

# LLD-AIO-003

## NanoPi NEO Module IoT Cap

操作手冊  
版本: V1.3

1	產品簡介.....	1
1.1	硬體規格.....	2
1.2	外接接頭定義.....	3
1.3	通訊及控制介面腳位定義.....	4
1.4	LED 指示燈說明.....	5
1.5	其他定義.....	5
1.6	各通訊及控制介面軟體定義.....	7
2	第一次操作 LLD-AIO-003.....	8
2.1	測試環境建議.....	8
2.2	主控口(Console)簡介.....	10
2.3	出廠設定值.....	10
2.4	操作 console 主控口.....	10
3	LLD-AIO-003 基本操作.....	16
3.1	變更網路設定.....	16
3.2	變更系統時間.....	17
3.3	下載新的 Linux 套件.....	17
3.4	FriendlyARM NanoPi NEO Module 核心模組參考技術資訊.....	19
附錄 A.	主機網路設定.....	20
附錄 B.	轉接線.....	23
附錄 C.	測試主機 COM 接口查詢.....	24



## 1 產品簡介

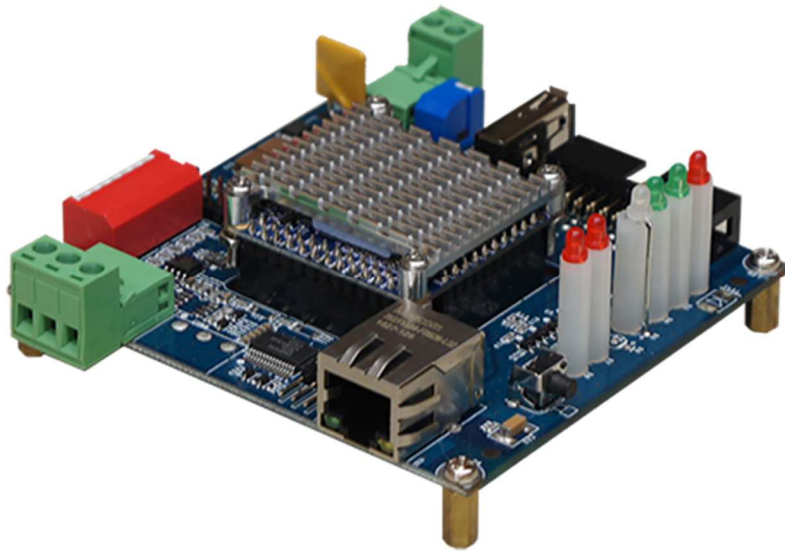
### 產品概述

LLD-AIO-003 是一款針對 FriendlyARM NanoPi NEO 核心模組做為 IoT 物聯網應用的功能載板，主要可應用於現場端設備或儀表的資料採集、即時監控、記錄及與遠端或雲端的數據傳輸。

LLD-AIO-003 搭配的 NanoPi NEO 模組內建的 Linux 作業系統是一種開放式的架構，使用者可以利用公開的 GNU 軟體開發工具，安裝免費的嵌入式系統專用的 C/C++ 程式編譯器(Compiler)及函數庫(Lib)後或更高階的開發語言如 Python、PHP...，進行二次應用程式的開發，使 LLD-AIO-003 成為一款專用的物聯網應用控制器或閘道器(Gateway)。

LLD-AIO-003 具備多元的通訊及控制功能。有乙太網路介面，用來連接網路(LAN 或 WAN)與後台系統連接；具備 RS-485 通訊介面，讓 LLD-AIO-003 可連接更多不同類別的設備或儀表。充份扮演好一個設備監控或資料處理的閘道器角色。LLD-AIO-003 同時也具備數位控制(GPIO)的功能，搭配專用的 Digital I/O 轉換模組與外接模組就可以進行現場端的開關控制。

工業級的設計，無風扇、低功耗，適用於各類長時間不間斷的監控應用。採用金屬外殼，除了增加產品的強度外，壁掛及導軌安裝的設計，讓 LLD-AIO-003 可輕易的安裝使用在各類的場域。RJ45、可插拔的端子(Terminal Block)，都是常見的接頭，種種的設計，都是易於現場施工人員的配線。



## 1.1 硬體規格

### 系統核心

- ▶ Friendly NanoPi NEO Core/Core2 System Module

### 網路介面

- ▶ 數量：1 組
- ▶ 類型：10/100BaseT 乙太網路(Ethernet)
- ▶ 接頭：RJ45

### 數位 I/O(GPIO)

- ▶ 數量：19 點
- ▶ 信號種類：3.3V CMOS
- ▶ 2.54mm 牛角針座 x 12 GPIO
- ▶ DIP Switch x 2 GPIO
- ▶ LED x 4 GPIO
- ▶ Beeper x 1 GPIO
- ▶ 電壓範圍：0 / 3.3VDC

### RS-485 串列埠介面

- ▶ 數量：1 組
- ▶ RS-485 信號：Data+, Data-, GND (支援自動流向控制)
- ▶ Multi-Drop Nodes：128
- ▶ 信號終端電阻：120/600Ω可選 (DIP Switch)
- ▶ 信號拉平電阻：1K/10KΩ可選 (DIP Switch)
- ▶ 保護：15KV ESD 靜電保護, 400W 突波保護
- ▶ 接頭：5.00mm 可插拔端子座

### RS-232 串列埠介面 (保留功能)

- ▶ 數量：1 組
- ▶ 信號：TxD, RxD, GND
- ▶ 保護：15KV ESD 靜電保護, 400W 突波保護
- ▶ 接頭：5.00mm 可插拔端子座

### 串列埠通訊參數

- ▶ Baud Rate：300 ~ 460,800 bps
- ▶ Parity：None, Even, Odd, Mark, Space
- ▶ Data Bits：5, 6, 7, 8
- ▶ Stop Bit：1, 2 bits

### USB 介面

- ▶ 數量：1 組
- ▶ 類型：USB2.0 相容
- ▶ 接頭：Single (Type A)

### 主控台串列埠(Console)

- ▶ 數量：1 組
- ▶ 信號：RS-232 (TxD, RxD, GND)
- ▶ 終端樣式：115,200 bps, VT-100

▷接頭：2.54mm 排針座

### SD 擴充介面

▷數量：1 組(於 NanoPi 模組，需開殼)

▷接頭：Micro SD 插槽

### 機構

▷PCB 尺寸：鍍鋅鋼板 / 92 x 87 x25mm

### 電源

▷工作電壓：DC 9~24VDC

▷電源接頭：5.00mm 端子座

▷功耗：<24W (不含 USB device)

### 其它

▷Real Time Clock : 1 組

▷Beeper : 1 組

▷LED 指示燈：電源，網路，串列埠，控制信號，使用者自定義

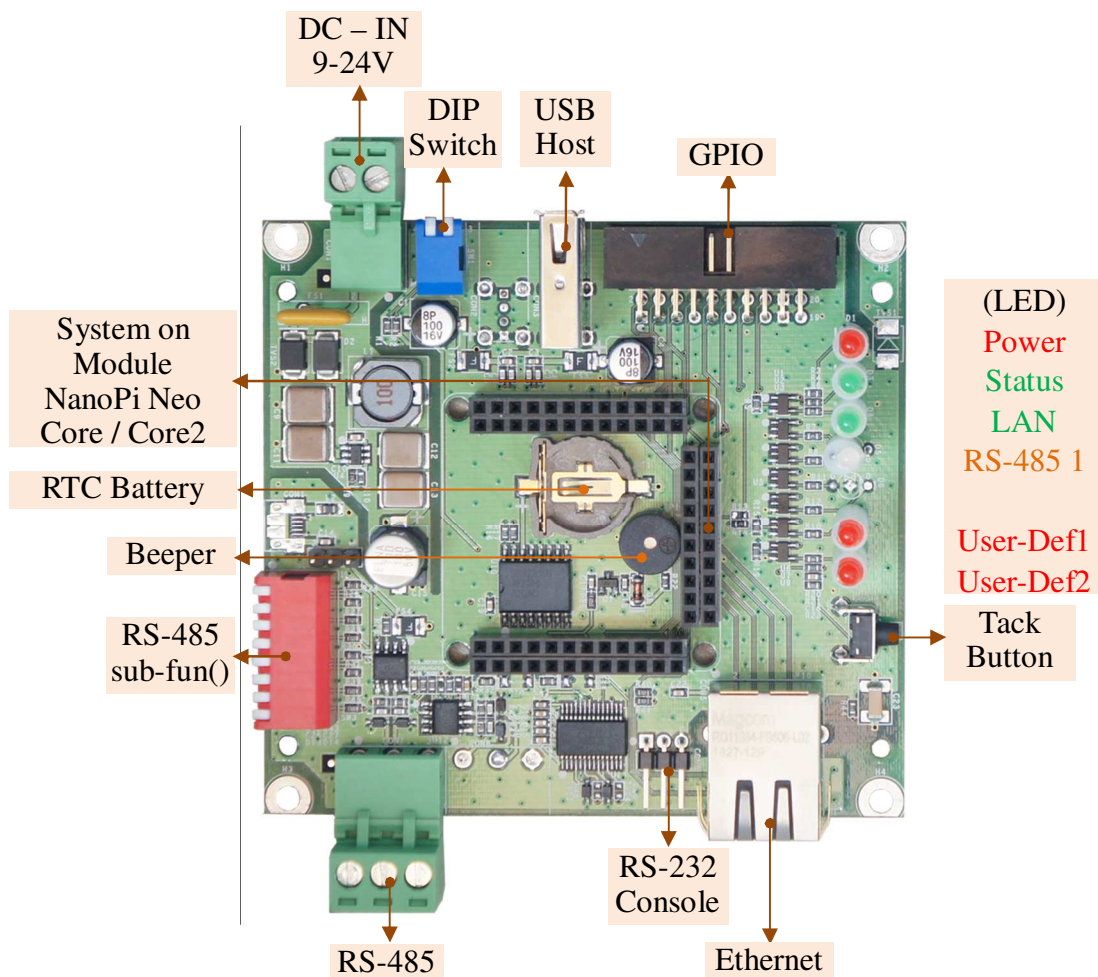
▷適用溫度：-20~70°C

▷適用濕度：20%~80% RHG

▷認證：CE, FCC

## 1.2 外接接頭定義

底板：LLD-AIO-003 Cape



### 1.3 通訊及控制介面腳位定義

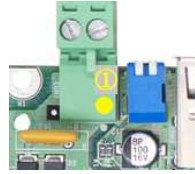
工作電壓輸入

零件編號: CON1

腳位	信號
1	GND
2	DC +

- DC Range: 9~24V DC

5.00mm 端子座



5VDC 電壓輸出(for 外接散熱風扇)

零件編號: J1

腳位	信號
1	5V+
2	GND
3	X

2.54mm 排針座

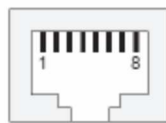


LAN 乙太網路介面(Ethernet)

零件編號: CON5

腳位	信號
1	ETX+
2	ETX-
3	ERX+
6	ERX-

RJ45

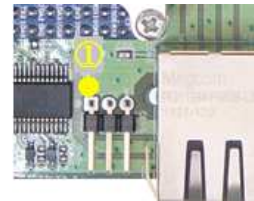


RS-232 Console

零件編號: CON6

腳位	信號
1	TxD
2	RxD
3	GND

2.54mm 排針座

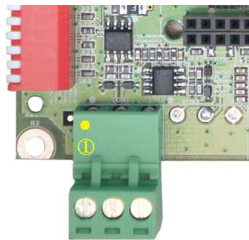


RS-485 (1)

零件編號: CON7

腳位	信號
1	D1+
2	D1-
3	GND

5.00mm 端子座

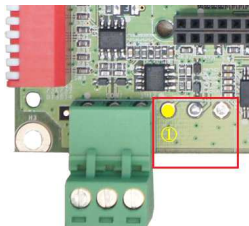


RS-232(保留)

CON8

腳位	信號
1	RxD
2	TxD
3	GND

5.00mm 端子座



## 1.4 LED 指示燈說明

自上到下



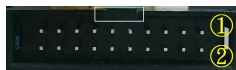
<b>Power</b>	電源指示燈 正確接上工作電壓時就會亮起 (不可透過程式控制 LED)
<b>Status</b>	可做為系統狀態指示燈用 是透過 GPIO 進行控制
<b>Link</b>	網路燈號 燈亮：正確連接網路線 燈閃：網路資料傳輸中 (不可透過程式控制 LED)
<b>RS-485</b>	RS-485 資料傳送燈號 綠燈：資料傳輸中(data out) 紅燈：資料接收中(data in) 橘燈：資料同時傳輸和接收中(data in+out) (不可透過程式控制 LED)
<b>RS-232 (保留功能)</b>	RS-232 資料傳送燈號 綠燈：資料傳輸中(data out) 紅燈：資料接收中(data in) 橘燈：資料同時傳輸和接收中(data in+out) (不可透過程式控制 LED)
<b>U-Def1</b>	可做為應用程式的狀態指示燈用 是透過 GPIO 進行控制
<b>U-Def2</b>	可做為應用程式的狀態指示燈用 是透過 GPIO 進行控制

## 1.5 其他定義

### 1.5.1 GPIO 定義

2.54mm

2x10 牛角排針座



零件編號: CON4

對應核心定義	定義	Pin#	Pin#	定義	對應核心定義
PIO_G08	GPIO-01	01	02	GPIO-02	PIO_G09
PIO_A12	GPIO-03	03	04	GPIO-04	PIO_A11
PIO_C03	GPIO-05	05	06	GPIO-06	PIO_A17
PIO_A07	GPIO-07	07	08	GPIO-08	PIO_L11
PIO_A16	GPIO-09	09	10	GPIO-10	PIO_A15
PIO_A14	GPIO-11	11	12	GPIO-12	PIO_A13
LRCK	I2S0	13	14	I2S0	BCK
DOUT	I2S0	15	16	I2S0	DIN
	GND	17	18	GND	
	X	19	20	X	

- 單雙號腳位”上、下”並排





### 1.5.2 RS-485 電位電阻設定

8-pin  
dip switch

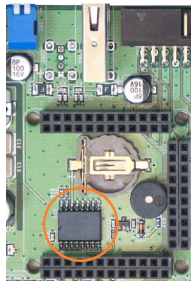


零件編號: SW3

腳位	RS-485	ON 定義	OFF 定義	備註
1	Port 1	data-腳位 1KΩ 電位拉低	data-腳位 10KΩ 電位拉低	腳位 1、2 需同時切換
2		data+腳位 1KΩ 電位拉高	data+腳位 10KΩ 電位拉高	
3		120Ω 終端電阻	無終端電阻	腳位 3、4 不可同時切換
4		600Ω 終端電阻	無終端電阻	
5	保留功能			
6				
7				
8				

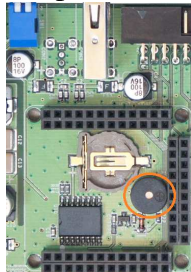
### 1.5.3 其它

RTC



- 以 I2C 控制
- 對應核心定義：I2C\_2
- RTC Battery size：CR1220

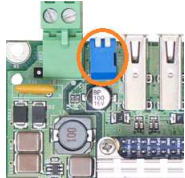
Beeper



- 以 GPIO 控制 (output)
- 對應核心定義：PIO\_G11
- 動作

GPIO status	Beeper status
Low	OFF
High	ON

DIP Switch



- 以 GPIO 控制 (input)
- 對應核心定義：PIO\_A02、PIO\_A03
- 動作

GPIO status	Switch status
Low	ON
High	OFF

Tack Button



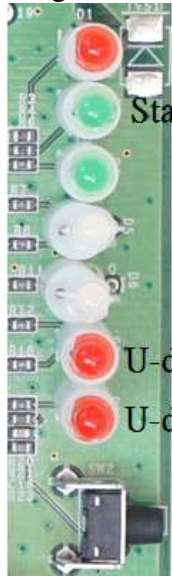
- 以 GPIO 控制 (input)
- 對應核心定義：PIO\_C02
- 動作

GPIO status	button status
Low	Push
High	X





### Programmable LED



### Status LED

- 以 GPIO 控制 (output)
- 對應核心定義：PIO\_A06
- 動作

GPIO status	LED status
Low	Light ON
High	Off

### U-def1 LED

- 以 GPIO 控制 (output)
- 對應核心定義：PIO\_C00
- 動作

GPIO status	LED status
Low	Light ON
High	Off

### U-def2 LED

- 以 GPIO 控制 (output)
- 對應核心定義：PIO\_C01
- 動作

GPIO status	LED status
Low	Light ON
High	Off

## 1.6 各通訊及控制介面軟體定義

介面型態	標示	軟體定義
網路	LAN1	eth0
Console	RS-232	/dev/ttyS0
RS-485	RS-485	/dev/ttyS1
	RS-232 (保留功能)	/dev/ttyS2
外掛記憶體	USB Disk	/dev/sda? (依設備先後插入順序有所不同)
	SD	/dev/mmc0

## 2 第一次操作 LLD-AIO-003

### 2.1 測試環境建議

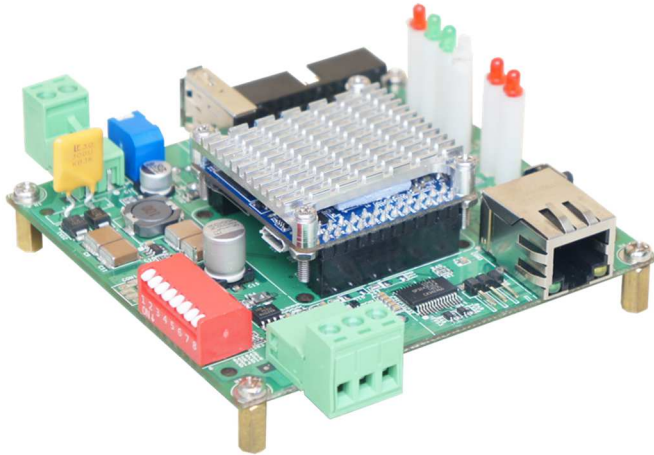
#### 2.1.1. 測試所需器材：

- LLD-AIO-003 一台
- DC 9~24V 電源供應器 一個(輸出功耗需大於>10W)  
(如電源供應器接頭與 LLD-AIO-003 不同,可自製轉接線轉換為一般接線,參考附錄 B. 轉接線參考)
- 網路線或網路交叉線一條
- USB to RS-232 轉接線一條
- LLD-AIO-003 專用 RS-232 Console 線(品名: CB-AIOCON-10, 自製方式請參考附錄 B. 轉接線參考)

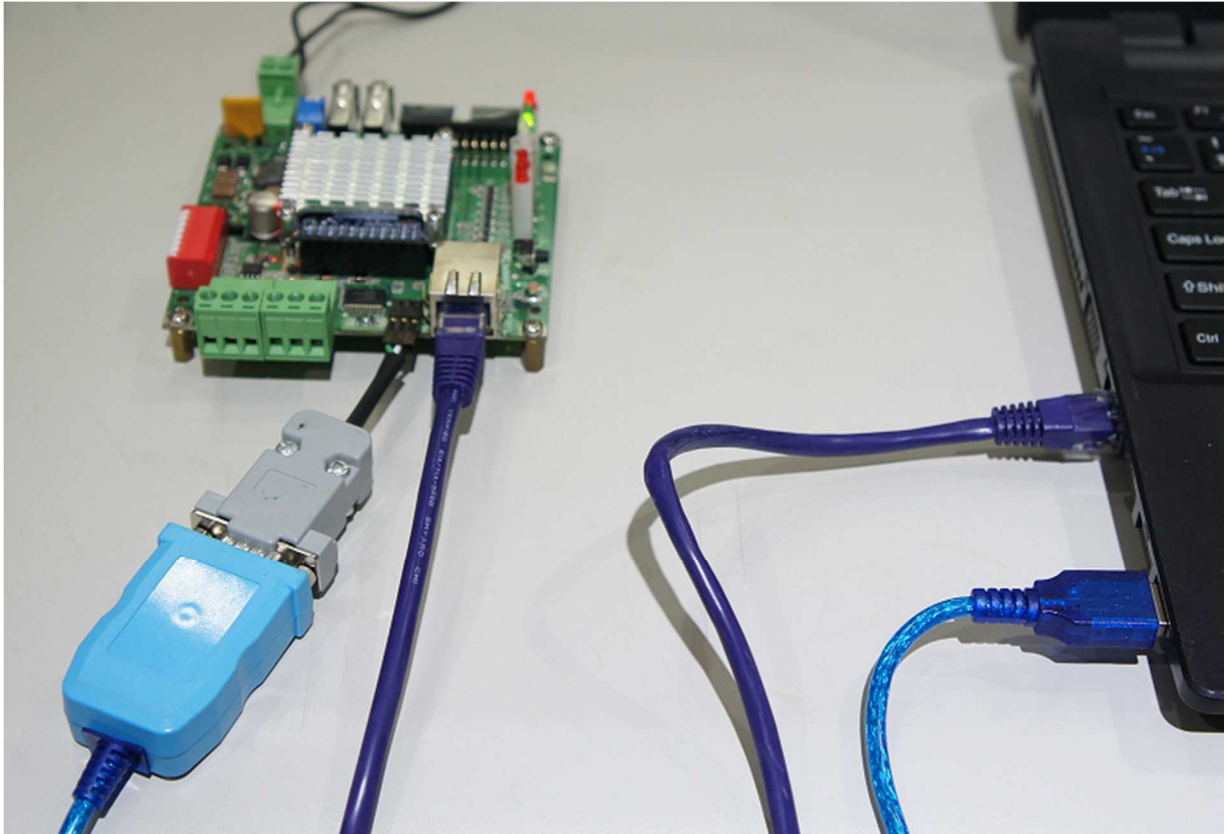


### 2.1.2. 連接安裝

建議先透過銅柱或其他絕緣部件將 LLD-AIO-003 墊高做隔離，以保護在開發及測試過程的產品的安全性



將網路與 RS-232 Console 都與測試主機連接，做為開發主機於 LLD-AIO-003 的通訊橋樑。



## 2.2 主控口(Console)簡介

LLD-AIO-003 出廠具備兩個主控口介面，一個是透過網路，另一個可透過 RS-232 介面，系統開發者可以透過主控口對 LLD-AIO-003 進行完全的控制及開發。除了一般的參數變更外，更可以直接進行系統、程式開發或系統套件的安裝及更新。

### 2.3 出廠設定值

#### A. 登入密碼 (有大小寫之分)

管理者帳戶	Login Name	root
root	Password	fa

#### B. 主控口通訊參數

網路	Ethernet (LAN1)	IP address : DHCP
RS-232	Console	Baud Rate : 115,200 bps Data Format : N-8-1 Flow Control : None Terminal Type : VT-100

## 2.4 操作 console 主控口

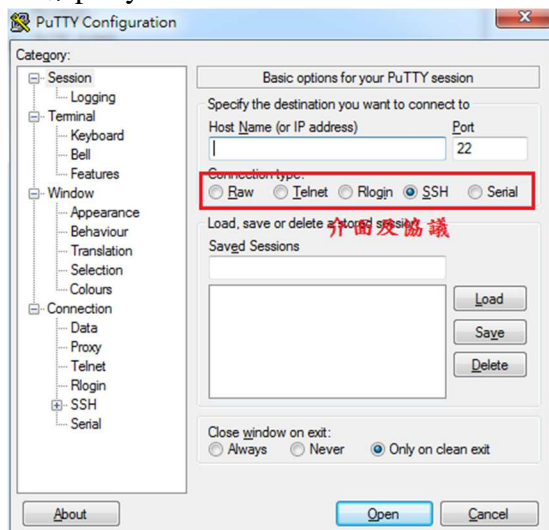
### 2.4.1 console 工具程式 - putty

putty 為 windows 作業系統中用來進行遠端通訊的一套工具軟體，常用連接於有主控接口的開發平台，如 LLD-AIO-003

#### ■ 下載 putty

<https://putty.org>

#### ■ 啟動 putty



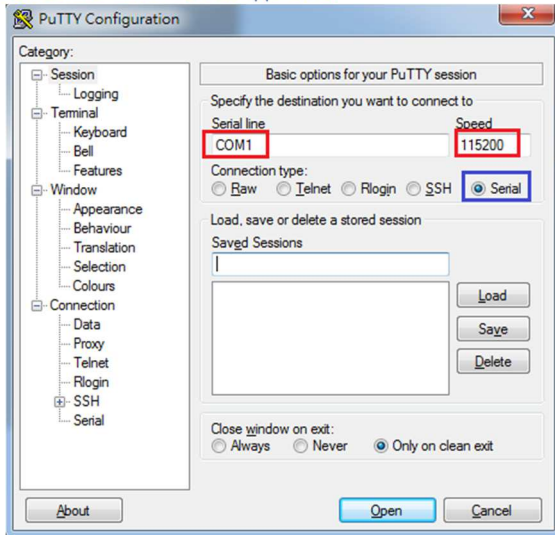
### 2.4.2 進入 RS-232 Console

選擇 “Serial” 通訊介面

輸入 “COM1” (註\*)及選擇通訊速度 ”115200”

註\* : COM# 為主機 RS-232 接口，統稱 COM 口的編號，透過 USB 轉接線擴充的編號需自主機系統中查詢。(可參考 “附錄 C.”)



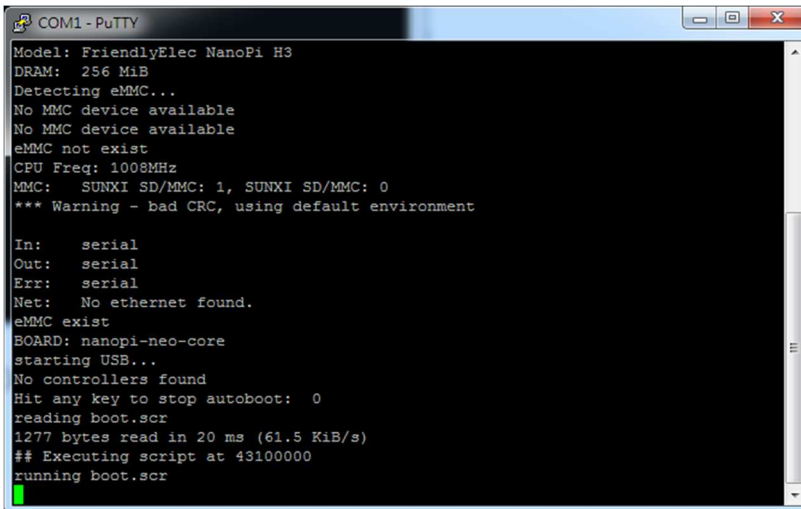


■ 啟動 putty



■ LLD-AIO-003 (通電)

顯示 Junminx-AIO-003 開機的信息



```
COM1 - PuTTY
[ OK ] Started Hostname Service.
[ OK ] Started Network Manager Wait Online.
[ OK ] Reached target Network is Online.
Starting /etc/rc.local Compatibility...
Starting LSB: Start NTP daemon...
Starting LSB: disk temperature monitoring daemon...
[ 8.944321] rc.local[534]: ctp = 0
[ OK ] Started /etc/rc.local Compatibility.
[ OK ] Started LSB: disk temperature monitoring daemon.
[ OK ] Started Getty on tty1.
[ OK ] Started Serial Getty on ttyS0.
[ OK ] Reached target Login Prompts.
[ OK ] Started LSB: Start NTP daemon.
[ OK ] Reached target Multi-User System.
[ OK ] Reached target Graphical Interface.
Starting Update UTMP about System Runlevel Changes...
[ OK ] Started Update UTMP about System Runlevel Changes.

Ubuntu 16.04.2 LTS NanoPi-NEO-Core ttyS0
NanoPi-NEO-Core login: pi (automatic login)
Last login: Thu Feb 11 16:28:05 UTC 2016 on tty1
```

### LLD-AIO-003 系統啟動完成

```
COM1 - PuTTY
[ OK ] Started Update UTMP about System Runlevel Changes.

Ubuntu 16.04.2 LTS NanoPi-NEO-Core ttyS0
NanoPi-NEO-Core login: pi (automatic login)
Last login: Thu Feb 11 16:28:05 UTC 2016 on tty1

          _ _ _ _ _
         / / / / /
        / / / / /
       / / / / /
      / / / / /
     / / / / /
    / / / / /
   / / / / /
  / / / / /
 / / / / /
/ / / / /

Welcome to Ubuntu 16.04.2 LTS 4.14.111
System load:  0.16           Up time:      10 sec
Memory usage: 15 % of 238Mb IP:
CPU temp:    22.2°C
Usage of /:   58% of 1.1G

* Documentation: http://wiki.friendlyarm.com/Ubuntu
* Forum: http://www.friendlyarm.com/Forum/

pi@NanoPi-NEO-Core:~$
```

註：出廠模式，透過 RS-232 Console 連接進入 LLD-AIO-003 無需做密碼登入

### 2.4.3 進入 Ethernet console

因 LLD-AIO-003 的出廠網路設定為 DHCP mode，所以需將 LLD-AIO-003 連接到有 DHCP server 的局網(LAN)中，才可以分配到有效的網路 IP address，也才能夠進行 Ethernet console 的接入。

- 透過 RS-232 console 查看網路設定

可透過指令 ifconfig

- ◆ LLD-AIO-003 未連接到有 DHCP Server 的網路

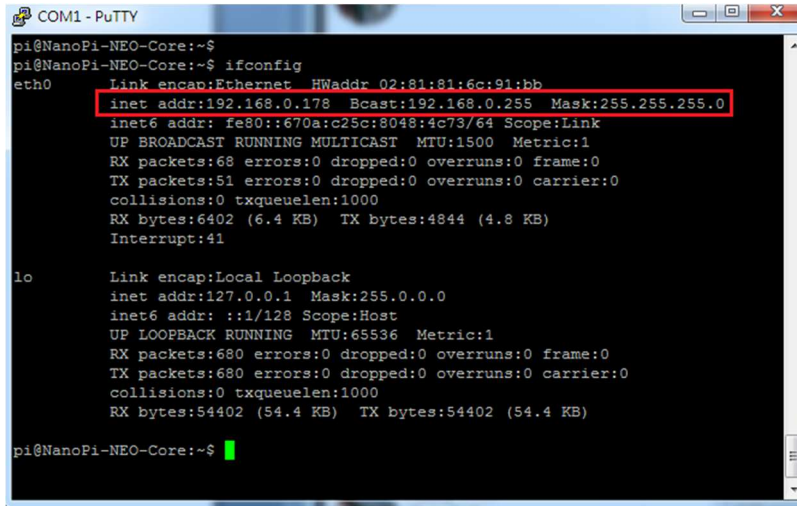
```
COM1 - PuTTY
pi@NanoPi-NEO-Core:~$
pi@NanoPi-NEO-Core:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 02:81:81:6c:91:bb
          UP BROADCAST MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)
          Interrupt:41

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:560 errors:0 dropped:0 overruns:0 frame:0
          TX packets:560 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:44576 (44.5 KB)  TX bytes:44576 (44.5 KB)

pi@NanoPi-NEO-Core:~$
```



◆ LLD-AIO-003 連接到有 SHCP Server 的網路  
取得有效 IP address : 192.168.0.178

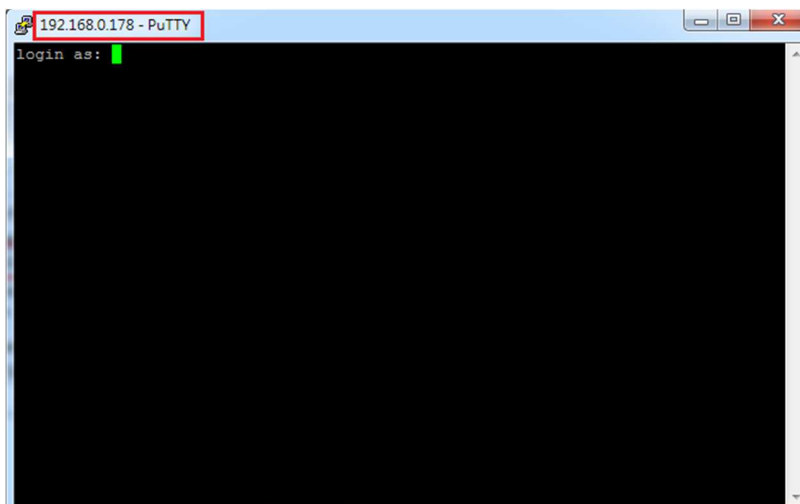
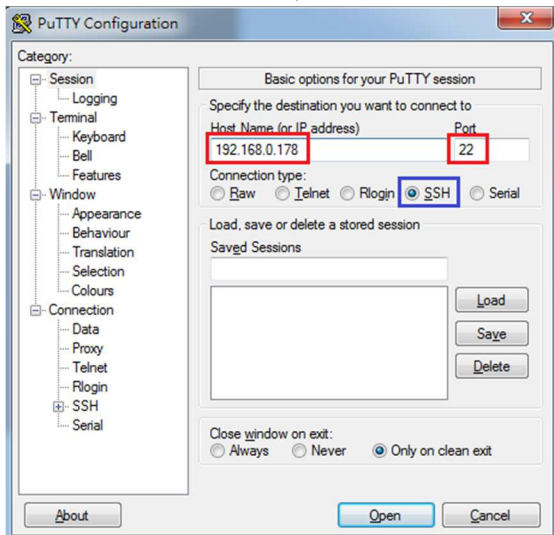


```
pi@NanoPi-NEO-Core:~$  
pi@NanoPi-NEO-Core:~$ ifconfig  
eth0      Link encap:Ethernet  HWaddr 02:81:81:6c:91:bb  
          inet addr:192.168.0.178  Bcast:192.168.0.255  Mask:255.255.0  
          inet6 addr: fe80::670a:c25c:8048:4c73/64 Scope:Link  
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1  
          RX packets:68  errors:0  dropped:0  overruns:0  frame:0  
          TX packets:51  errors:0  dropped:0  overruns:0  carrier:0  
          collisions:0 txqueuelen:1000  
          RX bytes:6402 (6.4 KB)  TX bytes:4844 (4.8 KB)  
          Interrupt:41  
  
lo        Link encap:Local Loopback  
          inet addr:127.0.0.1  Mask:255.0.0.0  
          inet6 addr: ::1/128 Scope:Host  
          UP LOOPBACK RUNNING  MTU:65536  Metric:1  
          RX packets:680  errors:0  dropped:0  overruns:0  frame:0  
          TX packets:680  errors:0  dropped:0  overruns:0  carrier:0  
          collisions:0 txqueuelen:1000  
          RX bytes:54402 (54.4 KB)  TX bytes:54402 (54.4 KB)  
  
pi@NanoPi-NEO-Core:~$
```

■ 啟動 putty

選擇 “SSH”通訊介面

輸入 “192.168.0.178 (接上續、所讀得的 LLD-AIO-003 IP address)”

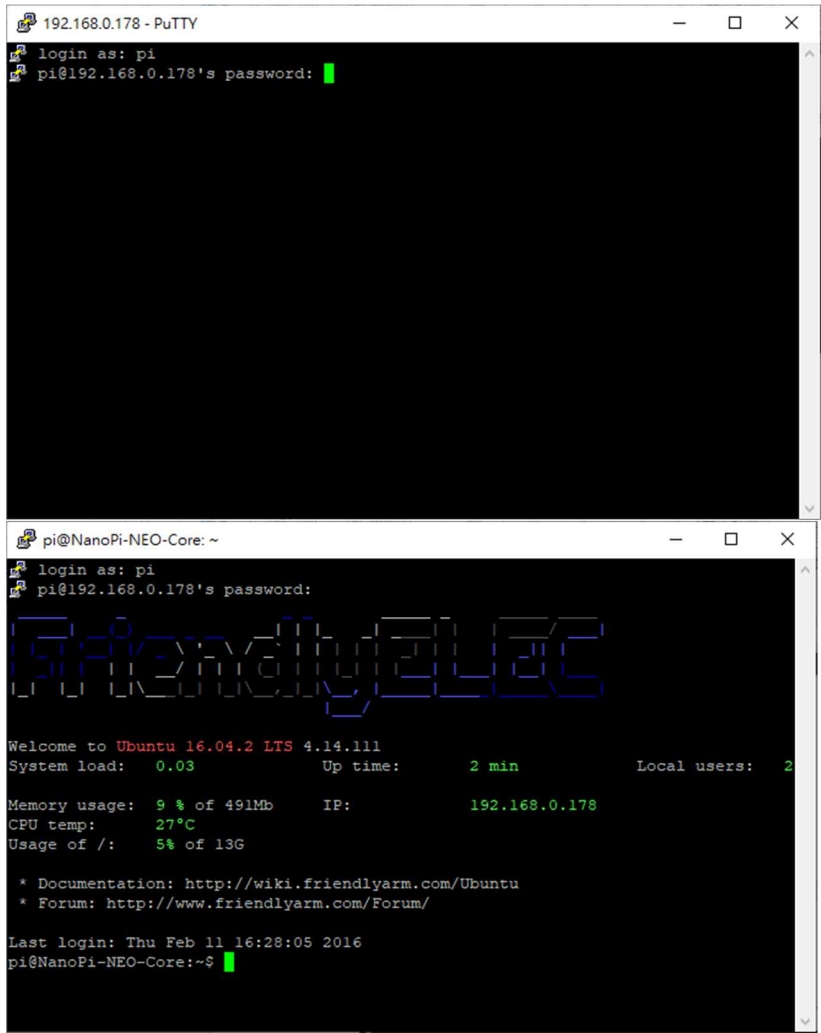


■ 進入 LLD-AIO-003 Ethernet console

輸入 LLD-AIO-003 出廠的一般用戶名及密碼

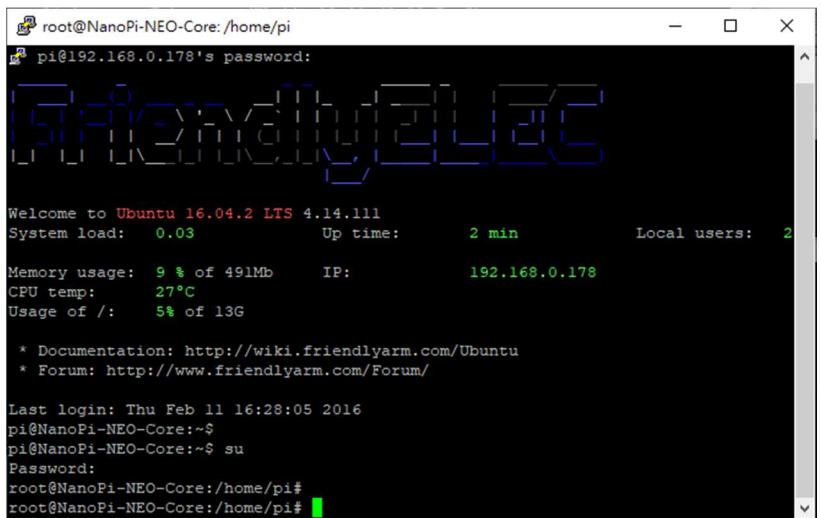
Login Name	pi
------------	----

Password	pi
----------	----

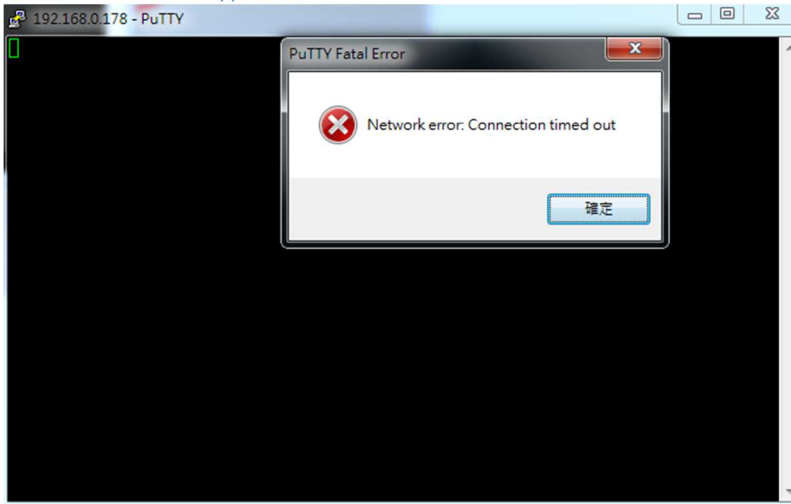


如需升級為管理層級，需執行指令”su”，並輸入 root 的密碼  
輸入 LLD-AIO-003 出廠的一般用戶名及密碼

Password	fa
----------	----



- 無法連入 LLD-AIO-003 Ethernet console



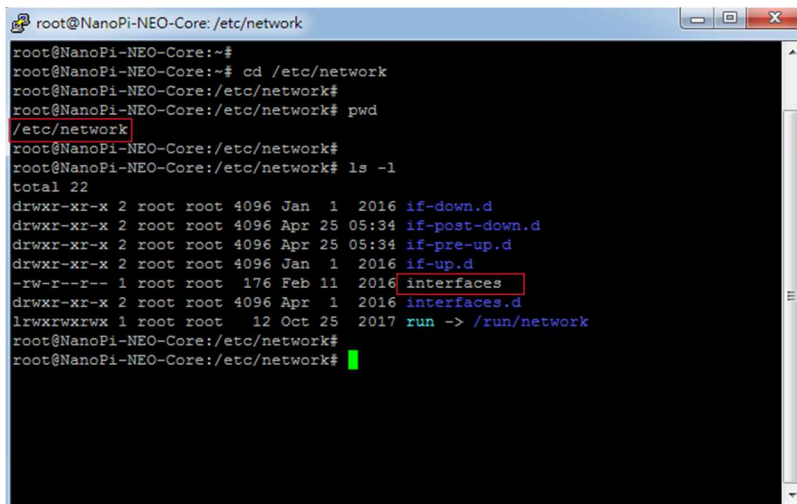
此狀況主要發生的原因常見有 3:

1. 輸入錯誤的 LLD-AIO-003 的 IP address
2. LLD-AIO-003 的網路有問題
  - LLD-AIO-003 的網路線沒接好
  - 不知 LLD-AIO-003 的網路 IP address 設定
3. LLD-AIO-003 和測試主機的網路設定在不同的網段  
需查看主機的網路設定是否和 LLD-AIO-003 在同一 IP 網段，可參考 "附錄 A"

### 3 LLD-AIO-003 基本操作

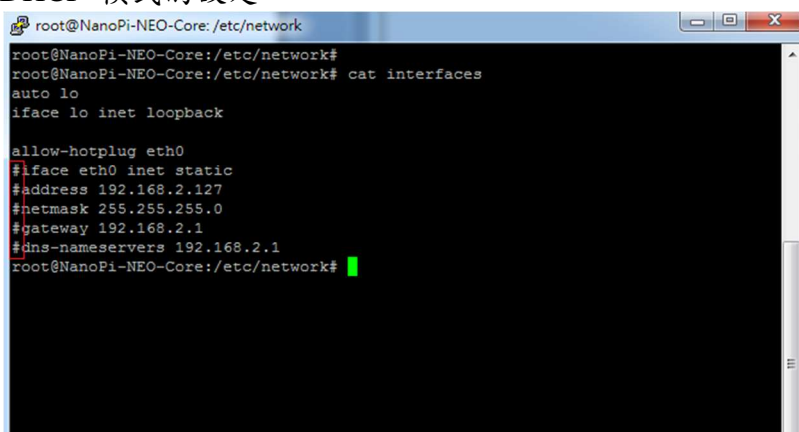
#### 3.1 變更網路設定

LLD-AIO-003 的出廠設定是 DHCP mode，所連接的網路上需有一台 DHCP Server 功能的主機分配 IP address 給 LLD-AIO-003，如要將 LLD-AIO-003 變更為自定義的固定 IP 模式，需變更一系統設定檔 “interfaces”，所在路徑: "/etc/network"



```
root@NanoPi-NEO-Core: /etc/network
root@NanoPi-NEO-Core:~#
root@NanoPi-NEO-Core:~# cd /etc/network
root@NanoPi-NEO-Core:/etc/network#
root@NanoPi-NEO-Core:/etc/network# pwd
/etc/network
root@NanoPi-NEO-Core:/etc/network#
root@NanoPi-NEO-Core:/etc/network# ls -l
total 22
drwxr-xr-x 2 root root 4096 Jan  1 2016 if-down.d
drwxr-xr-x 2 root root 4096 Apr 25 05:34 if-post-down.d
drwxr-xr-x 2 root root 4096 Apr 25 05:34 if-pre-up.d
drwxr-xr-x 2 root root 4096 Jan  1 2016 if-up.d
-rw-r--r-- 1 root root 176 Feb 11 2016 interfaces
drwxr-xr-x 2 root root 4096 Apr  1 2016 interfaces.d
lrwxrwxrwx 1 root root 12 Oct 25 2017 run -> /run/network
root@NanoPi-NEO-Core:/etc/network#
root@NanoPi-NEO-Core:/etc/network#
```

#### ■ DHCP 模式的設定



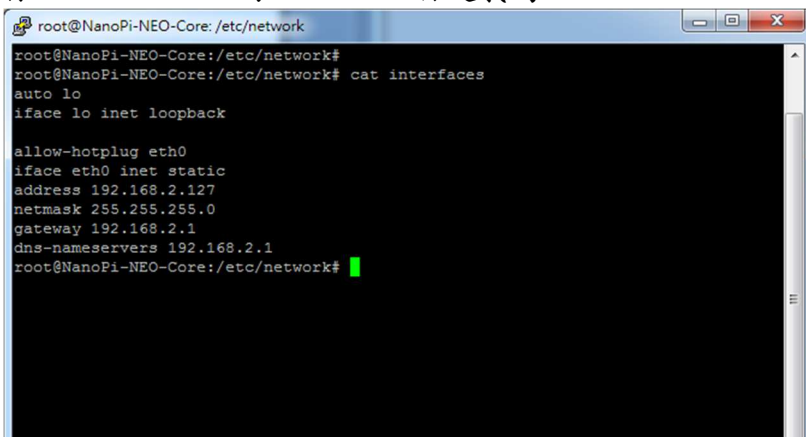
```
root@NanoPi-NEO-Core: /etc/network
root@NanoPi-NEO-Core:/etc/network#
root@NanoPi-NEO-Core:/etc/network# cat interfaces
auto lo
iface lo inet loopback

allow-hotplug eth0
#iface eth0 inet static
#address 192.168.2.127
#netmask 255.255.255.0
#gateway 192.168.2.1
#dns-nameservers 192.168.2.1
root@NanoPi-NEO-Core:/etc/network#
```

設定命令列前端多了 “#”，此命令列就不執行

#### ■ 固定 IP Address 的設定

將 LLD-AIO-003 的 IP address 自定義為: 192.168.2.127




```
root@NanoPi-NEO-Core: /etc/network
root@NanoPi-NEO-Core:/etc/network#
root@NanoPi-NEO-Core:/etc/network# cat interfaces
auto lo
iface lo inet loopback

allow-hotplug eth0
iface eth0 inet static
address 192.168.2.127
netmask 255.255.255.0
gateway 192.168.2.1
dns-nameservers 192.168.2.1
root@NanoPi-NEO-Core:/etc/network#
```

### 3.2 變更系統時間

LLD-AIO-003 正常運行時，會有兩個時間同時運行，一個是系統時間，可透過 “date” 這個指令查看或變更，另一個是 HW RTC(實時時鐘)的時間，可透過 ”hwclock” 這指令查看及變更。

系統時間易受元件老化、運行的環境溫度等等影響，較容易發生時間不準的狀況，通常可透過網路校時，如 NTP 協議或命令，另外就是透過與 LLD-AIO-003 本機的 HW RTC 進行時間同步。

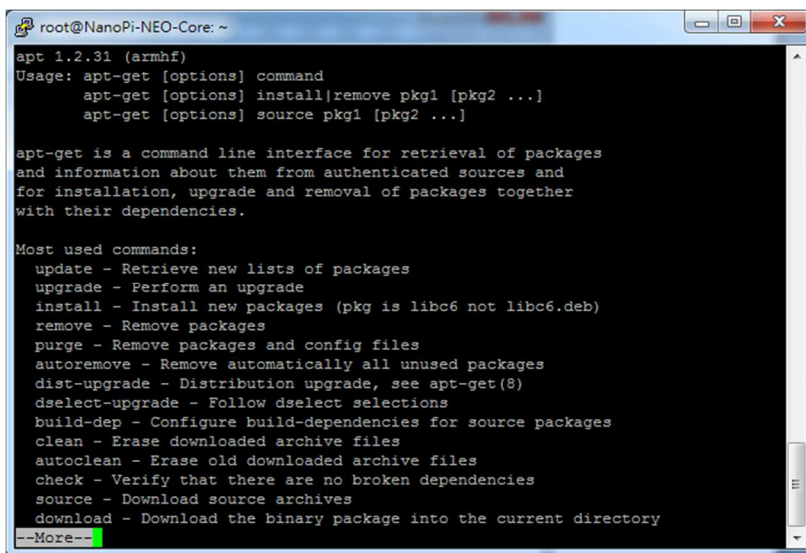


```
root@NanoPi-NEO-Core: ~  
root@NanoPi-NEO-Core:~#  
root@NanoPi-NEO-Core:~# date 查看系統時間  
Wed Aug 14 20:01:25 UTC 2019  
root@NanoPi-NEO-Core:~#  
root@NanoPi-NEO-Core:~# date 081421002019 變更系統時間  
Wed Aug 14 21:00:00 UTC 2019  
root@NanoPi-NEO-Core:~#  
root@NanoPi-NEO-Core:~# hwclock -s 自HW RTC讀回時間成系統時間  
root@NanoPi-NEO-Core:~#  
root@NanoPi-NEO-Core:~# date 驗證HW RTC取回的時間  
Wed Aug 14 20:02:02 UTC 2019  
root@NanoPi-NEO-Core:~#  
root@NanoPi-NEO-Core:~#
```

### 3.3 下載新的 Linux 套件

LLD-AIO-003 目前搭載的是 Linux 作業系統，開發或使用 LLD-AIO-003 可能需要安裝其它的開發套件或應用服務。

目前最常使用的方式，就是將 LLD-AIO-003 安裝在有連接到 Internet 的網路環境，透過”apt-get” 這個安裝套件的工具程式進行軟體安裝。



```
root@NanoPi-NEO-Core: ~  
apt 1.2.31 (armhf)  
Usage: apt-get [options] command  
       apt-get [options] install|remove pkg1 [pkg2 ...]  
       apt-get [options] source pkg1 [pkg2 ...]  
  
apt-get is a command line interface for retrieval of packages  
and information about them from authenticated sources and  
for installation, upgrade and removal of packages together  
with their dependencies.  
  
Most used commands:  
update - Retrieve new lists of packages  
upgrade - Perform an upgrade  
install - Install new packages (pkg is libc6 not libc6.deb)  
remove - Remove packages  
purge - Remove packages and config files  
autoremove - Remove automatically all unused packages  
dist-upgrade - Distribution upgrade, see apt-get(8)  
dselect-upgrade - Follow dselect selections  
build-dep - Configure build-dependencies for source packages  
clean - Erase downloaded archive files  
autoclean - Erase old downloaded archive files  
check - Verify that there are no broken dependencies  
source - Download source archives  
download - Download the binary package into the current directory  
--More--
```

- 例：安裝 Apache Web Server  
執行 apt-get install apache2

```
root@NanoPi-NEO-Core:~  
root@NanoPi-NEO-Core:~#  
root@NanoPi-NEO-Core:~# apt-get install apache2
```

安裝過程中，或許會有些提問要做確認

```
root@NanoPi-NEO-Core:~  
root@NanoPi-NEO-Core:~#  
root@NanoPi-NEO-Core:~# apt-get install apache2  
Reading package lists... Done  
Building dependency tree  
Reading state information... Done  
The following additional packages will be installed:  
  apache2-bin apache2-data apache2-utils libapr1 libaprutil1  
  libaprutil1-dbd-sqlite3 libaprutil1-ldap liblua5.1-0 ssl-cert  
Suggested packages:  
  www-browser apache2-doc apache2-suexec-pristine | apache2-suexec-custom ufw  
  openssl-blacklist  
The following NEW packages will be installed:  
  apache2 apache2-bin apache2-data apache2-utils libapr1 libaprutil1  
  libaprutil1-dbd-sqlite3 libaprutil1-ldap liblua5.1-0 ssl-cert  
0 upgraded, 10 newly installed, 0 to remove and 231 not upgraded.  
Need to get 1,411 kB of archives.  
After this operation, 4,974 kB of additional disk space will be used.  
Do you want to continue? [Y/n] y
```

不同的套件，安裝需要的時間不一樣

```
root@NanoPi-NEO-Core:~  
Setting up apache2-utils (2.4.18-2ubuntu3.10) ...  
Setting up apache2-data (2.4.18-2ubuntu3.10) ...  
Setting up apache2 (2.4.18-2ubuntu3.10) ...  
Enabling module mpm_event.  
Enabling module authz_core.  
Enabling module authz_host.  
Enabling module authn_core.  
Enabling module auth_basic.  
Enabling module access_compat.  
Enabling module authn_file.  
Enabling module authz_user.  
Enabling module alias.  
Enabling module dir.  
Enabling module autoindex.  
Enabling module env.  
Enabling module mime.  
Enabling module negotiation.  
Enabling module setenvif.  
Enabling module filter.  
Enabling module deflate.  
Enabling module status.  
Enabling conf charset.  
Enabling conf localized-error-pages.  
Enabling conf other-vhosts-access-log.  
Enabling conf security.
```



```

root@NanoPi-NEO-Core: ~
Enabling module authz_host.
Enabling module authn_core.
Enabling module auth_basic.
Enabling module access_compat.
Enabling module authn_file.
Enabling module authz_user.
Enabling module alias.
Enabling module dir.
Enabling module autoindex.
Enabling module env.
Enabling module mime.
Enabling module negotiation.
Enabling module setenvif.
Enabling module filter.
Enabling module deflate.
Enabling module status.
Enabling conf charset.
Enabling conf localized-error-pages.
Enabling conf other-vhosts-access-log.
Enabling conf security.
Enabling conf serve-cgi-bin.
Enabling site 000-default.
Setting up ssl-cert (1.0.37) ...
Processing triggers for libc-bin (2.23-0ubuntu7) ...
Processing triggers for systemd (229-4ubuntu16) ...
root@NanoPi-NEO-Core:~#

```

確認 Apache 已安裝，並且已運行

```

root@NanoPi-NEO-Core: ~
pi          733    554    0 19:29 tty1      00:00:00 -bash
pi          734    551    0 19:29 ttyS0     00:00:00 -bash
nobody     758    468    0 19:29 ?           00:00:00 /usr/sbin/dnsmasq --no-resolv --
root       848      2    0 19:30 ?           00:00:00 [kworker/0:1H]
root      1529      2    0 19:37 ?           00:00:00 [kworker/u8:1]
root      1556      2    0 19:37 ?           00:00:00 [kworker/2:0]
root      1559    519    0 19:37 ?           00:00:00 sshd: root@pts/0
root      1569      1    0 19:37 ?           00:00:00 /lib/systemd/systemd --user
root      1572    1569    0 19:37 ?           00:00:00 (sd-pam)
root      1652    1559    0 19:37 pts/0       00:00:00 -bash
root      1708      2    0 19:44 ?           00:00:00 [kworker/0:0]
root      1786    519    0 19:45 ?           00:00:00 sshd: root@pts/1
root      1888   1786    0 19:45 pts/1       00:00:00 -bash
root      2081   1888    0 19:58 pts/1       00:00:00 su
root      2086   2081    0 19:58 pts/1       00:00:00 bash
root      2192      2    0 20:02 ?           00:00:01 [kworker/0:1]
root      2219      2    0 20:14 ?           00:00:00 [kworker/u8:0]
root      2438      2    0 20:19 ?           00:00:00 [kworker/3:0]
root      2451      2    0 20:21 ?           00:00:00 [kworker/1:1]
root      2461      2    0 20:23 ?           00:00:00 [kworker/u8:2]
root      2478      2    0 20:26 ?           00:00:00 [kworker/1:0]
root      2895      1    0 20:26 ?           00:00:00 /usr/sbin/apache2 -k start
www-data  2898   2895    0 20:26 ?           00:00:00 /usr/sbin/apache2 -k start
www-data  2899   2895    0 20:26 ?           00:00:00 /usr/sbin/apache2 -k start
root      3019   2086    0 20:28 pts/1       00:00:00 ps -ef
root@NanoPi-NEO-Core:~#

```

### 3.4 FriendlyARM NanoPi NEO Module 核心模組參考技術資訊

如需更詳細的系統核心模組的相關資訊，可至系統核心模組的產品官網進行查閱

NanoPi NEO Core : [http://wiki.friendlyarm.com/wiki/index.php/NanoPi\\_NEO\\_Core](http://wiki.friendlyarm.com/wiki/index.php/NanoPi_NEO_Core)

NanoPi NEO Core2 : [http://wiki.friendlyarm.com/wiki/index.php/NanoPi\\_NEO\\_Core2](http://wiki.friendlyarm.com/wiki/index.php/NanoPi_NEO_Core2)

## 附錄 A. 主機網路設定

### WINDOWS 7 的環境設定

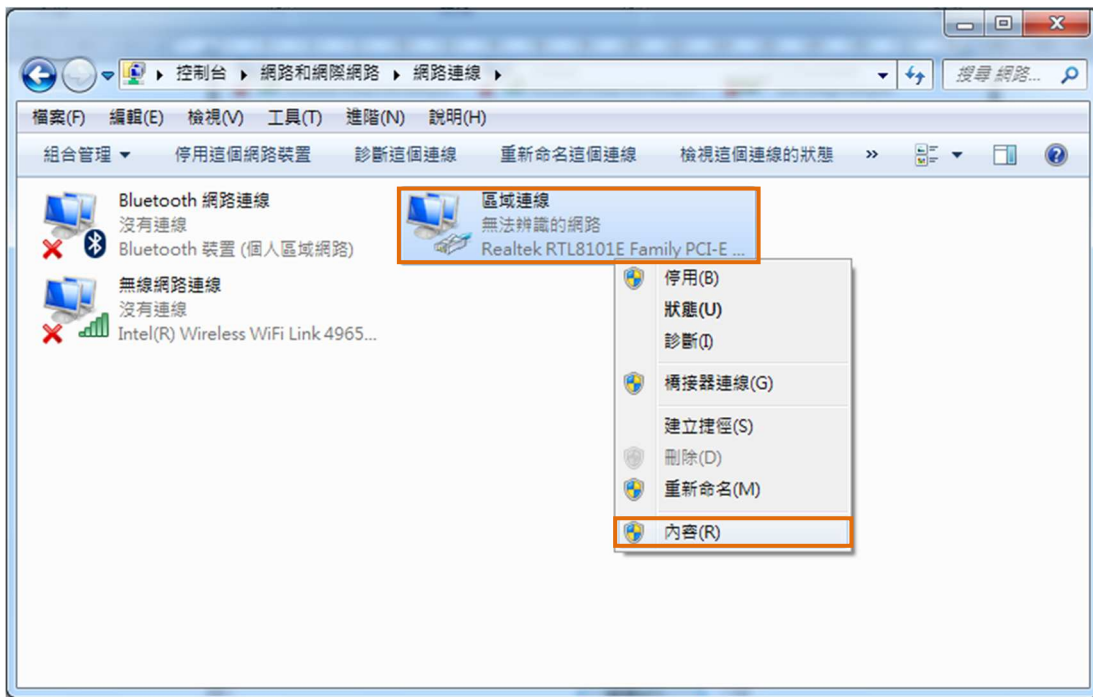
Step.01 開始 → 控制台 → 網路和網際網路-檢視網路狀態及工作。



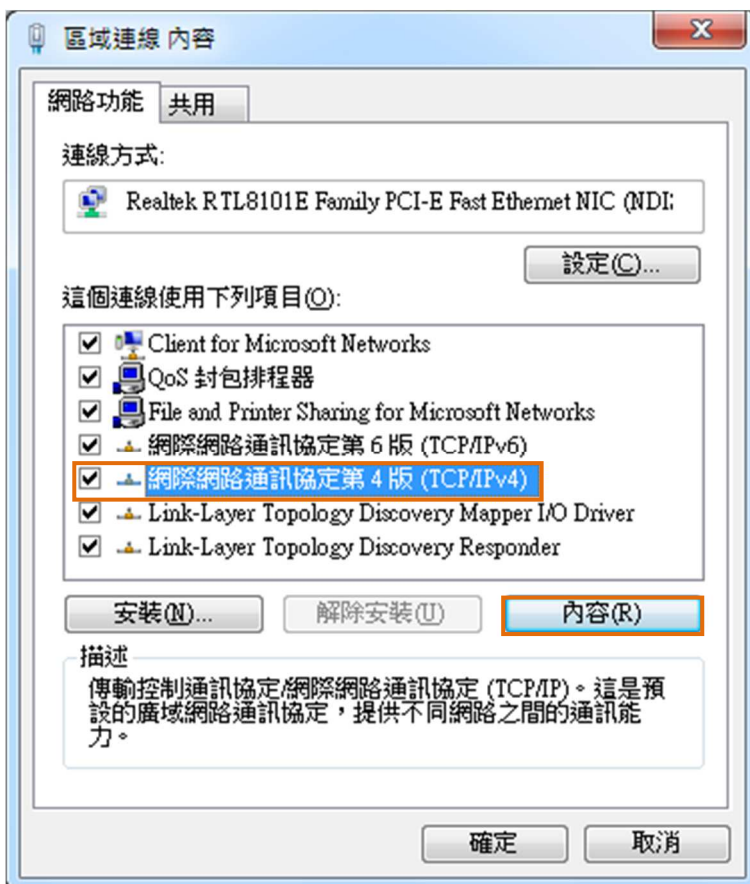
Step.02 變更介面卡設定。



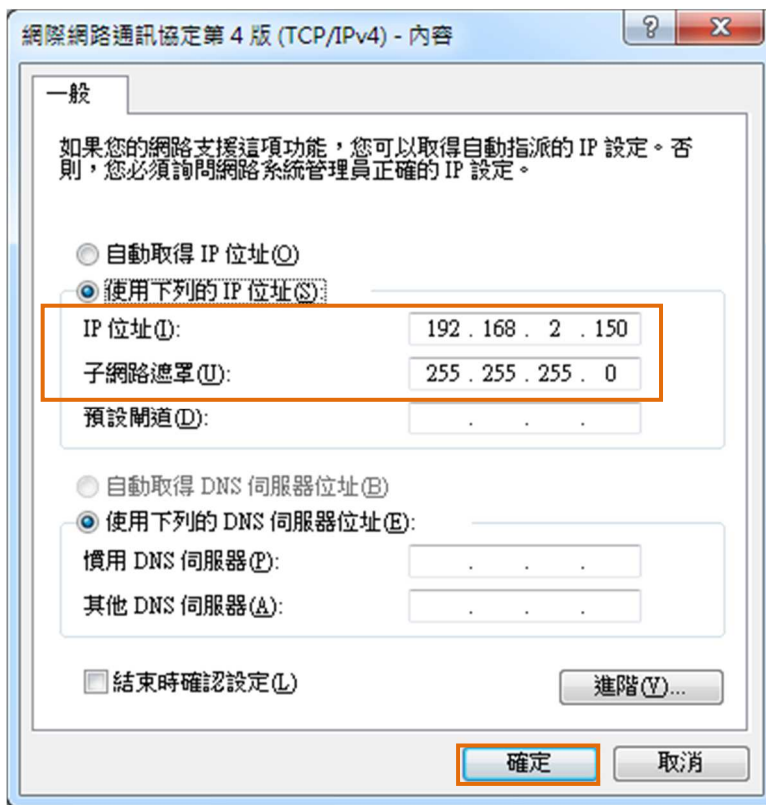
Step.03 區域連線 →按右鍵 內容。



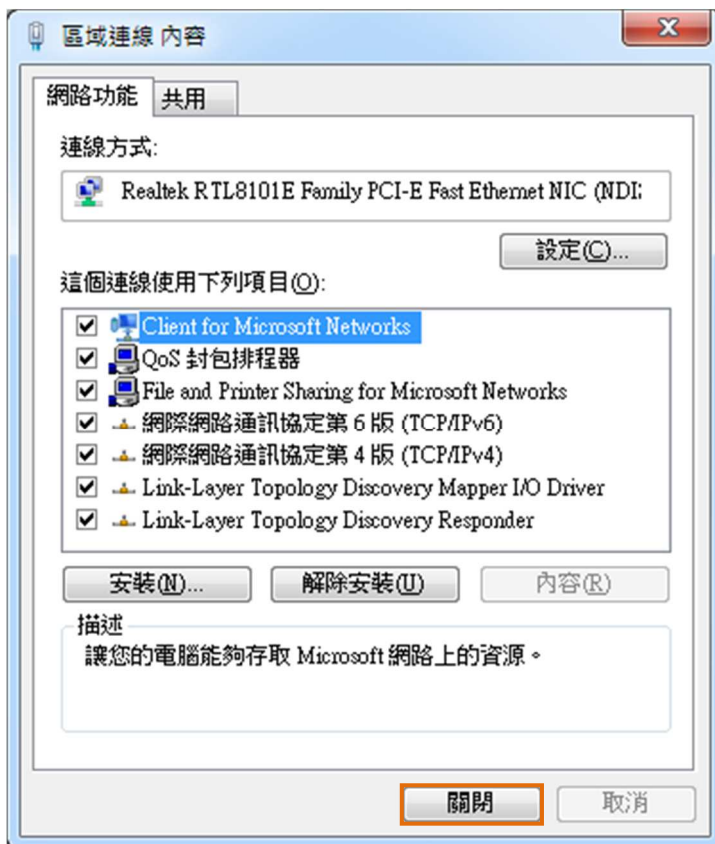
Step.04 網際網路通訊協定第 4 版(TCP/IPv4) → 內容。



Step.05 IP 位址(I) : 192.168.2.150 ; 子網路遮罩(U) : 255.255.255.0 → 確定鍵。



Step.06 設定完畢點擊關閉鍵。



## 附錄 B. 轉接線

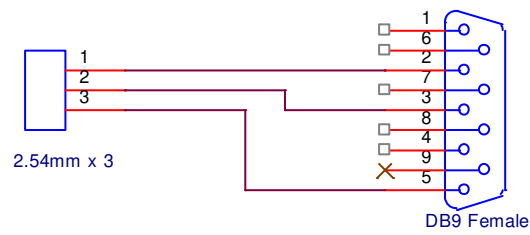
### B1. Console 線

3-pin 2.54mm 排針座轉換 DB9 母頭



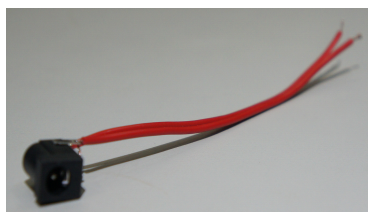
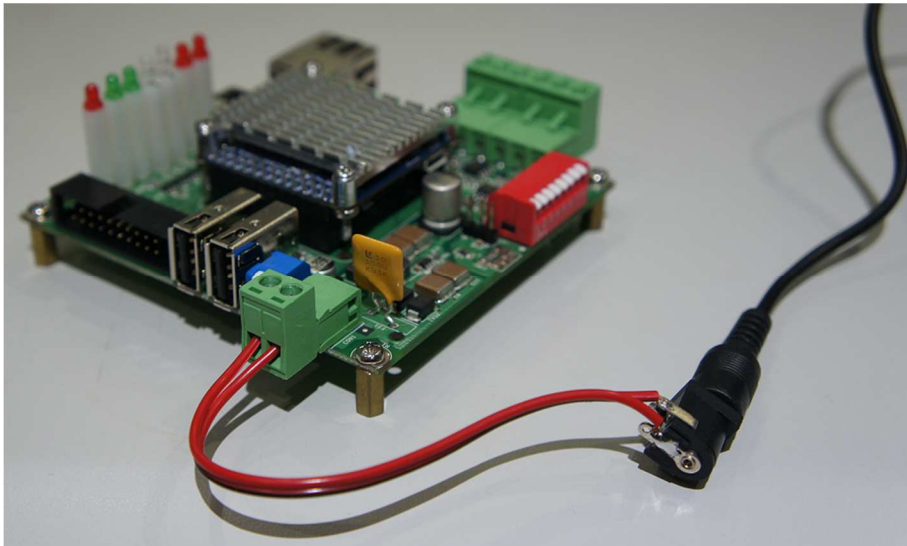
接腳定義

3-PIN HEADER 腳位編號	信號定義	DB9 母 腳位編號
1	TXD	2
2	RXD	3
3	GND	5



### B2. 電源轉接線

DC Power Jack 轉 Terminal Block



內徑  $\Phi 2.1\text{mm}$  powerjack



### 附錄 C. 測試主機 COM 接口查詢

