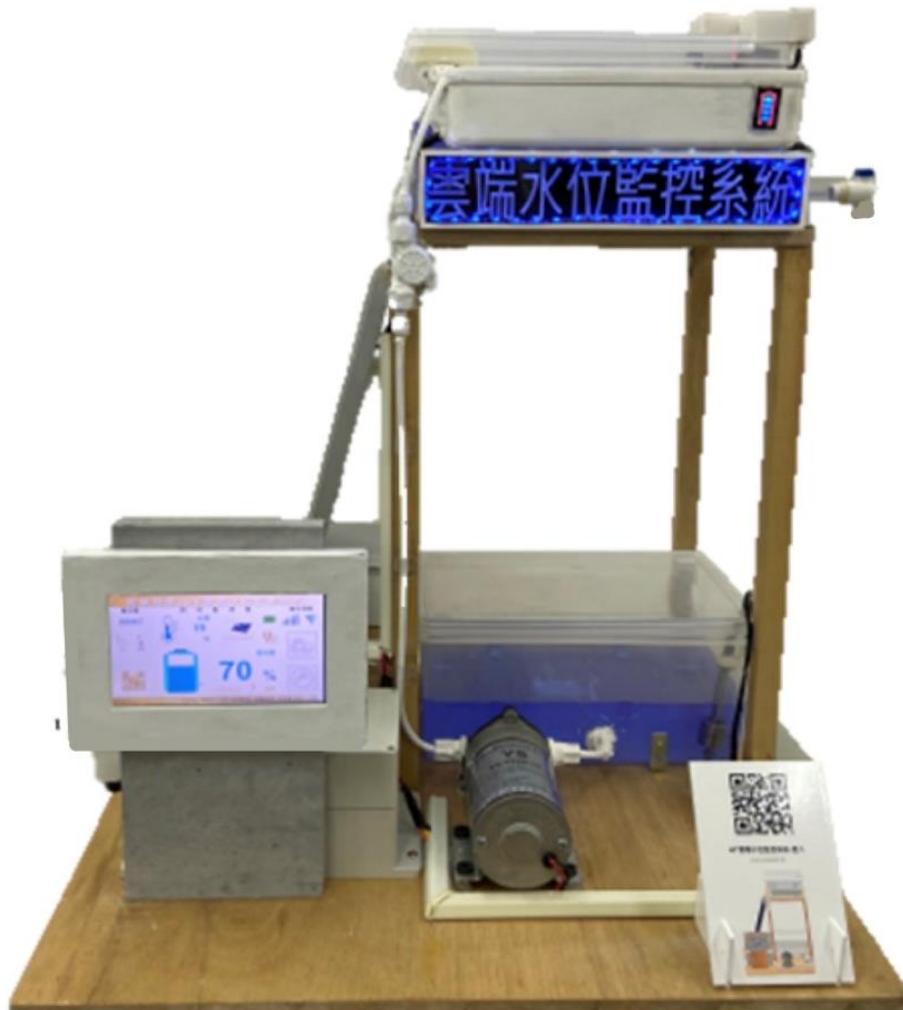


臺北市立大安高級工業職業學校專題製作競賽
「專題組」作品說明書



群別：電機與電子群

作品名稱：智慧居家物聯網-雲端水位監控系統

關鍵詞：智慧居家、物聯網、水位監控

目錄

壹、摘要.....	1
貳、研究動機.....	1
參、主題與課程之相關性或教學單元之說明.....	1
一、3D 繪圖與列印.....	1
二、雷射雕刻.....	2
三、電路板雕刻.....	2
四、硬體電路設計.....	3
五、軟體程式撰寫.....	3
肆、研究方法.....	4
一、研究流程.....	4
(一)、研究步驟.....	4
(二)、動作流程.....	5
二、使用材料.....	6
(一)、結構材料.....	6
(二)、電機材料.....	6
三、使用軟體與服務.....	10
(一)、Arduino IDE.....	10
(二)、USART HMI.....	10
(三)、Visual Studio.....	11
(四)、MQTT.....	11
(五)、Apache.....	12
(六)、MySQL.....	12
(七)、Altium Designer.....	13
(八)、Autodesk Inventor.....	13
(九)、RDWorks.....	14
(十)、Blender.....	14
二、使用設備.....	15
(一)、3D 列印機.....	15
(二)、雷射雕刻機.....	15
(三)、電路板雕刻機.....	15
伍、研究結果.....	16
一、主體結構.....	16
(一)、下水塔.....	16
(二)、上水塔.....	16
(三)、上防水盒.....	16

(四)、中防水盒	17
(五)、下防水盒	17
二、HMI 觸控螢幕	17
(一)、抽水控制區	17
(二)、即時監控區	17
(三)、警示區	17
三、邏輯電路	18
(一)、上水塔主控系統	18
(二)、屋內主控系統	18
四、電源供應	18
(一)、交流供電系統	18
(二)、太陽能供電系統	18
五、MQTT 通訊協定	18
六、網頁連線	19
七、成果展示	20
(一)、成品外觀	20
(二)、HMI 介面	20
(三)、偷水偵測及 Gmail 通知	21
(四)、網頁監控即時顯示	21
(四)、大數據分析	22
(五)、異常通知紀錄與網頁連結分享	22
陸、討論	23
一、水管選用	23
二、ESP32 NodeMcu 傳輸問題	23
三、電量顯示	23
柒、結論	24
捌、參考資料及其他	25
一、書籍資料	25
二、網路資料	25

表目錄

表 1	研究時間分配.....	4
表 2	ESP32 無線模組規格.....	6
表 3	超音波感測器規格.....	7
表 4	紅外線避障感測器規格.....	7
表 5	不鏽鋼浮球開關規格.....	7
表 6	流量感測器規格.....	8
表 7	抽水馬達規格.....	8
表 8	繼電器模組規格.....	8
表 9	太陽能板規格.....	9
表 10	HMI 規格.....	9
表 11	溫度感測器規格.....	9

圖目錄

圖 01	固定座 3D 設計圖	2
圖 02	3D 列印成品	2
圖 03	RDWorks 軟體	2
圖 04	雷射雕刻成品	2
圖 05	使用 Altium Designer 繪製電路圖	2
圖 06	PCB 電路板設計	2
圖 07	智慧居家監控實習的開發板	3
圖 08	臺科大太陽能模組實作成品	3
圖 09	Arduino IDE 程式開發	3
圖 10	藍芽無線控制 App 開發實作	3
圖 11	研究步驟	4
圖 12	木板結構材料	6
圖 13	ESP32 無線模組	6
圖 14	超音波感測器	7
圖 15	紅外線避障感測器	7
圖 16	不鏽鋼浮球開關	7
圖 17	流量感測器	8
圖 18	抽水馬達	8
圖 19	繼電器模組	8
圖 20	太陽能板	9
圖 21	7 吋液晶觸控螢幕 HMI	9
圖 22	溫度感測器	9
圖 23	Arduino IDE 軟體介面	10
圖 24	USART HMI 軟體介面	10
圖 25	Visual Studio logo	11
圖 26	Visual Studio 軟體介面	11
圖 27	MQTT logo	11
圖 28	Apache logo	12
圖 29	MySQL logo	12
圖 30	MySQL 軟體介面	12
圖 31	Altium Designer logo	13
圖 32	Altium Designer 軟體介面	13
圖 33	Autodesk Inventor 軟體介面	13
圖 34	RDWorks 軟體介面	14
圖 35	Blender 軟體介面	14
圖 36	3D 列印機	15

圖 37	雷射雕刻機.....	15
圖 38	電路板雕刻機.....	15
圖 39	專題成品.....	16
圖 40	HMI 觸控螢幕功能區塊.....	17
圖 41	mosquitto logo.....	19
圖 42	MQTT 傳輸過程	19
圖 43	網站登入畫面.....	19
圖 44	網站監控頁面.....	19
圖 45	專題成品外觀.....	20
圖 46	HMI 介面設計	20
圖 47	HMI 實品.....	20
圖 48	偷水示意圖	21
圖 49	Gmail 通知.....	21
圖 50	網頁監控即時顯示.....	21
圖 51	資料庫紀錄的每月用水量大數據分析	22
圖 52	手機端的網頁監控與大數據顯示畫面	22
圖 53	異常通知紀錄與網頁連結分享.....	22
圖 54	電量感測顯示模組.....	23
圖 55	光耦合器電路模組.....	23

【智慧居家物聯網-雲端水位監控系統】

壹、摘要

本專題以智慧居家物聯網-雲端水位監控系統的概念，完成水塔水位監控及水溫量測的設計，結合太陽能板及鋰電池組，作為綠能供電系統，整合 7 吋 HMI，作為居家室內的操控及資訊顯示的平台，使用 2 個 ESP32 無線模組，作為周邊硬體電路的控制、讀取與資料的傳遞，建置網頁及伺服器資料庫系統，並申請 www.tankiot.tk 作為專屬網站，應用資料庫系統的紀錄進行大數據分析，以圖表方式呈現在網頁。

本專題的研究成果並非僅侷限應用在智慧居家環境或是水位監控領域，只要依使用情境，改用相應規格的感測器，結合相關硬體電路控制，就可將本專題物聯網的技術，廣泛應用在科技業、工業、石化業、農林漁牧業、水產養殖業及水庫等，成為多功能的智慧物聯網-雲端監控系統的應用。

貳、研究動機

台灣夏雨冬乾的情形是非常嚴重的，住在中南部的人更是有切身之痛，在水情緊張時，甚至每個禮拜僅供五停二，這時如何知道家用水塔水位是否不足？如何確保家用水塔提前儲滿用水，又不用爬到頂樓去手拉浮球開關，採手動方式讓抽水馬達運作，提升水位？如果遇到家中只有年長的長輩時，如何實現不用爬到頂樓水塔，就能在屋內了解水塔水位及手動控制抽水馬達動作，甚至由在外上班的年輕人協助用手機軟體以雲端控制水塔的抽水馬達及監控水位，提前儲備用水。因此我們想利用在高職所學的知識及技能，開發智慧居家物聯網-雲端水位監控系統，讓使用者能夠隨時掌握家用水塔水位情況，提前儲備用水及有效掌握停水期間的用水分配量，利用科技來改善居家生活問題。

參、主題與課程之相關性或教學單元之說明

一、3D 繪圖與列印

我們利用高二跨領域課程中，在製圖科上課所學到的 Inventor 軟體，來繪製紅外線避障感測器固定座、超音波感測器防水固定座及浮球固定支架等 3D 圖，並利用課堂中所學到 3D 列印機的操作使用，將設計好的 3D 圖檔輸出連接到 3D 列印機，列印成組件成品，進行組裝使用。

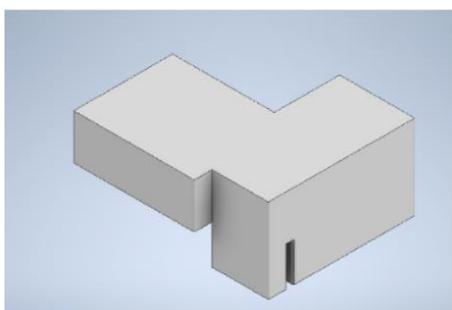


圖 1 固定座 3D 設計圖

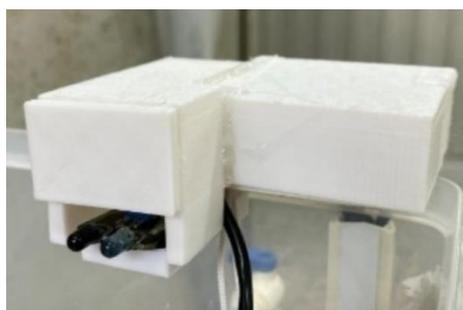


圖 2 3D 列印成品

二、雷射雕刻

我們利用高二跨領域課程中，在機械科上課所學到的 RDWorks 雷射雕刻設計軟體，畫出透明壓克力及電源供應器木質固座的設計圖，並利用課堂中所學到雷射雕刻機的操作使用，將設計好的圖檔輸出連接到雷射雕刻機，製作成組件成品，進行組裝使用。

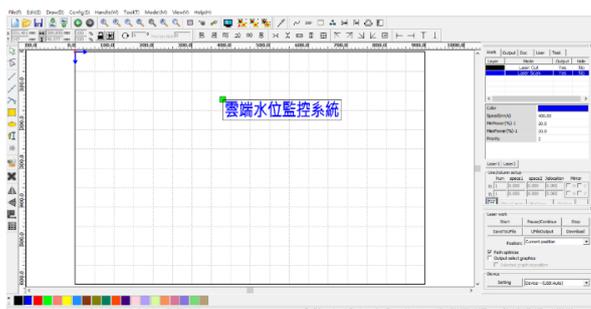


圖 3 RDWorks 軟體



圖 4 雷射雕刻成品

三、電路板雕刻

我們利用高三上學期「微電腦應用實習」課程中所學到的 Altium Designer 電路設計軟體，來繪製電路圖及設計 PCB 電路板，利用課堂中學到電路板雕刻機的操作使用，將設計好的 PCB 電路板轉換鑽孔檔與成型檔，輸出連接到電路板雕刻機，進行 PCB 電路板製作。

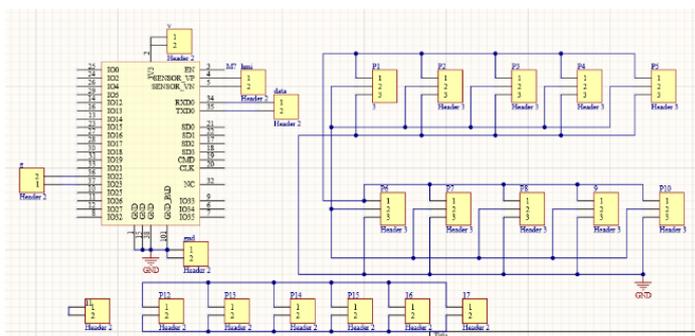


圖 5 使用 Altium Designer 繪製電路圖

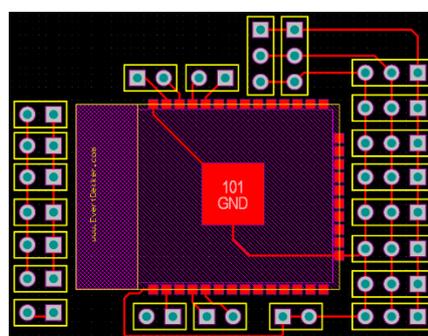


圖 6 PCB 電路板設計

四、硬體電路設計

我們應用高二「智慧居家監控實習」課程所學到的繼電器模組來控制抽水馬達動作，紅外線避障感測器來感測人體紅外線進行偷水偵測判斷，溫度感測器來量測水溫；應用高一下學期在臺灣科技大學七星學程之「智慧化綠能科技概論」課程所學到的太陽能儲能系統設計與實作知識，來整合設計太陽能板、太陽能充電模組及鋰電池組；應用「基本電學實習」及「電子學實習」所學到的交流、直流電源知識，來設計交流、直流電源供應系統。



圖 7 智慧居家監控實習的開發板

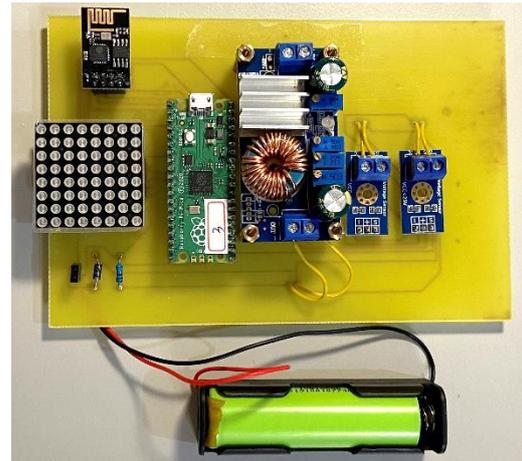


圖 8 臺科大太陽能模組實作成品

五、軟體程式撰寫

我們應用高二「智慧居家監控實習」所學到的 Arduino IDE 開發環境，來開發控制程式；使用 ESP32 無線模組來進行整合控制；應用行動裝置智慧監控、雲端電腦智慧監控的課程單元知識，來設計手機應用程式及網頁。應用在高二「可程式控制實習」所學到的人機介面課程單元知識，來設計 HMI 使用者介面及設定傳輸協定；應用在高二「單晶片微處理機實習」所學到的 C 語言程式知識，來編寫及整合控制程式。

```
ESP8266 Pin Map (ESP8266) Arduino 1.8.13
ESP8266 Pin Map (ESP8266)
16 #define MQTT_SERVER_IP "192.168.1.100"
17 #define MQTT_SERVER_PORT 1883
18
19 #define MQTT_SERVER_HOST "192.168.1.100"
20 #define MQTT_SERVER_PORT 1883
21 #define MQTT_SERVER_HOST "192.168.1.100"
22 #define MQTT_SERVER_PORT 1883
23 #define MQTT_SERVER_HOST "192.168.1.100"
24 #define MQTT_SERVER_PORT 1883
25 #define MQTT_SERVER_HOST "192.168.1.100"
26 #define MQTT_SERVER_PORT 1883
27 #define MQTT_SERVER_HOST "192.168.1.100"
28 #define MQTT_SERVER_PORT 1883
29 #define MQTT_SERVER_HOST "192.168.1.100"
30 #define MQTT_SERVER_PORT 1883
31 #define MQTT_SERVER_HOST "192.168.1.100"
32 #define MQTT_SERVER_PORT 1883
33 #define MQTT_SERVER_HOST "192.168.1.100"
34 #define MQTT_SERVER_PORT 1883
35 #define MQTT_SERVER_HOST "192.168.1.100"
36 #define MQTT_SERVER_PORT 1883
37 #define MQTT_SERVER_HOST "192.168.1.100"
38 #define MQTT_SERVER_PORT 1883
39 #define MQTT_SERVER_HOST "192.168.1.100"
40 #define MQTT_SERVER_PORT 1883
41 #define MQTT_SERVER_HOST "192.168.1.100"
42 #define MQTT_SERVER_PORT 1883
43 #define MQTT_SERVER_HOST "192.168.1.100"
44 #define MQTT_SERVER_PORT 1883
45 #define MQTT_SERVER_HOST "192.168.1.100"
46 #define MQTT_SERVER_PORT 1883
47 #define MQTT_SERVER_HOST "192.168.1.100"
48 #define MQTT_SERVER_PORT 1883
49 #define MQTT_SERVER_HOST "192.168.1.100"
50 #define MQTT_SERVER_PORT 1883
51 #define MQTT_SERVER_HOST "192.168.1.100"
52 #define MQTT_SERVER_PORT 1883
53 #define MQTT_SERVER_HOST "192.168.1.100"
54 #define MQTT_SERVER_PORT 1883
55 #define MQTT_SERVER_HOST "192.168.1.100"
56 #define MQTT_SERVER_PORT 1883
57 #define MQTT_SERVER_HOST "192.168.1.100"
58 #define MQTT_SERVER_PORT 1883
59 #define MQTT_SERVER_HOST "192.168.1.100"
60 #define MQTT_SERVER_PORT 1883
61 #define MQTT_SERVER_HOST "192.168.1.100"
62 #define MQTT_SERVER_PORT 1883
63 #define MQTT_SERVER_HOST "192.168.1.100"
64 #define MQTT_SERVER_PORT 1883
65 #define MQTT_SERVER_HOST "192.168.1.100"
66 #define MQTT_SERVER_PORT 1883
67 #define MQTT_SERVER_HOST "192.168.1.100"
68 #define MQTT_SERVER_PORT 1883
69 #define MQTT_SERVER_HOST "192.168.1.100"
70 #define MQTT_SERVER_PORT 1883
71 #define MQTT_SERVER_HOST "192.168.1.100"
72 #define MQTT_SERVER_PORT 1883
73 #define MQTT_SERVER_HOST "192.168.1.100"
74 #define MQTT_SERVER_PORT 1883
75 #define MQTT_SERVER_HOST "192.168.1.100"
76 #define MQTT_SERVER_PORT 1883
77 #define MQTT_SERVER_HOST "192.168.1.100"
78 #define MQTT_SERVER_PORT 1883
79 #define MQTT_SERVER_HOST "192.168.1.100"
80 #define MQTT_SERVER_PORT 1883
81 #define MQTT_SERVER_HOST "192.168.1.100"
82 #define MQTT_SERVER_PORT 1883
83 #define MQTT_SERVER_HOST "192.168.1.100"
84 #define MQTT_SERVER_PORT 1883
85 #define MQTT_SERVER_HOST "192.168.1.100"
86 #define MQTT_SERVER_PORT 1883
87 #define MQTT_SERVER_HOST "192.168.1.100"
88 #define MQTT_SERVER_PORT 1883
89 #define MQTT_SERVER_HOST "192.168.1.100"
90 #define MQTT_SERVER_PORT 1883
91 #define MQTT_SERVER_HOST "192.168.1.100"
92 #define MQTT_SERVER_PORT 1883
93 #define MQTT_SERVER_HOST "192.168.1.100"
94 #define MQTT_SERVER_PORT 1883
95 #define MQTT_SERVER_HOST "192.168.1.100"
96 #define MQTT_SERVER_PORT 1883
97 #define MQTT_SERVER_HOST "192.168.1.100"
98 #define MQTT_SERVER_PORT 1883
99 #define MQTT_SERVER_HOST "192.168.1.100"
100 #define MQTT_SERVER_PORT 1883
```

圖 9 Arduino IDE 程式開發



圖 10 藍芽無線控制 App 開發實作

肆、研究方法

一、研究流程

(一)、研究步驟

在七月決定專題題目後，便接著開始分工進行資料蒐集，同時展開元件及材料採購，接著進行水塔結構的設計及製作，同時展開 ESP32 無線模組及 HMI 的程式設計撰寫，接著進行電路板設計及製作，同時開始進行網頁設計，接著進行成品整合，之後進行成品測試與除錯，最後完成專題成品。專題的研究時間分配及研究步驟分別如下表 1 及圖 11：

表 1 研究時間分配

	7月	8月	9月	10月	11月	12月	1月
1. 資料蒐集							
2. 元件採購							
3. 水塔製作							
4. 程式撰寫							
5. 電路製作							
6. 網頁設計							
7. 成品整合							
8. 成品測試							

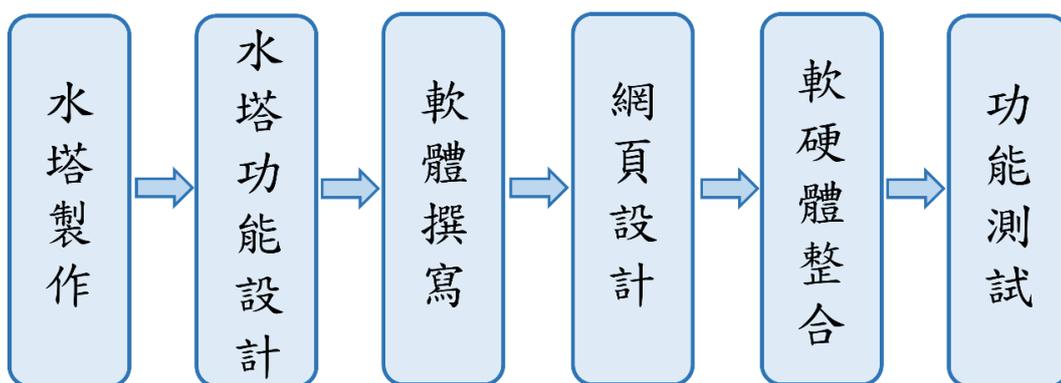


圖 11 研究步驟

(二)、動作流程

1、上水塔 ESP32

- (1)、讀取超音波模組偵測到的水位高度；
- (2)、以水位高度換算水塔蓄水量；
- (3)、讀取水溫；
- (4)、讀取浮球開關信號；
- (5)、讀取紅外線感測器的偷水偵測信號；
- (6)、讀取交流電源輸入信號；
- (7)、讀取鋰電池電壓的類比信號；
- (8)、換算成鋰電池蓄電量；
- (9)、以 ESP-Now 通訊協定與下水塔 ESP32 無線通訊
- (10)、將資料以傳送到下水塔 ESP32

2、下水塔 ESP32

- (1)、讀取太陽能充電輸入信號；
- (2)、判斷是否已連上無線路由器；
- (3)、偵測 WiFi 無線信號強度；
- (4)、換算 WiFi 無線信號刻度
- (5)、以 ESP-Now 通訊協定接收上水塔 ESP32 資料；
- (6)、將接收到的資料解析；
- (7)、將資料以 UART 通訊傳輸給 HMI；
- (8)、接收 HMI 資料；
- (9)、依 HMI 通訊協定解析資料；
- (10)、將所有資料以 WiFi 無線傳輸到網站；

3、HMI 人機介面

- (1)、偵測觸控事件；
- (2)、將觸控事件以 UART 傳輸到下水塔 ESP32；
- (3)、以 UART 通訊接收下水塔 ESP32 資料；
- (4)、依接收到的資料顯示到對應的物件；

4、管理網頁

- (1)、偵測是否有使用者訪問；
- (2)、判斷使用者名稱是否已建立；
- (3)、判斷使用者的密碼是否正確；
- (4)、顯示登入結果；
- (5)、更新網頁；
- (6)、依接收到資料顯示在網頁；
- (7)、更新網頁；

二、使用材料

(一)、結構材料

在主體結構的部分，為了滿足支撐水塔的機械強度，我們選用厚度 18mm 的木板，用鋸台切割成 11 塊木板來搭建，並使用自攻螺絲來進行鎖固，作為水塔、水路、電路系統及 HMI 的支撐固定使用。

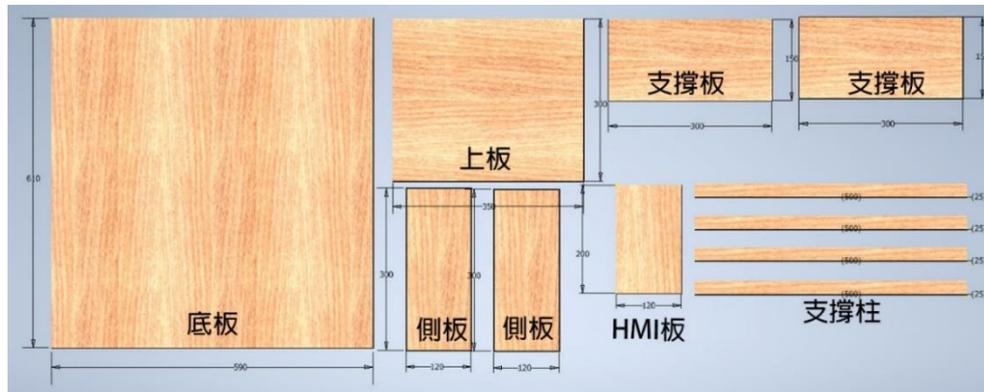


圖 12 木板結構材料

(二)、電機材料

1、ESP32 無線模組

ESP32 無線模組是一款雙核心，並結合 WiFi 和藍牙功能的 32 位元微控制器，採用 Tensilica Xtensa LX6 微處理器，內建天線開關，RF 變換器，功率放大器，低雜訊接收放大器，濾波器和電源管理模組。

表 2 ESP32 無線模組規格

廠牌	安信可
型號	NodeMCU-32S
腳位數	38 腳
核心處理器	Tensilica Xtensa LX6
核心	雙核 160/240 MHz
資料寬度	32 位元

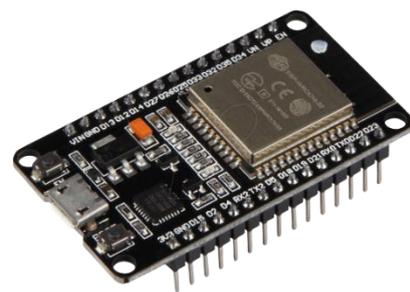


圖 13 ESP32 無線模組

2、超音波感測器

模組會送出 8 個 40kHz 的方波，如果前方有障礙物，信號就會返回，模組收到信號後，再利用返回的時間，去計算該障礙的距離，類似於聲納的原理。

表 3 超音波感測器規格

型號	HC-SR04
工作電壓	DC 5V
發射頻率	40kHz
探測距離	2cm~450cm
精度	±3 mm



圖 14 超音波感測器

3、紅外線避障感測器

紅外線避障感測器本身會發射特定頻率的紅外線，當紅外線被物體擋住後，紅外線就會反射，接收端會接收反射的紅外線，做出動作，並且可以透過電位器旋鈕調節檢測距離。

表 4 紅外線避障感測器規格

型號	IRS-180
工作電壓	DC 3.3~5V
檢測角度	35°
檢測距離	2cm~30cm



圖 15 紅外線避障感測器

4、不鏽鋼浮球開關

不鏽鋼浮球開關是靠著水的浮力來讓不鏽鋼軸心上的浮球上下移動，而不鏽鋼浮球內部有一個環型磁鐵，利用環型磁鐵來讓磁簧開關感應作動，用來作為開關使用。我們利用浮球開關將信號接到 ESP32 無線模組進行信號讀取，判斷是否到達滿水位或低水位。

表 5 不鏽鋼浮球開關規格

最大功率	10VA
最大開關電壓	100V AC/DC
最大開關電流	0.5A
使用溫度範圍	-30°C~120°C
材質	不鏽鋼



圖 16 不鏽鋼浮球開關

5、流量感測器

流量感測器主要由塑料閥體、水流轉子組件和霍爾感測器組成。可以裝在進水端，用於檢測進水流量，當水通過水流轉子組件時，磁性轉子轉動的轉速會隨水流量變化，霍爾感測器就會輸出相應脈衝信號，反饋給控制器，由控制器判斷水流量的大小。

表 6 流量感測器規格

型號	YF-S401
工作電壓	DC 5~24V
耐水壓	0.35MPa
流量範圍(Q)	0.15~3L/min
流量脈衝特性	$F=(98*Q) \pm 2\%$



圖 17 流量感測器

6、抽水馬達

使用抽水馬達模擬水塔抽水幫浦。DC24V 1.2A，抽水速率每分鐘 1.35 公升。

表 7 抽水馬達規格

型號	YS-5228-75
工作電壓	DC 24V
工作電流	1.2A
抽水速率	1.35L/min



圖 18 抽水馬達

7、繼電器模組

我們使用 8 路可以單獨控制每個繼電器的模組，作為大負載的控制開關，例如作為抽水馬達、直流電源回路等的控制開關。

表 8 繼電器模組規格

最大開關電壓	AC 125V/DC 30V
額定通過電流	10A
接點組成	8C
工作電壓	DC 5V
接點機構	單點



圖 19 繼電器模組

8、太陽能板

我們使用太陽能板，搭配太陽能充電電路板進行鋰電池充電，可以作為輔助電源。

表 9 太陽能板規格

額定電壓	DC 19V
額定電流	0.52A
尺寸（長）	345mm
尺寸（寬）	285mm
尺寸（厚）	17mm



圖 20 太陽能板

9、7 吋液晶觸控螢幕 HMI

我們使用較大尺寸的 7 吋液晶觸控螢幕 HMI，來作為操作介面設計，主要是考慮到尺寸太小的話，不太適合年長者操作。我們使用免費的 USART HMI 軟體來設計，並直接使用 RX/TX 串列傳輸方式燒錄到 HMI。HMI 也是利用 RX/TX 串列傳輸方式與 ESP32 無線模組進行通訊。

表 10 HMI 規格

廠牌	淘晶馳
型號	TJC8048K070_011R
尺寸	7 吋
解析度	800×480



圖 21 7 吋液晶觸控螢幕 HMI

10、溫度感測器

我們使用不銹鋼防水探頭型式的 DS18B20 溫度感測器，用來測量上水塔的水溫，而且 Arduino 已有內建 DS18B20 的函式庫可應用，對於溫度量測的開發非常方便，而且溫度量測的範圍廣，準確度也高。

表 11 溫度感測器規格

溫度感測器	DS18B20
量測範圍	-55°C~125°C
準確度	±0.5°C
探頭材質	不銹鋼



圖 22 溫度感測器

三、使用軟體與服務

(一)、Arduino IDE

Arduino IDE 是一個免費的整合式開發環境，其使用的語法和 C/C++ 相似，內建許多模組化的函式庫可供使用，可縮短開發時程，因此我們選用 Arduino IDE 來進行 ESP32 無線模組的開發。



圖 23 Arduino IDE 軟體介面

(二)、USART HMI

USART HMI 是一套免費的 HMI 人機介面設計軟體，可以結合淘晶馳 7 吋觸控式 LCD 模組，透過 UART 傳輸方式與 ESP32 模組進行通訊，方便 HMI 人機介面的開發，因此我們利用 USART HMI 軟體來設計專題的 HMI 人機介面。

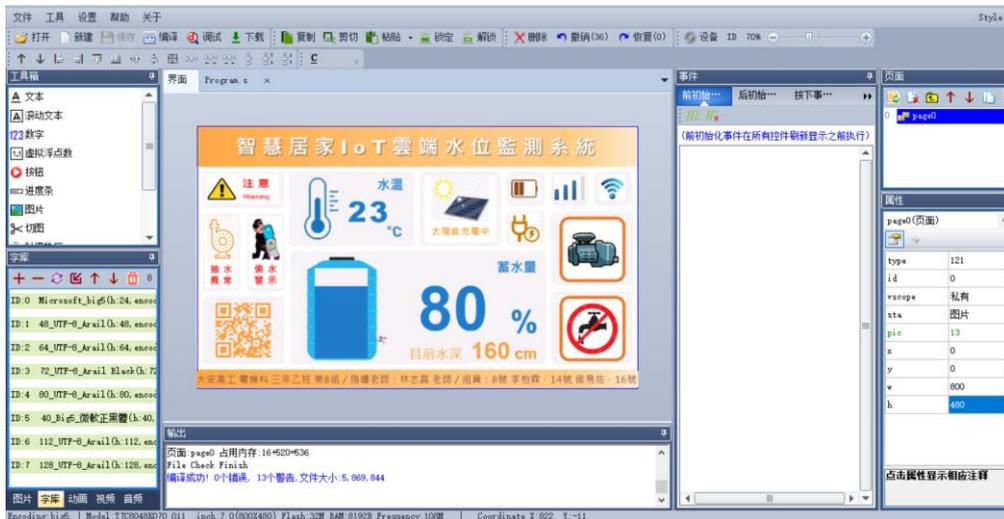


圖 24 USART HMI 軟體介面

(三)、Visual Studio

Visual Studio 是一套以 C/C++ 為基礎的整合式開發環境，它包括了整個軟體開發所需要的大部分工具，如程式碼管理工具等，同時也提供了偵錯以及單行執行的功能，可以讓我們能夠容易檢查程式的錯誤。我們使用 Visual Studio 開發環境，結合 Visual C# 的程式語言來進行網頁開發。



圖 25 Visual Studio logo

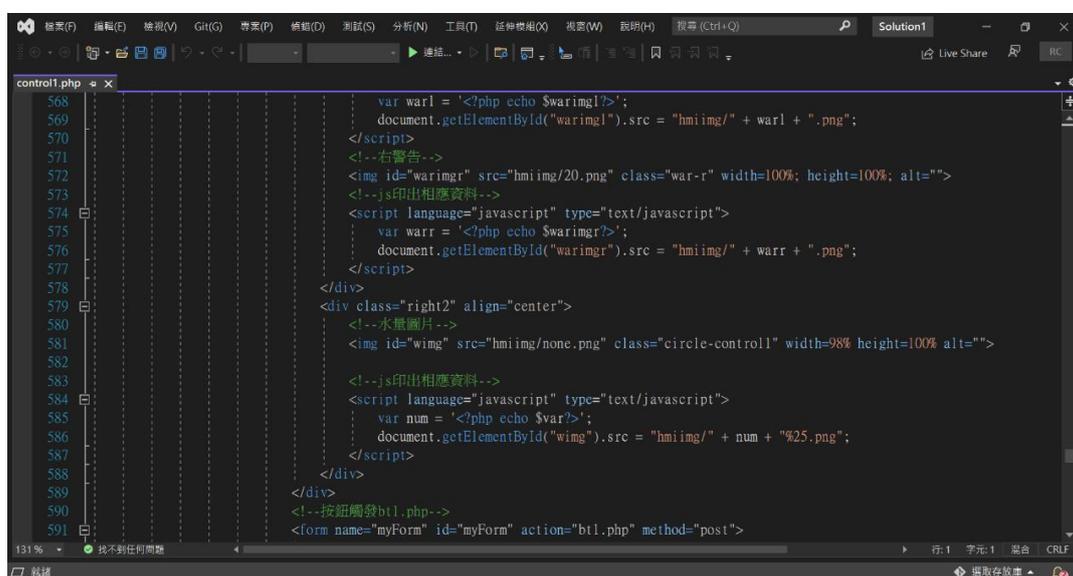


圖 26 Visual Studio 軟體介面

(四)、MQTT

MQTT 是 Message Queueing Telemetry Transport 的縮寫，是由 IBM 公司所制定的傳輸模式，透過 TCP/IP 協定進行資料傳輸，具有低功耗、低傳輸流量、非同步傳輸的特性，非常適合用在分散式的物聯網裝置，作為 WiFi 傳輸資料協定使用。

MQTT 通訊協定是透過一個代理伺服器「Broker」，利用「發布」和「訂閱」的方式來接收和傳送資料，只要兩端的裝置同時連線到 MQTT 的代理伺服器，不需要知道發佈者和訂閱者的 IP，就可以讓兩端進行通訊。



圖 27 MQTT logo

(五)、Apache

Apache 是一台支援許多特性的伺服器架設軟體，擁有通用的程式語言支援，Apache 紀錄檔可以通過網頁瀏覽器使用免費的指令碼 AWStats 或 Visitors 來進行分析。



圖 28 Apache logo

(六)、MySQL

MySQL 是一個小型的 C 語言嵌入式資料庫，特色是一個不需要建立的獨立資料庫系統，只要在撰寫網頁時，一起編入指令即可使用；而且它附有 Web 介面讓管理者以簡易方式輸入複雜 SQL 語言，這個 Web 介面也方便管理者遠端管理 MySQL 資料庫。



圖 29 MySQL logo

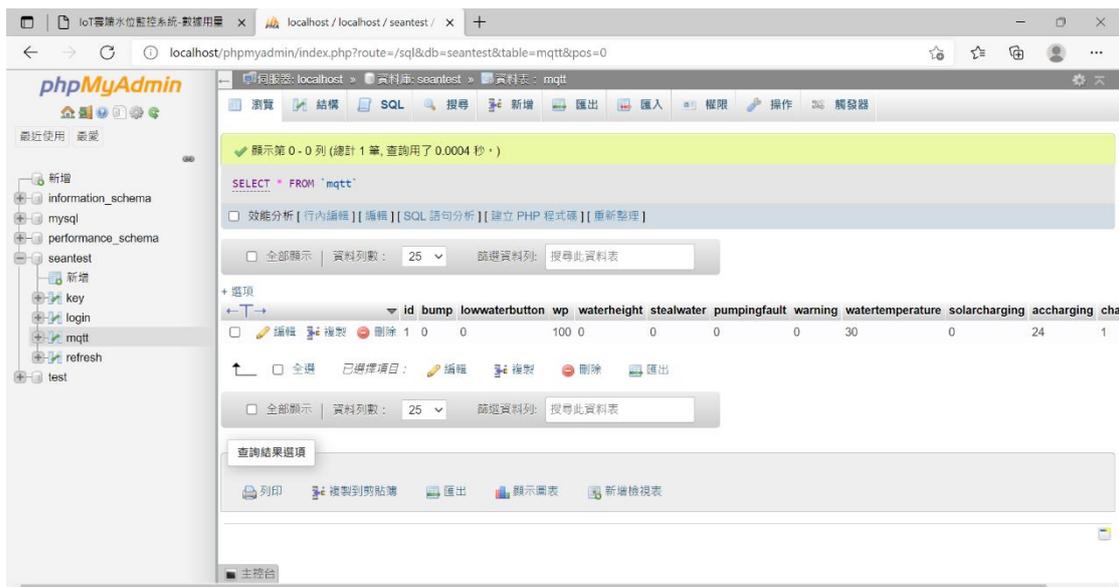


圖 30 MySQL 軟體介面

(七)、Altium Designer

Altium Designer 是一種電腦輔助電路設計軟體，可以進行電子零件、電路圖及 PCB 佈線設計，我們利用此軟體設計出結合 ESP32 無線模組、電源供應和繼電器控制的 PCB，轉檔後製作出電路板，相比於麵包板，PCB 電路板可以讓我們的電路系統更加穩定。



圖 31 Altium Designer logo

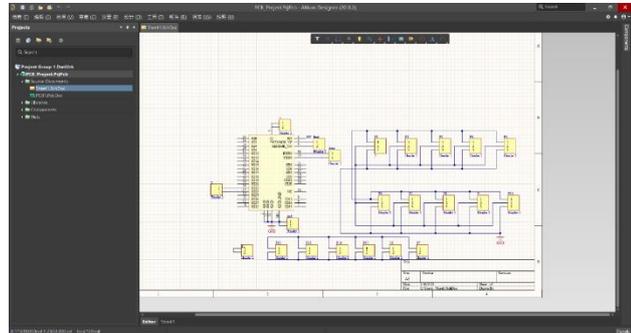


圖 32 Altium Designer 軟體介面

(八)、Autodesk Inventor

Autodesk Inventor 是一款用於 3D 建模的軟體，可以實現腦中的構圖，用於在作品初期模擬機構和設計外型，並在發表時讓觀眾們能夠更容易且充分的了解機構。

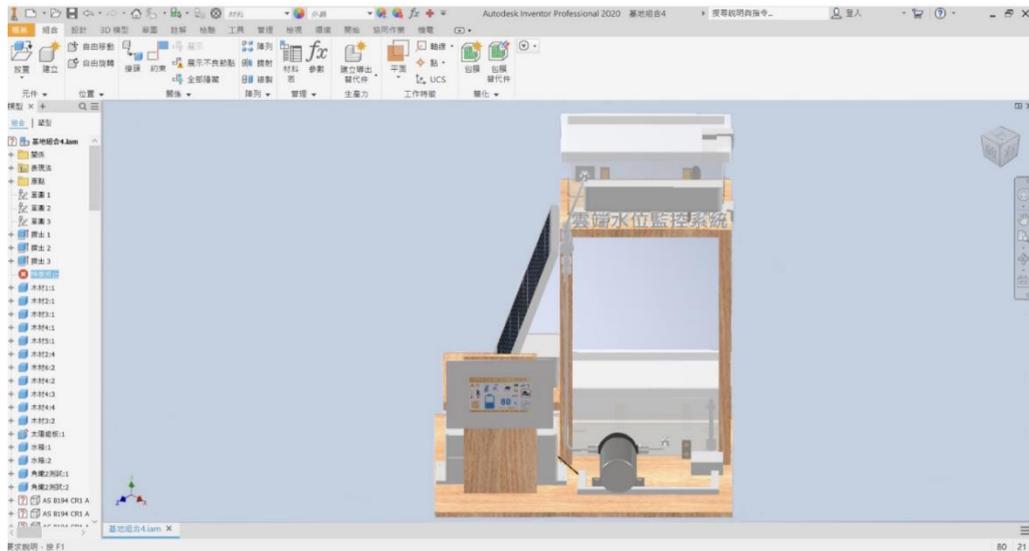


圖 33 Autodesk Inventor 軟體介面

(九)、RDWorks

RDWorks 是一款繪製雷射切割圖的軟體，我們使用此軟體繪製，並切割出部分機構的支架。

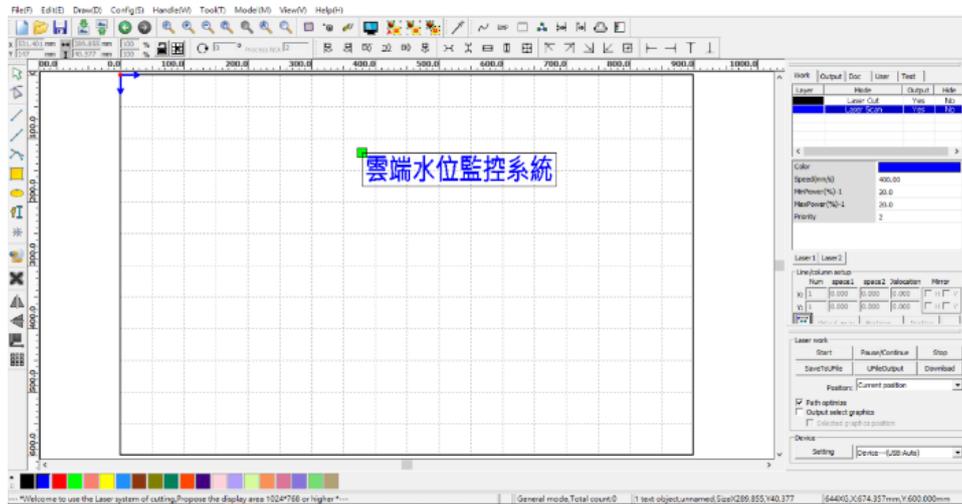


圖 34 RDWorks 軟體介面

(十)、Blender

Blender 是一款開源的 3D 繪圖軟體，可以繪製動畫、建模、渲染以及多種物理效果的模擬。我們使用此軟體在報告時以動畫呈現水路的流動以及機構的功能，讓觀眾獲得更多身歷其境的感受。

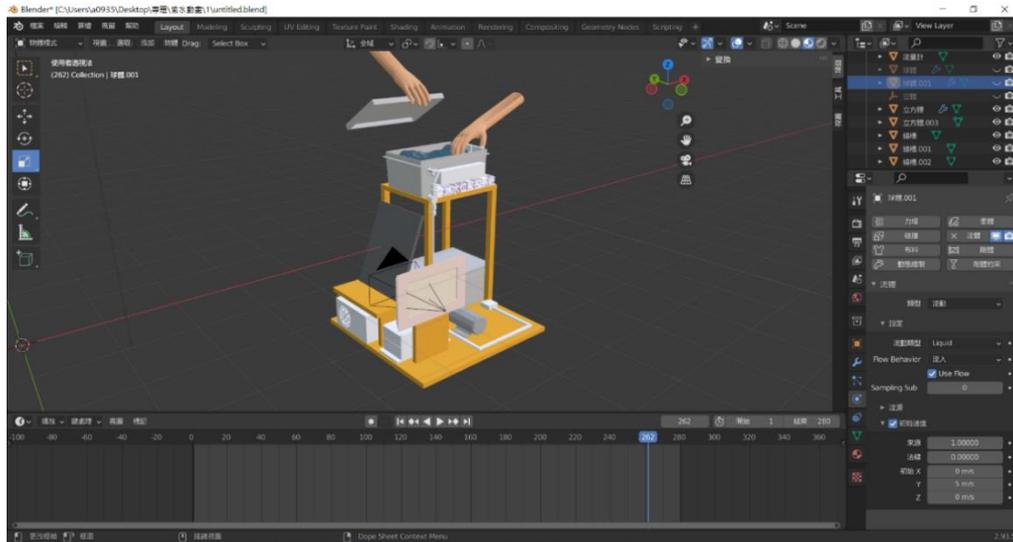


圖 35 Blender 軟體介面

二、使用設備

本專題使用了 3D 列印機、雷射雕刻機及電路板雕刻機等三款設備進行加工，設備的功能說明及外觀如下：

(一)、3D 列印機

3D 列印機是使用熔融堆積成型技術，把噴頭溫度加熱到 200°C，再把融化的 PLA 線材依照設計好的 inventor 3D 圖輸出到 3D 列印機，層層堆疊硬化後形成 3D 成品。

(二)、雷射雕刻機

雷射雕刻機是利用高功率的雷射光來進行掃描和切割加工，具有精準和快速的加工特性，被我們用在後期外觀的裝飾，像是位於水塔中心處的專題名稱，就是用雷射雕刻機掃描透透明壓克力板加工的，最後呈現出來的效果超乎預期的優異。另外還有位於左側的電源供應器蓋板，也是利用雷射雕刻機精準的加工木板外型，完美的契合在我們機構上。

(三)、電路板雕刻機

位於下防水盒裡面的電路板是用 Altium Designer 繪製電路圖及設計 PCB 佈線後，再利用電路板雕刻機把覆銅板不用的地方用各種不同尺寸的刀頭刮除，最後完成一塊兼具美觀和體積優勢的電路板，因為是用固定在 PCB 上的銅板來當成線路，因此穩定度遠勝於用杜邦線插麵包版。

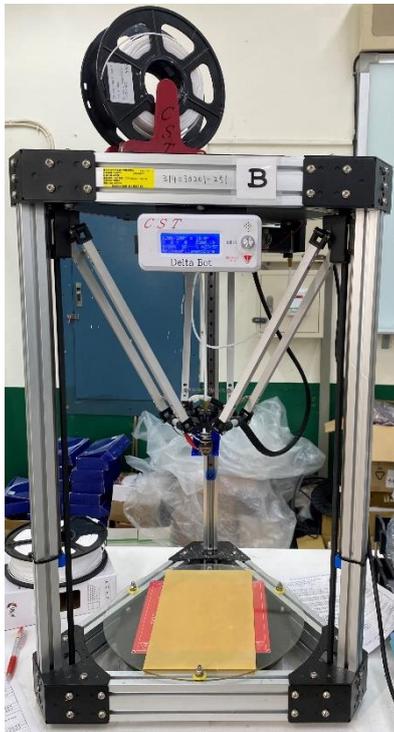


圖 36 3D 列印機



圖 37 雷射雕刻機

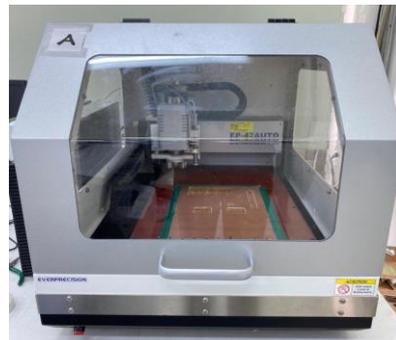


圖 38 電路板雕刻機

伍、研究結果

本專題是一套智慧居家物聯網雲端水位控制系統，主要可分為主體結構、HMI 觸控螢幕、邏輯電路、電源供應、MQTT 通訊協定、網頁連線等。

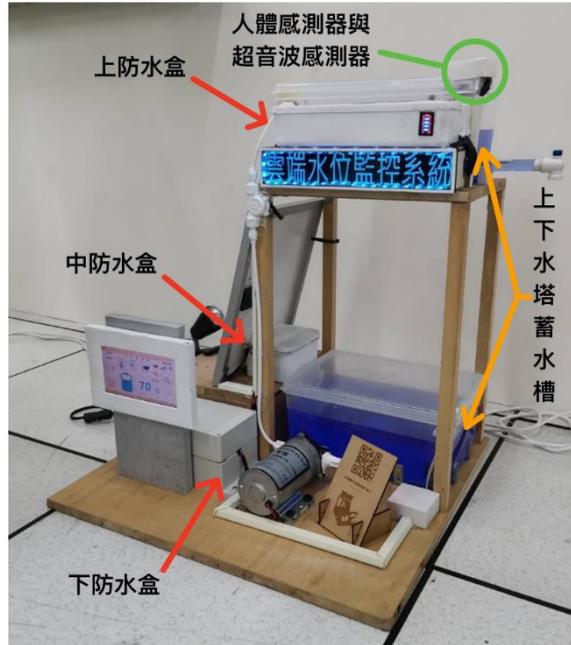


圖 39 專題成品

一、主體結構

主體結構是由 11 塊木板裝配組成支撐定位座，依定位區塊可分為下水塔、上水塔、上防水盒、中防水盒、下防水盒。

(一)、下水塔

模擬作為自來水供水系統進入居家的下水塔蓄水區，內部有 1 支浮球開關，用來偵測下水塔水位是否過低，判斷是否缺水，作為強制關閉抽水馬達的保護信號，防止抽水馬達空轉過久燒毀。

(二)、上水塔

模擬作為居家住宅的上水塔蓄水區，由下水塔經水路、抽水馬達、流量感測器等，將下水塔的蓄水抽到上水塔儲水。流量感測器用來判斷抽水馬達動作是否正常，作為抽水馬達的第二道保護。

上水塔內置超音波感測器量測水位高度及蓄水量，防水型溫度感測器量測水溫，紅外線感測器偵測人體，作為偷水偵測判斷。內部有 1 支浮球開關，用來偵測上水塔水位是否過低或滿水位，作為超音波感測器信號失效的二次保護。

(三)、上防水盒

用來固定上水塔主控系統的 ESP32 無線模組及電池電量模組。

(四)、中防水盒

用來固定太陽能充電模板及鋰電池組。

(五)、下防水盒

用來固定屋內主控系統的 ESP32 無線模組及 8 路繼電路模組。

二、HMI 觸控螢幕

以 7 吋 LCD 觸控螢幕作為 HMI 操作介面，主要可分為三大功能區塊，包括抽水控制區、即時監控區及警示區。

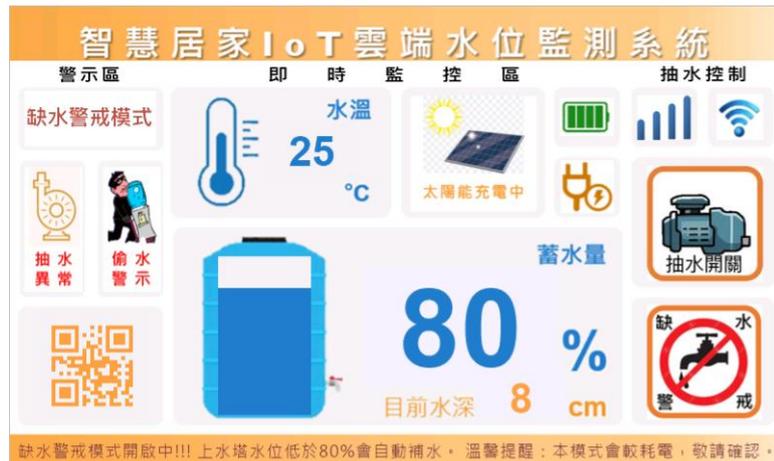


圖 40 HMI 觸控螢幕功能區塊

(一)、抽水控制區

一般開機是處於「自動模式」，上水塔水位低於 20%自動補水，高於 60%停止補水。按下抽水開關圖示會進入「手動模式」，可單次將上水塔水量直接補滿到 100%，之後會自行切換回「自動模式」。在獲得停水通知或颱風警報發佈前，可以按下缺水警戒圖示進入「缺水警戒模式」，上水塔會將原本 60%的高水位提升至 80%，當水位低於 80%自動補水，讓水位維持 80%。另外 HMI 也可提供監控顯示上水塔無線控制模組的 WiFi 連接狀態、WiFi 無線信號強度、太陽能充電狀態、交流電源供電狀態、鋰電池蓄電狀態、即時水溫、蓄水量、水位高度、設定模式、抽水馬達異常警示、偷水警示、監控網頁 QR code 及即時資訊等顯示。

(二)、即時監控區

即時監控區可即時提供上水塔的資訊，包括 ESP32 無線模組連線狀態、WiFi 無線信號強度、太陽能充電狀態、交流電源供電狀態、鋰電池蓄電狀態、水溫、蓄水量、水位高度等資訊等顯示。

(三)、警示區

警示區可提供目前設定模式、抽水馬達異常警示、偷水警示、監控網頁 QR code 及警示資訊。

三、邏輯電路

邏輯電路主要可分為二大區塊，包含上水塔主控系統和屋內主控系統。

(一)、上水塔主控系統

上水塔主控系統主要是以上水塔的 ESP32 無線模組為控制核心，負責讀取上水塔的溫度感測器信號來判斷水溫；控制超音波模組的發射及讀取超音波信號來換算水位高度及蓄水量；讀取上水塔浮球開關信號，判斷上水塔滿水位狀態，作為超音波模組失效時的二次保護；讀取太陽能板充電狀態；讀取鋰電池電壓，判斷蓄電量；判斷鋰電池電量不足時，控制繼電器模組切換成交流電源供電及對鋰電池充電；讀取交流電源供電狀態；接收流量感測器信號，判斷抽水馬達動作是否正常；偵測 WiFi 連線狀態；偵測 WiFi 無線信號強度；將上述所有信號，透過 ESP-NOW 通訊技術及協定，將傳送到居家室內的 ESP32 無線模組。

(二)、屋內主控系統

以居家室內的 ESP32 無線模組為控制核心，負責接收上水塔的 ESP32 無線模組所傳送的信號；利用 UART 傳輸方式，依通訊協定傳給 HMI 進行顯示；讀取 HMI 的觸控操作事件；讀取下水塔浮球開關信號，判斷下水塔缺水狀態，作為抽水馬達停止動作的保護，避免抽水馬達長時間空轉造成過熱損壞；控制抽水馬達的動作；負責 WiFi 網路對外連線，將資料傳到自建的網路伺服器資料庫系統。

四、電源供應

提供二種電源供應方式，包含交流供電系統與太陽能供電系統。

(一)、交流供電系統

交流供電系統主要是提供交流電源轉換成直流電源，提供給抽水馬達和邏輯電路的電源供應。

(二)、太陽能供電系統

利用 10W 的太陽能板連接太陽能充電模組電路，再接到鋰電池組進行充電儲能，可作為上水塔主控系統及週邊硬體電路額外的供電來源。太陽能供電的設計主要是考慮到未來應用在郊外場合，如水庫、農漁業等，太陽能供電方式較交流電源容易，且安全又環保。同時也可作為上水塔主控系統在白天的供電來源，達到節能省電。

五、MQTT 通訊協定

我們使用 Mosquitto 作為伺服器軟體，把固定 IP 透過分享器設定，將 1883 埠導至伺服器上，實現不限距離的無線傳輸。只要在撰寫程式時，定義好網頁端和硬體端的通訊協定，就可以達成利用網頁來監控系統，並且在未來更新時，擁有良好的擴充性。



圖 41 mosquitto logo



圖 42 MQTT 傳輸過程

六、網頁連線

我們的網頁建置是使用 Apache 及 MySQL 來架設。透過這兩個軟體將我們編寫好的網頁以及資料庫架設在伺服器上，並在分享器中把 80 埠導向給網頁伺服器主機，之後申請一個免費的專用網域名稱 <http://www.tankiot.tk> 及 DNS 服務，實現無論身在何處，只要有網路，就可以無遠弗屆的遠端操控雲端水位監控系統。



圖 43 網站登入畫面



圖 44 網站監控頁面

七、成果展示

(一)、成品外觀

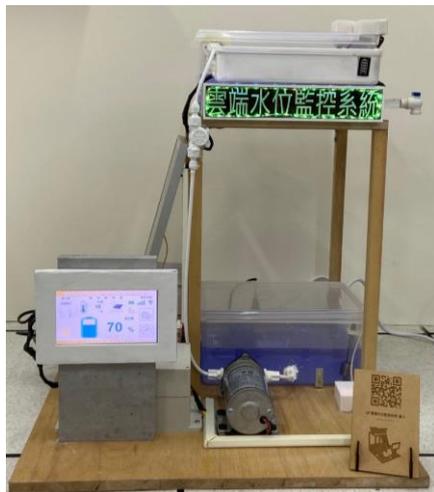


圖 45 專題成品外觀

(二)、HMI 介面

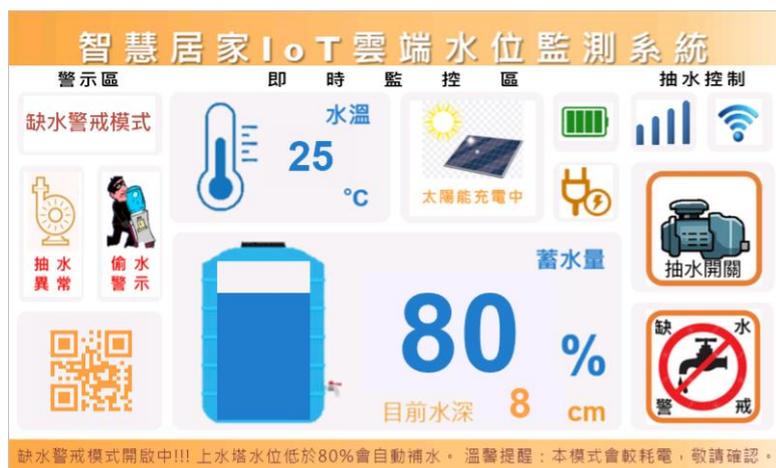


圖 46 HMI 介面設計



圖 47 HMI 實品

(三)、偷水偵測及 Gmail 通知



圖 48 偷水示意圖



圖 49 Gmail 通知

(四)、網頁監控即時顯示

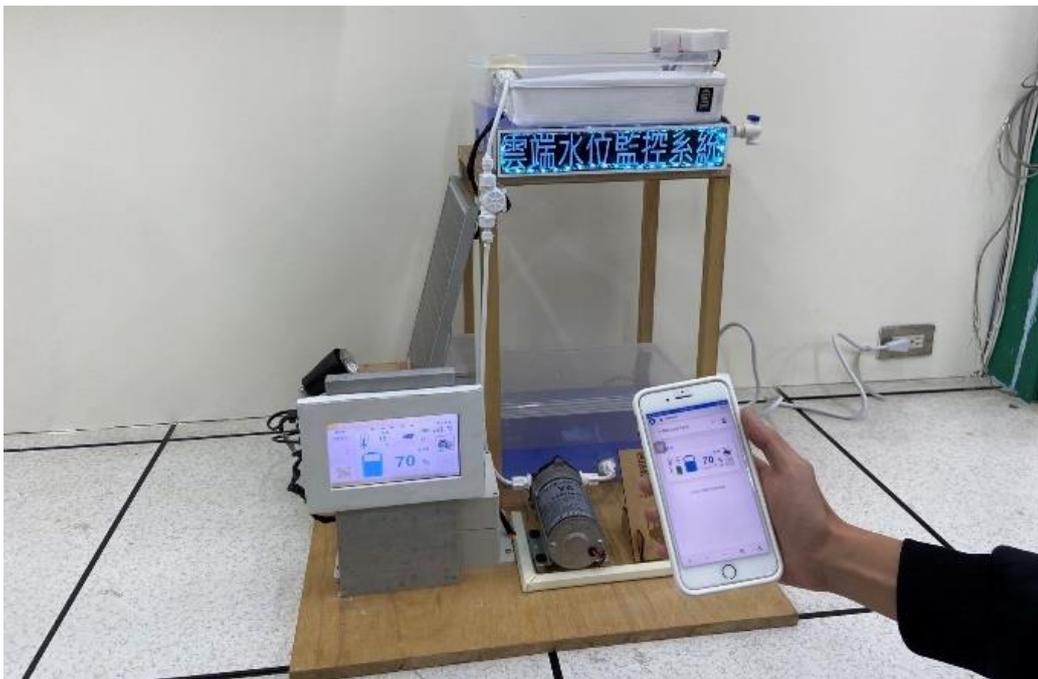


圖 50 網頁監控即時顯示

(四)、大數據分析

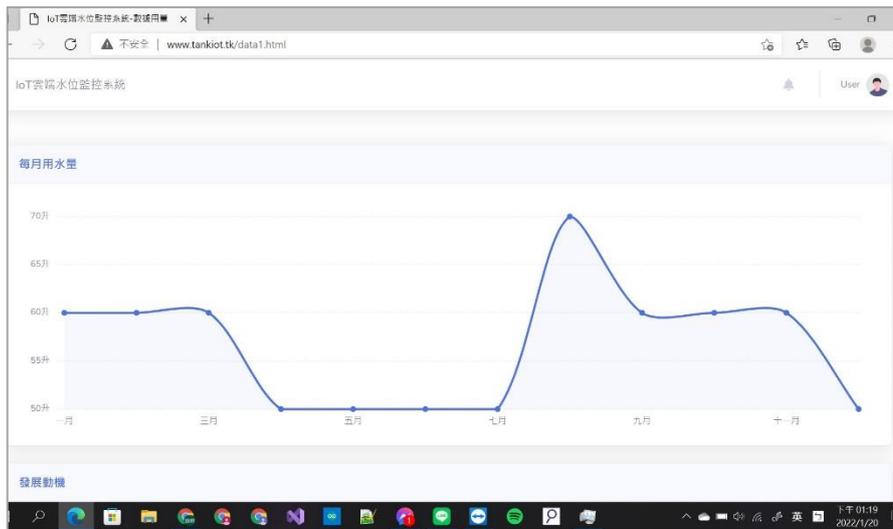


圖 51 資料庫紀錄的每月用水量大數據分析



圖 52 手機端的網頁監控與大數據顯示畫面

(五)、異常通知紀錄與網頁連結分享

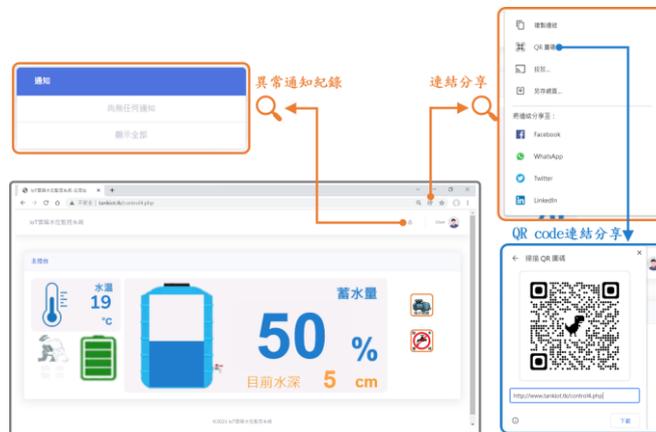


圖 53 異常通知紀錄與網頁連結分享

陸、討論

一、水管選用

在專題初期評估水路水管材料選用時，我們針對 PVC 硬管和矽膠軟管兩種材料進行評估，最後選定使用矽膠軟管來設計水路，主要是矽膠軟管的水路配置較靈活，可縮短專題開發時程，而且不易受搬運外力造成機械結構扭曲影響，若是開發過程中發生漏水狀況時，可快速維修而不影響專題整體進度，也容易擴充，若是發生漏水突發狀況，補救也較容易、維修工具較易取得。PVC 硬管體積大、材質硬，而且專題使用兩個水箱，搬運時容易發生脫膠，造成水箱與水管結合處容易破裂漏水，甚至可能直接扯破整個水箱側板。另外我們使用 10W 的抽水幫浦，若選用 PVC 水管的話，因管徑大、抽水揚程負載大，可能會造成抽水效果不佳、抽水時間過長，影響專題開發效率。

二、ESP32 無線模組傳輸問題

我們一開始是使用 WiFi_AP (網路基地台) 的模式，以區域網路內網伺服器方式架設網站，後來發現該模式有傳輸距離限制，造成專題的應用受限，考量未來專題應用的擴充性及實用性，可以讓使用者任意從遠端監控等需求，因此我們後來改成 WiFi_STA (無線終端) 模式，讓其可以透過 MQTT Broker 連線至伺服器並與其溝通，達到真正的無距離限制的遠端監控應用。

三、電量顯示

我們原本是使用電量感測顯示模組測量後，將燈號的信號線拉出來後，接到上水塔的 ESP32 無線模組收集資料，但是我們後來發現從模組上拉出來的信號電位過低，導致 ESP32 無線模組無法準確辨認其電量，所以我們後來多購入了一個光耦合器電路模組做中繼後，放大電量感測模組測量後的信號大小，讓 ESP32 無線模組可以準確的讀取。



圖 54 電量感測顯示模組

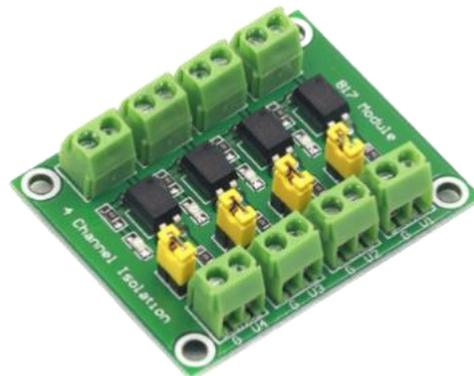


圖 55 光耦合器電路模組

柒、結論

本專題原先是想要設計出一套可以提供居家控制的水塔水位監控系統，讓年長者在居家室內就可以透過 HMI 來控制水塔的抽水馬達動作，方便用來作為停水前的儲水控制，以及颱風侵襲前的上水塔儲水加載荷重，避免上水塔被颱風吹落衍生安全問題；同時還能從 HMI 上即時得知上水塔水位及蓄水狀況，甚至即時監控抽水馬達有無空轉異狀、抽水過程是否異常等，完全不需要攀爬到頂樓上水塔去確認，應用在學校所學習的知識與技能來改善生活問題。

隨著專題製作不斷地發展，我們深入思考這套系統是否也能讓熟悉科技產品操作的年輕人一同參與使用，因此增加了物聯網與雲端控制的概念，想要讓使用者也能透過手機、平板電腦或桌上型電腦，利用 WiFi 無線信號來控制水塔水位及取得水位資訊，並且能在異常狀況發生時即時傳送 e-mail 通知提醒，甚至在遠端也能夠隨時登入網站來進行監控。隨著專題成品功能不斷地實現，我們發現智慧居家物聯網的構想可能會侷限我們的技術發展，於是我們自我要求地增加了 WiFi 連線狀態顯示、WiFi 無線訊號強度偵測、溫度監控、太陽能儲能供電系統、充電電池電量顯示、專用網站網址申請、自架網路伺服器及自建資料庫系統，最後發展至水情資訊大數據統計分析。

本專題的研究並非僅侷限應用在智慧居家物聯網雲端水位監控，只要依照使用場合與環境，改用合適的感測器，例如防水、防潮、耐壓、耐寒、耐高溫、耐酸鹼等功能的感測器，再結合相關硬體控制，就可以更廣泛地應用在科技業、工業、石化業、農林業、漁牧養殖業及水庫等，成為多功能的智慧物聯網雲端監控系統，再結合自架的網路伺服器資料庫大數據分析與專屬網頁的資訊呈現，相信本專題未來延伸的發展之路是無可限量的。期望未來進入科技大學就讀時，能深入學習 AI 人工智慧及相關演算法，加強無線通訊及高速運算技術，進一步整合擴展本專題成果，延伸技術持續應用發展。

捌、參考資料及其他

一、書籍資料

1. 尤濬哲(2021.08)。IoT 物聯網應用 - 使用 ESP32 開發板與 Arduino C 程式語言。新北市：台科大圖書股份有限公司。
2. 裘炯濤、陳眾聚(2019.11)。物聯網，So Easy! 基於 Blynk 平台的 IoT 項目實戰。北京市：人民郵電出版社。
3. 黃峰達(2018.08)。物聯網原來這麼近：立即手動實作一個。臺北市：佳魁資訊股份有限公司。
4. 林聖泉(2018.07)。從 Arduino 到 AVR 微控制器 - 嵌入式系統原理與應用。臺北市：旗標科技。
5. 趙英傑(2016.05)。超圖解物聯網 IoT 實作入門：使用 JavaScript/Node.JS/Arduino/Raspberry Pi/ESP8266/Espruino。臺北市：旗標出版股份有限公司。

二、網路資料

1. php 實現 mqtt 釋出/傳送訊息到主題。2021 年 12 月 11 日。取自 <https://reurl.cc/MbXjkn>
2. php-mqtt/client examples。2021 年 12 月 11 日。取自 <https://reurl.cc/qOZvO0>
3. How to Connect HTML Form with MySQL Database using PHP。2021 年 12 月 12 日。取自 <https://reurl.cc/Zr1Vrg>
4. HTML <link href="" rel="stylesheet">標籤(tag)-載入 CSS 樣式表。2021 年 12 月 12 日。取自 <https://reurl.cc/KpXlpm>
5. phpmailer 運用 gmail 來寄信。2021 年 12 月 18 日。取自 <https://reurl.cc/GoXpxd>
6. 【PHP】用 php 修改 html 屬性。2021 年 12 月 18 日。取自 <https://reurl.cc/MbXj0X>
7. 【PHP】如何自動提交複選框。2021 年 12 月 19 日。取自 <https://reurl.cc/VjRMDN>
8. CSS 位置(Position)屬性。2021 年 12 月 19 日。取自 <https://reurl.cc/zMrDZp>
9. 使用 onclick 執行 PHP 函式。2021 年 12 月 25 日。取自 <https://reurl.cc/l9Zyol>
10. javascript-點擊按鈕改變圖片。2021 年 12 月 25 日。取自 <https://reurl.cc/pWZv14>