教育部 5G 行動寬頻人才培育跨校教學聯盟計畫

5G 行動網路協定與核網技術聯盟中心示範課程

4G/5G 行動寬頻協同網路

實驗二
DC 效能量測與分析

副教授：吳俊興

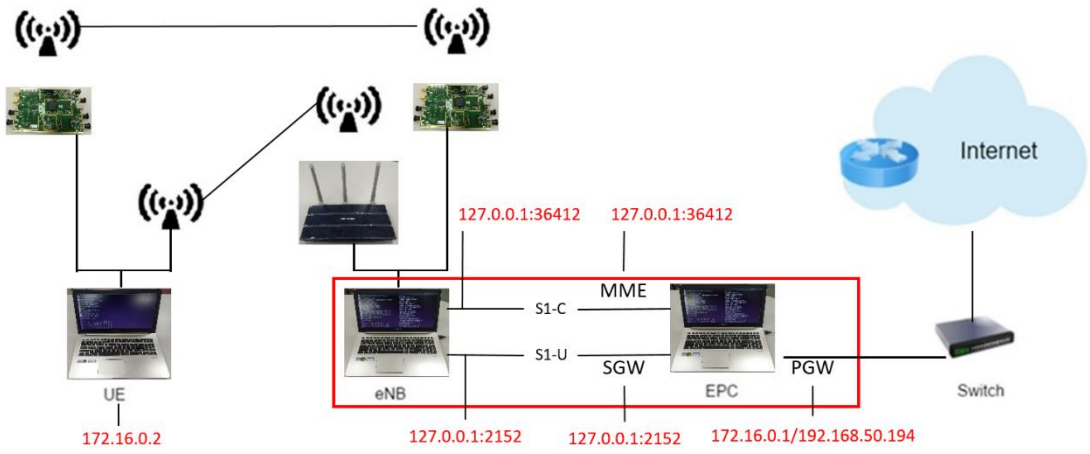
助教：林原進、吳振宇

國立高雄大學 資訊工程學系

目錄

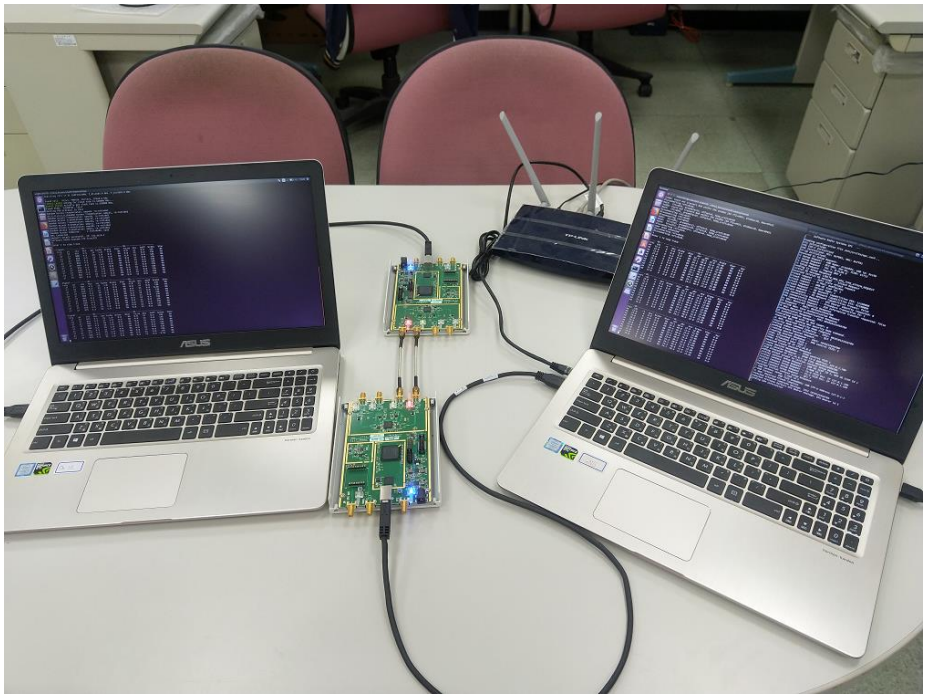
一、 平台架構

1. 實驗架構



本實驗架構如圖所示，共分成兩個部分，第一部分由 EPC 和 eNB 組成，透過程式指定 ip 位址由同一台電腦開啟，利用乙太網路線連接外網，第二部分為 UE，兩台電腦皆有連接 USRP 並且透過 SMA 線對連，同時連上 HUB-Wifi。

2. 實驗環境



二、 軟硬體需求

1. 硬體

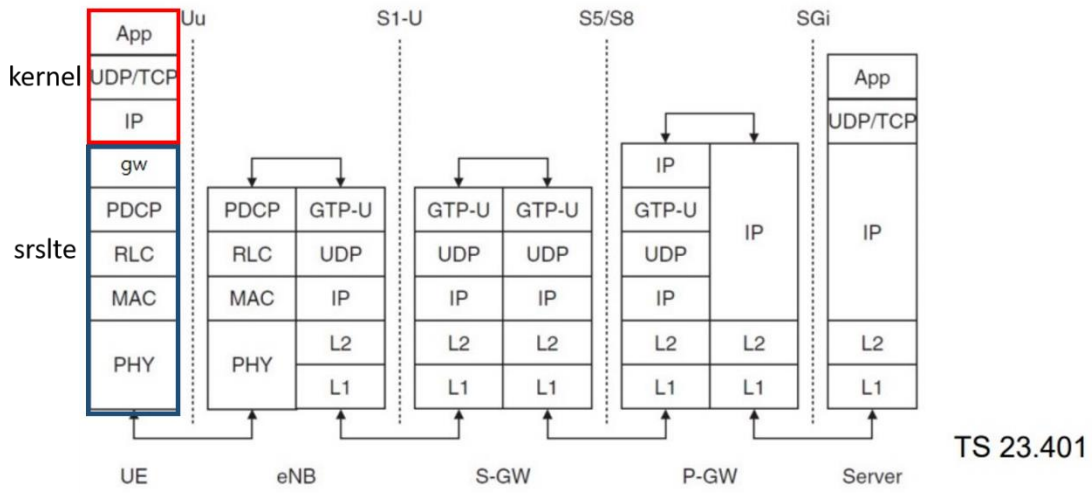
名稱	規格	數量	目的
EPC+eNB	電腦型號： ASUS NB M580V	1	啟動 MME,S-GW,P-GW
	USRP B210	1	啟動 srsLTE eNB
UE	電腦型號： ASUS NB M580V	1	模擬 UE
	USRP B210	1	啟動 srsLTE UE
無線分享器	TP-LINK TL-WR1043ND	1	實現無線分享器功能

2. 軟體

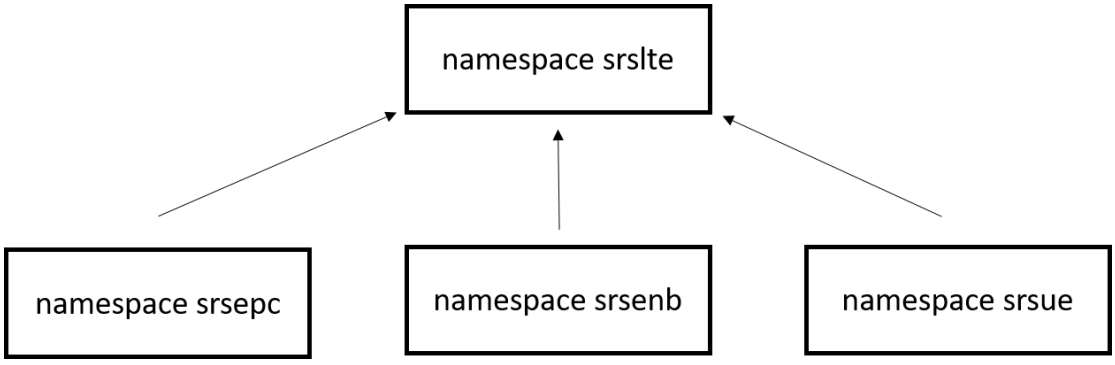
名稱	軟體	版本	目的
EPC+ eNB	Ubuntu	Ubuntu 16.04	啟動 HSS, MME, S-GW, P-GW 功能
		Kernel: linux-image-4.13.16-041316-lowlatency	
	srsLTE	srsLTE 18.6.1 470953bf9c5875646e4d5049c8f213d202fa84fd	
UE	Ubuntu	Ubuntu 16.04	啟動 UE 功能
		Kernel: linux-image-4.13.16-041316-lowlatency	
	srsLTE	srsLTE 18.6.1 470953bf9c5875646e4d5049c8f213d202fa84fd	
PC	Wireshark	2.6.8 (Ubuntu)	觀察協定
	Iperf	3.1.3 (Client) 3.1.3(Server)	測量網路頻寬

三、 介紹 srsLTE 架構

1. Data Flow



2. 程式結構



srsepc、srsenb、srsue 有一些程式碼是共用的，共用的程式碼會寫在 namespace srslte，當有需要使用的時候會呼叫 srslte 裏面的程式碼。

2.1. pdcp.cc

檔案位置 ：srsenb\src\upper\pdcp.cc

程式碼 ：

```
void pdcp::init(rlc_interface_pdcp* rlc_, …, srslte::log* pdcp_log_)
{
    rlc     = rlc_;
    rrc     = rrc_;
    gtpu    = gtpu_;
    log_h = pdcp_log_;
    pool = srslte::byte_buffer_pool::get_instance();
    pthread_rwlock_init(&rwlock, NULL);
```

}

srsenb 界面的運作部份會寫在 srsenb\src\，同時如果 srsenb 的界面有需要使用其他層的 function，它在初始化的時候會取得其他層的界面，有需要時再呼叫其他層的界面來使用其他層的 function。

pdcp.init 會呼叫 srsenb\src\upper\pdcp.cc 裏面的 init()

2.2. enb.cc

檔案位置 : srsenb\src\enb.cc

程式碼 :

```
namespace srsenb
{
	bool enb::init(all_args_t *args_)
	{
		pdcp_log.init("PDCP ", logger);
		pdcp_log.set_level(level(args->log.pdcp_level));
		pdcp_log.set_hex_limit(args->log.pdcp_hex_limit);
		pdcp.init(&rlc, &rrc, &gtpu, &pdcp_log);
	}
}
```

srsenb 所有程式的界面初始化會在 srsenb\src\enb.cc 開始

2.3. ue_interface.h

檔案位置 : srslte\lib\include\srslte\interfaces.ue_interface.h

程式碼 :

```
namespace srsue
{
	class pdcp_interface_rrc{…};// pdcp function for rrc
	class rlc_interface_pdcp{…};// rlc function for pdcp
}
```

UE 所有的界面會寫在 ue_interfaces.h

2.4. ue.cc

檔案位置 : srsue\src\ue.cc

程式碼 :

```
namespace srsue
{
	bool ue::init(all_args_t *args_)
	{
```

```
            pdcp_log.init("PDCP ", logger);
            pdcp_log.set_level(level(args->log.pdcp_level));
            pdcp_log.set_hex_limit(args->log.pdcp_hex_limit);
            pdcp.init(&rlc, &rrc, &gw, &pdcp_log, …);
        }
}
```

srsue 所有程式的界面初始化會在 srsue\src\ue.cc 開始

3. 封包流程

3.1. eNB 封包流程

```
UE -> eNB -> EPC        //eNB 收到從 UE 收到封包，rlc 層收到封包
srsenb::pdcp::write_pdu()      //rlc 呼叫 pdcp 的界面把封包送到 pdcp 層
-|-> srslte::pdcp::write_pdu() //enb 的 pdcp 界面呼叫 srslte 的 pdcp 界面
   |-> srslte::pdcp_entity::write_pdu()   // srslte 的 pdcp 界面再呼叫運作程式
     |-> srsenb::gtpu::write_pdu()   //pdcp 再呼叫 gtpu 界面

UE <- eNB <- EPC        //eNB 收到從 EPC 收到封包，gtpu 層收到封包
srsenb::gtpu::run_thread(){   recv();   }       //gtpu 層收到 EPC 送來的封包
-|-> srsenb::pdcp::write_sdu()//gtpu 層呼叫 pdcp 的界面把封包送到 pdcp 層
   |-> srslte::pdcp::write_sdu()      //enb 的 pdcp 界面呼叫 srslte 的 pdcp 界面
     |-> srslte::pdcp_entity::write_sdu()   //srslte 的 pdcp 界面再呼叫運作程式
       |-> srsenb::rlc::write_sdu()      //pdcp 再呼叫 rlc 界面
         |->…
```

3.2. UE 封包流程

```
UE -> eNB                //UE 把封包送到 eNB
srsue::gw::run_thread()        //UE 取得封包
-|-> srslte::pdcp::write_sdu()  //gw 層呼叫 pdcp 界面並把封包送到 pdcp 層
   |-> srslte::pdcp_entity::write_sdu()
     |->…
UE <- eNB                //UE 底層收到 eNB 的封包
|->...
  |-> srslte::pdcp::write_pdu() //底層呼叫 pdcp 界面並把封包送到 pdcp 層
   |-> srslte::pdcp_entity::write_pdu()//pdcp 界面呼叫 pdcp 運作程式
     |-> srsue::gw::write_pdu() {   write();   }//pdcp 呼叫 gw 層的界面
```

四、 環境安裝

1. Linux Kernel 安裝

1.1. 下載及安裝 Kernel

開啟一個終端機(Terminal)，並且依序輸入

wget -P ~/Downloads/kernel http://kernel.ubuntu.com/~kernel-ppa/mainline/v4.13.16/linux-headers-4.13.16-041316_4.13.16-041316.201711240901_all.deb
wget -P ~/Downloads/kernel http://kernel.ubuntu.com/~kernel-ppa/mainline/v4.13.16/linux-headers-4.13.16-041316-lowlatency_4.13.16-041316.201711240901_amd64.deb
wget -P ~/Downloads/kernel http://kernel.ubuntu.com/~kernel-ppa/mainline/v4.13.16/linux-image-4.13.16-041316-lowlatency_4.13.16-041316.201711240901_amd64.deb
sudo dpkg -i ~/Downloads/kernel/*.deb

1.2. 修改開機選單和設定

開啟終端機(Terminal)輸入以下指令

sudo gedit /etc/default/grub

找到下列文字

GRUB_HIDDEN_TIMEOUT=0

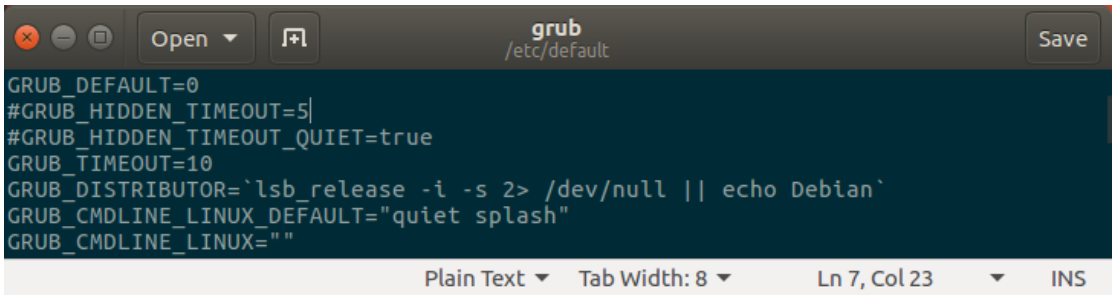
GRUB_HIDDEN_TIMEOUT_QUIET=true

改成

#GRUB_HIDDEN_TIMEOUT=0

#GRUB_HIDDEN_TIMEOUT_QUIET=true

儲存後離開



1.3. 更新 grub 設定

開啟終端機並輸入以下指令

```
sudo update-grub2
```

接著輸入以下指令，重新啟動電腦

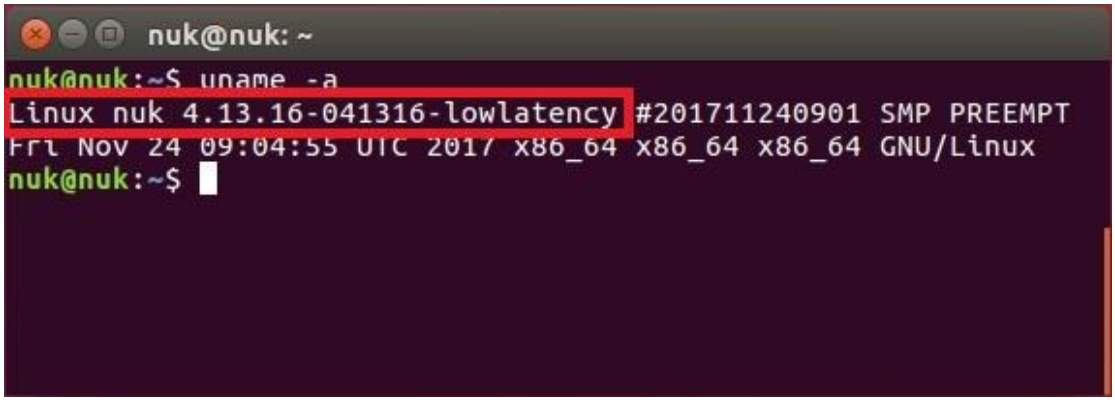
```
sudo reboot
```

然後在開機選單選擇剛才安裝的 lowlatency

1.4. 檢查 Kernel 版本

重新開機後在終端機輸入指令，確認版本

```
uname -r
```



2. 安裝相關套件

2.1. 一般套件

開啟終端機並輸入

```
sudo apt-get install cmake libfftw3-dev libboost-all-dev libconfig++-dev libsctp-dev
```

2.2. RF Front-end Driver

開啟終端機並依序輸入

```
sudo add-apt-repository ppa:ettusresearch/uhd
sudo apt-get update
sudo apt-get install libuhd-dev libuhd003 uhd-host
python3 /lib/uhd/utils/uhd_images_downloader.py
```

2.3. mbed TLS

開啟終端機並依序輸入

wget https://tls.mbed.org/download/start/mbedtls-2.6.0-apache.tgz
tar zxvf mbedtls-2.6.0-apache.tgz
sudo mv ~/Donwload/mbedtls-2.6.0 /usr/local
cd cd /usr/local/mbedtls-2.6.0
cmake .
make
make test
cmake -DENABLE_TESTING=Off .
cmake -DUSE_SHARED_MBEDTLS_LIBRARY=On .
sudo make install library

2.4. srsGUI

開啟終端機並依序輸入

sudo apt-get install libboost-system-dev libboost-test-dev libboost-thread-dev libqwt-dev libqt4-dev
git clone https://github.com/srsLTE/srsGUI.git
cd ~/srsgui
mkdir build
cd build
cmake ../
make
make test

2.5. srsLTE

開啟終端機並依序輸入

git clone https://github.com/nukcsie2066/nukxDC.git
cd srsLTE
mkdir build
cd build
cmake ../
make
make test
sudo make install

```
ue@ue-X580VD: ~/Desktop/srsLTE/build
ue@ue-X580VD:~$ cd ~/Desktop/srsLTE/
ue@ue-X580VD:~/Desktop/srsLTE$ mkdir build
ue@ue-X580VD:~/Desktop/srsLTE$ cd build/
ue@ue-X580VD:~/Desktop/srsLTE/build$ cmake ../
-- The C compiler identification is GNU 5.4.0
-- The CXX compiler identification is GNU 5.4.0
-- Check for working C compiler: /usr/bin/cc
-- Check for working C compiler: /usr/bin/cc -- works
-- Detecting C compiler ABI info
-- Detecting C compiler ABI info - done
-- Detecting C compile features
-- Detecting C compile features - done
-- Check for working CXX compiler: /usr/bin/c++
-- Check for working CXX compiler: /usr/bin/c++ -- works
-- Detecting CXX compiler ABI info
-- Detecting CXX compiler ABI info - done
-- Detecting CXX compile features
-- Detecting CXX compile features - done
-- CMAKE_SYSTEM: Linux-4.15.0-51-generic
-- CMAKE_SYSTEM_PROCESSOR: x86_64
-- CMAKE_CXX_COMPILER: /usr/bin/c++
-- Build type not specified: defaulting to Release.
-- Looking for pthread.h
-- Looking for pthread.h - found
-- Looking for pthread_create
-- Looking for pthread_create - not found
```


```
ue@ue-X580VD: ~/Desktop/srsLTE/build
ue@ue-X580VD:~$ cd ~/Desktop/srsLTE/
ue@ue-X580VD:~/Desktop/srsLTE$ mkdir build
ue@ue-X580VD:~/Desktop/srsLTE$ cd build/
ue@ue-X580VD:~/Desktop/srsLTE/build$ cmake ../
-- The C compiler identification is GNU 5.4.0
-- The CXX compiler identification is GNU 5.4.0
-- Check for working C compiler: /usr/bin/cc
-- Check for working C compiler: /usr/bin/cc -- works
-- Detecting C compiler ABI info
-- Detecting C compiler ABI info - done
-- Detecting C compile features
-- Detecting C compile features - done
-- Check for working CXX compiler: /usr/bin/c++
-- Check for working CXX compiler: /usr/bin/c++ -- works
-- Detecting CXX compiler ABI info
-- Detecting CXX compiler ABI info - done
-- Detecting CXX compile features
-- Detecting CXX compile features - done
-- CMAKE_SYSTEM: Linux-4.15.0-51-generic
-- CMAKE_SYSTEM_PROCESSOR: x86_64
-- CMAKE_CXX_COMPILER: /usr/bin/c++
-- Build type not specified: defaulting to Release.
-- Looking for pthread.h
-- Looking for pthread.h - found
-- Looking for pthread_create
-- Looking for pthread_create - not found
```


```
ue@ue-X580VD: ~/Desktop/srsLTE/build
ue@ue-X580VD:~/Desktop/srsLTE/build$ sudo make install
[sudo] password for ue:
[  1%] Built target rrc_asn1
[  2%] Built target srslte_asn1
-- Generating build_info.h
[  2%] Built target gen_build_info
[  7%] Built target srslte_common
[  8%] Built target arch_select
[  9%] Built target srslte_enb
[ 10%] Built target srslte_agc
[ 11%] Built target srslte_ch_estimation
[ 12%] Built target srslte_phy_common
[ 17%] Built target srslte_fec
[ 17%] Built target srslte_mimo
[ 22%] Built target srslte_phch
[ 24%] Built target srslte_sync
[ 27%] Built target srslte_utils
[ 28%] Built target srslte_channel
[ 29%] Built target srslte_dft
[ 30%] Built target srslte_io
[ 32%] Built target srslte_modem
[ 33%] Built target srslte_resampling
[ 34%] Built target srslte_scrambling
[ 35%] Built target srslte_ue
[ 35%] Built target srslte_phy
[ 35%] Built target refsignal_ul_test_all
```

3. 設定及編譯 srsLTE

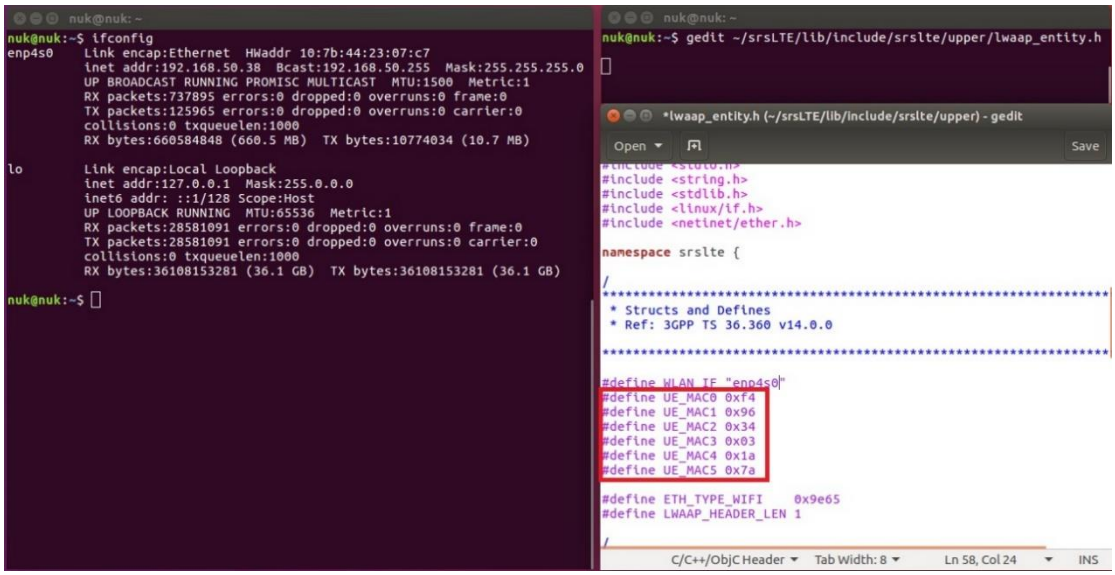
3.1. 設定 UE MAC

在 eNB 的終端機輸入以下指令

```
gedit /path/to/srsLTE/lib/include/srslte/upper/lwaap_entity.h
```

如下圖，在 eNB 主機上設定 LWA 的 DST MAC

把 UE 的 MAC 設成如下圖 header 樣式



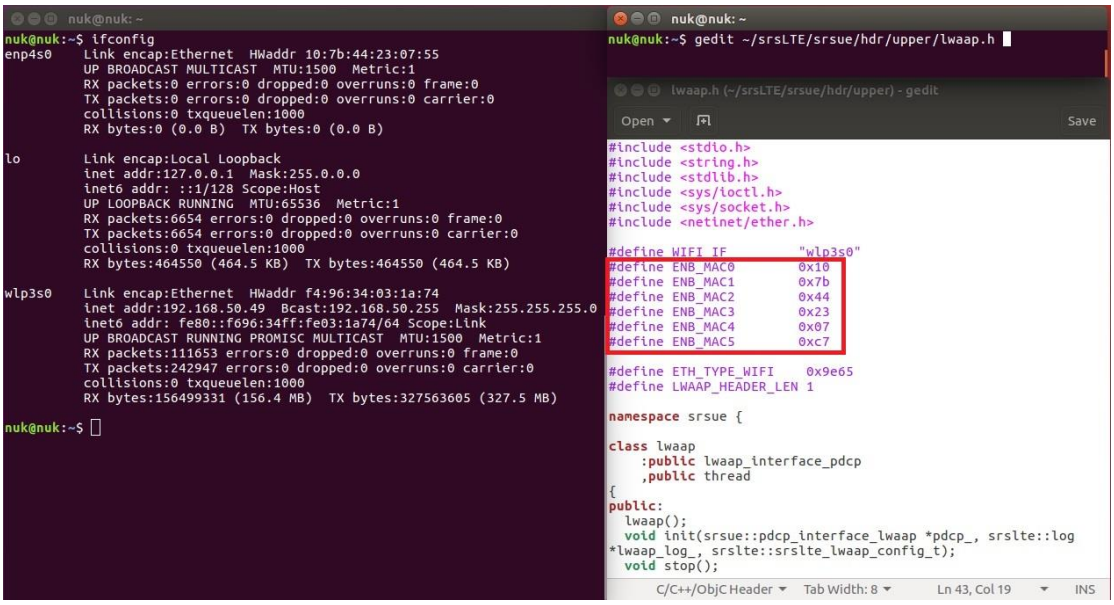
3.2. 設定 eNB MAC

在 UE 的終端機輸入以下指令

```
gedit /path/to/srsLTE/srsue/hdr/upper/lwaap.h
```

如下圖，在 UE 主機上設定 LWA 的 DST MAC

把 eNB 的 MAC 設成如下圖 header 樣式

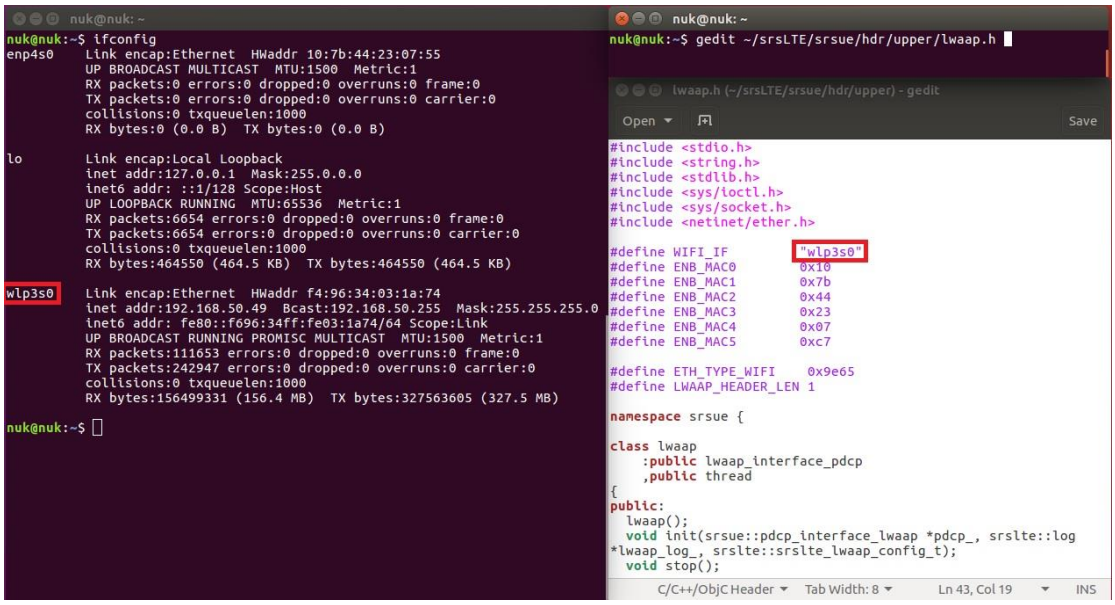


3.3. 設定 UE NIC Name

在 UE 的終端機輸入以下指令

```
gedit /path/to/srsLTE/srsue/hdr/upper/lwaap.h
```

如下圖，在 UE 設定 LWA 的網卡名稱

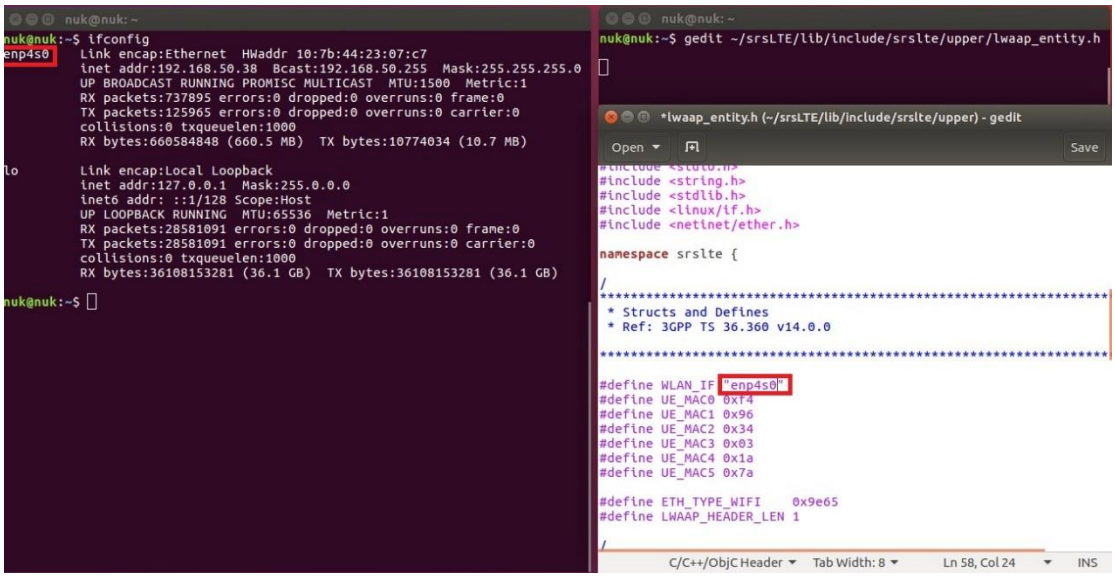


3.4. 設定 eNB NIC Name

在 eNB 的終端機輸入以下指令

```
gedit /path/to/srsLTE/lib/include/srslte/upper/lwaap_entity.h
```

如下圖，在 eNB 設定 LWA 的網卡名稱



3.5. 編譯 srsLTE

在 EPC、eNB 及 UE 的終端機輸入

cmake ../
cd /path/to/srsLTE/build
make
sudo make install
sudo ldconfig

3.6. 修改 srsEPC 設定檔

在 EPC 開啟終端機輸入指令

cd /path/to/srsLTE/srsepc
gedit epc.conf

mme:

tac、mcc、mnc 需要與 eNB 設定相同

```
1 ##########################################################################
2 #                    srsEPC configuration file
3 ##########################################################################
4
5 ##########################################################################
6 # MME configuration
7 #
8 # mme_code:          8-bit MME code identifies the MME within a group.
9 # mme_group:         16-bit MME group identifier.
10 # tac:               16-bit Tracking Area Code.
11 # mcc:               Mobile Country Code
12 # mnc:               Mobile Network Code
13 # apn:                     Set Access Point Name (APN)
14 # mme_bind_addr:     IP bind addr to listen for eNB S1-MME connnection
15 # dns_addr:          DNS server address for the UEs
16 #
17 ##########################################################################
18 [mme]
19 mme_code = 0x1a
20 mme_group = 0x0001
21 tac = 0x0007
22 mcc = 001
23 mnc = 01
24 mme_bind_addr = 192.168.10.12
25 apn = srsapn
26 dns_addr = 8.8.8.8
```

需與eNB設定相同

```
17 ##########################################################################
18 [mme]
19 mme_code = 0x1a
20 mme_group = 0x0001
21 tac = 0x0007
22 mcc = 001
23 mnc = 01
24 mme_bind_addr = 192.168.10.12
25 apn = srsapn
26 dns_addr = 8.8.8.8
27
28 ##########################################################################
29 # HSS configuration
30 #
31 # algo:            Authentication algorithm (xor/milenage)
32 # db_file:         Location of .csv file that stores UEs information.
33 #
34 ##########################################################################
35 [hss]
36 auth_algo = xor
37 db_file = user_db.csv
```

與eNB連接的IP和DNS

資料庫與認證演算法

spgw:

將 gtpu_bind_addr 設定成對外連接 IP

```
40 ##########################################################################
41 # SP-GW configuration
42 #
43 # gtpu_bind_addr:   GTP-U bind adress.
44 #
45 ##########################################################################
46
47 [spgw]
48 gtpu_bind_addr=192.168.50.194
49 sgi_if_addr=172.16.0.1
```

EPC對外連接IP

3.7. 修改 srsEPC 資料庫

在 EPC 開啟終端機輸入指令

cd /path/to/srsLTE/srsepc
gedit epc.conf

```
 2 # .csv to store UE's information in HSS
 3 # Kept in the following format: "Name,IMSI,Key,OP_Type,OP,AMF,SQN,QCI"
 4 #
 5 # Name:    Human readable name to help distinguish UE's. Ignored by the HSS
 6 # IMSI:    UE's IMSI value
 7 # Key:     UE's key, where other keys are derived from. Stored in hexadecimal
 8 # OP_Type: Operator's code type, either OP or OPc
 9 # OP/OPc:  Operator Code/Cyphered Operator Code, stored in hexadecimal
10 # AMF:     Authentication management field, stored in hexadecimal
11 # SQN:     UE's Sequence number for freshness of the authentication
12 # QCI:     QoS Class Identifier for the UE's default bearer.
13 #
14 # Note: Lines starting by '#' are ignored and will be overwritten
15 ue2,001010123456780,00112233445566778899aabbccddeeff,opc,63bfa50ee6523365ff14c1f45f88737d,8000,000000001234,7
16 ue1,001010123456789,00112233445566778899aabbccddeeff,opc,63bfa50ee6523365ff14c1f45f88737d,9001,00000000148b,7
```

7 RLC UM

9 RLC AM

UE預設SIM卡資訊

3.8. 修改 srseNB 設定檔

在 eNB 開啟終端機輸入指令

cd /path/to/srsLTE/srsenb
gedit enb.conf

```
1 ##########################################################################
2 #                      srsENB configuration file
3 ##########################################################################
4
5 ##########################################################################
6 # eNB configuration
7 #
8 # enb_id:           20-bit eNB identifier.
9 # cell_id:          8-bit cell identifier.
10 # tac:              16-bit Tracking Area Code.
11 # mcc:              Mobile Country Code
12 # mnc:              Mobile Network Code
13 # mme_addr:         IP address of MME for S1 connnection
14 # gtp_bind_addr:    Local IP address to bind for GTP connection
15 # n_prb:            Number of Physical Resource Blocks (6,15,25,50,75,100)
16 #
17 ##########################################################################
18 [enb]
19 enb_id = 0x19B
20 cell_id = 0x01
21 phy_cell_id = 1
22 tac = 0x0001
23 mcc = 001
24 mnc = 01
25 mme_addr = 192.168.10.254
26 gtp_bind_addr = 192.168.10.12
27 n_prb = 25
28
```

需與EPC設定相同

```
1 ##########################################################################
2 #                      srsENB configuration file
3 ##########################################################################
4
5 ##########################################################################
6 # eNB configuration
7 #
8 # enb_id:           20-bit eNB identifier.
9 # cell_id:          8-bit cell identifier.
10 # tac:              16-bit Tracking Area Code.
11 # mcc:              Mobile Country Code
12 # mnc:              Mobile Network Code
13 # mme_addr:         IP address of MME for S1 connnection
14 # gtp_bind_addr:    Local IP address to bind for GTP connection
15 # n_prb:            Number of Physical Resource Blocks (6,15,25,50,75,100)
16 #
17 ##########################################################################
18 [enb]
19 enb_id = 0x19B
20 cell_id = 0x01
21 phy_cell_id = 1
22 tac = 0x0001
23 mcc = 001
24 mnc = 01
25 mme_addr = 192.168.10.254
26 gtp_bind_addr = 192.168.10.12
27 n_prb = 25
```

mme_addr MME的IP位址
gtp_bind_addr eNB與EPC連接的IP位址

3.9. 修改 srsUE 設定檔

在 UE 開啟終端機輸入指令

cd /path/to/srsLTE/srsue
gedit ue.conf

```
[rf]
dl_earfcn = 500
freq_offset = 0
tx_gain = 60
rx_gain = 40

#nof_rx_ant = 1
#device_name = auto
#device_args = auto
#time_adv_nsamples = auto
#burst_preamble_us = auto
#continuous_tx     = auto
```

設定頻段(請參考下列網址)

調整收送功率(請參考之後投影片)

Band	Name	Downlink (MHz)			Bandwidth DL/UL (MHz)	Uplink (MHz)			Duplex spacing (MHz)	Geographical area	3GPP release
		Low	Middle	High		Low	Middle	High			
		Earfcn				Earfcn					
1	2100	2110	**2140**	2170	60	1920	**1950**	1980	190	Global	8
		0	300	599		18000	18300	18599			

資料來源：http://niviuk.free.fr/lte_band.php

```
##################################################################
# USIM configuration
#
# mode: USIM mode (soft/pcsc)
# algo: Authentication algorithm (xor/milenage)
# op:   128-bit Operator Variant Algorithm Configuration Field (hex)
# k:    128-bit subscriber key (hex)
# imsi: 15 digit International Mobile Subscriber Identity
# imei: 15 digit International Mobile Station Equipment Identity
# pin:  PIN in case real SIM card is used
# reader: Specify card reader by it's name as listed by 'pcsc_scan'. If empty, try all available readers.
##################################################################
[usim]
mode = soft
algo = xor
opc  = 63BFA50EE6523365FF14C1F45F88737D
k    = 00112233445566778899aabbccddeeff
imsi = 001010123456789
imei = 353490069873319
#reader =
#pin  = 1234
```

需與資料庫設置相同

五、 srsLTE 測試

1. 執行 EPC

在 EPC 開一個新的終端機輸入指令

cd ~/path/to/srsLTE/srsepc
./srsepc_if_masq.sh enp4s0 #enp4s0 是本例使用的對外網卡名稱
sudo srsepc epc.conf

```
asus-medium@asusmedium-UN65H: ~/Desktop/lwa_enb/srsepc
asus-medium@asusmedium-UN65H:~$ cd ~/Desktop/lwa_enb/srsepc/
asus-medium@asusmedium-UN65H:~/Desktop/lwa_enb/srsepc$ ./srsepc_if_masq.sh wlp3s0
[sudo] password for asus-medium:
Masquerading Interface wlp3s0
asus-medium@asusmedium-UN65H:~/Desktop/lwa_enb/srsepc$ sudo srsepc epc.conf

---  Software Radio Systems EPC  ---

Reading configuration file epc.conf...
HSS Initialized.
MME GTP-C Initialized
MME Initialized.
SP-GW Initialized.
```

2. 執行 eNB

在 eNB 再開一個新的終端機輸入

cd ~/path/to/srsLTE/srsenb
sudo srsenb enb.conf

```
asus-medium@asusmedium-UN65H: ~/Desktop/lwa_enb/srsenb
asus-medium@asusmedium-UN65H:~$ cd ~/Desktop/lwa_enb/srsenb/
asus-medium@asusmedium-UN65H:~/Desktop/lwa_enb/srsenb$ sudo srsenb enb.conf
[sudo] password for asus-medium:
---  Software Radio Systems LTE eNodeB  ---

Reading configuration file enb.conf...
[INFO] [UHD] linux; GNU C++ version 5.4.0 20160609; Boost_105800; UHD_3.14.
0.0-release
Opening USRP with args: type=b200,master_clock_rate=30.72e6
[INFO] [B200] Detected Device: B210
[INFO] [B200] Operating over USB 3.
[INFO] [B200] Initialize CODEC control...
[INFO] [B200] Initialize Radio control...
[INFO] [B200] Performing register loopback test...
[INFO] [B200] Register loopback test passed
[INFO] [B200] Performing register loopback test...
[INFO] [B200] Register loopback test passed
[INFO] [B200] Asking for clock rate 30.720000 MHz...
[INFO] [B200] Actually got clock rate 30.720000 MHz.
Setting frequency: DL=2160.0 Mhz, UL=1970.0 MHz
[INFO] [B200] Asking for clock rate 23.040000 MHz...
[INFO] [B200] Actually got clock rate 23.040000 MHz.
Setting Sampling frequency 5.76 MHz

==== eNodeB started ===
Type <t> to view trace
```

3. 執行 UE

在 UE 開一個新的終端機輸入

cd ~/path/to/srsLTE/srsue
sudo srsue ue.conf

```
ue@ue-X580VD: ~/Desktop/lwaap_ue/srsue
ue@ue-X580VD:~$ cd ~/Desktop/lwaap_ue/srsue/
ue@ue-X580VD:~/Desktop/lwaap_ue/srsue$ sudo srsue ue.conf
[sudo] password for ue:
Reading configuration file ue.conf...

Built in Release mode using commit 0a69e56 on branch develop_ue.

Buffer capacity 10240
Buffer capacity 40960
---  Software Radio Systems LTE UE  ---

Opening RF device with 1 RX antennas...
[INFO] [UHD] linux; GNU C++ version 5.4.0 20160609; Boost_105800; UHD_3.14.0.0-r
elease
Opening USRP with args: type=b200,master_clock_rate=30.72e6
[INFO] [B200] Detected Device: B210
[INFO] [B200] Operating over USB 3.
[INFO] [B200] Initialize CODEC control...
[INFO] [B200] Initialize Radio control...
[INFO] [B200] Performing register loopback test...
[INFO] [B200] Register loopback test passed
[INFO] [B200] Performing register loopback test...
[INFO] [B200] Register loopback test passed
[INFO] [B200] Asking for clock rate 30.720000 MHz...
[INFO] [B200] Actually got clock rate 30.720000 MHz.
LWAAP MAC f4:96:34:3:6a:a6
LWAAP IP packet receiver thread run_enable
Waiting PHY to initialize...
...
Attaching UE...
Searching cell in DL EARFCN=500, f_dl=2160.0 MHz, f_ul=1970.0 MHz
.
Found Cell:  PCI=1, PRB=25, Ports=1, CFO=0.5 KHz
[INFO] [B200] Asking for clock rate 23.040000 MHz...
[INFO] [B200] Actually got clock rate 23.040000 MHz.
Found PLMN:  Id=00101, TAC=7
Random Access Transmission: seq=9, ra-rnti=0x2
Random Access Transmission: seq=42, ra-rnti=0x2
Random Access Transmission: seq=9, ra-rnti=0x2
RRC Connected
Random Access Complete.     c-rnti=0x48, ta=0
Network attach successful. IP: 172.16.0.2
Software Radio Systems LTE (srsLTE)
```

3.1. UE Attach 成功

EPC :

```
root@NUK: /home/enb/srsLTE-eNB_LWIP/srsepc
SPGW Allocated IP 172.16.0.2 to ISMI 001010123456789
Adding attach accept to Initial Context Setup Request
Initial Context Setup Request -- eNB UE S1AP Id 1, MME UE S1AP Id 1
Initial Context Setup Request -- E-RAB id 5
Initial Context Setup Request -- S1-U TEID 0x1. IP 192.168.50.194
Initial Context Setup Request -- S1-U TEID 0x1. IP 192.168.50.194
Initial Context Setup Request -- QCI 9
Received Initial Context Setup Response
E-RAB Context Setup. E-RAB id 5
E-RAB Context -- eNB TEID 0x460003; eNB GTP-U Address 127.0.0.1
Integrity Protected UL NAS: Received Attach Complete
Unpacked Attached Complete Message. IMSI 1010123456789
Unpacked Activate Default EPS Bearer message. EPS Bearer id 5
Packing EMM Information
Sending EMM Information, bytes 67
DL NAS: Sent Downlink NAS Message. DL NAS Count=2, UL NAS count=1
DL NAS: MME UE S1AP id 1
SCTP Association Shutdown. Association: 128
Deleting eNB context. eNB Id: 0x19b
Releasing UEs context
Releasing UE ECM context. UE-MME S1AP Id: 1
```

UE :

```
nuk@nuk: ~/srsLTE-eNB_LWIP/srsue
[INFO] [B200] Register loopback test passed
[INFO] [B200] Asking for clock rate 30.720000 MHz...
[INFO] [B200] Actually got clock rate 30.720000 MHz.

lwipep lib init rnti = 0x0
lwipep MAC f4:96:34:3:1a:74
Waiting PHY to initialize...
...
Attaching UE...
Searching cell in DL EARFCN=500, f_dl=2160.0 MHz, f_ul=1970.0 MHz
.
Found Cell:  PCI=1, PRB=50, Ports=1, CFO=-0.8 KHz
[INFO] [B200] Asking for clock rate 11.520000 MHz...
[INFO] [B200] Actually got clock rate 11.520000 MHz.
Found PLMN:  Id=00101, TAC=1
Random Access Transmission: seq=5, ra-rnti=0x2
RRC Connected
Random Access Complete.     c-rnti=0x46, ta=18
lwipep rnti = 0x46
Network attach successful. IP: 172.16.0.2
Software Radio Systems LTE (srsLTE)
```

3.2. UE Attach 失敗

```
nuk_lab@lab: ~/srsLTE/srsue
[INFO] [CORES] Performing timer loopback test...
[INFO] [CORES] Timer loopback test passed
[INFO] [CORES] Performing timer loopback test...
[INFO] [CORES] Timer loopback test passed
LWAAP MAC f4:96:34:3:66:5a
Waiting PHY to initialize...
...
Attaching UE...
Searching cell in DL EARFCN=500, f_dl=2160.0 MHz, f_ul=1970.0 MHz
.
Found Cell:  PCI=1, PRB=25, Ports=1, CFO=-1.7 KHz
[INFO] [B200] Asking for clock rate 23.040000 MHz...
[INFO] [B200] Actually got clock rate 23.040000 MHz.
[INFO] [CORES] Performing timer loopback test...
[INFO] [CORES] Timer loopback test passed
[INFO] [CORES] Performing timer loopback test...
[INFO] [CORES] Timer loopback test passed
Found PLMN:  Id=00101, TAC=1
Random Access Transmission: seq=9, ra-rnti=0x2
RRC Connected
Random Access Complete.     c-rnti=0x46, ta=0
Network attach successful. IP: 172.16.0.2
Software Radio Systems LTE (srsLTE)
```

PLMN不同：沒有找到eNB

沒有RRC Connected：與eNB連接失敗

沒有IP：與EPC連接失敗

3.3. 開啟後 UE 設定

在 UE 開一個新的終端機輸入

sudo route add default gw 172.16.0.2 tun_srsue
sudo gedit /etc/resolv.conf

彈出新的視窗 resolv.conf，如下圖所示來修改，然後關閉它

```
nuk@nuk: ~
nuk@nuk:~$ sudo route add default gw 172.16.0.2 tun_srsue
nuk@nuk:~$ sudo gedit /etc/resolv.conf
```

```
Open ▾   resolv.conf
         /etc                                   Save
# Dynamic resolv.conf(5) file for glibc resolver(3)
generated by resolvconf(8)
#     DO NOT EDIT THIS FILE BY HAND -- YOUR CHANGES WILL BE
OVERWRITTEN
nameserver 8.8.8.8
Plain Text ▾   Tab Width: 8 ▾   Ln 3, Col 12 ▾   INS
```

4. 流量測試

在 EPC 開一個新的終端機輸入

iperf3 -s -B 172.16.0.1

```
nuk@nuk: ~/iperf
nuk@nuk:~/iperf$ iperf3 -s -B 172.16.0.1
-----------------------------------------------------------
Server listening on 5201
-----------------------------------------------------------
Accepted connection from 172.16.0.2, port 44411
[  5] local 172.16.0.1 port 5201 connected to 172.16.0.2 port 38249
[ ID] Interval           Transfer     Bitrate         Total Datagrams
[  5]   0.00-1.00   sec  11.9 MBytes  99.9 Mbits/sec  8759
[  5]   1.00-2.00   sec  11.9 MBytes   100 Mbits/sec  8765
[  5]   2.00-3.00   sec  11.9 MBytes   100 Mbits/sec  8766
[  5]   3.00-4.00   sec  11.9 MBytes   100 Mbits/sec  8766
[  5]   4.00-5.00   sec  11.9 MBytes   100 Mbits/sec  8766
[  5]   5.00-6.00   sec  11.9 MBytes   100 Mbits/sec  8765
[  5]   6.00-7.00   sec  11.9 MBytes   100 Mbits/sec  8766
```

在 UE 開一個新的終端機輸入

```
iperf3 -c 172.16.0.1 -B 172.16.0.2 -u -l 1426b -t 120 -b 100m -R
```

```
nuk@nuk: ~

nuk@nuk:~$ iperf3 -c 172.16.0.1 -B 172.16.0.2 -l 1426b -t 120 -u -b 100m -R
Connecting to host 172.16.0.1, port 5201
Reverse mode, remote host 172.16.0.1 is sending
[  5] local 172.16.0.2 port 52864 connected to 172.16.0.1 port 5201
[ ID] Interval           Transfer     Bitrate         Jitter    Lost/Total Datagrams
[  5]   0.00-1.00   sec  1.82 MBytes  15.3 Mbits/sec  0.456 ms  7042/8382 (84%)
[  5]   1.00-2.00   sec  1.73 MBytes  14.5 Mbits/sec  0.435 ms  7488/8757 (86%)
[  5]   2.00-3.00   sec  1.73 MBytes  14.5 Mbits/sec  0.484 ms  7497/8767 (86%)
[  5]   3.00-4.00   sec  1.73 MBytes  14.5 Mbits/sec  0.434 ms  7504/8774 (86%)
[  5]   4.00-5.00   sec  1.73 MBytes  14.5 Mbits/sec  0.458 ms  7488/8757 (86%)
```

在 UE/eNB 的 Console 視窗內輸入 t 後按 Enter

```
enb@NUK: ~/srsLTE/srsenb
LWAAP add user rnti=0x46
User 0x46 connected
t
Enter t to stop trace.

------DL------------------------------------------------UL-------------------------------------------
rnti   cqi     ri    mcs   tbits   brate    bler    snr    phr    mcs   rbits   brate    bler     bsr
46    15.0      0    0.0     0.0     0.0      0%   29.8   40.0   15.0   12.8k   12.8k      0%     0.0
46    15.0      0    0.0     0.0     0.0      0%   29.8   40.0   15.0   12.8k   12.8k      0%     0.0
46    15.0      0    0.0     0.0     0.0      0%   29.8   40.0   15.0   12.8k   12.8k      0%     0.0
46    15.0      0    0.0     0.0     0.0      0%   29.4   40.0   15.0   12.8k   12.8k      0%     0.0
46    15.0      0    0.0     0.0     0.0      0%   29.2   40.0   15.0   12.8k   12.8k      0%     0.0
46    15.0      0    0.0     0.0     0.0      0%   29.4   40.0   15.0   12.8k   12.8k      0%     0.0
46    15.0      0    0.0     0.0     0.0      0%   29.6   40.0   15.0   12.8k   12.8k      0%     0.0
46    15.0      0    0.0     0.0     0.0      0%   29.8   40.0   15.0   12.8k   12.8k      0%     0.0
46    15.0      0    0.0     0.0     0.0      0%   29.8   40.0   15.0   12.8k   12.8k      0%     0.0
46    15.0      0    0.0     0.0     0.0      0%   29.7   40.0   15.0   12.8k   12.8k      0%     0.0
46    15.0      0    0.0     0.0     0.0      0%   30.0   40.0   15.0   12.8k   12.8k      0%     0.0

------DL------------------------------------------------UL-------------------------------------------
rnti   cqi     ri    mcs   tbits   brate    bler    snr    phr    mcs   rbits   brate    bler     bsr
46    15.0      0    0.0     0.0     0.0      0%   30.0   40.0   15.0   12.8k   12.8k      0%     0.0
46    15.0      0    0.0     0.0     0.0      0%   30.0   40.0   15.0   12.8k   12.8k      0%     0.0
```

數值越大傳越快(搭配iPerf測試)

```
nuk_lab@lab: ~/srsLTE/srsue
   -55     55   -1.8k    7.0     42    1.0     0.0     0.0      0%     15     0.0    13k    13k      0%
   -55     55   -1.8k    7.0     42    1.0     0.0     0.0      0%     15     0.0    13k    13k      0%
   -55     55   -1.8k    7.0     41    1.0     0.0     0.0      0%     15     0.0    13k    13k      0%
   -55     55   -1.6k    7.0     41    1.0     0.0     0.0      0%     15     0.0    13k    13k      0%
   -55     55   -1.8k    7.0     40    1.0     0.0     0.0      0%     15     0.0    13k    13k      0%

--Signal------------- DL- ---------------------------UL-----------------------------------------
  rsrp     pl     cfo    mcs    snr  turbo   bits   brate    bler    mcs    buff   bits    brate    bler
   -55     55   -1.8k    7.0     41    1.0     0.0     0.0      0%     15     0.0   8.5k    8.5k      0%
   -55     55   -1.8k    7.0     41    1.0     0.0     0.0      0%     15     0.0    13k     13k      0%
   -55     55   -1.9k    7.0     42    1.0     0.0     0.0      0%     15     0.0    13k     13k      0%
   -55     55   -1.8k    7.0     41    1.0     0.0     0.0      0%     15     0.0    13k     13k      0%
   -55     55   -1.9k    7.0     41    1.0     0.0     0.0      0%     15     0.0    13k     13k      0%
   -55     55   -1.9k    7.0     41    1.0     0.0     0.0      0%     15     0.0    13k     13k      0%
   -55     55   -1.9k    7.0     41    1.0     0.0     0.0      0%     15     0.0    13k     13k      0%
   -55     55   -1.9k    7.0     42    1.0     0.0     0.0      0%     15     0.0    13k     13k      0%
   -55     55   -1.8k    7.0     41    1.0     0.0     0.0      0%     15     0.0    13k     13k      0%
   -55     55   -1.6k    7.0     41    1.0     0.0     0.0      0%     15     0.0    13k     13k      0%

--Signal------------- DL- ---------------------------UL-----------------------------------------
  rsrp     pl     cfo    mcs    snr  turbo   bits   brate    bler    mcs    buff   bits    brate    bler
   -55     55   -1.6k    7.0     40    1.0     0.0     0.0      0%     15     0.0    13k     13k      0%
```

數值盡量相近

5. MCS

UE 測量 PRB(Physical Resource Block)

- 接收功率和干擾得到 SINR 值，在 BLER 值不超過 10%
- 將測量值轉換成 CQI
- eNodeB 會根據 CQI 值選擇最合適的 MCS

LTE 傳輸效能通過 MCS（Modulation and Coding Scheme，調製與編碼策略）速率表來決定

5.1. Download Link

MCS Table

< 36.213 Table 7.1.7.1-1 >

MCS Index I_{MCS}	Modulation Order Q_m	TBS Index I_{TBS}
0	2	0
1	2	1
2	2	2
3	2	3
4	2	4
5	2	5
6	2	6
7	2	7
8	2	8
9	2	9
10	4	9
11	4	10
12	4	11
13	4	12
14	4	13
15	4	14
16	4	15
17	6	15
18	6	16
19	6	17
20	6	18
21	6	19
22	6	20
23	6	21
24	6	22
25	6	23
26	6	24
27	6	25
28	6	26
29	2	reserved
30	4	
31	6	

TBS Table

Table 7.1.7.2.1-1: Transport block size table (dimension 27×110)

I_{TBS}	N_{PRB}									
	1	2	3	4	5	6	7	8	9	10
0	16	32	56	88	120	152	176	208	224	256
1	24	56	88	144	176	208	224	256	328	344
2	32	72	144	176	208	256	296	328	376	424
3	40	104	176	208	256	328	392	440	504	568
4	56	120	208	256	328	408	488	552	632	696
5	72	144	224	328	424	504	600	680	776	872
6	328	176	256	392	504	600	712	808	936	1032
7	104	224	328	472	584	712	840	968	1096	1224
8	120	256	392	536	680	808	968	1096	1256	1384
9	136	296	456	616	776	936	1096	1256	1416	1544
10	144	328	504	680	872	1032	1224	1384	1544	1736
11	176	376	584	776	1000	1192	1384	1608	1800	2024
12	208	440	680	904	1128	1352	1608	1800	2024	2280
13	224	488	744	1000	1256	1544	1800	2024	2280	2536
14	256	552	840	1128	1416	1736	1992	2280	2600	2856
15	280	600	904	1224	1544	1800	2152	2472	2728	3112
16	328	632	968	1288	1608	1928	2280	2600	2984	3240
17	336	696	1064	1416	1800	2152	2536	2856	3240	3624
18	376	776	1160	1544	1992	2344	2792	3112	3624	4008
19	408	840	1288	1736	2152	2600	2984	3496	3880	4264
20	440	904	1384	1864	2344	2792	3240	3752	4136	4584
21	488	1000	1480	1992	2472	2984	3496	4008	4584	4968
22	520	1064	1608	2152	2664	3240	3752	4264	4776	5352
23	552	1128	1736	2280	2856	3496	4008	4584	5160	5736
24	584	1192	1800	2408	2984	3624	4264	4968	5544	5992
25	616	1256	1864	2536	3112	3752	4392	5160	5736	6200
26	712	1480	2216	2984	3752	4392	5160	5992	6712	7480

5.2. Upload Link

MCS Table

< 36.213 Table 8.6.1-1 >

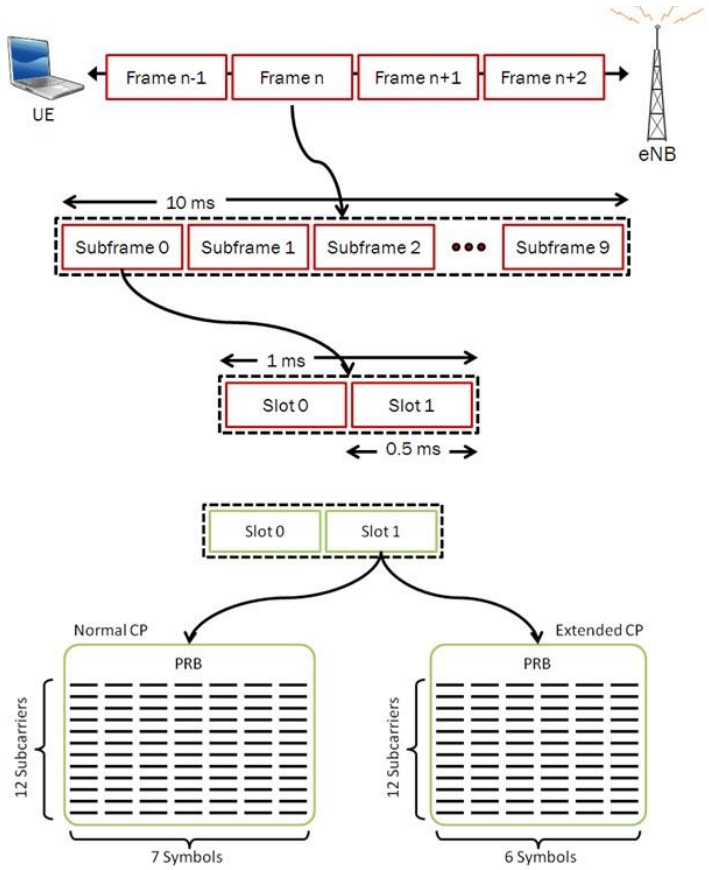
MCS Index I_{MCS}	Modulation Order Q'_m	TBS Index I_{TBS}	Redundancy Version rv_{idx}
0	2	0	0
1	2	1	0
2	2	2	0
3	2	3	0
4	2	4	0
5	2	5	0
6	2	6	0
7	2	7	0
8	2	8	0
9	2	9	0
10	2	10	0
11	4	10	0
12	4	11	0
13	4	12	0
14	4	13	0
15	4	14	0
16	4	15	0
17	4	16	0
18	4	17	0
19	4	18	0
20	4	19	0
21	6	19	0
22	6	20	0
23	6	21	0
24	6	22	0
25	6	23	0
26	6	24	0
27	6	25	0
28	6	26	0
29	reserved		1
30			2
31			3

TBS Table

Table 7.1.7.2.1-1: Transport block size table (dimension 27×110)

I_{TBS}	N_{PRB}									
	1	2	3	4	5	6	7	8	9	10
0	16	32	56	88	120	152	176	208	224	256
1	24	56	88	144	176	208	224	256	328	344
2	32	72	144	176	208	256	296	328	376	424
3	40	104	176	208	256	328	392	440	504	568
4	56	120	208	256	328	408	488	552	632	696
5	72	144	224	328	424	504	600	680	776	872
6	328	176	256	392	504	600	712	808	936	1032
7	104	224	328	472	584	712	840	968	1096	1224
8	120	256	392	536	680	808	968	1096	1256	1384
9	136	296	456	616	776	936	1096	1256	1416	1544
10	144	328	504	680	872	1032	1224	1384	1544	1736
11	176	376	584	776	1000	1192	1384	1608	1800	2024
12	208	440	680	904	1128	1352	1608	1800	2024	2280
13	224	488	744	1000	1256	1544	1800	2024	2280	2536
14	256	552	840	1128	1416	1736	1992	2280	2600	2856
15	280	600	904	1224	1544	1800	2152	2472	2728	3112
16	328	632	968	1288	1608	1928	2280	2600	2984	3240
17	336	696	1064	1416	1800	2152	2536	2856	3240	3624
18	376	776	1160	1544	1992	2344	2792	3112	3624	4008
19	408	840	1288	1736	2152	2600	2984	3496	3880	4264
20	440	904	1384	1864	2344	2792	3240	3752	4136	4584
21	488	1000	1480	1992	2472	2984	3496	4008	4584	4968
22	520	1064	1608	2152	2664	3240	3752	4264	4776	5352
23	552	1128	1736	2280	2856	3496	4008	4584	5160	5736
24	584	1192	1800	2408	2984	3624	4264	4968	5544	5992
25	616	1256	1864	2536	3112	3752	4392	5160	5736	6200
26	712	1480	2216	2984	3752	4392	5160	5992	6712	7480

6. FDD Throughput



RE = Symbols * (PRB * Subcarries)

CR = (TBS * CRC) / (RE * Bits per RE)

- TBS 請至 3GPP 查詢
- CRC = Throughput = TBS * CR
- Bits per RE = Modulation scheme

Throughput = TBS * CR

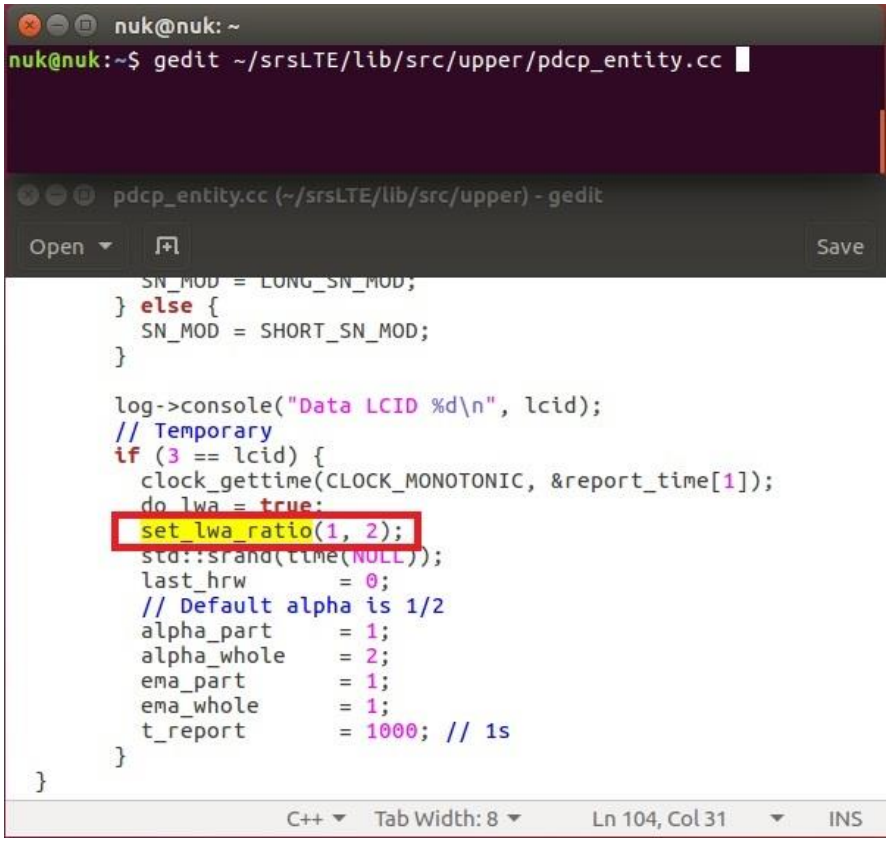
六、 實驗

1. 傳輸比例

1.1. 設定 LTE 與 WLAN 比例

在 eNB 的終端機輸入

gedit /path/to/srsLTE/lib/src/upper/pdcp_entity.cc



set_lwa_ratio(x,y)更改數字，請注意設為非負整數(0,1,2,3…)。
圖中所示為 1:2 代表 LTE:WLAN 比例。

1.2. 重新編譯 srsLTE

在 eNB 的終端機輸入

cd /path/to/srsLTE/build
cmake ../
make
sudo make install
sudo ldconfig

```
nuk@nuk: ~/srsLTE/build
nuk@nuk:~/srsLTE/build$ cmake ../
nuk@nuk:~/srsLTE/build$ make
nuk@nuk:~/srsLTE/build$ sudo make isntall
nuk@nuk:~/srsLTE/build$ sudo ldconfig
nuk@nuk:~/srsLTE/build$
```

1.3. 執行 srsEPC

在 EPC 開一個新的終端機輸入

cd ~/path/to/srsLTE/srsepc
./srsepc_if_masq.sh enp4s0
sudo srsepc epc.conf

enp4s0 是本例使用的對外網卡名稱，請自行查詢系統所使用的網卡名稱，因不同的系統或是硬體可能會有不同的名稱。

```
asus-medium@asusmedium-UN65H: ~/Desktop/lwa_enb/srsepc
asus-medium@asusmedium-UN65H:~$ cd ~/Desktop/lwa_enb/srsepc/
asus-medium@asusmedium-UN65H:~/Desktop/lwa_enb/srsepc$ ./srsepc_if_masq.sh wlp3s0
[sudo] password for asus-medium:
Masquerading Interface wlp3s0
asus-medium@asusmedium-UN65H:~/Desktop/lwa_enb/srsepc$ sudo srsepc epc.conf

---  Software Radio Systems EPC  ---

Reading configuration file epc.conf...
HSS Initialized.
MME GTP-C Initialized
MME Initialized.
SP-GW Initialized.
```

1.4. 執行 srseNB

在 eNB 再開一個新的終端機輸入

cd ~/path/to/srsLTE/srsenb
sudo srsenb enb.conf

```
asus-medium@asusmedium-UN65H: ~/Desktop/lwa_enb/srsenb
asus-medium@asusmedium-UN65H:~$ cd ~/Desktop/lwa_enb/srsenb/
asus-medium@asusmedium-UN65H:~/Desktop/lwa_enb/srsenb$ sudo srsenb enb.conf
[sudo] password for asus-medium:
---  Software Radio Systems LTE eNodeB  ---

Reading configuration file enb.conf...
[INFO] [UHD] linux; GNU C++ version 5.4.0 20160609; Boost_105800; UHD_3.14.
0.0-release
Opening USRP with args: type=b200,master_clock_rate=30.72e6
[INFO] [B200] Detected Device: B210
[INFO] [B200] Operating over USB 3.
[INFO] [B200] Initialize CODEC control...
[INFO] [B200] Initialize Radio control...
[INFO] [B200] Performing register loopback test...
[INFO] [B200] Register loopback test passed
[INFO] [B200] Performing register loopback test...
[INFO] [B200] Register loopback test passed
[INFO] [B200] Asking for clock rate 30.720000 MHz...
[INFO] [B200] Actually got clock rate 30.720000 MHz.
Setting frequency: DL=2160.0 Mhz, UL=1970.0 MHz
[INFO] [B200] Asking for clock rate 23.040000 MHz...
[INFO] [B200] Actually got clock rate 23.040000 MHz.
Setting Sampling frequency 5.76 MHz

==== eNodeB started ===
Type <t> to view trace
```

1.5. 執行 srsUE

在 UE 開一個新的終端機輸入

cd ~/path/to/srsLTE/srsue
sudo srsue ue.conf

```
ue@ue-X580VD: ~/Desktop/lwaap_ue/srsue
ue@ue-X580VD:~$ cd ~/Desktop/lwaap_ue/srsue/
ue@ue-X580VD:~/Desktop/lwaap_ue/srsue$ sudo srsue ue.conf
[sudo] password for ue:
Reading configuration file ue.conf...

Built in Release mode using commit 0a69e56 on branch develop_ue.

Buffer capacity 10240
Buffer capacity 40960
---  Software Radio Systems LTE UE  ---

Opening RF device with 1 RX antennas...
[INFO] [UHD] linux; GNU C++ version 5.4.0 20160609; Boost_105800; UHD_3.14.0.0-r
elease
Opening USRP with args: type=b200,master_clock_rate=30.72e6
[INFO] [B200] Detected Device: B210
[INFO] [B200] Operating over USB 3.
[INFO] [B200] Initialize CODEC control...
[INFO] [B200] Initialize Radio control...
[INFO] [B200] Performing register loopback test...
[INFO] [B200] Register loopback test passed
[INFO] [B200] Performing register loopback test...
[INFO] [B200] Register loopback test passed
[INFO] [B200] Asking for clock rate 30.720000 MHz...
[INFO] [B200] Actually got clock rate 30.720000 MHz.
LWAAP MAC f4:96:34:3:6a:a6
LWAAP IP packet receiver thread run_enable
Waiting PHY to initialize...
...
Attaching UE...
Searching cell in DL EARFCN=500, f_dl=2160.0 MHz, f_ul=1970.0 MHz
.
Found Cell:  PCI=1, PRB=25, Ports=1, CFO=0.5 KHz
[INFO] [B200] Asking for clock rate 23.040000 MHz...
[INFO] [B200] Actually got clock rate 23.040000 MHz.
Found PLMN:  Id=00101, TAC=7
Random Access Transmission: seq=9, ra-rnti=0x2
Random Access Transmission: seq=42, ra-rnti=0x2
Random Access Transmission: seq=9, ra-rnti=0x2
RRC Connected
Random Access Complete.     c-rnti=0x48, ta=0
Network attach successful. IP: 172.16.0.2
Software Radio Systems LTE (srsLTE)
```

1.6. 流量測試

在 EPC 開一個新的終端機輸入

```
iperf3 -s -B 172.16.0.1
```

```
nuk@nuk: ~/iperf
nuk@nuk:~/iperf$ iperf3 -s -B 172.16.0.1
-----------------------------------------------------------
Server listening on 5201
-----------------------------------------------------------
Accepted connection from 172.16.0.2, port 44411
[  5] local 172.16.0.1 port 5201 connected to 172.16.0.2 port 38249
[ ID] Interval           Transfer     Bitrate         Total Datagrams
[  5]   0.00-1.00   sec  11.9 MBytes  99.9 Mbits/sec  8759
[  5]   1.00-2.00   sec  11.9 MBytes   100 Mbits/sec  8765
[  5]   2.00-3.00   sec  11.9 MBytes   100 Mbits/sec  8766
[  5]   3.00-4.00   sec  11.9 MBytes   100 Mbits/sec  8766
[  5]   4.00-5.00   sec  11.9 MBytes   100 Mbits/sec  8766
[  5]   5.00-6.00   sec  11.9 MBytes   100 Mbits/sec  8765
[  5]   6.00-7.00   sec  11.9 MBytes   100 Mbits/sec  8766
```

在 UE 開一個新的終端機輸入

```
iperf3 -c 172.16.0.1 -B 172.16.0.2 -u -l 1426b -t 120 -b 100m -R
```

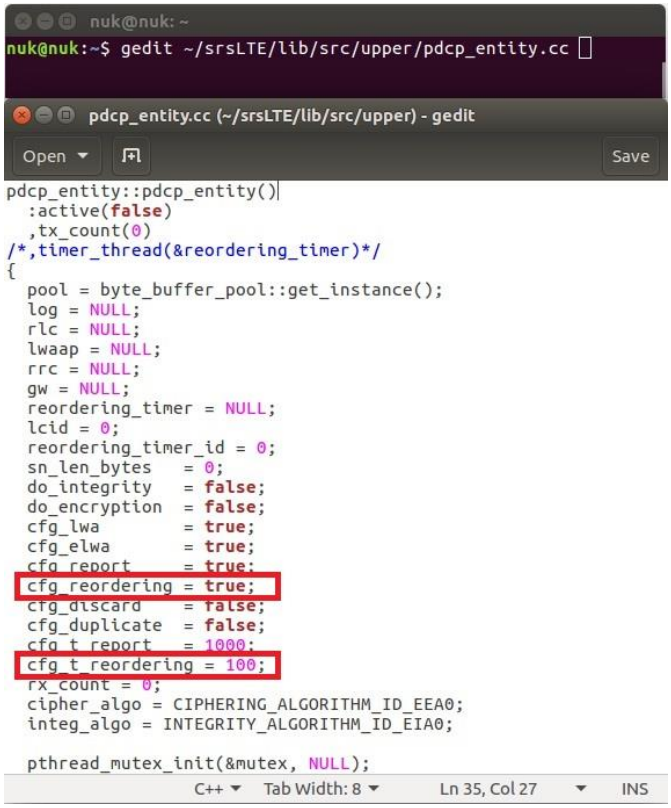
```
nuk@nuk: ~
nuk@nuk:~$ iperf3 -c 172.16.0.1 -B 172.16.0.2 -l 1426b -t 120 -u -b 100m -R
Connecting to host 172.16.0.1, port 5201
Reverse mode, remote host 172.16.0.1 is sending
[  5] local 172.16.0.2 port 59703 connected to 172.16.0.1 port 5201
[ ID] Interval           Transfer     Bitrate         Jitter    Lost/Total Datagrams
[  5]   0.00-1.00   sec  10.1 MBytes  85.1 Mbits/sec  29.219 ms  1743/9206 (19%)
[  5]   1.00-2.00   sec  9.67 MBytes  81.1 Mbits/sec  29.333 ms  1652/8765 (19%)
[  5]   2.00-3.00   sec  9.67 MBytes  81.1 Mbits/sec  31.073 ms  1653/8766 (19%)
[  5]   3.00-4.00   sec  9.67 MBytes  81.2 Mbits/sec  29.649 ms  1652/8766 (19%)
[  5]   4.00-5.00   sec  9.67 MBytes  81.1 Mbits/sec  25.812 ms  1654/8766 (19%)
[  5]   5.00-6.00   sec  9.67 MBytes  81.1 Mbits/sec  25.955 ms  1651/8764 (19%)
```

2. 封包排序

2.1. 設定 LTE WLAN 排序功能

在 UE 的終端機輸入

gedit /path/to/srsLTE/lib/src/upper/pdcp_entity.cc



cfg_reordering = true

啓動 LWA 的重新排序功能

cfg_t_reordering = 100

啓動 LWA 重新排序的等待時間

2.2. 重新編譯 srsLTE

在 UE 的終端機輸入

cd /path/to/srsLTE/build
cmake ../
make
sudo make install
sudo ldconfig

```
nuk@nuk: ~/srsLTE/build
nuk@nuk:~/srsLTE/build$ cmake ../
nuk@nuk:~/srsLTE/build$ make
nuk@nuk:~/srsLTE/build$ sudo make isntall
nuk@nuk:~/srsLTE/build$ sudo ldconfig
nuk@nuk:~/srsLTE/build$
```

2.3. 執行 srsEPC

在 EPC 開一個新的終端機輸入

cd ~/path/to/srsLTE/srsepc
./srsepc_if_masq.sh enp4s0
sudo srsepc epc.conf

enp4s0 是本例使用的對外網卡名稱，請自行查詢系統所使用的網卡名稱，因不同的系統或是硬體可能會有不同的名稱。

```
asus-medium@asusmedium-UN65H: ~/Desktop/lwa_enb/srsepc
asus-medium@asusmedium-UN65H:~$ cd ~/Desktop/lwa_enb/srsepc/
asus-medium@asusmedium-UN65H:~/Desktop/lwa_enb/srsepc$ ./srsepc_if_masq.sh wlp3s0
[sudo] password for asus-medium:
Masquerading Interface wlp3s0
asus-medium@asusmedium-UN65H:~/Desktop/lwa_enb/srsepc$ sudo srsepc epc.conf

---  Software Radio Systems EPC  ---

Reading configuration file epc.conf...
HSS Initialized.
MME GTP-C Initialized
MME Initialized.
SP-GW Initialized.
```

2.4. 執行 srseNB

在 eNB 再開一個新的終端機輸入

cd ~/path/to/srsLTE/srsenb
sudo srsenb enb.conf

```
asus-medium@asusmedium-UN65H: ~/Desktop/lwa_enb/srsenb
asus-medium@asusmedium-UN65H:~$ cd ~/Desktop/lwa_enb/srsenb/
asus-medium@asusmedium-UN65H:~/Desktop/lwa_enb/srsenb$ sudo srsenb enb.conf
[sudo] password for asus-medium:
---  Software Radio Systems LTE eNodeB  ---

Reading configuration file enb.conf...
[INFO] [UHD] linux; GNU C++ version 5.4.0 20160609; Boost_105800; UHD_3.14.
0.0-release
Opening USRP with args: type=b200,master_clock_rate=30.72e6
[INFO] [B200] Detected Device: B210
[INFO] [B200] Operating over USB 3.
[INFO] [B200] Initialize CODEC control...
[INFO] [B200] Initialize Radio control...
[INFO] [B200] Performing register loopback test...
[INFO] [B200] Register loopback test passed
[INFO] [B200] Performing register loopback test...
[INFO] [B200] Register loopback test passed
[INFO] [B200] Asking for clock rate 30.720000 MHz...
[INFO] [B200] Actually got clock rate 30.720000 MHz.
Setting frequency: DL=2160.0 Mhz, UL=1970.0 MHz
[INFO] [B200] Asking for clock rate 23.040000 MHz...
[INFO] [B200] Actually got clock rate 23.040000 MHz.
Setting Sampling frequency 5.76 MHz

==== eNodeB started ===
Type <t> to view trace
```

2.5. 執行 srsUE

在 UE 開一個新的終端機輸入

cd ~/path/to/srsLTE/srsue
sudo srsue ue.conf

```
ue@ue-X580VD: ~/Desktop/lwaap_ue/srsue
ue@ue-X580VD:~$ cd ~/Desktop/lwaap_ue/srsue/
ue@ue-X580VD:~/Desktop/lwaap_ue/srsue$ sudo srsue ue.conf
[sudo] password for ue:
Reading configuration file ue.conf...

Built in Release mode using commit 0a69e56 on branch develop_ue.

Buffer capacity 10240
Buffer capacity 40960
---  Software Radio Systems LTE UE  ---

Opening RF device with 1 RX antennas...
[INFO] [UHD] linux; GNU C++ version 5.4.0 20160609; Boost_105800; UHD_3.14.0.0-r
elease
Opening USRP with args: type=b200,master_clock_rate=30.72e6
[INFO] [B200] Detected Device: B210
[INFO] [B200] Operating over USB 3.
[INFO] [B200] Initialize CODEC control...
[INFO] [B200] Initialize Radio control...
[INFO] [B200] Performing register loopback test...
[INFO] [B200] Register loopback test passed
[INFO] [B200] Performing register loopback test...
[INFO] [B200] Register loopback test passed
[INFO] [B200] Asking for clock rate 30.720000 MHz...
[INFO] [B200] Actually got clock rate 30.720000 MHz.
LWAAP MAC f4:96:34:3:6a:a6
LWAAP IP packet receiver thread run_enable
Waiting PHY to initialize...
...
Attaching UE...
Searching cell in DL EARFCN=500, f_dl=2160.0 MHz, f_ul=1970.0 MHz
.
Found Cell:  PCI=1, PRB=25, Ports=1, CFO=0.5 KHz
[INFO] [B200] Asking for clock rate 23.040000 MHz...
[INFO] [B200] Actually got clock rate 23.040000 MHz.
Found PLMN:  Id=00101, TAC=7
Random Access Transmission: seq=9, ra-rnti=0x2
Random Access Transmission: seq=42, ra-rnti=0x2
Random Access Transmission: seq=9, ra-rnti=0x2
RRC Connected
Random Access Complete.     c-rnti=0x48, ta=0
Network attach successful. IP: 172.16.0.2
Software Radio Systems LTE (srsLTE)
```

2.6. 流量測試

在 EPC 開一個新的終端機輸入

```
iperf3 -s -B 172.16.0.1
```

```
nuk@nuk: ~/iperf
nuk@nuk:~/iperf$ iperf3 -s -B 172.16.0.1
-----------------------------------------------------------
Server listening on 5201
-----------------------------------------------------------
Accepted connection from 172.16.0.2, port 44411
[  5] local 172.16.0.1 port 5201 connected to 172.16.0.2 port 38249
[ ID] Interval           Transfer     Bitrate         Total Datagrams
[  5]   0.00-1.00   sec  11.9 MBytes  99.9 Mbits/sec  8759
[  5]   1.00-2.00   sec  11.9 MBytes   100 Mbits/sec  8765
[  5]   2.00-3.00   sec  11.9 MBytes   100 Mbits/sec  8766
[  5]   3.00-4.00   sec  11.9 MBytes   100 Mbits/sec  8766
[  5]   4.00-5.00   sec  11.9 MBytes   100 Mbits/sec  8766
[  5]   5.00-6.00   sec  11.9 MBytes   100 Mbits/sec  8765
[  5]   6.00-7.00   sec  11.9 MBytes   100 Mbits/sec  8766
```

在 UE 開一個新的終端機輸入

```
iperf3 -c 172.16.0.1 -B 172.16.0.2 -u -l 1426b -t 120 -b 100m -R
```

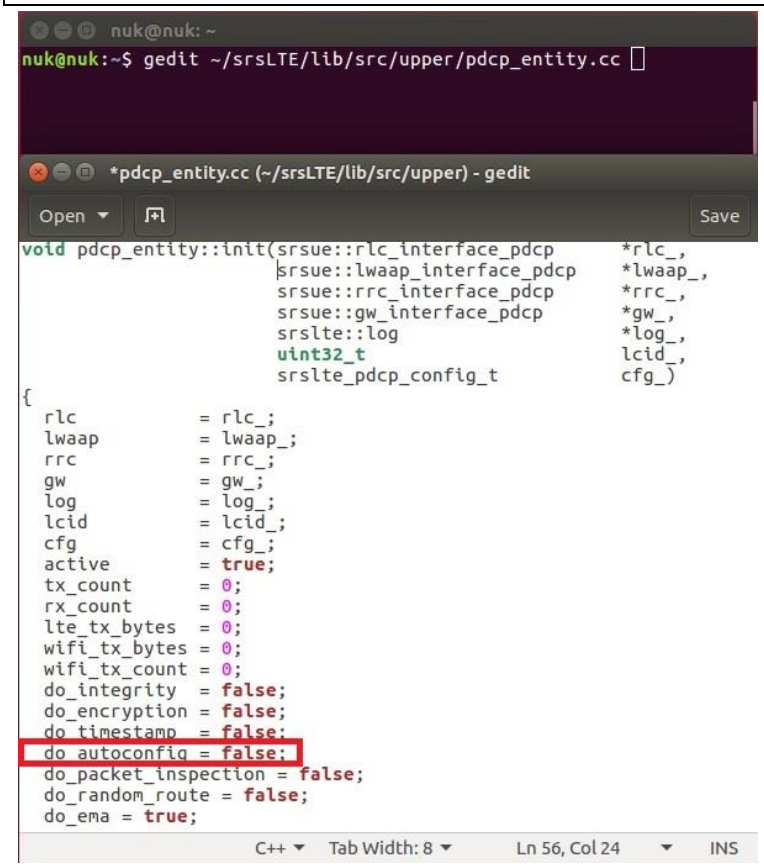
```
nuk@nuk: ~
nuk@nuk:~$ iperf3 -c 172.16.0.1 -B 172.16.0.2 -l 1426b -t 120 -u -b 100m -R
Connecting to host 172.16.0.1, port 5201
Reverse mode, remote host 172.16.0.1 is sending
[  5] local 172.16.0.2 port 59703 connected to 172.16.0.1 port 5201
[ ID] Interval           Transfer     Bitrate         Jitter    Lost/Total Datagrams
[  5]   0.00-1.00   sec  10.1 MBytes  85.1 Mbits/sec  29.219 ms  1743/9206 (19%)
[  5]   1.00-2.00   sec  9.67 MBytes  81.1 Mbits/sec  29.333 ms  1652/8765 (19%)
[  5]   2.00-3.00   sec  9.67 MBytes  81.1 Mbits/sec  31.073 ms  1653/8766 (19%)
[  5]   3.00-4.00   sec  9.67 MBytes  81.2 Mbits/sec  29.649 ms  1652/8766 (19%)
[  5]   4.00-5.00   sec  9.67 MBytes  81.1 Mbits/sec  25.812 ms  1654/8766 (19%)
[  5]   5.00-6.00   sec  9.67 MBytes  81.1 Mbits/sec  25.955 ms  1651/8764 (19%)
```

3. 自動調整傳輸比例

3.1. 設定 LTE WLAN 自動調配功能

在 eNB 的終端機輸入

```
gedit /path/to/srsLTE/lib/src/upper/pdcp_entity.cc
```



啓動 LWA 的自動調配功能

do_autoconfig = true

在 UE 的終端機輸入

```
gedit /path/to/srsLTE/lib/src/upper/pdcp_entity.cc
```

```
nuk@nuk: ~
nuk@nuk:~$ gedit ~/srsLTE/lib/src/upper/pdcp_entity.cc
```

pdcp_entity.cc (~/srsLTE/lib/src/upper) - gedit

Open Save

```
pdcp_entity::pdcp_entity()
  :active(false)
  ,tx_count(0)
/*,timer_thread(&reordering_timer)*/
{
  pool = byte_buffer_pool::get_instance();
  log = NULL;
  rlc = NULL;
  lwaap = NULL;
  rrc = NULL;
  gw = NULL;
  reordering_timer = NULL;
  lcid = 0;
  reordering_timer_id = 0;
  sn_len_bytes    = 0;
  do_integrity    = false;
  do_encryption   = false;
  cfg_lwa         = true;
  cfg_elwa        = true;
  cfg_report      = true;
  cfg_reordering  = true;
  cfg_discard     = false;
  cfg_duplicate   = false;
  cfg_t_report    = 1000;
  cfg_t_reordering = 100;
  rx_count = 0;
  cipher_algo = CIPHERING_ALGORITHM_ID_EEA0;
  integ_algo = INTEGRITY_ALGORITHM_ID_EIA0;

  pthread_mutex_init(&mutex, NULL);
```

C++ Tab Width: 8 Ln 35, Col 27 INS

啓動 LWA 回報網路狀況功能

cfg_report = true

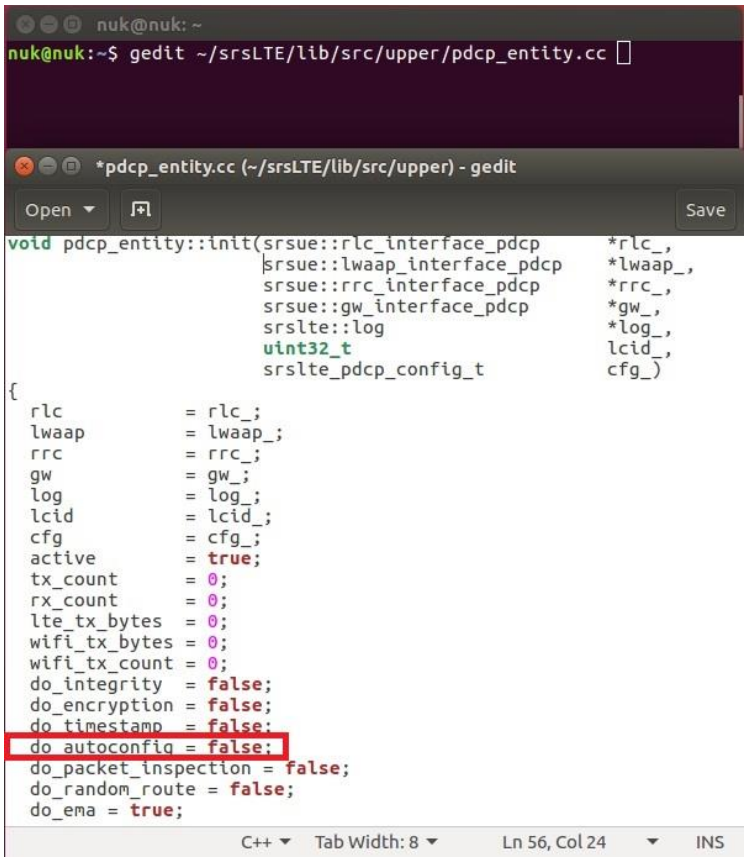
設定 LWA 回報網路狀況的時間

cfg_t_report = 5000

3.2. 設定 LTE WLAN 自動調配功能

在 eNB 的終端機輸入

```
gedit /path/to/srsLTE/lib/src/upper/pdcp_entity.cc
```



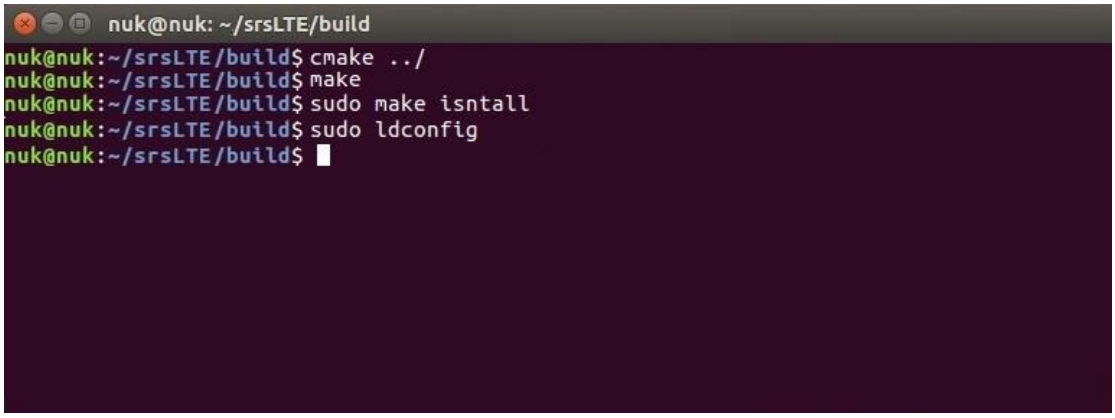
啓動 LWA 的自動調配功能

do_autoconfig = true

3.3. 重新編譯 srsLTE

在 eNB 及 UE 的終端機輸入

cd /path/to/srsLTE/build
cmake ../
make
sudo make install
sudo ldconfig

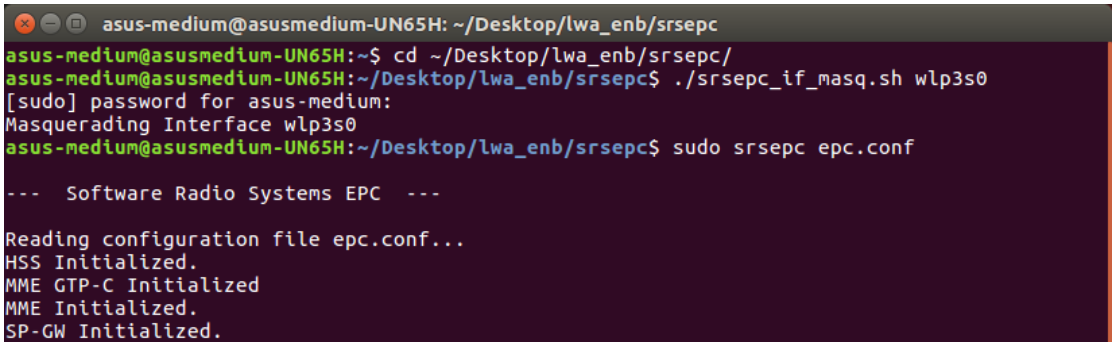


3.4. 執行 srsEPC

在 EPC 開一個新的終端機輸入

cd ~/path/to/srsLTE/srsepc
./srsepc_if_masq.sh enp4s0
sudo srsepc epc.conf

enp4s0 是本例使用的對外網卡名稱，請自行查詢系統所使用的網卡名稱，因不同的系統或是硬體可能會有不同的名稱。



3.5. 執行 srseNB

在 eNB 再開一個新的終端機輸入

cd ~/path/to/srsLTE/srsenb
sudo srsenb enb.conf

```
asus-medium@asusmedium-UN65H: ~/Desktop/lwa_enb/srsenb
asus-medium@asusmedium-UN65H:~$ cd ~/Desktop/lwa_enb/srsenb/
asus-medium@asusmedium-UN65H:~/Desktop/lwa_enb/srsenb$ sudo srsenb enb.conf
[sudo] password for asus-medium:
---  Software Radio Systems LTE eNodeB  ---

Reading configuration file enb.conf...
[INFO] [UHD] linux; GNU C++ version 5.4.0 20160609; Boost_105800; UHD_3.14.
0.0-release
Opening USRP with args: type=b200,master_clock_rate=30.72e6
[INFO] [B200] Detected Device: B210
[INFO] [B200] Operating over USB 3.
[INFO] [B200] Initialize CODEC control...
[INFO] [B200] Initialize Radio control...
[INFO] [B200] Performing register loopback test...
[INFO] [B200] Register loopback test passed
[INFO] [B200] Performing register loopback test...
[INFO] [B200] Register loopback test passed
[INFO] [B200] Asking for clock rate 30.720000 MHz...
[INFO] [B200] Actually got clock rate 30.720000 MHz.
Setting frequency: DL=2160.0 Mhz, UL=1970.0 MHz
[INFO] [B200] Asking for clock rate 23.040000 MHz...
[INFO] [B200] Actually got clock rate 23.040000 MHz.
Setting Sampling frequency 5.76 MHz

==== eNodeB started ===
Type <t> to view trace
```

3.6. 執行 srsUE

在 UE 開一個新的終端機輸入

```
ue@ue-X580VD: ~/Desktop/lwaap_ue/srsue
ue@ue-X580VD:~$ cd ~/Desktop/lwaap_ue/srsue/
ue@ue-X580VD:~/Desktop/lwaap_ue/srsue$ sudo srsue ue.conf
[sudo] password for ue:
Reading configuration file ue.conf...

Built in Release mode using commit 0a69e56 on branch develop_ue.

Buffer capacity 10240
Buffer capacity 40960
---  Software Radio Systems LTE UE  ---

Opening RF device with 1 RX antennas...
[INFO] [UHD] linux; GNU C++ version 5.4.0 20160609; Boost_105800; UHD_3.14.0.0-r
elease
Opening USRP with args: type=b200,master_clock_rate=30.72e6
[INFO] [B200] Detected Device: B210
[INFO] [B200] Operating over USB 3.
[INFO] [B200] Initialize CODEC control...
[INFO] [B200] Initialize Radio control...
[INFO] [B200] Performing register loopback test...
[INFO] [B200] Register loopback test passed
[INFO] [B200] Performing register loopback test...
[INFO] [B200] Register loopback test passed
[INFO] [B200] Asking for clock rate 30.720000 MHz...
[INFO] [B200] Actually got clock rate 30.720000 MHz.
LWAAP MAC f4:96:34:3:6a:a6
LWAAP IP packet receiver thread run_enable
Waiting PHY to initialize...
...
Attaching UE...
Searching cell in DL EARFCN=500, f_dl=2160.0 MHz, f_ul=1970.0 MHz
.
Found Cell:  PCI=1, PRB=25, Ports=1, CFO=0.5 KHz
[INFO] [B200] Asking for clock rate 23.040000 MHz...
[INFO] [B200] Actually got clock rate 23.040000 MHz.
Found PLMN:  Id=00101, TAC=7
Random Access Transmission: seq=9, ra-rnti=0x2
Random Access Transmission: seq=42, ra-rnti=0x2
Random Access Transmission: seq=9, ra-rnti=0x2
RRC Connected
Random Access Complete.     c-rnti=0x48, ta=0
Network attach successful. IP: 172.16.0.2
Software Radio Systems LTE (srsLTE)
```

cd ~/path/to/srsLTE/srsue
sudo srsue ue.conf

3.7. 流量測試

在 EPC 開一個新的終端機輸入

iperf3 -s -B 172.16.0.1

```
nuk@nuk: ~/iperf
nuk@nuk:~/iperf$ iperf3 -s -B 172.16.0.1
-----------------------------------------------------------
Server listening on 5201
-----------------------------------------------------------
Accepted connection from 172.16.0.2, port 44411
[  5] local 172.16.0.1 port 5201 connected to 172.16.0.2 port 38249
[ ID] Interval           Transfer     Bitrate         Total Datagrams
[  5]   0.00-1.00   sec  11.9 MBytes  99.9 Mbits/sec  8759
[  5]   1.00-2.00   sec  11.9 MBytes   100 Mbits/sec  8765
[  5]   2.00-3.00   sec  11.9 MBytes   100 Mbits/sec  8766
[  5]   3.00-4.00   sec  11.9 MBytes   100 Mbits/sec  8766
[  5]   4.00-5.00   sec  11.9 MBytes   100 Mbits/sec  8766
[  5]   5.00-6.00   sec  11.9 MBytes   100 Mbits/sec  8765
[  5]   6.00-7.00   sec  11.9 MBytes   100 Mbits/sec  8766
```

在 UE 開一個新的終端機輸入

iperf3 -c 172.16.0.1 -B 172.16.0.2 -u -l 1426b -t 120 -b 100m -R

```
nuk@nuk: ~
nuk@nuk:~$ iperf3 -c 172.16.0.1 -B 172.16.0.2 -l 1426b -t 120 -u -b 100m -R
Connecting to host 172.16.0.1, port 5201
Reverse mode, remote host 172.16.0.1 is sending
[  5] local 172.16.0.2 port 59703 connected to 172.16.0.1 port 5201
[ ID] Interval           Transfer     Bitrate         Jitter    Lost/Total Datagrams
[  5]   0.00-1.00   sec  10.1 MBytes  85.1 Mbits/sec  29.219 ms  1743/9206 (19%)
[  5]   1.00-2.00   sec  9.67 MBytes  81.1 Mbits/sec  29.333 ms  1652/8765 (19%)
[  5]   2.00-3.00   sec  9.67 MBytes  81.1 Mbits/sec  31.073 ms  1653/8766 (19%)
[  5]   3.00-4.00   sec  9.67 MBytes  81.2 Mbits/sec  29.649 ms  1652/8766 (19%)
[  5]   4.00-5.00   sec  9.67 MBytes  81.1 Mbits/sec  25.812 ms  1654/8766 (19%)
[  5]   5.00-6.00   sec  9.67 MBytes  81.1 Mbits/sec  25.955 ms  1651/8764 (19%)
```