

- WELCOME TO -

賽馬會科藝共融計劃

JC-Project-IDEA

Day01

上午

Ready Your Material

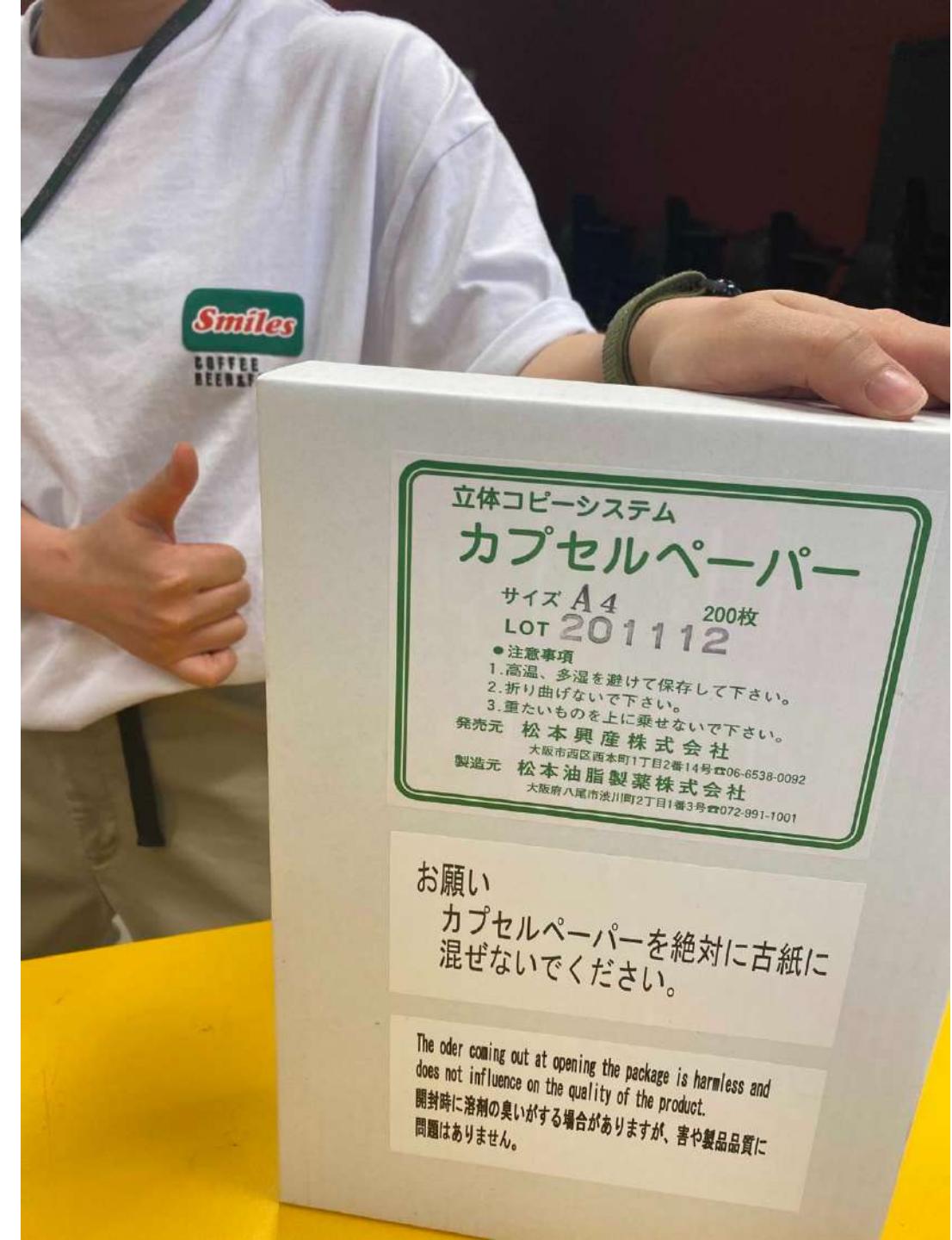
老師們需要集齊以下物料：

- 紙 及 紙膠帶

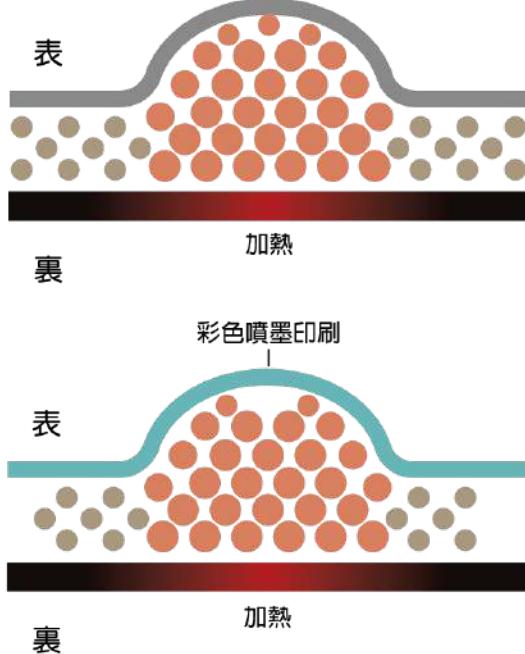
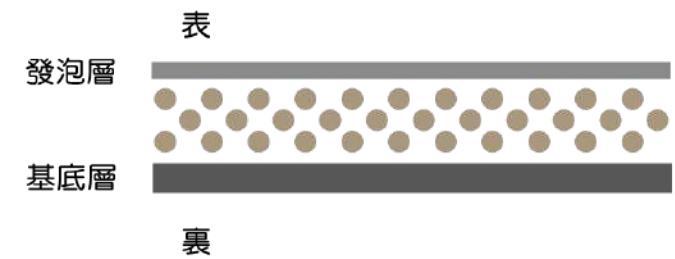
有關於物料選用

小知識

- 我們將會使用的紙張名叫「熱敏式專用紙」
專用紙張的表面有一層遇熱即會發泡的塗層。印刷時，熱敏印刷頭會從專用紙的背面(基底層)將印字的所在位置進行加熱，使發泡層膨脹，紙張的表面即製造出各種精細的立體線條及圖形。



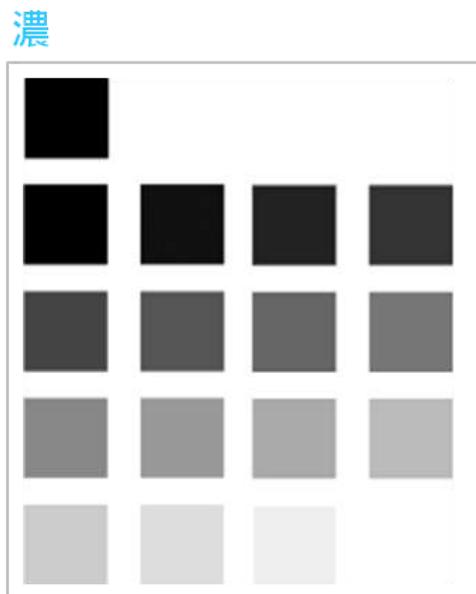
藉於發泡紙內面加熱，可印製出精細的立體圖形。



與彩色噴墨式印刷機搭配使用，
可呈現真實性更高、更美觀的圖形。

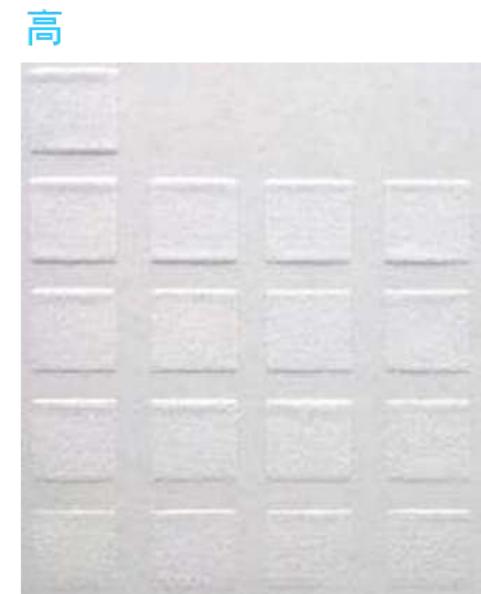
※ 彩色印刷機需另備

印刷檔案

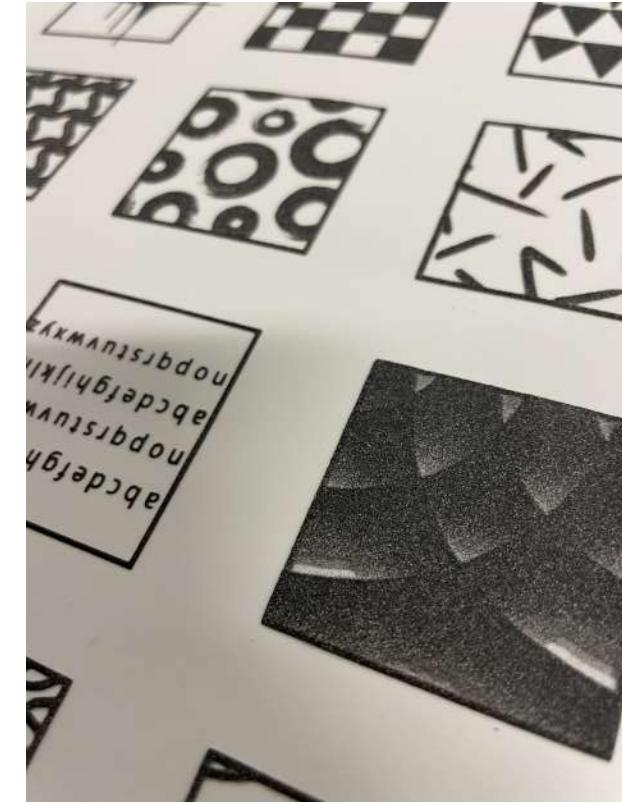
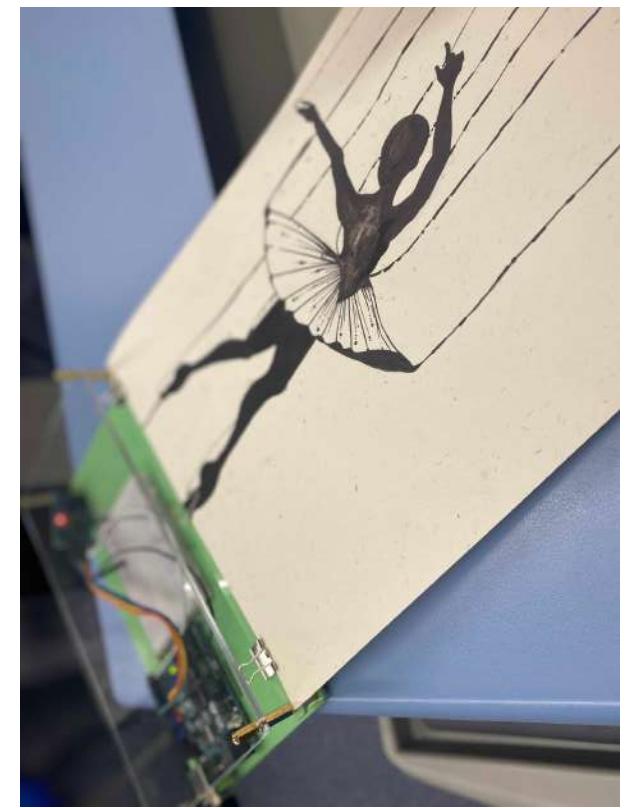


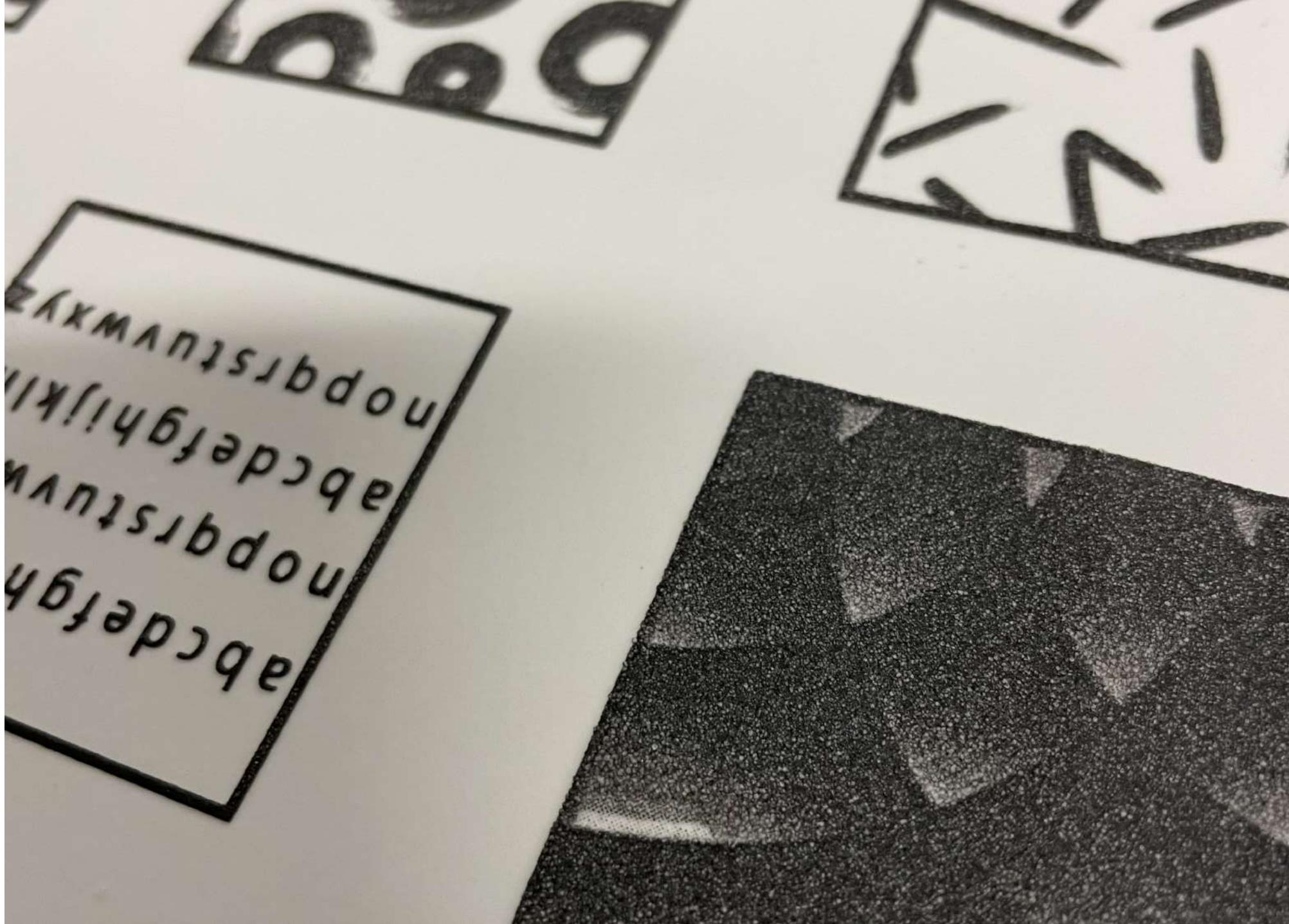
淡

印刷結果



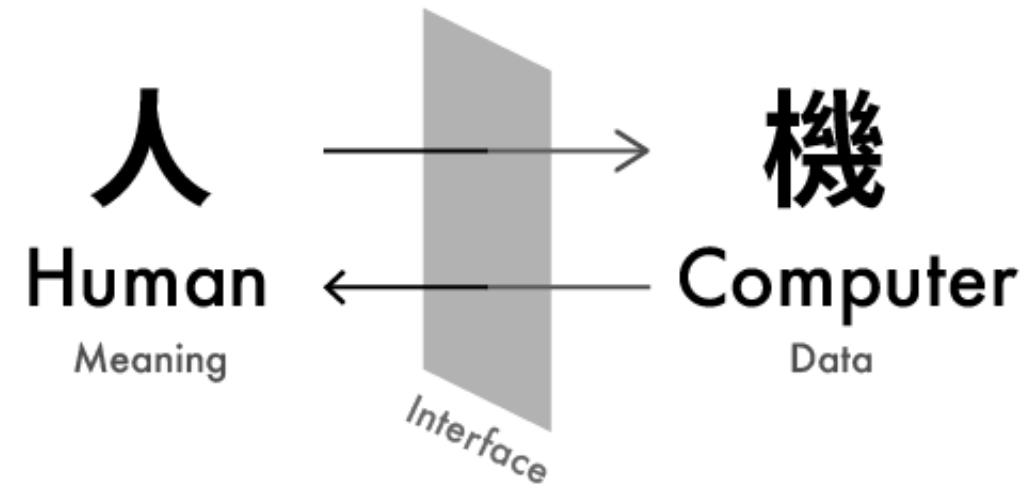
立體化





動手繪畫前先了解 ~

使用者體驗 (User Experience) 與
人機互動介面 (Human–Computer
Interface)



Materiable

美國 MIT Media Lab 的 Tangible Media 實體媒介實驗室，嘗試各種材料與方法，超越平面的螢幕，讓人們可以接觸實體三維的物件，就像是立體的顯示器，直接操作其代表的數位資訊。

[https://vimeo.com/165798784?
embedded=true&source=video_title
&owner=11720996](https://vimeo.com/165798784?embedded=true&source=video_title&owner=11720996)



MATERIALE

RENDERING DYNAMIC MATERIAL PROPERTIES IN
RESPONSE TO DIRECT PHYSICAL TOUCH WITH SHAPE
CHANGING INTERFACES

Ken Nakagaki, Luke Vink, Jared Counts, Daniel Windham,
Daniel Leithinger, Sean Follmer, Hiroshi Ishii



tangible
media

Please Touch!

這是一個包容性的藝術界面，挑戰觀眾與藝術品的傳統關係。它打破禁忌，邀請觀眾觸摸藝術品。觀眾參與其中，通過數字製作的觸感浮雕、計算機視覺和多媒體剪輯，探索藝術家的世界。

[https://www.youtube.com/watch?
v=uDPI0ZNTfaM](https://www.youtube.com/watch?v=uDPI0ZNTfaM)





有關裝置結構

先完成畫面部份

**繪畫時候都要顧及以後會加入的聲音元素 !!
以及經過加熱後的質感 !!**





畫面



電路

完成畫面後，建議老師可以把聲音內容先標記好。

城市聲

交通工具聲

小鼓聲

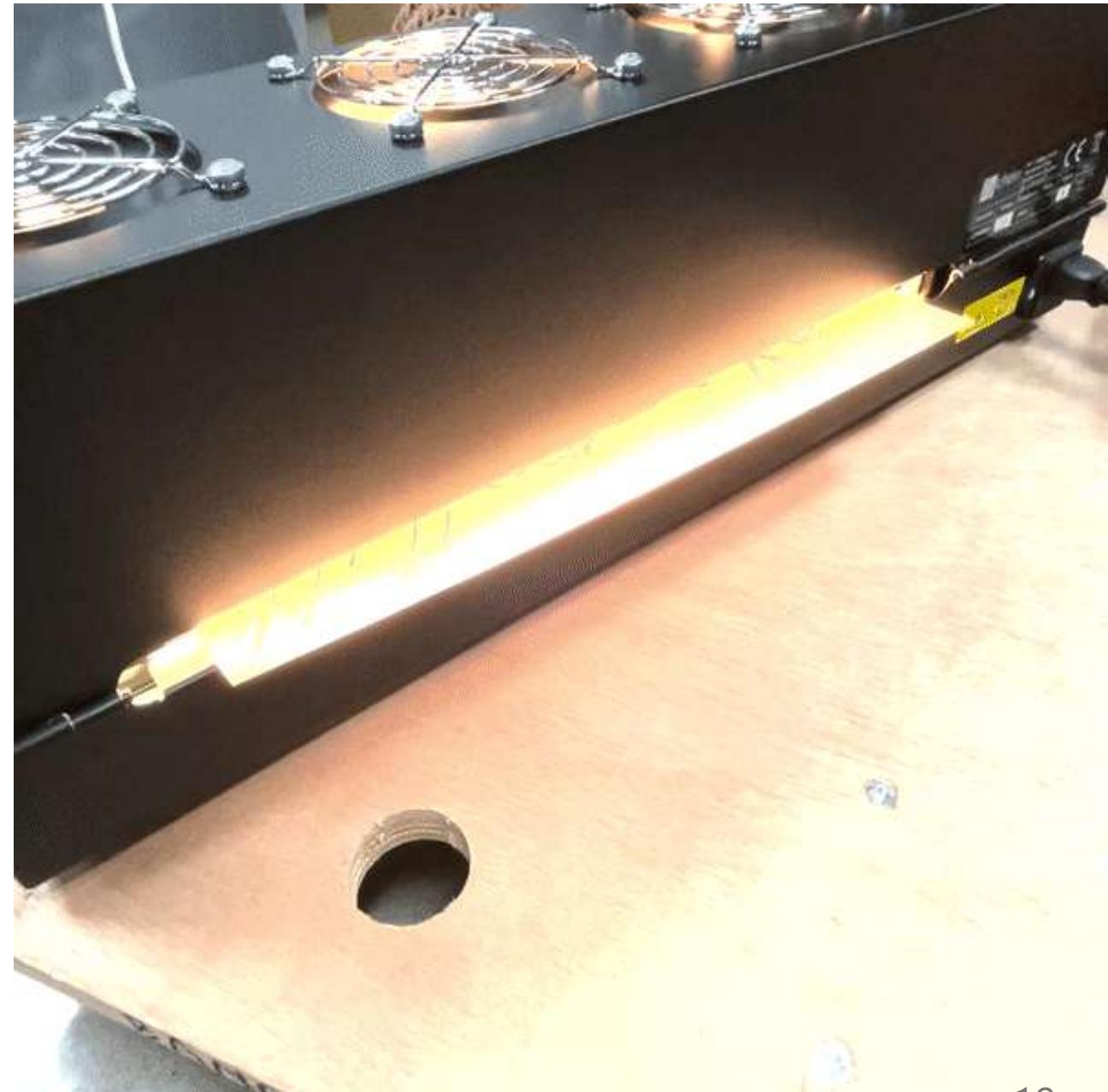
歡呼聲

歌聲



**完成畫面後，請老師把
作品交給助理。**

我們會先用印刷機把作品複製，下午
再放到加熱機器。



上午部份完成

下午部份會有關

- 使用導電油墨製作電路
- 聲音主題 擬聲音效 (Foley) 及 現場錄音 (Field Recording)

需要準備任何可發聲的日常用品

例如：保鮮紙、紙杯、各種文具

- WELCOME TO -

賽馬會科藝共融計劃

JC-Project-IDEA

Day01

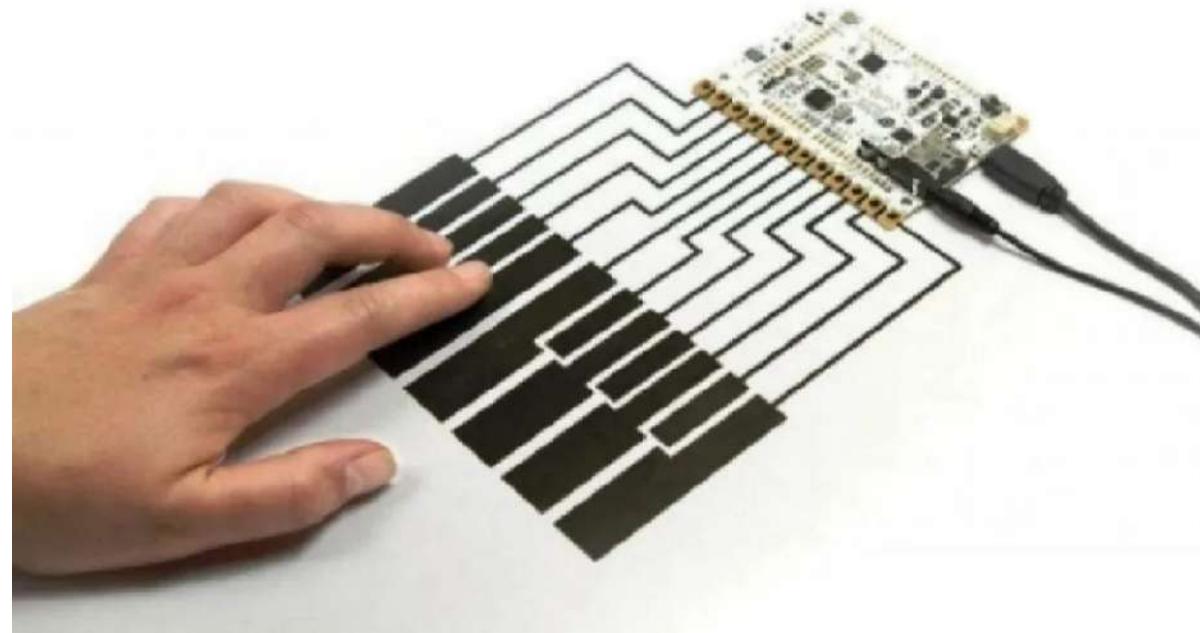
下午

準備加熱!!!!



簡介 - Conductive Ink (導電油墨)

導電墨水是一種具有導電性能的特殊墨水，通常由導電顆粒（如銀、碳）和樹脂基質組成。它可用於創建彈性、可印刷的電子電路，擴展應用領域。



在開始前，先補充基本知識。

以免



Circuit 電路

電路是由電子元件組成的路徑，讓電流能夠流通，用於控制、處理、傳輸和存儲電信號和電能。是電子技術中的基礎。

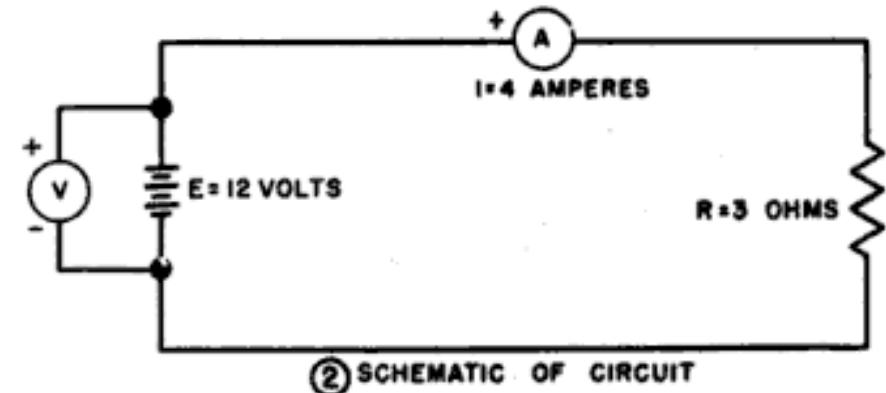
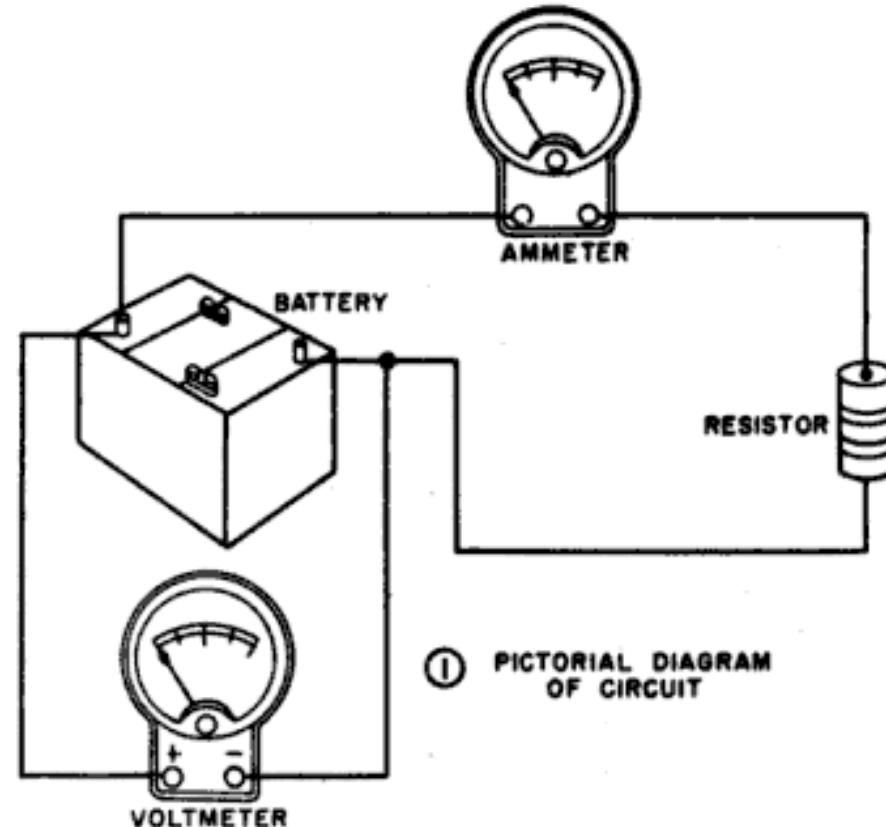
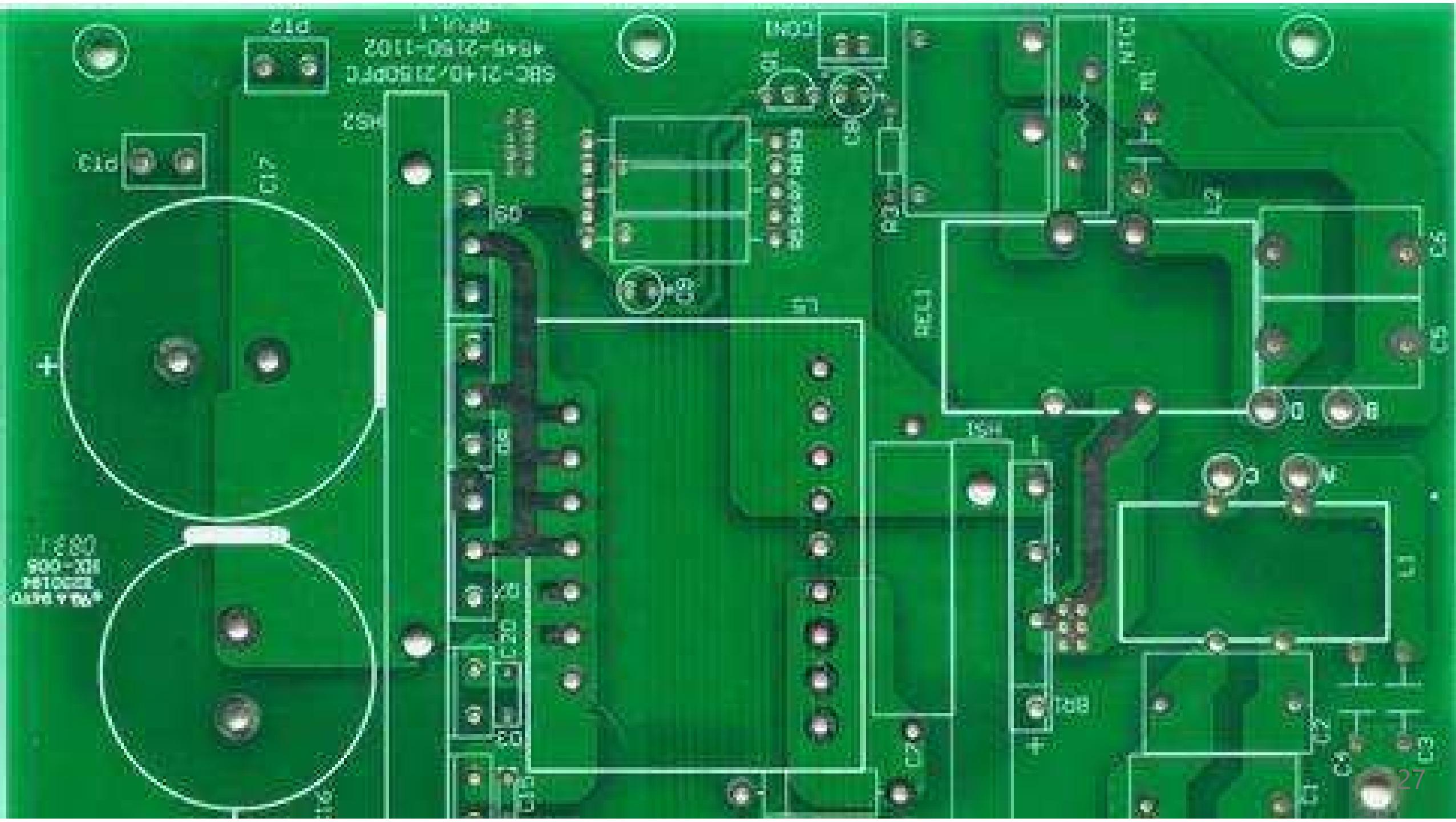
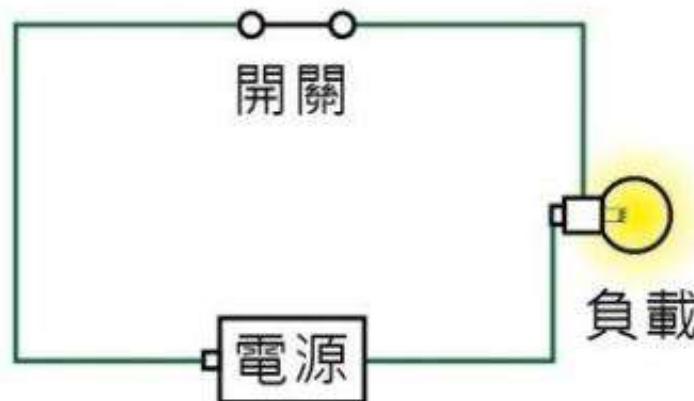


Figure 48. Diagram of a basic circuit.

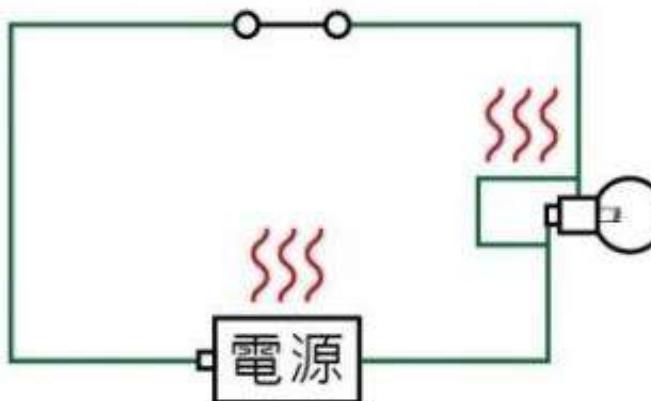


你所不知道的秘密

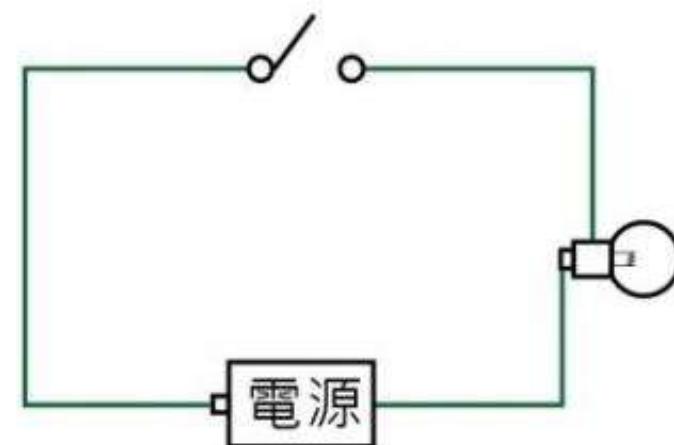
通路就是按下開關(on)，線路上通電，電器就能運作；短路就是在電路中直接讓正、負極接觸，產生的電流極高，會發熱甚至著火；斷路是電路呈斷開狀態(off)，此時電力系統無法構成迴路不能通電，電器就無法運作。



通路



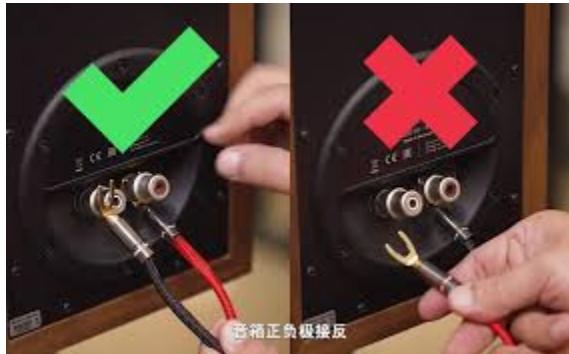
短路



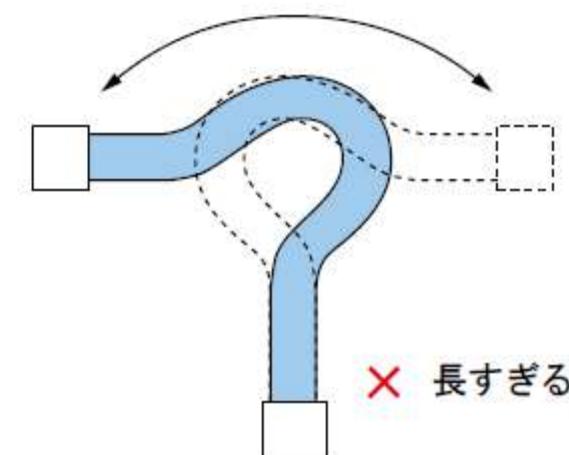
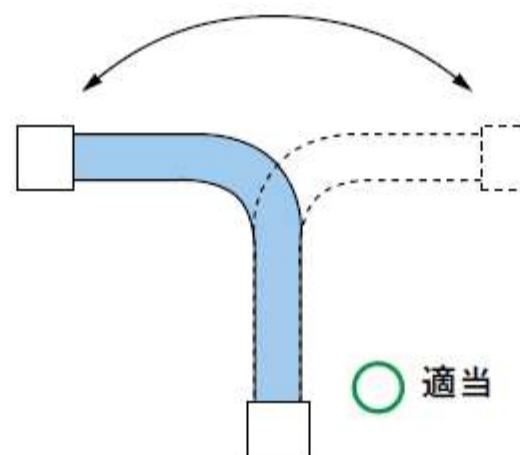
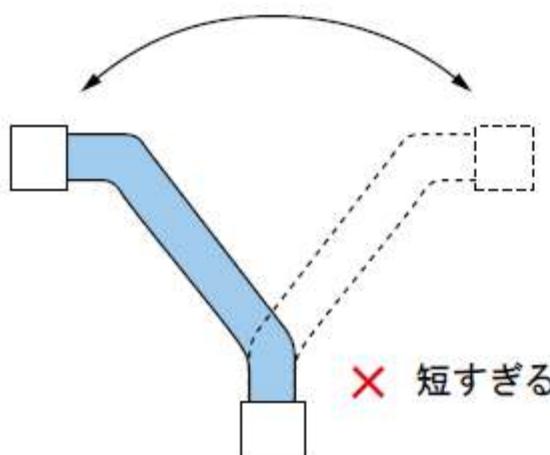
斷路

在接下來需要留意的事項：

- 插電前檢查電路是否已經接通
- 小心正負極接反導致電子元件損壞



- 不要過度屈曲電線



先在另一張紙上做好電路分佈計劃

城市聲

交通工具聲

小鼓聲

歡呼聲

歌聲





Now, Draw the Circuit !



現在檢查電路

現在工作人員會幫大家簡單測試一下電路運作

為明日課堂準備聲音素材

城市聲

交通工具聲

小鼓聲

歡呼聲

歌聲



wouter.hisschemoller Messages Settings Log Out Upload Sounds

search sounds

Home Sounds Forums People Help

Sounds

Random sound of the day

RETRO LASER SHOT_02.wav ★★★★
Mattix
January 6th, 2018
398 downloads
0 comments

-00:00

action gun shooter weapon target laser shoot game zombie fire videogame sound spaceship

Latest sounds

Lion roars 1.wav ★★★★
eardeer
February 8th, 2019
0 downloads
0 comments

-01:33

Male lions roaring. Recorded at UkuTula Game Reserve, South Africa. Mics used: Rode NT1A stereo pair in ORTF position Recorder: ...
ukutula lions lion ambiance south-africa africa roar nature field-recording

1 more sounds from eardeer in the last 48 hours

THE LIBRARY by soundpocket 聲音圖書館

sound out your support · click to make a sound donation
聲援我們 按此作聲音捐獻

Project info 本館簡介

Listen 來聽

editorial 編輯室

sound scoop 細聲公

main 大觀

meanderings 岑尺

網上聲音資源

**除了於 網上下載聲音資料 或 在城市錄音之外
還有甚麼方法取得聲音？**

甚麼是 擬聲音效 (Foley) ?

擬聲音效 (Foley) 是指使用人聲或物品模擬出特定聲音，用於影片、遊戲、廣告等多媒體製作，營造更真實的聲音效果，增加觀眾的身臨其境感受。

https://www.youtube.com/watch?v=_6tkhCNv9Ek







Foley Artist 的工具

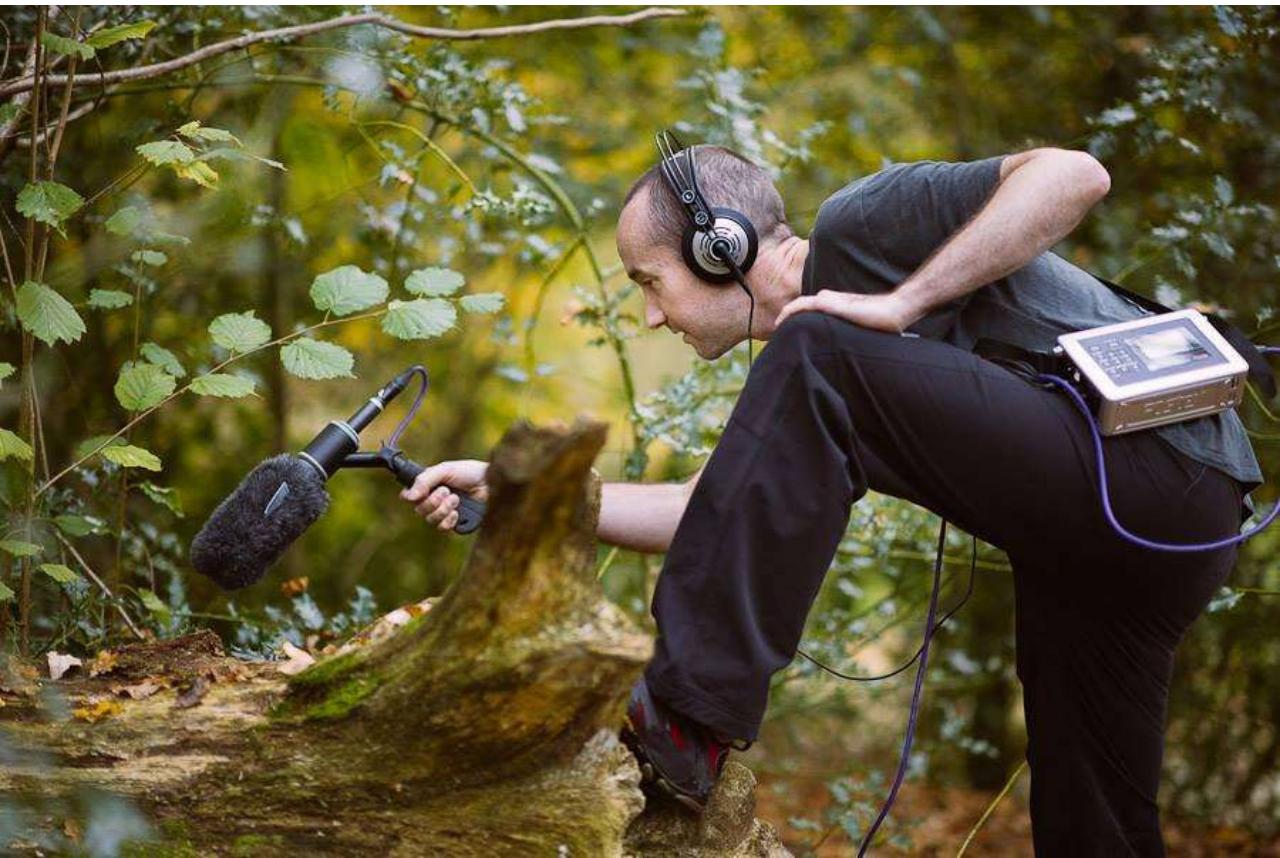


Foley Artist 的工具

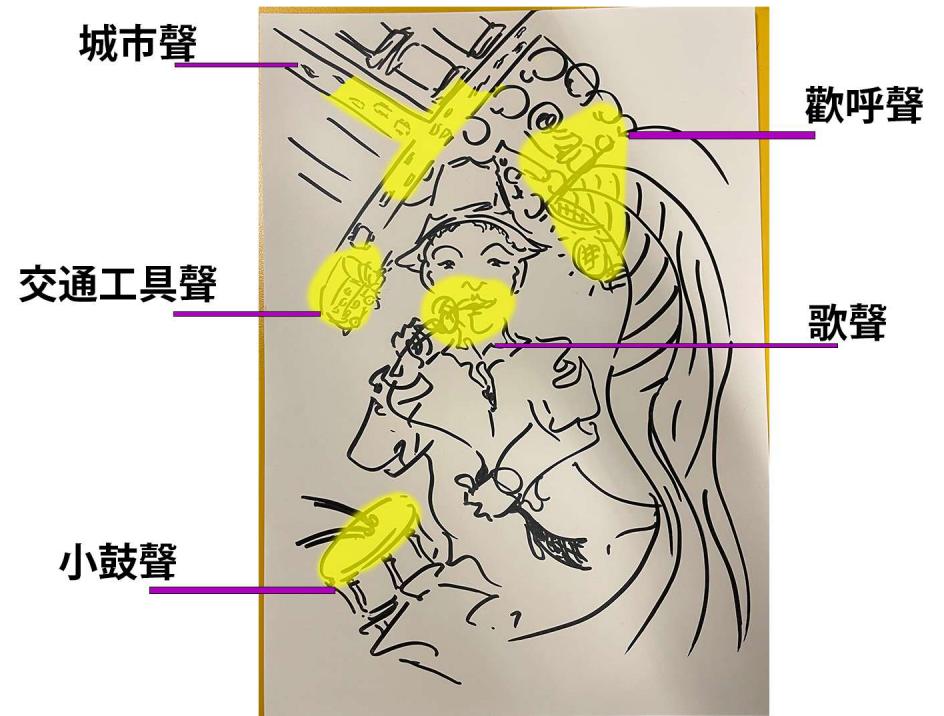
甚麼是 現場錄音 (Field Recording) ?

現場錄音 (Field Recording) 是指在戶外或室內環境中使用錄音設備進行錄音，捕捉自然或人造聲音，用於音樂、電影、電視、遊戲等製作，增加真實感和氛圍。

[https://www.youtube.com/watch?
v=G0nhh6Jkb1U](https://www.youtube.com/watch?v=G0nhh6Jkb1U)



現在為你的作品創作聲音



集齊聲音檔案後，我們可以做一些後製~

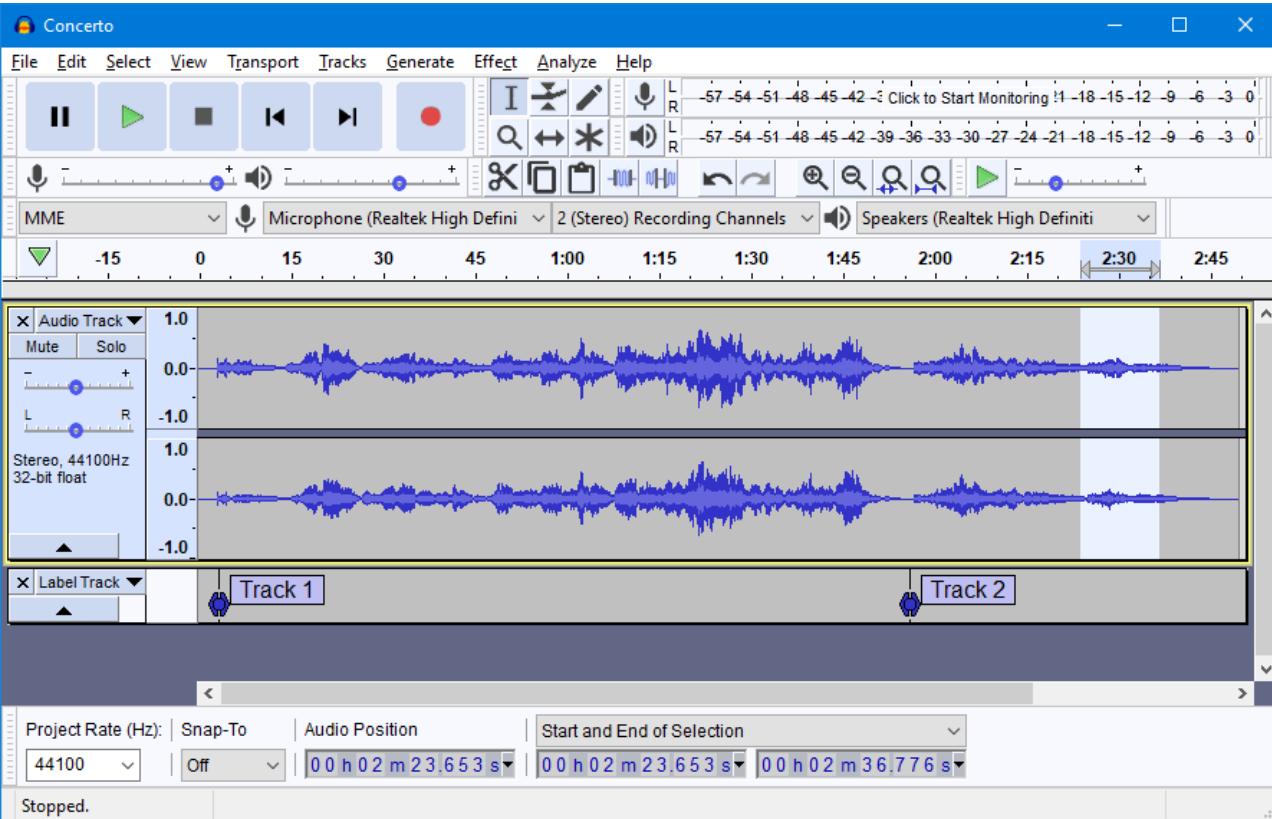
甚麼是 數位音訊工作站 (Digital Audio Workstation ,DAW) ?

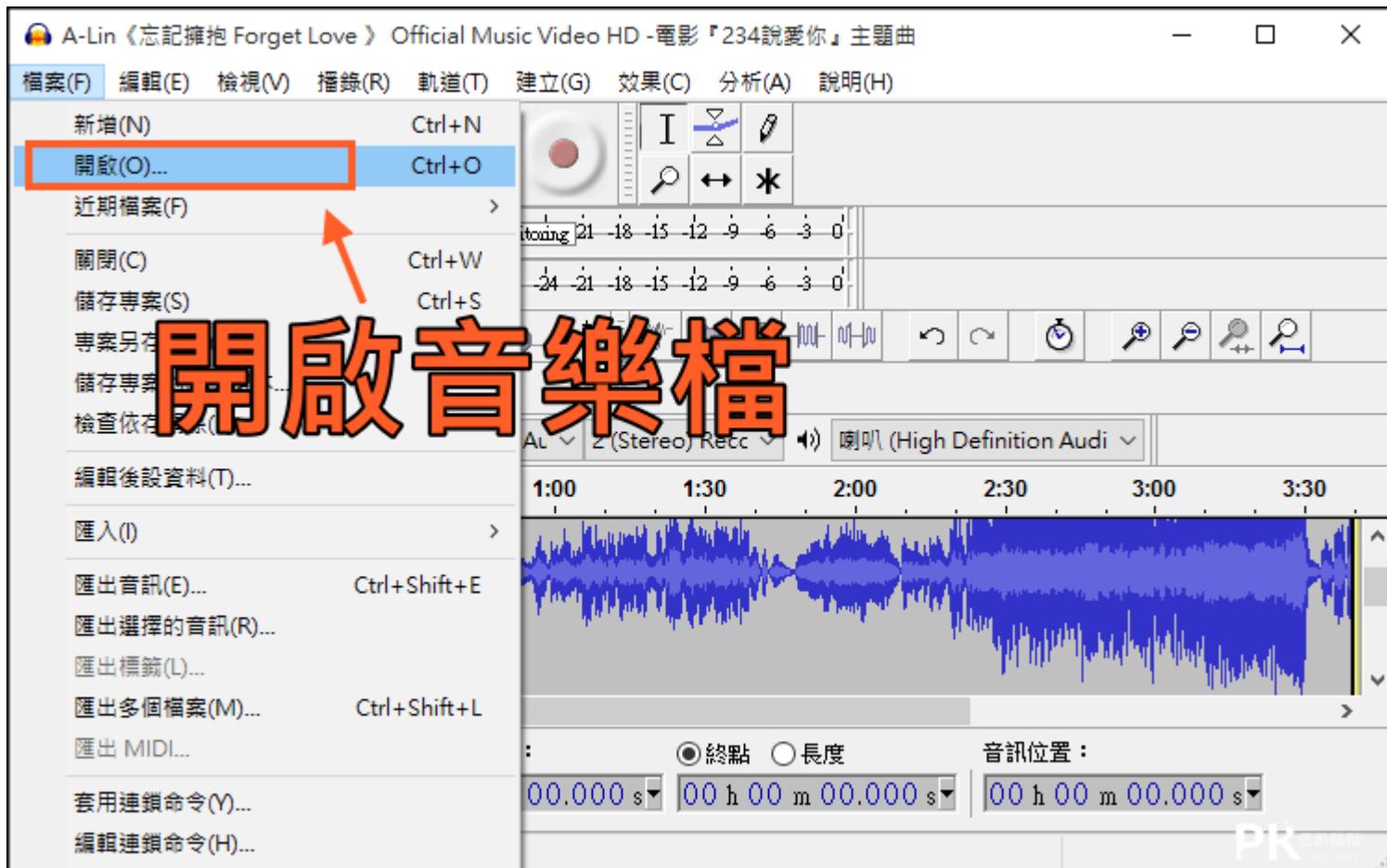
音訊工作站是一種專業數位音頻製作軟體，可用於錄音、混音、編曲、編輯及後製處理等音樂製作工作，適用於聲音藝術家、音樂製作人、錄音師、唱片製作人等。



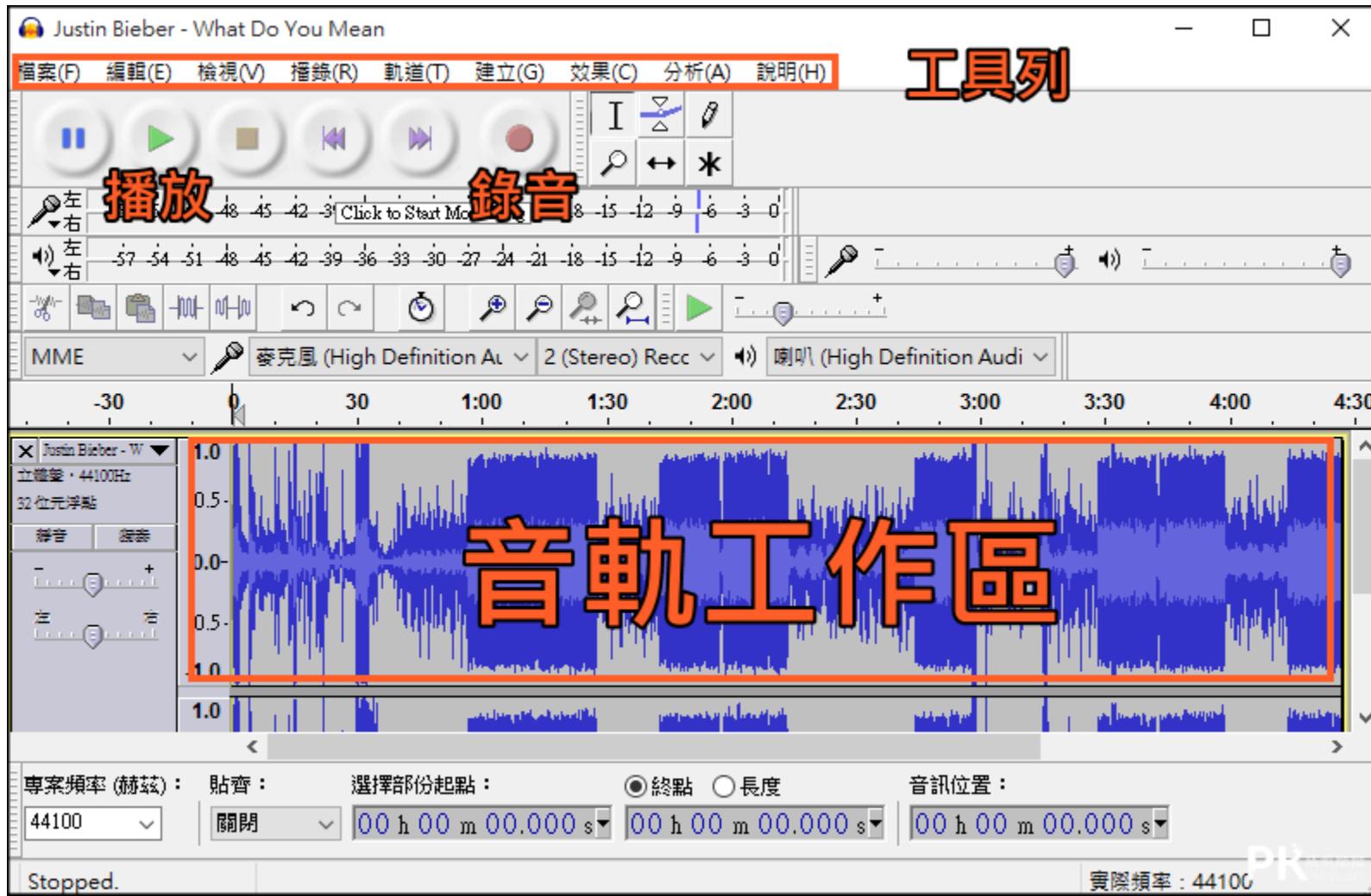
我們會使用開源軟件

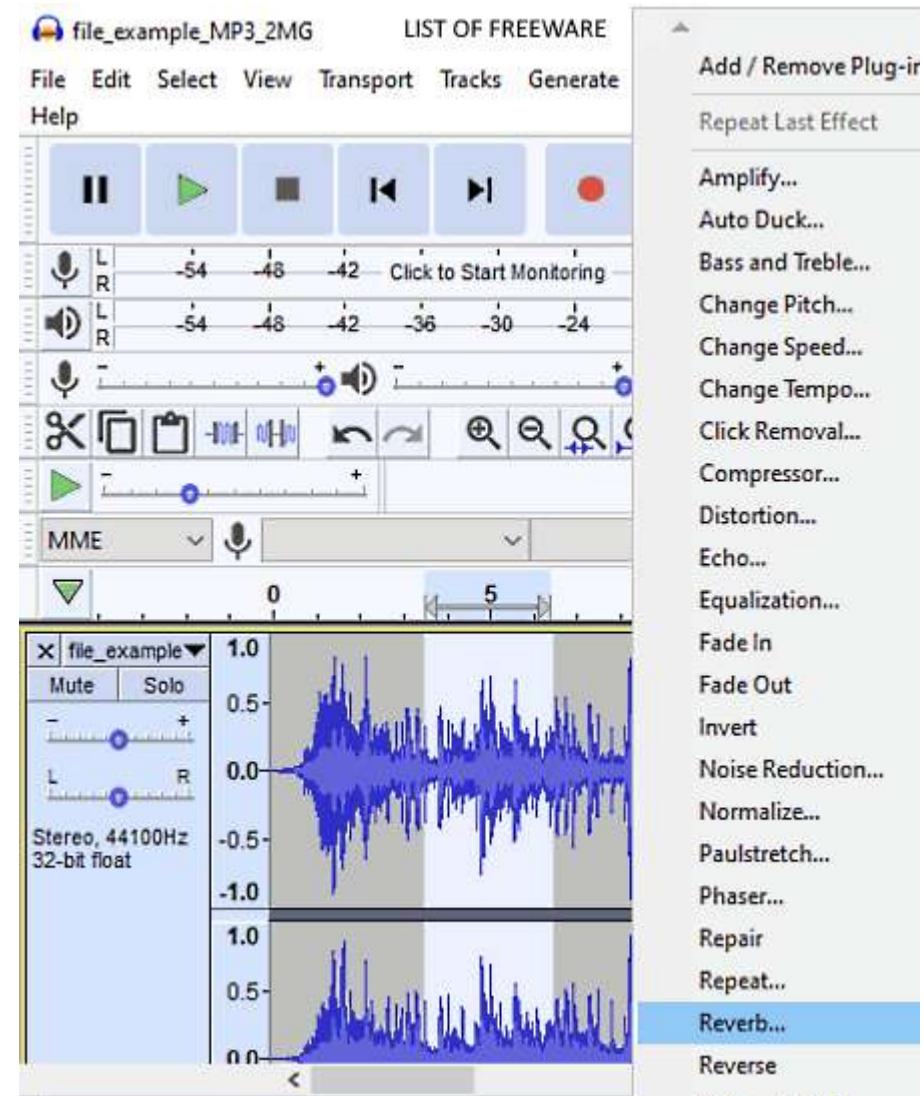
Audacity

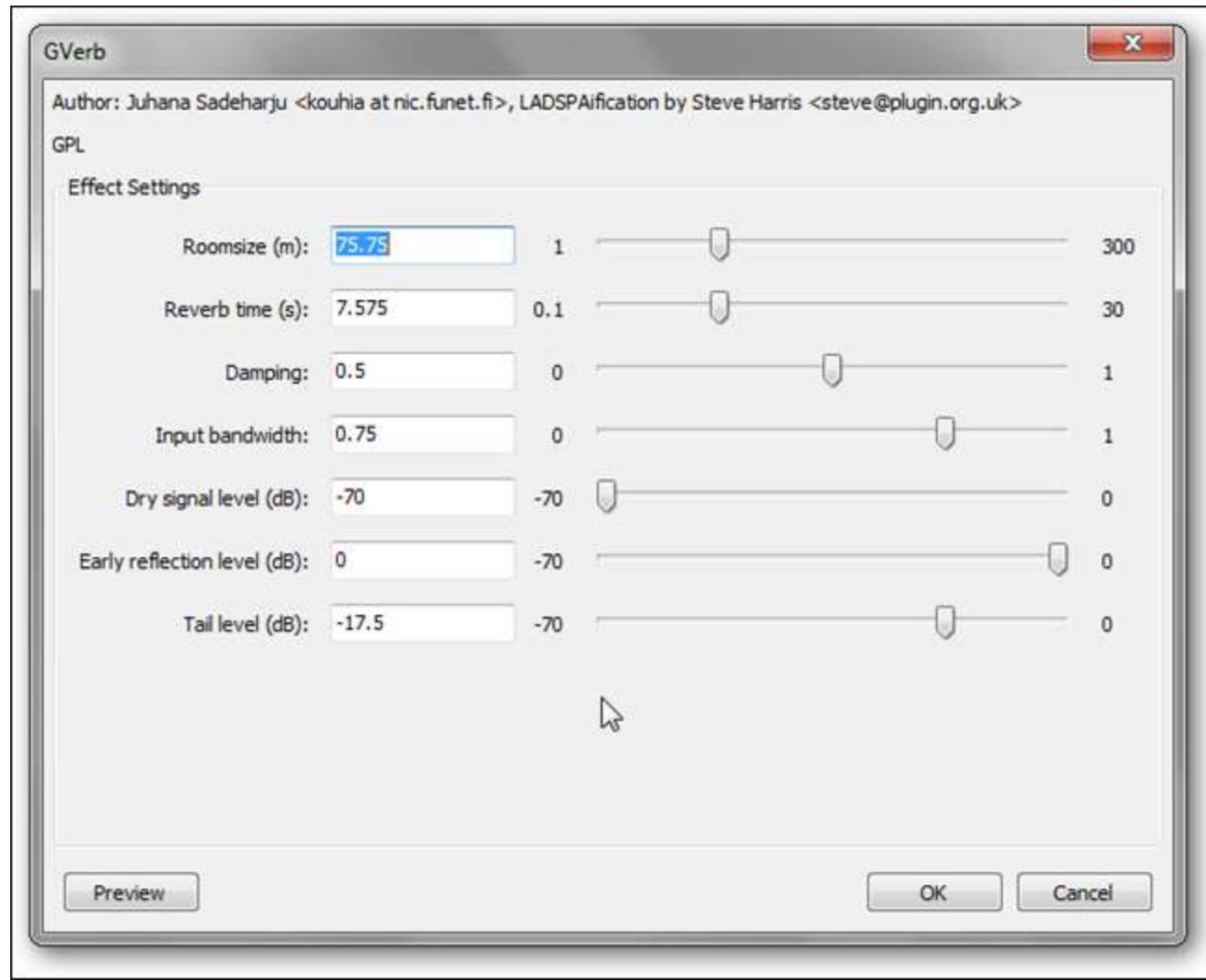


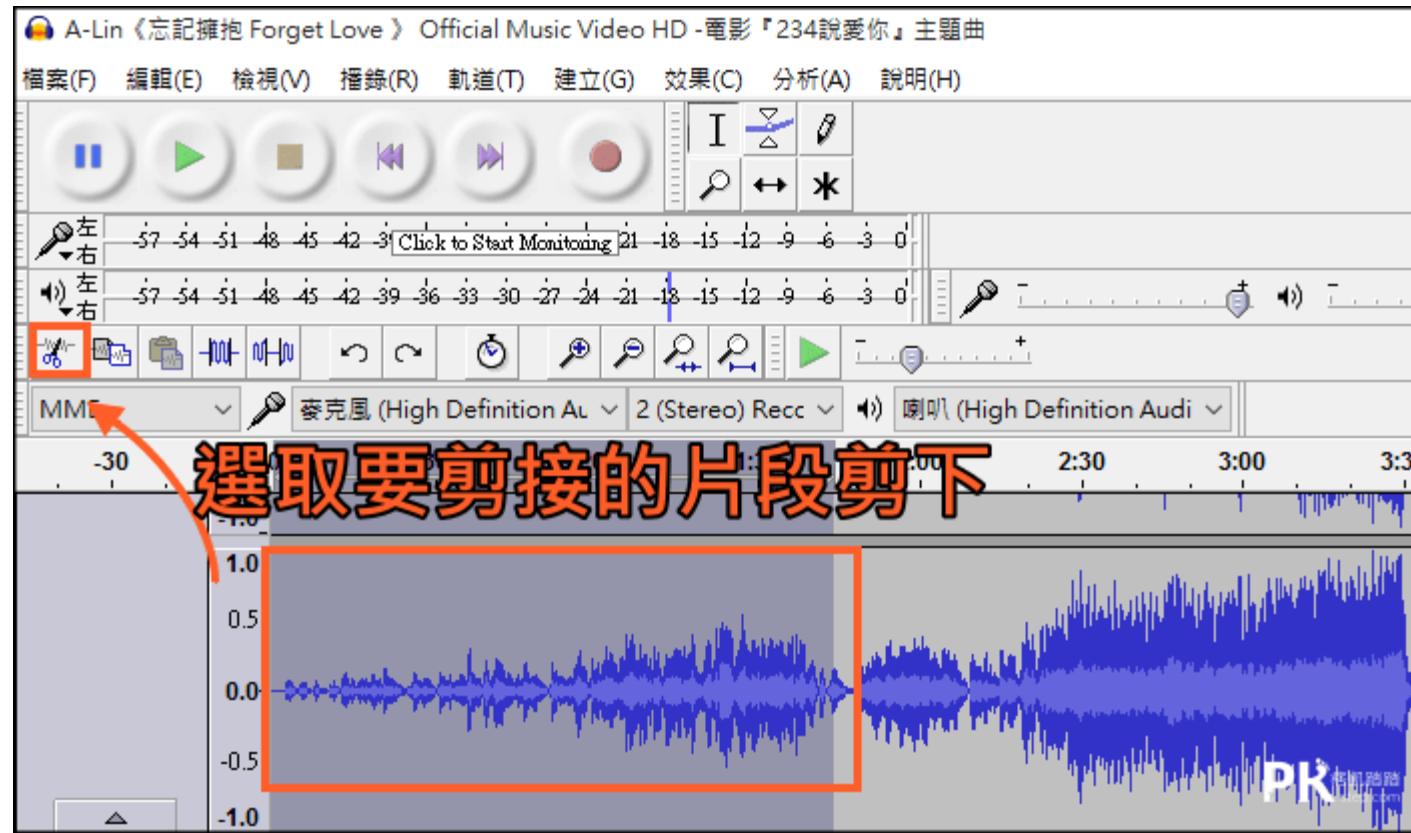


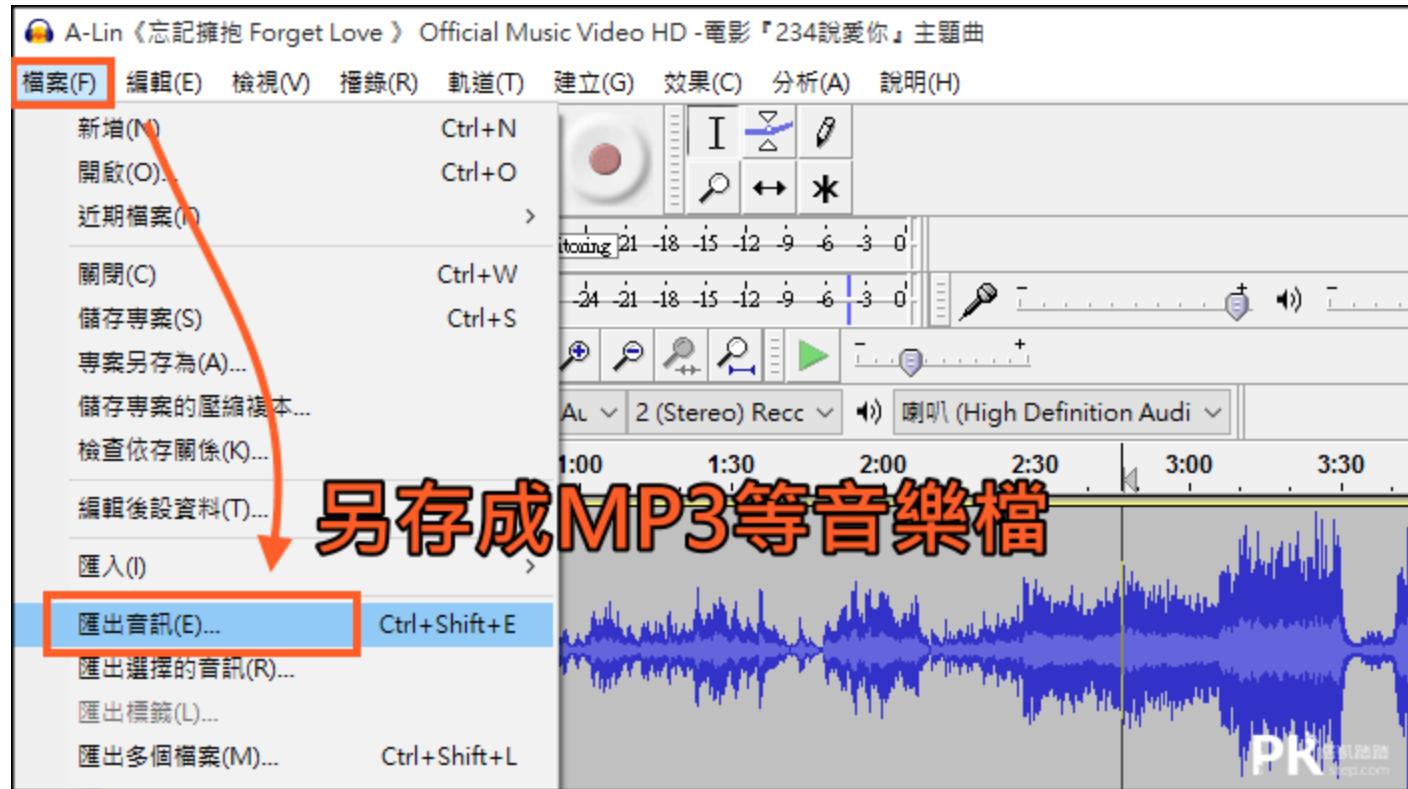
開啟音樂檔











下午部份完成

明日課堂部份

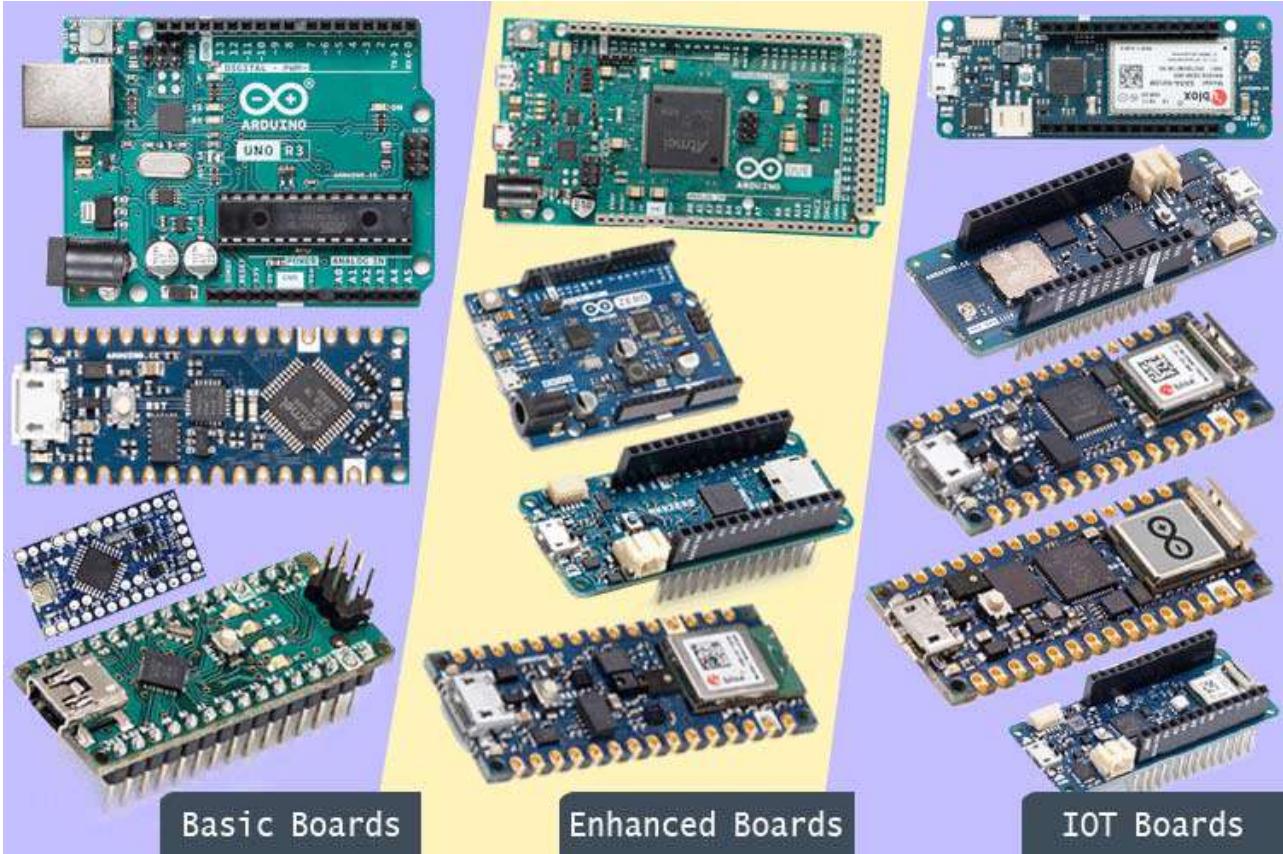
- 基礎編程、更多有關聲音藝術的例子

Q&A

附加

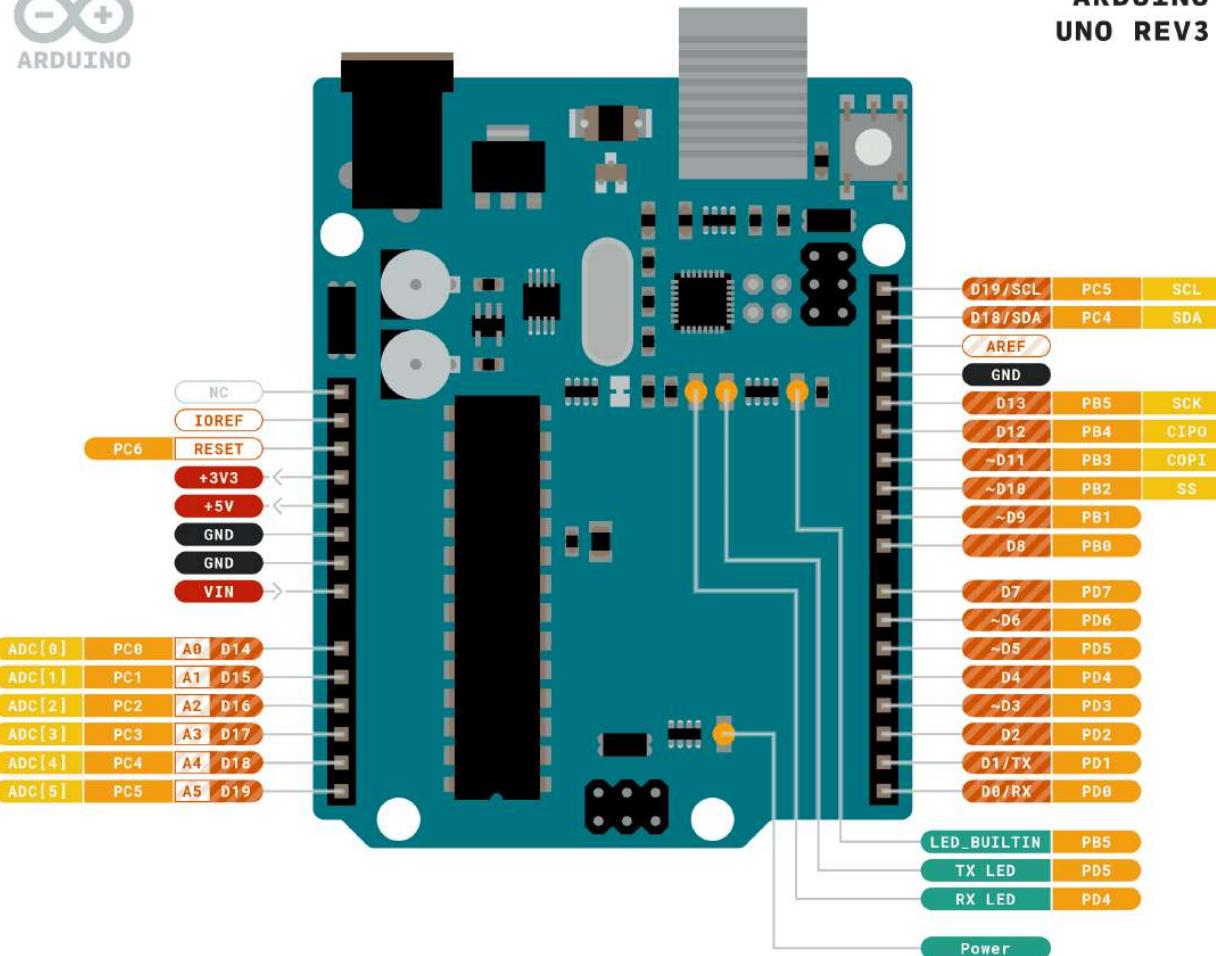
甚麼是 Arduino?

Arduino 是一個開源電子原型平台，允許用戶輕鬆創建互動式電子項目。它基於用戶友好的硬件和軟件，為創客、工程師和學生提供了一個靈活且易於使用的開發環境，促進創新和學習。源於2005年，Arduino 由一個意大利團隊開發，旨在為非專業人士提供低成本、易於使用的工具。



Basic Pinout Diagram

https://content.arduino.cc/assets/Pinout-UNOrev3_latest.pdf?_gl=1*jm7fcz*_ga*MTQ2MTMwNzk2OC4xNjgyMjc0MjYx*_ga_NEXN8H46L5*MTY4MjI3NDI2MC4xLjAuMTY4MjI3NDI2MC4wLjAuMA..



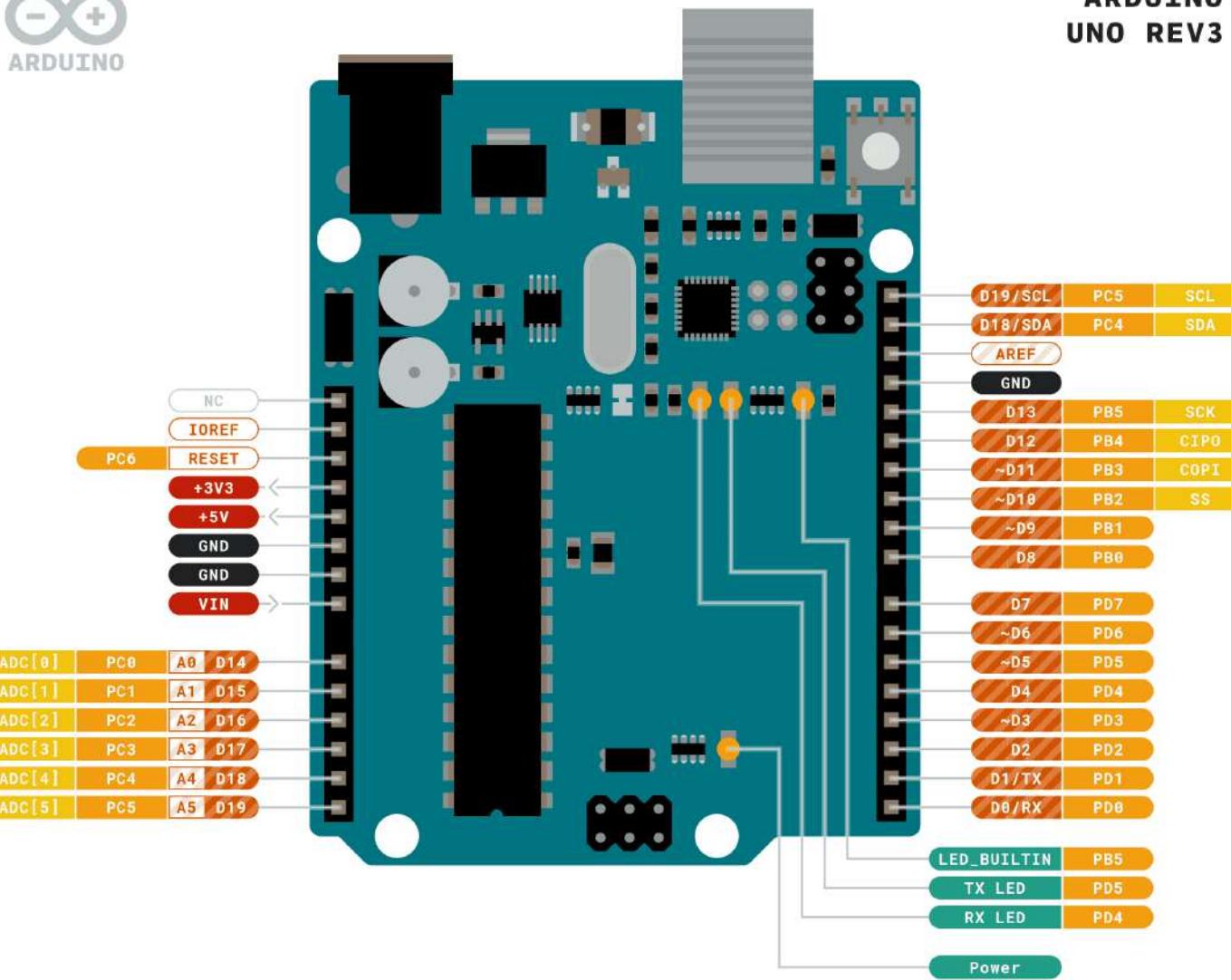
■ Ground ■ Internal Pin ■ Digital Pin ■ Microcontroller's Port
■ Power ■ SWD Pin □ Analog Pin
■ LED □ Other Pin ■ Default



This work is licensed under the Creative Commons Attribution-ShareAlike 4.0 International License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-sa/4.0/> or send a letter to Creative Commons, PO Box 1866, Mountain View, CA 94042, USA.



ARDUINO UNO REV3



■ Ground ■ Internal Pin ■ Digital Pin ■ Microcontroller's Port
■ Power ■ SWD Pin ■ Analog Pin
■ LED ■ Other Pin ■ Default



This work is licensed under the Creative Commons
Attribution-ShareAlike 4.0 International License. To view
a copy of this license, visit <http://creativecommons.org/licenses/by-sa/4.0/> or send a letter to Creative
Commons, PO Box 1866, Mountain View, CA 94034, USA

安裝 Arduino IDE 1.8.19

link to download Arduino IDE 1.8

<https://www.arduino.cc/en/software#future-version-of-the-arduino-ide>



Arduino IDE 1.8.19

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. This software can be used with any Arduino board.

Refer to the [Getting Started](#) page for Installation instructions.

SOURCE CODE

Active development of the Arduino software is [hosted by GitHub](#). See the instructions for [building the code](#). Latest release source code archives are available [here](#). The archives are PGP-signed so they can be verified using [this](#) gpg key.

DOWNLOAD OPTIONS

Windows Win 7 and newer
Windows ZIP file
Windows app Win 8.1 or 10 [Get](#)

Linux 32 bits
Linux 64 bits
Linux ARM 32 bits
Linux ARM 64 bits

Mac OS X 10.10 or newer

[Release Notes](#)
[Checksums \(sha512\)](#)

File Edit Sketch Tools Help



sketch_sep06a



```
void setup() {
    // put your setup code here, to run once:

}

void loop() {
    // put your main code here, to run repeatedly:

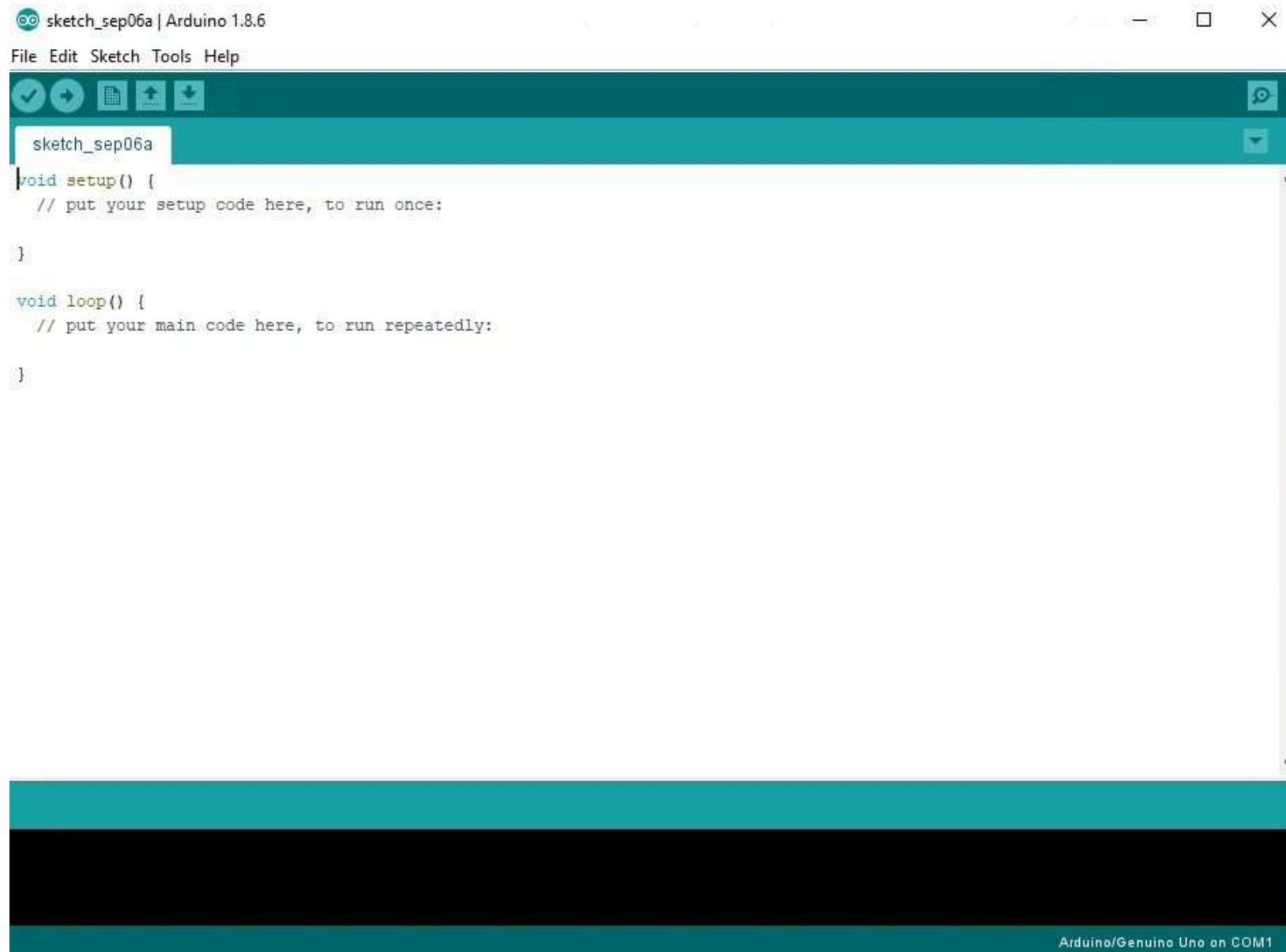
}
```

Arduino IDE

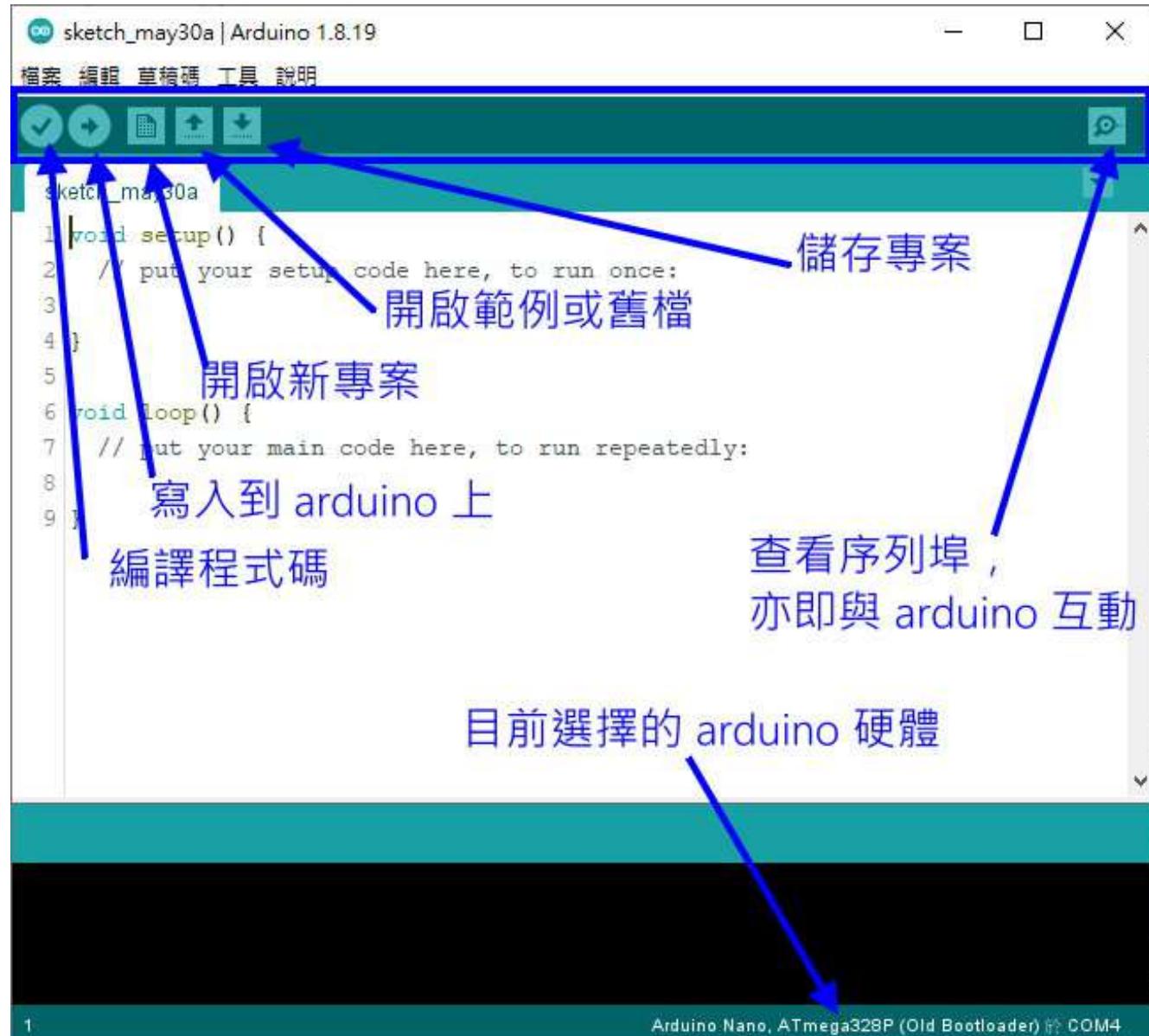
Arduino IDE 1.8.19 是一個的開發環境，提供代碼編輯器、實時錯誤檢查。



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。



檔案 編輯 草稿碼 工具 說明



自動格式化 Ctrl+T

封存草稿碼

修正編碼並重新載入

管理程式庫...

序列埠監控視窗 Ctrl+Shift+M

序列繪圖家 Ctrl+Shift+L

WiFi101 / WiFiNINA Firmware Updater

開發板: "Arduino Nano"

處理器: "ATmega328P (Old Bootloader)"

序列埠: "COM4"

取得開發板資訊

燒錄器: "BusPirate as ISP"

燒錄Bootloader



開發板管理員...

Arduino Yún

Arduino Uno

Arduino Duemilanove or Diecimila

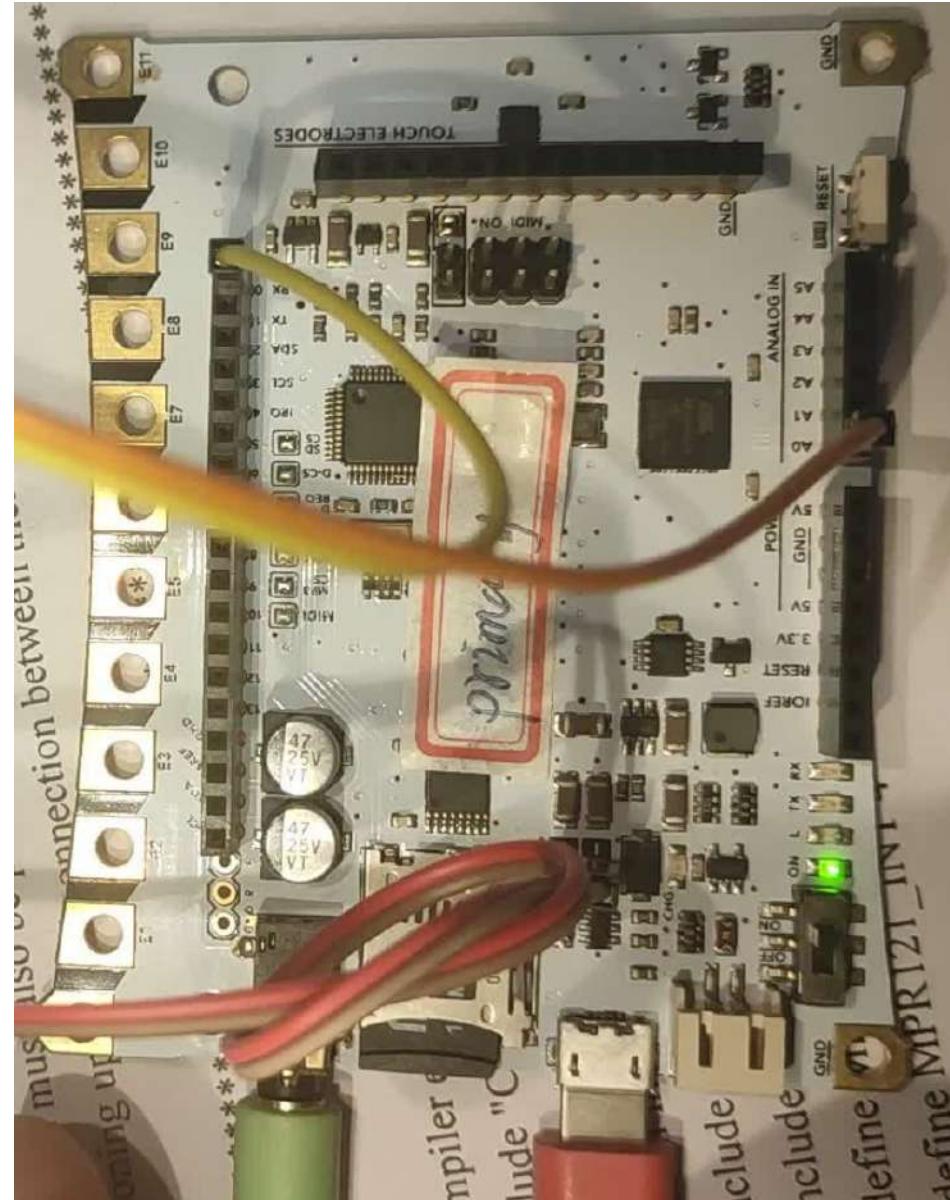
● Arduino Nano

Arduino Mega or Mega 2560

Arduino Mega ADK

Arduino Leonardo

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

檔案 編輯 草稿碼 工具 說明



sketch_may30a

```
void setup ()  
  // put your  
}  
  
void loop () {  
  // put your  
}
```

自動格式化

Ctrl+T

封存草稿碼

修正編碼並重新載入

管理程式庫...

Ctrl+Shift+I

序列埠監控視窗

Ctrl+Shift+M

序列埠國家

Ctrl+Shift+L

WiFi101 / WiFiNINA Firmware Updater

開發板: "Arduino Nano"

處理器: "ATmega328P (Old Bootloader)"

序列埠: "COM4"

取得開發板資訊

燒錄器: "BusPirate as ISP"

燒錄Bootloader

序列埠

COM1

COM2

COM3

COM4

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。

The screenshot shows the Arduino IDE interface with the title bar "sketch_may30a | Arduino 1.8.19". The menu bar includes "檔案" (File), "編輯" (Edit), "草稿碼" (Sketch), "工具" (Tools), and "說明" (Help). Below the menu is a toolbar with icons for saving, opening, and running code. The main window displays the code editor with the sketch name "sketch_may30a". The code is organized into two main sections:

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

The first section, "setup()", is labeled "初始化 arduino 硬體的程式碼" (Initialize Arduino hardware code). The second section, "loop()", is labeled "實際運作的程式碼" (Actual runtime code). Both sections are highlighted with blue boxes.

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);
}
```

先按這裡做編譯

這裡會顯示編譯的結果！
若有錯誤，就得要修訂才行

```
22 https://www.arduino.cc/en/Tutorial/BuiltinExamples/Blink
23 */
24
25 // the setup function runs once when you press reset or power the board
26 void setup() {
27     // initialize digital pin LED_BUILTIN as an output.
28     pinMode(LED_BUILTIN, OUTPUT);
29 }
30
31 // the loop function runs over and over again forever
32 void loop() {
33     digitalWrite(LED_BUILTIN, HIGH);      // turn the LED on (HIGH is the voltage level
34     delay(1000);                      // wait for a second
35     digitalWrite(LED_BUILTIN, LOW);     // turn the LED off by making the voltage LOW
36     delay(1000);                      // wait for a second
37 }
```

編譯完畢。

草稿碼使用了 924 bytes (3%) 的程式儲存空間。上限為 30720 bytes。
全域變數使用了 9 bytes (0%) 的動態記憶體，剩餘 2039 bytes 給區域變數。上限為 2048 bytes。

Arduino Nano, ATmega328P (Old Bootloader) 於 COM4

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。

先按下上傳按鈕之後

會顯示上傳的結果在這裡！

有問題時，參考這裡的方案

```
22 https://www.arduino.cc/en/Tutorial/BuiltinExamples/Blink
23 */
24
25 // the setup function runs once when you press reset or power the board
26 void setup() {
27     // initialize digital pin LED_BUILTIN as an output.
28     pinMode(LED_BUILTIN, OUTPUT);
29 }
30
31 // the loop function runs over and over again forever
32 void loop() {
33     digitalWrite(LED_BUILTIN, HIGH);      // turn the LED on (HIGH is the voltage level
34     delay(1000);                      // wait for a second
35     digitalWrite(LED_BUILTIN, LOW);     // turn the LED off by making the voltage LOW
36     delay(1000);                      // wait for a second
37 }
```

上傳完畢。

```
avrduke: 924 bytes of flash verified

avrduke done. Thank you.
```

1 Arduino Nano, ATmega328P (Old Bootloader) 於 COM4

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

- WELCOME TO -

賽馬會科藝共融計劃

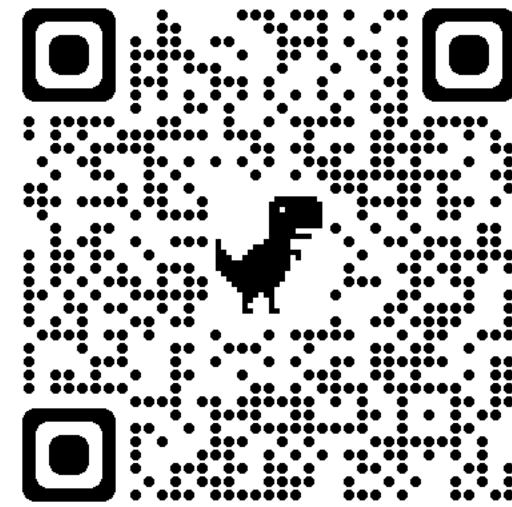
JC-Project-IDEA

Day02

上午



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



在使用 開發板 做到影片中的效果前

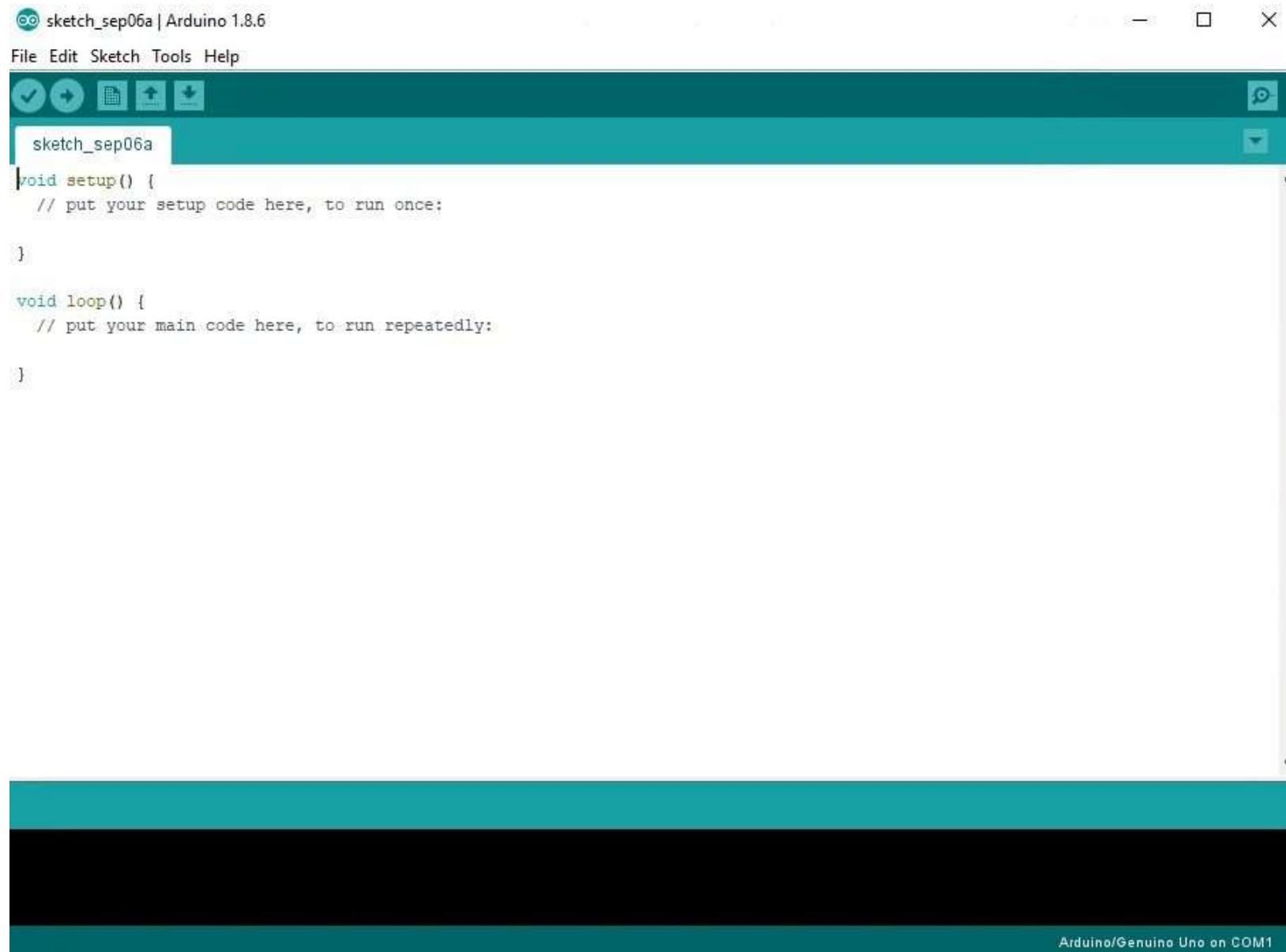
要做的準備工作

Arduino IDE

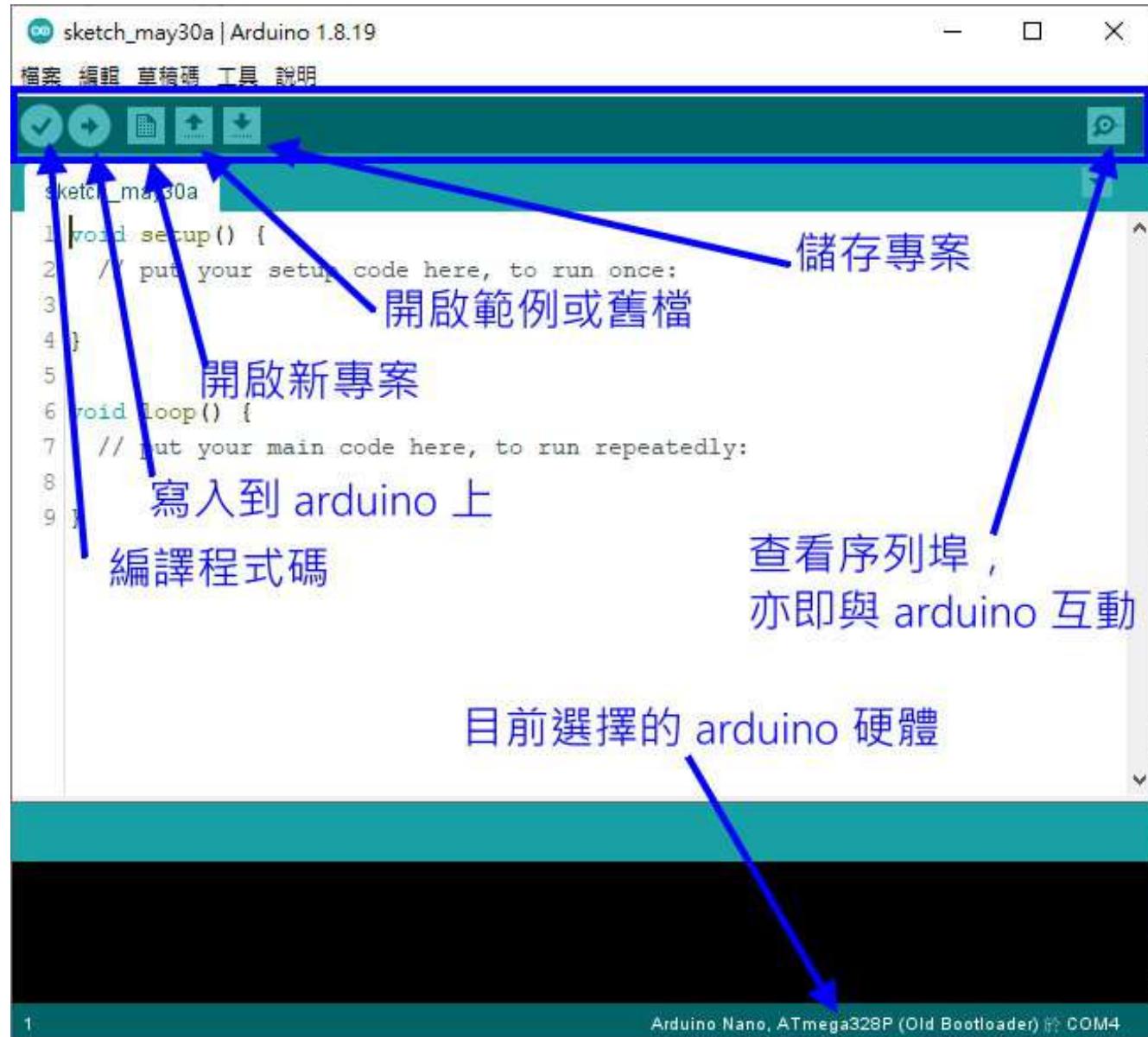
Arduino IDE 1.8.19 是一個的開發環境，提供代碼編輯器、實時錯誤檢查。



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。



檔案 編輯 草稿碼 工具 說明



自動格式化 Ctrl+T

封存草稿碼

修正編碼並重新載入

管理程式庫...

序列埠監控視窗

序列繪圖家

WiFi101 / WiFiNINA Firmware Updater

開發板: "Arduino Nano"

處理器: "ATmega328P (Old Bootloader)"

序列埠: "COM4"

取得開發板資訊

燒錄器: "BusPirate as ISP"

燒錄Bootloader



開發板管理員...

Arduino Yún

Arduino Uno

Arduino Duemilanove or Diecimila

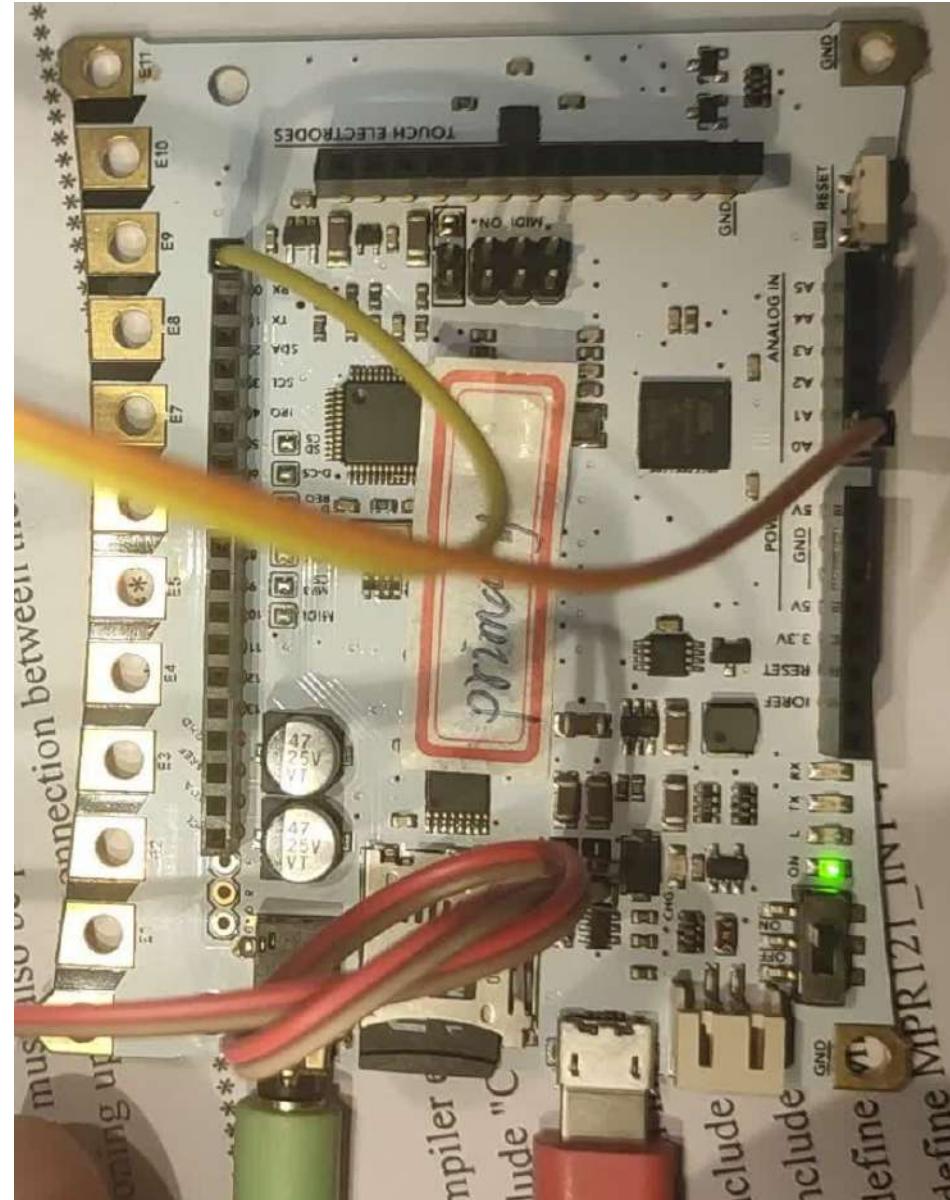
● Arduino Nano

Arduino Mega or Mega 2560

Arduino Mega ADK

Arduino Leonardo

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

檔案 編輯 草稿碼 工具 說明



sketch_may30a

```
void setup ()  
  // put your  
}  
  
void loop () {  
  // put your  
}
```

自動格式化

Ctrl+T

封存草稿碼

修正編碼並重新載入

管理程式庫...

Ctrl+Shift+I

序列埠監控視窗

Ctrl+Shift+M

序列埠國家

Ctrl+Shift+L

WiFi101 / WiFiNINA Firmware Updater

開發板: "Arduino Nano"

處理器: "ATmega328P (Old Bootloader)"

序列埠: "COM4"

取得開發板資訊

燒錄器: "BusPirate as ISP"

燒錄Bootloader

序列埠

COM1

COM2

COM3

COM4

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 | Arduino 1.8.19
檔案 編輯 草稿碼 工具 說明
sketch_may30a
1 void setup() {
2 // put your setup code here, to run once:
3 }
4 } 初始化 arduino 硬體的程式碼
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) ▾ 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);
}
```

先按這裡做編譯

這裡會顯示編譯的結果！
若有錯誤，就得要修訂才行

```
22 https://www.arduino.cc/en/Tutorial/BuiltinExamples/Blink
23 */
24
25 // the setup function runs once when you press reset or power the board
26 void setup() {
27     // initialize digital pin LED_BUILTIN as an output.
28     pinMode(LED_BUILTIN, OUTPUT);
29 }
30
31 // the loop function runs over and over again forever
32 void loop() {
33     digitalWrite(LED_BUILTIN, HIGH);      // turn the LED on (HIGH is the voltage level
34     delay(1000);                      // wait for a second
35     digitalWrite(LED_BUILTIN, LOW);       // turn the LED off by making the voltage LOW
36     delay(1000);                      // wait for a second
37 }
```

編譯完畢。

草稿碼使用了 924 bytes (3%) 的程式儲存空間。上限為 30720 bytes。
全域變數使用了 9 bytes (0%) 的動態記憶體，剩餘 2039 bytes 給區域變數。上限為 2048 bytes。

Arduino Nano, ATmega328P (Old Bootloader) 於 COM4

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。

先按下上傳按鈕之後

會顯示上傳的結果在這裡！

有問題時，參考這裡的方案

```
22 https://www.arduino.cc/en/Tutorial/BuiltinExamples/Blink
23 */
24
25 // the setup function runs once when you press reset or power the board
26 void setup() {
27     // initialize digital pin LED_BUILTIN as an output.
28     pinMode(LED_BUILTIN, OUTPUT);
29 }
30
31 // the loop function runs over and over again forever
32 void loop() {
33     digitalWrite(LED_BUILTIN, HIGH);      // turn the LED on (HIGH is the voltage level
34     delay(1000);                      // wait for a second
35     digitalWrite(LED_BUILTIN, LOW);     // turn the LED off by making the voltage LOW
36     delay(1000);                      // wait for a second
37 }
```

上傳完畢。

```
avrduke: 924 bytes of flash verified

avrduke done. Thank you.
```

1 Arduino Nano, ATmega328P (Old Bootloader) 於 COM4

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()`。

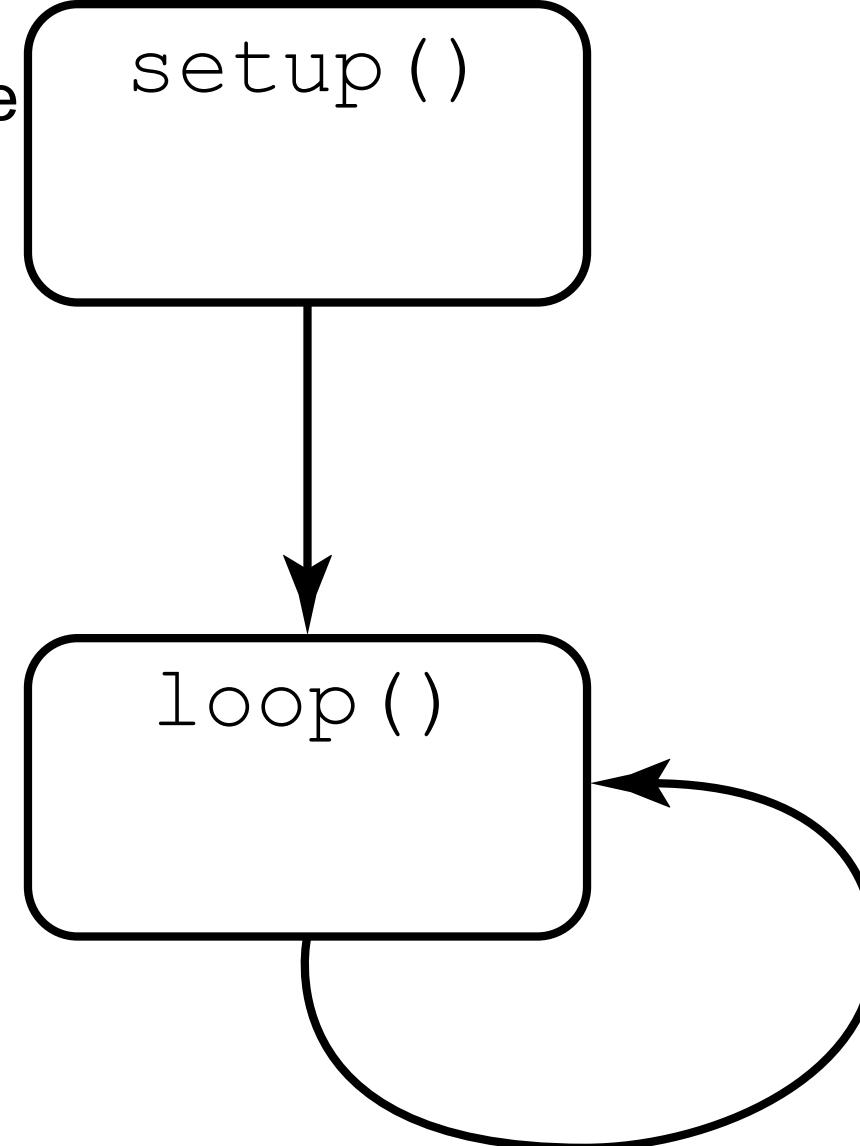
`void loop()` 是 Arduino 程式的另一個函式，用於執行主要的程式邏輯，例如讀取腳位數值、判斷條件、控制輸出等。`loop()` **會一直執行**，直到板子關機或重置。

Execution starts here
Execute once.

setup ()

loop ()

Execute repeatedly.

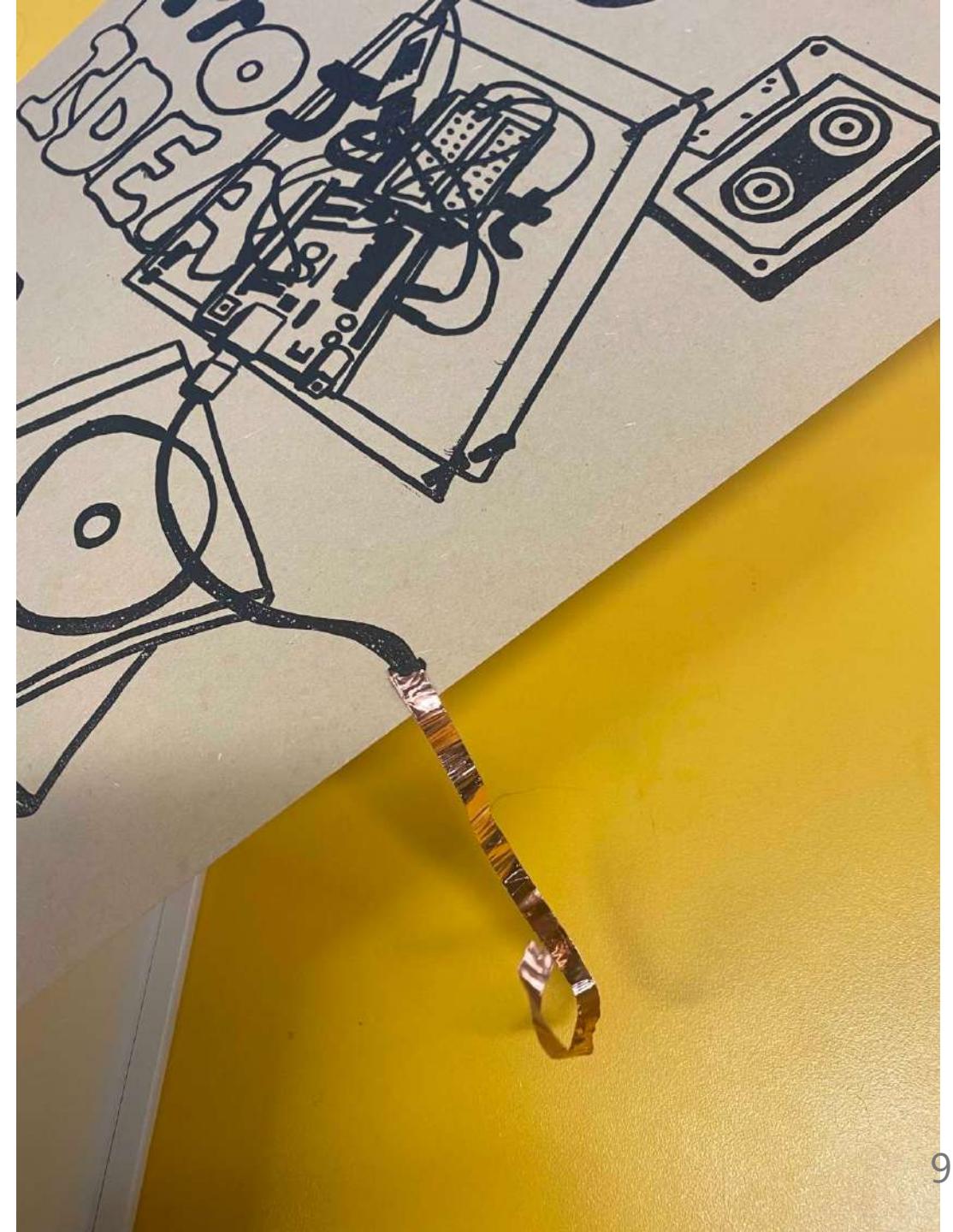


把聲音檔案準備好，下一節將會組裝硬件。

硬件組裝

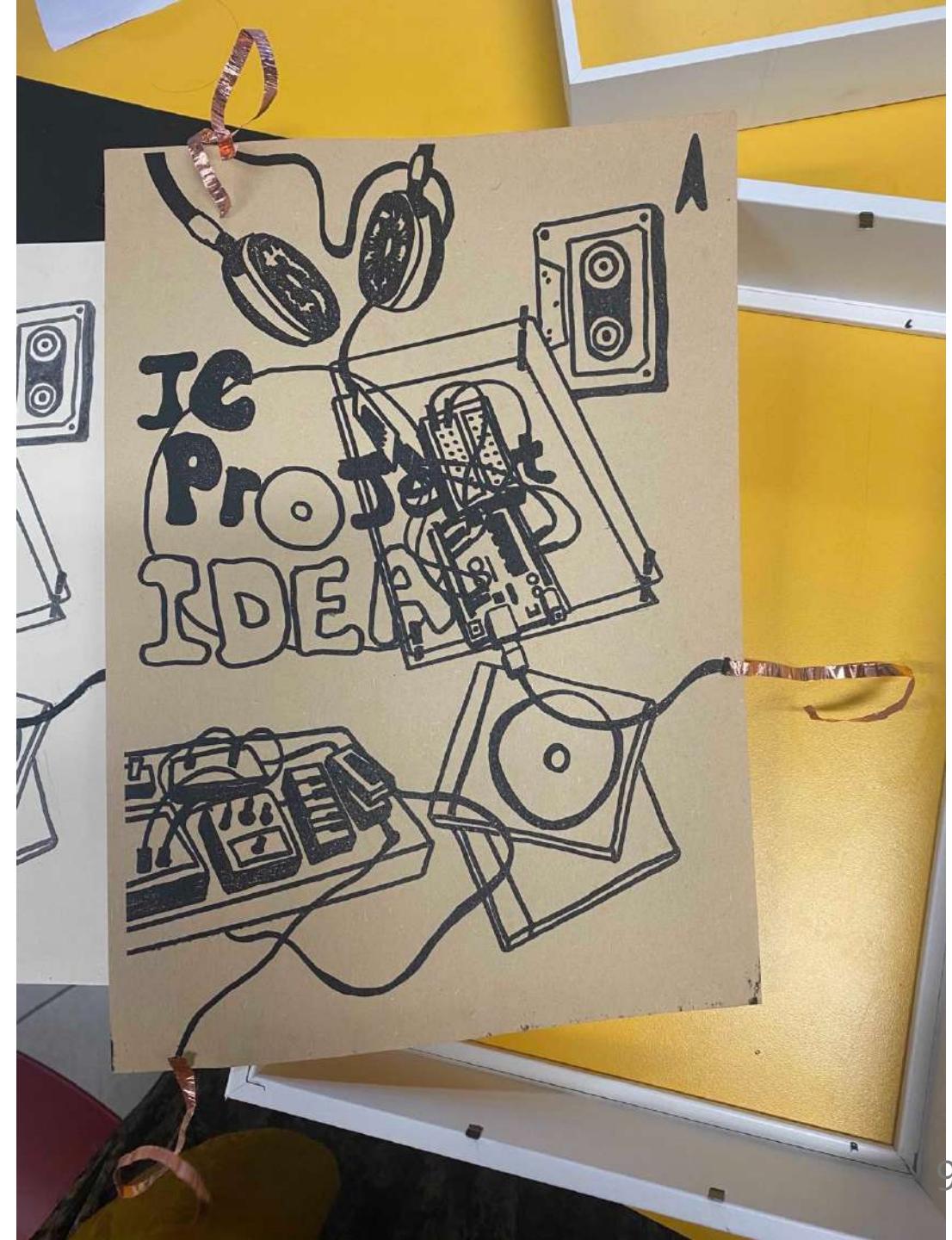
第一步

先把導電銅箔膠帶接到使用導電油墨
畫好的電路



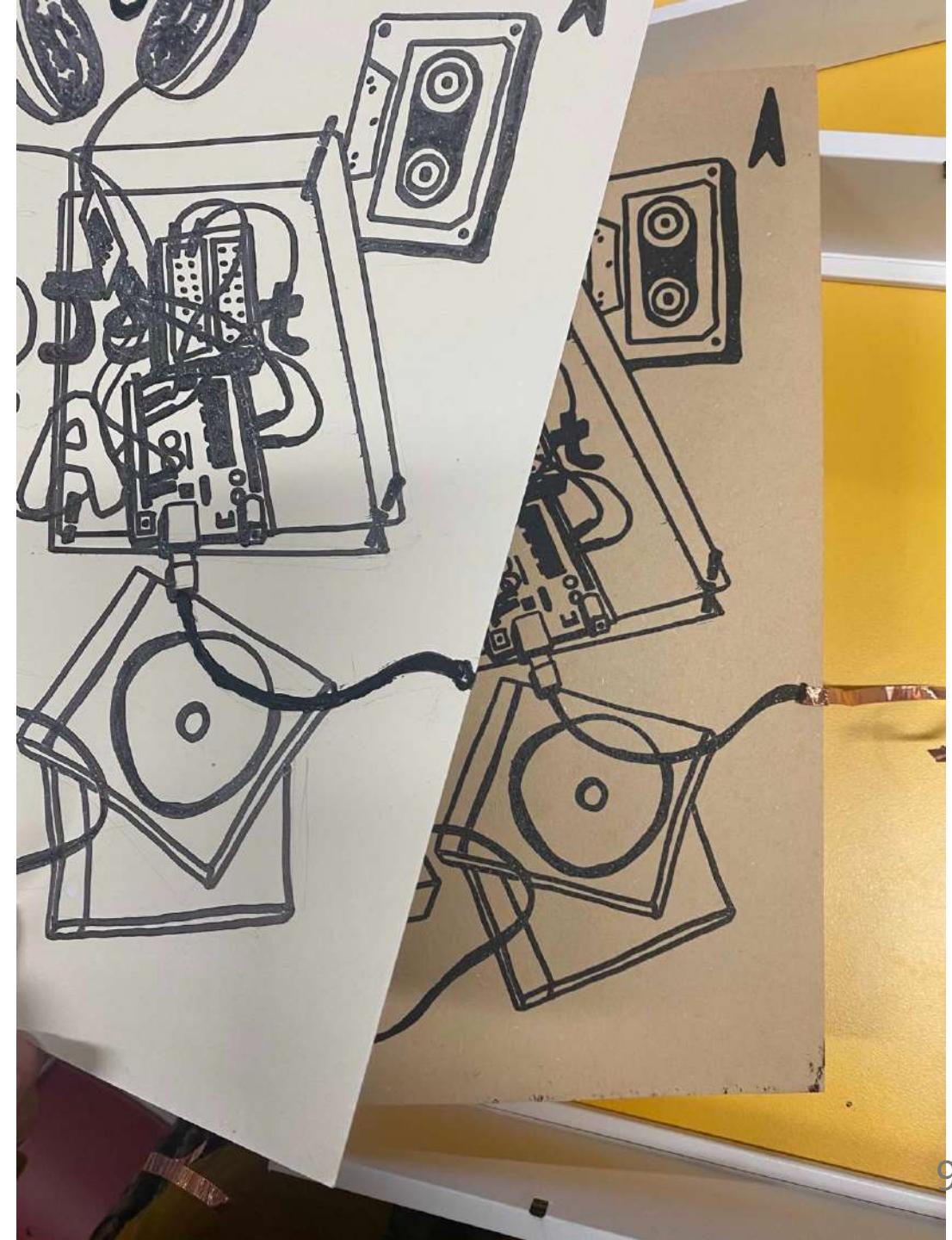
第二步

保留導電銅箔膠帶有足夠的長度



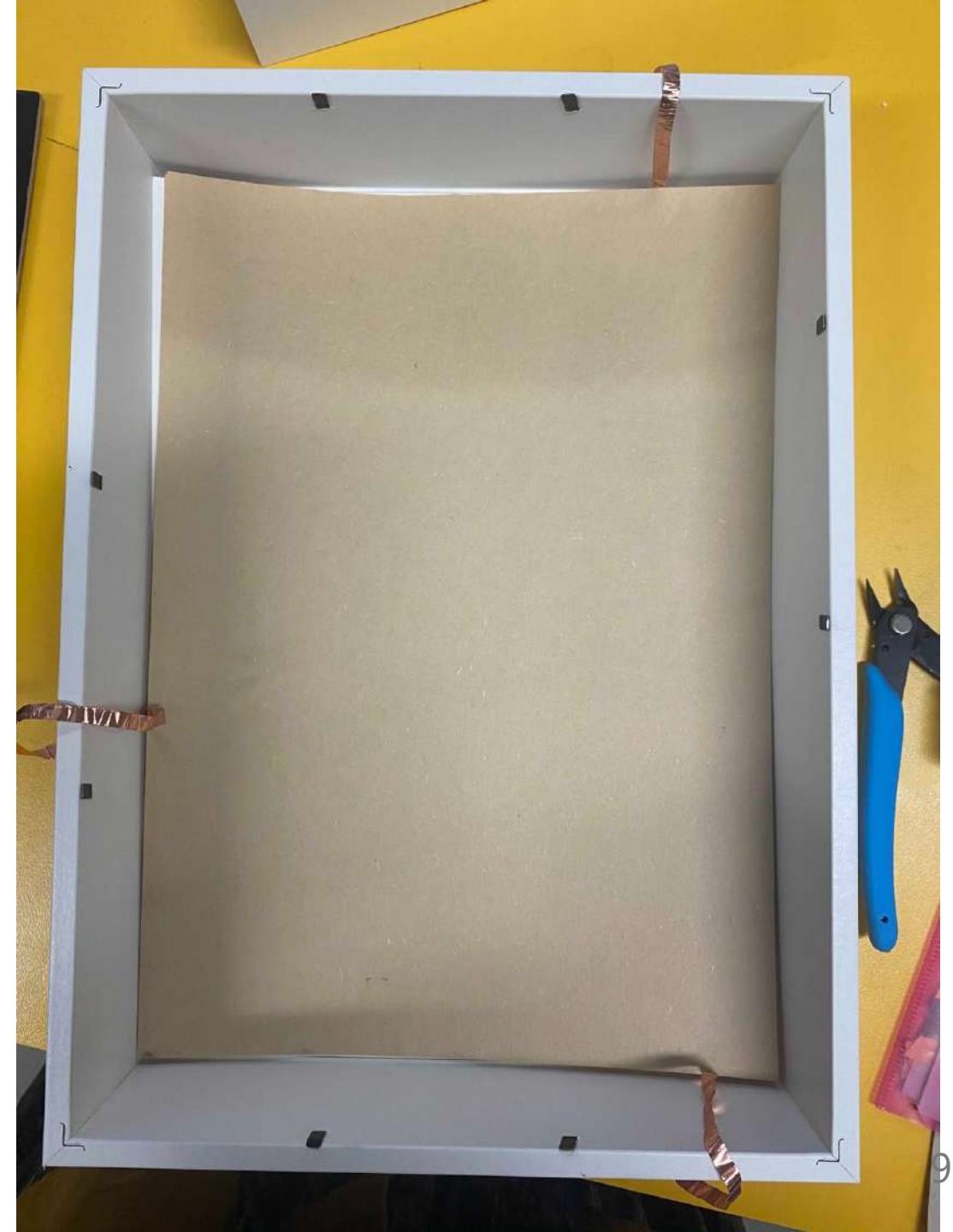
第三步

把畫好的畫對準放在電路圖的上方



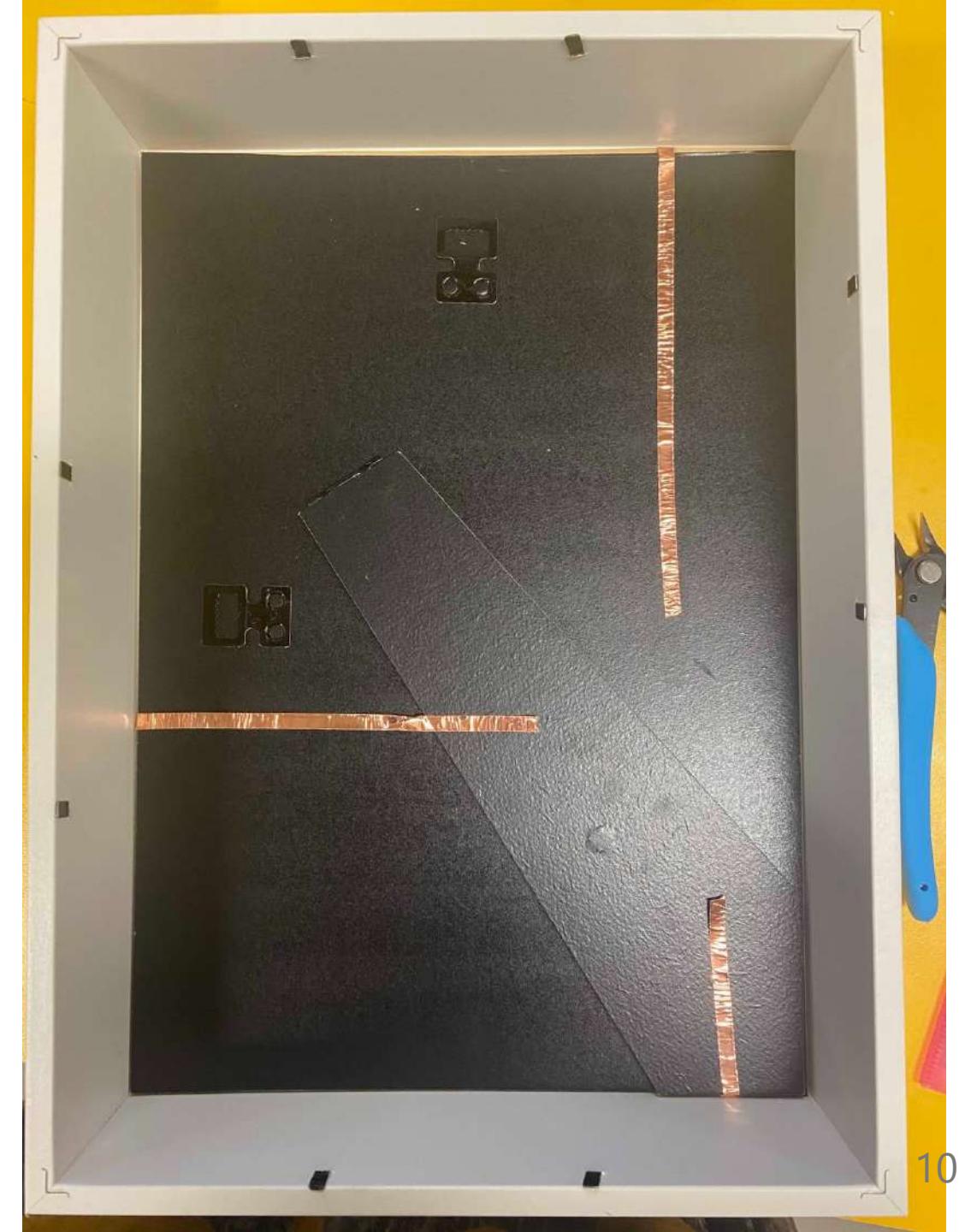
第四步

反轉畫框，把畫及電路圖放在畫框內



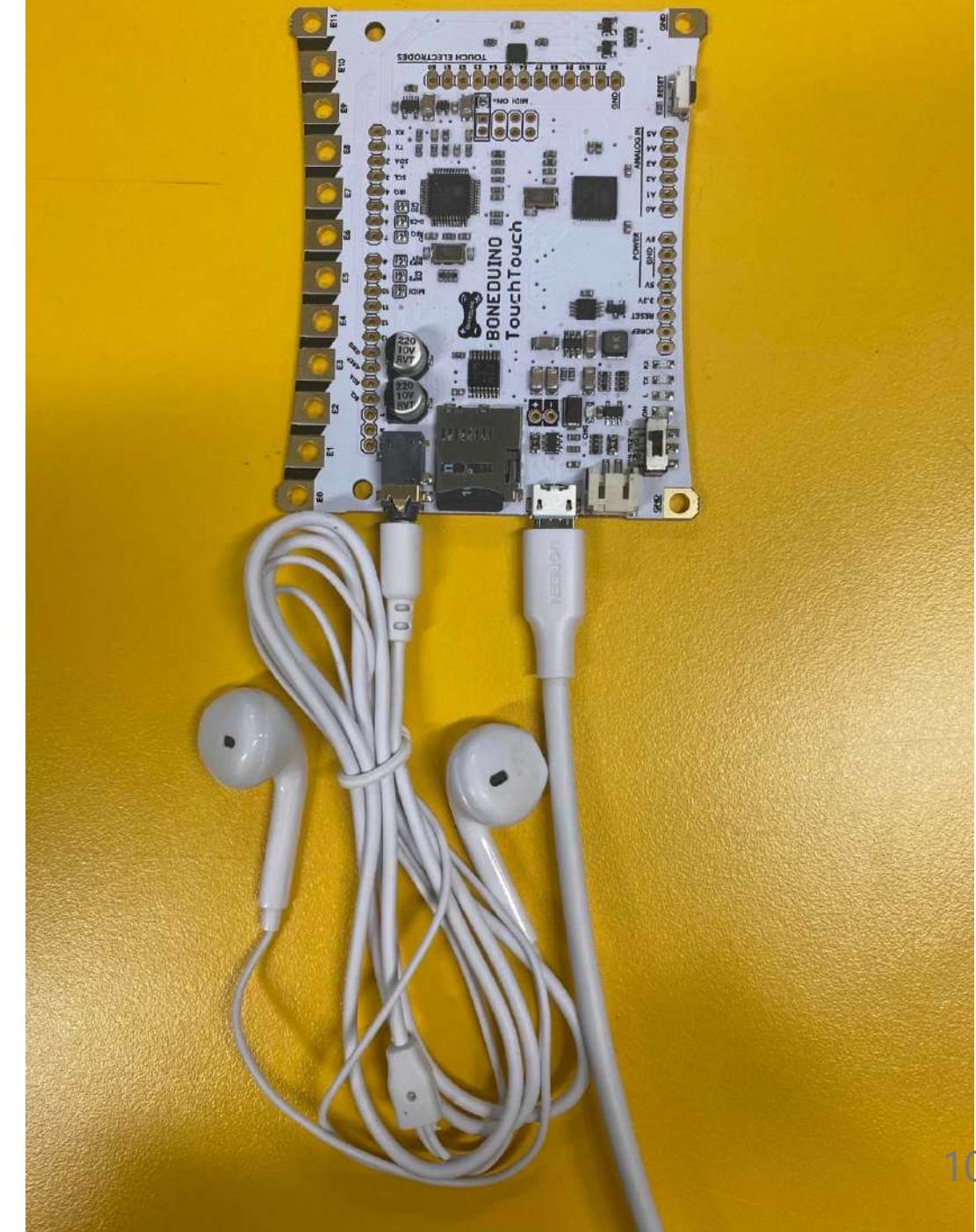
第五步

放入畫框背板，把導電銅箔膠帶保留
在背板用作連接開發板



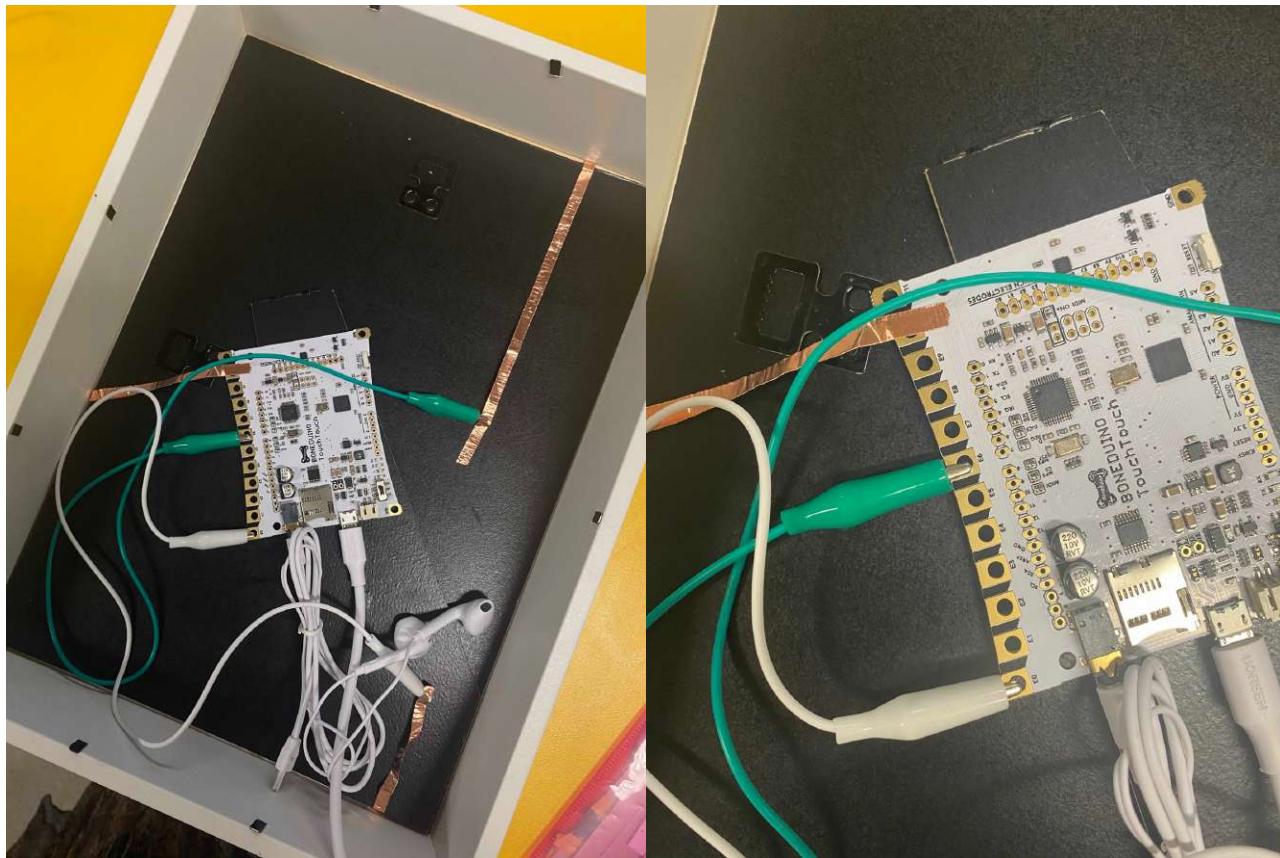
第六步

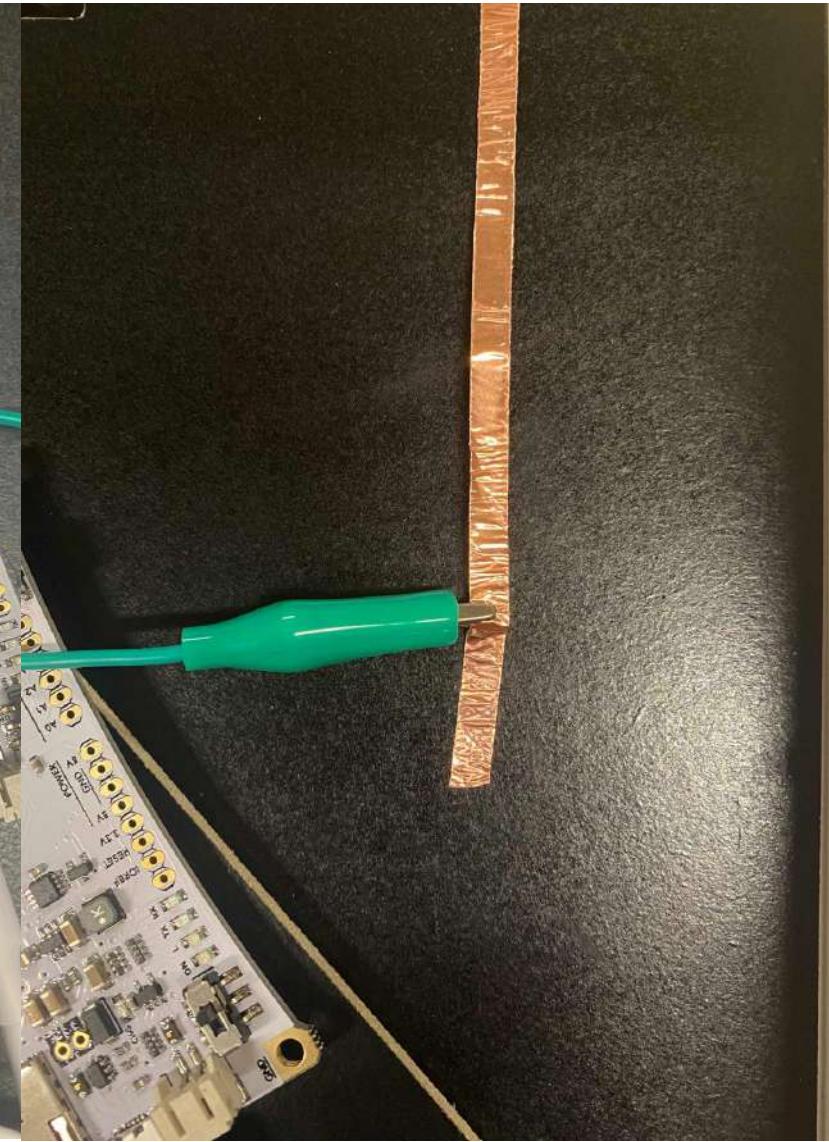
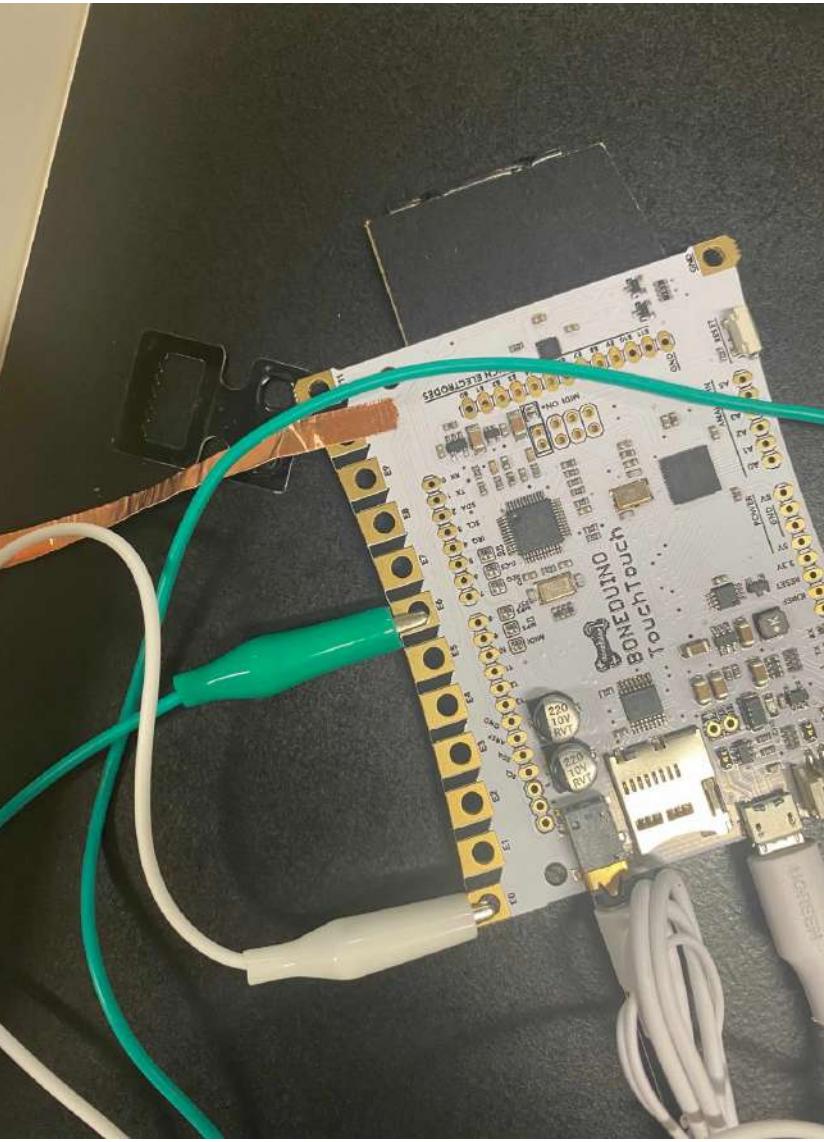
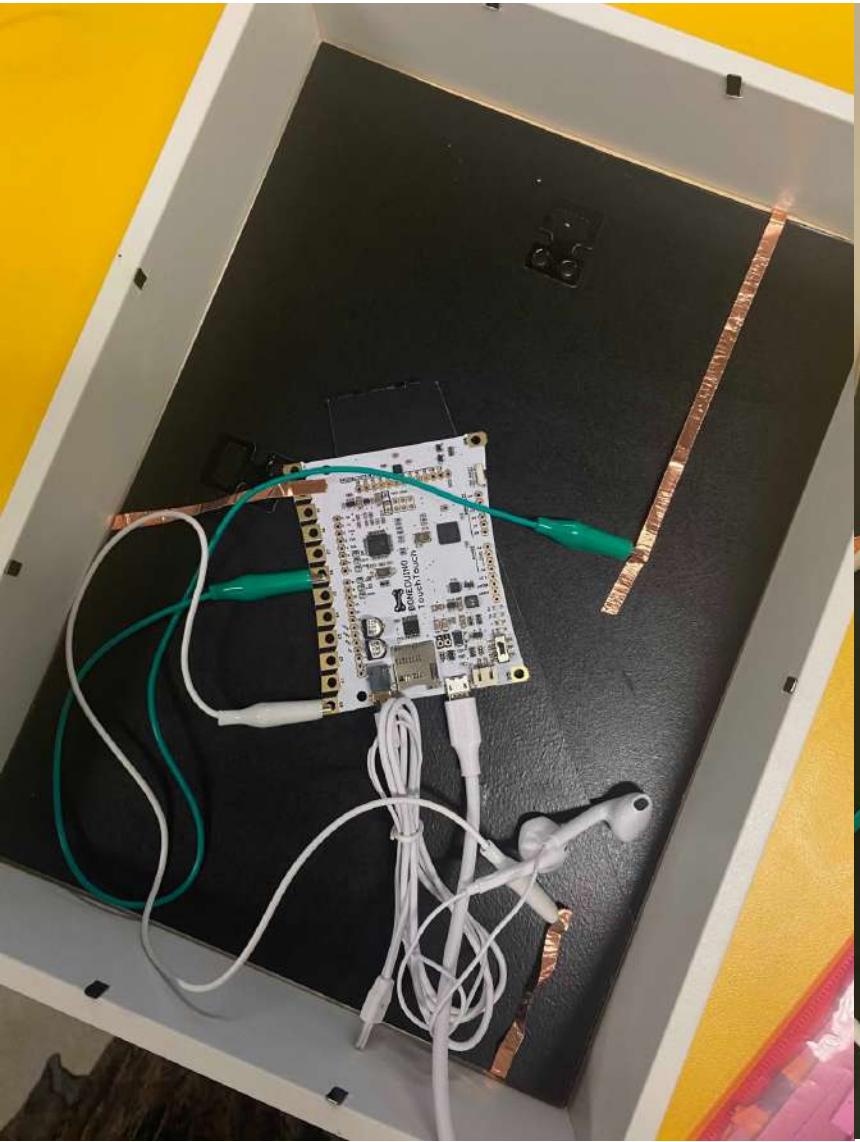
把開發板準備好
接上USB連接線及耳機



第七步

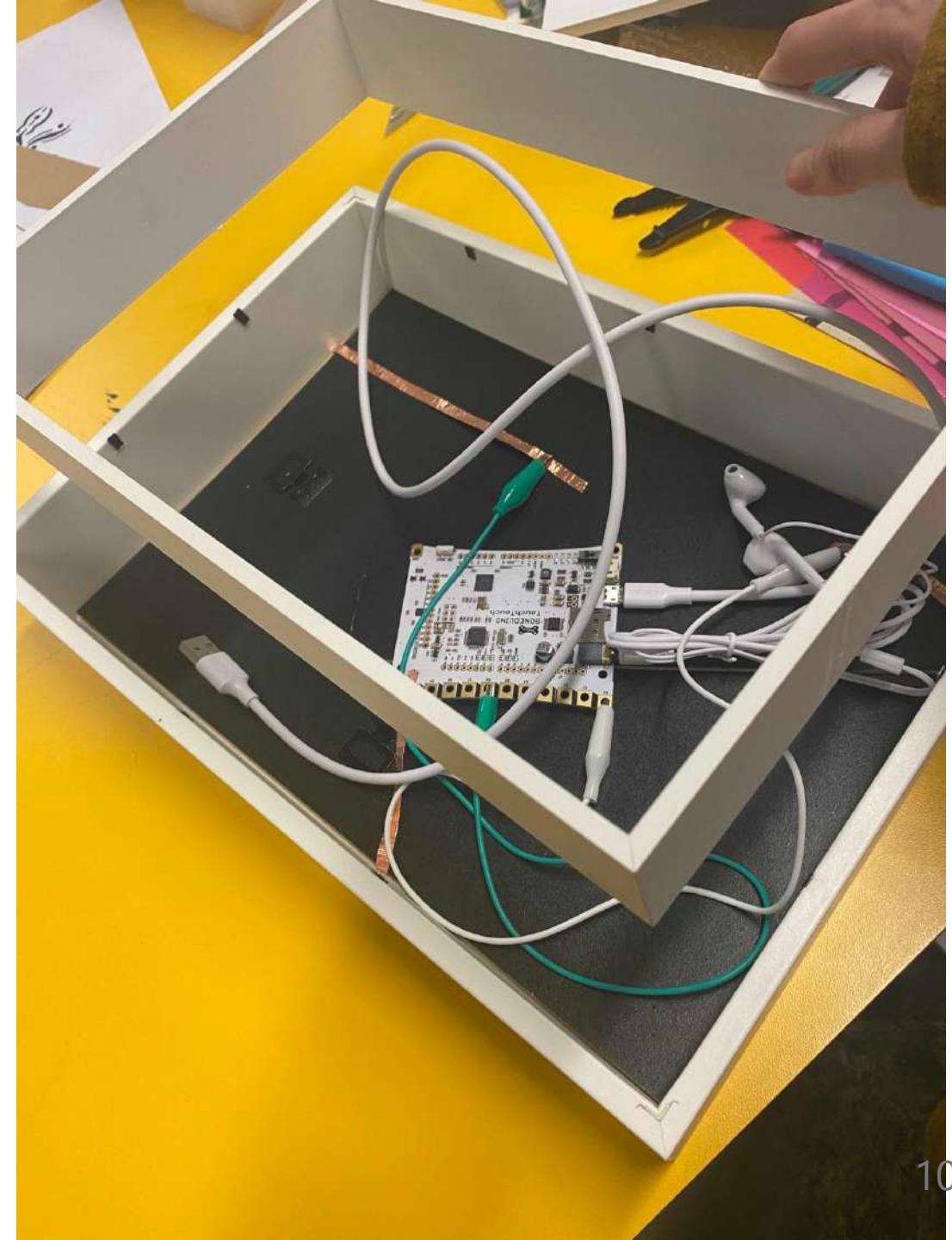
利用電夾連接開發板及導電銅箔膠帶





第八步

把畫框背部木框安置



第九步

把畫框背部的小鐵片把木框固定





組裝完成效果



- WELCOME TO -

賽馬會科藝共融計劃

JC-Project-IDEA

Day02

下午

sketch_apr29c §

```
#include "Compiler_Errors.h"

#include <MPR121.h>
#include <MPR121_Datastream.h>
#include <Wire.h>

#include <SPI.h>
#include <SdFat.h>
#include <FreeStack.h>
#include <SFEMP3Shield.h>

const uint32_t BAUD_RATE = 115200;
const uint8_t MPR121_ADDR = 0x5C;
const uint8_t MPR121_INT = 4;

const bool WAIT_FOR_SERIAL = false;

const bool MPR121_DATASTREAM_ENABLE = false;

uint8_t result;
uint8_t lastPlayed = 0;

SFEMP3Shield MP3player;

const bool REPLAY_MODE = true;

SdFat sd;

void setup() {

}

void loop() {

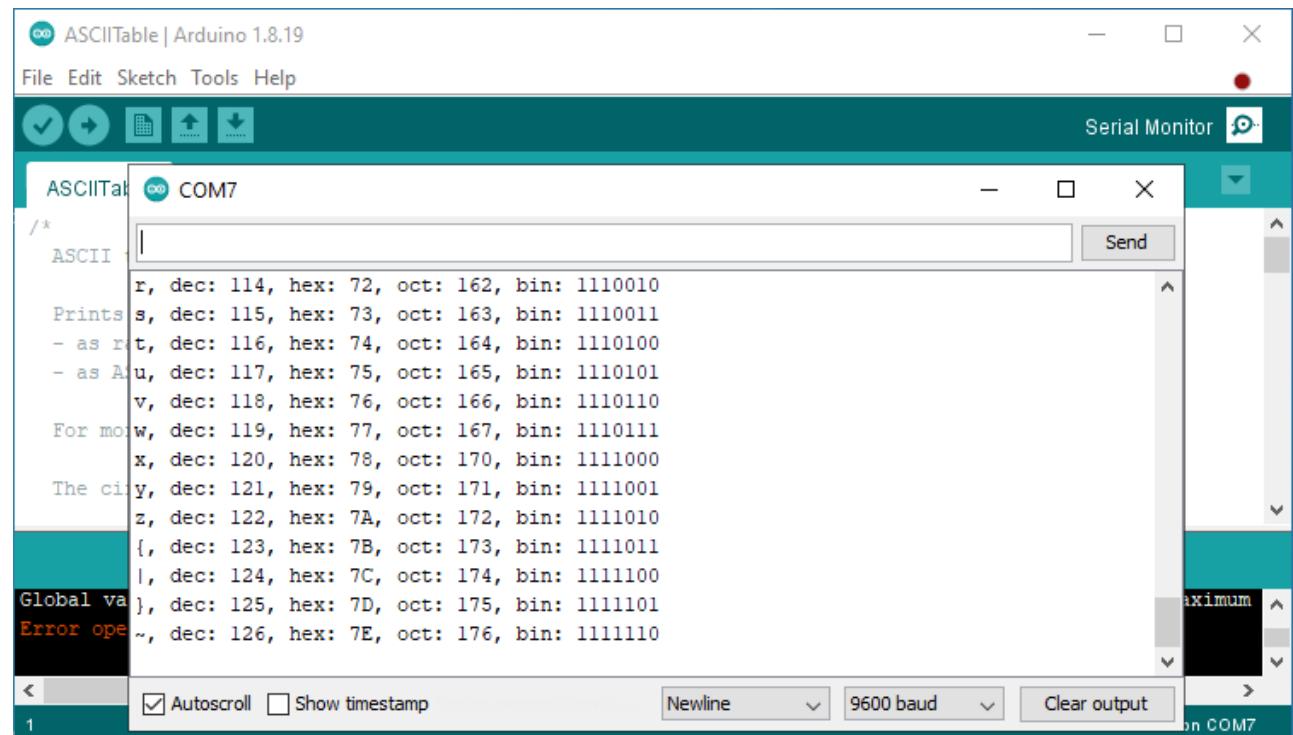
}
```

編碼內容

- *Serial.begin();*

Serial.begin() 是 Arduino 程式語言的一個函式，用於啟動序列通訊介面，在程式執行時可以透過該介面與外部設備或電腦進行資料傳輸。

- 右圖內的是 Serial Monitor



有關上一段編碼的補充

- *pinMode();*

`pinMode()` 是 Arduino 程式語言的一個函式，用於設定腳位的輸入或輸出模式，以便與外部電路或設備進行數位或類比訊號的傳輸。

- *Serial.println();*

`Serial.println()` 是 Arduino 程式語言的一個函式，用於透過序列通訊介面輸出文字或數字訊息，並在最後自動換行。

除錯階段

sketch_apr29c §

```
#include "Compiler_Errors.h"

#include <MPR121.h>
#include <MPR121_Datastream.h>
#include <Wire.h>

#include <SPI.h>
#include <SdFat.h>
#include <FreeStack.h>
#include <SFEMP3Shield.h>

const uint32_t BAUD_RATE = 115200;
const uint8_t MPR121_ADDR = 0x5C;
const uint8_t MPR121_INT = 4;

const bool WAIT_FOR_SERIAL = false;

const bool MPR121_DATASTREAM_ENABLE = false;

uint8_t result;
uint8_t lastPlayed = 0;

SFEMP3Shield MP3player;

const bool REPLAY_MODE = true;

SdFat sd;

void setup() {

}

void loop() {

}
```

有機會遇上的問題

1. 在編碼中遺漏了特定符號 如 ; () ,
2. 編碼不在 {} 之內

留意!!! Syntax 句法

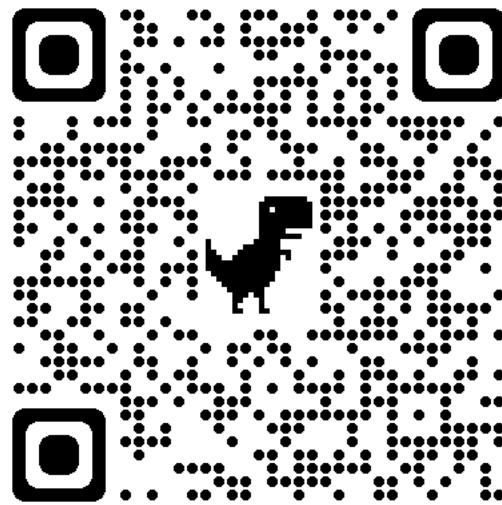
- 指一門語言裡支配句子結構, 直到組成句子的規則或過程。
<https://en.wikipedia.org/wiki/Syntax>

如發現編碼排位混亂 可以使用 CTRL + T 自動整理編碼功能



Physics of AUM/OM mantra made
visible - CYMATICS - Sound of
Creation

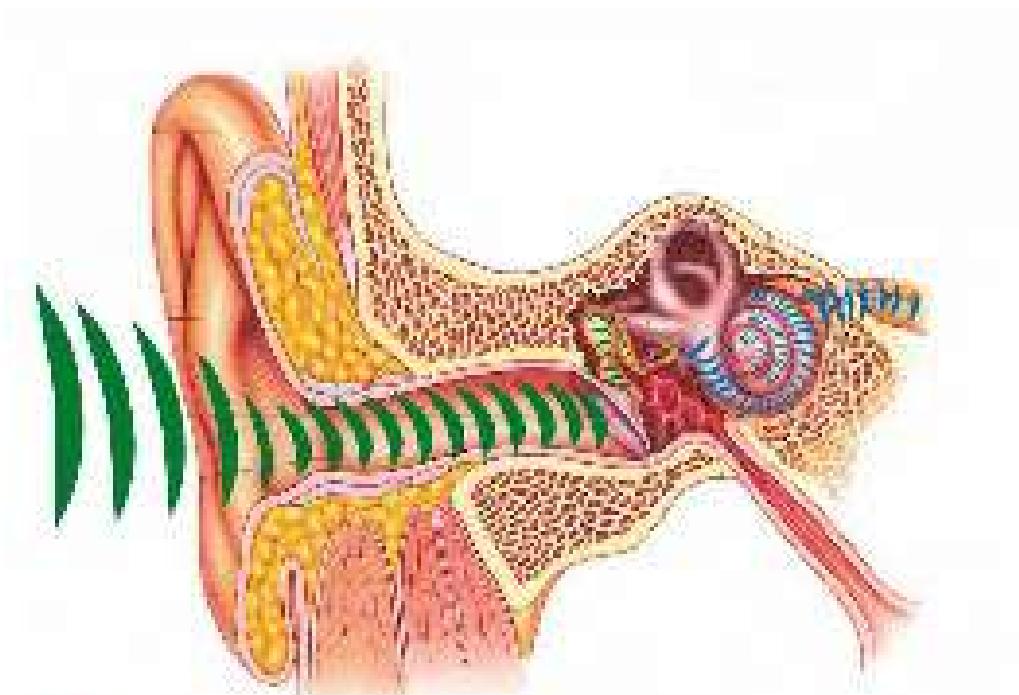
[https://www.youtube.com/watch?
v=fozINrtdzrQ](https://www.youtube.com/watch?v=fozINrtdzrQ)



聲音是？

聲音是振動產生的聲波，通過介質（氣體、固體、液體）傳播並能被人或動物聽覺器官所感知的波動現象。

聲音的頻率一般會以赫茲表示，記為Hz，指每秒鐘週期性震動的次數。而分貝是用來表示聲音強度的單位，記為dB。



綠色代表聲波以震動能傳遞到耳膜及聽小骨

藍色代表由內耳轉換為電波能

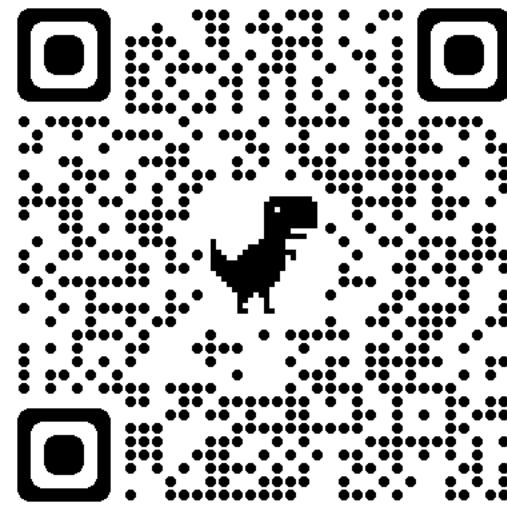
再經由聽神經傳達至大腦聽覺區

Oscillator 振盪器

Oscillator 是指能夠產生定期波形的電子電路或元件，可用於產生時鐘訊號、調整頻率和相位等。在電子學和通訊系統中，oscillator 常用於產生高頻信號，如無線電訊號，以及在數位電路中作為時序控制器。

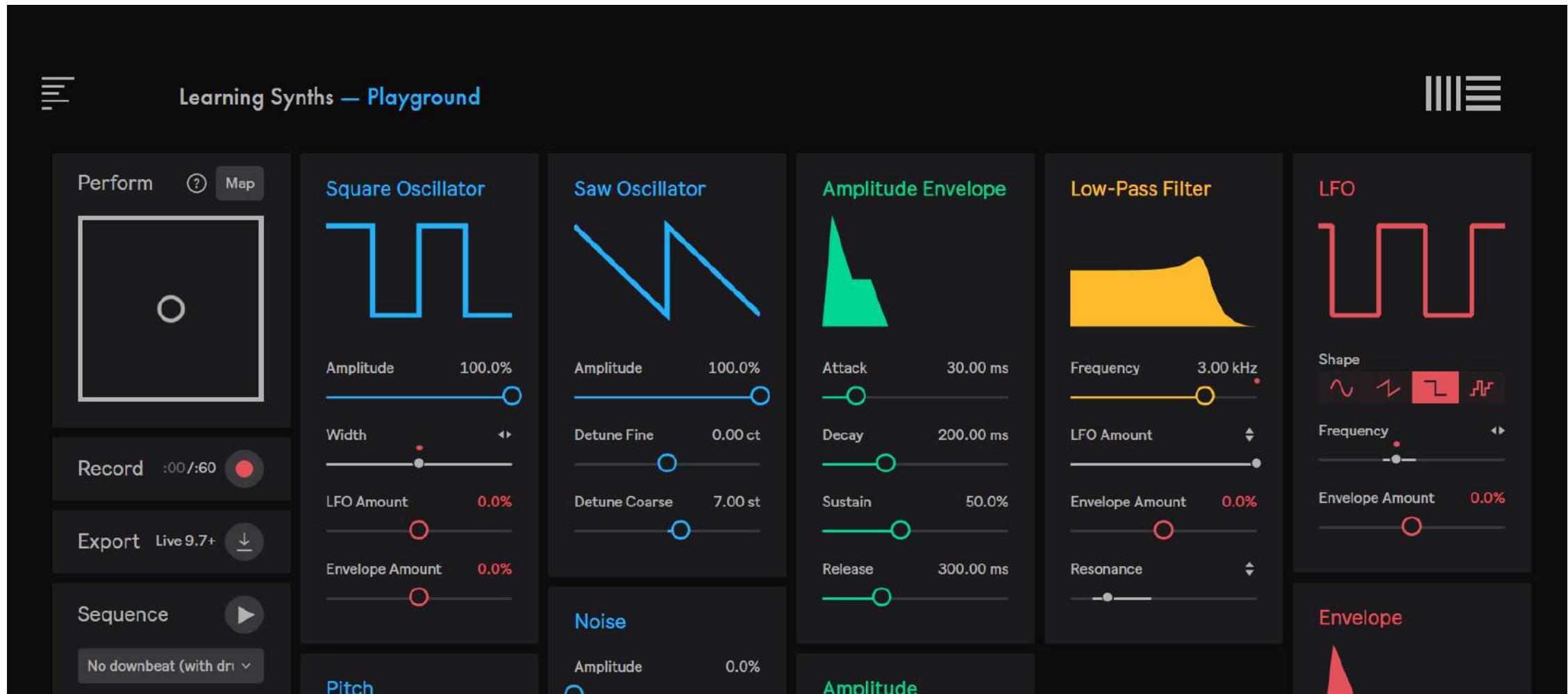
A Brief History of Synthesizers

[https://www.youtube.com/watch?
v=5sjreF6H_rY](https://www.youtube.com/watch?v=5sjreF6H_rY)



學一下 Synthesizer (合成器)

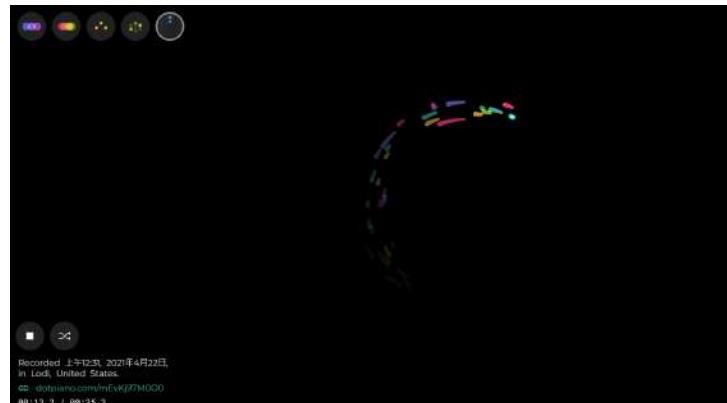
<https://learningsynths.ableton.com/en/playground>



有趣的網上音樂創作工具



<https://splice.com/sounds/beatmaker/026f92d3afda>



<https://dotpiano.com/mEvKj97M0O0>

除錯時段

作品在硬件上或編碼上有任何問題的話，請告知身邊的工作人員及老師。

簡報時間

現在邀請大家分享自己的作品

(※今まで)

こうじゃなくて



最後一節完成