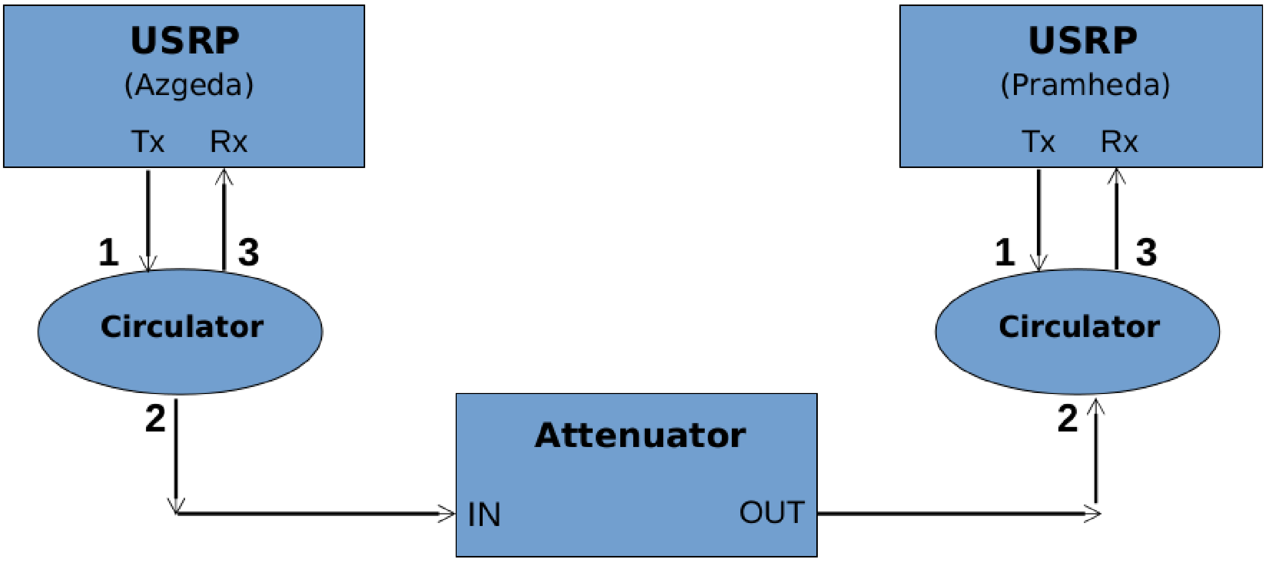
Guidelines

1) Deployment:

The scheme for the addressed setup:



2) Initial setting

2.1) Booting the mini-PC

When booting the mini-PC, press PowerOn button for 3 seconds, and then F4 option. See the Grub2 installation menu, and choose lowlatency kernel option. For instance, 4.15.0-141-lowlatency. After booting, check the loaded kernel by typing: uname -r

2.2) Remote connection

Execute the following commands to establish ssh connections:

```
ssh -X azgeda@192.168.12.215
ssh -X pramheda@192.168.12.160
```

The required password: linux

2.3) Compilation

Each time, the mini-PC is rebooted, a compilation has to be done in order to load the new UE_IP kernel, so that it is better to avoid booting the mini-PC.

The D2D demo is located in /home/pramheda/LTE-D2D (in a similar way for the azgeda).

```
cd ~/openairinterface5g/cmake_targets
./build_oai --UE -w USRP
./build_oai --UE -w USRP -c (if needed)
```

Here a list of folder that will be used to run the D2D demo:

```
cd ~/LTE-D2D/openairinterface5g
cd ~/LTE-D2D/openairinterface5g/cmake_targets/tools
cd ~/LTE-D2D/openairinterface5g/cmake_targets
cd ~/LTE-D2D/openairinterface5g/cmake_targets/lte_build_oai/build
cd ~/d2d-l3-stub
```

3) Running a D2D link

At **Pramheda** side:

```
# 1) Configurations
cd LTE-D2D/openairinterface5g/cmake_targets/tools
source init_nas_s1 UE

# 2) IP and MAC addresses
cd LTE-D2D/openairinterface5g/cmake_targets/

sudo ifconfig oip0 10.0.0.1
sudo ifconfig oip0 hw ether 00:00:00:00:00:01
sudo ip neigh add 10.0.0.2 lladdr 00:00:00:00:00:02 dev oip0 nud permanent

# 3) Routing
sudo iptables -t mangle -F
sudo iptables -t nat -F
sudo iptables -A POSTROUTING  -t mangle -o oip0 -d 10.0.0.2 -j MARK --set-mark 4

# 4) Running
cd LTE-D2D/openairinterface5g/cmake_targets/lte_build_oai/build
sudo ./lte-uesoftmodem -C 5900000000 -r 50 --ue-synchref --ue-enable-sl --ue-sl-only --ue-rxgain
100 --ue-txgain 0 --usrp-clksrc gpsdo
```

At **Azgeda** side:

```
# 1) Configurations
cd LTE-D2D/openairinterface5g/cmake_targets/tools
source init_nas_s1 UE

# 2) IP and MAC addresses - check it with ip neigh
sudo ifconfig oip0 10.0.0.2
sudo ifconfig oip0 hw ether 00:00:00:00:00:02
sudo ip neigh add 10.0.0.1 lladdr 00:00:00:00:00:01 dev oip0 nud permanent

# 3) Routing
sudo iptables -t mangle -F
sudo iptables -t nat -F
sudo iptables -A POSTROUTING  -t mangle -o oip0 -d 10.0.0.1 -j MARK --set-mark 4
```

```
# 4) Running
LTE-D2D/openairinterface5g/cmake_targets/lte_build_oai/build
sudo ./lte-uesoftmodem -C 5900000000 -r 50 --ue-enable-sl --ue-sl-only --ue-rxgain 100 --ue-txgain 0 --usrp-clksrc gpsdo
```

Note that under this configuration Pramheda (with 0x01 address) provides a synchronization signal (ue-synchref parameter) to Azgeda (with 0x02 address).

3.1) Establishing PC5

To establish this link, we must go to the folder d2d-l3-stub to find the executable **d2d_app**, and execute it first at the Pramheda side, and later on Azgeda.

```
# Pramheda
./d2d_app -r 0x01 0x02

# Azgeda
./d2d_app -s 0x02 0x01
```

If necessary we can compile these utilities as follows:

```
gcc -I . d2d_app.c -o d2d_app -lpthread
gcc -I . vencore_app.c -o vencore_app -lpthread
```

4) Testing D2D

To test the performance, as sever at the Pramheda side,

```
sudo iperf -s -i 1 -u -B 10.0.0.1
sudo wireshark
```

As client at the Azgeda side,

```
sudo iperf -c 10.0.0.2 -u -b 0.1M --bind 10.0.0.1 -t 100
```

If we want to send pcapng files:

```
sudo tcpreplay -i oip0  pcapng_filename
```

5) Comments

If using the Ethernet Attenuator, it could be better to slightly increase the value of the transmission power to 10, or 20, so that it can compensate some attenuations.

```
# Pramheda
sudo ./lte-uesoftmodem -C 5900000000 -r 50 --ue-synchref --ue-enable-sl --ue-sl-only --ue-rxgain 100 --ue-txgain 20 --usrp-clksrc gpsdo

# Azgeda
sudo ./lte-uesoftmodem -C 5900000000 -r 50 --ue-enable-sl --ue-sl-only --ue-rxgain 100 --ue-txgain 20 --usrp-clksrc gpsdo
```

For the transmission of packets, the kernel was modified to recognize L2 packets, for which a new folder named LTE-D2D-PCAP has been included. The procedures are the same as the ones followed until now.

6) Source code

The code for the OAI and PC5 can be downloaded from EURECOM gitlab,

```
# OAI
git clone https://gitlab.eurecom.fr/oai/openairinterface5g.git
git checkout LTE-sidelink

#Documentation:
cd openairinterface5g/targets/DOCS

# PC5
git clone https://gitlab.eurecom.fr/tien-thinh.nguyen/d2d-l3-stub
```