

投稿類別：工程技術類

篇名：

紅外線萬能遙控器

作者：

溫勇威。臺北市立松山高級工農職業學校。資三仁
蔡孟勳。臺北市立松山高級工農職業學校。資三仁
蔡書田。臺北市立松山高級工農職業學校。資三仁

指導老師：

林敬堯老師

壹●前言

一、研究動機

科技越來越進步，紅外線的控制已被普遍運用在許多居家設備，每個人家中都必須使用許多遙控器來控制家電，這樣不僅容易造成混淆，也造成了更多資源的浪費，如果可以統整家中所有遙控器，只需透過手機 APP 就可以對所有設備進行控制，這樣不僅可以節省資源的浪費，還能透過 WiFi 隨時隨地進行控制，就能讓生活更加便利。

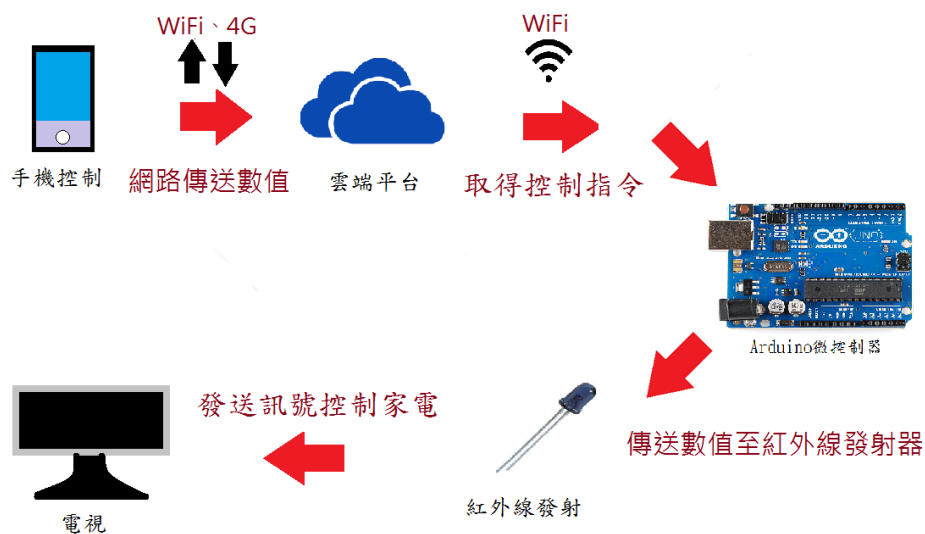
二、研究目的

紅外線控制技術不需要實體連線，使用上容易且成本低、耗電少，因而被廣泛運用在電器設備的控制當中，像是電視、風扇、冷氣、機上盒等等，但是受限於紅外線發射訊號的種類不同，每項電器控制都需要一台遙控器，容易造成混淆或是收納上比較麻煩，如果可以將所有遙控器整合在一起，只需透過手機 APP 就能達到所有控制功能，就能在居家生活上有非常大的幫助。

貳●正文

一、系統架構

本研究是以手機 APP 作為萬能遙控器，將訊息發送到網路 Server 端做儲存，再以 ESP8266 無線 WiFi 模組連接到 Server 端讀取訊息，當讀取到控制訊號時，就把訊號傳輸給微控制器，讓微控制器指示紅外線發射以控制家中設備。



【圖一】系統架構圖

二、硬體介紹

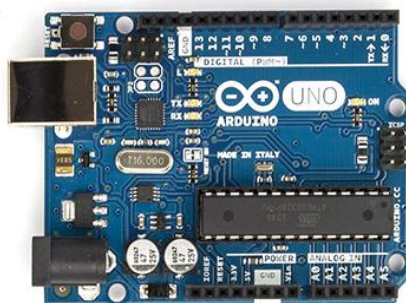
我們以 Arduino 開發版作微控制核心，結合紅外線收發功能，再以 ESP8266 作為連網的橋樑，希望能夠達到無遠弗屆的控制。本作品所使用的材料，如【表一】所示。

【表一】材料表

項目	數量
Arduino Uno 板	1
紅外線發射 LED	1
紅外線接收模組	1
ESP8266 WiFi 模組	1
LM2596 穩壓模組	1

(一) Arduino 單晶片

Arduino 是一個軟、硬體開放原始碼的單晶片微電腦，如【圖二】所示，為 Arduino 中的一個版本，使用了 Atmel AVR 單晶片，程式撰寫的語法類似 C 語言。



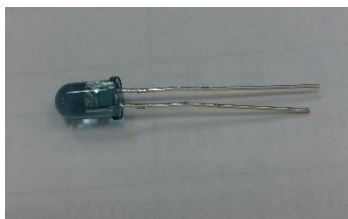
【圖二】Arduino UNO

(二) 紅外線發射 LED

紅外線 (Infrared, 簡稱 IR) 又俗稱紅色光芒，是波長介乎微波與可見光之間的電磁波，其波長在 760 奈米 (nm) 至 1 毫米 (mm) 之間，是波長比紅光長的非可見光，對應頻率約是在 430THz 到 300GHz 的範圍內，室溫下物體所發出的熱輻射多都在此波段。

紅外線發光二極體，如【圖三】所示，由紅外輻射效率高的材料 (常用砷化鎵 GaAs) 製成 PN 結，外加正向偏壓向 PN 結注入電流激發紅外光。光譜功率分佈為中心波長 830~950nm，半峰帶寬約 40nm 左右。

紅外線萬能遙控器



【圖三】紅外線發射 LED

(三) 紅外線接收模組

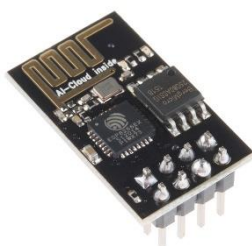
紅外線接收頭(又稱紅外線接收模組，IRM)是集成紅外線接收 PD 二極管、放大、濾波和比較器輸出等的 IC 模塊。我們不再製作接收放大電路，而是利用這模組簡化了電路，如【圖四】所示。



【圖四】紅外線接收模組

(四) ESP8266 WiFi 模組

ESP8266 是一款超低功耗的 UART-WiFi 透傳模組，擁有業內極富競爭力的封裝大小和超低能耗技術，專為移動裝置和物聯網應用設計，可將用戶的物理裝置連線到 WiFi 無線網路上，進行互聯網或區域網路通訊，如【圖五】所示。



【圖五】ESP8266 ESP-01

(五) LM2596 穩壓模組

LM2596 穩壓模組是可調的降壓模組，藉由改變模組上面的可調電阻來改變輸出電壓。輸入輸出壓差為 2V，最小可以輸出 1.25V 電壓。我們以這模組將總輸入的 5V 轉為 3V 供 ESP8266 使用，如【圖六】所示。

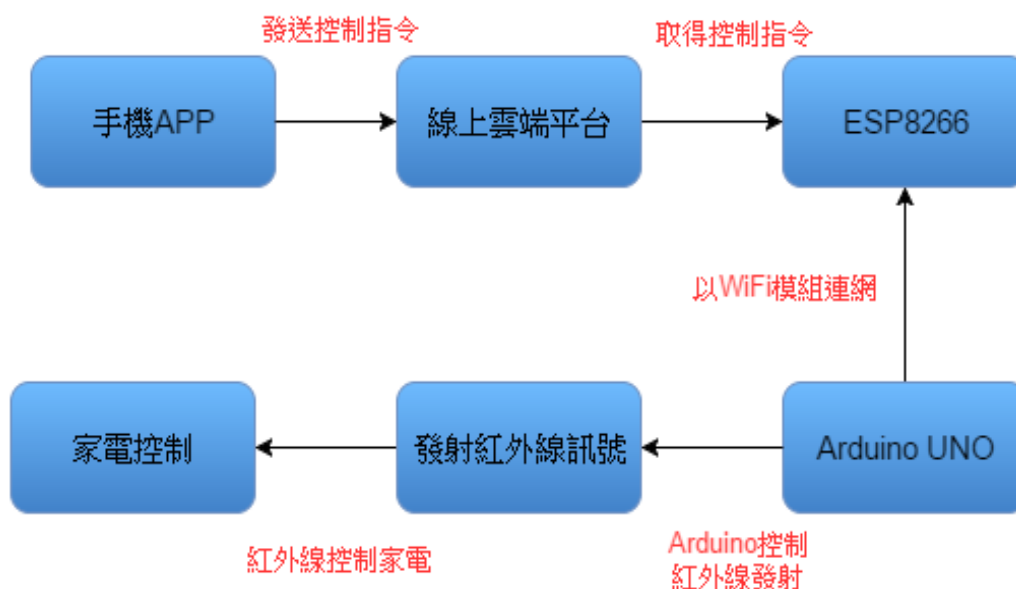
紅外線萬能遙控器



【圖六】 LM2596 穩壓模組

三、軟體核心

透過手機發送控制指令到線上雲端平台，再讓 Arduino 微控制器透過 ESP8266 這顆 WiFi 模組連網，取得線上雲端平台的控制指令，再依照指令對家電發射紅外線控制，架構圖如【圖七】所示。



【圖七】 軟體架構圖

(一) 開發介面

本研究所使用的開發軟體如【表二】所示。

【表二】 開發軟體

軟體名稱	版本
Arduino IDE	1.6.5
Eclipse Neon	4.6.1

1、Arduino Software IDE

Arduino Software IDE 是由 arduino.cc 所提供的一套以 Java 編寫的跨平台開發軟體，使用與 C 語言相似的程式語法，並且提供了包含常見的輸入/輸出函式的 Wiring 軟體函式庫。它常被用來介紹程式編寫給藝術家和不熟悉程式設計的人們，且包含了一個擁有語法突顯、括號匹配、自動縮排和一鍵編譯，並將執行檔燒寫入 Arduino 硬體中的編輯器。



【圖八】 Arduino 軟體開發環境

2、Eclipse

Eclipse 是由 Open Source Community 所創建的開發軟體，並被廣泛地使用在許多不同的領域。例如：可作為 Java 應用程式與 AndroidApp 的開發環境。大多數人都知道 Eclipse 是 Java 的 Integrated Development Environment (IDE)。至今，它已佔據超過全世界 Java 開發環境市場的 50%。Eclipse 計畫是由 Eclipse 基金會所管理，Eclipse 基金會是一個非營利性的組織，大多由以科技公司為背景的成員所組成。



【圖九】 Eclipse 開發環境

(二) 程式碼

1、紅外線收發程式碼：

我們先由紅外線接收器來解碼原本遙控器的紅外線，之後再藉由紅外線 LED 來發射所解碼出的編碼，藉這實驗可以看出只要解碼出原本遙控器的紅外線，再用程式的方式作儲存，之後整合成一個遙控器發射裝置，就能取代現有的所有遙控器。以下是接收紅外線解碼的程式，如【圖十】所示，以及紅外線發射程式，如【圖十一】所示。

紅外線萬能遙控器

```
void loop() {  
  if (irrecv.decode(&results)) {  
    Serial.println(results.value, HEX);  
    dump(&results);  
    irrecv.resume(); // Receive the next value  
  }  
}
```

【圖十】 紅外線接收解碼程式片段

```
void loop() {  
  int d;  
  int kbs = 32; // 38kHz carrier frequency for the NEC protocol  
  if( (d = Serial.read()) != -1) { // 讀取序列埠  
    unsigned long v = 0x0;  
    unsigned int irSignal[24];  
    unsigned int irSignal1[] = {1200,400,1300,350,450,1200,1300,400,1250,400,450,1200,  
    unsigned int irSignal2[] = {1250,400,1300,400,400,1250,1250,400,1300,400,400,1250,  
    unsigned int irSignal3[] = {1250,450,1250,400,450,1250,1250,400,1250,400,400,1300,  
    unsigned int irSignal4[] = {1300,400,1200,450,450,1200,1250,450,1250,400,400,1250,  
    switch(d) { // 根據讀取到的資料，換成代表六個按鈕的紅外線編碼  
      case '1':  
        v = 0x1;  
        break;  
      case '2':  
        v = 0x2;  
        break;  
      case '3':  
        v = 0x3;  
        break;  
      case '4':  
        v = 0x4;  
        break;  
      case '5':  
        v = 0x5;  
        break;  
      case '6':  
        v = 0x6;  
        break;  
      case '7':  
        v = 0x7;  
        break;  
      case '8':  
        v = 0x8;  
        break;  
      case '9':  
        v = 0x9;  
        break;  
      case '0':  
        v = 0x0;  
        break;  
      case 'A':  
        v = 0xA;  
        break;  
      case 'B':  
        v = 0xB;  
        break;  
      case 'C':  
        v = 0xC;  
        break;  
      case 'D':  
        v = 0xD;  
        break;  
      case 'E':  
        v = 0xE;  
        break;  
      case 'F':  
        v = 0xF;  
        break;  
    }  
  }  
}
```

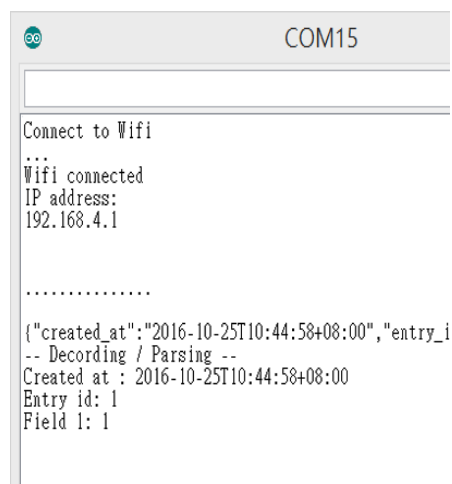
【圖十一】 紅外線發射程式片段

2、WiFi 程式碼：

手機 APP 將控制訊號上傳到線上雲端平台後，我們讓 Arduino 利用 ESP8266 接收雲端平台上的控制訊號，這樣就能不受距離限制的控制設備。程式碼，如【圖十二】所示，透過 Arduino 序列觀看讀取下來的資料，如【圖十三】所示。

```
String getStr = GET + " HTTP/1.1\r\n";  
client.print( getStr );  
client.print( "Host: api.thingspeak.com\r\n" );  
client.print( "Connection: close\r\n\r\n" );  
  
delay(10);  
  
// 讀取所有從 ThingSpeak IoT Server 的回應並輸出到序列埠  
String section="HEAD";  
while(client.available())  
{  
  String line = client.readStringUntil('\r');  
  
  /** parsing the JSON response here ! **/  
  // parse the HTML body  
  if(section == "HEAD" ) // HEAD
```

【圖十二】 讀取數值的程式碼



【圖十三】 Arduino 序列讀取到的數值

(三) 手機 APP

以下是我們的 APP 控制介面，透過手機作為控制端，能夠將紅外線解碼的結果儲存之後自由的定義按鍵說明，如【圖十四】所示。

紅外線萬能遙控器

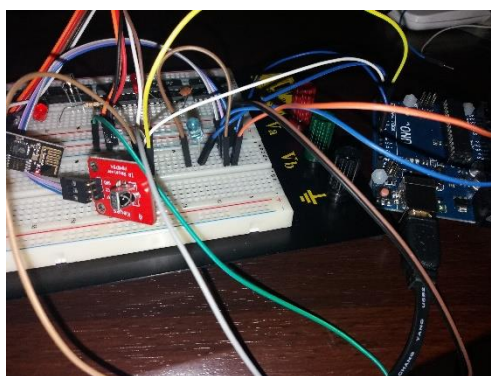


【圖十四】 APP 人機介面

四、研究成果

(一) 模擬電路

本作品的模擬版電路如【圖十五】所示，由左至右分別為 ESP8266、紅外線接收模組、藍色紅外線發射 LED 及 Arduino 微控制器。



【圖十五】 模擬電路

(二) 學習遙控器的紅外線控制訊號

首先我們要透過紅外線接收模組來解析遙控器控制訊號如【圖十六】所示，等解析完遙控器的訊號後，就能透過 ESP8266 來傳送解碼的訊號到我們的手機，並將接收到的訊號顯示在 APP 上方，如【圖十七】所示。

紅外線萬能遙控器



【圖十六】進行紅外線訊號解析



【圖十七】顯示接收到的訊號

(三) 朝向設備進行控制

將我們的模擬版電路朝向家中的風扇，並透過手機將控制訊號傳至雲端，再用 Arduino 讀取雲端上的訊號以做控制。風扇開啟與關閉功能，如【圖十八】、【圖十九】所示。



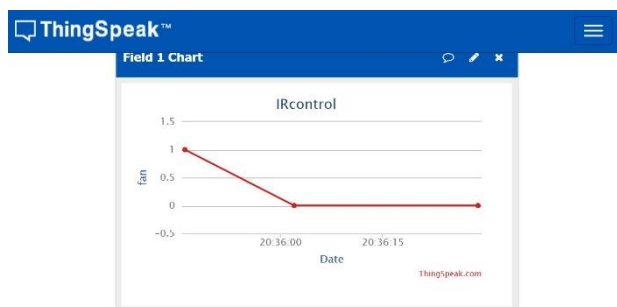
【圖十八】紅外線控制設備(開啟)



【圖十九】紅外線控制設備(關閉)

(四) 雲端監控使用狀況

藉由 APP 連線至雲端平台可以看到風扇控制的狀態，以及監控開關的運作時間，並且將家中設備的使用狀況上傳到線上雲端平台，方便使用者隨時隨地監控設備，如【圖二十】所示。



【圖二十】 控制訊號雲端紀錄(1=開啟，0=關閉)

參●結論

一、研究發現及心得

由於日常生活中常使用到遙控器，原以為不會太過困難，經過這次研究發現原來紅外線也有分許多協定，像是 NEC、RC5、RC6 等協定，所以造成在解析紅外線訊號時，程式的撰寫有許多的不便，之後我們發現有辦法以數值的方式呈現紅外線完整的載波，使我們在研究進度上有了大幅的成長。

在製作這項研究的過程中，雖然不斷的碰到問題，但是經過反覆的查詢資料、翻閱書籍、參考別人的論文，重複的進行嘗試，並試著解決所碰到的困難，最終才能有如此的成果，也讓我們更加瞭解日常生活中所使用到的紅外線相關原理。

二、未來展望

目前我們的作品受限於紅外線發射角度的問題，造成控制時十分的不穩定，未來我們希望將紅外線變成朝向四周發射，讓這作品不再只是遙控器，而搖身變為一座基地台，只要放在家中便能對所有紅外線設備進行控制，同時還能學習各種不同的遙控器控制訊號，以達到更加完善的控制，讓居家生活更加便利。

由於資料庫的撰寫比較複雜，目前我們在接收完紅外線訊號後，必須透過網路上的雲端網站，才能將訊號傳輸至 APP 端儲存，造成額外的記憶體占用，將來希望可以克服這項困難，透過架設資料庫網站的方式儲存數據，變成能夠自行新增及刪減紅外線的控制訊號，並統計長時間的使用數據，讓使用者更加明白家中電器的使用狀況。

肆●引註資料

- 註一、林弘偉（2016）。**架構於無線網路之可學習與可擴充式紅外線遙控系統**。國立臺北科技大學碩士論文。
- 註二、李奇峰（2012）。**使用微控制器實作可搭配 Android 手機遙控之萬用遙控器**。南台科技大學碩士論文。
- 註三、怯婷芳（2012）。**實作一個具備情境感知與個人化使用介面框架之萬用遙控器**。輔仁大學碩士論文。
- 註四、楊明豐（2014）。**Arduino 最佳入門與應用--打造互動設計輕鬆學**。台北市：碁峰資訊出版社。
- 註五、曾吉弘（譯）（2015）。**實戰數位家庭自動化 | 使用 Arduino**。台北市：碁峰資訊出版社。
- 註六、孫駿榮、吳明展、盧聰勇（2013）。**最簡單的互動設計 Arduino 一試就上手(第二版)**。臺北市：碁峯資訊股份有限公司。
- 註七、Martin Evans、Joshua Noble、Jordan Hochenbaum（2014）。王冠勛/譯；陳錦輝/審校。**Arduino 完全實戰手冊(Arduino in action)**。新北市：博碩文化股份有限公司。
- 註八、宮本信二（2014）。**Eclipse 完全攻略(第二版)**。新北市：博碩文化股份有限公司。
- 註九、Cooper Maa，紅外線遙控原理與 NEC IR Protocol。取自於：
<http://coopermaa2nd.blogspot.tw/2010/01/nec-ir-protocol.html>
- 註十、遊戲自己做，打造 WiFi 遙控車(之二)：整合 ESP8266 與 Arduino。取自於：
<http://lets-make-games.blogspot.tw/2015/05/wifi-esp8266-arduino.html>