

# 專題名稱：改變吧!傳統風扇控制



作者：

國立屏東高工 電子科 三年甲班 許育誠 15 號

國立屏東高工 電子科 三年甲班 李樂華 5 號

指導老師：范佐輝、吳盈震

關鍵字：手機藍芽遙控、感測溫度、自動控制

中華民國 114 年 1 月 10 日

## 目錄

摘要.....	3
壹、創意動機及目的.....	3
貳、作品特色及創意特質.....	4
一、手機藍牙遙控操作.....	4
二、環境感測與自動化控制.....	4
三、模組化設計與簡易開發.....	4
四、提升傳統電風扇價值.....	4
五、親民且易於普及.....	4
參、創意發想與研究過程.....	5
一、創意構想與可行性分析.....	5
二、軟體和材料選用.....	5
三、目標功能設計.....	7
肆、設計相關原理.....	8
伍、製作歷程說明.....	10
陸、作品功能敘述.....	14
柒、研究結論.....	16
捌、參考資料.....	17

## 摘要

智慧家庭技術的興起為日常生活帶來極大便利，但許多家電仍停留在傳統手動操控的階段，缺乏智慧化和個性化功能。針對電風扇這一常見設備，本專題提出利用ESP32藍牙模組與MIT App Inventor 2 (AI2) 平台相結合，改造傳統電風讓其具備手機藍芽遙控與智能控制的功能。透過手機應用程式可實現遠程操作，包括風速調節、模式選擇與開關控制，提升使用者的操作體驗。

另外本專題進一步融入環境感測技術，透過收集溫度與濕度數據，使電風扇能根據環境變化智能運行，實現節能與自動化的效果。此專題旨在探索物聯網技術於智慧家居中的應用，著重於硬體與軟體整合的實踐，為傳統家電賦予全新價值，並提供便利、高效的功能，展示未來智慧家電的潛力。

## 壹、 創意動機及目的

隨著智慧家電技術的普及，越來越多的用電設備被賦予智能化功能，使得人們的生活更加便捷。然而在電風扇這類常見家用電器中，現今仍然以傳統操作方式為主，缺乏智慧控制與移動設備的聯動功能，也無法實現遠程和個性化控制。基於以上種種原因啟發了我，是否可以透過ESP32的藍芽模組與手機開發設計工具 (MIT App Inventor 2, 簡稱AI2) 結合，改造開發出一台具有智能遙控功能的電風扇，為用大家帶來更便利且高效的操作體驗。

此外我們還觀察到現代生活中，許多人常需要在不同的房間使用電風扇，但手動調節不僅麻煩，還可能忘記關閉導致能源浪費。

因此我們需要設計出不同地電扇控制功能，除了有手機遙控更透過整合環境感測數據 (例如溫度、濕度)，讓電風扇有根據環境條件自動控制，達到節能與便利的雙重目的。

## 貳、 作品特色及創意特質

### 一、手機藍牙遙控操作

利用 ESP32 藍牙模組與 MIT App Inventor 2 開發平台，實現透過手機應用程式進行電風扇的遠程操控。保留手動功能的同時用戶也可以使用手機輕鬆調整風速、切換模式及控制開關，突破傳統電風扇的操作限制，提供更便捷的用戶體驗。

### 二、環境感測與自動化控制

結合溫濕度感測器，實時監測環境數據。電風扇可根據預設的舒適範圍自動調整風速或啟停運行，達到節能和提升舒適度的效果。這一功能使電風扇更具智慧性，適應不同使用場景。

### 三、模組化設計與簡易開發

採用模組化設計，硬體與軟體清晰分工，便於維護與擴展。以 ESP32 為核心控制器，結合 AI2 圖形化開發工具，使開發過程簡化，降低技術門檻，為未來的功能擴展留出空間。

### 四、提升傳統電風扇價值

針對傳統電風扇缺乏智能化的痛點，結合物聯網技術，為傳統電風扇注入新功能與價值，展現智慧家電的未來應用。

### 五、親民且易於普及

本設計採用成本較低的硬體模組（如 ESP32），並以開放性平台開發，整體系統經濟實惠，適合推廣至家庭、學校及其他日常應用場景，具有良好的市場應用潛力。

## 參、 創意發想與研究過程

### 一、 創意構想與可行性分析

#### 1. 現狀與問題確認

傳統電風扇以手動操作為主，缺乏遠程操控與智能化功能，無法滿足現代智慧家電需求。基於此，我們構想利用ESP32與手機應用程式，實現電風扇智能升級，提升操作便利性與使用體驗。

#### 2. 解決方案與可行性

為了提升風扇的智慧化與使用便利性，本方案採用 **ESP32 模組** 實現 **手機藍牙遙控功能**，讓使用者能夠透過手機遠端調整風速、切換模式及控制開關，增強操作靈活度。同時，**整合溫濕度感測器**，使系統能根據即時環境數據自動調整運行狀態，達到節能與舒適並重的效果。此外，為了降低開發成本與技術門檻，我們選用 **MIT App Inventor 2** 平台開發操作介面，使開發流程更直覺化，進一步提升設計的可行性與普及性。

### 二、 軟體和材料選用

#### 1. 硬體選用



圖1 使用ESP32模組為核心



圖2 使用DHT11溫濕度感測器與繼電器模組搭配ESP32，構建控制電路，實現風扇開關與風速調節的智能化控制

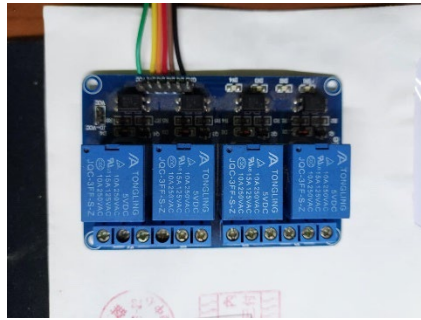


圖3 使用繼電器控制電路運作



圖4 使用LED燈顯示工作狀態

## 2. 軟體設計



圖5 使用MIT App Inventor 2開發手機應用程式，實現直觀操作介面與藍牙連接功能



圖6 使用Arduino IDE撰寫ESP32控制程式，實現藍牙通訊、環境數據處理與設備控制的邏輯整合

### 三、 目標功能設計

在目標功能設計方面，本專題風扇具備三種控制模式：手動控制、App 遙控與智能控制，並透過互斥設計確保操作不衝突。使用者可透過實體按鈕進行手動控制，或透過手機 App 遙控開關與風速調整（高、中、低），當任一模式啟動時，其他模式將失效。此外，智能控制模式則運用 ESP32 連接 DHT11 溫濕度感測器，根據環境溫度自動調整風速，達到節能與便利並重的效果。整體設計以 ESP32 為核心，搭配繼電器模組實現風扇開關控制，確保系統穩定運作並提升傳統風扇的智慧化價值。

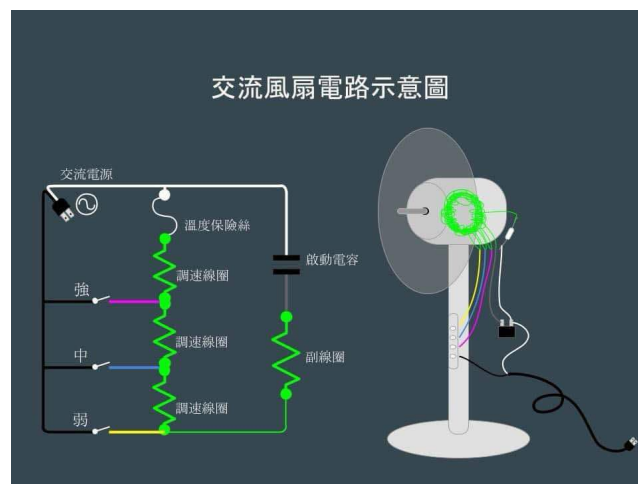


圖7 了解電風扇整體接線原理

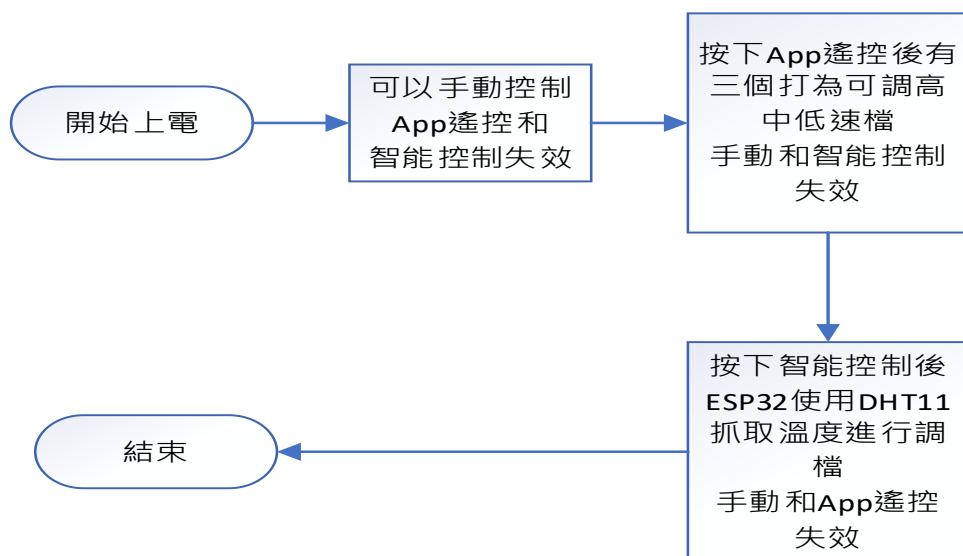


圖8 原理圖

## 肆、設計相關原理

本專題基於 ESP32 微控制器，結合 BLE 無線通訊、DHT11 溫度感測、繼電器控制與 MIT App Inventor 2 應用開發，實現風扇的智慧化控制，使其具備遠端操控、自動調速與節能運行的特性。以下為各技術的整合與應用原理：

### 1. ESP32 控制與 BLE 無線通訊

ESP32 作為核心控制單元，負責處理藍牙通訊、溫度感測數據分析及風扇控制指令傳輸。BLE（藍牙低功耗）技術讓 ESP32 以伺服器模式運行，透過 BLE UUID（通用唯一識別碼）與手機 App 建立連線，接收開關與風速調整指令，確保裝置間的即時通訊與低功耗運行。當使用者在手機 App 操作風扇開關或調整風速時，指令透過 BLE 傳輸至 ESP32 解析數據後，根據不同模式（手動、遠端、智慧）切換相應的運行狀態，並透過 GPIO 端口控制風扇的運作。

### 2. DHT11 溫度感測與智慧調速

DHT11 為數位溫濕度感測器，負責監測環境溫度，並將數據回傳 ESP32 進行處理。在智慧模式下，系統根據設定的溫度閾值，透過 ESP32 自動調整風速：

- 低於 10°C：風扇關閉（LED 全滅）
- 10°C~20°C：低速運行（LED 綠燈）
- 20°C~30°C：中速運行（LED 黃燈）
- 高於 30°C：高速運行（LED 紅燈）

ESP32 持續讀取 DHT11 數據，並根據變化即時調整風速，以達到最佳散熱效果與節能目標。

### 3. 繼電器控制風扇開關

繼電器（Relay）作為電子開關，負責控制風扇電源。ESP32 透過 GPIO 端口發送高低電位訊號，實現風扇的開啟與關閉：

- 高電位（HIGH）→ 繼電器導通 → 風扇運行
- 低電位（LOW）→ 繼電器斷開 → 風扇停止

此設計確保 ESP32 能夠透過低電壓訊號安全控制高電壓風扇，提升系統穩定性與使用壽命。

## 4. MIT App Inventor 2 手機控制介面

手機 App 透過 MIT App Inventor 2 開發，提供直觀的使用者介面，使使用者能夠透過 BLE 遠端控制風扇。主要功能如下：

- **BLE 連線管理**：搜尋並連接 ESP32 設備
- **手動模式**：用戶直接在 App 內調整風扇開關與風速
- **遠端模式**：透過 App 遙控風扇，避免手動操作的不便
- **智慧模式**：讓 ESP32 根據 DHT11 溫度數據自動調節風速

MIT App Inventor 2 的視覺化程式設計降低開發難度，使系統更容易維護與擴展。

## 5. 模式切換與互斥機制

為避免不同模式間的衝突，系統設計了**互斥控制機制**，確保一次僅能啟動一種模式，防止誤操作與邏輯錯亂。其運行規則如下：

模式	控制方式	其他模式狀態
手動模式	透過實體按鈕或 App 手動調節風速	遠端與智慧模式失效
遠端模式	使用手機 App 進行遙控	手動與智慧模式失效
智慧模式	依據環境溫度自動調速	手動與遠端模式失效

表 1 互斥控制邏輯

此機制確保模式切換流暢，防止系統因多重控制信號產生異常行為，提高可靠性與使用體驗。

## 伍、製作歷程說明

APP 設計和製作



圖 12 這是另一位組員在製作手機 App

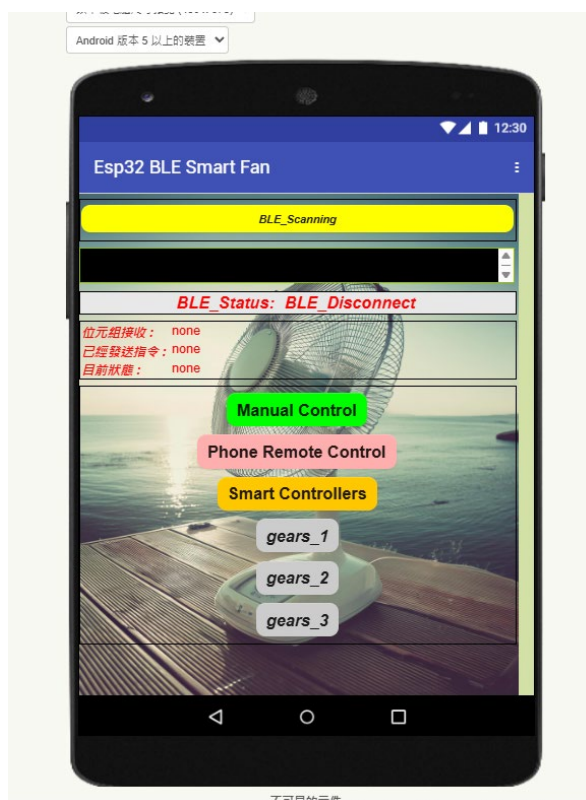


圖 13 這是這是另一位組員設計的控制介面

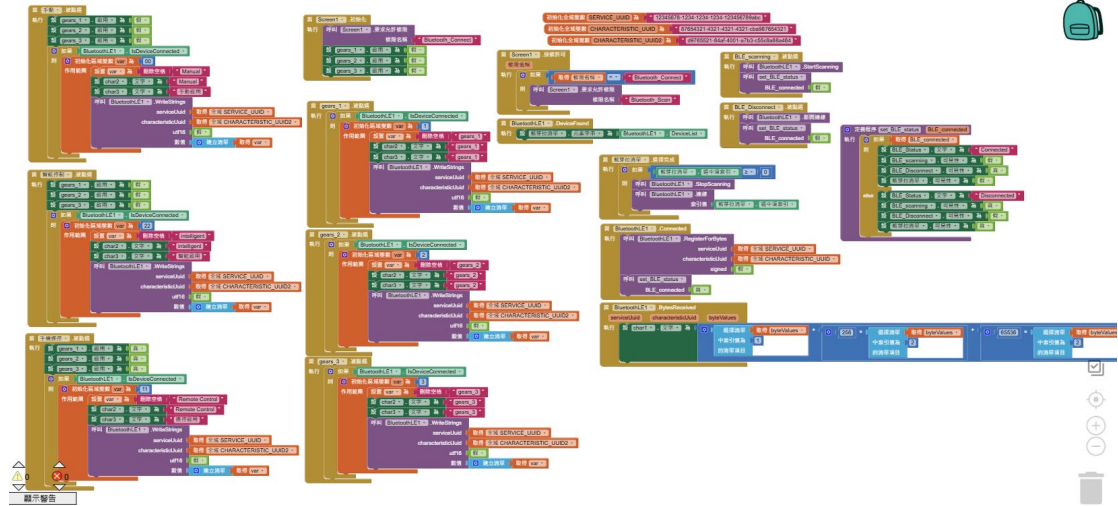


圖 14 這是這是另一位組員製作的手機相關控制程式執行邏輯



圖 15 這是在實作 ESP32 連接要控制的硬體設備

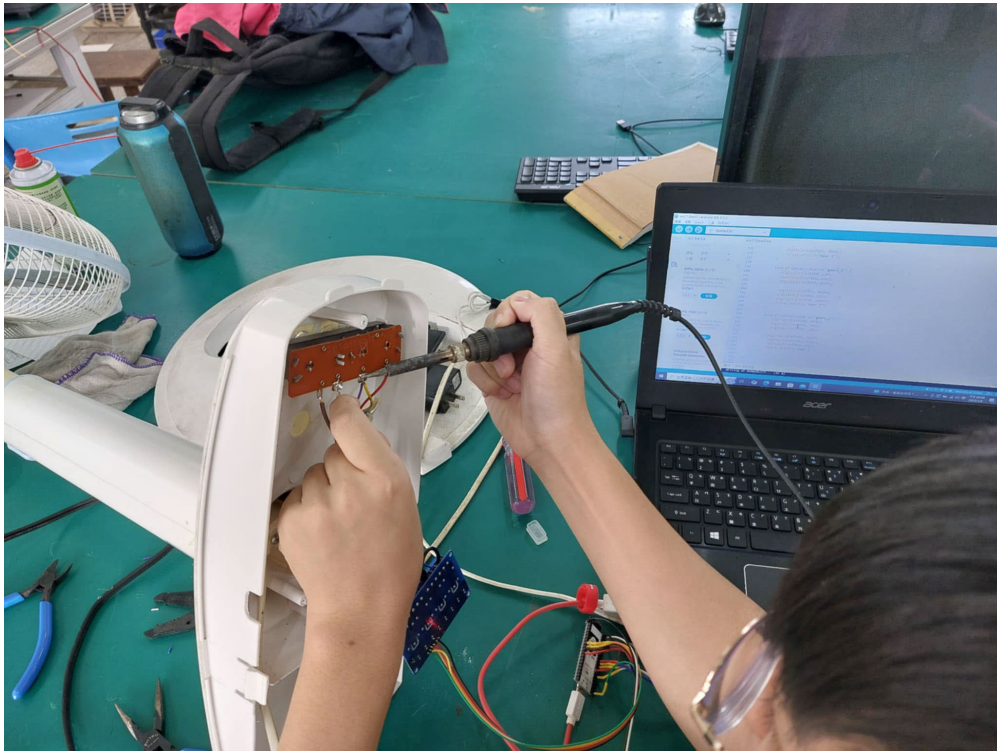


圖 16 這是另一位組員在連接整體線路



圖 17 是我在使用 arduino 撰寫 ESP32 控制程式搭配 App



圖 18 這是在測試和 Debug



圖 19 這是最終成品。

## 陸、作品功能敘述

功能敘述：



圖 9 這是手動模式，在剛開始插電 110V 時，LED 燈全亮代表手動調檔啟用，手機 App 其他功能鍵是點不了的，只能手動調檔，只有用戶在 App 按下 Manual Control 後手動調檔才會失效，進而使用其他功能。



圖 10 這是手機調檔，高速時 LED 顯示紅燈、中速時 LED 顯示黃燈、低速時 LED 顯示綠燈，用戶在 App 按下 Phone Remote Control 之後，按鈕 gears1、gears2、gears3 按了才會有條檔功能之反應，然後 智能控制和手動 不動作。



圖 11 這是智能溫度調檔，高速時 LED 顯示紅燈(超過  $30^{\circ}\text{C}$ )、中速時 LED 顯示黃燈( $20^{\circ}\text{C}\sim 30^{\circ}\text{C}$ )、低速時 LED 顯示綠燈( $10^{\circ}\text{C}\sim 20^{\circ}\text{C}$ )、 $10^{\circ}\text{C}$  以下 LED 全滅並且電扇不做動，用戶在 App 按下 Phone Remote Control 之後，gears1、gears2、gears3 按鈕按了才會有調檔功能，然後 手動和 App 控制 不動作。

## 柒、研究結論

本專題透過 ESP32 微控制器、BLE 無線通訊、DHT11 溫度感測、繼電器控制與 MIT App Inventor 2，成功將傳統風扇改造成智慧化設備，實現遠端操控與環境自適應調速功能。透過手動模式、遠端模式與智慧模式的整合，用戶能夠更靈活地控制風扇，提高便利性與使用體驗，同時減少能源浪費，達到節能與智慧化的雙重目標。

### 研究成果與貢獻

1. **智慧化風扇控制設計：**
  - **三種模式整合**（手動、遠端、智慧），提升風扇使用彈性。
  - **BLE 無線操控** 讓風扇可透過手機 App 進行遠端調節，突破傳統按鍵操作的限制。
2. **環境感測與節能應用：**
  - **DHT11 溫度感測器** 監測環境變化，風扇可自動調速，減少不必要的能源消耗。
  - **溫控智慧模式** 確保用戶在不同氣溫條件下均可獲得最佳舒適度。
3. **軟硬體技術整合與學習：**
  - 深入學習 **嵌入式系統開發、BLE 無線通訊、電子電路控制** 等核心技術。
  - 透過 MIT App Inventor 2 開發手機 App，簡化使用者操作，提升互動性與可行性。
4. **問題解決與系統優化：**
  - **BLE 連線穩定性優化**，提升設備配對與操作的即時反應能力。
  - **錯誤數據處理機制** 確保 DHT11 讀取的溫度數據準確，避免風速異常變化。

---

### 未來發展與應用

雖然本專題已成功驗證智慧風扇的可行性，但仍可進一步優化並擴展應用：

1. **Wi-Fi 遠端控制：**
  - 透過 Wi-Fi 或 MQTT 協議，讓用戶可在更遠距離內控制風扇，實現雲端監控與遠端操控。
2. **語音控制與智慧家居整合：**
  - 結合 Google Assistant、Alexa 等語音助理，使風扇支援語音指令，進一步提升智能化應用。

### 3. 多感測器擴展：

- 除了溫度感測外，可整合 PM2.5、CO<sub>2</sub>、濕度感測器，讓風扇能根據空氣品質進行智慧調節。

### 4. 系統安全性與穩定性強化：

- 增加 BLE 訪問權限驗證，防止未授權設備進行控制，提高設備安全性。
- 採用更耐用的電子元件，提高設備長期運行的穩定性。

本專題成功驗證物聯網技術在智慧家電領域的可行性，透過 BLE 連線、環境感測、自動調速與遠端控制，提升傳統風扇的智慧化程度，並提供更高效、便捷的使用體驗。

在開發過程中，我們不僅學習了物聯網應用開發與嵌入式系統設計，也深入理解了藍牙通訊、感測技術與電子電路控制的整合應用。此外，透過解決 BLE 連線不穩定、感測數據異常與軟硬體整合挑戰，我們提升了問題解決能力，並學會運用程式開發與硬體調適來優化系統效能。

本作品未來可進一步拓展，如加入 Wi-Fi 遠端控制、雲端數據分析、語音控制等功能，提升系統的智慧化程度與市場應用價值。本專題不僅讓我們深化電子與程式設計的專業技能，也為未來進一步探索智慧家居與物聯網技術奠定了堅實的基礎。

## 捌、參考資料

《超圖解Arduino 互動設計入門》第四版 趙英傑 旗標科技股份有限公司

手機應用程式設計超簡單：App Inventor 2 零基礎入門班 鄧文淵-總監製；文淵閣工作室-編 出版社：碁峰資訊

<https://community.appinventor.mit.edu/t/app-to-turn-on-flashlight-no-extensions/137296>

<https://seeklogo.com/vector-logo/483005/arduino-ide>