Connect to the Internet via USB with LTE EG95 Using HAT on Raspberry Pi

Hardware Prerequisites

- Raspberry Pi
- Raspberry PI HAT for IoT modules
- Quectel EG95 USB module
- LTE antenna

Hardware Setup

- Plug Quectel EG95 USB module to the Raspberry PI
- Connect required antennas to the Quectel EG95 module.

Software Setup

- First run update

```
sudo apt update && sudo apt upgrade
```

- Check Linux version:

root@raspberrypi:/home/pi# uname -a

Linux raspberrypi 5.4.79-v7l+ #1373 SMP Mon Nov 23 13:27:40 GMT 2020 armv7l GNU/Linux

- Check device driver:

root@raspberrypi:/home/pi# dmesg

[12.544039] usb 1-1.4: new high-speed USB device number 3 using xhci_hcd

[12.684076] usb 1-1.4: New USB device found, idVendor=2c7c, idProduct=0195, bcdDevice= 3.18

[12.684085] usb 1-1.4: New USB device strings: Mfr=1, Product=2, SerialNumber=0

[12.684092] usb 1-1.4: Product: Android

[12.684099] usb 1-1.4: Manufacturer: Android

[12.854908] bcmgenet: Skipping UMAC reset

[12.856650] bcmgenet fd580000.ethernet: configuring instance for external RGMII

[12.857027] bcmgenet fd580000.ethernet eth0: Link is Down

[13.048212] usbcore: registered new interface driver cdc_wdm

[13.052479] usbcore: registered new interface driver usbserial_generic

[13.052507] usbserial: USB Serial support registered for generic

[13.154447] qmi_wwan 1-1.4:1.4: cdc-wdm0: USB WDM device

[13.155684] qmi_wwan 1-1.4:1.4 wwan0: register 'qmi_wwan' at usb-0000:01:00.0-1.4, WWAN/QMI device, b2:d6:44:ac:01:a2

[13.155853] usbcore: registered new interface driver qmi_wwan

[13.165068] usbcore: registered new interface driver option

[13.165103] usbserial: USB Serial support registered for GSM modem (1-port)

[13.165254] option 1-1.4:1.0: GSM modem (1-port) converter detected

[13.166760] usb 1-1.4: GSM modem (1-port) converter now attached to ttyUSB0

[13.166920] option 1-1.4:1.1: GSM modem (1-port) converter detected

[13.167105] usb 1-1.4: GSM modem (1-port) converter now attached to ttyUSB1

[13.167235] option 1-1.4:1.2: GSM modem (1-port) converter detected

[13.167405] usb 1-1.4: GSM modem (1-port) converter now attached to ttyUSB2

[13.167561] option 1-1.4:1.3: GSM modem (1-port) converter detected

[13.167733] usb 1-1.4: GSM modem (1-port) converter now attached to ttyUSB3

- Check connection:
  sudo qmicli -d /dev/cdc-wdm0 --dms-get-operating-mode
  sudo qmicli -d /dev/cdc-wdm0 --nas-get-signal-strength
  sudo qmicli -d /dev/cdc-wdm0 —nas-get-home-network

- Change qmi_wwan driver to use Raw-IP.
  Disable the network interfaces exposed by the cellular module:
  ip link set dev wwan0 down
  Trigger the Raw-IP support:
  echo Y > /sys/class/net/wwan0/qmi/raw_ip
  Enable the network interfaces again:
  ip link set dev wwan0 up

- Activate the data connection in the cellular module:
  qmicli --device=/dev/cdc-wdm0 --device-open-proxy --wds-start-network="ip-type=4,apn=<YOUR_APN>" --client-no-release-cid

- Once "Network started" is displayed, you can send a DHCP request on the network interface.
  udhcpc -q -f -n -i wwan0
- If the connection was successfully set up established, you now have data connectivity. A ping to a remote server using the cellular network interface can for example prove this:
  ping -I wwan0 8.8.8.8
- Disconnect the data bearer and data connection over QMI by command bellow and providing the network handle and CID returned at connection activation:
  qmicli --device=/dev/cdc-wdm0 --device-open-proxy --wds-stop-network=NETWORK_HANDLE —client-cid=CID

Additional useful commands:

Request module manufacturer:
qmicli --device=/dev/cdc-wdm0 --device-open-proxy --dms-get-manufacturer

Get module model:
qmicli --device=/dev/cdc-wdm0 --device-open-proxy --dms-get-model

Get firmware version:
qmicli --device=/dev/cdc-wdm0 --device-open-proxy --dms-get-revision

Get module IDs (IMEI etc.):
qmicli --device=/dev/cdc-wdm0 --device-open-proxy --dms-get-ids

Get SIM card status:
qmicli --device=/dev/cdc-wdm0 --device-open-proxy --uim-get-card-status

Recent cellular modules like Sierra Wireless EM7565 require at least libqmi V1.20. Check version with command:
qmicli --version

If the connection was successfully set up established, you now have data connectivity. A ping to a remote server using the cellular network interface can for example prove this:
ping -I wwan0 8.8.8.8

The ifconfig Linux tool can show the current details for the network interface:
ifconfig wwan0