

# 2024仰望盃全國科學HomeRun實作大賽

## 決賽成果報告書

隊伍名稱: 菸鹼醯胺腺嘌呤二核苷酸磷酸

---

作品名稱: 緊急車輛號誌優先系統

---

科學概念1: 衛星導航系統(GNSS)

衛星導航系統(Global Navigation Satellite System, GNSS)是利用地球軌道上的衛星提供使用者導航、定位和時間服務。包括GPS、GLONASS、伽利略和北斗等,利用三角測量原理,通過測量從衛星到接收器的信號傳播時間來確定接收器的位置。

---

科學概念2: 訊息佇列遙測傳輸(MQTT)

訊息佇列遙測傳輸(Message Queuing Telemetry Transport, MQTT)是用於物聯網的通訊協定,分成訂閱者和發布者,並利用MQTT Broker來當作通訊中介,進行物聯網(IOT)的訊息傳輸。其中分成不同的服務品質(QoS),在這系統中,我們使用QoS 2來確保訊息都有進行準確的傳輸,避免傳輸過程失敗。

# 決賽成果報告書內文

## 壹 發想動機：

「一台載送患者轉院的民間救護車，行經XX路時，闖紅燈通過一處路口，另向一輛黑色轎車綠燈直行，疑似未發現有救護車，當場攔腰撞上救護車，造成6人輕重傷，其中轎車上的70歲婦人一度命危送醫後救回一命，雙方都表示不知車禍如何發生，事故原因仍待警方調查釐清。」

當我們在路上行車時，總是會先聽到緊急車輛救護車的緊急鳴笛聲，但是無法立即判斷其方向，即使駕駛人想要禮讓，也無法立即做正確反應，等到確定緊急車輛的方向時，往往沒有足夠的時間去閃避緊急車輛，有時反而會導致另一事件發生。

雖然目前已經推出緊急車輛優先號誌，但現有的是改變號誌的秒數使緊急車輛通行，但這樣卻會讓其他與緊急車輛同方向及對向的普通車輛會進入路口，導致緊急車輛無法順利通過路口，其他用路人也難以得知救護車的方向，無法提早避讓。

但假設將所有將進入路口的車輛阻擋，並告知用路人緊急車輛的方向，就能夠預防其它車輛進入路口阻擋，提早靠邊避讓，能避免緊急車輛減速或閃避而導致救援延誤等問題。因此我們希望能改良現有的緊急車輛的號誌通行系統，來改善上方所述的窘境。

## 貳 作品創意性：

### 一、保持路口通暢

相較於現有的系統，我們改用特殊燈號來告知用路人緊急車輛的方向，並阻擋其它車輛進入路口，能使用路人提前知道緊急車輛的方向，以解決車輛阻擋路口問題，增加緊急車輛通過的安全性。

### 二、使用設備門檻低

緊急車輛端只需要在車輛放置具有網路連線功能的手機，再安裝APP即可，並放在緊急車輛內就能傳資料給號誌，降低系統所需的材料與空間，也能減少安裝系統的難度。

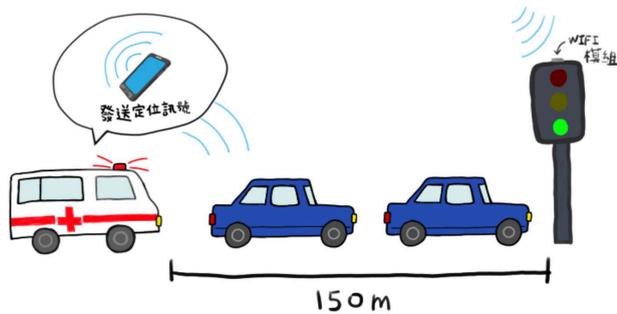
### 三、利用現有號誌

我們是利用現有的號誌來建置，無需增設其他號誌，能縮短安裝系統的成本。當決定普及時，能快速的擴展本系統。對於用路人而言，由於燈號上沒有重大變動，容易去適應本系統。

參 硬體及電路架構圖：

### 一、系統架構

將手機APP放置在緊急車輛內，並且開啟APP設定目的地之後，APP會判斷需變號的號誌，並利用網際網路與MQTT Broker進行資料傳輸，緊急號誌端則藉由ESP32取得號誌訊號後，便可控制號誌燈號。



圖一 系統架構示意圖

### 二、手機APP

利用MIT AppInventor製作緊急車輛端的APP，並且在使用者開啟APP並選擇目的地之後，會利用GNSS定位，並規劃到目的地的路徑，並檢視是否有重疊的號誌，進而判斷與其距離並進行聯繫。

#### 1.MQTT:

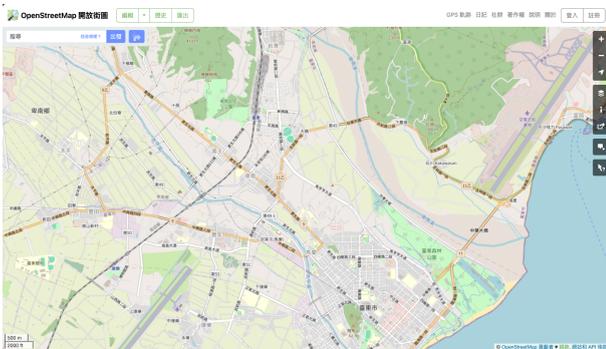
利用App Inventor的UrsPahoMqttClient擴充套件進行與MQTT Broker的連線，並將服務品質(QoS)設定為QoS2(訊號只進行一次傳輸)，以避免受到干擾又能準確的傳輸。

```
當 UrsPahoMqttClient1 .ConnectionStateChanged
  NewState  StateString
  執行 設 開關1 .在 為 真
        呼叫 UrsPahoMqttClient1 .PublishEx
          Topic "TS0001"
          Message "App Connect"
          RetainFlag 真
          QoS 2
```

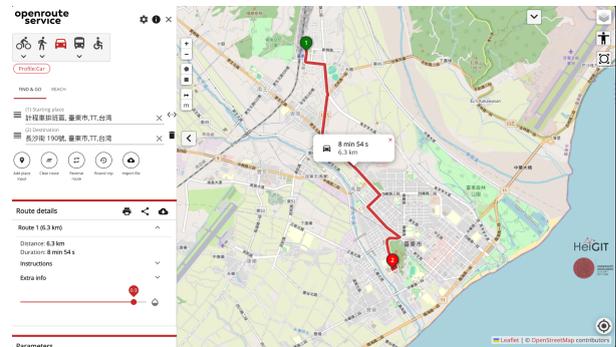
圖二 UrsPahoMqttClient程式示意圖

## 2.地圖與導航系統：

本次系統利用開放街圖(Open Street Map)作為緊急車輛導航系統的地圖，開放街圖是由世界各地群眾共同繪製的地圖，具有快速更新且方便使用等優點。導航伺服器則是使用OpenRouteService，是基於開放街圖所規劃路徑的導航系統，開源且可以選擇不同模式。



圖三 開放街圖示意圖



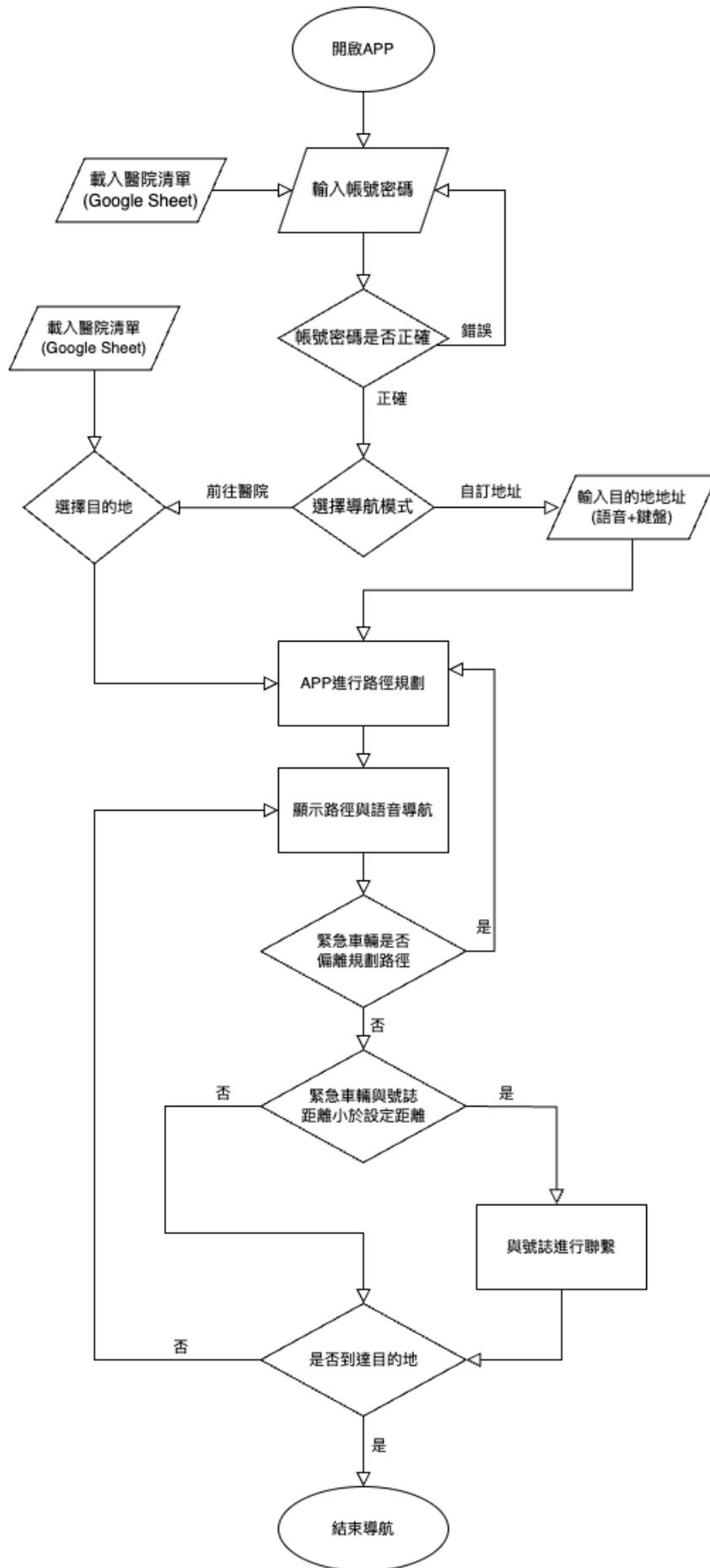
圖四 OpenRouteService示意圖

## 3.判斷經過號誌系統

利用App Inventor中地圖元件，判斷所繪製的路徑是否距離紅綠燈小於10公尺，如果小於10公尺，則將號誌新增到經過號誌的清單中，並主動判別與號誌之間的距離。



圖五 判斷經過號誌程式示意圖(節錄)

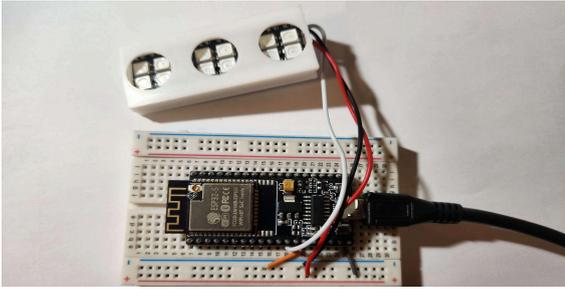


圖六 APP系統流程圖

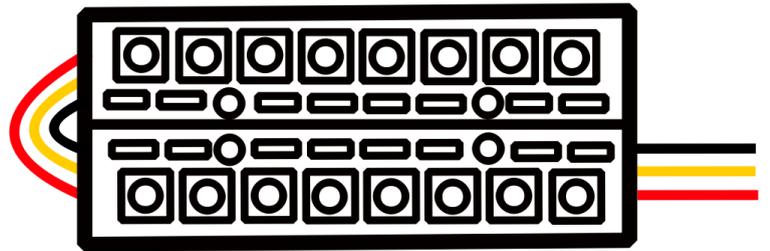
### 三、號誌系統

ESP32作為系統主機並使用Arduino IDE來撰寫本系統的程式，將號誌系統分成兩個部分，分別為負責顯示燈號的號誌部分，以及控制號誌的主機部分。

1.號誌設計：利用FastLED.h函式庫撰寫控制WS2812B燈條的程式，再將燈條與ESP32連接，模擬本系統的燈號。最後加上將燈條加上3D列印外殼，以避免裝置脫落。



圖七 模擬號誌示意圖



圖八 燈條示意圖

2.控制主機設計：利用WiFi.h使ESP32連線到WiFi網路，並且與MQTT Broker僅行聯繫，如有從MQTT Broker取得手機APP所送出的訊號，則根據收到的訊息進行判斷變更號誌。

```
1 #include <WiFi.h>
2 #include <PubSubClient.h>
3
4 const char* mqtt_server = "mqttgo.io";
5 const int mqtt_port = 1883;
6
7 const char* client_id = "ts0001";
8
9 const char* topic = "TS0001";
10 void callback(char* topic, byte* payload, unsigned int length) {
11     Serial.print("Message arrived on topic: ");
12     Serial.println(topic);
13     Serial.print("Message: ");
14     for (int i = 0; i < length; i++) {
15         Serial.print((char)payload[i]);
16     }
17     Serial.println();
18 }
19
20 WiFiClient espClient;
21 PubSubClient client(espClient);
22
23 void setup() {
```

圖九 ESP32連線MQTT Broker示意程式

## 肆 作品成果報告：

### 一.手機APP

利用MIT App Inventor製作緊急車輛端的APP，緊急車輛駕駛登入系統並選擇目的地之後，APP會自動規劃路徑以及與號誌聯繫，並提供語音播放導航，使駕駛能將視線專注於道路狀況。

	A	B	C
1	username↓	password↓	備註
2	gibbon0813	www1234	A車
3	camel0316	qwq12321	B車
4	homerun1234	hohome88	C車
5	Administrator	goodpassword	管理員

圖十 Google試算表資料庫示意圖

#### 1.登入系統

開啟APP之後，為防止遭到不當使用，本系統將會要求使用者輸入帳號密碼。密碼儲存在Google試算表，方便進行管理。



The screenshot shows a mobile application interface for login. At the top, there is a status bar with the time 12:41, signal strength, and 5G connectivity. Below the status bar, there are two input fields: '帳號' (Account) and '密碼' (Password). A '送出' (Submit) button is located below the password field.

圖十一 登入介面

#### 2.選擇導航模式

根據目的地的不同，可以選擇給「自訂地址」以及「前往醫院」模式。

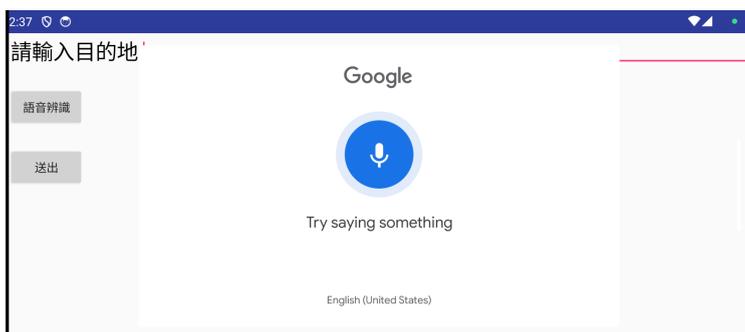


The screenshot shows a mobile application interface for selecting a navigation mode. At the top, there is a status bar with the time 12:20, signal strength, and 5G connectivity. Below the status bar, there are two large buttons: '自定地址' (Custom Address) and '前往醫院' (Go to Hospital). A '登出' (Logout) button is located at the bottom left.

圖十二 選擇模式介面

### 3-1自訂地址模式

當欲前往的目的地非在醫院清單內時，可利用自訂地址模式來設定，利用語音輸入，並且藉由鍵盤來修正輸入的文字。



圖十三 自訂地址模式

### 3-2前往醫院模式

當目的地在醫院清單內時，可以從清單選擇器中選擇欲前往的醫院，節省輸入地址時間，加速救援進行。



圖十四 選擇醫院清單

(備註：醫院清單儲存於Google試算表資料庫中，方便進行管理。)

	A	B	C
1	馬偕醫院	22.751028837184574,121.14144710244152	臺東縣台東市長沙街303巷1號
2	臺東醫院	22.757756458187647,121.15086110051787	臺東縣台東市五權街1號
3	榮民醫院	22.771682385036033,121.13257557381064	臺東縣台東市更生路1000號
4	基督教醫院	22.76415290608553,121.14652633447558	臺東縣台東市開封街350號
5	測試用醫院	22.79778852961966,121.12417891022908	臺東縣台東市樂利路00號

圖十五 醫院清單

#### 4.APP自動導航

在APP取得目的地位置之後，將進行規劃前往目的地的路徑，並自動分析路徑上會經過的號誌，進而與號誌進行聯繫。



圖十六 自動導航介面

## 二、號誌系統

### 1.連線到MQTT Broker

當系統開機時，根據設定的資料主動連線到WiFi網路，並與MQTT Broker進行聯繫，並訂閱指定主題，接收來自APP的訊號。

### 2.根據收到的訊號分析燈號

控制主機會根據從MQTT Broker取得訊息，自動變換號誌。

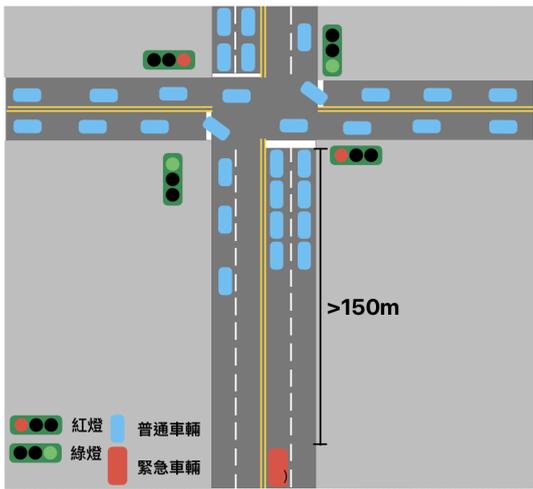
1.當緊急車輛距離號誌超過150公尺時，號誌維持原燈號。

2.當緊急車輛距離號誌150公尺時，綠燈方向變成黃燈，紅燈方向維持紅燈。

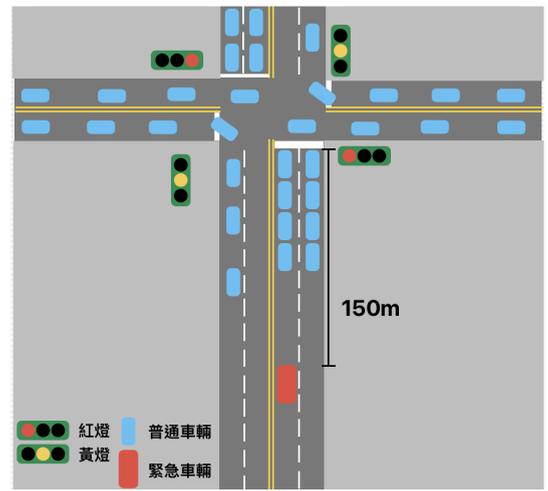
3.當距離號誌100公尺時，將所有號誌變成紅燈，並把緊急車輛來處的號誌變為緊急號誌(紅燈加上閃黃燈)以通知用路人，所有車輛及行人立即停止通行。

4.當號誌已經完全變號時，車輛有時間進行閃避，能夠讓普通車輛駕駛員及行人，有足夠的時間進行閃避。

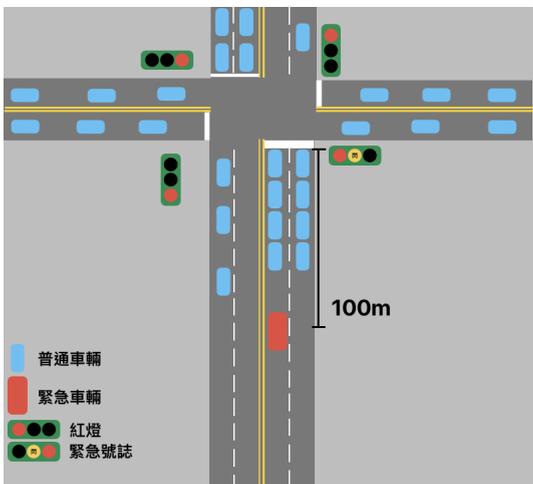
5.當緊急車輛通過後，APP會將該路口標記為已通過，號誌變為原本的循環。



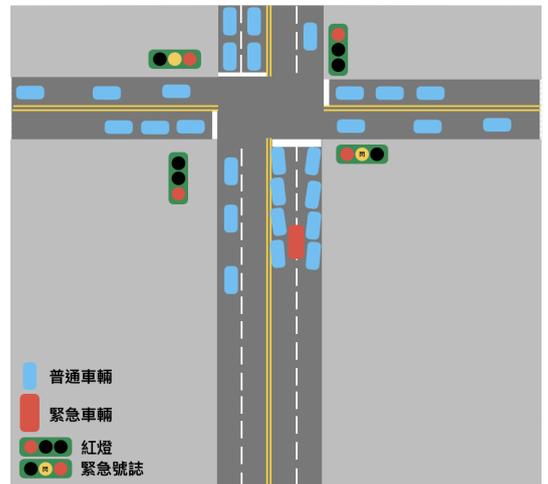
圖十七 號誌正常循環



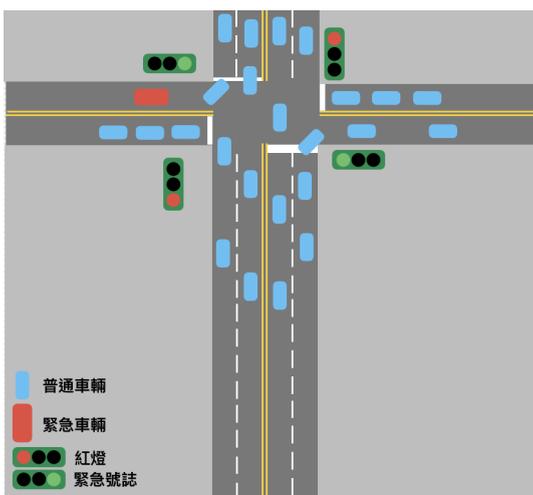
圖十八 準備改變燈號



圖十九 改變成緊急號誌



圖二十 緊急號誌通過路口



圖二十一  
緊急車輛號誌通過路口 恢復原循環

## 伍 參考文獻：

### 一、台北市政府新聞稿 臺北市建置優先號誌助緊急車輛安全快速到達

[https://www.gov.taipei/News\\_Content.aspx?n=F0DDAF49B89E9413&s=FDF918FEFC89E74F](https://www.gov.taipei/News_Content.aspx?n=F0DDAF49B89E9413&s=FDF918FEFC89E74F) (12.09.2023)

### 二、高雄市政府交通局 即刻救援不容堵，優先通行暢無阻 鳳頂路四處路口試辦緊急車輛優先通行號誌

<https://www.tbkc.gov.tw/Message/Bulletin/News?ID=92fa3ad9-57d5-40e2-bf3e-dcc5ef6d5701>(12.09.2023)

### 三、國家時間與頻率標準實驗 全球導航衛星(GNSS)時間傳遞

[https://www.stdtime.gov.tw/chrono/index\\_3\\_1.html](https://www.stdtime.gov.tw/chrono/index_3_1.html) (12.20.2023)

### 四、Wi-Fi聯盟(03.22.2023)

<https://www.wi-fi.org/zh-hant>

### 五、IBM-Processing MQTT messages(03.22.2023)

<https://www.ibm.com/docs/en/app-connect/12.0?topic=applications-processing-mqtt-messages>

### 六、MQTT: The Standard for IoT Messaging (03.20.2024)

<https://mqtt.org/>