

中華民國第 63 屆中小學科學展覽會
作品說明書

國小組 生活與應用科學科(一)

佳作

082821

STOMP-自製電音 BAND

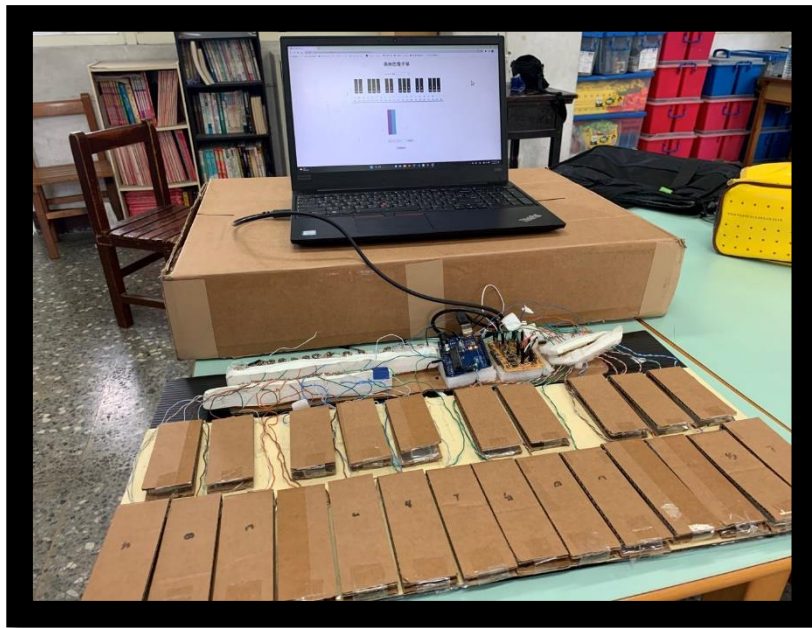
學校名稱：臺中市大里區大元國民小學

| | |
|---------------------------------------|-------------------------|
| 作者： 小五 施天海 小六 陳宣頤 小六 李昕穎 | 指導老師： 洪嘉黛 林建毅 |
|---------------------------------------|-------------------------|

關鍵詞：Arduino、樂器

摘要

本研究目標旨在完成多人合奏、原音重現的自製樂器，透過 Arduino 作為傳遞訊號的中介，並以電腦對應輸出不同樂器的實際音色。在系統架構上，我們利用 Arduino 的類比訊號，讀取電壓值轉換成字串對應音階，實際錄製我們想要的樂音建立音源資料庫，以視覺化圖形輔助音階表現，隨時對設備進行調整，完成撥放系統的模組。實驗過程中，我們測試了兩個世代的訊號傳輸系統，最終以分壓原理為知識架構，焊接可變電阻，形成有效、穩定的訊號傳輸。其次，以生活上的紙箱素材、Cat6 雙絞銅線、鋁箔紙自製耐受恢復力高、導電性高的琴鍵結構。最終，透過以上的研究歷程製作出**虛實**整合的電音樂團(BAND)。



壹、前言

一、研究動機

我們就讀的是一所有打擊樂團、直笛團的音樂型發展學校。在校園中，音樂演奏是生活的一部分，我們也參加打擊樂團的練習與表現，主要練習馬林巴琴的演奏。馬林巴琴是一種昂貴的大型敲擊樂器，平時除在校可使用馬林巴琴練習外，在家的練習僅能使用自製的練習琴，而練習琴只能產生節拍噪音而無樂音，在練習時常想，如果在家練習也能如同在校一樣演奏好聽的音樂就好了。

此時，我們想起打擊樂啟蒙樂團：STOMP。STOMP（破銅爛鐵樂隊）是一個結合了敲擊音樂、舞蹈，與情境喜劇的表演團體(圖 1-1)。STOMP 運用日常用品如掃把、垃圾桶蓋、打火機... 等等破銅爛鐵發出富有不同音頻、豐富音樂性的節奏。因此，我們開始想動手打造減少噪音、加入樂音、多人演奏，屬於我們的自製敲擊琴。



圖 1-1 STOMP 劇照 (摘自網路)

在研究初期，我們查詢在過去的研究中是否有類似的作品。我們透過「音樂」、「樂器」等關鍵字進行作品摘要的搜尋，獲得許多科展相關文獻，其中以民國 72 年高中組應用科學科的「新式電腦音樂合成器」、民國 106 年高中組電腦與資訊學科的「MIDI 樂器（電子琴）」、民國 109 年國小組生活應用科學(一)「微鋼琴三部曲」最為相關，站在巨人的肩膀看世界，其作品分析詳見文獻探討(p.5)。

最後，透過我們的想像與過去科展的研究，決定以回收材料自製多個的鍵盤樂器，在校期間錄製不同樂器的樂音，使用 Arduino（微電腦）分別控制不同的電子訊號，由電腦（家用桌機或筆電）同時判定不同訊號，發送至電腦並同時撥放，達到多人同時演奏同一大組樂器，卻能產生不同樂音自製電音 Band。(圖 1-2)

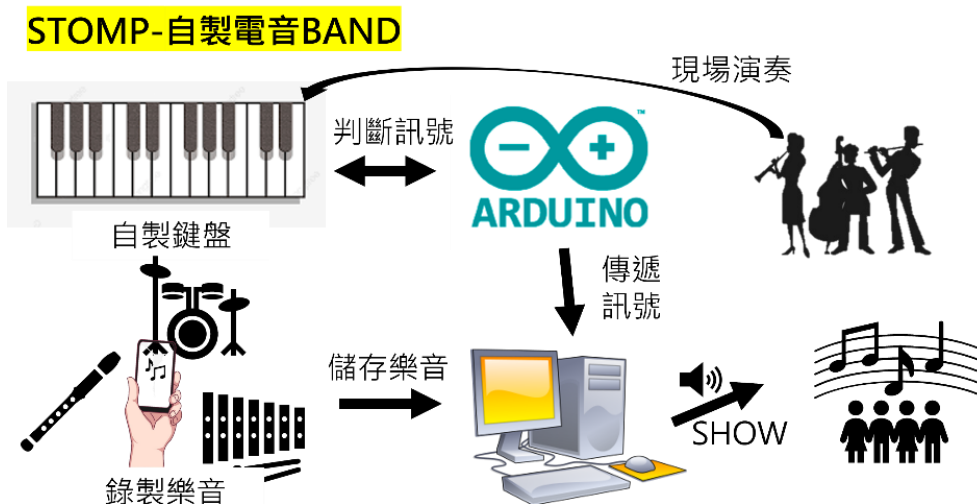


圖 1-2 本研究的想像與實際呈現目標

二、研究目的及問題

(一)自製多點電位差打擊樂器的系統架構

1. Arduino 在打擊樂器的偵測上扮演何種角色？
2. 音源設計第一代：離線 HTML5 API 如何掛載多音音色的開源音檔？
3. 音源設計第二代：如何實際錄製不同樂器的樂音並有效撥放呢？
4. 離線 HTML5 API 如何加入視覺化圖形幫助實測呢？

(二)測試有效的訊號傳輸系統

1. 不同傳輸導線電阻值的大小為何？
2. 第一代薄膜鍵盤的訊號傳輸效果為何？
3. 第二代 Arduino 訊號傳輸系統的修正過程為何？

(三)設計符合可攜式、可耐用的琴鍵結構

1. 不同材質、結構設計的琴鍵觀察比較為何？
2. 延續問題 2-1，其恢復力(抗變形)的能力為何？
3. 延續問題 2-1，其耐受力(有效導電)的能力為何？

(四)實際展現我們設計的敲擊樂器

三、文獻探討

(一) 歷屆科展作品分析

在研究初期，我們查詢全國歷屆科展的得獎作品，想了解在過去的研究中是否有類似的作品產生。首先以「音樂」為關鍵字進行作品摘要的搜尋，獲得 66 筆資料；進一步使用作品名稱搜尋時，獲得 22 筆資料，但報告書也包含「程式、電」等關鍵字的，僅有 4 篇。包含 1 篇國小、1 篇國中、2 篇高中的報告書，其中以民國 72 年高中組應用科學科的「新式電腦音樂合成器」最為相關。

接著，我們改以「樂器」為關鍵字進行作品摘要的搜尋，獲得 52 筆資料；進一步以作品名稱搜尋時，獲得 11 筆資料，包含 6 篇國小、1 篇國中、4 篇高中的報告書，其中以民國 106 年高中電腦與資訊學科的「MIDI 樂器（電子琴）」、民國 109 年國小組生活應用科學(一)的「微鋼琴三部曲」最為相關。當我們進一步分析三篇文獻後，自「微鋼琴三部曲」科展報告書中又找到相關文獻：民國 106 年國小生活與應用科學科的「真"水"鋼琴」。其相關文獻比較資料(表 1-1)，分析如下：

表 1-1 歷屆科展作品與本研究比較分析

| 作品名稱 | 新式電腦音樂合成器 | 真"水"鋼琴 | MIDI 樂器 (電子琴) | 微鋼琴三部曲 | STOMP-自製電音 BAND |
|------|-----------|--------------|---------------|-----------------------|--------------------------------|
| 屆次 | 民國 72 年 | 民國 106 年 | 民國 106 年 | 民國 109 年 | 民國 111 年 |
| 組別 | 高中 | 國小 | 高中 | 國小 | 國小 |
| 輸出方式 | 電腦 (MIDI) | 電腦 (Scratch) | IC 晶片+喇叭 | 電腦 (micro:bit) | 電腦 (開源網頁) |
| 音檔來源 | MIDI 音檔 | Scratch 內建音源 | MIDI 音檔 | Micro:bit 內建音源 | 實際收錄 真實樂器音 |
| 演奏方式 | 電腦鍵盤 | 自製按壓鍵盤 | 鋼琴鍵盤 | 自製按壓鍵盤 | 自製 敲擊鍵盤 |
| 播放音階 | 無說明 | 無說明 | 無說明 | 約 8-10 音階 | 12-72 音階 |
| 鍵盤製作 | 無說明 | 研究不同水溶液的導電效果 | 設計不同的矩陣鍵盤及函式 | <u>描述比較</u> 鍵盤的導電性 | 數據比較 鍵盤導電性與 耐受恢復力 |

本研究期待結合上述多篇科展研究的優點，透過不同的製作過程見證了時代的進步。我們以「微鋼琴三部曲」作品為基礎，以回收材料自製鍵盤並**數據分析**鍵盤優勢，並將「彈奏」轉換為「敲擊」；捨棄原有的音源撥放方式，將「軟體合成音樂」轉換為「真實錄製樂音」，透過 Arduino 突破撥放樂音的數量，符合多樂器、同時演奏的樂團目標。

(二) 系統平台-Arduino 微電腦

Arduino 是一款微控制器板，附帶易於學習和使用的開發環境 (IDE)，其特色為開發簡便且資源豐富。使用者可以透過 Arduino 板接上多種電子元件，例如 LED 燈、喇叭、馬達、開關、溫濕度感測器、紅外線發射與接收器、LCD 顯示器，以及各種通訊模組。配合自動控制的程式，就能輕鬆實現各種自動化應用，例如使用溫度感測器控制風扇運轉、調整燈光亮度、控制馬達轉速、使用紅外線遙控家電、控制機械手臂或機器人、以及製作自走車、飛行器等。

一般常見的 Arduino 有兩種，Arduino UNO 及 Arduino NANO(圖 1-3、圖 1-4)，大小約與手掌差不多，利用 USB 線接上電腦後，透過電腦程式語言的撰寫，再裝上其他電子裝置或通訊模組，就能做到許多不同的事情，這也是一台電腦的基本模型。



圖 1-3 Arduino UNO



圖 1-4 NANO(右)

(三) 琴鍵設計初探

1. 壓電裝置

根據 ROHM 公司網頁，壓電元件是一種透過施加壓力產生電壓 (壓電效應)，或透過施加電壓產生變形 (逆壓電效應) 的元件，包括多種壓電開關，如各種開關 (圖 1-5) 和微動開關 (圖 1-6)。在與 Arduino 搭配的感測應用中，常使用數位 I/O 訊號。



圖 1-5 常見的壓電開關



圖 1-6 本研究預計使用的微動開關

2. 鍵盤矩陣 (Keyboard Matrix)

所有的鍵盤控制器都會需要一個 I/O 腳偵測每一個按鍵的開或關狀態，因此當按鍵越多，所需的 I/O 就要越多。但多數的控制器都沒有這麼多 I/O 腳，因此必須透過鍵盤矩陣，將電路分成行與列，透過掃描的方式逐一掃描行與列，利用偵測電位差，驗證按鍵的位置。

透過上述原理，我們首先拆卸一電腦鍵盤作為鍵盤矩陣的基礎，並以電線連接至不同紙板，透過不同金屬的導通與敲擊回饋，回傳至 Arduino 進行演算。

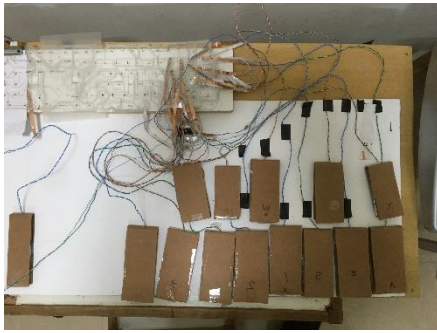


圖 1-7 以紙板為敲擊回饋測試



圖 1-8 以銅箔為導通按鍵測試

3. 分壓原理

呈上壓電開關的文獻可知，若以微動開關進行研究，可能會遇上以數位訊號分析不同音階的情況，若需要開源更多音階的設定，僅用數位輸入的方式明顯不足，因此改用 Arduino 開發板的類比輸入腳位(A0 至 A5 腳位)，偵測每瞬間通過類比腳位的電壓，並以該電壓值的結果轉換成訊號（字串），再由該字串對應鍵盤上的音階，達到訊號連結音階的效果。

在實驗過程中，我們事先將 A0 腳位串接可變電阻，接著旋轉可變電阻(如圖 1-9)，調整電阻大小，並觀察 Arduino 的序列埠上的電壓偵測數值，結果發現確實在不同電阻值下會有不同電壓值。上網查找電阻分壓原理公式帶入(如圖 1-10)，則可以計算出電阻與電壓的關係。

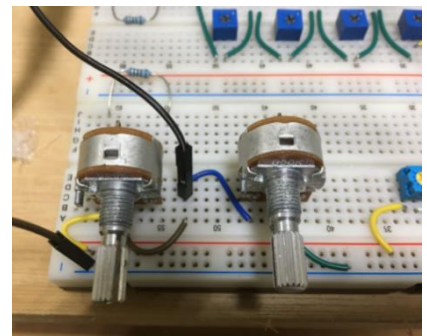


圖 1-9

旋轉 R1、R2 觀察不同的電壓值

理論上，Arduino 所用 AVR 晶片類比解析度理論值為 0.00489V (5V/1023)。但透過實測可知，Arduino 類比腳位僅可辨 0.5V 的電位差，加上最小值的 0V 及最大值的 5V，單一類比腳位約可辨識無按壓及 12 個音階。

$$V_{out} = V_1 \frac{IR_2}{I(R_1 + R_2)} = \frac{V_1 R_2}{(R_1 + R_2)}$$

分壓電路 公式

輸出電壓A0 = 輸入電壓vcc * 電阻r2 / (電阻r2 + 電阻r1)

電阻r1=0 則 輸出電壓5V = 輸入電壓5V * 電阻r2 / (電阻r2+ 0)

電阻r2=0 則 輸出電壓0v = 輸入電壓5V * 0 / (0 +電阻r1)

琴鍵依序漸變遞增A0腳位偵測電壓，求相對應的r2電阻值

$$\begin{aligned} (r_2+r_1) * A0 &= vcc * r2 / (r2 + r1) \\ (r2*A0 + r1 * A0) &= vcc*r2 \\ r1 * A0 &= vcc* r2 - r2 * A0 \\ r1 * A0 &= r2 * (vcc - A0) \\ r1 * A0 &= r2 * (vcc - A0) \\ (r1 * A0) / (vcc - A0) &= r2 * (vcc - A0) / (vcc - A0) \\ (r1 * A0) / (vcc - A0) &= r2 * 1 / 1 \\ (r1 * A0) / (vcc - A0) &= r2 \quad // \text{-----解出 } r2 \text{ 電路阻值} \end{aligned}$$

電阻R1 琴鍵數

圖 1-10 模擬分壓電阻值計算

貳、研究設備及器材

一、研究設備

windows 10 系統筆電一台、Arduino UNO 一片

二、研究器材

鋁箔、銅箔、導電凝膠、2B 鉛筆(石墨)、多芯導線、網路線 Cat6 雙絞銅線、USB 薄膜鍵盤、陶瓷電阻、可變電阻、麵包板、游標卡尺、LED 燈泡。

參、研究過程或方法

一、研究架構圖

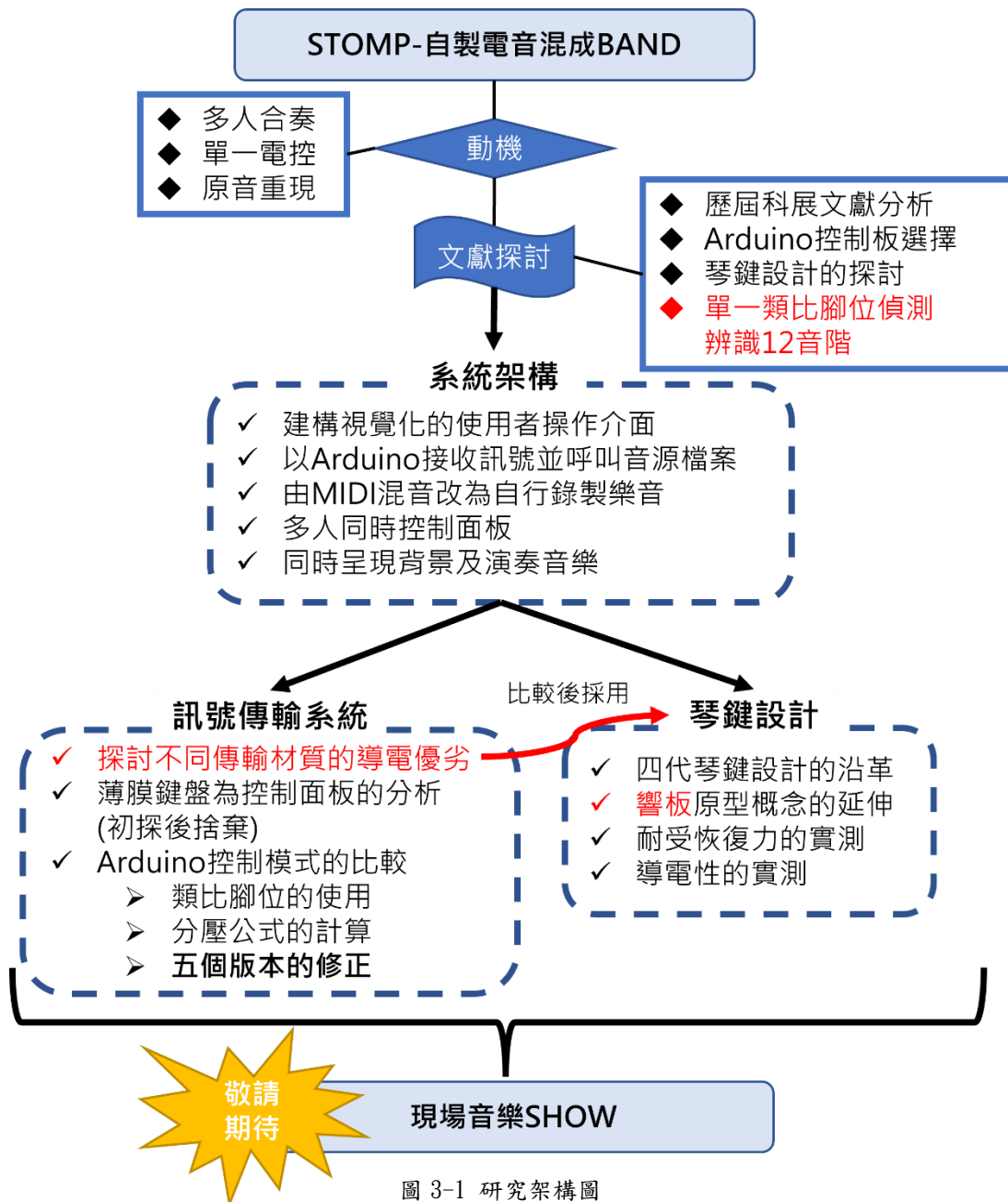


圖 3-1 研究架構圖

二、研究流程說明

研究一：系統架構

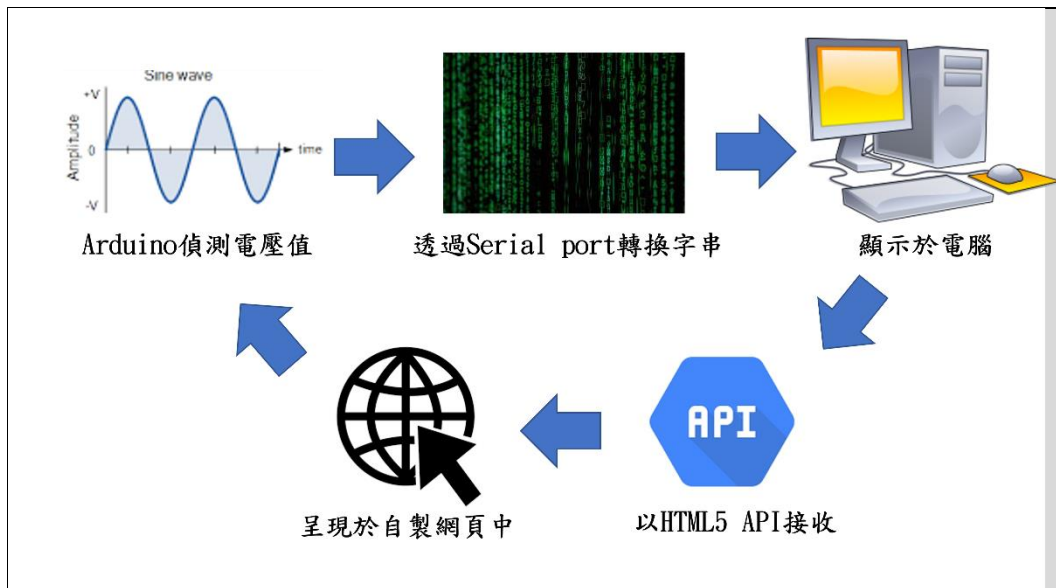


圖 3-2 程式的設計概念

本研究在系統架構的建立上，主要分為 Arduino 的電壓偵測與字串轉換程式、HTML5 的 API 接收與結合開源軟體的自製網頁呈現等(見圖 3-2)，其分述如下：

(一) Arduino 的程式設計

本研究中的 Arduino 微電腦板的應用簡單，主要來自於讀取 A0-A5 腳位的電壓，並轉換成字串顯示於電腦上，使用 HTML5 的 API 可以判讀字串。其程式流程圖(圖 3-3)如下：

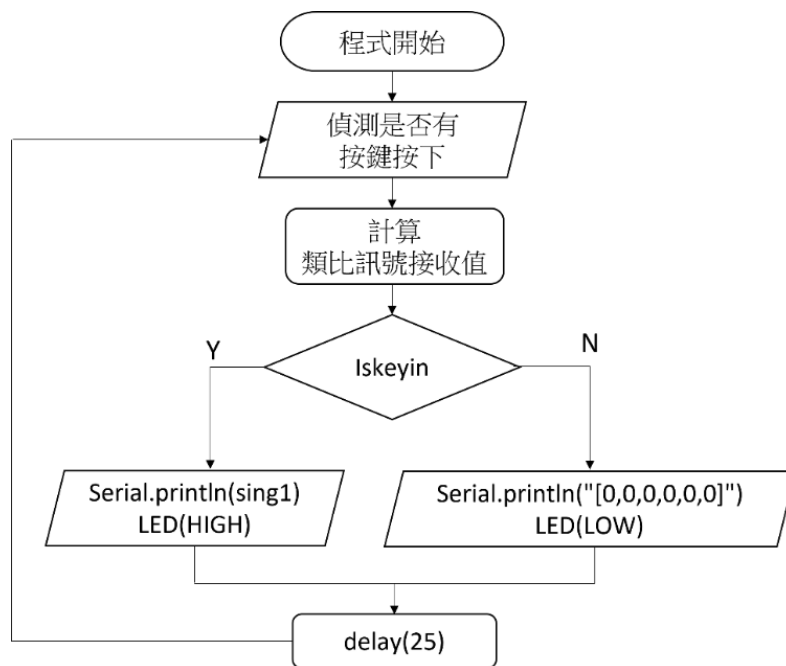


圖 3-3 Arduino 的程式流程圖

(二) HTML5 API 的設計

1. 建構使用者操作介面外觀。
2. 連結 Arduino 鍵盤，接收序列埠傳來的字串資料。
3. 建構離線網頁：連結鍵盤連線、序列埠連線視窗、選取 USB 序列裝置。
4. 網頁持續接收琴鍵傳來的字串資料，然後將字串轉換成陣列變數(A0-A5)。
5. 判別類比偵測電壓, 替換成鍵盤的 ASCII 送出按鍵，再轉換成相對應的 keycode 按鍵字符編碼。
6. 連結開源音源檔以播放聲音。
7. 原始碼參考來源

(1) Arduino 連線原始碼

(2) 視覺化

(三) 第一代音源：下載開源音檔

1. 開源音源檔
2. 本研究模擬 Keith William Horwood(2013) 分享在 GitHub 上的開源音檔，下載修正後做為字串呼叫的音源檔。



(四) 第二代音源：實際錄製樂音

經過市賽之後，我們持續修正琴組的聲音表現。我們期待能以更自然、更真實的樂音，來表現我們獨特的音樂性。因此使用直接演奏的獨立樂音呈現結果。

其次，透過 Arduino 訊號傳輸 2.5 版本的建立，線路的配置越發完整；琴鍵結構透過敲擊、導電測試後也越來越穩定。我們也將琴鍵配置，從原本的 12 鍵擴充至 36 鍵至 60 鍵。以不同樂音同時呈現的方式，表現多人合奏的目標。

1. 錄音實驗流程：

分五個不同樂器和三個不同音域來錄製樂音，並用網站將 m4a 轉成 mp3。

- (1) 直笛：我們請了直笛團的小朋友來替我們吹奏三個音域的樂音，並用手機將它錄製起來，最後轉成 mp3 檔。
- (2) 鋼琴：我們請了有學過鋼琴的同學來替我們彈奏三個音域的樂音，並用手機將它錄製起來，最後轉成 mp3 檔。
- (3) 木琴、鐵琴：我們請了打擊樂的老師來替我們敲擊三個音域的樂音，並用手機將它錄製起來，最後轉成 mp3 檔。



直笛



木琴



手搖鐘



鐵琴



鋼琴

2. 音檔測試

- (1) 將 m4a 轉成 mp3 檔：我們利用線上網站將錄音檔轉成 mp3 檔以方便程式運行
- (2) 音檔測試：我們將 m4a 轉成 mp3 檔後，就開始進行了音檔測試



手機原始錄音



轉 mp3



存檔



直笛音檔測試



木琴音檔測試



鋼琴音檔測試

研究二：訊號傳輸系統測試

訊號傳輸系統主要研究如何有效傳遞訊號、以何種傳輸方式（薄膜鍵盤、Arduino）將訊號傳至電腦，使電腦進行訊號的判讀。此部分分為三個實驗。第一是比較不同傳輸導線的優劣；其次是第一代訊號傳輸系統：薄膜鍵盤；最後是第二代訊號傳輸系統：Arduino。每個實驗中又各自有部分的修正，其細節描述如下：

（一）比較不同傳輸導線的優劣

1. 不同種類導線的電阻值比較

準備以下導電材料與測量電阻的三用電表：鋁箔、銅箔、Cat6 雙絞網路線、導電銀膠、2B 鉛筆(石墨)。

- (1) 取鋁箔、銅箔、Cat6 雙絞網路線、導電銀膠、2B 鉛筆(石墨)等不同材質的導線 10cm，平貼於桌面上。(圖 3-4)
- (2) 將三用電表連接導線兩端測量電阻。反覆 10 次取其平均值。



圖 3-4 三用電表量電阻

2. 不同長度導線的電阻值比較

準備不同長度的鋁箔、銅箔、Cat6 雙絞網路線進行實驗。

- (1) 取鋁箔、銅箔、Cat6 雙絞網路線等不同材質的導線各 1 m，平貼於桌面上。將三用電表連接導線兩端測量電阻。反覆 10 次，取其平均值。
- (2) 延伸實驗：取銅箔 2m、Cat6 雙絞網路線 2m、Cat6 雙絞網路線 4m，重複步驟 1、2 進行實驗。

（二）第一代訊號傳輸系統：薄膜鍵盤

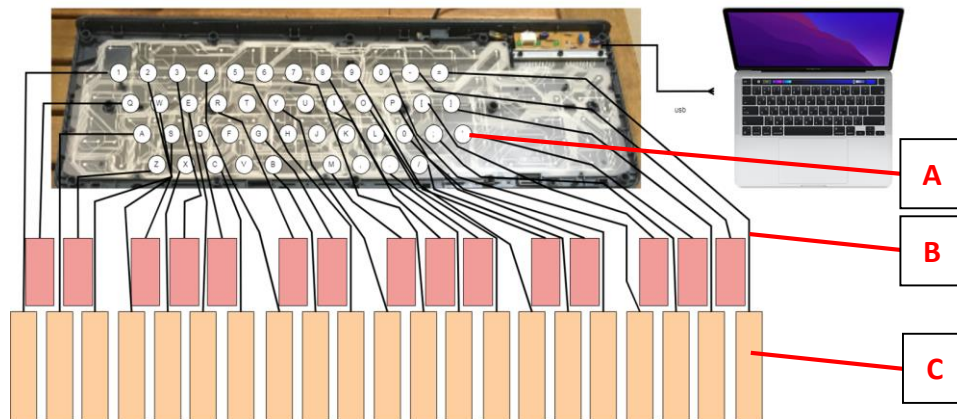


圖 3-5 薄膜鍵盤連結自製紙琴鍵

如上圖 3-5，預計透過薄膜鍵盤將訊號傳至電腦進行音階判別。以薄膜鍵盤為訊號轉介中心，再延伸設計敲擊鍵盤。

1. 硬體架構設計，首先需要改造 USB 薄膜鍵盤，設法將導線連接到薄膜鍵盤上的接觸點，將接觸點延伸到紙製琴鍵開關上。
2. 根據研究 1-1 的實驗結果，在 **A 點** 處我們使用導電銀膠補強接觸點；在 **B 點** 處使用 Cat6 雙絞網路線減少訊號的散失；在 **C 點** 處使用大面積的鋁箔增加接觸面積。
3. 自製紙琴鍵開關，內側上下各貼上鋁鉚，以導線連接延伸連接到薄膜鍵盤，薄膜鍵盤按鍵處導線用銅鉚連接後平貼到電路相對的觸點接點將琴槌槌打的壓力開關，透過薄膜鍵盤，傳鍵盤訊號到電腦程式，電腦程式讀取琴鍵被按下後，即可發出對應著八度音的一個音階。

(三) 第二代訊號傳輸系統：Arduino 訊號傳輸系統

根據第一代實測結果發現薄膜鍵盤的可行性低，因此改以 Arduino 作為訊號收集、辨識的裝置。根據不同做法，進行了五個版本的修正(圖 3-6)。其說明如下：

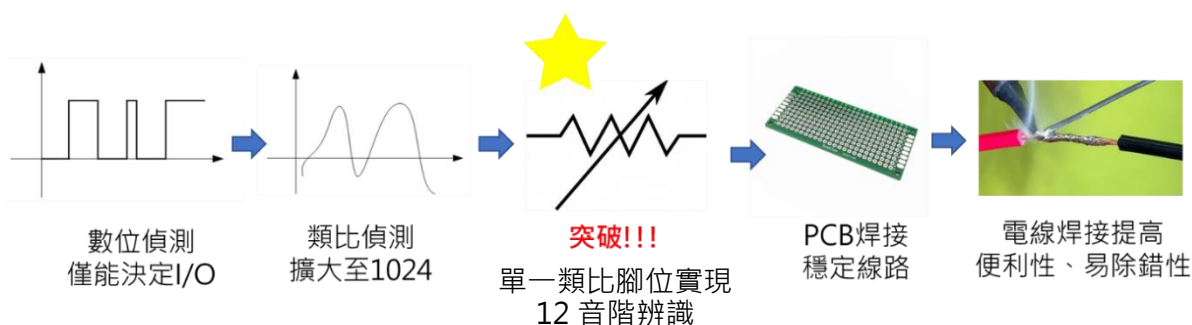
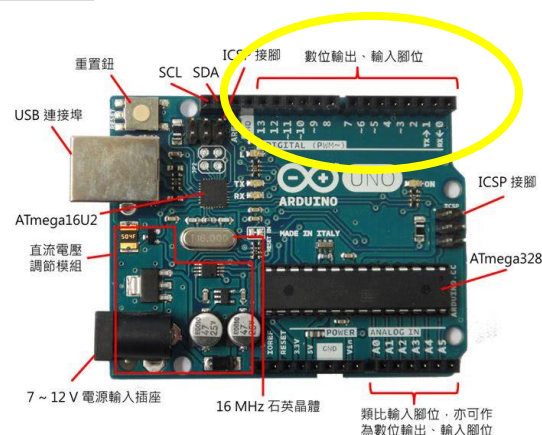


圖 3-6 Arduino 訊號傳輸系統修正歷程簡述

1. Arduino 訊號傳輸系統 2.1 版：數位腳位的訊號傳遞

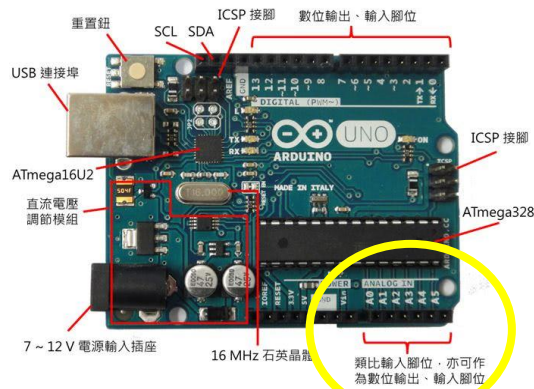
使用 Arduino 中的數位腳位進行訊號辨識，Arduino 中數位腳位分為別 D1 至 D13，扣除其中 D3、D5、D6、D10、D11 腳位為輸出腳位、D13 腳位為 LED 供電使用等，Arduino 可提供 8 個有效數位腳位給電腦進行辨識。

將每個製作的琴鍵對應一個 Arduino 的數位腳位時，按下琴鍵即回傳腳位給電腦，再由電腦轉換成對應音階。



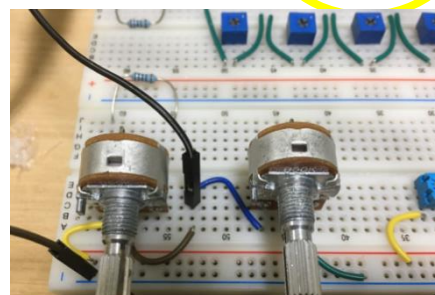
2. Arduino 訊號傳輸系統 2.2 版：類比腳位的訊號傳遞

依據 Arduino 類比訊號可分為 0 至 1023 等 1024 階解析度的 ADC，作為更多音階設計的方向。原理上則是使用串聯電阻(陶瓷電阻 109 歐姆)，使 Arduino 測得不同的電壓值，回傳電壓值轉換為相對應的音階。



3. Arduino 訊號傳輸系統 2.3 版

(1) 根據 2.2 版的測試發現，理論上應能測得不同解析的類比訊號，為了解 Arduino 的單一類比腳位可以偵測到多少不同的電壓值？我們採用兩顆可變電阻進行電壓偵測(圖 3-7)。



(2) 根據實驗結果，我們查找相關科學原理，透過分壓公式(文獻探討 3-3, p. 7)計算結果，調整類比訊號能辨識的電壓數值。並使用三用電表驗證分壓公式所得之電阻值，對應 Arduino 測得之電壓值，最後透過不同的電壓值呼叫相對應的樂音及音高。

圖 3-7 旋轉可變電阻測試可偵測電壓值

4. Arduino 訊號傳輸系統 2.4 版

經過 2.3 版的測試後，雖可實際測得 12 個相對的電壓值，在序列埠上顯示並轉換為對應音階，但因透過麵包板設計的介面極容易因震動、拉扯而產生脫落或接觸不良。我們以 2.3 版本的裝置配備，以 PCB 版焊接的形式改良，達到穩定訊號的目標。

5. Arduino 訊號傳輸系統 2.5 版

在經過市賽比賽過程後，我們發現 PCB 版焊接的效果雖然好，但因希望擴充更多琴鍵，又發現 PCB 焊接具備「立體」線路的特性，不容易除錯。最終考量焊接的便利性、易除錯性，改由電線與電線直接焊接，再加上熱縮管包覆，達到有效設計的目標(圖 3-8)。

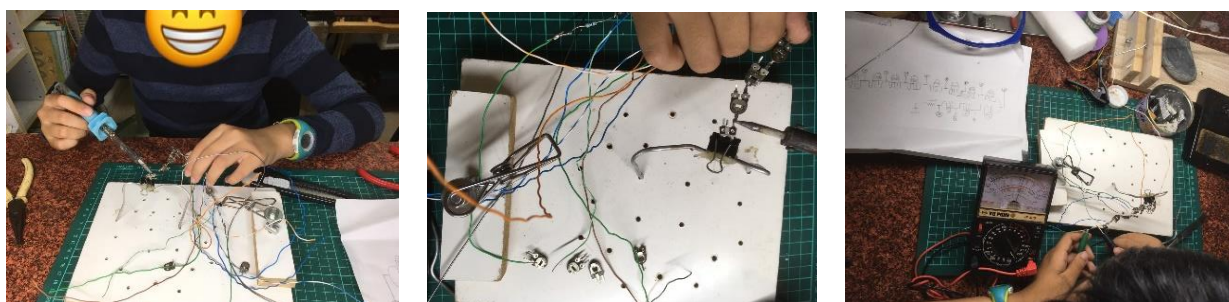


圖 3-8 捨棄 PCB 板，改用電線焊接

研究三：琴鍵結構

琴鍵結構設計延伸自訊號傳輸系統，本研究重視琴鍵的回饋效果（包含耐用度、恢復程度及感應靈敏度），其迴路配置上，沿用研究一結果測得的最佳解：雙絞網路線作為與訊號傳輸系統的連結。其沿革說明如下：

（一）琴鍵設計概念

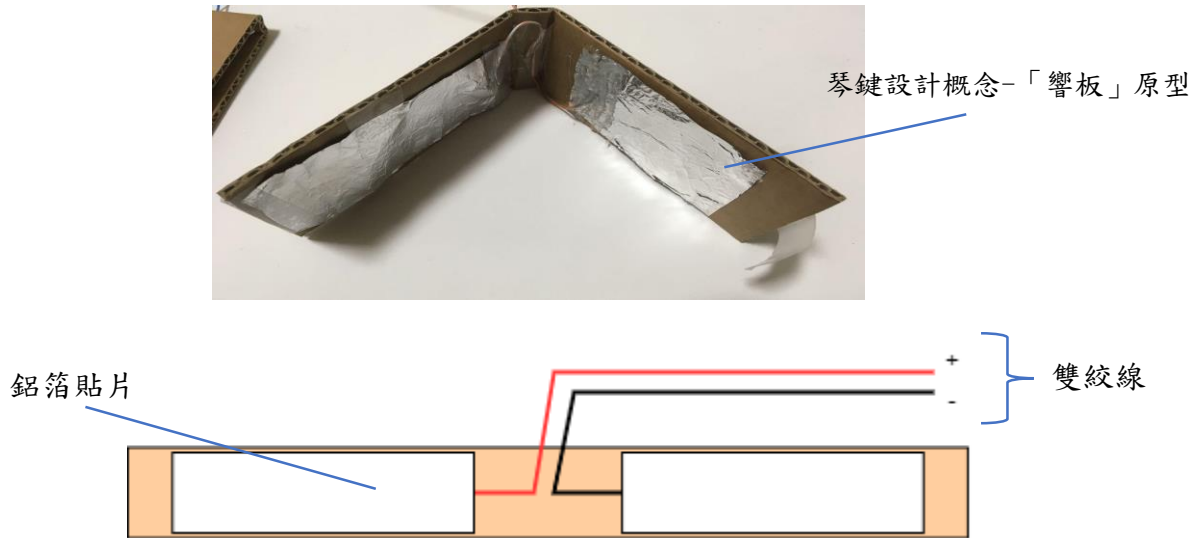


圖 3-9 琴鍵設計概念-鋁箔與雙絞線形成開關

1. 琴鍵的設計概念來自於「響板」，預設的琴鍵具備可恢復性、高彈力、耐受力佳等優點。(圖 3-9)
2. 因琴鍵觸發 Arduino 的關鍵在於通電，為能達到有效通電的目標，我們思考在「響板」的內側各貼上大片的鋁箔（便宜、易取得、大面積的導電性佳），並在上下鋁箔面各接上雙絞網路線達到【接觸後通電】的目標。(圖 3-9)

（二）琴鍵整體結構比較

琴鍵整理結構歷經了幾種不同的形態變化(圖 3-8)，並針對其形態的結構進行耐受恢復力、導電性測試，以求得到具有良率、可長時間使用的效果。

1. 琴鍵結構沿革(圖 3-10)：

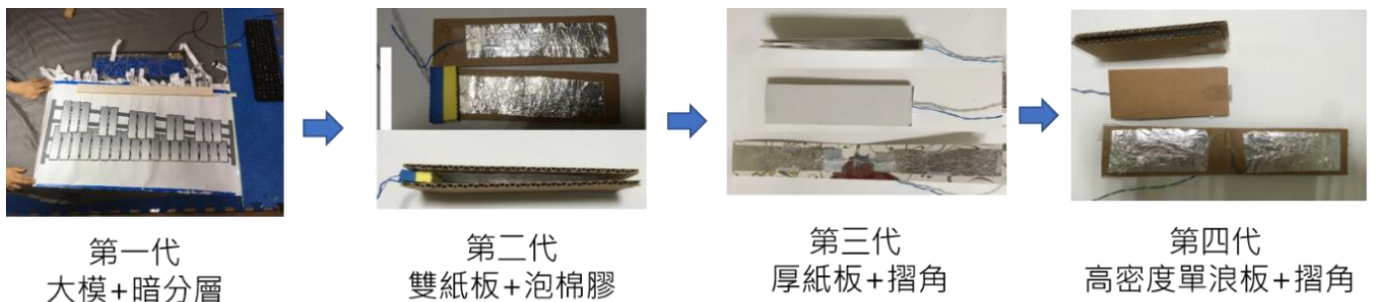


圖 3-10 琴鍵設計沿革

(1) 第一代：大模+暗分層

最初我們以馬林巴琴作為模型，期待設計的琴架能與馬林巴琴一致，因此複印了大張海報平貼於紙板上，並在對應的琴鍵下方做出分層，達到琴面整齊的效果。

(2) 第二代：雙紙板+泡棉膠

在第一代結構的失敗後，我們捨棄馬林巴琴的圖樣，改以個別琴鍵的方式呈現，以兩片分離的紙板+泡棉膠作為彈性回復的軟墊。

(3) 第三代：厚紙板+摺角

在歷經泡棉膠因按壓而塌陷的結果後，我們想到反覆摺紙時，紙張會因為纖維的剛性與彈性無法撫平，因此我們透過厚紙板摺角的方式進行測試。

(4) 第四代：高密度單浪板+摺角

第四代為我們最終選定的琴鍵模式，其製作流程說明如下：

- A. 裁切紙板 4cm × 22cm。
- B. 找紙板中心點，各往外延伸 0.5cm，並用鐵尺壓出痕跡對折。
- C. 裁切鋁箔 3cm × 10cm，並用膠帶黏外側紙板。
- D. 在紙板中心點鑽孔，並把去掉外皮的雙絞線由外插進去。
- E. 把雙絞線分別貼在鋁箔上，用膠帶固定住琴鍵的開合範圍。
- F. 再去掉雙絞線外皮，用三用電表測電阻值(確定數值在 4 歐姆以下)。



A. 裁切紙板



B. 對折



C. 裁切鋁箔



D-1. 鑽洞



D-2. 固定雙絞線



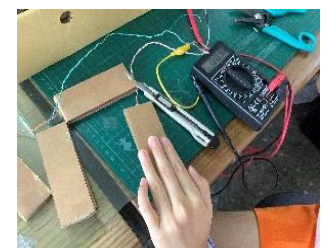
E-1. 貼在鋁箔上



E-2. 膠帶固定琴鍵



F-1. 去掉線的外皮



F-2. 檢測電阻值

2. 琴鍵結構測試：

- (1) **耐受恢復力：**先用游標卡尺量測製作好的厚度，將琴槌舉高至 30 公分(圖 3-11)，之後量測進行按壓 100 次、200 次、300 次的厚度變化，用游標卡尺測量每次敲擊後的厚度變化(圖 3-12)。經過三天後，重複上述實驗。

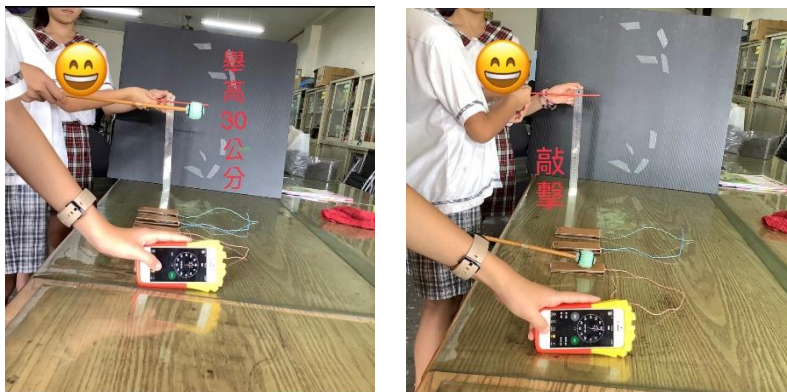


圖 3-11 琴鍵敲擊耐受力測試(高 30cm)



圖 3-12 琴鍵厚度前後測

- (2) **導電性測試：**以每秒鐘 2 下的節奏進行連續按壓，使串聯的 LED 發光，觀察能穩定閃爍幾次，上述實驗重複 5 次取平均值(圖 3-13)。

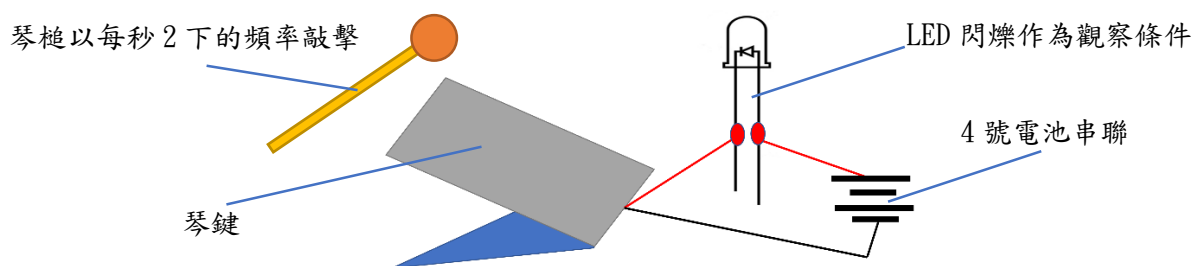


圖 3-13 琴鍵敲擊導電性測試

肆、研究結果

研究一：系統架構

本研究在程式設計的部分，主要分為 Arduino 的電壓偵測與字串轉換程式、HTML5 的 API 接收與結合開源軟體的自製網頁呈現等三部分，其程式碼、測試過程說明如下：

(一) Arduino 的程式設計

1. Arduino 開啟 A0-A5 的類比腳位，透過分壓原理計算不同電阻，使單一類比腳位能偵測 0、0.5、1.0 至 5V 的不同電壓，讓單一類比腳位能偵測 12 種不同電壓值，轉換為字串，達到單一腳位完成 12 音的目標。

2. 使用 Serial port 提取電壓轉換為字串，並提高鮑率至 115200，讓數值偵測的速度可提高到 25ms。

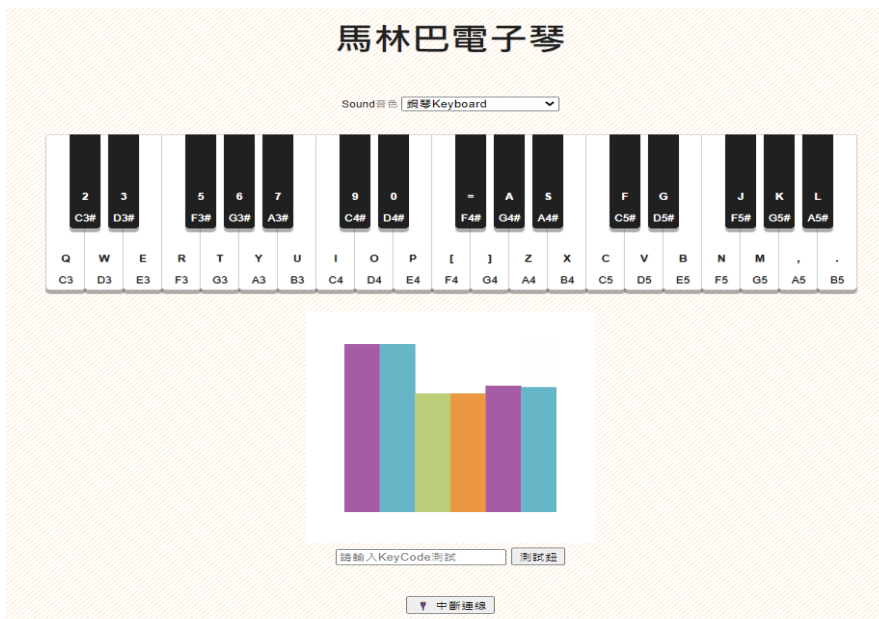
3. 小結：至報告截止前，單一片 Arduino 使用 A0 與 A1 腳位，可同時偵測 60 片琴鍵的電壓值，並準確提取字串，生成 60 個不同的音源撥放。

檔案 編輯 草稿碼 工具 說明

```
sketch_feb26a $
52 }
53 void loop() {
54   iskeyin = false ; // 準備偵測是否有按鍵按下
55   counter = counter + 1 ;
56   singl = "" ;
57   a0val = analogRead(A0) ;
58   a1val = analogRead(A1) ;
59   a2val = analogRead(A2) ;
60   a3val = analogRead(A3) ;
61   a4val = analogRead(A4) ;
62   a5val = analogRead(A5) ;
63
64   singl = "[ " ;
65   singl = singl + cA0 + d + cA1 + d + cA2 + d + cA3 + d + cA4 + d + cA5 + " ]";
66   // 偵測變化
67   if ( a0val != cA0 ) { cA0 = a0val ; iskeyin = true ; }
68   if ( a1val != cA1 ) { cA1 = a1val ; iskeyin = true ; }
69   if ( a2val != cA2 ) { cA2 = a2val ; iskeyin = true ; }
70   if ( a3val != cA3 ) { cA3 = a3val ; iskeyin = true ; }
71   if ( a4val != cA4 ) { cA4 = a4val ; iskeyin = true ; }
72   if ( a5val != cA5 ) { cA5 = a5val ; iskeyin = true ; }
73
74   if ( iskeyin ) {
75     Serial.println(singl);
76     Serial.flush();
77     digitalWrite(13, HIGH);
78   } else {
79     // Serial.println("[0,0,0,0,0,0]");
80     digitalWrite(13, LOW);
81   }
82   delay(25); // 1/50 秒
83 }
```

(二) HTML5 的 API

1. 建構使用者操作介面外觀：以我們學習的馬林巴琴音色為例。



2. 連結 Arduino 鍵盤，接收序列埠傳來的字串資料：

點選鍵盤連線，離線網頁跳出要求與序列埠連線視窗，選取 USB 序列裝置 (Arduino)。連結 Arduino 鍵盤，接收序列埠傳來的字串資料。



```

馬林巴琴HTML5 API - 記事本
檔案 編輯 檢視

連線程式摘錄如下:
const connectButton = document.getElementById('connect-button');
let port;
if ('serial' in navigator) ///
{
  connectButton.addEventListener('click',
    function() {
      if (port) {
        port.close();
        port = undefined;
        connectButton.innerHTML = '鍵盤連線';
        //Connect
      } else {
        // 連結Arduino鍵盤
        getReader();// 接收序列埠傳來的字串資料
      }
    }
  );
  connectButton.disabled = false;
} else {
  //
  // const noSerialSupportNotice = document.getElementById('connect-button');
  //noSerialSupportNotice.innerHTML = "不支援 WEB Serial api"
}

```

3. 判別類比偵測電壓，再轉換成相對應的 keycode 按鍵字符編碼。

(1) Arduino 訊號傳輸系統 2.3 版套用分壓公式(文獻探討 p.7)計算電阻值，形成的 12 琴鍵電壓與音階對照表。(「無」代表未偵測到觸碰)

區分每一琴鍵電壓:0.42

- 琴鍵1 [r2電阻值: 0.00Ω = 電壓0.00v * 1097Ω / (5v -0.00v)] 增量:0.00Ω (43< A0 && A0 <43)
- 琴鍵2 [r2電阻值: 99.73Ω = 電壓0.42v * 1097Ω / (5v -0.42v)] 增量:99.73Ω (43< A0 && A0 <128)
- 琴鍵3 [r2電阻值: 219.40Ω = 電壓0.83v * 1097Ω / (5v -0.83v)] 增量:119.67Ω (128< A0 && A0 <213)
- 琴鍵4 [r2電阻值: 365.67Ω = 電壓1.25v * 1097Ω / (5v -1.25v)] 增量:146.27Ω (213< A0 && A0 <298)
- 琴鍵5 [r2電阻值: 548.50Ω = 電壓1.67v * 1097Ω / (5v -1.67v)] 增量:182.83Ω (298< A0 && A0 <384)
- 琴鍵6 [r2電阻值: 783.57Ω = 電壓2.08v * 1097Ω / (5v -2.08v)] 增量:235.07Ω (384< A0 && A0 <469)
- 琴鍵7 [r2電阻值: 1097.00Ω = 電壓2.50v * 1097Ω / (5v -2.50v)] 增量:313.43Ω (469< A0 && A0 <554)
- 琴鍵8 [r2電阻值: 1535.80Ω = 電壓2.92v * 1097Ω / (5v -2.92v)] 增量:438.80Ω (554< A0 && A0 <639)
- 琴鍵9 [r2電阻值: 2194.00Ω = 電壓3.33v * 1097Ω / (5v -3.33v)] 增量:658.20Ω (639< A0 && A0 <725)
- 琴鍵10 [r2電阻值: 3291.00Ω = 電壓3.75v * 1097Ω / (5v -3.75v)] 增量:1097.00Ω (725< A0 && A0 <810)
- 琴鍵11 [r2電阻值: 5485.00Ω = 電壓4.17v * 1097Ω / (5v -4.17v)] 增量:2194.00Ω (810< A0 && A0 <895)
- 琴鍵12 [r2電阻值: 12067.00Ω = 電壓4.58v * 1097Ω / (5v -4.58v)] 增量:6582.00Ω (895< A0 && A0 <980)

| 琴鍵 0~11 | 電阻 R1 | 電阻 R1 | 輸出電壓 Vout | 代表音高 |
|---------|-------|--------|-----------|------|
| 0 | 1096 | 0 | 0 | C |
| 1 | 1096 | 100 | 0.42 | C# |
| 2 | 1096 | 219 | 0.83 | D |
| 3 | 1096 | 365 | 1.25 | D# |
| 4 | 1096 | 548 | 1.67 | E |
| 5 | 1096 | 783 | 2.08 | F |
| 6 | 1096 | 1096 | 2.50 | F# |
| 7 | 1096 | 1534 | 2.91 | G |
| 8 | 1096 | 2192 | 3.33 | G# |
| 9 | 1096 | 3288 | 3.75 | A |
| 10 | 1096 | 5480 | 4.16 | A# |
| 11 | 1096 | 12056 | 4.58 | B |
| 無 | 1096 | 546904 | 4.99 | 無 |

Arduino 約可辨識 0.4-0.5V 的落差

(2) 網頁持續接收琴鍵傳來的字串資料，然後將字串轉換成陣列變數(A0-A5)，敲擊琴鍵的分壓電阻進行電壓估算。

```
*馬林巴琴HTML5 API - 記事本
檔案 編輯 檢視

// Arduino 偵測(A0-A5)類比訊號將琴鍵的分壓輸出電壓值 合成為字串 以序列傳輸輸出.
singl = "[" + a0 + "," + a1 + "," + a2 + "," + a3 + "," + a4 + "," + a5 + "]";

// 網頁接收到琴鍵傳來的字串latestValue 後 轉成 陣列變數
keyList = JSON.parse(latestValue);
DetectKeyCode(keyList) // 判別是那個琴鍵被按下
```

(3) 當程式收到陣列變數，以輸出電壓數值判別是那一一個琴鍵被按下，轉換成相對應的keycode 按鍵字符編碼。

```
*馬林巴琴HTML5 API - 記事本
檔案 編輯 檢視

function DetectKeyCode(keys) { // 偵測是那個 琴鍵按下
  var keycode = 0;
  var a0 = keys[0]; // 接腳A0類比訊號

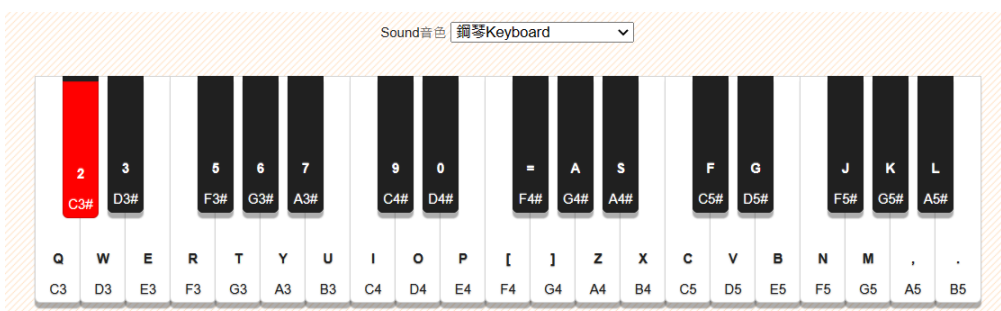
  if (a0 <= 43) keycode = 81; // 'Q'
  else if (43 < a0 && a0 <= 128) keycode = 50; // '2'
  else if (128 < a0 && a0 <= 213) keycode = 87; // 'W'
  else if (213 < a0 && a0 <= 298) keycode = 51; // '3'
  else if (298 < a0 && a0 <= 384) keycode = 69; // 'E'
  else if (384 < a0 && a0 <= 469) keycode = 82; // 'R'
  else if (469 < a0 && a0 <= 554) keycode = 53; // '5'
  else if (554 < a0 && a0 <= 639) keycode = 84; // 'T'
  else if (639 < a0 && a0 <= 725) keycode = 54; // '6'
  else if (725 < a0 && a0 <= 810) keycode = 89; // 'Y'
  else if (810 < a0 && a0 <= 895) keycode = 55; // '7'
  else if (895 < a0 && a0 <= 980) keycode = 85; // 'U'

  var a1 = keys[1]; // 接腳A1類比訊號

  if (a1 <= 43) keycode = 73; // 'I'
  else if (43 < a1 && a1 <= 128) keycode = 57; // '9'
  else if (128 < a1 && a1 <= 213) keycode = 79; // 'O'
  else if (213 < a1 && a1 <= 298) keycode = 48; // '0'
  else if (298 < a1 && a1 <= 384) keycode = 80; // 'P'
  else if (384 < a1 && a1 <= 469) keycode = 219; // '['
  else if (469 < a1 && a1 <= 554) keycode = 187; // '='
  else if (554 < a1 && a1 <= 639) keycode = 221; // ']'
  else if (639 < a1 && a1 <= 725) keycode = 65; // 'A'
  else if (725 < a1 && a1 <= 810) keycode = 98; // 'Z'
  else if (810 < a1 && a1 <= 895) keycode = 83; // 'S'
  else if (895 < a1 && a1 <= 980) keycode = 88; // 'X'

  reDraw(keys); // 顯示
  return keycode;
}
```

4. 連結開源音源檔、視覺化讀取資料以播放聲音。(完整程式碼如附件)
(第二代音檔也是直接讀取資料，僅需要轉換音源檔案的讀取位置)



研究二：訊號傳輸系統測試

(一)比較不同傳輸導線的差異

表 4-1 不同導線的電阻值測試(電線皆長 10cm)

| 導線材質 測試 | 鋁箔 1cm 寬 | 銅箔 6mm 寬 | 導電銀膠 | 2B 鉛筆(石墨) 1mm 寬 | Cat6 雙絞網路線 |
|------------|-------------|-------------|------|--------------------|------------|
| Test 1 | 0.04 | 0.03 | 8.25 | 284000 | 0.03 |
| Test 2 | 0.05 | 0.04 | 8.31 | 285000 | 0.03 |
| Test 3 | 0.05 | 0.04 | 8.52 | 286000 | 0.03 |
| Test 4 | 0.04 | 0.03 | 8.40 | 286000 | 0.04 |
| Test 5 | 0.05 | 0.04 | 8.50 | 285000 | 0.03 |
| Test 6 | 0.04 | 0.03 | 8.45 | 285000 | 0.03 |
| Test 7 | 0.05 | 0.04 | 8.43 | 286000 | 0.03 |
| Test 8 | 0.05 | 0.03 | 8.47 | 285000 | 0.03 |
| Test 9 | 0.04 | 0.04 | 8.42 | 284000 | 0.03 |
| Test 10 | 0.05 | 0.03 | 8.62 | 285000 | 0.03 |
| 平均值 | 0.05 | 0.04 | 8.44 | 285100 | 0.03 |

表 4-2 不同長度的電線電阻值測試

| 導線材質 測試 | 鋁箔 1m | 銅箔 1m | 銅箔 2m | 雙絞網路線 1m | 雙絞網路線 2m | 雙絞網路線 4m |
|------------|----------|----------|----------|-------------|-------------|-------------|
| Test 1 | 1542 | 0.31 | 0.51 | 0.31 | 0.41 | 0.56 |
| Test 2 | 1574 | 0.35 | 0.56 | 0.32 | 0.43 | 0.57 |
| Test 3 | 1562 | 0.32 | 0.57 | 0.31 | 0.42 | 0.56 |
| Test 4 | 1489 | 0.36 | 0.56 | 0.30 | 0.41 | 0.58 |
| Test 5 | 1542 | 0.31 | 0.57 | 0.32 | 0.41 | 0.57 |
| Test 6 | 1593 | 0.30 | 0.54 | 0.31 | 0.42 | 0.56 |
| Test 7 | 1574 | 0.35 | 0.59 | 0.30 | 0.41 | 0.57 |
| Test 8 | 1562 | 0.32 | 0.55 | 0.30 | 0.42 | 0.58 |
| Test 9 | 1546 | 0.34 | 0.54 | 0.31 | 0.41 | 0.56 |
| Test 10 | 1562 | 0.31 | 0.55 | 0.32 | 0.42 | 0.57 |
| 平均值 | 1555 | 0.33 | 0.55 | 0.31 | 0.42 | 0.57 |

採用

比較

(二) 比較第一代訊號傳輸系統：薄膜鍵盤

1. 薄膜鍵盤導線連接時必需防止導線交錯，處理好絕緣才不會造成短路，導致鍵盤電路損壞。(圖 4-1)
2. 佈線需要仔細安排將各個音階依序由上到下由左到右排列好，經過多次反覆嘗試後將八度音階對照鍵盤排列如下表所示：



圖 4-1 薄膜鍵盤連結自製紙琴鍵

| | | | | | | | | | | | | |
|------|---|----|---|----|---|---|----|---|----|---|----|---|
| 按鍵符號 | A | X | Z | Q | S | 1 | W | 2 | D | C | E | 3 |
| 音階 | C | C# | D | D# | E | F | F# | G | G# | A | A# | B |

(三) 比較第二代訊號傳輸系統：Arduino 訊號傳輸系統

表 4-3 Arduino 訊號傳輸系統 2.1 至 2.4 版本比較表

| 版本 | 2.1 版 | 2.2 版 | 2.3 版 | 2.4 版 |
|------|---|---|--|---|
| 器材 | 麵包板、杜邦線、陶瓷電阻 | | 麵包板、杜邦線、可變電阻、陶瓷電阻 | PCB 板、可變電阻、陶瓷電阻 |
| 原理 | 數位訊號偵測通電琴鍵的 0/I | 相同電阻串聯，以類比訊號辨識通電琴鍵的解析度 | 分壓原理計算電阻值，以類比訊號偵測通電琴鍵的電壓值，提高序列鮑率至 115200 | |
| 圖例 |  |  |  |  |
| 音階 | 至多 8 階 | 理論值約 20 階 | 理論值約 60 階 (實測至 24 階) | 理論值約 60 階 (實測至 24 階) |
| 觀察描述 | 單片 Arduino 晶片至多完成 8 個音階，不符合需求。 | 類比腳位偵測陶瓷電阻的串聯，單一腳位至多偵測 4-5 個音階。單片 Arduino 晶片約可完成 20 個音階。(A0-A5) | 類比腳位偵測可變電阻，單一腳位即可偵測 12 音階。單片 Arduino 晶片約可完成 60 個音階。但麵包板與杜邦線會因震動脫落。 | 優點如同 2.3 版本，且避免震動脫落 |

Arduino 訊號傳輸系統 2.5 版本

將 Arduino 訊號傳輸系統 2.4 版本自 PCB 板，改為電線焊接。提高焊接的便利性、易除錯性，達到穩定傳輸的目標。

研究三：琴鍵結構

針對琴鍵結構進行耐受恢復力、導電性測試，以求得到具有良率、可長時間使用的效果。其實驗描述如下：

(一) 琴鍵結構耐受恢復力測試

表 4-4 製作完成第一天與第三天的敲擊測試結果

| 時間 | 厚度變化 | 第一代 | 第二代 | 第三代 | 第四代 |
|------|-----------------|------|------|------|------|
| Day1 | 原厚度(mm) | 12.5 | 11.8 | 11.4 | 10.1 |
| | 按壓 100 次後厚度(mm) | 12.4 | 10.6 | 11.4 | 10.0 |
| | 按壓 200 次後厚度(mm) | 12.5 | 10.5 | 11.3 | 10.0 |
| | 按壓 300 次後厚度(mm) | 12.4 | 10.5 | 11.3 | 10.0 |
| Day3 | 原厚度(mm) | 12.1 | 10.5 | 11.3 | 10.0 |
| | 按壓 100 次後厚度(mm) | 12.1 | 10.4 | 11.2 | 10.0 |
| | 按壓 200 次後厚度(mm) | 12.0 | 10.5 | 11.2 | 10.0 |
| | 按壓 300 次後厚度(mm) | 12.0 | 10.4 | 11.2 | 10.0 |

不變形

(二) 琴鍵連續按壓可連續偵測通電的次數

表 4-5 連續按壓時，能連續明確通電的次數結果

| 可連續按壓次數 | 第一代 | 第二代 | 第三代 | 第四代 |
|-----------|-----|-----|-----|-------|
| 連續按壓 1(次) | 3 | 15 | 61 | 143 |
| 連續按壓 2(次) | 2 | 12 | 57 | 157 |
| 連續按壓 3(次) | 4 | 7 | 52 | 139 |
| 連續按壓 4(次) | 3 | 8 | 50 | 148 |
| 連續按壓 5(次) | 2 | 6 | 45 | 156 |
| 平均值 | 2.8 | 9.6 | 53 | 148.6 |

最穩定

伍、討論

研究一：系統架構

(一) Arduino 的程式設計

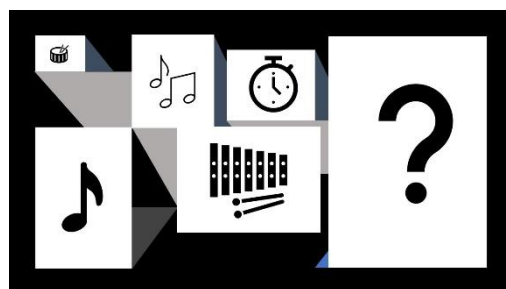
1. 根據原理，Arduino 的類比腳位有 A0-A5 等六個腳位，並依照研究二實驗結果可知，每一腳位能夠提供 12 個不同的電壓值，理論上能產生共 72 組不同的電壓值。
2. 而我們本次研究使用 Arduino 僅作為偵測電壓與回傳電壓字串使用，提高序列埠的鮑率至 115200，讓數值偵測的速度可提高到 25ms 後，透過 keycode 按鍵字符編碼將電壓值轉換成字串，使偵測電壓、對應音階時，能快速(25ms)識別對應的音源。
且每一個類比腳位的獨立偵測邏輯，電腦可以同時判讀最多 6 個檔案，同時撥放。

(二) HTML5 的 API

1. 首先建構使用者操作介面外觀，以 Keith William Horwood(2013) 分享在 GitHub 上的開源音檔模板為基礎，建立自己的 API。
2. 在建置 HTML 的 API 時，也想過呼叫的音源檔案雖然屬於開源軟體，但此音源皆為電腦混成的樂音，雖然音高準確，卻少了真實情境的表現。
3. 在市賽過後，我們思考製作的初衷，【期待自製樂器能表現在校練習的樂音】，因此決定進行另一個大突破，錄製原音的呈現。

(三) 錄製真實的音源檔

1. 錄製真實的音源檔案放在電腦中，提供電腦讀取音源，概念上是相當容易的。僅需要修正呼叫檔案的目的地而已，但其考量的過程卻相當需要討論。
2. 我們思考的問題如下：
 - (1) 每個音檔要錄製多少時間呢？
 - (2) 4 分音符/8 分音符是否都要分別錄製？
 - (3) 要錄製那些樂音呢？
 - (4) 是否每種錄音檔案都可以使用呢？
3. 根據上述問題，我們解決方式如下：
 - (1) 每個音檔的錄音時間皆是 2 秒鐘，使樂音能完整表現。
 - (2) 不同節拍的樂音使用程式來解決。我們透過程式邏輯撰寫「**當重新偵測到訊號時，就直接撥放新的樂音**」的概念，使「**敲擊速度**」成為節奏的來源。
 - (3) 為了達到電音 BAND 的目標，我們錄製了鋼琴、木琴、鐵琴、手搖鐘、直笛，使樂音的種類多樣化又具有節拍性。
 - (4) 我們在程式呼叫過程中發現，需使用 mp3 檔案呼叫。透過手機錄製的音源檔案為 M4A 檔，需藉由網路軟體轉檔才能撥放。



研究二：訊號傳輸系統測試

(一) 不同傳輸導線的優劣討論

1. 相同長度不同導線的電阻值測試

- (1) 根據實驗可知，1mm 寬的 2B 鉛筆(石墨)所塗出來電路的電阻值過大，無法作為實際運用。而鋁箔、銅箔及 Cat6 雙絞網路線之電阻值較低，可進行不同長度的電阻比較。
- (2) 思考自 36 鍵鋼琴為例，實際大小尺寸約為長 2.4 公尺、寬 1.3 公尺。因此訊號線的尺寸也必須至少延長線到琴長一半以上的長度 1.5~2 公尺長。推論要測試不同導線能否在 2 公尺長時依然能夠觸發鍵盤開關，測量比較其電阻歐姆值大小。

2. 不同長度的電線電阻值測試

- (1) 用寬度約 1cm 鋁箔去延伸到 1 公尺以上時，鋁箔測得電阻值的話是約為 15k 歐姆。根據文獻，鋁箔能夠大面積與短距離使用，因此鋁箔僅測試 1m 的長度。
- (2) 自銅箔與雙絞網路線的數據而言，兩者差異不大。但自 Cat6 產品使用說明書提到，雙絞網路線還可以避免雜訊的干擾，避免一些線圈磁場的一些效應，因此我們依照實驗數據建議延長 1 公尺以上改用雙絞網路線。實驗過程中，我們也試著使用不同的導體去延伸出我們所需要的鍵鍵盤的長度，雙絞線實驗甚至可以延長到達 10 公尺以上距離依然可以觸動開關。顯示雙絞網路線最符合需求。

3. 小結：

根據上述實驗結果得知，在琴鍵與訊號傳輸系統的連結上，使用最穩定且不受雜訊干擾的 Cat6 雙絞網路線；琴鍵接觸面以市面最常見、價格最便宜的大片鋁箔方便使用。

(二) 第一代訊號傳輸系統：薄膜鍵盤

使用薄膜鍵盤進行訊號傳輸時，會遇上幾個問題：

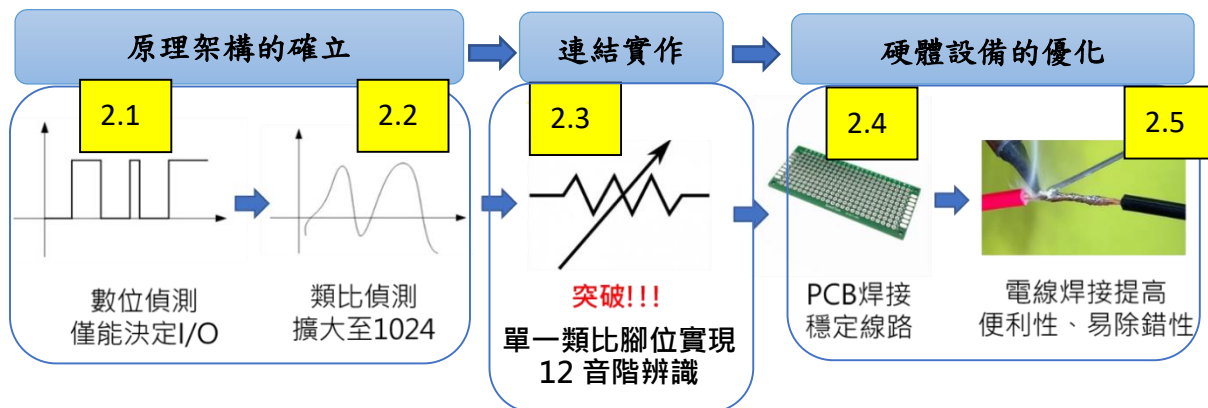
1. 透過鍵盤延長電線進行訊號傳遞，看似具擴充性，但就訊號傳遞的效能而言，無異是畫蛇添足。
2. 薄膜鍵盤為求輕巧，鍵盤的通電接點皆是細小電線，透過 Cat6 網路線或其他電線進行安裝，一直無法有效克服穩定傳輸的困境。在很早期的測試中確定棄用。(圖 5-1)



圖 5-1 薄膜鍵盤連接琴鍵測試

(三) 第二代訊號傳輸系統：Arduino 訊號傳輸系統

在使用 Arduino 進行訊號傳輸測試，經過 2.1 至 2.5 的版本，其過程主要分為：原理架構的確立、連結實作（突破）、設備優化等三個時期（比較圖詳見 p.19）：

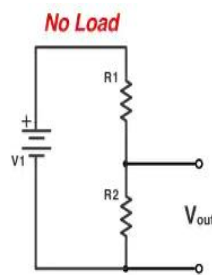


1. 原理架構的確立

使用 Arduino 便會思考訊號連接的腳位，因此開始分析數位腳位(0/I)和類比腳位(0-1023)的差異。透過此分析結果，觀察到**多點感測**必須透過類比訊號才能完成，因此很快確立了 A0-A5，是我們需要使用的腳位。

2. 連結實作

(1) 確立了類比腳位的使用，接著需要考量理論與實務的差異。雖然單一類比腳位「號稱」能感測 1024 種變化，但具體結果仍需要進行實驗。因此我們透過在兩顆可變電阻(R1/R2)通過 5V 的電壓，在 A0 腳位上進行實際電壓的偵測，在過程中旋轉兩顆可變電阻(R1/R2)，觀察偵測的電壓值變化。



$$V_{out} = V_1 \frac{R_2}{R_1 + R_2} = \frac{V_1 R_2}{R_1 + R_2}$$

圖 5-2 分壓公式

(2) 在實測的過程中，我們可以發現電阻的些許變化，不容易在偵測的電壓上顯示，實際每次電壓的變化值約為 0.5V 左右。因此，我們即可透過分壓公式計算每 0.5V 的電壓變化(圖 5-2)，約需要多少的電阻值。

3. 硬體設備的優化

經過了 Arduino 訊號傳輸系統 2.3 版的突破後，便需要進行硬體設備的優化，使我們的樂器能穩定運行。首先透過 PCB 板的焊接，連接可變電阻及電線。也透過此版本參與了市賽，獲取佳績。

偶後發現，PCB 板雖能穩定電路，但因 PCB 板本身無法一眼看到電路，導致在訊號不良時，維修成為相當大的問題。所以再進一步修改為「電線焊接」的版本(圖 5-3)。達到穩定線材、維修方便的目標。

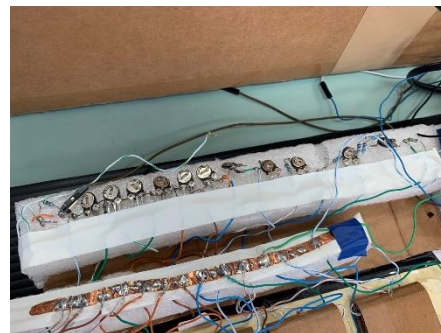
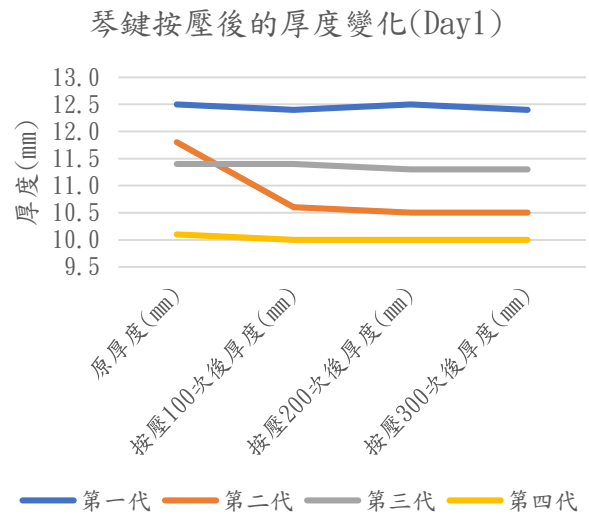


圖 5-3 電線焊接的成品可一目了然

研究三：琴鍵結構

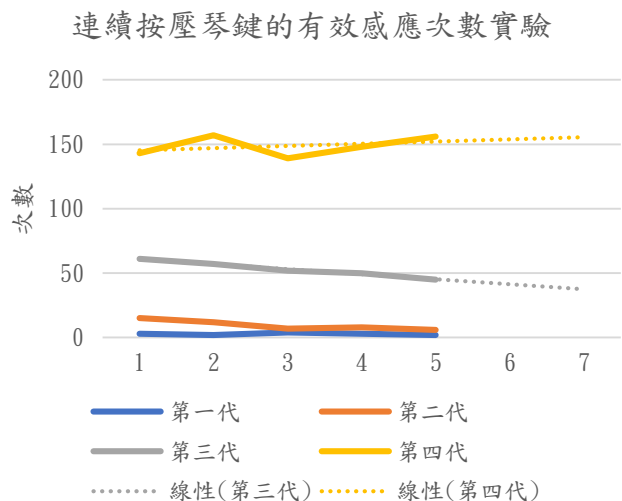
(一) 琴鍵結構耐受恢復力測試

1. 自耐受力測試發現，第二代的泡棉膠（右圖，橘色線）所製成的琴鍵，因泡棉膠易受力變形的因素，在經過 100 次按壓時急遽的「變扁」。但變形有限，至 200 下以後，甚至三天後後都無劇烈變化。
2. 不論是第一代、第三代及第四代在 Day1、Day3 實驗中，琴鍵多次按壓的結果上，表現皆是變化不大。因此進一步進行連續按壓的通電測試。



(二) 琴鍵連續按壓可連續偵測通電的次數

1. 本實驗主要模擬樂器演奏時的連續敲擊是否能穩定產生琴音，根據結果發現，第一代及第二代琴鍵的效果極差，都在短短三至五秒內進會產生感應上的錯誤。而第三代約可連續按壓 30 秒(60 次)；第四代更可達到 1 分鐘(120 下)以上。
2. 觀察第一代琴鍵失誤主要在於一大面的琴鍵模型，容易在敲擊時感應到其他琴鍵，而無法有效、細膩的偵測結果。
3. 第二代琴鍵主要受到泡棉膠影響，在前一個實驗中即發現泡棉受力會急遽變形，進而影響通電感應的效果。
4. 而第三代與第四代的差異在於紙板的剛性程度。第三代使用厚紙板，而第四代進一步使用高密度單浪板（購買 3C 產品時，外包裝的較硬紙箱即是），能達到比起第三代長時間使用後的衰減（見上圖「(線性趨勢) 第三代」，灰色虛線），第四代高密度單浪板的衰減更微小（見上圖「(線性趨勢) 第四代」，黃色虛線）。



陸、結論

- 一、本研究真實解決了我們在家練習馬林巴琴時無樂音的現況。使得我們的自製樂器中【多人合奏、多重音色、原音重現】的期待得以完整的表現。並更進一步設計小型電子演奏團，重現在校練習樂團的複合樂音表現。
- 二、本研究的歷程可以分為三大項：
- (一)以回收材料自製鍵盤並數據分析鍵盤優勢，且將「彈奏」轉換為「敲擊」。
 - (二)捨棄原有的音源撥放方式，將「軟體合成音樂」轉換為「真實錄製樂音」。
 - (三)透過 Arduino 突破撥放樂音的數量，符合多樂器、同時演奏的樂團目標。
- 三、本次使用的網頁設計軟體、音源檔案皆使用開源資料。為響應網路上開源資料的分享，也在本次報告中公開發碼、音源檔案，並上傳至 GitHub，有助於未來有興趣的研究者取用測試。
- 四、最後針對本作品與市售樂器的分析比較如下：

| 產品 | 電鋼琴 | 馬林巴琴 | 鐵琴 | STOMP-自製電音 BAND |
|------|-------------------|--------------|----------------|--------------------|
| 價格 | \$5000-\$200000 ↑ | \$1200000 ↑ | \$ 450000 ↑ | 成本\$5,000 以內 |
| 可攜帶性 | 固定大小 | 固定大小 | 固定大小 | 依需求攜帶音域及大小 |
| 音域 | 3-5 個音域 | 5 個音域 | 3 個音域 | 1-6 個音域 (實測 5 個音域) |
| 音色 | 可產生多元化混成音 | 單一 | 單一 | 可錄製多元化的真實樂器音 |
| 音量 | 可精準調整大小 | 須依經驗調整力道控制音量 | 須依經驗判斷調整力道控制音量 | 可精準調整大小 |
| 使用人數 | 一次一人 | 一次一人 | 一次一人 | 設定不同樂器音可同時多人演奏 |

柒、參考資料及其他

民國 72 年高中科展-新式電腦音樂合成器 <https://twsf.ntsec.gov.tw/activity/race-1/23/pdf/23h/196.pdf>

民國 106 年國小科展-真"水"鋼琴 <https://www.ntsec.edu.tw/Science-Content.aspx?cat=&a=0&fld=&key=&isd=1&icop=10&p=226&sid=13877>

民國 106 年高中科展-「MIDI 樂器 (電子琴)」 <https://www.ntsec.edu.tw/Science-Content.aspx?a=6821&fld=&key=&isd=1&icop=10&p=1&sid=13767>

民國 109 年國小科展-「微」鋼琴三部曲 <https://www.ntsec.edu.tw/Science-Content.aspx?cat=&a=0&fld=&key=&isd=1&icop=10&p=59&sid=16656>

壓電開關介紹 <https://www.rohm.com.tw/electronics-basics/piezo/what1>

天花板隨記- Arduino 筆記(85)：電壓感測器(Voltage Sensor) <https://atceiling.blogspot.com/2020/10/arduino85voltage-sensor.html>

Arduino 連線原始碼 <https://github.com/svendahlstrand/web-serial-api>

開源音源檔 <https://keithwhor.com/music/>
<https://keithwhor.github.io/audiosynth>

視覺化 <https://css-tricks.com/making-an-audio-waveform-visualizer-with-vanilla-javascript/>

附件：Arduino 程式碼

```
int a0val = 0 ; // 3 組 類比訊號接收值
int alval = 0 ;
int a2val = 0 ;
int a3val = 0 ; // 3 組 類比訊號接收值
int a4val = 0 ;
int a5val = 0 ;
int counter = 0 ;
String d = "," ;
String singl = "" ;
int cA0 = 0 ;
int cA1 = 0 ;
int cA2 = 0 ;
int cA3 = 0 ;
int cA4 = 0 ;
int cA5 = 0 ;
boolean iskeyin = false ;
// 如果要讓 Arduino 接收來自外部電腦的資料，我
// 們可以用 Serial.available() 跟
// Serial.read() 這兩個函式：

/*
 * // 接收
 * if(Serial.available()) {
 *   int num = Serial.parseInt();
 *   Serial.print("The temperature is: ");
 *   Serial.print(num * 9 / 5 + 32);
 *   Serial.println(" °F");
 * }
 */
void setup() {
  // 接收 3 組 類比訊號
  pinMode(13, OUTPUT);
  pinMode(A0, INPUT);
  pinMode(A1, INPUT);
  pinMode(A2, INPUT);
  pinMode(A3, INPUT);
  pinMode(A4, INPUT);
  pinMode(A5, INPUT);

  // 接收 3 組 類比訊號
  // Serial.begin(9600); //19200
  Serial.begin(115200); //19200
  //115200
  //115200
  a0val = 0 ; // 3 組 類比訊號接收值
  alval = 0 ;
  a2val = 0 ;
  a3val = 0 ;
  a4val = 0 ;
  a5val = 0 ;
  counter = 0 ;
  d = "," ;
  singl = "" ;

  cA0 = analogRead(A0) ; // 量化取值減少精度
  1024 / 4 = 256
  cA1 = analogRead(A1) ; //
  cA2 = analogRead(A2) ; //
  cA3 = analogRead(A3) ; //
  cA4 = analogRead(A4) ; //
  cA5 = analogRead(A5) ; //
}
void loop() {
  iskeyin = false ; // 準備偵測是否有按鍵按下
  counter = counter + 1 ;
  singl = "" ;
  a0val = analogRead(A0) ;
  alval = analogRead(A1) ;
  a2val = analogRead(A2) ;
  a3val = analogRead(A3) ;
  a4val = analogRead(A4) ;
  a5val = analogRead(A5) ;

  singl = "[ " ;
  singl = singl + cA0 + d + cA1 + d + cA2
  + d + cA3 + d + cA4 + d + cA5 + " ]" ;
  // 偵測變化
  if ( a0val != cA0 ) { cA0 = a0val ;
  iskeyin = true ; }
  if ( alval != cA1 ) { cA1 = alval ;
  iskeyin = true ; }
  if ( a2val != cA2 ) { cA2 = a2val ;
  iskeyin = true ; }

  if ( a3val != cA3 ) { cA3 = a3val ;
  iskeyin = true ; }
  if ( a4val != cA4 ) { cA4 = a4val ;
  iskeyin = true ; }
  if ( a5val != cA5 ) { cA5 = a5val ;
  iskeyin = true ; }

  if (iskeyin){
    Serial.println(singl);
    Serial.flush();
    digitalWrite(13, HIGH);
  }else {
    // Serial.println("[0,0,0,0,0,0]");
    digitalWrite(13, LOW);
  }
  delay(25); // 1/50 秒
}
}
```

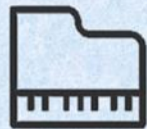
【評語】 082821

1. 利用 Arduino 設計出多音階敲擊樂器，第二代使用自行錄製直笛，鋼琴，木琴音源。
2. 雖雅使用紙板最為敲擊琴鍵成本低廉，卻能發出真實樂器聲音且有一定耐受力。
3. 比過去的模擬樂器有較多數目的琴鍵 12-72 音階。
4. 未來結合 A I 技術可以延伸至虛擬樂團。
5. 報告呈現建議可以再清楚些。

作品海報



STOMP-



自製電音BAND



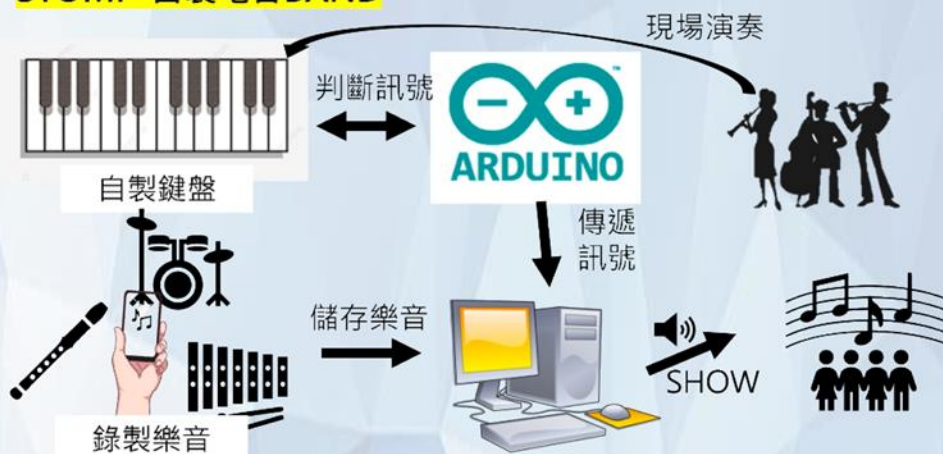
摘要

本研究目標旨在完成**多人合奏、原音重現**的自製樂器，透過 Arduino 作為傳遞訊號的中介，以電腦對應輸出不同樂器的實際音色。在系統架構上，我們利用 Arduino 的類比訊號，讀取電壓值轉換成字串對應音階，實際錄製我們想要的樂音建立音源資料庫，以視覺化圖形輔助音階表現，隨時對設備進行調整，完成播放系統的模組。實驗過程中，我們測試了 2 個世代、5 個版本的訊號傳輸系統，最後以分壓原理為知識架構，焊接可變電阻，形成有效、穩定的訊號傳輸。其次，以生活上的紙箱素材、Cat6 雙絞銅線、鋁箔紙自製耐受恢復力高、導電性高的琴鍵結構，製作出虛實整合的電音樂團(BAND)。

研究動機

我們就讀的是一所音樂型發展學校。在校園中，打擊樂演奏是生活的一部分，我們參加打擊樂團主要練習馬林巴琴。馬林巴琴是一種昂貴的大型敲擊樂器，平時除在校可使用馬林巴琴練習外，在家的練習僅能使用自製的練習琴，而練習琴只能產生節拍噪音而無樂音，在練習時常想，如果在家練習也能如同在校一樣演奏好聽的音樂就好了。此時，我們想起打擊樂啟蒙樂團：STOMP (破銅爛鐵樂隊)。STOMP 運用日常用品如掃把、垃圾桶蓋、打火機...等等破銅爛富有不同音頻、豐富音樂性的節奏。因此，我們開始想動手打造減少噪音、加入樂音、多人演奏，屬於我們的自製敲擊琴。透過我們的想像與過去科展的研究，決定以回收材料自製多個的鍵盤樂器，在校期間錄製不同樂器的樂音，使用 Arduino 分別控制不同的電子訊號，由電腦同時判定不同訊號，發送至電腦並同時播放，達到多人同時演奏同一大組樂器，卻能產生不同樂音自製電音 BAND。

STOMP-自製電音BAND



研究目的與問題

(一) 自製多點電位差打擊樂器的系統架構

1. Arduino 在打擊樂器的偵測上扮演何種角色？
2. 離線 HTML5 API 如何掛載多音音色的開源音檔？
3. 如何實際錄製不同樂器的樂音並有效播放呢？
4. 如何加入視覺化圖形幫助實測呢？

(二) 測試有效的訊號傳輸系統

1. 不同傳輸導線電阻值的大小為何？
2. 第一代薄膜鍵盤的訊號傳輸效果為何？
3. 第二代 Arduino 訊號傳輸系統的修正過程為何？

(三) 設計符合可攜式、可耐用的琴鍵結構

1. 不同材質、結構設計的琴鍵觀察比較為何？
2. 延續問題 3-1，其恢復力(抗變形)的能力為何？
3. 延續問題 3-1，其耐受力(有效導電)的能力為何？

(四) 實際展現我們設計的敲擊樂器



文獻探討

(一) 歷屆科展作品分析

| 作品名稱 | 新式電腦音樂合成器 | 真"水"鋼琴 | MIDI樂器(電子琴) | 微鋼琴三部曲 | STOMP-自製電音BAND |
|------|-----------|--------------|--------------|---------------|----------------|
| 屆次 | 民國72年 | 民國106年 | 民國106年 | 民國109年 | 民國111年 |
| 組別 | 高中 | 國小 | 高中 | 國小 | 國小 |
| 輸出方式 | 電腦(MIDI) | 電腦(Scratch) | IC晶片+喇叭 | 電腦(micro:bit) | 電腦(開源網頁) |
| 音檔來源 | MIDI音檔 | Scratch內建音源 | MIDI音檔 | Micro:bit內建音源 | 實際收錄真實樂器音 |
| 演奏方式 | 電腦鍵盤 | 自製按壓鍵盤 | 鋼琴鍵盤 | 自製按壓鍵盤 | 自製敲擊鍵盤 |
| 播放音階 | 無說明 | 無說明 | 無說明 | 約8-10音階 | 12-72音階 |
| 鍵盤製作 | 無說明 | 研究不同水溶液的導電效果 | 設計不同的矩陣鍵盤及函式 | 描述比較鍵盤導電性 | 數據比較鍵盤導電性與恢復力 |

我們以「微鋼琴三部曲」作品為基礎，以回收材料自製鍵盤並數據分析鍵盤優勢，並將「彈奏」轉換為「敲擊」；捨棄原有的音源播放方式，將「軟體合成音樂」轉換為「真實錄製樂音」，透過 Arduino 突破播放樂音的數量，符合多樂器、同時演奏的樂團目標。

(二) 系統平台-Arduino微電腦

Arduino 是一款微控制器板，附帶易於學習和使用的開發環境 (IDE)，特色為開發簡便且資源豐富。使用者可以透過 Arduino 板接上多種電子元件，例如 LED 燈、喇叭、馬達、開關、溫濕度感測器、紅外線發射與接收器、LCD 顯示器，以及各種通訊模組。配合自動控制的程式，就能輕鬆實現各種自動化應用。



(三) 琴鍵設計初探

1. 壓電裝置：

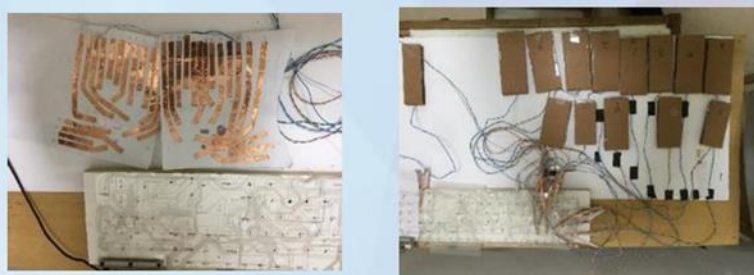
壓電元件透過施加壓力產生電壓或施加電壓產生變形，包括多種壓電開關，在與 Arduino 搭配的感測應用中，常使用數位 I/O 訊號。

2. 鍵盤矩陣 (Keyboard Matrix)

鍵盤矩陣將電路分成行與列，透過掃描的方式逐一掃描行與列，利用偵測電位差，驗證按鍵的位置，解決控制器缺少 I/O 腳的問題。



透過上述原理，拆卸一電腦鍵盤作為鍵盤矩陣的基礎，並以電線連接至不同紙板，透過不同金屬的導通與敲擊回饋，回傳至 Arduino 進行演算。



3. 電阻分壓原理

呈上壓電開關的文獻可知，改用 Arduino 開發的類比輸入腳位，偵測電壓並轉換成訊號對應鍵盤上的音階可有效擴大辨識範圍。Arduino 類比解析度理論值為 0.00489V，但實測類比腳位僅可辨 0.4-0.5V 的電位差，單一腳位可辨識無按壓及 12 個音階。

分壓電路 公式
 輸出電壓 $V_{out} = V_{in} \times \frac{R_2}{R_1 + R_2}$
 電阻 $R_1 = 0$ 則 輸出電壓 $V_{out} = V_{in}$
 電阻 $R_2 = 0$ 則 輸出電壓 $V_{out} = 0$

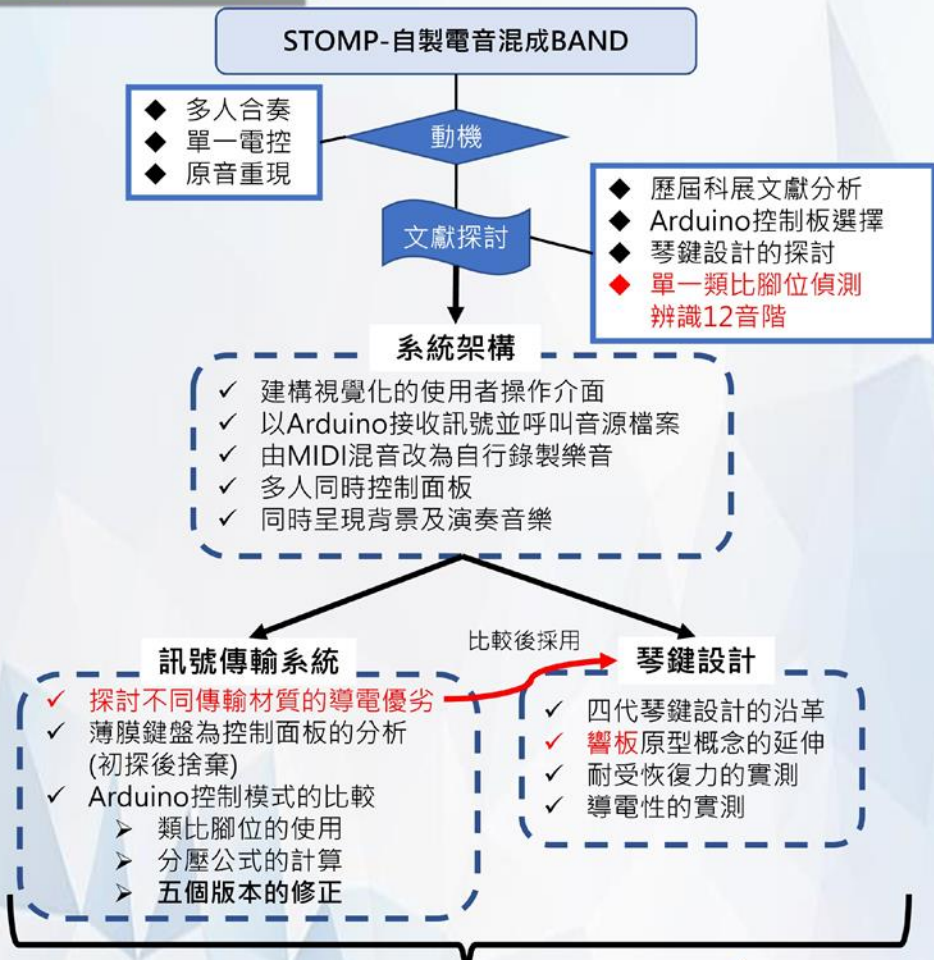
琴鍵順序對應 AB 腳位偵測電壓，求相對應的 R2 電阻值

$$V_{out} = V_1 \frac{R_2}{R_1 + R_2} = \frac{V_1 R_2}{R_1 + R_2}$$

電阻 R1 1097 琴鍵數 12

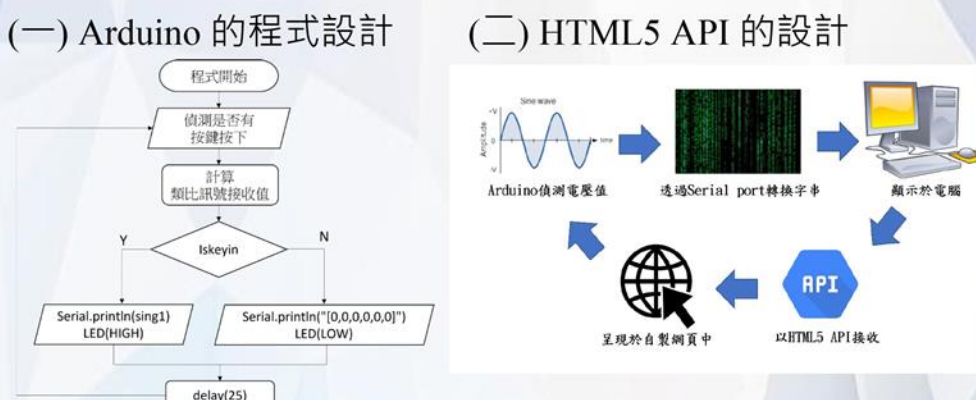
R2 電阻值估測

研究架構



研究方法

研究一：系統架構
系統架構的建立，分為 Arduino 的電壓偵測與字串轉換程式、HTML5 的API 接收與結合開源軟體的自製網頁呈現。



(三) 第二代音源：實際錄製樂音

- 每個音檔的錄音時間皆是**2秒鐘**。
- 不同節拍的樂音使用程式來解決。我們透過程式邏輯撰寫「當重新偵測到訊號時就直接播放新的樂音」，使「**敲擊速度**」成為節奏來源。

研究二：訊號傳輸系統

(一) 比較不同傳輸導線的優劣

- 不同種類導線的電阻值比較
取鋁箔、銅箔、Cat6雙絞網路線、導電銀膠、2B鉛筆等測量電阻。
- 不同長度導線的電阻值比較
取不同長度鋁箔、銅箔、Cat6雙絞網路線測量電阻。

(二) 第一代訊號傳輸系統：薄膜鍵盤
在 A 點處我們使用導電銀膠補強接觸點；在 B 點處使用 Cat6 雙絞網路線減少訊號的散失；在 C 點處使用大面積的鋁箔增加接觸面積。

(三) 第二代訊號傳輸系統：Arduino訊號傳輸系統
Arduino進行了多個版本的改良，考量焊接的便利性、易除錯性，改由電線直接焊接，再加上熱縮管包覆。



研究三：琴鍵結構

- 琴鍵的設計具備可恢復性、高彈力、耐受力佳等優點。
- 為提高琴鍵觸發效能，在「響板」內側貼上大片鋁箔（便宜、易取得且大面積導電性佳），在上下鋁箔面各接上雙絞網路線使【接觸後通電】。

3. 琴鍵結構測試：

(1) 耐受恢復力
先用游標卡尺量測製作好的厚度，將琴槌舉高至30公分，量測進行按壓100次、200次、300次厚度變化。

(2) 導電性測試
以每秒鐘2下的節奏進行連續按壓，使串聯的LED發光，觀察能穩定閃爍幾次，上述實驗重複5次取平均值。

研究結果

研究一：系統架構

(一) Arduino 的程式設計
Arduino開啟A0-A5腳位，以分壓原理計算不同電阻，讓單一類比腳位能偵測12種不同電壓值，轉換為字串，達到單一腳位完成12音的目標。

(二) HTML5 的 API

- 建構使用者操作介面外觀，以馬林巴琴音色為例。
- 連結 Arduino 鍵盤，接收序列傳來的字串資料資料。

3. 判別類比偵測電壓，轉換成相對應的 keycode 按鍵字符編碼。

| 音階 0-11 | 電阻 R1 | 電阻 R2 | 輸出電壓 Vout | 代表音高 |
|---------|-------|--------|-----------|------|
| 0 | 1996 | 0 | 0 | C |
| 1 | 1996 | 100 | 0.42 | CB |
| 2 | 1996 | 219 | 0.83 | D |
| 3 | 1996 | 365 | 1.25 | DB |
| 4 | 1996 | 548 | 1.67 | E |
| 5 | 1996 | 783 | 2.08 | F |
| 6 | 1996 | 1096 | 2.50 | FB |
| 7 | 1996 | 1534 | 2.91 | G |
| 8 | 1996 | 2192 | 3.33 | GB |
| 9 | 1996 | 3288 | 3.75 | A |
| 10 | 1996 | 5480 | 4.16 | AB |
| 11 | 1996 | 12056 | 4.58 | B |
| 12 | 1996 | 548904 | 4.99 | BB |

Arduino 約可辨識 0.4-0.5V 的落差

研究二：訊號傳輸系統

(一) 比較不同傳輸導線的優劣

1. 不同導線的電阻值測試(電線皆長10cm)

| 導線材質測試 | 鋁箔 1cm寬 | 銅箔 6mm寬 | 導電銀膠 | 2B鉛筆(石墨) 1mm寬 | Cat6雙絞網路線 |
|---------|---------|---------|------|---------------|-----------|
| Test 1 | 0.04 | 0.03 | 8.25 | 284000 | 0.03 |
| Test 2 | 0.05 | 0.04 | 8.31 | 285000 | 0.03 |
| Test 3 | 0.05 | 0.04 | 8.52 | 286000 | 0.03 |
| Test 4 | 0.04 | 0.03 | 8.40 | 286000 | 0.04 |
| Test 5 | 0.05 | 0.04 | 8.50 | 285000 | 0.03 |
| Test 6 | 0.04 | 0.03 | 8.45 | 285000 | 0.03 |
| Test 7 | 0.05 | 0.04 | 8.43 | 286000 | 0.03 |
| Test 8 | 0.05 | 0.03 | 8.47 | 285000 | 0.03 |
| Test 9 | 0.04 | 0.04 | 8.42 | 284000 | 0.03 |
| Test 10 | 0.05 | 0.03 | 8.62 | 285000 | 0.03 |
| 平均值 | 0.05 | 0.04 | 8.44 | 285100 | 0.03 |

2. 不同長度的電線電阻值測試

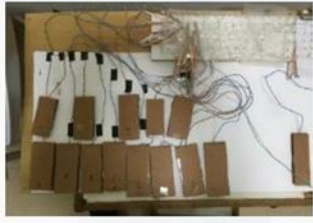
| 導線材質測試 | 鋁箔 1m | 銅箔 1m | 銅箔 2m | 雙絞網路線 1m | 雙絞網路線 2m | 雙絞網路線 4m |
|---------|-------|-------|-------|----------|----------|----------|
| Test 1 | 1542 | 0.31 | 0.51 | 0.31 | 0.41 | 0.56 |
| Test 2 | 1574 | 0.35 | 0.56 | 0.32 | 0.43 | 0.57 |
| Test 3 | 1562 | 0.32 | 0.57 | 0.31 | 0.42 | 0.56 |
| Test 4 | 1489 | 0.36 | 0.56 | 0.30 | 0.41 | 0.58 |
| Test 5 | 1542 | 0.31 | 0.57 | 0.32 | 0.41 | 0.57 |
| Test 6 | 1593 | 0.30 | 0.54 | 0.31 | 0.42 | 0.56 |
| Test 7 | 1574 | 0.35 | 0.59 | 0.30 | 0.41 | 0.57 |
| Test 8 | 1562 | 0.32 | 0.55 | 0.30 | 0.42 | 0.58 |
| Test 9 | 1546 | 0.34 | 0.54 | 0.31 | 0.41 | 0.56 |
| Test 10 | 1562 | 0.31 | 0.55 | 0.32 | 0.42 | 0.57 |
| 平均值 | 1555 | 0.33 | 0.55 | 0.31 | 0.42 | 0.57 |

(二) 第一代訊號傳輸系統：薄膜鍵盤

發現：

薄膜鍵盤導線連接時必需防止導線造成短路，導致鍵盤電路損壞。佈線需要仔細安排將各個音階依序排列，經過多次反覆嘗試後將八度音階對照鍵盤排列如下表：

| | | | | | | | | | | | | |
|------|---|----|---|----|---|---|----|---|----|---|----|---|
| 按鍵符號 | A | X | Z | Q | S | 1 | W | 2 | D | C | E | 3 |
| 音階 | C | C# | D | D# | E | F | F# | G | G# | A | A# | B |

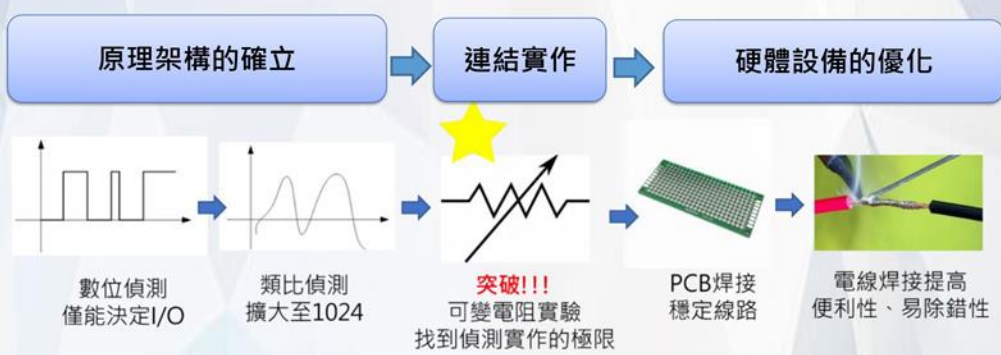


(三) 第二代訊號傳輸系統：Arduino訊號傳遞系統

| 版本 | 2.1版 | 2.2版 | 2.3版 | 2.4版 |
|------|----------------------------|--|--|---------------------|
| 器材 | 麵包板、杜邦線、陶瓷電阻 | 麵包板、杜邦線、可變電阻、陶瓷電阻 | 麵包板、杜邦線、可變電阻、陶瓷電阻 | PCB板、可變電阻、陶瓷電阻 |
| 原理 | 數位訊號偵測通電琴鍵的O/I | 相同電阻串聯，類比訊號辨識通電琴鍵的解析度 | 分壓原理計算電阻值，以類比訊號偵測通電琴鍵的電壓值，提高序列速率至115200 | |
| 圖例 | | | | |
| 音階 | 至多8階 | 理論值約20階 | 理論值約60階 (實測至24階) | 理論值約60階 (實測至24階) |
| 觀察描述 | 單片Arduino晶片至多完成8個音階，不符合需求。 | 類比腳位偵測陶瓷電阻的串聯，單一腳位至多偵測4-5個音階。單片Arduino晶片約可完成20個音階(A0-A5) | 類比腳位偵測可變電阻，單一腳位即可偵測12音階。單片Arduino晶片約可完成60個音階。但麵包板與杜邦線會因震動脫落。 | 優點如同2.3版本，且避免震動脫落 |

Arduino 訊號傳輸系統 2.5 版本

將 Arduino 訊號傳輸系統 2.4 版本自 PCB 板,改為電線焊接。提高焊接的便利性、易除錯性。達到穩定傳輸的目標。



結論

一、本研究真實解決了我們在家練習馬林巴琴時無樂音的現況。使得我們的自製樂器中【多人合奏、多重音色、原音重現】的期待得以完整的表現。並更進一步設計小型電子演奏團，重現在校練習樂團的複合樂音表現。

二、本研究的歷程可以分為三大項：

- (一)以回收材料自製鍵盤並數據分析鍵盤優勢，將「彈奏」轉換為「敲擊」。
- (二)捨棄原有的音源播放方式，將「軟體合成音樂」轉換為「真實錄製樂音」。
- (三)透過 Arduino 突破播放樂音的數量，符合多樂器、同時演奏的樂團目標。

三、本次使用的網頁設計軟體、音源檔案皆使用開源資料。為響應網路上開源資料的分享，也在本次報告中公開程式碼、音源檔案，並上傳至 GitHub，有助於未來有興趣的研究者取用測試。

參考文獻

- 民國 72 年高中科展-新式電腦音樂合成器 <https://twsf.ntsec.gov.tw/activity/race/1/23/pdf/23h/196.pdf>
- 民國 106 年國小科展-真"水"鋼琴 <https://www.ntsec.edu.tw/ScienceContent.aspx?cat=&a=0&fld=&key=&isd=1&icop=10&p=226&sid=13877>
- 民國 106 年高中科展-「MIDI 樂器 (電子琴)」 <https://www.ntsec.edu.tw/ScienceContent.aspx?a=6821&fld=&key=&isd=1&icop=10&p=1&sid=13767>
- 民國 109 年國小科展-「微」鋼琴三部曲 <https://www.ntsec.edu.tw/ScienceContent.aspx?cat=&a=0&fld=&key=&isd=1&icop=10&p=59&sid=16656>
- 壓電開關介紹 <https://www.rohm.com.tw/electronics-basics/piezo/what1>
- 天花板隨記- Arduino 筆記(85)：電壓感測器 <https://atceiling.blogspot.com/2020/10/arduino85voltage-sensor.html>
- Arduino 連線原始碼 <https://github.com/svendahlstrand/web-serial-api>
- 開源音源檔 <https://keithwhor.com/music/>
<https://keithwhor.github.io/audiosynth>
- 視覺化 <https://css-tricks.com/making-an-audio-waveform-visualizer-with-vanilla-javascript>

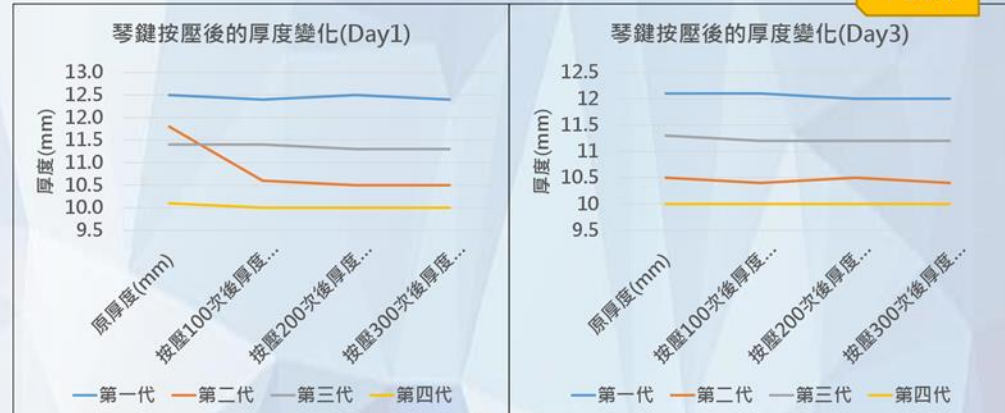
研究三：琴鍵結構

(一) 琴鍵結構耐受恢復力測試

針對琴鍵結構進行耐受恢復力、導電性測試，以求得到具有良率、可長時間使用的效果。其實驗描述如下：

| 時間 | 厚度變化 | 第一代 | 第二代 | 第三代 | 第四代 |
|------|----------------|------|------|------|------|
| Day1 | 原厚度(mm) | 12.5 | 11.8 | 11.4 | 10.1 |
| | 按壓 100 次後厚(mm) | 12.4 | 10.6 | 11.4 | 10.0 |
| | 按壓 200 次後厚(mm) | 12.5 | 10.5 | 11.3 | 10.0 |
| | 按壓 300 次後厚(mm) | 12.4 | 10.5 | 11.3 | 10.0 |
| Day3 | 原厚度(mm) | 12.1 | 10.5 | 11.3 | 10.0 |
| | 按壓 100 次後厚(mm) | 12.1 | 10.4 | 11.2 | 10.0 |
| | 按壓 200 次後厚(mm) | 12.0 | 10.5 | 11.2 | 10.0 |
| | 按壓 300 次後厚(mm) | 12.0 | 10.4 | 11.2 | 10.0 |

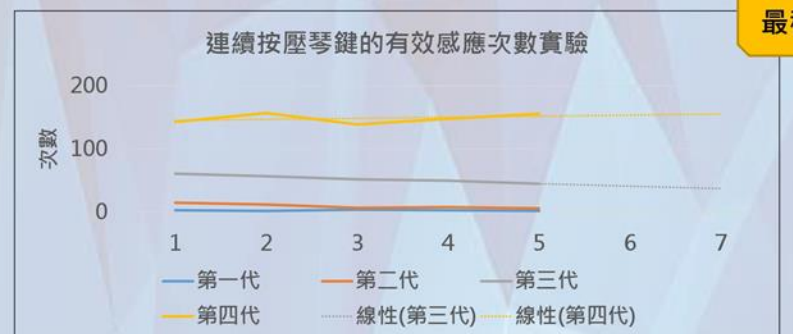
不變形



(二) 琴鍵連續按壓可連續偵測通電的次數

| 可連續按壓次數 | 第一代 | 第二代 | 第三代 | 第四代 |
|-----------|-----|-----|-----|-------|
| 連續按壓 1(次) | 3 | 15 | 61 | 143 |
| 連續按壓 2(次) | 2 | 12 | 57 | 157 |
| 連續按壓 3(次) | 4 | 7 | 52 | 139 |
| 連續按壓 4(次) | 3 | 8 | 50 | 148 |
| 連續按壓 5(次) | 2 | 6 | 45 | 156 |
| 平均值 | 2.8 | 9.6 | 53 | 148.6 |

最穩定



四、針對本作品與市售樂器的分析比較如下：

| 產品 | 電鋼琴 | 馬林巴琴 | 鐵琴 | STOMP-自製電音 BAND |
|------|-------------------|--------------|----------------|-----------------|
| 價格 | \$5000-\$200000 ↑ | \$1200000 ↑ | \$450000 ↑ | 成本\$5,000以內 |
| 可攜帶性 | 固定大小 | 固定大小 | 固定大小 | 依需求攜帶音域及大小 |
| 音域 | 3-5個音域 | 5個音域 | 3個音域 | 1-6個音域 (實測5個音域) |
| 音色 | 可產生多元化混成音 | 單一 | 單一 | 可錄製多元化的真實樂器音 |
| 音量 | 可精準調整大小 | 須依經驗調整力道控制音量 | 須依經驗判斷調整力道控制音量 | 可精準調整大小 |
| 使用人數 | 一次一人 | 一次一人 | 一次一人 | 設定不同樂器音可同時多人演奏 |