

# 國立中興大學資訊工程學系

## 資訊專題競賽報告

### 基於強化學習的智能風扇及應用

#### Smart Fan Based on Reinforcement Learning and Its Applications

專題題目說明、價值與貢獻自評（限100字內）：

本專題藉由強化學習當中的Q-learning來訓練風扇，讓風扇可以不斷地進行自我學習和優化，達到能根據使用者的體溫自動調整風速，最終能讓使用者維持在最合適的溫度，並提高舒適度。

專題隊員：

姓名	E-mail	負責項目說明	專題內貢獻度(%)
施昶宇	chang001124gmail.com	蒐集資料、實作	50%
陳政鴻	Angel65020765@gmail.com	蒐集資料、實作	50%

【說明】上述表格之專題內貢獻度累計需等於100%。

指導教授簡述及簡評：

指導教授簽名：

中 華 民 國 年 月 日

國立中興大學  
資訊科學與工程學系  
專題

基於強化學習的智能風扇及應用

Smart Fan Based on Reinforcement Learning and Its Applications

指導教授:黃德成

組員:施昶宇、陳政鴻

# 目錄

圖目錄.....	III
表目錄.....	V
一、 摘要.....	1
二、 專題研究動機與目的.....	1
三、 專題重要貢獻.....	1
四、 團隊合作方式.....	2
五、 設計原理、研究方法與步驟.....	3
5.1 機器學習原理.....	3
5.2 Q-learning 原理.....	4
5.3 藉由 PWM 控制風扇轉速.....	5
5.4 選擇強化學習— Q-learning.....	6
5.5 設計主要的訓練演算法.....	7
六、 系統實現與實驗.....	8
6.1 硬體方面.....	8
6.1.1 Arduino Mega2560.....	8
6.1.2 ESP8266.....	9
6.1.3 L298N.....	10
6.1.4 DS18B20.....	11

6.2 風扇和各個模組的實際接線圖.....	12
6.3 軟體方面.....	13
6.3.1 使用者登入.....	13
6.3.2 ESP8266 連上 WIFI、TCP 連線.....	13
6.3.3 連線資料庫取得資料.....	13
6.3.4 用 q-learning 開始訓練.....	14
6.3.5 訓練結果更新資料庫.....	14
6.3.6 整體流程圖.....	14
6.4 實驗設計.....	15
七、效能評估與成果.....	16
八、結論.....	17
九、參考文獻.....	18

# 圖目錄

圖 5.1.....	3
圖 5.2.....	4
圖 5.3.....	4
圖 5.4.....	5
圖 5.5.....	5
圖 5.6.....	7
圖 6.1.....	8
圖 6.2.....	9
圖 6.3.....	9
圖 6.4.....	10
圖 6.5.....	11
圖 6.6.....	12
圖 6.7.....	12
圖 6.8.....	13
圖 6.9.....	13
圖 6.10.....	14
圖 6.11.....	15
圖 7.1.....	16

圖 7.2.....	16
圖 7.3.....	17
圖 7.4.....	17

# 表目錄

表 4.1.....	2
表 5.1.....	6
表 6.1.....	10
表 6.2.....	11

## 一、摘要

一般市面上的風扇只能固定轉速，並且沒有考慮人體溫度，缺乏便利和舒適性，為了解決這個問題，我們使用 Q-Learning 結合電風扇，實現出個人化的電扇，並利用 ESP8266 將風扇連上 WIFI，建立與資料庫的連線，將 q\_table 上傳至資料庫儲存，讓下一位使用者登入時，能夠從資料庫存取自己的 q\_table，經過長時間的訓練之後，能夠根據使用者狀態自動調整到最佳的狀態。

關鍵詞: Reinforcement learning、智能風扇、ESP8266、資料庫。

## 二、專題研究動機與目的

電風扇是日常生活必備的家電，但是市面上大多僅考慮環境溫度而沒有考慮使用者體溫的風扇。夏天在冷氣房時，一般風扇只能固定風速，剛開始覺得很熱，但因熟睡後風速不變，體溫下降，經常因為冷醒才把電扇關掉，十分缺乏便利和舒適性。因此我們希望能設計一個能根據人體溫度自動調整風速的電扇，來讓使用者能維持在最舒適的溫度。

## 三、專題重要貢獻

本次專題實作之風扇，利用 q-learning 找到每個溫度下的最適風速，並且將 q-table 上傳資料庫，使用者登入後，就能藉由 ESP8266 連線 wifi 從資料庫中下載相應的 q-table，達到個人化風扇的效果。



## 四、團隊合作方式

表 4.1 是我們團隊的合作方式，分為三個階段，前期主要以查詢資料，決定方向為主，中期完成硬體的配置，以及軟體實作，後其統整所有資料及製作實驗數據。

表 4.1 團隊合作方式

前期	<ol style="list-style-type: none"><li>1. 決定專題題目</li><li>2. 閱讀相關論文</li><li>3. 蒐集相關資料</li></ol>
中期	<ol style="list-style-type: none"><li>1. 完成 q-learning 的設計</li><li>2. 完成硬體的配置</li><li>3. 實現登入頁面、將風扇連上資料庫</li></ol>
後期	<ol style="list-style-type: none"><li>1. 評估數據</li><li>2. 調整參數</li><li>3. 撰寫報告</li></ol>

## 五、設計原理、研究方法與步驟

### 5.1 機器學習原理

機器學習是一門開發演算法和統計模型的科學，旨在讓電腦從大量的數據中學習並自主地進行預測、分類、集群等任務，而不需要明確的程序或規則。主要分成三大類，監督式學習、非監督式學習和強化學習三種類型，如圖 5.1。

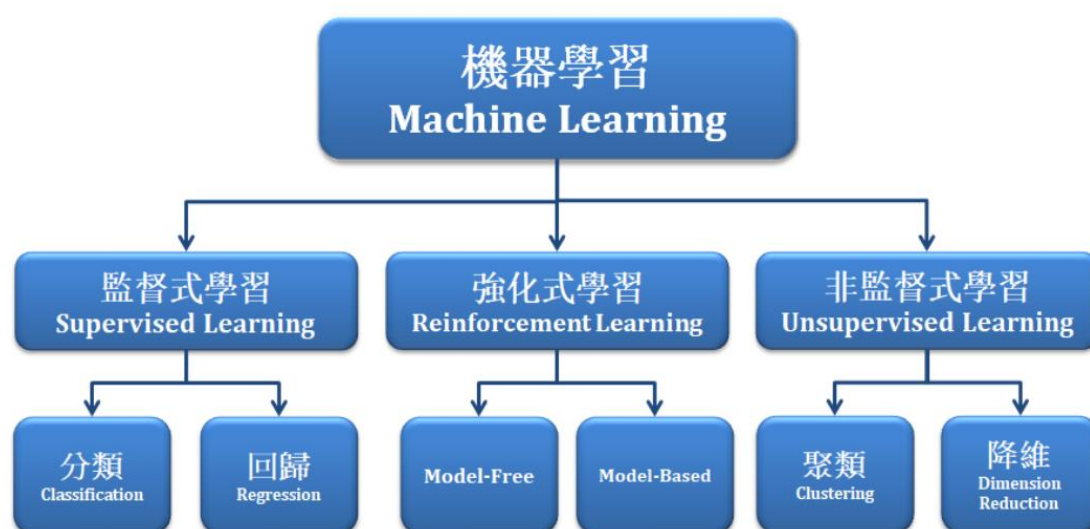


圖 5.1 機器學習架構圖[12]

本次專題主要利用其中的強化學習，在強化學習中，agent 通過觀察環境的狀態，進行行動，並根據環境的反饋學習最佳行動策略。強化學習不需要正確的輸出輸入，而是強調如何基於環境而行動，以取得最大化的預期利益，十分符合我們專題所需，圖 5.2 為題例說明。

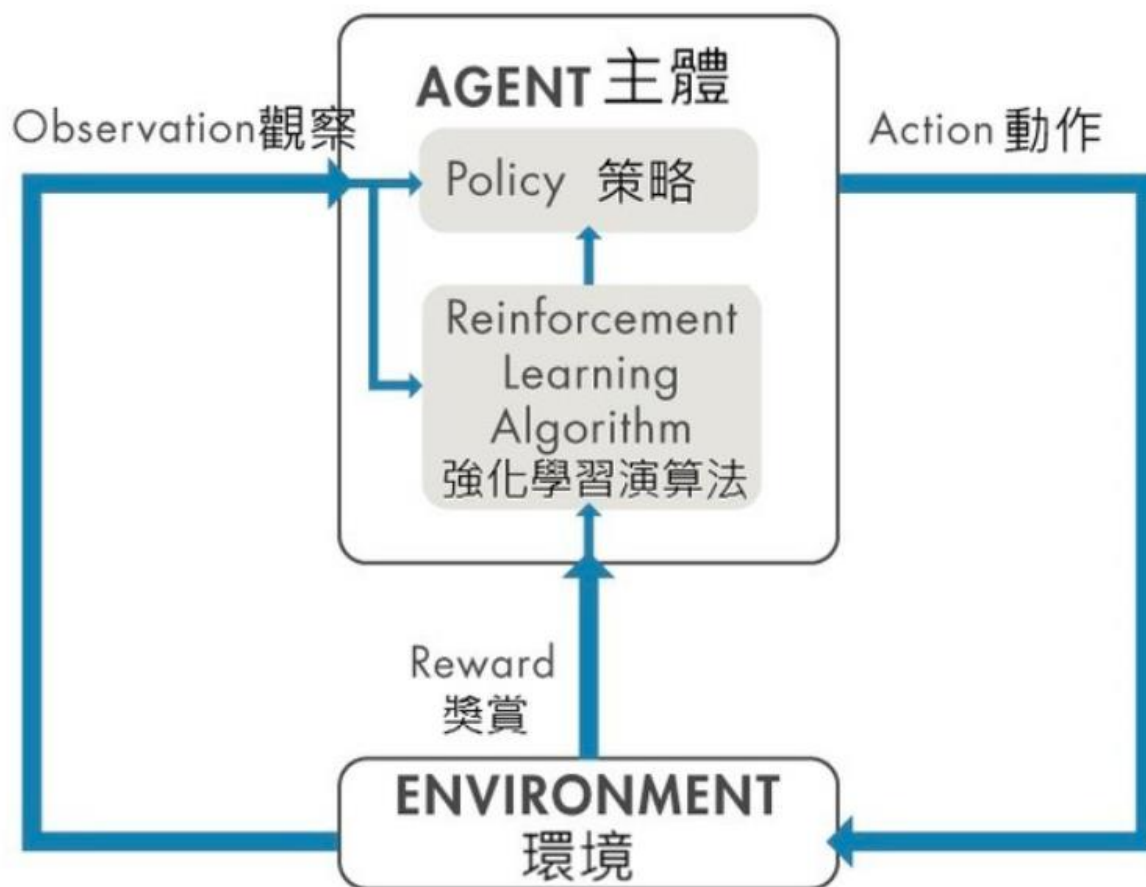


圖 5.2 強化學習主要流程[11]

## 5.2 Q-learning 原理

Q-learning 中會需要一個代理人(Agent)，用來表示在不同環境下做出決定的角色，而我們需要決定角色面對到的狀態(State)，和定義面對此狀態所做出的動作(Action)，最後每個 state 和 action 對應到的即為 Q 值，Q 值越大 Agent 越容易以其 action 做為下一步的動作，圖 5.3 為 Q-learning 演算法。

```

Initialize  $Q(s, a)$  arbitrarily
Repeat (for each episode):
    Initialize  $s$ 
    Repeat (for each step of episode):
        Choose  $a$  from  $s$  using policy derived from  $Q$  (e.g.,  $\epsilon$ -greedy)
        Take action  $a$ , observe  $r, s'$ 
         $Q(s, a) \leftarrow Q(s, a) + \alpha[r + \gamma \max_{a'} Q(s', a') - Q(s, a)]$ 
         $s \leftarrow s'$ 
    until  $s$  is terminal

```

圖 5.3 Q-learning 演算法[9]

其中的核心公式 Q-function 由圖 5.4 所示， $Q(s, a)$  是在狀態  $s$  下採取行動  $a$  的  $Q$  值， $\alpha$  是學習率控制了每次更新  $Q$  值時新信息的貢獻程度。如果學習率太高，更新  $Q$  值時可能會太過於激進，導致算法無法收斂或收斂緩慢。 $r$  是在狀態  $s$  下採取行動  $a$  後收到的， $\gamma$  是衰減因子， $\gamma$  值越大，未來回報的價值越高，Agent 更加重視未來的回報，將更加注重長期利益， $\max_a Q(s', a')$  是在狀態  $s'$  下所有可能的行動  $a'$  的最大  $Q$  值。

$$Q^{new}(s_t, a_t) \leftarrow \underbrace{Q(s_t, a_t)}_{\text{old value}} + \underbrace{\alpha}_{\text{learning rate}} \cdot \underbrace{\left( \underbrace{r_t}_{\text{reward}} + \underbrace{\gamma}_{\text{discount factor}} \cdot \underbrace{\max_a Q(s_{t+1}, a)}_{\text{estimate of optimal future value}} - \underbrace{Q(s_t, a_t)}_{\text{old value}} \right)}_{\text{temporal difference}} \underbrace{\quad}_{\text{new value (temporal difference target)}}$$

圖 5.4 Q-function[10]

### 5.3 藉由 PWM 控制風扇轉速

脈波寬度調變(Pulse Width Modulation) 簡單 PWM，是一種控制脈波工作週期 (duty cycle) 來達到控制輸出電壓的一種控制方式。由下方公式可知方波的直流值是與工作週期成正比，因此我們只要改變工作週期，就可以改變輸出的電壓(圖 5.5)進而改變風扇的轉數，在 arduino 中可以設定的週期範圍為 0 到 255。

$$V_{DC} = \text{Max } V_{in} \times \text{Duty Cycle}$$

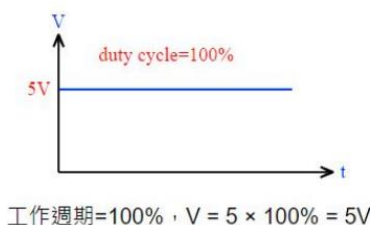


圖 5.5 輸出電壓對工作週期折線圖

## 5.4 選擇強化學習— Q-learning

本次專題主要考量人體溫度和風扇轉速的關係，溫度的變化是連續不斷的，而且對於使用者來說，冷熱的感受是十分主觀的，我們無法事先將資料做標記，因此我們選強化學習當中的 Q-learning，就由讓風扇探索環境累積經驗，在未來遇到類似經驗時就能選出最優的決策。

### 建構 Q-table

我們將 action 從-100 每格遞增 10 到 100，總共 21 格，以及 state 為根據目前體溫和目標溫度的差值以每格 0.2°C 的差距，從+8.0°C到-8.0°C，一共 81 格如表 5.1 所示。

表 5.1 風扇之 Q-table

		PWM								
溫 差		-100	-90	...	-10	0	+10	...	+90	+100
	+8.0									
	...									
	+0.4									
	+0.2									
	0.0									
	-0.2									
	-0.4									
	...									
	-8.0									

## 5.5 設計主要的訓練演算法

(1)偵測溫度

(2)查詢 q-table 並選擇最大 q 值的 action

(3)檢查溫度是否有往目標溫度前進有則加分，反之扣分

(4)利用圖 5.4 中的公式更新 Q-table

回到(1)繼續訓練直到結束，圖 5.6 為其流程圖

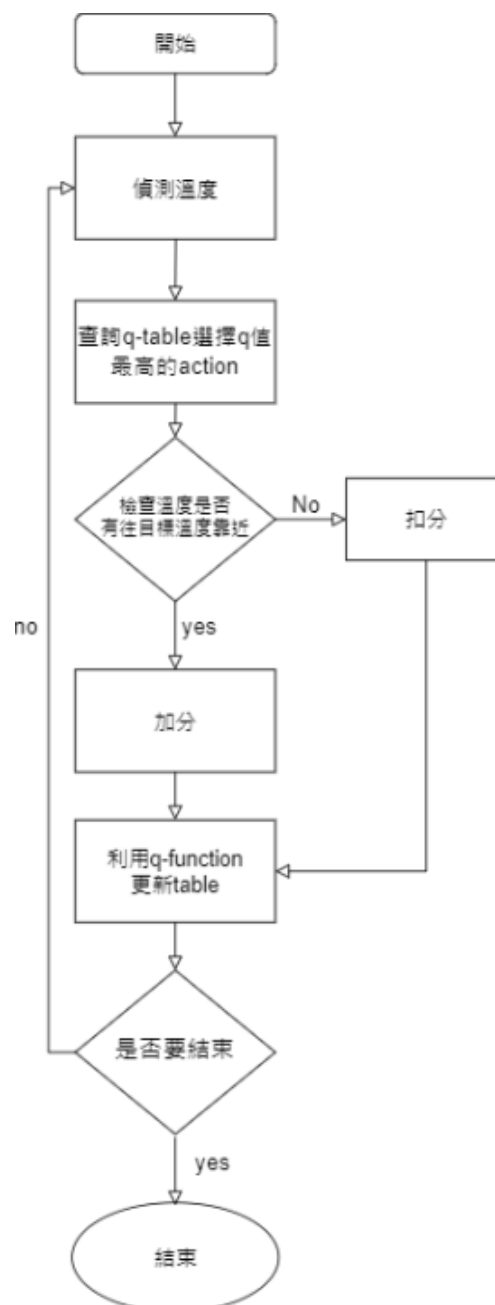


圖 5.6 訓練驗算法流程圖

## 六、系統實現與實驗

### 6.1 硬體方面：

- Arduino Mega2560
- ESP8266
- L298N
- DS18B20

#### 6.1.1 Arduino Mega2560

於 Arduino 系列開發板的一員(圖 6.1)。它擁有 54 個數字 I/O 引腳（其中 15 個可作為 PWM 輸出），16 個類比輸入引腳，4 個 UART 串口，16 MHz 的晶體振盪器，256 KB 的 Flash 存儲器，8 KB 的 SRAM 和 4 KB 的 EEPROM。它支持 USB 接口和外部電源供應，可通過 USB 接口與電腦連接進行溝通和數據傳輸。應為其儲存空間較大，因此我們使用此來當作風扇和軟體之間的橋樑。

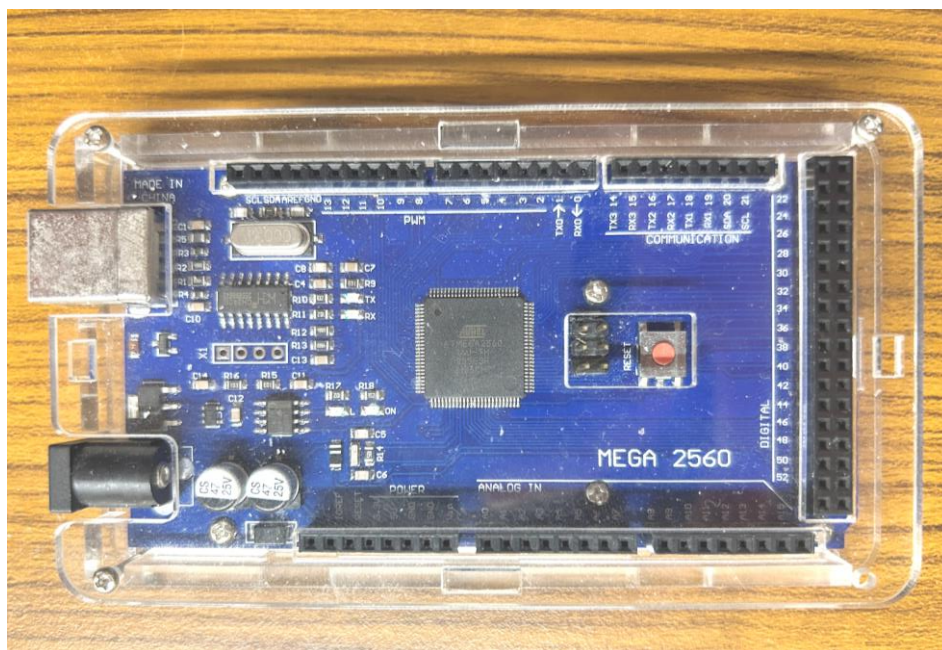


圖 6.1 Arduino mega 2560



### 6.1.2 ESP8266

此模組能讓上述的 arduino 開發版能連上 wifi，ESP8266 有許多系列，我們是使用 ESP-01s 來讓我們的風扇能跟資料庫連結。ESP-01s 模組具有 8 個引腳，包括 2 個 GPIO、UART 串口通訊口和 ADC（類比數位轉換器）引腳，並且具有內置的 WiFi 天線。它可以透過 AT 指令進行控制，支持 STA（station）和 AP（access point）模式，並且可以通過 TCP/IP 協議進行無線通訊。此外，ESP-01s 還具有支持 WPA/WPA2 安全協議的加密功能，可保護無線通訊的安全性，圖 6.2、圖 6.3 為其與 arduino 開發板的接腳圖。

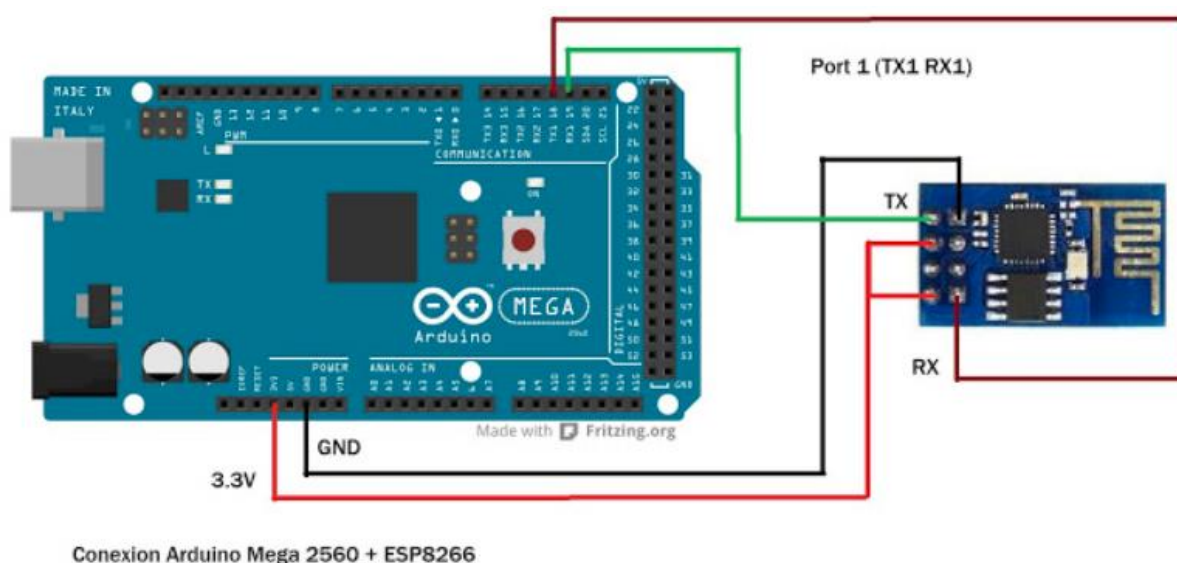


圖 6.2 Arudino 和 ESP01s 的接線圖[13]

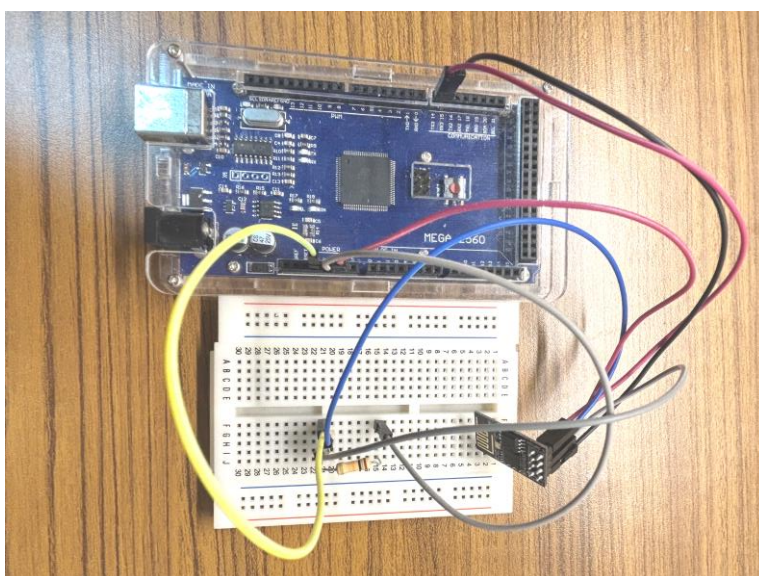


圖 6.3 Arudino 和 ESP01s 的實際接線圖



### 6.1.3 L298N

L298N(圖 6.4)是馬達驅動模組，可用於控制直流馬達和步進馬達等負載，電壓範圍為 5V 至 35V，它可以通過控制輸入端的信號控制馬達的轉向和速度，正好符合我們挑整風扇轉數的需求。

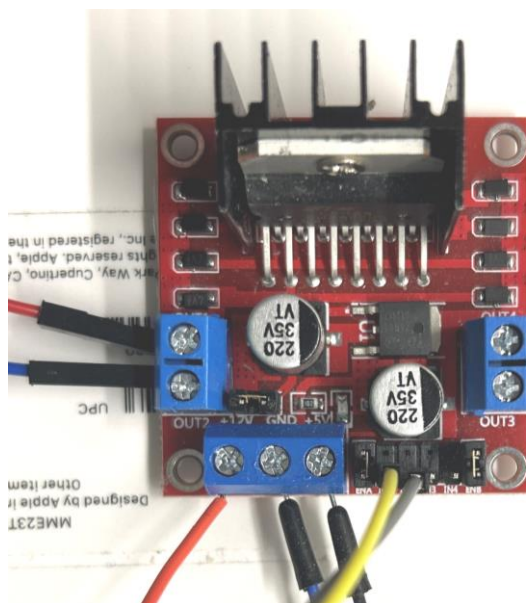


圖 6.4 L298N

表 6.1 為 L298N 和 Arduino mega 2560 的接線關係，其中 L298N 需要外接 12V 的電源，並將 L298N 上的 GND 接到 Arduino mega 2560 上，使其能夠順利運行。

表 6.1 mega 對 L298N 腳位

Arduino mega 2560	L298N
	12V 外部電源
GND	GND
Pin8(PWM)	IN1
Pin9(PWM)	IN2

## 6.1.4 DS18B20

它使用 1-Wire (One-Wire) 資料傳輸協定，資料傳輸只需一條線即可，需要三個腳位（電源、資料、接地）。此溫度感測元件溫度感測範圍-55°C 到+125°C，而且在 -10 °C~85 °C 這個範圍內保證  $\pm 0.5^{\circ}\text{C}$  的精確度，偵測的溫度讀數已在內部校正為攝氏刻度，因此我們選擇此元件來偵測人體的體溫，如圖 6.5。



圖 6.5 DS18B20

表 6.2 mega 對 L298N 腳位

Arduino mega2560	DS18B20
GND	GND
5V	VCC
Pin5	Data

## 6.2 風扇和各個模組的實際接線圖

圖 6.6、圖 6.7 為風扇的實際接線圖

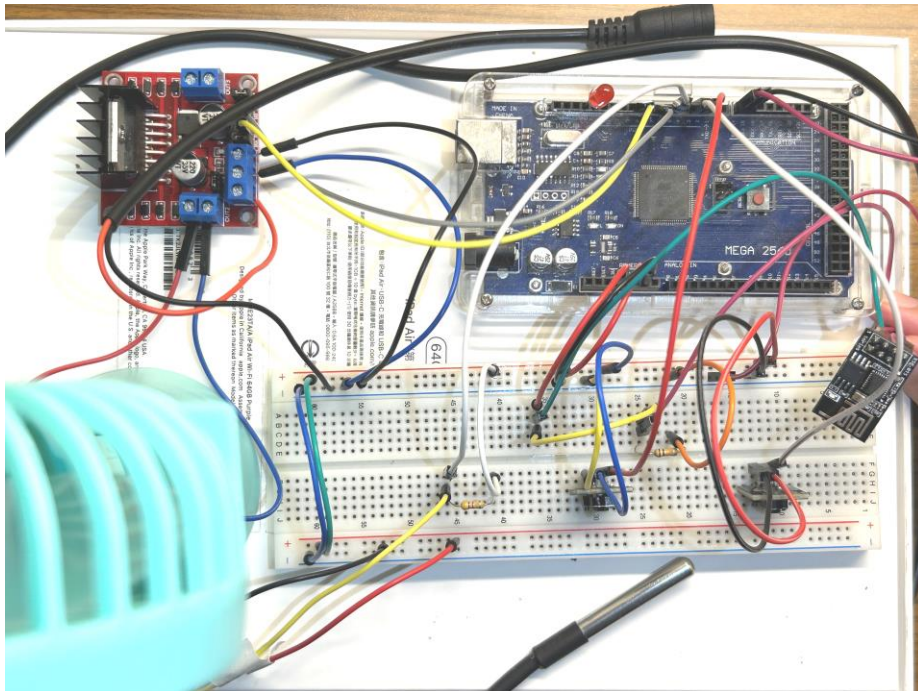


圖 6.6 風扇實際接線圖

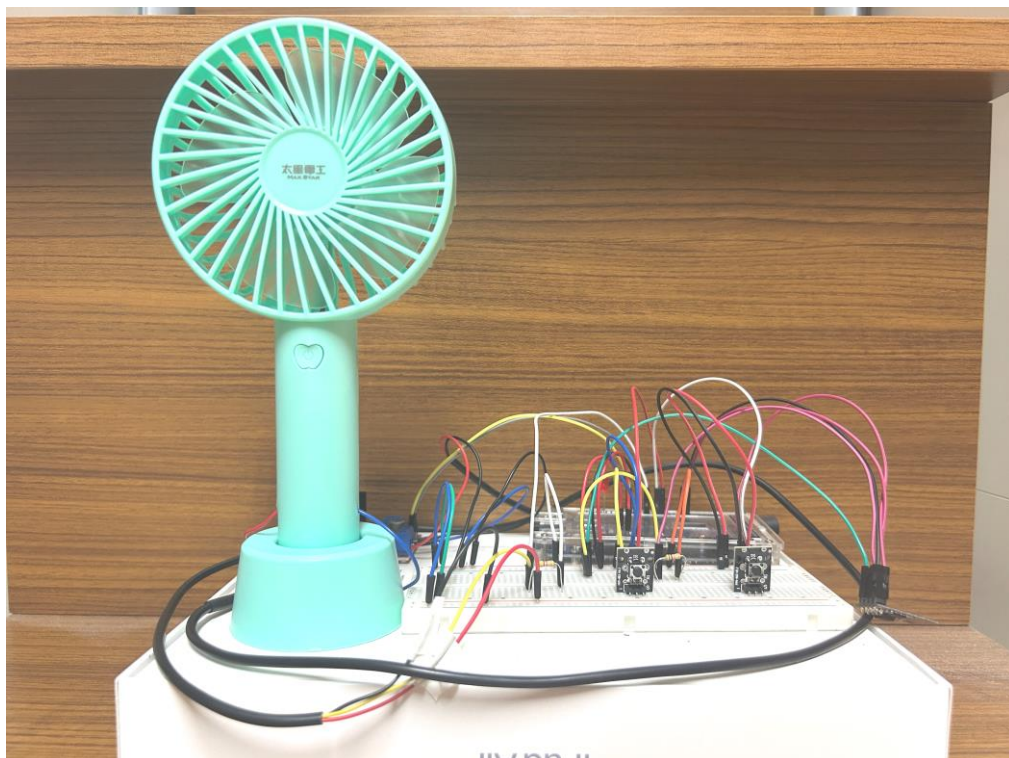


圖 6.7 風扇實際接線圖

## 6.3 軟體方面

### 6.3.1 使用者登入

使用者可利用 QR code 掃描登入，用戶可註冊新帳號或登入舊有的帳號如圖 6.8。

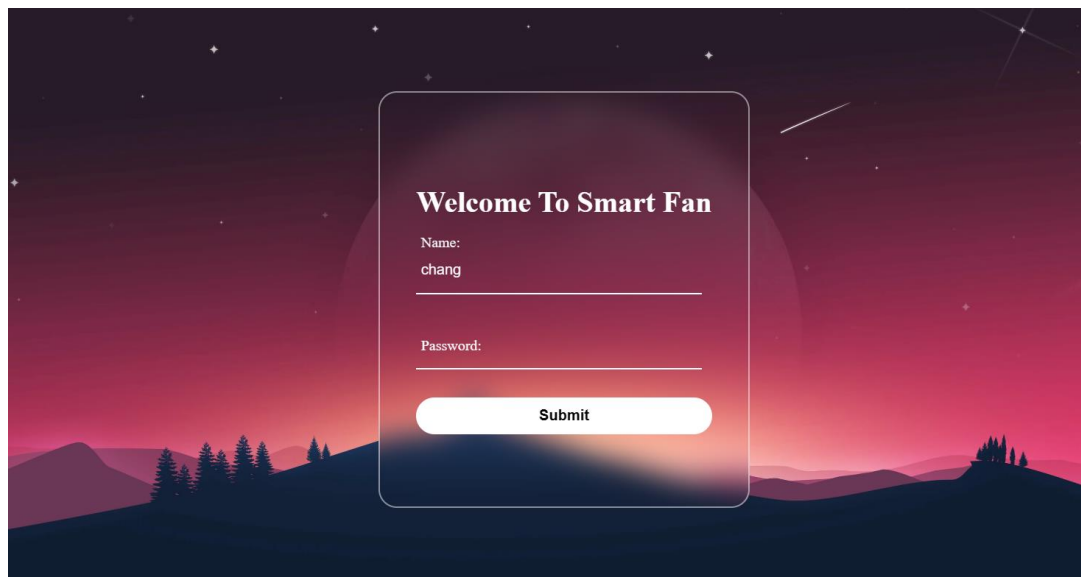


圖 6.8 登入畫面

### 6.3.2 ESP8266 連上 WIFI、TCP 連線

利用 ESP8266 函式庫連到指定的 WIFI，風扇和伺服器建立 TCP 連線。

### 6.3.3 連線資料庫取得資料

使用者登入後，系統會抓取之前的 q-table，並將其傳送給風扇，如未登入過，將初始化新的 table 再傳給風扇，如圖 6.9 所示。

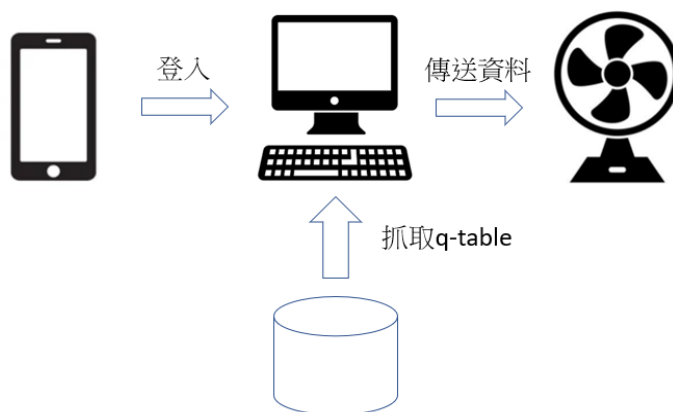


圖 6.9 登入及傳送資料示意圖

### 6.3.4 用 q-learning 開始訓練

一開始會設定一個基本風速，記錄使用者的感受並改變風速，例如感覺冷會使風速降低，反之風速增加，當使用者感受到一次冷和一次熱時，利用二分法，推斷使用者最適溫度落在此區間內，藉由兩個溫度取中間值當目標溫度開始訓練。

每 3 秒偵測溫度，從 q-table 中選出 q 值最高的 action，並計算 reward，計算方式如下：

- 判斷此 action 是否有使溫度靠近目標溫度，如果是，reward 為此溫度差乘 10，不是則為負的。
- 溫度不變但溫度為目標溫度則 reward 為 20，反之如果不是目標溫度則 reward 為-20

最後根據 q-function 來更新 q-table。

### 6.3.5 訓練結果更新資料庫

利用 ESP8266 跟資料庫建立 TCP 連線，將 q-table 傳給資料庫，如圖 6.10 所示。



圖 6.10 Q\_table 上傳示意圖

### 6.3.6 整體流程圖

圖 6.11 為風扇整體架構的流程圖

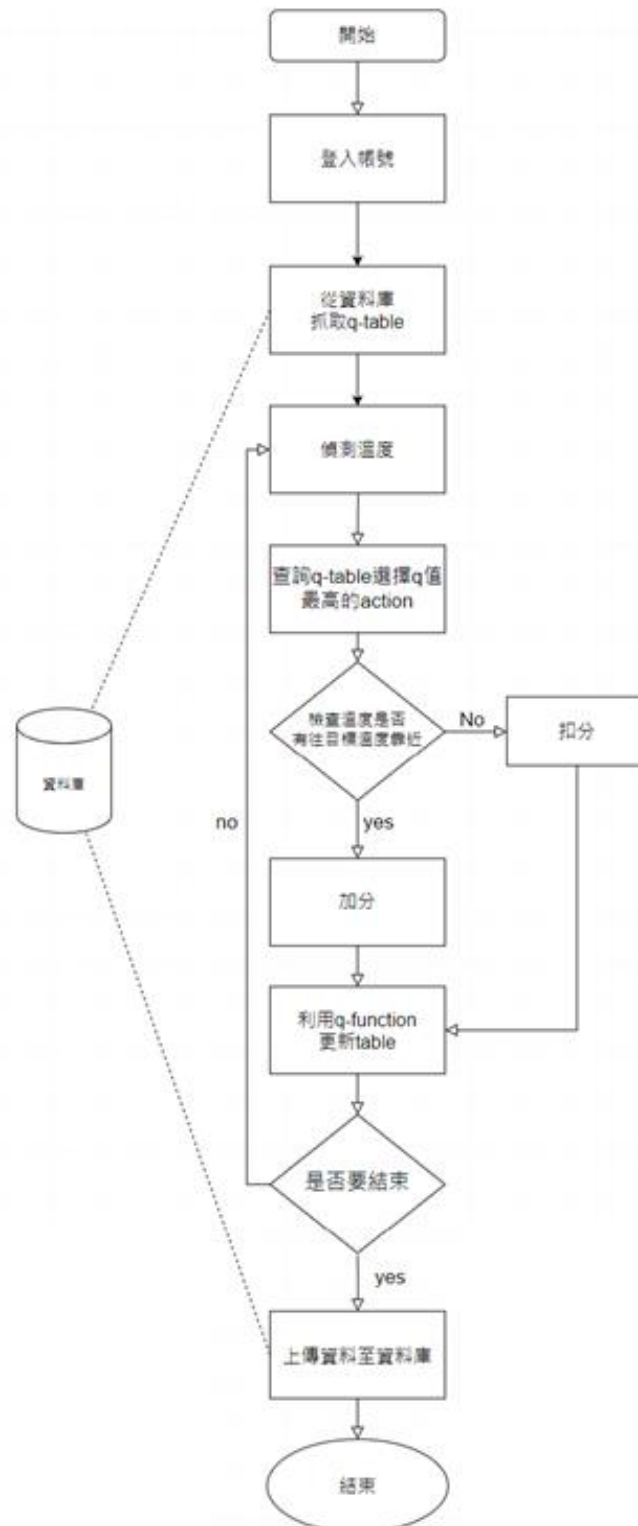


圖 6.11 整體流程圖

## 6.4 實驗設計

將風扇放置於離受試者約 40cm 處，吹向胸口至脖子位置，檢測溫度變化，最後根據其測量值來觀看效果，用同樣方式測量一般風扇對溫度的影響，並比較兩者間的差異。



## 七、效能評估與成果

圖 7.1 為使用 q-learning 訓練的結果，Y 軸為溫度和 X 軸為訓練次數的關係圖，一開始 epsilon 值較高，會採取較多隨機 action，造成前期溫度較不穩定，隨著訓練次數增加，降低 epsilon 的比率，讓風扇可以根據 q-table 來選擇 action，溫度也因此逐漸朝目標溫度收斂。

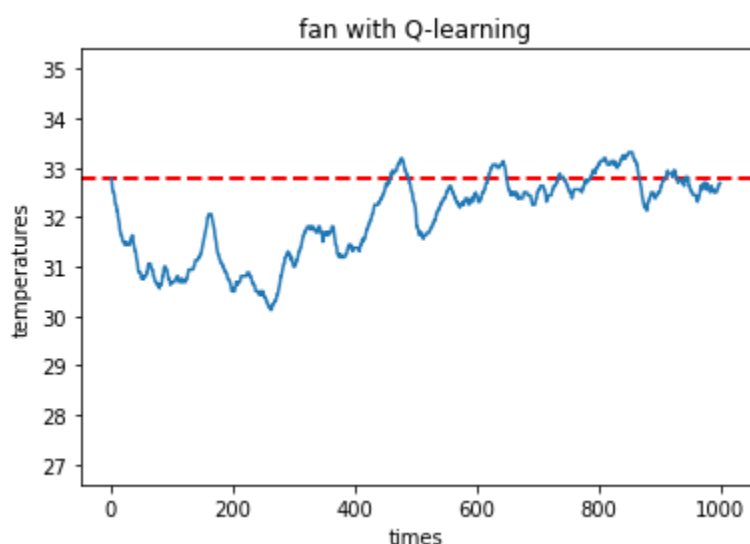


圖 7.1 溫度對次數的折線圖，紅線為目標溫度 32.78°C

圖 7.2 為 PWM 值和次數的關係圖，相關係數為-0.16，從圖中可得知 PWM 值隨著訓練次數增加，PWM 值越小，由圖 7.3 可看出呈現負相關。根據圖 7.1，溫度最後會逐漸在目標溫度附近，因此風扇會選擇 PWM 較小的值來穩定溫度。

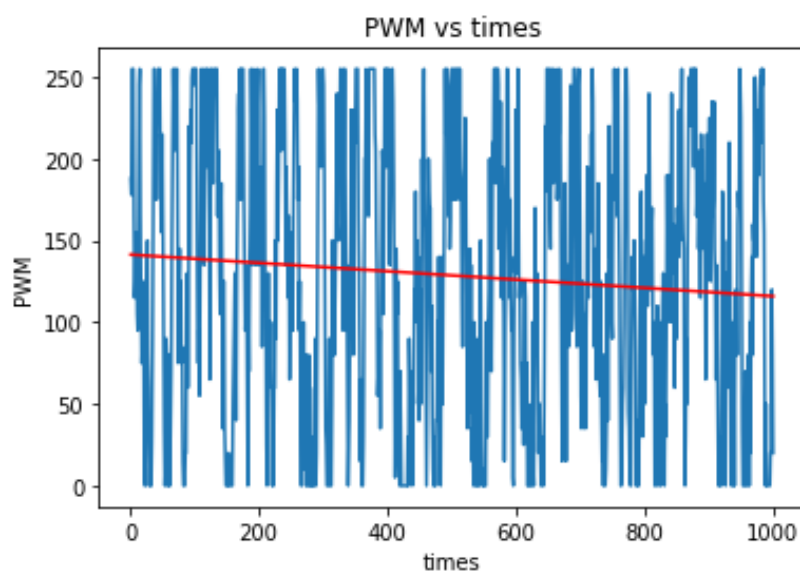


圖 7.2 PWM 對次數的折線圖，紅線為 PWM 和 times 的線性關係

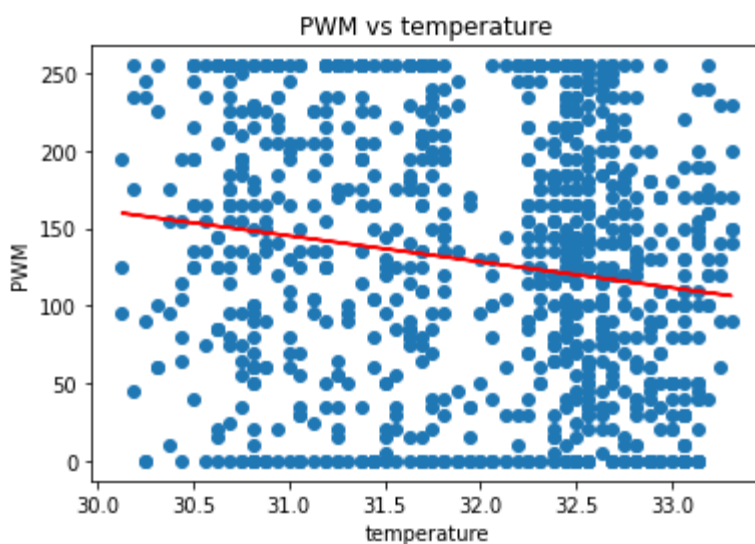


圖 7.3 PWM 對溫度的點陣圖，紅線為 PWM 和 times 的線性關係

圖 7.4 為一般固定轉速之風扇，隨著次數的增加，體溫逐漸下降，偏離目標溫度

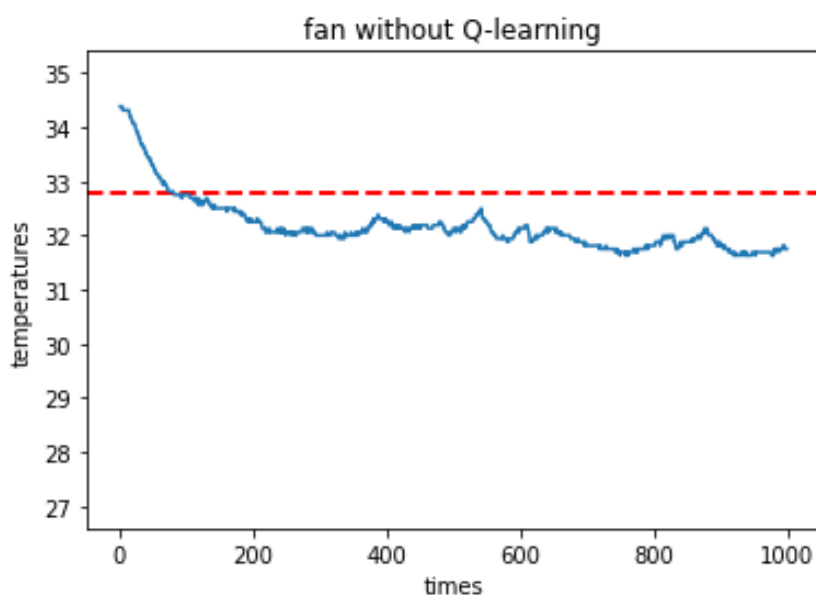


圖 7.4 溫度對次數的折線圖，紅線為目標溫度 32.78°C

## 八、結論:

有 q-learning 訓練的風扇在一開始訓練時雖然會花較多時間在訓練，但是隨著時間推移，很明顯較一般市售風扇在維持最適溫度有更好的表現。不過因為風扇無法對環境溫度和濕度有太大的影響，因此在不同的室溫和環境下，都可能會影響最適溫度的決定。



## 九、參考文獻：

- [1] Van Otterlo, M. Wiering, M. Reinforcement learning and markov decision processes, Reinforcement Learning, Adaptation, Learning, and Optimization. Vol. 12. pp.3–42 ,2012.
- [2] ITEADLIB\_Arduino\_WeeESP8266  
[https://github.com/itead/ITEADLIB\\_Arduino\\_WeeESP8266](https://github.com/itead/ITEADLIB_Arduino_WeeESP8266)
- [3] esp8266 基本測試  
[http://maker.tn.edu.tw/modules/tad\\_book3/page.php?tbsn=21&tbsn=390](http://maker.tn.edu.tw/modules/tad_book3/page.php?tbsn=21&tbsn=390)
- [4] \_ESP8266 WiFi 模組(ESP-01)韌體更新  
<http://www.twbts.work/2019/05/esp8266-wifi-esp-01.html?m=1>
- [5] 溫度感測器 DS18B20  
<https://shop.mirotek.com.tw/arduino/arduino-adv-1/>
- [6] 強化學習導論六：時間差分學習(TD Learning)  
<https://www.getit01.com/p2018020833426502/>
- [7] Q-Learning Algorithm: From Explanation to Implementation  
<https://towardsdatascience.com/q-learning-algorithm-from-explanation-to-implementation-cbda2ea187>
- [8] 使用 L298N 模組控制直流馬達  
<http://coopermaa2nd.blogspot.com/2012/09/l298n.html>
- [9] Open AI Gym 簡介與 Q learning 演算法實作  
<https://blog.techbridge.cc/2017/11/04/openai-gym-intro-and-q-learning/>
- [10] qlearning 基於價值的強化學習算法  
[https://blog.csdn.net/weixin\\_26729375/article/details/109070321](https://blog.csdn.net/weixin_26729375/article/details/109070321)
- [11] 強化學習：入門指南  
<https://smartauto.ctimes.com.tw/DispArt-tw.asp?O=HK3AT6N81A2ARASTDX>
- [12] 探討機器學習與深度學習之差異  
<https://www.wpgdadatong.com/tw/blog/detail/40355>

[13] Test Arduino Mega 2560 and ESP8266 (AT Commands)  
<https://pdacontrolen.com/prueba-arduino-mega-2560-y-esp8266/>