

【國產 IC 開發套件】

型號：HUB 8735 ultra

(RTL8735)

Arduino 使用說明

V1.0 (20240508)

Getting Started with HUB 8735 ultra

指導單位：經濟部工業局

主辦單位：財團法人資訊工業策進會

執行單位：物聯網智造基地



合作單位：振邦科技股份有限公司 **WinsTec**

目 錄

一、 REQUIRED ENVIRONMENT	1
二、 INTRODUCTION TO HUB 8735 ULTRA.....	1
(一) HUB 8735 ULTRA 腳位圖	2
(二) HUB 8735 ULTRA 硬體規格	2
三、 SETTING UP DEVELOPMENT ENVIRONMENT	4
(三) 安裝 ARDUINO IDE 開發環境	5
四、 USE FIRST EXAMPLE.....	9
(四) 編譯並上傳	9
(五) 測試 VIDEOONLY 範例.....	14
五、 WI-FI EXAMPLE	18
(一) 編譯並上傳	18
六、 BLE EXAMPLE.....	20
(一) 編譯並上傳	20
七、 I2C EXAMPLE	21
(一) 編譯並上傳	21
八、 SPI EXAMPLE	23
(一) 編譯並上傳	23
九、 例外處理.....	24
(一) UCFG FAIL.....	25
十、 參考資料.....	26
十一、 版本紀錄.....	26

圖目錄

圖 1、HUB 8735 ultra 與 Arduino Uno 尺寸示意.....	1
圖 2、HUB 8735 ultra 腳位圖.....	2
圖 3、HUB 8735 ultra USB download & debug connector.....	4
圖 4、確認裝置 com port.....	5
圖 5、尋找 HUB 8735 ultra 設定檔.....	6
圖 6、偏好設定.....	7
圖 7、開發版管理員.....	7
圖 8、選擇開發版.....	8
圖 9、開發板選 HUB-8735 ultra.....	8
圖 10、打開範例	9
圖 11、程式碼開啟	10
圖 12、程式碼驗證編譯.....	11
圖 13、編譯後完成訊息.....	12
圖 14、透過按鍵進入燒錄模式.....	13
圖 15、透過短路進入燒錄模式.....	13
圖 16、upload finish 字樣	14
圖 17、RTSP 的 IP	15
圖 18、開啟 VLC player.....	15
圖 19、輸入 RTSP 的網址	16
圖 20、Camera 的影像	17
圖 21、打開 WiFi 範本	18
圖 22、輸入 ssid, password	19
圖 23、WIFI 連線指令	19
圖 24、開啟 BLE 範例.....	20
圖 25、Source code	20
圖 26、手機搜尋畫面	21
圖 27、開啟 I2C 範例	22
圖 28、HUB 8735 ultra 與 LCD 1602 I2C 接線圖	22
圖 29、開啟 SPI 範例.....	23
圖 30、HUB 8735 ultra 與 ILI9341_TFT 接線圖.....	23
圖 31、UCFG 錯誤圖.....	25
圖 32、設定 921600.....	25

表目錄

表 1、硬體功能表.....	2
表 2、ILI9341_TFT PIN 定義.....	24

一、 Required Environment

目前 HUB 8735 ultra 支援 Windows 7/8/10 及更高版本的 32 位元/64 位元，Linux Ubuntu 以及 MAC OS 作業系統。並且使用 Arduino IDE 1.8.19 之後的版本。

二、 Introduction to HUB 8735 ultra

HUB 8735 ultra 適合開發各式的感測器或物聯網應用的開發板。它上面的介面有 Wi-Fi, BLE, GPIO, I2C, UART, SPI, PWM, ADC，這些介面可以接一些電子元件像是 LED 燈、開關、壓力計、溫濕度感測器、PM2.5 粉塵感測器等等。這些資料可以由內建的 Wi-Fi 上傳到雲端，搭配手機的 App 實現物聯網的實作。

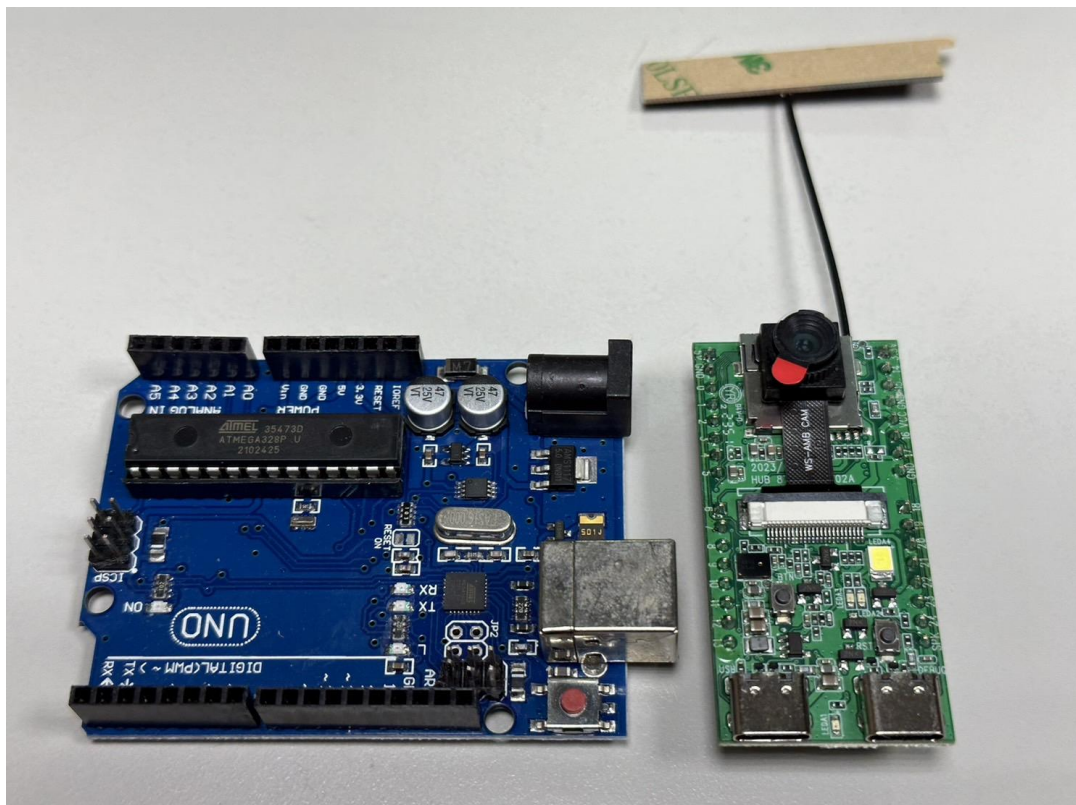


圖 1、HUB 8735 ultra 與 Arduino Uno 尺寸示意

資料來源：本計畫整理

HUB 8735 ultra 的大小大概是 Arduino Uno 的一半不到，圖 1 示意圖右邊是 HUB 8735 ultra，左邊則是 Arduino Uno。

HUB 8735 ultra 可透過 USB Type C 供電，引腳圖和功能請參考圖 2 和表 1。

(一) HUB 8735 ultra 腳位圖

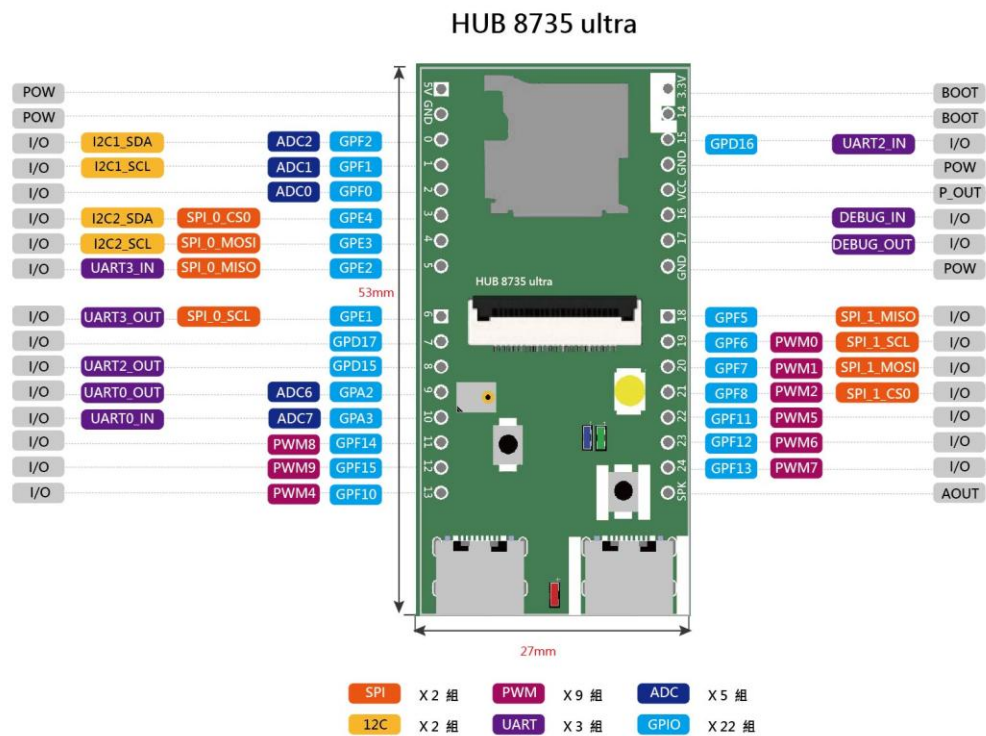


圖 2、HUB 8735 ultra 腳位圖

資料來源：本計畫整理

(二) HUB 8735 ultra 硬體規格

表 1、硬體功能表

	GPIO pin	ADC	PWM	UART	SPI	I2C	LED
0	PF2	ADC2				I2C1_SDA	
1	PF1	ADC1				I2C1_SCL	
2	PF0	ADC0					
3	PE4				SPI_SS	I2C_SDK	
4	PE3				SPI_MOSI	I2C_SCL	

5	PE2			SERIAL3_RX	SPI_MISO		
6	PE1			SERIAL3_TX	SPI_SCLK		
7	PD17						
8	PD15			SERIAL2_TX			
9	PA2	ADC6		SERIAL1_TX			
10	PA3	ADC7		SERIAL1_RX			
11	PF14		✓				
12	PF15		✓				BUTTON
13	PF10		✓				CAMERA FLASH LED
14	PA5						FLASH MODE
15	PD16			SERIAL2_RX			
16	PF3			LOG_RX			
17	PF4			LOG TX			
18	PF5		✓		SPI1_MISO		
19	PF6		✓		SPI1_SCLK		
20	PF7		✓		SPI1_MOSI		
21	PF8		✓		SPI1_SS		
22	PF11		✓				
23	PF12		✓				

24	PF13		✓				
25	PE6						LED_G (green)
26	PF9						LED_B (blue)
27	PA0	ADC4				I2C2_SCL	
28	PA1	ADC5				I2C2_SDA	
29	PE5						

資料來源：本計畫整理

三、Setting up Development Environment

HUB8735 ultra 內建 USB 轉 TTL 序列訊號的 IC，使用者可以直接透過 DEBUG USB 與電腦連接做燒錄及偵錯。

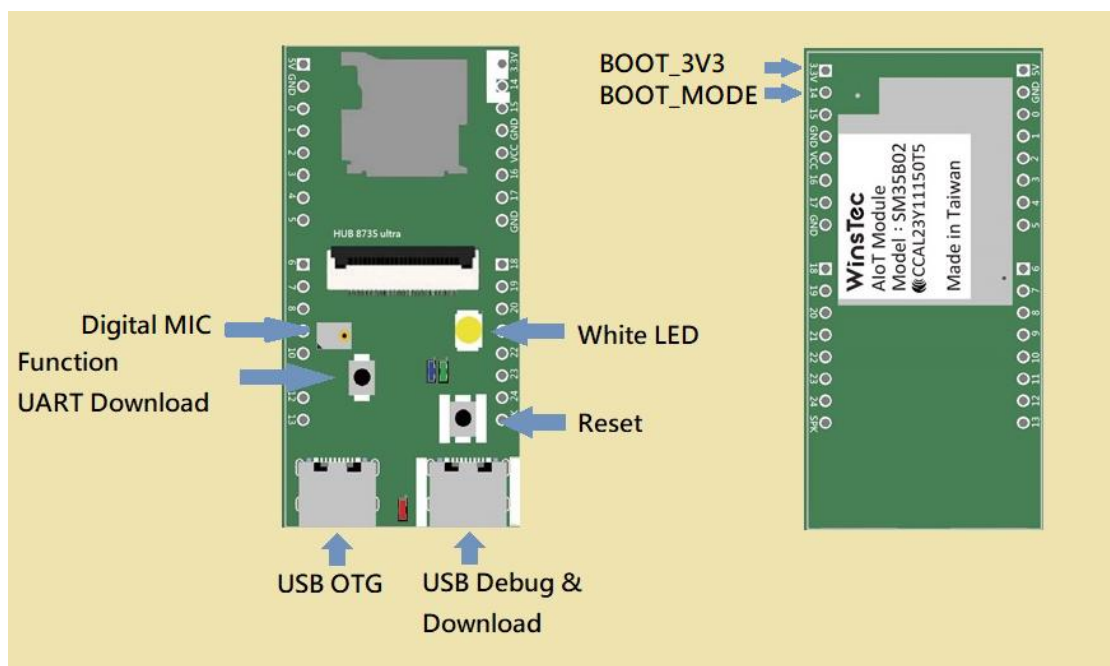


圖 3、HUB 8735 ultra USB download & debug connector

資料來源：本計畫整理

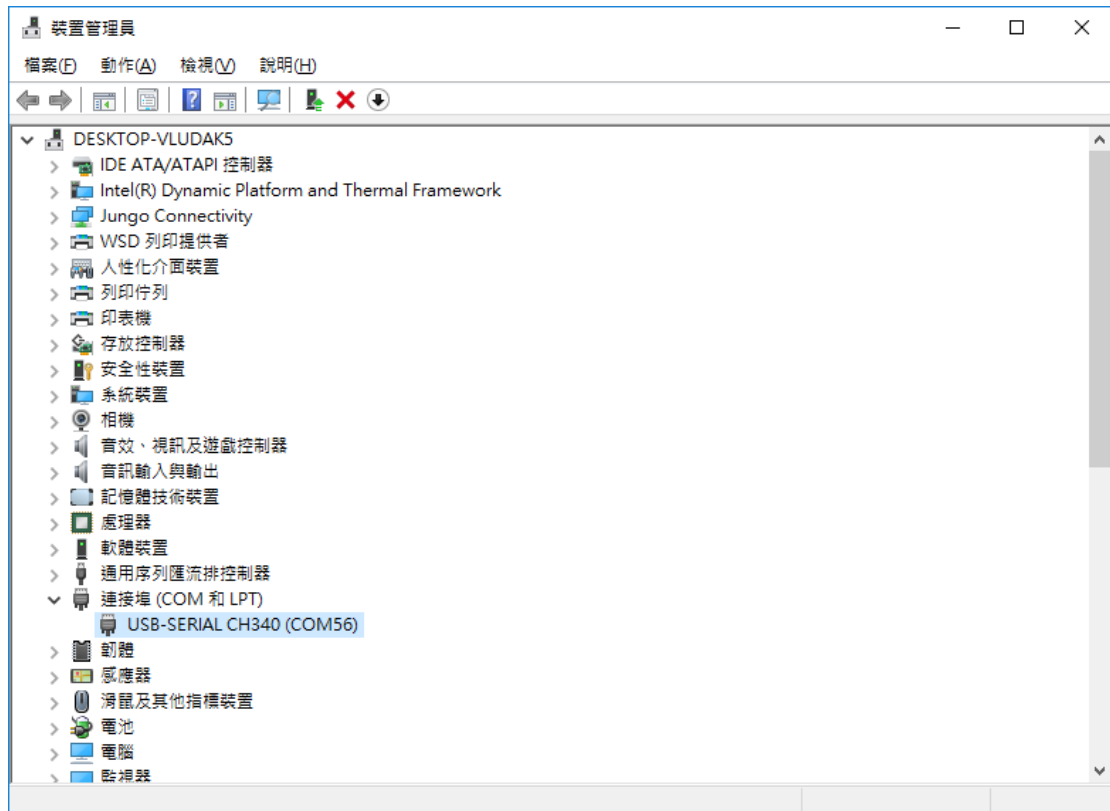


圖 4、確認裝置 com port

資料來源：本計畫整理

(三) 安裝 Arduino IDE 開發環境

Arduino IDE 在 1.8.19 版之後，支援第三方的硬體，因此我們可以在 Arduino IDE 上開發 HUB 8735 ultra，並共享 Arduino 上面的範例程式。在 Arduino 官方網站上可以找到下載程式：

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

安裝完之後，打開 Arduino IDE，為了讓 Arduino IDE 找到 HUB 8735 ultra 的設定檔，先到 “檔案” -> “偏好設定”

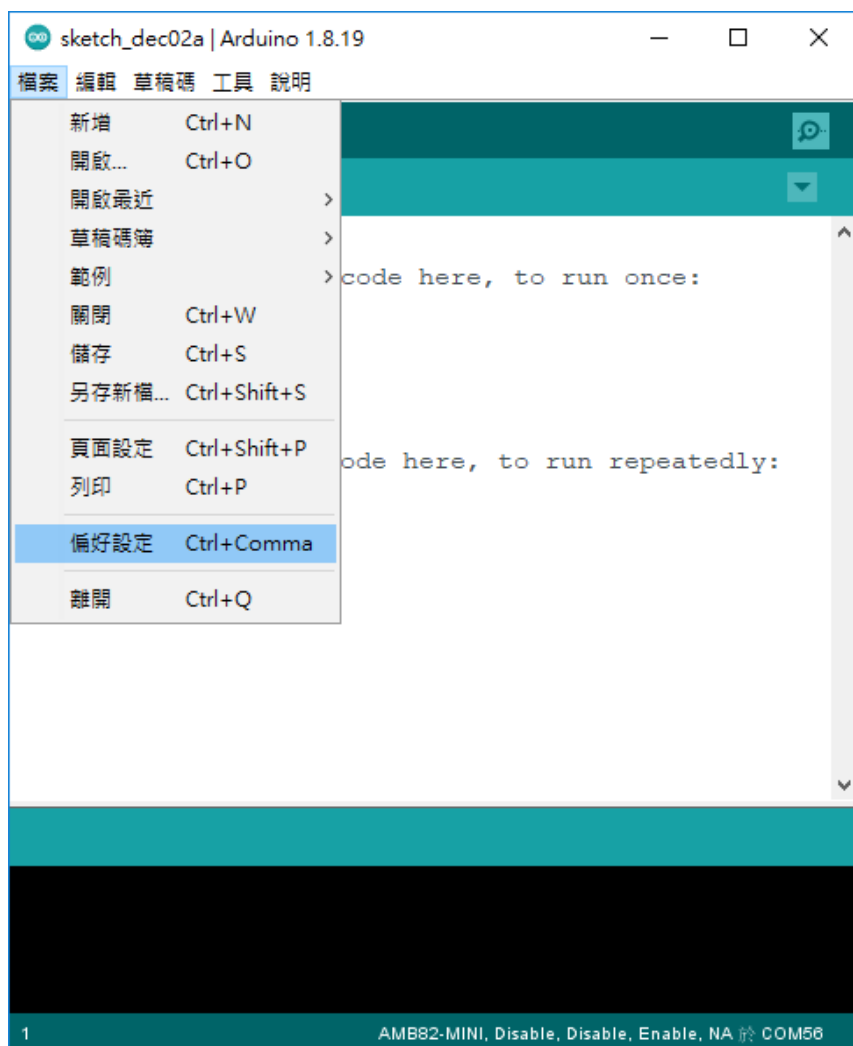


圖 5、尋找 HUB 8735 ultra 設定檔

資料來源：本計畫整理

然後在 其他開發板管理員網址(Additional Boards Manager URLs): 填入：

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



圖 6、偏好設定

資料來源：本計畫整理

接著準備選板子，到 “工具” -> “開發板:” -> “開發板管理員”

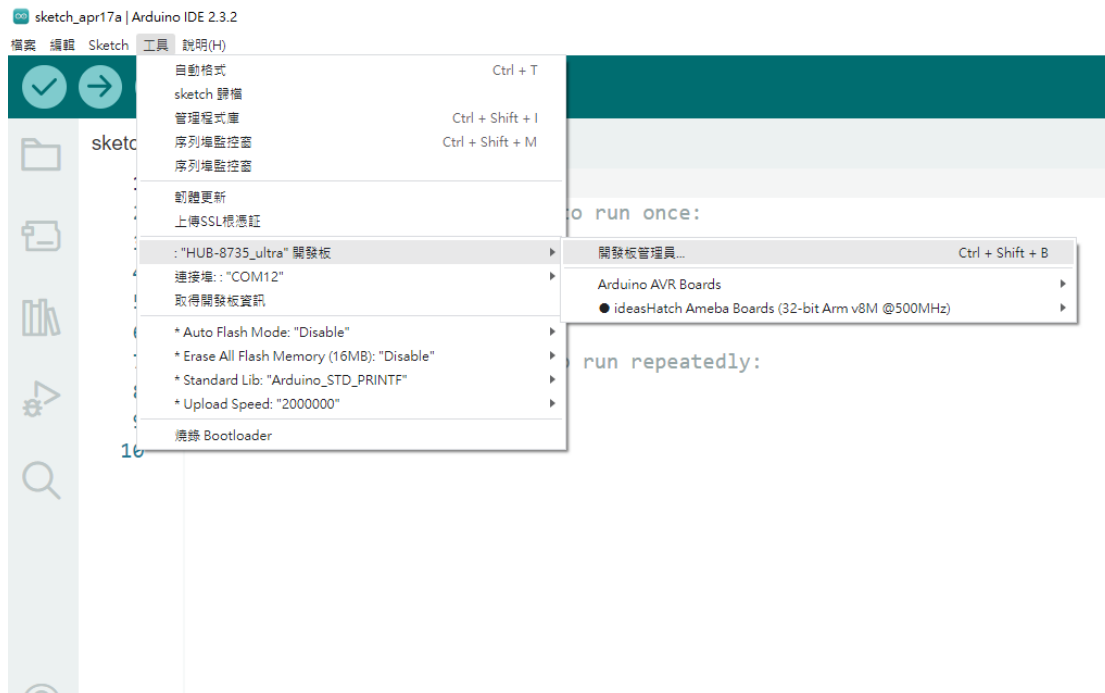


圖 7、開發版管理員

資料來源：本計畫整理

在搜尋視窗搜尋 “HUB 8735 ultra” 或 “ideasHatch” ，確認是包含開發板 HUB 8735 ultra(RTL8735B)的。選擇版本 4.0.10-Release 或較新版本後安裝。

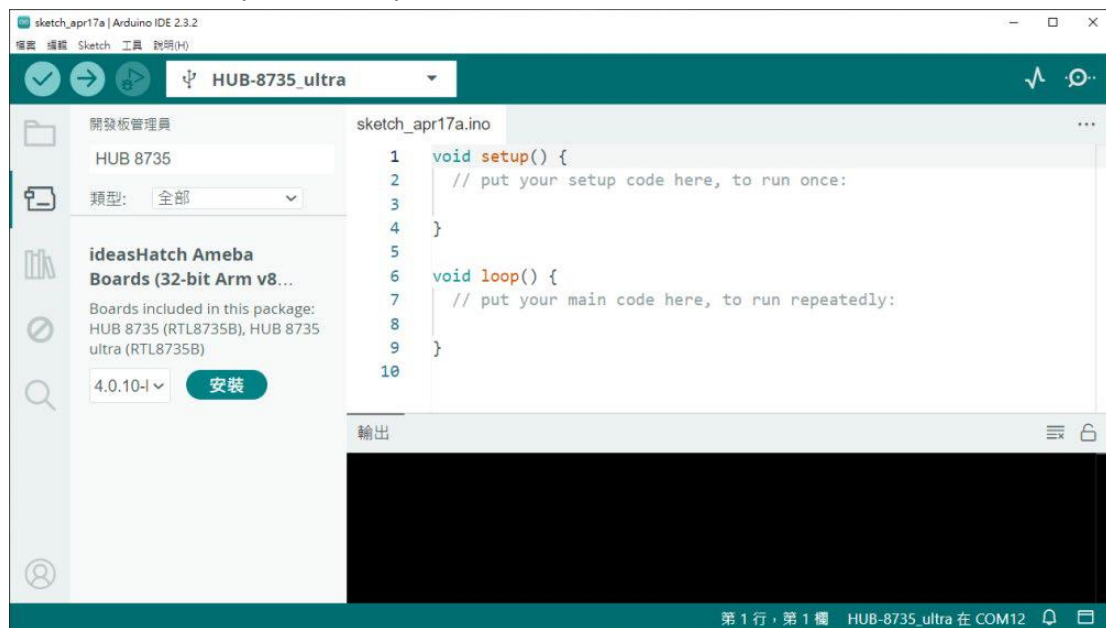


圖 8、選擇開發版

資料來源：本計畫整理

接著將開發板選擇 HUB 8735 ultra，選取 “工具” -> “開發板” -> “AmebaPro2 ARM(32-bits) Boards - ideasHatch” -> “HUB-8735 ultra”，這樣開發環境就設定完成了。

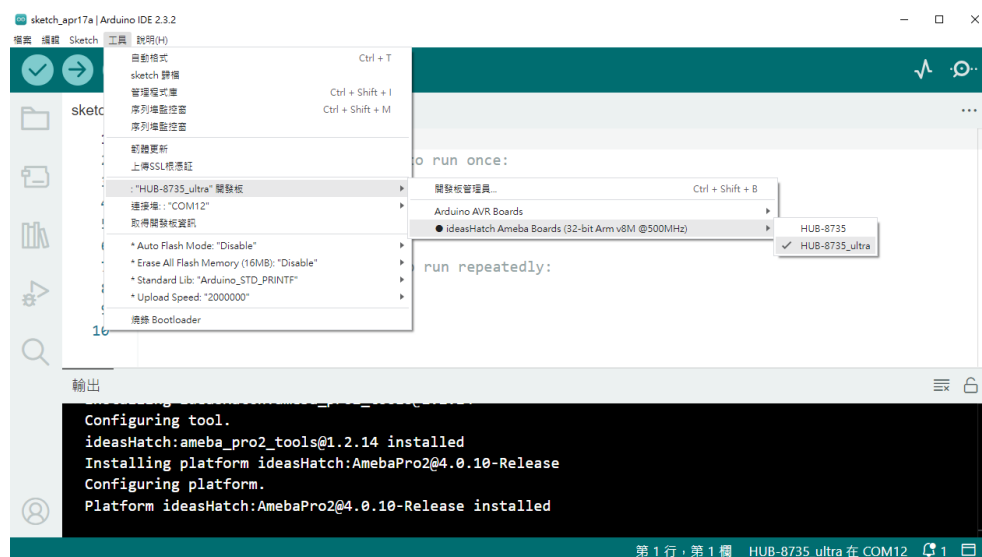


圖 9、開發板選 HUB-8735 ultra

資料來源：本計畫整理

也可以在 Additional Boards Manager URLs: 填入：

```
https://github.com/ambiot/ambpro2_arduino/raw/main/Arduino_package/  
package_realtek.com_amebapro2_index.json
```

來參考原廠開發板的更新。

四、Use First Example

(四) 編譯並上傳

Arduino IDE 提供許多內建的範例，這些範例只要打開經過編譯上傳之後就可以在板子上執行。我們直接嘗試 AmebaVideo 範例。

打開 “File” -> “Examples” -> “AmebaMultimedia” ->
“StreamRTSP” -> “VideoOnly”

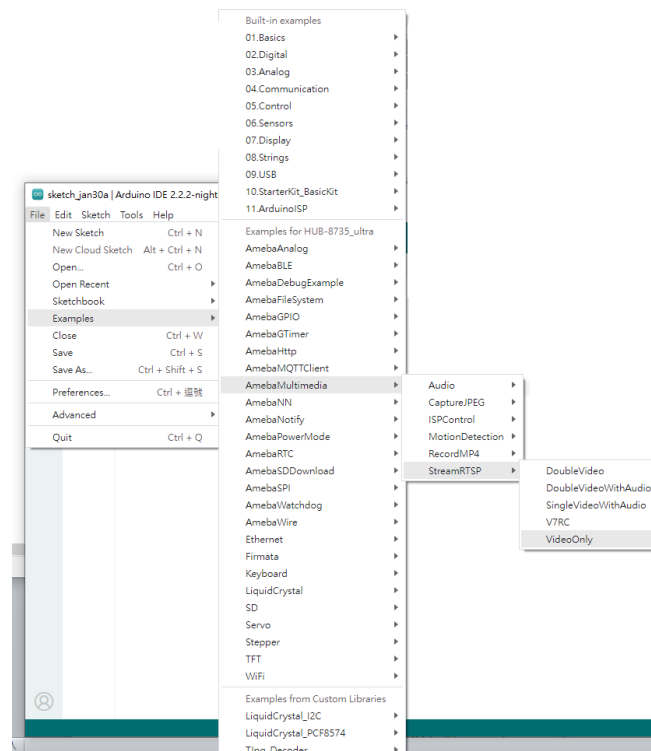
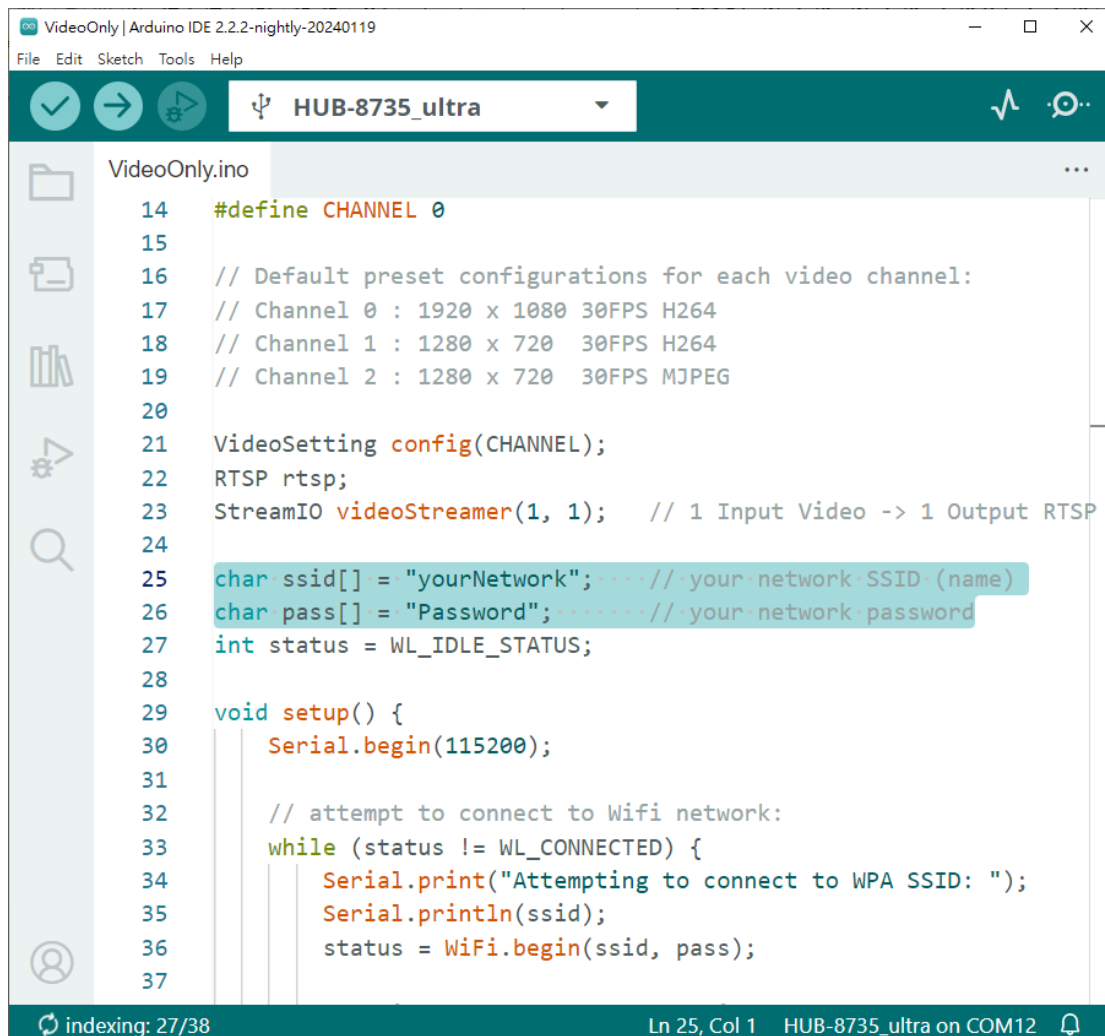


圖 10、打開範例

資料來源：本計畫整理

選了範例之後，Arduino IDE 會打開另一個視窗，裡面已經有寫好的程式碼。



```
VideoOnly.ino
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  int status = WL_IDLE_STATUS;
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      }
```

圖 11、程式碼開啟

資料來源：本計畫整理

由於 RTSP 需要連網，所以在範例的 25.26 行輸入 WiFi 的 SSID 及 Password。

注意：RTSP 連網需在同一個網域下，也就是電腦跟 HUB 8735 ultra 須連上同一個網域。

以上修改完後編譯程式碼，點選 “草稿碼” -> “驗證/編譯”

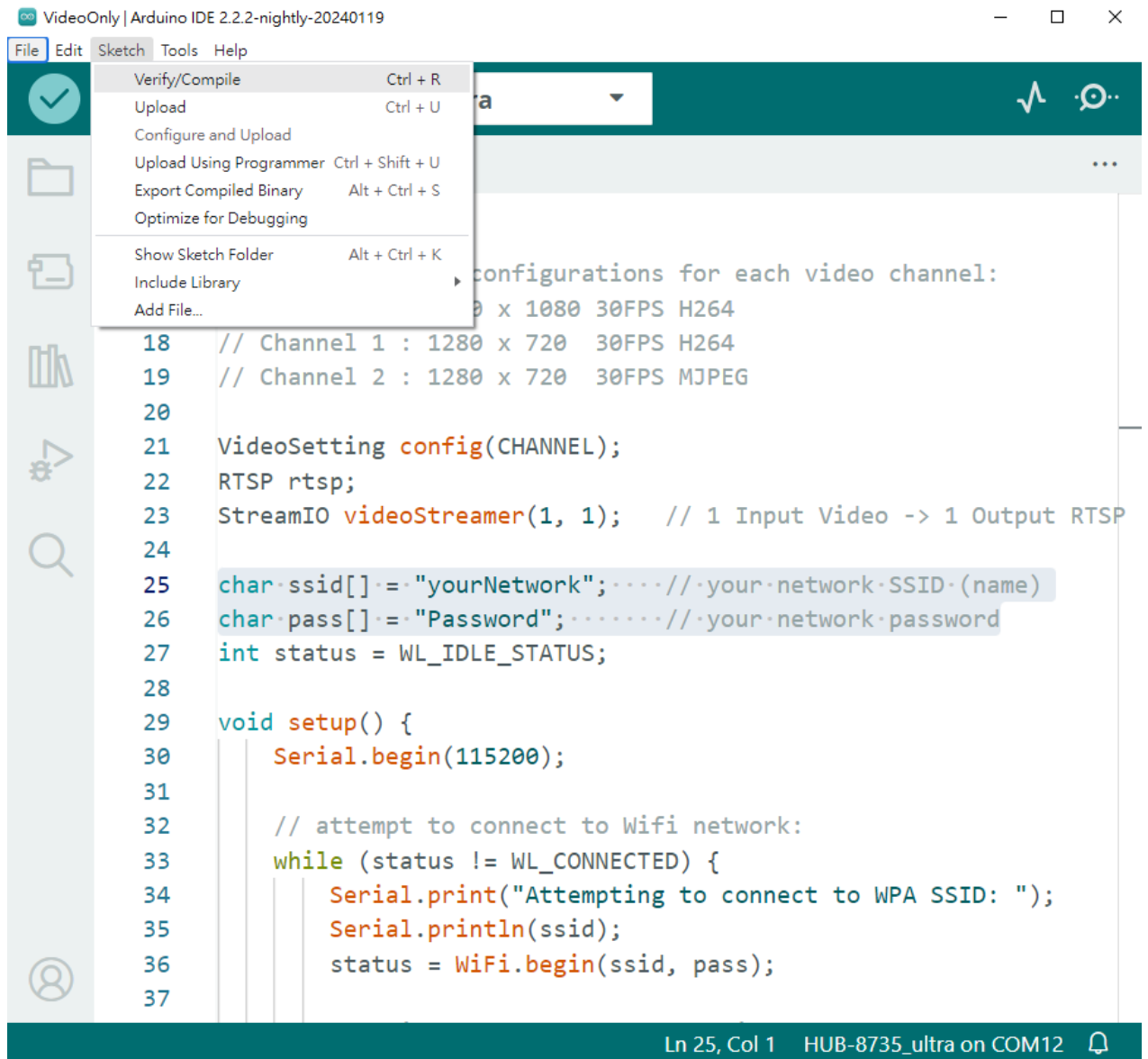
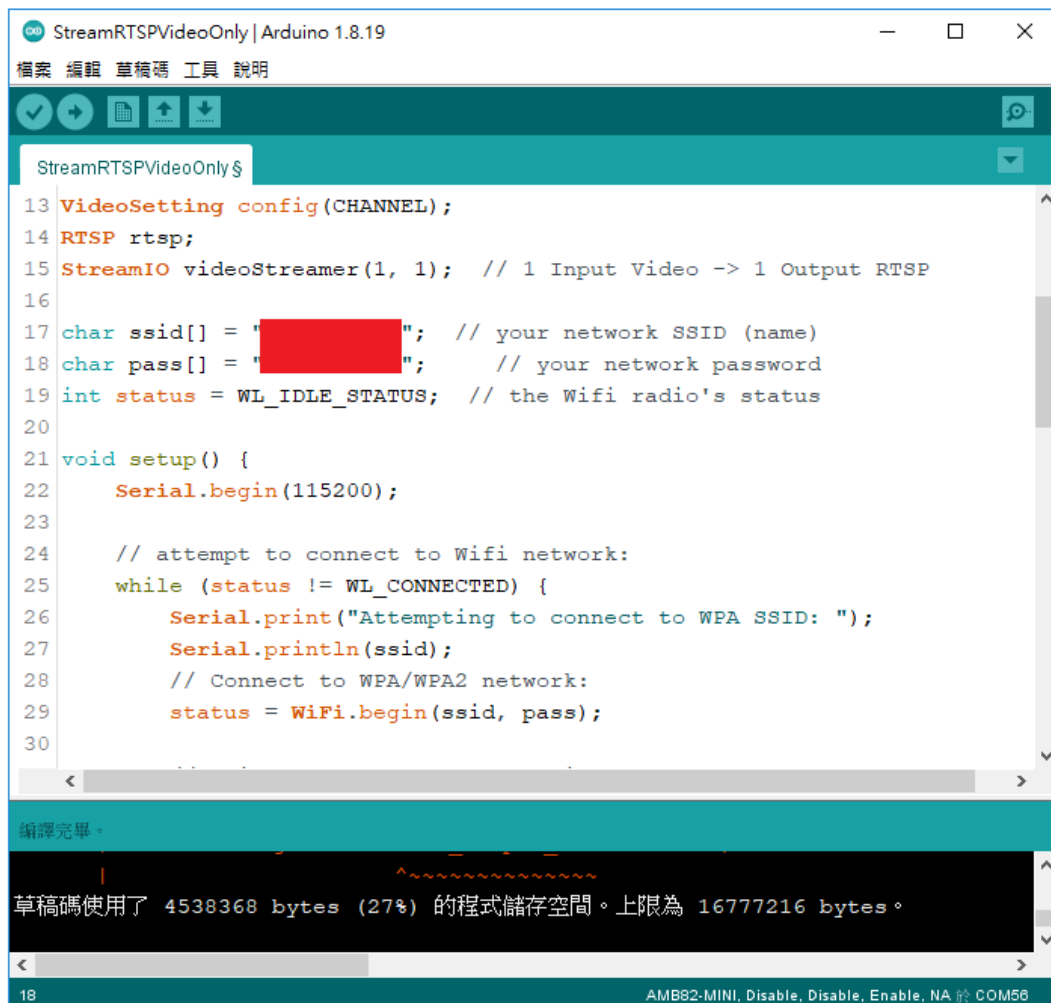


圖 12、程式碼驗證編譯

資料來源：本計畫整理

接著會看到 Arduino IDE 下方的訊息列出現一些訊息，代表它正在編譯。編譯完成之後，會看到類似底下的訊息。（沒有錯誤訊息就代表沒問題）



```
StreamRTSPVideoOnly | Arduino 1.8.19
檔案 編輯 草稿碼 工具 說明

StreamRTSPVideoOnly$
13 VideoSetting config(CHANNEL);
14 RTSP rtsp;
15 StreamIO videoStreamer(1, 1); // 1 Input Video -> 1 Output RTSP
16
17 char ssid[] = " "; // your network SSID (name)
18 char pass[] = " "; // your network password
19 int status = WL_IDLE_STATUS; // the Wifi radio's status
20
21 void setup() {
22     Serial.begin(115200);
23
24     // attempt to connect to Wifi network:
25     while (status != WL_CONNECTED) {
26         Serial.print("Attempting to connect to WPA SSID: ");
27         Serial.println(ssid);
28         // Connect to WPA/WPA2 network:
29         status = WiFi.begin(ssid, pass);
30
21
編譯完畢。
草稿碼使用了 4538368 bytes (27%) 的程式儲存空間。上限為 16777216 bytes。
18 AMB82-MINI, Disable, Disable, Enable, NA 於 COM58
```

圖 13、編譯後完成訊息

資料來源：本計畫整理

接著我們就可以上傳編譯好的程式碼到 HUB 8735 ultra 上面。先確定 HUB 8735 ultra 進入燒錄模式。要進入燒錄模式方法有兩種，先壓著功能鍵，然後按下 Reset 鍵在放開，最後再放開功能鍵。

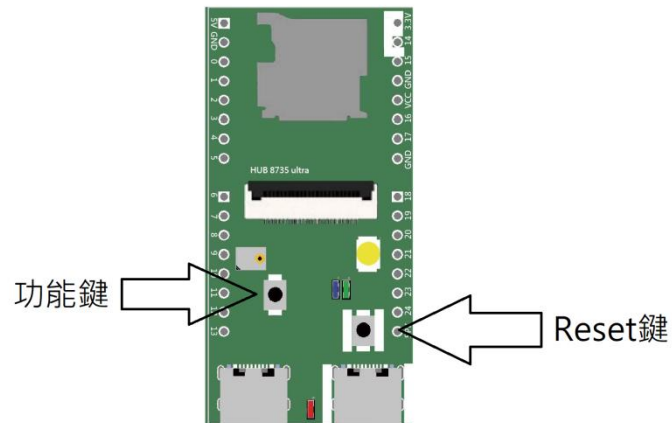


圖 14、透過按鍵進入燒錄模式

資料來源：本計畫整理

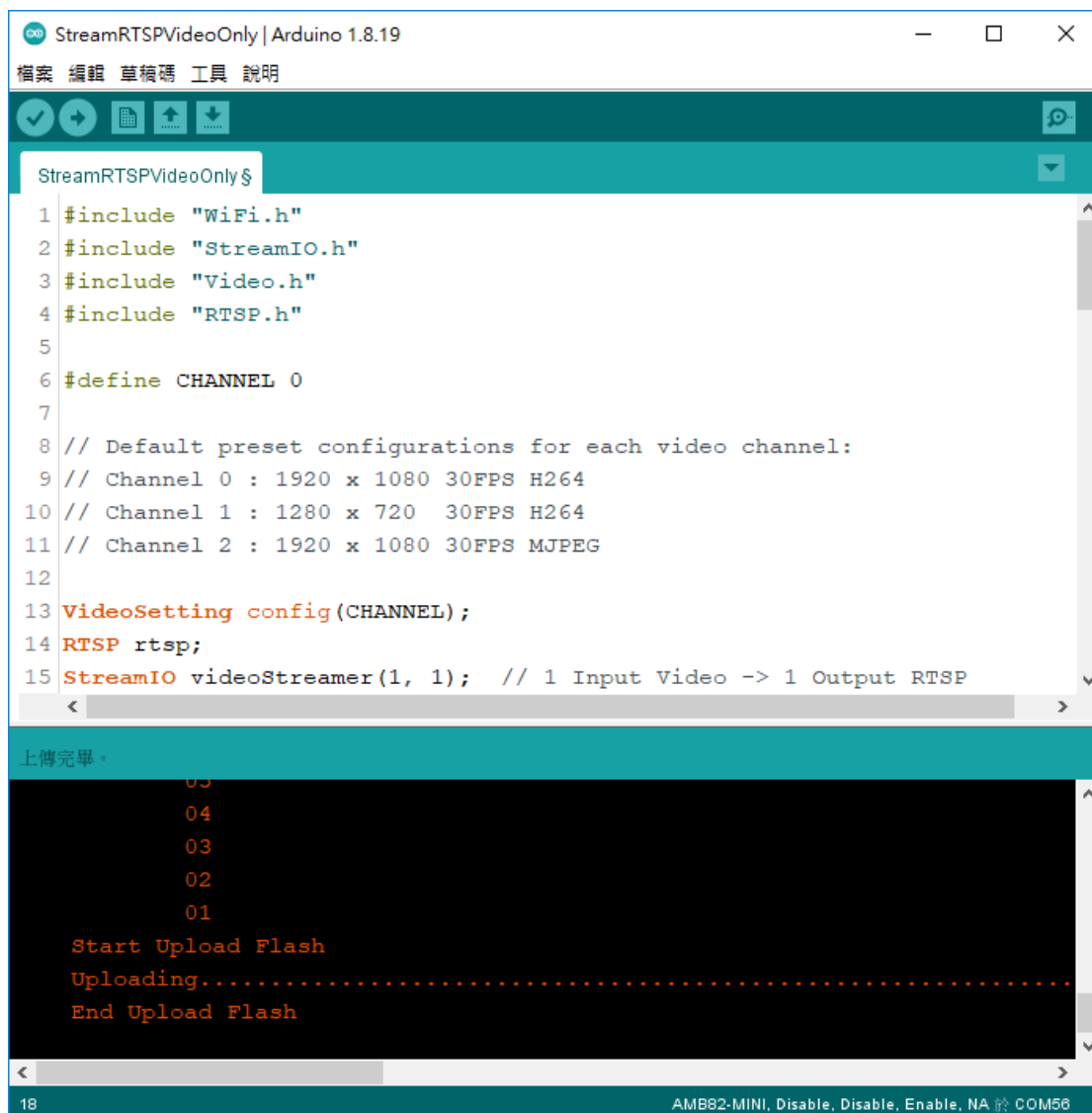
可以透過 Arduino IDE 內建序列埠監控窗檢查是否進入燒錄模式，同時注意監控窗的鮑率需選擇 115200。上傳完成後，會在下方的輸出窗口印出成功訊息。

另一種進入燒錄方法則是短路 HUB 8735 ultra 的 BOOT_MODE (14)跟 BOOT_V3P3 (3.3V)後按下 RESET 按鍵進入燒錄模式



圖 15、透過短路進入燒錄模式

資料來源：本計畫整理



```
StreamRTSPVideoOnly | Arduino 1.8.19
檔案 編輯 草稿碼 工具 說明

StreamRTSPVideoOnly $
1 #include "WiFi.h"
2 #include "StreamIO.h"
3 #include "video.h"
4 #include "RTSP.h"
5
6 #define CHANNEL 0
7
8 // Default preset configurations for each video channel:
9 // Channel 0 : 1920 x 1080 30FPS H264
10 // Channel 1 : 1280 x 720 30FPS H264
11 // Channel 2 : 1920 x 1080 30FPS MJPEG
12
13 VideoSetting config(CHANNEL);
14 RTSP rtsp;
15 StreamIO videoStreamer(1, 1); // 1 Input Video -> 1 Output RTSP

上傳完畢。
05
04
03
02
01
Start Upload Flash
Uploading.....
End Upload Flash

18 AMB82-MINI, Disable, Disable, Enable, NA 於 COM56
```

圖 16、upload finish 字樣

資料來源：本計畫整理

(五) 測試 VideoOnly 範例

VideoOnly 範例可以在 RTSP 上看到 Camera 影像。

首先移除短路 HUB 8735 ultra 的 BOOT_MODE (14)跟 BOOT_V3P3 (3.3V)

後按下 RESET 按鍵。透過 UART console 可以看到系統啟動後會主動去連設定

的 SSID。連上後會看到一組 RTSP 的 IP。

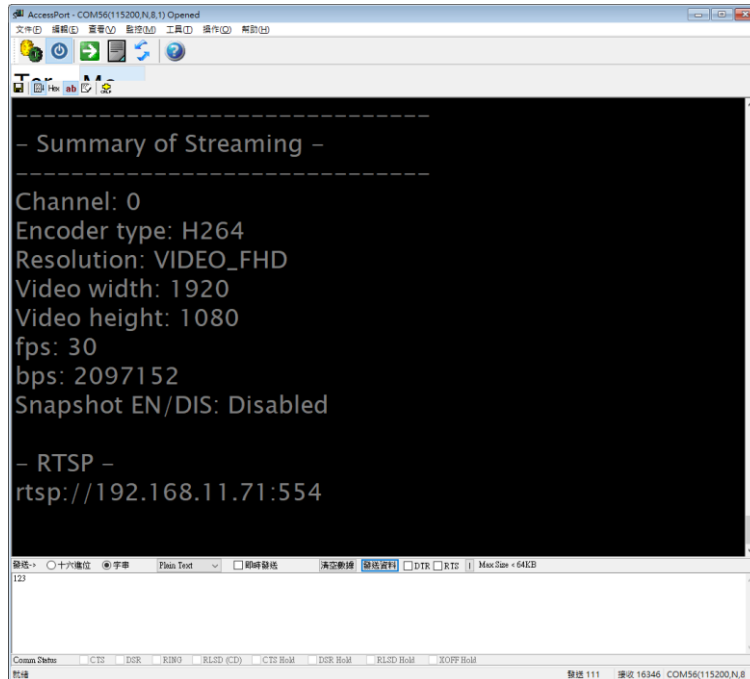


圖 17、RTSP 的 IP

資料來源：本計畫整理

開啟 VLC player，並選擇開啟網路串流。

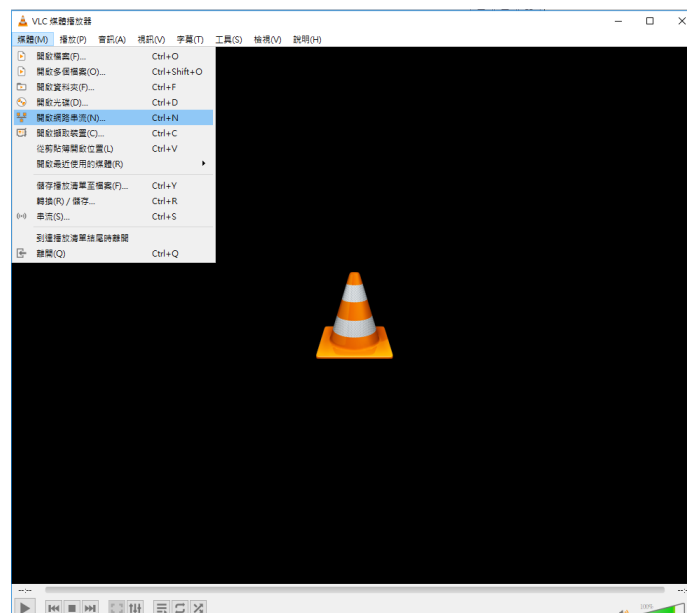


圖 18、開啟 VLC player

資料來源：本計畫整理

輸入 RTSP 的位置後按播放，就可以看到 Camera 的影像。

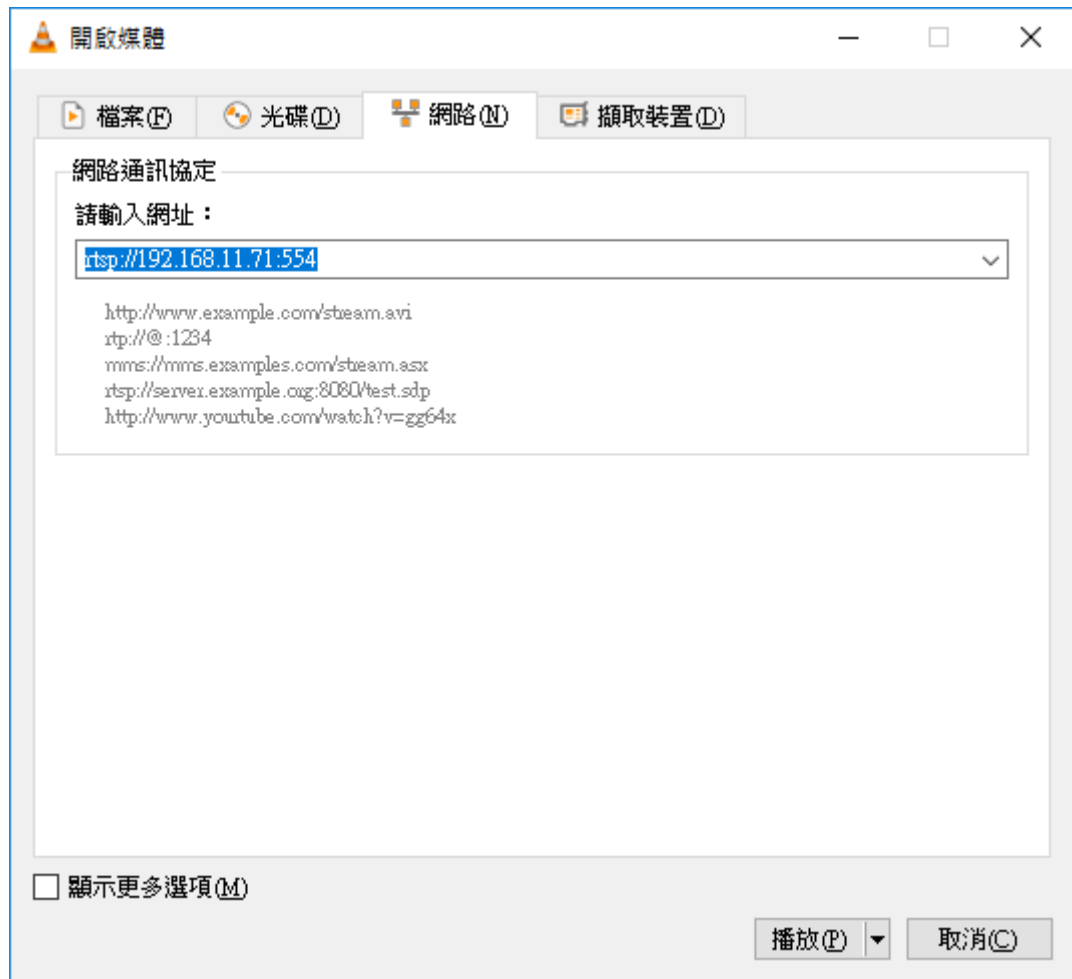


圖 19、輸入 RTSP 的網址

資料來源：本計畫整理

如果要讓影片更流暢，可以勾選顯示更多選項，並將快取改成 0 毫秒。

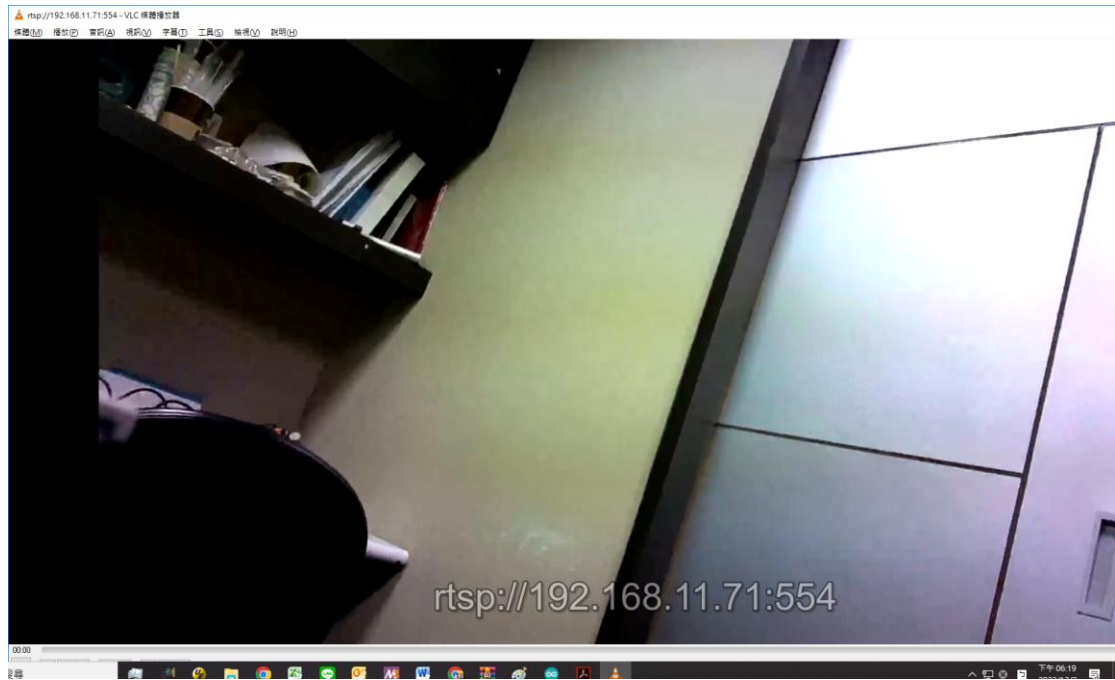


圖 20、Camera 的影像

資料來源：本計畫整理

五、Wi-Fi Example

(一) 編譯並上傳

如何啓用 Wi-Fi 功能，可以從 Arduino WiFi 範本中找到相關範例，以 Connect WPA_Security 爲範例。

打開“File”->“Examples”->“WiFi”->“ConnectToWiFi”->“WPA_Security”。

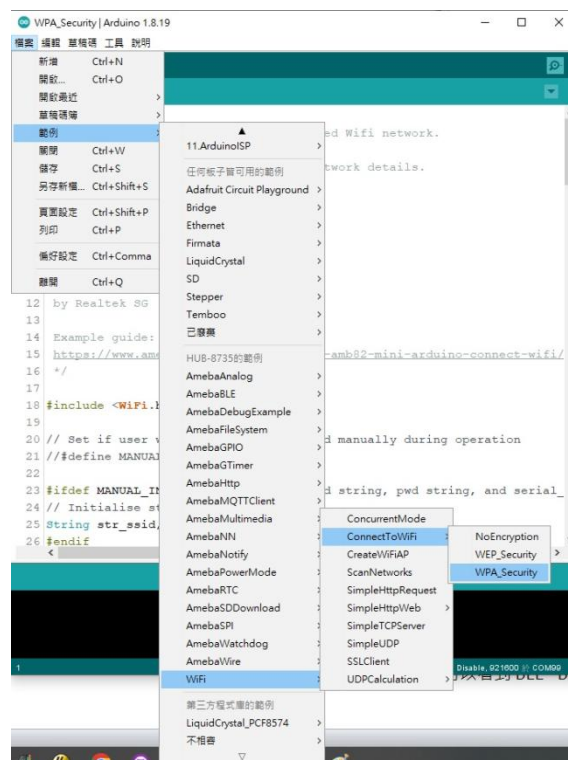


圖 21、打開 WiFi 範本

資料來源：本計畫整理

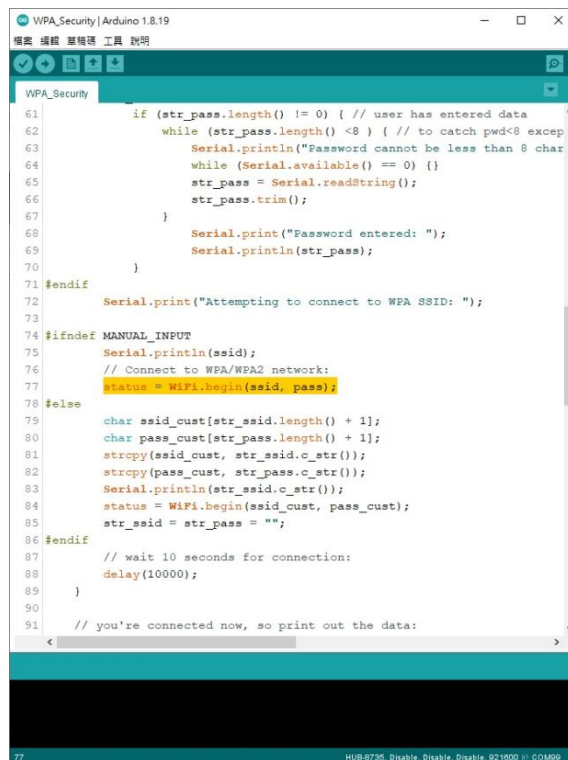
開啓後可以看到 WiFi 設定帳號密碼及初始化相關設定。



```
31
32 // UTF-8 encoding can also be used for SSID with emoji characters
33 // Emoji characters can be converted into UTF-8 at https://mothereff.in
34 // char ssid[] = "\xe2\x9c\x8c\xef\x88\x8f Ameba \xe2\x9c\x8c\xef\x88\x8f";
35
36 char ssid[] = "Network_SSID"; // your network SSID (name)
37 char pass[] = "Password"; // your network password
38 int status = WL_IDLE_STATUS; // Indicator of Wifi status
39
40 void setup() {
41   //Initialize serial and wait for port to open:
42   Serial.begin(115200);
43   while (!Serial) {
44     ; // wait for serial port to connect. Needed for native USB port
45   }
46
47   // attempt to connect to Wifi network:
48   while (status != WL_CONNECTED) {
49     #ifdef MANUAL_INPUT
50       Serial.println("Enter your ssid");
51       while (Serial.available() == 0) {}
52       str_ssid = Serial.readString();
53       str_ssid.trim();
54       Serial.print("SSID entered: ");
55       Serial.println(str_ssid);
56
57       Serial.println("Enter your password");
58       while (Serial.available() == 0) {}
59       str_pass = Serial.readString();
60       str_pass.trim();
61       if (str_pass.length() != 0) { // user has entered data
```

圖 22、輸入 ssid, password

資料來源：本計畫整理



```
61       if (str_pass.length() != 0) { // user has entered data
62         while (str_pass.length() < 8) { // to catch pwd<8 except
63           Serial.println("Password cannot be less than 8 char");
64           while (Serial.available() == 0) {}
65           str_pass = Serial.readString();
66           str_pass.trim();
67         }
68         Serial.print("Password entered: ");
69         Serial.println(str_pass);
70       }
71     #endif
72     Serial.print("Attempting to connect to WPA SSID: ");
73
74     #ifdef MANUAL_INPUT
75       Serial.println(ssid);
76       // Connect to WPA/WPA2 network:
77       status = WiFi.begin(ssid, pass);
78     #else
79       char ssid_cust[str_ssid.length() + 1];
80       char pass_cust[str_pass.length() + 1];
81       strcpy(ssid_cust, str_ssid.c_str());
82       strcpy(pass_cust, str_pass.c_str());
83       Serial.println(str_ssid.c_str());
84       status = WiFi.begin(ssid_cust, pass_cust);
85       str_ssid = str_pass = "";
86     #endif
87     // wait 10 seconds for connection:
88     delay(10000);
89   }
90
91   // you're connected now, so print out the data:
```

圖 23、WIFI 連線指令

資料來源：本計畫整理

六、BLE Example

(一) 編譯並上傳

如何啓用 BLE 功能，可以從 Arduino Ameba_BLE 範本中找到相關範例，以 BLEBeacon 爲範例。

打開“File”->“Examples”->“AmebaBLE”->“BLEBeacon”。

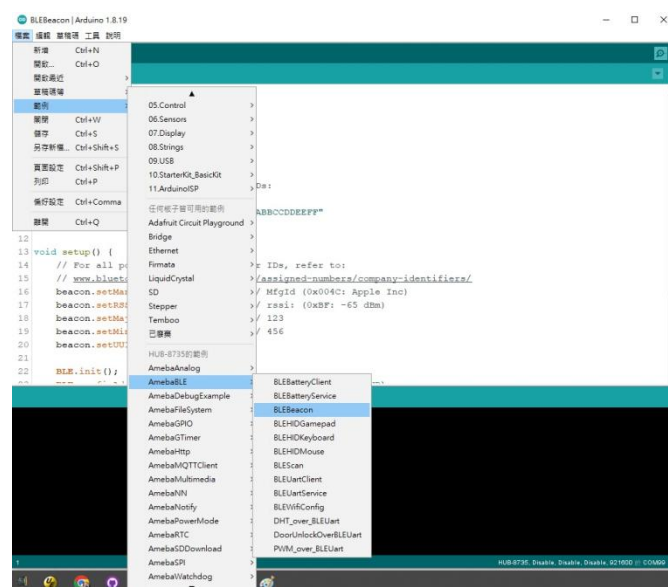


圖 24、開啟 BLE 範例

資料來源：本計畫整理

開啓後可以看到 BLE Beacon 初始化相關設定。

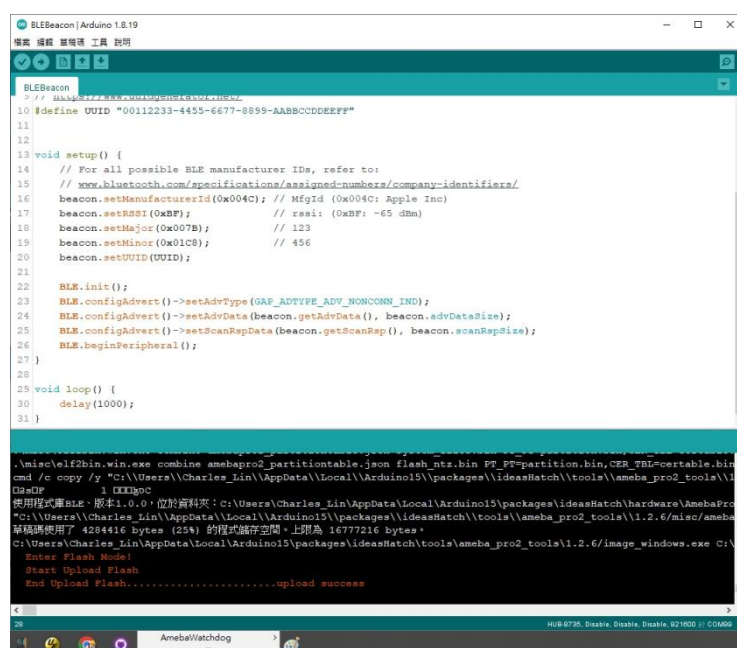


圖 25、Source code

資料來源：本計畫整理

燒錄到 HUB 8735 ultra 後，重新啟動即可搜尋到 BLE 資料。

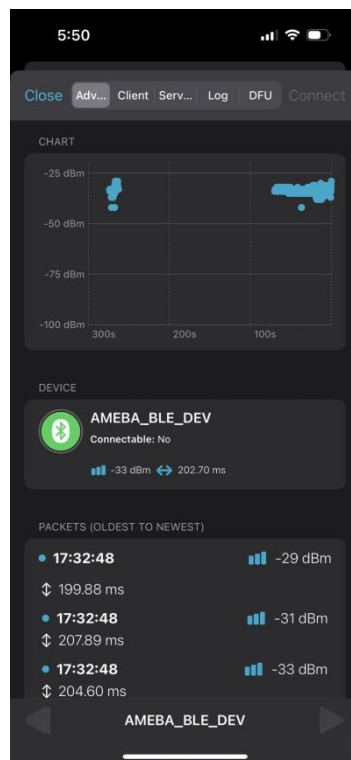


圖 26、手機搜尋畫面

資料來源：本計畫整理

七、I2C Example

(一) 編譯並上傳

如何啟用 I2C 功能，可以從 Arduino Ameba_Wire 範本中找到相關範例，以 LCD_HelloWorld 為範例。

打開“File”->“Examples”->“AmebaWire”->“LCD_HelloWorld”。此範例使用的是 I2C 介面，如果需要更改為 I2C1，則須在 setup 修改 Wire 成 Wire1，相關文件參考在 Wire.cpp 及 Wire.h。

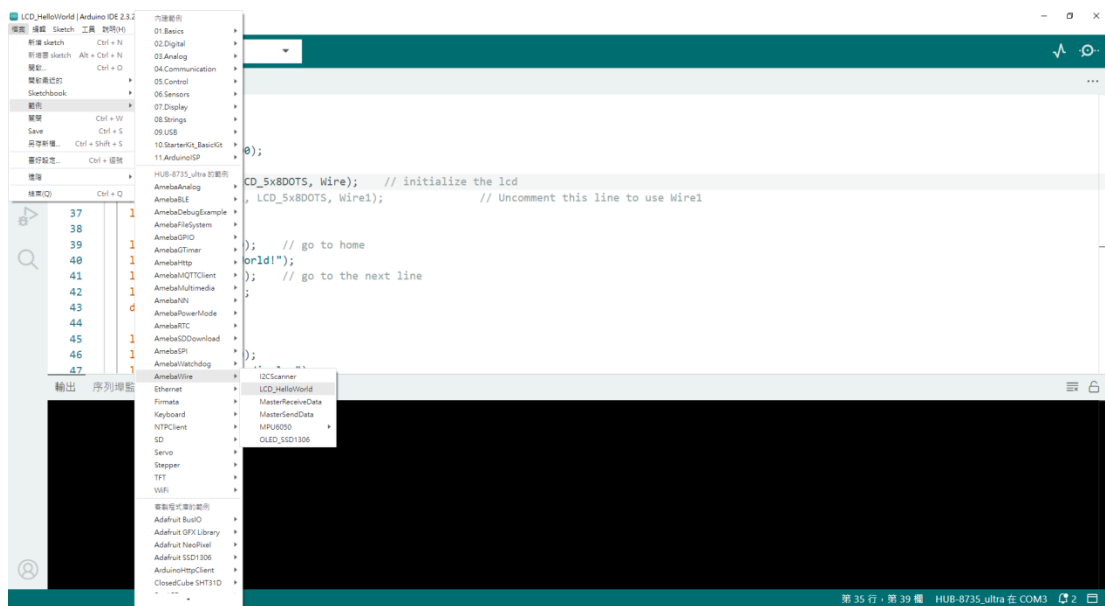


圖 27、開啟 I2C 範例

資料來源：本計畫整理

開啟範例後可以看到 lcd 的初始化設定，包含 I2C slave address 等等。

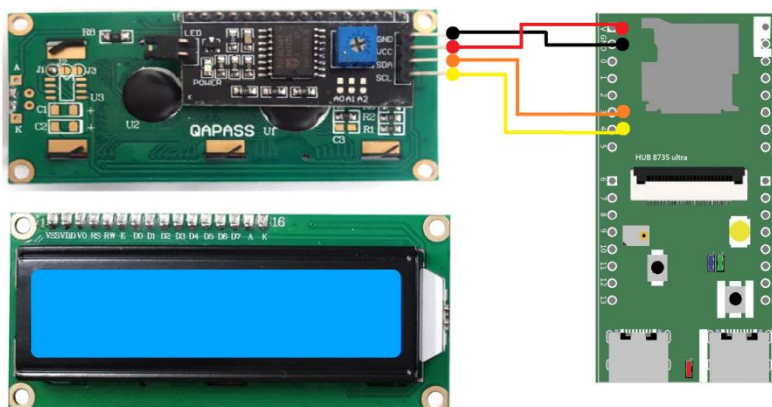


圖 28、HUB 8735 ultra 與 LCD 1602 I2C 接線圖

資料來源：本計畫整理

燒錄完成後，按壓 Reset 按鈕。螢幕上 LCD 第一行出現“Hello World”，第二行出現“Ameba”。

八、SPI Example

(一) 編譯並上傳

如何啓用 SPI 功能，可以從 Arduino Ameba_SPI 範本中找到相關範例，以 LCD_Screen_ILI9341_TFT 為範例。

打開 “File” -> “Examples” -> “AmebaSPI” -> “LCD_Screen_ILI9341_TFT”。

此範例使用的是 SPI0 介面，如果需要更改為 SPI1，則須修改 cpp file 中的 SPI 宣告。開啓後可以看到 SPI 及 TFT LCD 腳位的初始化相關設定。

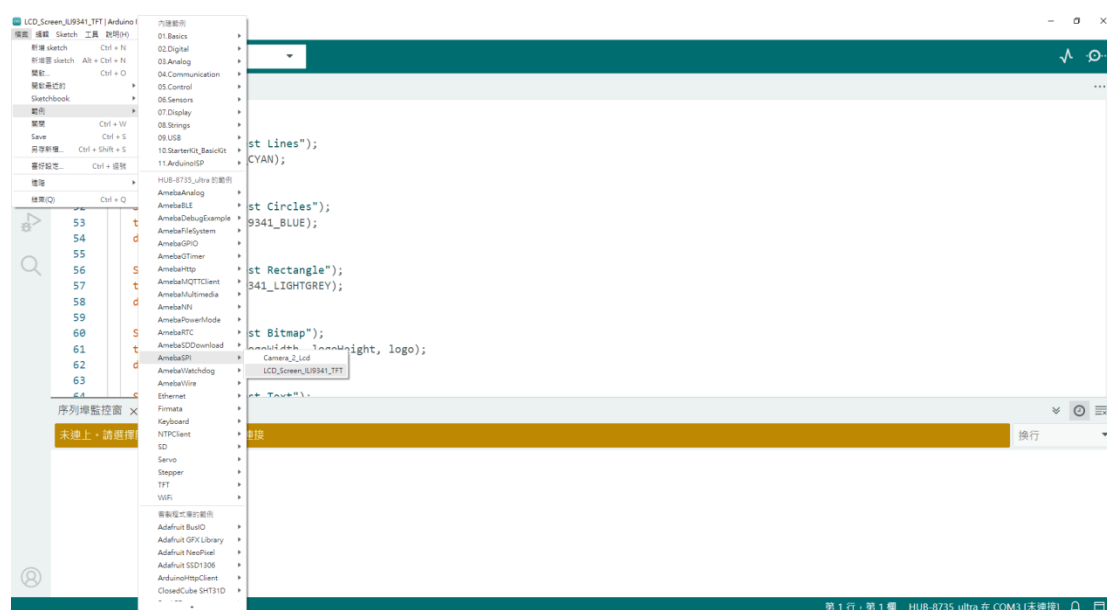


圖 29、開啟 SPI 範例

資料來源：本計畫整理

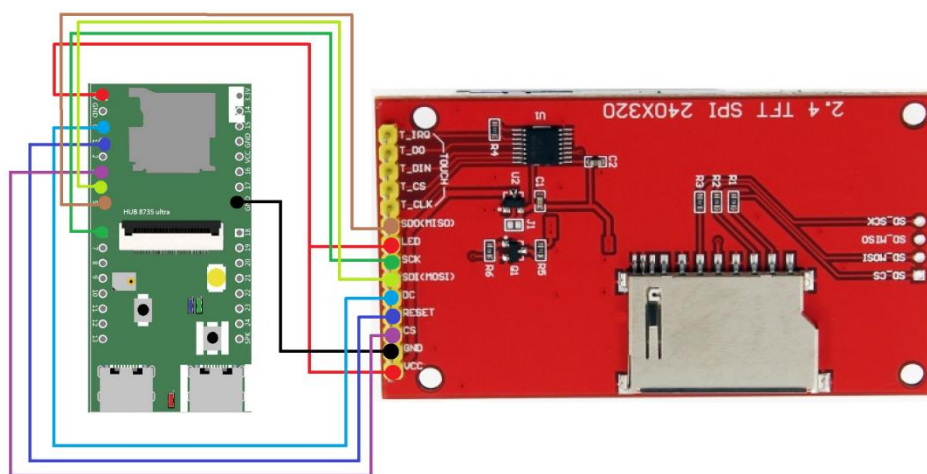


圖 30、HUB 8735 ultra 與 ILI9341_TFT 接線圖

資料來源：本計畫整理

表 2、ILI9341_TFT pin 定義

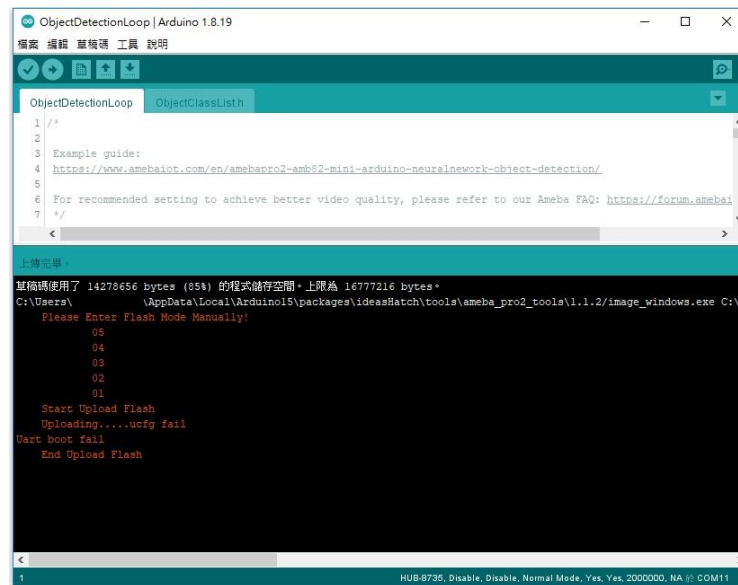
Number	Pin Label	Description
1	VCC	5V/3.3V power input
2	GND	Ground
3	CS	LCD chip select signal, low level enable
4	RESET	LCD reset signal, low level reset
5	DC/RS	LCD register / data selection signal, high level: register, low level: data
6	SDI(MOSI)	SPI bus write data signal
7	SCK	SPI bus clock signal
8	LED	Backlight control, high level lighting, if not controlled, connect 3.3V always bright
9	SDO(MISO)	SPI bus read data signal, if you do not need to the read function, you can not connect it
(The following is the touch screen signal line wiring, if you do not need to touch function or the module itself does not have touch function, you can not connect them)		
10	T_CLK	Touch SPI bus clock signal
11	T_CS	Touch screen chip select signal, low level enable
12	T_DIN	Touch SPI bus input
13	T_DO	Touch SPI bus output
14	T_IRQ	Touch screen interrupt signal, low level when touch is detected

燒錄完成後，按壓 **Reset** 按鈕。螢幕上會跑一些測試，如刷新螢幕顏色，畫各種線條，圓圈，矩形，且顯示不同大小的文字等等。

九、例外處理

例外處理指在開發編譯上，有可能遇到一些編譯失敗等錯誤問題，提供發生現象跟處理方法。

(一) UCFG Fail



十、參考資料

- 物聯網智造基地 - 國產 IC 智造工具包

https://www.ideas-hatch.com/mem_evb.jsp

十一、版本紀錄

時間	版本	備註
2024/05/08	V1.0	[HUB 8735 ultra] Arduino 使用說明 初版