

使用 ICDT MQTT xHMI 控制 Arduino ESP32 三色燈

元米科技 2021/8/19

關於 ICDT MQTT xHMI 方案

ICDT MQTT xHMI 是兼具 MQTT Server、WEB Server 的雲端圖控方案，可快速將 IOT 設備連接至雲端，並且依照自己編輯的顯示介面以瀏覽器顯示操控。其他的特點

- 簡易線上編輯除錯的 DDC 可程式控制語法，可使簡易的定時、算數、條件運算由雲端執行，不需要改變終端設備的韌體(每秒或每十秒運算一次，可由 DDC 程式設定)
- 定時數值記錄，可採報表或曲線方式呈現，每點約 10,000 筆每分鐘記錄一次可儲存達一周的資料最快一分鐘，時間可由 DDC 程式設定)
- 具備警報監視功能，除了於警報發生時除了瀏覽網頁會跳出相關訊息外，也可依照設定連動 Line 或 Telegram 帳戶發出提示訊息

系統架構

MQTT xHMI Server
(MQTT Broker & HTTP Server)
<http://projectname.icdt.com.tw:8888>

xHMI 專案編輯

HTTP 上傳專案



MQTT
訂閱

HTTP



瀏覽器操控



申請 ICDT MQTT xHMI 客戶端帳號

依照 FB 粉絲專頁說明申請

<https://fb.watch/7s8wmoUuBz/> (基本上於貼文下回復+1, 並發送訊息提供希望的專案名稱等資料即可)

完成申請後會提供一個帳號密碼, 以下以 aiqwebhmi 為帳號範例, 12345678 為密碼範例

測試帳號與修改密碼

瀏覽器輸入網址

<http://aiqwebhmi.icdt.com.tw:8888/project?cmd=states> (最前方為帳號名稱), 此時需輸入提供的帳號密碼 (例如aiqwebhmi/12345678), 完成後畫面會出現本帳號的各項連線數據, 此表示帳戶密碼無誤

瀏覽器輸入網址 <http://aiqwebhmi.icdt.com.tw:8888/account> 輸入目前密碼與兩次新密碼, 以修改帳號密碼。

注意 aiqwebhmi 只是帳號範例

← → ↻ 🏠 ⚠️ 不安全 | aiqwebhmi.icdt.com.tw:8888/project?

```
Project:aiqwebhmi
Log count:0
Upload count:0
Web count:3
Alarm count:0
Notify count:0
Login Fail count:0
MQTT Login count:0
MQTT Login Fail count:0
MQTT Pulish count:0
MQTT Subscript count:0
MQTT Client Message count:0
```

← → ↻ 🏠 ⚠️ 不安全 | aiqwebhmi.icdt.com.tw:8888/accou... ☆ ★ 👤 ⋮

Change password for [aiqwebhmi]

Current Password

New Password

Confirm Password

Confirm

Access token :

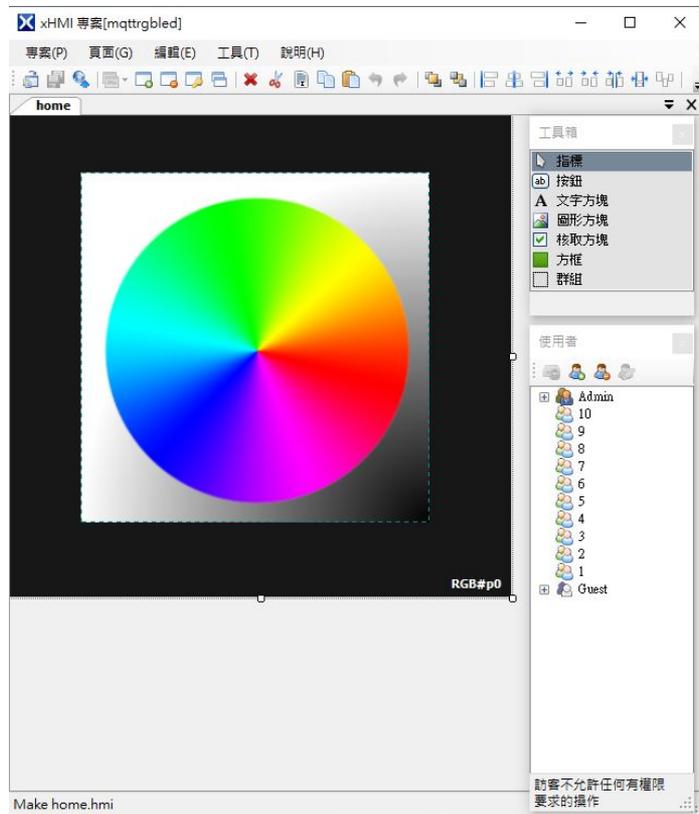
A2nI6F1CwF4Qexryjo8bdhJJEwaFNnEvLHR0yZMIxa45JmwFKU3RwxCD

安裝 xHMI 圖控專案編輯程式

xHMI 圖控專案編輯程式是在 Windows 10 作業系統環境執行，因此必須準備 Windows 10 電腦，於以下路徑下在安裝 <https://bit.ly/3iXxFdA>

完成安裝後於以下路徑下載專案範例檔案
http://icdt.com.tw/menu/matt_rgb_led.zip

解壓縮後開啟 xHMI 程式，選擇專案->開啟專案，點選 `mqtrrgbled.phm`，彈出的使用者直接選擇確定即可(先使用預設的 `root/root`)帳號



使用者設定

使用者必須有至少一個管理員(Admin)，預設的帳號密碼為root/root，可以依照需求修正

1~10 使用者層級可以依照需求設定，於後續按鈕或數值操作希望限制使用者權限時使用

訪客 Guest 如包含 All 則除非有限定權的操作外限外，否則所有人只要知道專案帳號皆可以開啟畫面，如果專案不希望被瀏覽則建議刪除該All)帳號

注意，此處的使用者針對專案操作設定，與申請的客戶端帳號密碼無關



設定標籤(Tags)

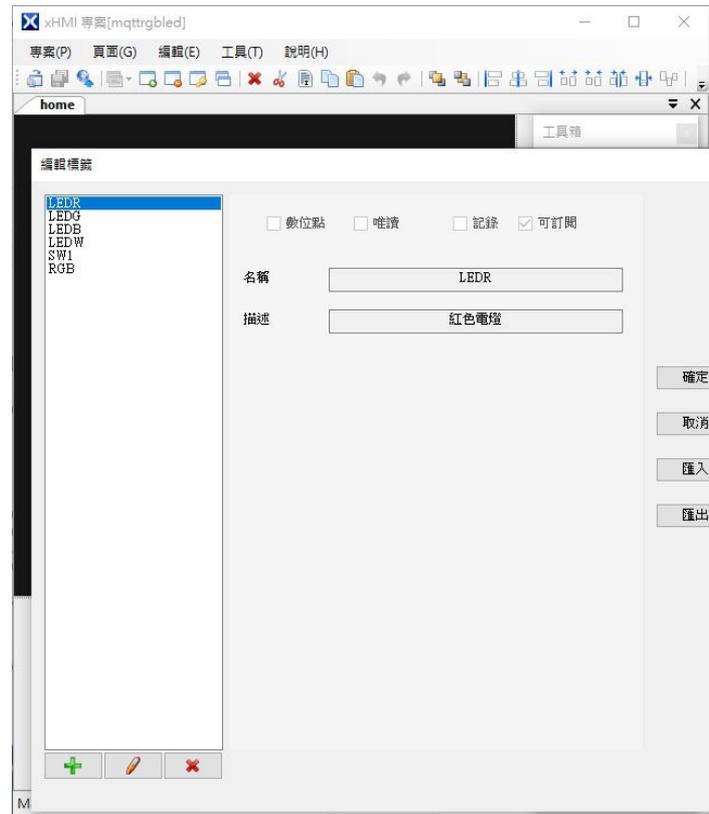
點選工具>物件標籤可檢視與設定目前的標籤。

編輯名稱即 MQTT 發佈與訂閱 Topic 的關鍵字, 例如專案名為 aiqwebhmi, 標籤名為 LEDW, 則該點的 Topic 即為 aiqwebhmi/LEDW

標籤名為英文或數字, 區分大小寫, 不得使用 DDC 函數或變數保留字

勾選數位點將數值僅接受 0 與 1; 勾選唯讀則該點預設不可以由圖控改變狀態; 勾選記錄會定時記錄數值; 勾選可訂閱會在改變或收到發佈後, 發送數值給訂閱者

大量修改可以匯出為 CSV 檔案, 由 Excel 編輯後再做匯入

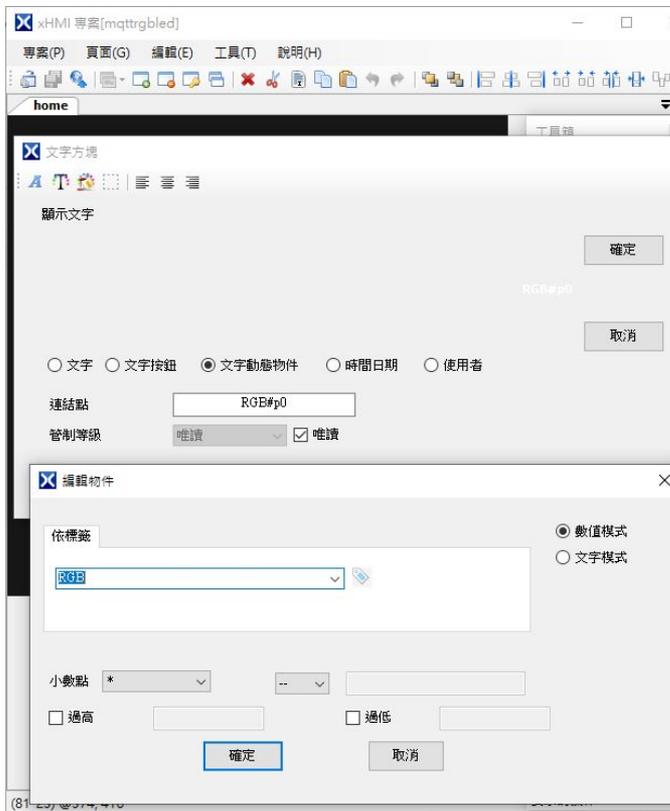


編輯頁面

滑鼠雙擊畫面上的元件可開啟並進行設定，個元件的應用請參考選單說明相關單元

此案例只使用一個文字方塊顯示目前RGB 的數值，並使用一個自定義方框，用以擷取點選處的顏色，自定義方框的程式碼可以參考

mqtt_rgb_led\mqttrgbled_customized\colorpicker.html 與 圖控介面 標籤“RGB” 互動的是 colorpicker.html 的 Update() 函數，點選後將顏色以 RGB565 的數值格式回傳給 標籤“RGB”。點選的位置在色環右側或下側(X >278 或 Y >278) 會改變色環的透明度



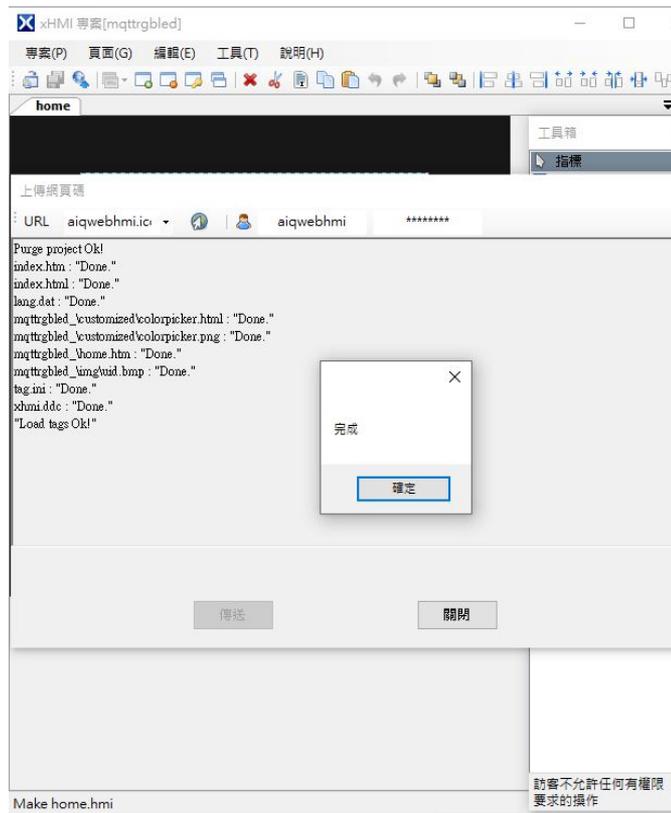
```
19 function Update ()
20 {
21   ..if(--loadcnt >=0.)
22     ..loadimage() ;
23   ..if(getInput)
24     ..{
25     ..getInput=.false;
26     ..return rgb565;
27   }..
28 }
29 function toRGB565(orig)
30 {
31   ..var rgb = orig.replace(/
32   ..r = (rgb[1] & 0xf8) >> 3
33   ..g = (rgb[2] & 0xfc) >> 2
34   ..b = (rgb[3] & 0xf8) >> 3
35   ..return (r << 11) + (g <<
36 }
37
38 function getCursorPosition
39 ..const rect = canvas.getB
40 ..const x = event.clientX
41 ..const y = event.clientY
42 ..if(x>278)
43 ..{
44   ..//context.fillStyle="b
45   ..context.fillRect(0,0,3
46   ..context.globalAlpha =
47   ..context.drawImage(img,
48   ..console.log(context.glo
49 ..}
50 ..else if(y>278)
51 ..{
52   ..//context.fillStyle="b
53   ..context.fillRect(0,0,3
54   ..context.globalAlpha =
55   ..context.drawImage(img,
56   ..console.log(context.glo
57 ..}
58 ..else
59 ..{
60   ..console.log(x+" ", "+y)
61   ..var colorData = .getCanv
62   ..var color = colorData.
63
```

上傳專案

完成編輯後選擇專案>上傳網頁碼。URL 處輸入帳號網址 `http://aiqwebhmi.icdt.com.tw:8888`，右側使用者帳號密碼輸入 `aiqwebhmi` 與更新後的密碼，按下傳送已進行上傳。(以帳號 `aiqwebhmi` 為例)

上傳完成後可開啟瀏覽器輸入帳號網址即可瀏覽成果，瀏覽器除了電腦外，手機平板瀏覽器皆可正常瀏覽)

注意 aiqwebhmi 只是帳號範例



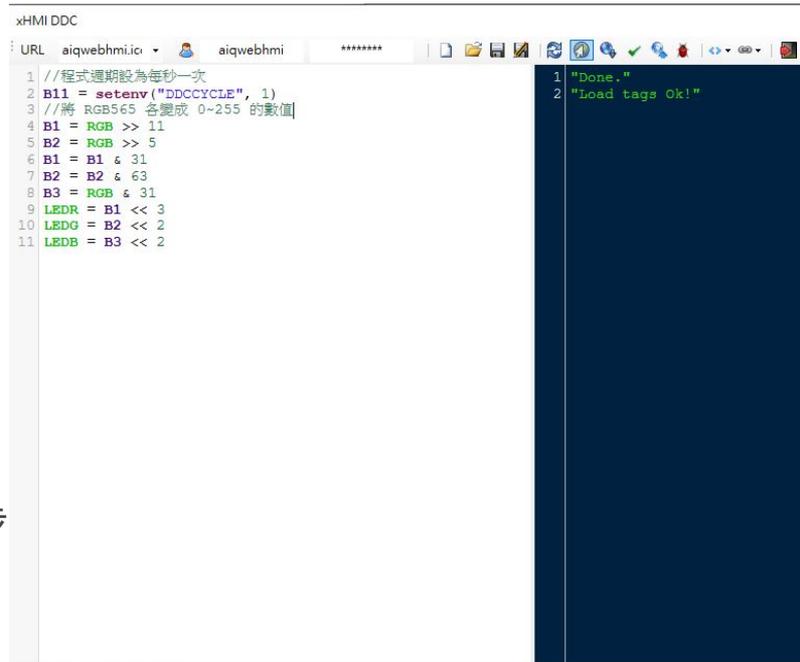
修改 DDC 程式

點選工具 -> DDC 可以開啟 DDC 程式編輯器，於 URL 處輸入帳號網址 <http://aiqwebhmi.icdt.com.tw:8888>，右側使用者帳號密碼輸入 aiqwebhmi 與更新後的密碼，按下傳送已進行上傳。(以帳號 aiqwebhmi 為例)

按下 Reflash 按鈕會顯示目前專案內儲存的 DDC 程式；按下 Download 可以至雲端帳戶下載目前的 DDC 程式；按下 Check 會將目前編輯器內容上傳雲端做語法檢查；語法無錯誤時 Send 鈕才會生效，按下 Send 可將目前程式上傳至雲端，並重新啟動程式

DDC 程式上傳同時會儲存於電腦專案路徑中，於下次上傳專案時同步上傳

由於 DDC 程式的標籤變數是取自專案的物件標籤名稱，因此如果物件標籤名稱變更必須先上傳專案才會生效



The screenshot shows a web browser window titled 'xHMI DDC' with the URL 'aiqwebhmi.icdt.com.tw:8888'. The main content area displays a DDC program code with line numbers 1 through 11. The code includes comments and assignments for variables B1, B2, B3, LEDR, LEDG, and LEDB. A right-hand panel shows the execution output with two lines: '1 Done.' and '2 Load tags Ok!'.

```
1 //程式週期設為每秒一次
2 B11 = setenv("DDCCYCLE", 1)
3 //將 RGB565 各變成 0~255 的數值
4 B1 = RGB >> 11
5 B2 = RGB >> 5
6 B1 = B1 & 31
7 B2 = B2 & 63
8 B3 = RGB & 31
9 LEDR = B1 << 3
10 LEDG = B2 << 2
11 LEDB = B3 << 2
```

```
1 Done.
2 Load tags Ok!
```

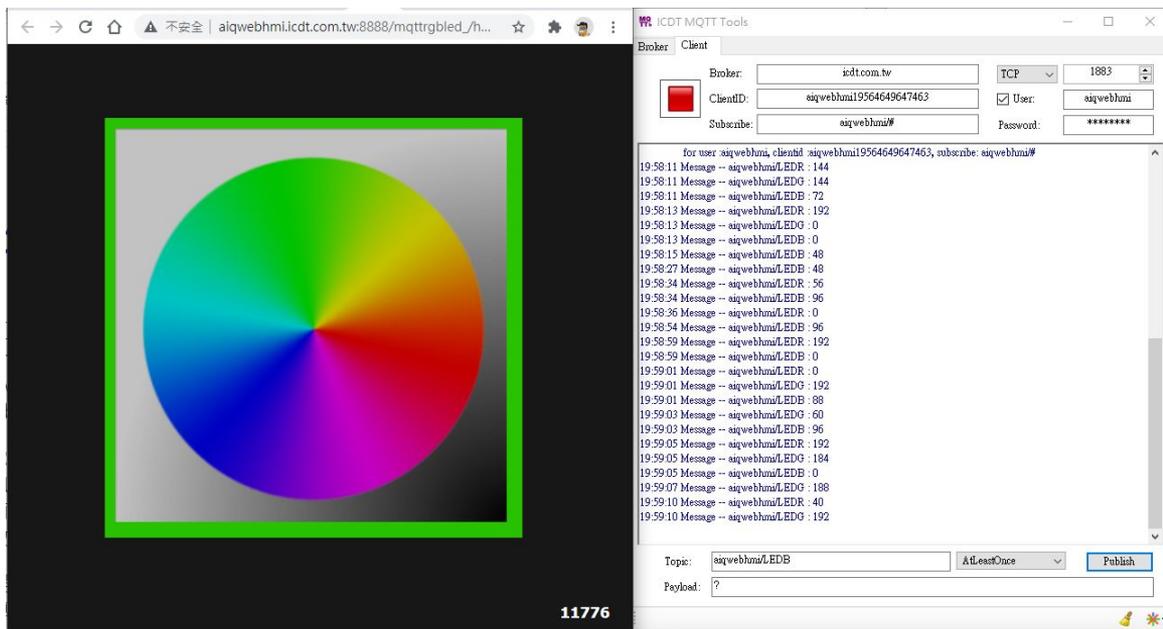
注意 aiqwebhmi 只是帳號範例

驗證 MQTT 發布-使用 ICDT MQTT Tools

此處以 ICDT MQTT Tools 進行 MQTT Client 發布驗證, 下載處

<https://bit.ly/2SqRe5X>

於 Broker 處輸入 icdt.com.tw TCP 1883, ClientID 以帳號名稱開始例如 aiqwebhmixx, Subscribe 為 aiqwebhmi/# 或 aiqwebhmi/+, User 輸入 aiqwebhmi 與新密碼。Topic 處輸入 aiqwebhmi/LEDB (以標籤 LEDB 為例) Payload 處輸入 ?, 按下發布即可看到目前 LEDB 的數值, 將 Payload 處輸入 0~255 的數值則因為被 DDC 程式蓋過因此無法發布



注意 aiqwebhmi 只是帳號範例

編輯 ESP32-C3S 開發板

驗證使用的是購自尤教授的露天賣場的 ESP32-C3S 單核心低價板，板上附一個 RGB LED 燈與白色 LED 燈，以 Arduino 編輯器依照說明文件部落格完成設定，並開啟下載之 mqtt_rgb_led.ino 完成編譯上傳後，三色 LED 可依照操作顯示顏色

賣場連結：

<https://www.ruten.com.tw/item/show?22119396842939>

說明文件部落格：

<https://youyouyou.pixnet.net/blog/post/121068678-esp32-c3f-%e5%96%ae%e6%a0%b8%e4%bd%8e%e5%83%b9%e6%94%af%e6%8f%b4%e6%a6%82%e8%a7%80>

```
mqtt_rgb_led

/*
ESP32-C3S 賣場連結：https://www.ruten.com.tw/item/show?22119396842939
設定說明：https://youyouyou.pixnet.net/blog/post/121068678-esp32-c3f-%e5%96%ae%e6%a0%b8%e4%bd%8e%e5%83%b9%e6%94%af%e6%8f%b4%e6%a6%82%e8%a7%80
開發板:"ESP32C3 Dev Module"
Flash Size: "2MB (16Mb)"
Partition Scheme: "Default 4MB with spiffs (1.2MB APP/1.5MB SPIFFS)"
打開串列埠視窗會造成無法動作，須改為 ESP32 Dev Module 才可開啟，但又改程式時必須記住上述參數要調回來
*/
#include <stdio.h>
#include <PubSubClient.h>
#include <WiFi.h>

const char* ssid = "mywifissid";
const char* pass = "mywifipassword";
const char* mqttServer = "icdt.com.tw";
const int mqttPort = 1883;

const char* mqttUser = "aiqwebhmi";
const char* mqttPassword = "12345678";
//MqttSubscribe 開頭必須與 mqttUser 相同
const char* MqttSubscribe = "aiqwebhmi/#";
//clientId 開頭必須與 mqttUser 相同
String clientId = String(mqttUser);
bool Wifi_Ready = false;

WiFiClient espClient;
PubSubClient client(espClient);

// setting PWM properties
const int freq = 5000;
const int ledRChannel = 0;
const int ledGChannel = 1;
const int ledBChannel = 2;
const int ledWChannel = 3;
<

儲存完畢
Writing at 0x00008000... (100 %)
Wrote 3072 bytes (118 compressed) at 0x00008000 in 0.1 seconds (effective 256.5 kbit/s)...
```



補充說明



此範例僅提供簡易的網頁操作取色控制 RGB LED 燈，搭配 DDC 程式與其他感測元件，可以達成定時控制、亮度控制等進階功能。

ESP32 適性價比極高，搭配 Arduino 開發 MQTT 物聯網應用極為便利，配合 ICDT MQTT xHMI Server 方案，更容易完成 Broker、操控介面、資料記錄與雲端程式操控。元米科技除了提供該方案的免費體驗，更提供教育工作者免費教育板伺服器授權，以利物聯網教學推廣，歡迎多加利用