

GAT4553 - 課程1

參考資料：清大教材PPT

[7-2. NB-IoT傳輸模組.pdf](#)

基礎使用

架構

目標1

目標2

作業：(現場檢查)

基礎使用

教材

- 請清點今天會使用之教材

- Micro-USB線 _____



- NB-IoT 天線 _____



- NB-IoT Sim卡 _____

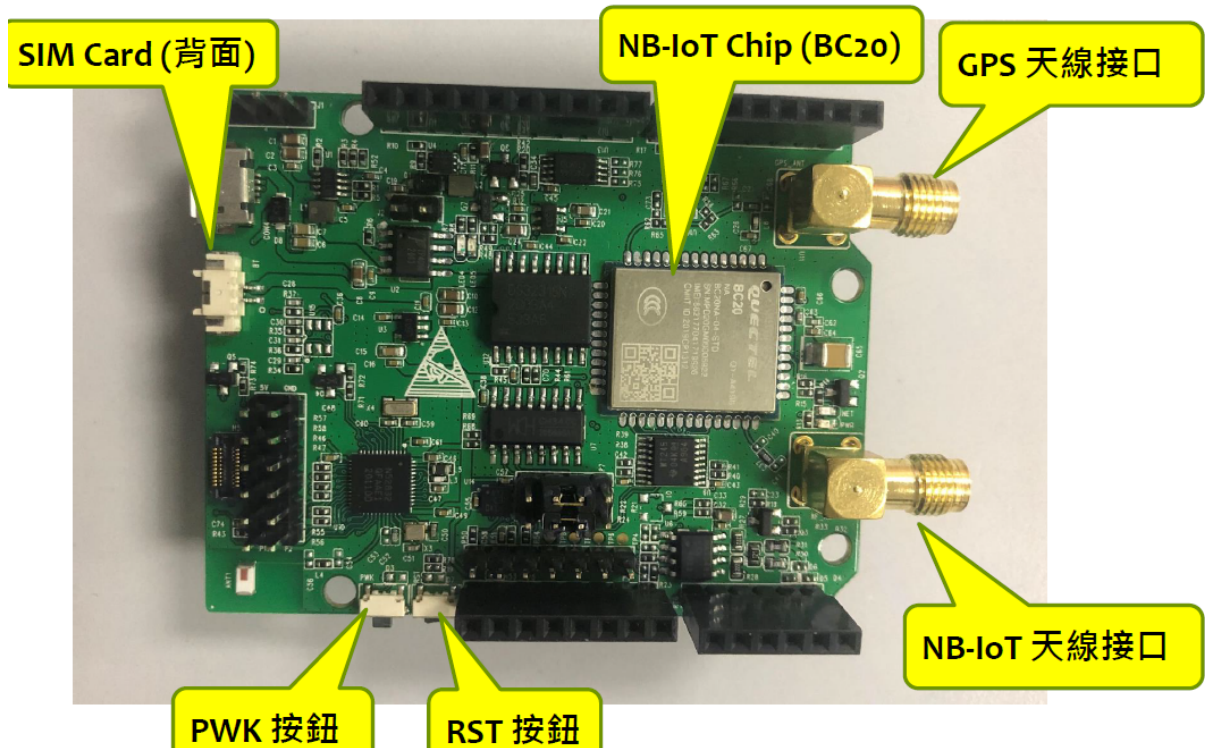


- NB-IoT模組開發板 _____



- 完成實驗後請將各項教材繳回給助教!!

NB-IoT 通訊模組

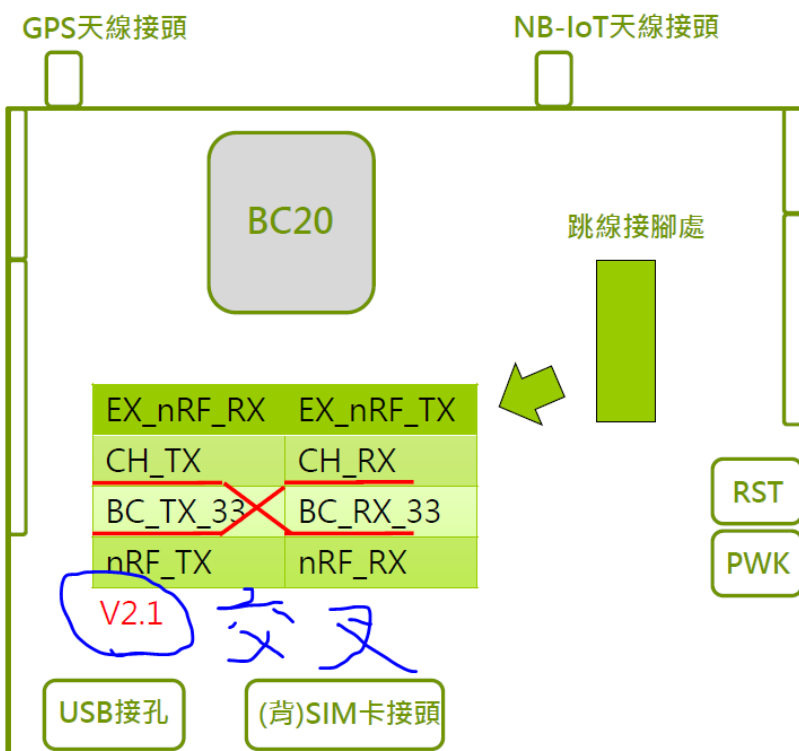


!!請注意板子版本，有分V2.1與V2.2，腳位不相同!!

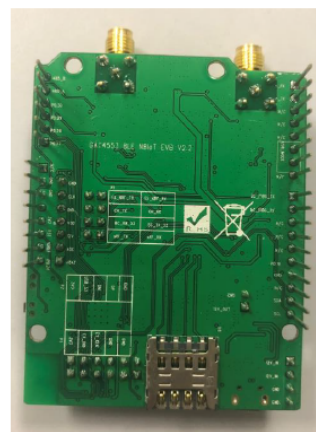
V2.1

NB-IoT 通訊模組

以正面觀看之各腳位



翻面可看到各腳位說明



V2.1 請將
BC_TX_33接到CH_RX
BC_RX_33接到CH_TX

BC : NB-IoT晶片
CH : USB

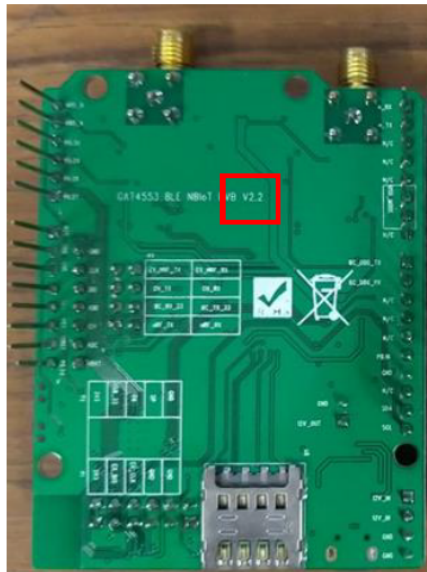
*請注意V2.1和V2.2板子腳位略有不同

14

V2.2

NB-IoT 通訊模組

*請注意V2.1和V2.2板子腳位略有不同



V2.2 (正面可用跳線連接下表腳位)

| | |
|-----------|-----------|
| EX_nRF_RX | EX_nRF_TX |
| CH_RX | CH_TX |
| BC_TX_33 | BC_RX_33 |
| nRF_RX | nRF_TX |

並排

AT指令

NB-IoT 通訊模組的 AT Commands

| | 用途 | UE Return |
|---|-----------|------------------------------|
| AT+QBAND? | 確認設置頻段信息 | +QBAND: 5,8 OK |
| AT+CGSN=1 | 查詢設備IMEI號 | +CGSN: 123456789012347 OK |
| Note: IMEI如有錯誤，自動模式下先設置AT+CFUN=0關閉MT功能後寫入 | | |
| AT+CFUN? | 確認MT的功能 | +CFUN: 1 OK |
| Note: 如果開啟自動聯網功能，重啟後默認CFUN=1 | | |

NB-IoT 通訊模組的 AT Commands

| | 用途 | UE Return |
|--|--------------|-----------------------|
| AT+CIMI | 查詢USIM卡IMSI號 | 460001357924680 OK |
| Note: 如查不到IMSI號，請確認卡片是否開通與是否插好 | | |
| AT+CSQ | 確認信號強度 | +CSQ: 21,99 OK |
| Note: 保持正常通信狀態下，信號強度值不能低於15即-80dBm左右 信號強度值為99時說明沒有信號。 | | |

| 數值 | 信號強度 |
|--------|-----------------------------|
| 0 | -113dBm or less |
| 1 | -111dBm |
| 2...30 | -109...-53dBm |
| 31 | -51dBm or greater |
| 99 | Not known or Not detectable |

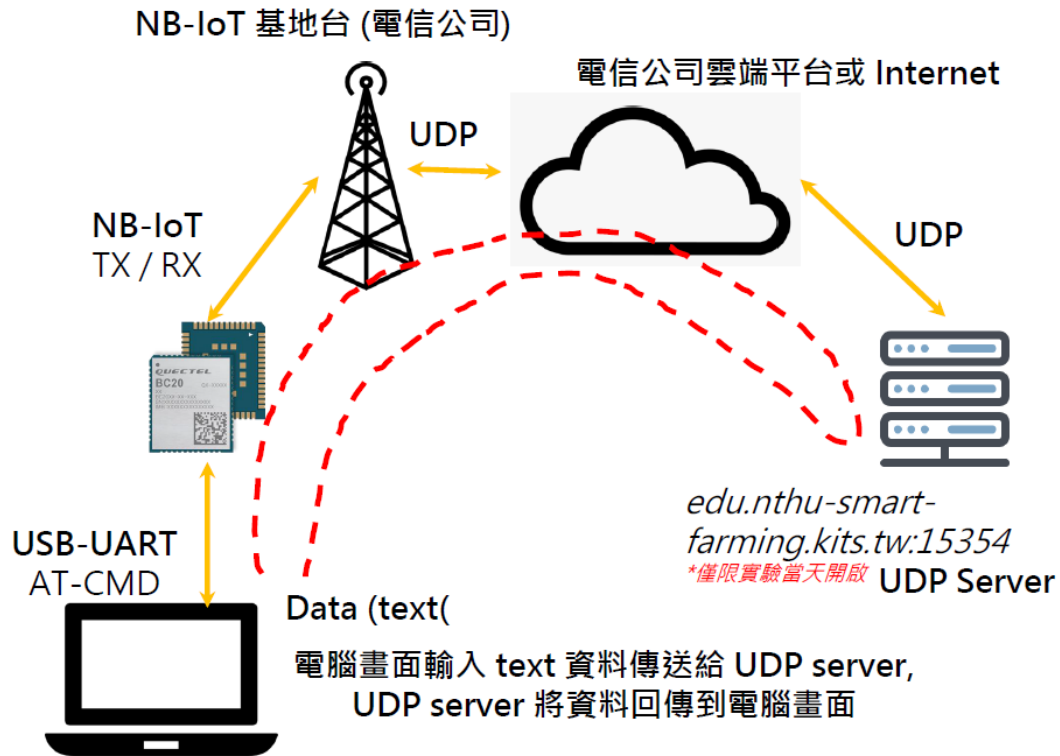
18

NB-IoT 通訊模組的 AT Commands

| | 用途 | UE Return |
|---|------------|-------------------|
| AT+CGATT? | 查詢是否有連上網路 | +CGATT: 1 OK |
| Note: 如果反為+CGATT:0說明未上網成功，有30s的延時，可一直查詢。過了30s後仍未成功，則判定為失敗 | | |
| AT+CEREG? | 查詢當前網路註冊狀態 | +CEREG: 1,1 OK |
| Note: 第一個參數1表示是否能註冊網路，第二個參數1代表成功註冊網路 | | |
| AT+CSCON? | 查詢連接狀態 | +CSCON: 0,1 OK |
| Note: 第一個參數0表示關閉非請求結果碼，第二個參數1表示為連接狀態 | | |

架構

NB-IoT 通訊模組實驗架構



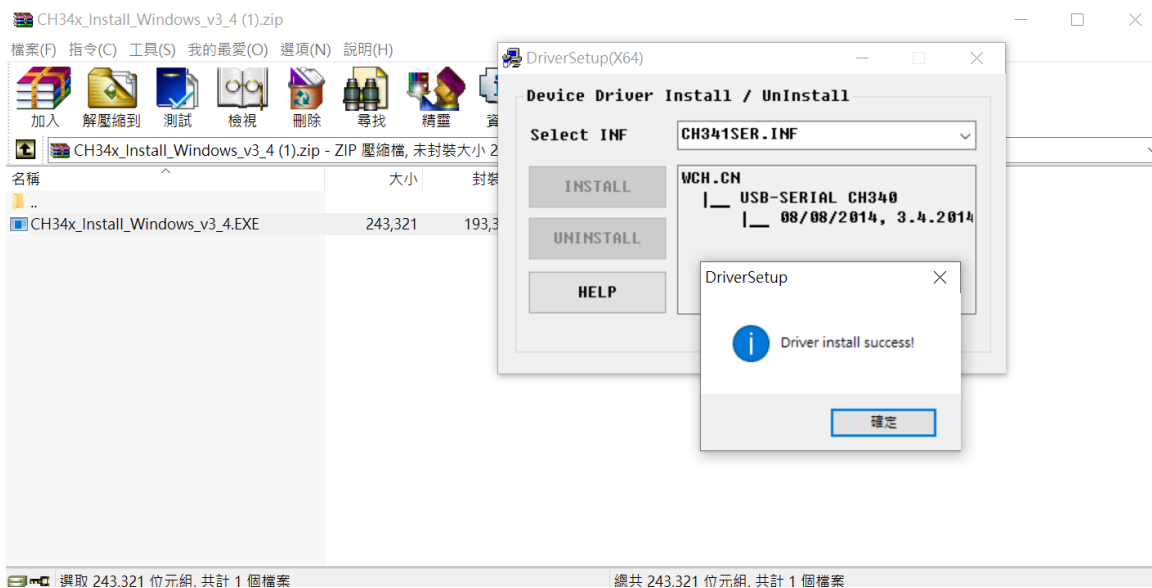
目標1

1. 安裝驅動

CH340 Drivers for Windows, Mac and Linux

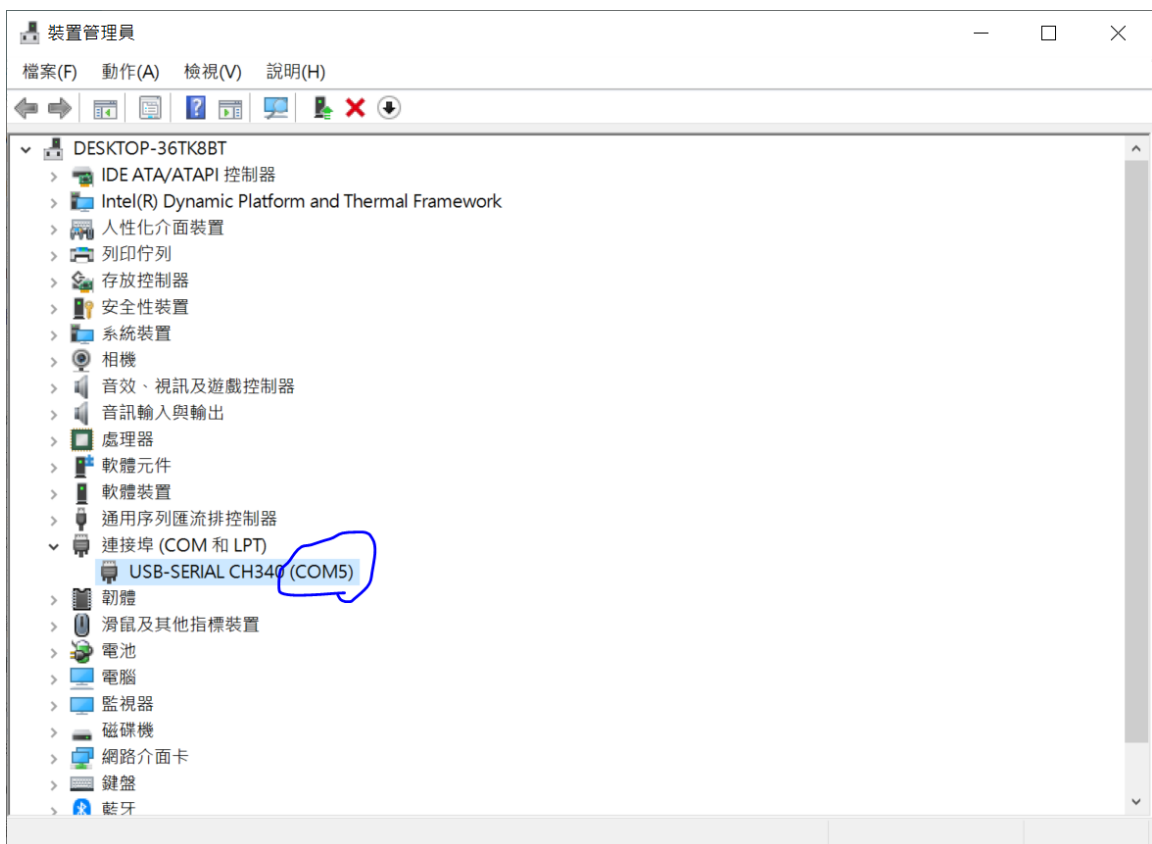
<https://sparks.gogo.co.nz/ch340.html>

[CH34x_Install_Windows_v3_4.zip](#)

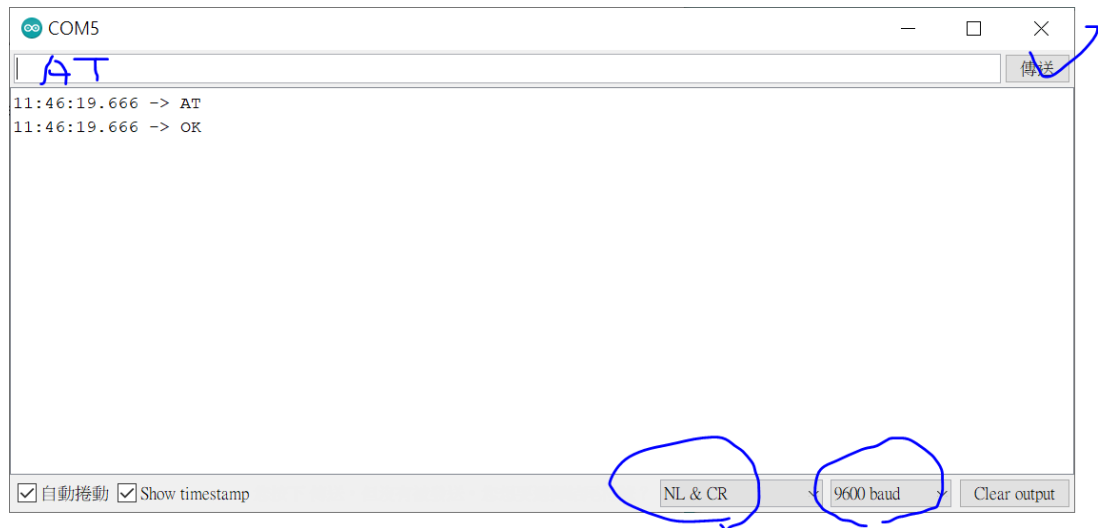


2. 確認模組狀態

Com port setting hint: Baud rate: 115200 / NL & CR



1. 安裝 UART 通訊軟體 Arduino IDE 即可，注意序列埠
2. 打開序列埠監控視窗，設定 board rate(115200)
NL&CR(Newline & Carriage Return)



3. 長按 4553 板子的開機鍵 PWK(約 2 秒)

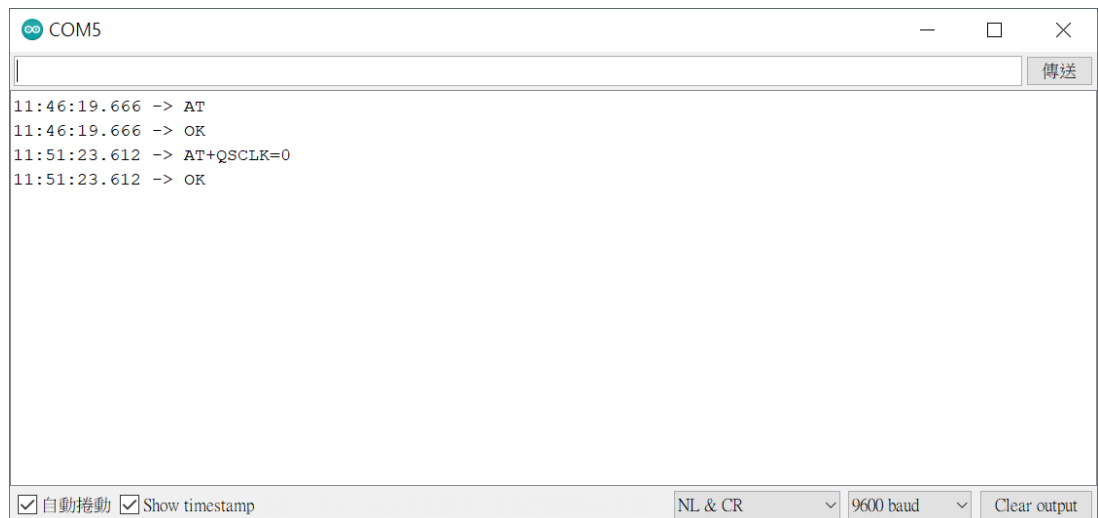
4. 確認通訊成功：輸入 **AT**，須回復 **OK**

補充：有時候輸入會失敗(因為延遲)，多試幾次

AT

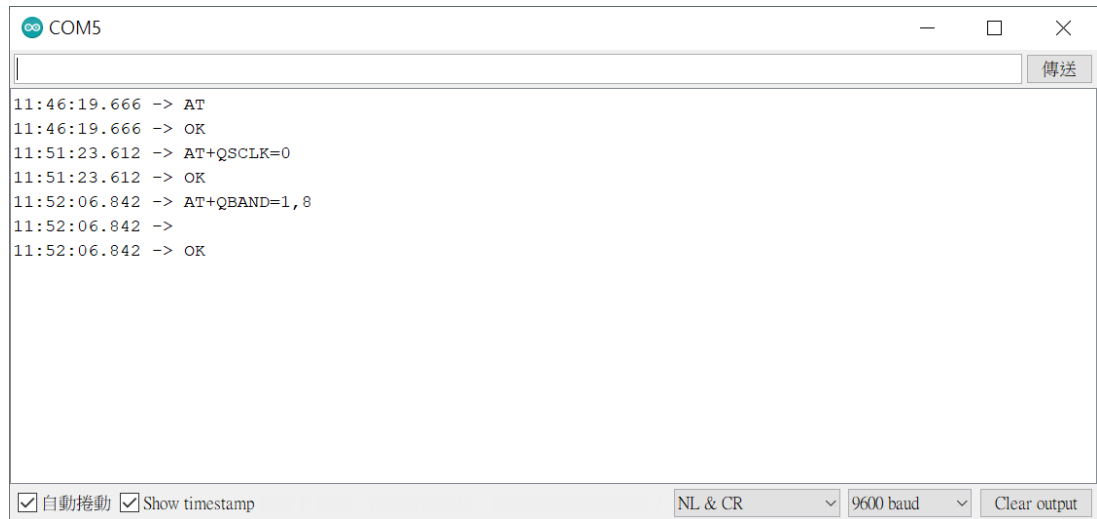
5. 關閉設備休眠

AT+QSClk=0



6. 確認設置頻段 8 需要開啟

AT+QBAND=1,8



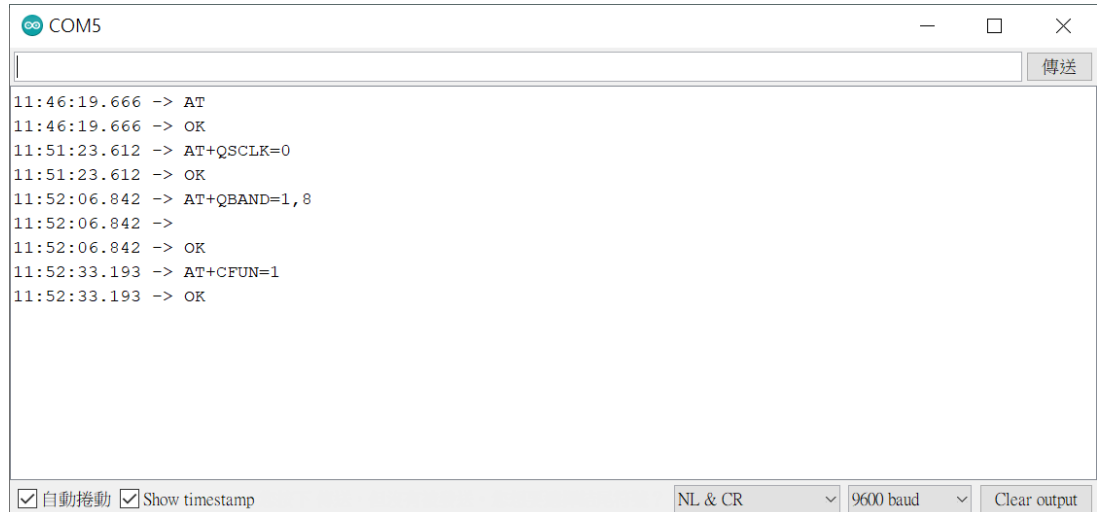
The screenshot shows a terminal window titled 'COM5' with a text input field and a '傳送' (Send) button. The output area displays the following log:

```
11:46:19.666 -> AT
11:46:19.666 -> OK
11:51:23.612 -> AT+QSClk=0
11:51:23.612 -> OK
11:52:06.842 -> AT+QBAND=1,8
11:52:06.842 -> 
11:52:06.842 -> OK
```

At the bottom, there are checkboxes for '自動捲動' (Auto scroll) and 'Show timestamp', both of which are checked. To the right, there are dropdown menus for 'NL & CR' and '9600 baud', and a 'Clear output' button.

7. 啟動 NB IoT 模組

AT+CFUN=1



This screenshot is similar to the previous one but includes an additional command and its response:

```
11:46:19.666 -> AT
11:46:19.666 -> OK
11:51:23.612 -> AT+QSClk=0
11:51:23.612 -> OK
11:52:06.842 -> AT+QBAND=1,8
11:52:06.842 -> 
11:52:06.842 -> OK
11:52:33.193 -> AT+CFUN=1
11:52:33.193 -> OK
```

The interface elements at the bottom (checkboxes, dropdowns, and buttons) are identical to the previous screenshot.

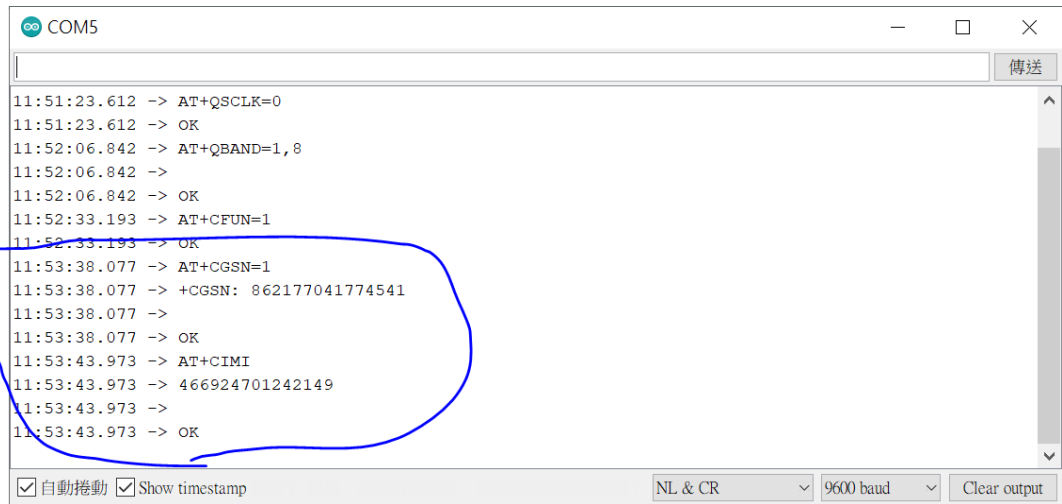
8. 查詢 IMEI / IMSI 號碼 確認 SIM 卡正常

IMEI

AT+CGSN=1

IMSI

AT+CIMI



```
11:51:23.612 -> AT+QSClk=0
11:51:23.612 -> OK
11:52:06.842 -> AT+QBAND=1,8
11:52:06.842 -> OK
11:52:06.842 -> OK
11:52:33.193 -> AT+CFUN=1
11:52:33.193 -> OK
11:53:38.077 -> AT+CGSN=1
11:53:38.077 -> +CGSN: 862177041774541
11:53:38.077 -> OK
11:53:38.077 -> OK
11:53:43.973 -> AT+CIMI
11:53:43.973 -> 466924701242149
11:53:43.973 -> OK
```

目標2

透過 AT CMD 傳送與接收資料

1. 啟動 NB 模組

AT+CFUN=1

2. 去附著(離開)網路

AT+CGATT=0

3. 設定傳輸 IPV4 / IPV6 位址

AT+CGDCONT=1, "IPV4V6"

若 AT+CGDCONT 無法設置，請先將 AT+CGATT=0

4. 附著(加入)NB 網路

AT+CGATT=1

5. 設定傳送與接收格式為文本格式

▼ AT+QICFG指令

2.3.12. AT+QICFG 配置可选参数

该命令可为 TCP/IP 各项功能配置可选参数。

| AT+QICFG 配置可选参数 | |
|---|---|
| 测试命令 AT+QICFG=? | 响应 +QICFG: "dataformat", (支持的 <send_data_format> 列表), (支持的 <recv_data_format> 列表) +QICFG: "viewmode", (支持的 <view_mode> 列表) +QICFG: "showlength", (支持的 <show_length_mode> 列表) +QICFG: "echomode", (支持的 <echo_mode> 列表) OK |
| 设置命令 设置发送/接收数据的格式 AT+QICFG="dataformat"[,<send_data_format>,<recv_data_format>] | 响应 若缺省参数<send_data_format>和<recv_data_format>, 则查询当前配置: +QICFG: "dataformat",<send_data_format>,<recv_data_format> OK 若不缺省参数<send_data_format>和<recv_data_format>, 则设置发送数据格式和接收数据格式: |

| | |
|---|--|
| | <p>OK</p> <p>若有任何错误: ERROR</p> |
| <p>设置命令</p> <p>设置已接收数据的输出格式</p> <p>AT+QICFG="viewmode"[,<view_mode>]</p> | <p>响应</p> <p>若缺省参数<view_mode>, 则查询当前配置: +QICFG: "viewmode",<view_mode></p> <p>OK</p> <p>若不缺省参数<view_mode>, 则设置已接收数据的输出格式: OK</p> <p>若有任何错误: ERROR</p> |
| <p>设置命令</p> <p>设置在缓存模式下是否显示可选数据长度参数¹⁾</p> <p>AT+QICFG="showlength"[,<show_length_mode>]</p> | <p>响应</p> <p>若缺省参数<show_length_mode>, 则查询当前配置: +QICFG: "showlength",<show_length_mode></p> <p>OK</p> <p>若不缺省参数<show_length_mode>, 则设置在缓存模式下是否显示可选数据长度参数: OK</p> <p>若有任何错误: ERROR</p> |
| <p>设置命令</p> <p>设置在数据模式下是否向 UART 回显输入数据</p> <p>AT+QICFG="echomode"[,<echo_mode>]</p> | <p>响应</p> <p>若缺省参数<echo_mode>, 则查询当前配置: +QICFG: "echomode",<echo_mode></p> <p>OK</p> <p>若不缺省参数<echo_mode>, 则设置回显模式: OK</p> <p>若有任何错误: ERROR</p> |
| 最大响应时间 | 300 毫秒 |
| 保存机制 | <p>1. <send_data_format>、<recv_data_format>、<view_mode>和<show_length_mode>参数配置自动保存至 NVRAM; 深睡眠唤醒后仍有效。</p> <p>2. <echo_mode>参数配置不保存至 NVRAM; 深睡眠唤醒后</p> |

无效。

参数

| | |
|--------------------|--|
| <send_data_format> | 整型。发送数据的格式。 0 文本格式 1 十六进制格式 |
| <recv_data_format> | 整型。接收数据的格式。 0 文本格式 1 十六进制格式 |
| <view_mode> | 整型。已接收数据的输出格式。 0 已接收数据输出格式: data header\r\n data 1 已接收数据输出格式: data header,data |
| <show_length_mode> | 整型。在缓存模式下是否显示可选数据长度参数 ¹⁾ 。 0 不显示 1 显示 |
| <echo_mode> | 整型。是否在数据模式下向 UART 回显输入数据。 0 不回显 1 回显 |

备注

- ¹⁾ 所述可选数据长度参数包含:
 - 下行数据已达 URC +QIURC: "recv",<connectID>[,<current_recv_length>]中的参数<current_recv_length>;
 - AT+QIRD 命令返回值中的参数<remaining_length>。
- 参数<echo_mode>仅在数据模式下才有效, 且仅 BC26NxR01A07/BC20NxR01A08 及之后的版本中才支持该参数。

```
AT+QICFG="dataformat",0,0
```

6. 建立 UDP Socket

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

```
AT+QIOPEN=1,0,"UDP","iiot.niu.edu.tw",18800,0,0,0
```

7. 傳送 UDP 資料

傳送文本資料(其中8為data長度)

```
AT+QISEND=0,8,"B0942103"
```

```
12:09:06.700 -> AT+QISENDEX=0,3,313233
12:09:06.700 -> OK
12:09:06.700 ->
12:09:06.700 -> SEND OK
12:09:07.356 ->
12:09:07.356 -> +QIURC: "recv",0
```

8. 接收 UDP 資料

目標二 (70 %)

| 指令 | 用途 |
|-----------------|---------------------------------------|
| AT+QIRD=0,<Len> | 接收UDP資料的指令格式 Receive data from UDP |

Len → 指定讀取長度 (in bytes)

* 亦可直接設定最大長度512



```
AT+QIRD=0,512
```

作業：(現場檢查)

請傳送所有組員學號到UDP server並自己接收截圖

請截圖伺服器收到訊息與本機接收到訊息兩張截圖，舉例來說，有B0942103與B01234567，則需要傳送兩個UDP資料到 iiot.niu.edu.tw:18800，截圖arduino的序列埠介面、截圖老師電腦中server收到命令，並現場請助教檢查