



Firebase

<https://firebase.google.com/>



App 即時影像推播

<https://www.e-sharer.com/tw/course/index.aspx?kind1=56>

Android 拍照上傳 Firebase

```
@Override  
protected void onActivityResult(int requestCode, int resultCode, Intent data) {  
    //拍照後顯示圖片  
    if (resultCode == RESULT_OK) {  
  
        //取出拍照後回傳資料  
        Bundle extras = data.getExtras();  
  
        //將資料轉換為圖像格式  
        Bitmap bmp = (Bitmap) extras.get("data");  
  
        //imageView.setImageBitmap(bmp);  
        //將照片傳送至 firebase  
        myRef.setValue(utility.BitmapToString(bmp));  
    }  
}
```

Bitmap與String互轉

```
public static String BitmapToString(Bitmap bitmap){  
    ByteArrayOutputStream baos=new ByteArrayOutputStream();  
    bitmap.compress(Bitmap.CompressFormat.PNG,100, baos);  
    byte [] b=baos.toByteArray();  
    return Base64.encodeToString(b, Base64.DEFAULT);  
}  
  
public static Bitmap StringToBitmap(String encodedString){  
    try {  
        byte [] encodeByte=Base64.decode(encodedString, Base64.DEFAULT);  
        Bitmap bitmap= BitmapFactory.decodeByteArray(encodeByte, 0, encodeByte.length);  
        return bitmap;  
    } catch(Exception e) {  
        e.getMessage();  
        return null;  
    }  
}
```

ESP8266 範例應用

Examples for WeMos D1 R2 & mini

- ▶ ArduinoOTA
- ▶ DNSServer
- ▶ EEPROM
- ▶ ESP8266
- ▶ ESP8266AVRISP
- ▶ ESP8266HTTPClient
- ▶ ESP8266httpUpdate
- ▶ ESP8266HTTPUpdateServer
- ▶ ESP8266mDNS
- ▶ ESP8266SSDP
- ▶ ESP8266WebServer

ESP8266WiFi

- ▶ ESP8266WiFiMesh
- ▶ Ethernet(esp8266)
- ▶ Hash
- ▶ SD(esp8266)
- ▶ Servo(esp8266)
- ▶ SoftwareSerial
- ▶ TFT_Touch_Shield_V2
- ▶ Ticker

第三方程式庫的範例

- ▶ HTTPSRequest
- ▶ NTPClient
- WiFiAccessPoint**
- ▶ WiFiClient
- ▶ WiFiClientBasic
- ▶ WiFiClientEvents
- ▶ WiFiMulti
- ▶ WiFiScan
- ▶ WiFiTelnetToSerial
- ▶ WiFiWebServer



Android II、III 課程驗收範例~



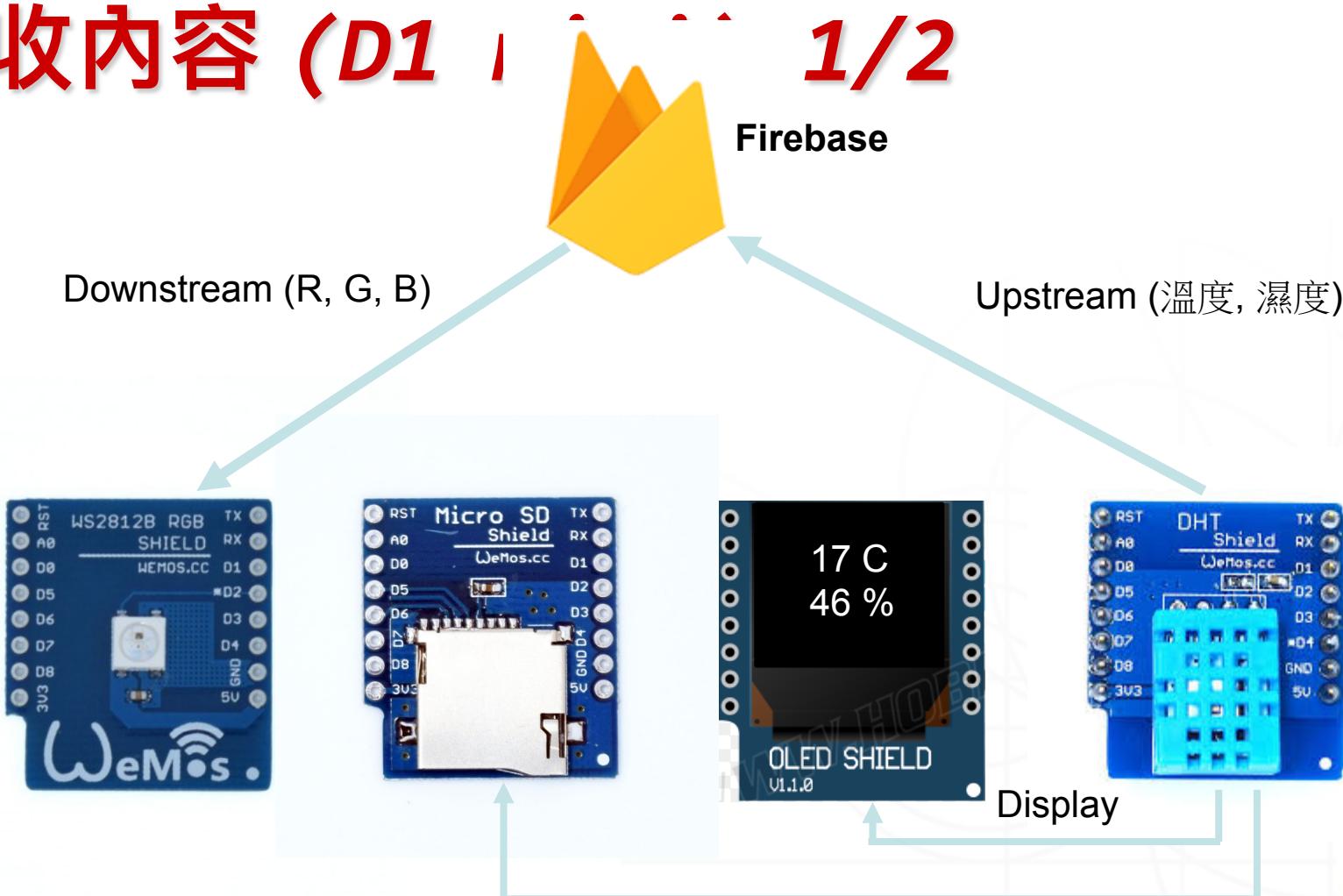
講師驗收

- 已經Android授課之講師不驗第一階
- 第二階與第三階同時驗收
- 將套件組組裝完畢並且可以用手機操控飛行與控制智能家電
- 手機APP開發需將製作出來的APP內有裝置後端資料庫與雲端，程式碼需提供給驗收之講師與PM
- 統一驗收時間：



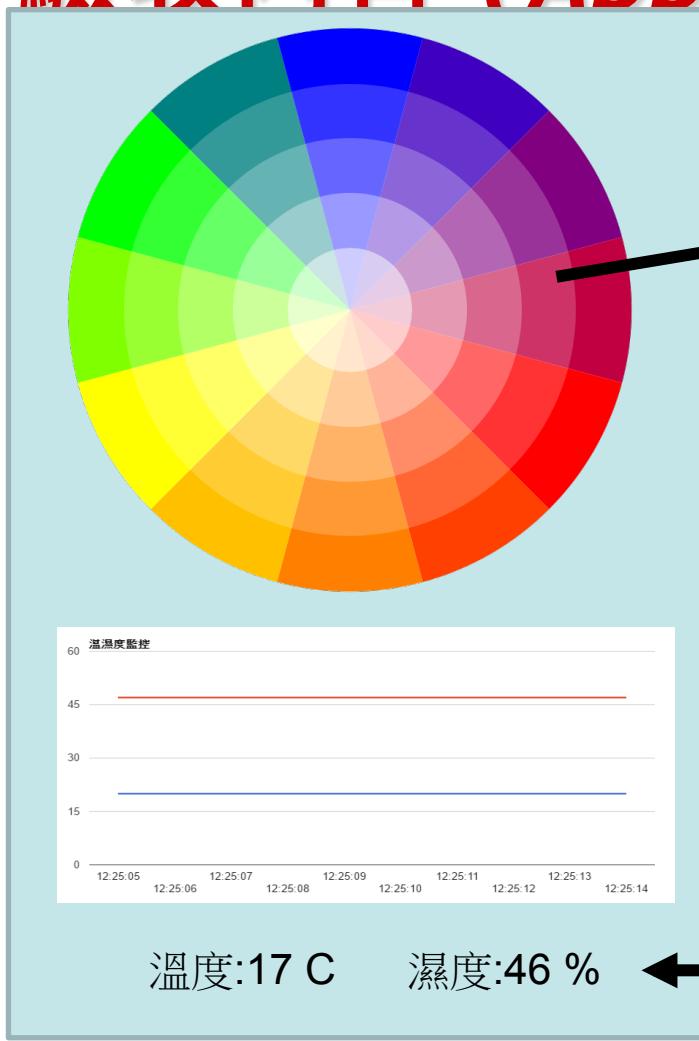
驗收內容 (D1)

1/2



Storage : 儲存 [溫度,濕度,時間] 歷史紀錄

驗收內容 (App) 2/2



Upstream (R, G, B)

使用者可以自由吸色



Firebase

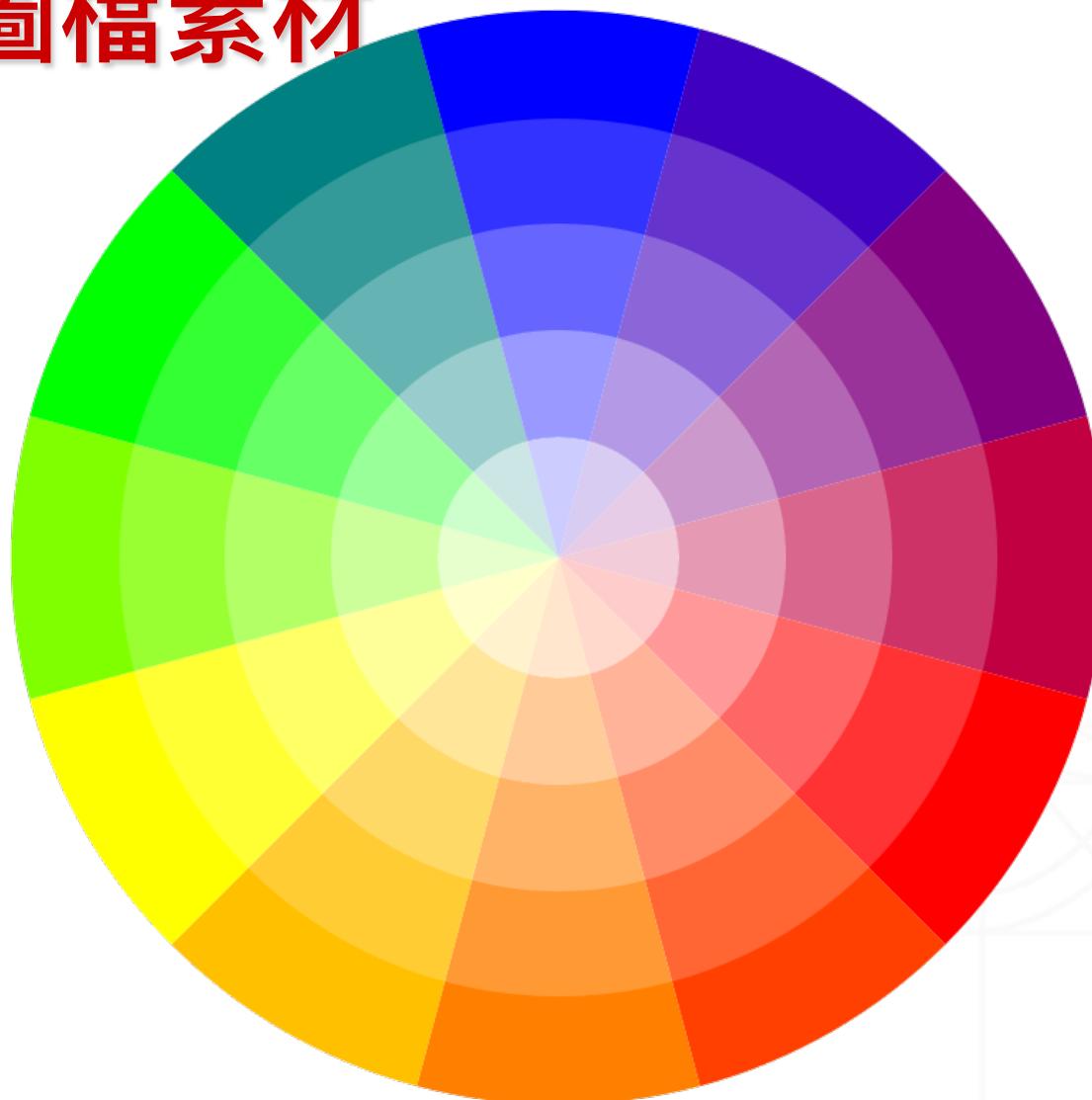
Downstream (溫度, 濕度)

繪製即時溫溼度變化圖
使用 Google Line Chart

顯示目前最新溫溼度資料



RGB 圖檔素材





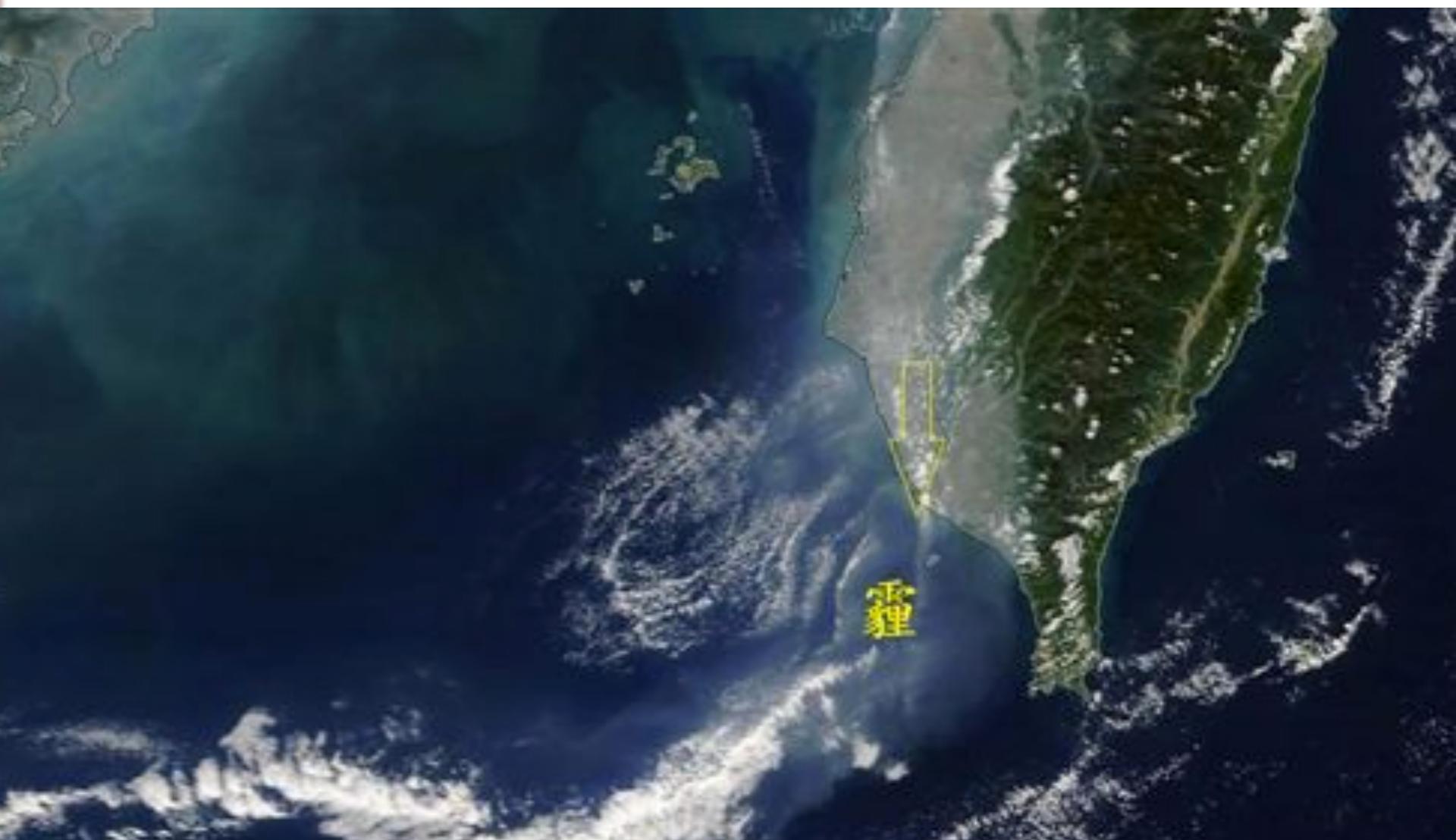
課後補充...



PM 2.5 懸浮微粒



霾害空氣品質有多糟？



你所不知道的 PM2.5 ?!

什麼是PM2.5

空氣中大多懸浮微粒體積約 $10\mu\text{m}$ ，而PM2.5則小於 $2.5\mu\text{m}$ 。細小的體積可能穿透肺泡進入血管內，也許會進一步對人體造成影響。



PM2.5細懸浮微粒直徑 大約為頭髮直徑的1/28



空氣中存在許多污染物，其中漂浮在空氣中類似灰塵的粒狀物稱為懸浮微粒(particulate matter, PM)，PM粒徑大小有別，小於或等於2.5微米(μm)的粒子，就稱為PM_{2.5}，通稱細懸浮微粒，單位以微克/立方公尺($\mu\text{g}/\text{m}^3$)表示之，它的直徑還不到人的頭髮絲粗細的1/28，非常微細可穿透肺部氣泡，並直接進入血管中隨著血液循環全身，故對人體及生態所造成之影響是不容忽視的。

粒徑 (μm)	粒徑大小說明
<100	稱 總懸浮微粒(TSP)，約為海灘沙粒，可懸浮於空氣中。
<10	稱 懸浮微粒(PM ₁₀)，約為沙子直徑的1/10，容易通過鼻腔之鼻毛與彎道到達喉嚨
2.5~10	稱 粗懸浮微粒(PM _{2.5-10})，約頭髮直徑的1/8~1/20大小，可以被吸入並附著於人體的呼吸系統
<2.5	稱 細懸浮微粒(PM _{2.5})，約頭髮直徑的1/28，可穿透肺部氣泡，直接進入血管中隨著血液循環全身。



PM 2.5 怎麼來的 ?!

- 懸浮粒子成分複雜

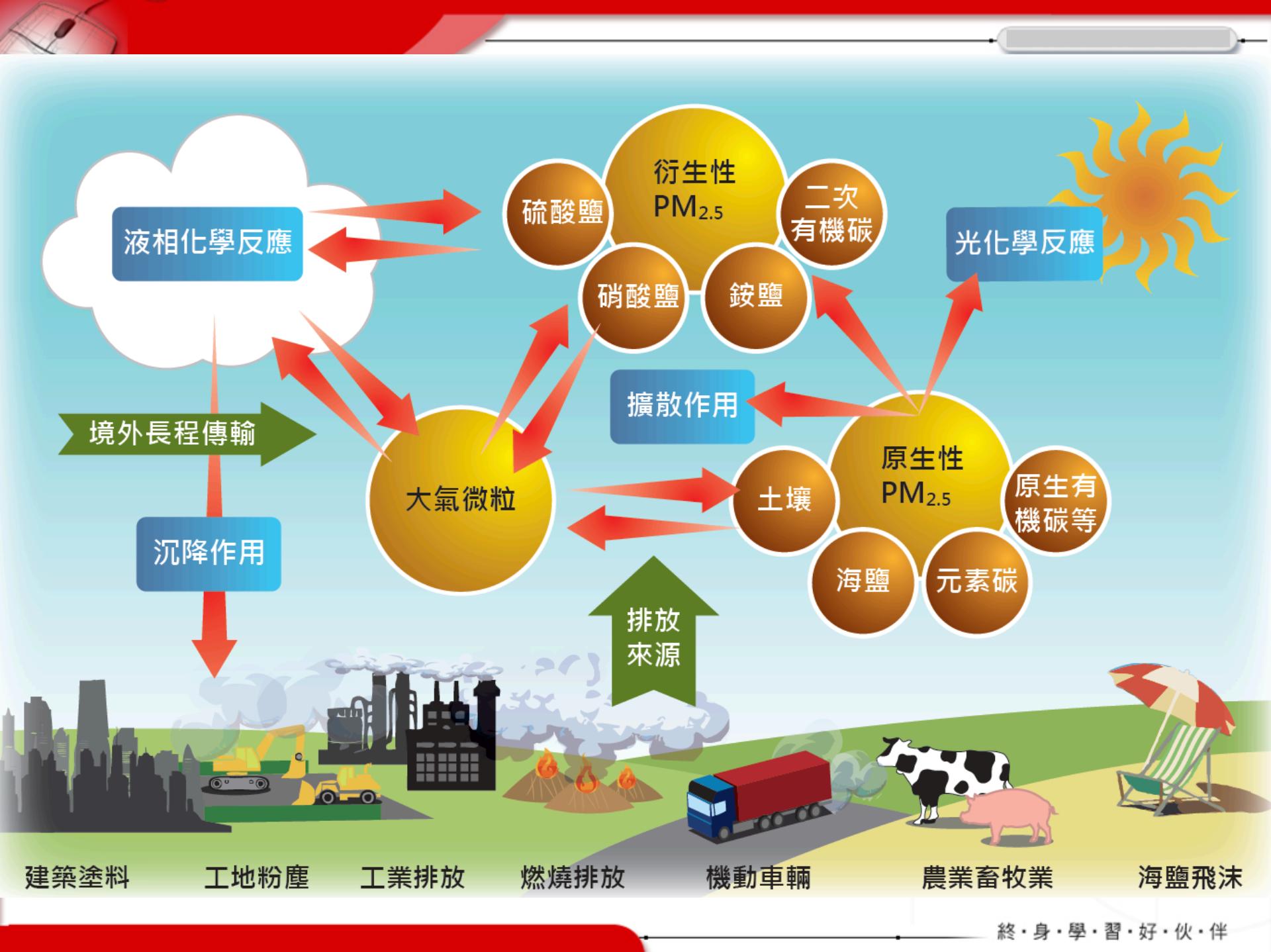
- 室外

- 主要是地面上的塵土、海鹽、無廢氣處理的柴油車，甚至是燃燒垃圾的氣體都是污染來源。
 - 中國大陸季風的影響；火力發電。

- 室內

- 則是以塵蟎與二手菸為主要來源。
 - 金紙燃燒、焚香及燃燒蚊香也是汙染源之一。





圖解 PM2.5 致病全過程

資料來源: sina weibo

北京嚴重空氣污染本週持續，PM2.5 濃度值處極高水平，大部分地區達達 700 微克 / 立方米以上。醫院呼吸器官病例激增。

**PM
2.5**

大氣中直徑小於或等於
2.5 微米的顆粒物

**PM
2.5 ~ 10**

大氣中直徑 2.5 微米至
10 微米的顆粒物

**PM
10+**

大氣中直徑等於或大於
10 微米的顆粒物

直徑小於 **2.5** 微米的顆粒物
可以直接進入支氣管以及肺泡，
從而被人體吸收。

被人體吸收的微塵可以損害血
紅蛋白的輸送氧能力，使人體
喪失血液。並且引發全身各系
統疾病

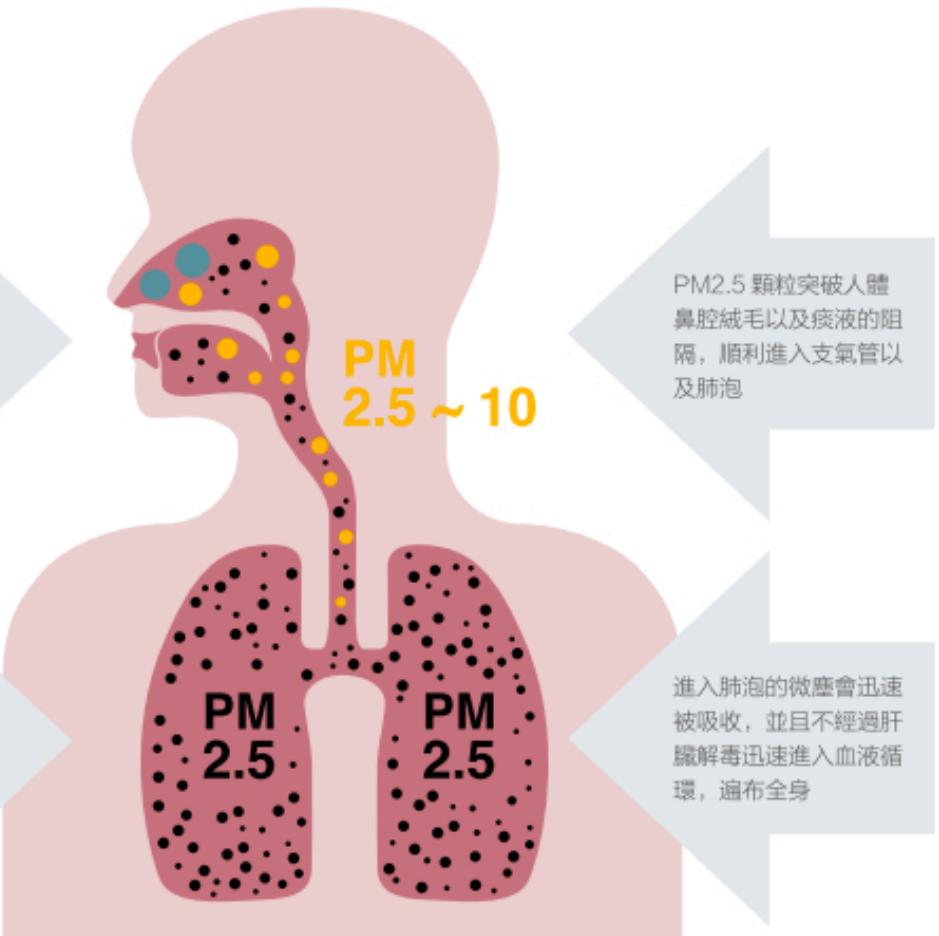
**PM
2.5 ~ 10**

PM2.5 顆粒突破人體
鼻腔絨毛以及痰液的阻
隔，順利進入支氣管以
及肺泡

**PM
2.5**

**PM
2.5**

進入肺泡的微塵會迅速
被吸收，並且不經過肝
臟解毒迅速進入血液循
環，遍布全身



不同懸浮微粒粒徑分布對呼吸系統影響



2.5~10 μm

分布特性：
沉積於上部鼻腔與深呼吸道。

影響：
造成纖維麻痺、支氣管黏膜過度分泌、使黏液腺增生，引起可逆性支氣管痙攣，抑制深呼吸、並蔓延至小支氣管道。



>10 μm

分布特性：
沉積於鼻咽。

影響：
容易造成過敏性鼻炎，引發咳嗽、氣喘等症狀。



<0.1 μm

分布特性：
沉積於肺泡組織內。

影響：
促使肺部之巨噬細胞明顯增加，形成肺氣腫並破壞肺泡。



<2.5 μm

分布特性：
10%以下沉積於支氣管，約20~30%於肺泡。

影響：
形成慢性支氣管炎、細支氣管擴張、肺水腫或支氣管纖維化等症狀。



空氣品質標準		WHO ¹				歐盟	美國		加拿大	澳洲	日本	南韓	香港	中國大陸		泰國	我國
		IT-1	IT-2	IT-3	AQG		聯邦	加州						一級	二級		
PM _{2.5} µg/m ³	年平均值	35	25	15	10	25 ²	12	12	10	8	15	25	35	15	35	25	15
	24小時 平均值	75	50	37.5	25	-	35	-	28	25	35	50	75	35	75	50	35

資料來源: Air quality guidelines-global update 2005, World Health Organization

Air Quality Standards, Environment, European Commission

National Ambient Air Quality Standards (NAAQS), U.S. Environmental Protection Agency, United States

California Ambient Air Quality Standards (CAAQS), California Environmental Protection Agency

National Ambient Air Quality Objectives (NAAQOs), Health Canada, Canada

Air quality standards, Department of the Environment, Water, Heritage and the Arts, Australian Government

大氣汚染に係る環境基準，環境省，日本

Changes in and Strengthening of Air Quality Standards, Ministry of Environment, Republic of Korea

空气质量素指标，环境保护署，香港特别行政区政府

环境空气质量标准，环境保护部，中华人民共和国

Ambient Air Standards, Pollution Control Department, Ministry of Natural Resources and Environment, Thailand

空氣品質標準，行政院環境保護署

備註: 1.WHO: 第一段過渡期目標(IT-1)

第二段過渡期目標(IT-2)

第三段過渡期目標(IT-3)

空氣品質準則(AQG)

2.2010年目標濃度，2015年法規濃度

3.中國大陸:一級標準適用於自然保護區、風景名勝區及其它需要特殊保護的地區

二級標準適用於居民區、商業交通居民混合區、文化區、工業區和農村地區

GB3095-2012環境空氣質量標準於2012年2月29發布，於2016年1月1日實施

濃度每增加 **10** $\mu\text{g}/\text{m}^3$

總的死亡風險會上升**4%**

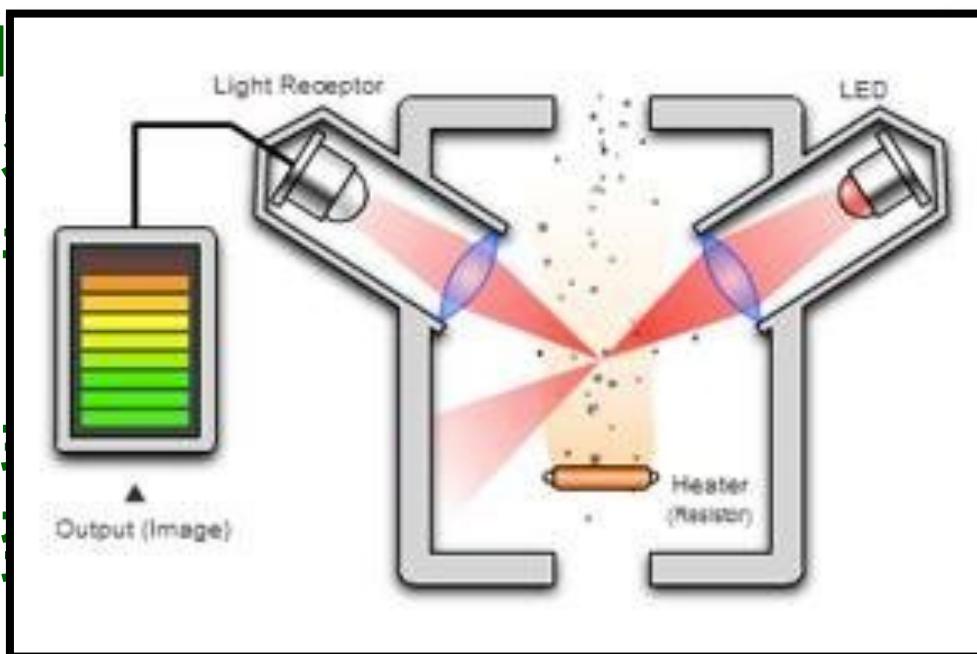
GP2Y1010AU0F(PM 2.5)



GP2Y1010AU0F (PM 2.5)

- GP2Y1010AU0F (PM 2.5)

- 是SH
計用。
其內
晶體
即使
檢測

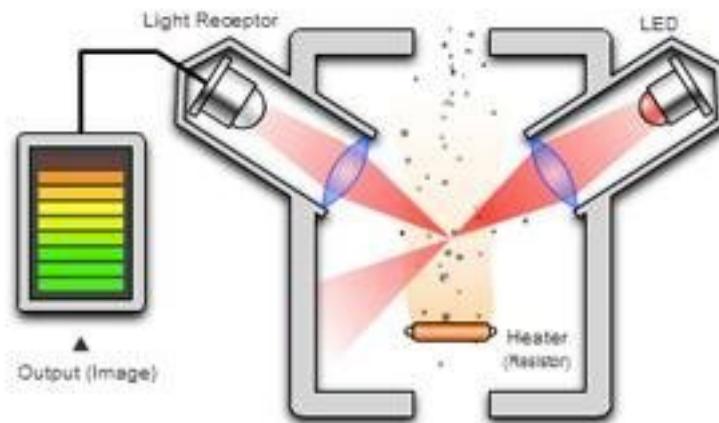


測器，設
豐和光電
反射光，
也能夠被
充中。

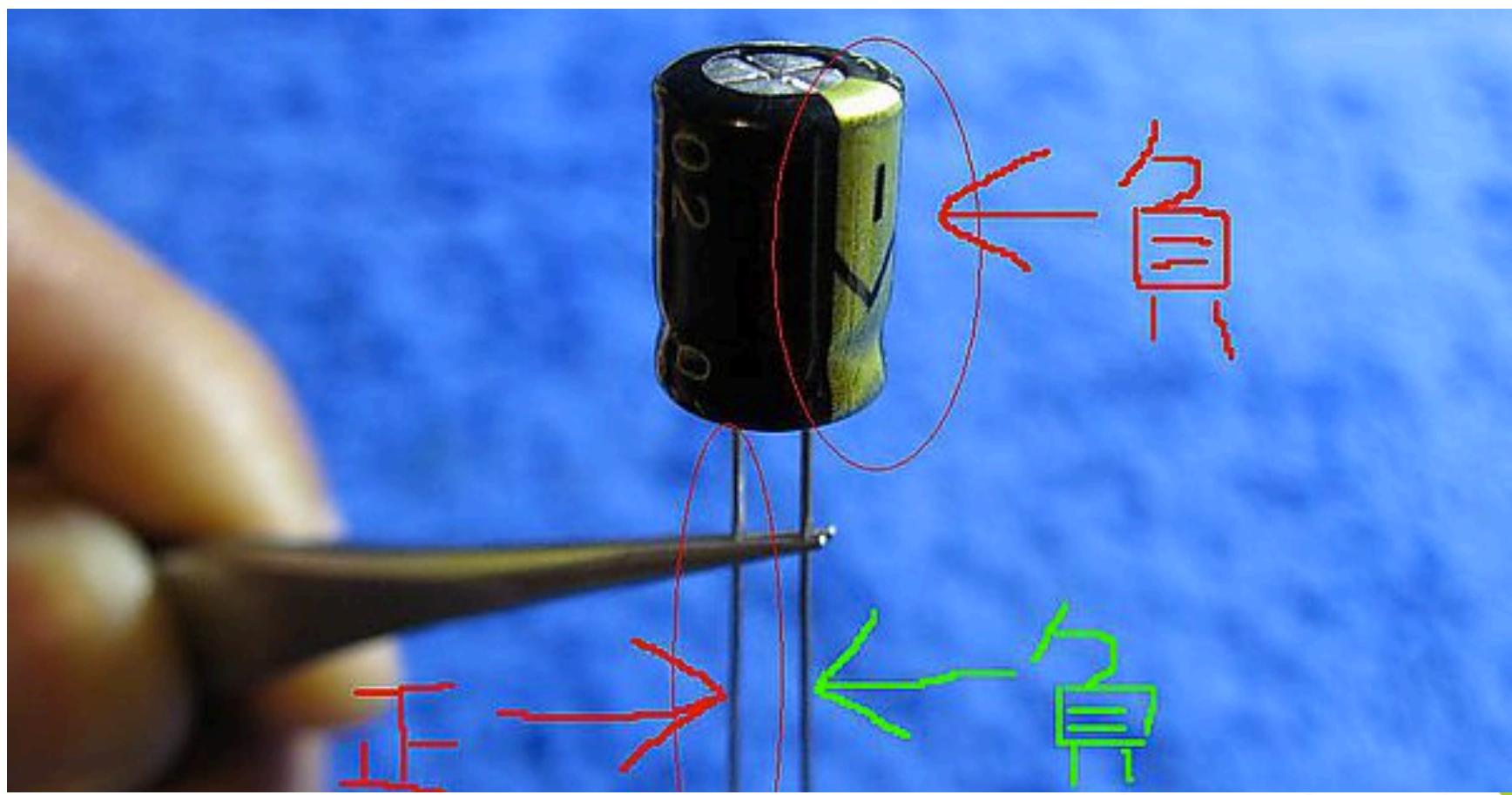


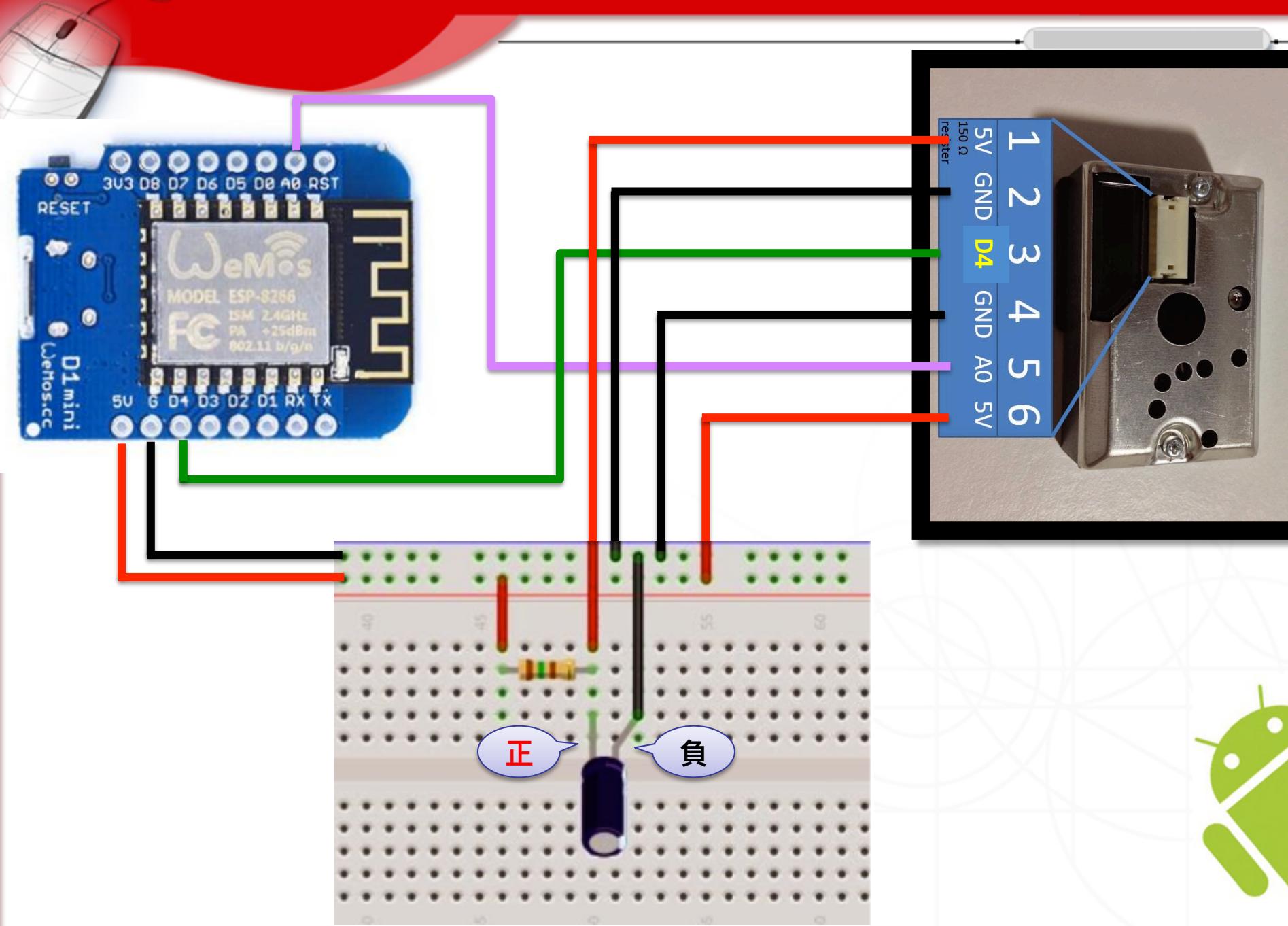
GP2Y1010AU0F (PM 2.5)

- 拆解說明
- 運作原理



電容正負





PM 2.5 (-)

```
int dustPin=0; // D1 mini 接 A0
int dustVal=0; // 儲存 pm2.5 的值 ug/m3微米/立方m

int ledPower=2; // D1 mini 接 D4
int delayTime=280;
int delayTime2=40;
float offTime=9680;
void setup(){
    Serial.begin(9600);
    pinMode(ledPower,OUTPUT);
    pinMode(4, OUTPUT);
}

void loop(){
    digitalWrite(ledPower,LOW); // power on the LED
    delayMicroseconds(delayTime);
    dustVal=analogRead(dustPin); // read the dust value
    delayMicroseconds(delayTime2);
    digitalWrite(ledPower,HIGH); // turn the LED off
    delayMicroseconds(offTime);

    delay(3000);
    Serial.print(dustVal);

    Serial.println(" ug/m3 ");
    delay(1000);
}
```



PM 2.5 (二)

```
int measurePin = 0; // D1 mini 接 A0
int ledPower = 2; // D1 mini 接 D4

int samplingTime = 280;
int deltaTime = 40;
int sleepTime = 9680;

float voMeasured = 0;
float calcVoltage = 0;
float dustDensity = 0;

void setup() {
    Serial.begin(9600);
    pinMode(ledPower, OUTPUT);
}

void loop() {
    digitalWrite(ledPower, LOW); // power on the LED
    delayMicroseconds(samplingTime);

    voMeasured = analogRead(measurePin); // read the dust value

    delayMicroseconds(deltaTime);
    digitalWrite(ledPower, HIGH); // turn the LED off
    delayMicroseconds(sleepTime);
    if (voMeasured > 0) {
        // 0 - 5V mapped to 0 - 1023 integer values
        // recover voltage
        calcVoltage = voMeasured * (5.0 / 1024.0);

        // linear equation taken from http://www.howmuchsnow.com/arduino/airquality/
        dustDensity = 0.17 * calcVoltage - 0.1;
        if (dustDensity > 0) {
            Serial.print("Raw Signal Value (0-1023): ");
            Serial.print(voMeasured);

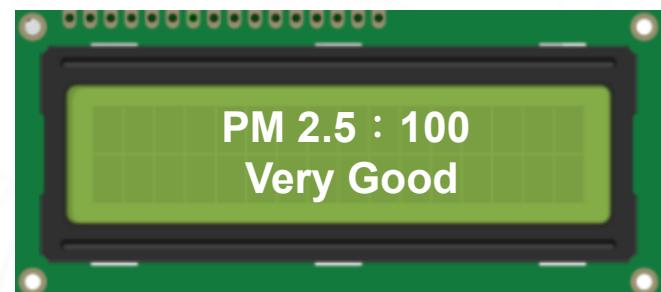
            Serial.print(" - Voltage: ");
            Serial.print(calcVoltage);

            Serial.print(" - Dust Density: ");
            // 這裡將數值呈現改成較常用的單位( ug/m3 )
            Serial.print(dustDensity * 1000);
            Serial.println(" ug/m3 ");
            delay(3000);
        }
    }
}
```

PM 2.5

- 利用 LCD 呈現 PM 2.5

- 3000 以上 : Very Bad
- 1050-3000 : Bad
- 300-1050 : Normal
- 150-300 : Good
- 75-150 : Very Good
- 0-75 : Perfect



PM 2.5 雲端 App

- 將 PM 2.5 的數值送入 Firebase
 - 並寫一個 App 取得 PM 2.5 數值資料
 - 亦可透過App控制 Relay



