

Mbed Lab 10 Report

WiFi and MQTT

109033130 唐振家

一、Lab Description

1、Connect B-L4S5I-IOT01A to WiFi

說明：

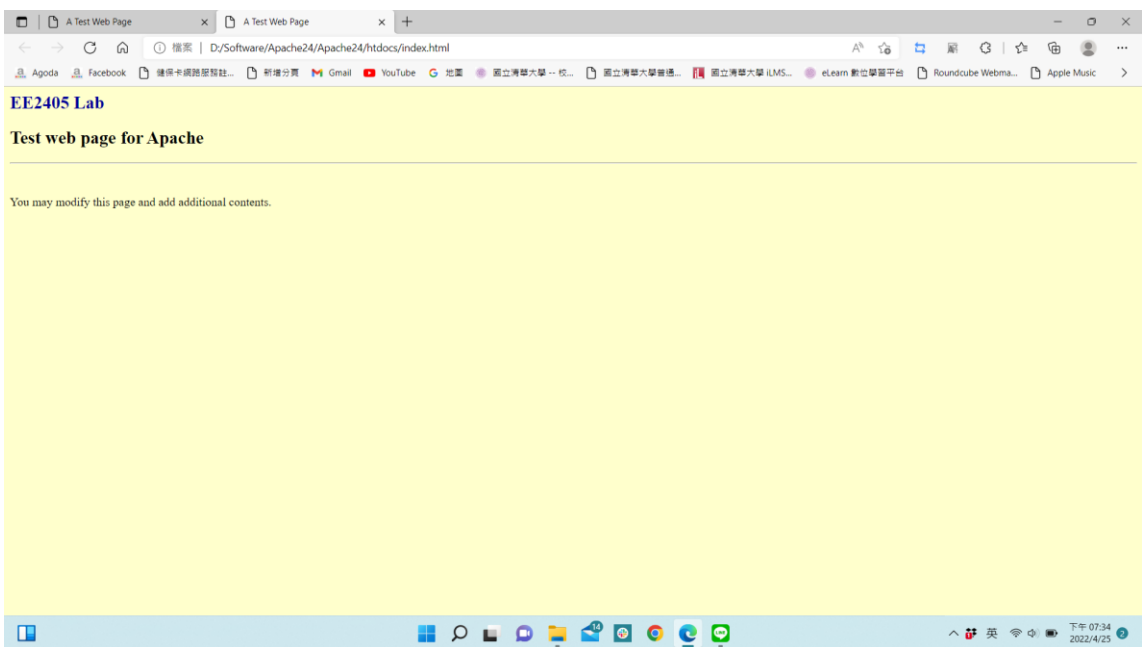
此部分旨在利用mbed上的接口，將mbed以及電腦連在同一個區域網路之下，並確認是否有成功連接上。

先將電腦連上收機私人網路，利用終端輸入ipconfig找出IP位置，或是直接利用電腦介面查詢，為 [172.20.10.2]。之後進行apache2的下載在以及測試連線是否可行。

```
Wireless LAN adapter Wi-Fi:
Connection-specific DNS Suffix . : 
Link-local IPv6 Address . . . . . : fe80::690f:4272:6b25:1f8c%9
IPv4 Address. . . . . : 172.20.10.2
Subnet Mask . . . . . : 255.255.255.240
Default Gateway . . . . . : 172.20.10.1
```

說明：

此為將Apache server下載下來之後，設定好網路端口為80(Input、Output需相同)，最後將網頁裡index.html的code複製進去檔案中，執行./httpd.exe檔案，再輸入本地網路的IP位址，即可確認是否輸入正確的網路地址。



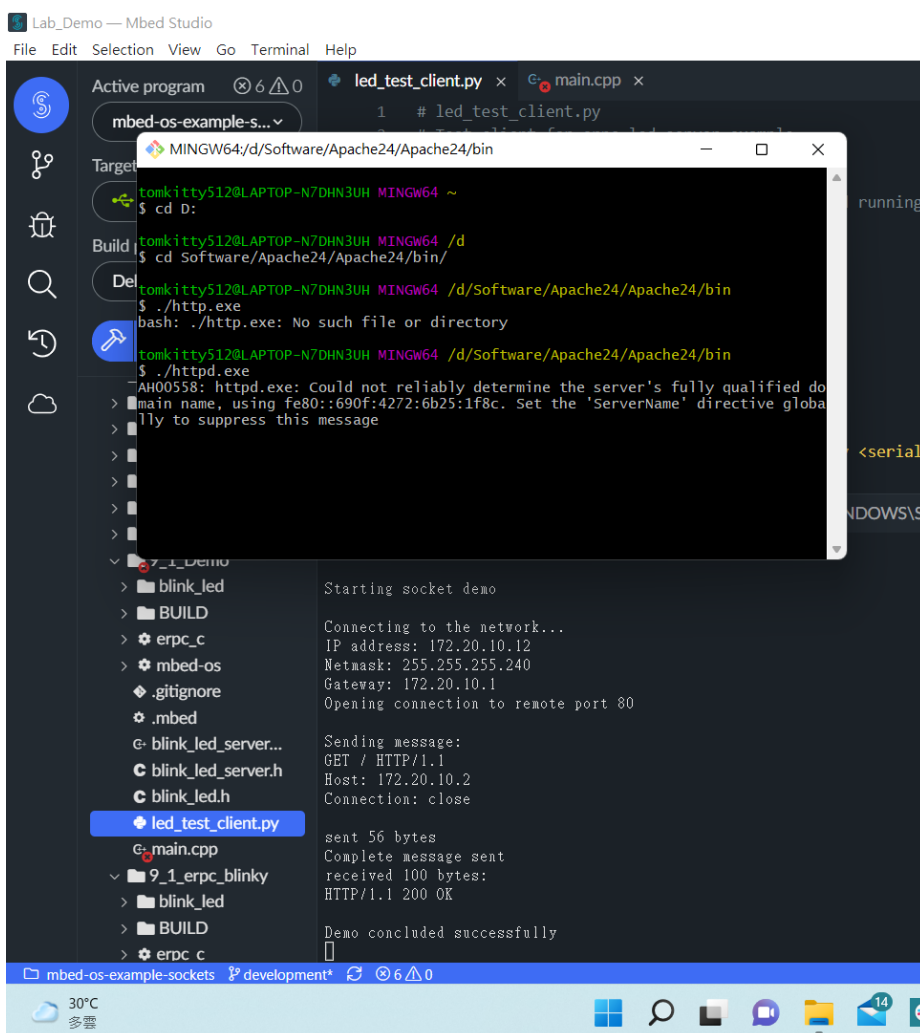
一、Lab Description

1、Connect B-L4S5I-IOT01A to WiFi

說明：

接下來將課程網頁上的URL引入Program中，將所有的HOST_IP部份改成剛剛的本地網路的IP，且在mbed_app.json，將SSID以及PASSWORD改成私人網路的名稱及密碼，如此一來mbed才能夠連上我們自己的私人網路。

利用Gitbash開啟Apache server，執行mbed裡的program，如果有成功，即會顯示以下資訊，我們可以看見，本地的HOST為剛剛設定的172.20.10.2，而port為80



The screenshot shows the Mbed Studio IDE interface. The main window displays a terminal window titled "MINGW64/d/Software/Apache24/Apache24/bin". The terminal output shows the following commands and results:

```
tomkitty512@LAPTOP-N7DHN3UH MINGW64 ~  
$ cd D:  
$ cd Software/Apache24/Apache24/bin/  
$ ./http.exe  
bash: ./http.exe: No such file or directory  
$ ./httpd.exe  
-AH00558: httpd.exe: Could not reliably determine the server's fully qualified domain name, using fe80::690f:4272:6b25:1f8c. Set the 'ServerName' directive globally to suppress this message  
Starting socket demo  
Connecting to the network...  
IP address: 172.20.10.12  
Netmask: 255.255.255.240  
Gateway: 172.20.10.1  
Opening connection to remote port 80  
Sending message:  
GET / HTTP/1.1  
Host: 172.20.10.2  
Connection: close  
sent 56 bytes  
Complete message sent  
received 100 bytes:  
HTTP/1.1 200 OK  
Demo concluded successfully
```

一、Lab Description

1、MQTT Broker and Client

說明：

此部分旨在，利用剛剛連接好的Wifi網路，將我們想要丟出去的message，利用MQTT的方式丟出。

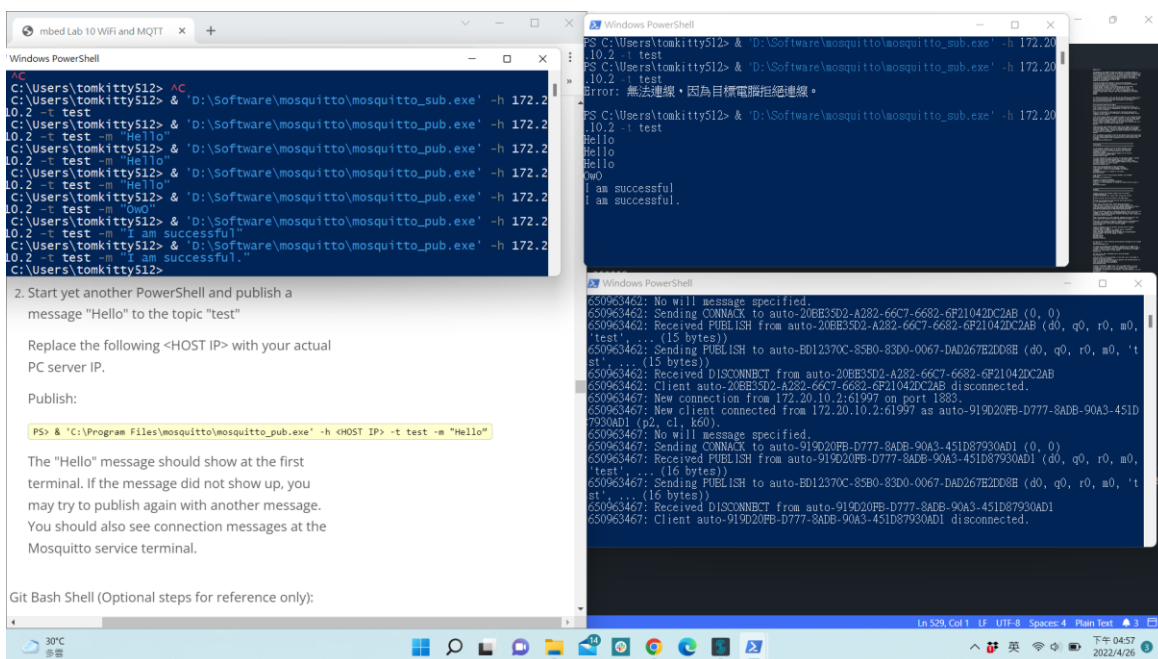
一樣需先下載MQTT broker service，然後設置1883 network port專門每MQTT的通訊使用。

接著利用PowerShell，進到Program所在的資料夾，執行Mosquitto service。

```
PS D:\> cd D:
PS D:\> cd .\Embedded_system\Lab_Demo\
PS D:\Embedded_system\Lab_Demo> D:\Software\mosquitto\mosquitto.exe -c mosquitto.conf -v
650889375: mosquitto version 2.0.14 starting
650889375: Config loaded from mosquitto.conf.
650889375: Starting in local only mode. Connections will only be possible from clients running on this machine.
650889375: Create a configuration file which defines a listener to allow remote access.
650889375: For more details see https://mosquitto.org/documentation/authentication-methods/
650889375: Opening ipv4 listen socket on port 1883.
650889375: Opening ipv6 listen socket on port 1883.
650889375: mosquitto version 2.0.14 running
```

說明：

我們可以先利用一些指令測試Mosquitto service是否可行，透過開啟其預設的publish以及subscribe，如果可以將訊息文字傳遞成功，即表示Mosquitto service是可以使用的。



一、Lab Description

1、MQTT Broker and Client

說明：

將Python的MQTT libraries下載下來並創建新的program，引入Wifi、mqtt的library (因為我們要將其二做結合)，同樣的將SSID以及PASSWORD改成私人網路的名稱及密碼，IP也訂成自己的HOST IP，成功即顯示我的網路名稱以及本地位址以及network port等資訊，並且測試五組數據是否穩定。

```
Connecting to jiajia...
Connecting to TCP network...
address is 172.20.10.2/1883
Successfully connected!
Message arrived: QoS0, retained 0, dup 0, packetID 816
Payload Message from Python!

Message arrived: QoS0, retained 0, dup 0, packetID 816
Payload Message from Python!

Message arrived: QoS0, retained 0, dup 0, packetID 816
Payload Message from Python!

Message arrived: QoS0, retained 0, dup 0, packetID 816
Payload Message from Python!

Message arrived: QoS0, retained 0, dup 0, packetID 816
Payload Message from Python!
```

說明：

將Python MQTT Client設定完成，執行後，會出現以下結果，將收到連接成功的提示以及穩定接收五組資料。

```
d:\Embedded_system\Lab_Demo\10_2_MQTT>python3 mqtt_client.py
Connecting to 172.20.10.2/Mbed
Connected rc: 0
Subscribed OK
[Received] Topic: Mbed, Message: b'Message from Python!\n'

[Received] Topic: Mbed, Message: b'Message from Python!\n'

[Received] Topic: Mbed, Message: b'Message from Python!\n'

[Received] Topic: Mbed, Message: b'Message from Python!\n'

[Received] Topic: Mbed, Message: b'Message from Python!\n'

650977571: Mbed (QoS 0)
650977571: auto-B5C8BA12-193F-550F-5708-5920D415B981 0 Mbed
650977571: Sending SUBACK to auto-B5C8BA12-193F-550F-5708-5920D415B981
650977571: Received PUBLISH from auto-B5C8BA12-193F-550F-5708-5920D415B981 (d0, q0, r0, m0, 'Mbed', ... (21 bytes))
650977571: Sending PUBLISH to Mbed (d0, q0, r0, m0, 'Mbed', ... (21 bytes))
650977571: Sending PUBLISH to auto-B5C8BA12-193F-550F-5708-5920D415B981 (d0, q0, r0, m0, 'Mbed', ... (21 bytes))
650977572: Received PUBLISH from auto-B5C8BA12-193F-550F-5708-5920D415B981 (d0, q0, r0, m0, 'Mbed', ... (21 bytes))
650977572: Sending PUBLISH to Mbed (d0, q0, r0, m0, 'Mbed', ... (21 bytes))
650977572: Sending PUBLISH to auto-B5C8BA12-193F-550F-5708-5920D415B981 (d0, q0, r0, m0, 'Mbed', ... (21 bytes))
650977574: Received PUBLISH from auto-B5C8BA12-193F-550F-5708-5920D415B981 (d0, q0, r0, m0, 'Mbed', ... (21 bytes))
650977574: Sending PUBLISH to Mbed (d0, q0, r0, m0, 'Mbed', ... (21 bytes))
650977574: Sending PUBLISH to auto-B5C8BA12-193F-550F-5708-5920D415B981 (d0, q0, r0, m0, 'Mbed', ... (21 bytes))
650977575: Received PUBLISH from auto-B5C8BA12-193F-550F-5708-5920D415B981 (d0, q0, r0, m0, 'Mbed', ... (21 bytes))
650977575: Sending PUBLISH to Mbed (d0, q0, r0, m0, 'Mbed', ... (21 bytes))
650977575: Sending PUBLISH to auto-B5C8BA12-193F-550F-5708-5920D415B981 (d0, q0, r0, m0, 'Mbed', ... (21 bytes))
650977577: Received PUBLISH from auto-B5C8BA12-193F-550F-5708-5920D415B981 (d0, q0, r0, m0, 'Mbed', ... (21 bytes))
650977577: Sending PUBLISH to Mbed (d0, q0, r0, m0, 'Mbed', ... (21 bytes))
650977577: Sending PUBLISH to auto-B5C8BA12-193F-550F-5708-5920D415B981 (d0, q0, r0, m0, 'Mbed', ... (21 bytes))
```

一、Lab Description

1、MQTT Broker and Client

說明：

按下按鍵時，mbed將message_num加一後pub給client(python端)，client subscribe下來以後，印出結果。

```
Message arrived: QoS0, retained 0, dup 0, packetID 816
Payload Message from Python!

Message arrived: QoS0, retained 0, dup 0, packetID 816
Payload Message from Python!

rc: 0
Publish message: QoS0 Hello, Python! #1
Message arrived: QoS0, retained 0, dup 0, packetID 816
Payload QoS0 Hello, Python! #1
rc: 0
Publish message: QoS0 Hello, Python! #2
Message arrived: QoS0, retained 0, dup 0, packetID 816
rc: 0
Publish message: QoS0 Hello, Python! #3
Payload QoS0 Hello, Python! #2
Message arrived: QoS0, retained 0, dup 0, packetID 816
rc: 0
Publish message: QoS0 Hello, Python! #4
Payload QoS0 Hello, Python! #3
rc: 0
Publish message: QoS0 Hello, Python! #5
Message arrived: QoS0, retained 0, dup 0, packetID 816
rc: 0
Publish message: QoS0 Hello, Python! #6
Payload QoS0 Hello, Python! #4
Message arrived: QoS0, retained 0, dup 0, packetID 816
Payload QoS0 Hello, Python! #5
Message arrived: QoS0, retained 0, dup 0, packetID 816
Payload QoS0 Hello, Python! #6
rc: 0
Publish message: QoS0 Hello, Python! #7
Message arrived: QoS0, retained 0, dup 0, packetID 816
Payload QoS0 Hello, Python! #7

C:\embedded_system\lab_demo\10_2_mqtt>python3 mqtt_client.py
Connecting to 172.20.10.2/Mbed
Connected rc: 0
Subscribed OK
[Received] Topic: Mbed, Message: b'Message from Python!\n'

[Received] Topic: Mbed, Message: b'Message from Python!\n'

[Received] Topic: Mbed, Message: b'Message from Python!\n'

[Received] Topic: Mbed, Message: b'Message from Python!\n'

[Received] Topic: Mbed, Message: b'Message from Python!\n'

[Received] Topic: Mbed, Message: b'QoS0 Hello, Python! #1\x00'
[Received] Topic: Mbed, Message: b'QoS0 Hello, Python! #2\x00'
[Received] Topic: Mbed, Message: b'QoS0 Hello, Python! #3\x00'
[Received] Topic: Mbed, Message: b'QoS0 Hello, Python! #4\x00'
[Received] Topic: Mbed, Message: b'QoS0 Hello, Python! #5\x00'
[Received] Topic: Mbed, Message: b'QoS0 Hello, Python! #6\x00'
```

二、Demo and Checkpoints

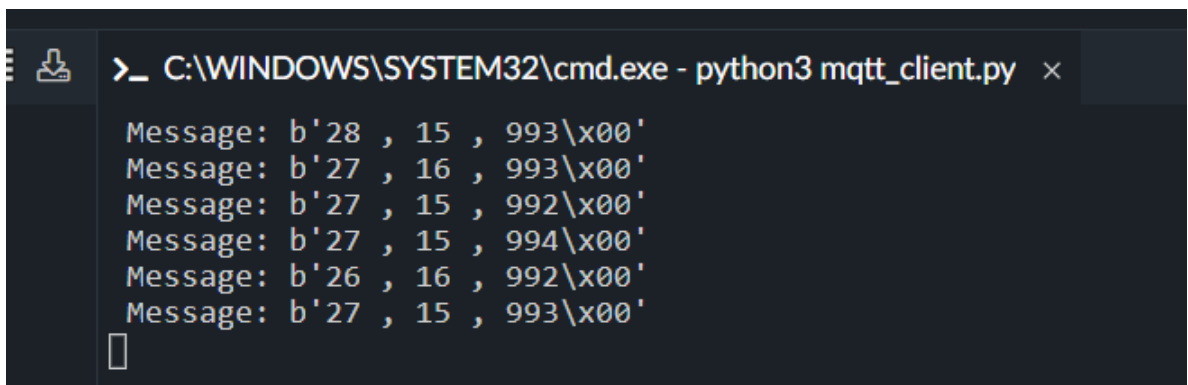
send the data of accelerometer

說明：

新增一個ticker，每0.5秒就call一次publish function，在每一次的publish function 中，都取一次acceleromete的data，之後pub出去。

```
Ticker ticker;
mqtt_thread.start(callback(&mqtt_queue, &EventQueue::dispatch_forever));
ticker.attach(mqtt_queue.event(&publish_message, &client), 500ms);
```

```
void publish_message(MQTT::Client<MQTTNetwork, Countdown>* client) {
    BSP_ACCELERO_AccGetXYZ(pDataXYZ);
}
```

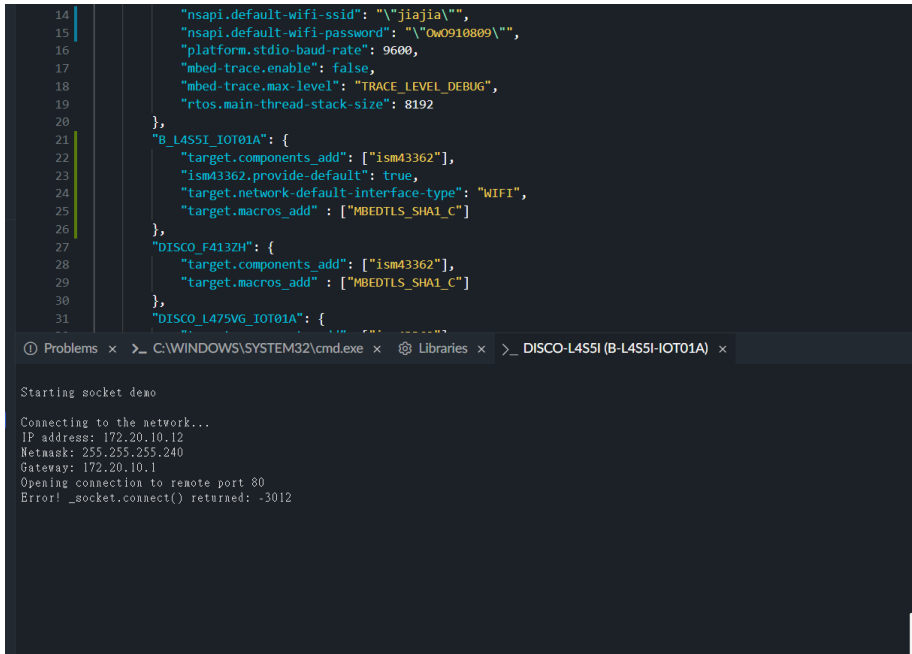


```
>_ C:\WINDOWS\SYSTEM32\cmd.exe - python3 mqtt_client.py ×
Message: b'28 , 15 , 993\x00'
Message: b'27 , 16 , 993\x00'
Message: b'27 , 15 , 992\x00'
Message: b'27 , 15 , 994\x00'
Message: b'26 , 16 , 992\x00'
Message: b'27 , 15 , 993\x00'
```

三、遇到的問題

Error -3012

後來發現是如果Powershell沒有開啟好，就會出現這類問題



```
14     "nsapi.default-wifi-ssid": "\"jiajia\"",
15     "nsapi.default-wifi-password": "\"0w0910809\"",
16     "platform.stdio-baud-rate": 9600,
17     "mbed-trace.enable": false,
18     "mbed-trace.max-level": "TRACE_LEVEL_DEBUG",
19     "rtos.main-thread-stack-size": 8192
20   },
21   "B_L4S5I_IOT01A": {
22     "target.components_add": ["ism43362"],
23     "ism43362.provide-default": true,
24     "target.network-default-interface-type": "WIFI",
25     "target.macros_add": ["MBEDTLS_SHA1_C"]
26   },
27   "DISCO_F413ZH": {
28     "target.components_add": ["ism43362"],
29     "target.macros_add": ["MBEDTLS_SHA1_C"]
30   },
31   "DISCO_L475WG_IOT01A": {
32     "target.components_add": ["ism43362"],
33     "ism43362.provide-default": true,
34     "target.network-default-interface-type": "WIFI",
35     "target.macros_add": ["MBEDTLS_SHA1_C"]
36   }
37 }
```

```
Starting socket demo
Connecting to the network...
IP address: 172.20.10.12
Netmask: 255.255.255.240
Gateway: 172.20.10.1
Opening connection to remote port 80
Error! _socket.connect() returned: -3012
```

SSID問題

手機的wifi名稱可以是符號或是其他文字，但是如果是用符號，可能與電腦等不相容，進而導致連線失敗

四、討論

git remote repository

MQTT的publis跟subscribe只要在相同的topic下就可以使用，並且可以一次傳遞多種型別的message，感覺在很多的介面溝通能夠有很多應用。