



微控制器應用實習

FINAL PROJECT PRESENTATION

Arduino 音樂播放器 實習報告

班級 Class

大安高工 控制二甲

座號 No.

3 號

學生 Student

徐家瑤

指導老師 Instructor

張漢民 老師

KEY TECHNOLOGIES

 OneButton Lib

 Buzzer

 LED Indicator

 Wokwi Sim

 Non-blocking

DATE

2026-03-11



音樂播放器專案

Arduino x OneButton

PROJECT OVERVIEW

這是一個什麼專案？



按鈕控制音樂

利用 Arduino 開發板和按鈕開關，製作一個可以播放音樂的小裝置。透過按壓按鈕，我們可以控制音樂的播放與停止。



智慧判斷按法

程式能夠分辨你是「短按一下」還是「長按不放」，分別執行不同的功能（如：播放/暫停 vs 從頭播放）。



聲光效果互動

除了聽到音樂，還有 LED 燈會隨著音樂播放狀態閃爍，讓操作更有回饋感。



什麼是 Arduino ?

Arduino 是一個開源的電子原型平台，包含硬體（各種型號的開發板）和軟體（Arduino IDE）。它就像是一個小型的電腦大腦，可以接收感測器的訊號，並控制燈光、馬達或其他裝置。

核心概念

輸入 (Input) → 處理 (Process) → 輸出 (Output)



開源 (Open Source)

軟硬體設計皆公開，全世界的開發者都能使用、修改與分享。



簡單易學

專為初學者設計，程式語法直觀，不需要深厚的電子背景也能上手。



擴充性強

有豐富的擴充板 (Shields) 和感測器模組可供搭配使用。



跨平台

開發環境支援 Windows, Mac OS 和 Linux 系統。

常見應用領域





發光二極體

Light Emitting Diode (LED)

類型	數位輸出 (Digital Output)
腳位	長腳(+), 短腳(-)
驅動	高電位 (HIGH) 發光

工作原理

1. 單向導電性

LED 是二極體的一種，電流只能從正極（陽極）流向負極（陰極）。如果接反了，電流無法通過，LED 就不會亮。

正極 (+) 長腳 → 負極 (-) 短腳

2. 需要限流電阻

LED 的電阻很小，直接接上電源會導致電流過大而燒毀 LED。因此必須串聯一個電阻（通常 220Ω 或 330Ω）來限制電流。

接線方式

- 1 將 LED 的長腳 (正極) 串聯一個 220Ω 電阻。
- 2 電阻的另一端接到 Arduino 的數位腳位 (例如 Pin 13)。
- 3 將 LED 的短腳 (負極) 接到 GND (接地)。





蜂鳴器

Piezo Buzzer

類型	數位/類比輸出 (Output)
種類	有源 (Active) / 無源 (Passive)
控制	tone() / PWM 訊號

工作原理與種類區分

● 無源蜂鳴器 (Passive)

內部無振盪源，通電不會直接叫。需要輸入不同頻率的方波訊號才會發聲。

優點：可以控制聲音頻率，製作出不同音階的音樂。

● 有源蜂鳴器 (Active)

內部自帶振盪源，一通電就會發出固定頻率的聲音（通常是單一音調）。

優點：程式控制簡單，適合做警報聲。

PWM Square Wave Signal 🎵

🔌 接線與程式控制 (tone 函式)

- 1 蜂鳴器長腳 (正極 +) 接 Arduino 數位腳位 (如 Pin 8)。
- 2 蜂鳴器短腳 (負極 -) 接 GND (接地)。
- 3 使用 `tone(PIN, frequency, duration)` 發聲。

</> 關鍵函式語法：

```
tone(8, 262, 500); // 在 Pin 8 發出 Do (262Hz)
```



1



變數 (Variable)

會改變的數值

就像一個置物箱，你可以給它一個名字（例如 score），然後隨時把裡面的東西（數值）換掉。

分數

計時

按鈕狀態

2



常數 (Constant)

不會改變的數值

就像刻在石頭上的字，一旦定義了就不能修改。通常用來設定硬體腳位，避免程式出錯。

腳位編號

圓周率

固定參數

basic_logic.ino

```
// --- 常數區 (設定後不能改) ---  
const int BTN_PIN = 2; // 按鈕接在腳位 2  
const int LED_PIN = 13; // LED 接在腳位 13  
  
// --- 變數區 (程式執行中會變) ---  
int clickCount = 0; // 記錄按了幾次  
bool isPlaying = false; // 記錄是否在播放  
  
void setup() {  
    // 使用常數來設定腳位，好讀又安全  
    pinMode(BTN_PIN, INPUT_PULLUP);  
    pinMode(LED_PIN, OUTPUT);  
}  
  
void loop() {  
    // 假設按鈕被按下 ...  
    if (digitalRead(BTN_PIN) == LOW) {  
        clickCount = clickCount + 1; // 變數值+1
```



1



if (如果...)

條件成立才執行

像是一個守門員，檢查條件是否為真 (True)。如果符合條件，就執行大括號 { } 內的程式碼。

✓ 按鈕按下時

✓ 分數大於60



2



else (否則...)

二選一的路徑

當 if 的條件**不成立**時，就會執行 else 區塊的內容。用來處理「另一種情況」。

✗ 按鈕放開時

✗ 音樂停止時

```
logic_control.ino

// --- 專案邏輯範例 ---
void loop() {

  // 1. 判斷音樂是否正在播放
  if (isPlaying == true) {

    // 如果是 (True): LED 閃爍
    flashLED();
    playMelody();

  } else {

    // 否則 (False): LED 保持亮著，安靜
    digitalWrite(LED_PIN, HIGH);
    noTone(BUZZER_PIN);

  }

  // 2. 判斷按鈕長按 (OneButton 內部邏輯)
  // if (buttonDuration > 800) { 重播音樂 ... }

}
```



1



開機初始狀態

當 Arduino 接上電源啟動後，音樂預設為**停止**狀態，且 LED 指示燈保持**恆亮**，等待使用者操作。

2



按一下 (Click)

短按按鈕一下，可切換播放狀態：
若正在播放則**暫停**；若停止中則**開始播放**。

3



長按 (Long Press)

持續按住按鈕不放（約 1 秒），會觸發重置功能，音樂將直接**從頭開始播放**。

4



LED 狀態指示

提供視覺回饋：

- 音樂播放中：LED **規律閃爍**
- 音樂停止時：LED **停止閃爍**

實作成果展示

Hardware Implementation Result



作品完成度

★★★★★ (接線整潔，功能完整)

1 核心控制器

使用 Arduino 相容開發板 (Nano)，結合多功能擴充板，方便杜邦線插拔與電路配置。

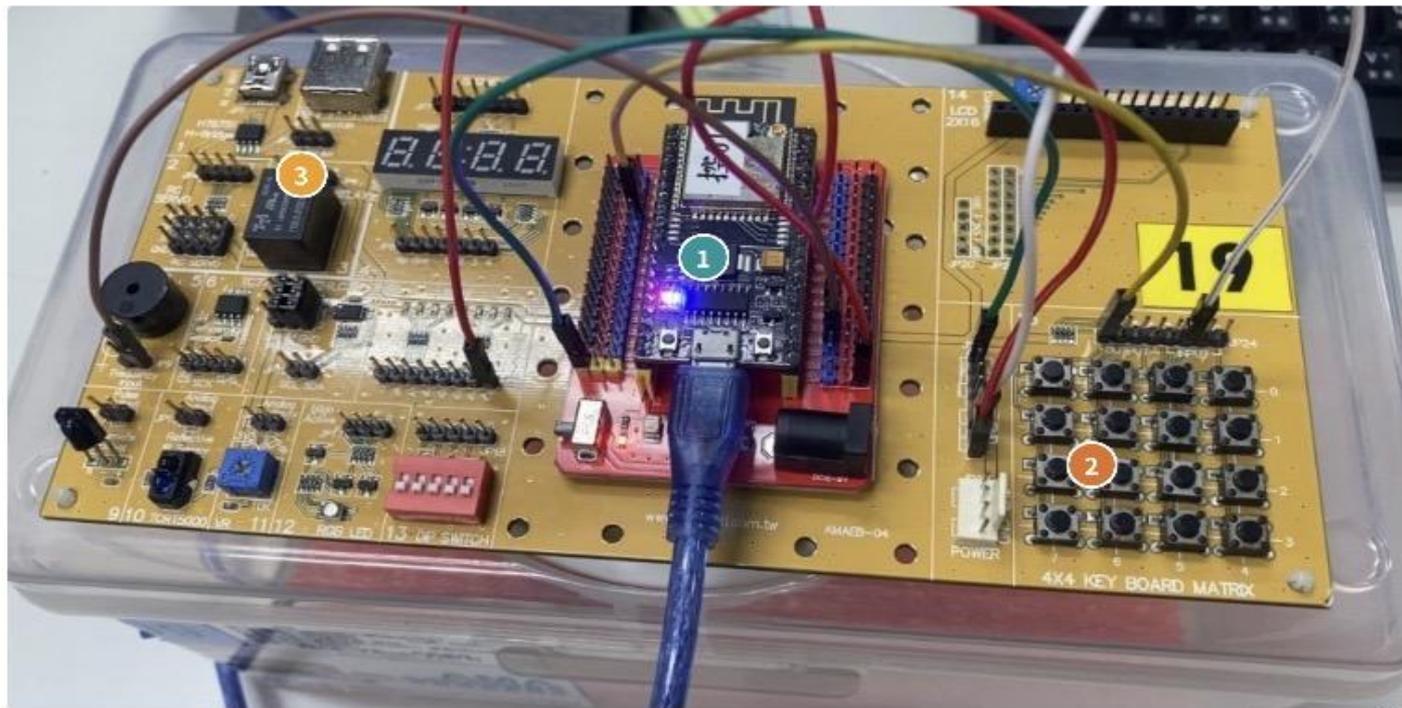
2 輸入控制區

利用矩陣鍵盤或獨立按鈕模組作為訊號輸入來源，透過程式邏輯判斷 Click 與 Long Press。

3 周邊電路

包含蜂鳴器 (Buzzer) 負責旋律輸出，以及 LED 指示燈 (左側區域) 顯示播放狀態。

✓ 硬體測試通過
所有接線已確認無誤



實惠接線圖



“「程式設計不只是寫出能動的程式碼，更是學習如何將複雜的問題，拆解成一個個可執行的邏輯步驟。」

💡 核心學習收穫

- ✔ **非阻塞式編程：**學會使用 `millis()` 取代 `delay()`，讓 LED 閃爍與按鈕偵測可以同時進行，互不卡頓。
- ✔ **函式庫應用：**透過 OneButton Library 輕鬆實現 Click 與 Long Press 的判斷，大幅簡化了程式碼。
- ✔ **硬體整合能力：**親手完成電路接線，了解數位輸入(按鈕)與輸出(蜂鳴器、LED)的控制原理。

🔧 遇到的挑戰

- 🔧 **按鈕彈跳問題：**一開始按一下會觸發多次訊號 (Bouncing)，後來透過函式庫內建的去彈跳機制解決了不穩定的現象。
- 🔧 **接線接觸不良：**麵包板接線時 LED 時亮時不亮，經過三用電表檢查後，發現是地線 (GND) 沒有插好。
- 🔧 **邏輯衝突：**在長按重置時，原本會誤判為短按，調整參數時間後成功區分兩種動作。

▶▶ 未來展望

- 🔧 **增加顯示螢幕：**加入 LCD 1602 或 OLED 螢幕，顯示目前的歌曲名稱與播放進度條。
- 🔧 **多曲目切換：**利用雙擊 (Double Click) 功能來切換下一首歌曲，讓播放器功能更完整。
- 🔧 **外殼製作：**學習 3D 列印技術，為這個電路設計一個專屬的外殼，使其成為真正的產品。