

中華民國第 65 屆中小學科學展覽會

作品說明書

高級中等學校組 電腦與資訊學科

第一名

052515

單手操作無障礙一鍵盤與滑鼠的整合設計研究

學校名稱：啟英學校財團法人桃園市啟英高級中等學校

作者：

高二 林魯林

高二 許家瑜

高二 王可安

指導老師：

連素玲

賴曉貞

關鍵詞：鍵鼠融合、單手操作、無障礙設計

得獎感言

用創意，補回被天使借走的一隻手

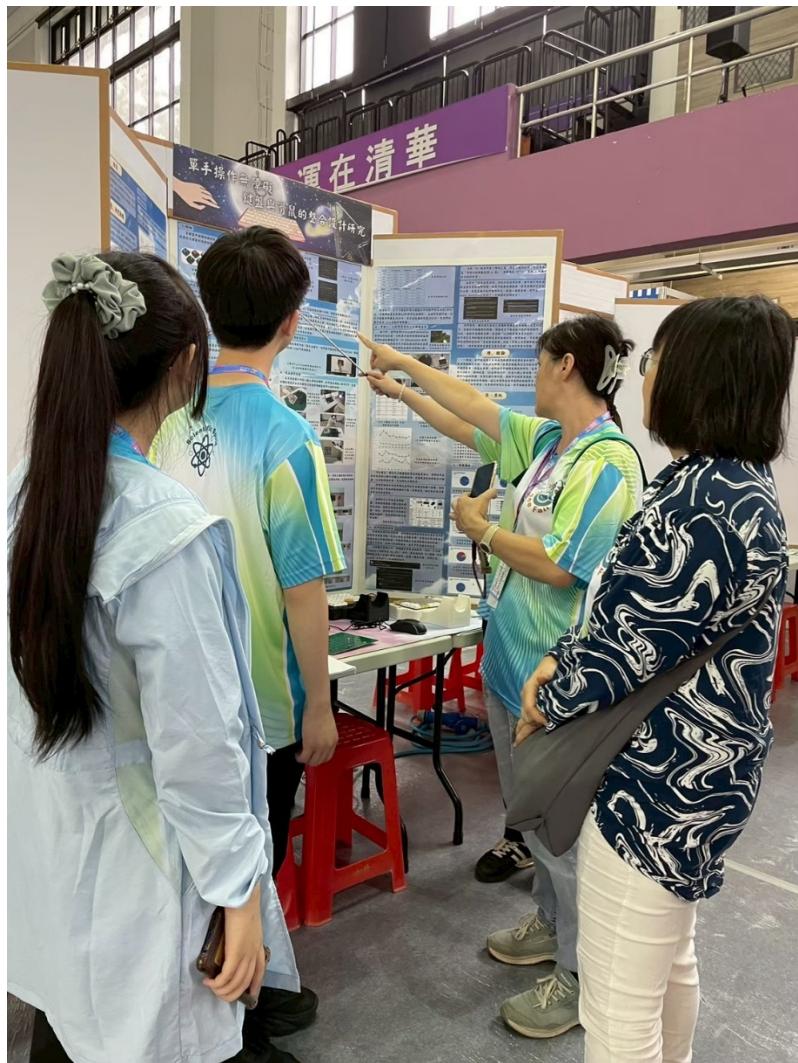
我們的專題，起點是一位身邊的朋友。我們的組員家瑜，她天生只有一隻手，卻依然熱愛使用電腦、打字、上網。但對她來說，傳統的鍵盤與滑鼠配置，總是讓操作變得笨拙又耗時。看著她一次次努力去克服不便，我們心裡想：「如果我們能用自己的專長，讓她能像雙手一樣自然地操作電腦，那該多好？」於是，我們決定開始這個專題，把「鍵盤與滑鼠合而為一」的想法變成現實。

從那一刻開始，我們的生活就被各種挑戰填滿。電路板第一次打印就出問題，封裝錯誤讓焊接變成不可能的任務；3D 列印外殼一次次的修改，調整到手托曲線能真正貼合手腕；韌體程式寫到半夜卡在一個 Bug 上，我們拖著疲憊的身軀盯著螢幕，一行一行檢查程式碼。這些挫折並沒有擊倒我們，反而讓我們一次次想出更好的解法，因為我們知道——這是為了幫助我們最在乎的人。

放假的日子，我們幾乎把圖書館的會議室當成我們的實驗室。有人負責焊接，有人敲程式，有人測試鍵帽手感；討論各種事情到深夜是常態，因為我們想把每一個細節做到最好。可安雖然通勤很辛苦，卻一次也沒有缺席；家瑜則用她的親身體驗，幫我們找出真正影響單手操作的細節；而我作為組長，最慶幸的是能和這兩位夥伴一起，把一個看似遙不可及的想法一步步做出來。

今天能站在這裡，拿到全國科展第一名，不只是榮耀，更像是一個溫暖的答案——告訴我們，努力真的能改變一些事情。謝謝指導老師在我們迷路時給予方向，謝謝家人無條件的支持，也謝謝所有參與測試、給我們建議的人。

我們想把這個獎獻給每一位在生活中面對不便卻依然不放棄的人。願我們的作品，能成為某些人的一雙新手，也希望我們的故事能告訴大家：當你心中有一個想幫助的對象時，你會找到千百種方法去實現它。限制或許存在，但我們的創意與努力永遠能把它們打破。



比賽佈展現場，組員與指導老師共同確認展示細節



專題佈展完成後的團隊合照，正式邁向比賽的舞台



全體組員與兩位指導老師的合影

單手操作無障礙一鍵盤與滑鼠的整合設計研究

摘要

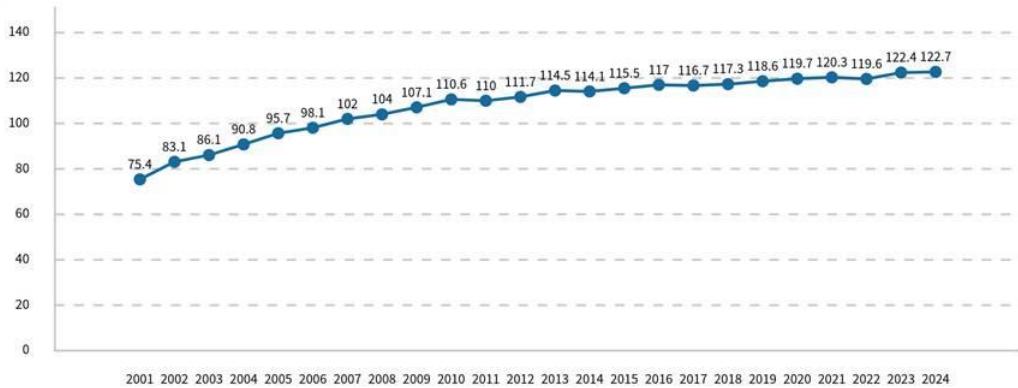
本研究旨在解決肢體障礙或單手使用者在操作電腦時所面臨的不便，設計一款融合鍵盤與滑鼠功能的單手操作裝置。透過自製 PCB 電路板、客製化鍵帽與鍵軸配置、3D 列印人體工學外殼，以及整合滑鼠感測器與 QMK 韌體技術，實現單手即可同時完成鍵入與游標控制的操作方式。我們的實驗結果顯示，單手使用時相較於傳統鍵鼠組合，本裝置能大幅減少手腕移動的距離與次數，有效提升使用效率並降低疲勞感。產品設計不僅針對身障者，也適用於特定單手使用情境，具備良好的通用性與實用價值。未來將持續優化硬體與軟體，期望推廣至更廣泛的使用族群，實踐科技輔具的社會意義。

壹、研究動機

隨著科技的快速發展，尤其是數位裝置與人機互動界面的進步，現代社會中對於電腦操作的需求愈加普遍。然而，對於某些特定族群（例如有肢體障礙、單手操作需求的使用者等）來說，一般設計的雙手電腦操作方式並不方便他們使用。

發想來自我們的一位組員是位肢體障礙者，因為先天因素僅能使用右手生活，在討論過後發現她有操作電腦不便的問題，為了解決這件事，我們開始朝著解決肢障者使用鍵鼠的困難設計這項作品。上網查閱資料後發現，市面上並沒有專為單手使用者開發、能「同時」操作鍵鼠的相關產品。這對於有肢體障礙或需要單手操作的使用者而言，傳統的鍵盤與滑鼠設計已無法滿足他們的操作需求，對其工作效率和身體健康造成不利影響。為此，我們設計了一個鍵盤與滑鼠融合的工具，來解決這些使特殊使用者的操作問題，使其操作起來更加方便。

根據統計，在總人口數都約為 2300 萬左右的狀況下，身心障礙的總人數從 2001 年的約 75 萬人到 2024 年已增加至約 120 萬，其中「肢體障礙類」人數占比最高，從 30.6 萬人增加到了 39.67 萬人，傳統的鍵盤和滑鼠配置已無法滿足其操作需求，大大影響了他們的日常生活和工作效率。(人數由內政部統計)



(圖 1)2001~2024 身心障礙的總人數(內政部統計)折線圖 (單位：萬)
(出處詳附錄圖片來源 1)

貳、研究目的

傳統的鍵盤和滑鼠在設計之初，大多是以雙手操作為主要考量，這對於一般使用者來說已經非常便利。然而，對於需要單手操作的使用者而言，這種設計在實際使用過程中往往會帶來一些困擾，特別是單手操作時需要頻繁地在鍵盤與滑鼠之間來回切換、某些快捷鍵間距過大等問題，顯得格外不便。

我們的目標是解決「單手操作電腦」不便的問題，不只是針對有肢體障礙的使用者，甚至是可以給一般的使用者使用。我們設計了一款將鍵盤與滑鼠結合的設備，讓使用者能夠用單手同時進行打字和滑鼠操作，解決頻繁切換鍵鼠的困擾，提升操作效率和使用體驗。這樣的設計不僅適合身心障礙者，也能讓一般使用者在特定情境下更方便地使用電腦。

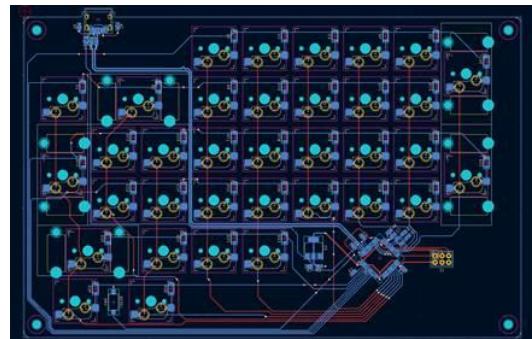
參、研究設備及器材

一、KiCad 7.0

KiCad 7.0 是一款開源的電子設計自動化 (EDA) 軟體，適用於電路原理圖設計與印刷電路板 (PCB) 佈局。該版本帶來更優化的使用者介面、強化的 3D 視圖、改進的設計規則檢查 (DRC)，以及更高效的元件管理。KiCad 7.0 支援多層 PCB 設計、差分對與阻抗匹配等高級功能，並能與多種格式相容，方便與其他 EDA 軟體協作。其強大的功能與開源特性，使其成為我們設計電路板的理想選擇。



(圖 2) KiCad 7.0 logo
(出處詳附錄圖片來源 2)



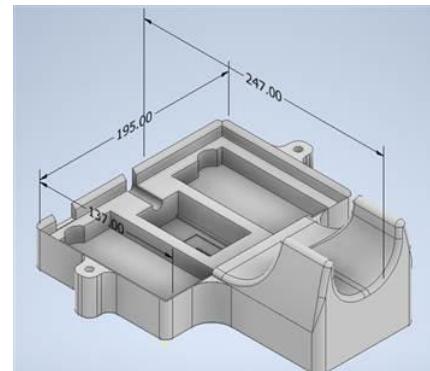
(圖 3)PCB 設計界面
(出處詳附錄圖片來源 1)

二、Autodesk Inventor

Autodesk Inventor 是一款由 Autodesk 開發的 3D 機械設計與建模軟體，專為工程師與設計師打造。它提供強大的參數化建模、裝配設計、模擬分析與工程圖功能，使使用者能夠高效創建與優化產品設計。Inventor 支援多種 CAD 格式並與 AutoCAD 無縫整合，適用於機械設計、製造與產品開發領域。



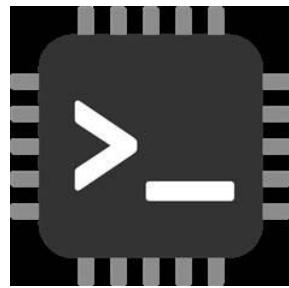
(圖 4)Autodesk Inventor logo
(出處詳附錄圖片來源 3)



(圖 5)模型設計界面
(出處詳附錄圖片來源 1)

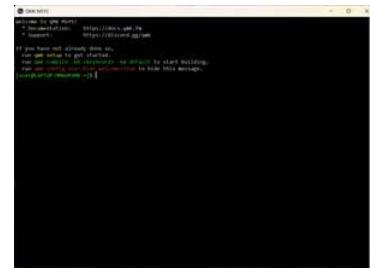
三、QMK MSYS

QMK MSYS 是一個專為 QMK Firmware 開發環境設計的工具集，基於 MSYS2，提供 Windows 使用者簡單的方式來編譯、修改和燒錄 QMK 韌體。它包含必要的工具鏈，如 GCC、Make 和 Python，讓開發者能夠在 Windows 平台上無縫使用 QMK CLI 來管理鍵盤配置。QMK MSYS 降低了 Windows 使用者設置開發環境的門檻，使客製化鍵盤韌體的編譯和測試變得更加方便。



(圖 6)QMKS logo

(出處詳附錄圖片來源 4)

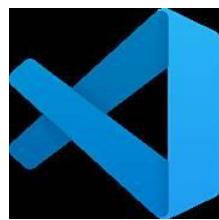


(圖 7)操作界面

(出處詳附錄圖片來源 1)

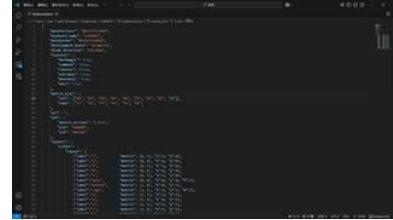
四、Visual Studio Code

Visual Studio Code (VS Code) 是一款由微軟開發的免費、開源程式碼編輯器，支援 Windows、macOS 和 Linux。它具備強大的語法高亮、智慧感知 (IntelliSense)、內建 Git 控制、除錯工具及擴充功能市場，適用於多種程式語言，如 Python、JavaScript、C++ 等。VS Code 輕量但功能強大，透過各種擴充套件可客製化開發環境，使其成為我們編寫程式的最佳選擇。



(圖 8)Visual Studio Code logo

(出處詳附錄圖片來源 5)



(圖 9)編輯介面

(出處詳附錄圖片來源 1)

五、AutoHotkey

AutoHotkey 是一款免費且開源的自動化腳本工具，專為 Windows 系統設計。可用於快速建立鍵盤快捷鍵、巨集、自動化重複動作，甚至開發簡單的 GUI 工具。透過簡潔易懂的語法，使用者可輕鬆自訂鍵位映射、滑鼠控制與文字輸入等功能，是提升工作效率與打造個人化操作環境的利器。



(圖 10)AutoHotkey logo

(出處詳附錄圖片來源 6)



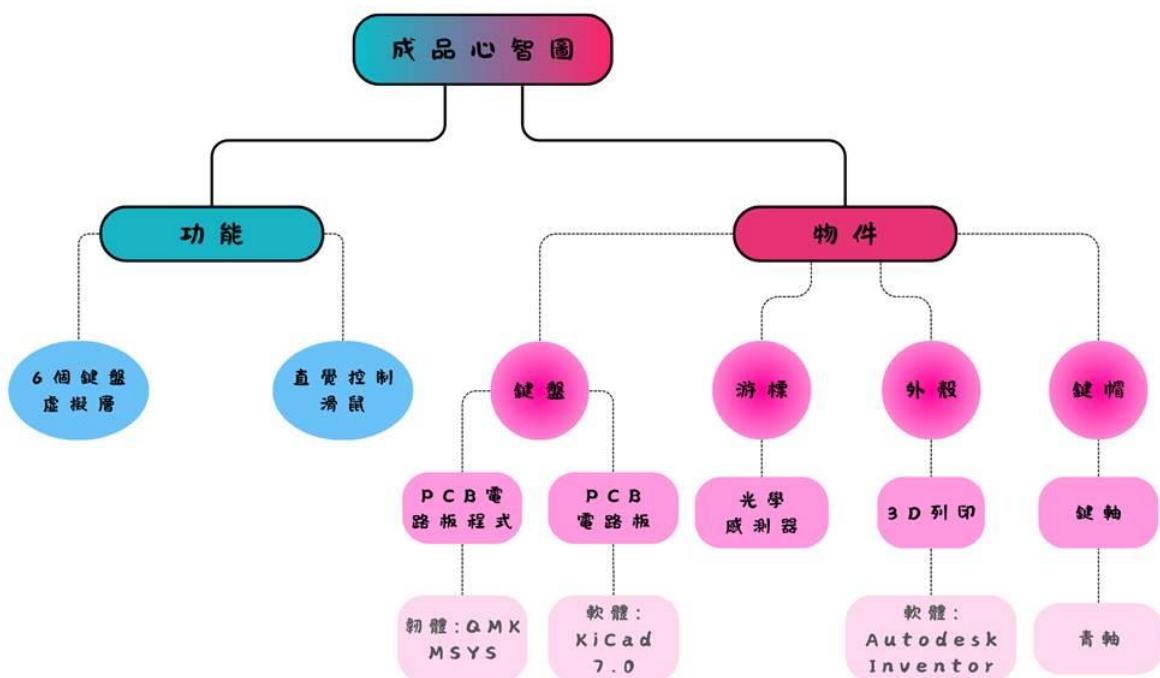
(圖 11)編輯介面

(出處詳附錄圖片來源 1)

肆、研究過程或方法

一、設計過程

在主題方向決定好後，我們便開始著手進行下一步的準備工作。在經過大家的通力合作與努力，將相關的資料充分蒐集整理完畢後，我們進一步展開了具體的工作分配。整個專題的製作內容主要被分為兩個部分：一部分是進行書面的內容撰寫與設計，另一部分則是專注於作品的實際製作。在完成這兩個部分後，我們再將所有的成果進行彙整，並組裝成完整的成品。最後經過反覆的測試與檢查，確認成品沒問題就製作完成了。



(圖 12)成品心智圖 (出處詳附錄圖片來源 1)

二、設計相關原理

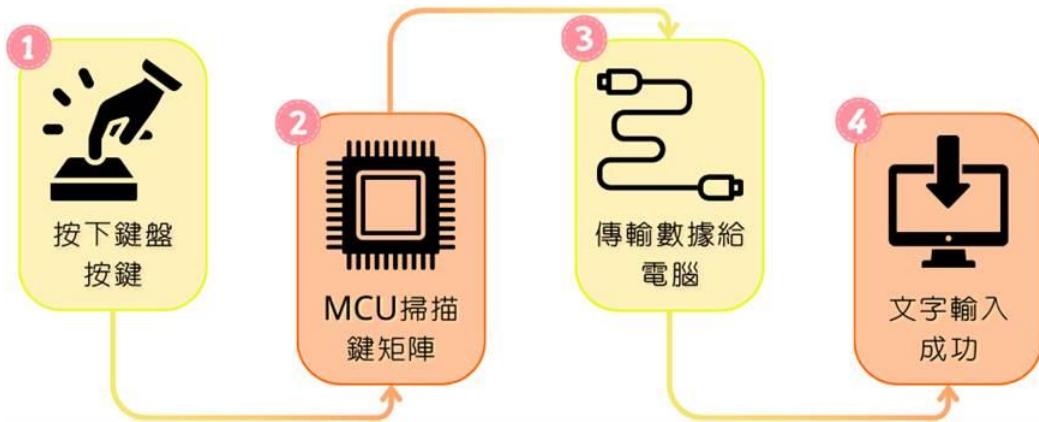
我們的專案可以簡單分為三大部分：

1. 完全客製化的按鍵佈局(鍵盤)
2. 控制游標功能(滑鼠)
3. 用於整合以上兩者的外殼(3D 列印)

自製鍵盤部分可分為屬於硬體的：鍵帽、鍵軸、PCB 電路板、外殼(含定位板)，以及屬於韌體的 QMK (燒錄在鍵盤微控制器裡的鍵盤程式)；控制游標部分需要一個滑鼠用

的光學感測器，安裝在外殼的最底下；外殼部分則是使用 3D 建模軟體畫出後，交由廠商印出並進行整合。

三、單手鍵盤製作



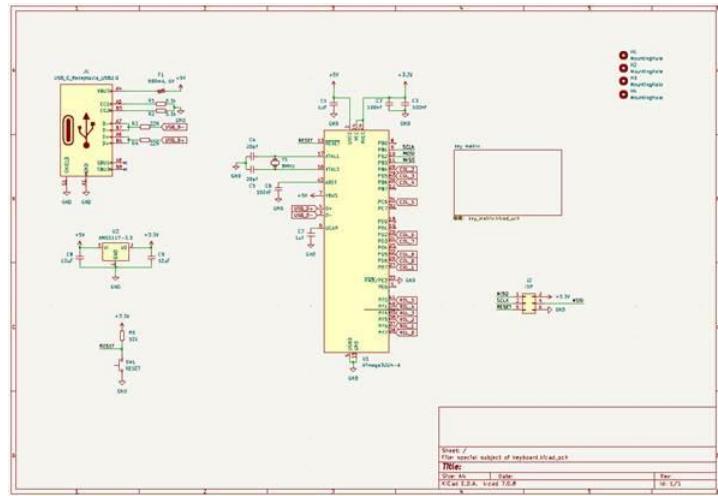
(圖 13)鍵盤輸入流程圖 (出處詳附錄圖片來源 1)

(一)PCB 電路板

1. 繪製電路藍圖

在繪製藍圖前，我們需先決定鍵盤所使用的微控制器(MCU)。在參考了許多網路資料後，我們選擇了 ATmega32U4，主要優點包括：容易焊接、價格合理、周邊電路佔用空間較少，且未來可擴充藍牙功能。

下圖(圖 14)分別是 ATmega32U4 微控制器(圖 14 中間)、TYPE-C 母座(圖 14 左上)、方框縮小的鍵矩陣，以及其他構成電路所需的周邊電路(圖 14 較小部分 3 處)。



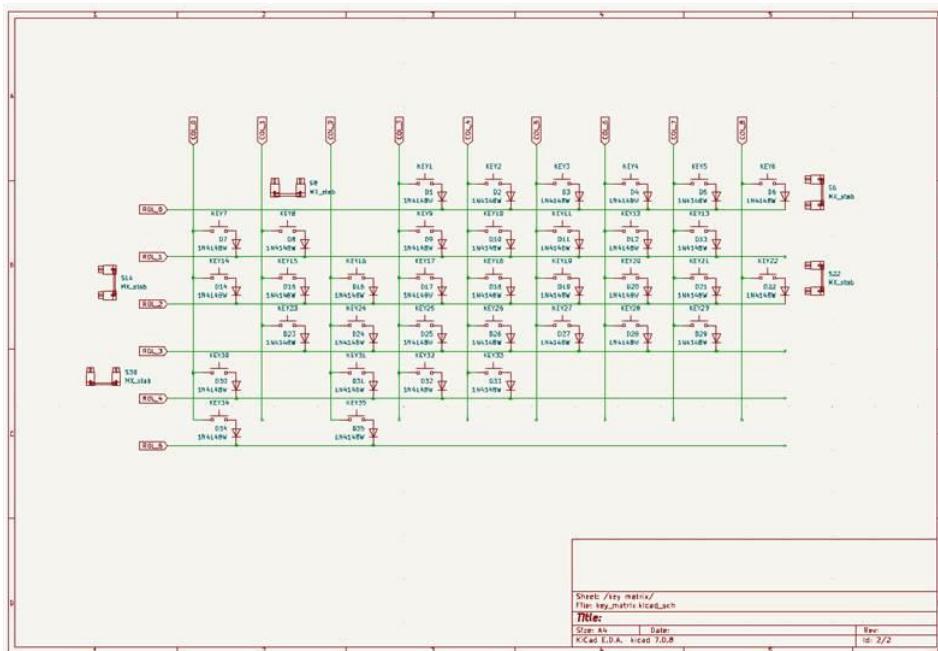
(圖 14)鍵盤 KiCad 電路藍圖 (出處詳附錄圖片來源 1)

2. 繪製鍵矩陣 (Key Matrix)

由於鍵盤的按鍵數量遠超過微控制器之針腳數量，所以要使用鍵矩陣（Key Matrix）來減少微控制器偵測全部按鍵所需的針腳數量。

簡單來說，鍵矩陣將按鍵排列成行 (Row) 和列 (Column) 兩個方向的交叉點：

1. 每個按鍵皆有一個特定的行和列交點。
 2. 控制器會依次掃描行與列，檢查是否有電流通過某個交點。
 3. 如果某個交點檢測到電流，控制器就能知道是哪個按鍵被按下。



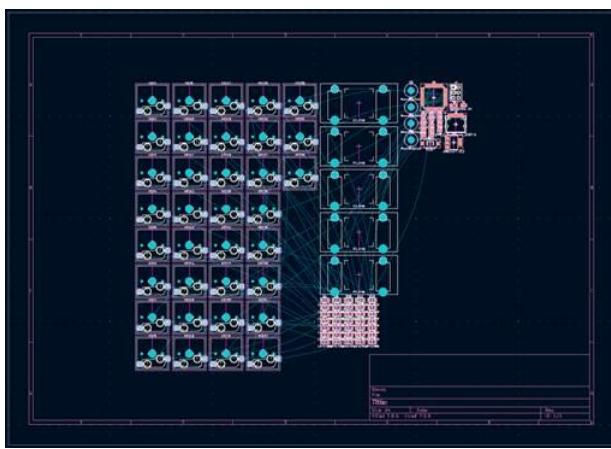
(圖 15)在 KiCad 中繪製的鍵矩陣 (Key Matrix) 藍圖 (出處詳附錄圖片來源 1)

使用鍵矩陣的優點

1. 節省線材：例如，4 行 \times 4 列的鍵矩陣只需要 8 條連接線，而不是 16 條。
2. 支持多鍵檢測：透過掃描技術，可以檢測同時按下的多個按鍵。

3. 佈置線路與擺放元件

在上一步驟中，我們已經完成了元件的選擇與線路的連接。接下來，我們要將電路圖 (SCH) 轉換為 PCB 佈局，規劃並擺放元件，最終製成 PCB 電路板的設計圖。繪製好電路圖 (SCH) 後，點擊『工具』 \rightarrow 『切換到 PCB 編輯器』（如圖 17）。切換後的畫面如（圖 18）。



(圖 16)KiCad 中的 PCB 編輯器

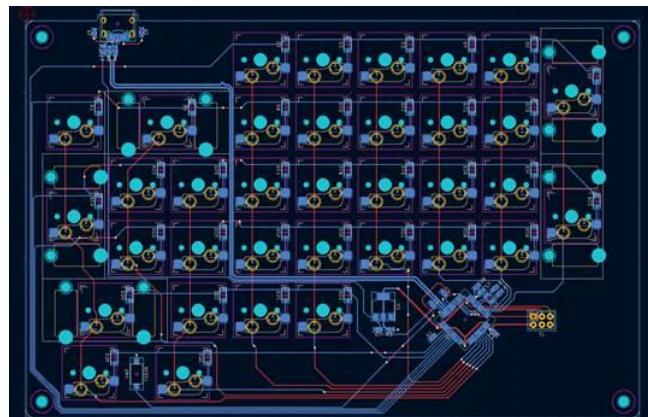
(出處詳附錄圖片來源 1)



(圖 17)KiCad 中的工具欄

(出處詳附錄圖片來源 1)

接下來，我們要規劃這些元件在電路板上的位置並安排走線。需要注意的是，不可將容易發熱的元件過於密集地放置，走線應盡量避免靠近電路板上的孔洞（例如螺絲孔），而電源相關線路應使用較粗的導線。完成佈置後的 PCB 電路板如下（圖 18）。

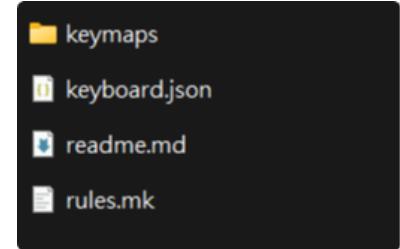


(圖 18)在 PCB 編輯器中擺放好元件 (出處詳附錄圖片來源 1)

4.QMK 韌體製作及燒錄（鍵盤程式）

再來要撰寫鍵盤程式，我們使用的軟體為 QMK MSYS。

首先，在 QMK MSYS 中建立一個專案，並選擇 ATmega32U4 作為微控制器。QMK 會自動生成預設鍵盤程式所需的各個檔案（如圖 19 所示）。接下來，需要修改檔案內容以匹配我們的專案需求，需調整的文件為 keyboard.json 和 keymap.c。



(圖 19)QMK 的資料夾

(出處詳附錄圖片來源 1)

keyboard.json 檔中，需要修改兩個部分：

- (1) layout-按鍵名稱&位置
- (2) matrix-指定微控制器的腳位

首先是 keyboard.json 的 layout 部分（圖 20）。「label」要填入每個按鍵的「名稱」，例如 Enter；「matrix」要填入每個按鍵在鍵矩陣中的位置〔row, col〕；X、Y 則要填入每個按鍵在現實的實際位置。

```
```json
"layout": [
 {"label": "1", "matrix": [0, 3], "x": 3, "y": 0},
 {"label": "2", "matrix": [0, 4], "x": 4, "y": 0},
 {"label": "3", "matrix": [0, 5], "x": 5, "y": 0},
 {"label": "4", "matrix": [0, 6], "x": 6, "y": 0},
 {"label": "5", "matrix": [0, 7], "x": 7, "y": 0},
 {"label": "back", "matrix": [0, 8], "x": 8, "y": 0, "h": 2},
 {"label": "delete", "matrix": [1, 0], "x": 0, "y": 1},
 {"label": "caps", "matrix": [1, 1], "x": 1, "y": 1, "w": 2},
 {"label": "q", "matrix": [1, 3], "x": 3, "y": 1},
 {"label": "w", "matrix": [1, 4], "x": 4, "y": 1},
 {"label": "e", "matrix": [1, 5], "x": 5, "y": 1},
 {"label": "r", "matrix": [1, 6], "x": 6, "y": 1},
 {"label": "t", "matrix": [1, 7], "x": 7, "y": 1},
 {"label": "shift", "matrix": [2, 0], "x": 0, "y": 2, "h": 2},
 {"label": "alt", "matrix": [2, 1], "x": 1, "y": 2},
 {"label": "ctrl", "matrix": [2, 2], "x": 2, "y": 2},
 {"label": "a", "matrix": [2, 3], "x": 3, "y": 2},
 {"label": "s", "matrix": [2, 4], "x": 4, "y": 2},
 {"label": "d", "matrix": [2, 5], "x": 5, "y": 2},
 {"label": "f", "matrix": [2, 6], "x": 6, "y": 2},
 {"label": "g", "matrix": [2, 7], "x": 7, "y": 2},
 {"label": "Enter", "matrix": [2, 8], "x": 8, "y": 2, "h": 2},
 {"label": "Tab", "matrix": [3, 1], "x": 1, "y": 3},
 {"label": "win", "matrix": [3, 2], "x": 2, "y": 3},
 {"label": "z", "matrix": [3, 3], "x": 3, "y": 3},
 {"label": "x", "matrix": [3, 4], "x": 4, "y": 3},
 {"label": "c", "matrix": [3, 5], "x": 5, "y": 3},
 {"label": "v", "matrix": [3, 6], "x": 6, "y": 3},
 {"label": "b", "matrix": [3, 7], "x": 7, "y": 3},
 {"label": "space", "matrix": [4, 0], "x": 0, "y": 4, "w": 2},
 {"label": "fn", "matrix": [4, 2], "x": 2, "y": 4},
 {"label": "wheel up", "matrix": [4, 3], "x": 3, "y": 4},
 {"label": "wheel down", "matrix": [4, 4], "x": 4, "y": 4},
 {"label": "Mouse left", "matrix": [5, 0], "x": 0, "y": 5, "w": 1.5},
 {"label": "Mouse right", "matrix": [5, 2], "x": 1.5, "y": 5, "w": 1.5}
]
```

```

(圖 20)keyboard.json 檔中的 layout 區

(出處詳附錄圖片來源 1)

再來是 matrix pins 的部分（圖 21）。這裡要填入鍵矩陣中的每條 row 和 col 指定的 ATmega32U4 腳位。

```
"matrix_pins": {  
    "cols": ["D6", "D7", "B4", "B5", "B6", "C6", "D2", "D3", "D5"],  
    "rows": ["F7", "F6", "F5", "F4", "F1", "F0"]  
},
```

(圖 21)keyboard.json 檔中的 matrix pins 區 (出處詳附錄圖片來源 1)

第二個需要修改的檔案是 keymap.c（圖 22）。這裡需要寫入所有按鍵對應的功能鍵碼，例如，Space 鍵對應的鍵碼為 KC_SPC，Enter 鍵對應的鍵碼為 KC_ENTER。

```
[0] = LAYOUT(  
    KC_1, KC_2, KC_3, KC_4, KC_5, KC_BSPC,  
    KC_DELETE, KC_CAPS_LOCK, KC_Q, KC_W, KC_E, KC_R, KC_T,  
    KC_LSFT, KC_LALT, KC_LCTL, KC_A, KC_S, KC_D, KC_F, KC_G, KC_ENTER,  
    KC_TAB, KC_LWIN, KC_Z, KC_X, KC_C, KC_V, KC_B,  
    KC_SPC, MO(1), MS_WHLU, MS_WHLD,  
    MS_BTN1, MS_BTN2  
,  
  
[1] = LAYOUT(  
    KC_6, KC_7, KC_8, KC_9, KC_0, KC_O, KC_P,  
    _____, _____, KC_Y, KC_U, KC_I, KC_O, KC_P,  
    _____, _____, KC_N, KC_H, KC_J, KC_K, KC_L, KC_COLON, _____,  
    _____, _____, KC_M, KC_KP_COMM, KC_KP_DOT, KC_KP_SLASH,  
    _____, _____  
)
```

(圖 22)keymap.c 檔案 (出處詳附錄圖片來源 1)

Fn 鍵是鍵盤中最有趣的按鍵。它的鍵碼為 MO_(n)，意思是切換到鍵盤的第 n 層。當單一層空間不足，需要一些組合鍵時，使用它會非常方便。

完成鍵盤程式後編譯並燒錄到 ATmega32U4 中後，鍵盤電路部分就完成了。

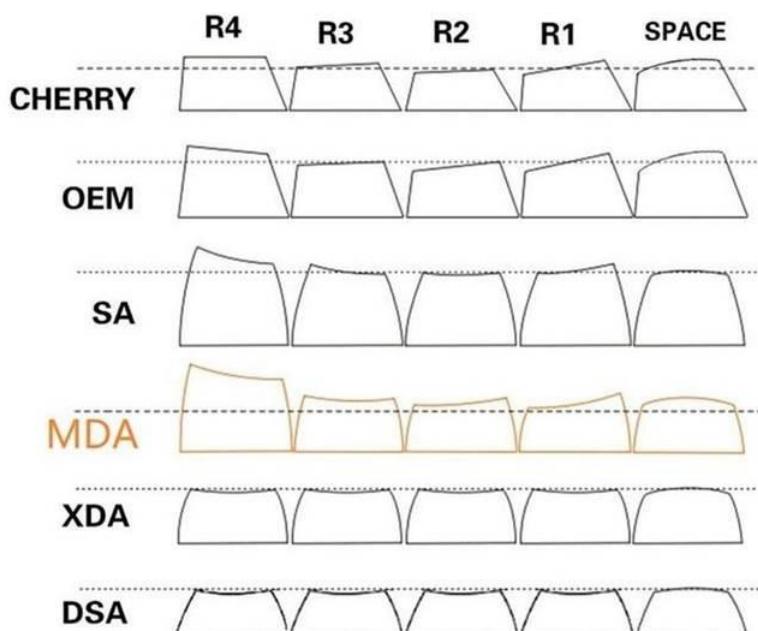
(二)鍵帽

鍵帽 (Keycap) 是指鍵盤上覆蓋在按鍵開關 (Switch) 上的塑膠或其他材質的帽子部分。它是使用者實際接觸到的表面，因此在觸感、設計與耐用性方面具有極高的重要性。

鍵帽的高度與形狀會影響打字的手感，常見的鍵帽高度包括 OEM、Cherry、SA、DSA 等。

(表 1)各種鍵帽類型及特色

| Profile | R1 | R2 | R3 | R4 | Space 鍵 | 特色 |
|---------|----|----|------|----|---------|---------------------------|
| Cherry | 低 | 中 | 坡度適中 | 最高 | 弧形 | 低高度，較符合人體工學，適合長時間打字或遊戲使用。 |
| OEM | 低 | 中 | 坡度適中 | 最高 | 弧形 | 與 Cherry 相似但稍高，常見於市售鍵盤 |
| SA | 高 | 高 | 圓頂形狀 | 高 | 弧形 | 高鍵帽，復古造型，手感飽滿 |
| MDA | 中 | 中 | 微圓頂 | 中 | 平緩 | 介於 SA 和 XDA 之間，較矮的球形鍵帽 |
| XDA | 均高 | 均高 | 平頂 | 均高 | 平緩 | 所有鍵帽高度一致，適合平面佈局 |
| DSA | 均低 | 均低 | 平頂 | 均低 | 平緩 | 低鍵帽且所有行列高度相同，適合任意排列 |



(圖 23)各種不同高度及形狀的鍵帽 (出處詳附錄圖片來源 7)

(三)鍵軸

鍵軸是機械式鍵盤中每個按鍵的核心組件，它負責檢測按鍵被按下的動作，並將訊號傳送至鍵盤的微控制器。鍵軸的設計影響著按鍵的觸感、回饋和聲音。

鍵軸主要可分為按壓過程平滑無段落感的線性軸 (Linear)、按壓時有明顯的段落感或觸發點的段落軸 (Tactile) 以及除了段落感還有會發出清脆按鍵聲響的點擊軸 (Clicky)。

常見的鍵軸有 Cherry MX 系列、Gateron 系列、Kailh 系列、Romer-G、Outemu 系列等。

(表 2)各種品牌之青軸比較

| 參數 | Cherry MX 青軸 | Gateron 青軸 | Kailh 青軸 | Romer-G 青軸 | Outemu 青軸 |
|------|--------------|------------|----------|------------|-----------|
| 觸發壓力 | 50g | 55g | 50g | 45g | 60g |
| 總行程 | 4.0mm | 4.0mm | 4.0mm | 3.2mm | 4.0mm |
| 觸發行程 | 2.2mm | 2.3mm | 2.0mm | 1.5mm | 2.4mm |
| 壽命 | 5,000 萬次 | 5,000 萬次 | 5,000 萬次 | 7,000 萬次 | 5,000 萬次 |
| 聲音 | 清脆 | 清脆 | 較脆 | 較悶 | 較響亮 |
| 手感 | 明顯段落感 | 順滑段落感 | 較輕快 | 段落感較輕 | 明顯段落感 |
| 回彈 | 適中 | 柔和 | 偏硬 | 柔和 | 偏硬 |
| 適用場合 | 打字、遊戲 | 打字、遊戲 | 打字 | 遊戲 | 打字、遊戲 |

我們為什麼選青軸？由於我們希望輸入文字時有明顯的回饋感，因此選擇有著明顯段落感且性價比高的 Outemu 青軸。



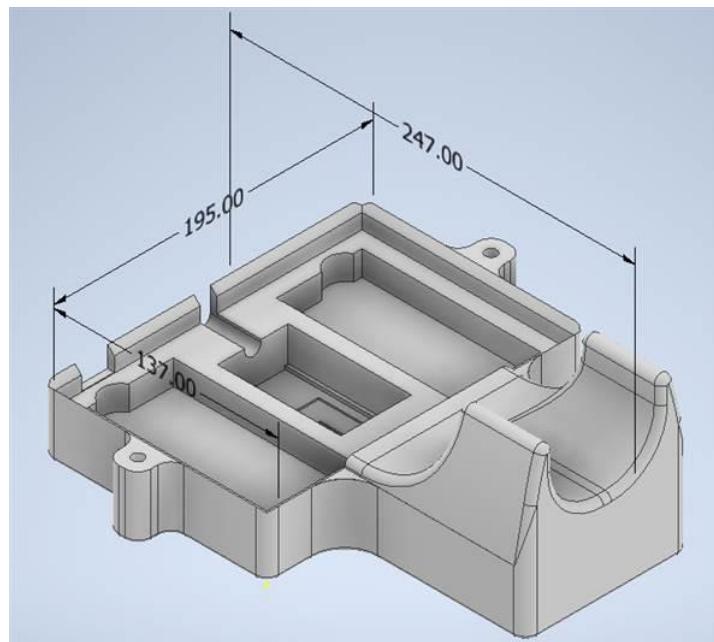
(圖 24)各種不同的鍵軸（分別為黑、紅、黃、綠、青、茶軸）

(出處詳附錄圖片來源 8)

四、外殼及定位版

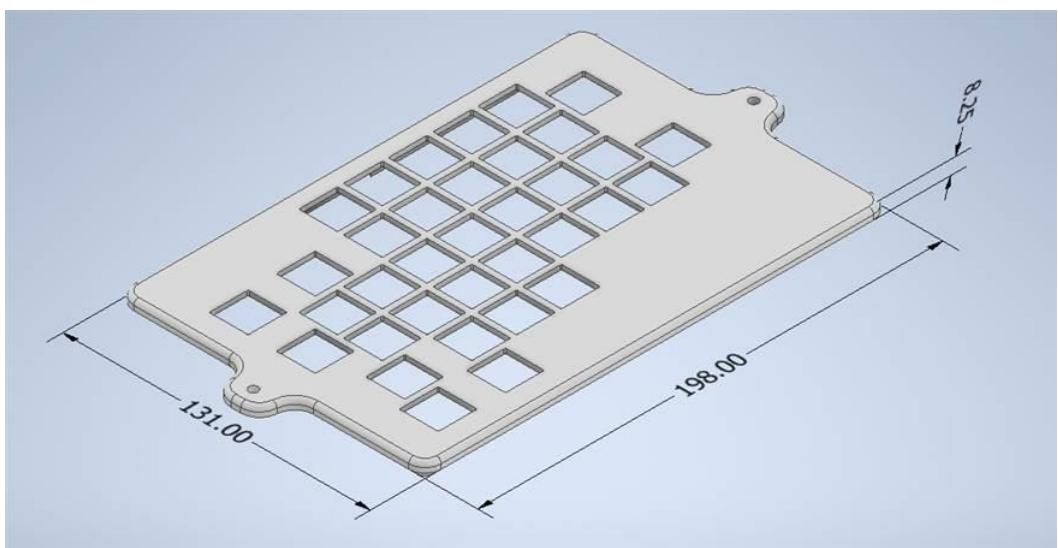
市面上大部分賣的一般鍵盤通常可以分解出外殼以及定位版(兩層)，我們將兩者合而為一，變成在外殼之設計中就帶有定位版的設計。

我們使用 3D 繪圖軟體 Autodesk Inventor 繪製鍵盤之外殼，分為底座以及上蓋兩部分，兩者使用螺絲連接。



(圖 25)鍵盤的「底座」模型。(出處詳附錄圖片來源 1)

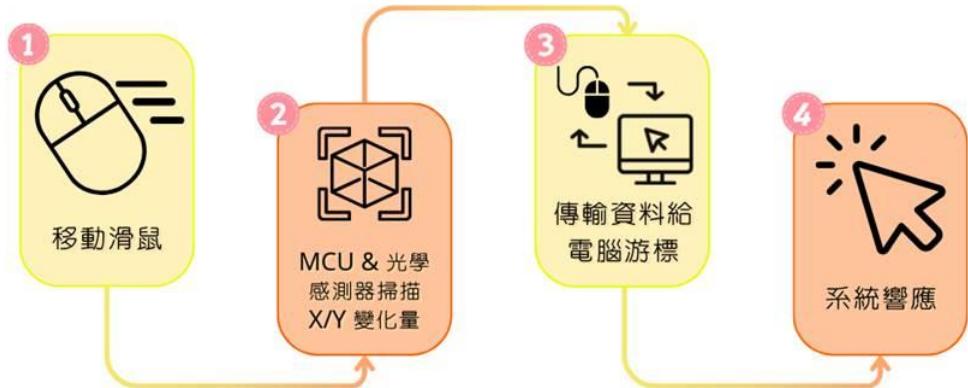
下圖(圖 26)為上蓋模型，設計包含了固定鍵軸的「定位板」之設計。



(圖 26)鍵盤的「上蓋」模型 (出處詳附錄圖片來源 1)

五、游標控制

我們將滑鼠的電路板置於底部，只利用其光學感測器來控制游標。



(圖 27)滑鼠游標流程圖

(出處詳附錄圖片來源 1)

六、整合組裝

經過上述設計後我們已經準備好了所有的零件。接下來，將滑鼠電路板和鍵盤的 PCB 電路板放入 3D 列印的殼中。再來將鍵軸與鍵帽插入上蓋的定位版中，最後，鎖上兩側螺絲就組裝完畢了。



(圖 28)將滑鼠的電路板放入 3D 列印中

(出處詳附錄圖片來源 1)

(圖 29)放入鍵盤 PCB 以及鍵軸

(出處詳附錄圖片來源 1)



(圖 30)裝上鍵帽後，蓋上上蓋就組裝完畢了 (出處詳附錄圖片來源 1)

伍、研究結果

一、游標部分

我們的作品採用直覺化的設計，使用者僅需將手掌自然地放置於按鍵表面，便可透過滑動整個鍵盤裝置來操控螢幕游標。此一操作方式不僅簡單易懂，亦能有效降低學習門檻。



(圖 31)一般滑鼠(左)和我們設計的按鍵(右)對照圖 (出處詳附錄圖片來源 1)

二、鍵盤部分

由於我們的鍵盤是為了單手使用所設計，所以我們縮減了整體大小與按鍵數量，將一般 100% 鍵盤的 104 鍵縮減到 51 個按鍵。因此，許多按鍵需要依靠「層」的概念來搭配使用。

一般的鍵盤按鍵分為兩個虛擬層。第一層是我們看到的最大圖示，按下後就是對應的鍵。例如，按下 1 鍵會輸入 1。第二層需要使用 Fn 鍵，其功能為「暫時切換鍵盤層」，按住會切換到第二層，放開就會返回第一層。

我們的鍵盤一共有 6 個虛擬層，包含 2 個主要層和 4 個附加層。每個主要層都配有两个附加層，分別使用 Fn 和 Shift 來切換。

| | | |
|-------|-------|-------|
| | 一般層 | 注音層 |
| Fn | 附加層 1 | 附加層 3 |
| Shift | 附加層 2 | 附加層 4 |

三、實際按鍵層之操作方式

一般層在不加其他鍵時，按下 1 鍵會輸入「1」。



(圖 32)一般層配置圖(上)
電腦畫面(左下)
按下 1 鍵(右下)

附加層 1 在按下 Fn + 1 鍵時會輸入「F1」。



(圖 33)附加層 1 配置圖(上)
電腦畫面(文件中按下 F1 後
帶往的網頁)(左下)
同時按下 Fn 和 1 鍵(右下)

附加層 2 在按下 Shift + 1 鍵時會輸入「！」。



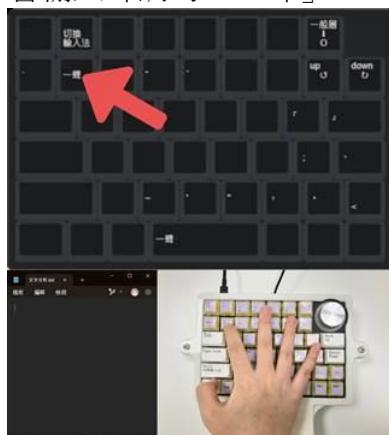
(圖 34)附加層 2 配置圖(上)
電腦畫面(左下)
同時按下 Shift 和 1 鍵(右下)

注音層在不加其他鍵時，按下 6 鍵會輸入「ㄉ」。



(圖 35)注音層配置圖(上)
電腦畫面(左下)
按下 6 鍵(右下)

附加層 3 在按下 Fn + 6 鍵時會輸入聲調的「一聲」。



(圖 36)附加層 3 配置圖(上)
電腦畫面(左下)
同時按下 Fn 和 6 鍵(右下)

附加層 4 在按下 Shift + 6 鍵時會輸入「@」。



(圖 37)附加層 4 配置圖(上)
電腦畫面(左下)
同時按下 Shift 和 6 鍵(右下)

(本頁出處詳附錄圖片來源 1)

四、兩大主要層說明與按鍵配置

(一) 一般層：

為日常使用所設計，包含滑鼠的左鍵、右鍵、數字、26 個英文字母以及各種符號。平常使用時，大部分情況下會停留在這一層。

(二) 注音層：

是專門為繁體中文使用者設計，包含 37 個注音符號、5 種聲調，以及中文輸入時會用到的所有符號。此外，因為注音輸入法需要選字，我們設計了旋鈕選字功能：當需要選字時，順時針轉動旋鈕即可選擇下拉文字，選好後按下 Enter 即可完成輸入。

下圖為完整鍵盤配置圖，上面詳細寫出了一般層和注音層的所有字母與符號的位置。



(圖 38)完整鍵盤配置圖 (出處詳附錄圖片來源 1)

五、特殊快捷鍵說明

(一)語音輸入功能：

當使用者按下 Win + H 鍵時，系統會啟動 Windows 內建的語音輸入功能。此功能需搭配麥克風使用，能夠即時將語音轉換為文字，適用於各種輸入欄位，對於想提升輸入效率的使用者而言，是一項相當實用的輔助工具。

(二)定住游標功能：

由於本裝置是透過滑動整體來控制游標，因此當使用者操作空間受限（例如裝置碰到牆壁或桌面邊緣）時，無法像傳統滑鼠那樣「提起再放下」以重設位置。為了解決此問題，我們使用 AutoHotkey 設計了一組快捷鍵功能：按下 Ctrl + B 時，螢幕上的游標會暫時固定（實際上是暫時中斷滑鼠控制），使用者便可在不影響游標位置的情況下，將裝置移動至較佳位置，再繼續操作。此設計有效解決了本產品相較於一般滑鼠在體積上的劣勢。

六、產品使用教程影片

為了讓使用者更快速地了解我們產品的操作方式，我們製作了數部教學影片，並上傳至 YouTube 平台。使用者可透過掃描裝置側邊的 QR Code，即可連結至教學影片清單。影片內容涵蓋了裝置的基本操作方式、快捷鍵設定方法，以及常見問題排除等詳盡說明，幫助使用者能更有效率地上手並發揮本產品的功能優勢。



(圖 39)上傳至 YouTube 的使用教學影片 (出處詳附錄圖片來源 1)

七、成品操作說明

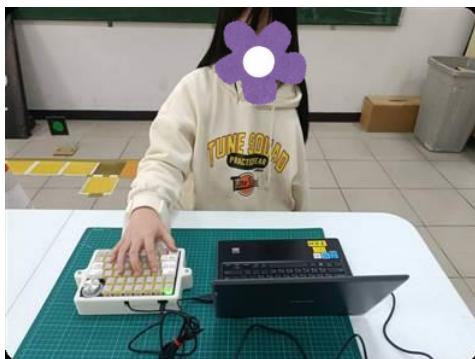
實際操作情況如下圖，右手手腕置於符合人體工學的凹槽中，手掌放在鍵盤上單手打字，靠著凹槽的固定和手掌的摩擦力滑動整個鍵盤就可以控制游標。這樣如此一來解決了平時操作電腦需要頻繁切換滑鼠與鍵盤的問題，以及一般的雙手鍵盤有快捷鍵間距過遠的問題。這樣鍵盤與滑鼠融為一體的設計，讓使用者在單手操作電腦時方便許多。



(圖 40)手腕置於符合人體工學的凹槽



(圖 41)連接於筆電上單手打字



(圖 42)使用鍵盤功能



(圖 43)使用滑鼠功能(滾輪)

另外，如果忘記了鍵盤上每個按鍵的功能(配置)，可以掃描成品右側的 QR Code (如下圖)，進入附有鍵盤配置圖(如圖 38)的網頁。



(圖 44)成品右側之 QR Code

(本頁出處詳附錄圖片來源 1)

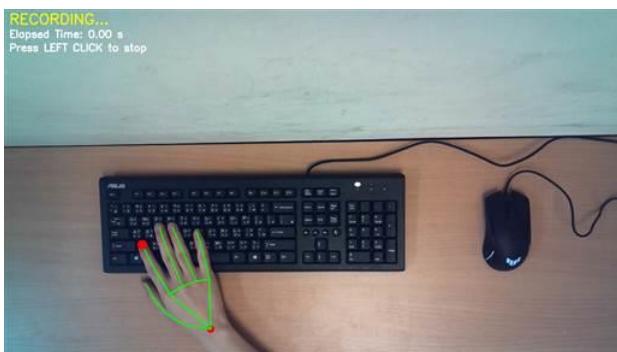
八、實驗一：不同設備之手腕移動距離差異

以下是使用 Python 結合 MediaPipe 骨架估計函式庫，測試三種不同設備之手腕移動距離差異的結果。

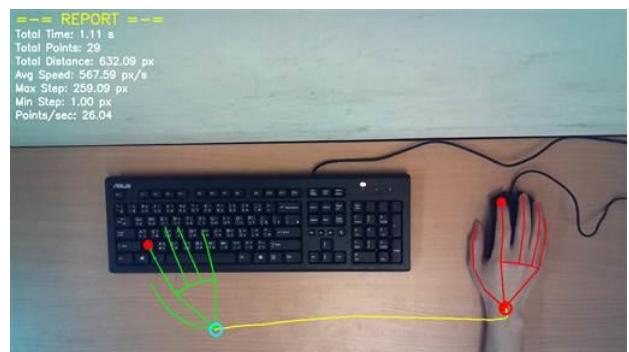
將 Z、1、0、M 鍵設為起點，測試不同距離按鍵的移動差異。終點皆為滑鼠左鍵。

當按下起點鍵時，開始紀錄時間並記錄手腕軌跡（作為距離依據）；當按下滑鼠左鍵，則結束該次實驗。

(一) [全尺寸鍵盤] 到 [滑鼠] 的比較圖



(圖 45) z 鍵起點



(圖 46) z 鍵終點



(圖 47)1 起點

(圖 48)1 終點

(圖 49)0 起點

(圖 50)0 終點

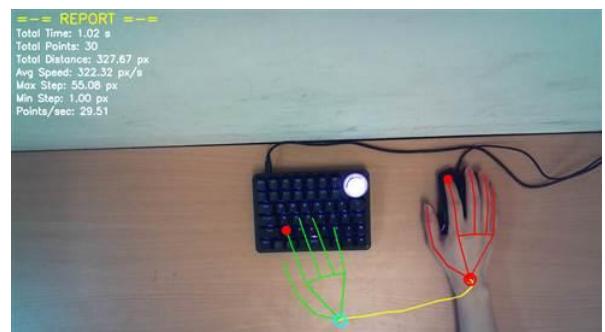
(圖 51)m 起點

(圖 52)m 終點

(二) [單手鍵盤] 到 [滑鼠] 的比較圖



(圖 53) z 鍵起點



(圖 54) z 鍵終點



(圖 55)1 起點

(圖 56)1 終點

(圖 57)0 起點

(圖 58)0 終點

(圖 59)m 起點

(圖 60)m 終點

(本頁出處詳附錄圖片來源 1)

(三) [我們的作品] 到 [滑鼠] 的比較圖



(圖 61) z 鍵起點



(圖 62) z 鍵終點



(圖 63)1 起點



(圖 64)1 終點



(圖 65)0 起點



(圖 66)0 終點



(圖 67)m 起點



(圖 68)m 終點

(本頁出處詳附錄圖片來源 1)

(表 3)綜合時間比較

| 時間比較 | 全尺寸鍵盤+滑鼠 | 單手鍵盤+滑鼠 | 我們的作品 |
|------|----------|---------|--------|
| z 鍵 | 1.11 秒 | 1.02 秒 | 0.31 秒 |
| 1 鍵 | 1.05 秒 | 1.41 秒 | 0.29 秒 |
| 0 鍵 | 0.96 秒 | 1.09 秒 | 0.27 秒 |
| m 鍵 | 1.11 秒 | 0.96 秒 | 0.28 秒 |
| 平均值 | 1.06 秒 | 1.12 秒 | 0.29 秒 |

(表 4)綜合手腕距離比較

| 距離比較 | 全尺寸鍵盤+滑鼠 | 單手鍵盤+滑鼠 | 我們的作品 |
|------|----------|----------|---------|
| z 鍵 | 50 cm | 26 cm | 0.9 cm |
| 1 鍵 | 54 cm | 31 cm | 1.2 cm |
| 0 鍵 | 41 cm | 25 cm | 0.8 cm |
| m 鍵 | 38 cm | 25 cm | 1.2 cm |
| 平均值 | 45.75 cm | 26.75 cm | 1.03 cm |

從(表 3)和(表 4)中可以看出，相較於其他設備，我們的作品在從鍵盤按鍵移動至滑鼠按鍵的時間與距離上皆顯著縮短。造成此差異的原因在於，其他設備需在鍵盤與滑鼠之間來回切換，而我們的作品則將滑鼠按鍵整合設計於鍵盤之中。此設計有助於減少因頻繁切換所造成的手部疲勞。

九、實驗二：比較實際操作上的數值差異

接下來，我們繼續使用 Python 結合 MediaPipe 骨架估計函式庫比較了單手操作情況下傳統鍵盤和我們的作品在實際操作中的數值差異。

本實驗以知名遊戲《Minecraft》進行，在招募數位自願者後，將每組實驗分為兩部分：一為使用全尺寸鍵盤與滑鼠，以單手完成指定任務；另一則為使用我們的裝置，同樣以單手完成任務。實驗開始時，系統會開始記錄手腕軌跡（作為距離依據）與手部移動次數（用於計算平均值）。當完成指定任務後，即結束該次實驗。



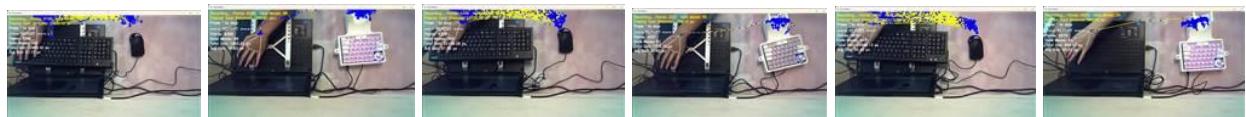
(圖 69) 實驗螢幕截圖



(圖 70) 第 1 組 全尺寸鍵盤+滑鼠 a



(圖 71) 第 1 組 我們的作品 b



(圖 72) 第 3 組 a (圖 73) 第 3 組 b (圖 74) 第 4 組 a (圖 75) 第 4 組 b (圖 76) 第 5 組 a (圖 77) 第 5 組 b

(本頁出處詳附錄圖片來源 1)



(圖 78)第 10 組 全尺寸鍵盤+滑鼠 a



(圖 79)第 10 組 我們的作品 b



(圖 80)第 6 組 a (圖 81)第 6 組 b (圖 82)第 7 組 a (圖 83)第 7 組 b (圖 84)第 8 組 a (圖 85)第 8 組 b

(本頁出處詳附錄圖片來源 1)

(表 5)[全尺寸鍵盤+滑鼠] 的測試數據及平均值

| Tr | 項目 | # 移動次數 | # 總距離 | # 平均移動距離 |
|----|-----|--------|-----------|----------|
| | 第一項 | 416 | 22769.74 | 54.73 |
| | 第二項 | 603 | 28746.36 | 47.67 |
| | 第三項 | 500 | 33866.42 | 67.73 |
| | 第四項 | 384 | 22138.08 | 57.65 |
| | 第五項 | 414 | 22962.13 | 55.46 |
| | 第六項 | 345 | 18891.24 | 54.76 |
| | 第七項 | 365 | 17468.02 | 47.86 |
| | 第八項 | 357 | 17085.23 | 47.86 |
| | 第九項 | 342 | 16185.67 | 47.33 |
| | 第十項 | 394 | 20952.59 | 53.1 |
| | 平均 | 412 | 22106.548 | 53.415 |

(表 6)[我們的作品] 的測試數據及平均值

| Tr | 項目 | # 移動次數 | # 總距離 | # 平均移動距離 |
|----|-----|--------|----------|----------|
| | 第一項 | 96 | 2122.53 | 22.11 |
| | 第二項 | 67 | 1827.13 | 27.27 |
| | 第三項 | 88 | 1665.61 | 18.93 |
| | 第四項 | 82 | 1756.1 | 21.42 |
| | 第五項 | 24 | 965.92 | 40.25 |
| | 第六項 | 28 | 914.06 | 32.64 |
| | 第七項 | 36 | 927.47 | 25.76 |
| | 第八項 | 37 | 796.3 | 21.52 |
| | 第九項 | 33 | 1184.54 | 35.9 |
| | 第十項 | 79 | 1390.65 | 17.6 |
| | 平均 | 65.2 | 1364.139 | 26.34 |

從(表 5)和(表 6)中可以看出，相較於全尺寸鍵盤與滑鼠的配置，我們的作品在手腕移動次數與移動距離上皆顯著縮短。造成此差異的原因在於，傳統設備需在鍵盤與滑鼠之間頻繁切換，而本裝置將滑鼠按鍵整合於鍵盤之中，使用者僅需滑動裝置本體即可控制游標。此設計有助於減少因反覆切換所造成的手部疲勞。

這兩項實驗我們以不同的角度觀察了我們的作品和其他輸入設備手腕移動距離上的差異，可以發現我們的作品可以有效減少手腕的移動距離和次數，間接減緩了使用時對手腕的疲勞。透過鍵鼠融合的設計，我們的產品不僅適合單手使用者，也能在特定情境（如遊戲）提供更高的操作靈活性。

十、各種鍵盤特性總比較

在設計本專題的單手鍵盤時，我們研究了市場上各類鍵盤，發現傳統鍵盤多數是以雙手操作為前提，對單手使用者並不友善。因此，我們整理了不同類型鍵盤的特點，並與我們的設計進行比較：

(表 7)各種鍵盤特性比較表 (出處詳附錄圖片來源 9~12)

| 鍵盤種類
特性 | 全尺寸鍵盤 | 迷你鍵盤 | 單手鍵盤 | 我們的鍵盤
(含滑鼠在內部) |
|------------|---------|---------|---------|-------------------|
| 按鍵數量 | 104 | 61 | 20-40 | 51 |
| 使用方式 | 雙手 | 雙手/單手 | 單手 | 單手 |
| 攜帶性 | 低 | 中等 | 高 | 中等 |
| 學習成本 | 低 | 低-中 | 中-高 | 高 |
| 打字效率 | 因學習程度而異 | 因學習程度而異 | 因學習程度而異 | 因學習程度而異 |
| 遊戲適應性 | 高 | 高 | 依遊戲類型而異 | 依遊戲類型而異 |
| 可自訂性 | 一般(鍵帽) | 高(可程式化) | 高(多功能鍵) | 高(全自訂) |
| 適合對象 | 一般使用者 | 遊戲、簡單工作 | 特殊需求、遊戲 | 需單手操作者 |
| 鍵鼠狀態 | 鍵鼠分離 | 鍵鼠分離 | 鍵鼠分離 | 鍵鼠合併 |

從表格中可以看出，我們的設計在單手操作便利性、鍵鼠整合度與可客製化程度方面具有明顯優勢。相較於傳統全尺寸鍵盤與迷你鍵盤，我們的單手鍵盤減少了不必要的按鍵，並透過多層鍵位切換提升功能完整性。

陸、討論

1. 光學感測器無法感應滑動

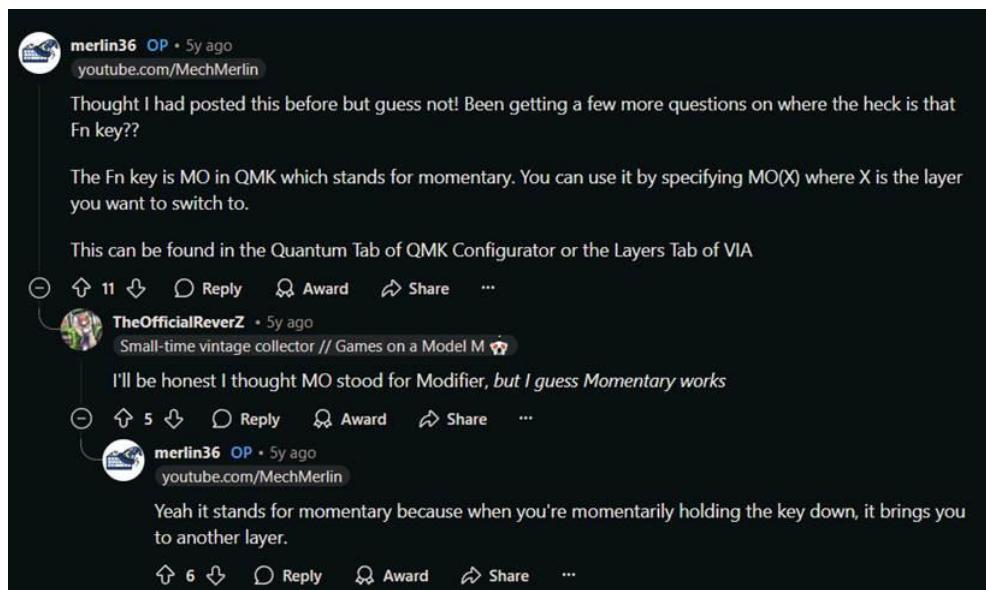
在組裝各部分的零件時，我們發現控制游標移動的光學感測器經常無法感應滑動。起初以為是感測器故障，但在取出感測器並手動在桌面上滑動測試後，才發現問題出在感測器與桌面的距離。經過多次嘗試，並輔以熱熔膠調整，最終才找到適當的位置。

2. 找不到 Fn 的對應鍵碼

在鍵盤韌體程式中，每個鍵都有專屬的鍵碼。例如，空白鍵（space）的鍵碼是 KC_SPC。然而，當我編寫到 Fn 鍵時，卻在鍵碼表上找不到與之對應的鍵碼。經過一番搜尋後，我終於在 Reddit 上找到了一則相關的貼文：



(圖 86)在 QMK 中 Fn 的鍵碼是什麼? (出處詳附錄圖片來源 13)

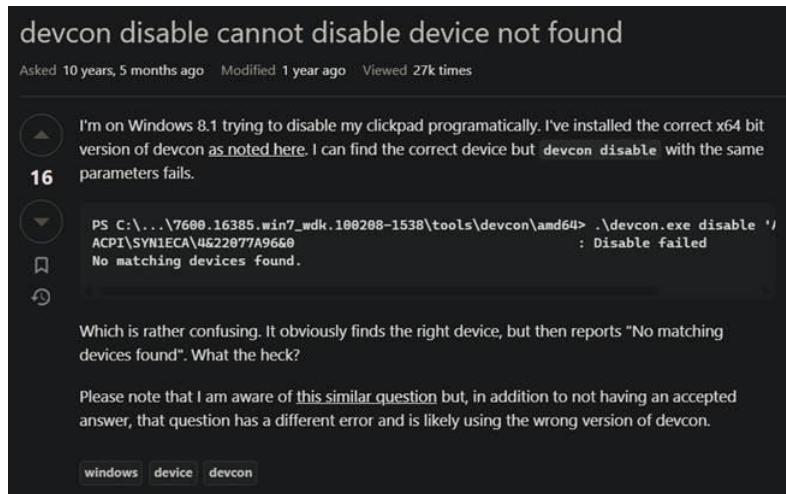


(圖 87)貼文留言區的解釋 (出處詳附錄圖片來源 13)

原來，Fn 鍵並不是一個文字鍵，而是一個功能鍵。它的作用是「暫時切換鍵盤到第 n 層」，其鍵碼為 MO(n)，其中 n 代表要切換到的層。

3.定住滑鼠功能失常

在製作「定住螢幕游標」功能的最後階段時，雖然程式碼經過檢查並未發現任何問題，但按下對應快捷鍵時，螢幕游標卻完全沒有反應，甚至未顯示任何錯誤訊息。經過一番搜尋後，我們在 Stack Overflow 上找到了一則相關的討論貼文：



devcon disable cannot disable device not found

Asked 10 years, 5 months ago Modified 1 year ago Viewed 27k times

16

I'm on Windows 8.1 trying to disable my clickpad programmatically. I've installed the correct x64 bit version of devcon as noted here. I can find the correct device but `devcon disable` with the same parameters fails.

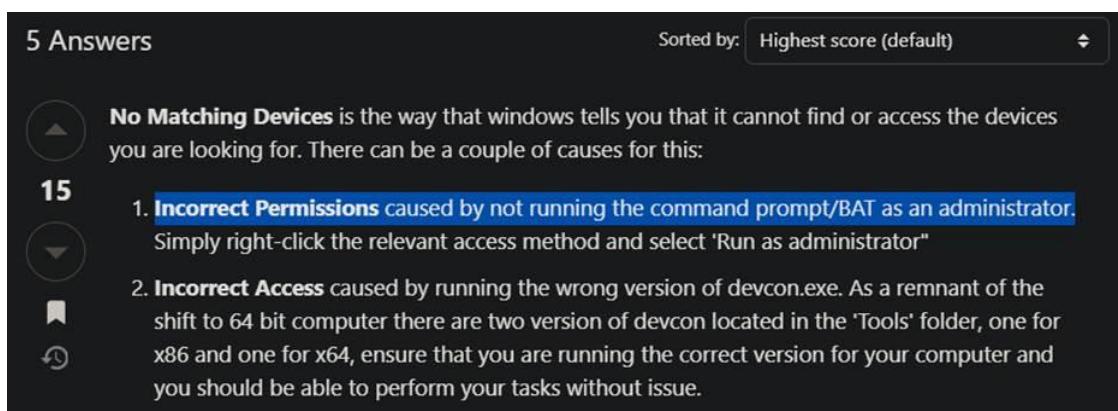
```
PS C:\...\7600.16385.win7_wdk.100208-1538\tools\devcon\amd64> ./devcon.exe disable /ACPI\SYN1ECA\4622077A9660
: Disable failed
No matching devices found.
```

Which is rather confusing. It obviously finds the right device, but then reports "No matching devices found". What the heck?

Please note that I am aware of this similar question but, in addition to not having an accepted answer, that question has a different error and is likely using the wrong version of devcon.

windows device devcon

(圖 88)裝置無法正常斷開 (出處詳附錄圖片來源 14)



5 Answers

Sorted by: Highest score (default)

No Matching Devices is the way that windows tells you that it cannot find or access the devices you are looking for. There can be a couple of causes for this:

1. **Incorrect Permissions** caused by not running the command prompt/BAT as an administrator. Simply right-click the relevant access method and select 'Run as administrator'
2. **Incorrect Access** caused by running the wrong version of devcon.exe. As a remnant of the shift to 64 bit computer there are two version of devcon located in the 'Tools' folder, one for x86 and one for x64, ensure that you are running the correct version for your computer and you should be able to perform your tasks without issue.

(圖 89)貼文留言區的解釋 (出處詳附錄圖片來源 14)

我們發現，雖然程式本身已具備斷開與重新連接輸入裝置的能力，但由於系統權限的限制，預設情況下無法成功執行相關操作。因此，在執行這個程式時，要選擇「以系統管理員身分執行」，以賦予程式足夠的權限來控制裝置的連線狀態。唯有如此，才能順利實現我們所設計的「定住螢幕游標」功能（實際上為暫時中斷滑鼠連線），確保使用者在重置裝置位置時，游標不會意外移動。

柒、結論

一開始，我們發現許多使用者因肢體障礙或僅能單手操作，在使用傳統鍵盤與滑鼠時經常面臨不便。為此，我們決定開發一款結合鍵盤與滑鼠功能的單手操作裝置，以提升操作效率與便利性。為了實現這個目標，我們投入大量時間進行市場調查與技術研究，確保產品設計能真正滿足目標族群的需求。

在硬體方面，我們學習並應用了 PCB 設計、鍵軸配置、3D 建模與列印技術，打造符合人體工學的裝置結構；在軟體方面，則深入研究 QMK 韌體，並根據需求客製化鍵位與功能，使其更適合單手操作。

專題過程中，我們也遇到不少挑戰，例如如何整合鍵盤與滑鼠功能、設計合適的鍵位配置，以及在成本與品質間取得平衡。經過我們不斷測試與改進，才逐步克服這些困難。

這段經歷不僅加深了我們對電子與資訊技術的理解，也讓我們深刻體會團隊合作與解決問題的重要性。每位成員發揮專長、互相支援，才能順利完成這項專題。

本專題成功實現了單手鍵鼠整合的設計，為肢體障礙者與單手操作者提供了更高效、便利的電腦操作體驗。未來，我們計畫進一步優化產品，在硬體方面探索更輕巧耐用的材料，改善鍵帽與佈局設計，提升人體工學與使用舒適度；在軟體方面，則持續優化 QMK 韌體，加入更多自訂功能，導入如 AI 學習使用習慣等智慧化功能，進一步提升使用效率。最終，我們希望此產品能推廣至市場，與相關機構合作，真正實現科技輔助生活的初衷。



(圖 90)製作過程 (出處詳附錄圖片來源 1)

捌、參考資料及其他

一、參考資料

1. 身心障礙人數統計，(2025/02/10)取自:<https://tinyurl.com/23uant4p>
2. 自製單手鍵盤參考資料，(2025/02/15)取自 <https://tinyurl.com/2yjz2sk4>
3. 按鍵字母排列資料參考，(2025/02/15)取自 <https://tinyurl.com/2cjdglay>
4. 自製機械鍵盤參考資料，(2025/02/15)取自 <https://tinyurl.com/2arjbyde>
5. 客製化鍵盤參考資料，(2025/02/10)取自 <https://www.youtube.com/watch?v=xcXgFiaGPCQ>
6. 鍵帽挑選參考資料，(2025/02/12)取自 https://www.youtube.com/watch?v=Bgs_pk-MWRk&t=707s
7. 套件挑選參考資料，(2025/02/12)取自 <https://www.youtube.com/watch?v=4TgCPMDHGNQ>
8. 自訂鍵盤快捷鍵軟體(尚未使用)(2025/02/16)<https://www.autohotkey.com/>
9. 客製化鍵盤參考資料，(2025/02/16)取自 <https://tinyurl.com/275r2bkk>
10. 鍵盤佈局編輯工具，(2025/02/17)來自 <https://www.keyboard-layout-editor.com/#/>
11. 編輯鍵盤佈局 (Layout)(2025/02/14)<https://blog.ziteh.dev/posts/diyqmkkeyboard-kle/>
12. 電路設計相關資料，(2025/02/17)來自 <https://www.youtube.com/watch?v=yPJ-4nEXykE>
13. 電路設計軟體(2025/02/17)<https://easyeda.com/>
14. 機械鍵盤的基本電路參考，(2025/02/18)來自 <https://ergotaiwan.tw/self-keyboard-basic-1/>
15. 鍵盤 PCB 繪製方式 1，(2025/02/18)來自 <https://www.youtube.com/watch?v=DOluUYmqIs4>
16. 鍵盤 PCB 繪製方式 2，(2025/02/18)來自 <https://blog.ziteh.dev/posts/diyqmkkeyboard-pcb-layout-sch/>
17. 鍵盤 PCB 繪製方式 3，(2025/02/18)來自 <https://www.youtube.com/watch?v=zhCffPA7798>
18. 鍵盤 PCB 繪製方式 4，(2025/02/18)來自 <https://tinyurl.com/258d6kbt>
19. 鍵帽規格尺寸對照表，(2025/02/19)來自 <https://tinyurl.com/24aos8hy>
20. 鍵盤 PCB 線路佈置工具，(2025/02/20)來自 <https://kb.xyz.is/>
21. ATmega32u4 周邊電路參考，(2025/02/20)來自 <https://tinyurl.com/yyqb5pmf>
22. 定位版生成工具網站，(2025/02/20)來自 <https://eswai.github.io/plategen/plategen.html>
23. 葉雅婷. 單臂截肢者之輸入輔具設計研究. 2001 年
24. 余彥儒. 適用單手使用之智慧型手機觸控鍵盤. 2013 年
25. 陳柏州. 滑鼠設計與操作之評估研究. 2001 年
26. 陳皇任. 肢障者改良式指標器按鍵之研發及效能評估系統之建立. 2005 年

27. 鍵盤數據照片，作者 Cangjie6，取自 <https://tinyurl.com/24x8lx9z>
28. 電子學 作者:葉倍宏 出版日期:2021/07/10
29. 基本電學 作者:黃仲宇 出版社:台科大圖書 出版日期:2020/04/01
30. 精解 KiCad：從電腦輔助電路設計到 PCB 實踐 作者：陳茂璋, 劉進德 出版社：台科大圖書 出版日期：2023/11/14
31. Autodesk Inventor 2024 基礎特訓教材 作者：黃穎豐, 陳明鈺 出版社：全華圖書 出版日期：2024/09/27

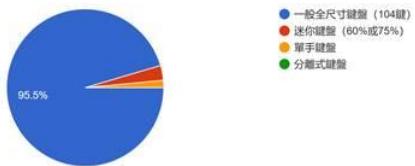
二、圖片來源

1. 本圖片由作者親自拍攝或製作
2. 本圖片引用自 Github，網址 <https://github.com/kicad>
3. 本圖片引用自 Brandlogos，網址 <https://brandlogos.net/autodesk-inventor-logo-108187.html>
4. 本圖片引用自 QMK MSYS，網址 <https://msys.qmk.fm/>
5. 本圖片引用自維基百科，網址 <https://tinyurl.com/22mxqzdu>
6. 本圖片引用自 Autohotkey 官方網站，網址 <https://tinyurl.com/283tzuk3>
7. 本圖片引用自 Reddit，網址 <https://tinyurl.com/22b22y8l>
8. 本圖片引用自 ikanchai，網址 <http://m.ikanchai.com/pcarticle/369290>
9. 本圖片引用自維基百科，網址 <https://tinyurl.com/26zvmd3z>
10. 本圖片引用自 MOMO，網址 <https://tinyurl.com/2cf7y4w8>
11. 本圖片引用自 Coupang，網址 <https://tinyurl.com/2b9x8685>
12. 本圖片引用自 Logitech，網址 <https://tinyurl.com/294ap5j7>
13. 本圖片引用自 Reddit，網址 <https://tinyurl.com/23jvkm4c>
14. 本圖片引用自 Stackoverflow，網址 <https://tinyurl.com/24btqz4j>

三、問卷調查

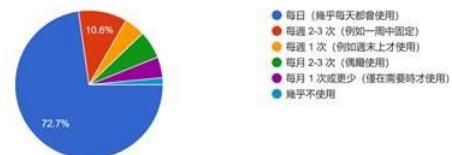
您目前主要使用的鍵盤類型為？

66 例回應



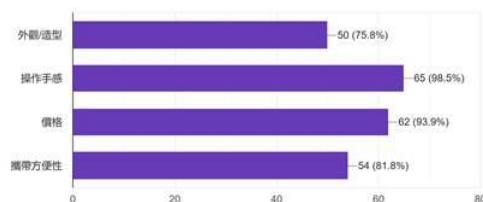
您平均多久使用一次鍵盤與滑鼠？

66 例回應



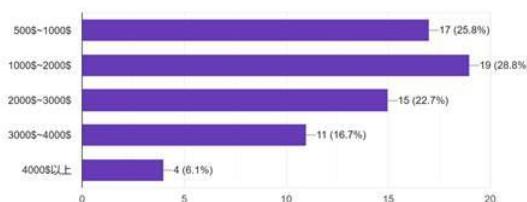
您選擇鍵盤產品時最重視的功能是？

66 例回應



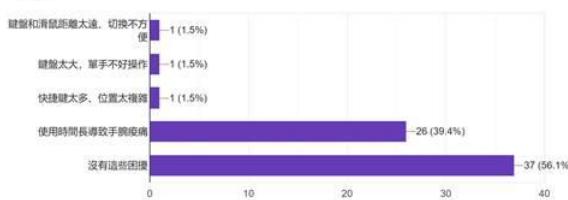
您認為我們的產品售價應為多少？

66 例回應



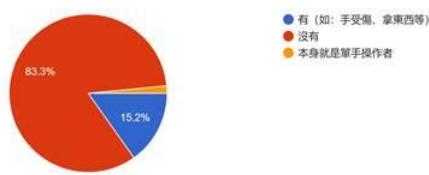
您是否曾經遇過以下哪種使用上的困擾？ (可複選)

66 例回應



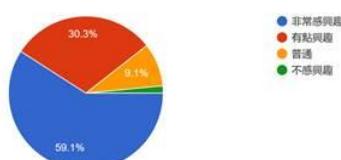
您是否有過只能單手使用鍵盤的經驗？

66 例回應



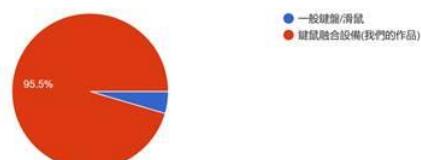
如果有一個裝置可以讓您用「一隻手同時操作鍵盤與滑鼠」，您會感興趣嗎？

66 例回應



經過這次測試，您認為在「單手」情況下，哪種設備更為方便的玩遊戲？

66 例回應



如果您身為一個單手操作者，會願意購買我們的設備嗎？

66 例回應



【評語】052515

本研究結合鍵盤與滑鼠於一體的單手操作裝置，透過自製PCB、客製化鍵帽鍵軸、3D列印外殼與QMK韌體，實現針對單手或身障者使用情境之創新人機介面設計。問題設定明確，針對單手或身障者操作困難提出實際可行解法，具備社會價值，值得肯定。建議可多注重該裝置實用性、便利性等的實驗設計與結果論述，落實科學實驗的精神。

作品海報



單手操作無障礙

鍵盤與滑鼠的整合設計研究

摘要

本研究針對肢體障礙與單手使用者設計一款結合鍵盤與滑鼠功能的單手操作裝置。並具備語音輸入游標固定等多項功能，實現單手同時輸入文字與游標控制。本裝置可有效減少手腕移動、提升效率並降低疲勞，具備通用性與實用價值。

壹、研究動機

根據統計，在總人口皆約2300萬的情況下，身心障礙者從2001年的75萬增至2024年的120萬，其中「肢體障礙」占比最高，從30.6萬增至39.67萬。這使傳統鍵盤與滑鼠的配置已無法滿足其操作需求，影響日常生活與工作效率。（數據由內政部統計）



2001~2024身心障礙的總折線圖(單位:萬)(出處詳附錄圖片來源1)

貳、研究目的

我們的目標是解決「單手操作電腦」不便的問題，特別是針對有肢體障礙或需要單手輸入的使用者，讓使用者能夠用單手「同時」進行「操作鍵盤」與「控制滑鼠」，解決在單手使用傳統輸入裝置時需頻繁切換鍵盤與滑鼠的問題，提升操作效率和使用體驗。這樣的設計不僅適合身心障礙者，也能讓一般使用者在特定情境下更方便地使用電腦。

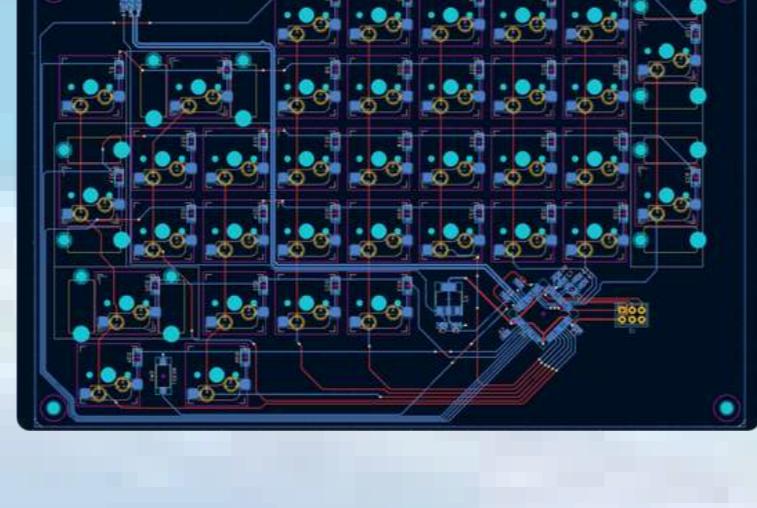
參、研究設備及器材

一、KiCad 7.0

一款免費開源的電子設計自動化軟體，有提供電路設計、PCB佈局及產生製造文件等功能。



KiCad 7.0
logo(出處
詳附錄圖片
來源2)



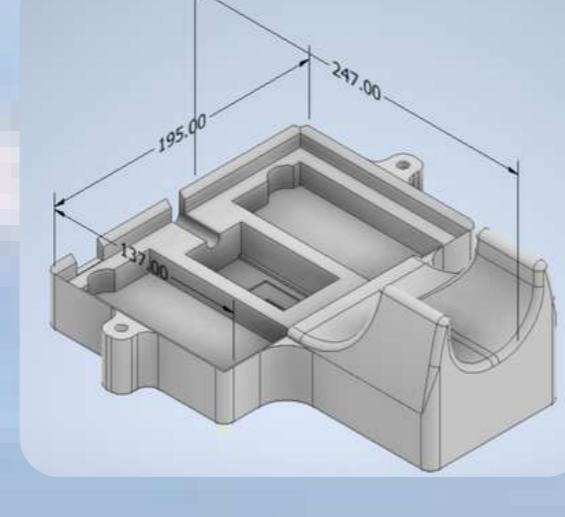
PCB
設計界面
(出處詳附
錄圖片來
源1)

二、Autodesk Inventor

一款專業3D設計與機械工程軟體，提供強大的建模工具，適用於產品設計與工程開發。



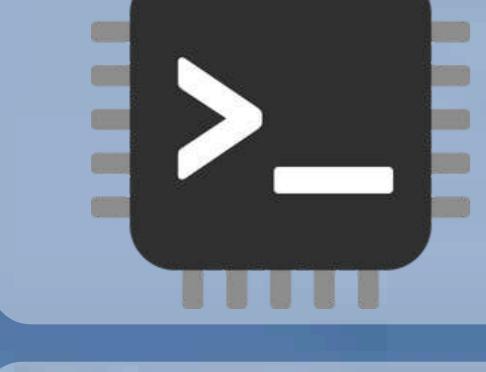
Autodesk
Inventor
logo(出處詳
附錄圖片來
源3)



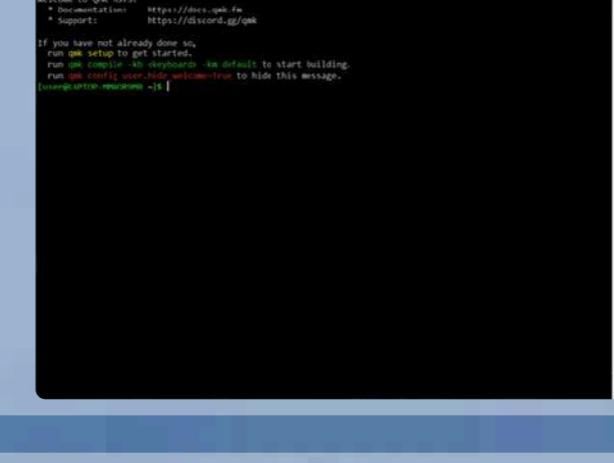
模型設計界面
(出處詳附
錄圖片來
源1)

三、QMK MSYS

基於MSYS2的鍵盤韌體程式開發工具，提供簡單方式編譯、修改與燒錄韌體程式。



QMK MSYS
logo(出處詳
附錄圖片來
源4)



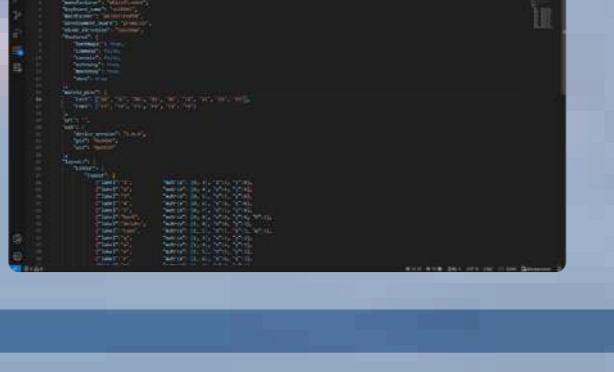
操作界面(出處詳
附錄圖片來
源1)

四、Visual Studio Code

由微軟開發的免費、輕量級且可擴展的程式碼編輯器，軟體輕量但功能十分強大，成為我們編寫程式的最佳選擇。



Visual Studio
Code logo(出處詳
附錄圖片來
源5)



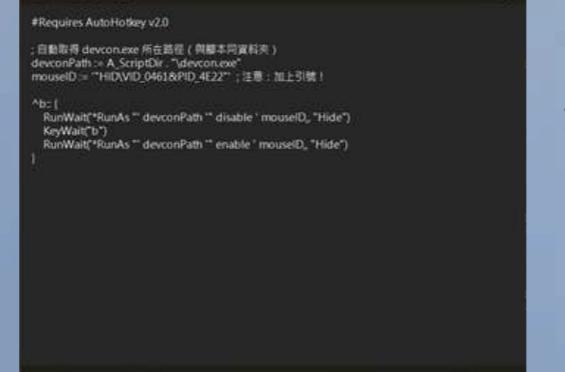
編輯介面(出處詳
附錄圖片來
源1)

五、AutoHotkey

是一款免費自動化工具，能用腳本控制鍵盤滑鼠，提高操作效率與自動化工作流程。



AutoHotkey
logo(出處詳
附錄圖片來
源6)



編輯介面(出處詳
附錄圖片來
源1)

肆、研究過程或方法

一、單手鍵盤製作

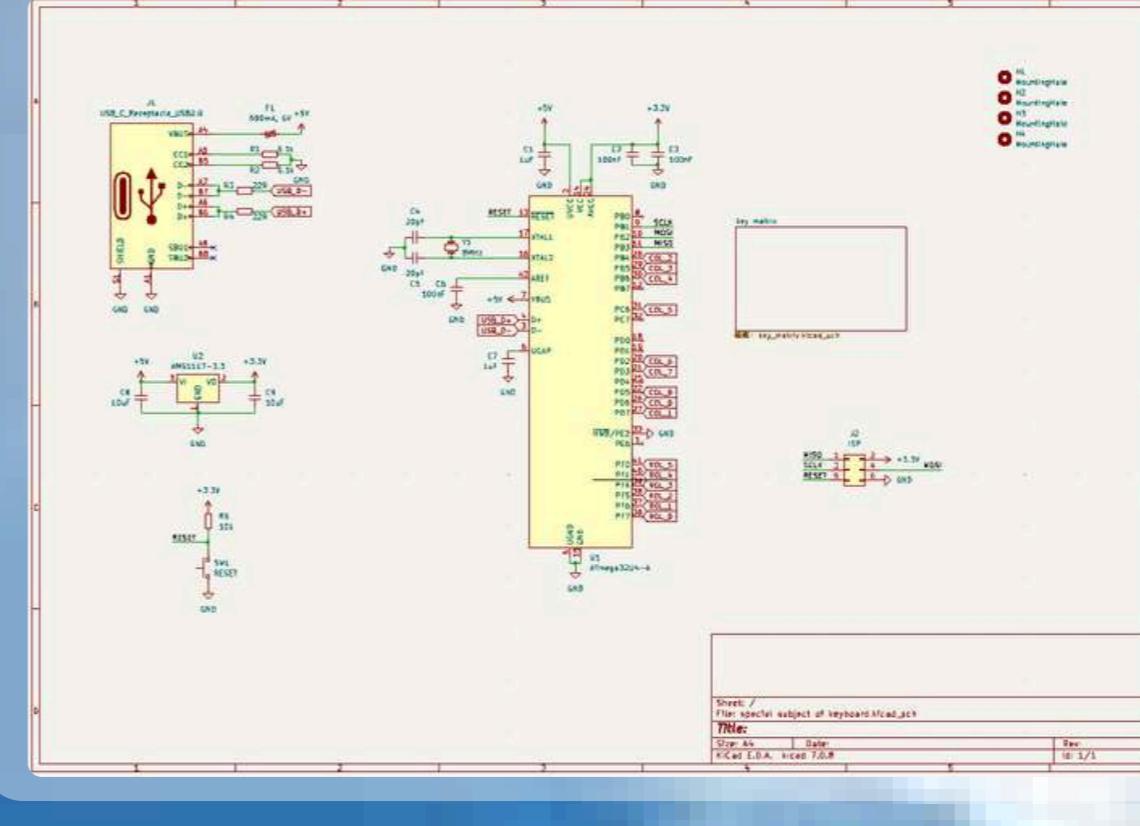


鍵盤輸入流程圖(出處
詳附錄圖片來
源1)

(一)PCB電路板

1.繪製電路藍圖

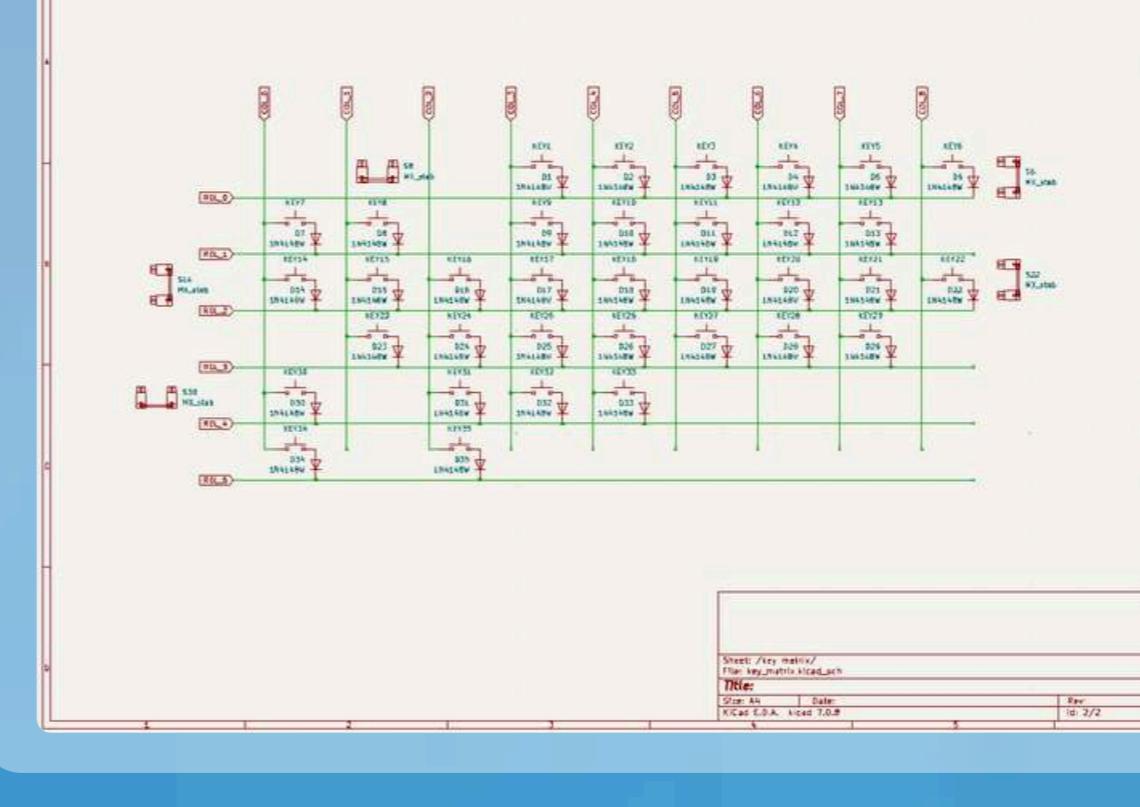
在繪製鍵盤電路藍圖前，我們需選擇鍵盤的微控制器MCU，我們選擇的是定易焊接、且可擴充藍牙的ATmega32U4，並搭配TYPE-C母座、鍵矩陣及周邊電路繪製。



KiCad中繪製的鍵盤電路藍圖
(出處詳附錄圖片來源1)

2.繪製鍵矩陣 (Key Matrix)

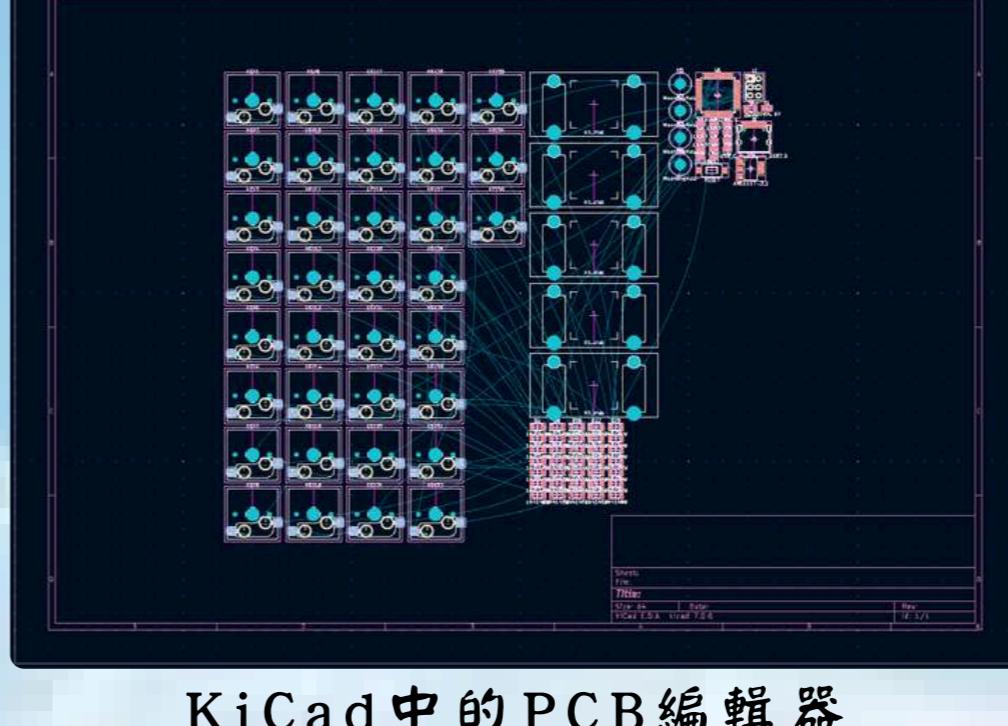
由於按鍵數超過微控制器針腳，因此採鍵矩陣設計，透過行列掃描檢測電流，以判斷按鍵是否被按下。



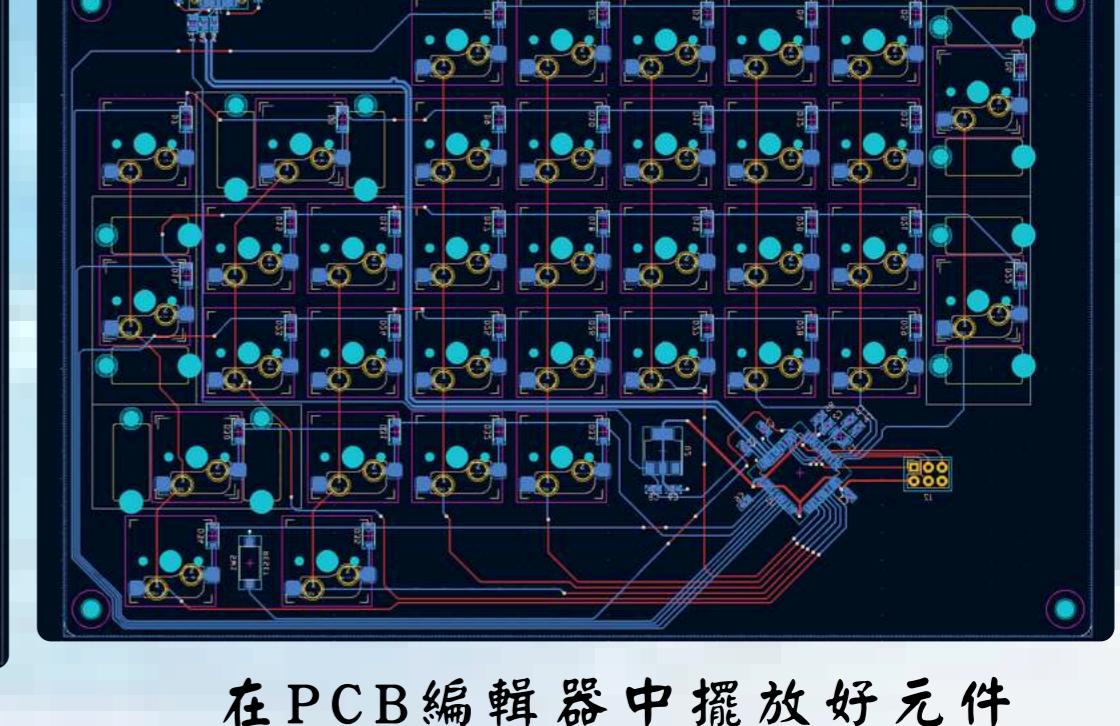
鍵矩陣 (Key Matrix) 藍
圖(出處詳附錄圖片來源1)

3.佈置線路與擺放元件

完成元件選擇、佈置與連接後，接下來將電路圖轉為PCB佈局，規劃元件與走線。繪製電路板需注意走線要避免過熱問題與訊號互相干擾。



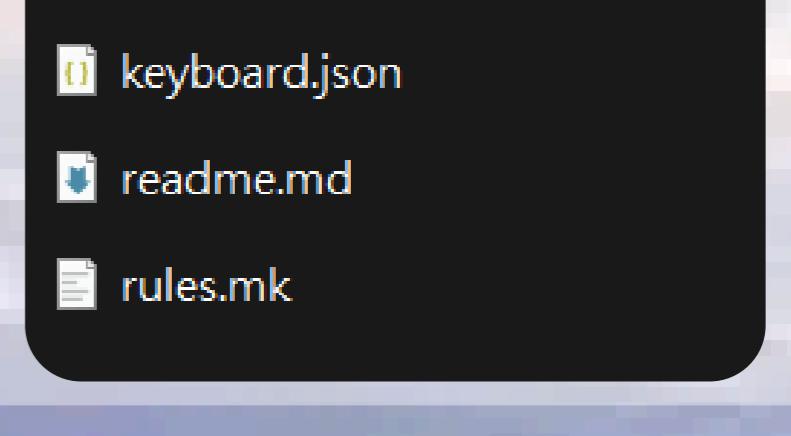
KiCad中的PCB編輯器
(出處詳附錄圖片來源1)



在PCB編輯器中擺放好元件
(出處詳附錄圖片來源1)

4.QMK韌體製作及燒錄 (鍵盤程式)

在QMK MSYS中建立所需文件，選擇ATmega32U4作為微控制器，QMK資料夾中會產生所需設定檔。需修改keyboard.json和keymap.c設定檔以滿足設定檔需求。

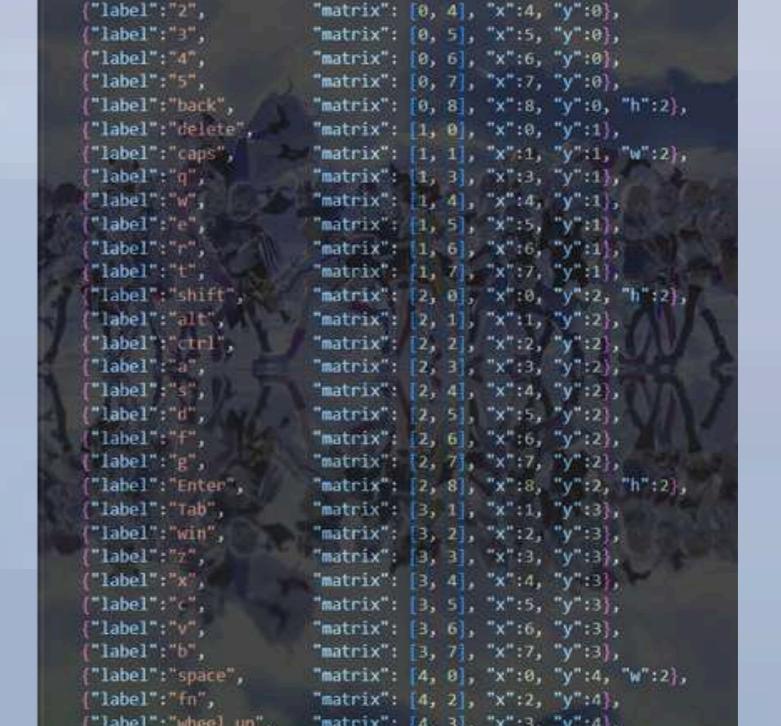


放置QMK韌體程式的資料夾
(出處詳附錄圖片來源1)

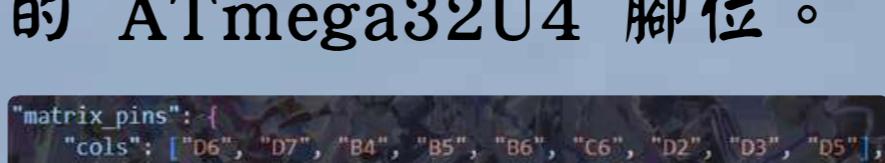
keyboard.json 的 layout

部分，label 要填入每個按鍵的「名稱」；matrix 要填入按鍵在鍵矩陣中的位置 [row, col]；X、Y 則要填入每個按鍵在現實的實際位置。

keyboard.json 檔中的 layout 區
(出處詳附錄圖片來源1)



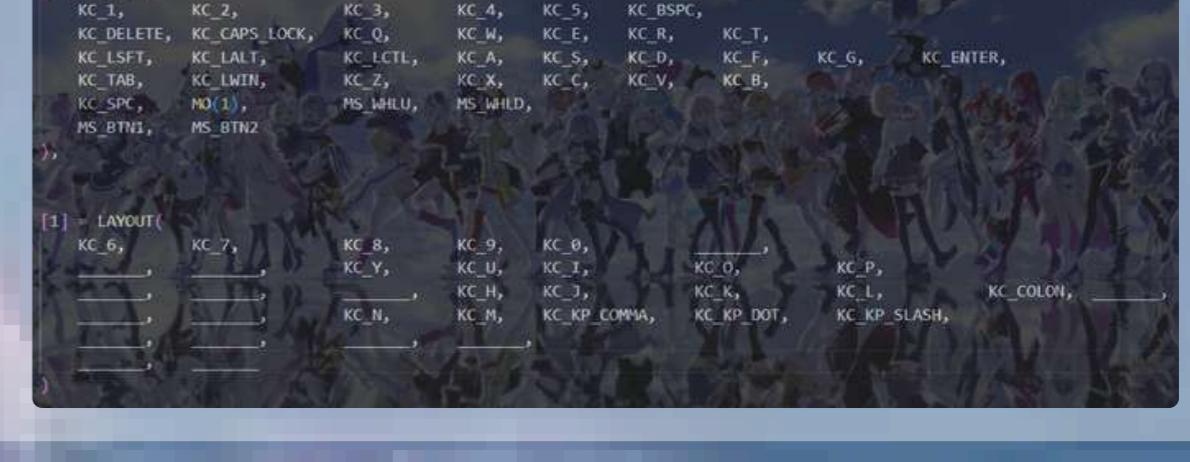
matrix pins 部分則要填入鍵矩陣中的每條 row 和 col 指定的 ATmega32U4 腳位。



keyboard.json 檔中的 matrix
pins 區(出處詳附錄圖片來源1)

keymap.c 中要寫入所有按鍵對應的功能鍵碼，例如，Space鍵對應的鍵碼為 KC_SPC。

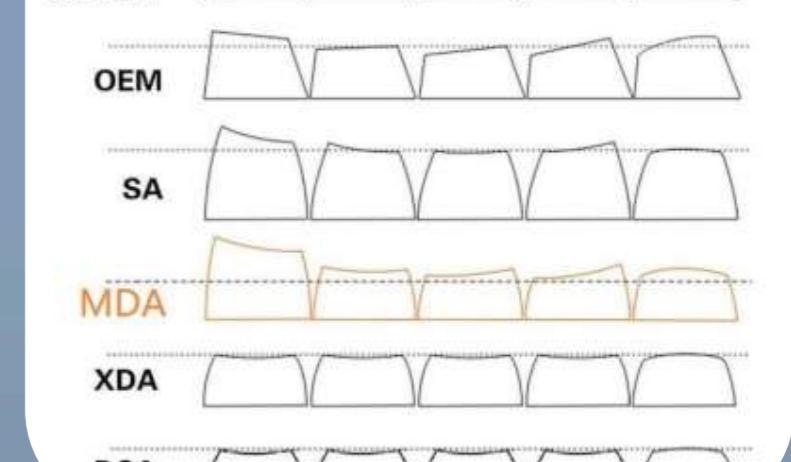
keymap.c 檔案
(出處詳附錄圖
片來源1)



(二)鍵帽

各種鍵帽類型及特色

| Profile | R1 | R2 | R3 | R4 | Space鍵 | 特色 |
|---------|----|----|------|----|--------|---------------------------|
| Cherry | 低 | 中 | 坡度適中 | 最高 | 弧形 | 低高度，較符合人體工學，適合長時間打字或遊戲使用。 |
| OEM | 低 | 中 | 坡度適中 | 最高 | 弧形 | 與Cherry相似但稍高，常見於市售鍵盤。 |
| SA | 高 | 高 | 圓潤形狀 | 高 | 弧形 | 高鍵帽，復古造型，手感飽滿。 |
| MDA | 中 | 中 | 微圓頂 | 中 | 平緩 | 介於SA和XDA之間，較矮的球形鍵帽。 |
| XDA | 均高 | 均高 | 平頂 | 均高 | 平緩 | 所有鍵帽高度一致，適合平面佈局。 |
| DSA | 均低 | 均低 | 平頂 | 均低 | 平緩 | 低鍵帽且所有行列高度相同，適合任意排列。 |



各種不同高度及形狀的鍵帽 (出處詳附錄圖片來源7)

(三)鍵軸

是鍵盤中每個按鍵的核心組件，它負責檢測按鍵被按下的動作，並將訊號傳送至鍵盤的微控制器。鍵軸的設計影響著按鍵的觸感、回饋和聲音。



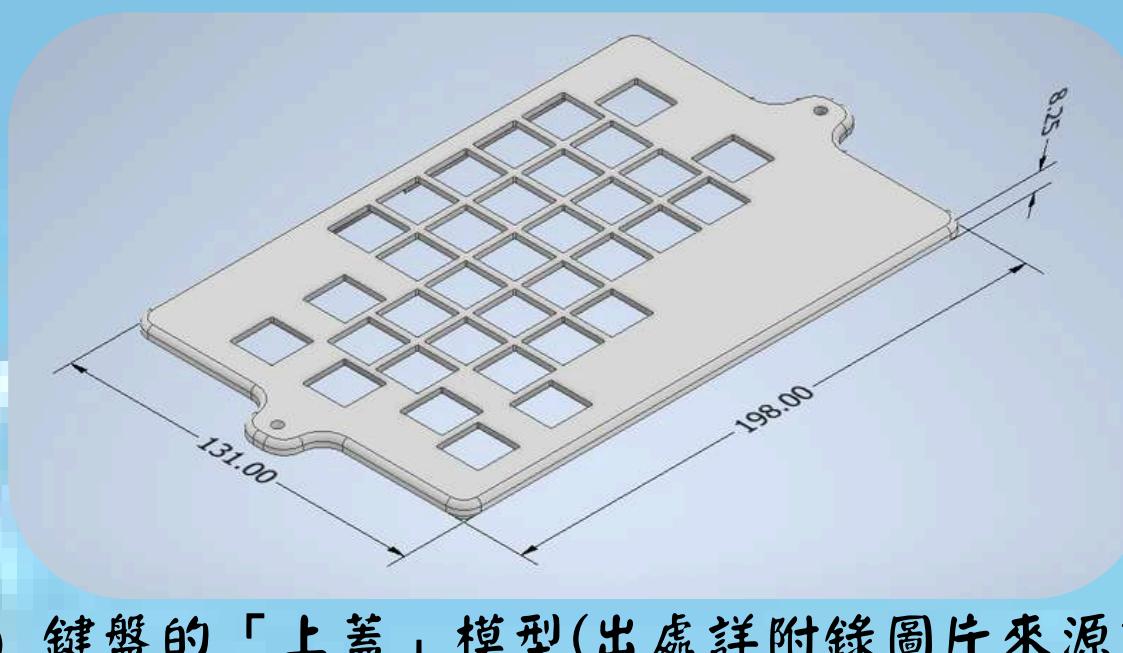
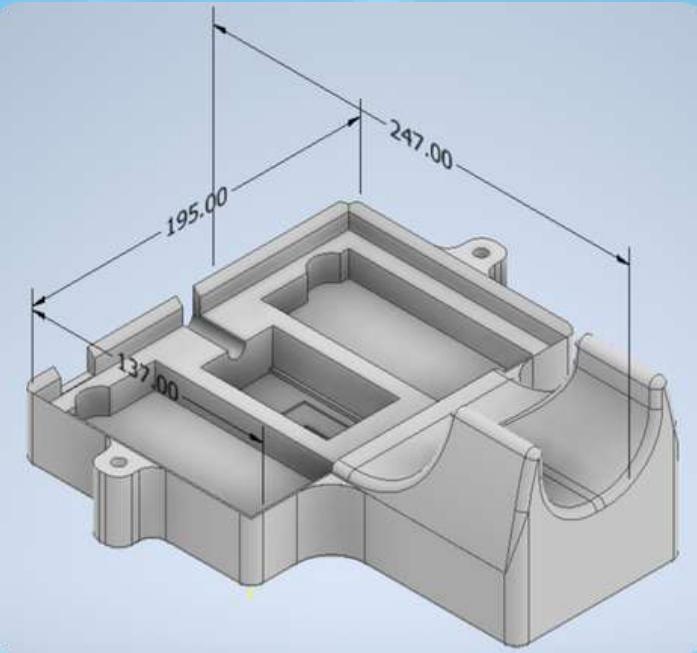
各種品牌之
青軸比較

| 參數 | Cherry MX 青軸 | Gateron 青軸 | Kailh 青軸 | Romer-G 青軸 | Outemu 青軸 |
|------|--------------|------------|----------|------------|-----------|
| 觸發壓力 | 50g | 55g | 50g | 45g | 60g |
| 總行程 | 4.0mm | 4.0mm | 4.0mm | 3.2mm | 4.0mm |
| 觸發行程 | 2.2mm | 2.3mm | 2.0mm | 1.5mm | 2.4mm |
| 壽命 | 5,000萬次 | 5,000萬次 | 5,000萬次 | 7,000萬次 | 5,000萬次 |
| 聲音 | 清脆 | 清脆 | 較脆 | 較悶 | 較響亮 |
| 手感 | 明顯段落感 | 順滑段落感 | 較輕快 | 段落感較輕 | 明顯段落感 |
| 回彈 | 適中 | 柔和 | 偏硬 | 柔和 | 偏硬 |
| 適用場合 | 打字、遊戲 | 打字、遊戲 | 打字 | 遊戲 | 打字、遊戲 |

各種不同的鍵軸(出處詳附錄圖片來源8)

二、外殼及定位版

我們使用3D繪圖軟體Inventor繪製鍵盤之外殼，分為上蓋以及底座兩部分，兩者使用螺絲連接。



鍵盤的「底座」模型
(出處詳附錄圖片來源1) 鍵盤的「上蓋」模型(出處詳附錄圖片來源1)

三、游標控制

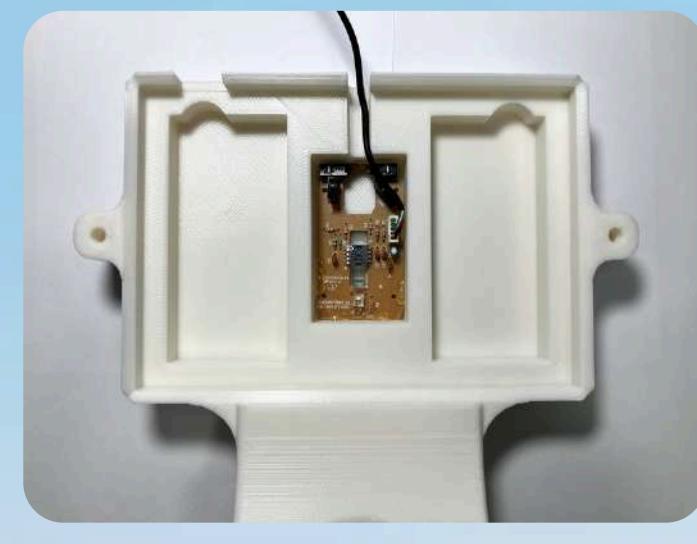
我們將滑鼠的電路板置於模型底部，只利用其光學感測器來控制游標。

滑鼠游標流程圖(出處詳附錄圖片來源1)



四、整合組裝

將滑鼠電路板和鍵盤的PCB電路板放入3D列印的殼中。再來將鍵軸與鍵帽插入上蓋的定位版中，最後，鎖上兩側螺絲就組裝完畢了。



光學感測器放入3D列印中
(出處詳附錄圖片來源1)



放入鍵盤PCB以及鍵軸
(出處詳附錄圖片來源1)



裝上鍵帽-硬體完成
(出處詳附錄圖片來源1)

伍、研究結果

一、游標部分

採用直覺設計，手掌放上按鍵表面後滑動整個鍵盤即可操控螢幕游標。



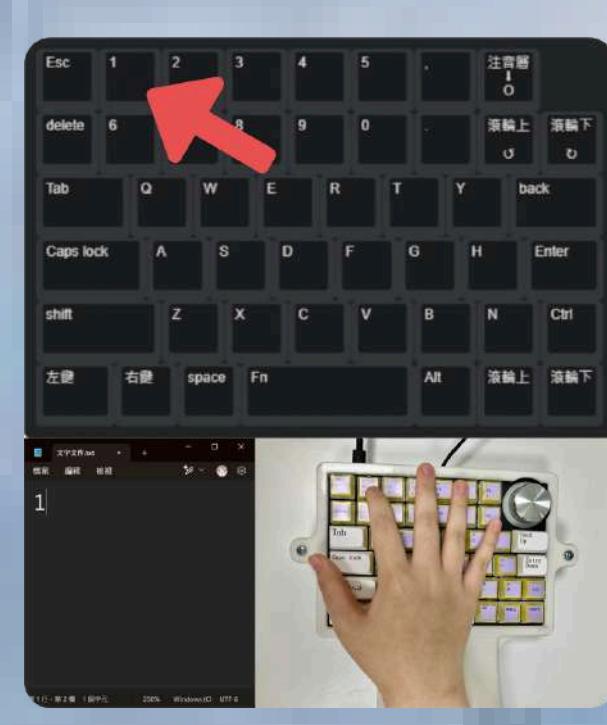
一般滑鼠(左圖)和
我們設計的滑鼠
按鍵(右圖)對照圖
(出處詳附錄圖片
來源1)

二、鍵盤部分

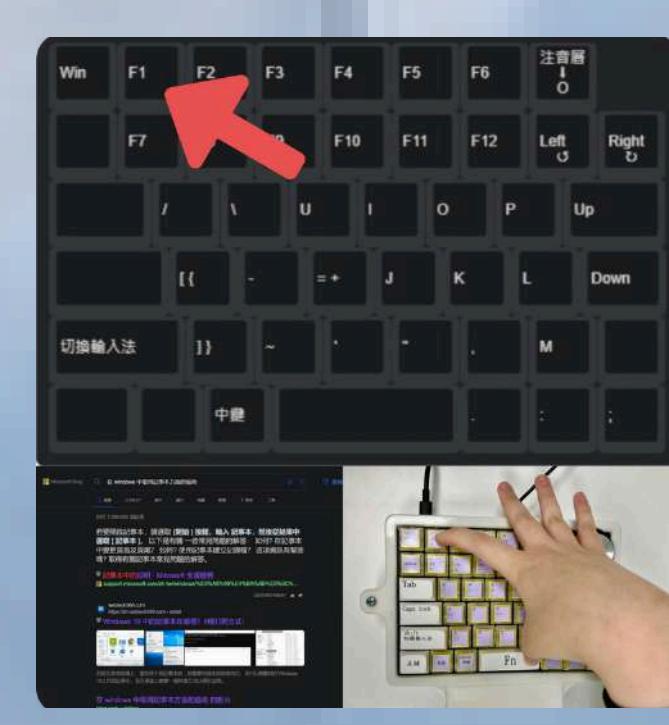
我們的鍵盤一共有6個虛擬層，包含2個主要層和4個附加層。每個主要層都配有兩個附加層，分別使用 Fn 和 Shift 來切換。

| | | |
|-------|------|------|
| Fn | 一般層 | 注音層 |
| | 附加層1 | 附加層3 |
| Shift | 附加層2 | 附加層4 |

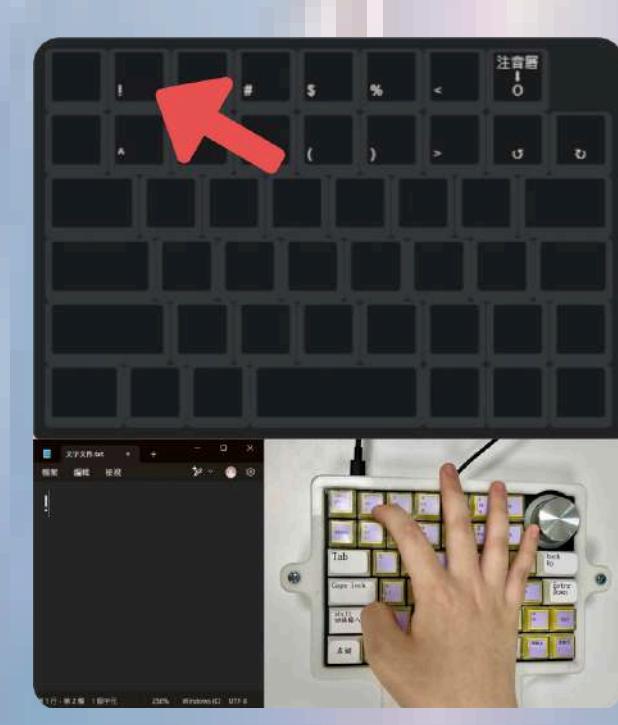
具體概念說明表



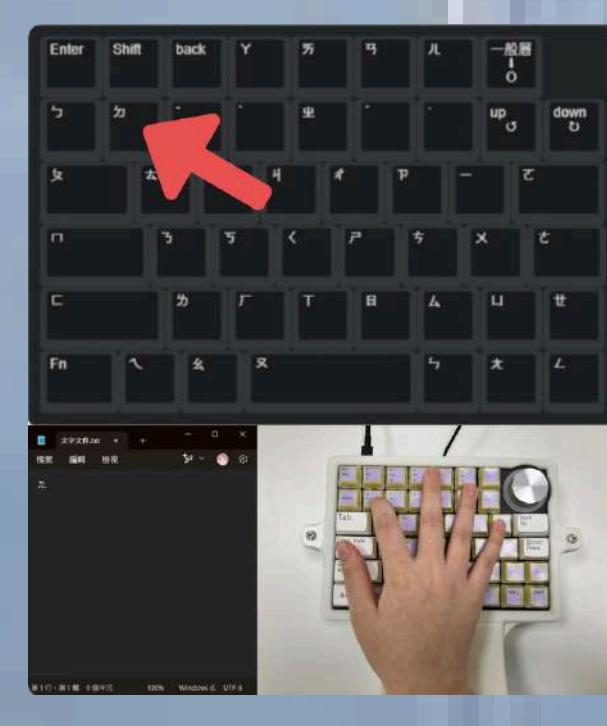
一般層在不加其他鍵時，按
下1鍵會輸入「1」。



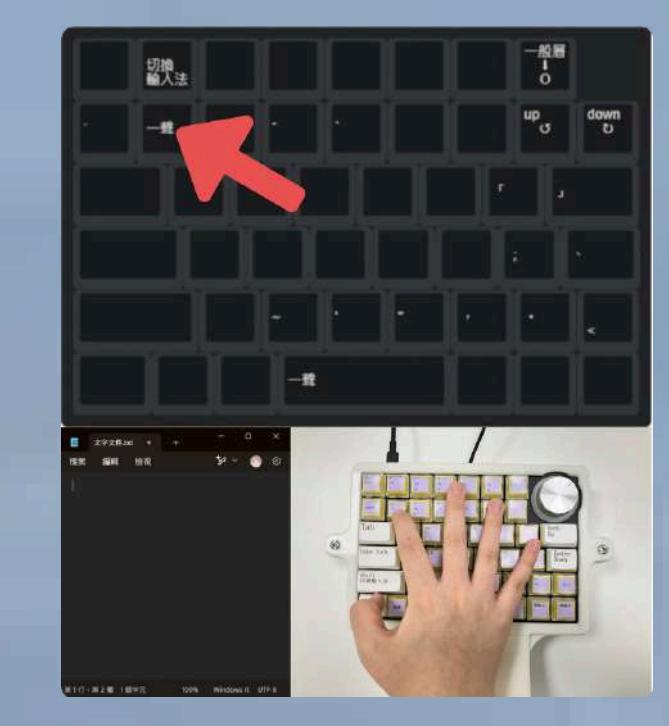
附加層1在按下Fn + 1鍵
時會輸入「F1」。



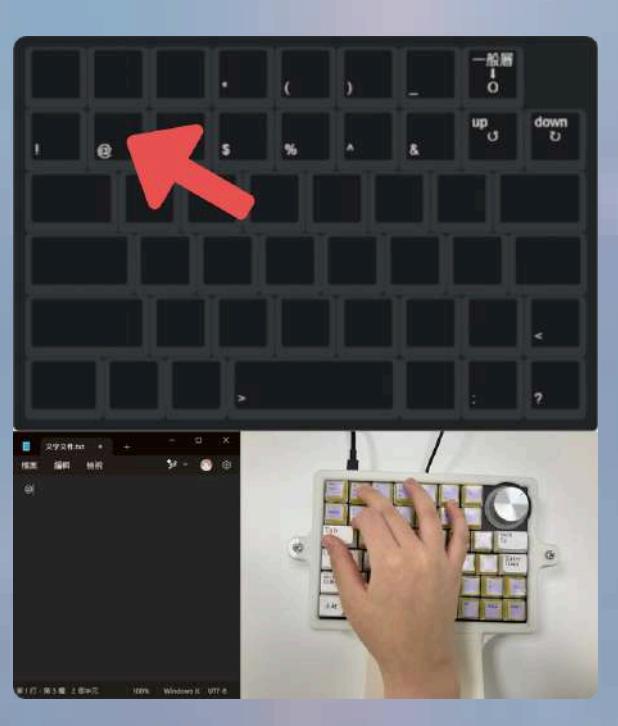
附加層2在按下Shift + 1
鍵時會輸入「！」。



注音層在不加其他鍵時，按
下6鍵會輸入「ㄅ」。



附加層3在按下Fn + 6鍵時
會輸入聲調的「一聲」。



附加層4在按下Shift + 6鍵時
會輸入「@」。

(以上六張出處詳附錄圖片來源1)

一般層：為日常使用所設計，包含滑鼠的左鍵、右鍵、數字、26個英文字母以及各種符號。



一般層 (出處詳附錄
圖片來源1)

注音層：是專門為繁體中文使用者設計，包含37個注音符號、5種聲調，以及所有中文符號。



注音層 (出處詳附錄
圖片來源1)

三、特殊快捷鍵說明

(一) 按下 Win + H，可啟用語音輸入功能，將語音即時轉為文字。

(二) 當使用者操作空間受限（如裝置碰到牆壁或桌緣）時，因本裝置透過滑動整體來控制游標，無法像傳統滑鼠般「提起再放下」來重設位置。為解決此問題，我們用AutoHotkey設計快捷鍵：按下 Ctrl + B 可暫時中斷滑鼠控制，固定游標位置，使用者可調整裝置位置後再繼續操作。此設計有效彌補本裝置體積較大帶來的限制。

四、產品使用教程影片

為了讓使用者快速了解產品操作，我們製作教學影片上傳YouTube，掃描裝置側邊QR Code可連結觀看。



上傳至 YouTube 的使用教學影片

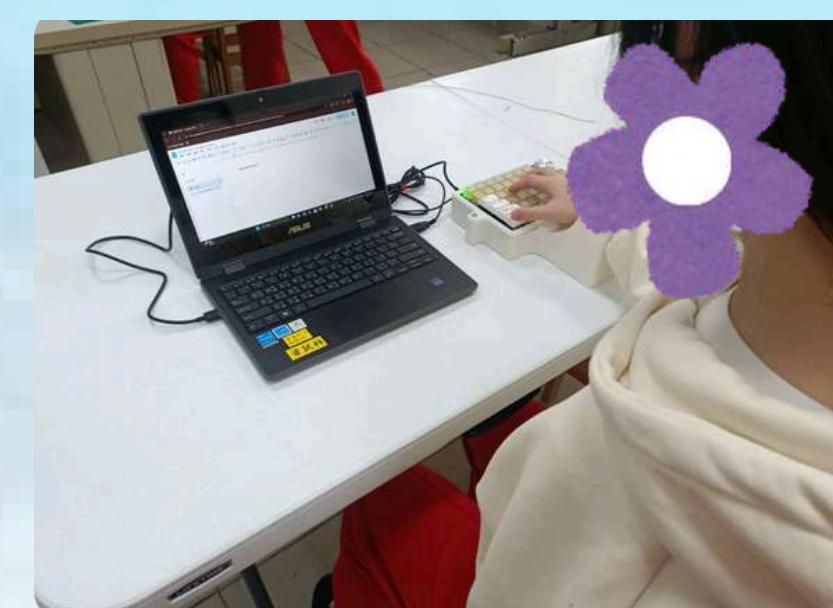
(出處詳附錄圖片來源1)

五、成品操作說明

右手手腕置於符合人體工學的凹槽中，手掌放在鍵盤上單手打字，靠著凹槽的固定和手掌的摩擦力滑動整個鍵盤就可以控制游標。



手腕置於符合
人體工學的凹
槽 (出處詳附
錄圖片來源1)



連接於筆電
上單手打字
(出處詳附
錄圖片來源
1)



使用鍵盤功能
(出處詳附錄
圖片來源1)



使用鍵盤功能
(出處詳附錄
圖片來源1)

掃描成品右側的 QR Code，可以進入附有完整鍵盤配置圖的網頁。



成品右側之
QR Code
(出處詳附錄
圖片來源1)

六、實驗一：不同設備之手腕移動距離差異

以下是Python結合MediaPipe骨架估計函式庫，測試三種設備手腕移動距離差異的結果。以Z、1、0、M鍵為起點，滑鼠左鍵為終點，記錄起按時間與手腕軌跡測算距離。



一般鍵盤滑鼠開始加結束 (出處詳附錄圖片來源1)



單手鍵盤滑鼠開始加結束 (出處詳附錄圖片來源1)



我們的作品開始加結束 (出處詳附錄圖片來源1)

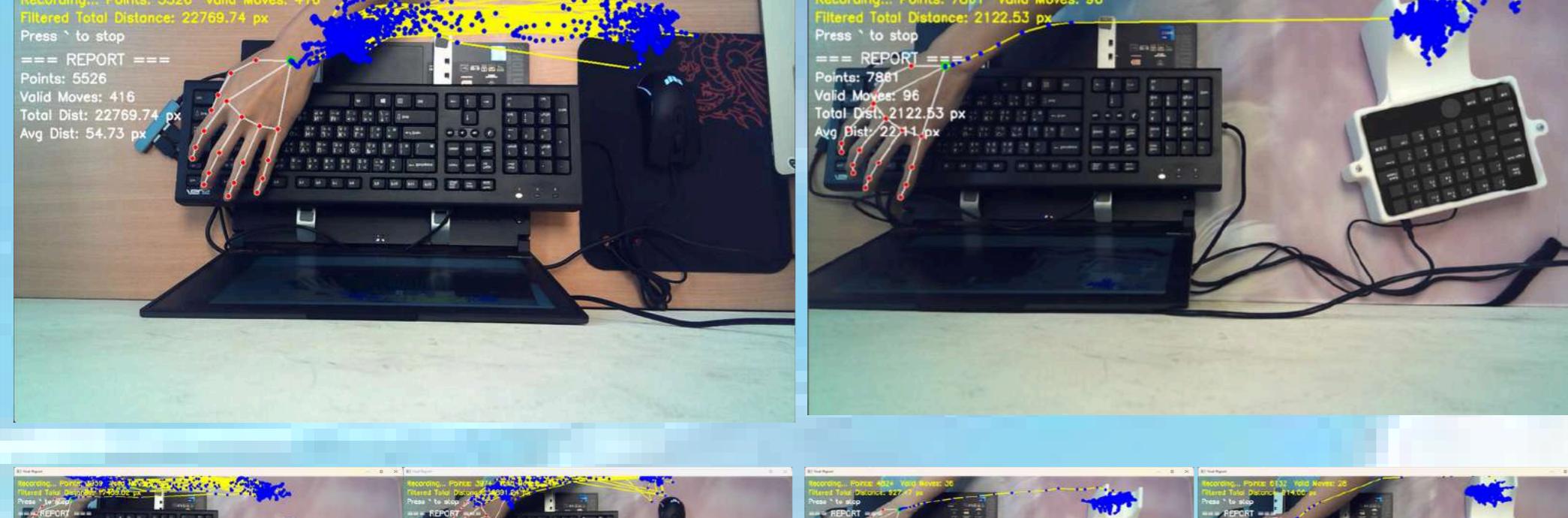
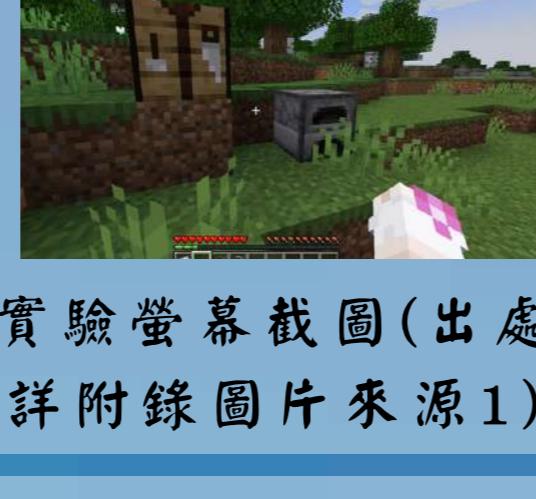
| 時間比較 | 全尺寸鍵盤+滑鼠 | 單手鍵盤+滑鼠 | 我們的作品 |
|------|----------|---------|--------|
| z 鍵 | 1.11 秒 | 1.02 秒 | 0.31 秒 |
| 1 鍵 | 1.05 秒 | 1.41 秒 | 0.29 秒 |
| 0 鍵 | 0.96 秒 | 1.09 秒 | 0.27 秒 |
| m 鍵 | 1.11 秒 | 0.96 秒 | 0.28 秒 |
| 平均值 | 1.06 秒 | 1.12 秒 | 0.29 秒 |

綜合時間比較

數據顯示，全尺寸鍵盤和單手鍵盤的平均操作時間均需1秒以上，而我們的產品僅需0.3秒即可完成操作。在操作距離方面，全尺寸鍵盤需伸展至45公分，單手鍵盤雖縮短至26公分，但我們的設計僅需1公分手腕移動距離。

七、實驗二：比較實際操作上的數值差異

接下來，我們運用Python結合MediaPipe骨架估計函式庫，比較單手操作時傳統鍵盤與我們作品的數值差異。以《Minecraft》為例，招募志願者分別用全尺寸鍵鼠與我們的裝置單手完成任務，記錄手腕軌跡與手部移動次數。



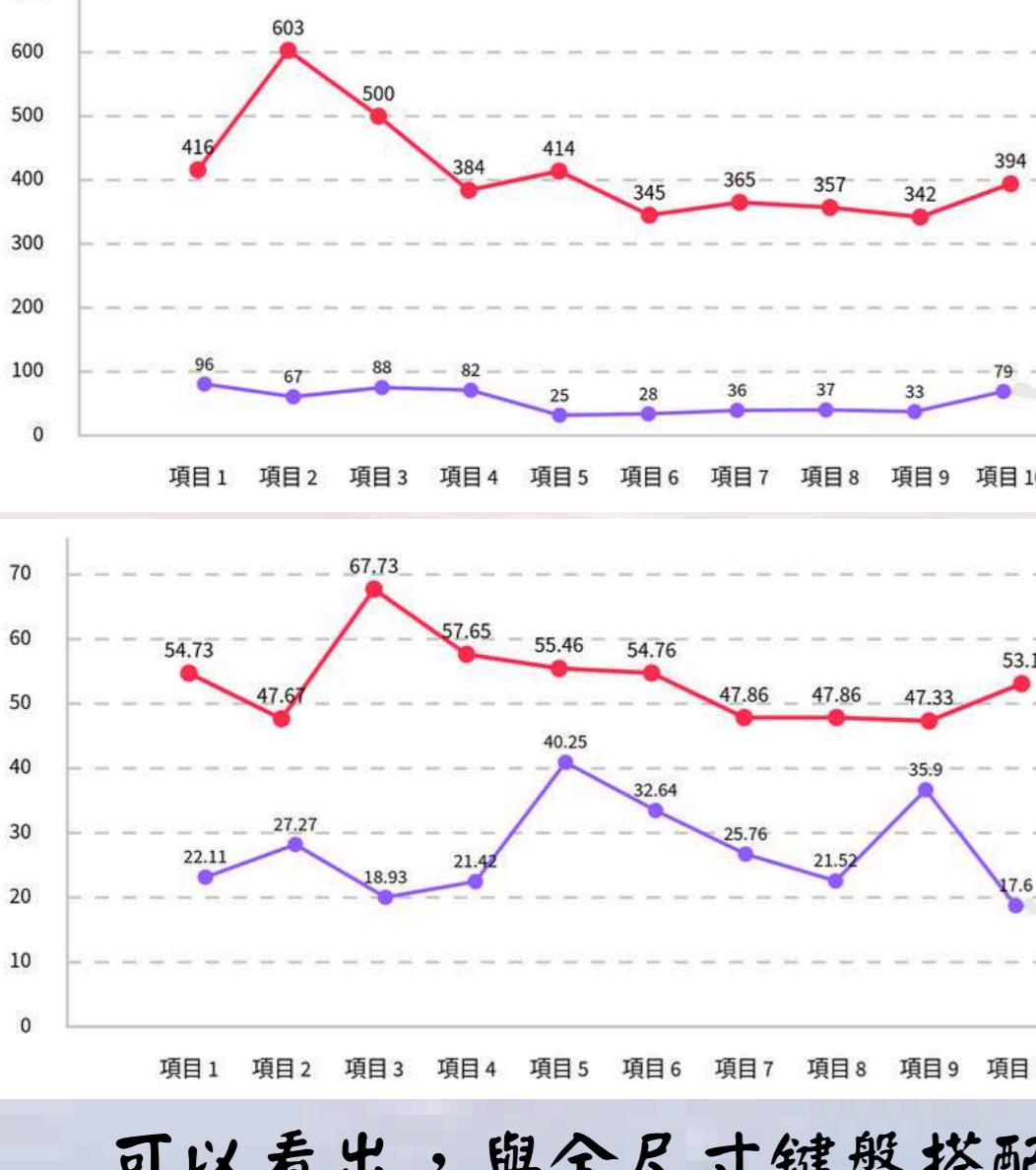
全尺寸鍵盤滑鼠和我們的作品實操比較 (出處詳附錄圖片來源1)

| Tr | 項目 | # 移動次數 | # 遊距離 | # 平均移動距離 |
|-----|-----|-----------|--------|----------|
| 第一項 | 416 | 22769.74 | 54.73 | |
| 第二項 | 603 | 28746.36 | 47.67 | |
| 第三項 | 500 | 33866.42 | 67.73 | |
| 第四項 | 384 | 22138.08 | 57.65 | |
| 第五項 | 414 | 22962.13 | 55.46 | |
| 第六項 | 345 | 18891.24 | 54.76 | |
| 第七項 | 365 | 17468.02 | 47.86 | |
| 第八項 | 357 | 17085.23 | 47.86 | |
| 第九項 | 342 | 16185.67 | 47.33 | |
| 第十項 | 394 | 20952.59 | 53.1 | |
| 平均 | 412 | 22106.548 | 53.415 | |

【全尺寸鍵盤+滑鼠】的測試數據及平均值

| Tr | 項目 | # 移動次數 | # 遊距離 | # 平均移動距離 |
|-----|------|----------|-------|----------|
| 第一項 | 96 | 2122.53 | 22.11 | |
| 第二項 | 67 | 1827.13 | 27.27 | |
| 第三項 | 88 | 1665.61 | 18.93 | |
| 第四項 | 82 | 1756.1 | 21.42 | |
| 第五項 | 24 | 965.92 | 40.25 | |
| 第六項 | 28 | 914.06 | 32.64 | |
| 第七項 | 36 | 927.47 | 25.76 | |
| 第八項 | 37 | 796.3 | 21.52 | |
| 第九項 | 33 | 1184.54 | 35.9 | |
| 第十項 | 79 | 1390.65 | 17.6 | |
| 平均 | 65.2 | 1364.139 | 26.34 | |

【我們的作品】的測試數據及平均值



移動次數折線圖 (出處詳附錄圖片來源1)

平均移動距離折線圖 (出處詳附錄圖片來源1)

可以看出，與全尺寸鍵盤搭配滑鼠的配置相比，我們的作品在手腕移動次數和距離上均明顯縮短。因其將滑鼠按鍵整合於鍵盤，以滑動控制游標，有效降低手部疲勞。

兩項實驗從不同角度對比我們的作品與其他輸入設備，發現其可有效減少手腕移動距離和次數，緩解疲勞，鍵鼠融合設計兼具靈活性與適用性。

我們研究了市場上各類鍵盤，發現傳統鍵盤多數以雙手操作為前提，對單手使用者並不友善。因此，我們整理了不同類型鍵盤的特點，並與我們的設計進行比較：

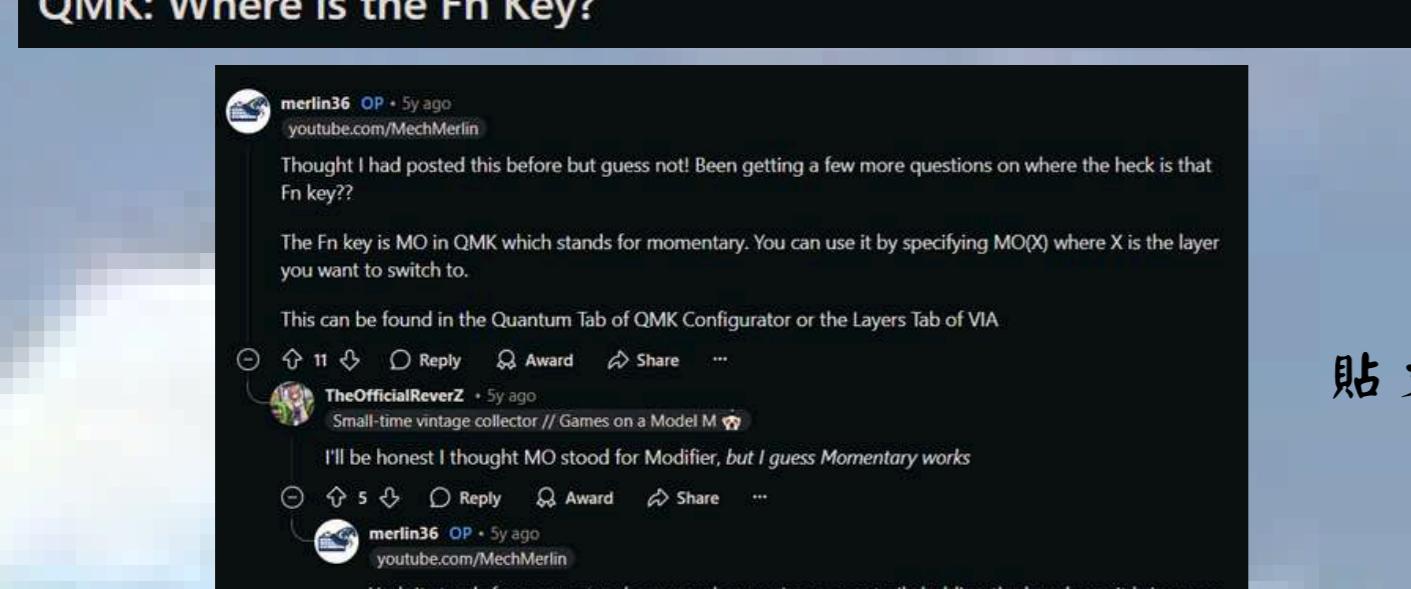
各種鍵盤特性比較表

| 特性 | 鍵盤種類 | 全尺寸鍵盤 | 迷你鍵盤 | 單手鍵盤 | 我們的鍵盤 (含滑鼠在內部) |
|-------|-----------------|---------|---------|---------|----------------|
| 按鍵數量 | 104 | 61 | 20-40 | 51 | |
| 使用方式 | 雙手 | 雙手/單手 | 單手 | 單手 | |
| 攜帶性 | 低 | 中等 | 高 | 中等 | |
| 學習成本 | 低 | 低, 中 | 中-高 | 高 | |
| 打字效率 | 因學習程度而異 | 因學習程度而異 | 因學習程度而異 | 因學習程度而異 | |
| 遊戲適應性 | 高 | 高 | 依遊戲類型而異 | 依遊戲類型而異 | |
| 可自訂性 | 一般(鍵腳, 高(可程式化)) | 高(可程式化) | 高(多功能鍵) | 高(全自訂) | |
| 適合對象 | 一般使用者 | 遊戲、簡單工作 | 特殊需求、遊戲 | 需單手操作者 | |
| 鍵鼠狀態 | 鍵鼠分離 | 鍵鼠分離 | 鍵鼠分離 | 鍵鼠合併 | |

八、我們遇到的問題

在組裝各部分的零件時，我們發現控制游標移動的光學感測器經常無法感應滑動。起初以為是感測器故障，但在取出感測器並手動在桌面上滑動測試後，才發現問題出在感測器與桌面的距離。經過多次嘗試，並輔以熱熔膠調整，最終才找到適當的位置。

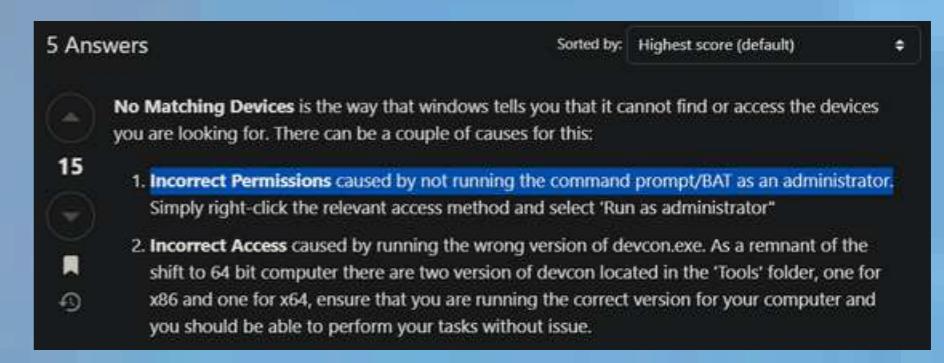
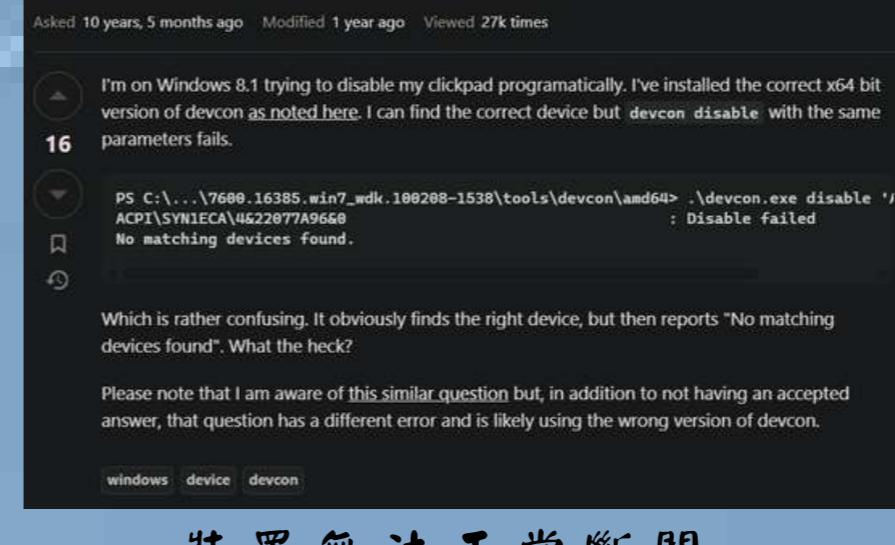
在鍵盤韌體程式中，每個鍵都有專屬的鍵碼。例如，空白鍵 (space) 的鍵碼是 KC_SPC。然而，當我編寫到 Fn 鍵時，卻在鍵碼表上找不到與之對應的鍵碼。經過一番搜尋後，我終於在 Reddit 上找到了一則相關的貼文：



在 QMK 中 Fn 鍵在哪裡？

原來，Fn 鍵並不是一個文字鍵，而是一個功能鍵。它的作用是「暫時切換鍵盤到第 n 層」，其鍵碼為 MO(n)，其中 n 代表要切換到的層。

在製作「定住螢幕游標」功能的最後階段時，雖然程式碼經過檢查並未發現任何問題，但按下對應快捷鍵時，螢幕游標卻完全沒有反應，甚至未顯示任何錯誤訊息。經過一番搜尋後，我們在 Stack Overflow 上找到了一則相關的討論貼文：



裝置無法正常斷開

我們發現，雖然程式本身已具備斷開與重新連接輸入裝置的能力，但由於系統權限的限制，預設情況下無法成功執行相關操作。因此，在執行這個程式時，要選擇「以系統管理員身分執行」，以賦予程式足夠的權限來控制裝置的連線狀態。唯有如此，才能順利實現我們所設計的「定住螢幕游標」功能（實際上為暫時中斷滑鼠連線），確保使用者在重置裝置位置時，游標不會意外移動。

陸、結論

一開始，我們發現許多使用者因肢體障礙或僅能單手操作，使用傳統鍵鼠不便，因此決定開發一款整合鍵盤與滑鼠功能的單手操作裝置。為達此目標，我們進行市場調查與技術研究，並學習PCB設計、鍵軸配置、3D建模與列印，打造人體工學結構；同時深入QMK韌體，客製化鍵位與功能。過程中克服整合技術、鍵位設計與成本控制等挑戰。這次專題不僅提升我們的技術實力，也體現團隊合作價值。未來將持續優化產品，朝市場推廣與智慧化發展邁進。

柒、其他

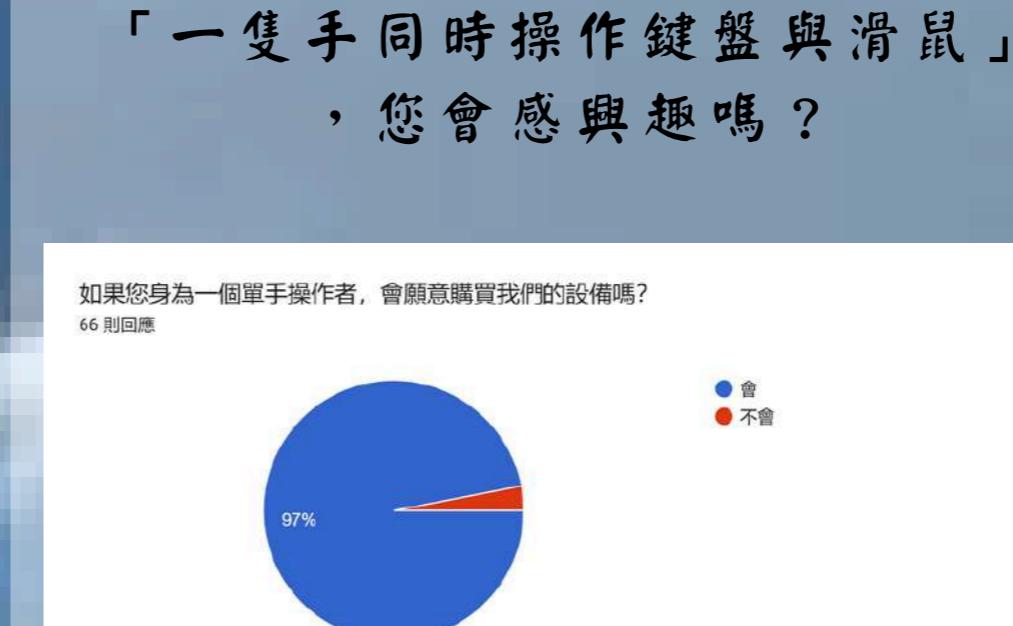
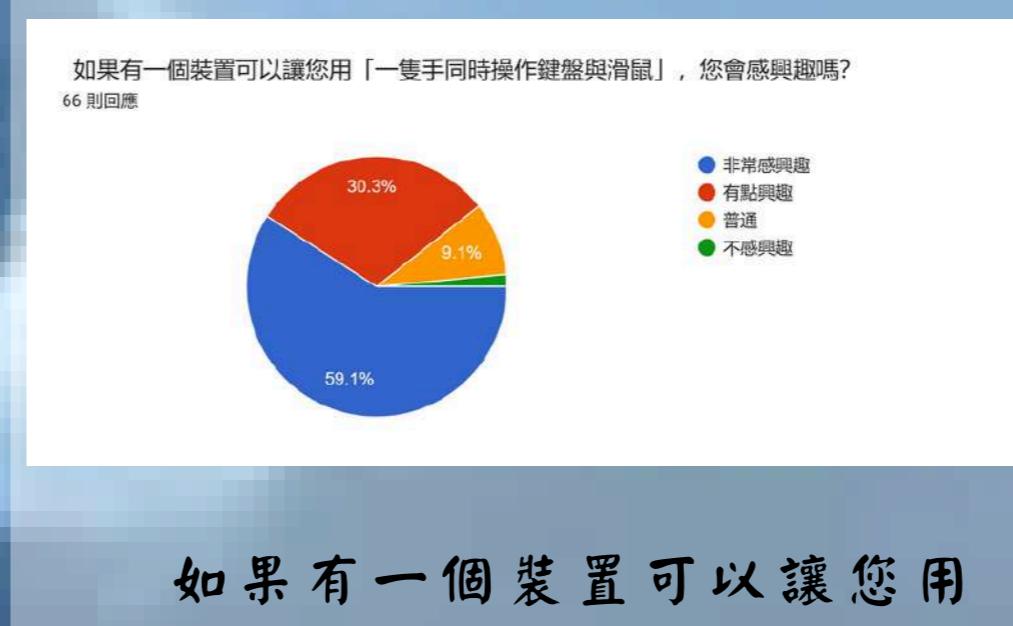
一、圖片來源

1. 本圖片由作者親自拍攝或製作
2. 本圖片引用自 Github 網址 <https://github.com/kicad>
3. 本圖片引用自 Brandlogos 網址 <https://brandlogos.net/autodesk-inventor-logo-108187.html>
4. 本圖片引用自 QMK MSYS 網址 <https://msys.qmk.fm/>
5. 本圖片引用自維基百科 網址 <https://tinyurl.com/22mxqzdu>
6. 本圖片引用自 Autohotkey 官方網站 網址 <https://tinyurl.com/283tzuk3>
7. 本圖片引用自 Reddit 網址 <https://tinyurl.com/22b22y81>
8. 本圖片引用自 ikanchai 網址 <http://m.ikanchai.com/pcarticle/369290>
9. 本圖片引用自維基百科 網址 <https://tinyurl.com/26zvmd3z>
10. 本圖片引用自 MOMO 網址 <https://tinyurl.com/2cf7y4w8>
11. 本圖片引用自 Coupang 網址 <https://tinyurl.com/2b9x8685>
12. 本圖片引用自 Logitech 網址 <https://tinyurl.com/294ap5j7>
13. 本圖片引用自 Reddit 網址 <https://tinyurl.com/23jvkm4c>
14. 本圖片引用自 Stackoverflow 網址 <https://tinyurl.com/24btqz4j>

二、問卷調查



您是否曾經遇過以下哪種使用上的困擾？



您平均多久使用一次鍵盤與滑鼠？

