CodeZoo CATM1 Hands-On

- Jetson Nano (with PPP)-

version 1.0

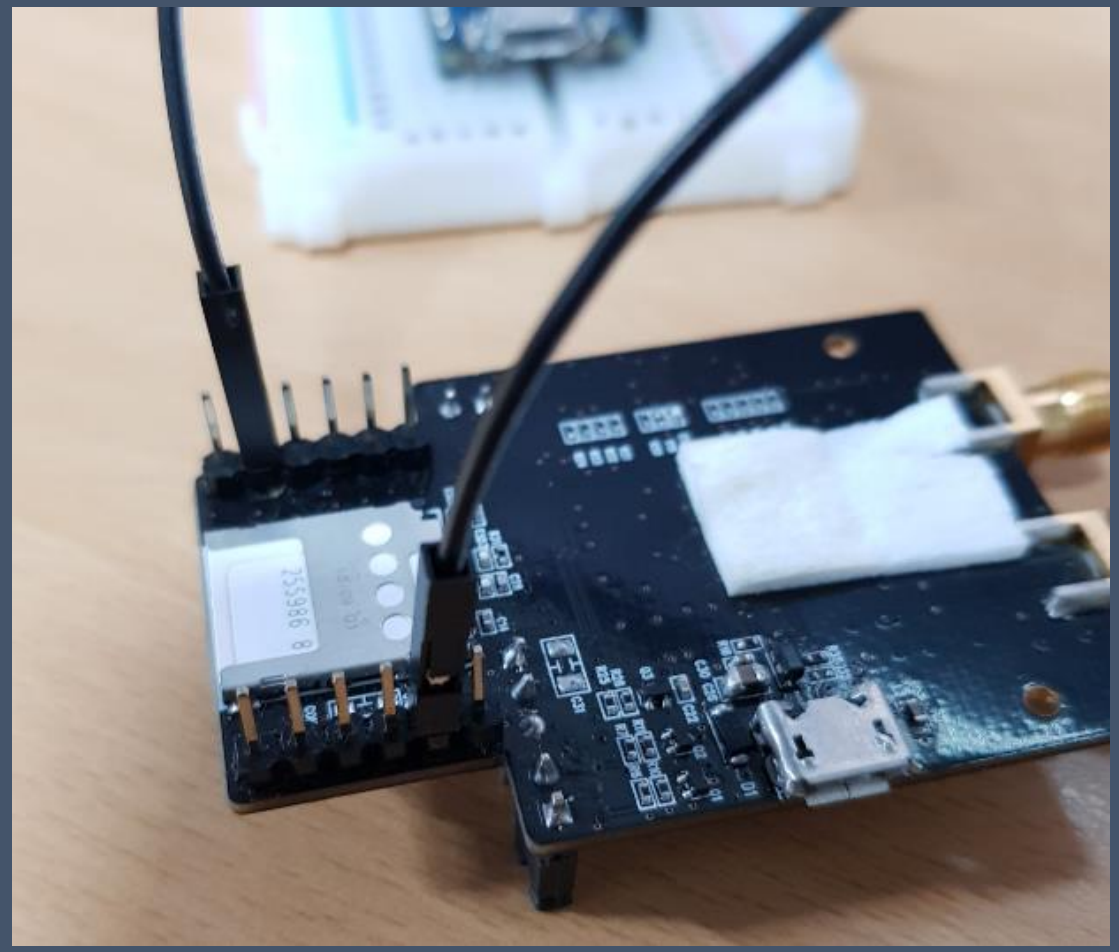
rooney.jang@codezoo.co.kr

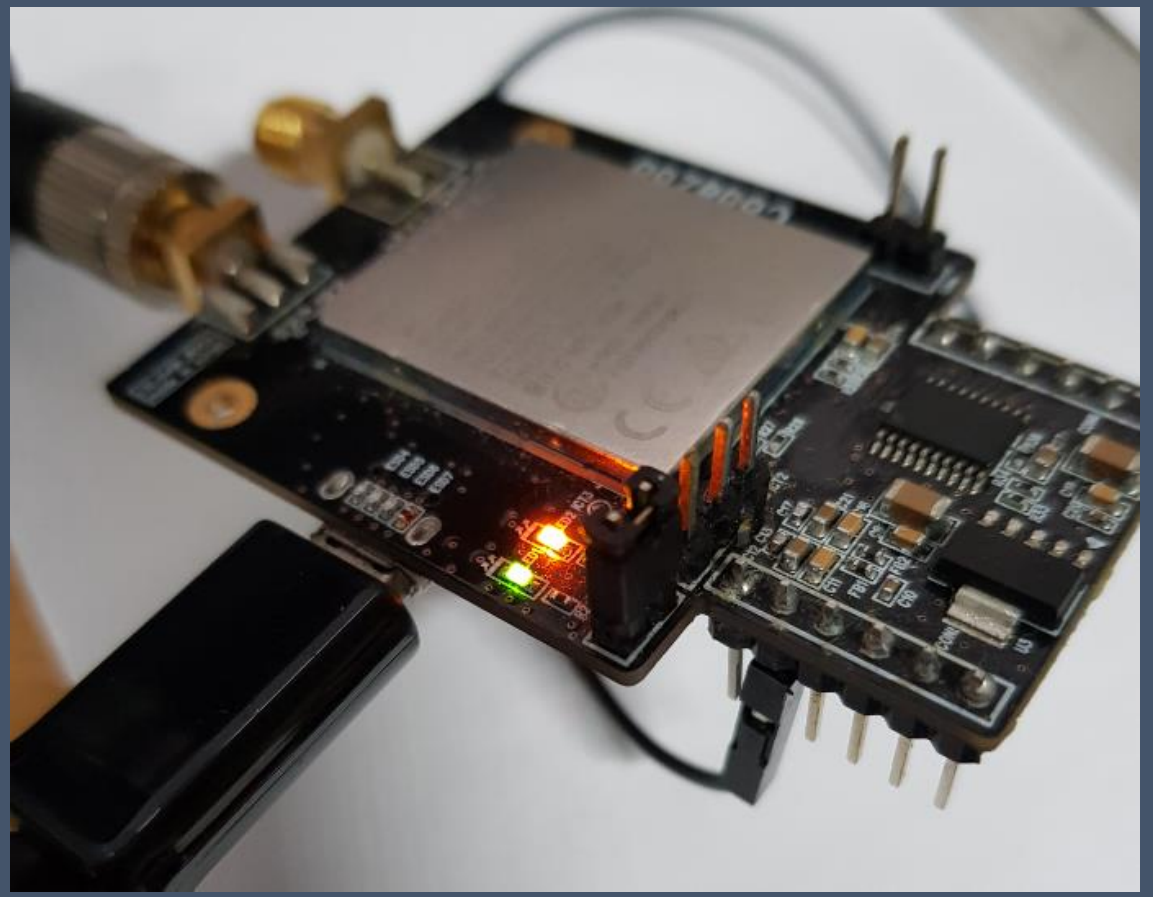
www.**Code**Zoo.co.kr

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

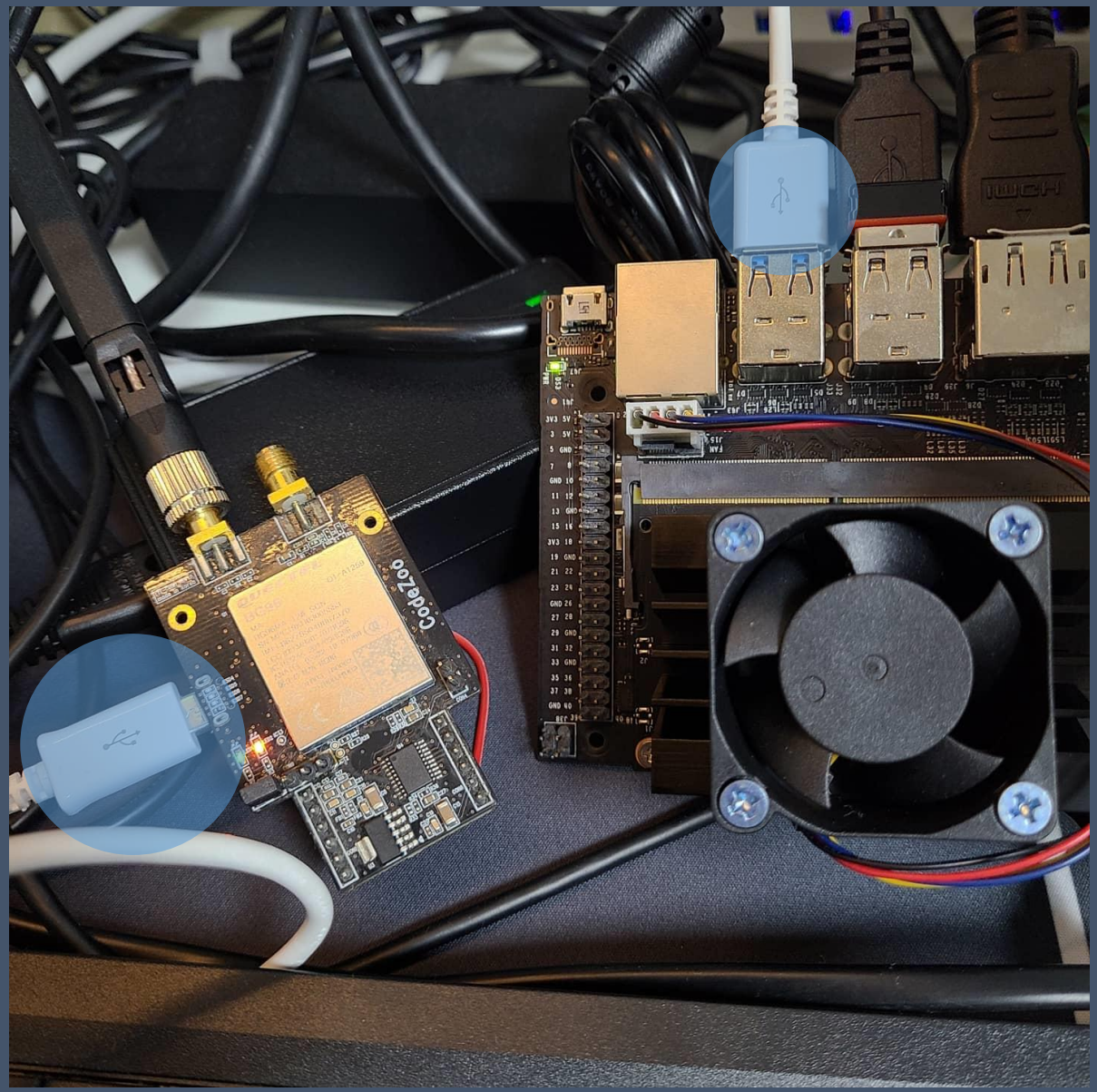
2. Prepare for Setup
https://developer.nvidia.com/embedded/learn/get-started-jetson-nano-devkit#prepare

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

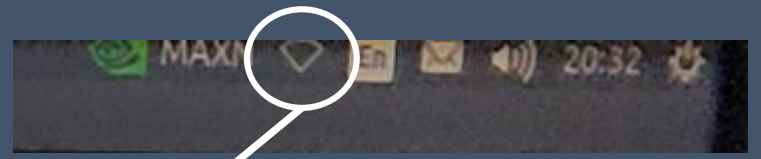
3. 파일 편집 후 시스템을 재부팅 합니다.

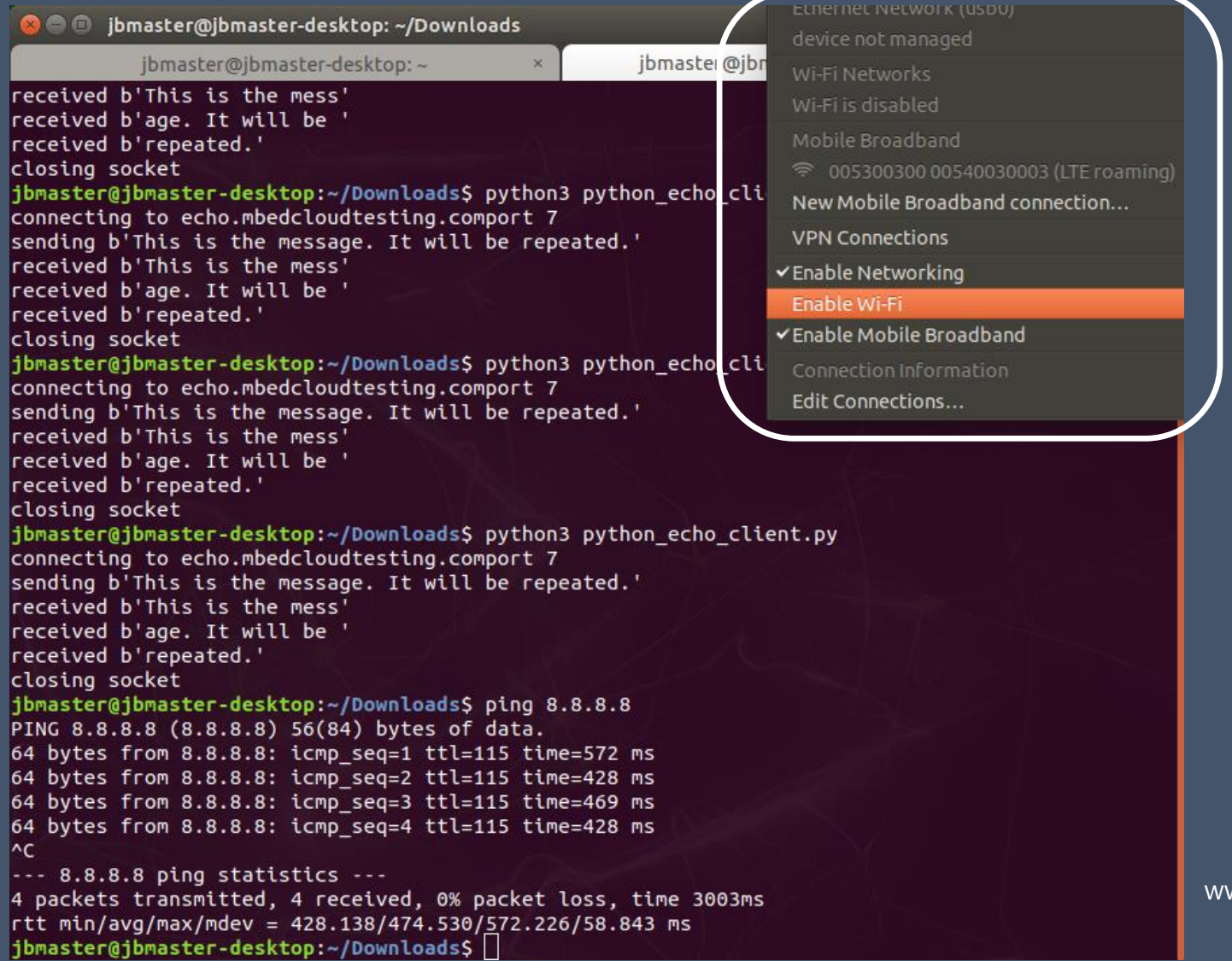
wvdial.conf

```
[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)

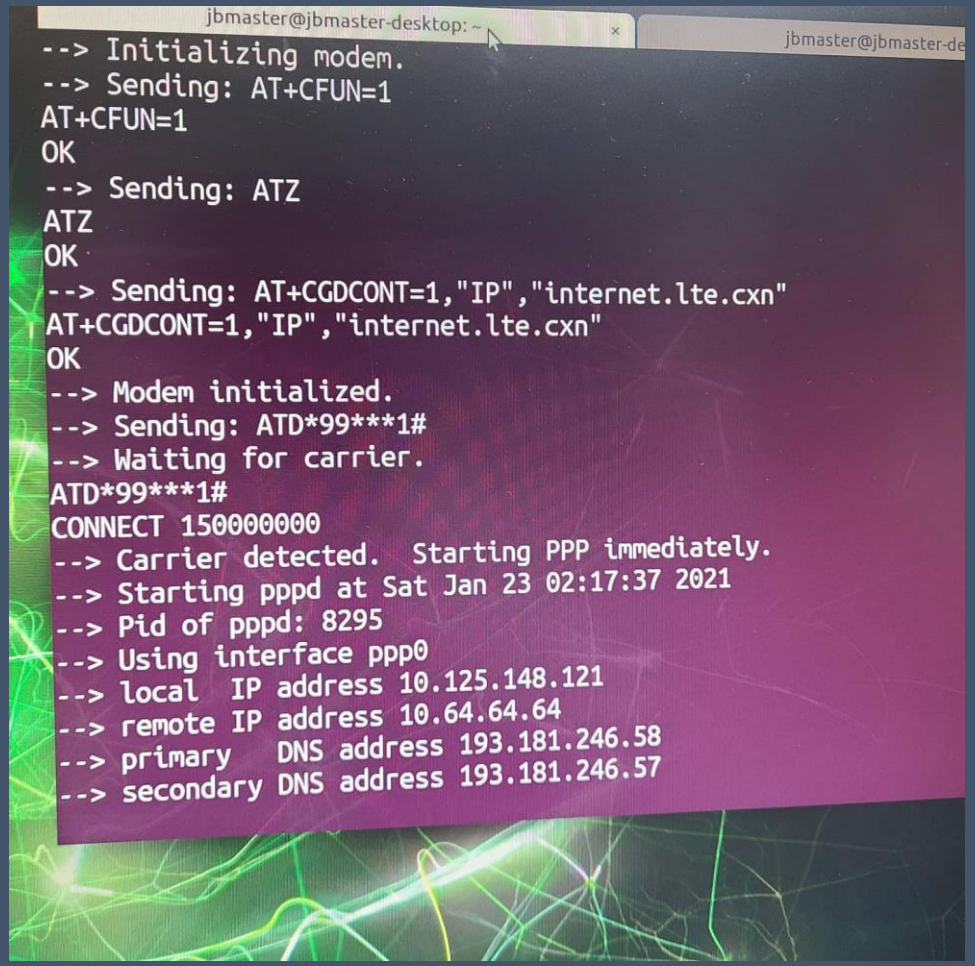




5. CAT.M1 PPP Test

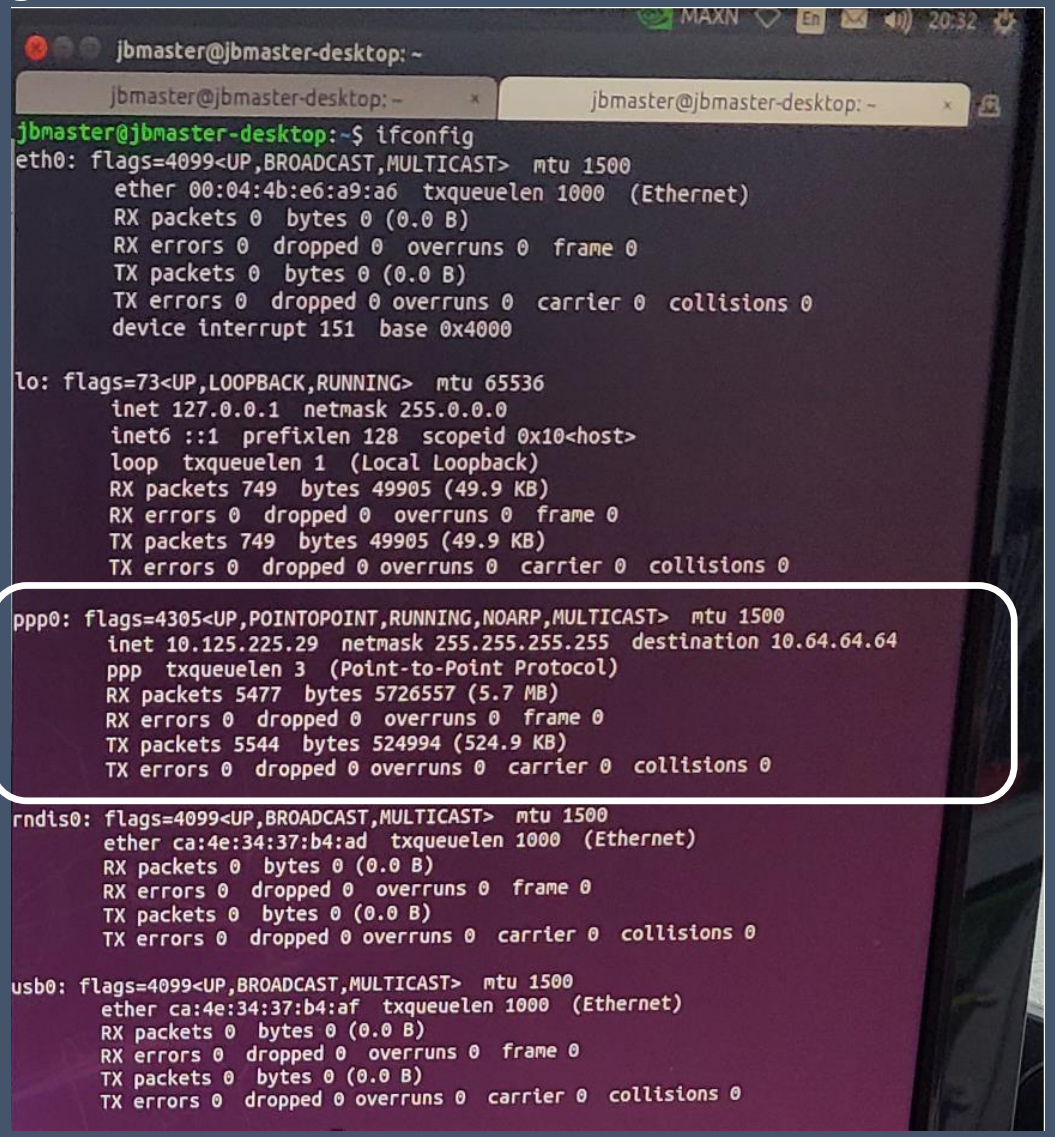
2. WvDial을 실행하여 연결을 확인하십시오.
sudo wvdial

연결된 경우 다음과 유사한 출력이 표시되어야 합니다.



5. CAT.M1 PPP 동작 확인

3. Jetson Nano 터미널에서 ifconfig 실행, PPP0 디바이스 확인



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.