

## **A. Rpi Network configuration when eth0 and eth1 is connected**

- **Eth0 is used for PLC**
- **Eth1 is used for Ethernet network**

**Use case1 - eth0 will connect to PLC using an ethernet cable and the network is given to eth1 with a physical ethernet cable.**

**Note –** Make sure that your virtual plc ip at eth0 is in same subnet as the original plc ip range, in my case its 172.16.51.189 with netmask 255.255.254.0

To enable Rpi to work on eth0 and eth1, do the following configuration

### **1. Go to /etc/network**

```
sudo nano /etc/network/interfaces
```

#### **Add the following lines**

```
auto eth0
iface eth0 inet static
address 172.16.51.189
netmask 255.255.254.0
```

```
auto eth1
iface eth1 inet dhcp
```

**After adding these lines restart the service as follows,**

```
sudo systemctl restart dhcpcd
sudo systemctl restart networking
sudo reboot
```

**Gateway Automation –** Now that we have set up the gateway configuration, its time to set up an automation job to run your python code at bootup which can be done by systemctl.

**Use the following commands for gateway automation**

```
sudo nano /etc/systemd/system/my_script.service
```

write the following –

```
[Unit]
Description=Run my Python script on startup
After=network.target
[Service]
Type=simple
ExecStart=/bin/bash -c 'cd /home/pi/wipro-pari/myenv/bin/python mycode.py
WorkingDirectory=/home/pi/WIPRO/
StandardOutput=journal+console
StandardError=journal+console
Restart=on-failure
RestartSec=5s
User=pi
```

[Install]

WantedBy=multi-user.target

```
sudo systemctl daemon-reload
sudo systemctl enable my_script.service
sudo systemctl start my_script.service
sudo systemctl status my_script.service
```

- 2. Now do “ifconfig” to check for interface ips, if you see any ip at eth0 or eth1, that means your interfaces are enabled.**

## **B. Rpi Network configuration when usb0(GSM) and eth0 is connected**

**Use case2 - eth0 will connect to PLC using an ethernet cable and the network is given to usb0 with a physical GSM module externally attached to Rpi.**

**Note** – Make sure that your virtual plc ip at eth0 is in same subnet as the original plc ip range, in my case its 172.16.51.189 with netmask 255.255.254.0 and the GSM module is working. The blinking will be continuous at the GSM module which means the GSM is working and in your ifconfig you will see a usb0 interface which is your GSM.

To enable Rpi to work on eth0 and usb0, do the following configuration

### **1. Go to /etc/network**

```
sudo nano /etc/network/interfaces
```

#### **Add the following lines**

```
auto eth0
iface eth0 inet static
address 172.16.51.189
netmask 255.255.254.0
```

```
auto usb0
iface usb0 inet dhcp
```

#### **After adding these lines restart the service as follows,**

```
sudo systemctl restart dhcpcd
sudo systemctl restart networking
sudo reboot
```

**Note** – If you want to access net on GSM module only i.e. your usb0, you can select only usb0 to be your network interface as follows, I have added these commands in my systemctl job command

```
sudo ifdown usb0 && sudo ifup usb0
sudo ip link set usb0 up
```

**Gateway Automation** – Now that we have set up the gateway configuration, its time to set up an automation job to run your python code at bootup which can be done by systemctl.

**Use the following commands for gateway automation**

```
sudo nano /etc/systemd/system/my_script.service
```

**write the following –**

```
[Unit]
Description=Run my Python script on startup
After=network.target
[Service]
Type=simple
ExecStart=/bin/bash -c ' sudo ifdown usb0 && sudo ip link set usb0 up &&sudo ifup
usb0&&cd /home/pi/wipro-pari/myenv/bin/python mycode.py
WorkingDirectory=/home/pi/WIPRO/
StandardOutput=journal+console
StandardError=journal+console
Restart=on-failure
RestartSec=5s
User=pi

[Install]
WantedBy=multi-user.target
```

```
sudo systemctl daemon-reload
sudo systemctl enable my_script.service
sudo systemctl start my_script.service
sudo systemctl status my_script.service
```