

Thingsboard 的樹莓派物聯網閘道器實作

IOT Gateway Implementation Based on Raspberry with Thingsboard

黃樹林

Shu-Lin Hwang

明志科技大學電子工程系副教授

摘要

近年來，工業 4.0 的技術越來越成熟。物聯網的意識逐漸抬頭，智慧工廠是未來的趨勢。本研究使用樹莓派實作一個多通訊協定的閘道器，透過此閘道器蒐集資料並上傳到伺服器。此閘道器於實作工業上常用的通訊協定，如：現場匯流排技術 Modbus、消息隊列遙測傳輸(MQTT)、OPC UA(OPC Unified Architecture)，讓接上閘道器的傳統設備重獲新生。在本研究中結合 Thingsboard 物聯網平台。透過此平台於樹莓派上建置為一個物聯網專屬閘道器，實作測試與驗證上述三種協定。閘道器成功將所取得的值傳到平台上並以圖形化網頁顯示，快速監控工廠內的設備狀態與數值，以達到智慧工廠的初步雛形。本研究最後搭配 Django 這個網頁伺服器，為使用者規劃閘道器的設定介面以及觀察閘道器上所連接的設備的資料。

關鍵詞：樹莓派、物聯網、Modbus、MQTT、Thingsboard

ABSTRACT

In recent years, the technology of Industry 4.0 has become more and more mature. The awareness of the Internet of Things is gradually rising, and the smart factory is the future trend. This study uses the Raspberry Pi to implement a multiple communication protocol gateway. This gateway can collect data and upload data to the server. This gateway implements communication protocols commonly used in the industry, such as Modbus, Message Queue Telemetry Transmission (MQTT) and OPC UA, then lets traditional equipment be reborn. This study uses the Thingsboard IOT platform. Through this platform, the Raspberry Pi will be built as an IoT dedicated gateway, and the three protocols will be tested and verified. The Gateway successfully passes the obtained value to the platform to display on the graphical web page. Then we can quickly monitor the status and values of the equipment in the factory to achieve the initial prototype of the smart factory. Finally, this study uses the Django web server to change the gateway's setting interface and to view the information of the devices connected to the gateway.

Keywords: Raspberry Pi, IoT, Modbus, MQTT, Thingsboard

一、前言

資策會 MIC 預估，全球智慧工廠市場規模 2015 年可達 179.88 億美元，到 2018 年，全球智慧工廠市場規模可達 235.29 億美元，2015 年之後，全球智慧工廠市場年成長率，可達 7%~11%，優於整體產業自動化年成長率[1]。Modbus 是一個非常古老的協定，支援 TCP、RTU、ASCII 三種模式，其中 RTU 與 ASCII 為傳統工廠最常用的協定。而 MQTT 則是近幾年來非常常用的物聯網資料傳輸協定，通過訂閱標題，就可以取得設備發佈的資料，最重要的是支援資訊加密，這對重視資訊安全物聯網是非

常適合的一個協定之一。智慧工廠最重要的就是能讓傳統的設備透過工業物聯網閘道器，讓資料可以透過網路讓老闆或是工程師達到遠端監控的能力，因此一個工業物聯網閘道器除了支援傳統設備資料的讀取，在透過網路傳遞時也需要把資料加密，避免有心人士竊取工廠資料，所以也需要有支援加密功能的通訊協定如 MQTT 與 OPC UA。

本研究的重點在於實作出一個用於智慧工廠的工業物聯網多通訊協定的閘道器，然後利用物聯網平台取得樹莓派所連接的設備數值，達到監控設備情況的目的[2]。搭配物聯網開放平台 Thingsboard，在樹莓派上建立專屬閘道器，可快速建立圖形化遠

端監控網頁操控系統，省去自行建置雲端伺服器及網頁服務的複雜工作，此平台也能透過特殊設定快速達到支援多通訊協定(Modbus、MQTT、OPC UA)的目的，快速有效建立智慧工廠的初步雛形。

最後利用 Django framework 設計出一個網頁伺服器，達到客製化的要求，提供使用者一個設定的畫面，用來設定所有連接在樹莓派的傳統設備。

二、背景知識

2.1 工業物聯網

在「MQTT 聯網架構實現工業物聯網閘道器」中提到所謂工業 4.0 就是結合工業物聯網(Industrial IoT, IIoT)與大資料分析，在各個不同領域製造現場，透過即時訊息蒐集達成預防故障、降低成本、增加收益與提高產能等目標[3]。工廠將具有智慧化、無人化、資訊化等特性，透過工業物聯網與大資料分析等基礎建設，將外部世界和工廠連結在一起，同時藉由採集、分析各種運作資訊提高整體營運效率。

2.2. 物聯網 (IoT) 閘道器

物聯網 (IoT) 閘道器是一種物理設備或軟體程式，用作雲和控制器，感測器和智慧設備之間的連接點。遷移到雲的所有資料（反之亦然）都通過閘道器，閘道器可以是專用硬體設備或軟體程式。IoT 閘道器也可以稱為智慧閘道器或控制層[4]。

物聯網閘道器的另一個好處是它可以為物聯網網路及其傳輸的資料提供額外的安全性。由於閘道器管理向兩個方向移動的資訊，因此它可以保護移動到雲的資料免受洩漏和物聯網設備的攻擊，這些攻擊具有篡改檢測，加密，硬體隨機數生成器和加密引擎等功能。

2.3 MODBUS

MODBUS 是一種需求-回應訊息的傳遞協定，位於 OSI 模型的第 7 級，採用主從架構實作而成如圖 1 所示。它提供在不同類型的匯流排或網路上連接的設備之間的客戶端/伺服器通信。主從架構的通訊作業會成雙成對的出現，必須有個裝置啟動需求並等候回應，這個啟動需求的裝置（主要裝置）會負責啟動每一次互動。一般而言，主要裝置會是人機介面 (HMI) 或監控與資料擷取 (SCADA) 系統，附屬裝置則是感測器、程式化邏輯控制器 (PLC) 或程式化自動控制器 (PAC)。這些需求和回應內容、傳送訊息所經過的網路層，都是由此協定的不同層級所定義的。透過 Modbus 存取的資料會存放在四

種資料組間或位址範圍[5]：Coil、離散輸入、保存暫存器、輸入暫存器。就像大多數的規格一樣，名稱可能會因為產業或應用而有所不同。ModbusTCP[6]訊息傳遞服務在以太網 TCP/IP 網路上連接的設備之間提供客戶端/伺服器通信。此客戶端/伺服器模型基於四種類型的消息：

- MODBUS 要求
- MODBUS 確認
- MODBUS 指示
- MODBUS 回覆

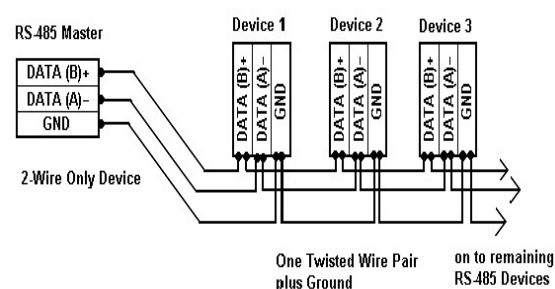


圖 1. Modbus 線路圖

2.4 MQTT

MQTT (Message Queuing Telemetry Transport) 是 ISO 標準(ISO/IEC PRF 20922)下基於發布/訂閱範式的消息協定。工作在 TCP/IP 協定族上，是為硬體性能低下的遠程設備及網路狀況糟糕情況下而設計的發布/訂閱型消息協定，它需要一個訊息代理。

MQTT 系統由與伺服器與通信的客戶端組成，通常稱為“代理”。客戶可以是發布者或訂閱者。每個客戶端都可以連接到代理[7]。

資訊按主題層次結構組織。當發布者具有要分發的新資料項時，它會將包含資料的控制訊息發送到連接的代理。然後，代理將資訊分發給已訂閱該主題的任何客戶端。發布者不需要有關於訂閱者數量或位置的任何資料，而訂閱者又不必配置有關發布者的任何資料。MQTT 以純文字格式發送連接憑證，不包括任何安全性或身份驗證措施。這可以通過底層 TCP 傳輸提供，使用措施保護傳輸信息的完整性不被攔截或複製。此架構示意圖如圖 2。

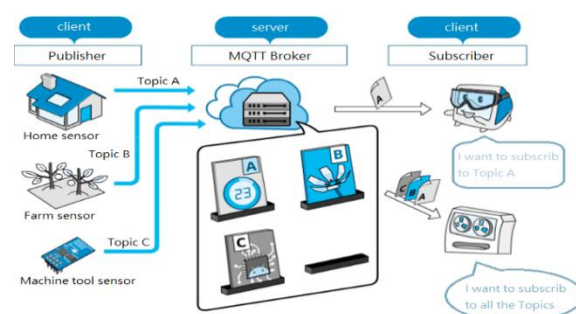


圖 2. MQTT 示意圖

2.5 OPC UA

開發 OPC UA 的目的是發展比原來 OPC 通訊架構（只使用 Microsoft Windows 進程交換 COM/DCOM）更理想的架構。除了 client/server 通訊協定外，新版 OPC UA 也加入了 publish/subscribe 的機制[8]。OPC 統一架構(UA)[9]於 2008 年發布，是一個獨立於平台的面向服務架構，將各個 OPC Classic 規範的所有功能集成到一個可擴展的框架中如圖 3 所示。

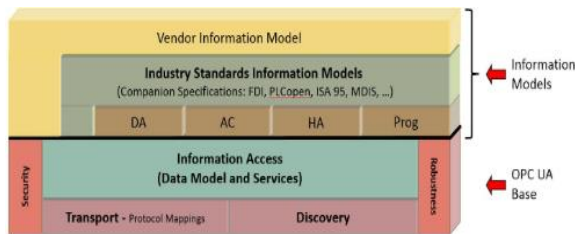


圖 3. OPC UA 架構圖

2.4 Thingsboard 物聯網平台

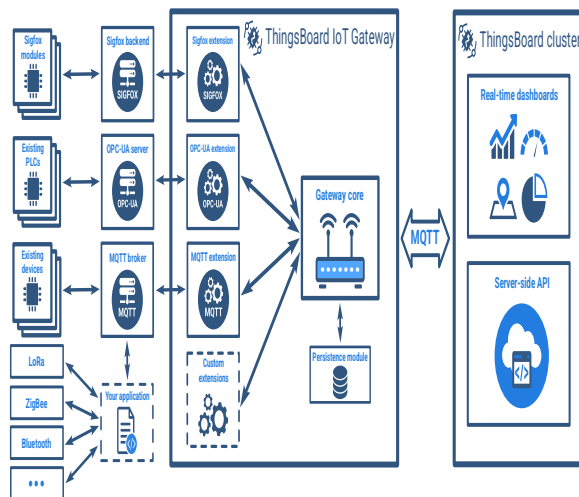


圖 4. Thingsboard 架構圖

ThingsBoard[10]是一個開源的物聯網平台，可以實現物聯網項目的快速開發，管理和擴展，支援常用的工業物聯網通訊協定如圖 4。目標是提供開箱即用的物聯網雲或本地解決方案，為您的物聯網應用提供伺服器端基礎架構。

三、系統架構

此系統主要架構就是在樹莓派上透過 Modbus 協定取得或寫入設備數值，再利用 MQTT 發佈與訂閱的特性，將取得的數值寫入自己架設的網頁或者是物聯網開放平台 Thingsboard，達到遠端監測與控制的目的，本系統規劃支援三種常見的國際標準通訊協定，分別為 Modbus、MQTT、OPC UA

架設由 Python 撰寫而成的網頁伺服器 Django，其中設計 3 個功能如圖 5 所示，分別是新增設備、即時資料、歷史資料，其中所有的資料都會存進資料庫中。

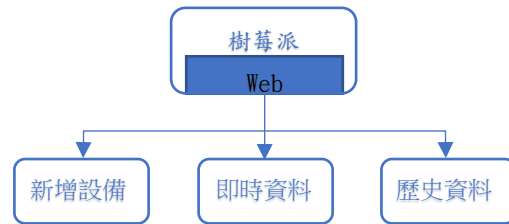


圖 5. 網頁伺服器架構圖

當有新設備接上樹莓派工業物聯網閘道器時，必須先到新增設備的網頁將設備的資訊輸入，如此一來就可以在即時資料的網頁上看到所有的設備列表以及設備所感測到的數值，其流程如圖 6 所示。



圖 6. 新增設備流程

在歷史資料的頁面上，以日期做分隔顯示在網頁上，其流程如圖 7 所示。



圖 7. 取得歷史資料流程

所有數值都將存在資料庫裡如圖 8、圖 9 所示。

| id | Name | device_id | register_no | register_no | register_no | register_no | register_no | mode |
|----|-------|-----------|-------------|-------------|-------------|-------------|-------------|------|
| 1 | 溫度感測器 | 1 | 0 | 1 | | | | RTU |
| 2 | 溫度感測器 | 2 | 3840 | | | | | RTU |

圖 8. 資料庫設備資料

| id | Name | value1 | value2 | value3 | value4 | value5 | time_str | time_dt |
|----|-------------|---------|--------|--------|--------|--------|------------------------------|---------|
| 1 | 21430 溫度感測器 | 24.7 °C | 76.2 % | | | | 20191107 2019/11/07 08:54:43 | |
| 2 | 21431 溫度感測器 | 24.8 °C | | | | | 20191107 2019/11/07 08:54:43 | |
| 3 | 21432 溫度感測器 | 24.7 °C | 76.3 % | | | | 20191107 2019/11/07 08:54:52 | |
| 4 | 21433 溫度感測器 | 24.8 °C | | | | | 20191107 2019/11/07 08:54:52 | |
| 5 | 21434 溫度感測器 | 24.7 °C | 76.4 % | | | | 20191107 2019/11/07 08:55:01 | |
| 6 | 21435 溫度感測器 | 24.8 °C | | | | | 20191107 2019/11/07 08:55:11 | |
| 7 | 21436 溫度感測器 | 24.8 °C | 76.5 % | | | | 20191107 2019/11/07 08:55:11 | |
| 8 | 21437 溫度感測器 | 24.8 °C | | | | | 20191107 2019/11/07 08:55:21 | |
| 9 | 21438 溫度感測器 | 24.8 °C | 76.5 % | | | | 20191107 2019/11/07 08:55:21 | |
| 10 | 21439 溫度感測器 | 24.8 °C | | | | | 20191107 2019/11/07 08:55:21 | |
| 11 | 21440 溫度感測器 | 24.8 °C | 76.6 % | | | | 20191107 2019/11/07 08:55:31 | |
| 12 | 21441 溫度感測器 | 24.8 °C | | | | | 20191107 2019/11/07 08:55:31 | |

圖 9. 資料庫設備數值資料

四、實驗環境與方法

4.1. 硬體模組介紹

樹莓派是基於 Linux 的單晶片電腦，由英國樹莓派基金會開發，目的是以低價硬體及自由軟體促進學校的基本電腦科學教育。操作系統採用開源的 Linux 系統：Debian、ArchLinux，自帶的 Iceweasel、KOffice 等軟體，能夠滿足基本的網路瀏覽、文字處理以及電腦學習的需要。分 A、B 兩種型號如圖 10。



圖 10. 樹莓派 3 model B+ 開發板

本研究使用 USB to RS485 轉換器如圖 11，可使傳統 Modbus 設備得以連接至樹莓派閘道器，連接後執行程式，即可取得感測器資料，此轉換器的特點是，簡單連接且不需要外接電源，即插即用。



圖 11. USB to RS485 轉換器

高精度工業級溫溼度感測器模組如圖 12 採用工業級晶片、高精度溫濕度探頭，確保產品的優異可靠性、高精度、互換性。採用 RS485 硬體接口(具有防雷設計)，協定層兼容標準的工業 Modbus-Rtu 協定。



圖 12. 高精度工業級溫溼度感測器模組

表 1. 暫存器位址與說明

| 暫存器位址 | 說明 |
|--------|-------|
| 0x0000 | 溫度測量值 |
| 0x0001 | 濕度測量值 |

單相交流電智慧電量監控器如圖 13 此設備採用 Modbus RTU 協定進行通訊，目前只支援 0x03(讀保持暫存器)、0x04(讀輸入暫存器)、0x06(寫單個暫存器)、0x41(校準)、0x42(電能清零)等功能碼。



圖 13. 單相交流電智慧電量監控器

表 2. 暫存器位址與說明

| 暫存器位址 | 說明 | 分辨率 |
|--------|-------------|----------------|
| 0x0000 | 電壓測量值 | 1LBS 對應 0.1V |
| 0x0001 | 電流測量值低 16 位 | 1LBS 對應 0.001A |
| 0x0002 | 電流測量值高 16 位 | |
| 0x0003 | 功率測量值低 16 位 | 1LBS 對應 0.1W |
| 0x0004 | 功率測量值高 16 位 | |
| 0x0005 | 電能測量值低 16 位 | 1LBS 對應 1Wh |
| 0x0006 | 電能測量值高 16 位 | |
| 0x0007 | 頻率測量值 | 1LBS 對應 0.1Hz |
| 0x0008 | 功率因數測量值 | 1LBS 對應 0.01 |

4.2. 開源物聯網平台

本研究選擇 Thingsboard 這個物聯網平台，它擁有能快速建立的圖表優點如圖 14，並且支援多平台，不管在 Windows、Linux 或者樹莓派它都有提供服務讓使用者能夠簡單的將電腦或是嵌入式裝置建置成一台閘道器。

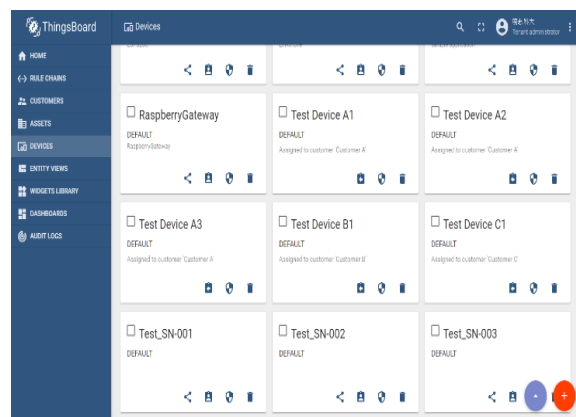


圖 14. 物聯網平台圖形介面

4.3. 網頁伺服器

本研究選擇 Django 這個框架來實作網頁伺服器，使用 Django 可以在幾小時內將 Web 應用程序

從概念帶到啟動。Django 解決了許多 Web 開發的麻煩，因此可以專注於編寫應用程序而無需重新撰寫重複的功能，Django 包含許多其他功能，可用於處理常見的 Web 開發任務，它負責開箱即用的用戶身份驗證，內容管理，站點地圖，RSS feed 和許多其他任務。最重要的是，它是免費和開源的。

為樹莓派安裝 Thingsboard 套件，並在其開放平台將樹莓派掛載，使其得以傳遞資料與建立圖形化，安裝服務之前，必須先確認樹莓派是否有 java 的環境，這次選擇的樹莓派作業系統恰好就有內建 java 開發環境，服務安裝後，必須修改設定檔讓樹莓派與 Thingsboard 確定相互連接，在每次編輯設定檔後都需重啟服務。在樹莓派閘道器設定檔修改完之後，在 Thingsboard 也必須在相對應的地方做修改，讓雙方確認是連通的狀態，圖 15 為 Thingsboard 與閘道器互為連接的狀態。

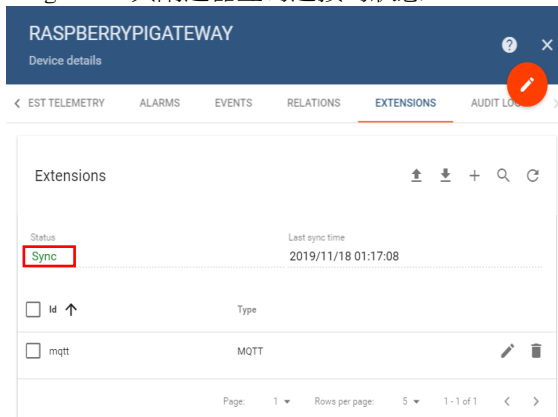


圖 15. Thingsboard 與閘道器連接狀態

本研究最後，由於 Thingsboard 內建的圖表不多，無法客製化，加上 Python 語言的趨勢，決定架設以 Python 為底層的網頁伺服器(Django)，提供使用者圖形介面，在新的設備接上之後，使用者可以直接在網頁上新增設備以及設定樹莓派物聯網閘道器，以便觀察已連接上閘道器的設備資料。

五、實驗結果及分析

5.1. Thingsboard 遠端溫溼度感測器量測結果

如前所述，將樹莓派取得資料透過 MQTT 傳遞到 MQTT Broker，Thingsboard 再到 Broker 取得資料，並建立圖形化曲線圖，圖 16、17 溫溼度感測器量測溫度及濕度結果，並用圖形化介面顯示出來。



圖 16. 溫度測量值



圖 17. 濕度測量值

5.2. Django 網頁伺服器及 Web API 功能

本研究利用 Django 在樹莓派上架設網頁伺服器提供 Web API，由於 Python 語言為未來的趨勢，所以透過 Django 架設一個閘道器的設定畫面，可以設定閘道器，也可以註冊新連接上的設備，更能夠觀察連接到閘道器上的設備資料，此網頁所顯示的資料為即時資料，每 5 秒更新一次。

圖 18 為樹莓派工業物聯網閘道器新增設備之基本畫面，可以指定名稱、Slave_ID、Modbus 模式、暫存器位址，還有根據不同的傳輸協定，輸入各個協定的參數等等。

樹莓派工業物聯網閘道器

新增設備

Name:

Slave id:

Modbus mode:
☐ RTU
☒ TCP

Register no:

Register no1:

Register no2:

Register no3:

Register no4:

Register no5:

IoT protocol:
☒ MQTT
☐ OPC UA

圖 18. 基本新增設備畫面

在圖 19 中，當我在 IoT protocol 選擇 MQTT 時，網頁上自動出現 MQTT 所有相關的欄位，以供使用者設定 MQTT 相關參數，如 MQTT Broker、Port、Topic、Username、Password 等等。

IoT protocol:
☒ MQTT
☐ OPC UA

MQTT

Broker ip:

Mqtt port:

Topic:

Username:

Password:

Telemetry no1:

Telemetry no2:

Telemetry no3:

Telemetry no4:

Telemetry no5:

圖 19. MQTT 參數設定

在圖 20 中，當我在 IoT protocol 選擇 OPC UA 時，網頁上自動出現 OPC UA 所有相關的欄位，以供使用者設定 OPC UA 相關參數，如 IP、Port、Parameter 等等。

OPC UA

Opcua ip:

Opcua port:

Parameter no1:

Parameter no2:

Parameter no3:

Parameter no4:

Parameter no5:

圖 20. OPC UA 參數

在圖 21 中，在網頁有兩個表格，一個是設備資料，一個是設備清單，在設備資料上可以看到所有連接至樹莓派閘道器的設備數值，在設備清單上可以看到所有設備的設定資料。

設備資料

| Name | value1 | value2 | value3 | value4 | value5 | History |
|---------|--------|--------|--------|--------|--------|----------------------|
| 溫溼度感測器 | 26.7 | 55.3 | | | | Link |
| 溫度控制器 | 28.1 | | | | | Link |
| 網路測試設備 | 12437 | | | | | Link |
| 網路測試設備2 | 1620 | 1617 | | | | Link |

設備清單

| slave_id | Name | mode | IoT_protocol | register_no | register_no1 | register_no2 | register_no3 | register_no4 | register_no5 |
|----------|---------|------|--------------|-------------|--------------|--------------|--------------|--------------|--------------|
| 1 | 溫溼度感測器 | RTU | MQTT | 2 | 0 | 1 | | | |
| 2 | 溫度控制器 | RTU | MQTT | 1 | 3840 | | | | |
| 3 | 網路測試設備 | TCP | MQTT | 1 | 0 | | | | |
| 4 | 網路測試設備2 | TCP | MQTT | 2 | 0 | 1 | | | |

新增設備 刪除

圖 21. 即時資料

除了即時資料外，本研究將所有資料存在資料庫中，以日期分隔，使用者可以選擇觀察哪一天的資料，以研究實驗使用的溫溼度感測器為例，透過歷史資料可以了解家中的溫度及濕度的變化，辨別家裡的冷氣是否溫度開得過低導致電費增加。圖 22、23 為兩個感測器的歷史資料。

Sensor_History

| Name | time | Value1 | Value2 | Value3 | Value4 | Value5 |
|-------|---------------------|---------|--------|--------|--------|--------|
| 溫度控制器 | 2019/10/29 20:54:10 | 24.2 °C | | | | |
| 溫度控制器 | 2019/10/29 20:54:36 | 24.2 °C | | | | |
| 溫度控制器 | 2019/10/29 21:54:40 | 24.6 °C | | | | |
| 溫度控制器 | 2019/10/29 22:25:40 | 24.2 °C | | | | |
| 溫度控制器 | 2019/10/29 23:25:44 | 24.7 °C | | | | |

回首頁

- 20191029
- 20191030
- 20191031

圖 22. 溫度控制器歷史資料

Sensor_History

| Name | time | Value1 | Value2 | Value3 | Value4 | Value5 |
|--------|---------------------|---------|--------|--------|--------|--------|
| 溫溼度感測器 | 2019/10/29 20:54:10 | 25.1 °C | 67.4 % | | | |
| 溫溼度感測器 | 2019/10/29 20:54:36 | 25.1 °C | 67.3 % | | | |
| 溫溼度感測器 | 2019/10/29 21:54:40 | 25.4 °C | 74.5 % | | | |
| 溫溼度感測器 | 2019/10/29 22:25:40 | 25.0 °C | 74.0 % | | | |
| 溫溼度感測器 | 2019/10/29 23:25:44 | 25.5 °C | 74.1 % | | | |

回首頁

- 20191029
- 20191030
- 20191031

圖 23. 溫溼度感測器歷史資料

將上述所有的硬體及軟體整合如圖 24 所示，由左到右分別是第二台樹莓派(訂閱端)、第一台樹莓派(物聯網閘道器)、溫度控制器、溫溼度感測器以及電源供應器。



圖 24. 系統整合設備圖

同時，本研究使用網路攝影機，達到遠端監控的目的，影像如圖 25 所示。

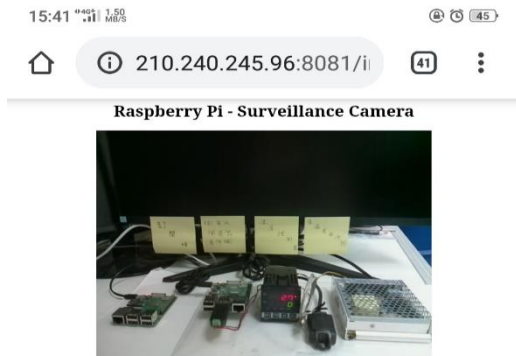


圖 25. 網路攝影機影像

六、結論與未來展望

本研究使用 Thingsboard 平台在樹莓派上建立專屬閘道器，可以快速建立圖形化遠端監控網頁操控系統，省去自行建置雲端伺服器及網頁服務的複雜工作，但平台內建的圖表太少以及設定方式複雜，因此在本研究的最後自行架設一個網頁伺服器，提供使用者更客製化的網頁介面。

本研究實作測試與驗證三種協定，成功將所取得的值傳到平台上並以圖形化網頁顯示，快速監控工廠內的設備狀態與數值，以達到智慧工廠的初步雛形。未來的設備都會趨向無線，使用藍牙或無線網路的方式傳遞資料。對多種設備進行讀值並利用物聯網平台做圖表的監控，近年來使用樹莓派做各種應用已成為各大企業與教育的趨勢，由於便宜與支援多種介面，如低功耗藍牙以及無線網路，加上 Python 強大的海量套件庫，有助於本研究的實作，這也是選擇樹莓派實作閘道器的原因之一。

七、參考文獻

1. 廖專崇, 從自動化邁向智動化 智慧工控發展全速前進, 新通訊, 11 月號 177 期, 2015。取自: http://www.2cm.com.tw/coverstory_content.asp?sn=1510230005

2. Robinson, A., Cook, M.: Raspberry Pi Projects. Wiley, Chichester (2014)
3. 陳響亮, 李宜靜, 陳文泉, 李桂銘, “MQTT 聯網架構實現工業物聯網閘道器”, TANET2018 臺灣國際網路研討會, 2018 / 11 / 01, P2167 - 2172
4. Brien Posey, Tréa Lavery, IoT Gateway, TechTarget (NASDAQ: TTGT)
5. 深入了解 Modbus 協定-- Engineer Ambitiously-NI, in <https://www.ni.com/zh-tw/innovations/white-papers/14/the-modbus-protocol-in-depth.html>
6. Texas Modbus-IDA, Modbus Messaging On Tcp/Ip Implementation Guide V1.0b, 2006
7. MQTT, in <https://zh.wikipedia.org/wiki/MQTT>
8. OPC UA, in https://zh.wikipedia.org/wiki/OPC_UA
9. OPC UA, in <https://opcfoundation.org/about/opc-technologies/opc-ua/>
10. ThingsBoard Documentation, in <https://thingsboard.io/docs/getting-started-guides/what-is-thingsboard/>

