

Quick Start Guide (QSG)

Arttha 5G-NR-FR1 (N77 band)

NLM Board

Contents

1	Introduction	3
1.1	Purpose	3
1.2	Abbreviations	3
2	Hardware & Software details	4
2.1	Hardware Components	4
2.2	Software Components	4
3	Application bring up	6
3.1	Files name introduction	6
3.2	Access details of NLM board	6
3.3	Steps to run the application	6
4	Running commands	11

1. Introduction

1.1 Purpose

The purpose of this Quick Start Guide is to provide the steps to create the setup for NLM (Network Listening Module) on LS1043 host and LA9310 baseband processor. This guide will further provide steps to run the 5G-NR NLM sub-6GHz application on the platform.

1.2 Abbreviations

NLM	Network listening module
SSB	Synchronization signal block
RSSI	Received signal strength indicator
VSG	Vector signal generator
GSCN	Global synchronization channel number
MIB	Master information block

2. Hardware & Software components

2.1 Hardware Components:

a) **LS1043ARDB** (motherboard) is the host and it is 64-bit Arm®-based processor for embedded networking.

b) **LA9310** (daughter-card) is a programmable baseband processor targeted at a sub-6GHz 5G such as network listening modules, repeaters, repeater controllers, etc.

2.2 Software Components:

a) **Linux**: Host device runs on Linux OS.

b) **Free RTOS**: Baseband device runs on Free RTOS.

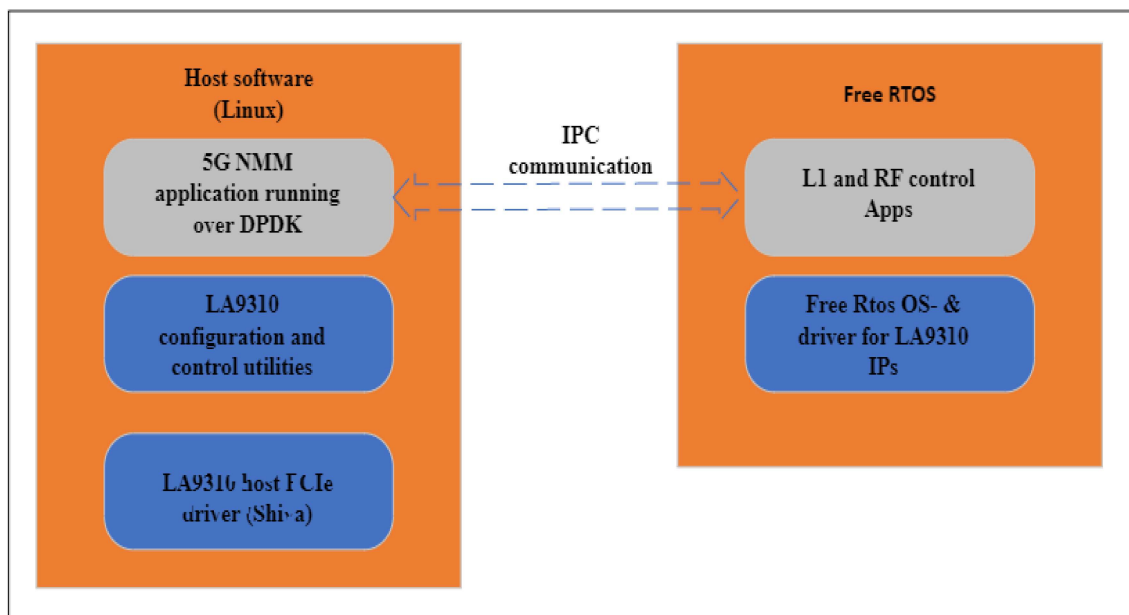


Fig.1 Host and controller side communication

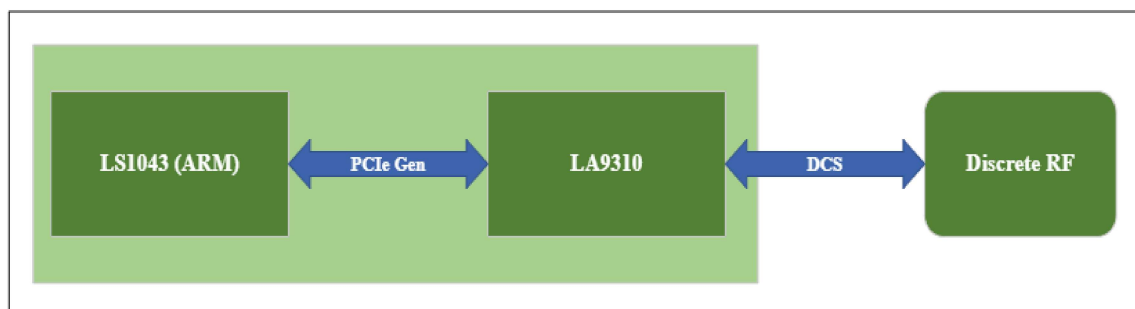


Fig. 2 Hardware connection of board



(a)



(b)

Fig. 3 Board setup and connections

3. Application bring up

3.1 Required application and data files

Boot source file: ls1043ardb_boot.scr

For controller: la9310.bin,

For VSPA: apm.eld

For NMM-test-app: nmm_test, libnmm.a, libnmmdlproc.a PSSDET_REF_TD.bin, PSS_REF_XCORR.bin

3.2 Access details of NLM board :

(a) Boardfarm IP - 192.168.2.183

(b) Host Console port – USB5

(c) Free RTOS port– USB3

3.3 Steps to run the application:

- 1) SSH into the boardfarm using boardfarm IP credentials.
- 2) Use the USB port number to minicom into the LS1043 on the boardfarm and login into the ARDB.
- 3) NMM Boot source file **ls1043ardb_boot.scr** is needed to run the application. There is one boot source file which comes with the BSP. This has to be replaced with the NMM version of boot.scr file. Copy the NMM provided .scr file to boot directory of the board with different name than what is already present in the boot directory.

```

drwx----- 2 root root      16384 Mar 17 09:38 lost+found
-rw-r--r--  1 root root       1018 May 20 05:07 ls1043ardb_boot.scr
-rw-r--r--  1 root root        965 May 30 11:02 ls1043ardb_boot.scr.g
-rw-r--r--  1 41642 user 33572380 Feb 25 15:37 lsd2004_yocto_tiny_LS_arm64.itb
drwxr-xr-x  3 41642 user      4096 Mar 17 09:46 modules
drwxr-xr-x  3 41642 user      4096 Mar 17 09:46 secboot_hdrs
-rw-r--r--  1 41642 user        997 Feb 25 15:37 srk_hash.txt
-rw-r--r--  1 41642 user 11025058 Feb 25 15:37 vmlinuz-4.19.90-rt35
root@localhost:/boot#
```

- 4) Use NMM provided ls1043ardb_boot.scr when running NLM application and original (BSP default) ls1043ardb_boot.scr to run LAN on the board to transfer compiled binaries for VSPA and M4 controller. LAN does not work with the NMM app ls1043ardb_boot.scr file, so boot scr files should be renamed if working in LAN mode or in NMM mode.

Important: Make sure after renaming of files that one version of **ls1043ardb_boot.scr file(name as it is) is always in the boot directory or else the board will never boot.**

As can be seen from above steps NMM ls1043ardb_boot.scr is of 1018 bytes and BSP ls1043ardb_boot.scr is of 965 bytes.

NOTE: To Configure LAN (to transfer file into the board) follow following steps:

```
cd /boot
mv ls1043ardb_boot.scr ls1043ardb_boot.scr.n
mv ls1043ardb_boot.scr.g ls1043ardb_boot.scr
ls -l
reboot
cpld reset altbank
boot
dhclient
ifconfig fml-mac4 192.168.2.203
```

To run nmm application (change from LAN mode to NMM app mode):

```
cd /boot
mv ls1043ardb_boot.scr ls1043ardb_boot.scr.g
mv ls1043ardb_boot.scr.n ls1043ardb_boot.scr
ls -l
reboot
cpld reset altbank
boot
5) Now do reboot
```

```
NOTICE: BL31: Boot: 22:16:20, Nov 15 2021
NOTICE: Welcome to LS1043 BL31 Phase

U-Boot 2019.10 (Nov 15 2021 - 22:16:03 +0530)

SoC: LS1043AE Rev1.1 (0x87920011)
Clock Configuration:
  CPU0(A53):1600 MHz  CPU1(A53):1600 MHz  CPU2(A53):1600 MHz
  CPU3(A53):1600 MHz
  Bus: 400 MHz  DDR: 1600 MT/s  FMAN: 500 MHz
Reset Configuration Word (RCW):
  00000000: 08100010 0a000000 00000000 00000000
  00000010: 14550002 80004012 e0025000 c1002000
  00000020: 00000000 00000000 00000000 00038800
  00000030: 00000000 00001101 00000096 00000001
Model: LS1043A RDB Board
Board: LS1043ARDB, boot from vBank 0
CPLD: V2.0
PCBA: V6.0
SERDES Reference Clocks:
SD1 CLK1 = 156.25MHZ, SD1 CLK2 = 100.00MHZ
DRAM: 1.9 GiB (DDR4, 32-bit, CL=11, ECC off)
Using SERDES1 Protocol: 5205 (0x1455)
SEC0: RNG instantiated
Firmware 'Microcode version 0.0.1 for LS1021a r1.0' for 1021 V1.0
QE: uploading microcode 'Microcode for LS1021a r1.0' version 0.0.1
Flash: 128 MiB
NAND: 512 MiB
MMC: FSL_SDHC: 0
Loading Environment from Flash... OK
EEPROM: NXID v1
In: serial
Out: serial
Err: serial
Net: Fman1: Uploading microcode version 106.4.18
PCIE0: pcie@3400000 disabled
PCIE1: pcie@3500000 Root Complex: no link
PCIE2: pcie@3600000 Root Complex: x1 gen2
FM1@DTSEC1, FM1@DTSEC2, FM1@DTSEC3 [PRIME], FM1@DTSEC4, FM1@DTSEC5, FM1@DTSEC6, FM1@TGEC1
=>
CTRL-A Z for help | 115200 8N1 | NOR | Minicom 2.7 | VT102 | Offline | ttyUSB4
```

Board will stop at above prompt

6) Type `cpld reset altbank` here at the prompt, it will start booting and a timer count down to 0 will appear.

7) Now press command 'boot'

8) Board will start booting up and stop at login prompt

```
[ OK ] Started Getty on tty1.
[ OK ] Reached target Login Prompts.
[ OK ] Started LSB: HPA's tftp server.
[ OK ] Started LSB: Allows network connections to serial ports.
[ OK ] Started OpenBSD Secure Shell server.
[ OK ] Started Dispatcher daemon for systemd-networkd.
[ OK ] Reached target Multi-User System.
[ OK ] Reached target Graphical Interface.
      Starting Update UTMP about System Runlevel Changes...
[ OK ] Started Update UTMP about System Runlevel Changes.

NXP LSDK 2004 main
localhost login:
CTRL-A Z for help | 115200 8N1 | NOR | Minicom 2.7 | VT102 | Offline | t
```

9) Login is root and password is root

```
localhost login: root
Password:
Last login: Mon May 30 14:39:57 UTC 2022 on ttyS0
Welcome to NXP LSDK 2004 main (GNU/Linux 4.19.90-rt35 aarch64)

* Support:      https://www.nxp.com/lSDK
* Documentation: https://lsdk.github.io/document
* Licensing:    https://lsdk.github.io/eula
root@localhost:~#
CTRL-A Z for help | 115200 8N1 | NOR | Minicom 2.7 | VT102 | Offline | t
```

10) Now we are inside host environment

11) Check for LA9310 binaries in `/lib/firmware` directory

12) NLM application is dependent on NMM host environment as we have not yet gone into host side yet

NLM need NMM `boot.scr` also needs NMM application(Host) from NXP.

13) Now all commands will be run from this folder as nmm is present here

14) Before running any other command, open FreeRTOS port in another terminal, and minicom into RTOS USB port.

15) Type following command on the host console from `/lib/firmware/lteFW` directory to load VSPA

```
echo 1 > /sys/bus/pci/rescan
```

```
echo 7 > /proc/sys/kernel/printk
```



```
insmod la9310shiva.ko scratch_buf_size=0x20000000 scratch_buf_phys_addr=0xc0000000
```

This will load VSPA binary

```
[ 841.159015] NXP-LA9310-Driver 0001:01:00.0: MSI:ATU: DBI 0xffff000023400054, DMA 1571000, EP b000000
[ 841.168227] NXP-LA9310-Driver 0001:01:00.0: MSI ATU done
[ 841.173862] NXP-LA9310-Driver 0001:01:00.0: irq mux 78 allocated successfully
[ 841.181604] NXP-LA9310-Driver 0001:01:00.0: nlm0: Initiating Reset handshake
[ 841.188647] NXP-LA9310-Driver 0001:01:00.0: [Reset HS] Waiting for FreeRTOS to write 3
[ 841.195121] pcieport 0001:00:00.0: AER: Multiple Corrected error received: 0001:00:00.0
[ 841.195137] pcieport 0001:00:00.0: PCIe Bus Error: severity=Corrected, type=Physical Layer, (Receive
[ 841.195141] pcieport 0001:00:00.0: device [1957:8080] error status/mask=00000001/00006000
[ 841.195145] pcieport 0001:00:00.0: [ 0] RxErr (First)
[ 841.219810] NXP-LA9310-Driver 0001:01:00.0: Host Handshake interrupt boom!! irq num 81
[ 841.237114] NXP-LA9310-Driver 0001:01:00.0: LA9310 Reset HSHAKE done, scratch 0x3
[ 841.244590] NXP-LA9310-Driver 0001:01:00.0: HIF Version : 1.0
[ 841.250328] NXP-LA9310-Driver 0001:01:00.0: nlm0:Initiating sub-drivers
[ 841.256934] NXP-LA9310-Driver 0001:01:00.0: subdrv [IPC] virqmap init
[ 841.263366] NXP-LA9310-Driver 0001:01:00.0: virqmap init, evtmask f, count 4
[ 841.270406] NXP-LA9310-Driver 0001:01:00.0: Inside la9310_ipc_probe function K_hif=2a0
[ 841.278492] NXP-LA9310-Driver 0001:01:00.0: IPC modem is ready!
[ 841.284416] NXP-LA9310-Driver 0001:01:00.0: Exiting function la9310_ipc_probe
[ 841.291547] NXP-LA9310-Driver 0001:01:00.0: subdrv [VSPA] virqmap init
[ 841.298067] NXP-LA9310-Driver 0001:01:00.0: virqmap init, evtmask 10, count 1
[ 841.305375] NXP-LA9310-Driver 0001:01:00.0: nlm0-vspa0: hwver 0x02011500, 16 AUs, dmem 6400 bytes
[ 841.314239] NXP-LA9310-Driver 0001:01:00.0: INFO:vspa_probe : VSPA Loading firmware initiated-
[ 842.023655] NXP-LA9310-Driver 0001:01:00.0: Downloaded f/w at 0xffff00001a801000 size: 1706764
[ 842.032322] NXP-LA9310-Driver 0001:01:00.0: Copy fw to ffff00001c000000, size 1706764
[ 843.083562] NXP-LA9310-Driver 0001:01:00.0: INFO:vspa_probe :VSPA FW image apm.eld loading finished
[ 843.092626] NXP-LA9310-Driver 0001:01:00.0: Boot Ok Msg Verified: msb = F1000000, lsb = 00000000
[ 843.101408] NXP-LA9310-Driver 0001:01:00.0: SW Version: vspa = 00000000, ippu = 00000000
[ 843.109493] NXP-LA9310-Driver 0001:01:00.0: SPM Ack Msg: msb = F0700000, lsb = 00000000
root@localhost:/lib/firmware/lteFW#
```

16) Now run the following command to execute NMM application

```
mkdir /dev/hugepages2M
mount -t hugetlbfs -o pagesize=2M none /dev/hugepages2M
chmod +x nmm-test
./nmm-test --vdev=bbdev_la93xx -n l
NMM prompt will show up
```

```
PMD: dpaa_sec-1 cryptodev init
PMD: dpaa_sec-2 cryptodev init
PMD: dpaa_sec-3 cryptodev init
PMD: dpaa_sec-4 cryptodev init
[ 931.973124] NXP-LA9310-Driver 0001:01:00.0: Huge Page Buff:0x98000000[H]-0xc0000000[M],s
[ 931.982170] la9310_dev->hif->ipc_regs.ipc_mdata_size: 352
libnmm: nmm_init: NMM Library initialized (libnmm.a version 1.1.0)
nmm>
CTRL-A Z for help | 115200 8N1 | NOR | Minicom 2.7 | VT102 | Offline | ttyUSB4
```

17) Type quit here and wait for app to close

```
libnmm: nmm_init: NMM Library initialized (libnmm.a version 1.1.0)
nmm> quit
[ 998.535669] USDPAA process leaking 16392 FQIDs
root@localhost:/lib/firmware/lteFW#
```

18) Run following command to configure rfic

```
cd /home/root/rf-ctrl/
python3 nlm.py
RFIC prompt will come, here configure rfic
```

```

root@localhost:/lib/firmware/lteFW# cd /home/root/rf-ctrl/
root@localhost:/home/root/rf-ctrl# python3 nlm.py
Starting RF prompt...
rfic-nlm>
CTRL-A 7 for help | 115200 8N1 | NOR | Minicom 2.7 | VT102 | Offline |

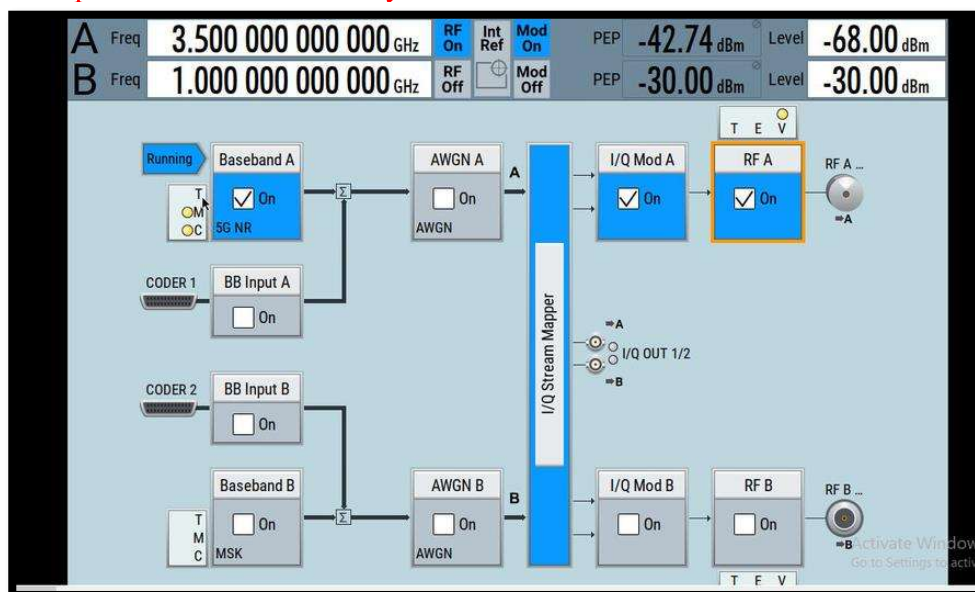
```

Configure LNA by using command

lna l

19) Check VSG connection Port A/B to the board and configuration of VSG, keep RF, BASEBAND, IQ on.

Important: Always off the RF on VSG while configuring VSG frequency and power level, and RF power level should always be set below -65dBm.



20) In RFIC prompt configure frequency equal to the value of VSG or the frequency to tune using freq command (This step is only required for ssb_scan).

```

Starting RF prompt...
rfic-nlm> freq 3500000
rfic-nlm> getfreq
3500000kHz
rfic-nlm>
CTRL-A Z for help | 115200 8N1 | NOR | Mir

```

Configure using freq <value> command and verify using getfreq command to see if configured properly

4. Running commands

All the commands will run on nmm application (on host console) and the corresponding output will be on controller console.

To get list of commands in the host type help, it will list out all the commands.

1) band_scan_n_77

This command will scan the whole n77 band for GSCN values (7711 to 8329), according to the gscn frequency set on VSG it will try to detect the respective GSCN value.

Type command band_scan_n_77 on nmm application (on host).

```
nmm> band_scan_n_77
libnmm: process_band_scan_n77: msg is 4945088
libnmm: build_band_scan_n77_cmd: entering into build_band_scan_n77_cmd
nmm> libnmm: process_rx: Received message address from controller to host: buf = 8811ea00, len = 54
```

It will give this type of output on controller console.

```
GSCN : 7904
Status: CELL NOT FOUND

GSCN : 7905
Status: CELL NOT FOUND

GSCN : 7906
Status: CELL NOT FOUND

GSCN : 7907
Status: CELL NOT FOUND

GSCN : 7908
Status: CELL NOT FOUND

GSCN : 7909
Status: CELL NOT FOUND

GSCN : 7910
Status: CELL NOT FOUND

GSCN : 7911
CellId: 111
Rssi : -55 dBm
Status: SSB_FOUND

GSCN : 7912
Status: CELL NOT FOUND

GSCN : 7913
Status: CELL NOT FOUND

GSCN : 7914
Status: CELL NOT FOUND

GSCN : 7915
Status: CELL NOT FOUND

GSCN : 7916
Status: CELL NOT FOUND

GSCN : 7917
Status: CELL NOT FOUND

GSCN : 7918
Status: CELL NOT FOUND

GSCN : 7919
Status: CELL NOT FOUND
```

2) ssb_scan

It will give the Cell id, rssi and C_{f0} details.

Type command ssb_scan on nmm application (on host).

```
nmm> ssb_scan
```

If frequency configuration on host and vsg are not same or if vsg is not connected to board then it will give following output on controller console.

```
Scanning ssb scan...
CellId: NA
Rssi : NA dBm
Cfo : NA KHz
Status: SSB_SCAN_TIMEOUT
CTRL-A Z for help | 115200 8N1 | NOR | Minicom 2.7 | VT102 | Offline | ttyUSB3
```

When frequency configuration are correct and vsg is connected to board it will give the following output.

```
Scanning ssb scan...
CellId: 111
Rssi : -55 dBm
Cfo : 0 KHz
Status: SSB_FOUND
CTRL-A Z for help | 115200 8N1 | NOR | Minicom 2.7 | VT102 | Offline | ttyUSB3
```

3) cell_follow gscn <enter gscn number>

It will track cell id for ever 20ms and also reduces C_{fo} .

Type command cell_follow gscn <gscn no.> on nmm application (on host).

```
nmm> cell_follow gscn 7911
nmm> Received fapi response: type = Cell follow response, err code = OK
libnmm: process_rx: Received message address from controller to host: buf = 8811eb80, len = 54
CTRL-A Z for help | 115200 8N1 | NOR | Minicom 2.7 | VT102 | Offline | ttyUSB5
```

A sample output is given below

```
Cell Follow for GSCN 7911 in Progress...
CellId : 111
Rssi : -55 dBm
Cfo : 0 KHz
ppbx_10 : 0
Status : CELL_FOUND
[SSB_TRACK_UPDATE]: SFN/SF: 2/3, CellId: 111, Rssi: -55 dBm, cfo_x1000: 5973136 Hz, ppb_x10: 16623
[SSB_TRACK_UPDATE]: SFN/SF: 200/3, CellId: 111, Rssi: -55 dBm, cfo_x1000: 0 Hz, ppb_x10: 0
[SSB_TRACK_UPDATE]: SFN/SF: 300/3, CellId: 111, Rssi: -55 dBm, cfo_x1000: 0 Hz, ppb_x10: 0
[SSB_TRACK_UPDATE]: SFN/SF: 400/3, CellId: 111, Rssi: -55 dBm, cfo_x1000: 0 Hz, ppb_x10: 0
[SSB_TRACK_UPDATE]: SFN/SF: 500/3, CellId: 111, Rssi: -55 dBm, cfo_x1000: 0 Hz, ppb_x10: 0
[SSB_TRACK_UPDATE]: SFN/SF: 600/3, CellId: 111, Rssi: -55 dBm, cfo_x1000: 0 Hz, ppb_x10: 0
[SSB_TRACK_UPDATE]: SFN/SF: 700/3, CellId: 111, Rssi: -55 dBm, cfo_x1000: 0 Hz, ppb_x10: 0
[SSB_TRACK_UPDATE]: SFN/SF: 800/3, CellId: 111, Rssi: -55 dBm, cfo_x1000: 0 Hz, ppb_x10: 0
[SSB_TRACK_UPDATE]: SFN/SF: 900/3, CellId: 111, Rssi: -55 dBm, cfo_x1000: 0 Hz, ppb_x10: 0
[SSB_TRACK_UPDATE]: SFN/SF: 1000/3, CellId: 111, Rssi: -55 dBm, cfo_x1000: 0 Hz, ppb_x10: 0
[SSB_TRACK_UPDATE]: SFN/SF: 76/3, CellId: 111, Rssi: -55 dBm, cfo_x1000: 0 Hz, ppb_x10: 0
[SSB_TRACK_UPDATE]: SFN/SF: 176/3, CellId: 111, Rssi: -55 dBm, cfo_x1000: 0 Hz, ppb_x10: 0
[SSB_TRACK_UPDATE]: SFN/SF: 276/3, CellId: 111, Rssi: -55 dBm, cfo_x1000: 0 Hz, ppb_x10: 0
[SSB_TRACK_UPDATE]: SFN/SF: 376/3, CellId: 111, Rssi: -55 dBm, cfo_x1000: 0 Hz, ppb_x10: 0
[SSB_TRACK_UPDATE]: SFN/SF: 476/3, CellId: 111, Rssi: -55 dBm, cfo_x1000: 0 Hz, ppb_x10: 0
[SSB_TRACK_UPDATE]: SFN/SF: 576/3, CellId: 111, Rssi: -55 dBm, cfo_x1000: 0 Hz, ppb_x10: 0
[SSB_TRACK_UPDATE]: SFN/SF: 676/3, CellId: 111, Rssi: -55 dBm, cfo_x1000: 0 Hz, ppb_x10: 0
[SSB_TRACK_UPDATE]: SFN/SF: 776/3, CellId: 111, Rssi: -55 dBm, cfo_x1000: 0 Hz, ppb_x10: 0
[SSB_TRACK_UPDATE]: SFN/SF: 876/3, CellId: 111, Rssi: -55 dBm, cfo_x1000: 0 Hz, ppb_x10: 0
[SSB_TRACK_UPDATE]: SFN/SF: 976/3, CellId: 111, Rssi: -55 dBm, cfo_x1000: 0 Hz, ppb_x10: 0
[SSB_TRACK_UPDATE]: SFN/SF: 52/3, CellId: 111, Rssi: -55 dBm, cfo_x1000: 0 Hz, ppb_x10: 0
[SSB_TRACK_UPDATE]: SFN/SF: 152/3, CellId: 111, Rssi: -55 dBm, cfo_x1000: 0 Hz, ppb_x10: 0
[SSB_TRACK_UPDATE]: SFN/SF: 252/3, CellId: 111, Rssi: -55 dBm, cfo_x1000: 0 Hz, ppb_x10: 0
```

4) cell_follow_mib gscn <enter gscn number>

It will detect mib (master information block) on the received signal.

Type command cell_follow mib gscn <gscn no.> on nmm application (on host).

```
nmm> cell_follow_mib gscn 7911
nmm> libnmm: process_rx: Received message address from controller to host: buf = 8811ed00, len = 54
```

A sample output is given below

```

cell Follow MIB for GSCN 7911 in Progress...
cell follow: Missed Detection
CellId      : 111
Rssi        : -55 dBm
Cfo         : -6 KHz
ppbX_10     : -16698
Status      : CELL FOUND
[SSB_TRACK_UPDATE]: SFN/SF: 86/3, CellId: 111, Rssi: -57 dBm, cfo_x1000: 0 Hz, ppb_x10: 0
[MIB_INFO]:SFN: 0
[MIB_INFO]:subCarrierSpacingCommon: scs15
[MIB_INFO]:ssb_subcarrierOffset: 0
[MIB_INFO]:dmrs_typeA_Position: pos2
[MIB_INFO]:pdcch_configIndex: 0
[MIB_INFO]:intraFreqReselection: allowed
[MIB_INFO]:cellBarred: barred
[MIB_INFO]:half_frame_bit: 0
[MIB_INFO]:k_ssb_msb: 0
[SSB_TRACK_UPDATE]: SFN/SF: 280/3, CellId: 111, Rssi: -55 dBm, cfo_x1000: 0 Hz, ppb_x10: 0
[SSB_TRACK_UPDATE]: SFN/SF: 380/3, CellId: 111, Rssi: -57 dBm, cfo_x1000: 0 Hz, ppb_x10: 0
[SSB_TRACK_UPDATE]: SFN/SF: 480/3, CellId: 111, Rssi: -55 dBm, cfo_x1000: 0 Hz, ppb_x10: 0
[SSB_TRACK_UPDATE]: SFN/SF: 580/3, CellId: 111, Rssi: -55 dBm, cfo_x1000: 1054 Hz, ppb_x10: 3
[SSB_TRACK_UPDATE]: SFN/SF: 680/3, CellId: 111, Rssi: -55 dBm, cfo_x1000: 0 Hz, ppb_x10: 0
[SSB_TRACK_UPDATE]: SFN/SF: 780/3, CellId: 111, Rssi: -57 dBm, cfo_x1000: 0 Hz, ppb_x10: 0
[SSB_TRACK_UPDATE]: SFN/SF: 880/3, CellId: 111, Rssi: -55 dBm, cfo_x1000: 0 Hz, ppb_x10: 0
[SSB_TRACK_UPDATE]: SFN/SF: 980/3, CellId: 111, Rssi: -55 dBm, cfo_x1000: 0 Hz, ppb_x10: 0
MIB CRC FAIL
[SSB_TRACK_UPDATE]: SFN/SF: 1080/3, CellId: 111, Rssi: -55 dBm, cfo_x1000: 0 Hz, ppb_x10: 0
[SSB_TRACK_UPDATE]: SFN/SF: 78/3, CellId: 111, Rssi: -55 dBm, cfo_x1000: 0 Hz, ppb_x10: 0
[SSB_TRACK_UPDATE]: SFN/SF: 176/3, CellId: 111, Rssi: -55 dBm, cfo_x1000: 0 Hz, ppb_x10: 0
[SSB_TRACK_UPDATE]: SFN/SF: 276/3, CellId: 111, Rssi: -55 dBm, cfo_x1000: 0 Hz, ppb_x10: 0
[SSB_TRACK_UPDATE]: SFN/SF: 376/3, CellId: 111, Rssi: -57 dBm, cfo_x1000: 0 Hz, ppb_x10: 0
[SSB_TRACK_UPDATE]: SFN/SF: 476/3, CellId: 111, Rssi: -55 dBm, cfo_x1000: 0 Hz, ppb_x10: 0
[SSB_TRACK_UPDATE]: SFN/SF: 576/3, CellId: 111, Rssi: -57 dBm, cfo_x1000: 0 Hz, ppb_x10: 0
GSCN      : 7955

```

5) cell_follow stop

After running cell_follow gscn or cell_follow mib it has to be stopped otherwise, it will run continuously, so this command will stop the cell follow.

```

nmm> cell_follow_stop
nmm> libnmm: process_rx: Received message address from controller to host: buf = 8811e880, len = 54
CTRL-A Z for help | 115200 8N1 | NOR | Minicom 2.7 | VT102 | Offline | ttyUSB5

```