**物聯網核心技術實習作業二：NB-IoT**

**國立中正大學資訊工程系 黃仁竑教授**

**實驗目的**

使用AT command 對 NB-IoT 模組下指令，進行查詢、設定、發送資料等功能。

**實驗所需軟體與設備**

環境：

* Windows 10
* Arduino IDE 1.8.12
* Python 3.8.5

硬體：

* GAT 4552
* Micro USB cable
* 杜邦線x2
* SIM card
* NB-IoT 天線

目錄

[一、 NB-IoT模組 1](#_Toc54317738)

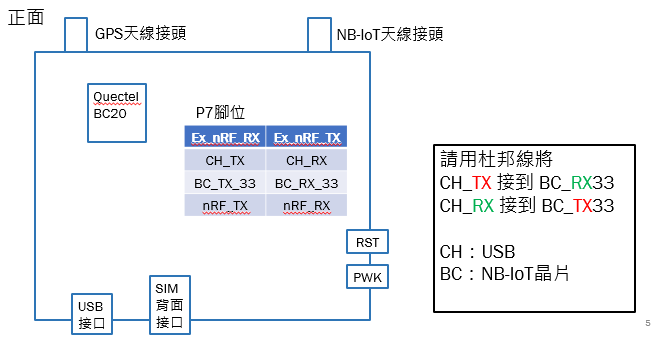
[二、 TCP & UDP 2](#_Toc54317739)

[三、 實驗介紹 3](#_Toc54317740)

[四、 實驗步驟 3](#_Toc54317741)

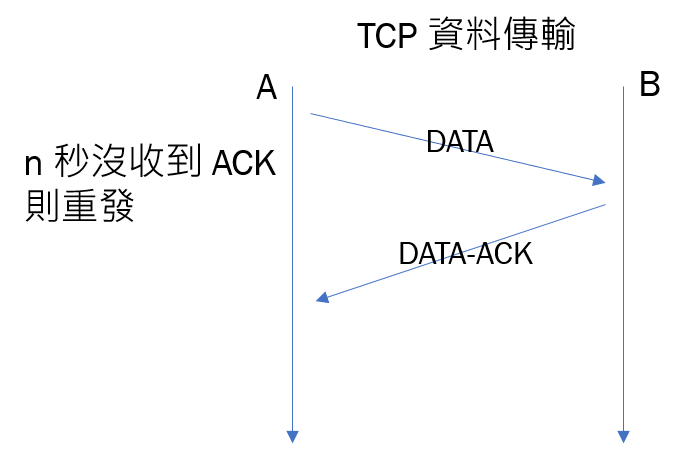
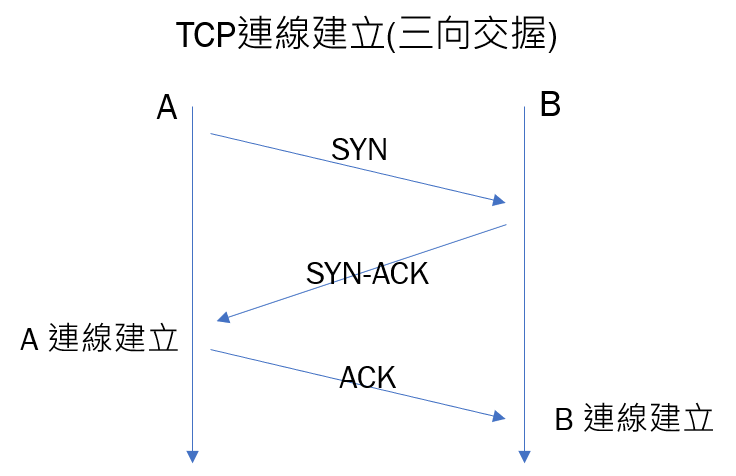
1. NB-IoT模組

* 操作環境
  + 3.3V 單電源供電
  + 操作溫度：-35°C ~ 75°C
* 特性
  + 可快速整合至感應器
  + 提供 AT 指令集
  + 18.7mm x 16.0mm x 2.1mm 針腳封裝
  + 介面：UART
  + 高靈敏度：-129dBm
  + 支援頻道：LTE Band 5, 8

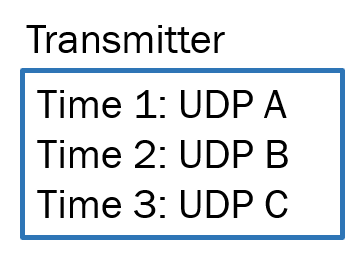


1. TCP & UDP

* TCP 與 UDP 為傳輸層中，TCP/IP 的兩個主要的協定
* Transmission Control Protocol(TCP)
  + TCP 提供可靠的傳輸，  
    包含封包的循序編號、資料確認與重送、錯誤檢查與流量控制等功能

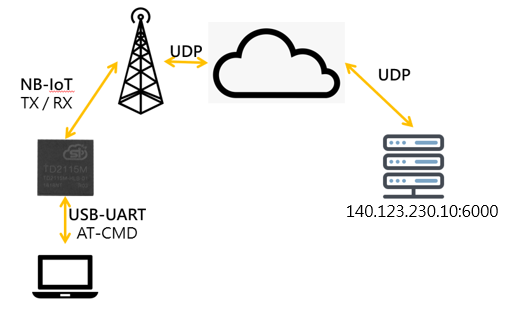


* User Datagram Protocol(UDP)
  + 非可靠的非連線型資料流傳輸服務
  + 在沒有事先建立連線的情況下，送出訊息，  
    在通訊上的可靠度較低，不過負擔較小，速度較快
  + 不保證資料正確接收，也不保證資料抵達順序
  + 適用於可容忍部分封包丟失且流量大的應用，例如影像、語音等

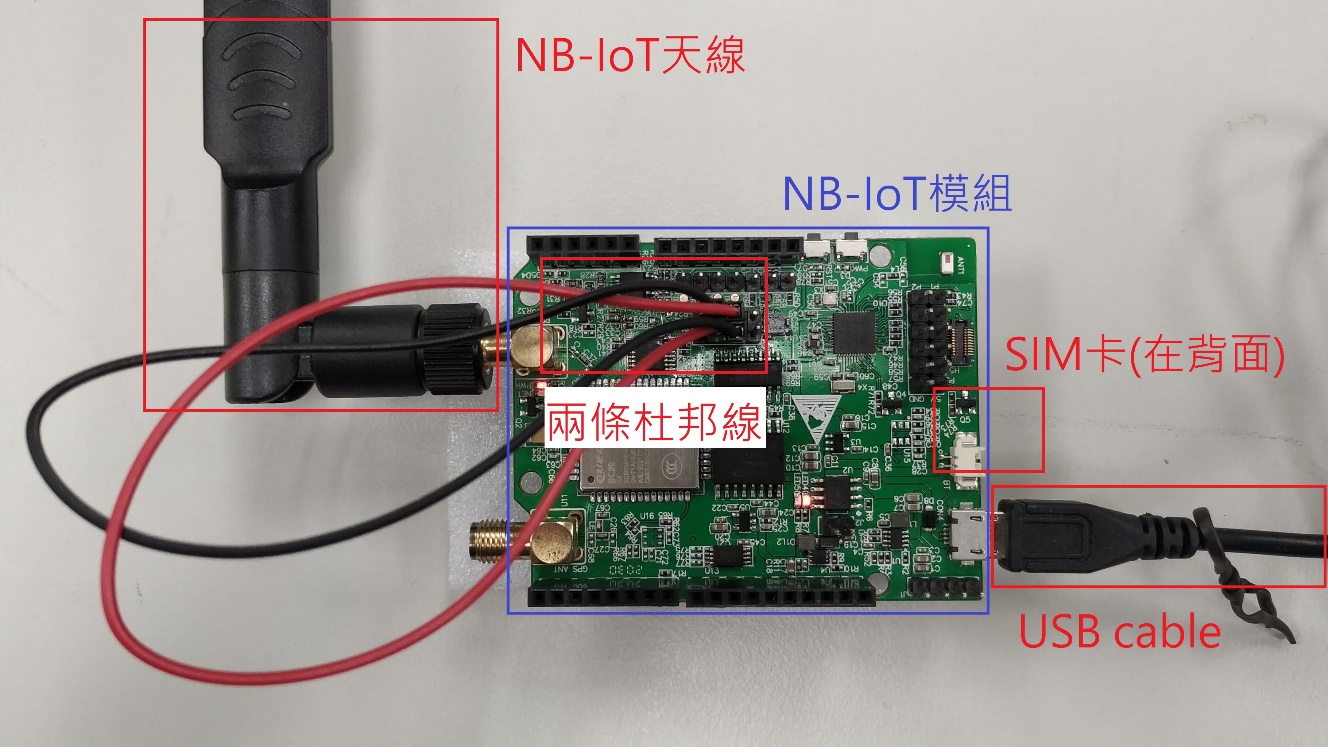


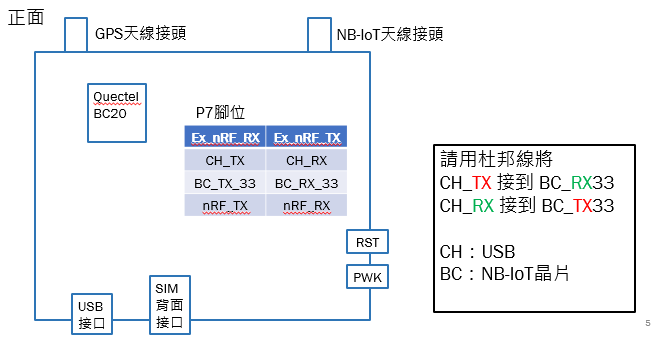
1. 實驗介紹

* 目標一：透過 COM port 下 AT command
* 目標二：透過 AT command 使模組收發資料
* References
  + [AT command 相關參考資料](https://drive.google.com/drive/folders/106XDGp3PlqA-3qpZMcBunnc6WpIJpUOR?usp=sharing)
* 實驗架構



1. 實驗步驟
2. 連接相關設備





1. 安裝兩個[開發版相關驅動](https://drive.google.com/drive/folders/1Zuh-HtRhm4mEYCJg0-w1UoGr8DQJ7des?usp=sharing)

解壓縮後分別執行安裝

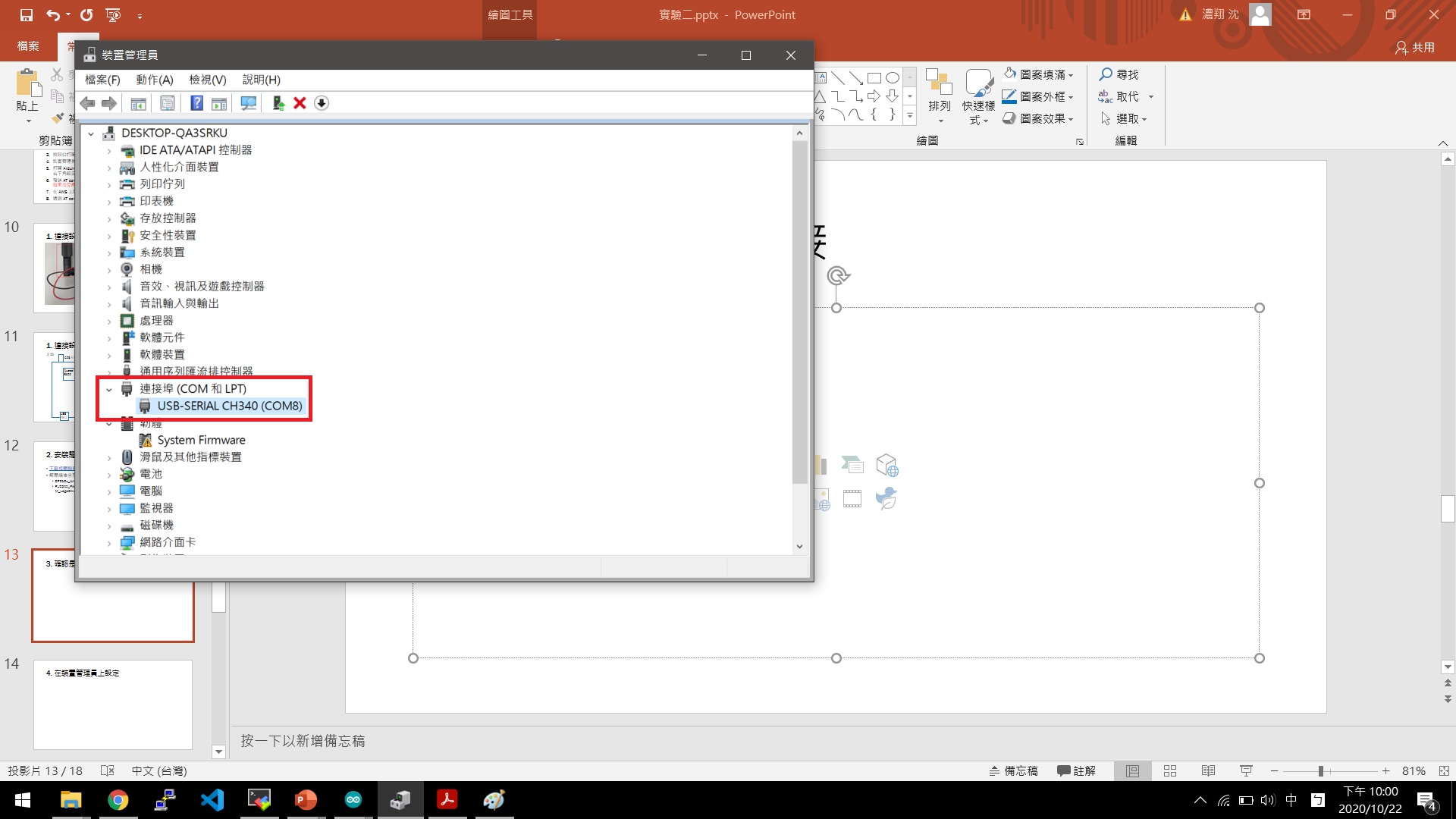
CP210x\_Universal\_Windows\_Driver (1)\CP210xVCPInstaller\_x64.exe

PL23XX\_Prolific\_DriverInstaller\_v202\PL23XX

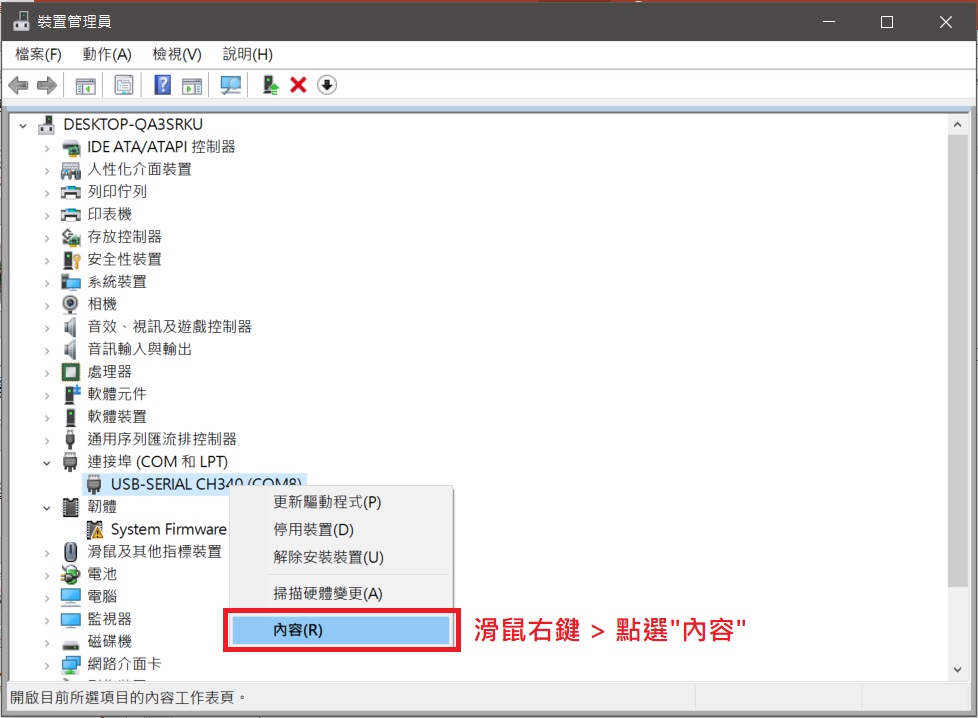
1. 控制台打開裝置管理員，確認裝置是否正常

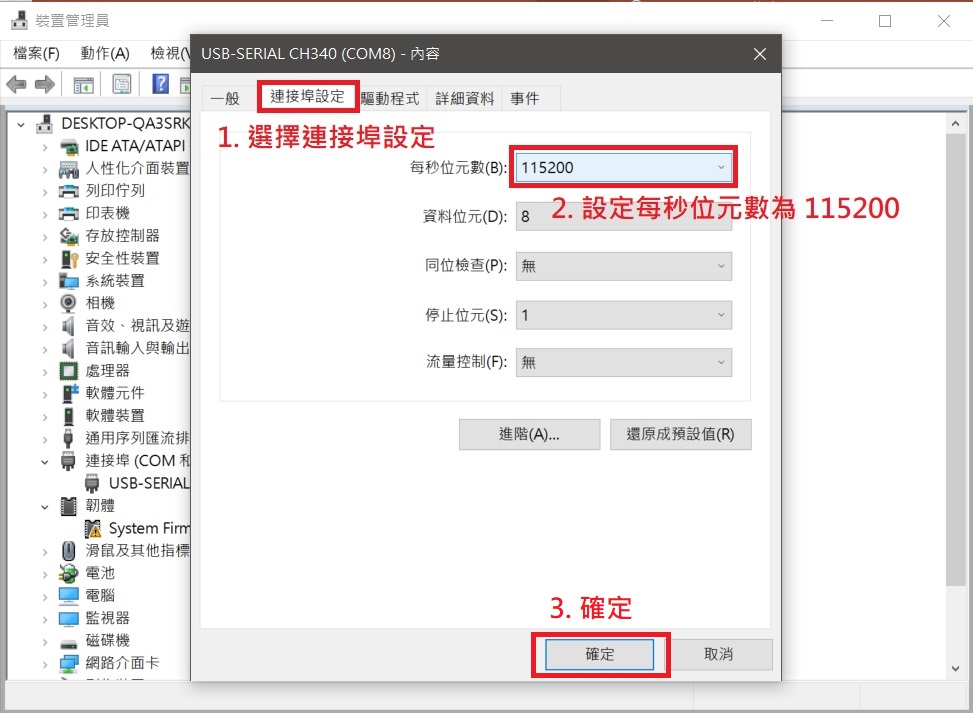
成功安裝驅動後，打開：控制台>硬體和音效>裝置管理員

確認裝置是否正常運行，裝置連接後應如下圖顯示



1. 裝置管理員，設定每秒位元數(B) = 115200





1. 打開 Arduino IDE 的 Serial monitor  
   右下角設定為 Both NL & CR、115200 baud



1. 發送 AT command 進行相關設定與測試，  
   如果沒反應，請按住板子上的 PWK 按鈕 2 秒放開之後重下指令

可參考[AT command references](https://drive.google.com/drive/folders/106XDGp3PlqA-3qpZMcBunnc6WpIJpUOR?usp=sharing)

可以透過 Serial monitor 對裝置下達 AT command

AT command 格式通則

查詢指令可支持的參數：AT+<cmd>=?

查詢指令目前的參數：AT+<cmd>?

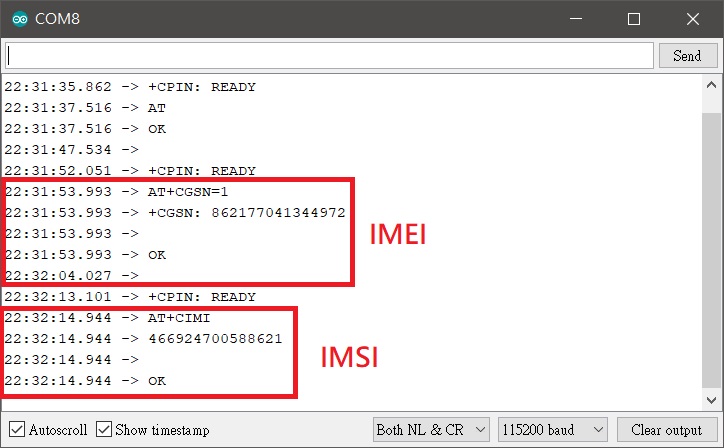
設置指令的參數：  
AT+<cmd>=<para\_1>[,<para\_2>[,<para\_3>[…]]]

如果下達指令後沒有收到 response，請長按 PWK 兩秒放開後  
重新下達 AT command

|  |  |
| --- | --- |
| **指令** | **用途** |
| AT+QSCLK=0 | 關閉設備休眠 |
| AT+QBAND=1,8 | 設定一個頻段：8 |
| AT+CFUN=1 | Enable NB module |
| AT+CGSN=1 | 查詢設備 IMEI |
| AT+CIMI | 查詢 SIM 卡卡號 |

紅色：只能在 enable NB module 後使用

成功顯示如下資訊即完成實驗目標一

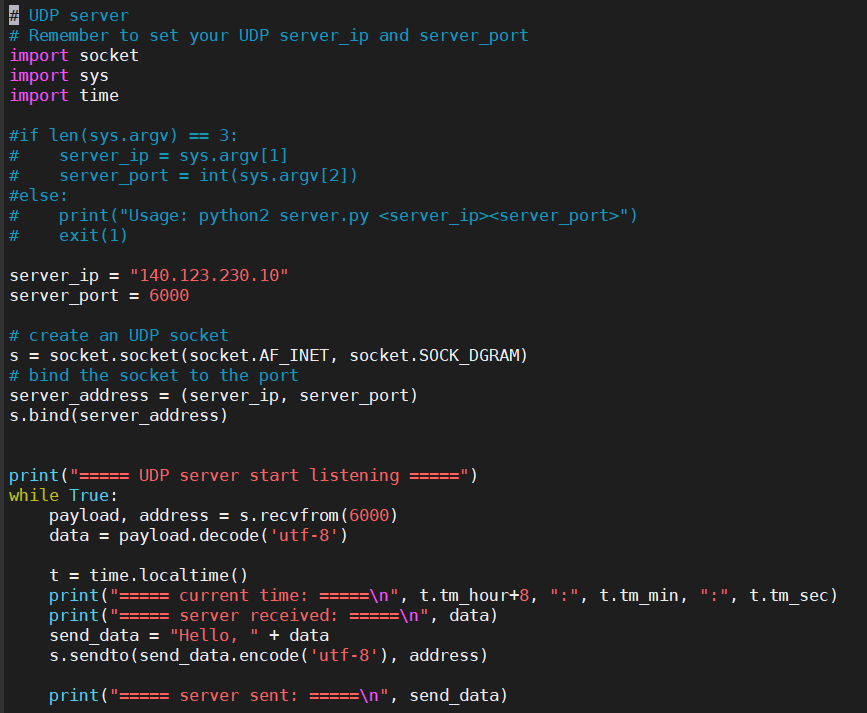


1. UDP server

UDP server python code

可以自架 server 執行本程式

也可以直接送 UDP request 到  
140.123.230.10:6000

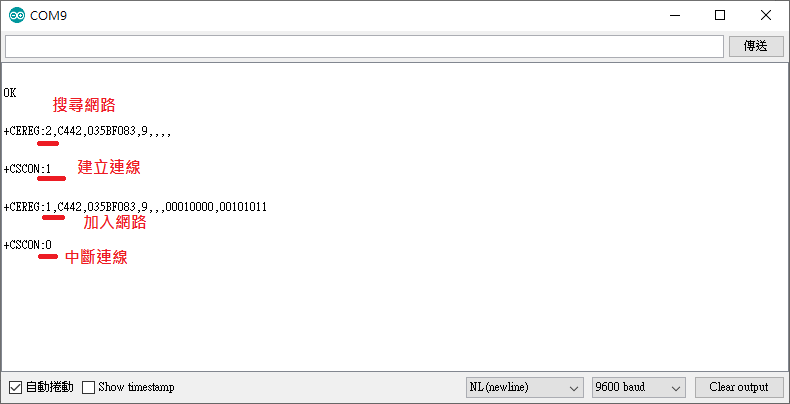


1. 透過 AT command 收發資料

|  |  |
| --- | --- |
| **指令** | **用途** |
| AT+CFUN=1 | Enable NB module |
| AT+CGDCONT=1,"IPV4V6",,,0,0,,,,,0,0 | 設定 IPV4/IPV6 |
| AT+CGATT=1 | 附著至NB網路 |
| AT+QICFG="dataformat",1,1 | 設定傳送與接收格式為HEX |

若 AT+CGDCONT 無法設定，請先將 AT+CGATT=0

|  |  |
| --- | --- |
| **指令** | **用途** |
| AT+CMEE=1 | 開啟進階錯誤回報(建議開啟) |
| AT+CEREG=5 | 開啟註冊狀態回報 |
| AT+CSCON=1 | 開啟基站連線狀態回報 |

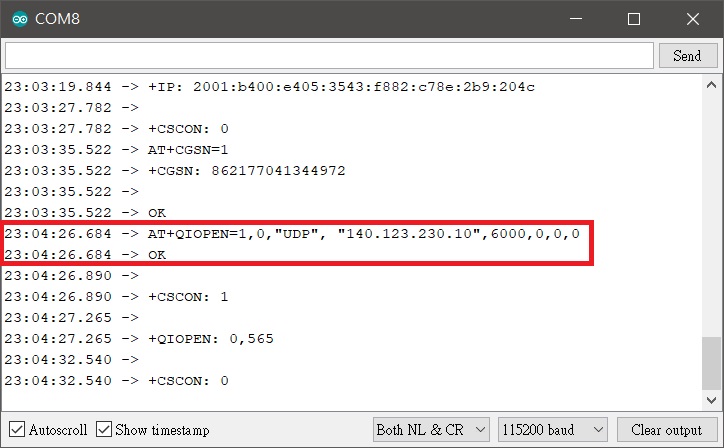


|  |  |
| --- | --- |
| **指令** | **用途** |
| AT+QIOPEN=1,0,"UDP",<Addr/DomainName>,<DstPort>,0,0,0 | 建立 Socket 指令格式 |

Addr：server IP address  
ex. 140.123.230.10  
也可輸入 domain name

DstPort：port of service  
ex. 6000 (UDP port)

Ex. AT+QIOPEN=1,0,"UDP","140.123.230.10",6000,0,0,0

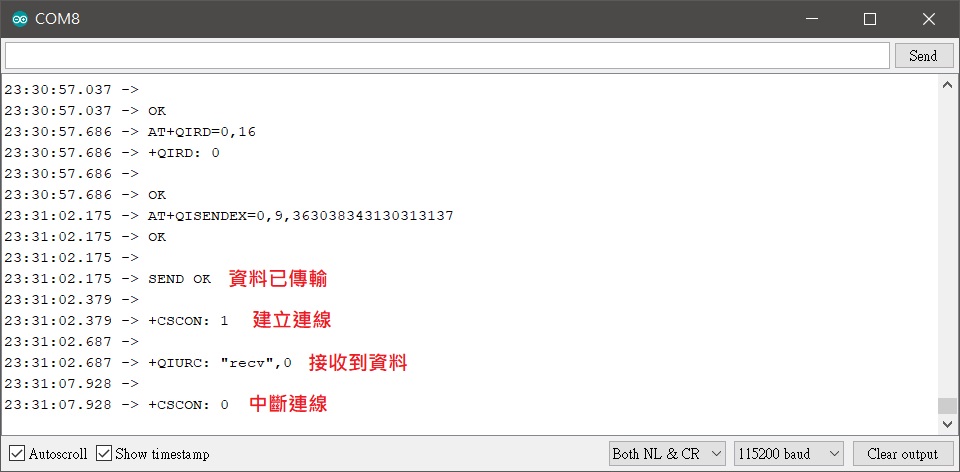


|  |  |
| --- | --- |
| **指令** | **用途** |
| AT+QISENDEX=0,<length>,<data> | 傳送 UDP 資料 |

length：資料長度(IN BYTES)

data：資料內容(HEX STRING)

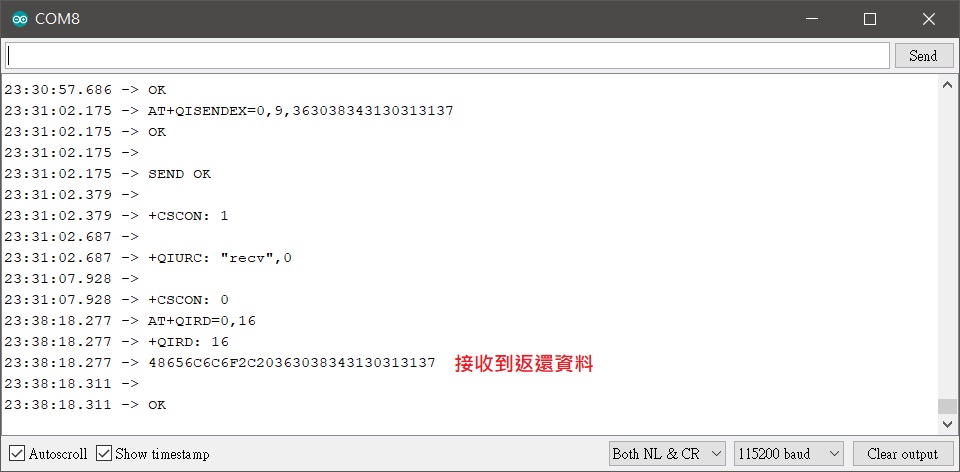
Ex. AT+QISENDEX=0,9,363038343130313137  
上述指令會將 "608410117"   
傳送至140.123.230.10:6000 ([ASCII](https://zh.wikipedia.org/wiki/ASCII))



|  |  |
| --- | --- |
| **指令** | **用途** |
| AT+QIRD=0,<length> | 接收 UDP 資料 |

length：資料長度(IN BYTES)  
也可以直接設定最大長度512

Ex. AT+QIRD=0,16



將自己的學號送到 UDP server 上

接收返還資料，應為 "48656C6C6F2C20<你學號的Hex String>"  
轉換為 ASCII 的字元應為 "Hello, <你的學號>"

Ex. 傳送 608410117，應收到48656C6C6F2C20363038343130313137

若正確，即完成實驗目標二