

- WELCOME TO -

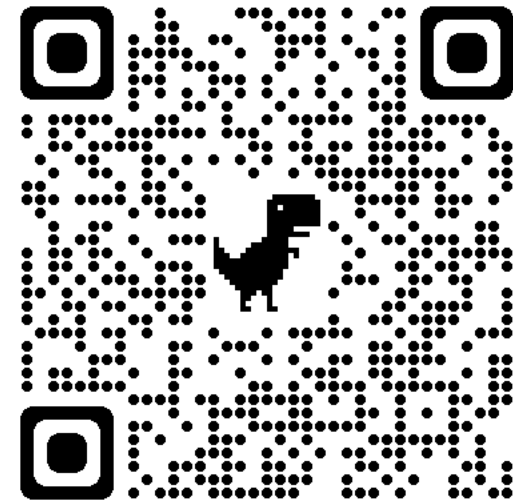
**賽馬會科藝共融計劃**

JC-Project-IDEA

## **Part 4 : Arduino(II) - Basic Coding**



Starter Project 1 - Graphic Sensors  
[https://www.youtube.com/watch?  
v=29ZliJJQTFI](https://www.youtube.com/watch?v=29ZliJJQTFI)



在使用Arduino 做到影片中的效果前

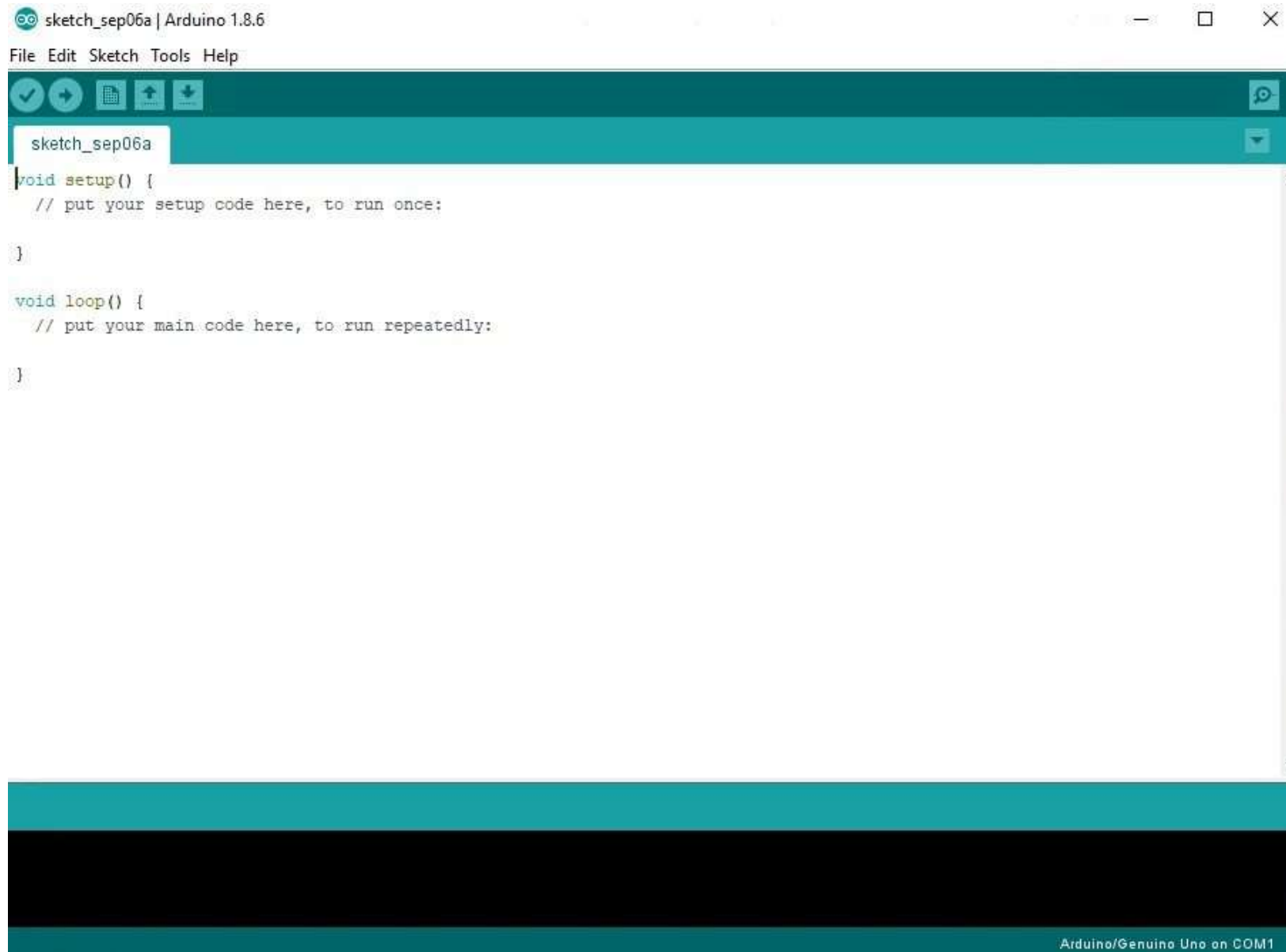
## **重溫第二節內容**

# Arduino IDE

Arduino IDE 1.8.19 是一個的開發環境，提供代碼編輯器、實時錯誤檢查。

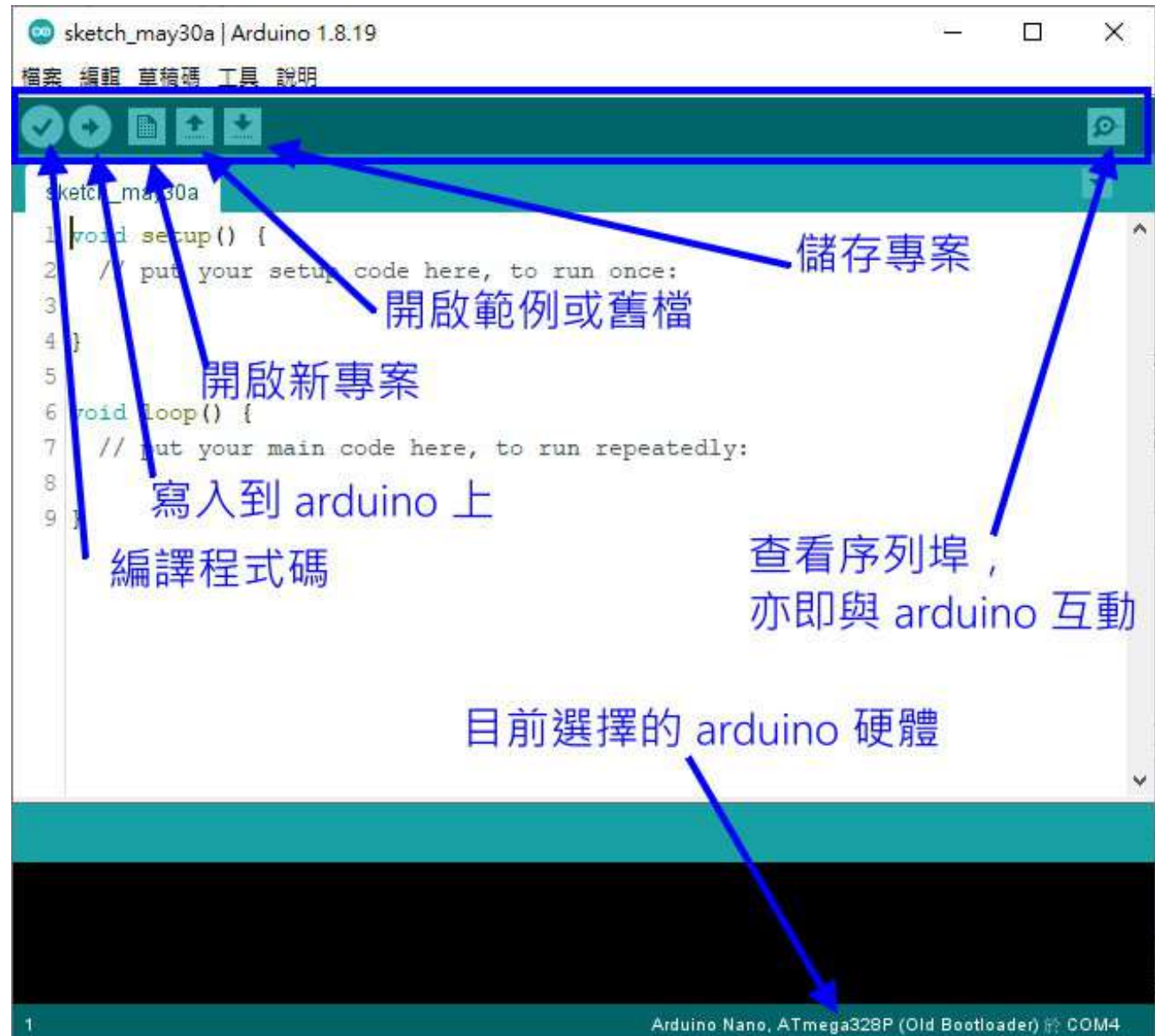


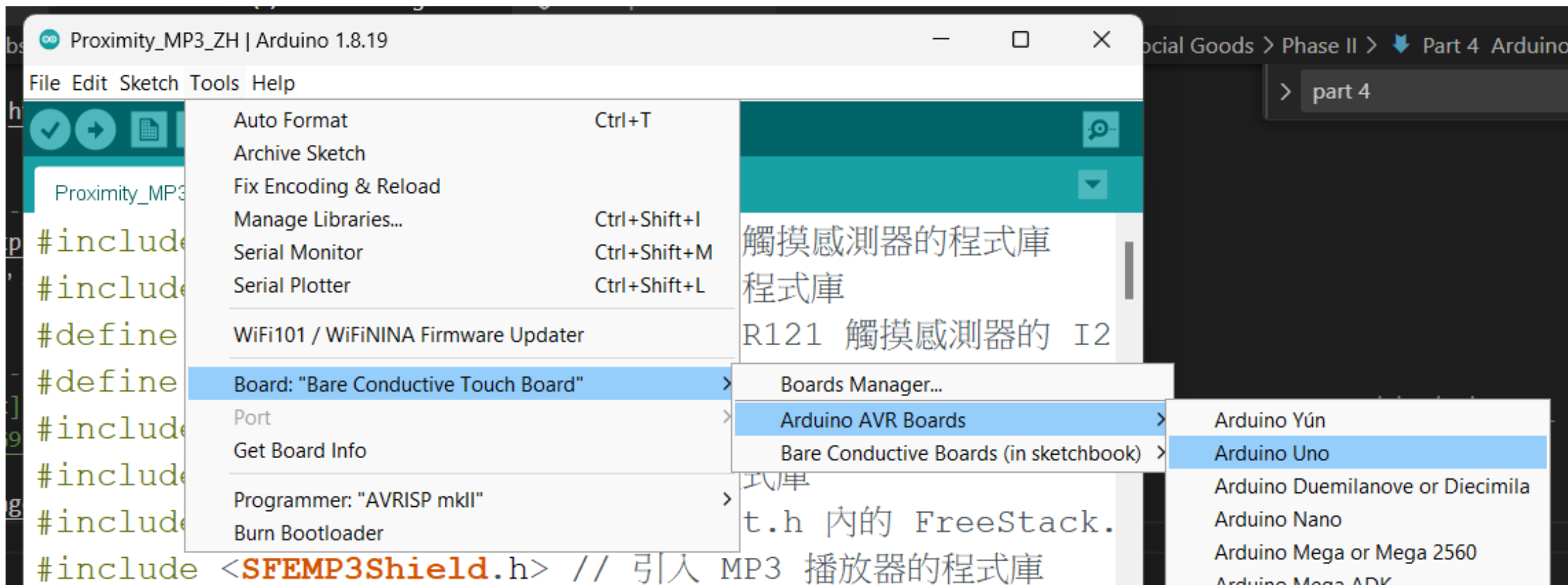
# 使用介面簡介



啟動 Arduino IDE：您將看到主界面，包括編輯區域、工具欄和菜單欄。







選擇開發板：在菜單欄，選擇「工具」>「開發板」，然後選擇您所使用的 Arduino 開發板。



先確定 LED 燈亮起



選擇連接埠：再次點擊「工具」，選擇「連接埠」，然後選擇 Arduino 開發板所連接的串口。

```
sketch_may30a

1 void setup() {
2   // put your setup code here, to run once:
3   初始化 arduino 硬體的程式碼
4 }
5
6 void loop() {
7   // put your main code here, to run repeatedly:
8
9   實際運作的程式碼
}
```

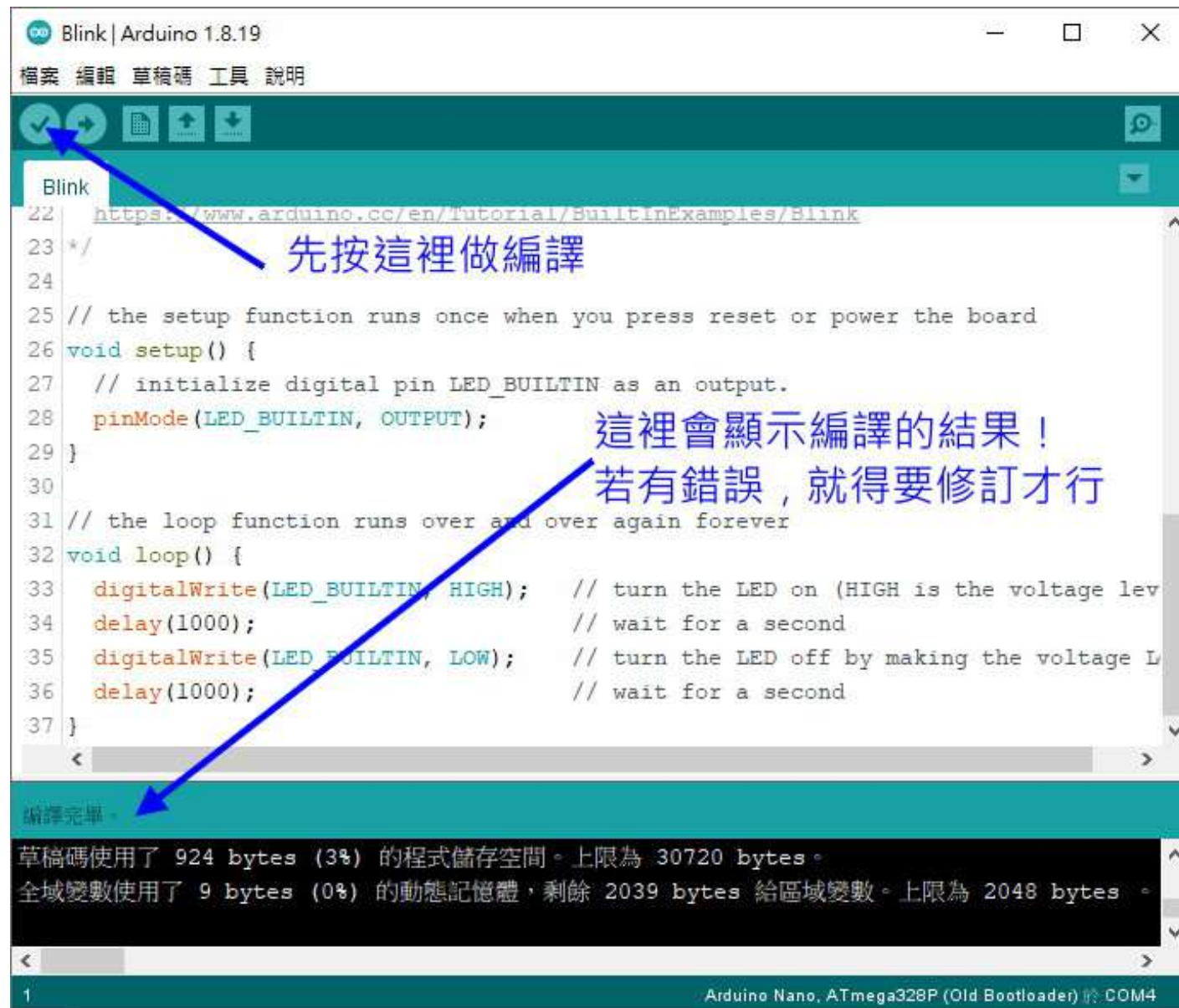
1 Arduino Nano, ATmega328P (Old Bootloader) 於 COM4

編寫程式碼：在編輯區域編寫您的 Arduino 程式碼，使用 `setup()` 和 `loop()` 函數組織程式結構。

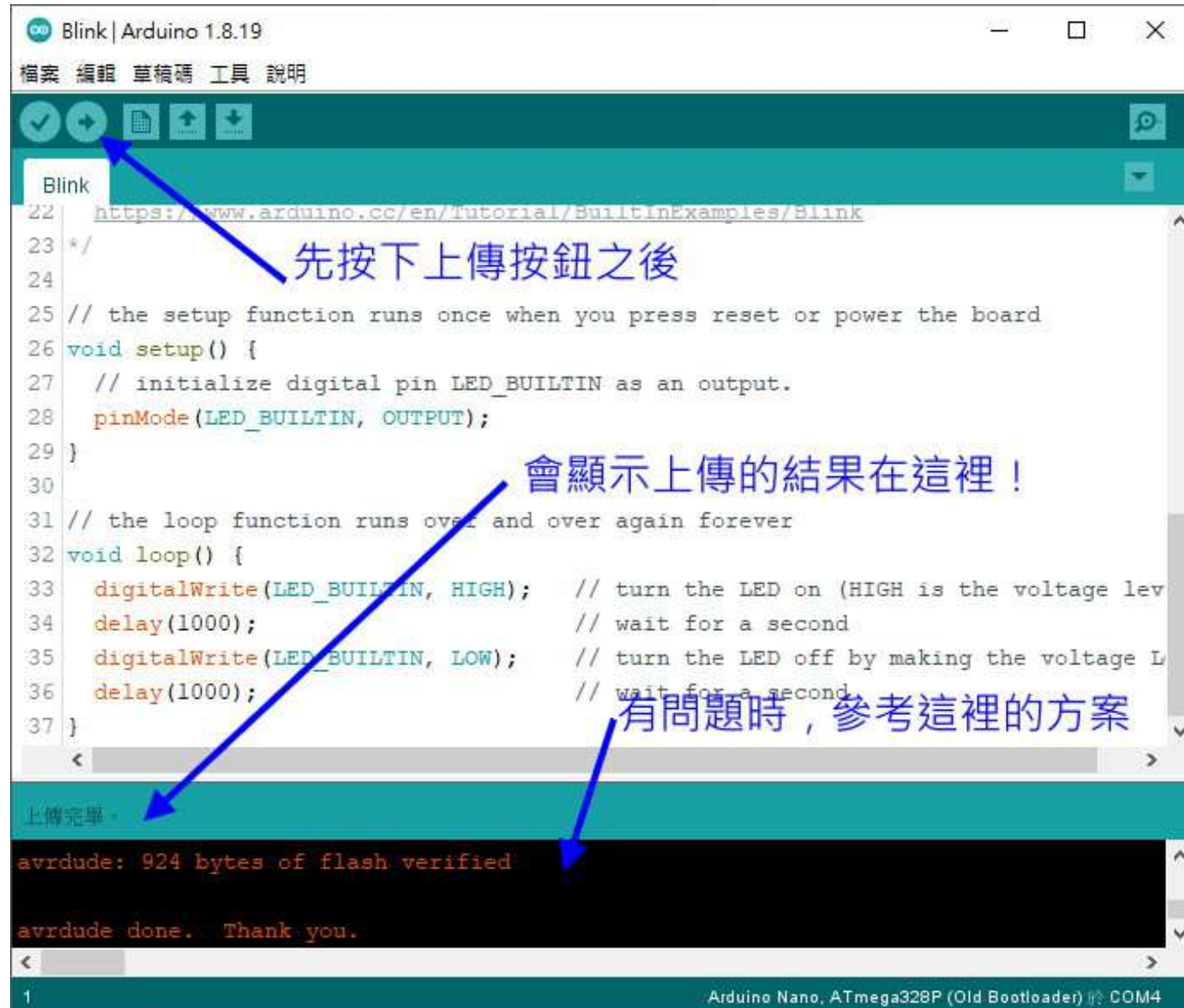
## 編碼例子：Blink

```
void setup() {  
    pinMode(LED_BUILTIN, OUTPUT);  
}  
  
void loop() {  
    digitalWrite(LED_BUILTIN, HIGH);  
    delay(1000);  
    digitalWrite(LED_BUILTIN, LOW);  
    delay(1000);  
}
```





驗證程式碼：點擊工具欄上的「✓」按鈕，Arduino IDE 將對您的程式碼進行編譯和錯誤檢查。



上傳程式碼：確保程式碼無誤後，點擊工具欄上的「→」按鈕，將程式碼上傳到 Arduino 開發板。



**完成以上所有步驟 = 電子零件正常運作**

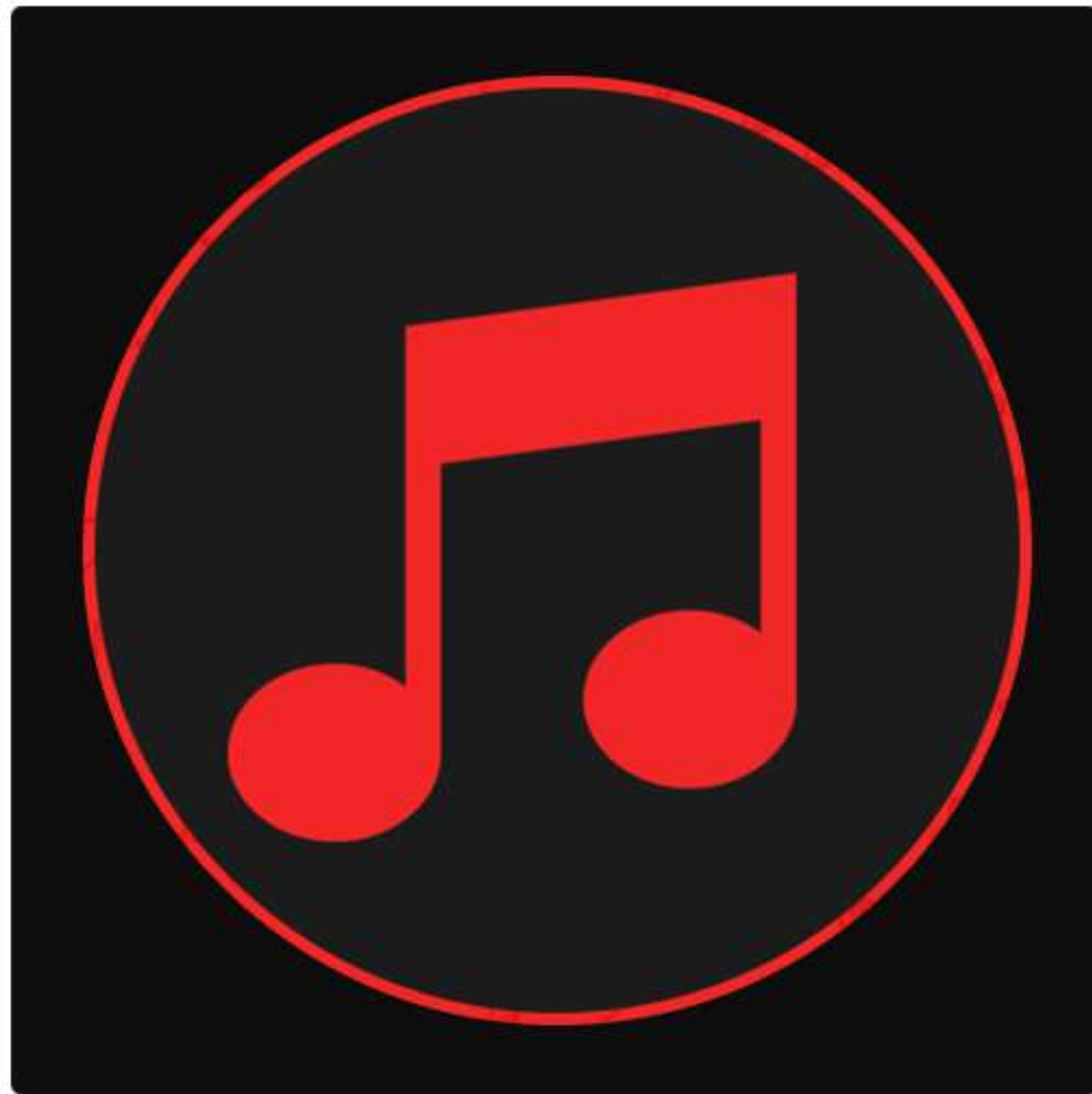
## 有發現到在 Blink 的例子中，編碼被分為兩個主要部份？

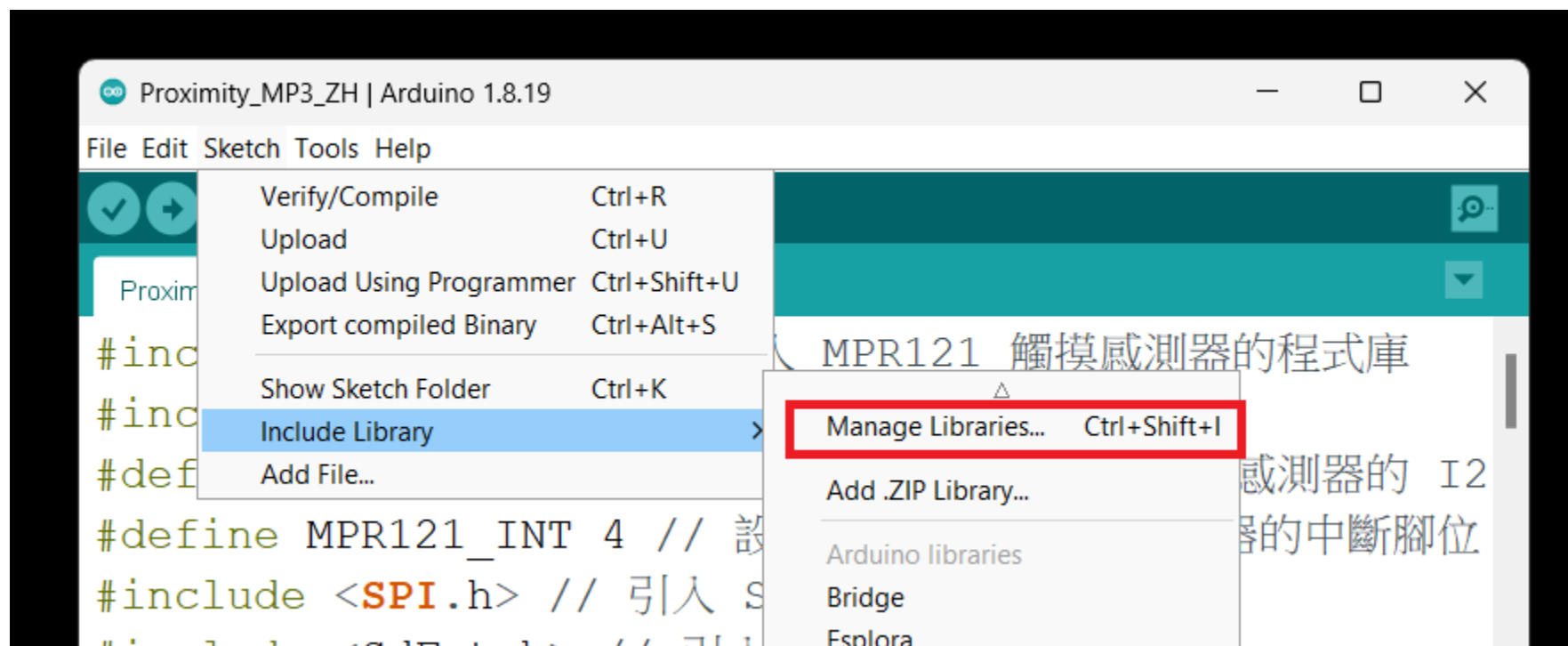
```
void setup() {  
    pinMode(LED_BUILTIN, OUTPUT);  
}  
  
void loop() {  
    digitalWrite(LED_BUILTIN, HIGH);  
    delay(1000);  
    digitalWrite(LED_BUILTIN, LOW);  
    delay(1000);  
}
```

- 分別被編寫於 setup(){ 編碼 } 及 loop(){ 編碼 } 的當中

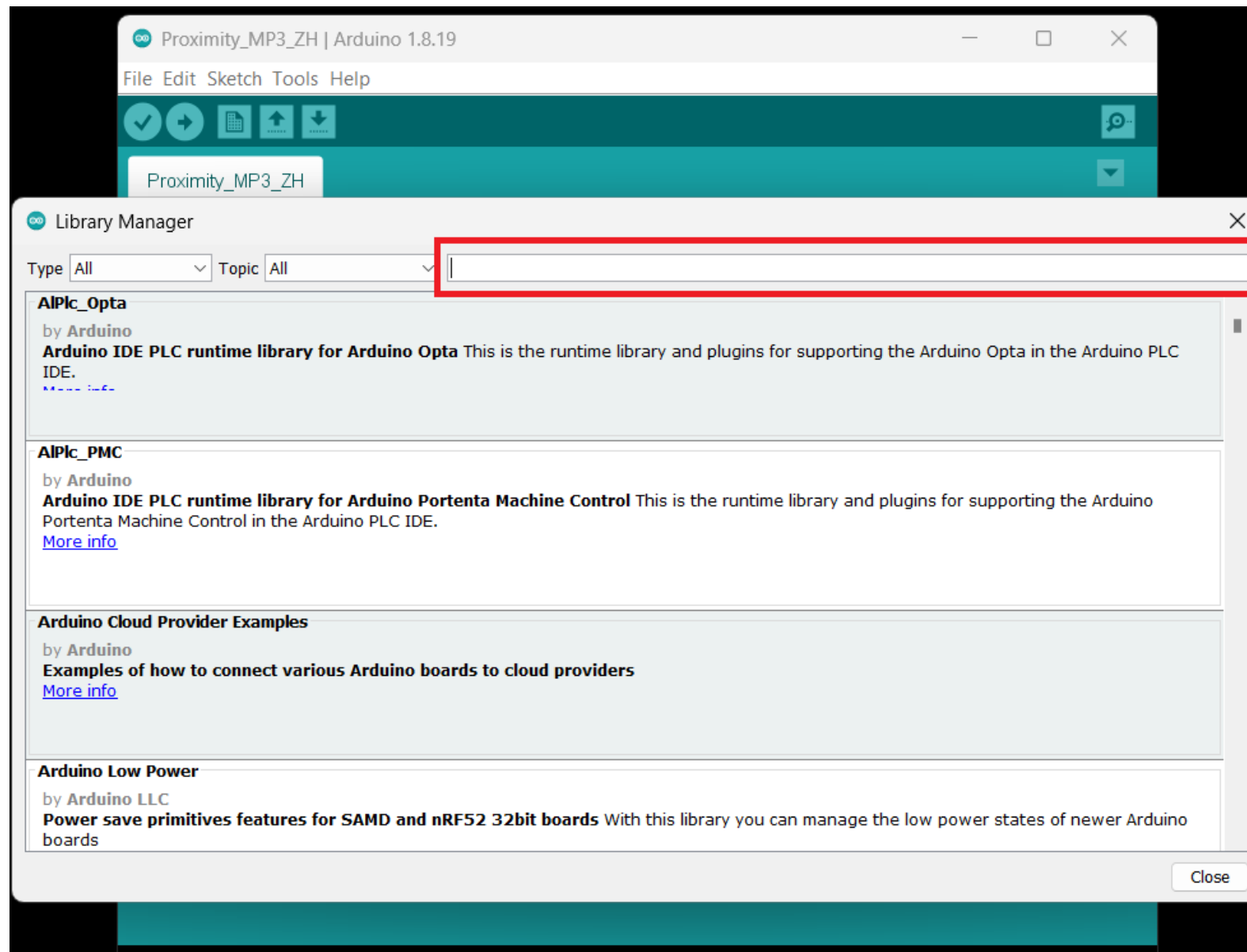
## 訓練 (一)

- 使用編碼寫一個簡單的 MP3 播放器

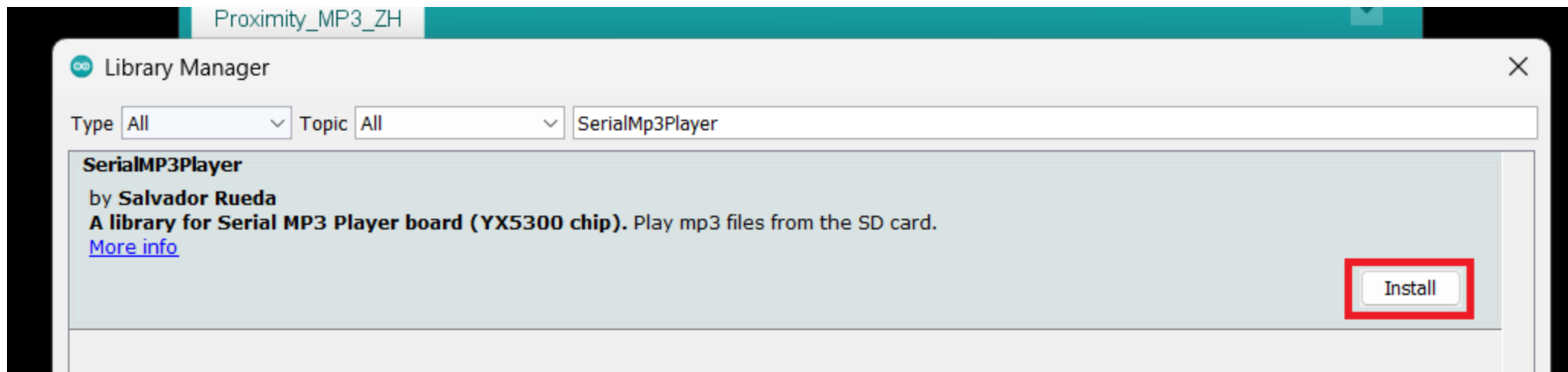




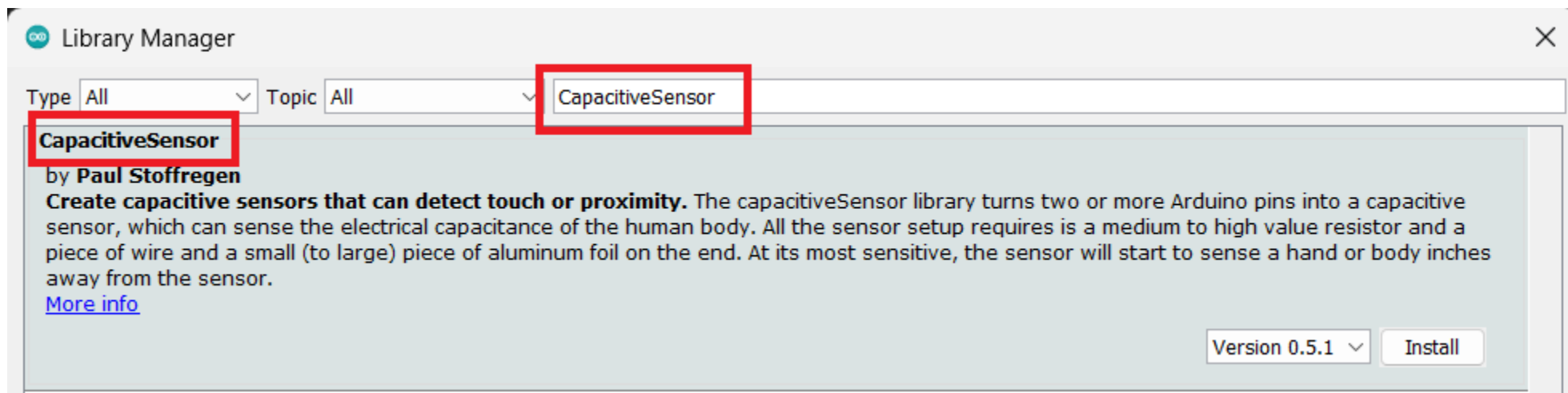
在 Sketch 工具欄下, 把滑鼠移到 Include Library,  
選取 Manage Libraries



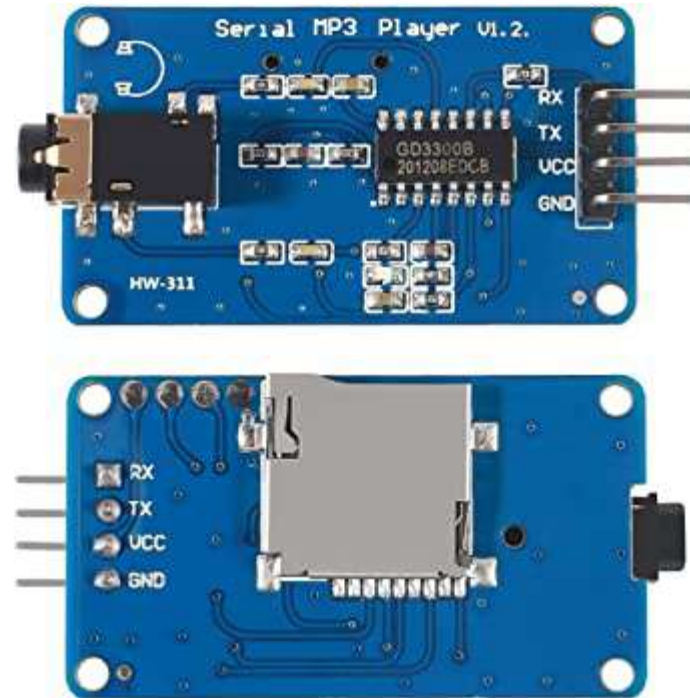
輸入 SerialMp3Player 搜索編碼庫



按 Install 安裝編碼庫

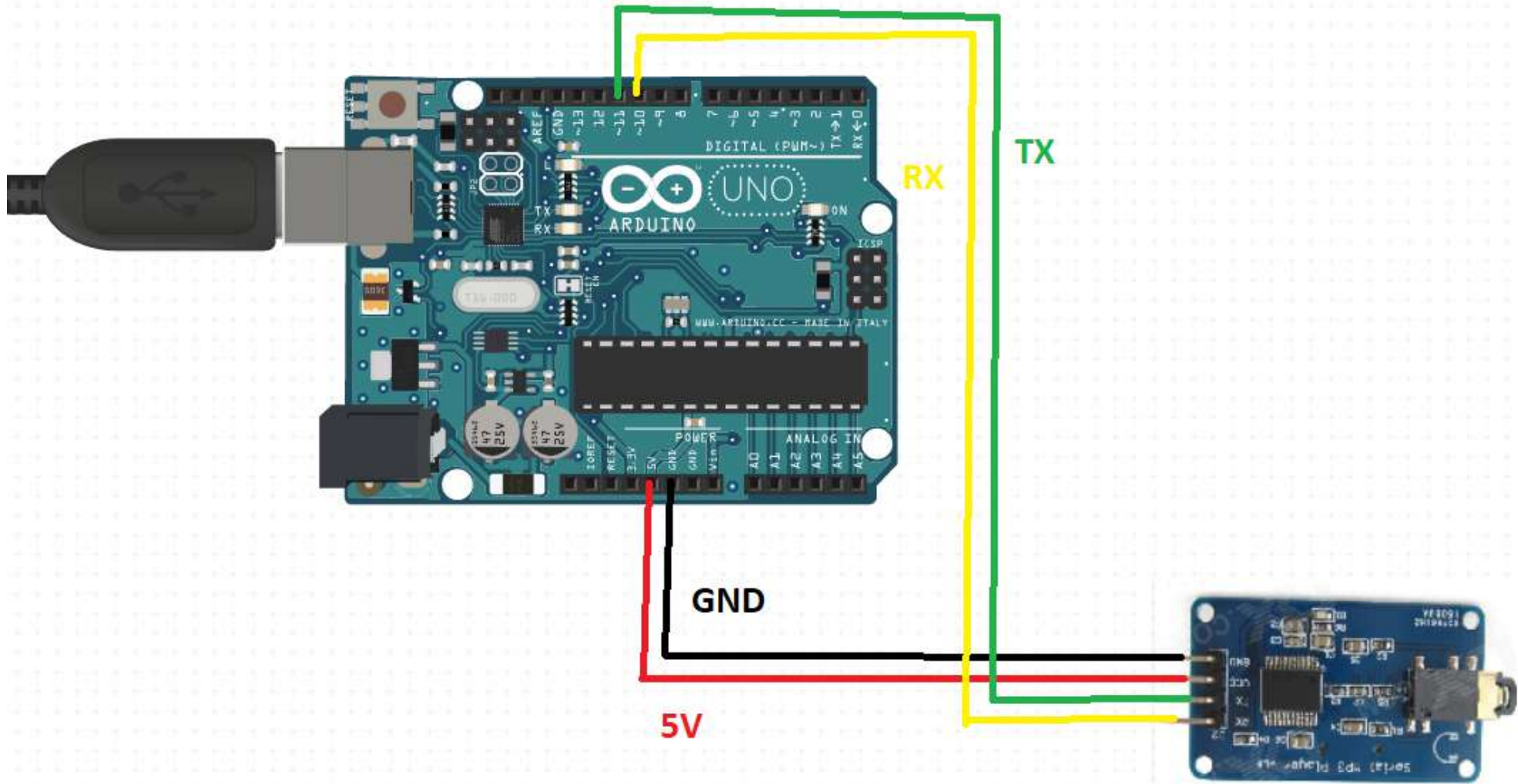


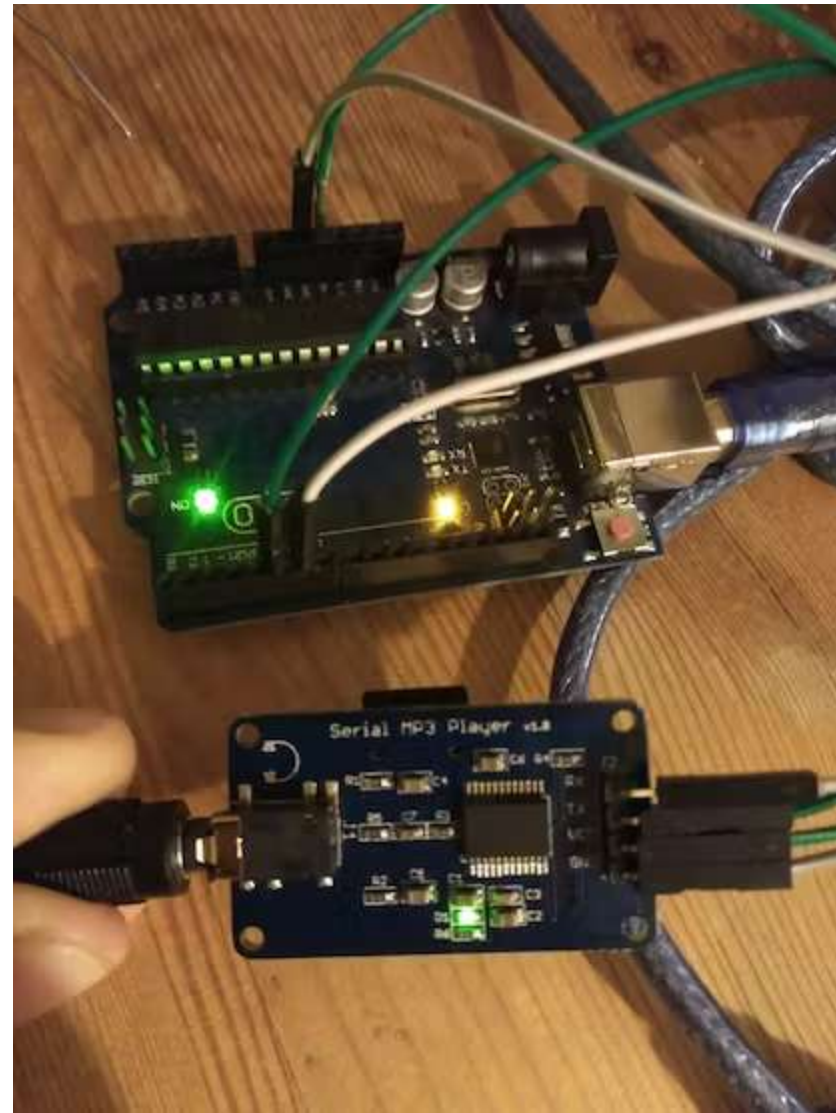
另外再安裝 CapacitiveSensor 編碼庫



把 Serial MP3播放模組YX5300/YX6300 接到 Arduino UNO







# 將以下的編碼複製到編輯區域

```
#include "SerialMP3Player.h"// 使用MP3版的編碼庫library

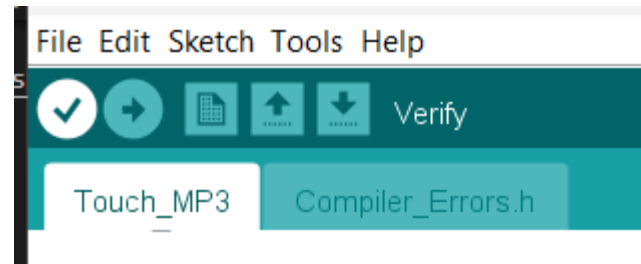
#define TX 11 //to MP3 board RX //定義ARDUINO TX到MP3 RX引腳連接
#define RX 10 //to MP3 board TX //定義ARDUINO RX到MP3 TX引腳連接

SerialMP3Player mp3(RX, TX);// 定義起動MP3相關的TX, RX

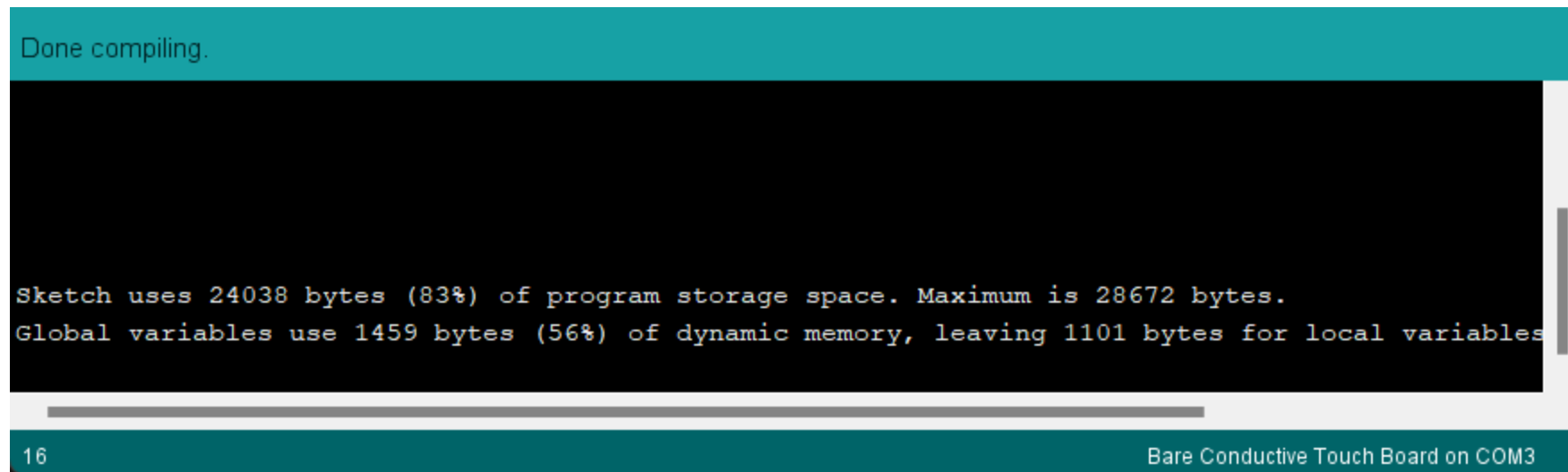
//設定：有電源起動時執行一次的程序
void setup() {
    Serial.begin(9600);        // 起動serial介面
    mp3.begin(9600);           // 開始MP3版的連接
    delay(500);                 // 等待起動
    mp3.sendCommand(CMD_SEL_DEV, 0, 2); //選取 sd-card
    delay(500);                 // 等待起動
    mp3.setVol(15); // 設定音量
    mp3.play(1);                //歌曲於SD CARD內的次序
}

//迴圈：處理器不停執行的程序
void loop() {

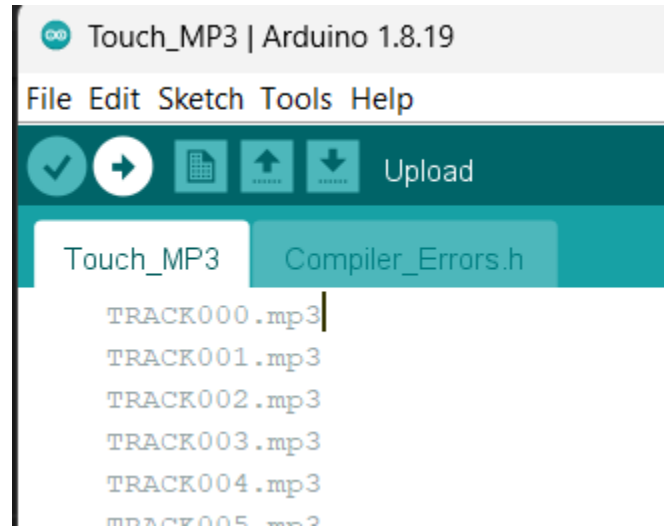
}
```



## 先編譯程式碼



如果系統控制台 (Console) 內沒有出現問題, 就可以做下一步。



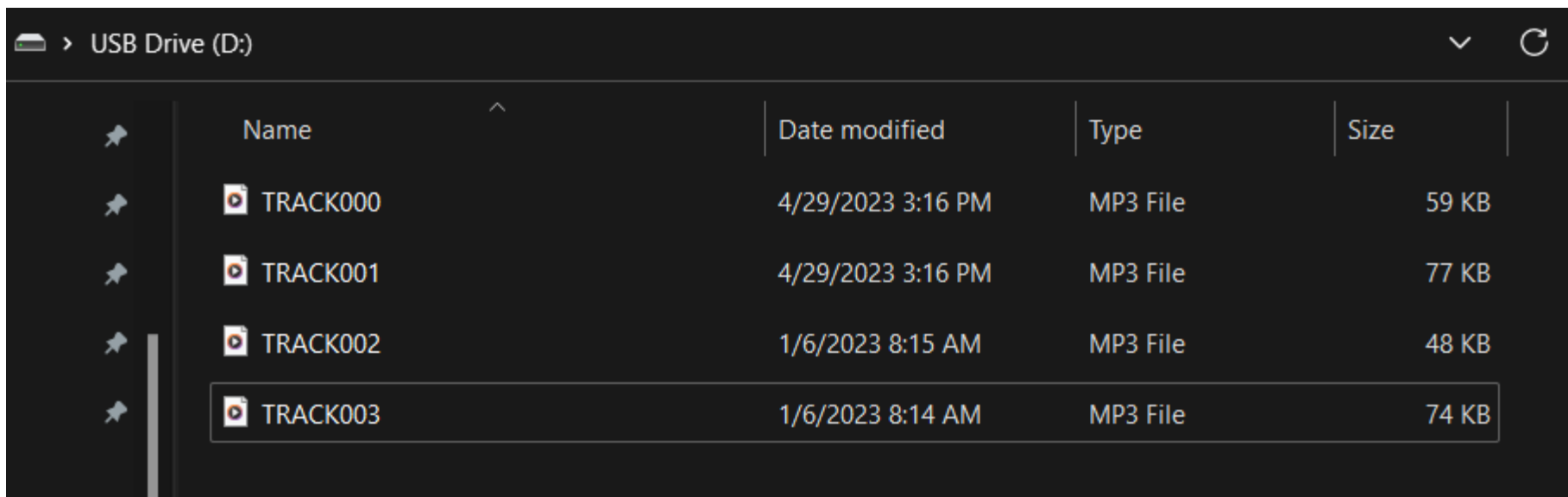
上傳程式碼





把準備好的聲音素材傳到 SD Card 內





USB Drive (D:)

Name	Date modified	Type	Size
TRACK000	4/29/2023 3:16 PM	MP3 File	59 KB
TRACK001	4/29/2023 3:16 PM	MP3 File	77 KB
TRACK002	1/6/2023 8:15 AM	MP3 File	48 KB
TRACK003	1/6/2023 8:14 AM	MP3 File	74 KB

把聲音檔案依次序修改名稱

TRACK000.mp3、TRACK001.mp3、TRACK002.mp3，以此類推。

## 在編碼部份的頂部

```
#include "SerialMP3Player.h" // 使用MP3版的編碼庫library
```

#include 是 Arduino 程式語言的一個預處理器指令，用於引入相關的程式庫，讓程式能夠使用庫中的功能。



```
#define TX 11 //to MP3 board RX //定義ARDUINO TX到MP3 RX引腳連接  
#define RX 10 //to MP3 board TX //定義ARDUINO RX到MP3 TX引腳連接  
  
SerialMP3Player mp3(RX, TX);// 定義起動MP3相關的TX, RX
```

Declare ( 宣告 ) 是指在 Arduino 程式中宣告變數、物件或函式，告知編譯器在後續的程式碼中使用這些元素。

## 有關 void setup() 及 void loop()

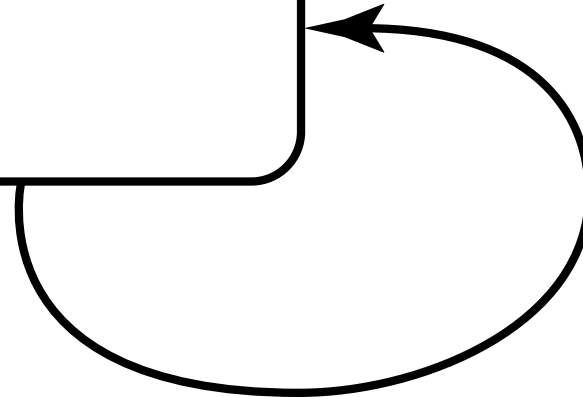
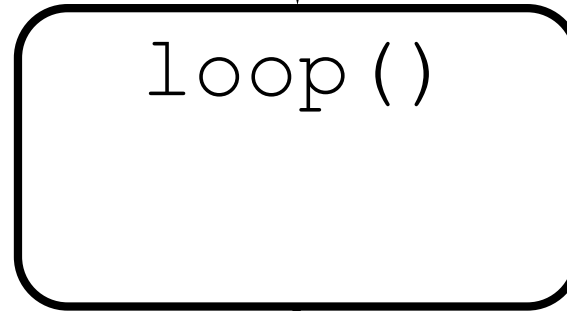
*void setup()* 是 Arduino 程式的一個函式，用於初始化設定，例如設定序列通訊、腳位模式、變數初值等。當 Arduino 板子啟動後，會**自動執行一次** setup()。

*void loop()* 是 Arduino 程式的另一個函式，用於執行主要的程式邏輯，例如讀取腳位數值、判斷條件、控制輸出等。loop() **會一直執行**，直到板子關機或重置。

Execution starts here  
Execute once.



Execute repeatedly.



**把聲音檔案準備好，下一節將會組裝硬件。**

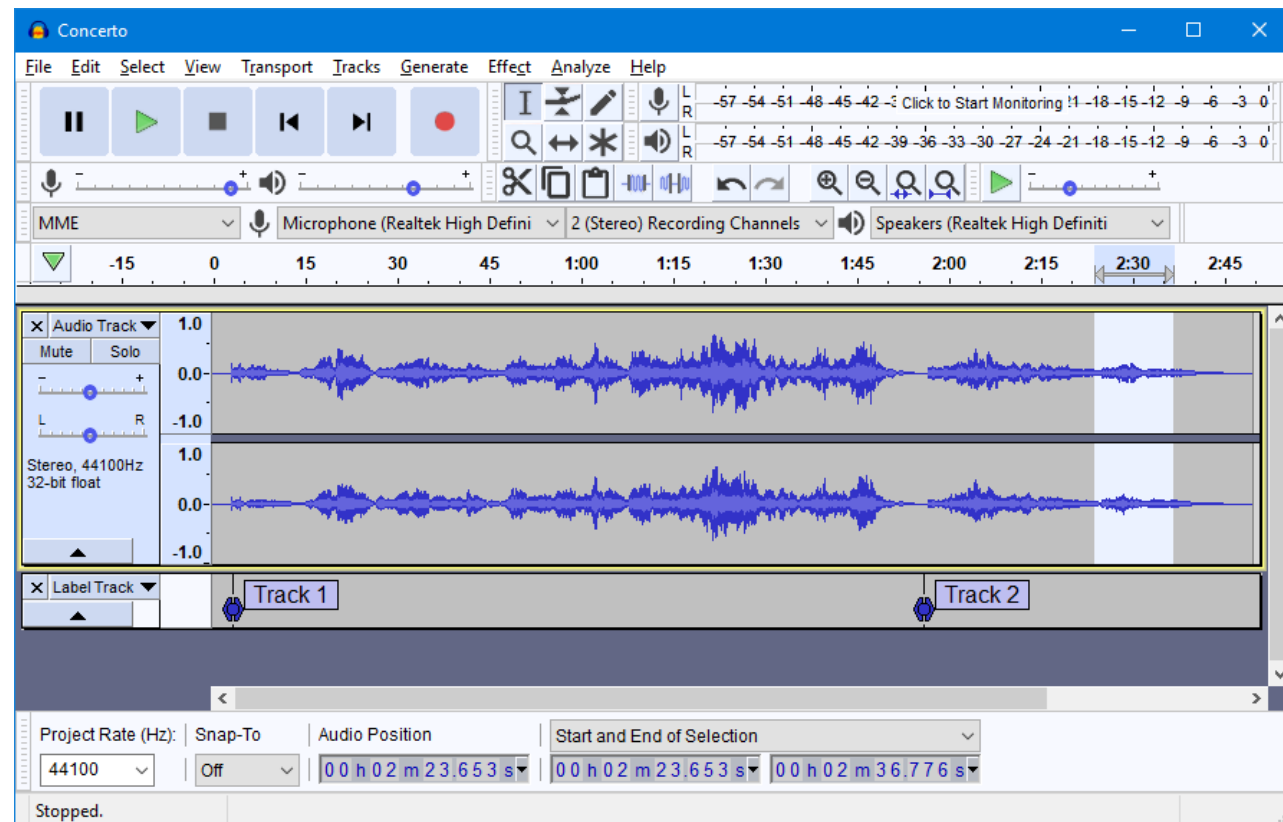
# 甚麼是 數位音訊工作站 (Digital Audio Workstation ,DAW) ?

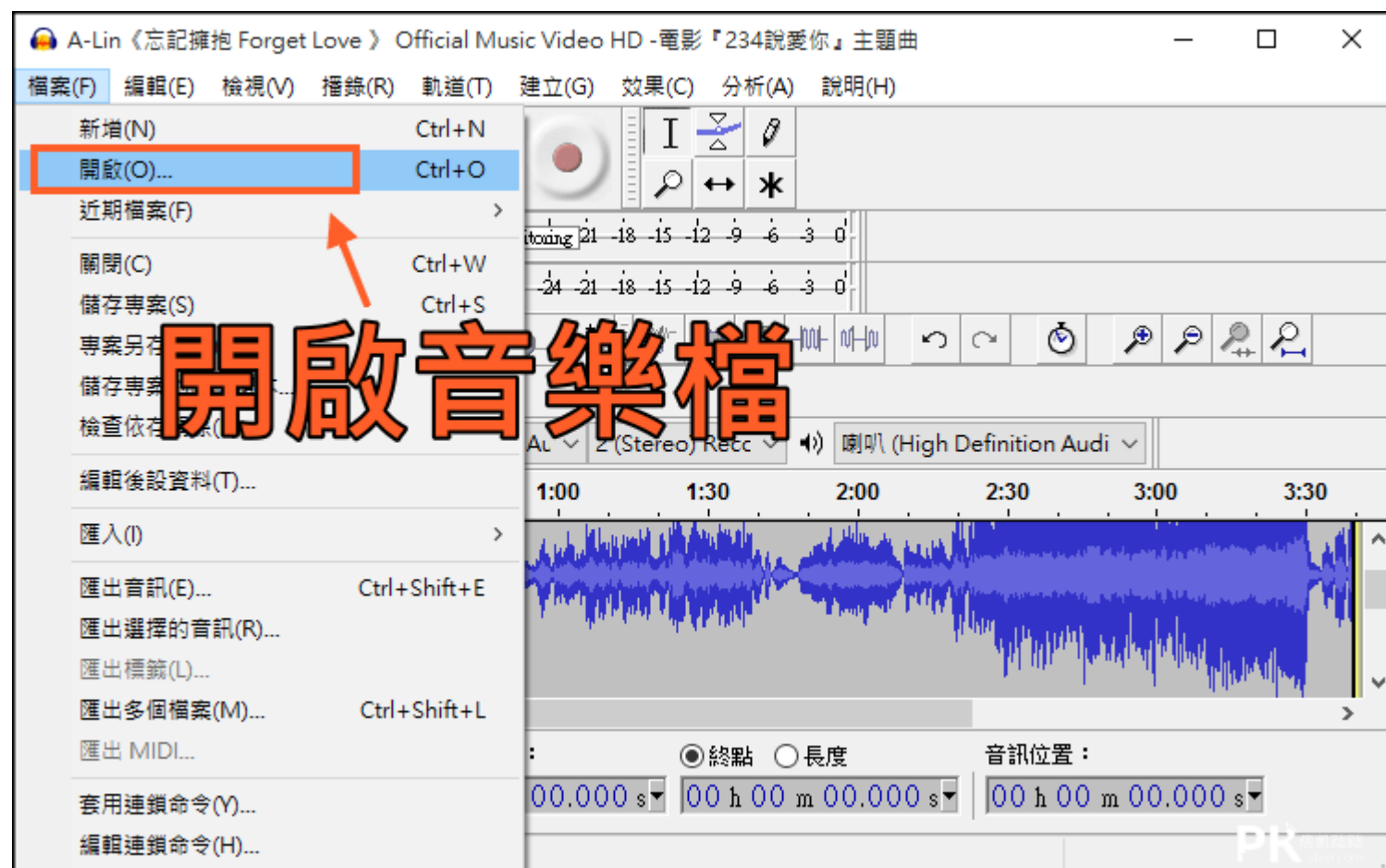
音訊工作站是一種專業數位音頻製作軟體，可用於錄音、混音、編曲、編輯及後製處理等音樂製作工作，適用於聲音藝術家、音樂製作人、錄音師、唱片製作人等。

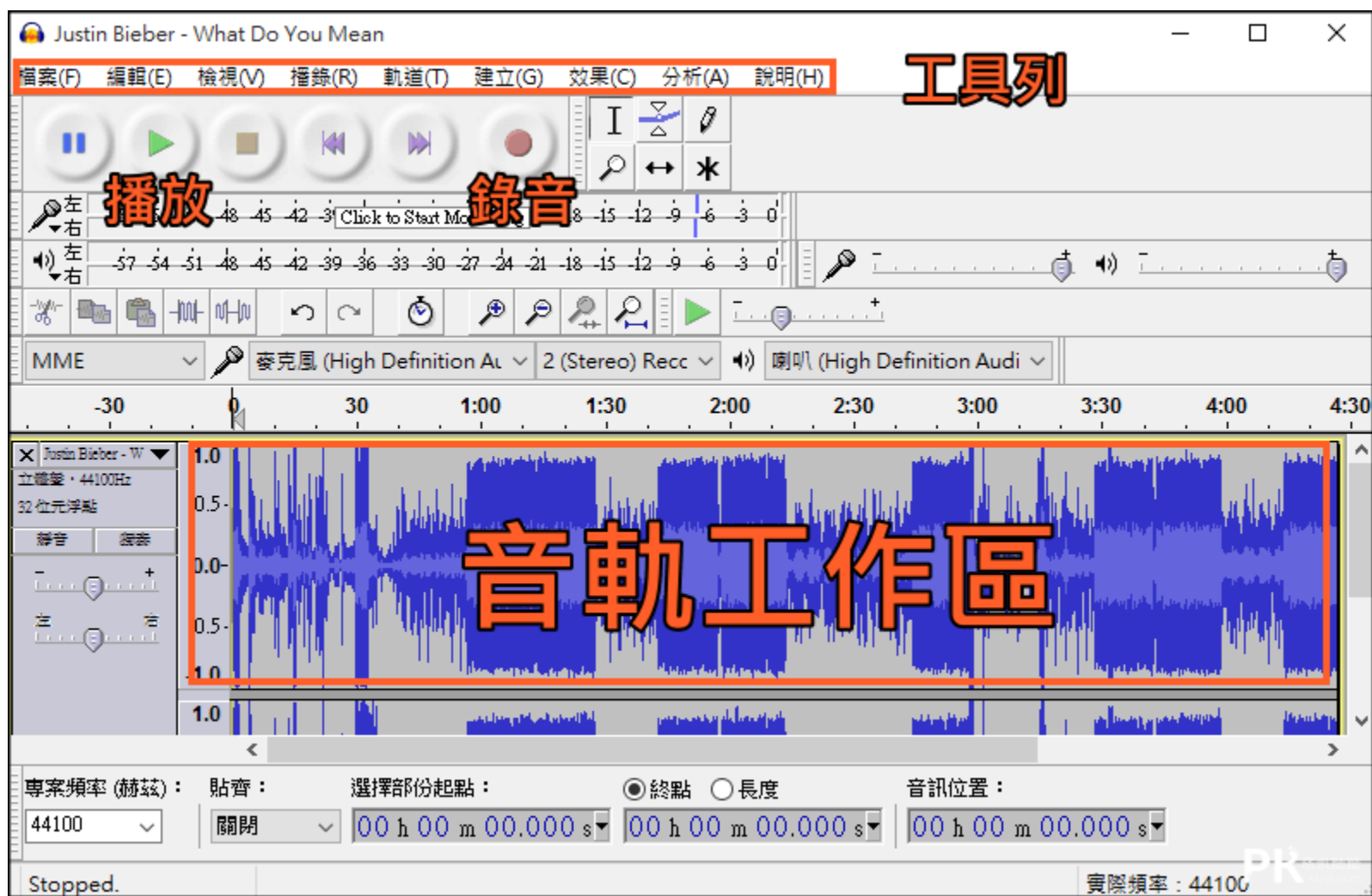


# 我們會使用開源軟件

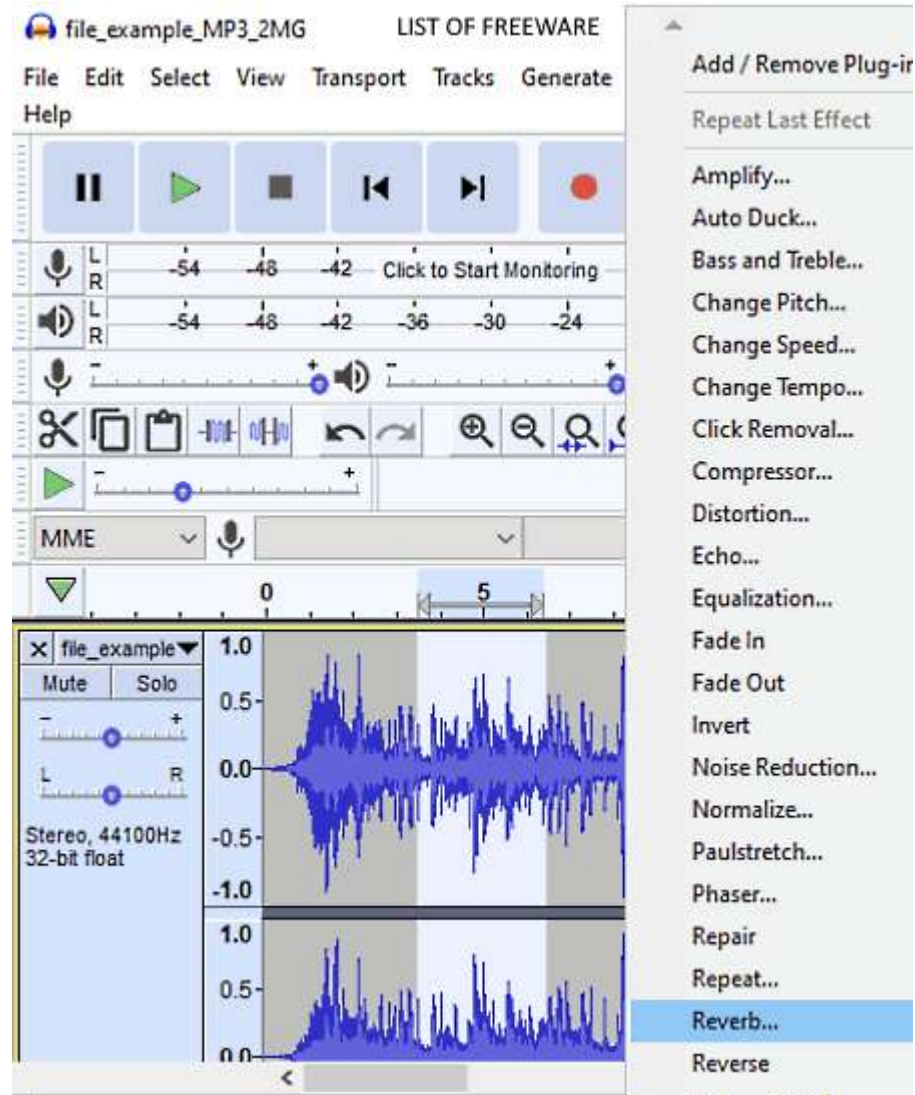
## Audacity

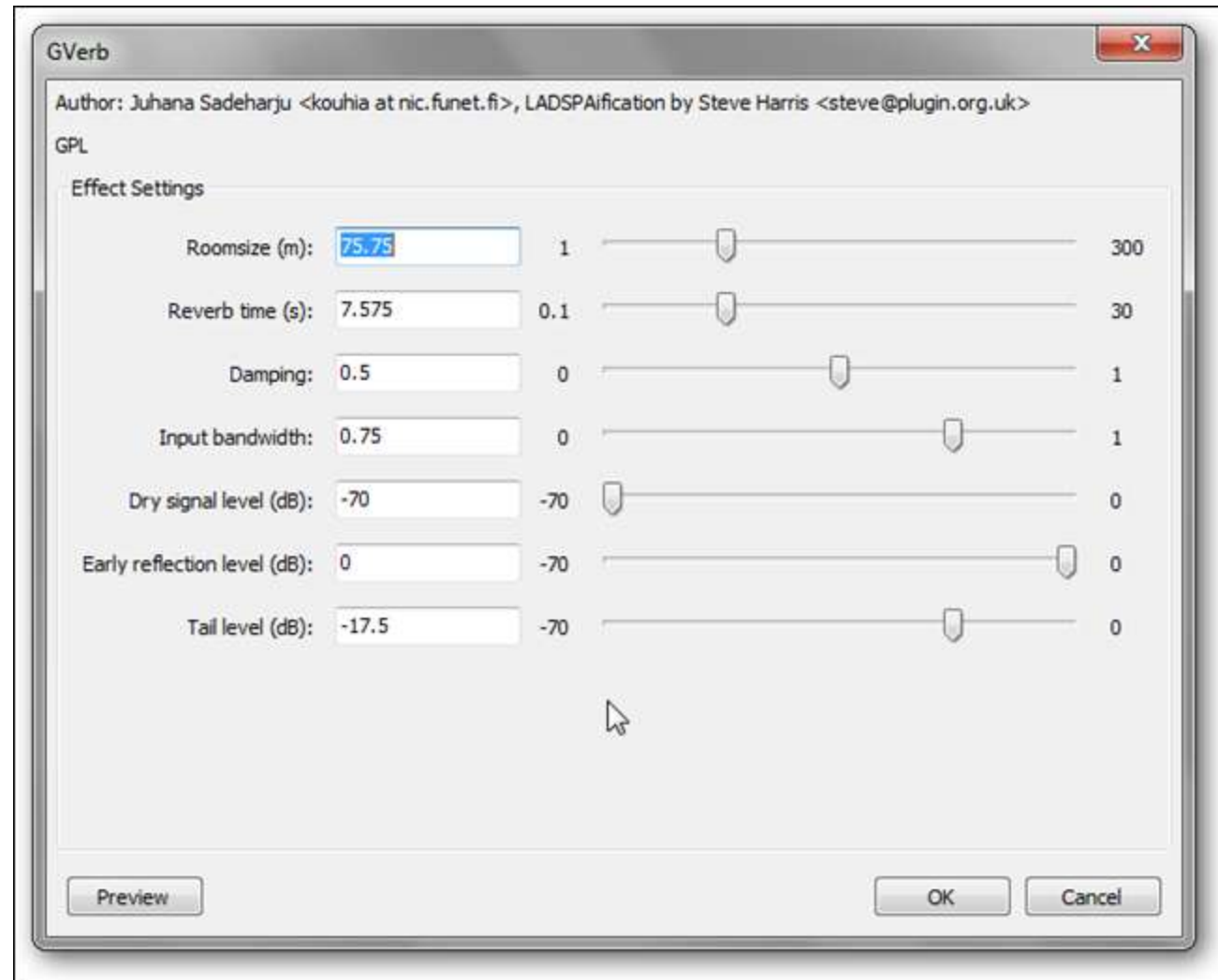


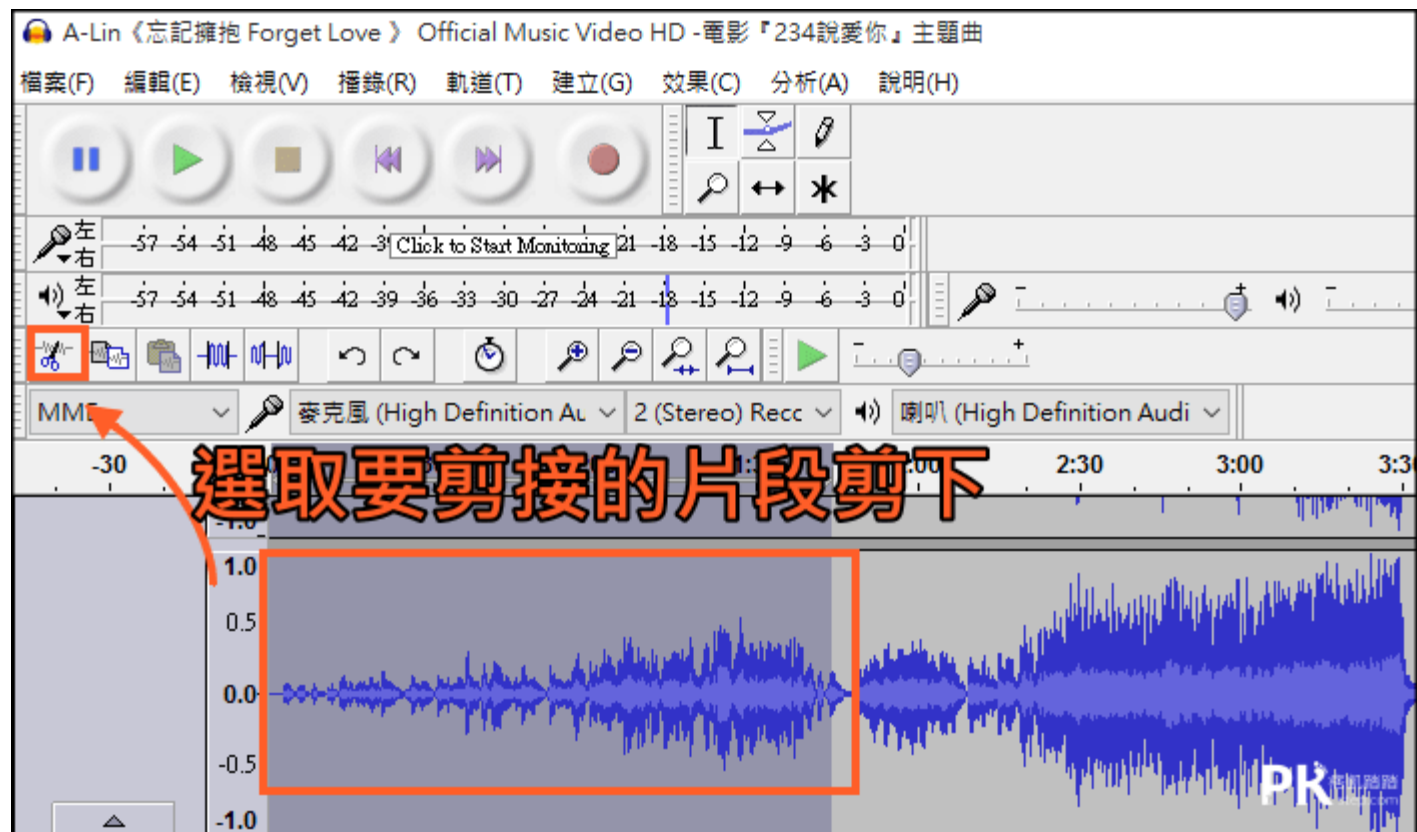


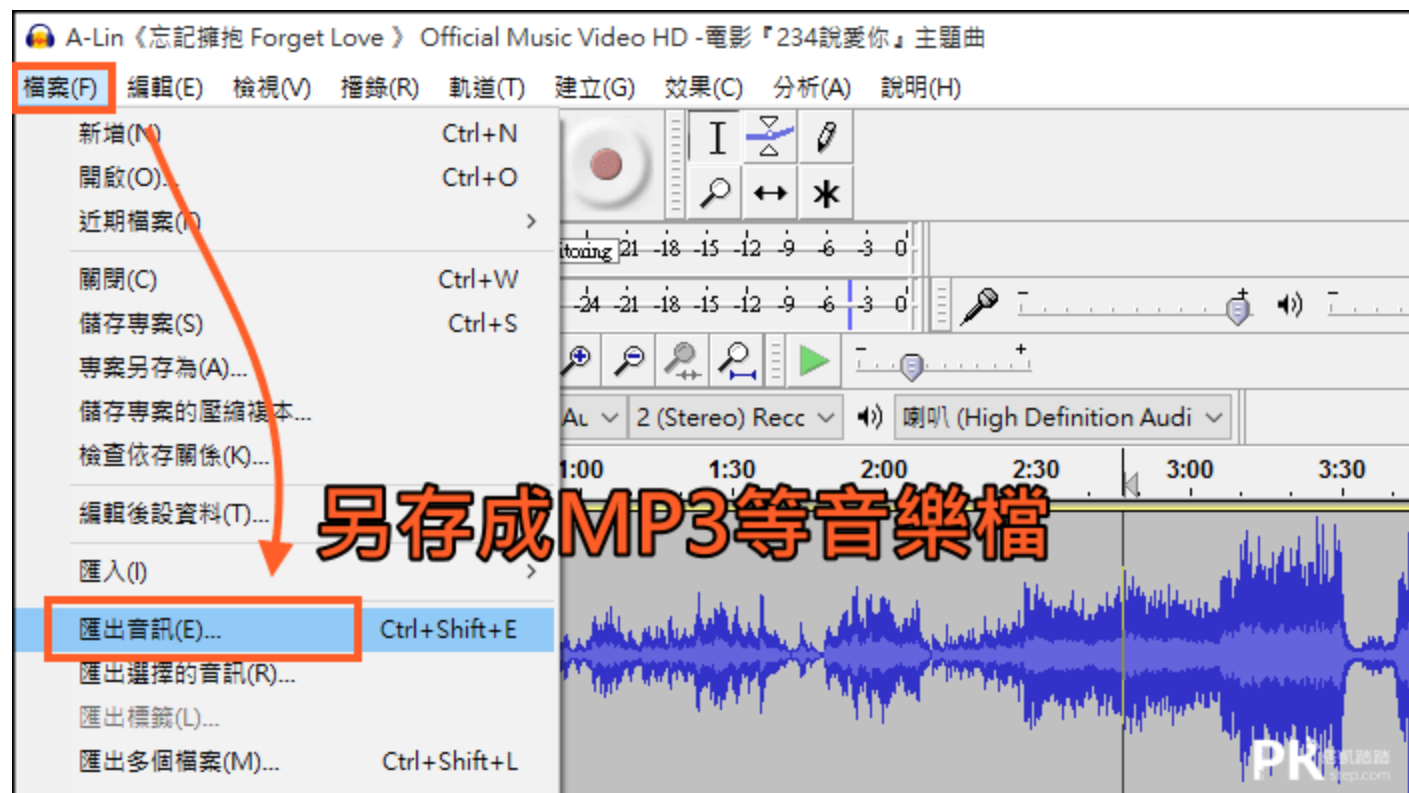












第四節完成

下一節課會有關

- 完成一半的編碼部份
- 硬件組裝