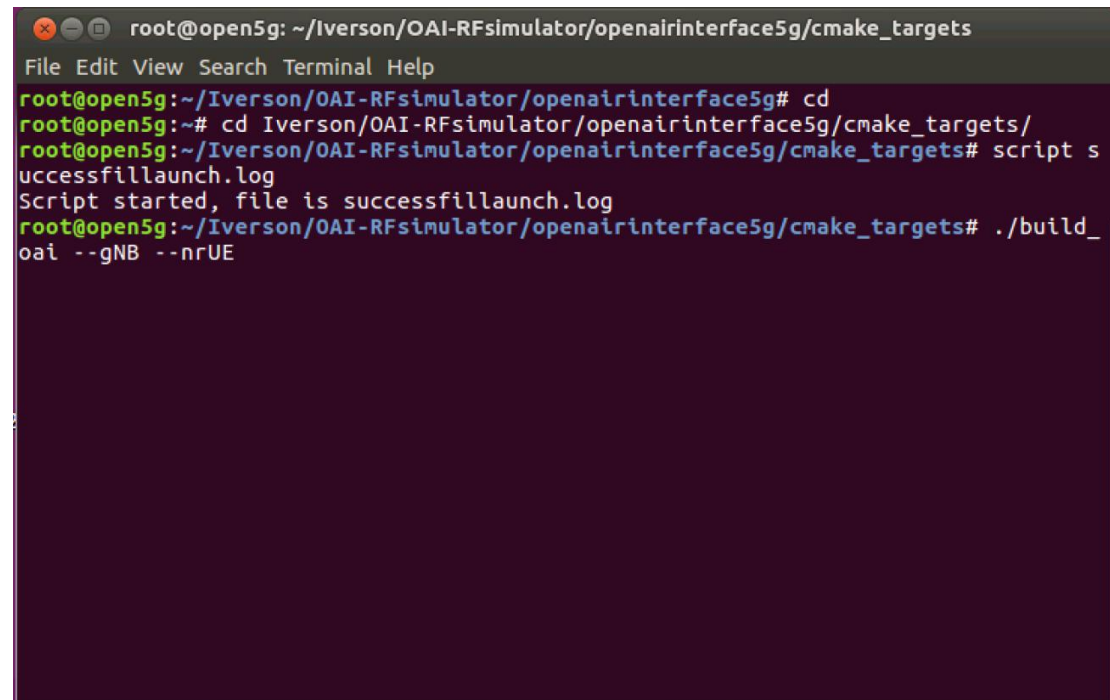


这篇文章讲述了如何正确运行 OAI rfsimulator，从而可以利用仿真的形式展示下行信道的信息，本次运行是在同一台主机上运行 gNB 和 nrUE。安装并正确编译好 OAI rfsimulator 之后便可以按照本教程操作。

第一步：

打开一个窗口按照如下图所示运行如下指令

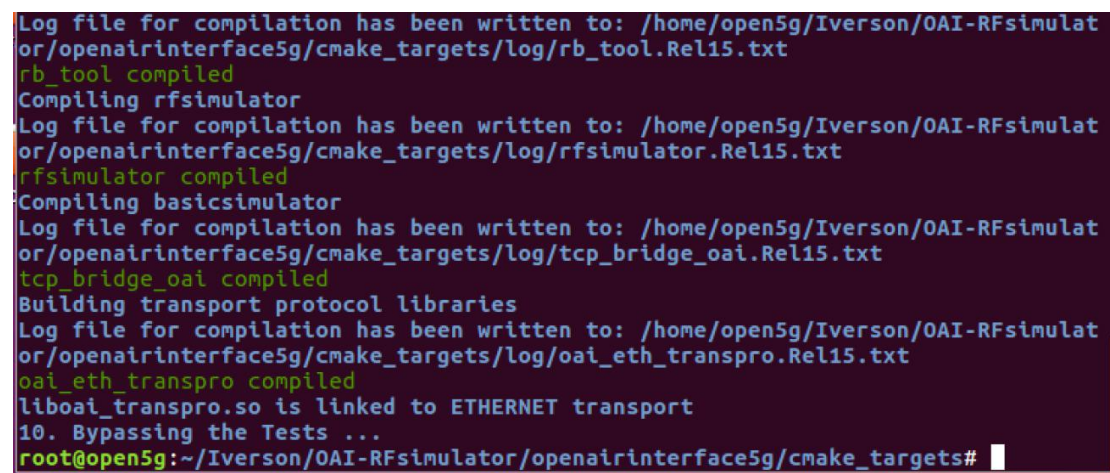


```
root@open5g: ~/Iverson/OAI-RFsimulator/openairinterface5g/cmake_targets
File Edit View Search Terminal Help
root@open5g:~/Iverson/OAI-RFsimulator/openairinterface5g# cd
root@open5g:~# cd Iverson/OAI-RFsimulator/openairinterface5g/cmake_targets/
root@open5g:~/Iverson/OAI-RFsimulator/openairinterface5g/cmake_targets# script s
uccessfillaunch.log
Script started, file is successfillaunch.log
root@open5g:~/Iverson/OAI-RFsimulator/openairinterface5g/cmake_targets# ./build_
oai --gNB --nrUE
```

等待 nr-softmodem 和 nr-uesoftmodem 以及其他依赖项编译成功，运行成功需要一点时间。

备注：script 的作用请自行查找

运行成功我们会看到如下图所示的提示



```
Log file for compilation has been written to: /home/open5g/Iverson/OAI-RFsimulat
or/openairinterface5g/cmake_targets/log/rb_tool.Rel15.txt
rb_tool compiled
Compiling rfsimulator
Log file for compilation has been written to: /home/open5g/Iverson/OAI-RFsimulat
or/openairinterface5g/cmake_targets/log/rfsimulator.Rel15.txt
rfsimulator compiled
Compiling basicsimulator
Log file for compilation has been written to: /home/open5g/Iverson/OAI-RFsimulat
or/openairinterface5g/cmake_targets/log/tcp_bridge_oai.Rel15.txt
tcp_bridge_oai compiled
Building transport protocol libraries
Log file for compilation has been written to: /home/open5g/Iverson/OAI-RFsimulat
or/openairinterface5g/cmake_targets/log/oai_eth_transpro.Rel15.txt
oai_eth_transpro compiled
liboai_transpro.so is linked to ETHERNET transport
10. Bypassing the Tests ...
root@open5g:~/Iverson/OAI-RFsimulator/openairinterface5g/cmake_targets#
```

第二步：

再打开一个运行 gNB 的窗口，运行如下图所示指令：

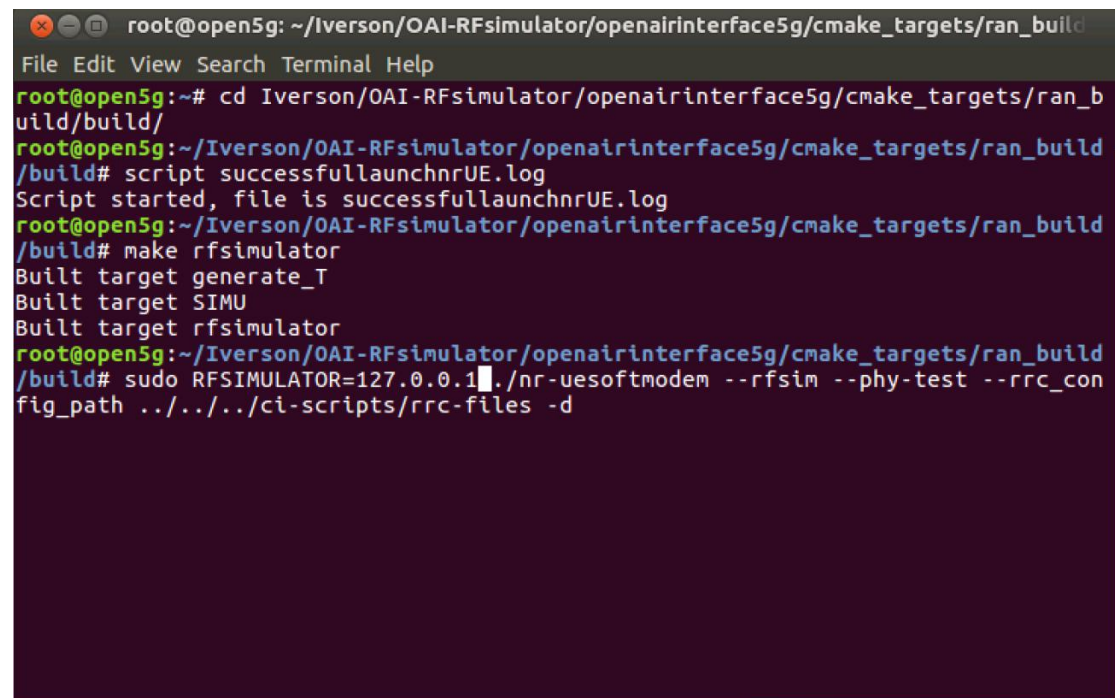
```
root@open5g: ~/Iverson/OAI-RFsimulator/openairinterface5g/cmake_targets/ran_build
File Edit View Search Terminal Help
root@open5g:~# cd Iverson/OAI-RFsimulator/openairinterface5g/cmake_targets/ran_build/build/build/
root@open5g:~/Iverson/OAI-RFsimulator/openairinterface5g/cmake_targets/ran_build/build# script successfullaunchgNB.log
Script started, file is successfullaunchgNB.log
root@open5g:~/Iverson/OAI-RFsimulator/openairinterface5g/cmake_targets/ran_build/build# make rfsimulator
Built target generate_T
Built target SIMU
Built target rfsimulator
root@open5g:~/Iverson/OAI-RFsimulator/openairinterface5g/cmake_targets/ran_build/build# sudo RFSIMULATOR=server ./nr-softmodem -O ../../../../targets/PROJECTS/GENERIC-LTE-EPC/CONF/gnb.band78.tm1.106PRB.usrpn300.conf --parallel-config PARALLEL_SINGLE_THREAD --rfsim --phy-test
```

运行结果如下图所示

```
root@open5g: ~/Iverson/OAI-RFsimulator/openairinterface5g/cmake_targets/ran_build
File Edit View Search Terminal Help
[PHY] ULSCH received ok
[PHY] ULSCH received ok
[PHY] ULSCH received ok
[PHY] ULSCH received ok
[HW] No UE, Generated void samples for Rx: 52224000
[PHY] ULSCH received ok
[PHY] ULSCH received ok
[PHY] ULSCH received ok
[MAC] Frame 88, slot 0: Adding BCH PDU in position 0 (length 3)
[PHY] 88.0 : pbch_pdu: 6f60b
[PHY] ULSCH received ok
[PHY] ULSCH received ok
[HW] No UE, Generated void samples for Rx: 55296000
[PHY] ULSCH received ok
[PHY] ULSCH received ok
[PHY] ULSCH received ok
[PHY] ULSCH received ok
[PHY] ULSCH received ok
[HW] No UE, Generated void samples for Rx: 58368000
[PHY] ULSCH received ok
[MAC] Frame 96, slot 0: Adding BCH PDU in position 0 (length 3)
[PHY] 96.0 : pbch_pdu: 6f60d
[PHY] ULSCH received ok
```

第三步：

打开第三个窗口，用于运行 nrUE，运行如下图所示指令：

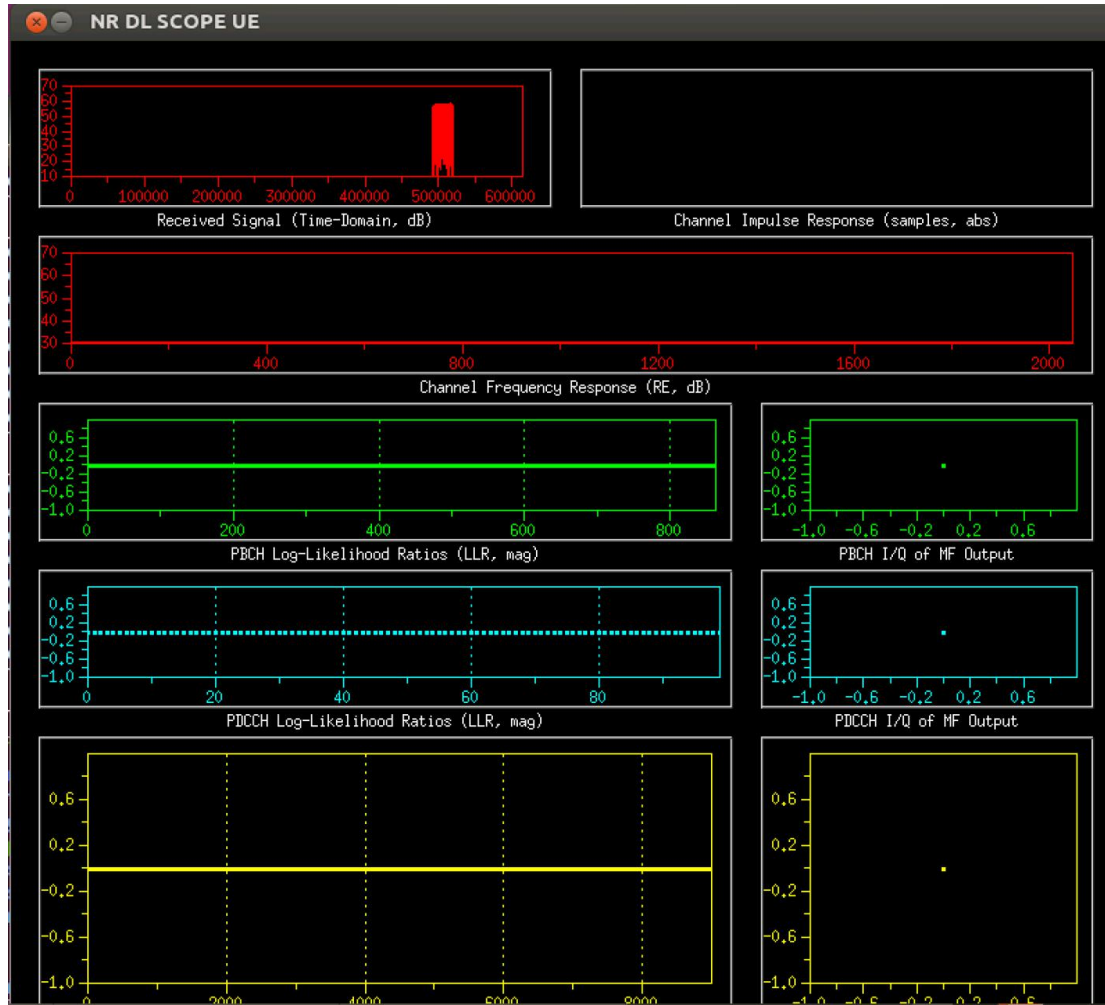


```
root@open5g: ~/Iverson/OAI-RFsimulator/openairinterface5g/cmake_targets/ran_build
File Edit View Search Terminal Help
root@open5g:~# cd Iverson/OAI-RFsimulator/openairinterface5g/cmake_targets/ran_build/build/build/
root@open5g:~/Iverson/OAI-RFsimulator/openairinterface5g/cmake_targets/ran_build/build# script successfullaunchnrUE.log
Script started, file is successfullaunchnrUE.log
root@open5g:~/Iverson/OAI-RFsimulator/openairinterface5g/cmake_targets/ran_build/build# make rfsimulator
Built target generate_T
Built target SIMU
Built target rfsimulator
root@open5g:~/Iverson/OAI-RFsimulator/openairinterface5g/cmake_targets/ran_build/build# sudo RFSIMULATOR=127.0.0.1 ./nr-uesoftmodem --rfsim --phy-test --rrc_config_path ../../../../ci-scripts/rrc-files -d
```

备注：由于我们是在同一台主机上运行的 gNB 和 nrUE，所以目标 gNB 地址采用默认的 127.0.0.1，如果 gNB 和 nrUE 不是在同一台机器运行的，只需将 127.0.0.1 改为运行 gNB 的主机地址。

备注：-d 只添加在 nrUE 窗口的命令行，用于获得 softscope

运行之后会有一个 NR DL SCOPE UE 窗口自动打开，结果如下图所示



显然此时的运行结果出现了问题，不要慌张，我们只需按 ctrl+c 退出 gNB 窗口，gNB 停止运行此时 nrUE 也会自动退出。此时，我们只需在刚刚打开的 gNB 窗口运行如下图所示指令

```
root@open5g:~/Iverson/OAI-RFsimulator/openairinterface5g/cmake_targets/ran_build/build# sudo RFSIMULATOR=server ./nr-softmodem -O ../../../../targets/PROJECTS/GENERIC-LTE-EPC/CONF/gnb.band78.tm1.106PRB.usrp300.conf --parallel-config PARALLEL_SINGLE_THREAD --rfsim --phy-test
```

在刚刚打开的 nrUE 窗口运行如下所示指令

```
root@open5g:~/Iverson/OAI-RFsimulator/openairinterface5g/cmake_targets/ran_build/build# sudo RFSIMULATOR=127.0.0.1 ./nr-uesoftmodem --rfsim --phy-test --rrc_config_path ../../../../ci-scripts/rrc-files -d
```

我们就会看到下图的结果：



第四步：

利用 iperf 进行网络测试：

打开第四个窗口（此时的测试是 gNB 和 nrUE 运行在相同的主机下进行的网络测试，如果 gNB 和 nrUE 运行在不同的主机上只需将 127.0.0.1 改成对应主机的 IP 地址即可），运行如下图所示指令（如果 gNB 和 nrUE 运行在不同的主机上，那么第四个窗口就是在 nrUE 主机上运行的）

```
root@open5g: ~  
File Edit View Search Terminal Help  
root@open5g:~# iperf -s -i 1 -u -B 127.0.0.1  
-----  
Server listening on UDP port 5001  
Binding to local address 127.0.0.1  
Receiving 1470 byte datagrams  
UDP buffer size: 208 KByte (default)  
-----
```

打开第五个窗口（此时的测试是 gNB 和 nrUE 运行在相同的主机下进行的网络测试，如果 gNB 和 nrUE 运行在不同的主机上只需将 127.0.0.1 改成对应主机的 IP 地址即可），运行如下图所示指令（如果 gNB 和 nrUE 运行在不同的主机上，那么第四个窗口就是在 gNB 主机上运行的）

```
root@open5g: ~  
File Edit View Search Terminal Help  
root@open5g:~# iperf -c 127.0.0.1 -u -b 0.1M --bind 127.0.0.1  
bind failed: Address already in use  
-----  
Client connecting to 127.0.0.1, UDP port 5001  
Binding to local address 127.0.0.1  
Sending 1470 byte datagrams  
UDP buffer size: 208 KByte (default)  
-----
```

运行结果如下图所示:

```
root@open5g: ~
File Edit View Search Terminal Help
root@open5g:~# iperf -s -i 1 -u -B 127.0.0.1
-----
Server listening on UDP port 5001
Binding to local address 127.0.0.1
Receiving 1470 byte datagrams
UDP buffer size: 208 KByte (default)
-----
[ 3] local 127.0.0.1 port 5001 connected with 127.0.0.1 port 46206
[ ID] Interval      Transfer      Bandwidth      Jitter      Lost/Total Datagrams
[ 3] 0.0- 1.0 sec   11.5 KBytes   94.1 Kbits/sec  0.008 ms    0/ 8 (0%)
[ 3] 1.0- 2.0 sec   12.9 KBytes   106 Kbits/sec   0.011 ms    0/ 9 (0%)
[ 3] 2.0- 3.0 sec   11.5 KBytes   94.1 Kbits/sec   0.011 ms    0/ 8 (0%)
[ 3] 3.0- 4.0 sec   12.9 KBytes   106 Kbits/sec   0.012 ms    0/ 9 (0%)
[ 3] 4.0- 5.0 sec   11.5 KBytes   94.1 Kbits/sec   0.013 ms    0/ 8 (0%)
[ 3] 5.0- 6.0 sec   12.9 KBytes   106 Kbits/sec   0.015 ms    0/ 9 (0%)
[ 3] 6.0- 7.0 sec   11.5 KBytes   94.1 Kbits/sec   0.019 ms    0/ 8 (0%)
[ 3] 7.0- 8.0 sec   12.9 KBytes   106 Kbits/sec   0.019 ms    0/ 9 (0%)
[ 3] 8.0- 9.0 sec   11.5 KBytes   94.1 Kbits/sec   0.020 ms    0/ 8 (0%)
[ 3] 9.0-10.0 sec   12.9 KBytes   106 Kbits/sec   0.020 ms    0/ 9 (0%)
[ 3] 0.0-10.2 sec   125 KBytes    100 Kbits/sec   0.021 ms    0/ 87 (0%)

root@open5g: ~
File Edit View Search Terminal Help
root@open5g:~# iperf -c 127.0.0.1 -u -b 0.1M --bind 127.0.0.1
bind failed: Address already in use
-----
Client connecting to 127.0.0.1, UDP port 5001
Binding to local address 127.0.0.1
Sending 1470 byte datagrams
UDP buffer size: 208 KByte (default)
-----
[ 3] local 127.0.0.1 port 46206 connected with 127.0.0.1 port 5001
[ ID] Interval      Transfer      Bandwidth
[ 3] 0.0-10.2 sec   125 KBytes    100 Kbits/sec
[ 3] Sent 87 datagrams
[ 3] Server Report:
[ 3] 0.0-10.2 sec   125 KBytes    100 Kbits/sec   0.021 ms    0/ 87 (0%)
```