



CodeZoo CATM1 Hands-On

- Jetson Nano (with PPP)-

version 1.0

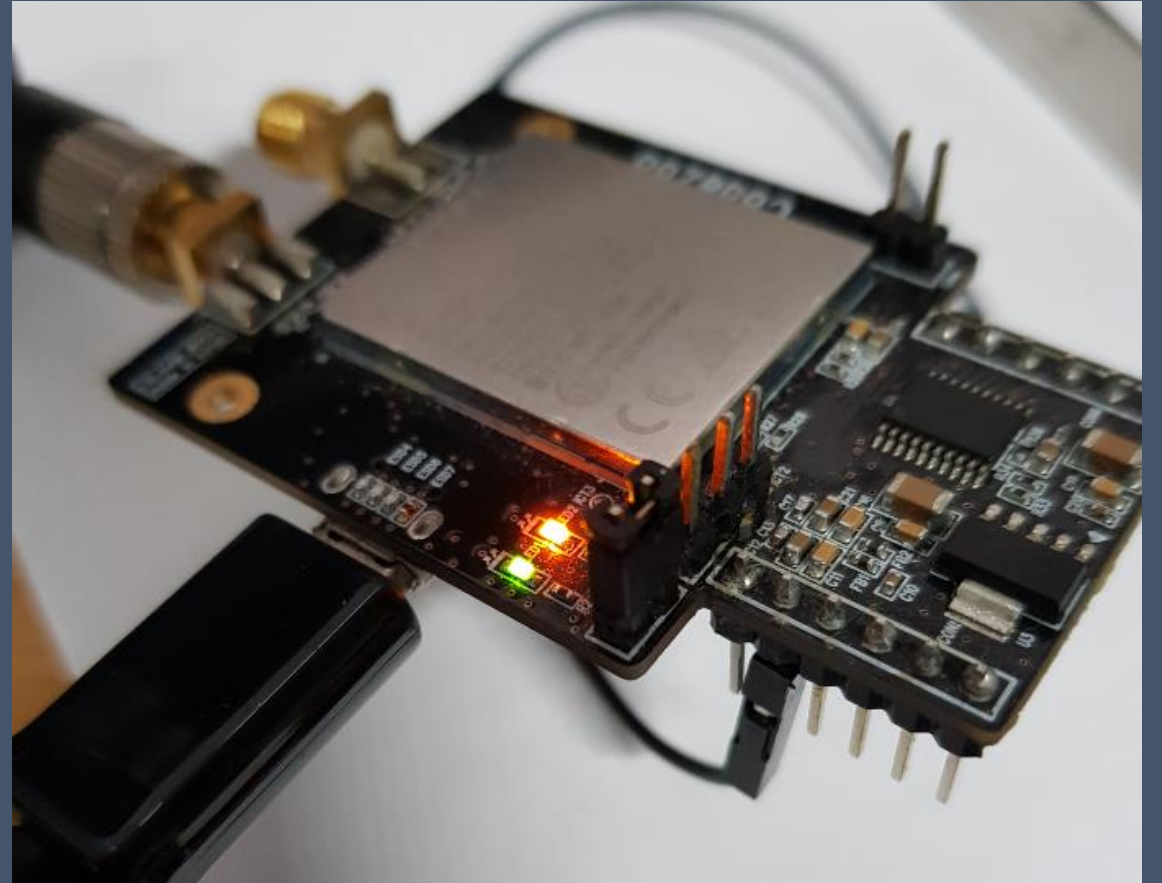
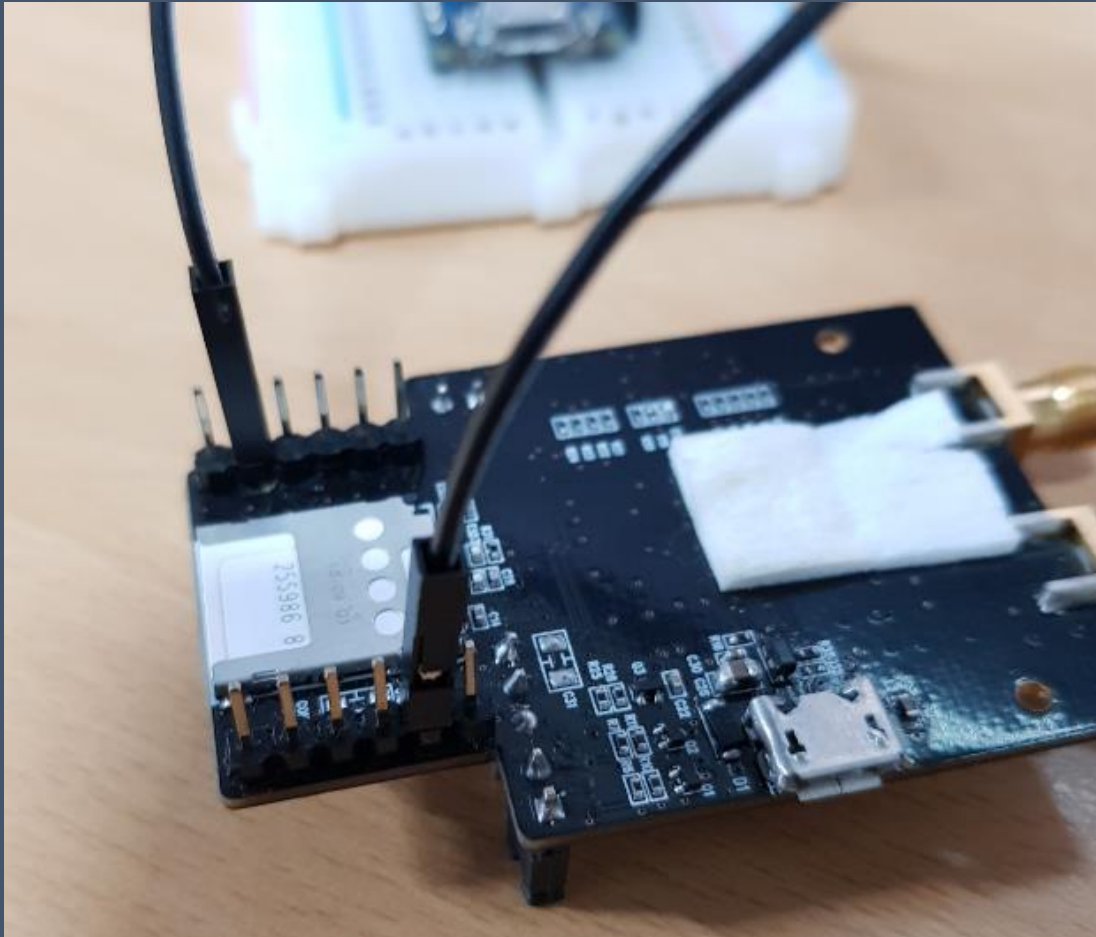
rooney.jang@codezoo.co.kr

www.CodeZoo.co.kr

1. CAT.M1 Hardware 준비

USB 연결 후 AT Command 테스트 절차

1. 후면 좌측 5번과 우측 2번을 점퍼선으로 연결
2. Micro USB 케이블을 라즈베리파이 USB 포트에 연결



2. CAT.M1 실습 (Jetson Nano Image install)

1. Introduction

<https://developer.nvidia.com/embedded/learn/get-started-jetson-nano-devkit#intro>

2. Prepare for Setup

<https://developer.nvidia.com/embedded/learn/get-started-jetson-nano-devkit#prepare>

3. Write Image to the microSD Card

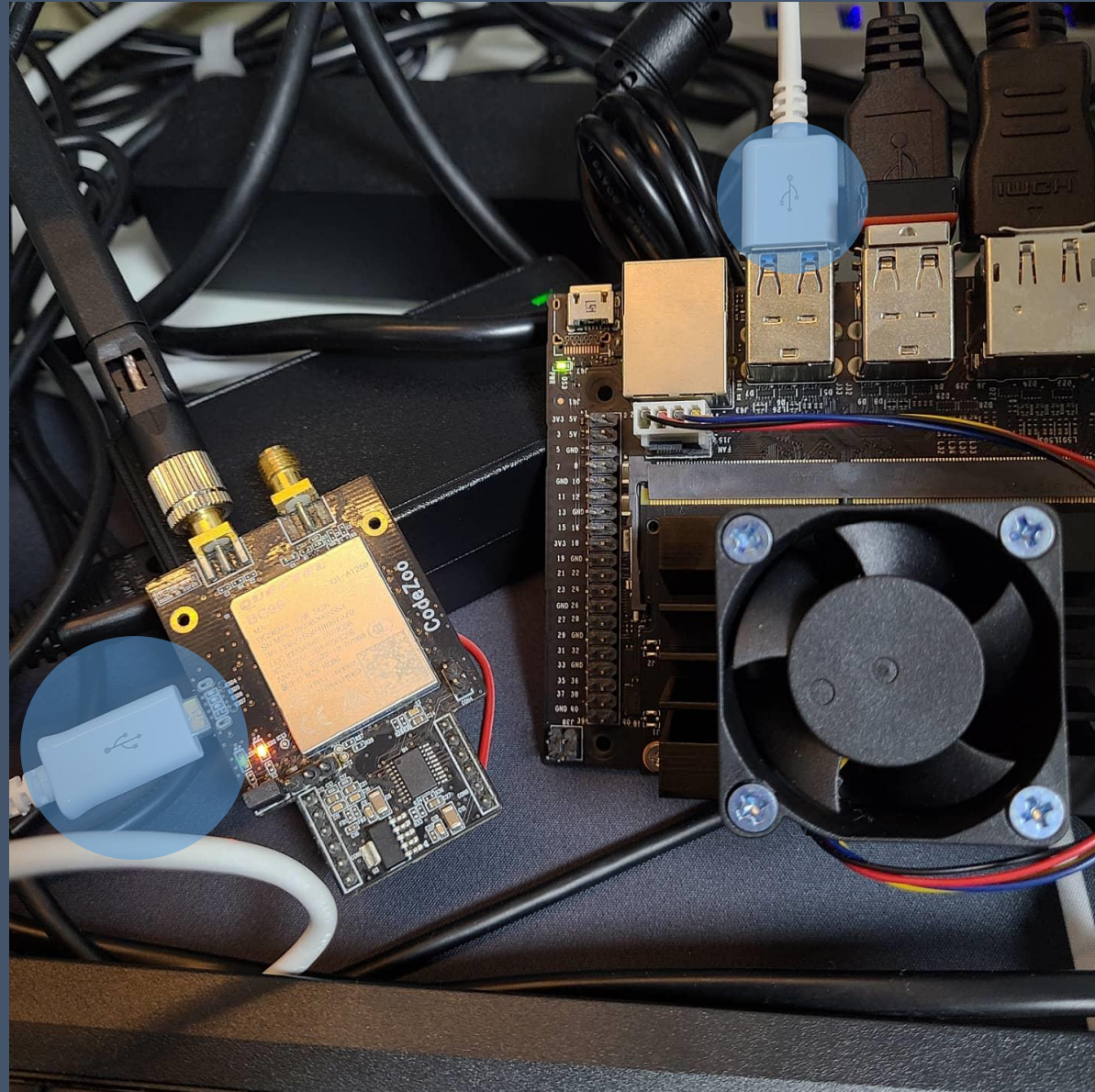
<https://developer.nvidia.com/embedded/learn/get-started-jetson-nano-devkit#write>

4. Setup and First Boot

<https://developer.nvidia.com/embedded/learn/get-started-jetson-nano-devkit#setup>

3. CAT.M1 연결

마이크로 USB 케이블로 CATM1과
Jetson Nano 연결



4. CAT.M1 WvDial install

Cat.M1 모뎀과 Jetson Nano를 PPP(Point to Point Protocol)로 연결하기 위해 WvDial 을 사용합니다.

1. WvDial 설치:

```
sudo apt-get install wvdial
```

2. 기본 WvDial 구성 파일을 편집 합니다.

```
sudo nano /etc/wvdial.conf
```

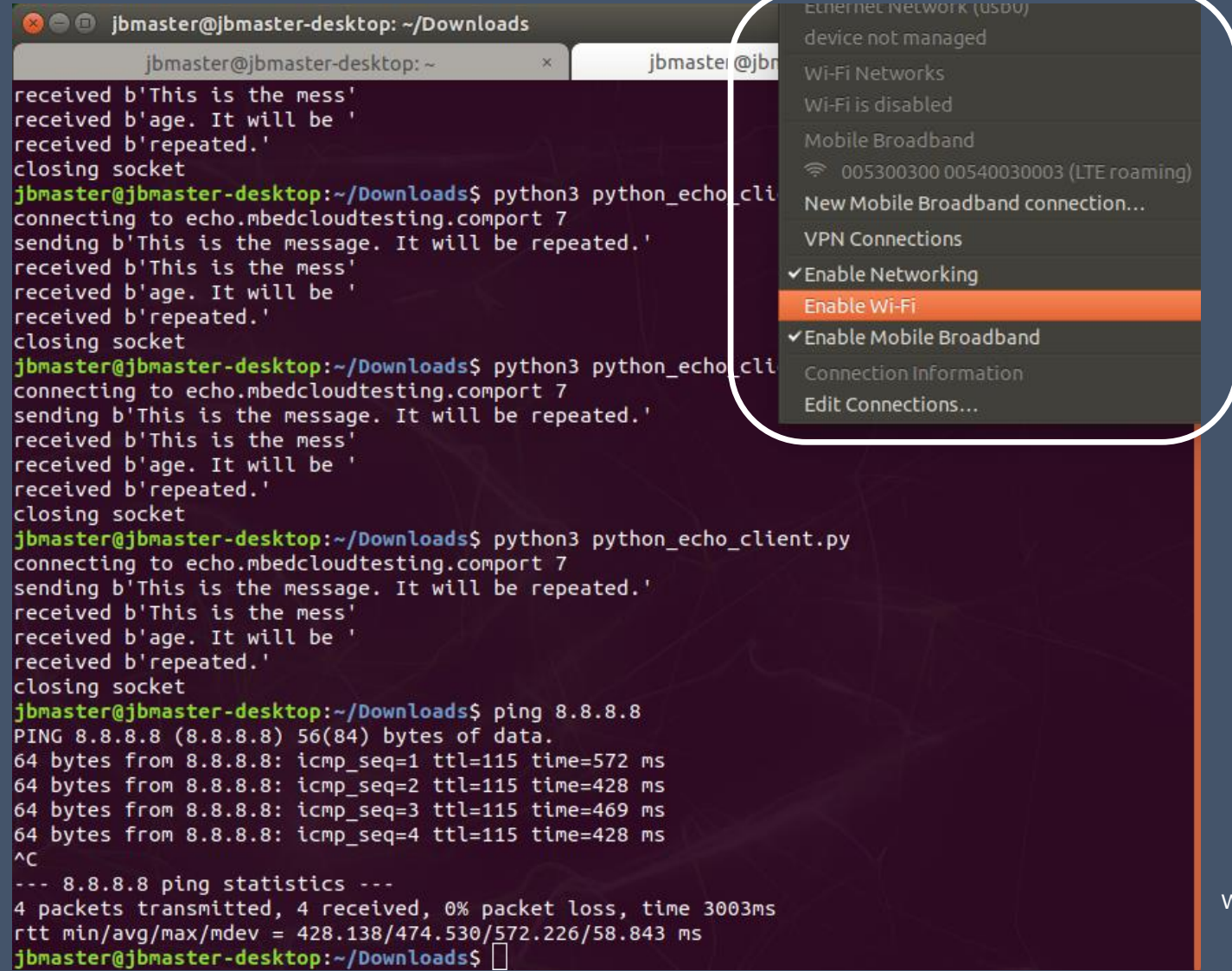
3. 파일 편집 후 시스템을 재부팅 합니다.

wvdial.conf

```
[Dialer Defaults]
Modem = /dev/ttyUSB3
Baud = 115200
Init1 = AT+CFUN=1
Init2 = ATZ
Init3 = AT+CGDCONT=1,"IP","internet.lte.cxn"
Phone = *99***1#
Dial Command = ATD
Username = codezoo
Password = codezoo
Auto DNS = 1
Check Def Route = 1
Carrier Check = 0
Stupid Mode = 1
Dial Attempts = 3
ISDN = 0
```

5. CAT.M1 PPP Test

1. Jetson Nano Wi-Fi 기능을 Disable 시킵니다. (클릭해서 Enable Wi-Fi 를 해제)



```
jbmaste@jbmaste-desktop: ~/Downloads
jbmaste@jbmaste-desktop: ~
received b'This is the mess'
received b'age. It will be '
received b'repeated.'
closing socket
jbmaste@jbmaste-desktop:~/Downloads$ python3 python_echo_cli
connecting to echo.mbedcloudtesting.comport 7
sending b'This is the message. It will be repeated.'
received b'This is the mess'
received b'age. It will be '
received b'repeated.'
closing socket
jbmaste@jbmaste-desktop:~/Downloads$ python3 python_echo_cli
connecting to echo.mbedcloudtesting.comport 7
sending b'This is the message. It will be repeated.'
received b'This is the mess'
received b'age. It will be '
received b'repeated.'
closing socket
jbmaste@jbmaste-desktop:~/Downloads$ python3 python_echo_client.py
connecting to echo.mbedcloudtesting.comport 7
sending b'This is the message. It will be repeated.'
received b'This is the mess'
received b'age. It will be '
received b'repeated.'
closing socket
jbmaste@jbmaste-desktop:~/Downloads$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=115 time=572 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=115 time=428 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=115 time=469 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=115 time=428 ms
^C
--- 8.8.8.8 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3003ms
rtt min/avg/max/mdev = 428.138/474.530/572.226/58.843 ms
jbmaste@jbmaste-desktop:~/Downloads$
```

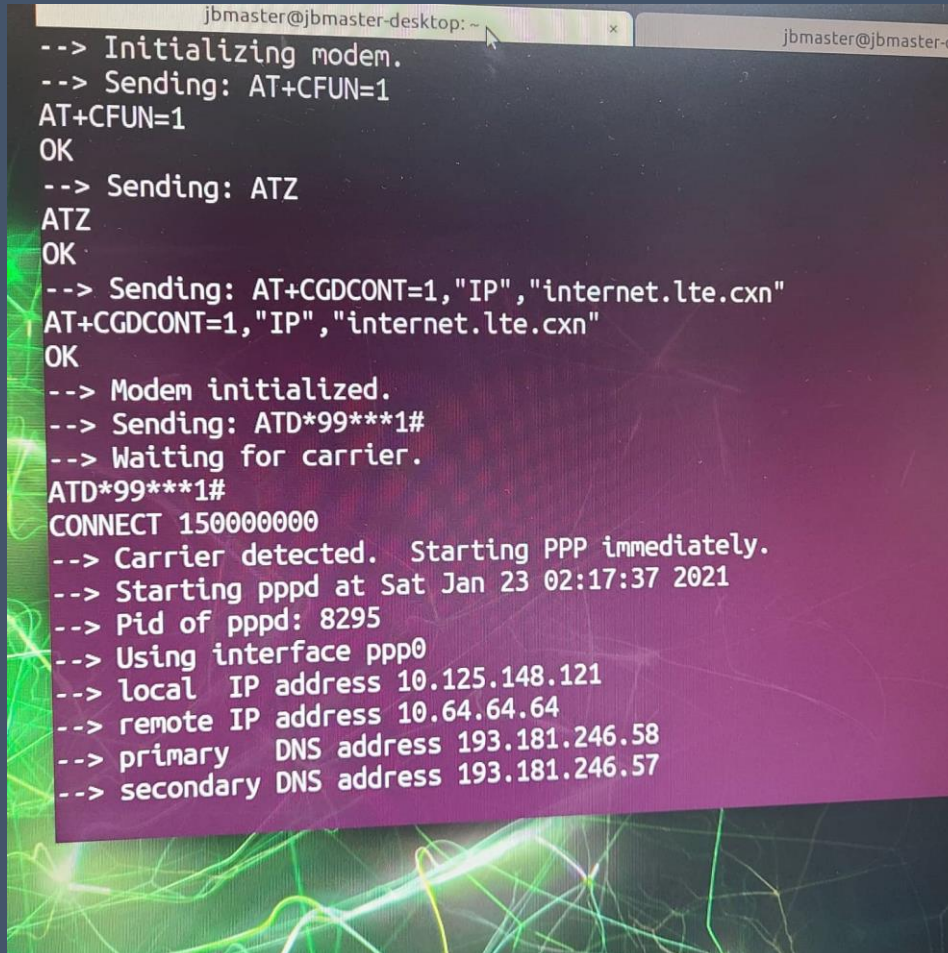
Network Settings Menu:

- Ethernet Network (usb0)
- device not managed
- Wi-Fi Networks
- Wi-Fi is disabled
- Mobile Broadband
- 005300300 00540030003 (LTE roaming)
- New Mobile Broadband connection...
- VPN Connections
- ✓ Enable Networking
- Enable Wi-Fi
- ✓ Enable Mobile Broadband
- Connection Information
- Edit Connections...

5. CAT.M1 PPP Test

2. WvDial을 실행하여 연결을 확인하십시오.
`sudo wvdial`

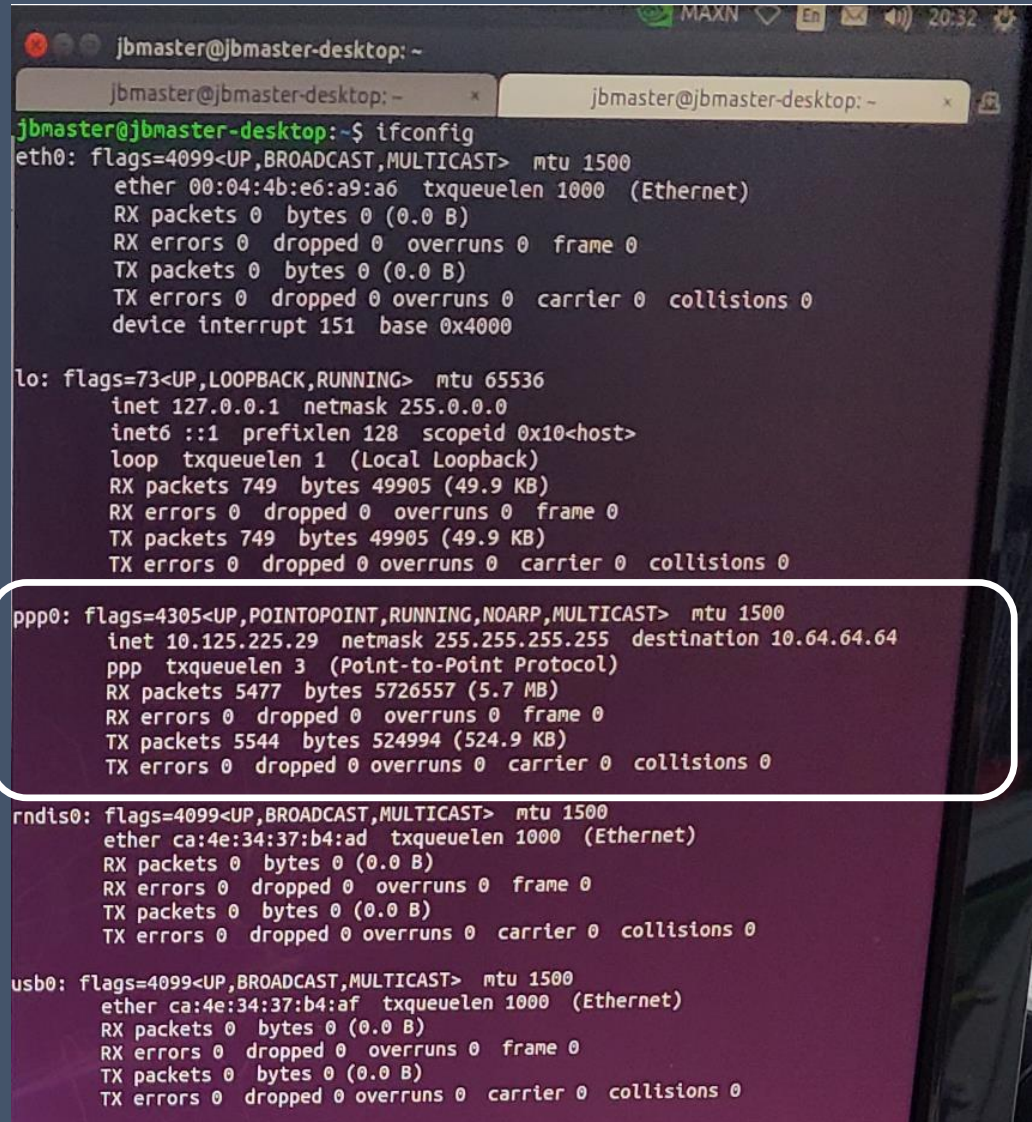
연결된 경우 다음과 유사한 출력이 표시되어야 합니다.

A terminal window screenshot showing the output of the 'wvdial' command. The terminal has a dark background with light green text. The output shows the modem initialization process, including sending AT commands, waiting for a carrier, and finally establishing a PPP connection with IP addresses and DNS settings.

```
jbmaste@jbmaste-desktop: ~  
--> Initializing modem.  
--> Sending: AT+CFUN=1  
AT+CFUN=1  
OK  
--> Sending: ATZ  
ATZ  
OK  
--> Sending: AT+CGDCONT=1,"IP","internet.lte.cxn"  
AT+CGDCONT=1,"IP","internet.lte.cxn"  
OK  
--> Modem initialized.  
--> Sending: ATD*99***1#  
--> Waiting for carrier.  
ATD*99***1#  
CONNECT 1500000000  
--> Carrier detected. Starting PPP immediately.  
--> Starting pppd at Sat Jan 23 02:17:37 2021  
--> Pid of pppd: 8295  
--> Using interface ppp0  
--> local IP address 10.125.148.121  
--> remote IP address 10.64.64.64  
--> primary DNS address 193.181.246.58  
--> secondary DNS address 193.181.246.57
```

5. CAT.M1 PPP 동작 확인

3. Jetson Nano 터미널에서 ifconfig 실행, PPP0 디바이스 확인



```
jbmater@jbmater-desktop: ~  
jbmater@jbmater-desktop: ~  
jbmater@jbmater-desktop: ~$ ifconfig  
eth0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500  
ether 00:04:4b:e6:a9:a6 txqueuelen 1000 (Ethernet)  
RX packets 0 bytes 0 (0.0 B)  
RX errors 0 dropped 0 overruns 0 frame 0  
TX packets 0 bytes 0 (0.0 B)  
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
device interrupt 151 base 0x4000  
  
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536  
inet 127.0.0.1 netmask 255.0.0.0  
inet6 ::1 prefixlen 128 scopeid 0x10<host>  
loop txqueuelen 1 (Local Loopback)  
RX packets 749 bytes 49905 (49.9 KB)  
RX errors 0 dropped 0 overruns 0 frame 0  
TX packets 749 bytes 49905 (49.9 KB)  
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
ppp0: flags=4305<UP,POINTOPOINT,RUNNING,NOARP,MULTICAST> mtu 1500  
inet 10.125.225.29 netmask 255.255.255.255 destination 10.64.64.64  
ppp txqueuelen 3 (Point-to-Point Protocol)  
RX packets 5477 bytes 5726557 (5.7 MB)  
RX errors 0 dropped 0 overruns 0 frame 0  
TX packets 5544 bytes 524994 (524.9 KB)  
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
rndis0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500  
ether ca:4e:34:37:b4:ad txqueuelen 1000 (Ethernet)  
RX packets 0 bytes 0 (0.0 B)  
RX errors 0 dropped 0 overruns 0 frame 0  
TX packets 0 bytes 0 (0.0 B)  
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
usb0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500  
ether ca:4e:34:37:b4:af txqueuelen 1000 (Ethernet)  
RX packets 0 bytes 0 (0.0 B)  
RX errors 0 dropped 0 overruns 0 frame 0  
TX packets 0 bytes 0 (0.0 B)  
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```


5. CAT.M1 PPP 동작 확인

4. 파이썬 테스트 코드로 통신 테스트

python3 python_echo_client.py

```
import socket
import sys

#Create a TCP/IP socket
sock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)

#Connect the socket to the port where the server is listening
server_address = ('echo.mbedcloudtesting.com',7)
print('connecting to {}port {}'.format(*server_address))
sock.connect(server_address)

try:

    #Send data
    message = b'This is the message. It will be repeated.'
    print('sending {!r}'.format(message))
    sock.sendall(message)

    #Look for the response
    amount_received = 0
    amount_expected = len(message)

    while amount_received < amount_expected:
        data = sock.recv(16)
        amount_received += len(data)
        print('received {!r}'.format(data))

finally:
    print('closing socket')
    sock.close()
```

감사합니다.