FC20 Series Third-Party Linux Platform Wi-Fi&Bluetooth User Guide

Wi-Fi&Bluetooth Module Series

Version: 1.0.0

Date: 2021-07-08

Status: Preliminary

Our aim is to provide customers with timely and comprehensive service. For any assistance, please contact our company headquarters:

Quectel Wireless Solutions Co., Ltd.
Building 5, Shanghai Business Park Phase III (Area B), No.1016 Tianlin Road, Minhang District, Shanghai 200233, China
Tel: +86 21 5108 6236
Email: info@quectel.com

Or our local office. For more information, please visit:
http://www.quectel.com/support/sales.htm.

For technical support, or to report documentation errors, please visit:
http://www.quectel.com/support/technical.htm
Or email to support@quectel.com.

General Notes

Quectel offers the information as a service to its customers. The information provided is based upon customers' requirements. Quectel makes every effort to ensure the quality of the information it makes available. Quectel does not make any warranty as to the information contained herein, and does not accept any liability for any injury, loss or damage of any kind incurred by use of or reliance upon the information. All information supplied herein is subject to change without prior notice.

Disclaimer

While Quectel has made efforts to ensure that the functions and features under development are free from errors, it is possible that these functions and features could contain errors, inaccuracies and omissions. Unless otherwise provided by valid agreement, Quectel makes no warranties of any kind, implied or express, with respect to the use of features and functions under development. To the maximum extent permitted by law, Quectel excludes all liability for any loss or damage suffered in connection with the use of the functions and features under development, regardless of whether such loss or damage may have been foreseeable.

Duty of Confidentiality

The Receiving Party shall keep confidential all documentation and information provided by Quectel, except when the specific permission has been granted by Quectel. The Receiving Party shall not access or use Quectel's documentation and information for any purpose except as expressly provided herein. Furthermore, the Receiving Party shall not disclose any of the Quectel's documentation and information to any third party without the prior written consent by Quectel. For any noncompliance to the above requirements, unauthorized use, or other illegal or malicious use of the documentation and information, Quectel will reserve the right to take legal action.

Copyright

About the Document

Revision History

Version	Date	Author	Description
-	2021-01-28	Qingzong MA	Creation of the document
1.0.0	2021-07-08	Qingzong MA	Preliminary

Contents

Table Index

Figure Index

1 Introduction

Quectel FC20 series modules are low-power and cost-effective Wi-Fi/Bluetooth modules which support Wi-Fi and Bluetooth features on the third-party Linux platform.

This document introduces how to enable Wi-Fi and Bluetooth features with FC20 series modules, including function verification method and the description of factory test mode.

2 Enabling Wi-Fi

This chapter takes IMX8QXP platform as an example to describe how to enable Wi-Fi of FC20 series modules on the third-party Linux platform, the function verification and factory test.

2.1. Building Environment

2.1.1. Hardware Environment

1. Prepare the following hardware:
   - Quectel FC20 series module
   - IMX8QXP demo board
2. Install the module and demo board as shown below (take NXP board MCIMX8QXP-CPU as an example):

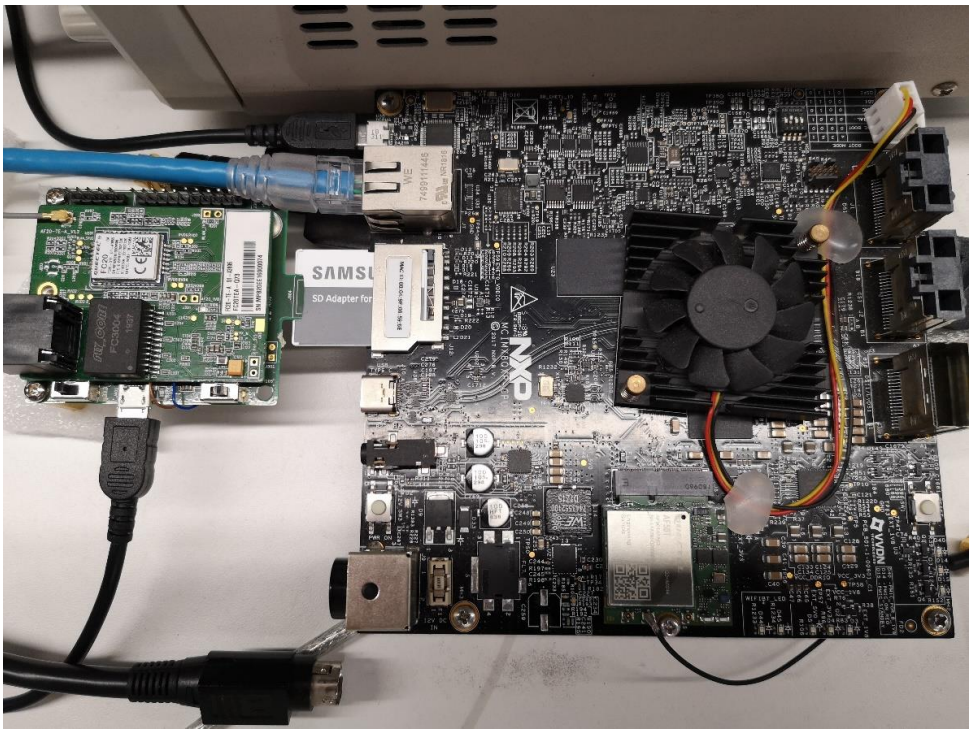


Figure 1: Hardware Environment

2.1.2. Software Environment

Recommended Kernel version: 4.14.78
IMX yocto environment downloading address (Git address):

```
git clone https://source.codeaurora.org/external/imx/linux-imx.git --branch imx_4.14.78_1.0.0_ga
```

2.1.2.1. Installing IMX8QXP Tool Chain

Execute the following command to install the IMX8QXP tool chain on host.

```
$ ./ fsl-imx-xwayland-glibc-x86_64-meta-toolchain-aarch64-toolchain-4.14-sumo.sh // Install toolchain
    NXP i.MX Release Distro SDK installer version 4.14-sumo
    ===========================================================
    Enter target directory for SDK (default: /opt/fsl-imx-xwayland/4.14-sumo): /home/qingzong/toolchain/
    You are about to install the SDK to "/home/qingzong/toolchain". Proceed[Y/n]? Y
    Extracting SDK.....done
    Setting it up...done
    SDK has been successfully set up and is ready to be used.
    Each time you wish to use the SDK in a new shell session, you need to source the environment setup script e.g.
  $ . /home/qingzong/toolchain/environment-setup-aarch64-poky-linux
```

2.1.2.2. Configuring IMX8QXP Tool Chain Environment

After the tool china is installed, execute **source** to configure the running environment. Here is an example:

```
$ source /home/qingzong/toolchain/environment-setup-aarch64-poky-linux
```

NOTE

In the actual practice, the directories in the command mentioned above should be changed to the actual directories.

2.1.2.3. Compiling IMX8QXP Images

Execute the following command to compile IMX8QXP images:

```
$ cd ${kernel-source}        // Please change to your directory
$ make defconfig
$ make menuconfig
$    LDFLAGS="" CC="$CC" make -j24
```

NOTE

Please open the following configuration items in *menuconfig* file:
CONFIG_CFG80211=y
CONFIG_HOSTAP=y
CONFIG_WIRELESS_EXT=y
CONFIG_WEXT_PRIV=y
CONFIG_NL80211_TESTMODE=y
CONFIG_CFG80211_WEXT=y
CONFIG_CFG80211_INTERNAL_REGDB=y
CONFIG_CFG80211_REG_DEBUG=y
CONFIG_BCMDHD=n
CONFIG_RFKILL=y
CONFIG_RFKILL_INPUT=y
CONFIG_RFKILL_REGULATOR=y
CONFIG_RFKILL_GPIO=y

2.2. Wi-Fi Host Driver

2.2.1. Obtaining Module SDK

Please contact Quectel Technical Support for SDK of the specific FC20 series module.

2.2.2. Setting Up SDK Compilation Environment

Create a folder named *tmp* and decompress the open source code of FC20 series module to the *tmp* folder:

```
# mkdir tmp
# cd tmp
# tar -xvf FC20_WiFi_SPs_xx.tar.bz2
# cd FC20_WiFi_SPs_xx
```

2.2.3. Configuring IMX8QXP Tool Chain Environment

Execute the following command to configure IMX8QXP tool chain environment. Here is an example:

```
$ source /home/qingzong/toolchain/environment-setup-aarch64-poky-linux
```

NOTE

In the actual practice, the directory in the command mentioned above should be changed to the actual directory.

2.2.4. Configuring Wi-Fi Compilation Environment

Execute the following command to configure Wi-Fi compilation environment. Here is an example:

```
$ cd <FC20_target_root>/cnss_host_LEA/chss_proc/host/AIO/build
$ vim scripts/te-f30/config.te-f30              // Modify KERNELPATH, KERNELARCH and TOOLPREFIX
        export KERNELPATH=/home/qingzong/code/kernel-source
        export KERNELARCH=arm64
        export TOOLPREFIX=${CROSS_COMPILE}
$ cd <FC20_target_root>/cnss_host_LEA/chss_proc/host/AIO/drivers/qcacld-new
$ vim Makefile                                  // Modify KERNEL_SRC
    KERNEL_SRC ?= /home/qingzong/code/kernel-source
```

NOTE

<FC20_target_root> is the root directory that stores the decompressed FC20 SDK.

2.2.5. Compiling Wi-Fi Driver

Execute the following commands to compile Wi-Fi driver. Here is an example:

```
$ cd <FC20_target_root>/cnss_host_LEA/chss_proc/host/AIO/build
$ make drivers
```

NOTE

<FC20_target_root> is the root directory that stores the decompressed FC20 SDK.

2.2.6. Installing WLAN Firmware and Wi-Fi Driver

Install the WLAN firmware and Wi-Fi driver to IMX8QXP demo board according to the following steps.

1. The WLAN firmware is located at *<FC20_target_root>/meta_build/load_meta/wlan_firmware/sdio*. Copy the binary files (*bdwlan30.bin*, *otp30.bin*, *qwlan30.bin* and *utf30.bin*) to */lib/firmware/* of IMX8QXP demo board with USB flash disk. Here is an example:

```
$ ls <FC20_target_root>/meta_build/load_meta/wlan_firmware/sdio/
```

2. Wi-Fi driver configuration file is located at *<FC20_target_root>/meta_build/load_meta/host/wlan_host/sdio*. Copy the configuration file *qcom_cfg.ini* to */lib/firmware/wlan/* of IMX8QXP demo board (please create the directory if the target directory */lib/firmware/wlan/* does not exist) with USB flash disk. Here is an example:

```
$ ls <FC20_target_root>/meta_build/load_meta/host/wlan_host/sdio/qcom_cfg.ini
```

3. The .ko file of the Wi-Fi driver is located at *<FC20_target_root>/cnss_host_LEA/chss_proc/host/AIO/drivers/qcacld-new/*. Copy file *wlan.ko* to */lib/modules/$(echo $(uname –r))/extra* of the IMX8QXP demo board with USB flash disk. Here is an example:

```
$ ls <FC20_target_root>/cnss_host_LEA/chss_proc/host/AIO/drivers/qcacld-new/
```

2.3. WLAN Driver

2.3.1. Enumerating SDIO

Execute the following command to check whether SDIO can be enumerated successfully. Here is an example:

```
root@imx8qxpmek:~# dmesg | grep mmc1
[    1.947683] mmc1: CQHCI version 5.10
[    2.007212] mmc1: SDHCI controller on 5b020000.usdhc [5b020000.usdhc] using ADMA
[    2.067920] mmc1: queuing unknown CIS tuple 0x01 (3 bytes)
[    2.075785] mmc1: queuing unknown CIS tuple 0x1a (5 bytes)
[    2.079252] mmc1: queuing unknown CIS tuple 0x1b (8 bytes)
[    2.079946] mmc1: queuing unknown CIS tuple 0x14 (0 bytes)
[    2.136241] mmc1: queuing unknown CIS tuple 0x80 (1 bytes)
[    3.484201] mmc1: queuing unknown CIS tuple 0x81 (1 bytes)
[    3.489830] mmc1: queuing unknown CIS tuple 0x82 (1 bytes)
[    3.495424] mmc1: new ultra high speed SDR104 SDIO card at address 0001
```

2.3.2. Loading WLAN Driver

Execute **insmod** to load WLAN driver. Here is an example:

```
# insmod wlan.ko
```

NOTE

insmod ./wlan.ko country_code=US is for setting Country as module parameter.

2.3.3. Bring up Wi-Fi

Execute the follow command to bring up Wi-Fi. Here is an example:

```
$ ifconfig wlan0 up
$ ifconfig wlan0
wlan0     Link encap:Ethernet  HWaddr 00:03:7f:10:72:12
          UP BROADCAST MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:3000
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)
```

2.4. Function Verification

The Wi-Fi working modes of Quectel FC20 series modules are STA mode and AP mode. This Chapter introduces the function verification methods for both STA mode and AP mode.

2.4.1. STA Mode

STA (Station) acts as a client in WLAN. The following steps show you how to test the Wi-Fi feature of FC20 series modules when the Wi-Fi working mode is STA mode.

2.4.1.1. Connecting AP in Open Mode

1. Set an AP, (take the AP that SSID is set to SoftAP and IP address is 192.168.1.1 as an example) and enable the dhcp server of the AP.

2. Modify the configuration file *wpa-sta.conf*. Here is an example:

```
ctrl_interface=/var/run/wpa_supplicant
update_config=1
network={
ssid="SoftAP"
key_mgmt=NONE
}
```

3. Connect the AP. Here is an example:

```
$ wpa_supplicant -D nl80211 -i wlan0 -c wpa-sta.conf &
```

4. Set IP address of wlan0 after connecting AP successfully. Here is an example:

```
$ ifconfig wlan0 192.168.1.2
```

5. Check the reachability of the IP address with **ping**. Here is an example:

```
$ ping 192.168.1.1
```

6. If the IP address can be pinged successfully, it indicates the module works normally in STA mode.

2.4.1.2. Connecting AP in Encryption Mode

1. Set an AP (take the AP that the encryption mode is WPA-PSK, SSID is SoftAP and IP address is 192.168.1.1 as an example), and enable the dhcp server of the AP.

2. Modify the configuration file *wpa-sta.conf*. Here is an example:

```
ctrl_interface=/var/run/wpa_supplicant
update_config=1
network={
ssid="SoftAP"
key_mgmt=WPA-PSK
auth_alg=OPEN
pairwise=CCMP
group=CCMP
psk="1234567890"
}
```

3. Connect the AP. Here is an example:

```
$ wpa_supplicant -Dnl80211 -iwlan0 -c wpa-sta.conf &
```

4. Set IP address of wlan0 after connecting AP successfully. Here is an example:

```
$ ifconfig wlan0 192.168.1.2
```

5. Check the reachability of the IP address with **ping**. Here is an example:

```
$ ping 192.168.1.1
```

6. If the IP address can be pinged successfully, it indicates the module works normally in STA mode.

2.4.2. AP Mode

AP indicates wireless access point. It is a creator of a wireless network and the central node of the network. The host has to install the programs hostapd and dhcp to configure wireless access point. Hostapd is a server end program running in user space and provides hotspot access and authentication. dhcp assigns IP address for the devices accessed through the wireless access point.

The configuration files *hostapd-2G.conf* and *hostapd-5G.conf* are located at *<FC20_target_root>/meta_build/load_meta/hostapd/*.

The following steps show you how to test the Wi-Fi feature of FC20 series modules when the Wi-Fi mode is AP mode with different wireless LAN standards (11bgn and 11ac).

2.4.2.1. Open Mode (11bgn)

1. Modify the default configuration file *hostapd.conf* and store it to */home/root/sbin* with hostapd program. Here is an example:

```
interface=wlan0
ssid=SoftAP
hw_mode=g
channel=1
auth_algs=1
ieee80211n=1
```

2. Execute the following command to run hostapd program.

```
# hostapd -dd hostapd.conf &
```

3. Set IP address of the SoftAP. Here is an example:

```
#ifconfig wlan0 192.168.11.1
```

4. Connect to the unencrypted AP with other Wi-Fi device that is in STA mode, and execute **ping** to check the reachability of the IP address.

2.4.2.2. Encryption Mode (11bgn)

1. Modify the configuration file *hostapd.conf*. Here is an example:

```
interface=wlan0
ssid=SoftAP
hw_mode=g
channel=1
auth_algs=3
ieee80211n=1
wpa=3
wpa_passphrase=12345678
wpa_key_mgmt=WPA-PSK
wpa_pairwise=CCMP
rsn_pairwise=CCMP
```

2. Execute the following command to run hostapd program:

```
#hostapd -dd hostapd.conf &
```

3. Set the IP address of SoftAP. Here is an example:

```
#ifconfig wlan0 192.168.11.1
```

4. Connect to the AP encrypted by WPA1/2 PSK with other Wi-Fi device that is in STA mode, and execute **ping** to check the reachability of the IP address.

2.4.2.3. Open Mode (11ac)

1. Modify the default configuration file *hostapd.conf* and store it to */home/root/sbin* with hostapd program. Here is an example:

```
interface=wlan0
ssid=SoftAP
hw_mode=a
channel=1
auth_algs=1
ieee80211n=1
ieee80211ac=1
```

2. Execute the following command to run hostapd program.

```
# hostapd -dd hostapd.conf &
```

3. Set IP address of the SoftAP. Here is an example:

```
#ifconfig wlan0 192.168.11.1
```

4. Connect to the unencrypted AP using other Wi-Fi devices that are in STA mode, and execute **ping** to check the reachability of the IP address.

2.4.2.4. Encryption Mode (11ac)

1. Modify the configuration file *hostapd.conf*. Here is an example:

```
interface=wlan0
ssid=SoftAP
hw_mode=a
channel=1
ieee80211n=1
ieee80211ac=1
wpa=3
wpa_passphrase=12345678
wpa_key_mgmt=WPA-PSK
wpa_pairwise=CCMP
rsn_pairwise=CCMP
```

2. Execute the following command to run hostapd program:

```
#hostapd -dd hostapd.conf &
```

3. Set the IP address of SoftAP. Here is an example:

```
#ifconfig wlan0 192.168.11.1
```

4. Connect to the AP encrypted by WPA1/2 PSK using other Wi-Fi devices that are in STA mode, and execute **ping** to check the reachability of the IP address.

2.5. Factory Test Mode

2.5.1. Configuring Factory Test Mode with QRCT

This chapter offers steps to perform factory test of Wi-Fi feature and the use of QRCT.

NOTES

1. The use of QRCT tool requires Qualcomm license. Please install and use the QRCT tool with the assistance of Quectel Technical Support (support@quectel.com).
2. The test device mentioned in this document takes LitePoint Solution as an example.

Set the module to factory test mode and start Wi-Fi. Please save the IP address of eth0 carefully for further use. Here is an example:

```
# insmod /lib/modules/wlan.ko con_mode=5
# ifconfig wlan0 up
# Qcmbr -v
```

```
root@imx6ullevk:~# ifconfig
eth0      Link encap:Ethernet  HWaddr 00:04:9F:04:F5:87
          inet addr:10.88.152.49  Bcast:10.88.152.255  Mask:255.255.255.0
          inet6 addr: fe80::204:9fff:fe04:f587%1995763091/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:3197 errors:0 dropped:74 overruns:0 frame:0
          TX packets:16 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:315874 (308.4 KiB)  TX bytes:2296 (2.2 KiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1%1995763091/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

wlan0     Link encap:Ethernet  HWaddr 00:0A:F5:89:89:80
          UP BROADCAST MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:3000
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)
```

```
imx6ullevk login: root
root@imx6ullevk:~# insmod /lib/modules/wlan.ko con mode=5
wlan: loading out-of-tree module taints kernel.
wlan: loading driver v4.5.25.46
hifDeviceInserted: Dumping clocks (50000000,132000000)
ol_download_firmware: chip_id:0x5020001 board_id:0x0
__ol_transfer_bin_file: Loading board data file utfbd30.b00
ar6k_wlan mmc0:0001:1: Direct firmware load for utfbd30.b00 failed with error -2
ar6k_wlan mmc0:0001:1: Falling back to user helper
__ol_transfer_bin_file: Failed to get utfbd30.b00:-11
__ol_transfer_bin_file: Trying to load default bdwlan30.bin
Board extended Data download address: 0x0
__ol_transfer_bin_file: no Setup file needed
__ol_transfer_bin_file: Loading firmware file utf30.bin
R0: wlan: [394:E :SAP] dfs_init_radar_filters[217]: Unknown dfs domain 0
Target Ready! : transmit resources : 10 size:1536, MaxMsgsPerHTCBundle = 32
TXRX: ol_tx_mark_first_wakeup_packet: pdev is NULL
cfg80211: failed to add phy80211 symlink to netdev!
ath_hif_sdio: HIF (Atheros/multi-bss)
wlan: driver loaded in 1120000
root@imx6ullevk:~# ifconfig wlan0 up
IPv6: ADDRCONF(NETDEV_UP): wlan0: link is not ready
root@imx6ullevk:~# Qcmbr -v
Enabling verbose mode
```

2.5.1.1. Running and Configuring Tool

1. Run QRCT.
2. Click "**Tool**" → "**User Defined Transport**" → "**OK**" and confirm the prompt message.

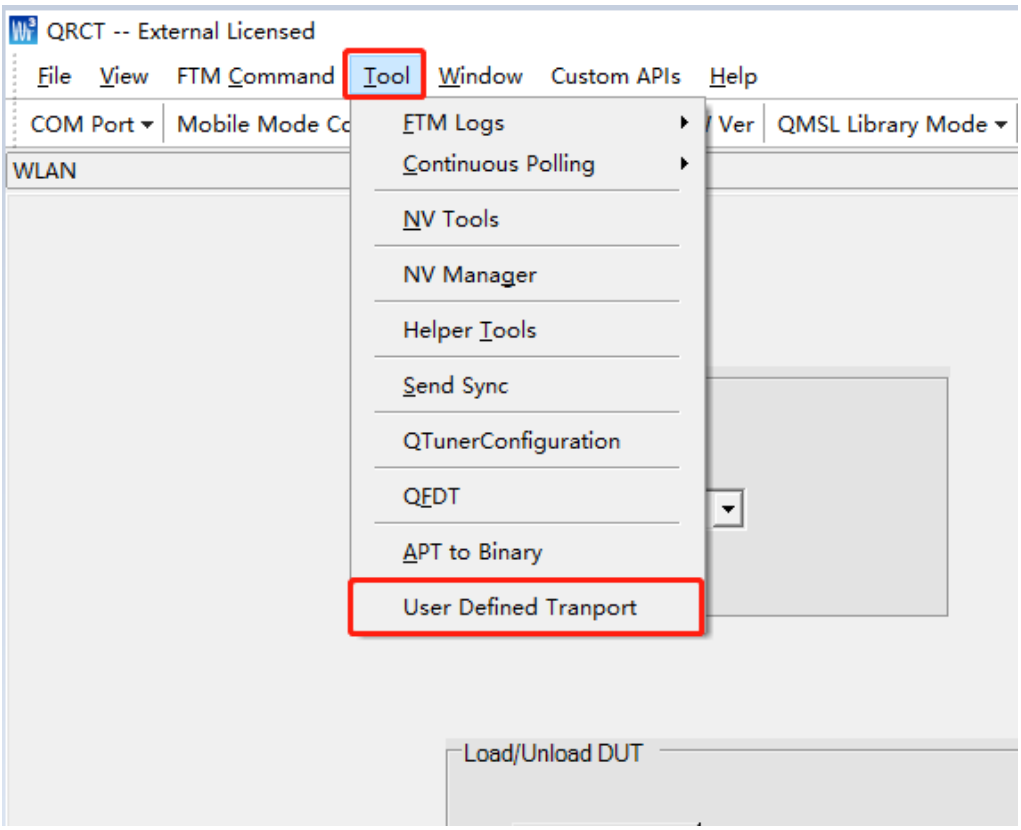


Figure 2: Enter "User Defined Transport"

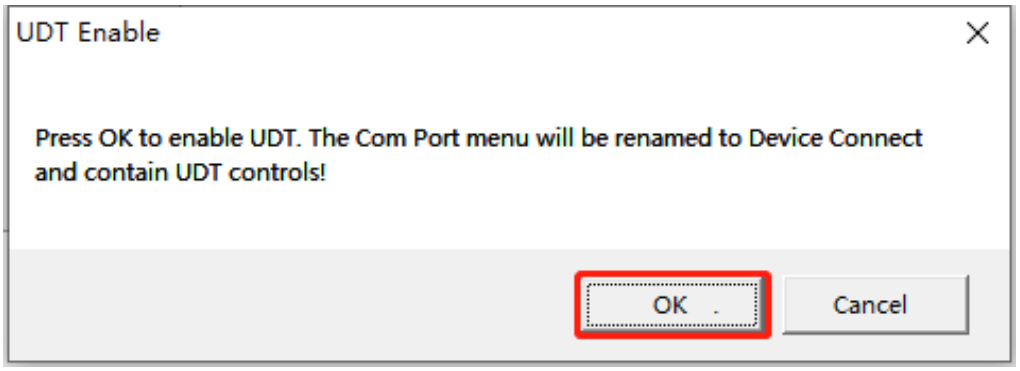


Figure 3: Confirm Enabling UDT

3. Continue to click "**Device Connect**" → "*** Add New UDT**".

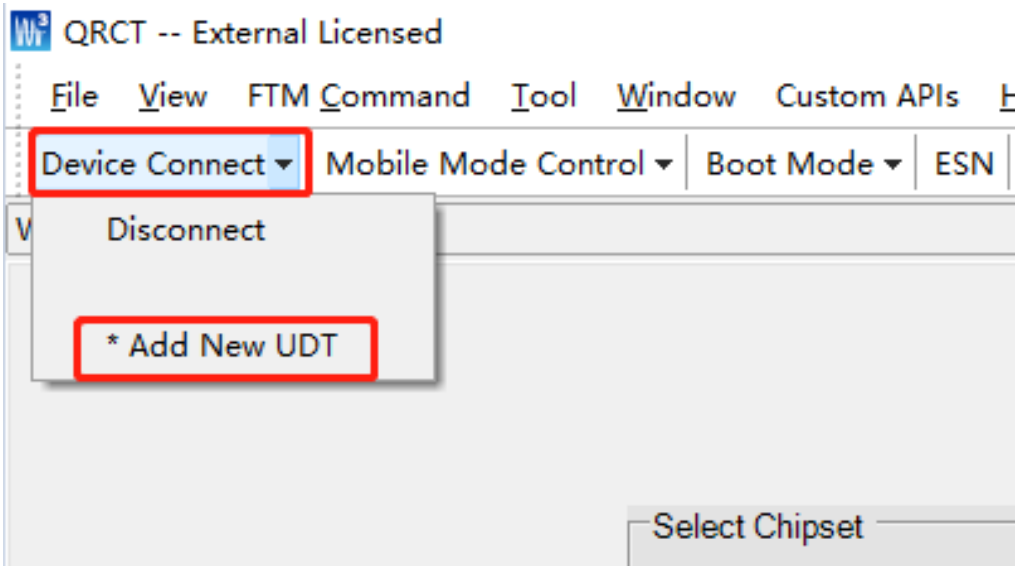


Figure 4: Add New UDT (WLAN)

In the popup box, select "Generic" from "Technology", input C:\Program Files (x86)\Qualcomm\QDART\bin\QMSL_WLAN_Transport.dll (the directory is created on the host automatically after installing QRCT. Please contact Quectel Technical Support if the directory does not exist) to "User Defined Transport DLL Path", and input the IP address of eth0 saved before to "IP Address" and "TCP Port". Then click "**Save**". Here is an example:

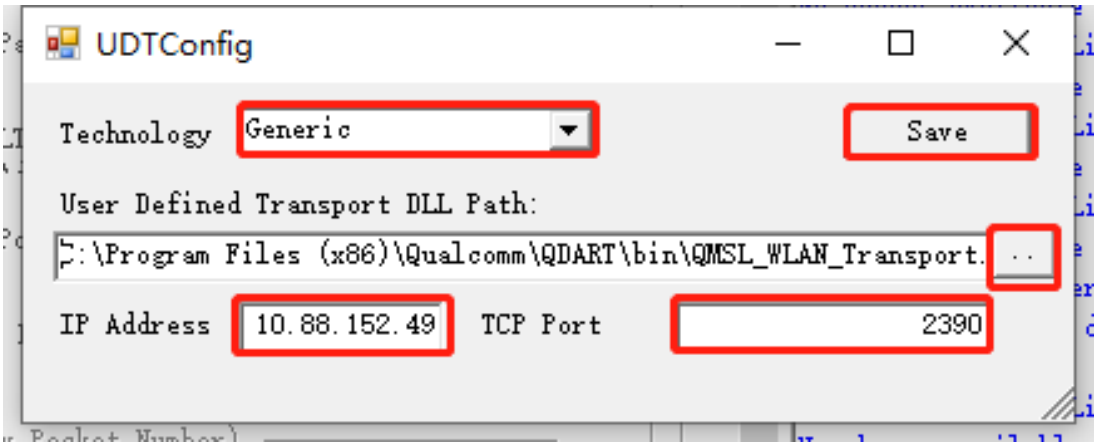


Figure 5: Configure UDT (WLAN)

4. After the configurations are saved, click "**Device Connect**" → "**10.88.152.49:2390(UDT Generic)**" to disconnect the connection.

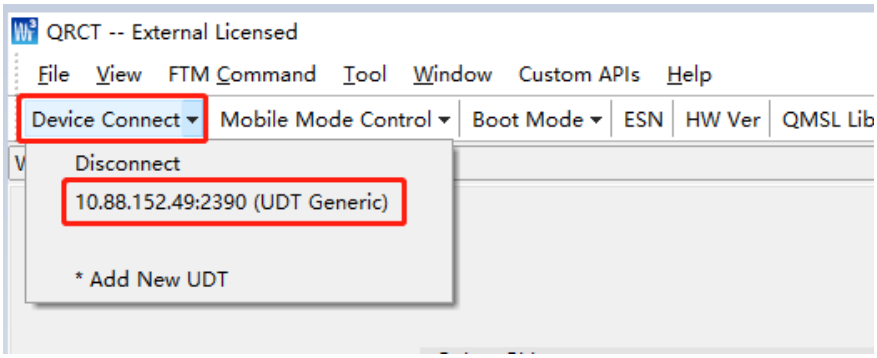


Figure 6: Disconnect (WLAN)

NOTE

This step must be performed, otherwise it may cause loading DUT failure. This step is also required when reopening the tool.

2.5.1.2. Testing WLAN

1. Click "**FTM Command**" → "**WLAN**" in sequence.

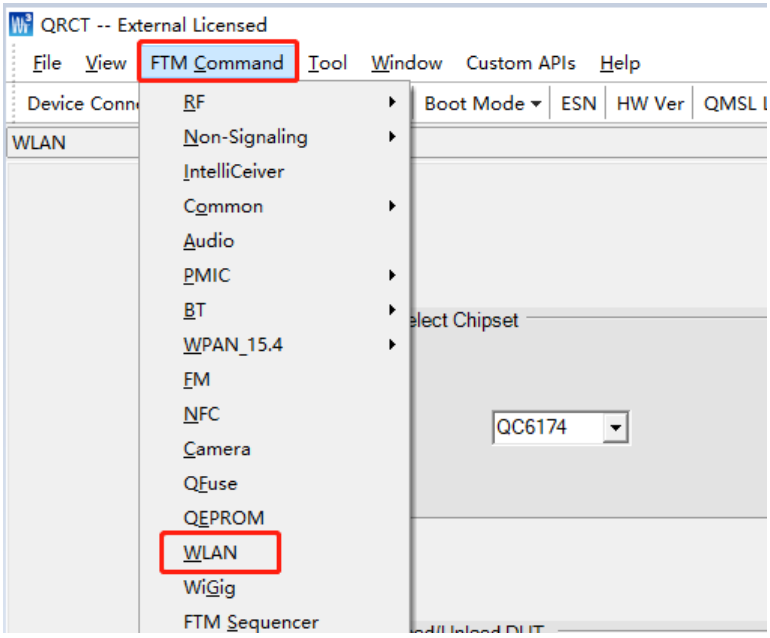


Figure 7: Select WLAN

Then select chipset QC6174, select BDF (*bdwlan.bin*), and click "**Load DUT**" to load DUT. Here is an example:

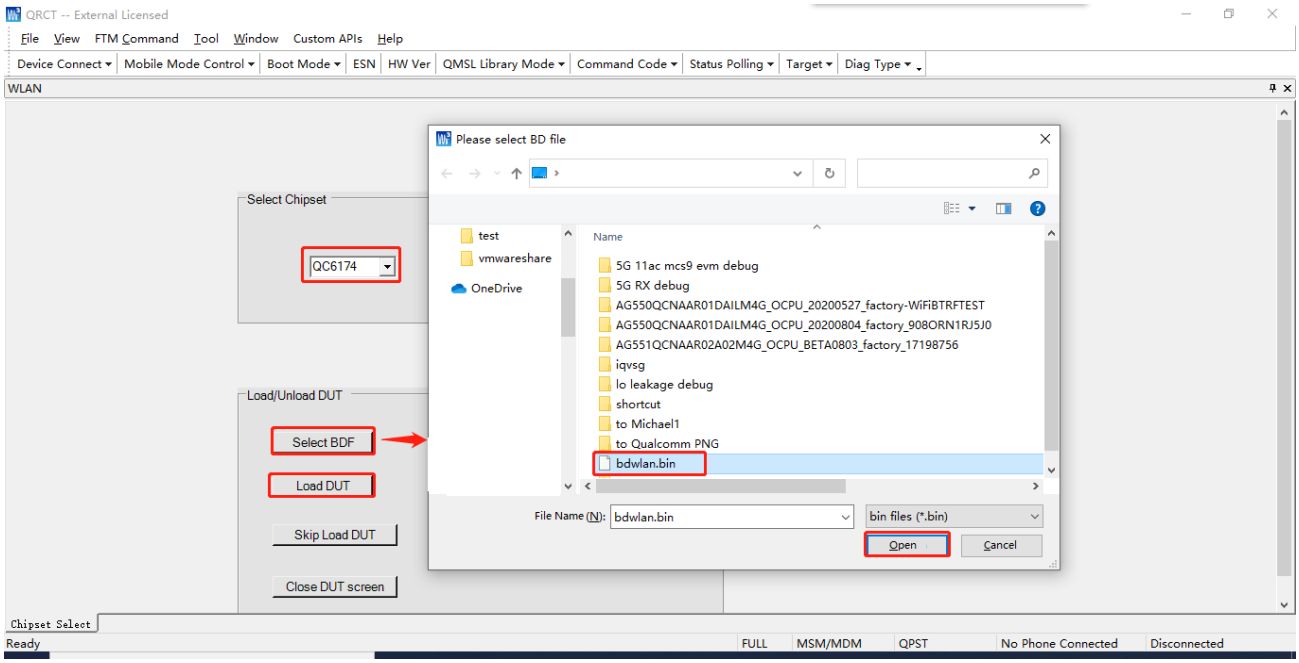


Figure 8: Load DUT

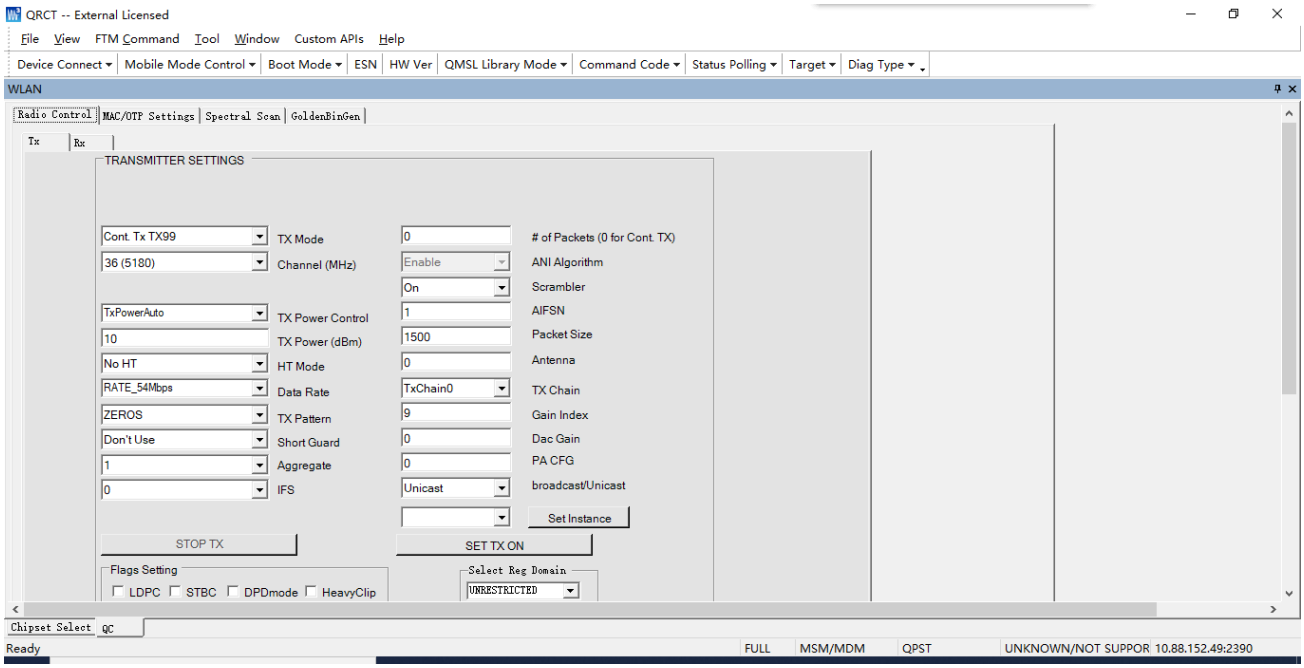


Figure 9: Load DUT Successfully

2. In “Tx” column, please select “Channel”, “HT Mode” and “Data Rate” (the data rate should be the test data) according to the actual practice. Then click “**SET TX ON**”. Here is an example:

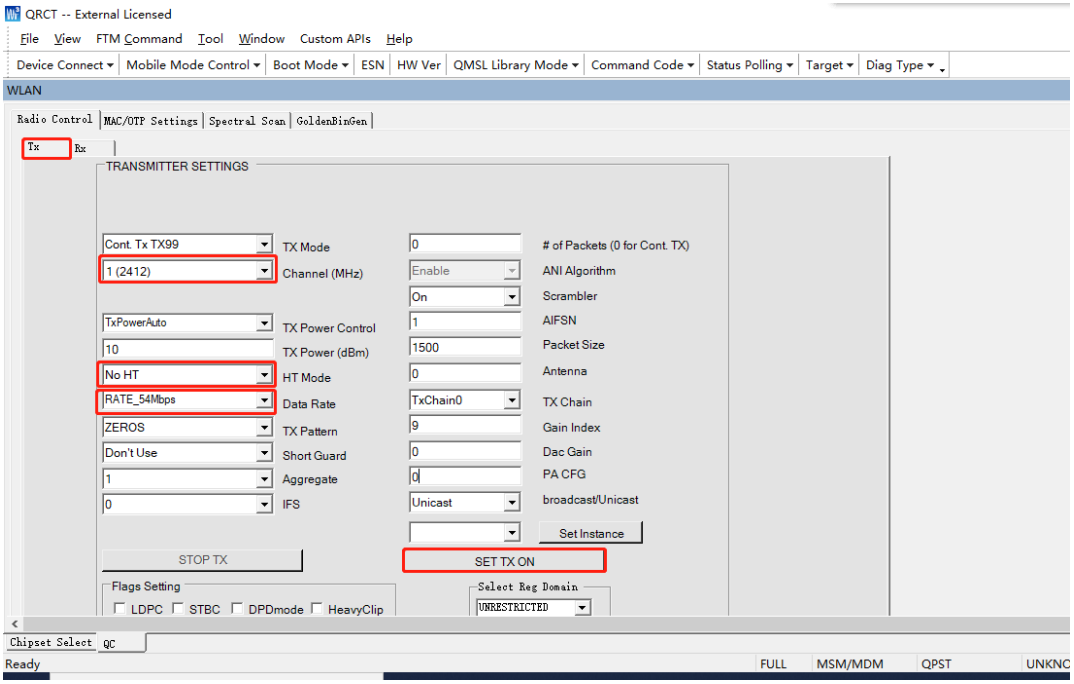


Figure 10: Configure TX

If TX power needs to be changed, please select “TxPowerForce_CLPC” as the figure below:

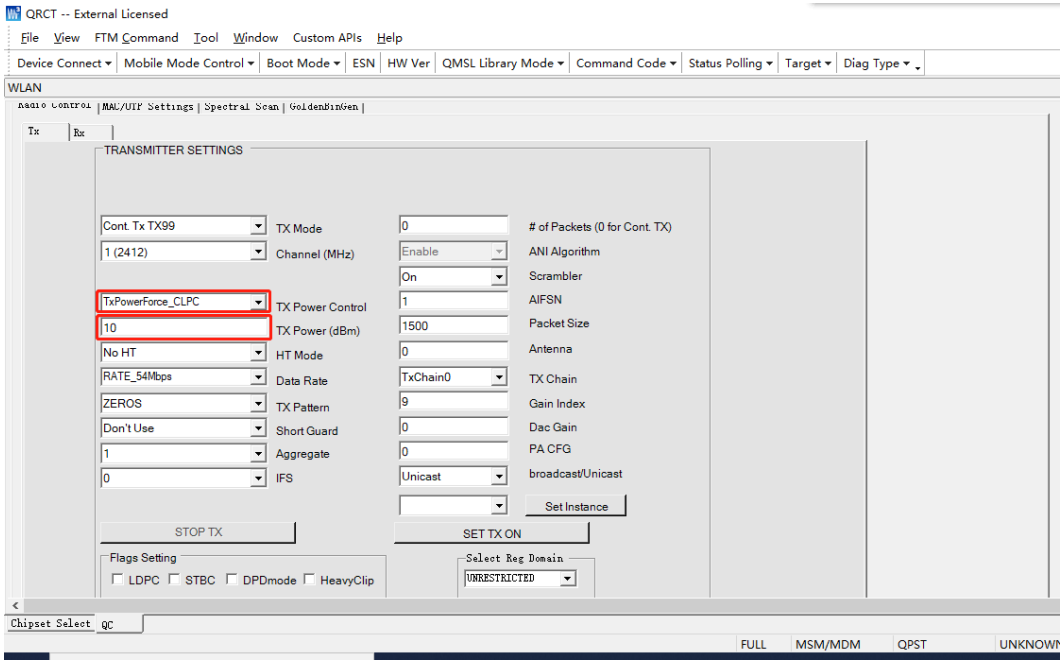


Figure 11: Change TX Power

After clicking "**SET TX ON**", the RF test device receives and displays the WLAN signal strength. Here is an example:

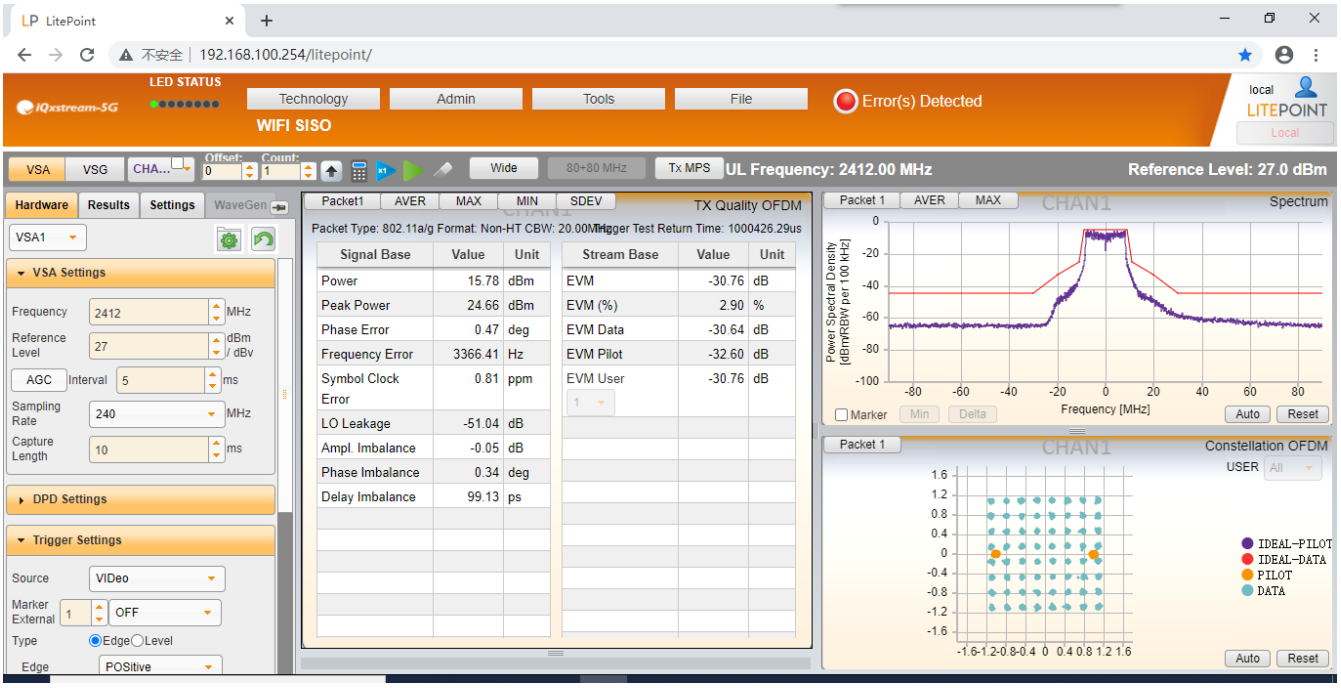


Figure 12: WLAN Signal Strength

3. The TX test of 5G is same as the TX test mentioned above. Here is an example:

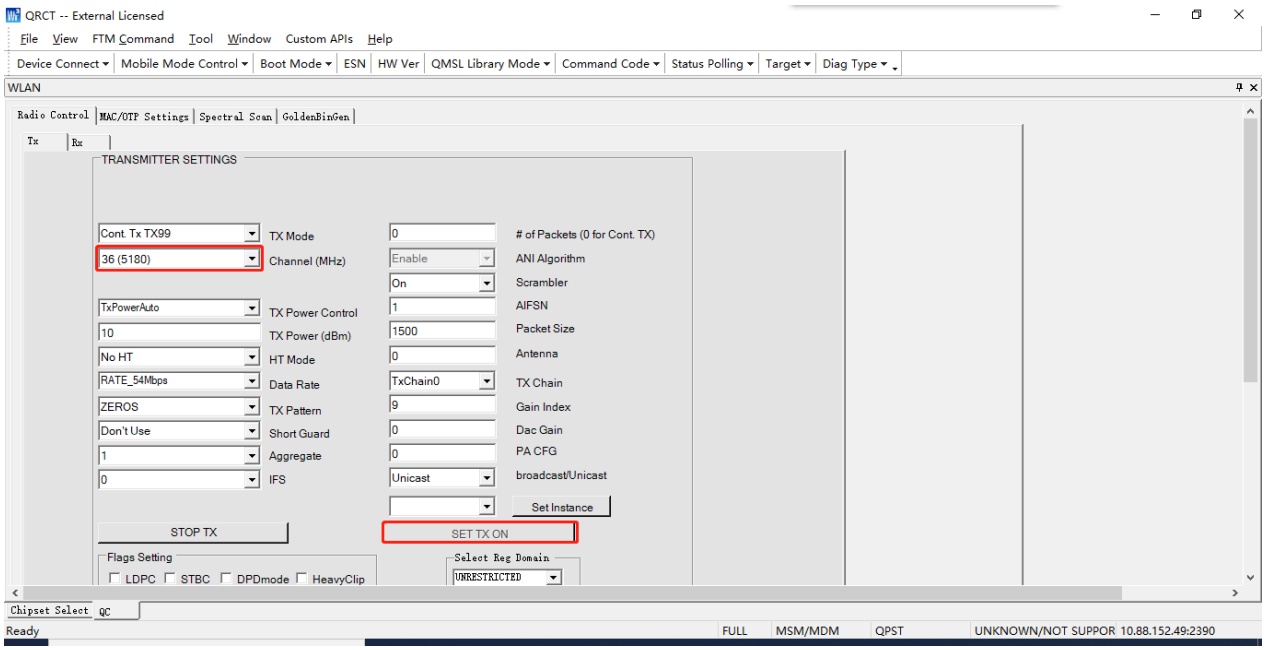


Figure 13: Set TX of 5G

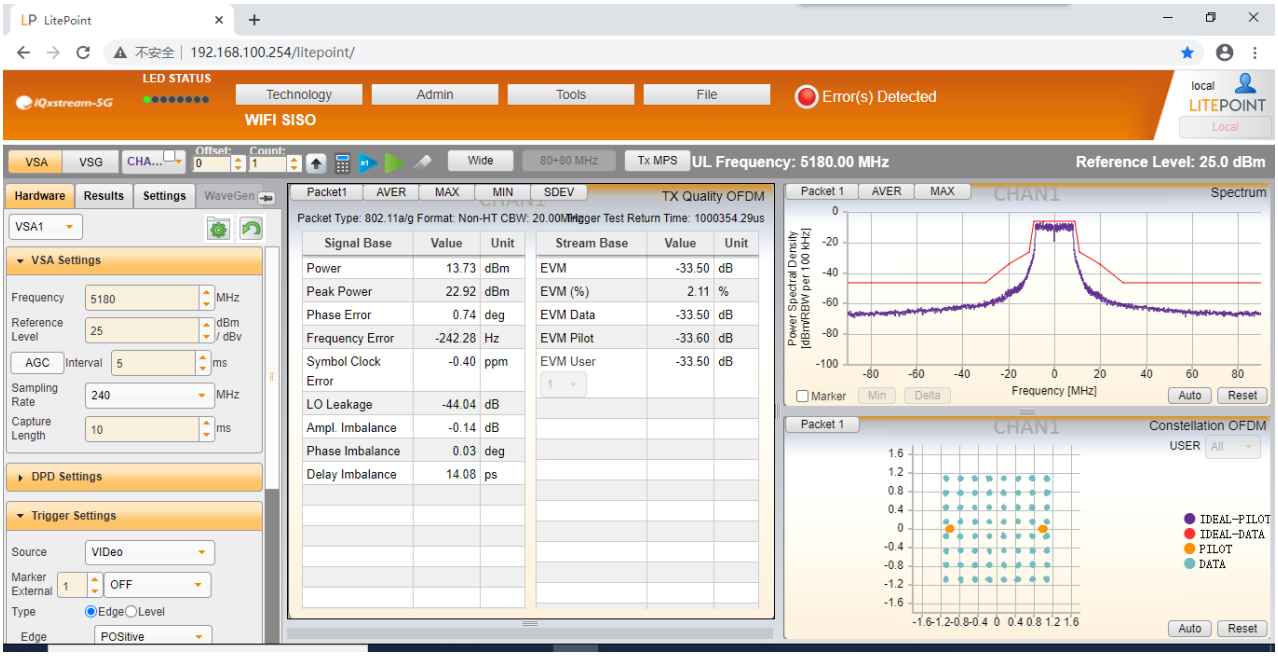


Figure 14: WLAN Signal Strength of 5G

4. In “Rx” column, select “Channel”, “HT Mode” and “Data Rate” (the data rate should be the test data) according to the actual practice, check “SET CONTINUOUS RX”, and then the RF test device starts sending signal automatically. Here is an example:

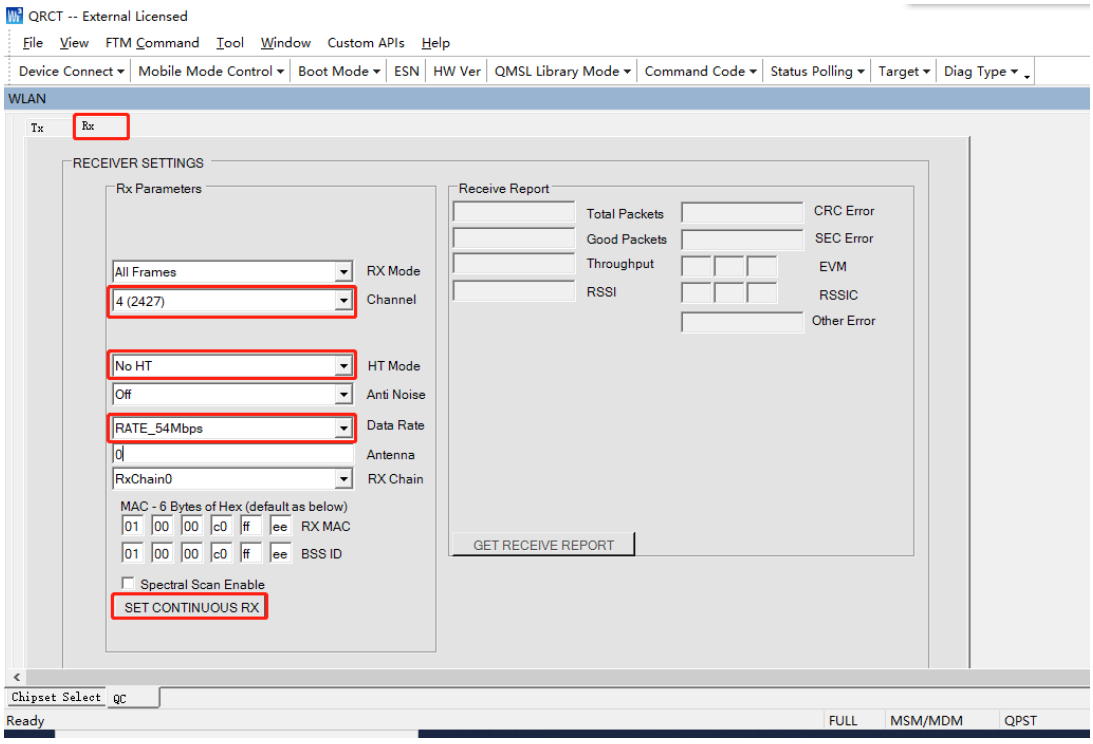


Figure 15: Configure RX

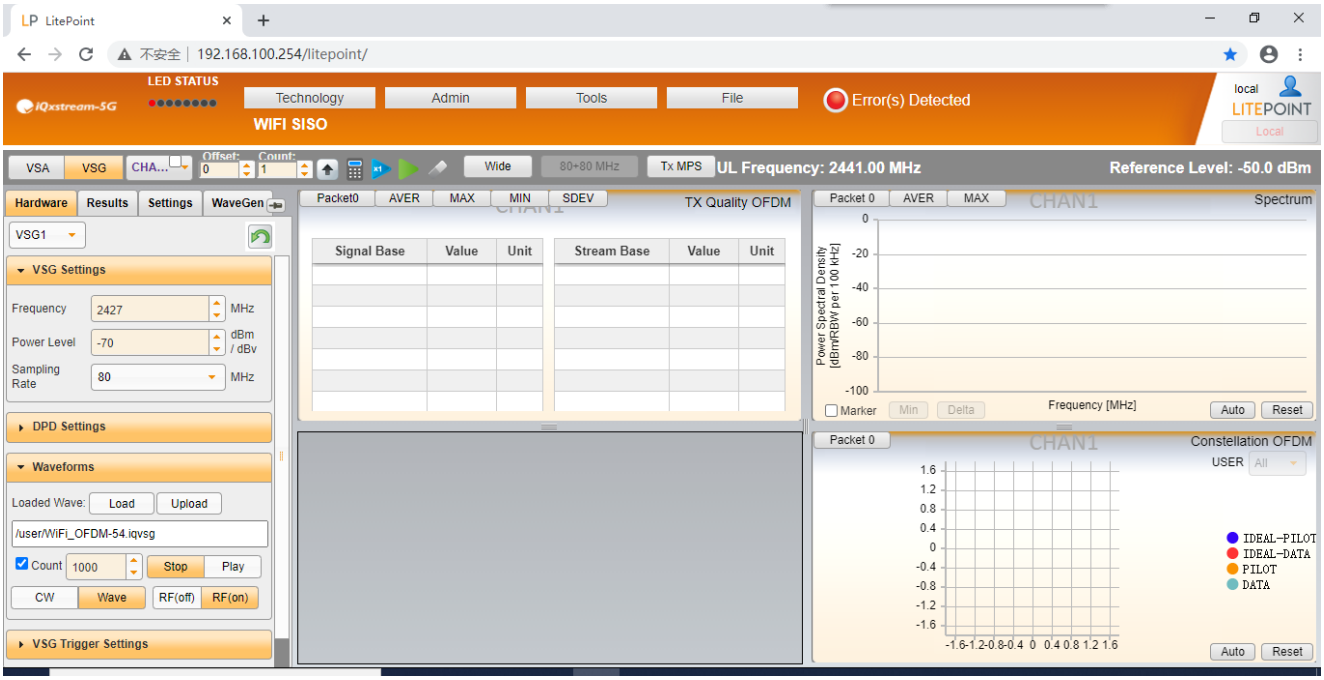


Figure 16: Device Sends Signal

Click "**GET RECEIVE REPORT**" on the QRCT tool to check the received data. Here is an example:

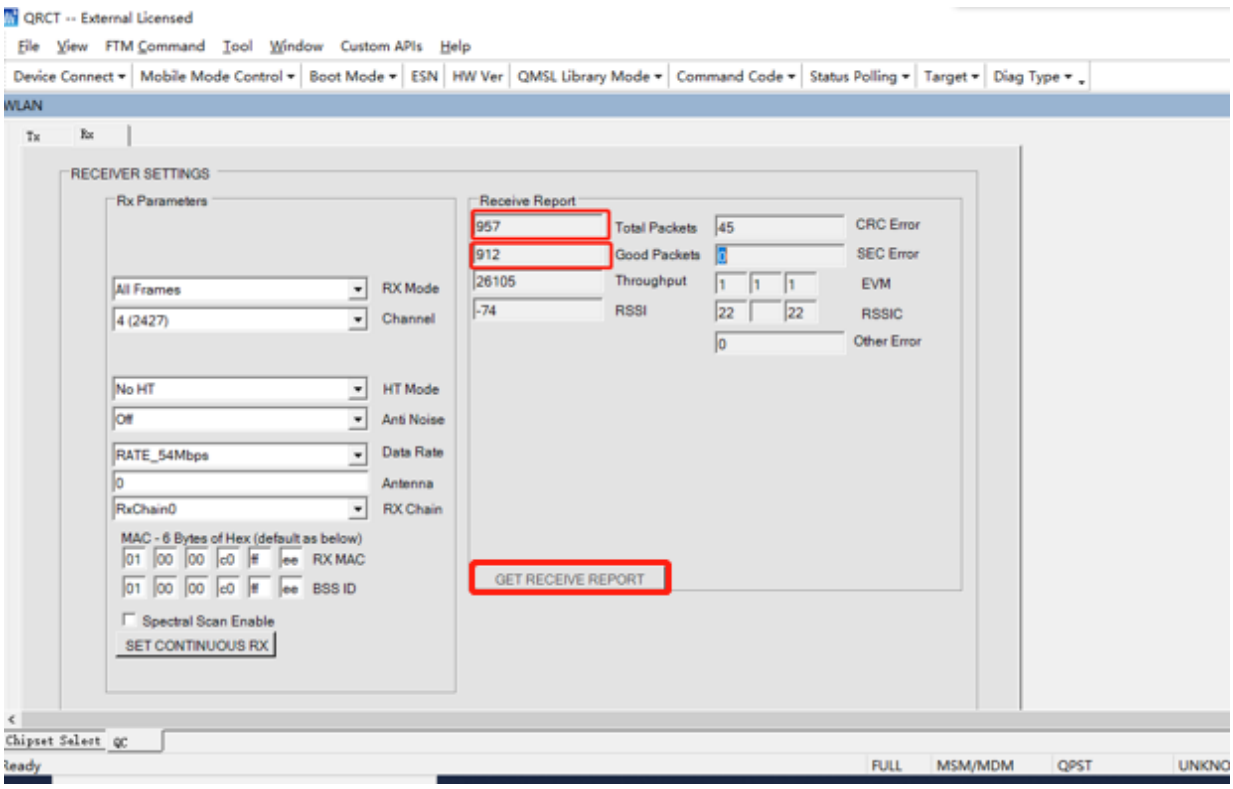


Figure 17: Check Received Data

5. RX test of 5G is same as the RX test mentioned above. Here is an example:

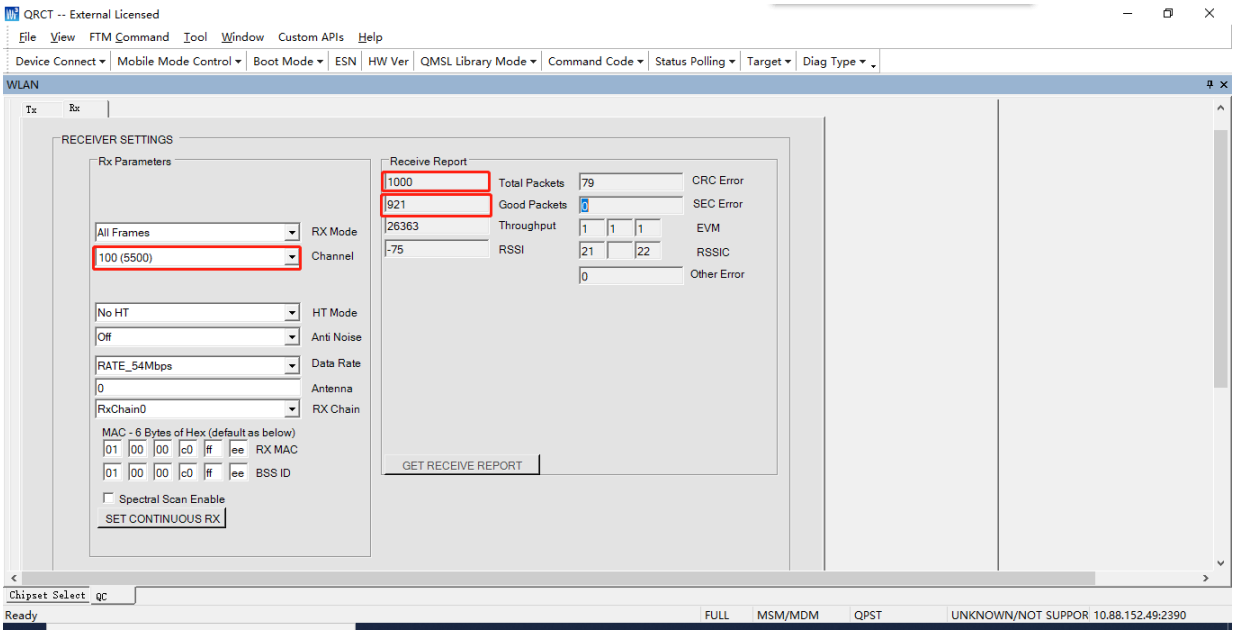
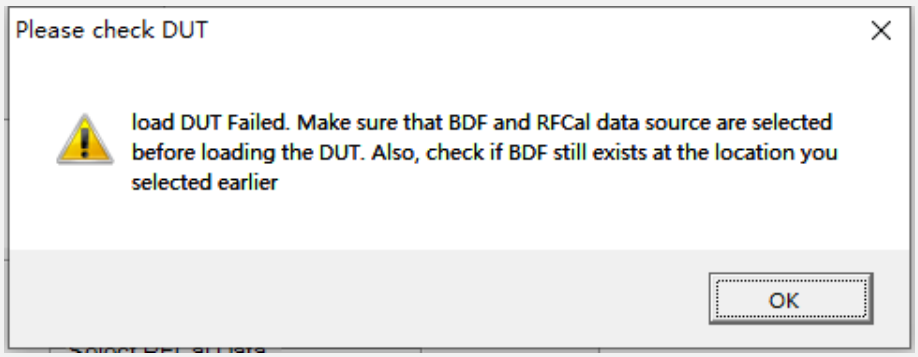


Figure 18: Set RX of 5G

NOTE

1. If the prompt of loading DUT failure, please input **Qcmbr –v** again after “Remote connection closed” is printed, and then reopen the QRCT.



2. The options of “HT Mode” in QRCT are defined as below:
   No HT indicates 11g mode;
   CCK indicates 11b mode;
   HT20 indicates 11n 20M mode;
   HT40 indicates 11n 40M mode;
   VHT20 indicates 11ac 20M mode;
   VHT40 indicates 11ac 40M mode;
   VHT80 indicates 11ac 80M mode.

2.5.2. Configuring Factory Test Mode with Command

2.5.2.1. Enabling WLAN in FTM

Execute the following command to enable WLAN in FTM:

```
insmod wlan.ko con_mode=5
ifconifg wlan0 up
```

2.5.2.2. Command List of MyFTM

MyFTM is a command-line tool that is used to perform the WLAN FTM test. Following are the commands to support TLV2.0 message format:

Table 1: Command List

Option	Functionality	Range	Description
-r	WlanATSetRate	1–30	See ***Table 2***.
-y	STBC	1\0	1 Enable 0 Disable
-z	LDPC	1\0	1 Enable 0 Disable
-f	WlanATSetWifiFreq	2412–5825	Frequencies.
-p	WlanATSetWifiTxPower	0–20	Command power, such as 15 (supports .5 increments, so 7.5 is a valid settings)
-a	WlanATSetWifiAntenna	1	Chain 0
		2	Chain 1
		3	Both Chain 0 and Chain 1 (2 x 2)
		65535	Smart antenna mode
-G	Gain Index	0–31	Gain index is considered only if TPC is set to 4
-D	Dac Gain	-48–128	Dac Gain is considered only if TPC is set to 4
-j	Number of packets	-	Number of packets to be transmitted; by default, it is 0, which means to continue Tx.
-k	Aggregation	0	Disable Aggregation (default value)

		1	Enable Aggregation
-M	WLAN mode	0	TCMD_WLAN_MODE_NOHT
		1	TCMD_WLAN_MODE_HT20
		2	TCMD_WLAN_MODE_HT40PLUS
		4	TCMD_WLAN_MODE_CCK
		5	TCMD_WLAN_MODE_VHT20
		6	TCMD_WLAN_MODE_VHT40PLUS
		8	TCMD_WLAN_MODE_VHT80_0
-C	PA CFG	0–3	
-c	TPC	0	TPC_TX_PWR
		1	TPC_FORCED_GAIN
		2	TPC_TGT_PWR
		3	TPC_TX_FORCED_GAIN
		4	TPC_FORCED_GAINIDX
		5	TPC_FORCED_TGTPWR
-I	Enable long preamble	-	Enable long preamble; by default, short preamble is enabled
-s	Tx packet size	Valid packet size	By default, it is set to 1500
-B	Phy RF Mode	dbs\phya\phyb\sbs	Sbs Configure Phy RF Mode
-H	Rx Mode	0\1\2	0 Promis (Default) 1 Filter 2 Report
-I	Phy Id	0\1	0 Phyid-0 1 Phyid-1
-J	Enable TLV2.0	-	Enable TLV 2.0 command format messages
-N	BSS MAC	BSS ID	BSS ID.
-O	Station MAC	STA Address	Station MAC Address
-o	BT MAC	BT Address	BT MAC Address
-Q	Frequency2	2412–5825	Secondary Frequency

-U	ShortGuard	0\1	0 Short Guard Off 1 Short Guard On
-X	TX Station MAC	TX STA Address	TX Station MAC Address
-Y	RX Station MAC	RX STA Address	RX Station MAC Address
-t	WlanATSetWifiTX	0	Tx off
		1	TCMD_CONT_TX_SINE
		2	TCMD_CONT_TX_FRAME
		3	TCMD_CONT_TX_TX99
-x	WlanATSetWifiRX	0	Rx stop
		1	Rx start
--fecpad	UL-OFDMA Tx afactor	255	AUTO. Take care of it by FW
--ldpc	UL-OFDMA Tx LDPCExtraSymbol	255	AUTO. Take care of it by FW
-nss	data stream	1–8	1 1 stream 2 2 streams 3 3 streams
--dcm	OFDMA Dual Carrier Modulation	0	Disable DCM
		1	Enable DCM
--gi	LTF GI	2	GI_800
		19	1x LFT + GI_1600
		35	2x LFT + GI_1600
		52	4x LFT + GI_3200
--ppdutype	PPDU type	0	Single user
		1	Multiple user
		2	ExtRange Single user
		3	Trigger
--linkdir	Link Direction	0	Up
		1	Down
--rateB	bandwidth for different	0	CCK

w	mode	1	LegacyOFDM
		2	11N_HT20
		3	11N_HT40
		4	11AC_VHT20
		5	11AC_VHT40
		6	11AC_VHT80
		8	11AX_HE20
		9	11AX_HE40
		10	11AX_HE80
		12	11AX_OFDMA_HE20
		13	11AX_OFDMA_HE40
		14	11AX_OFDMA_HE80
--toneplan	OFDMA tone plan	-	-
--dutycl	Duty cycle	1–9	TX duty cycle

Table 2: Rate Index

Rate Index	Data rate	Note
1	1 M	Legacy 802.11a/b/g data rate
2	2 M	
3	5.5 M	
4	6 M	
5	9 M	
6	11 M	
7	12 M	
8	18 M	
10	24 M	

12	36 M	
13	48 M	
14	54 M	
15	MCS0	802.11n or 802.11ac HT/VHT data rate
16	MCS1	
17	MCS2	
18	MCS3	
19	MCS4	
20	MCS5	
21	MCS6	
22	MCS7	
23	MCS8	
24	MCS9	
25	MCS10	
26	MCS11	
27	MCS12	
28	MCS13	
29	MCS14	
30	MCS15	

2.5.2.3. TX Performance Test Example (5G)

1. Execute the following command to Set the WLAN to continue to TX with 54Mbps rate on 11a mode on chain0 with TLV2 format:

```
myftm -J -I 0 --gl 2 --nss 1 --rateBw 1 -M 0 -r 14 -f 5500 -c 0 -p 16 -a 1 -X 00:03:7F:44:55:69 -Y 00:03:7F:44:55:68 -N 00:03:7F:44:55:70 -t 3
```

   The terminal prints the following logs:

```
doCommand() ifname wlan0 cmdStreamLen 512
cmdID 5 response needed
nl80211: sending message
setting timer
nl80211: waiting for response
nl80211: cb wrapper called
nl80211: resp received, calling custom cb
resetting timer

 parseTlv2Response() TLV2 length got 76
05 00 00 00 02 00 00 00 00 00 00 00 30 00 00 00
8e f1 00 00 00 00 00 00 00 00 00 00 0d 00 00 00
1b 54 d0 28 03 00 00 00 34 00 00 00 f8 7d d7 4b
0e 00 00 00 a6 00 00 00 14 52 e6 44 00 00 00 00
29 00 00 00 4f 7e a4 2c 00 00 00 00
Response cmdId: 14
Response status 0
qca6174TxCommand() Success = 0
W1anATSetWifiTX done with sucess
/ # _
```

2. Control the VSA to receive relevant packets.

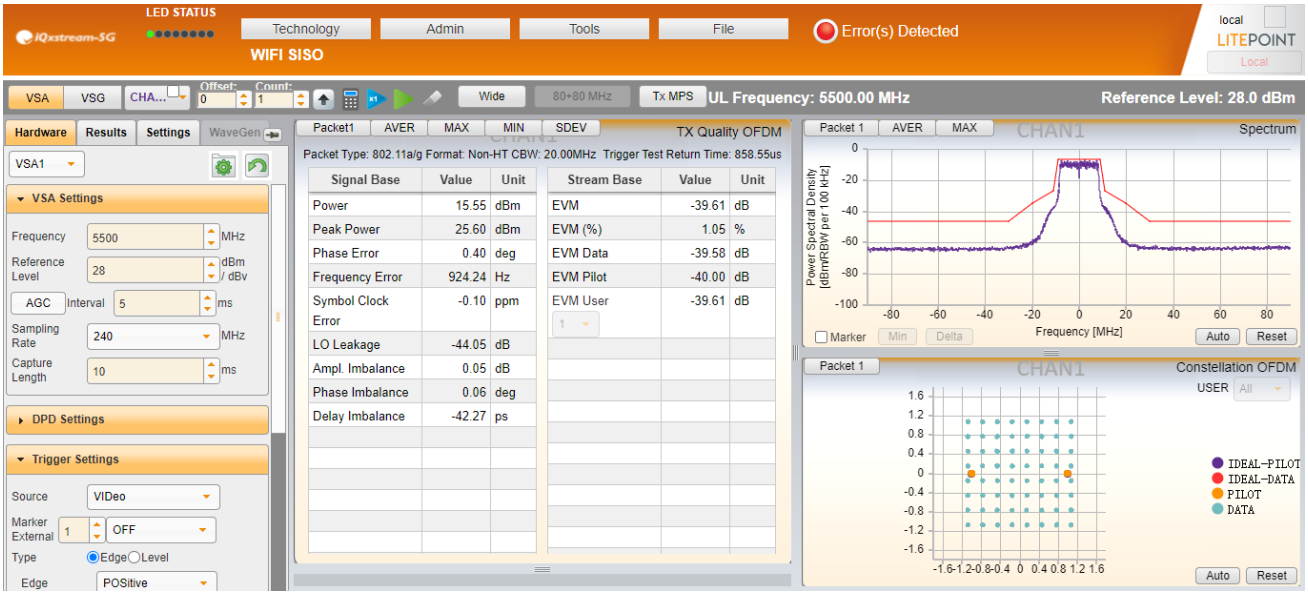


Figure 19: Receive Relevant Packets

3. Execute the following command to stop TX.

```
myftm -J -I 0 -t 0
```

2.5.2.4. RX Performance Test Example (5G)

- **Example A**

1. Execute the following commands to set the WLAN start to Rx 54Mbps rate packet of 11a on frequency 5500 and the antenna on chain0 with TLV2 format:

```
myftm -J -I 0 --nss 1 --rateBw 1 -M 0 -r 14 -f 5500 -a 1 -x 1
```

The terminal prints the following logs:

```
doCommand() ifname wlan0 cmdStreamLen 356
cmdID 5 response needed
n180211: sending message
setting timer
n180211: waiting for response
n180211: cb wrapper called
n180211: resp received, calling custom cb
resetting timer

 parseTlv2Response() TLV2 length got 76
05 00 00 00 02 00 00 00 00 00 00 00 30 00 00 00
c6 d7 00 00 00 00 00 00 00 00 00 00 15 00 00 00
75 28 3f 08 03 00 00 00 a6 00 00 00 14 52 e6 44
00 00 00 00 29 00 00 00 4f 7e a4 2c 00 00 00 00
70 00 00 00 73 3e c7 72 00 00 00 00
status 0

qca6174RxPacketStart() success = 0
/ # _
```

2. Control VSG to transmit relevant packets.

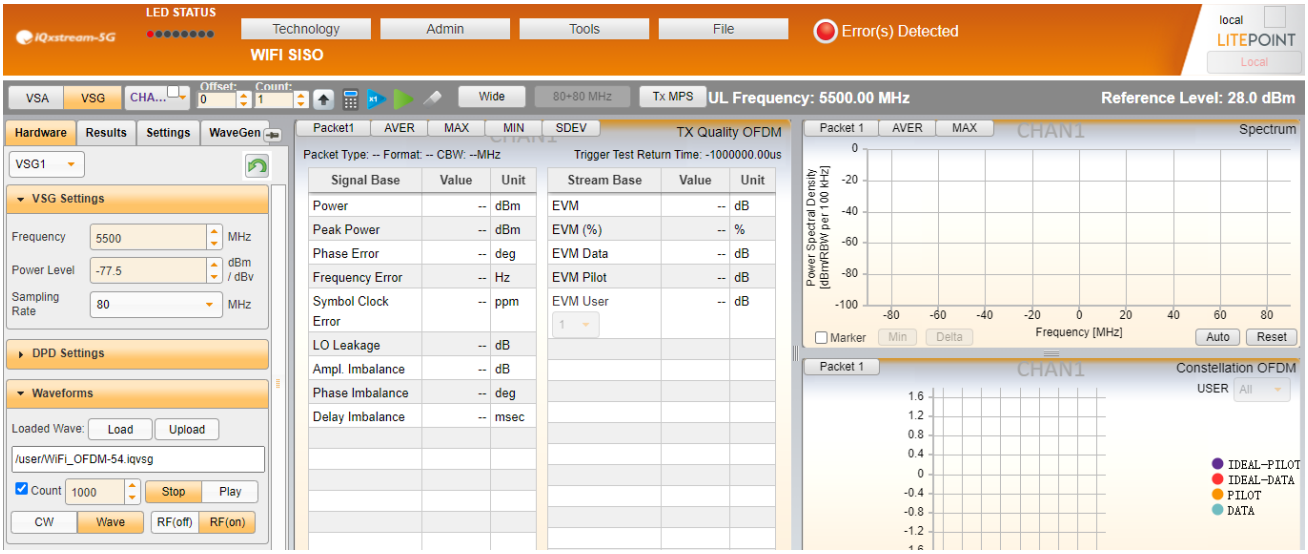


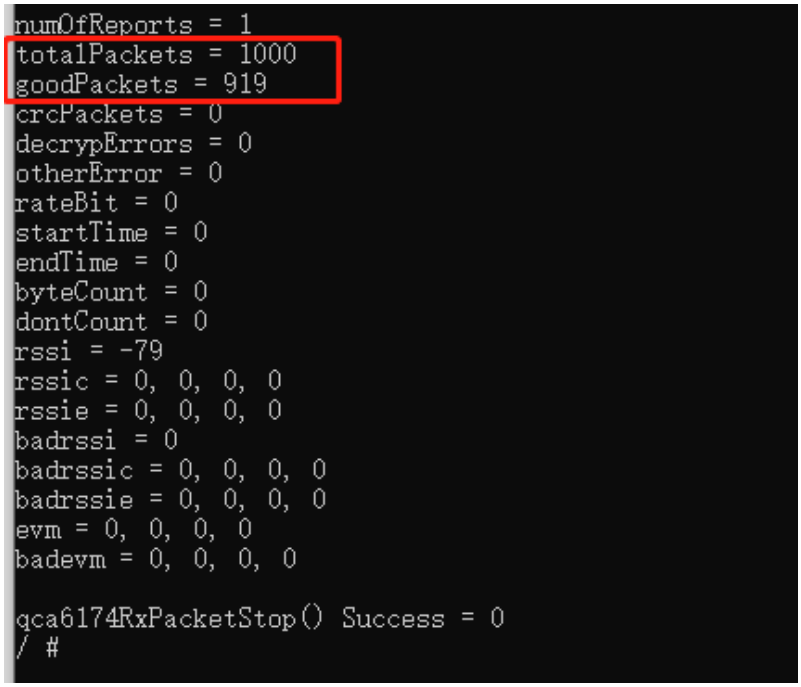
Figure 20: Transmit Relevant Packets

- **Example B**

1. Execute the following commands to Set the WLAN stop to Rx 54Mbps rate packet of 11a on frequency 5500 and the antenna on chain0 with TLV2 format, then exit RX mode:

```
myftm -J -I 0 --nss 1 --rateBw 1 -M 0 -r 14 -f 5500 -a 1 -x 0
```

The terminal prints the following logs:



2.5.2.5. RX&TX Performance Test Examples (2.4G)

1. Execute the following command to start TX (ANT0 2412 54M):

```
myftm -J -B dbs
myftm -J -I 1 --gl 2 --nss 1 --rateBw 1 -M 0 -r 14 -f 2412 -c 0 -p 16 -a 1 -X 00:03:7F:44:55:69 -Y 00:03:7F:44:55:68 -N 00:03:7F:44:55:70 -t 3
```

2. Execute the following command to stop TX:

```
myftm -J -I 1 -t 0
```

3. Execute the following command to start RX (ANT0 2412 54M):

```
myftm -J -B dbs
myftm -J -I 1 --nss 1 --rateBw 1 -M 0 -r 14 -f 2412 -a 1 -x 1
```

4. Execute the following command to stop RX:

```
myftm -J -I 1 --nss 1 --rateBw 1 -M 0 -r 14 -f 2412 -a 1 -x 0
```

2.6. Logs Capture

2.6.1. Getting WLAN Host and Firmware Version

1. Execute the following command to get WLAN host and firmware version:

```
# iwpriv wlan0 version
```

2. Execute the following command to get the configurations of Wi-Fi driver:

```
# iwpriv wlan0 getConfig
```

2.6.2. Enabling WLAN Host Logs

1. Add the following contents to the file *qcom_cfg.ini*, reload WLAN module and check kernel log along with issue.

```
vosTraceEnableTL=255
vosTraceEnableWDI=255
vosTraceEnableHDD=255
vosTraceEnableSME=255
vosTraceEnablePE=255
vosTraceEnablePMC=255
vosTraceEnableWDA=255
vosTraceEnableSYS=255
vosTraceEnableVOSS=255
vosTraceEnableSAP=255
vosTraceEnableHDDSAP=255
vosTraceEnableCFG=255
vosTraceEnableADF=255
vosTraceEnableTXRX=255
vosTraceEnableHTC=255
vosTraceEnableHIF=255
vosTraceEnableHDDSAPDATA=255
vosTraceEnableHDDDATA=255
vosTraceEnableHIF=255
vosTraceEnableHTC=255
vosTraceEnableTXRX=255
```

```
vosTraceEnableADF=255
vosTraceEnableCFG=255
```

2. The WLAN logging tool cnss_diag_lite routes WLAN driver verbose messages and firmware debug information to console or saves the logs in a file.

   Some of the options of the logging tool is listed as below:

   -f saves log to file by text format (same path with tool, folder is *wlan_logs*)
   -c enables host log (no parameter value)
   -x enables FW diagnose message (0 indicates disable, 1–31 indicate enable)
   -e enables FW diagnose event/log (0 indicates disable, 1 indicates enable)
   -l saves log to file by QXDM format (same path with tool, folder is *wlan_logs*)

NOTE

1. Currently only host log and FW diagnose message could be shown in console window.
2. FW diagnose event/log only could be saved into file and parsed by QXDM tool.
3. The options can be given in any order.

4. Execute the flowing command to enable host log and firmware log, and save the logs to console files by test format.

```
# cnss_diag_lite -c > host_wlan_log &
```

2.6.3. Enabling Module Firmware Debug Logs

1. You can enable WLAN firmware debug logs by modifying *qcom_cfg.ini*:

```
# Enable firmware uart print
gEnablefwprint=0

# Enable firmware log
gEnablefwlog=1
gMulticastHostFwMsgs=1
gFwDebugLogType=255
# Additional firmware log levels
gFwDebugLogLevel=0
gFwDebugModuleLoglevel=1,0,2,0,4,0,5,0,6,0,7,4,8,0,9,0
```

Or executing the following command:

```
#First enable firmware log
gEnablefwlog=1
```

2. Configure log level:

```
Syntax: iwpriv wlan0 dl_loglevel <0-5>
    Supported log levels
    DBGLOG_VERBOSE = 0,
    DBGLOG_INFO,
    DBGLOG_INFO_LVL_1,
    DBGLOG_INFO_LVL_2,
    DBGLOG_WARN,
    DBGLOG_ERR,
    DBGLOG_LVL_MAX
Syntax: iwpriv wlan0 dl_type <0-3>
    DBGLOG_PROCESS_DEFAULT = 0
    DBGLOG_PROCESS_PRINT_RAW = 1
    DBGLOG_PROCESS_POOL_RAW =2
    DBGLOG_PROCESS_NET_RAW = 3
Syntax: iwpriv wlan0 dl_report <0-1>
    0: report stops
    1: report starts
```

2.6.4. Obtaining WLAN Target Firmware Statistic Separately

1. Execute the following command to get the target physical device statistics:

```
# iwpriv wlan0 txrx_fw_stats 1
```

2. Execute the following command to receive reorder statistics:

```
# iwpriv wlan0 txrx_fw_stats 2
```

3. Execute the following command to receive rate control statistics

```
# iwpriv wlan0 txrx_fw_stats 3
```

4. Execute the following command to transmit rate control statistics

```
# iwpriv wlan0 txrx_fw_stats 6
```

5. Execute the following command to clear firmware statistics corresponding to bit set by <mask>. If <mask>=0x1f, all statistics would be cleared.

```
iwpriv wlan0 txrx_fw_st_rst <mask>
```

2.7. Commands to Fix Rate and Bandwidth

2.7.1. Configuring NSS and MCS

1. The following command configures NSS:

```
iwpriv < interface> nss <nss>
```

<nss> Number of spatial streams.
1 1x1 MCS
2 2x2 MCS
3 3x3 MCS

2. The following command configures MCS:

```
iwpriv < interface> set11ACRates <MCS>
```

<MCS> Modulation and coding scheme. Range: 0–9.

3. For HT rates, the following command controls both NSS and MCS:

```
iwpriv < interface> set11NRates 0x80..0x97 (for HT MCS0..23)
```

2.7.2. Configuring CCK Rates

1. For rate 1 Mbps:

```
iwpriv <interface> set11NRates 0x1b1b1b1b
```

2. For rate 2 Mbps:

```
iwpriv <interface> set11NRates 0x1a1a1a1a
```

3. For rate 5.5 Mbps:

```
iwpriv <interface> set11NRates 0x19191919
```

4. For rate 11 Mbps:

```
iwpriv <interface> set11NRates 0x18181818
```

2.7.3. Configuring OFDM Rates

1. For rate 6 Mbps:

```
iwpriv <interface> set11NRates 0x0b0b0b0b
```

2. For rate 9 Mbps:

```
iwpriv <interface> set11NRates 0x0f0f0f0f
```

3. For rate 12 Mbps:

```
iwpriv <interface> set11NRates 0x0a0a0a0a
```

4. For rate 18 Mbps:

```
iwpriv <interface> set11NRates 0x0e0e0e0e
```

5. For rate 24 Mbps:

```
iwpriv <interface> set11NRates 0x09090909
```

6. For rate 36 Mbps:

```
iwpriv <interface> set11NRates 0x0d0d0d0d
```

7. For rate 48 Mbps:

```
iwpriv <interface> set11NRates 0x08080808
```

8. For rate 54 Mbps:

```
iwpriv <interface> set11NRates 0x0c0c0c0c
```

2.7.4. Configuring Auto Rate Mode

The following command switches rate mode to auto rate run-time after fixing rate:

```
iwpriv <interface> set11NRates 0
```

2.7.5. Controlling Channel Bandwidth

The following command restricts the transmission bandwidth for both fixed and auto rate:

```
iwpriv <interface> chwidth <bw_idx>
```

<bw_idx> Bandwidth.
0 20 Mhz
1 40 Mhz
2 80 Mhz

3 Enabling Bluetooth

3.1. Operations

The introduction of how to enable FC20 Bluetooth function on the third-party platform is to be added later.

3.2. Factory Test Mode

This chapter offers steps to perform the factory test of Bluetooth feature and the use of QRCT.

NOTES

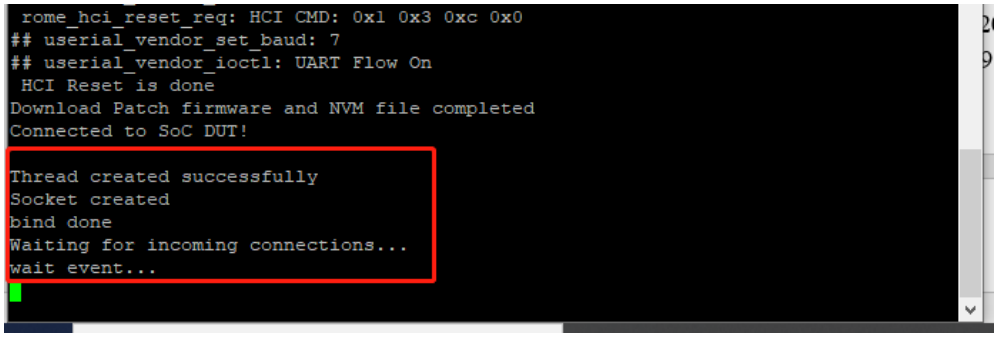
1. The use of QRCT tool requires Qualcomm license. Please install and use the QRCT tool with the assistance of Quectel Technical Support (support@quectel.com).
2. The test device mentioned in this document takes LitePoint Solution as an example.

3.2.1. Configuring Factory Test Mode with QRCT

Set the module to factory test mode. Here is an example:

```
#root
# Btdiag UDT=yes PORT=2390 IOType=SERIAL QDARTIOType=ethernet BT-DEVICE=/dev/ttymxc1
BT-BAUDRATE=115200 IPADDRESS=10.88.152.49
```

If the following logs are printed, it indicates that the module is configured completely:



3.2.2. Running and Configuring Tool

1. Run QRCT.
   Click "**Tool**" → "**User Defined Transport**" → "**Yes**" and confirm the prompt message. If this step is performed when testing WLAN, you can skip this operation.

2. Continue to click "**Device Connect**" → "*** Add New UDT**".

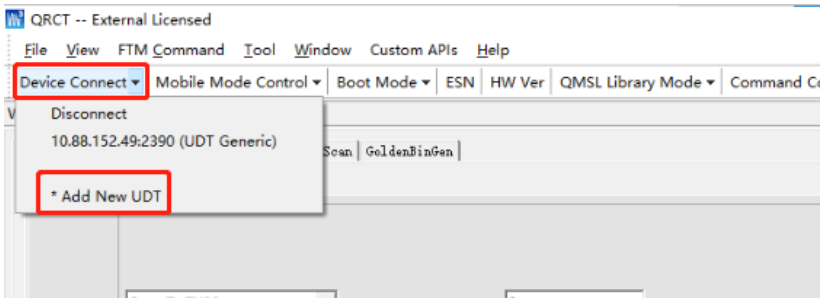


Figure 21: Add New UDT (BT)

In the popup box, select "Bluetooth" from "Technology", input C:\Program Files (x86)\Qualcomm\QDART\bin\QMSL_BT_Transport.dll (the directory is created on the host automatically after installing QRCT. Please contact Quectel Technical Support if the directory does not exist) to "User Defined Transport DLL Path", input the IP address of eth0 to "IP Address", and input the port number to "TCP Port". "ResourceID" is defaulted to COM1, and there is no need to change it. Then click "**Save**". Here is an example:

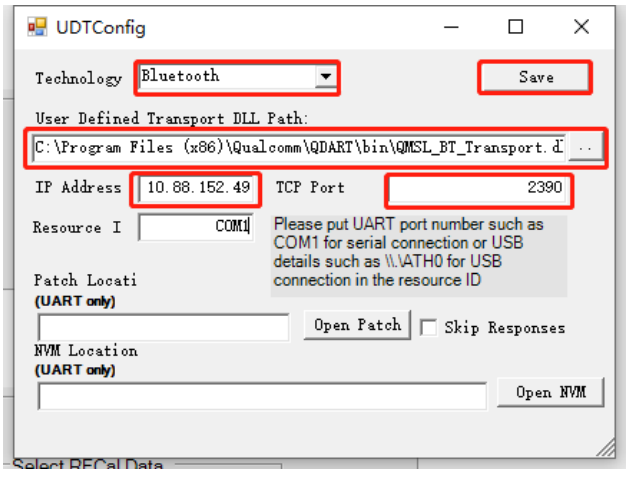


Figure 22: Configure UDT (BT)

3. After saving the configurations, click "**Device Connect**" → "**connectiondetails=COM1…….**".

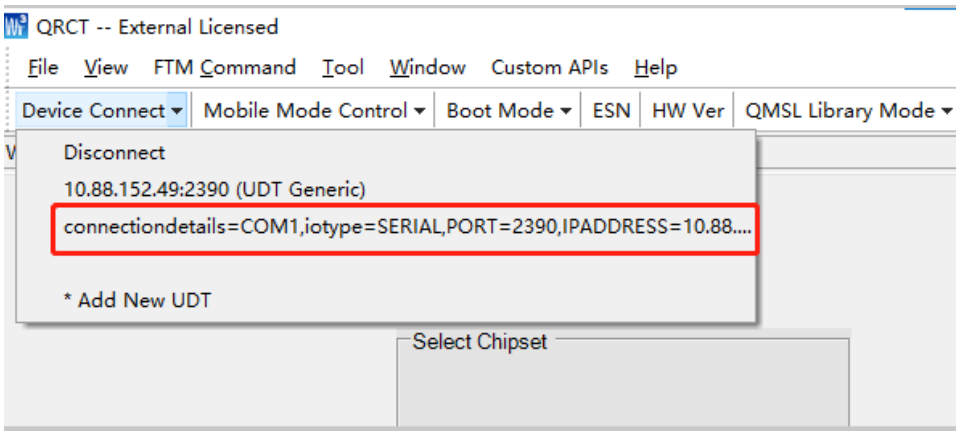


Figure 23: Disconnect (Bluetooth)

NOTE

This step must be performed, otherwise it may cause DUT loading failure. This step is also required when reopening the tool.

3.2.3. Testing Bluetooth

1. Click "**FTM Command**" → "**BT**" → "**HCI Commands**".

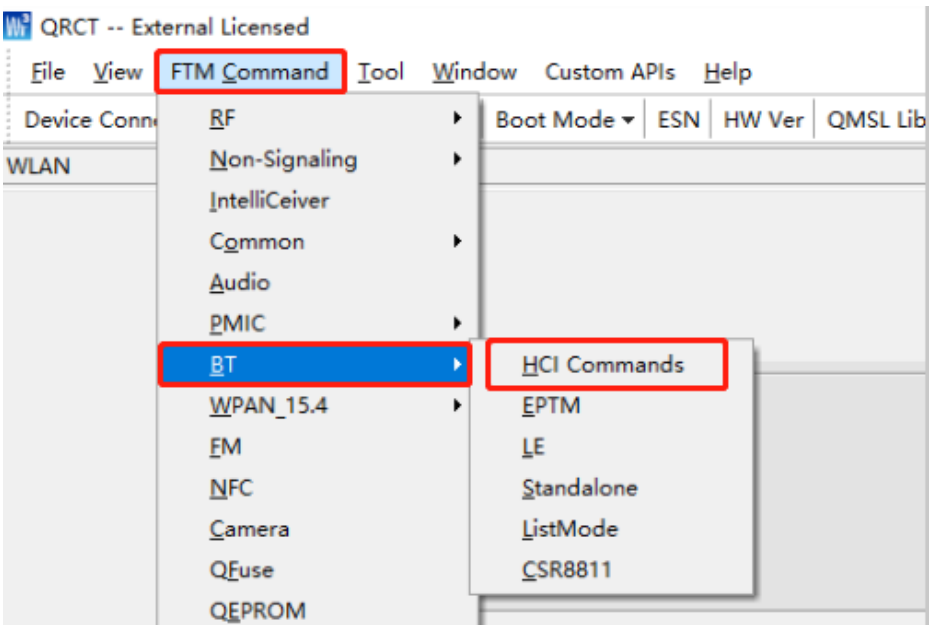


Figure 24: Select BT

2. Click “**View**” → “**QRCT Debug Message**”.

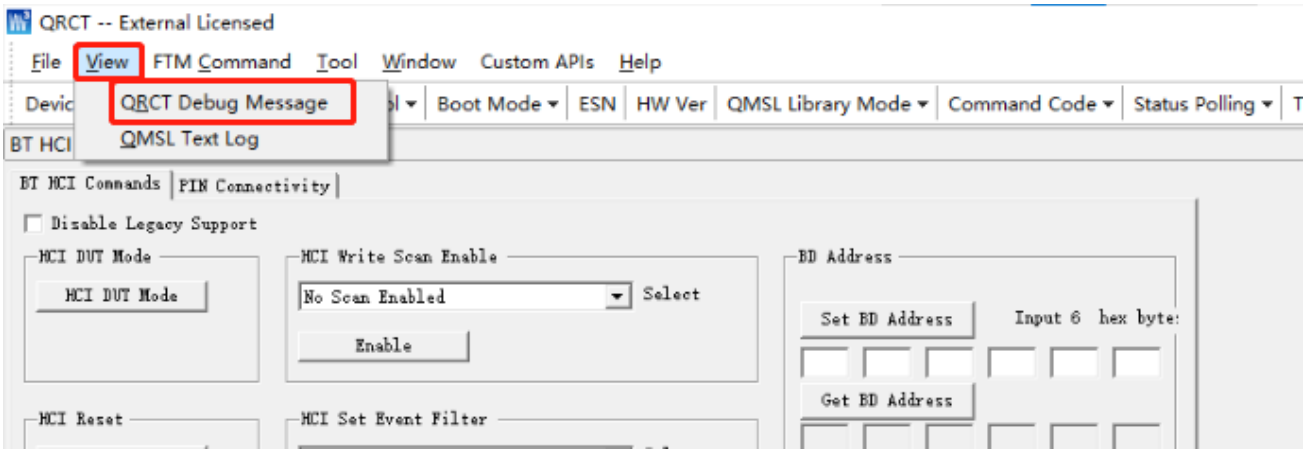


Figure 25: View Debug Message

Check “Disable Legacy Support”, click “**HCI Reset**” to view whether the debug message window has failure message, then click “**HCI DUT Mode**”.

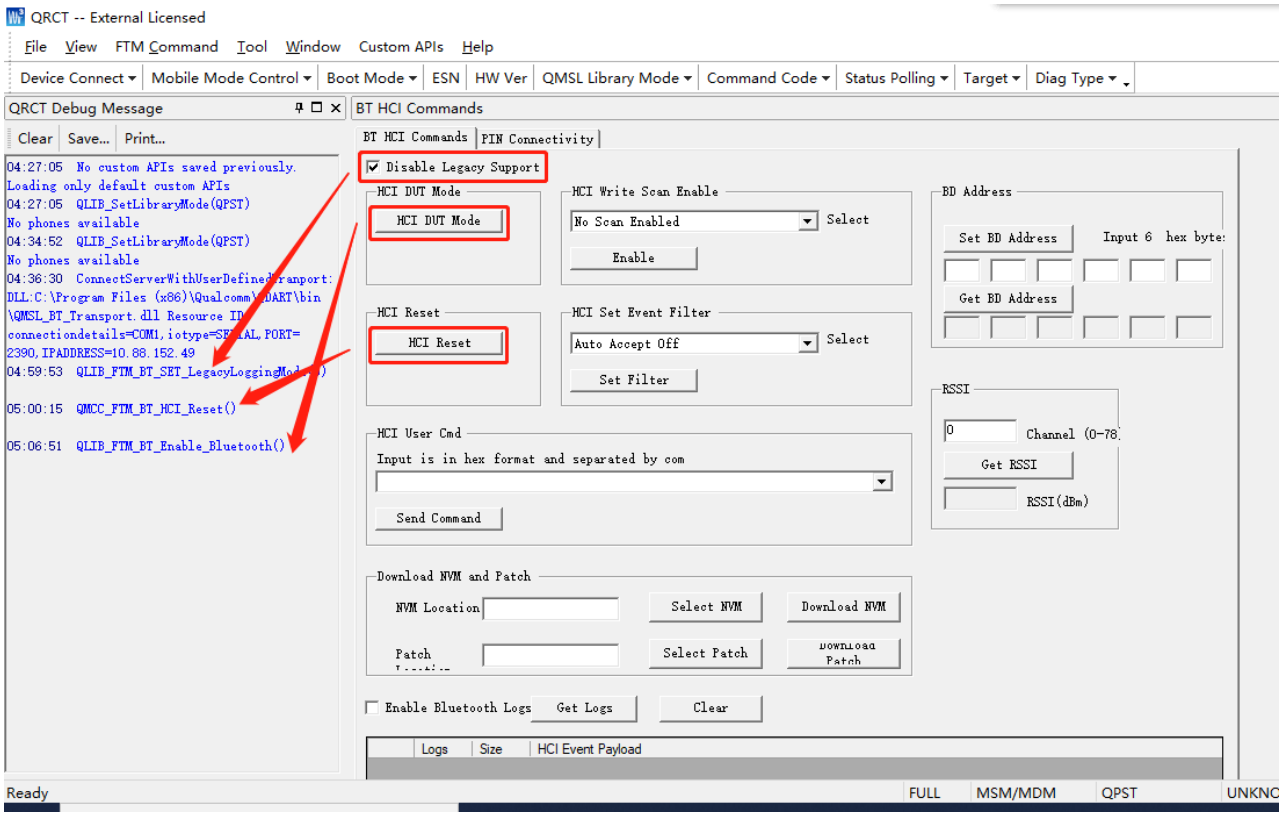


Figure 26: View Failure Message

3. Click "**FTM Command**" → "**BT**" → "**LE**".

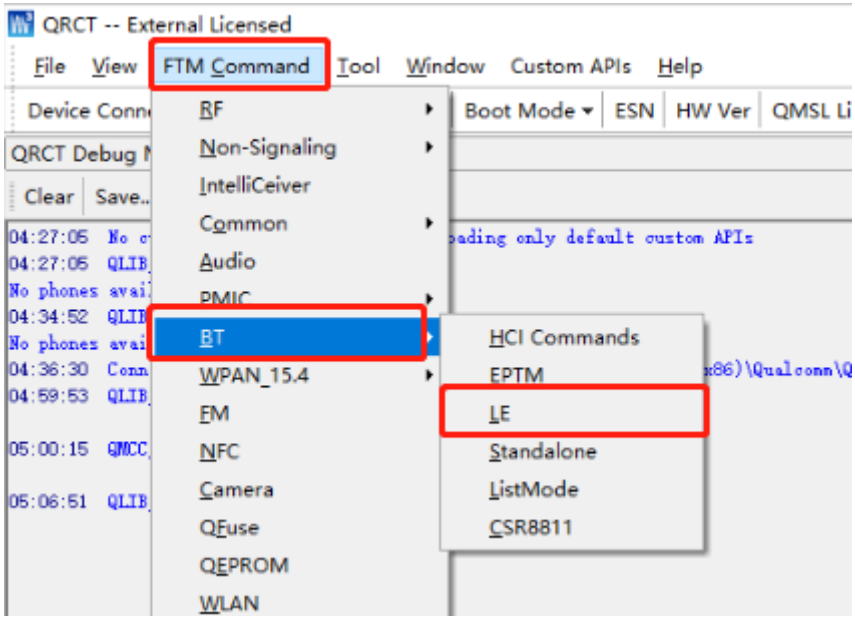


Figure 27: Select LE Mode

4. In "Transmitter Test" column, set "Test Frequency", "Test Payload Length" and "Payload Type" according to the actual practice. Then click "**Transmitter Test**". The test device receives and displays BT signal strength. Here is an example:

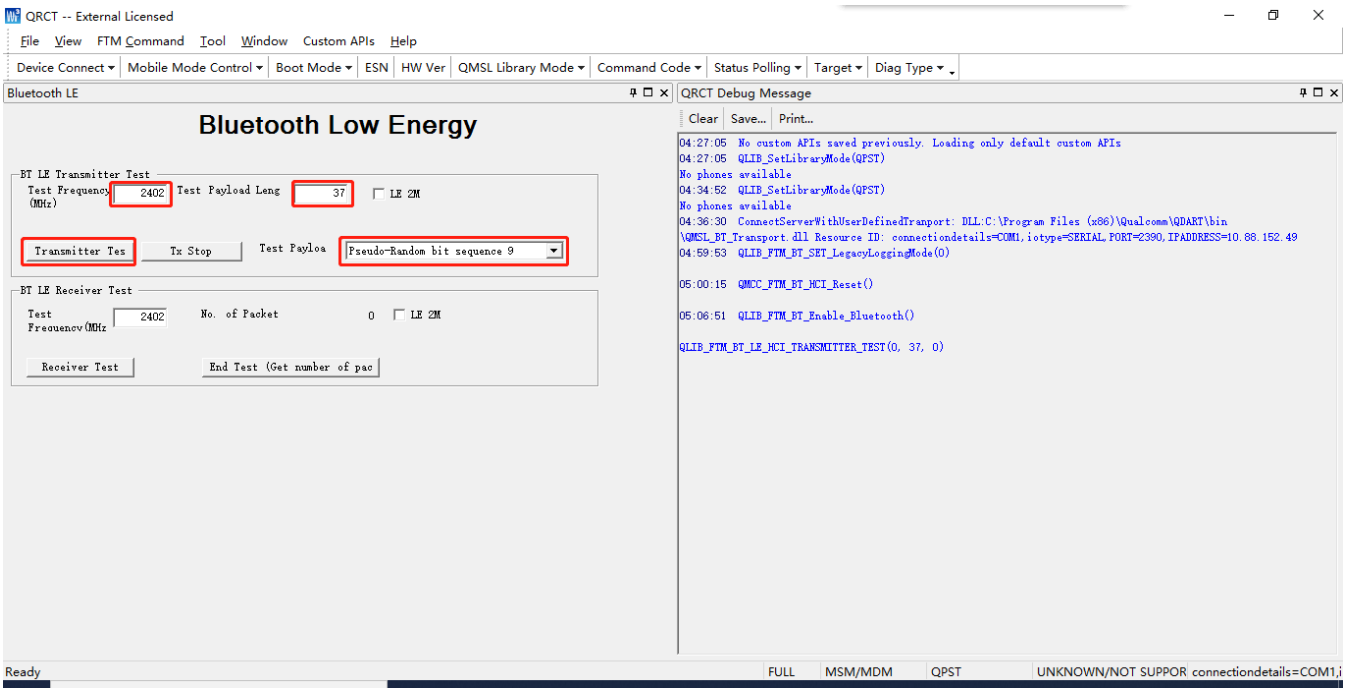


Figure 28: Configure BLE

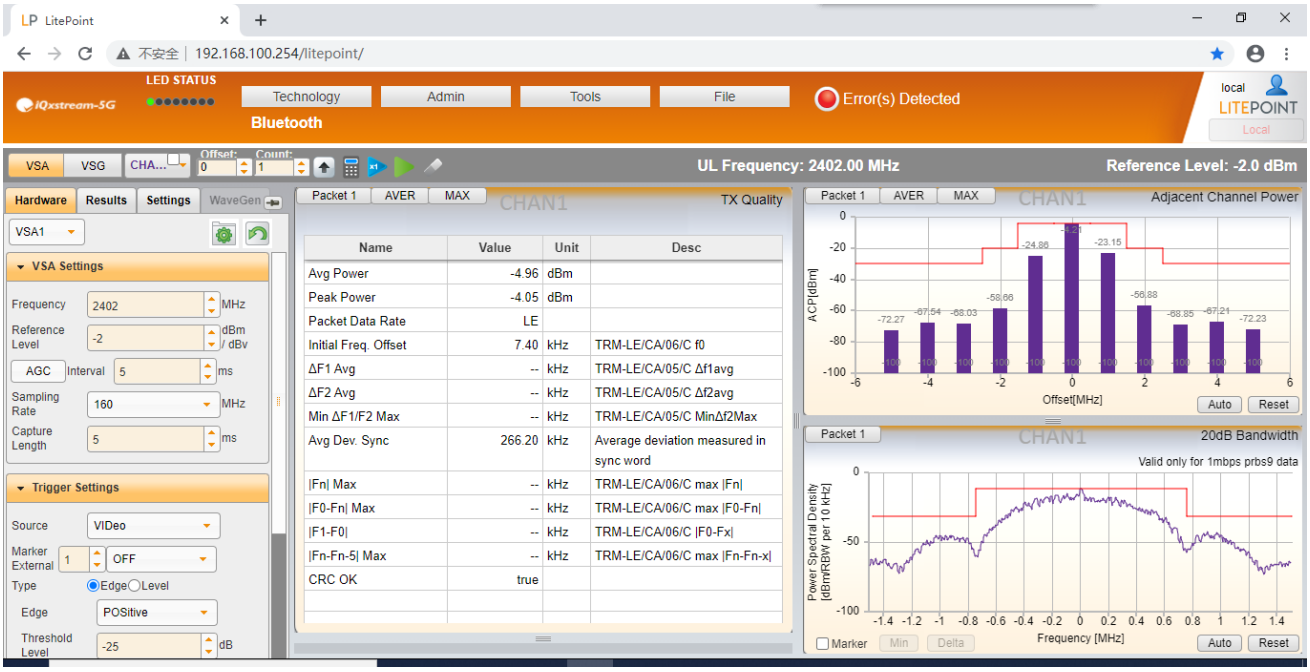


Figure 29: Bluetooth Signal Strength

5. Select “Test Frequency” under “Receiver Test” column in the QRCT tool, then click “**Receiver Test**”. At the same time, the test device begins to send signal. Here is an example:

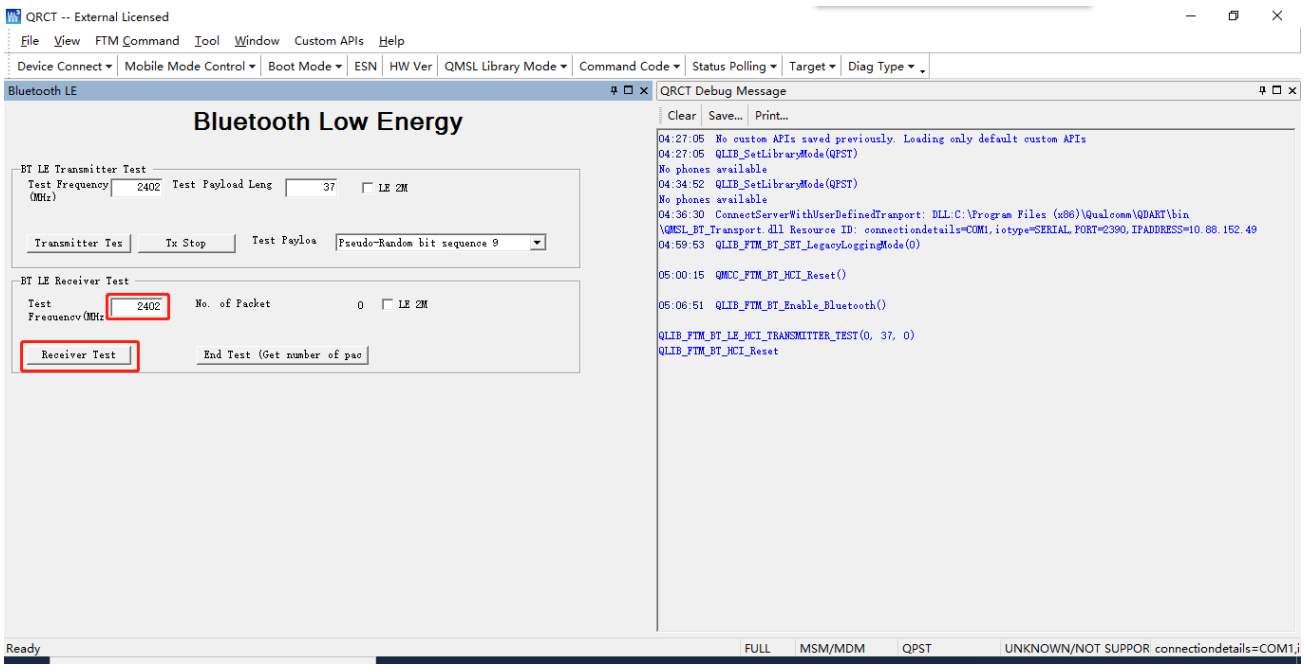


Figure 30: Configure Receiving Test

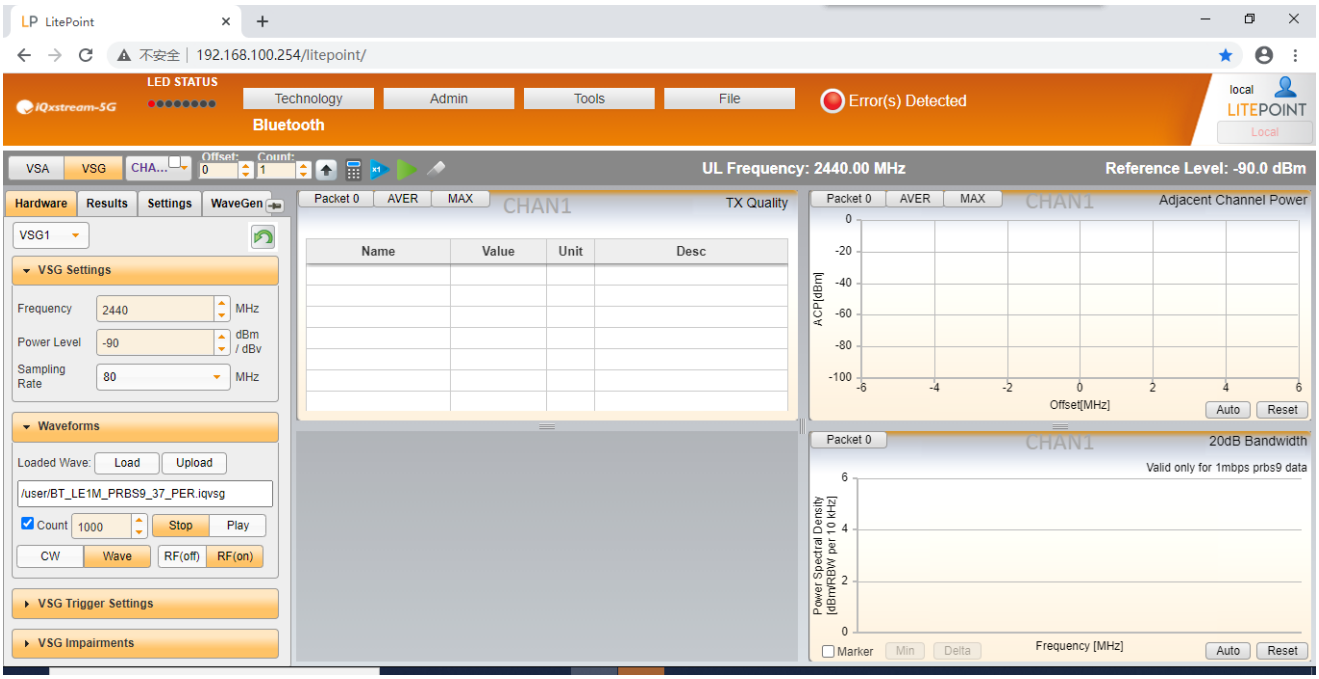


Figure 31: Send Signal From the Test Device

Click "**End Test**" in the QRCT and view the received data.

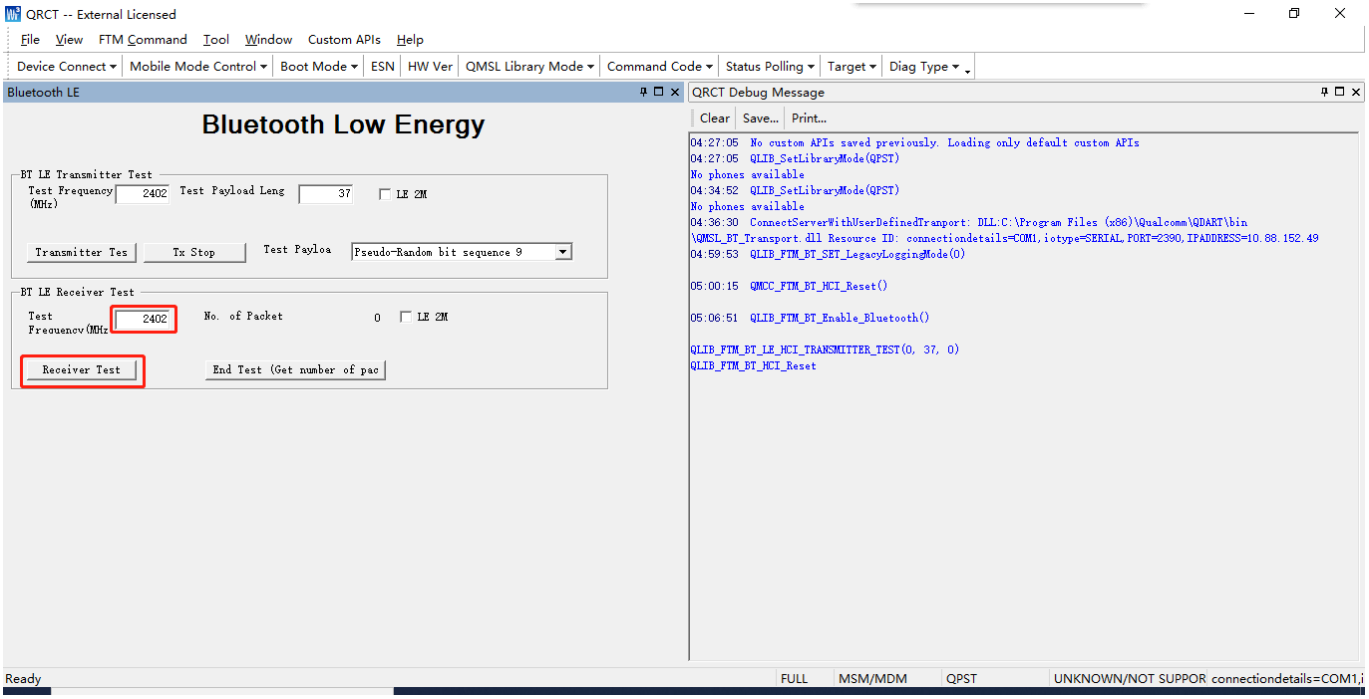


Figure 32: View Received Data

NOTE

If the QRCT is closed, the console prints “Client disconnected”. If it is necessary to test BT or WiFi again, reboot DUT and then execute **Btdiag UDT=yes PORT=2390 IOType=SERIAL QDARTIOType=ethernet BT-DEVICE=/dev/ttymxc1 BT-BAUDRATE=115200 IPADDRESS=10.88.152.49**. If the DUT is not rebooted, it returns an error after executing the above commands.

4 Appendix A References

Table 3: Terms and Abbreviations

Abbreviation	Description
AP	Access Point
BSS	Base Station System
BT	BlueTooth
CCK	Complementary Code Keying
CPU	Central Processing Unit
DHCP	Dynamic Host Configuration Protocol
DUT	Device Under Test
FTM	Factory Test Mode
GPIO	General-Purpose Input/Output
LE	Low Energy
MAC	Medium Access Control
MCS	Modulation and Coding Scheme
NSS	Number of Spatial Streams
OFDM	Orthogonal Frequency-Division Multiplexing
PSK	Pre-Shared Key
RF	Radio Frequency
RX	Receive
SDIO	Secure Digital Input/Output
SDK	Software Development Kit

SSID	Service Set Identifier
STA	Station
TCP	Transmission Control Protocol
TPC	Transmit Power Control
TX	Transmit
UDT	UDP-based Data Transfer Protocol
USB	Universal Serial Bus
Wi-Fi	Wireless Fidelity
WLAN	Wireless Local Area Network
WPA	Wi-Fi Protected Access