



國產IC開發套件 HUB 8735 示範案例

HUB 8735 Stranger Alert with PU02



大綱

- 開發板HUB 8735 介紹
- 開發環境Arduino建置
- 周邊介面與範例說明：
 - ◆ PU02毫米波雷達

HUB 8735 介紹

HUB 8735



晶片原廠
瑞昱半導體

晶片採用
Ameba RTL8735

【開發板特點】

- 兼容Arduino開發特性
- 具備多功能影像處理的高度集成模組
- 內置NPU AI 運算引擎加速處理AI模型
- 802.11 a/b/g/n 雙頻Wi-Fi與BLE低耗電藍牙傳輸
- 可廣泛應用於各種結合影像識別或AI運算之物聯網場域

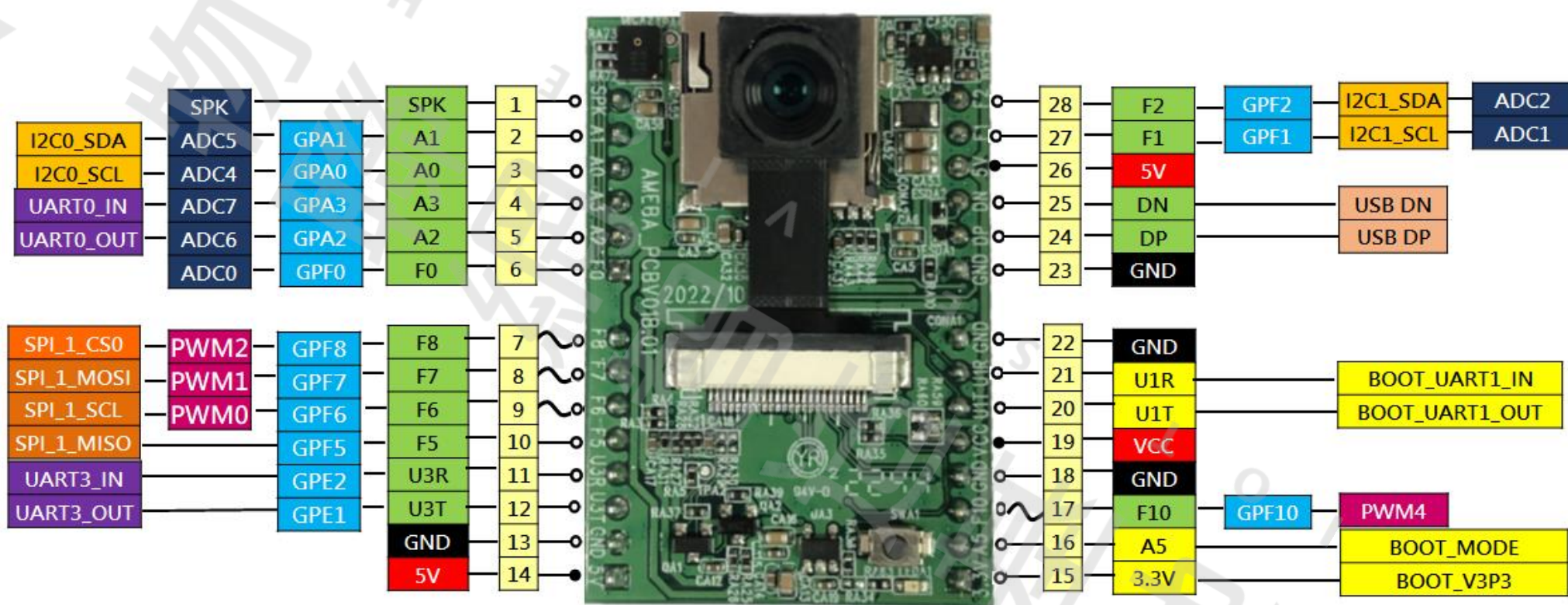


經濟部工業局廣告

HUB 8735 介紹

HUB 8735 腳位圖

POWER
GROUND
PIN INDEX
PIN NAME
GPIO
ANALOG
PWM
UART
SPI
I2C
USB
CONTROL
5V tolerant
Not 5V tolerant
PIN Mux
Max ±16mA per pin, ±4mA recommend



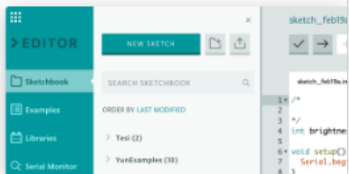


Install Arduino IDE

Arduino Web Editor

Start coding online and save your sketches in the cloud. The most up-to-date version of the IDE includes all libraries and also supports new Arduino boards.

[CODE ONLINE](#) [GETTING STARTED](#)



Downloads



Arduino IDE 2.1.0

The new major release of the Arduino IDE is faster and even more powerful! In addition to a more modern editor and a more responsive interface it features autocompletion, code navigation, and even a live debugger.

For more details, please refer to the [Arduino IDE 2.0 documentation](#).

Nightly builds with the latest bugfixes are available through the section below.

SOURCE CODE

The Arduino IDE 2.0 is open source and its source code is hosted on [GitHub](#).

DOWNLOAD OPTIONS

Windows Win 10 and newer, 64 bits
Windows MSI installer
Windows ZIP file

Linux AppImage 64 bits (X86-64)
Linux ZIP file 64 bits (X86-64)

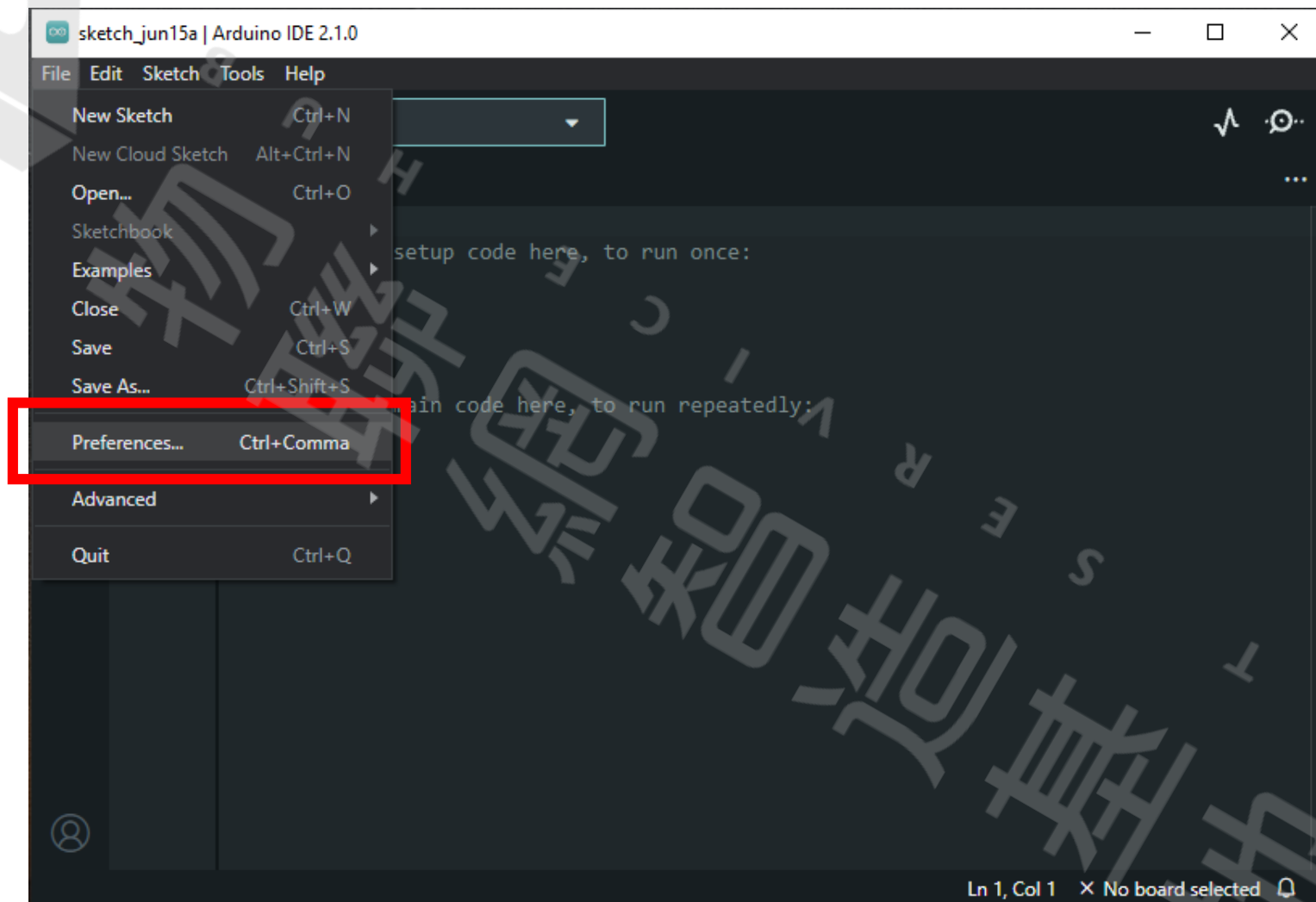
macOS Intel, 10.14: "Mojave" or newer, 64 bits
macOS Apple Silicon, 11: "Big Sur" or newer, 64 bits

[Release Notes](#)

建議版本 ≥ 1.8.19

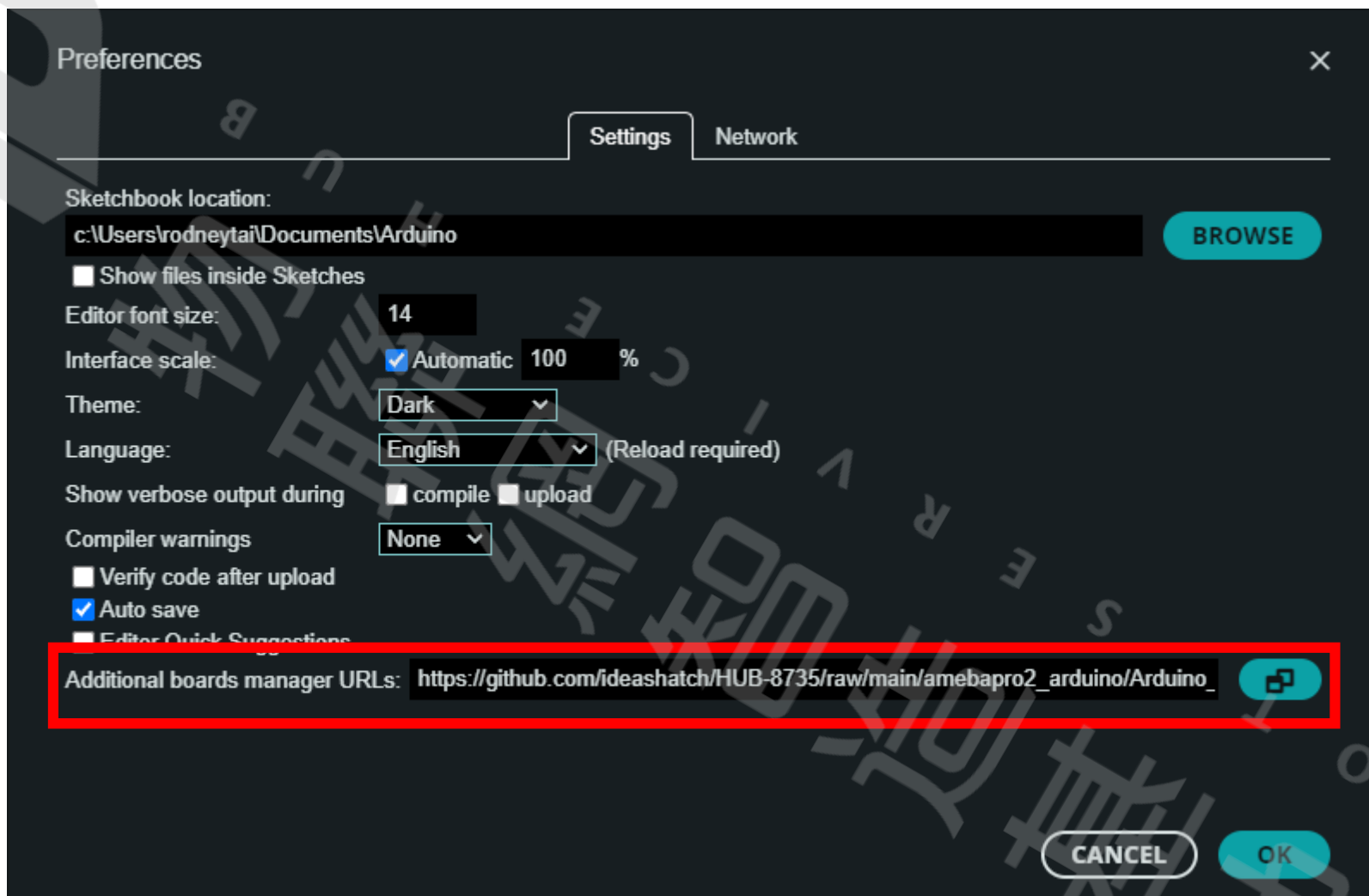
Download: <https://www.arduino.cc/en/software>

Arduino Setup





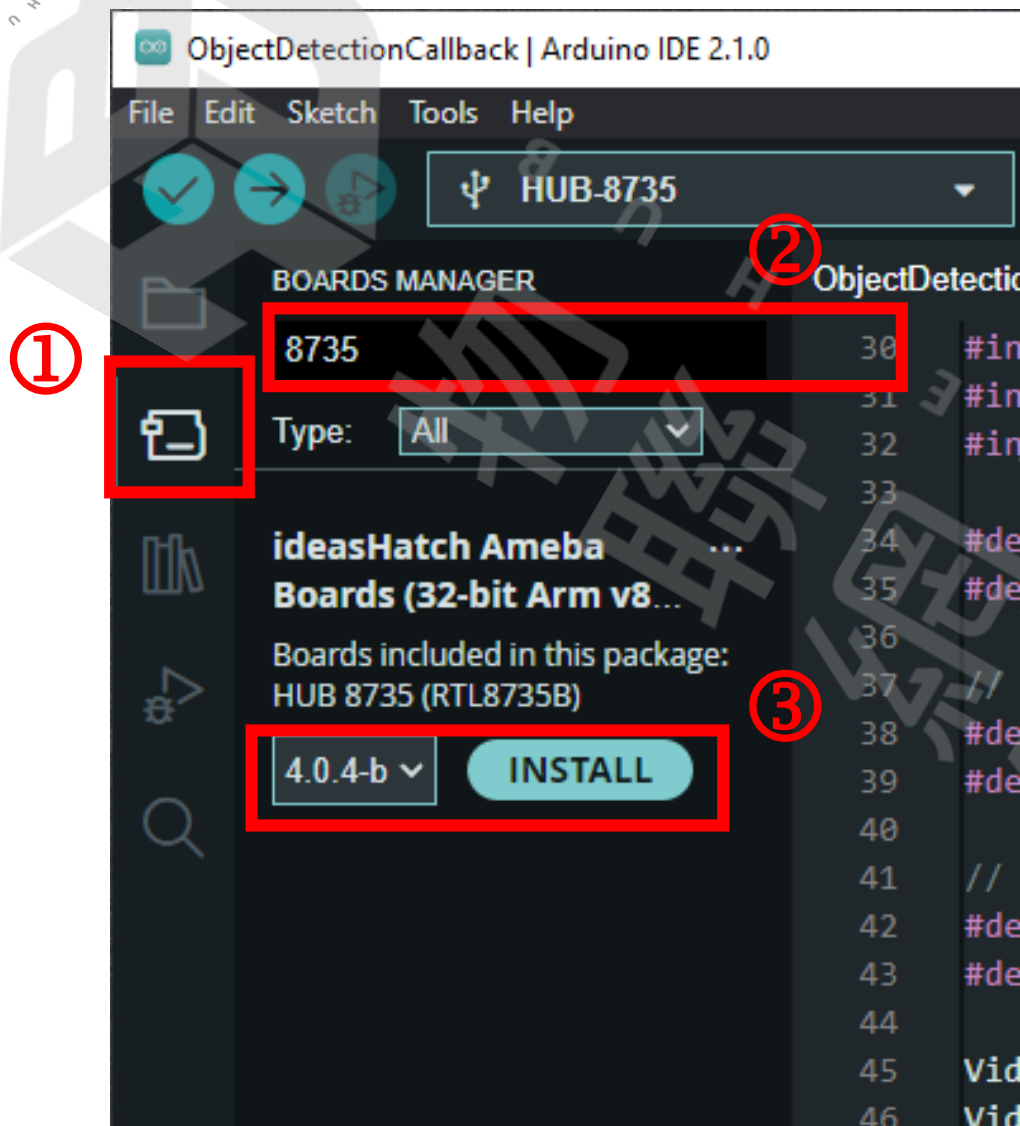
Arduino Setup



在 Additional boards manager URLs 加上以下連結並按OK：

https://github.com/ideashatch/HUB-8735/raw/main/amebapro2_arduino/Arduino_package/ideasHatch.json

Arduino Setup

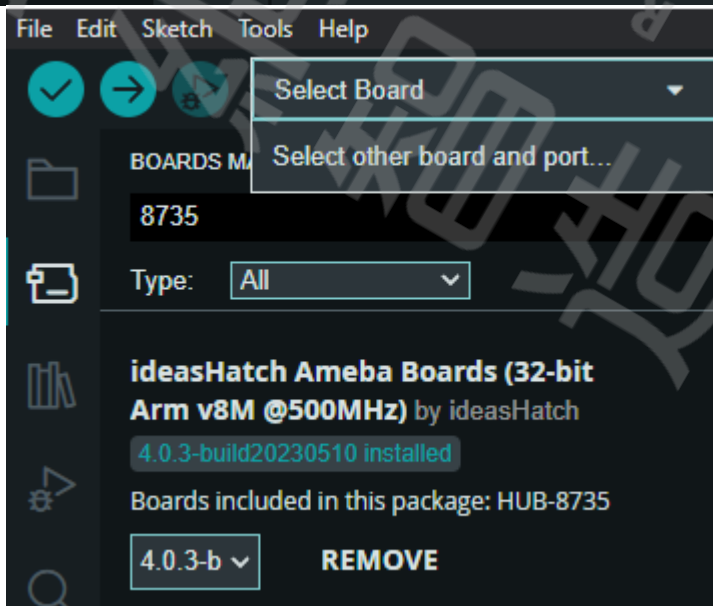


1. 點選Board Manager
2. 輸入8735 / ideasHatch
3. 選擇版本install

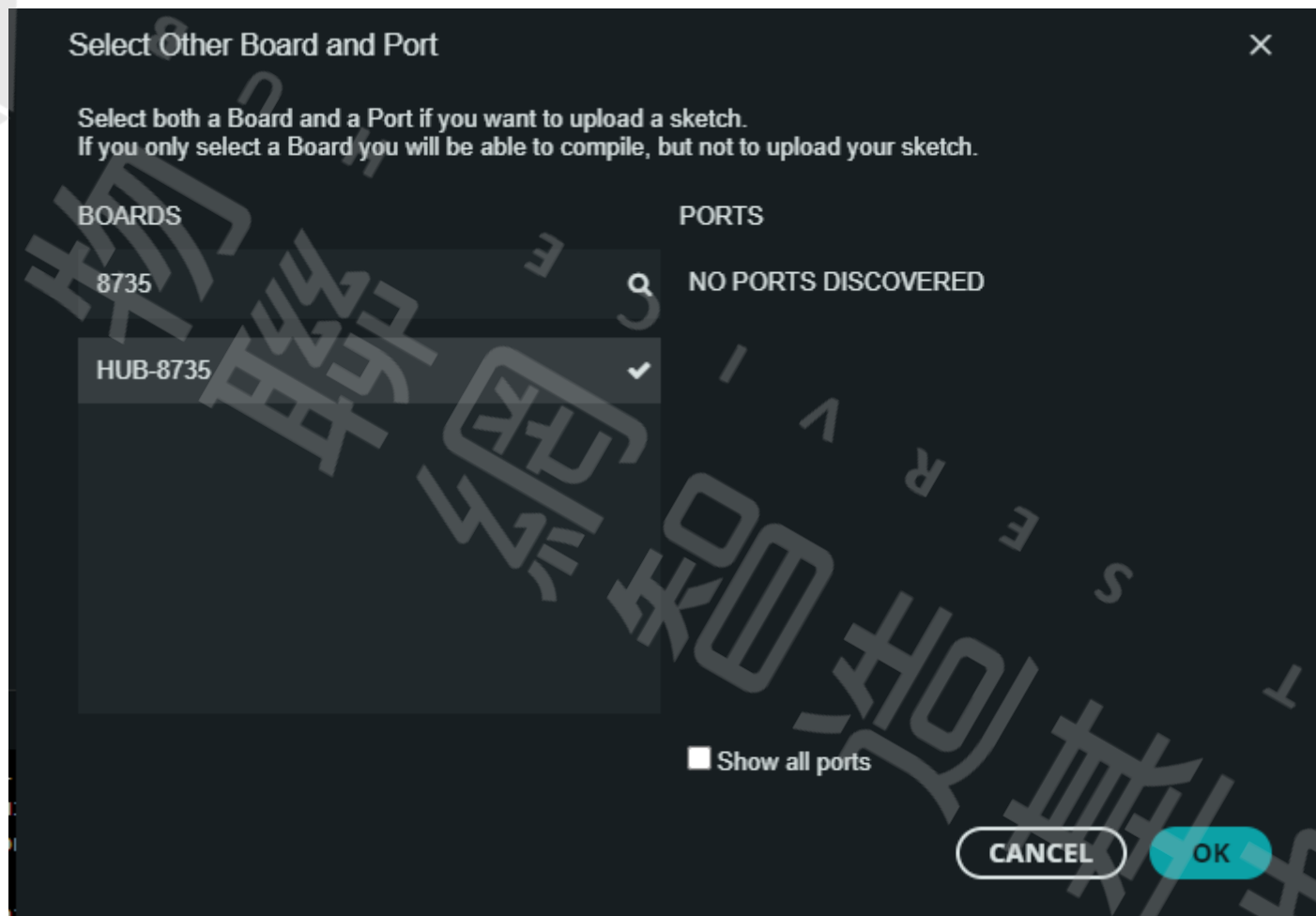
Arduino Setup



SDK安裝完畢後，
要選擇Board。
兩種方式都可作用。

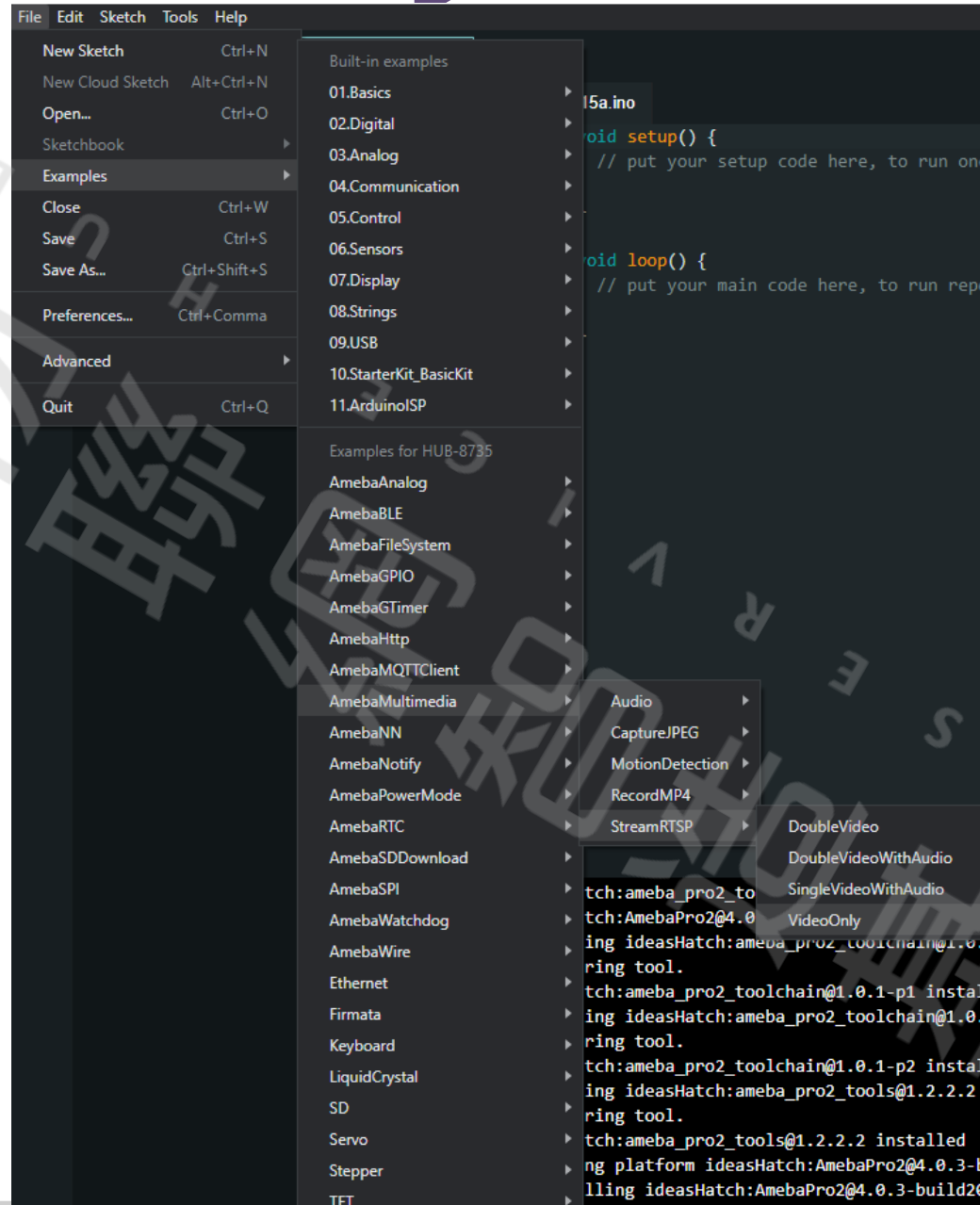


Arduino Setup





Testing 開啟範例





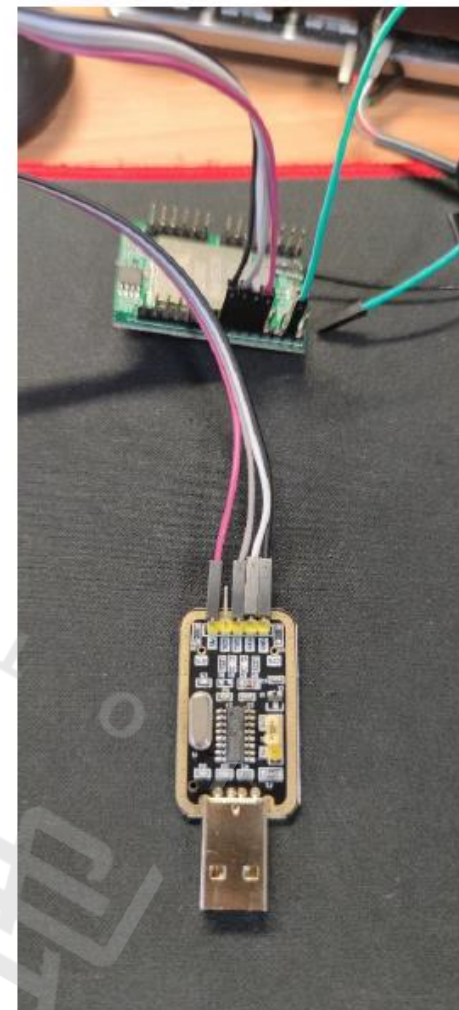
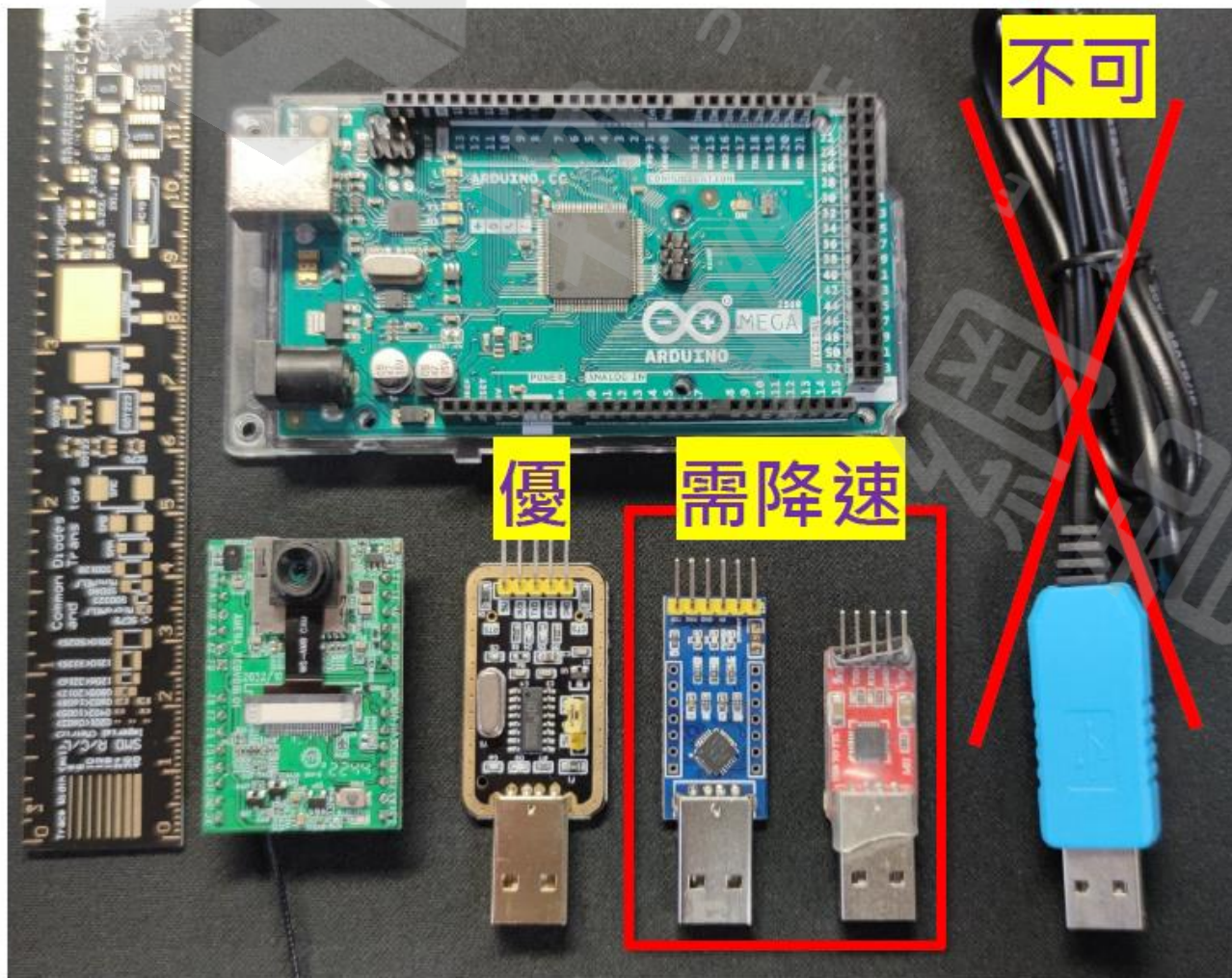
Stream Video

更改 Wi-Fi SSID & password

```
VideoOnly.ino
1  /*
2
3  Example guide:
4  https://www.amebaiot.com/en/amebapro2-amb82-mini-arduino-video-rtsp/
5
6  For recommended setting to achieve better video quality, please refer to our Ameba
7  */
8
9  #include "WiFi.h"
10 #include "StreamIO.h"
11 #include "VideoStream.h"
12 #include "RTSP.h"
13
14 #define CHANNEL 0
15
16 // Default preset configurations for each video channel:
17 // Channel 0 : 1920 x 1080 30FPS H264
18 // Channel 1 : 1280 x 720 30FPS H264
19 // Channel 2 : 1280 x 720 30FPS MJPEG
20
21 VideoSetting config(CHANNEL);
22 RTSP rtsp;
23 StreamIO videoStreamer(1, 1); // 1 Input Video -> 1 Output RTSP
24
25 char ssid[] = "yourNetwork"; // your network SSID (name)
26 char pass[] = "Password"; // your network password
27
28
29 void setup() {
30     Serial.begin(115200);
31
32     // attempt to connect to Wifi network:
33     while (status != WL_CONNECTED) {
34         Serial.print("Attempting to connect to WPA SSID: ");
35         Serial.println(ssid);
36         status = WiFi.begin(ssid, pass);
37     }
```

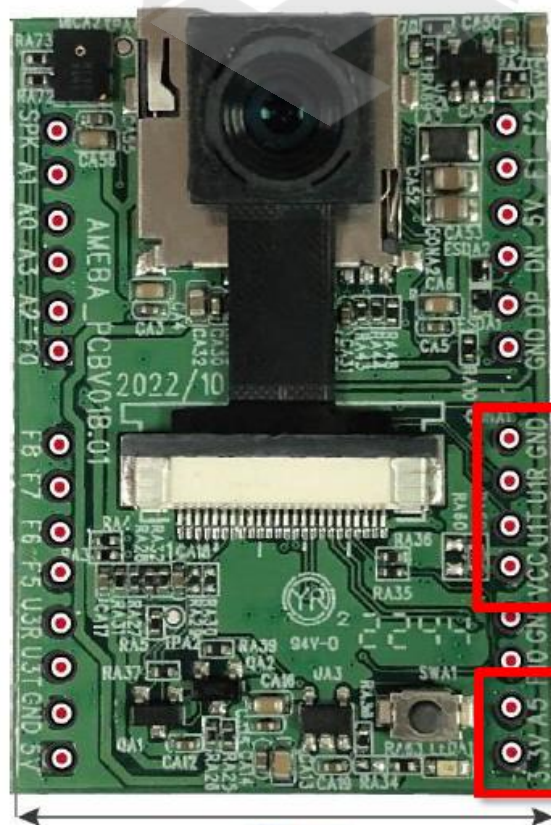
Connect to PC

CP210X不支援2M speed ⚠



連接方式

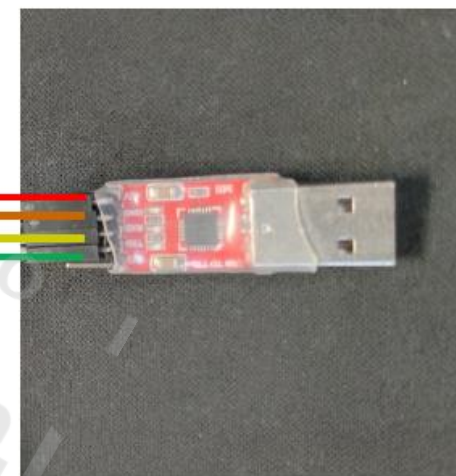
HUB 8735



F2	GPF2	I2C1_SDA	ADC2	I/O
F1	GPF1	I2C1_SCL	ADC1	I/O
5V				POW
DN				USB
DP				USB
GND				POW

GND			POW
U1R			I/O
U1T			I/O
VCC			P_OUT
GND			POW
F10	GPF10	PWM4	I/O
A5			BOOT
3.3V			BOOT

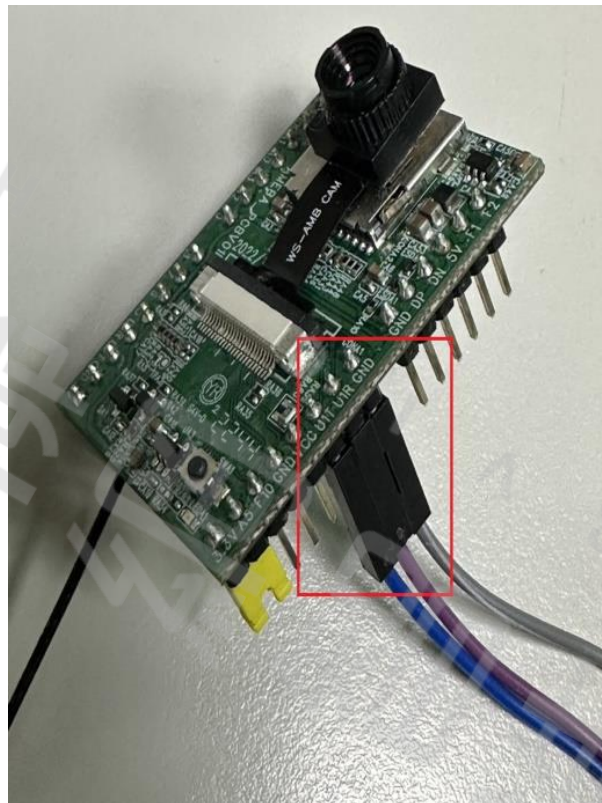
BOOT_MODE



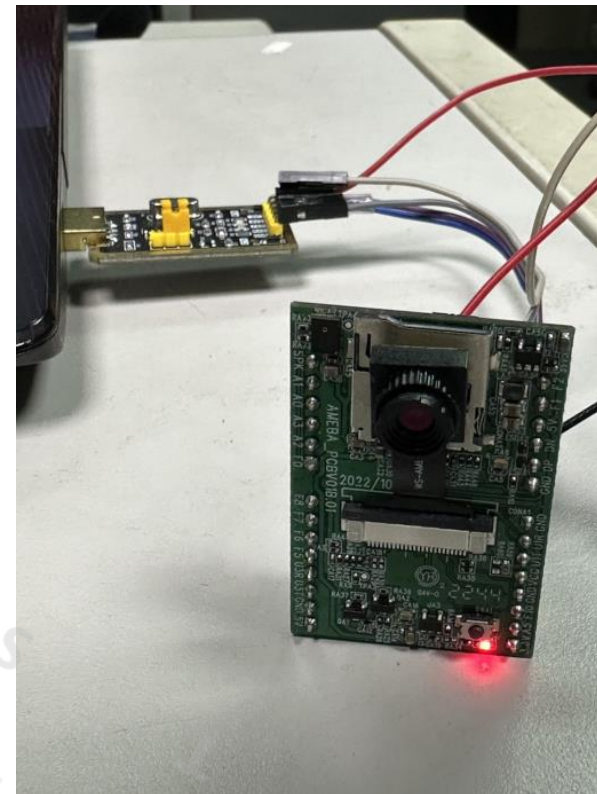
連結方式



短路 3V3 及 A5
(BOOT PIN)

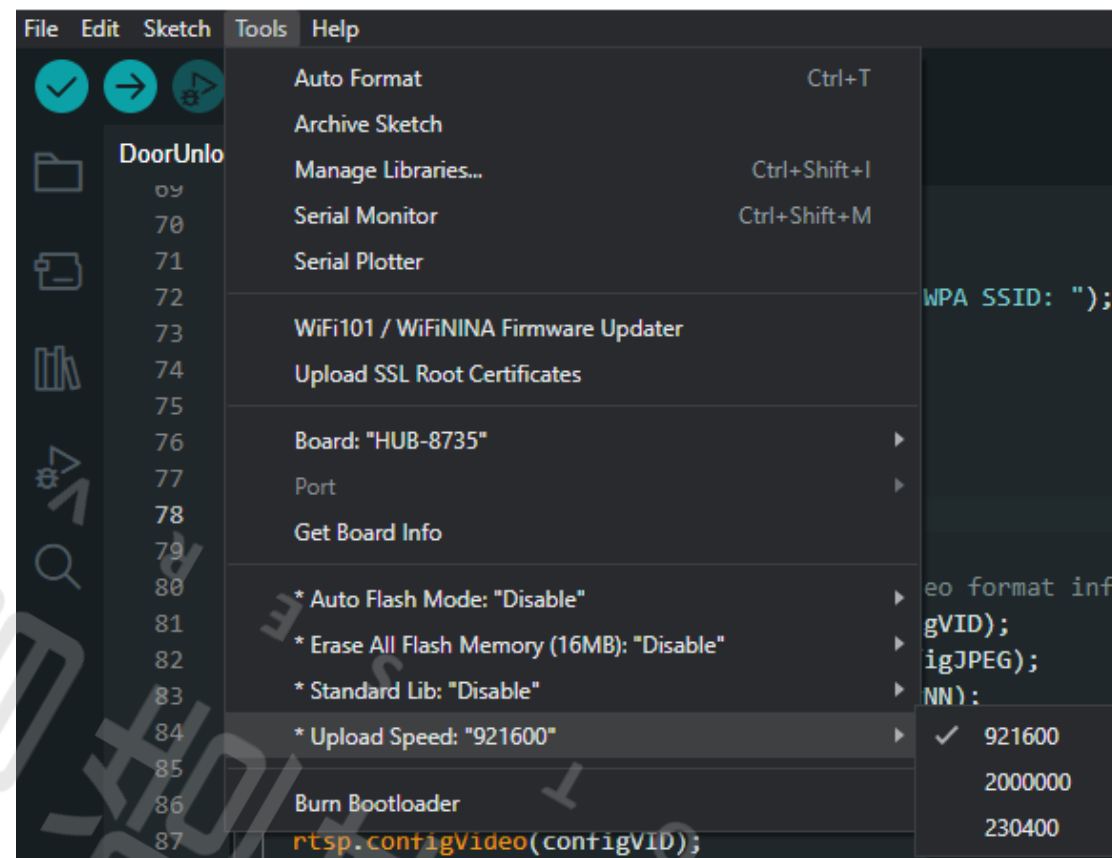
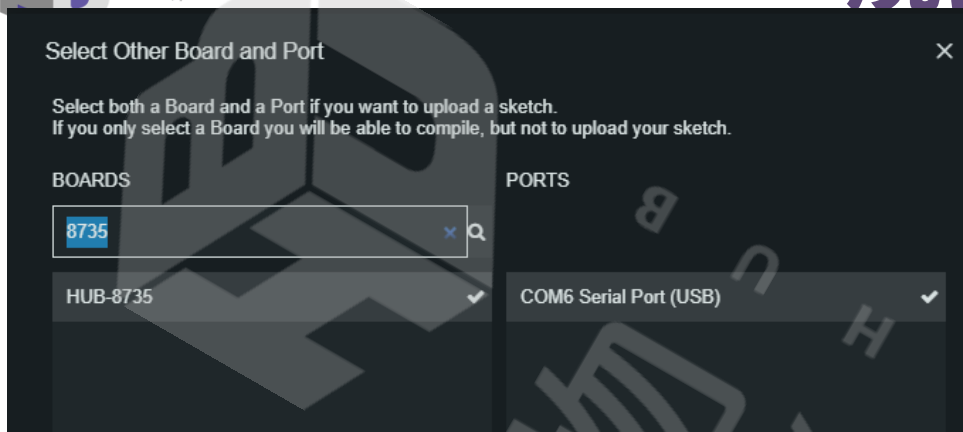


Serial Port 及
GND 對接
(U1T/U1R)



5V上電，進入燒錄模式

燒錄測試 (1)



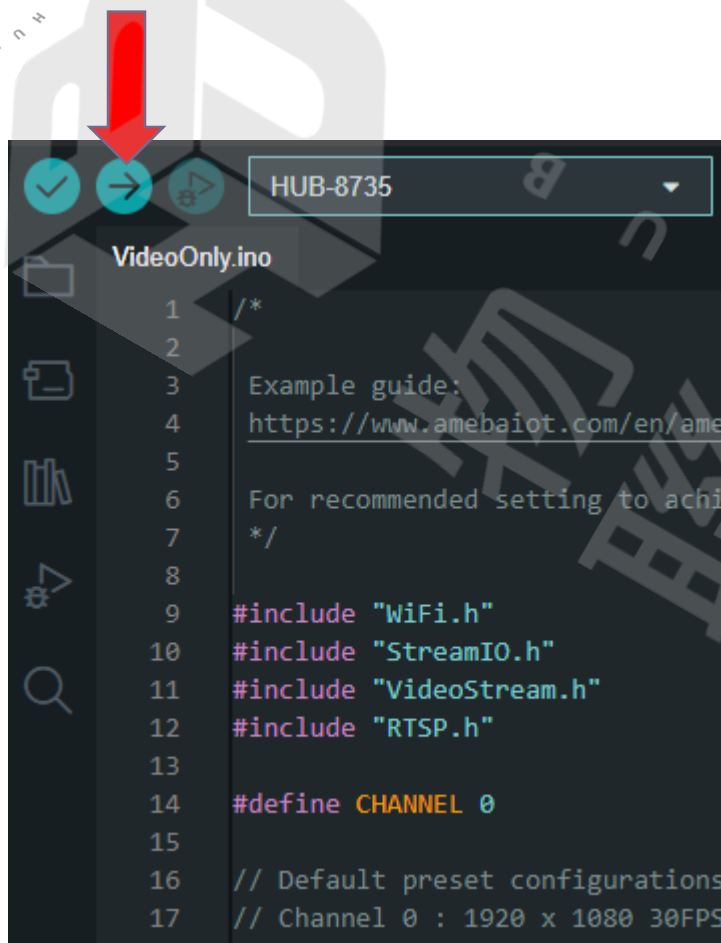
1. 確認PORTS

2. 確認Upload Speed

CP2102 => 921600

CH340G => 2000000

燒錄測試 (2)



燒錄成功會出現以下
message

```
Output
c:/users/rodneytai/appdata/local/arduino15/packages/ideashatch/
Sketch uses 4239360 bytes (25%) of program storage space. Maxim
Enter Flash Mode!
Start Upload Flash
Uploading.....upload success

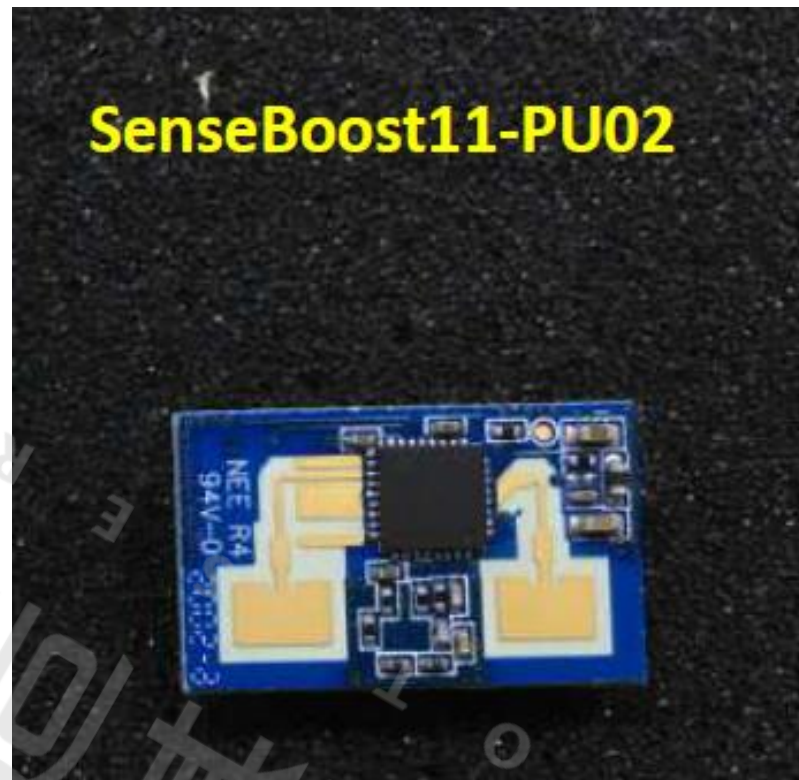
End Upload Flash
```

3. 確認後點下箭頭就會進行編譯上傳
4. 按下板子上的按鈕 (RESET)

毫米波雷達PU02

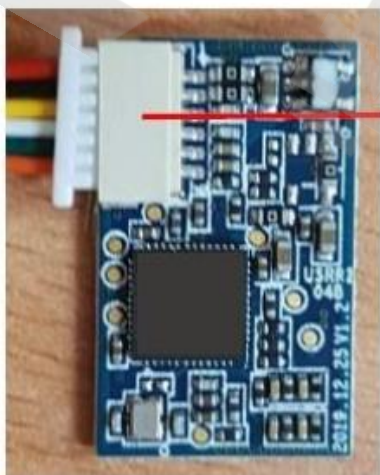
- 工作在24GHz的移動偵測雷達-ISM-頻段
- 偵測移動物體的方向和速度
- 集成低躁放大器
- 擴展操作溫度範圍從 -40°C 到 +70°C
- 外型尺寸極小化
- 客制化軟體配合不同產品應用

- ◆ 物體移動偵測
- ◆ 相對速度偵測
- ◆ 微動物體偵測
- ◆ 目標物距離偵測



PU02 腳位說明

PU02:



UART & GPIO Interface

Red	3.3V
Black	GND
Yellow	MCU-TX
White	MCU-RX
Green	GPIO1
Orange	GPIO2

腳位說明:

讀取資料只要Tx, Rx, 3.3V, GND共4個腳位

PIN#	I/O	In/Out	Description
1	Vcc	Input	Supply voltage
2	GND	Input	Analog ground
3	TX	Output	115200bps UART
4	RX	Input	115200bps UART
5	GPIO_1	Output	I/O port (Low: 0V, High: 2.6V)
6	GPIO_2	Output	I/O port (Low: 0V, High: 2.6V)

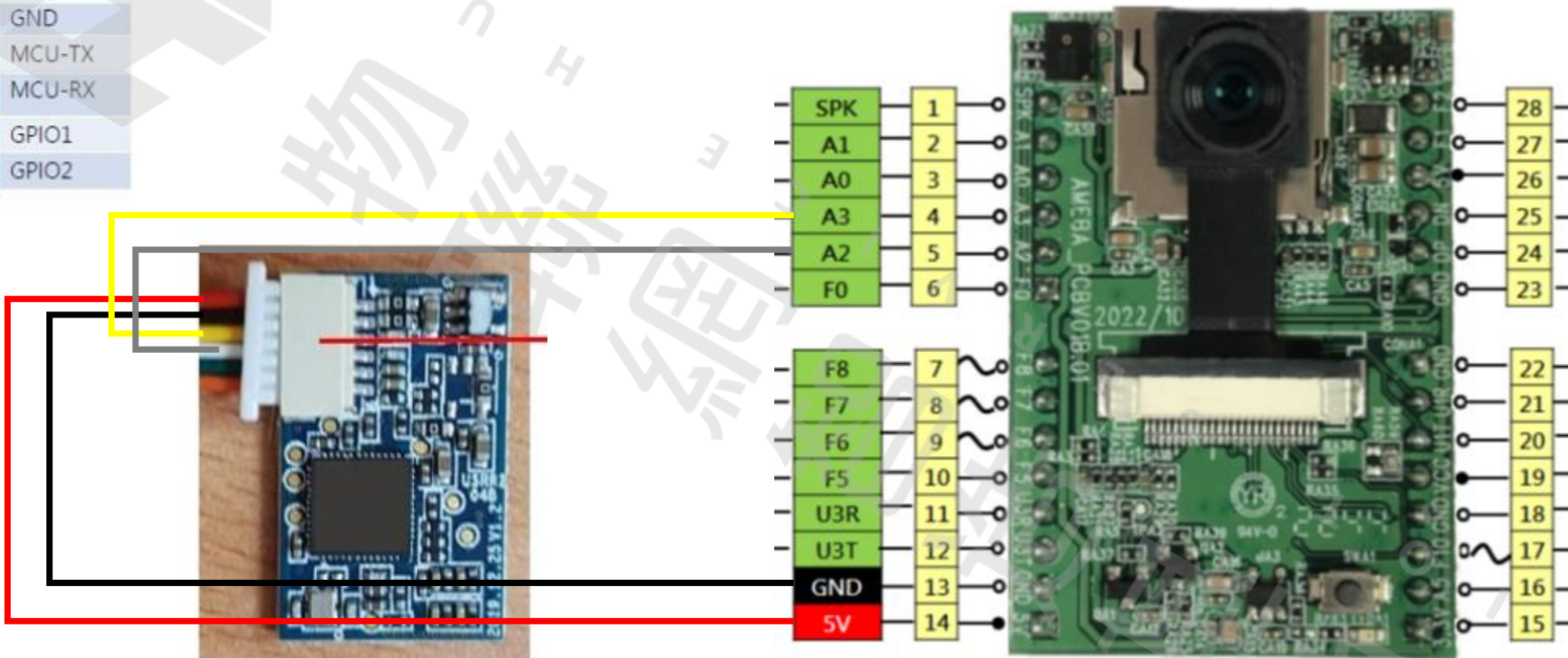
案例說明

- 利用PU02 與 HUB 8735 , 模擬門鈴通知。
- 當有物體/人接近時, 將開啟相機拍照儲存至SD Card, 並且發送LINE 文字訊息通知作為預警之用途。
- 過程使用Wi-Fi 透過NTP 取得網絡時間 (用以當作檔名) 。

PU02 連接HUB 8735

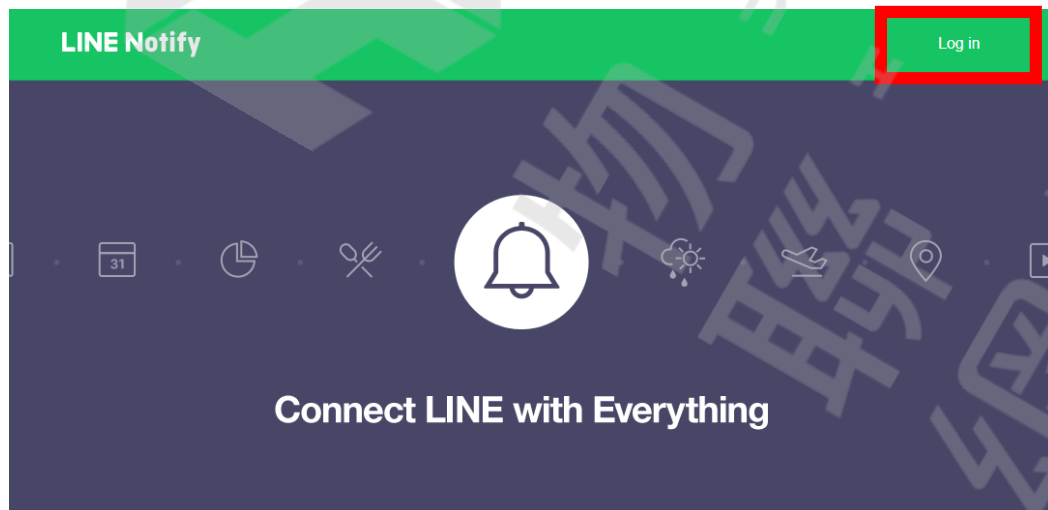
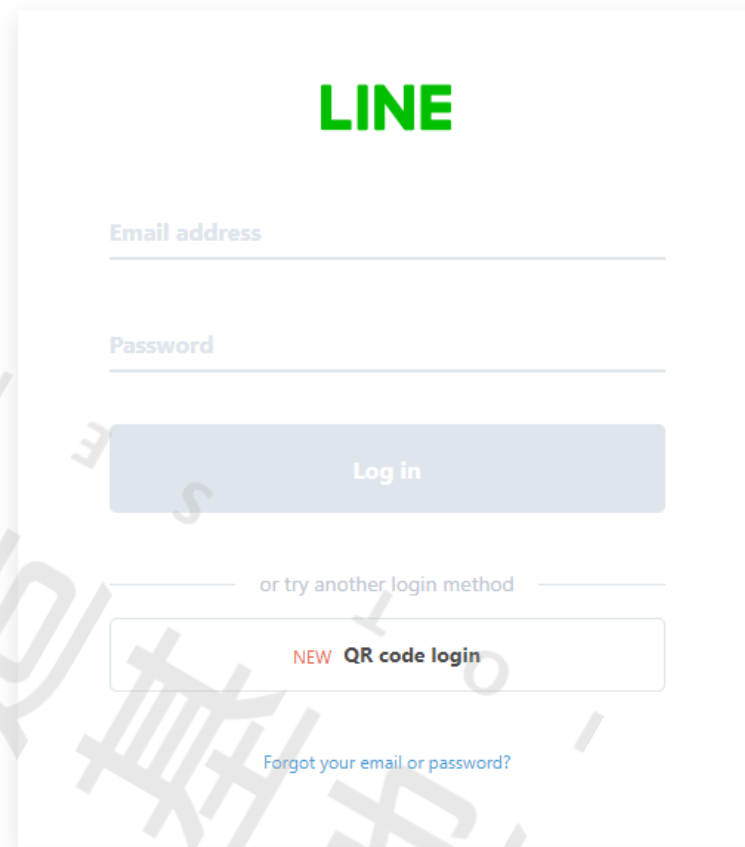
UART & GPIO Interface

Red	3.3V
Black	GND
Yellow	MCU-TX
White	MCU-RX
Green	GPIO1
Orange	GPIO2



LINE Notify

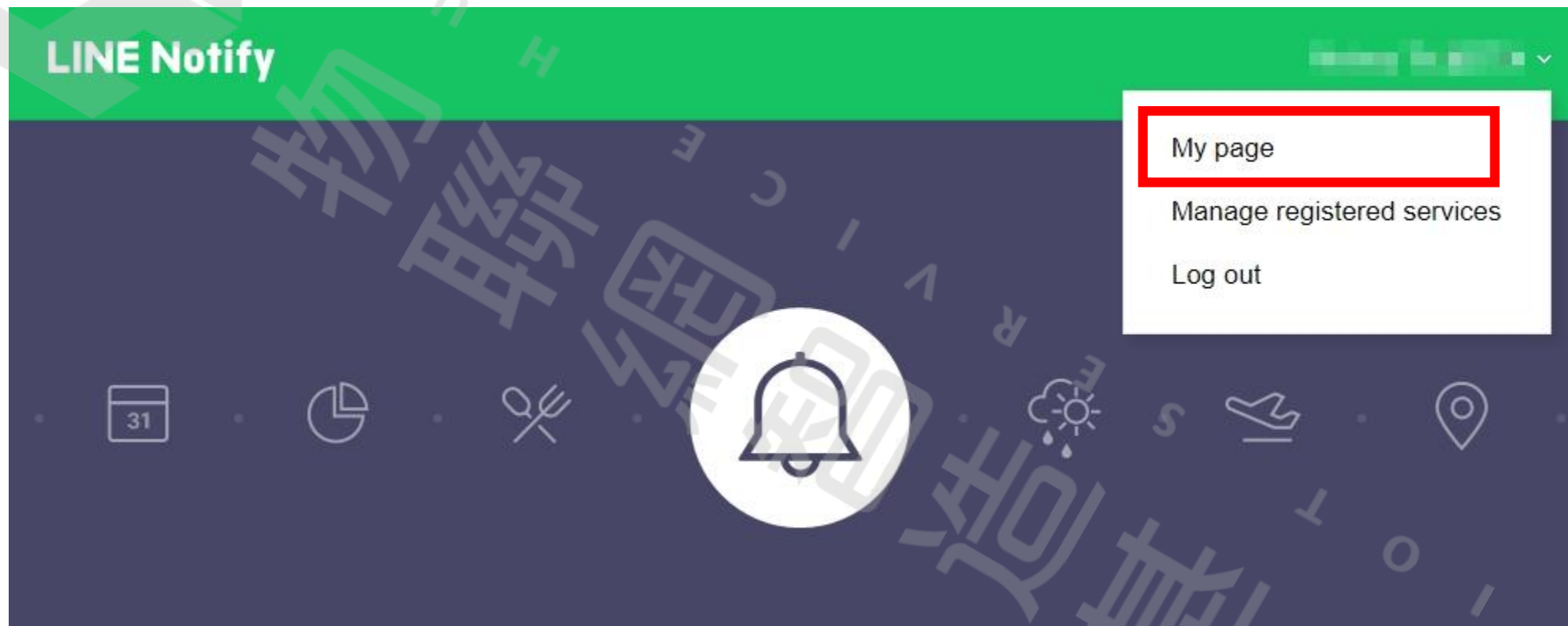
- <https://notify-bot.line.me/en/>

Receive web service notifications on LINE

Get notifications from LINE Notify's official account after connecting with your preferred web services.
You can receive notifications from multiple services in groups or 1-on-1 chats.

LINE Notify



LINE Notify

Generate token

Please enter a token name to be displayed before each notification.

Door Alert

Select a chat to send notifications to.

Search by group name

1-on-1 chat with LINE Notify

Note: Revealing your personal access token can allow a third party to obtain the names of your connected chats as well as your profile name.

Generate token

Your token is:

If you leave this page, you will not be able to view your newly generated token again. Please copy the token before leaving this page.

Copy

Close

務必存好Token，否則無法再次取得

注意事項

```
#define PU02_Serial Serial1
#include "PU02API.h"
```

Include PU02 的API

```
char ssid[] = " "; // your network SSID (name)
char pass[] = " "; // your network password
int status = WL_IDLE_STATUS;
```

輸入Wi-Fi SSID & Password

```
// LINE TOKEN
char* LINE_TOKEN = "{PASTE YOUR LINE TOKEN HERE}";
const char* LINE_SERVER = "notify-api.line.me";
const int LINE_PORT = 443;
```

```
// LINE TOKEN
char* LINE_TOKEN = "3";
const char* LINE_SERVER = "notify-api.line.me";
const int LINE_PORT = 443;
```

替換成自己的LINE TOKEN



程式簡說

運用Wi-Fi連接網絡
取得NTP時間

傳回來的時間為UNIX Time
需要再進行處理

```
// Get real-time network time(UNIX time), then parse to formatted time
time_t fetchNetworkTime()
{
    byte packetBuffer[NTP_PACKET_SIZE];
    // Clear the packet buffer
    memset(packetBuffer, 0, NTP_PACKET_SIZE);

    // Set the first byte of the packet to 0x1B
    packetBuffer[0] = 0x1B;

    // Send an NTP request to the time server
    udp.beginPacket(ntpServer, 123); // NTP port is 123
    udp.write(packetBuffer, NTP_PACKET_SIZE);
    udp.endPacket();

    // Wait for the response packet
    delay(1000);
    if (udp.parsePacket())
    {
        // Read the packet into the buffer
        udp.read(packetBuffer, NTP_PACKET_SIZE);

        // Extract the seconds since Jan 1, 1900
        unsigned long secondsSince1900 = (unsigned long)(packetBuffer[40] << 24) |
                                           (unsigned long)(packetBuffer[41] << 16) |
                                           (unsigned long)(packetBuffer[42] << 8) |
                                           (unsigned long)(packetBuffer[43]);

        // Convert NTP time to Unix timestamp
        const unsigned long seventyYears = 2208988800UL;
        unsigned long epochTime = secondsSince1900 - seventyYears;

        // Apply time zone offset
        time_t localTime = epochTime - (timeZone * 3600);

        return localTime;
    }
    return 0;
}
```

可更動數字
以更換時區

程式簡說

Initial FS 並開啟相機，
進行拍攝，後存入SD Card

```
// Capture and store images in SD Card
void capstoreImage(time_t unixTime)
{
    fs.begin();

    // Parse Unix Time to YYYY-MM-DD HH-mm
    struct tm *timeinfo;
    timeinfo = gmtime(&unixTime);

    char fileNameTime[75];

    snprintf(fileNameTime, sizeof(fileNameTime), "%04d-%02d-%02d_%02d-%02d",
             timeinfo->tm_year + 1900, timeinfo->tm_mon + 1, timeinfo->tm_mday,
             timeinfo->tm_hour, timeinfo->tm_min);

    String filePath = String(fs.getRootPath()) + "Stranger_" + String(fileNameTime) + ".jpg";
    File file = fs.open(filePath);

    char path[128];
    strcpy(path, filePath.c_str());

    delay(300);
    Camera.getImage(CHANNEL, &img_addr, &img_len);
    file.write((uint8_t*)img_addr, img_len);
    file.close();
    fs.setLastModTime(path,
                      timeinfo->tm_year + 1900,
                      timeinfo->tm_mon + 1,
                      timeinfo->tm_mday,
                      timeinfo->tm_hour,
                      timeinfo->tm_min,
                      timeinfo->tm_sec);

    fs.end();
}
```



程式簡說

發送LINE Message

```
// Send LINE Message (Only text)
void sendLineMsg(char *msg, bool isNotify)
{
    if (lineClient.connect(LINE_SERVER, 443))
    {
        lineClient.send(msg, !isNotify);
    }
}
```

防止連續發送訊息
所設之變數

```
// Variables for storing mmWave's previous values, and message sending period
int previousValue = 0;
bool initialReading = true;
unsigned long lastMessageTime = 0;
const unsigned long messageCooldown = 180000; // 180 seconds cooldown

const int threshold = 17;
const int fluctuationRange = 5; // Allow some fluctuation around the threshold value
bool initialMessageSent = false;
```



程式簡說

檢查PU02的值是否有低於threshold。
若有則進行拍照，發送LINE 通知動作。

比較目前&前一次的值，
若差異大則表示有物體接近，
此設計為避免false alarm

檢查是否與上一次發送訊息時間
有間隔，以免訊息氾濫。
除了開機後第一次訊息發送，
訊息之間的時間為180s。

```
// If PU02 returns value, check if the distance value matches condition,
// If so, send LINE Notify Message to alert or notify user
bool checkAndNotify(int sensorValue)
{
    // Check if the distance value is below the upper threshold (indicating something is close)
    if (sensorValue < threshold)
    {
        // If this is the initial reading, update the previousValue and set initialReading to false
        if (initialReading)
        {
            previousValue = sensorValue;
            initialReading = false;
            return false;
        }
        // Check if the absolute difference between the current value and the previous value
        // is greater than the fluctuation range
        int difference = abs(sensorValue - previousValue);
        if (difference > fluctuationRange)
        {
            // Enough fluctuation detected, trigger the notification
            unsigned long currentTime = millis();
            if (!initialMessageSent || (currentTime - lastMessageTime >= messageCooldown))
            {
                // Send the message here (you can modify this part to use your Line Notify function)
                Serial.println("Something is getting close!");
                capstoreImage(fetchNetworkTime());
                sendLineMsg("Something/Someone approached to your doorstep.&stickerPackageId=446&stickerId=2016", true);

                // Update the initial message status and the last message time
                initialMessageSent = true;
                lastMessageTime = currentTime;
                return true;
            }
            else if (currentTime - lastMessageTime <= messageCooldown)
            {
                Serial.println("Message Cool Down.");
            }
        }
    }

    // Update the previous value for the next check
    previousValue = sensorValue;
    return false;
}
```

結果示意

```

14:29:45.205 -> 48 get
14:29:45.813 -> 48 get
14:29:46.558 -> 45 get
14:29:47.460 -> 43 get
14:29:48.052 -> 43 get
14:29:50.915 -> 48 get
14:29:51.507 -> 48 get
14:29:52.110 -> 48 get
14:29:53.188 -> 12 get
14:30:00.800 -> 17 get
14:30:00.800 -> Something is getting close!
14:30:01.908 -> Register disk driver to Fatfs.
14:30:01.908 -> FATFS Register: disk driver 0
14:30:01.908 -> SD_Init 0
14:30:02.033 -> part_count = 0
14:30:02.079 -> SD_DeInit
14:30:02.487 -> connect host
14:30:02.487 -> connect host--
14:30:02.625 -> bcn_interval = 100
14:30:02.919 -> -----
14:30:02.919 -> POST /api/notify HTTP/1.1
14:30:02.919 -> Host: notify-api.line.me
14:30:02.919 -> Authorization: Bearer 3YqffGDd5t61ojL1JkzX0qK1CF7EAcZ0dkQ9cf17p58
14:30:02.919 -> Content-Length: 117
14:30:02.919 -> Content-Type: application/x-www-form-urlencoded
14:30:02.919 -> message=Something/Someone approached to your doorstep.&stickerPackageId=446&stickerId=2016&notificationDisabled=false
14:30:02.919 -> -----
14:30:06.109 -> 32 get
14:30:06.563 -> 37 get
14:30:06.641 -> bcn_interval = 100
14:30:06.984 -> 37 get
14:30:07.452 -> 37 get

```



Door Alert: Something/Someone approached to your doorstep.



2:30 PM

SD Card 內容

 Stranger_2023-07-28_16-56	7/28/2023 4:56 PM	JPG File	47 KB
 Stranger_2023-07-28_16-57	7/28/2023 4:57 PM	JPG File	77 KB
 Stranger_2023-07-28_16-59	7/28/2023 4:59 PM	JPG File	42 KB
 Stranger_2023-07-28_17-00	7/28/2023 5:00 PM	JPG File	38 KB
 Stranger_2023-07-28_17-12	7/28/2023 5:12 PM	JPG File	38 KB
 Stranger_2023-07-28_17-13	7/28/2023 5:13 PM	JPG File	37 KB
 Stranger_2023-07-28_17-15	7/28/2023 5:15 PM	JPG File	37 KB
 Stranger_2023-07-28_17-27	7/28/2023 5:27 PM	JPG File	38 KB
 Stranger_2023-07-28_17-31	7/28/2023 5:31 PM	JPG File	42 KB
 Stranger_2023-07-28_17-32	7/28/2023 5:32 PM	JPG File	41 KB
 Stranger_2023-07-28_17-36	7/28/2023 5:36 PM	JPG File	38 KB
 Stranger_2023-07-31_09-37	7/31/2023 9:37 AM	JPG File	44 KB
 Stranger_2023-07-31_09-42	7/31/2023 9:42 AM	JPG File	44 KB
 Stranger_2023-07-31_09-45	7/31/2023 9:45 AM	JPG File	38 KB



Thank you

