

2024仰望盃全國科學 HomeRun 實作大賽

決賽成果報告書

隊伍名稱： 星際探險家

作品名稱： 薄荷冰炫風

科學概念2： 電流熱效應

它的運作原理是利用金屬導體的電阻 (Resistance) 特性來發熱，當電流通過一金屬導體時，該金屬的電阻會將部分電流以熱能的形式消耗掉；在持續不斷的過電下，熱能會開始累積，最後造成發熱的現象。

科學概念2： 「風冷溫度」，可以利用溫度計來估算風速。測量實際溫度： 使用一個正確校準的溫度計來測量當前的實際溫度。查找風冷指數 (Wind Chill Index) 表： 風冷指數是一種評估風速對人體感覺的影響的指標。通常，這個指數是在特定溫度和風速條件下估算的。查找風冷指數表，可以在表中找到對應的風速和實際溫度的數值。

決賽成果報告書內文

1. 發想動機：

在寒冷的冬天，當室外冷風颼颼的時候，我們常常遇到開窗時太冷，關窗卻又太悶的狀況，甚至在班上同學會為此而產生口角紛爭。當理化課上過溫度與熱的單元後，因此有了研發新裝置的構想，用科學方法解決這個問題，讓教室的空氣既可保持新鮮又不至於太冷！

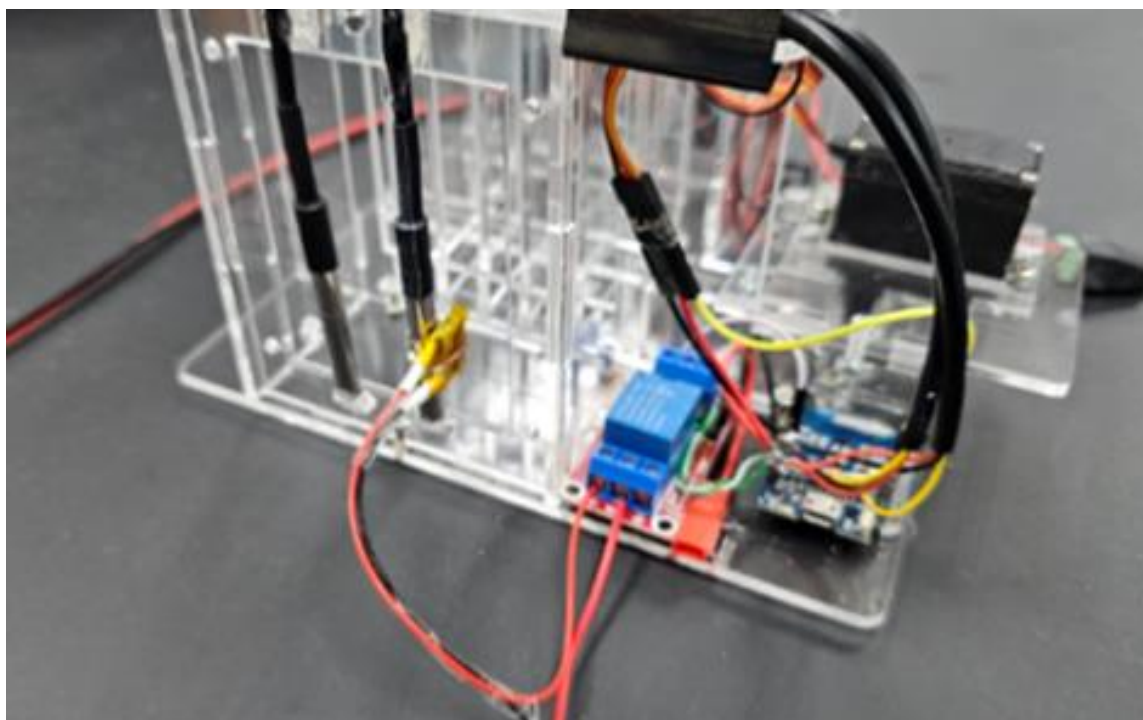
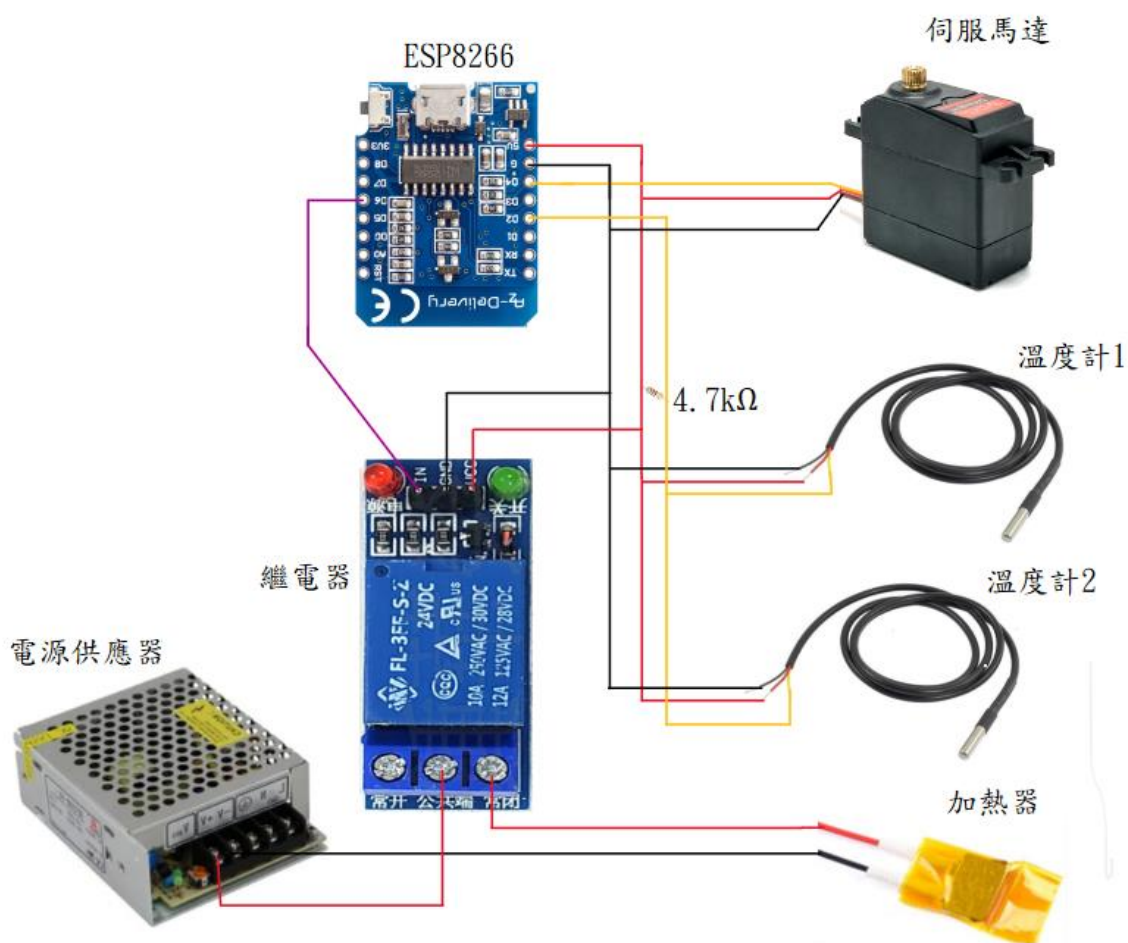
2. 作品創意性：

要兼顧空氣新鮮度與避免冷風大量灌入教室使溫度驟降，我們想偵測戶外的風速與氣溫，打造一個可以控制進風量的自動百葉窗。

而測量風速有許多方式，我們採用一種失溫法來推算風速，因為冷風會帶走溫度計的熱量，利用兩隻溫度計做實驗對照，在一隻與利用微型加熱器加熱與受冷風吹的溫度變化，可以由溫度變化速率推算風速。

根據實驗數據，找出溫度計的溫度變化速率與風速關係式，取代風速計，如此同時測氣溫與風速，進而調節窗戶的開闔大小，這是具有綠能環保意識且十分富有創意的裝置。

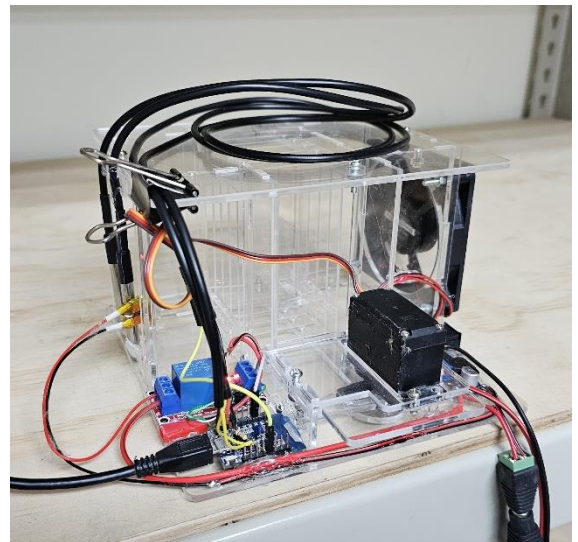
3. 硬體及電路架構圖：



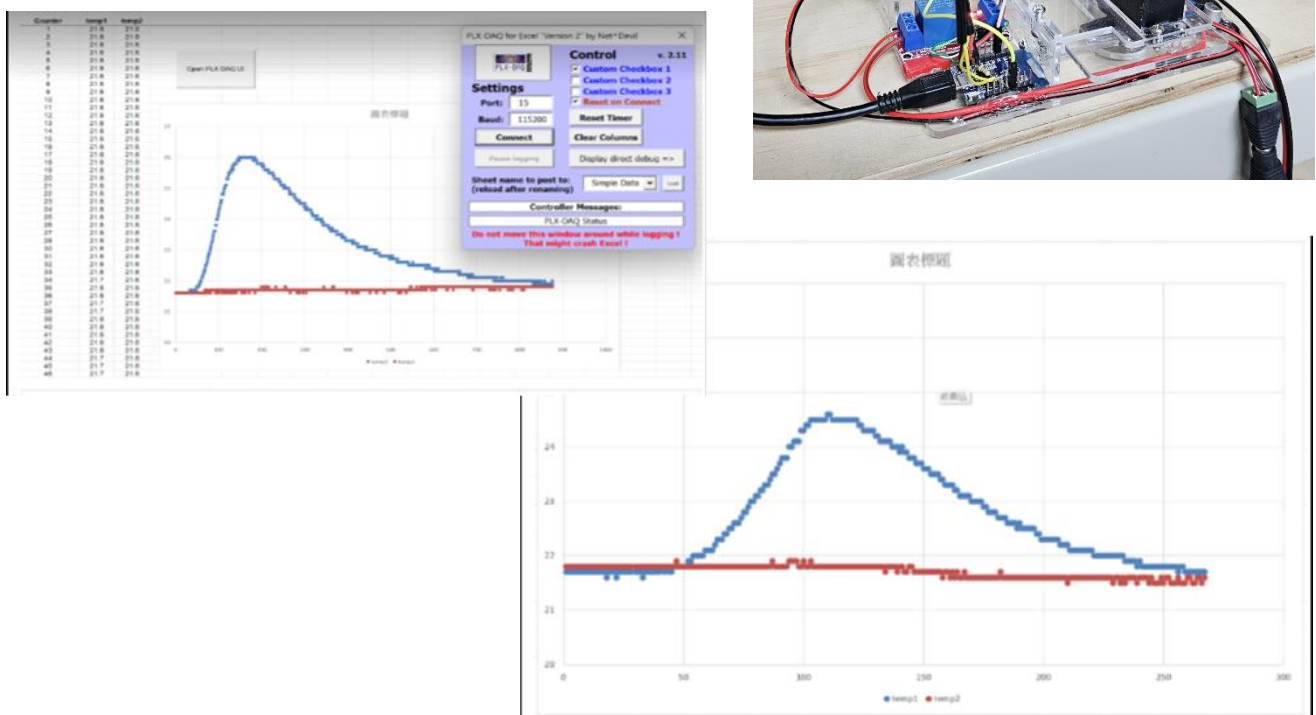
4. 作品成果報告：

這個專案旨在設計並製作一個自動控制系統，可以根據兩個環境溫度感測器與加熱器的數據來推算在不同溫度下的風速，進而找出冷風溫度與風速的關係式，並且撰寫程式透過開發板連結伺服馬達自動調整百葉窗的開合程度，調整室溫和舒適度。

[右圖]為經過數次修改而設計出來的研究裝置，將上述硬體安裝到相關位置。



[下圖]為 EXCEL 中收集數據之介面：



這個裝置是利用 ESP8266開發板：連接兩隻數位溫度計，並控制繼電器，在適當條件對加熱片通電加熱一隻溫度計，量測升溫之速率，到達設定條件時再斷電，讓其降溫，繼續記錄溫度變化。藉由序列埠連接至電腦，在 EXCEL 中收集相關數據進行分析研究與討論。

下面附上其中一段用來測試的程式碼(在 Arduino IDE 中撰寫編譯)

//ESP8266 + ds18b20 Sensor 測試不同溫度或不同風速下[繼電器通電加熱]之溫度變化

```
#include <OneWire.h>
#include <DallasTemperature.h>
#define One_Wire_Bus D6 //溫度計的腳位
OneWire oneWire(One_Wire_Bus);
DallasTemperature sensors(&oneWire);

#include <Servo.h>
#define Servo_Pin D5 //伺服馬達的腳位
Servo servo;

#define Relay_Pin D1 //繼電器的腳位

int temp_1 = 0;
int temp_2 = 0;
float real_temp_1 = 0.0;
float real_temp_2 = 0.0;
float temp_offset_1 = 0.0; //調整誤差值
float temp_offset_2 = 0.0; //調整誤差值

int gogo = 0; //為了執行加熱程序的預設值
int delay_time = 434; //設定時間差來調整讀取時間間隔
int temp_no = 0;
float heat_up_temp = 0.5; //預定加熱到室溫加 x 度
int start_d = 86; //百葉窗起始開闔角度
int dd = 0; //百葉窗實驗開闔角度

void setup() {
  Serial.begin(115200);
  pinMode(Relay_Pin, OUTPUT);
  digitalWrite(Relay_Pin, LOW);
  sensors.begin();
  sensors.setResolution(12);
  servo.attach(Servo_Pin, 500, 2500);
  servo.write(start_d + dd);
  Serial.println("CLEAR SHEET"); //clears sheet starting at row 1
  Serial.println("LABEL, Date, Time, Timer, Counter, temp1, temp2");
}

void loop() {

  static unsigned long timer = millis();
  if (millis() - timer > delay_time) { // 每讀一筆資料，大約間隔1秒，由
delay_Time 進行調整
    read_Temp();
```

(接下頁)

```

if (gogo == 0) {
  if (real_temp_2 < real_temp_1 + heat_up_temp) {
    digitalWrite(Relay_Pin, HIGH);
    Serial.println("Start Heat");
  } else if (real_temp_2 > real_temp_1 + heat_up_temp) {
    digitalWrite(Relay_Pin, LOW);
    Serial.println("Heat End");
    gogo = 9; //結束加熱程序
  }
}

if (gogo == 9) {
  if (real_temp_2 <= real_temp_1 + heat_up_temp * 0.5) {
    gogo = 0;
  }
}

Serial.print("DATA, DATE, TIME");
Serial.print(",");
Serial.print(millis() - timer);
Serial.print(",");
Serial.print(temp_no);
Serial.print(",");
Serial.print(real_temp_1, 1);
Serial.print(",");
Serial.println(real_temp_2, 1);
timer = millis();

}
}

//讀取溫度計的副程式
String read_Temp() {

  sensors.requestTemperatures();
  temp_1 = (sensors.getTempCByIndex(0) + temp_offset_1) * 10;
  temp_2 = (sensors.getTempCByIndex(1) + temp_offset_2) * 10;
  real_temp_1 = temp_1 / 10.0;
  real_temp_2 = temp_2 / 10.0;
  temp_no++;
  return String(real_temp_1, real_temp_2);

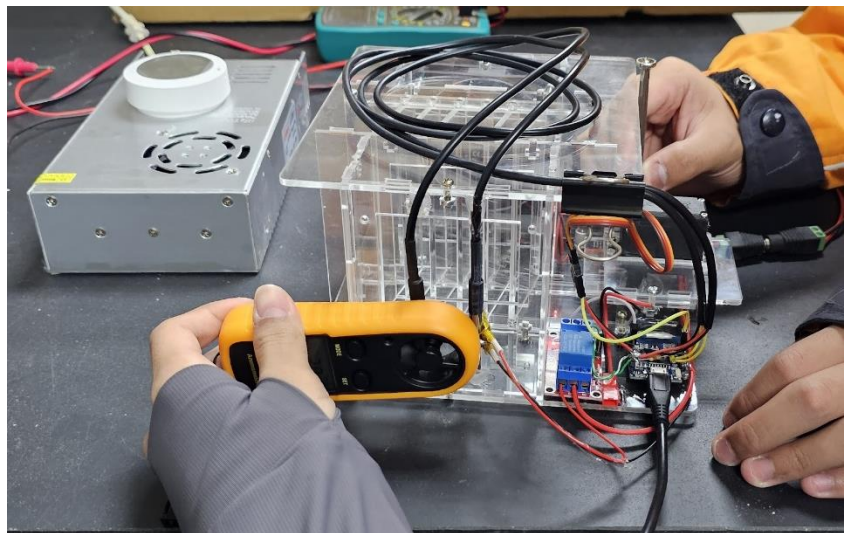
}

```

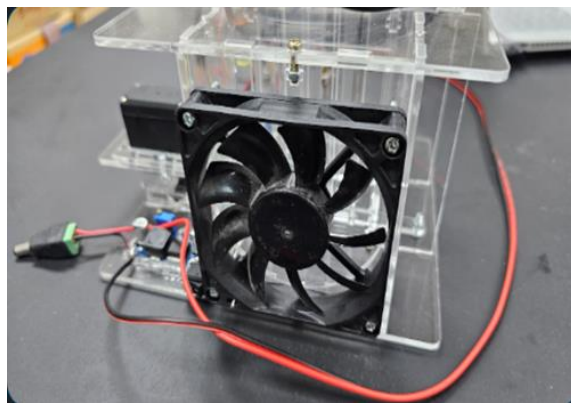

我們先在不同的環境溫度下量測溫度計的誤差，然後利用小風扇的風速去測試加熱與散熱速率，以這兩個系列的數據去回推風速，最後以這個公式來控制百葉窗的開闔，葉片啟閉大小由氣溫與風速決定，讓室內維持較為舒適的環境，希望能廣泛的運用在生活中，提供大眾生活便利與環保節能。

實驗步驟：

1. 根據設計圖，製作實驗裝置(3D繪圖與壓克力雷切)
2. 兩支溫度計的溫度差：我們以 arduino 程式編輯程式碼並輸入硬體作品中用於感測兩支溫度計的溫度，然後以其中一隻為標準，將另一支替加熱片的溫度計測出來的溫度加減平均溫差當成實驗值。
3. 測量小風扇的風速：將風速計置於溫度計後，來測量出小風扇的風速，小風扇的轉速是由輸入的電壓控制，我們分別測試輸入電壓為 5V、6V、7V、8V。
4. 測試風速與百葉窗開闔角度之關係式：我們先定義全開為零度，利用程式碼將伺服馬達調整到預設轉動角度，並以每轉15度測量一組數據，我們分別測試角度為 0° 、 15° 、 30° 、 45° 、 60° 、 75° (最大值，葉片已全關閉)。
5. 測試加熱升溫與散熱降溫之速率：(因素一)在無風與有風的，(因素二)當下氣溫之溫度，我們討論出幾種不同模式：(策略一)固定加熱相同時間，收集溫度變化，(策略二)固定加熱至相同溫度，收集溫度變化，(策略三)固定加熱到比參照溫度計高幾度，收集溫度變化。



6. 由數據回推不同溫度下，風速與溫度變化之函數公式：利用上述之方法應可研究出風速與變因之關聯，利用多次的實驗數據可以算出時間、溫度變化與風速的函數公式。



7. 函數公式嵌入硬體設備：將上述步驟中推導之函數公式融入到 Arduino 控制程式中，讓此系統能依建立的模型自動運作。

實驗結果：

1. 兩隻溫度計的溫度校正：兩隻數位溫度計大致相差 0.1°C ，藉由程式碼定義兩隻溫度計為 temp_1 和 temp_2

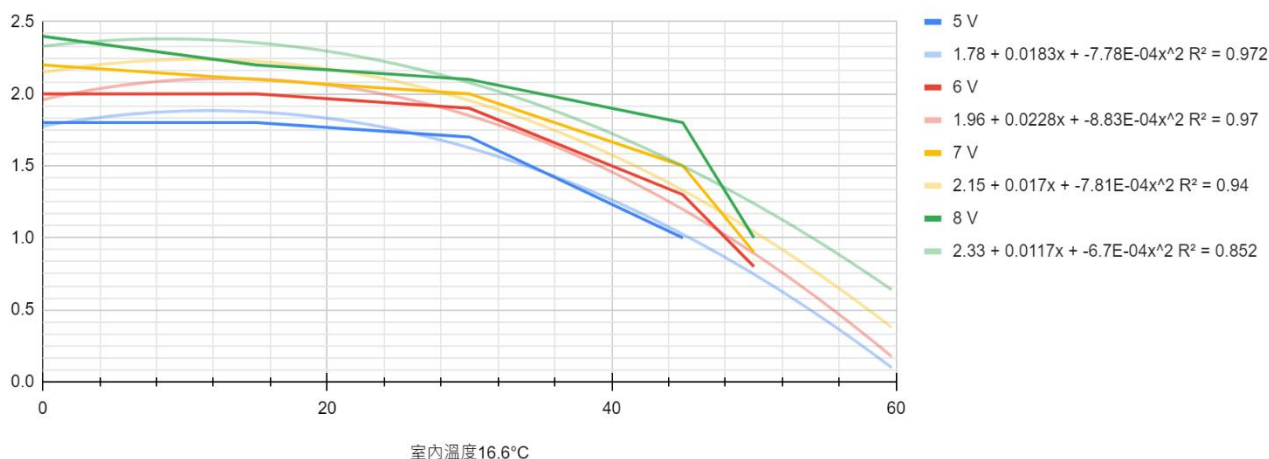
(實驗中： $\text{temp}_2 = \text{temp}_1 + 0.1$)

2. 小風扇預設電壓值以及百葉窗開闔角度的風速關係：在市售簡易風速計(右圖)的測量下得知風速(m/s)：

室內溫度 16.6°C	風扇電壓 5 V	風扇電壓 6 V	風扇電壓 7 V	風扇電壓 8 V
百葉窗開合 0°	1.8	2	2.2	2.4
百葉窗開合 15°	1.8	2	2.1	2.2
百葉窗開合 30°	1.7	1.9	2	2.1
百葉窗開合 45°	1	1.3	1.5	1.8
百葉窗開合 50°	0	0.8	0.9	1
百葉窗開合 $>50^{\circ}$	0	0	0	0



風扇電壓與風速關係圖



3. 計算風速與百葉窗開闔角度之關係式：

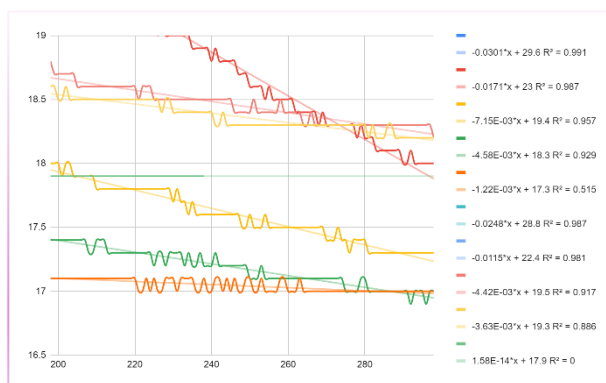
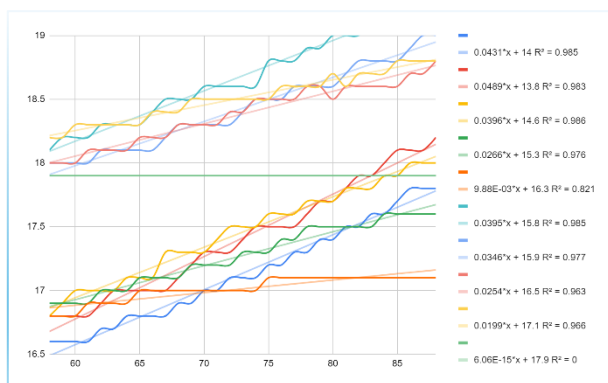
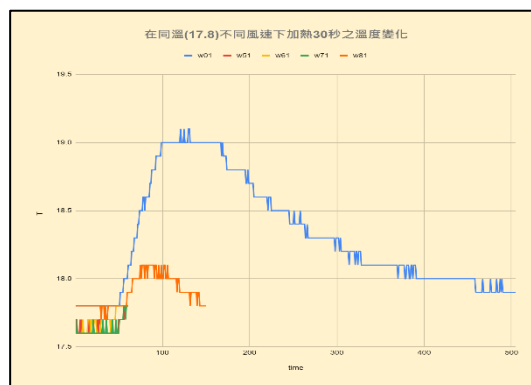
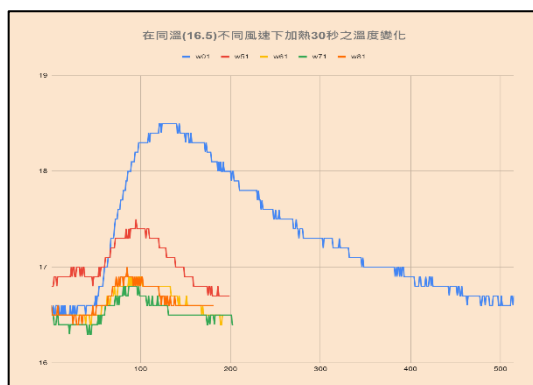
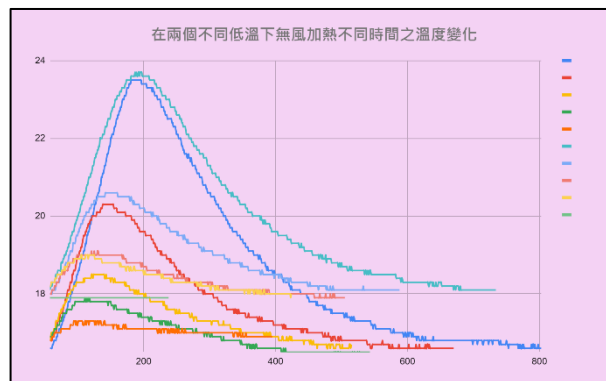
設 y 為固定風速(*我們要的風速)

設 x 為我們測得的風速 $(x-y)/0.23*15=$ 開闔角度

4. 加熱升溫與散熱降溫研究

如右圖及下圖(方程式在圖例說明)

取用適當的時間間隔求出過程中之線性關係式(以斜率為主要數據)



5. 回推風速的函數公式：依據上圖的吸熱與散熱過程斜率我們可以知道散熱在有無風的狀況下散熱需時間並且利用斜率可以回推算出風速。

6. 最後將函數公式嵌入硬體程式中，並驗證可行：我們假設在16度時風速為0.5m/s 為舒適基準。

實驗討論：

1. 時間溫度差和風速的關係是我們實驗的主軸：通過使用兩支溫度計，一隻有加熱片來實踐溫度變化，進一步找出與風速的關係。雖然最後使用兩支溫度計來量測溫度差，但過程有使用市售風速計來測量風扇的風速，風扇模擬自然風對於散熱速率的影響。這也顯示了溫度和風速之間的關係。
2. 百葉窗開合對風速的影響：我們進一步測試了百葉窗的開合對風速的影響，這有助於我們理解在不同條件下控制進入室內的風速。
3. 吸熱與放熱的評估：通過圖形可觀察出斜率變化，並利用 EXCEL 軟體可協助我們可以計算出線條方程式與斜率值。這有助於我們找出風速與時間溫差之關係式。
4. 研究環境是在密閉帶有分離式冷氣的教室中，利用恆溫空調來模擬不同寒冷天氣的狀態，但還是有些不穩定的因子，避開冷氣出風口及變頻冷氣溫度還是會有些許浮動。
5. 函數公式的應用：最後我們將實驗數據轉換為函數公式，以便將其嵌入到 Arduino 程式中，但溫度數據會因取用的時間範圍有不同斜率，推導的函數公式就會不同，但我們覺得人們也未必敏感到能察覺，且冷熱感受原本就因人而異，因此忽略此變數的影響。

總而言之，這個實驗培養了我們細心的洞察力，有助於我們理解風速、吸熱與放熱效率的影響，以及如何使用電控裝置量測記錄溫度差，並使用風速計來輔助校正風速關係式。雖然實驗的時候產生不少迷失，有許多第一次的經驗都在老師耐心指導下完成，這是一個讓我們難忘的探究與實作過程！

5. 參考文獻：

a. 電路板的參考

ESP8266：<https://zh.wikipedia.org/zh-tw/ESP8266%E5%BC%80%E5%8F%91%E6%9D%BF>

SERVO：<https://zh.wikipedia.org/zh-tw/Servo>

繼電器：<https://zh.wikipedia.org/zh-tw/%E7%BB%A7%E7%94%B5%E5%99%A8>

ds18b20溫度計：<https://shop.mirotek.com.tw/arduino/arduino-adv-1/>

b. 風寒指數：

<https://zh.wikipedia.org/zhtw/%E9%A2%A8%E5%AF%92%E6%8C%87%E6%95%B8>

c. 風速器模組(Wind Sensor Rev. C)

<https://shop.playrobot.com/products/sensor-dg0753>

d. 手持式風速計(Anemometer) <https://www.amazon.com/->

[/zh_TW/Anemometer-](#)

[%E6%89%8B%E6%8C%81%E5%BC%8F%E9%A2%A8%E9%80%9F%E8%A8%88-](#)

[%E7%94%A8%E6%](#)