AW-CB511NF

IEEE 802.11 a/b/g/n/ac WLAN 2T2R with Bluetooth 5.0 Combo Module (M.2 2230)

User Guide

Rev. 0.1

(For Standard)

Revision History

Document NO:

Version	Revision Date	DCN NO.	Description	Initials	Approved
0.1	2021/11/15		● Initial Version	Licheng Wang	NC Chen

Table of Contents

1 Overview

1.1 Device supported

This document supports AW-CB511NF and AW-CB511NF-BPF.

2. System Setup

2.1. Hardware Requirements

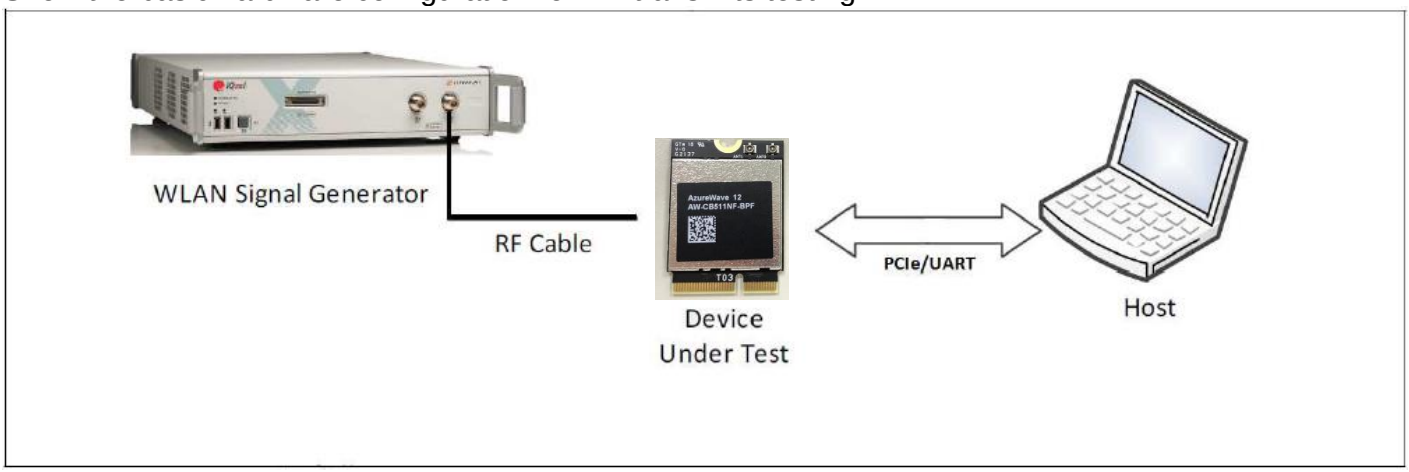
M.2 2230 Standard key E receptacle.
Host system need running the Linux operating system.
Vector Signal Analyzer/WLAN analyzer for transmit measurements.
WLAN signal generator for receiver measurements.
RF isolation chamber for receive measurements.
RF attenuators
RF cable

2.2. Software Requirements

Linux device driver package
(AW-CB511NF and AW-CB511NF-BPF DUT driver must be installed in the host system to run WL commands.)

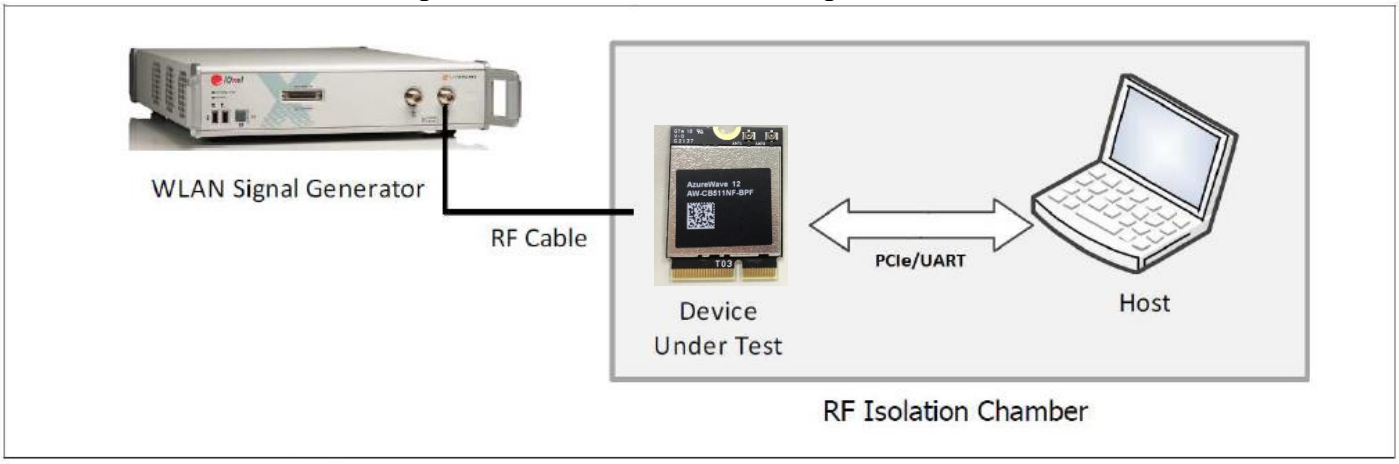
2.3. RF Transmit Test Setup

Show the basic hardware configuration for RF transmits testing.



2.4. RF Receive Test Setup

Show the basic hardware configuration for RF Receive testing.\



3. WLAN Basic Test

3.1.Throughput Test

CONNECTING TO WIRELESS NETWORKS

The examples in the following sections illustrate how to connect to both infrastructure and ad hoc networks, including infrastructure networks that use no security, WEP security, and WPA/PSK and WPS2/PSK security.

SCANNING FOR WIRELESS NETWORKS

To force the dongle to scan

• Run **wl up**.

• Run **wl scan**.

To force the dongle to return the results of the scan

• Run **wl scanresults**.

Example results returned when an AP is found:

• SSID: “Eval4325”

• Mode: Managed: RSSI: -48 dBm noise: -105 dBm Channel: 1

• BSSID: 00:10:18:90:2E:C1 Capability: ESS ShortSlot

• Supported Rates: [1(b) 2(b) 5.5(b) 11(b) 18 24 36 54 6 9 12 48]

Example results returned when an ad hoc network is found:

• SSID: “ADHOC#1”

• Mode: Ad Hoc RSSI: -41 dBm noise: -105 dBm Channel: 1

• BSSID: B2:51:28:6B:3C:A1 Capability: IBSS

• Supported Rates: [1(b) 2(b) 5.5(b) 11(b)]

CONNECTING TO AN INFRASTRUCTURE NETWORK WITH NO SECURITY (AP CONNECTION)

To connect to the network through an AP with SSID = Eval4325

Run **wl join Eval4325**.

CONNECTING TO AN INFRASTRUCTURE NETWORK WITH WEP SECURITY

To connect to the network that uses 12345 as the network key

• Run **wl join Eval4325 key 12345**.

CONNECTING TO AN INFRASTRUCTURE NETWORK WITH WPA-PSK/WPA2-PSK SECURITY

To specify TKIP or AES as the data encryption method

• Run **wl wsec 3/7**.

To enable the supplicant

• Run **wl sup_wpa 1**.

To specify the PSK passphrase (network key) to use

• Run **wl set_psk $passphrase**.

To connect to a network that uses WPA-PSK security

• Run **wl join Eval4325 imode bss amode wpapsk**.

To connect to a network that uses WPA2-PSK security

• Run **wl join Eval4325 imode bss amode wpa2psk**.

CONNECTING TO AN AD HOC NETWORK USING CHANNEL 1

To set the channel to channel 1

• Run **wl channel 1**.

To connect to the ad hoc network with SSID = 4325-ADHOC

• Run **wl join 4325-ADHOC imode ibss**.

MANAGING POWER CONSUMPTION

To disable Power Save (PS) mode (default)

• Run **wl PM 0**.

To enable legacy IEEE 802.11 Power Save (PS) mode

• Run **wl PM 1**.

To enable Fast IEEE 802.11 Power Save mode

• Run **wl PM 2**.

Note:

• The STA automatically transitions to Legacy PS mode when no data is being sent or received.

• The STA automatically disables PS mode when data is being sent or received.

MEASURING WLAN THROUGHPUT

The throughput measurement shows the performance of the TCP/IP layer over the wireless link. To achieve the best results, run the measurement test in a clean environment with as little interference as possible. The test can be run with the adapter connected to either an Infrastructure network (see Fig. 2.2) or an ad hoc network (see Fig. 2.3). An AP that is known to be in good working order should be used for the infrastructure mode test.

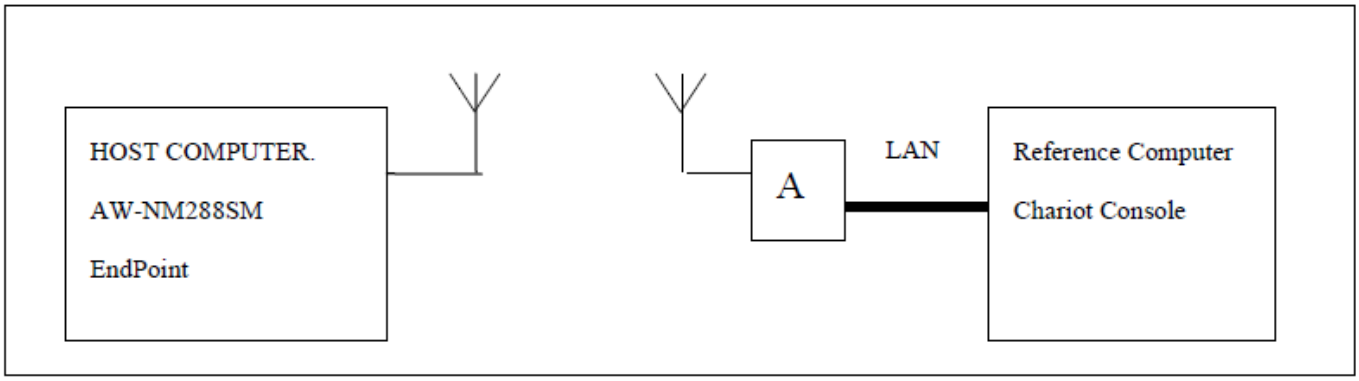


FIG. 2.2

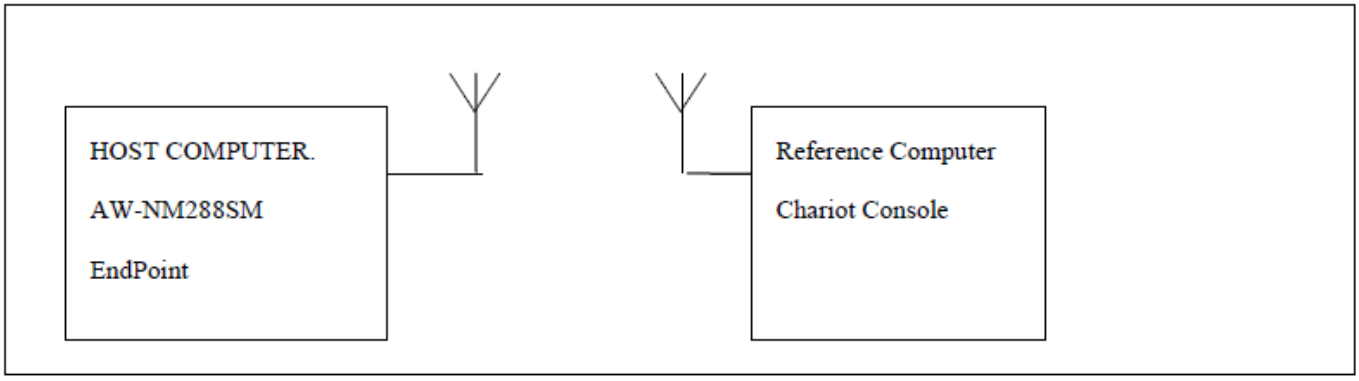


FIG. 2.3

3.2. RF Tx/Rx WL Command Test

3.2.1 Single carrier

./wl down
./wl mpc 0
./wl band b
./wl up
./wl channel 1
./wl phy_txpwrctrl 0
./wl phy_txpwrindex 120
./wl phy_txlo_tone 1
./wl fqacurcy 1

3.2.2 802.11b/g Tx in 2.4G

./wlfmac down
./wlfmac rsdb_mode 0
./wlfmac mpc 0
./wlfmac phy_watchdog 0
./wlfmac country ALL
./wlfmac band b
./wlfmac channel 7
./wlfmac mimo_txbw -1
./wlfmac txchain [X] → Core 0: X=1; Core 1: X=2
./wlfmac up
./wlfmac phy_forcecal 1
./wlfmac scansuppress 1
./wlfmac 2g_rate -r 11 -b 20 → -r=11b/g rate, -b=20MHz BW

for close loop power control
./wlfmac phy_txpwrctrl 1 → close loop power control
./wlfmac txpwr1 -o -d [X] → X=user defined TX power

for open loop power control
./wlfmac phy_txpwrctrl 0 → open loop power control
./wlfmac phy_txpwrindex [Y] → Tx power index in the range of 0-127

./wlfmac pkteng_start 00:11:22:33:44:55 tx 100 1024 0
./wlfmac pkteng_stop tx → stop Tx transmission

After the temperature is stable, please resend the command phy_forcecal 1.
./wlfac phy_forcecal 1

3.2.3 802.11a Tx in 5G

```
./wlfmac down
./wlfmac rsdb_mode 0
./wlfmac mpc 0
./wlfmac phy_watchdog 0
./wlfmac country ALL
./wlfmac band a
./wlfmac channel 36
./wlfmac mimo_txbw -1
./wlfmac txchain [X] → Core 0: X=1; Core 1: X=2
./wlfmac up
./wlfmac phy_forcecal 1
./wlfmac scansuppress 1
./wlfmac 5g_rate -r 54 -b 20 → -r=11a rate, -b=20MHz BW

./wlfmac phy_txpwrctrl 1
./wlfmac txpwr1 -o -d [X] → X=user defined TX power
./wlfmac pkteng_start 00:11:22:33:44:55 tx 100 1024 0
./wlfmac pkteng_stop tx → stop Tx transmission

# After the temperature is stable, please resend the command phy_forcecal 1.
./wlfac phy_forcecal 1
```

3.2.4 802.11n HT20 Tx in 2.4G/5G

./wlfmac down
./wlfmac rsdb_mode 0
./wlfmac mpc 0
./wlfmac phy_watchdog 0
./wlfmac country ALL
./wlfmac band [b/a] → b=2.4GHz, a=5GHz
./wlfmac chanspec 7/20
./wlfmac mimo_txbw -1
./wlfmac txchain [X] → Core 0: X=1; Core 1: X=2; MIMO: X=3
./wlfmac up
./wlfmac phy_forcecal 1
./wlfmac scansuppress 1

for SISO operation
./wlfmac [2g/5g]_rate -h [0-7] -b 20 → 2g_rate or 5g_rate, -h=MCS0-MCS7

for MIMO operation
./wlfmac [2g/5g]_rate -h [8-15] -b 20 → 2g_rate or 5g_rate, -h=MCS8-MCS15

./wlfmac phy_txpwrctrl 1
./wlfmac txpwr1 -o -d [X] → X=user defined TX power
./wlfmac pkteng_start 00:11:22:33:44:55 tx 100 1024 0
./wlfmac pkteng_stop tx → stop Tx transmission

After the temperature is stable, please resend the command phy_forcecal 1.
./wlfac phy_forcecal 1

3.2.5 802.11n HT40 Tx in 5G

./wlfmac down
./wlfmac rsdb_mode 0
./wlfmac mpc 0
./wlfmac phy_watchdog 0
./wlfmac country ALL
./wlfmac band a
./wlfmac chanspec 36l
./wlfmac mimo_txbw 4
./wlfmac txchain [X] → Core 0: X=1; Core 1: X=2; MIMO: X=3
./wlfmac up
./wlfmac phy_forcecal 1
./wlfmac scansuppress 1

for SISO operation
./wlfmac 5g_rate -h [0-7] -b 40 → -h=MCS0-MCS7

for MIMO operation
./wlfmac 5g_rate -h [8-15] -b 40 → -h=MCS8-MCS15

./wlfmac phy_txpwrctrl 1
./wlfmac txpwr1 -o -d [X] → X=user defined TX power
./wlfmac pkteng_start 00:11:22:33:44:55 tx 100 1024 0
./wlfmac pkteng_stop tx → stop Tx transmission

After the temperature is stable, please resend the command phy_forcecal 1.
./wlfac phy_forcecal 1

3.2.6 802.11ac VHT20 Tx in 5G

```
./wlfmac down
./wlfmac rsdb_mode 0
./wlfmac mpc 0
./wlfmac phy_watchdog 0
./wlfmac country ALL
./wlfmac band a
./wlfmac chanspec 36
./wlfmac mimo_txbw -1
./wlfmac txchain [X] → Core 0: X=1; Core 1: X=2; MIMO: X=3
./wlfmac up
./wlfmac phy_forcecal 1
./wlfmac scansuppress 1

# for SISO operation
./wlfmac 5g_rate -v [0-8] -b 20 → -v=MCS0-MCS8

# for MIMO operation
./wlfmac 5g_rate -v 8x2 -b 20

./wlfmac phy_txpwrctrl 1
./wlfmac txpwr1 -o -d [X] → X=user defined TX power
./wlfmac pkteng_start 00:11:22:33:44:55 tx 100 1024 0
./wlfmac pkteng_stop tx → stop Tx transmission

# After the temperature is stable, please resend the command phy_forcecal 1.
./wlfac phy_forcecal 1
```

3.2.7 802.11ac VHT40 Tx in 5G

./wlfmac down
./wlfmac rsdb_mode 0
./wlfmac mpc 0
./wlfmac phy_watchdog 0
./wlfmac country ALL
./wlfmac band a
./wlfmac chanspec 36l
./wlfmac mimo_txbw -1
./wlfmac txchain [X] → Core 0: X=1; Core 1: X=2; MIMO: X=3
./wlfmac up
./wlfmac phy_forcecal 1
./wlfmac scansuppress 1

for SISO operation
./wlfmac 5g_rate -v [0-9] -b 40 → -v=MCS0-MCS9

for MIMO operation
./wlfmac 5g_rate -v 9x2 -b 40

./wlfmac phy_txpwrctrl 1
./wlfmac txpwr1 -o -d [X] → X=user defined TX power
./wlfmac pkteng_start 00:11:22:33:44:55 tx 100 1024 0
./wlfmac pkteng_stop tx → stop Tx transmission

After the temperature is stable, please resend the command phy_forcecal 1.
./wlfac phy_forcecal 1

3.2.8 802.11ac VHT80 Tx in 5G

./wlfmac down
./wlfmac rsdb_mode 0
./wlfmac mpc 0
./wlfmac phy_watchdog 0
./wlfmac country ALL
./wlfmac band a
./wlfmac chanspec 36/80
./wlfmac mimo_txbw -1
./wlfmac txchain [X] → Core 0: X=1; Core 1: X=2; MIMO: X=3
./wlfmac up
./wlfmac phy_forcecal 1
./wlfmac scansuppress 1

for SISO operation
./wlfmac 5g_rate -v [0-9] -b 80 → -v=MCS0-MCS9

for MIMO operation
./wlfmac 5g_rate -v 9x2 -b 80

./wlfmac phy_txpwrctrl 1
./wlfmac txpwr1 -o -d [X] → X=user defined TX power
./wlfmac pkteng_start 00:11:22:33:44:55 tx 100 1024 0
./wlfmac pkteng_stop tx → stop Tx transmission

After the temperature is stable, please resend the command phy_forcecal 1.
./wlfac phy_forcecal 1

3.2.9 802.11b/g Rx in 2.4G

./wlfmac down
./wlfmac rsdb_mode 0
./wlfmac mpc 0
./wlfmac phy_watchdog 0
./wlfmac country ALL
./wlfmac band b
./wlfmac channel 7
./wlfmac mimo_txbw -1
./wlfmac rxchain [X] → SISO Core 0: X=1; SISO Core 1: X=2
./wlfmac up
./wlfmac phy_forcecal 1
./wlfmac scansuppress 1

./wlfmac pkteng_start 00:90:4c:99:00:1d rx
./wlfmac reset_cnts → to reset counters to 0.
./wlfmac counters → read pktengrxducast, take this as counter #1.

Litepoint >> generate a 11b/g waveform that contains X number of packets.

./wlfmac counters → Read pktengrxducast again, take this as counter #2.
PER% = { [X - (counter#2 - counter#1)] / X } * 100%

3.2.10 802.11a Rx in 5G

./wlfmac down
./wlfmac rsdb_mode 0
./wlfmac mpc 0
./wlfmac phy_watchdog 0
./wlfmac country ALL
./wlfmac band [b/a] → b=2.4GHz, a=5GHz
./wlfmac chanspec 7/20
./wlfmac mimo_txbw -1
./wlfmac rxchain [X] → SISO Core 0: X=1; SISO Core 1: X=2
./wlfmac up
./wlfmac phy_forcecal 1
./wlfmac scansuppress 1

./wlfmac pkteng_start 00:90:4c:99:00:1d rx
./wlfmac reset_cnts → to reset counters to 0.
./wlfmac counters → read pktengrxducast, take this as counter #1.

Litepoint >> generate a 11n 20M waveform that contains X number of packets.

./wlfmac counters → Read pktengrxducast again, take this as counter #2.
PER% = { [X - (counter#2 - counter#1)] / X } * 100%

3.2.11 802.11n HT20 Rx in 2.4G/5G

./wlfmac down
./wlfmac rsdb_mode 0
./wlfmac mpc 0
./wlfmac phy_watchdog 0
./wlfmac country ALL
./wlfmac band [b/a] → b=2.4GHz, a=5GHz
./wlfmac chanspec 7/20
./wlfmac mimo_txbw -1
./wlfmac rxchain [X] → SISO Core 0: X=1; SISO Core 1: X=2
./wlfmac up
./wlfmac phy_forcecal 1
./wlfmac scansuppress 1

./wlfmac pkteng_start 00:90:4c:99:00:1d rx
./wlfmac reset_cnts → to reset counters to 0.
./wlfmac counters → read pktengrxducast, take this as counter #1.

Litepoint >> generate a 11n 20M waveform that contains X number of packets.

./wlfmac counters → Read pktengrxducast again, take this as counter #2.
PER% = { [X - (counter#2 - counter#1)] / X } * 100%

3.2.12 802.11n HT40 Rx in 5G

./wlfmac down
./wlfmac rsdb_mode 0
./wlfmac mpc 0
./wlfmac phy_watchdog 0
./wlfmac country ALL
./wlfmac band a
./wlfmac chanspec 36l
./wlfmac mimo_txbw 4
./wlfmac rxchain [X] → SISO Core 0: X=1; SISO Core 1: X=2
./wlfmac up
./wlfmac phy_forcecal 1
./wlfmac scansuppress 1

./wlfmac pkteng_start 00:90:4c:99:00:1d rx
./wlfmac reset_cnts → to reset counters to 0.
./wlfmac counters → read pktengrxducast, take this as counter #1.

Litepoint >> generate a 11n 40M waveform that contains X number of packets.

./wlfmac counters → Read pktengrxducast again, take this as counter #2.
PER% = { [X - (counter#2 - counter#1)] / X } * 100%

3.2.13 802.11ac VHT20 Rx in 5G

./wlfmac down
./wlfmac rsdb_mode 0
./wlfmac mpc 0
./wlfmac phy_watchdog 0
./wlfmac country ALL
./wlfmac band a
./wlfmac chanspec 36/20
./wlfmac mimo_txbw -1
./wlfmac rxchain [X] → SISO Core 0: X=1; SISO Core 1: X=2
./wlfmac up
./wlfmac phy_forcecal 1
./wlfmac scansuppress 1

./wlfmac pkteng_start 00:90:4c:99:00:1d rx
./wlfmac reset_cnts → to reset counters to 0.
./wlfmac counters → read pktengrxducast, take this as counter #1.

Litepoint >> generate a 11ac 20M waveform that contains X number of packets.

./wlfmac counters → Read pktengrxducast again, take this as counter #2.
PER% = { [X - (counter#2 - counter#1)] / X } * 100%

3.2.14 802.11ac VHT40 Rx in 5G

./wlfmac down
./wlfmac rsdb_mode 0
./wlfmac mpc 0
./wlfmac phy_watchdog 0
./wlfmac country ALL
./wlfmac band a
./wlfmac chanspec 36l
./wlfmac mimo_txbw -1
./wlfmac rxchain [X] → SISO Core 0: X=1; SISO Core 1: X=2
./wlfmac up
./wlfmac phy_forcecal 1
./wlfmac scansuppress 1

./wlfmac pkteng_start 00:90:4c:99:00:1d rx
./wlfmac reset_cnts → to reset counters to 0.
./wlfmac counters → read pktengrxducast, take this as counter #1.

Litepoint >> generate a 11ac 40M waveform that contains X number of packets.

./wlfmac counters → Read pktengrxducast again, take this as counter #2.
PER% = { [X - (counter#2 - counter#1)] / X } * 100%

3.2.15 802.11ac VHT80 Rx in 5G

./wlfmac down
./wlfmac rsdb_mode 0
./wlfmac mpc 0
./wlfmac phy_watchdog 0
./wlfmac country ALL
./wlfmac band a
./wlfmac chanspec 36/80
./wlfmac mimo_txbw -1
./wlfmac rxchain [X] → SISO Core 0: X=1; SISO Core 1: X=2
./wlfmac up
./wlfmac phy_forcecal 1
./wlfmac scansuppress 1

./wlfmac pkteng_start 00:90:4c:99:00:1d rx
./wlfmac reset_cnts → to reset counters to 0.
./wlfmac counters → read pktengrxducast, take this as counter #1.

Litepoint >> generate a 11ac 80M waveform that contains X number of packets.

./wlfmac counters → Read pktengrxducast again, take this as counter #2.
PER% = { [X - (counter#2 - counter#1)] / X } * 100%

4.BT Basic Test

Software Requirements

1.bt_tool

2.hcd file.

3.brcm_patchram_plus_linux_32

Driver Installation and enter test mode.

1.Enter **/path/brcm_patchram_plus_linux_32 --baudrate 115200 --use_baudrate_for_download – patchram [hcd file].hcd –no2bytes /dev/ttyUSB0** and wait download finish.

2.Enter **/path/bt_tool /dev/ttyUSB0 115200 hcimd 03 03** and wait successful!

3.Enter **/path/bt_tool /dev/ttyUSB0 115200 hcimd 03 1a 03** and wait successful!

4.Enter **/path/bt_tool /dev/ttyUSB0 115200 hcimd 03 05 02 00 03** and wait successful!

5.Enter **/path/bt_tool /dev/ttyUSB0 115200 hcimd 06 03** and wait successful!