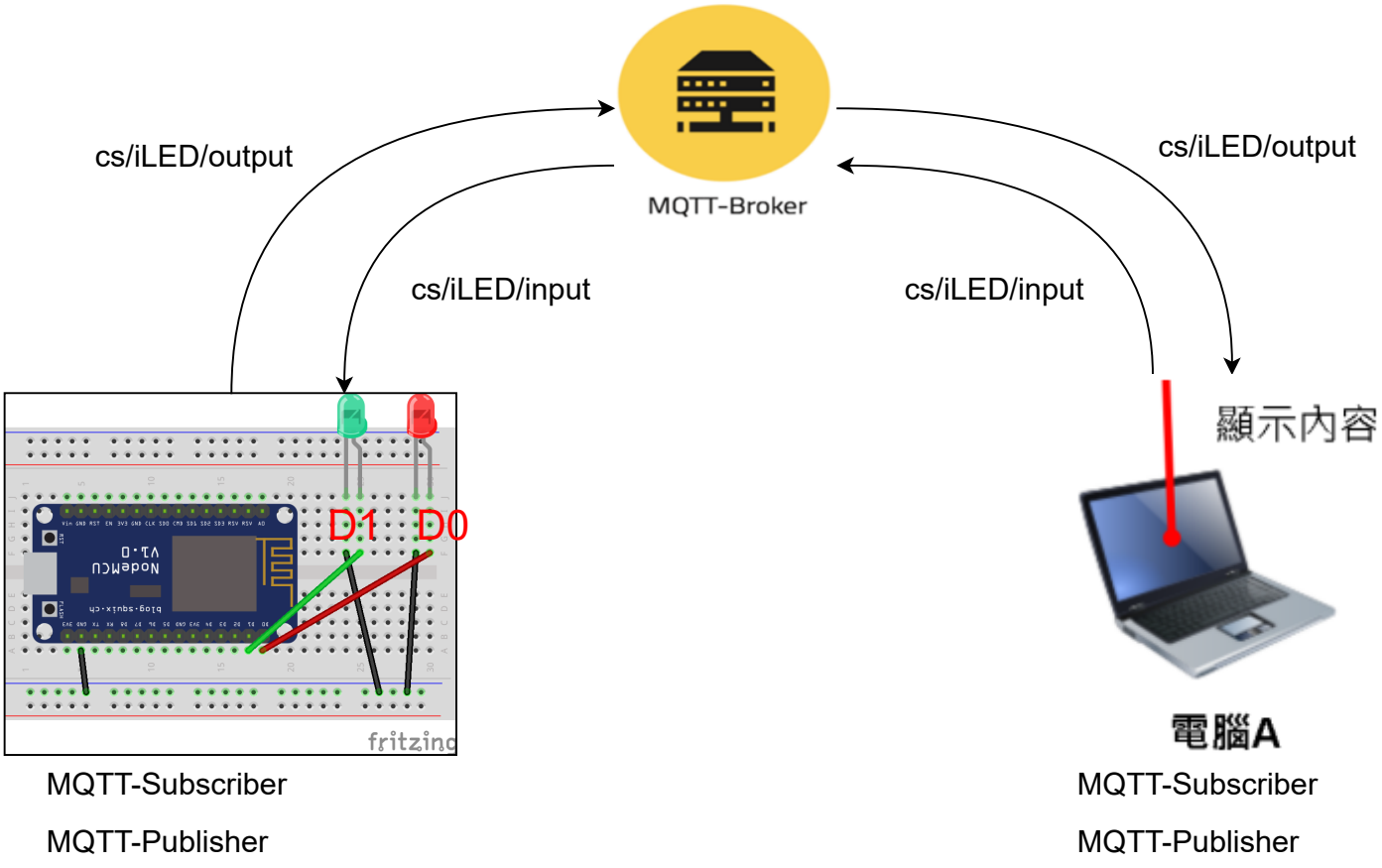
iLED

此章節解說智能燈泡(iLED)的實作方法，應用於遠端控制開關燈。
如果對於建置環境不了解，先參考「NodeMCU_HelloWorld」章節。
如果對於MQTT不了解，先參考「NodeMCU_MQTT」章節。
如果對於LED不了解，先參考「LED」章節

架構圖：
紅LED正極接在D0 Pin腳，負極接在GND
綠LED正極接在D1 Pin腳，負極接在GND



上圖，NodeMCU與電腦A同時具有Publisher、Subscriber角色，NodeMCU發送的topic為 cs/iLED/output，接收的topic為 cs/iLED/input，反之電腦A發送的topic為 cs/iLED/input，接收的topic為 cs/iLED/output。

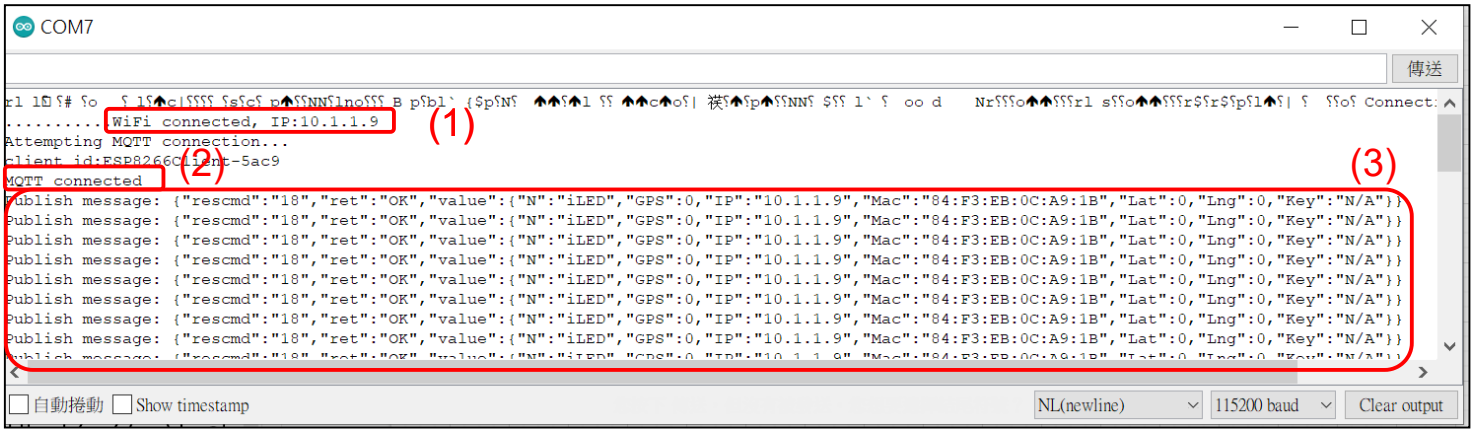
架構介紹：
1. NodeMCU連上 WiFi AP(2.4G only)，也連上MQTT-Broker
2. NodeMCU每2秒發送訊息至 topic(cs/iLED/output)
3. 電腦A連上 WiFi AP，也連上MQTT-Broker
4. 電腦A接收 topic(cs/iLED/output)，也可發送控制LED指令至 topic(cs/iLED/input)

1. 編寫草稿碼 -> 上傳至 NodeMCU 開發板

(1) 定義紅LED、綠LED的Pin腳為D1、D2
(2) 設定WiFi SSID與密碼
(3) 設定MQTT-Broker、Port、發送topic與接收topic

2. 觀看結果

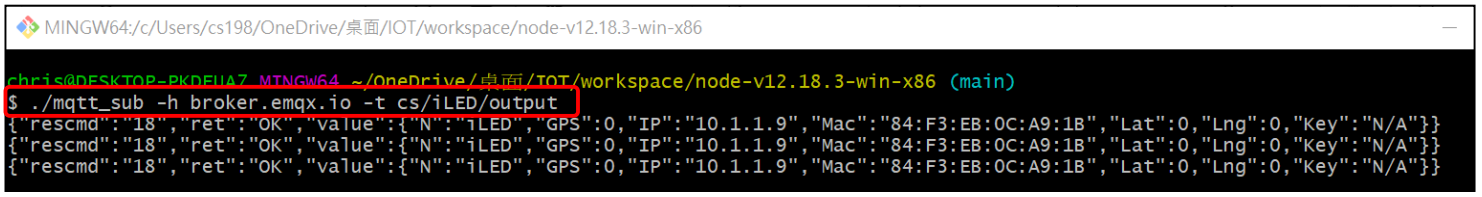
2.1 命令列



(1) NodeMCU 連上WiFi，且拿到IP
(2) NodeMCU連上MQTT-Broker
(3) 每2秒發送訊息至 topic(cs/iLED/output)

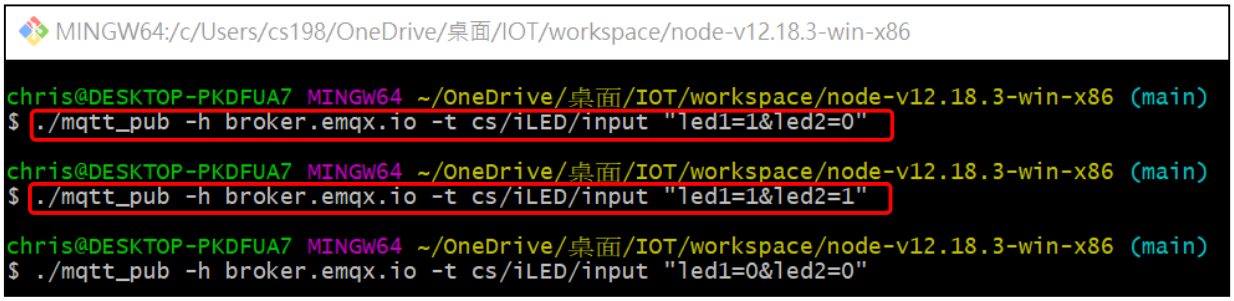
2.2 電腦A - Git Bash介面

Subscriber



(1) 用Node.js - mqtt 工具接收topic(cs/iLED/output)訊息

Publisher



(1) 用Node.js - mqtt 工具發送紅led1亮的訊息至topic(cs/iLED/input)
(2) 用Node.js - mqtt 工具發送紅led1與綠led2都亮的訊息至 topic(cs/iLED/input)