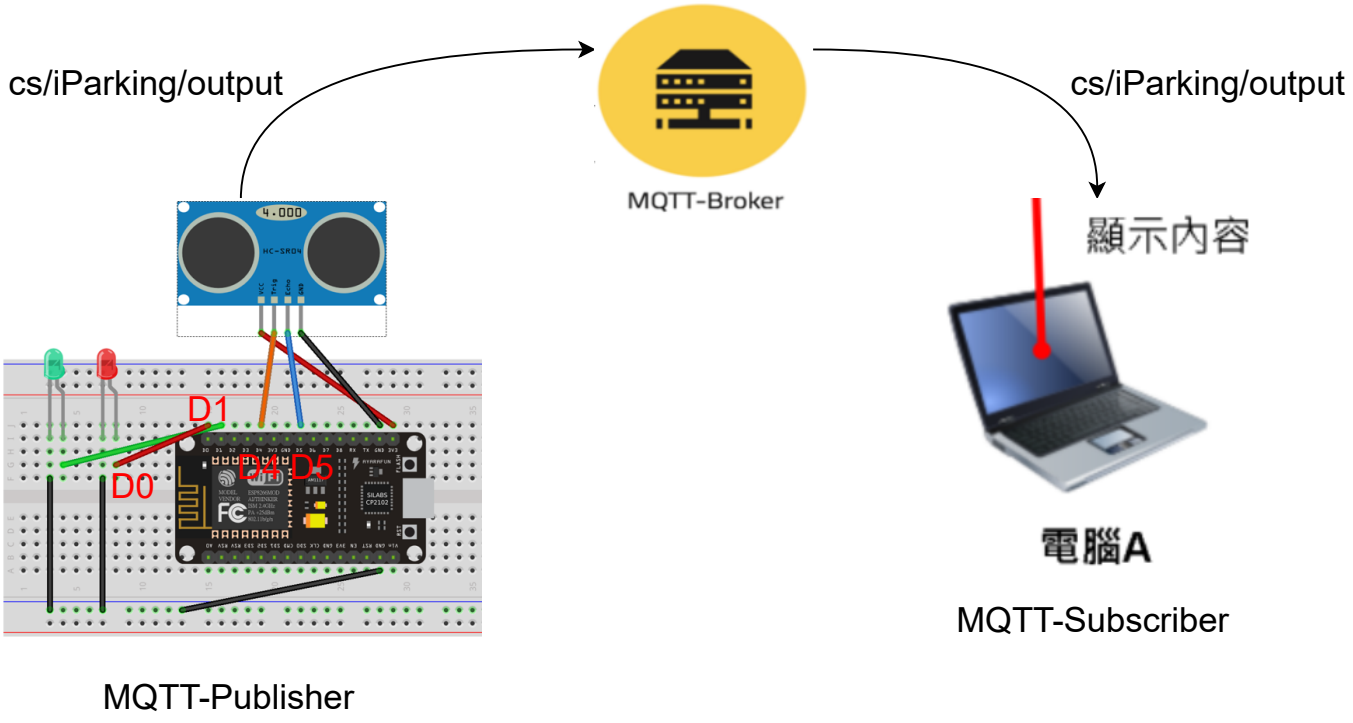
iParking

此章節解說智能停車格(iParking)的實作方法，應用於停車位上偵測車輛距離小於30 CM時，以紅燈表示；以綠燈表示，並回報資訊給中控中心。
如果對於建置環境不了解，先參考「NodeMCU_HelloWorld」章節。
如果對於HC-SR04不了解，先參考「HC_SR04」章節。
如果對於LED不了解，先參考「LED」章節。
如果對於MQTT不了解，先參考「NodeMCU_MQTT」章節。

架構圖：

(1) 將 VCC 與 GND接上
(2) 發送聲波接在 D4 Pin 腳，接收聲波接在 D5 Pin 腳
(3) 紅燈正極接在 D0 Pin 腳
(4) 綠燈正極接在 D1 Pin 腳



上圖，NodeMCU為Publisher角色，電腦A為Subscriber角色。
NodeMCU發送的topic為cs/iParking/output，反之電腦A接收的topic為cs/iParking/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/iParking/output)
3. 電腦A連上WiFi AP，也連上MQTT-Broker
4. 電腦A接收topic(cs/iParking/output)，知道該停車位狀態

1. 原理解說

HC-SR04 透過 40kHz的超聲波運作，此頻率人類無法聽到。當 Trigger 發送超聲波後，超聲波遇到障礙物(**Object**)而反射，此時Echo接收到反射的超聲波，得到測量時間(Duration)。我們知道聲音的速度約 340M/sec，因此可從公式得出聲音走的距離(Sonar Distance)：

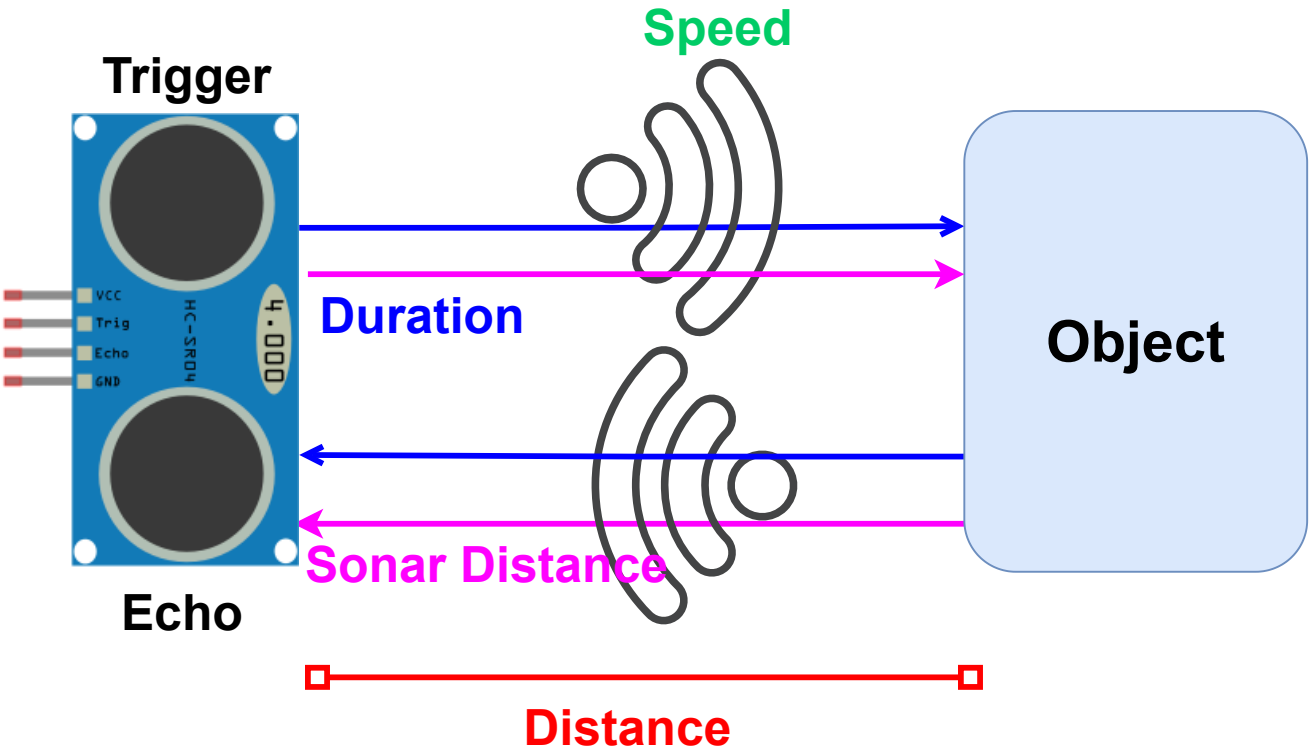
Sonar Distance = Speed * Duration

進一步得到測量距離(Distance)：

Distance = Sonar Distance / 2

公式總結為：

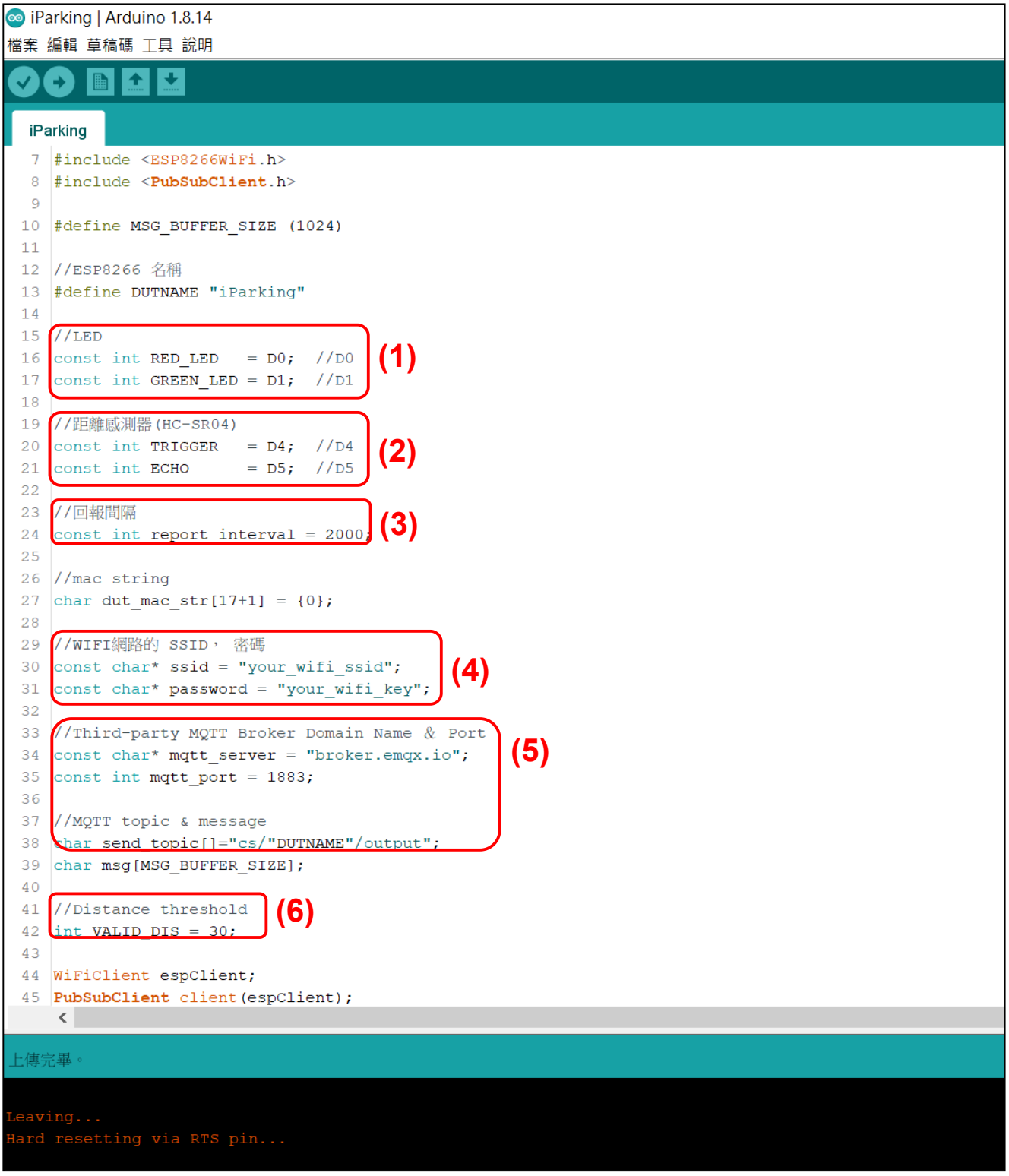
Distance = Speed * Duration / 2



2. 應用說明

從步驟1可得到感測HC-SR04與障礙物的距離，設計出每2秒偵測與障礙物的距離，距離單位為公分(CM)，並設定當距離小於30 CM時，紅燈亮起； 反之綠燈亮起。
紅燈代表：車位有車 / 離障礙物很近。
綠燈代表：車位無車 / 離障礙物很遠。

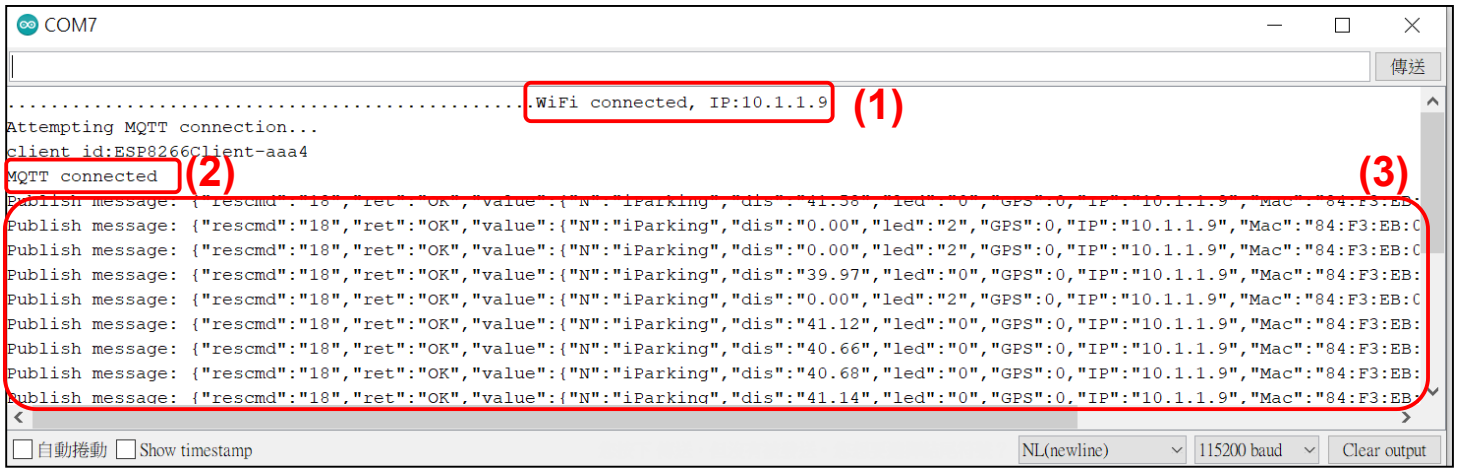
3. 編寫草稿碼 -> 上傳至 NodeMCU 開發板



(1) 定義紅綠燈pin腳為 D0、D1
(2) 定義HC-SR04 trigger pin腳為D4、echo pin腳為D5
(3) 設定回報間隔時間，單位毫秒(ms)
(4) 設定WiFi SSID與密碼
(5) 設定MQTT-Broker、Port、及發送的topic
(6) 定義與物體的距離，單位公分(cm)

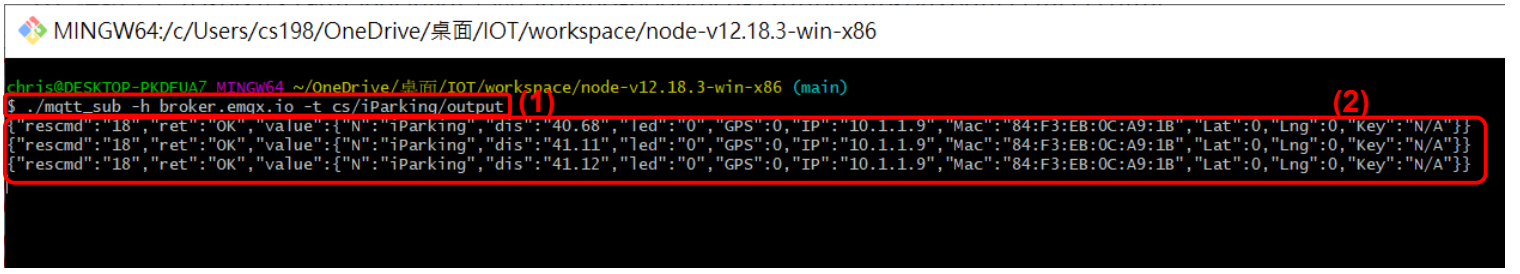
4. 觀看結果

4.1 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/iParking/output)

4.2 電腦A - Git Bash介面



(1) 用Node.js - mqtt 工具接收 topic(cs/iParking/output)訊息
(2) 接收到的訊息內容帶有dis欄位，代表與車輛的距離