

聯發科  
Filogic130A (MT7933)  
開發板教學

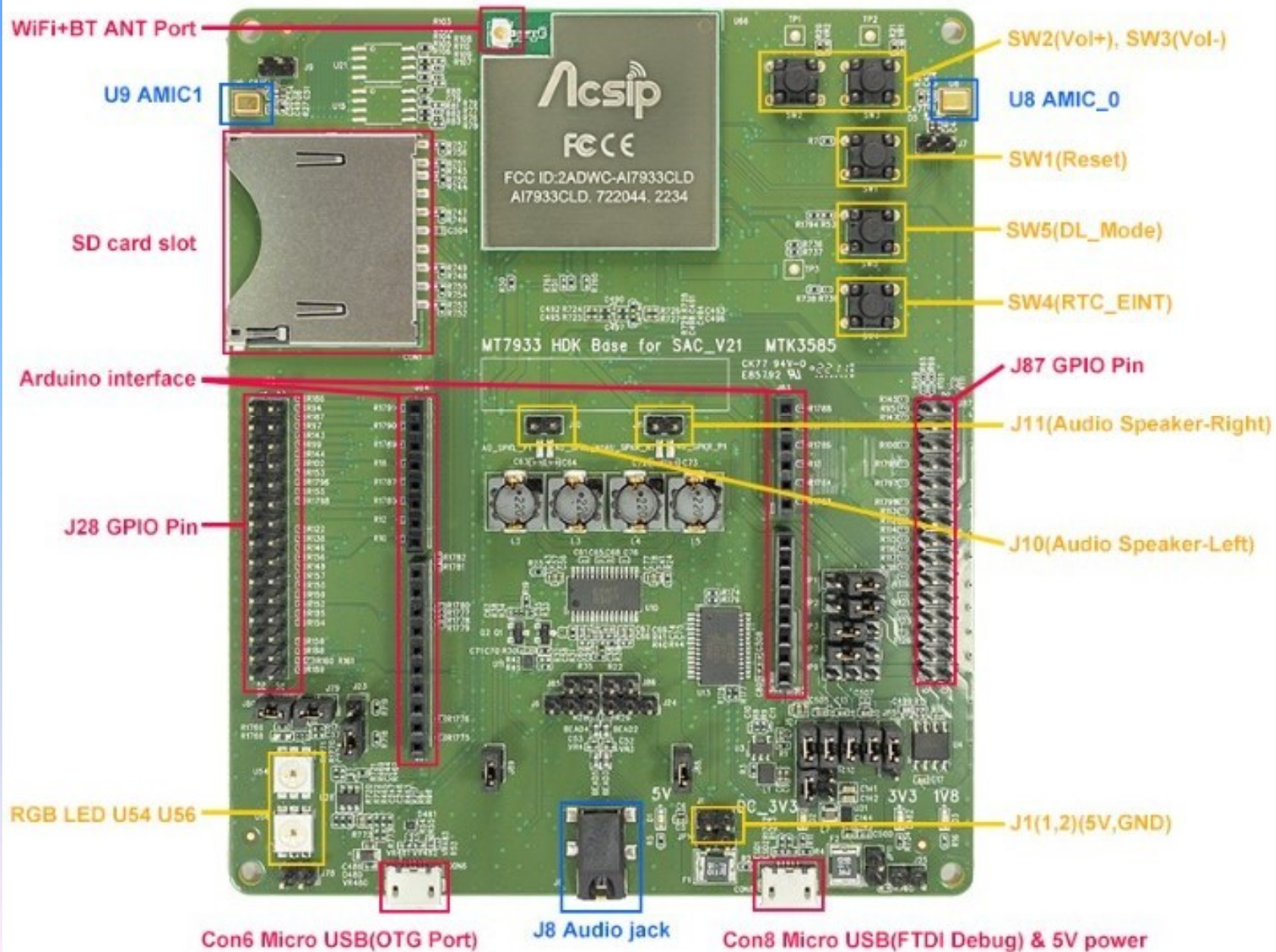
主講人：盧怡仁 (Peter Lu)

YT: Peter Lu @漢亞科技  
<https://han-ya.blogspot.com>

# 大綱

- 開發板 Filogic130A 介紹
- Arduino SDK 安裝與韌體上傳
- 周邊介面與範例說明
  - 溫溼度感測模組 (I<sup>2</sup>C)
  - 顯示模組LCD與OLED (SPI)
  - 播放SD卡音檔 (Audio / SDIO)
  - 網路雲端數據 (WiFi / TCP / UDP)
  - 藍芽BLE通訊 (BT)

# 開發板硬體外觀



# MT7933的周邊介面

- **UART x 3**
- **I<sup>2</sup>C x 2**
- **SPI / SDIO**
- **ADC x 12 (輸入範圍0~1.8V)**
- **PWM x 12**

**MCU**

- **Wi-Fi 6.0 (2.4/5GHz)**
- **BT 5.0**

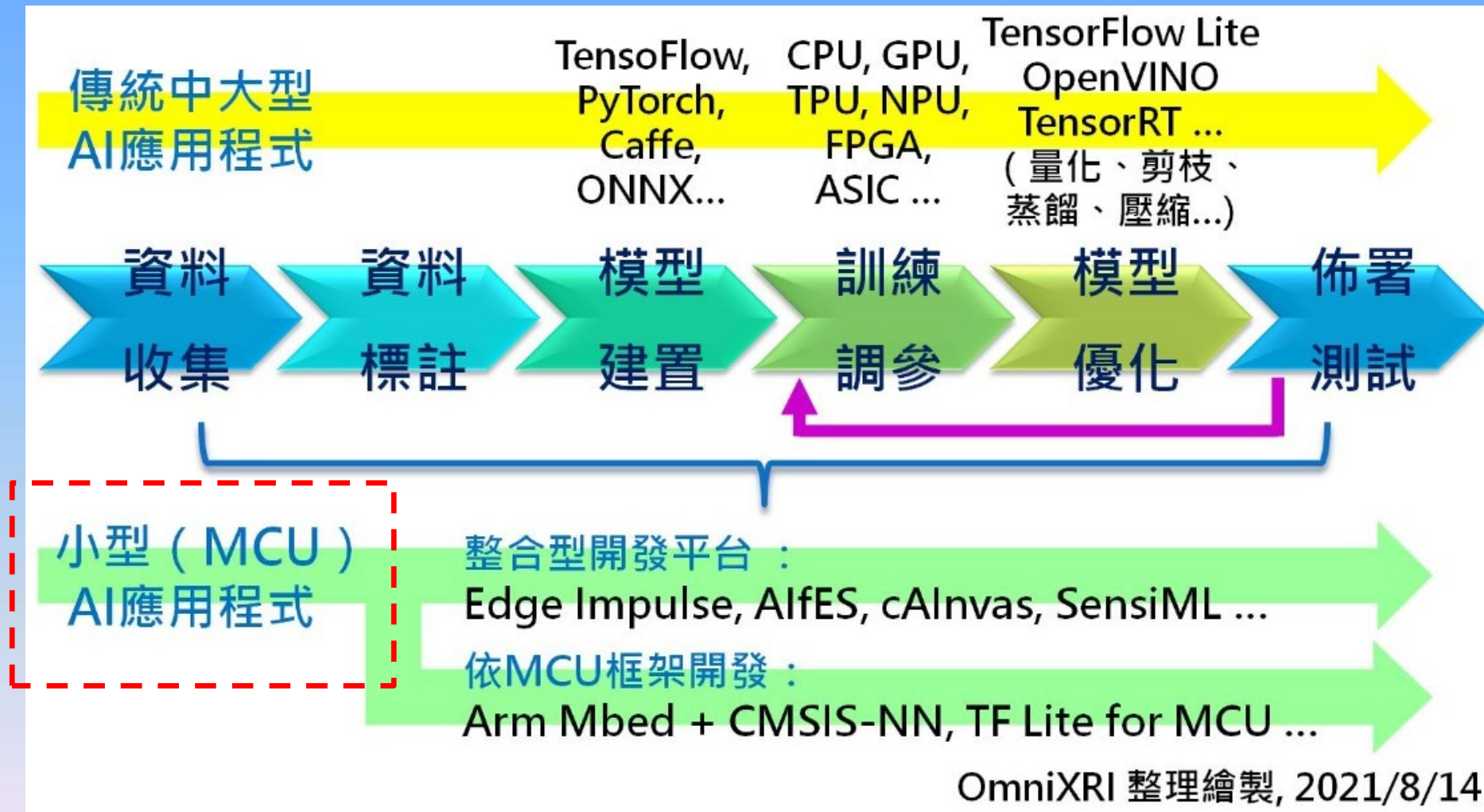
**無線連網**

- **Stereo speaker / line out**
- **Microphone x 2**

**音訊**

- **系統單晶片 SoC**
- **應用**
  - 智慧音箱
  - 數據收集
  - TinyML
  - AIOT
  - Edge AI

# AI + MCU 邊緣運算





# WiFi + BT 的無線模組

- WiFi + BT 整合的單晶片 SoC，且能支援 Arduino 環境
- 比較三款模組：聯發科、瑞昱、樂鑫 (詳細規格依官網為主)

	Fillogic 130A	RTL8722	ESP32
CPU	M33 + HiFi4 DSP	M33 + M23	Xtensa 雙核心
最高內頻	300 MHz	300 MHz	240 MHz
Flash	16 MB	4 MB	2-4 MB
SRAM	1 MB	512 KB	512 KB
WiFi	2.4G / 5G	2.4G / 5G	2.4G
BT	BLE 5.0	BLE 5.0	BLE 4.2 (新系列 可支援 5.0)

# Filogic SDK 下載

- 大聯大的大大通網站 [www.wpgdadatong.com](http://www.wpgdadatong.com)
  - 技術論壇：提問、討論、文件、技術分享...等。
- 建立開發環境
  - **IOT SDK** (Linux編譯環境)：基於Ubuntu v20.04
  - **Arduino SDK**：Arduino IDE v1.6.13以上



**MEDIATEK**

## Filogic IoT SDK

版本號碼 / Version : 1.1.3  
版本日期 / Release : 2022/2/17  
適用範圍 / For product : Filogic 130/130A

↓ 下載 / download



**ARDUINO**

## Filogic IoT Arduino SDK

版本號碼/Version: 1.0.0  
版本日期/Release: 2022/07/04  
適用範圍/For Product: Filogic 130/13

↓ 下載 / download

# 如何建立 Linux 編譯環境

- 參考YouTube影片介紹 <https://youtu.be/3JwzQ6YpsY0>





# Arduino IDE 編譯環境

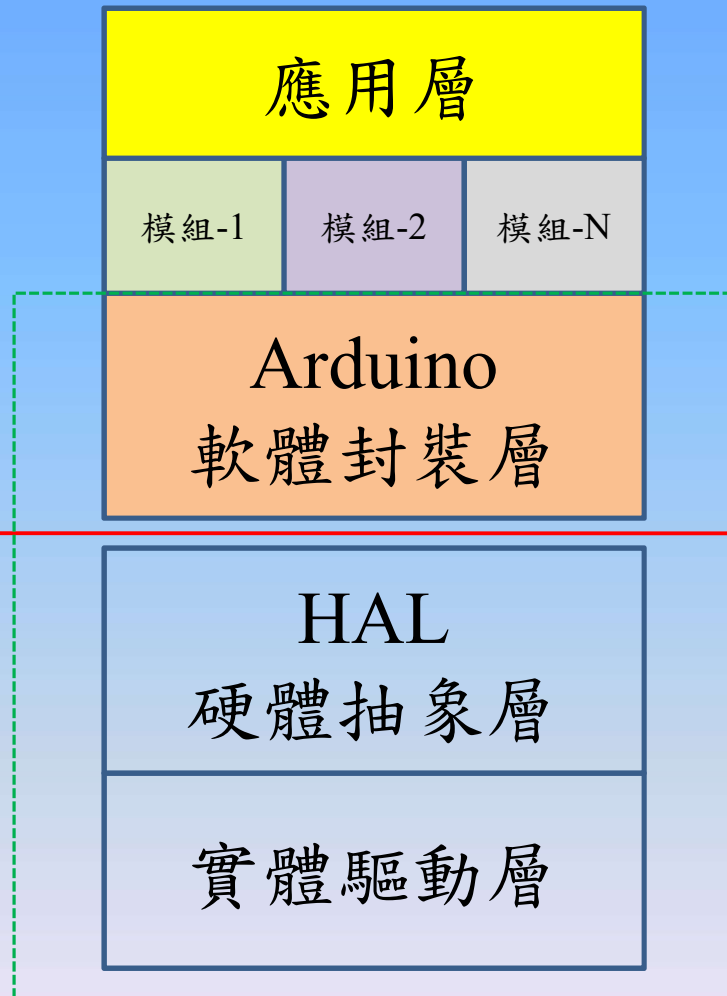
- 優點

- 快速入手，編輯介面簡潔易懂
- 建立統一的編譯環境
- ✓ 在軟體架構下，更多市面的模組能支援各類開發板(下頁的比較)
- 網路資源豐富，開發過程更為便利與快速

- 操作前的準備

- 安裝Arduino IDE v1.6.13以上，勿用v2.0
- 下載Filologic Arduino SDK (v1.0.0 ~ 1.2.0)
- 安裝Tomcat 8.5以上 (~~http網路服務器~~)

# Arduino 架構



**Arduino SDK環境開發架構**



**Linux SDK環境開發架構**

# 安裝步驟說明 (1)

- 參考YouTube影片介紹 <https://youtu.be/oZ9UFsgVlXw>



# Filogic SDK 下載

- Filogic130 **Arduino SDK**的各個版本

- ✓從資策會雲端硬碟下載

- <https://drive.google.com/drive/folders/1z7HscksgsalRfHW7T49nVbI60gY-weV7>

- ✓依序下載後(解壓縮)，並放到Tomcat目錄底下

- ✓在SDK v1.0.0裡面，提供開發板的電路圖

- ✓文件與範例程式碼：

- [https://github.com/yijenlu1971/Filogic onArduino](https://github.com/yijenlu1971/Filogic_onArduino)



Q：在 Arduino IDE 環境下，若發生無法上傳韌體時，上傳過程出現錯誤訊息。

✓ Bootloader 開機程式：

板子出廠時所預載的 bootloader 版本是給 Linux SDK 使用，如果要在 Arduino IDE 開發必須重新燒錄 Arduino SDK 版本的 bootloader，連結如下：

[https://www.ideashatch.com/filogic\\_130/arduino\\_bootloader\\_and\\_flash\\_tool.zip](https://www.ideashatch.com/filogic_130/arduino_bootloader_and_flash_tool.zip)

## 安裝步驟說明 (2)

- 安裝tomcat後，SDK檔案放到 \webapps\ROOT目錄下
- 運行tomcat  
\\bin\\Tomcat8.exe

Apache Software Foundation > Tomcat 8.5 > webapps > ROOT

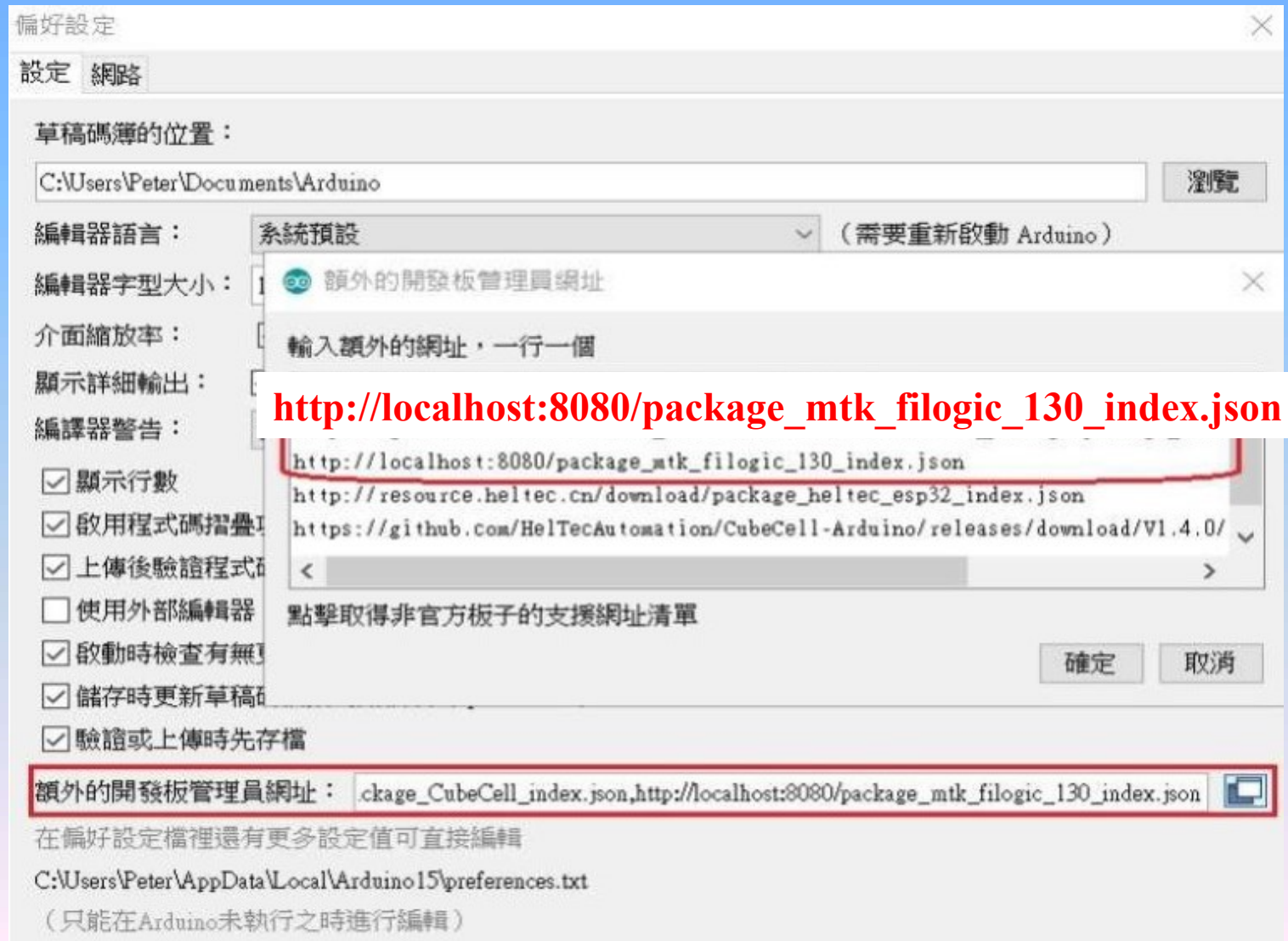
名稱

修改日期

WEB-INF	2022/7/9 上午 6:16
asf-logo-wide.svg	2021/7/1 上午 2:00
bg-button.png	2021/7/1 上午 2:00
bg-middle.png	2021/7/1 上午 2:00
bg-nav.png	2021/7/1 上午 2:00
bg-upper.png	2021/7/1 上午 2:00
favicon.ico	2021/7/1 上午 2:00
filogic_130_toolchain-1.0.0.tar.gz	2022/7/6 上午 10:01
index.jsp	2021/7/1 上午 2:00
mediatek_filogic_130_flash_tool-1.0.0.tar.bz2	2022/7/6 上午 10:01
mediatek_filogic_130_flash_tool-1.1.0.tar.bz2	2022/8/31 下午 10:29
mediatek_filogic_130-v1.0.0.tar.bz2	2022/7/6 上午 10:12
mediatek_filogic_130-v1.1.0.tar.bz2	2022/8/5 下午 5:20
mediatek_filogic_130-v1.2.0.tar.bz2	2022/8/31 下午 10:29
package_mtk_filogic_130_index.json	2022/9/2 下午 5:04

# 安裝步驟說明 (3)

- 安裝IDE後，點選「偏好設定」，並設定SDK下載網址



設定SDK下載網址，可改至ideas-hatch

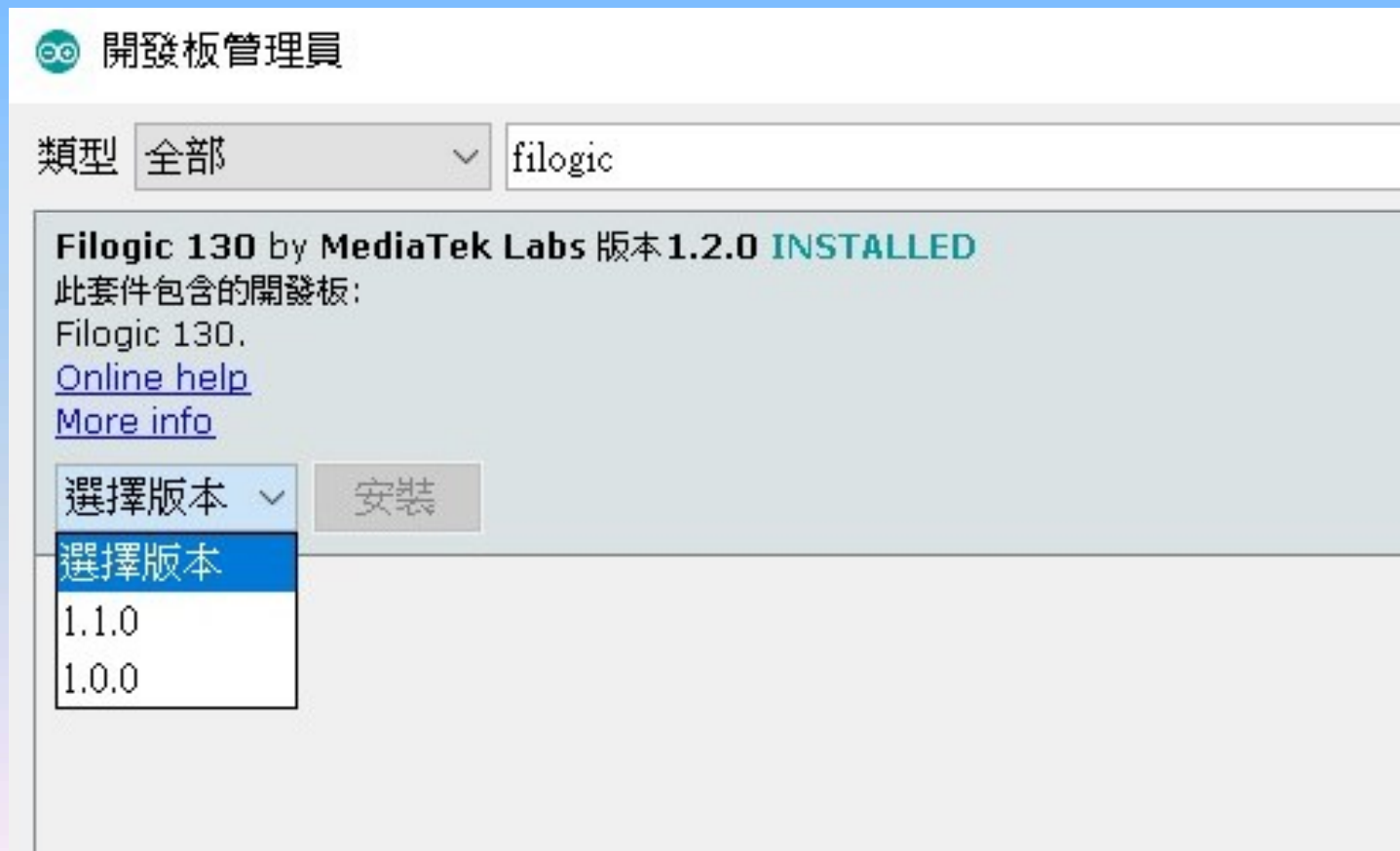
[https://www.ideas-hatch.com/filologic\\_130/arduino\\_sdk/package\\_mtk\\_filologic\\_130\\_index.json](https://www.ideas-hatch.com/filologic_130/arduino_sdk/package_mtk_filologic_130_index.json)





# 安裝步驟說明 (4)

- 開啟IDE的“開發板管理員”
- 安裝 Filogic 130開發板的SDK



# 安裝步驟說明 (5)

## ➤ 下載測試程式 GPIO

[https://github.com/yijenlu1971/Filologic\\_onArduino](https://github.com/yijenlu1971/Filologic_onArduino)

## ➤ 編譯程式，並上傳韌體。

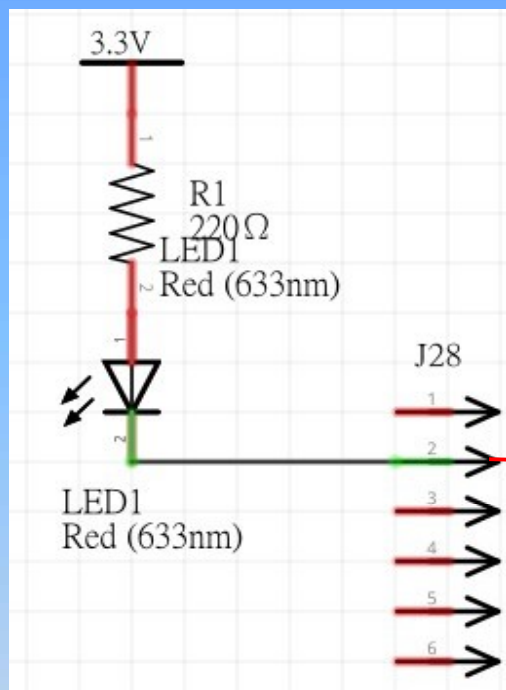
## ➤ 留意上傳過程：

- 按著 SW1 Reset 鍵，當看到訊息 “**INFO: Goto open COM**” 的時候，再放開SW1鍵。.....開始上傳，直到看到訊息 “**Finished!**”。
- 再按一次 SW1 鍵 (Reset)，重啟系統。

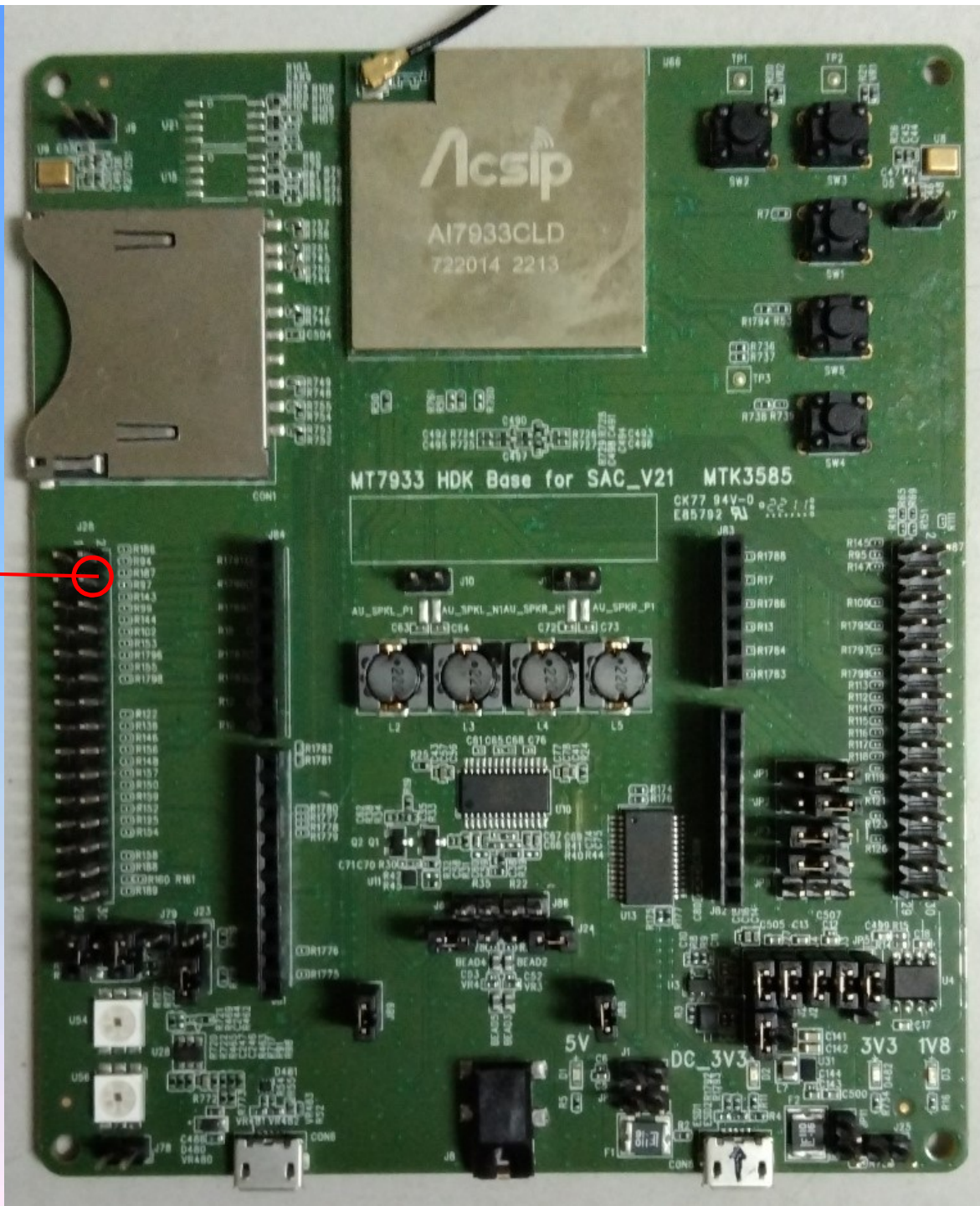
## ➤ 當系統運行後：

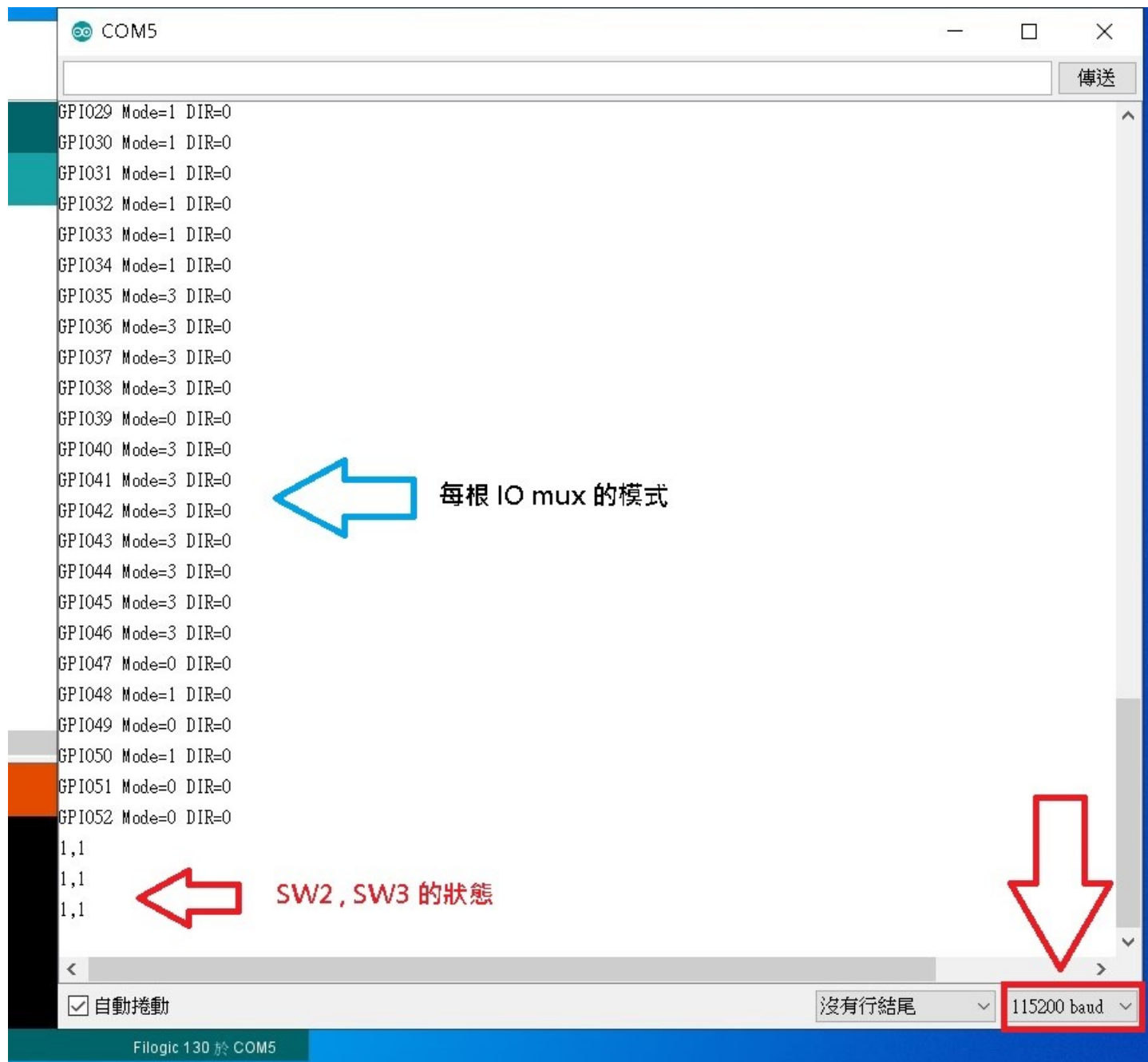
- 從 serial port 視窗能看到訊息輸出，代表 GPIO 程式上傳成功，且正在運行中，如下圖所示。
- 按一按板子上的 SW2 或 SW3 鈕，從訊息能看出狀態變化。
- 接上一顆LED，如下圖，燈號會閃爍。

**3.3V**電源接在板子的  
J87外側 pin-26



**LED**的短腳接在板子的  
J28內側 pin-4







# 範例說明：IO mux

- MT7933晶片的每一腳位具有多項功能，透過 IO mux 方式來切換pin腳的功能。
- 範例程式GPIO，調用底層的HAL函數，並列出53根 IO 腳的「當下」功能。
- 下圖：對照開發板電路圖與 IO 腳位的對應關係。

```
GPIO6 Mode=3 DIR=1
GPIO7 Mode=3 DIR=1
GPIO8 Mode=3 DIR=0
GPIO9 Mode=3 DIR=0
GPIO10 Mode=3 DIR=0
GPIO11 Mode=3 DIR=0
GPIO12 Mode=6 DIR=0
GPIO13 Mode=3 DIR=0
GPIO14 Mode=3 DIR=0
GPIO15 Mode=3 DIR=0
GPIO16 Mode=6 DIR=0
GPIO17 Mode=3 DIR=0
GPIO18 Mode=3 DIR=0
GPIO19 Mode=4 DIR=0
GPIO20 Mode=4 DIR=0
GPIO21 Mode=4 DIR=0
GPIO22 Mode=4 DIR=0
GPIO23 Mode=4 DIR=0
GPIO24 Mode=4 DIR=0
GPIO25 Mode=3 DIR=0
GPIO26 Mode=3 DIR=0
```

當下的  
功能模式

```
MT7933_PIN_6_FUNC_SPI0_SCK
MT7933_PIN_7_FUNC_SPI0_CS_N
MT7933_PIN_8_FUNC_SPI0_MISO
MT7933_PIN_9_FUNC_SPI0_MOSI
MT7933_PIN_10_FUNC_I2SIN_DAT0
MT7933_PIN_11_FUNC_I2SO_DAT0
MT7933_PIN_12_FUNC_ANT_SEL3
MT7933_PIN_13_FUNC_I2SO_BCK
MT7933_PIN_14_FUNC_I2SO_LRCK
MT7933_PIN_15_FUNC_I2SO_MCK
MT7933_PIN_16_FUNC_ANT_SEL7
MT7933_PIN_17_FUNC_TDMIN_MCLK
MT7933_PIN_18_FUNC_TDMIN_BCK
MT7933_PIN_19_FUNC_I2SIN_LRCK
MT7933_PIN_20_FUNC_I2SIN_BCK
MT7933_PIN_21_FUNC_DMIC_DAT1
MT7933_PIN_22_FUNC_DMIC_DAT2
MT7933_PIN_23_FUNC_DMIC_DAT3
MT7933_PIN_24_FUNC_DMIC_CLK1
MT7933_PIN_25_FUNC_UART0_RX
MT7933_PIN_26_FUNC_UART0_TX
```

```
(C_SDIO_CLK)
(C_SDIO_CMD)
(C_SDIO_DAT0)
(C_SDIO_DAT1)
(C_SDIO_DAT2)
(C_SDIO_DAT3)
(GPIO_B_1)
(GPIO_B_2)
(GPIO_B_3)
(GPIO_B_4)
(GPIO_B_5)
(GPIO_B_6)
(GPIO_B_7)
(GPIO_B_8)
(GPIO_B_9)
(GPIO_B_10)
(GPIO_B_11)
(GPIO_B_12)
(GPIO_B_13) : HAL_UART_1
(GPIO_B_14) : HAL_UART_1
```

電路圖  
標示的  
名稱

✓ IO mux概念：

舉某IC晶片的腳位資料為例，PA4腳位功能，預設功能為SPI1\_NSS 或 UART2\_CK 或 ADC12\_IN4 (三功能選一)。

開機通電後，先要init GPIO腳位的功能(設定相關的暫存器)

脚位					管脚名称	类型 (1)	I/O电平(2)	主功能 (复位后) (3)	可选功能	
BGA100	LQFP48	LQFP64	LQFP100	VFQFPN36					默认功能	重定义功能
G3	14	20	29	11	PA4	I/O		PA4	SPI1_NSS(7)/US ART2_CK(7)/ADC 12_IN4	
H3	15	21	30	12	PA5	I/O		PA5	SPI1_SCK(7)/AD C12_IN5	
J3	16	22	31	13	PA6	I/O		PA6	SPI1_MISO(7)/A DC12_IN6/TIM3_ CH1(7)	TIM1_BKIN
K3	17	23	32	14	PA7	I/O		PA7	SPI1_MOSI(7)/A DC12_IN7/TIM3_ CH2(7)	TIM1_CHIN
G4	-	24	33	-	PC4	I/O		PC4	ADC12_IN14	
H4	-	25	34	-	PC5	I/O		PC5	ADC12_IN15	
J4	18	26	35	15	PB0	I/O		PB0	ADC12_IN8/TIM3_ CH3(7)	TIM1_CH2N
K4	19	27	36	16	PB1	I/O		PB1	ADC12_IN9/TIM3_ CH4(7)	TIM1_CH3N

# 範例說明：ADC

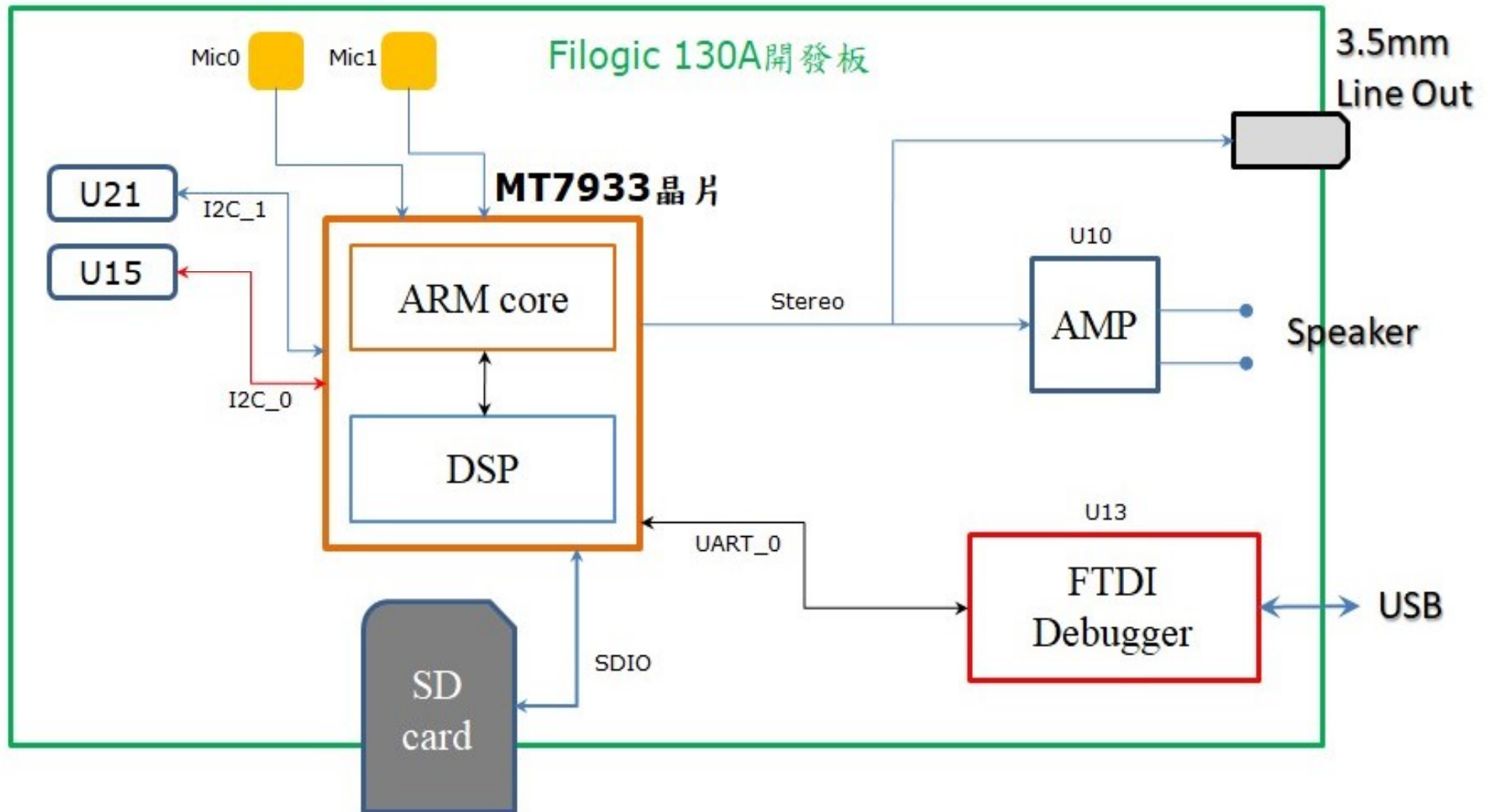
- 應用：
  - 氣壓、流體速度...等感測元件
  - 荷重元(load cell)：測拉力、秤重...
  - 磁感應(hall sensor)：搖桿軸、測電流...
- 規格：
  - 提供 **12** 組ADC channel，取樣率2M Samples/sec，解析度**12-bit**，量測電壓範圍**0~1.8V**。
- 範例程式 **ADC**
  - 開發板上**J83**單排針的前四個為ADC通道0~3。
  - 另外，HAL\_GPIO\_27和HAL\_GPIO\_28則為ADC通道10和11。
  - 呼叫 **analogRead** 函數便能取到ADC數值

# 傳輸介面

- Serial bus
  - **UART** :  $\leq 115.2$  Kbps (開發板的debug: 921.6K)
  - **I<sup>2</sup>C** (Inter-Integrated Circuit) : 100~400 Kbps
    - **SCL**和**SDA** : 兩訊號線
    - 可選 DMA 傳輸
  - **SPI** :  $\leq 50$  MHz
    - **CS** 、**SCK** 、**MISO** 、**MOSI** : 四訊號線
    - 可設定為 Master 或 Slave 模式
    - 可選 DMA 傳輸
  - **SDIO**
    - **CMD** 、**CLK** 、**D0~D3**等六訊號線



# 電路的方塊圖

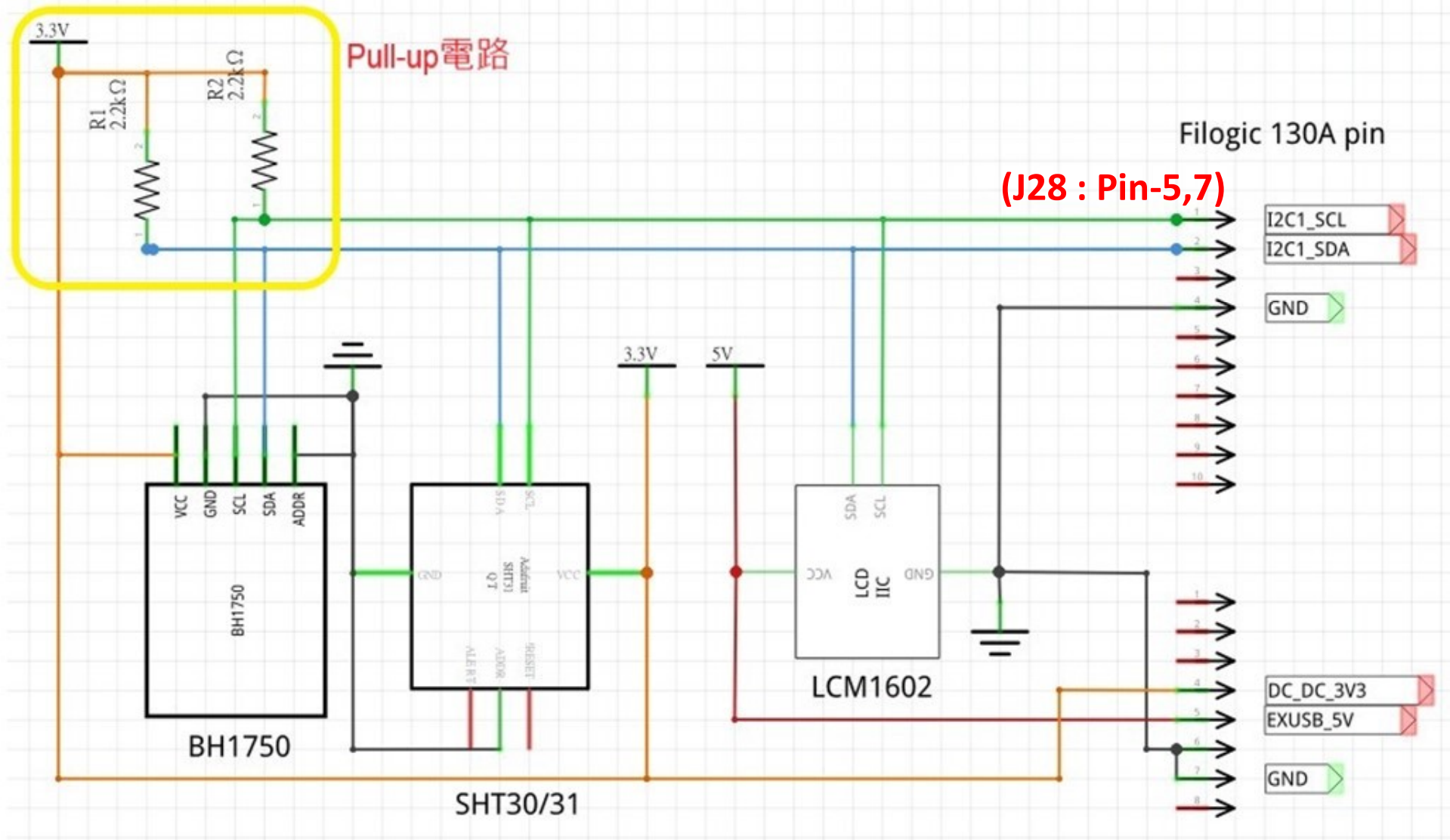


# 通訊介面說明：I<sup>2</sup>C (1)

- MT7933 晶片提供兩組I<sup>2</sup>C介面，可串接各種模組或其他晶片。
- J28雙排針的(1、3) 和(5、7) 對應兩組I<sup>2</sup>C，其中 **Arduino** 軟體架構下只能操作第一組(SDK v1.1.0以上)，而第二組能透過**HAL** (Hardware Abstraction Layer)架構使用。
- 範例中，使用的I<sup>2</sup>C模組為：
  - BH1750 光照度感測 (位址**0x23**)
  - SHT30 溫溼度感測 (位址**0x44**)
  - LCM1602 顯示模組 (位址**0x27**)

# 通訊介面說明：I<sup>2</sup>C (2)

- 串接的電路圖，如下圖。請參考文件說明

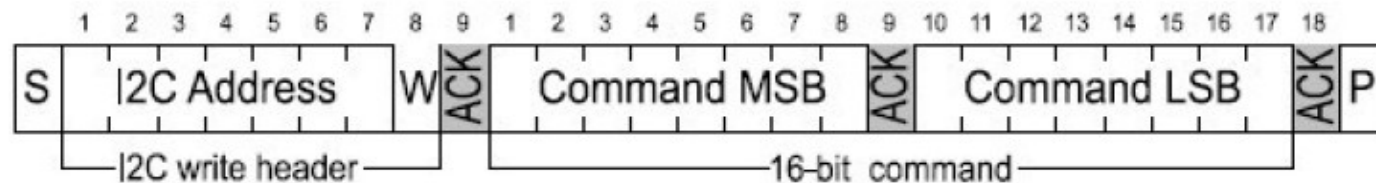


## 通訊介面說明：I<sup>2</sup>C (3)

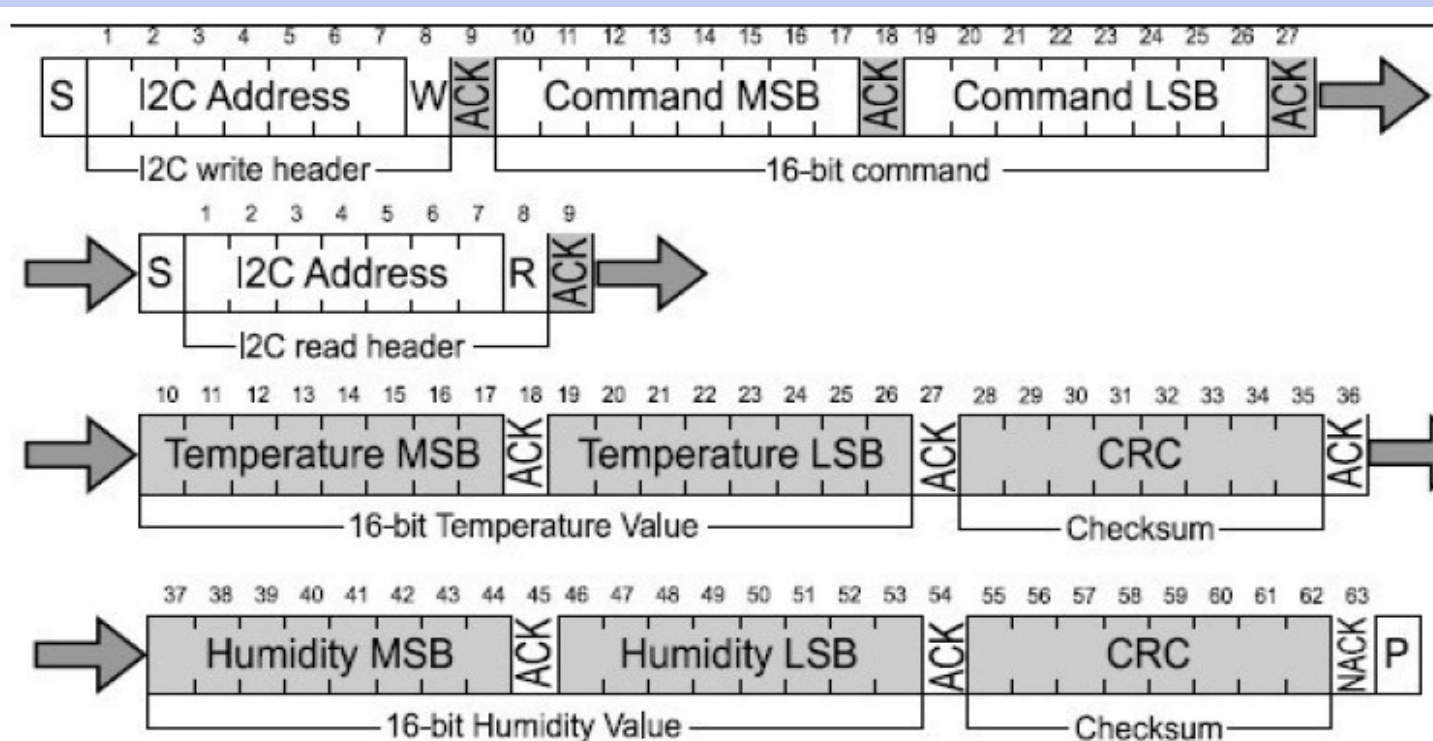
- 串接的電路圖
  - I2C\_SCL 接到開發板的**J28 pin7** (雙排外側)
  - I2C\_SDA 接到開發板的**J28 pin5** (雙排外側)
  - 3.3V電源接到開發板的**J87 pin 26** (或J82 pin4)
  - 5V電源接到開發板的**J87 pin 30** (或J82 pin5)
  - GND接到開發板的**J87 pin 28** (或J82 pin6)
- 載入範例程式
  - 下載GitHub上面的**I2C**
  - 依據HAL架構自行開發的感測器驅動程式

# 通訊介面說明：I<sup>2</sup>C (4)

- 介面的時序圖：寫入和讀取



圖二：master 發送一個寫入的通訊格式 (取自 SHT30 datasheet)



圖三：master 發送一個讀取的通訊格式 (取自 SHT30 datasheet)



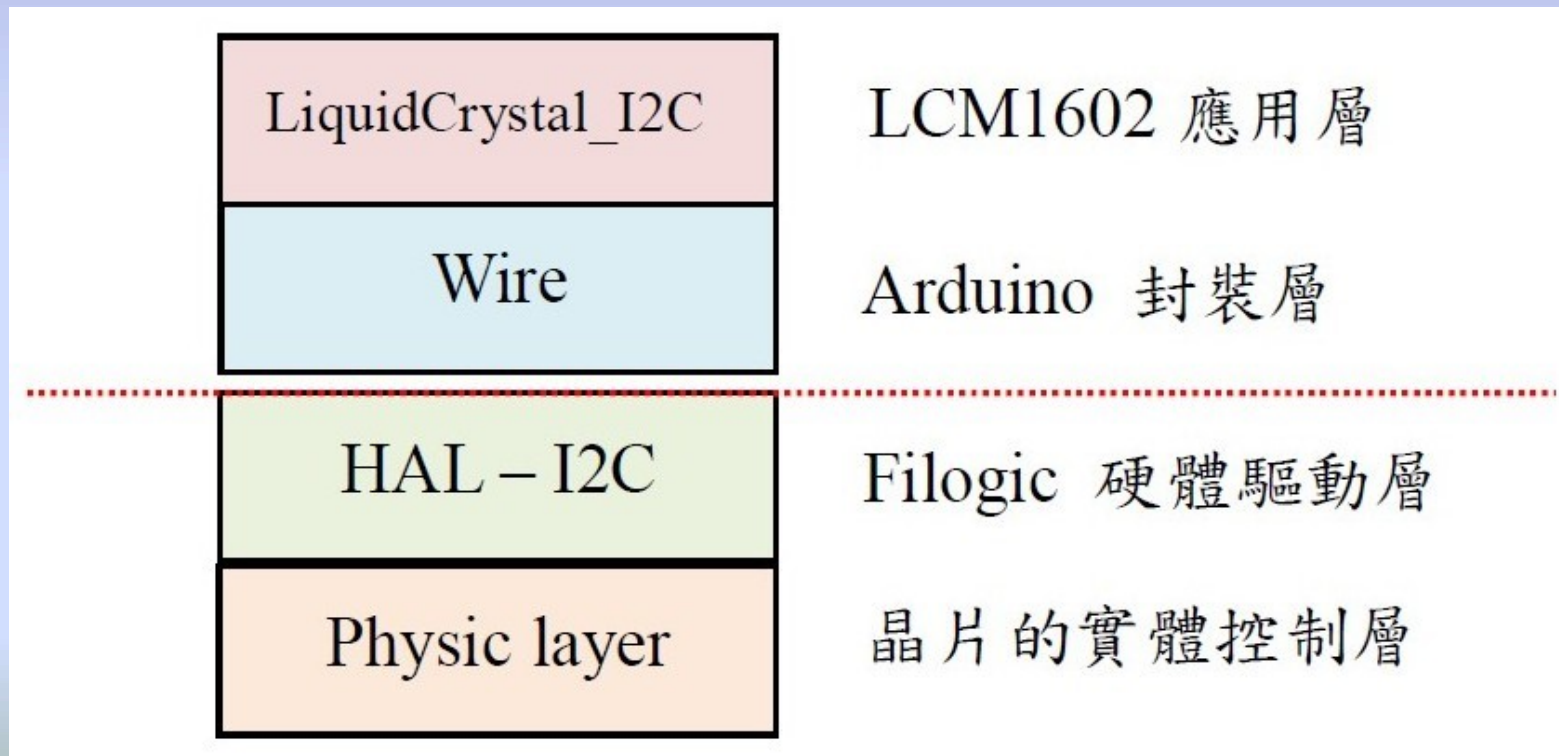
# HAL架構的範例：I<sup>2</sup>C (5)

- 影片連結 <https://youtu.be/UQumrcPdIuA>

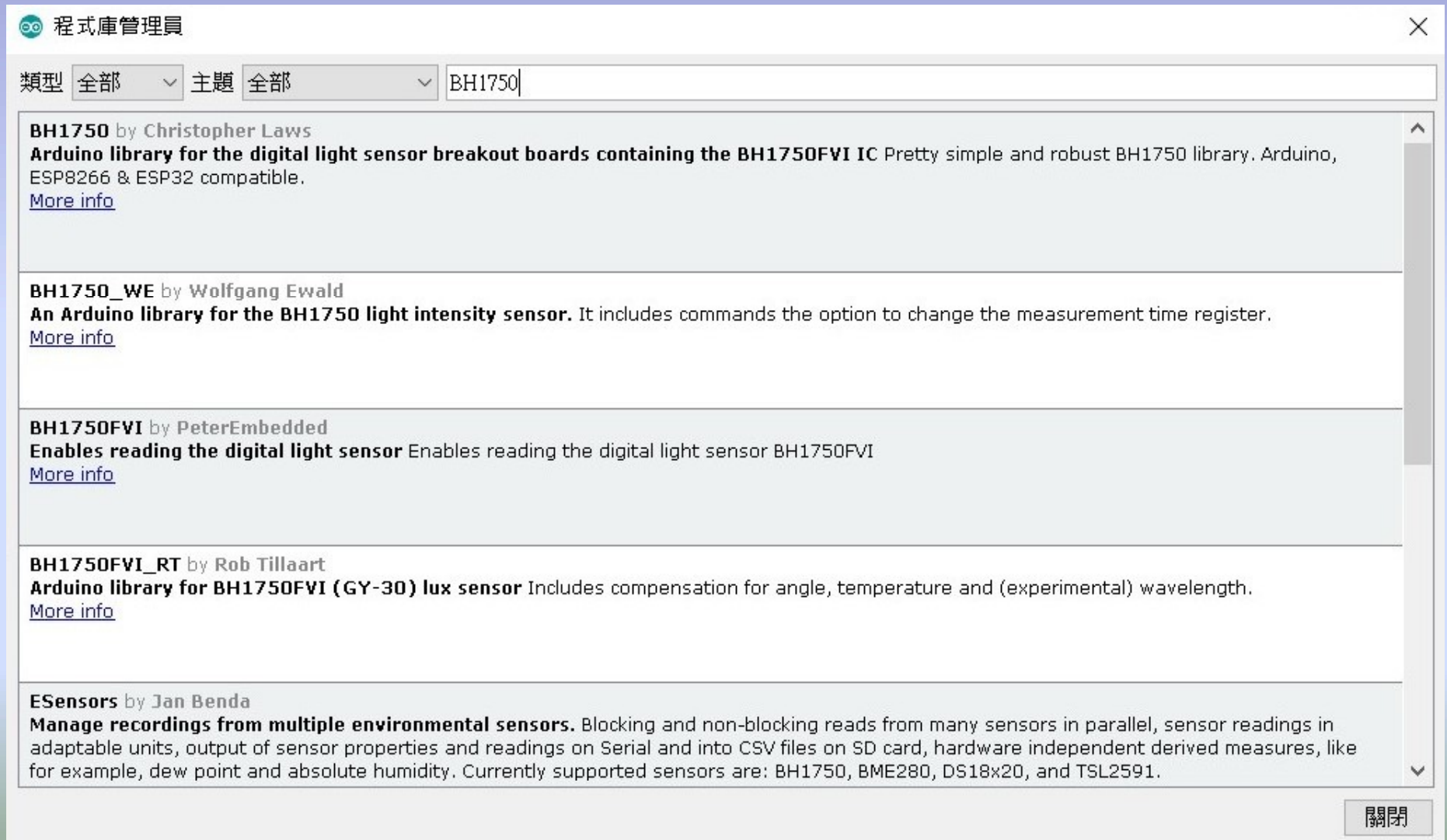


## SDK 架構：I<sup>2</sup>C (6)


- Arduino SDK v1.1.0版本提供完整I<sup>2</sup>C控制介面。原廠提供的範例程式，LCM1602模組。



網路上，各界大神提供模組的驅動程式，我們可以從「程式庫管理員」下載 **BH1750** 感測模組的library。



從「程式庫管理員」下載 SHT30 感測模組的程式庫。

 程式庫管理員

類型  主題

**AirGradient Air Quality Sensor** by AirGradient  
**ESP8266 library for an air quality sensor featuring PM2.5, CO2, Temperature, TVOC and Humidity with OLED display.** Air quality monitoring library supporting the Plantower PMS5003 particle sensor, the Senseair S8 CO2 sensor and the SHT30/31 sensor for humidity and temperature. Kits with all components including a nice enclosure are available in our online shop. You can also connect an OLED display or send the air quality data to the AirGradient platform or any other backend. Optionally you can connect the Sensirion SGP40 TVOC module from AirGradient.  
[More info](#)

**ClosedCube SHT31D** by ClosedCube  
**Arduino library for Sensirion SHT30-D, SHT31-D and SHT35-D Digital I2C Temperature & Humidity Sensors** Arduino library for Sensirion SHT30-D, SHT31-D and SHT35-D Digital I2C Digital I2C Temperature & Humidity Sensors  
[More info](#)

**SHT85** by Rob Tillaart  
**Arduino library for the SHT85, SHT30, SHT31, SHT35 Senserion temperature and humidity sensor**  
[More info](#)

# 通訊介面說明：SPI (1)

- MT7933晶片提供 1 組 SPI 介面，也與 SDIO 介面共用。傳輸速度  $\leq 50$  MHz
- 開發板上面有兩顆**RGB**燈，由SPI介面所控制
  - **J79**和**J80**接通才能連接 RGB 燈，如果想要接上其他模組，必須將這兩個斷開。
  - 範例程式 **RGB\_AdafruitBusIO**，下載 Adafruit 函數庫，模擬 SPI 訊號控制 RGB 燈號的顏色變化。
  - 原廠提供的範例 **LEDWidget\_Blink**，直接呼叫 HAL 函數。



# Arduino程式庫：SPI (2)

- 下載並安裝 Adafruit 程式庫
  - BusIO
  - GFX\_Library
  - ST7735

程式庫管理員

類型  主題

**Adafruit BusIO** by Adafruit 版本1.13.2 **INSTALLED**  
This is a library for abstracting away UART, I2C and SPI interfacing This is a library for abstracting away UART, I2C and SPI interfacing  
[More info](#)

類型  主題

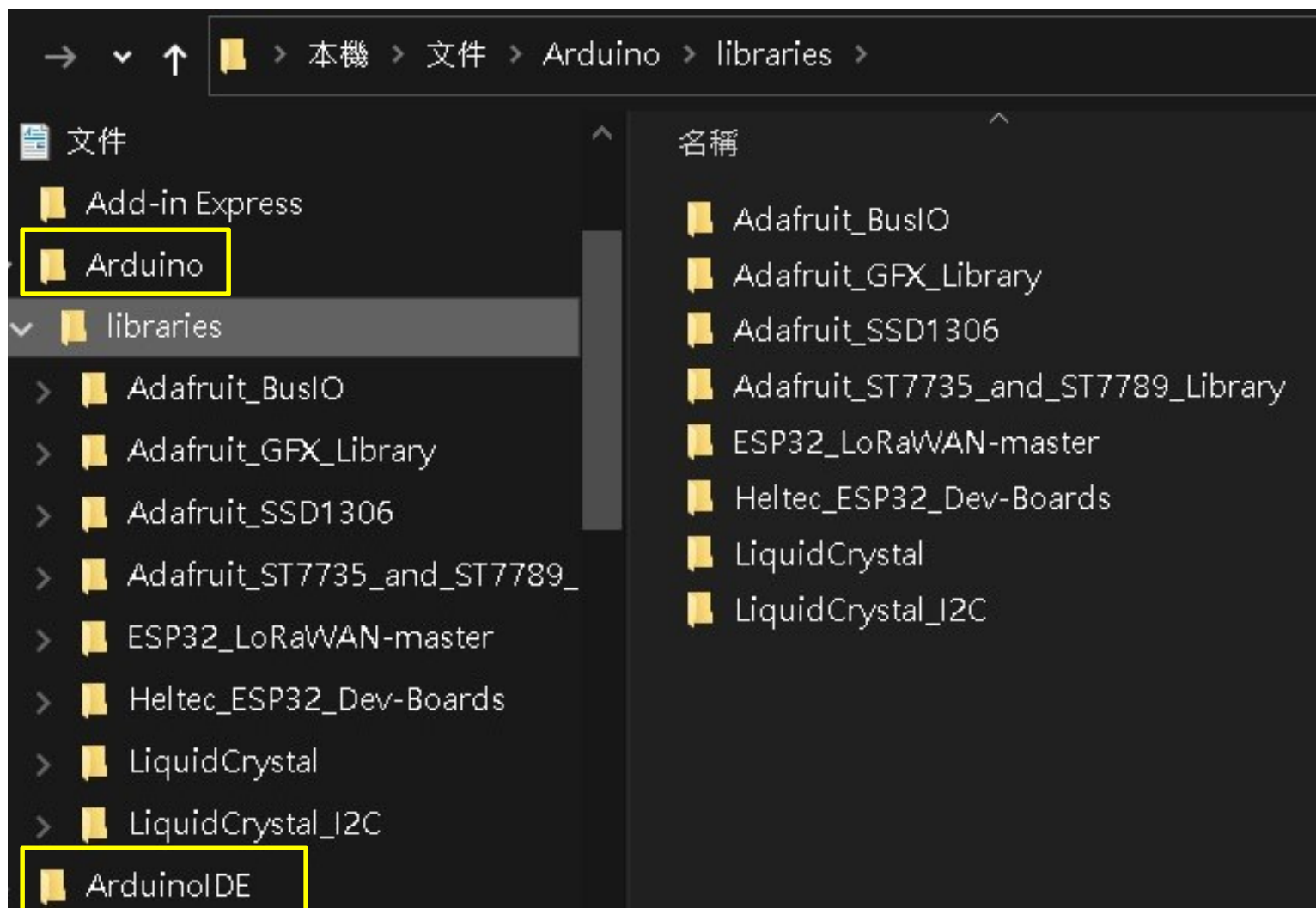
**Adafruit GFX Library** by Adafruit 版本1.11.3 **INSTALLED**  
**Adafruit GFX graphics core library, this is the 'core' class that all our other graphics libraries derive from.** Install this library in addition to the display library for your hardware.  
[More info](#)

類型  主題

**Adafruit ST7735 and ST7789 Library** by Adafruit 版本1.9.3 **INSTALLED**  
**This is a library for the Adafruit ST7735 and ST7789 SPI displays.** This is a library for the Adafruit ST7735 and ST7789 SPI displays.  
[More info](#)

✓程式庫安裝在哪裡？

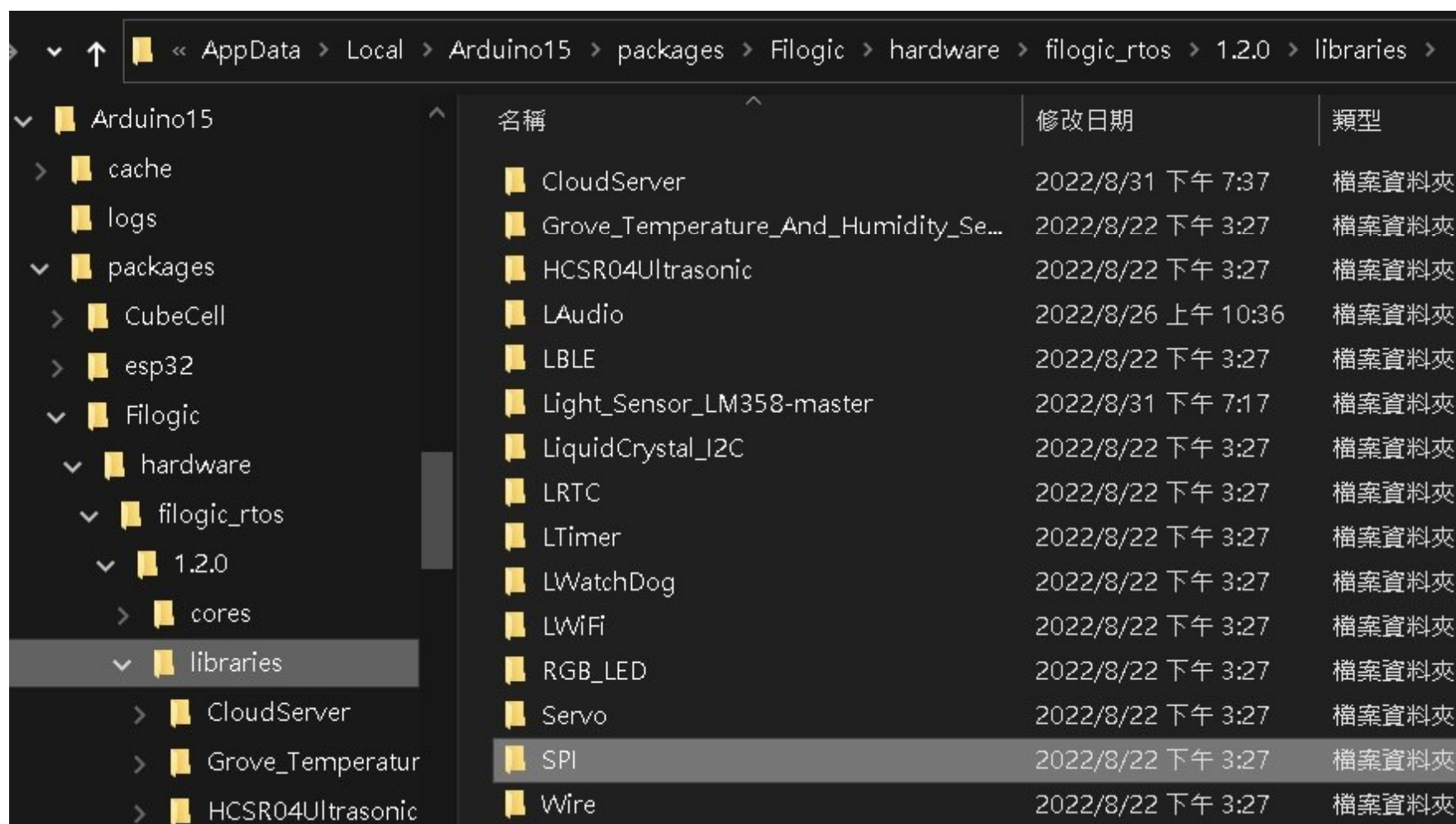
A：在“文件”目錄下的 **Arduino** 或 **ArduinoIDE** (與版本有關)，如下圖所示，libraries底下各個程式庫。



✓編譯 SPI 程式庫過程出現錯誤？

A：下圖的目錄，SPI class裡的函數需要修正，請試著修改。

請參考Github上面的**Filogic\_update**目錄裡是修正後的函數。

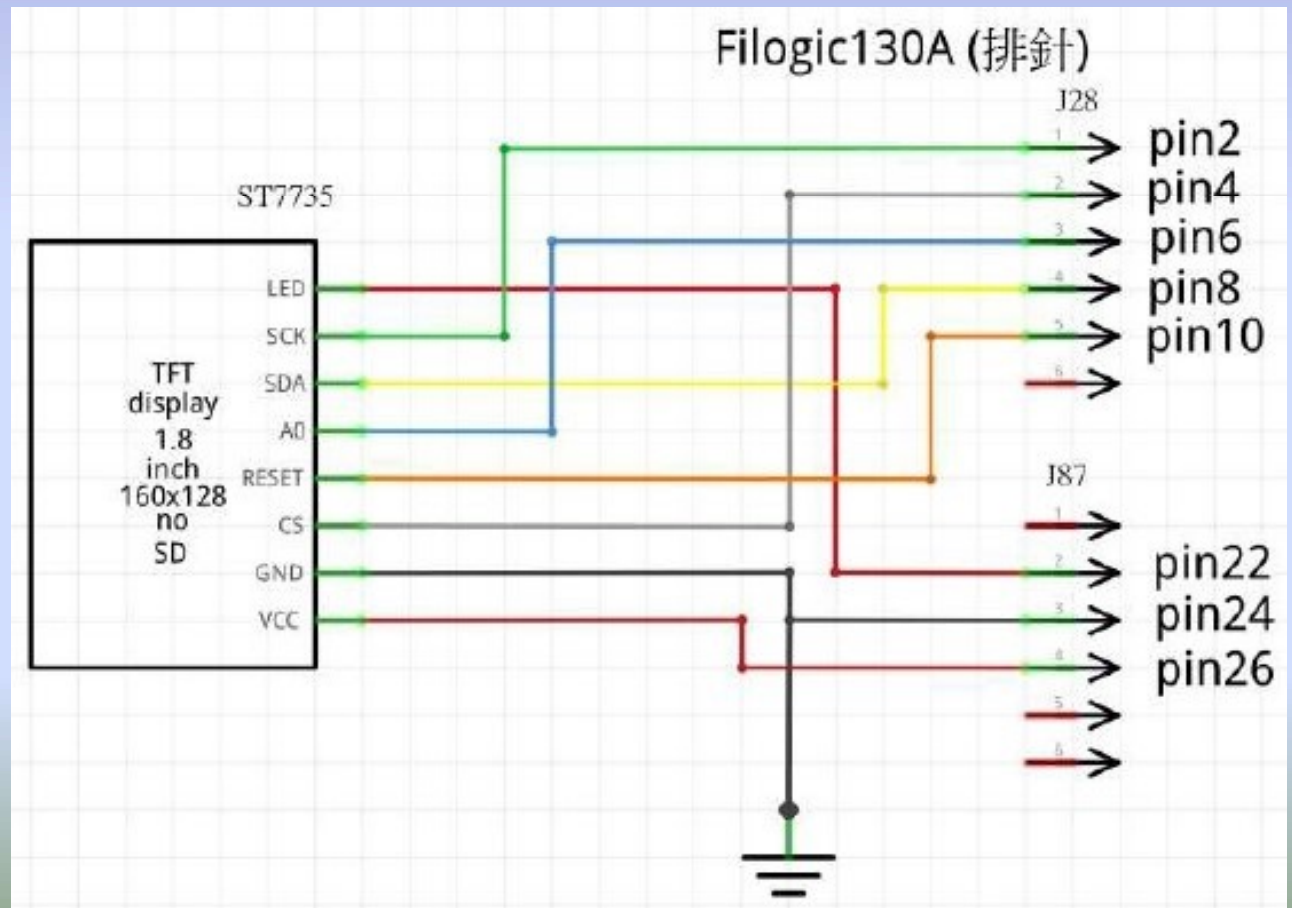


# 介面模組：SPI (3)

- 範例程式 **Display\_Adafruit**，接上兩個顯示模組：
  - TFT-LCD ST7735 (128 x 160)：SPI介面
  - OLED SSD1306 (128 x 64)：I<sup>2</sup>C介面

ST7735 pins:

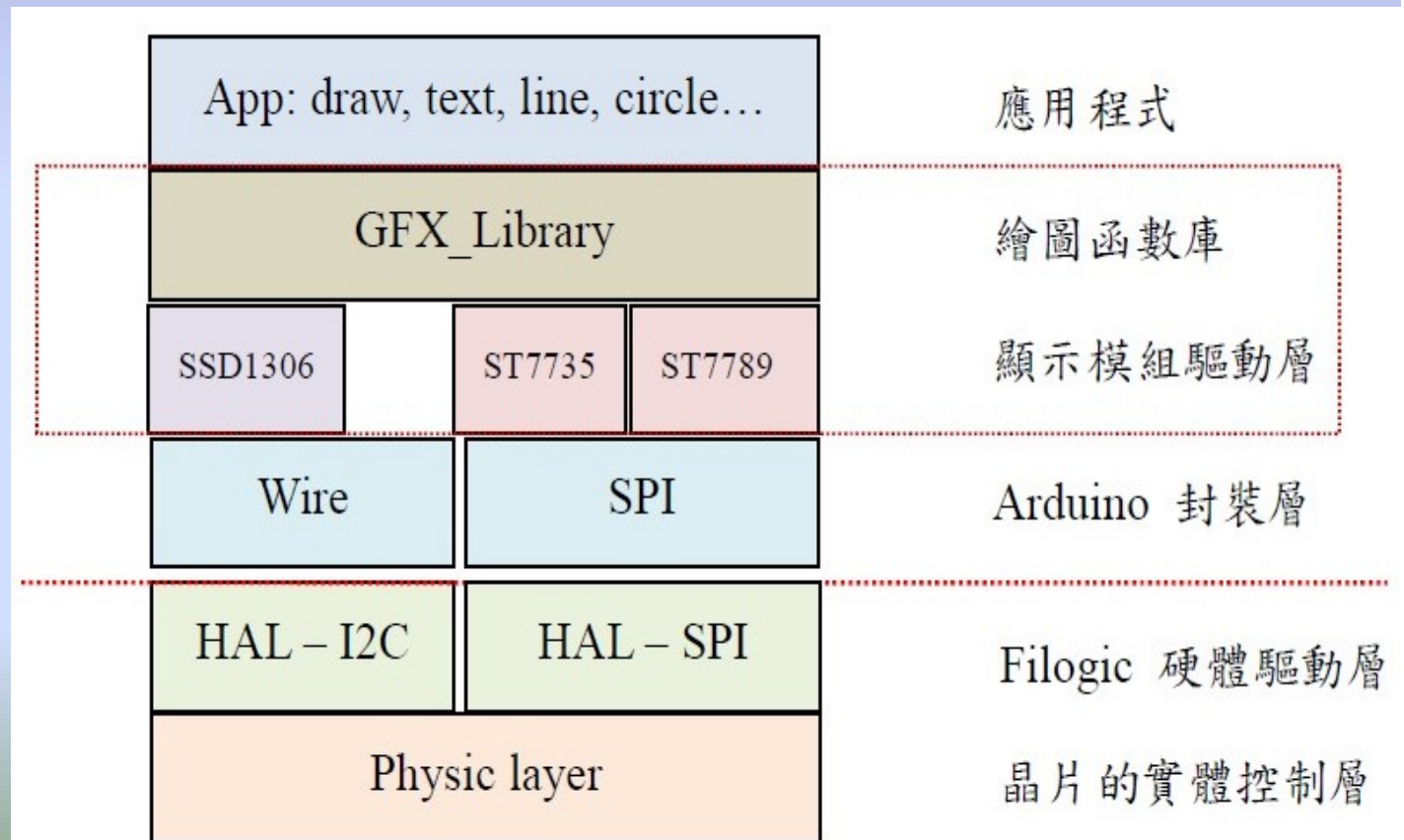
SCK ---> CLK  
SDA ---> MOSI  
A0 ---> DC  
RES ---> Reset  
CS ---> CS



# SDK 架構：SPI (4)

- Arduino SDK v1.1.0版本提供I<sup>2</sup>C與SPI控制介面，軟體封裝架構如下：
- Adafruit library 架構圖

- GFX
- BusIO
- SSD1306
- ST7735





# 範例程式碼：SPI (5)

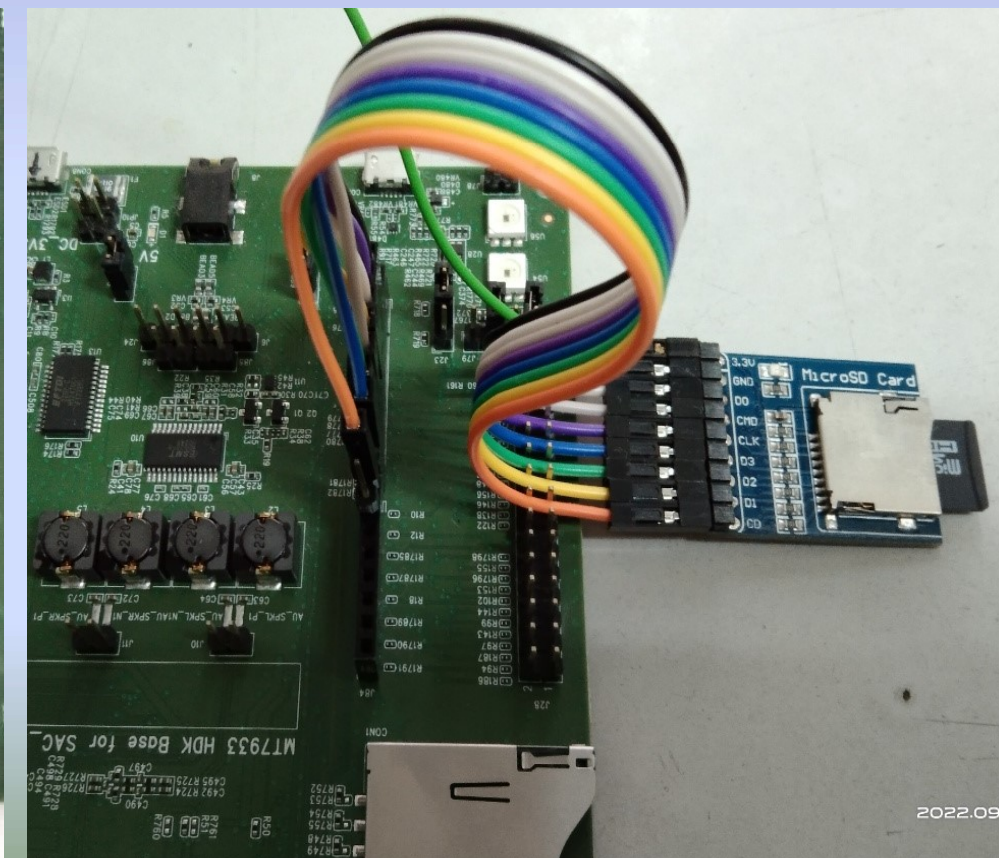
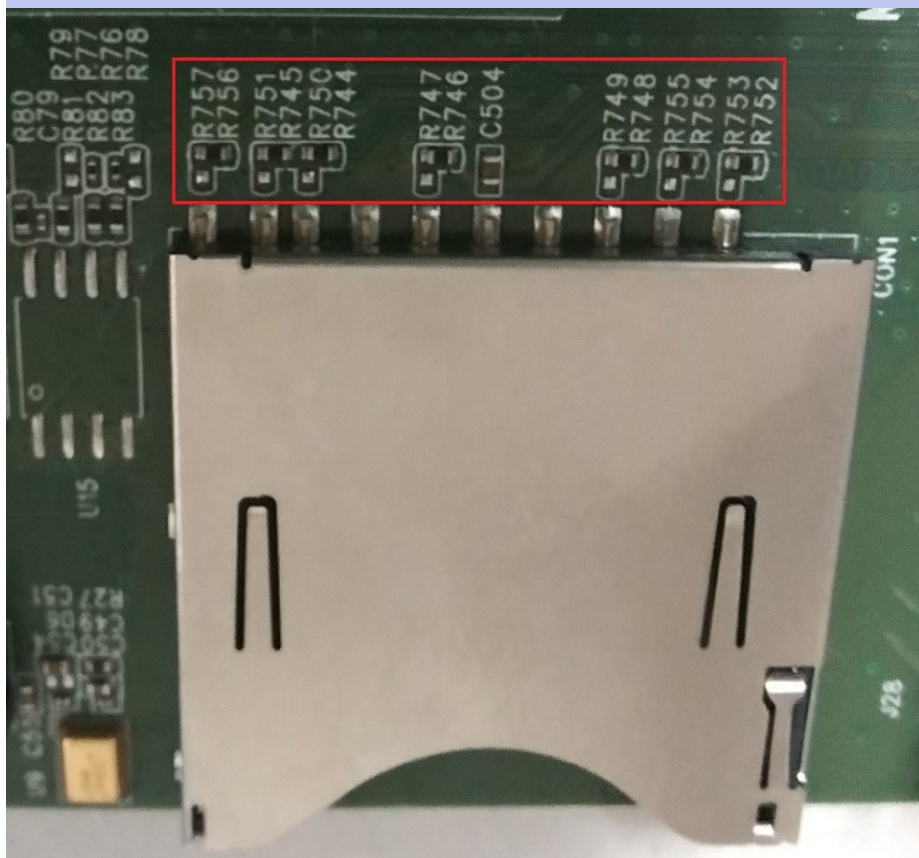
- 下載[github.com/yijenlu1971/Filogic\\_onArduino](https://github.com/yijenlu1971/Filogic_onArduino)
- 影片連結 <https://youtu.be/JF1PXoPQ5IE>



# 音訊與SD介面實驗

# SDIO 介面

- 開發板上的SD插槽，需要焊接才能接通介面
- 或者，外接SD插槽，接到J81排針上



- 開發板上的SD卡槽
  - R747, R749, R750, R751, R753, R755, R757七個點要焊接導通。
- 外接SD卡槽到**J81**
  - J81的pin4 GND
  - J81的pin5 SDIO\_CLK
  - J81的pin6 SDIO\_DAT0
  - J81的pin7 SDIO\_DAT1
  - J81的pin8 SDIO\_CMD
  - J81的pin9 SDIO\_DAT2
  - J81的pin10 SDIO\_DATA3

# SDIO 的檔案管理

- SDK v1.2.0版本已經整合第三方開源庫**fatfs** (FAT File System)
  - 標準化的檔案管理函數：f\_open、f\_read、f\_write、f\_close...等。
  - 記錄數據
- 範例程式**PlayWavFromSD**
  - 讀取SD卡的檔案，撥放wav音檔



## 音源介面

- 開發板上的J6和J24，以及J10和J11與聲音輸出有關。J10和J11分別用來外接左右聲道的喇叭。
- J6和J24用來將音源切換到U10放大器或者Line-Out輸出。
  - 一切到U10：J6 (2,3)和J24 (1,2)，J10/J11接上喇叭。
  - 一切到Line-Out：J6 (1,2)和J24 (2,3)
  - Line-Out端子需要接上有電源的喇叭，無法驅動耳機。
  - 下載範例程式**PlayAudio**

# 音源輸出路由

- MT7933 內部處理聲音輸出的三種路由：
  - 直接輸出到DAC，如下圖
  - 輸出TDM (數位音訊)，如I2S，可參考IO mux
  - 輸出GA src (???)

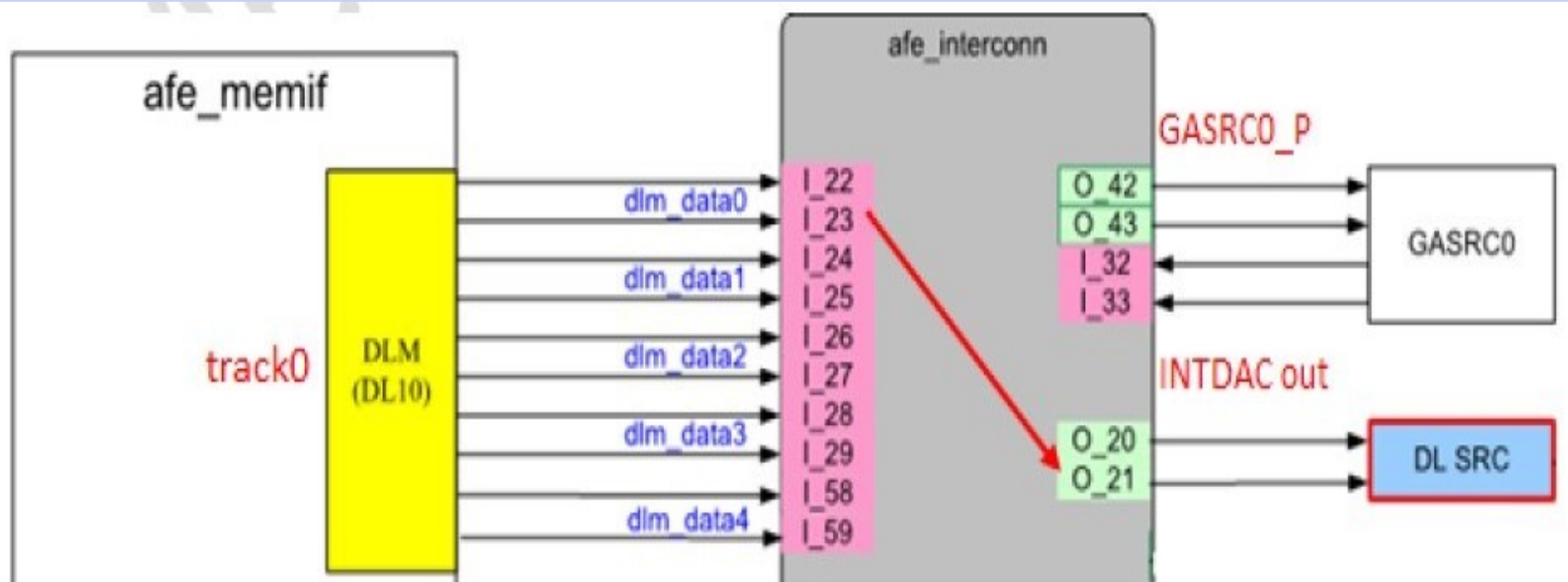


Figure 4-9. *dlm\_intdac* path

# 音源輸出路由

- 範例程式碼，直接輸出到DAC埠

```
41 void TaskPlayMusic(void *param)
42 {
43     int idx = 0;
44
45     for(;;) {
46         while( fNum == 0 ) vTaskDelay( 1000 );
47         printf("# of wav_file: %d\n", fNum);
48
49         connect_route("track0", "INTDAC out", 1, CONNECT_FE_BE);
50         connect_route("I_22", "O_20", 1, CONNECT_IO_PORT);
51         connect_route("I_23", "O_21", 1, CONNECT_IO_PORT);
52
53         while(bMount) {
54             PlayWavFile( flist[idx].name );
55
56             if( ++idx >= fNum ) idx = 0;
57             vTaskDelay( 1500 );
58         }
59     }
60 }
```

# 音源輸入路由

- MT7933 內部處理音源輸入的三種路由：
  - 直接從ADC輸入，如下圖
  - 輸入TDM (數位音訊)，如I2S
  - 輸入DMIC (Digital Microphone Interface)

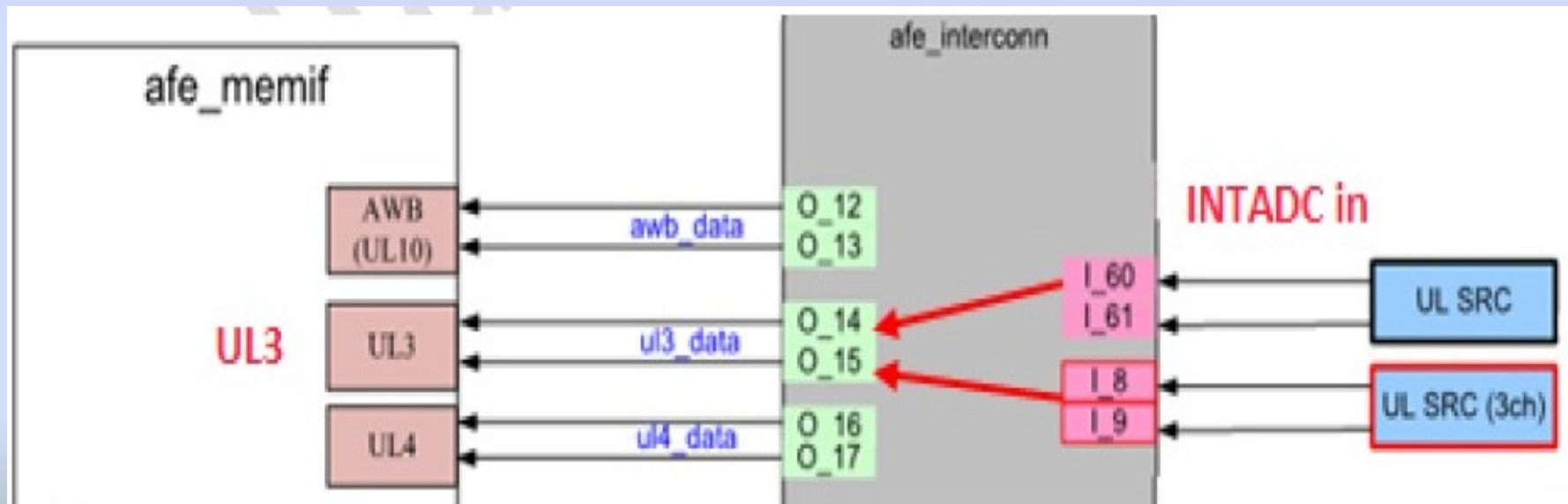


Figure 4-13. ul3\_intadc path

# 範例程式碼：SD + 音源

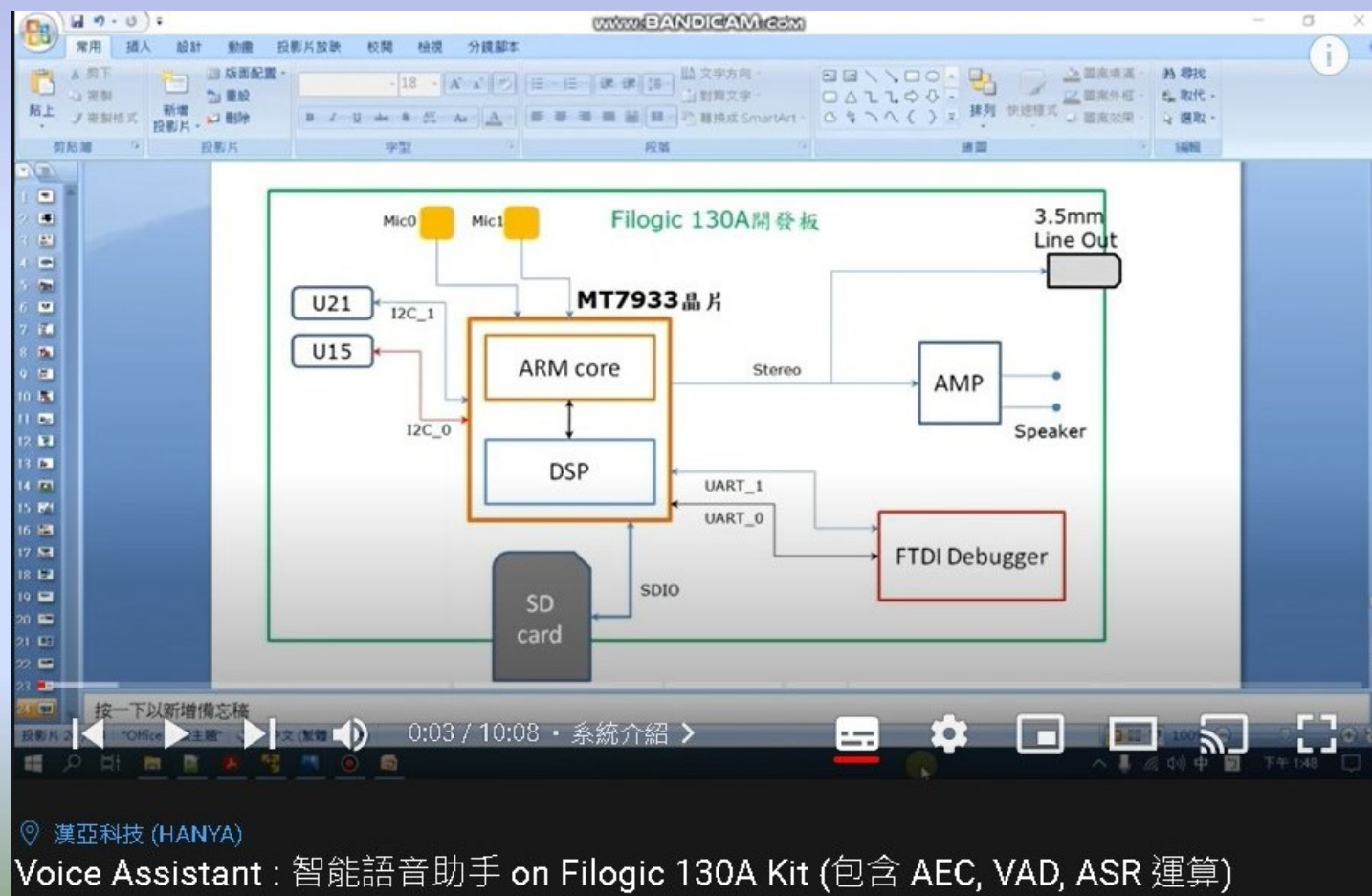
- 下載 [github.com/yijenlu1971/Filogic\\_onArduino](https://github.com/yijenlu1971/Filogic_onArduino)
  - 範例程式 **PlayWavFromSD**，**PlayAudio**
- 影片連結 <https://youtu.be/D6tEVUoJPQ8>





# 語音辨識 ASR

- 聯發科與第三方ASR開發商尚有license細節需要處理，Arduino SDK版本暫時沒有包含ASR功能。
- Linux版本的SDK有提供ASR demo的範例，參考YT影片說明。

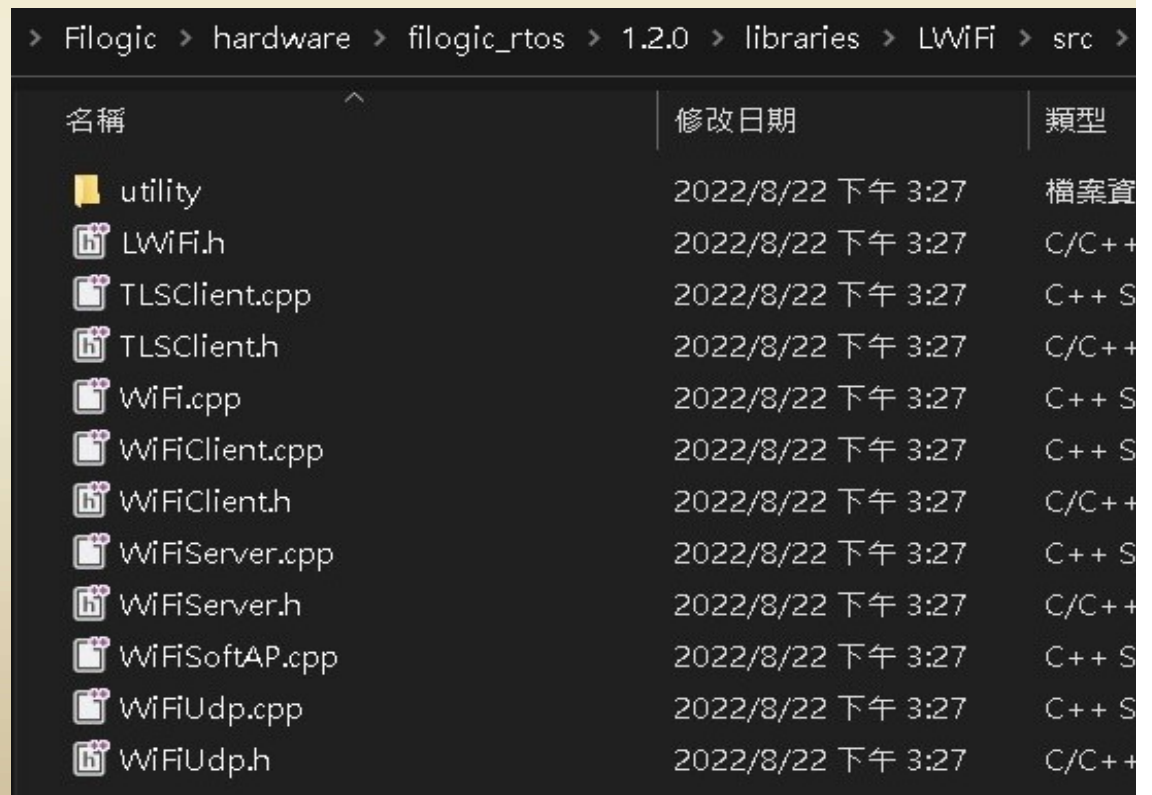


# 網路雲端數據實驗

WiFi / TCP / UDP

# 無線網路 (1)

- WiFi具有
  - AP (Access Point)
  - Station (NoEncrypt、WEP、WPA)
- WiFi class介面的呼叫方式與Arduino架構相同
- 定義網路的類別
  - WiFi
  - UDP
  - Server (TCP)
  - Client (TCP)
  - TLS



The screenshot shows a file explorer window with the path: > Filogic > hardware > filogic\_rtos > 1.2.0 > libraries > LWiFi > src >. The table below lists the files and folders in this directory.

名稱	修改日期	類型
utility	2022/8/22 下午 3:27	檔案夾
LWiFi.h	2022/8/22 下午 3:27	C/C++
TLSClient.cpp	2022/8/22 下午 3:27	C++ S
TLSClient.h	2022/8/22 下午 3:27	C/C++
WiFi.cpp	2022/8/22 下午 3:27	C++ S
WiFiClient.cpp	2022/8/22 下午 3:27	C++ S
WiFiClient.h	2022/8/22 下午 3:27	C/C++
WiFiServer.cpp	2022/8/22 下午 3:27	C++ S
WiFiServer.h	2022/8/22 下午 3:27	C/C++
WiFiSoftAP.cpp	2022/8/22 下午 3:27	C++ S
WiFiUdp.cpp	2022/8/22 下午 3:27	C++ S
WiFiUdp.h	2022/8/22 下午 3:27	C/C++

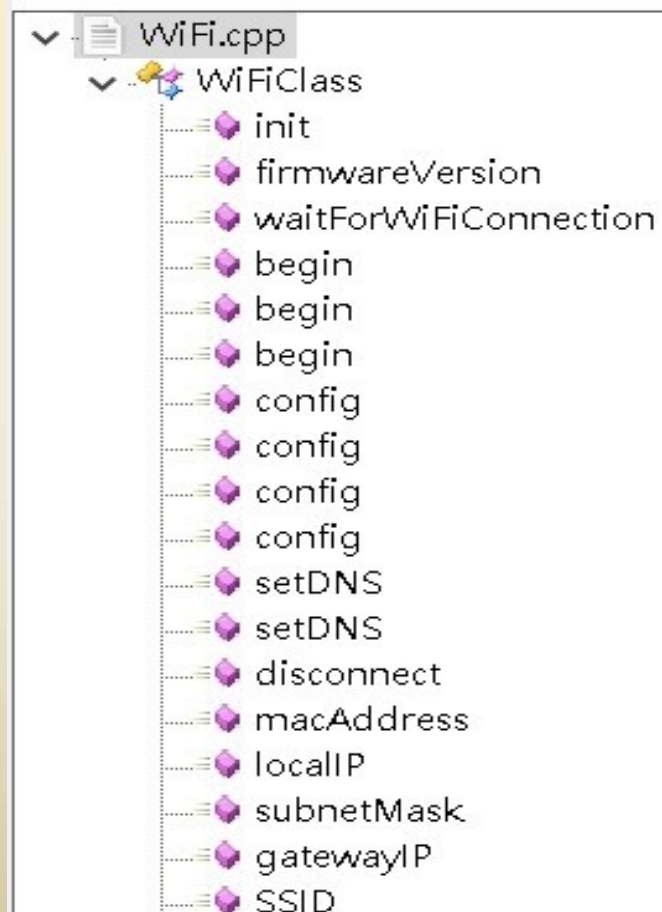
# 無線網路 (2)

## Arduino WiFi class 函數

### Wifi Class

- WiFi.begin()
- WiFi.disconnect()
- WiFi.config()
- WiFi.setDNS()
- WiFi.SSID()
- WiFi.BSSID()
- WiFi.RSSI()
- WiFi.encryptionType()
- WiFi.scanNetworks()
- WiFi.status()
- WiFi.getSocket()
- WiFi.macAddress()

## Fillogic WiFi class 函數



## ✓ 準備無線網路

WiFi-AP設備，取得AP的SSID名稱與密碼。或者開啟手機的熱點分享。

## ✓ 連上無線網路

學習如何讓開發板連上WiFi-AP設備，並取得開發板的IP位址。



# 無線網路 (3)

- WiFi - station
  - 呼叫 **begin** 函數
  - ssid, password 連線AP
  - 檢查連線狀態
  - 取得被分配的IP位址
- WiFi - access point
  - 呼叫 **softAP** 函數
  - ssid, password 等待連線
  - 取得本身的 IP 位址
  - 處理每個client連線

```
// attempt to connect to Wifi network:
while (status != WL_CONNECTED) {
  Serial.print("Attempting to connect to WPA SSID: ");
  Serial.println(ssid);
  // Connect to WPA/WPA2 network:
  status = WiFi.begin(ssid, pass);
}
```

```
/* You can remove the password parameter if you want the AP to be open. */
WiFi.softAP(ssid, password);
IPAddress myIP = WiFi.softAPIP();
Serial.println("AP ready.");
Serial.print("Connect to AP ");
Serial.print(ssid);
```

# 無線網路連線

## ✓ 範例 **ConnectWithWPA**

在底下的程式碼中，填入自己AP的 SSID 和連線密碼。

```
ConnectWithWPA $  sntp.h  sntpWiFiUdp.cpp
15  */
16  #include <WiFi.h>
17  #include <LRTC.h>
18  #include <TLSClient.h>
19  #include "sntp.h"
20
21  #define LED_BUILTIN 7
22
23  char ssid[] = "ssid"; // your network SSID (name)
24  char pass[] = "1234567890"; // your network password
25  int status = WL_IDLE_STATUS; // the Wifi radio's status
26  //TLSClient iotClient;
27  bool bOn = false;
28
```

# 網路範例：NTP (1)

- 取得網路時間
  - 原廠提供範例：WiFiUdpNtpClient
  - 下載範例 **ConnectWithWPA**
- 範例
  - 設定DNS位址，並取得NTP服務器的IP位址
  - 宣告一個 **WiFiUdp** 的 ntpClient 連線
  - 開啟晶片的 **LRTC** 功能
  - 取得時間後，再設定到RTC (Real-Time Clock)

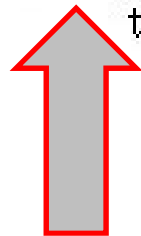
```
WiFi.setDNS(dns);  
if( WiFi.hostByName(hostname, ntpAddr) )  
{  
    Serial.print("NTP IP: "); Serial.println(ntpAddr);
```

```
82  
83 WiFiUDP ntpClient;  
84 uint8_t sntpBuf[SNTP_MSG_LEN];  
85 uint8_t sntpRxBuf[SNTP_MSG_LEN*2];  
86 u32_t sntp_timestamp;
```

台灣的時區 +8



```
77 ut = time(NULL) + sntp_timestamp + (8*60*60);  
78 tmInfo = localtime(&ut);  
79 LRTC.set(tmInfo->tm_year+1900, tmInfo->tm_mon+1, tmInfo->tm_mday,  
80 tmInfo->tm_hour, tmInfo->tm_min, tmInfo->tm_sec);
```



將時間設定到晶片的RTC

# 網路範例：NTP (2)

- NTP測試結果

You're connected to the networkSSID: ASUS

BSSID: 4C:ED:FB:8D:E0:18

signal strength (RSSI):-34 dBm

Encryption Type:9

WPA-PSK/WPA2-PSK

Name stl

IP Address: 192.168.1.229

[14517]<226>[common][I][get\_mac][436]get mac 9f:6c:55:f9:65:9c

MAC address: 9C:65:F9:55:6C:9F

Dump rtc setting, pwr1:c6, pwr2:9a, rtc\_key:59, prot1:a3, prot2:57, prot3:67, prot4:d2

[14549]<227>[common][I][foundHostByName][20]foundHostByName: Found Host: name=time.stdtime.gov.tw

NTP IP: 118.163.81.61

Sending request:48

[18229]<228>[common][I][parsePacket][209]lwip\_recvfrom received 48 bytes

sntp\_process: 1663556356

rtc read time, yea:22, mon:9, dom:19, dow:0, hou:10, min:59, sec:36

2022/9/19 10:59:36

rtc read time, yea:22, mon:9, dom:19, dow:0, hou:10, min:59, sec:47

2022/9/19 10:59:47



# 網路雲端數據 (1)

- 雲端數據 **ThingSpeak**
  - 註冊個人帳號
  - 創建一個新的channel，並命名。
  - 取得連線的API key (分別有讀、寫兩把key)
  - 設定channel裡面，建立欲觀看的欄位，欄位名
  - Private裡面，可以檢視Dashboard

	<b>FREE</b> For small non-commercial projects
Scalable for larger projects	<b>×</b> No. Annual usage is capped.
Number of messages	<b>3 million/year</b> (~8,200/day) <sup>(2)</sup>
Message update interval limit	<b>Every 15 seconds</b>
Number of channels	<b>4</b>
MATLAB Compute Timeout	<b>20 seconds</b>
Private channel sharing	<b>Limited to 3 shares</b>

## 網路雲端數據 (2)

- ThingSpeak網站上設定的步驟

ThingSpeak™ Channels ▾ Apps ▾ Devices ▾ Support ▾

### My Channels

New Channel ← 1

Search by tag 🔍

Name ⬆	Created ⬆	Updated ⬆
4 🔒 Filologic 130A 3	2022-05-04	2022-05-06 23:19
Private Public Settings Sharing API Keys 2 Data Import / Export		

## 網路雲端數據 (3)

- 開發板如何上傳數據？
  - 雲端網站提供REST API 和 MQTT API 兩種方式：實作範例以REST為主。
  - 下載範例**HttpClient**，連線的程式碼取自Linux版本的SDK，提供了第三方 http 連線原始碼。
  - 以Http POST方式連線 ThingSpeak 網站，建立連線 `httpclient_connect`，如下

```
sprintf(get_url, "https://api.thingspeak.com/update");  
printf("httpclient_connect:%d", httpclient_connect(&iotClient, get_url));
```

## 網路雲端數據 (4)

– Http內容格式：

**api\_key=xxx&field1=xxx&field2=xxx**

– 傳送內容 httpclient\_post 函數

```
iotData.response_buf = iotRespBuf;
iotData.response_buf_len = BUF_SIZE;

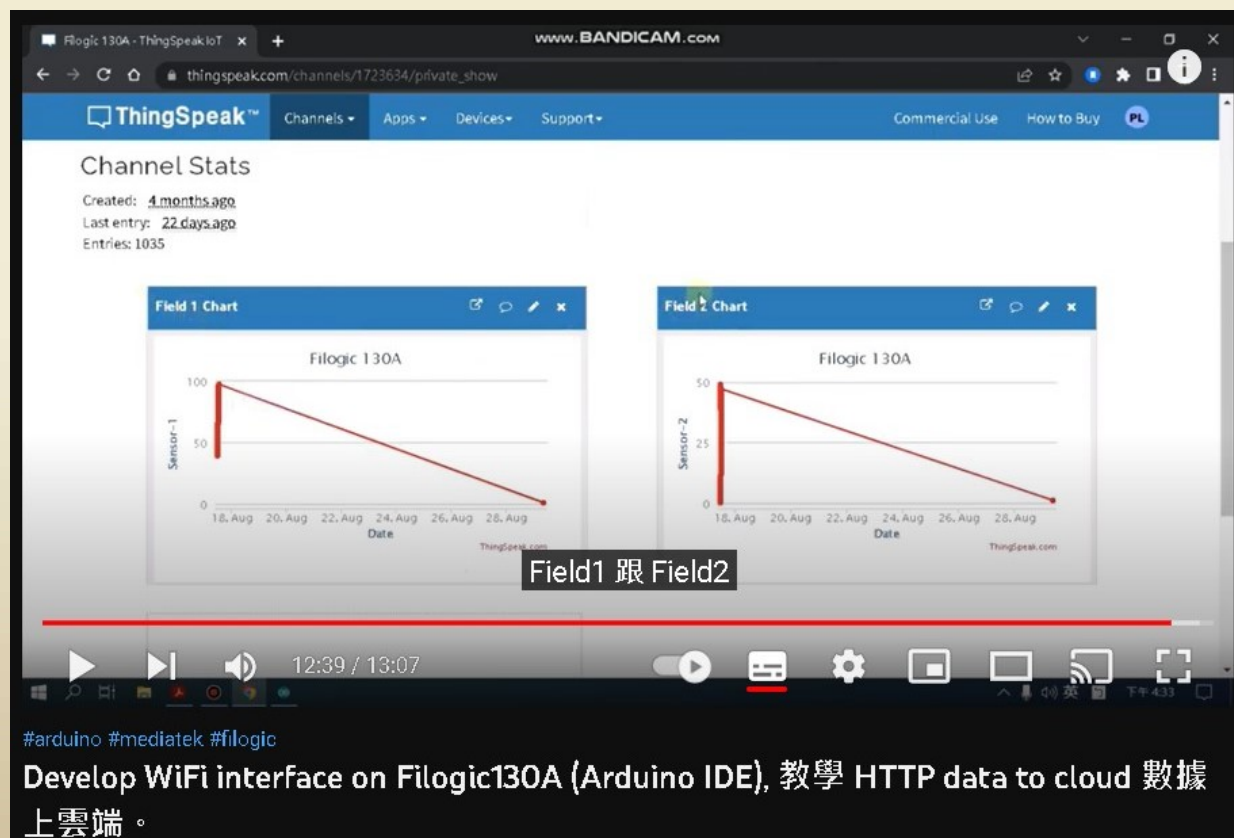
sprintf(iotPostBuf, "api_key=XXXXXXXXXXXXXXXXXXXX&field1=%d&field2=%d", ++tempVal1 % 100, ++tempVal2 % 50);
iotData.post_buf = iotPostBuf;
iotData.post_buf_len = strlen(iotPostBuf);

sprintf((char*)buf, "application/x-www-form-urlencoded");
iotData.post_content_type = (char*)buf;

printf("connecting to ThingSpeak for update...\n");
if( httpclient_post(&iotClient, get_url, &iotData) == HTTPCLIENT_OK)
{
    printf("data_len=%d %s\n", iotData.response_content_len - iotData.retrieve_len, iotData.response_buf);
}
```

# 實作結果

- 參考影片連結
  - <https://youtu.be/OTzjujqmcdo>
  - <https://youtu.be/jbtSzYi-x7M>





# 藍芽BLE通訊實驗

連線：手機、電腦、開發板

# 藍芽 Profile

- 底下這些常見的Profile，在SDK裡面尚未提供
  - A2DP
  - Hand-free Profile (HFP)
  - Headset Profile (HSP)
- 目前Fillogic SDK提供藍芽的**GATT**通訊協定

# 藍芽BLE (1)

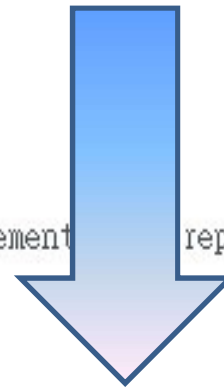
- Filogic SDK：藍芽的**GATT**通訊協定
- BLE裝置的角色分為
  - Peripheral (server端)：GATT協定提供服務、特徵、屬性，向周遭發出 advertisement 廣播封包
  - Central (client端)：掃描周遭環境，建立連線

> Filogic > hardware > filogic_rtos > 1.2.0 > libraries > LBLE > src >		
名稱	修改日期	類型
utility	2022/8/22 下午 3:27	檔案資料夾
LBLE.cpp	2022/8/22 下午 3:27	C++ Source
LBLE.h	2022/8/22 下午 3:27	C/C++ Header
LBLECentral.cpp	2022/8/22 下午 3:27	C++ Source
LBLECentral.h	2022/8/22 下午 3:27	C/C++ Header
LBLEPeripheral.cpp	2022/8/22 下午 3:27	C++ Source
LBLEPeripheral.h	2022/8/22 下午 3:27	C/C++ Header

## 藍芽BLE (2)

- Filogic開發板當作 Peripheral端
  - 下載範例 **SimplePeripheral**
  - 可修改廣播的名稱 **MTK\_BLE Lab** 方便識別。
- Central端分別用以下裝置測試連線：
  - 手機：安裝“BLE Scanner”或“B-BLE”APP。
  - 電腦：安裝“BLEConsole”工具程式
  - 開發板：使用另一塊Filogic開發板，下載範例 **ConnectPeripheral**。或者，第三方開發板(如：ESP32、Ameba)具有Central端的連線功能。

實作時，修改廣播的  
名稱，不要重複才能  
區分。

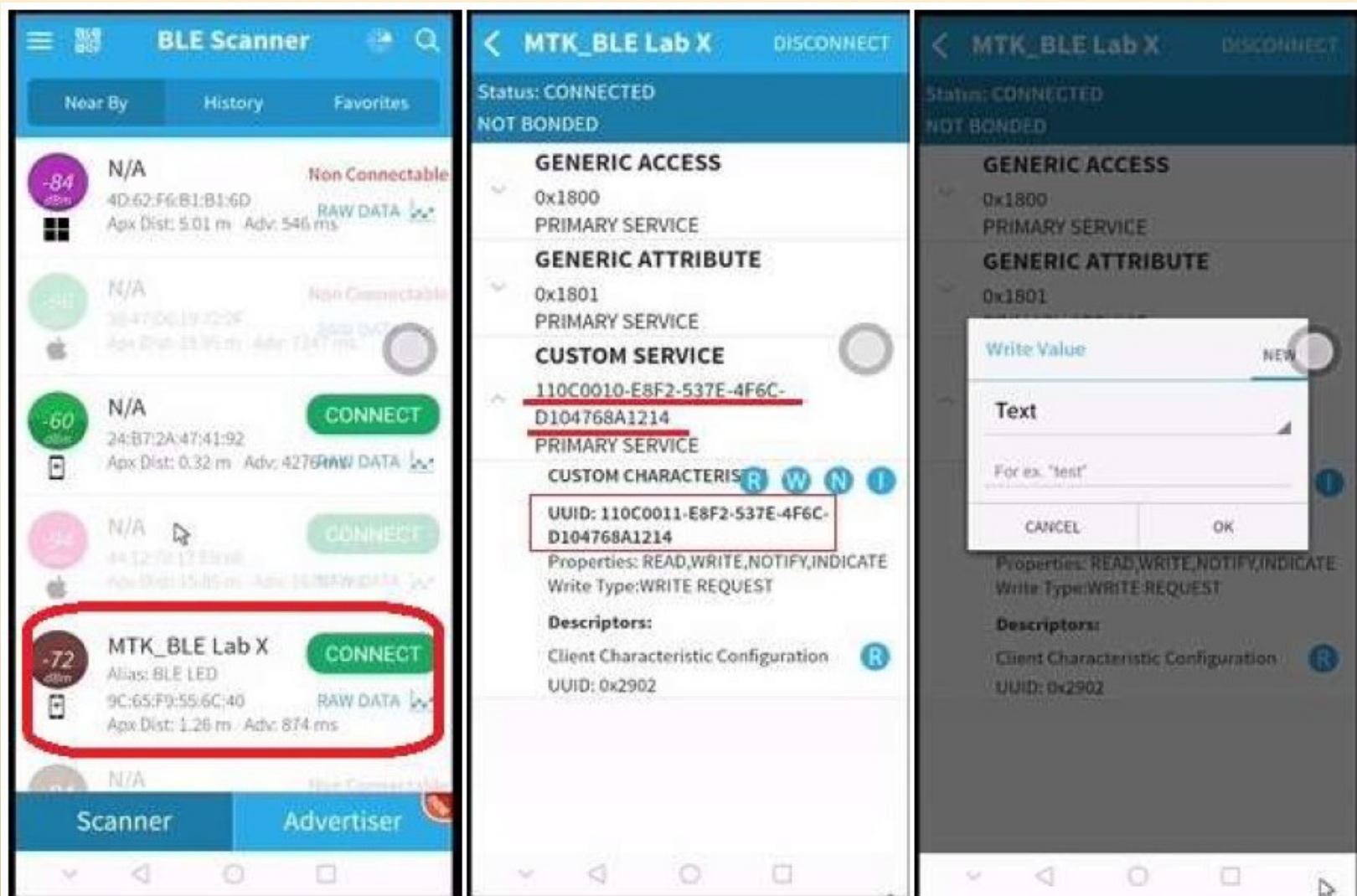


```
47 // configure our advertisement data.
48 // In this case, we simply create an advertisement that represents an
49 // connectable device with a device name
50 LBLEAdvertisementData advertisement;
51 advertisement.configAsConnectableDevice("MTK_BLE Lab X");
52
53 // Configure our device's Generic Access Profile's device name
54 // Usually this is the same as the name in the advertisement data.
55 LBLEPeripheral.setName("MTK_BLE Lab X");
56
```



# 藍芽BLE - 手機

- BLE Scanner連線開發板，取得裝置的服務訊息



# 實作結果


參考影片連結

<https://youtu.be/esCSL31hE9I>



# 藍芽BLE - 電腦

- 從Microsoft Store上，安裝BLEConsole應用程式



BLE

**BLEConsole**

SeNSSoft

開啟

NT\$33.00

-- ★ 0  
平均 評等

螢幕擷取畫面

```
blec_mon_5868a54caccd1ac_28d9dc7b218db0f
BLE: help
BLEConsole ver. 1.5.6

help, ?      : show help information
quit, q      : quit from application
list, ls [v] : show available BLE devices
open <name> or <ID> : connect to BLE device
delay <ms>   : pause execution for a certain number of milliseconds
timeout <ms> : show/change connection timeout, default value is 3 sec
close        : disconnect from currently connected device
stat, st     : shows current BLE device status
print, p <text&vars> : prints text and variables to stdout, where are variables are
                  : %id - Bluetooth device ID
                  : %addr - device BT address
                  : %mac - device MAC address
                  : %name - device Bluetooth name
                  : %stat - device connection status
                  : %NDU, %NDU, %HU, %hb, %aa, %ss, %D, %d, %T, %t, %X - date/time variables
format [data_format], fmt : show/change display format, can be ASCII/UTF8/Dec/Hex/Bin
set <service_name> or <S> : set current service (for read/write operations)
write, w <name>=<value>   : write value to specific characteristic
subs <name>=<v>           : subscribe to value change for specific characteristic
unsubs <name>=<v> [all]    : unsubscribe from value change for specific characteristic or unsubs all for all
wait                      : wait for notification event on value change (you must be subscribed, see above)
foreach [device_mask]     : starts devices enumerating loop
if <cmd> {<params>}       : start conditional block dependent on function returning v/o error
elif                      : another conditional block
else                      : if condition == false block
endif                    : end conditional block

* You can also use standard C language string formatting characters like \t, \n etc.
** <name> could be "Service/characteristic", or just a char name or # (for selected service)
For additional information and examples please visit https://github.com/shelltechnetworks/BLEConsole

BLE: In
#00: FlangeLights00001
#01: FlangeLights00002
#02: Pebble
BLE: open #2
Connecting to Pebble.
Found 2 services:
#00: GenericAccess
#01: GenericAttribute
#02: DeviceInformation
#03: ImmediateAlert
#04: LinkLoss
#05: TxPower
#06: Battery
BLE:
```

描述

Windows 10 command line tool for interacting with Bluetooth LE devices. For more information, run a and press enter, or visit a web site <https://sensboston.github.io/BLEConsole>  
Application has no limitation but if you like it or found this app useful, you may purchase as donation.

# 藍芽BLE - 電腦

- BLEConsole連線開發板，透過命令列下指令
  - list，列出周遭的藍芽裝置，找到 MTK\_BLE 的編號
  - open，連線裝置
  - set
  - write

```
C:\Program Files\WindowsApps\52745SeNSSoft.BLEConsole_1.5.6.0_x64__9t942y
BLEConsole ver. 1.5.6

BLE: list
#00: ACTON II
#01: Apple Pencil
#02: HID
#03: MTK_BLE Lab X
#04: N/A
BLE: open #03
Connecting to MTK_BLE Lab X.
Found 3 services:
#00: GenericAccess
#01: GenericAttribute
#02: Custom Service: 110c0010-e8f2-537e-4f6c-d104768a1214
BLE:
```



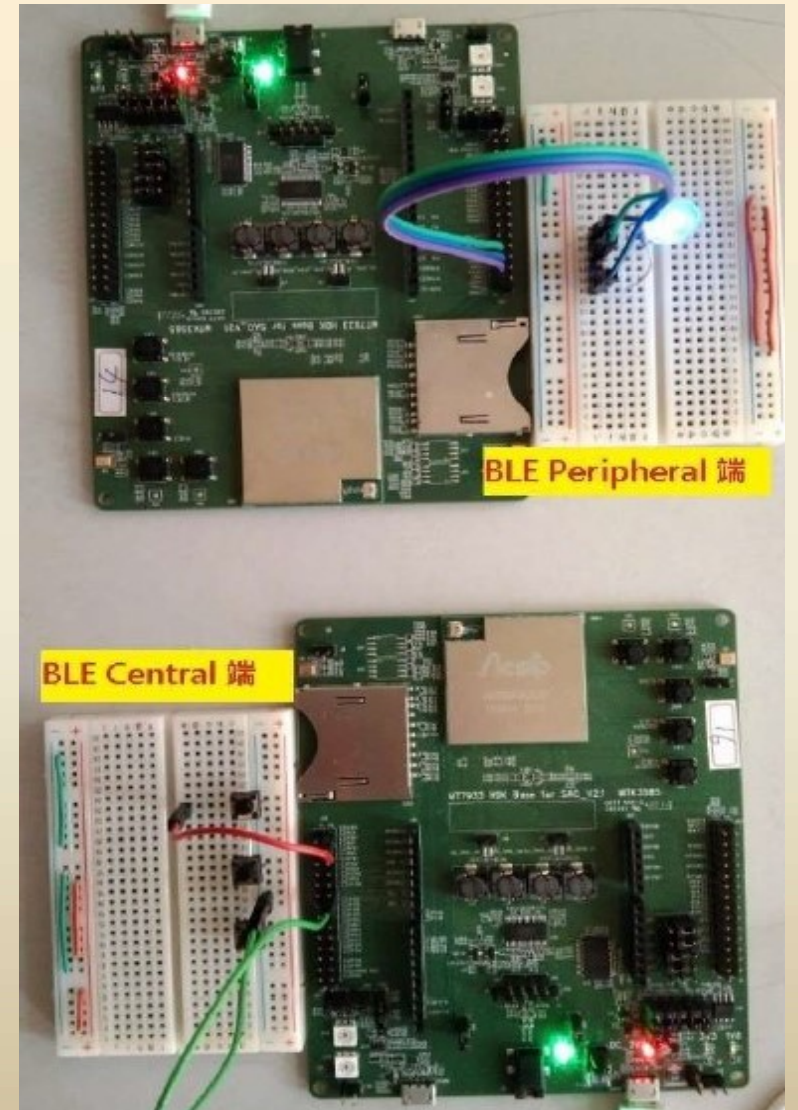
# 藍芽BLE - 電腦

```
Connecting to MTK_BLE Lab X.  
Found 3 services:  
#00: GenericAccess  
#01: GenericAttribute  
#02: Custom Service: 110c0010-e8f2-537e-4f6c-d104768a1214  
BLE: set #02  
Selected service Custom Service: 110c0010-e8f2-537e-4f6c-d104768a1214.  
#00: Custom Characteristic: 110c0011-e8f2-537e-4f6c-d104768a1214 RWN  
BLE: w #00 0  
BLE: w #00 1  
BLE:
```



# 藍芽BLE - 開發板

- 下載範例 **ConnectPeripheral**，讓兩張開發板能夠互連
  - 一塊是LED燈號顯示 (接收命令)
  - 一塊是按鈕控制 (發送命令)



謝謝指教  
Q&A