
# RF Performance Test

# User Manual

*Version：1.0*



*www.bouffalolab.com*

# Contents

# List of Figures

# 1 Version record

Table 1.1: Version record

| Version | Update content |
|---|---|
| V1.0 | Initial release |

# 2 Overview

The RF performance test tool (RF MFG) is a tool provided by Bouffalo Lab for RF evaluation and testing. It includes two parts: a test tool and a test image (MFG Firmware). The interface of the test tool is shown in the figure below.

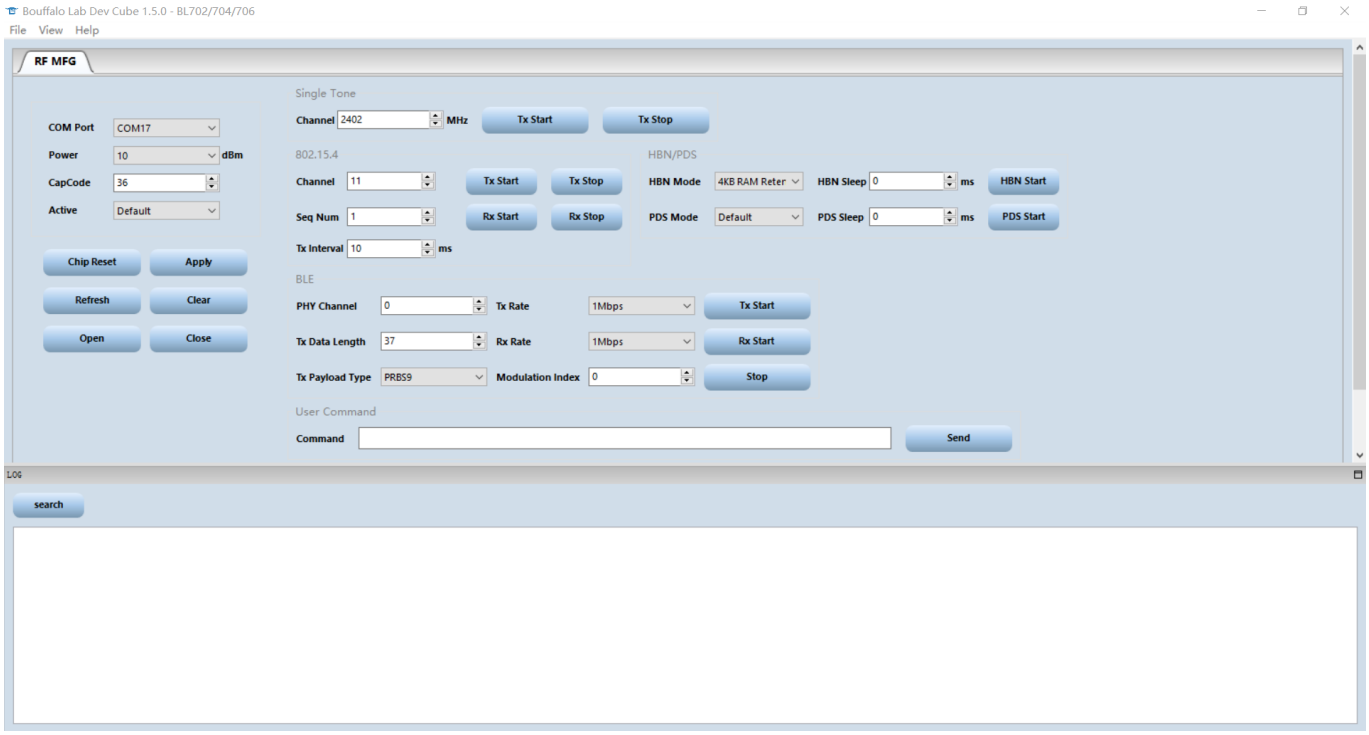


Fig. 2.1: Interface of test tool

The functions that can be performed by the RF performance test tool (RF MFG) include:

- Single Tone test
- 802.15.4 packet transmission
- 802.15.4 packet reception
- BLE packet transmission

- BLE packet reception
- PDS (Power Down Sleep) test
- HBN (Hibernate) test

# 3 Programming tool kit

If user does not have the programming tool kit (which is also called Dev Cube), please download it from Bouffalo Lab Dev Cube. It contains the RF test firmware, firmware programming tool, RF test tool, etc. The content of the tool kit is shown in the figure below. It is a tool set used by customers to develop various types of Bouffalo Lab chips.


Fig. 3.1: Programming tool kit

# 4 Program test firmware

## 4.1 Program test firmware to IOT module (with Flash)

In the programming tool kit (Bouffalo Lab Dev Cube), the RF test firmware of each chip is included. The following takes BL704 IOT module evaluation kit as an example to introduce the programming process.

The BL704 IOT module evaluation kit BL704_IoT_DevKit is shown in the figure below.

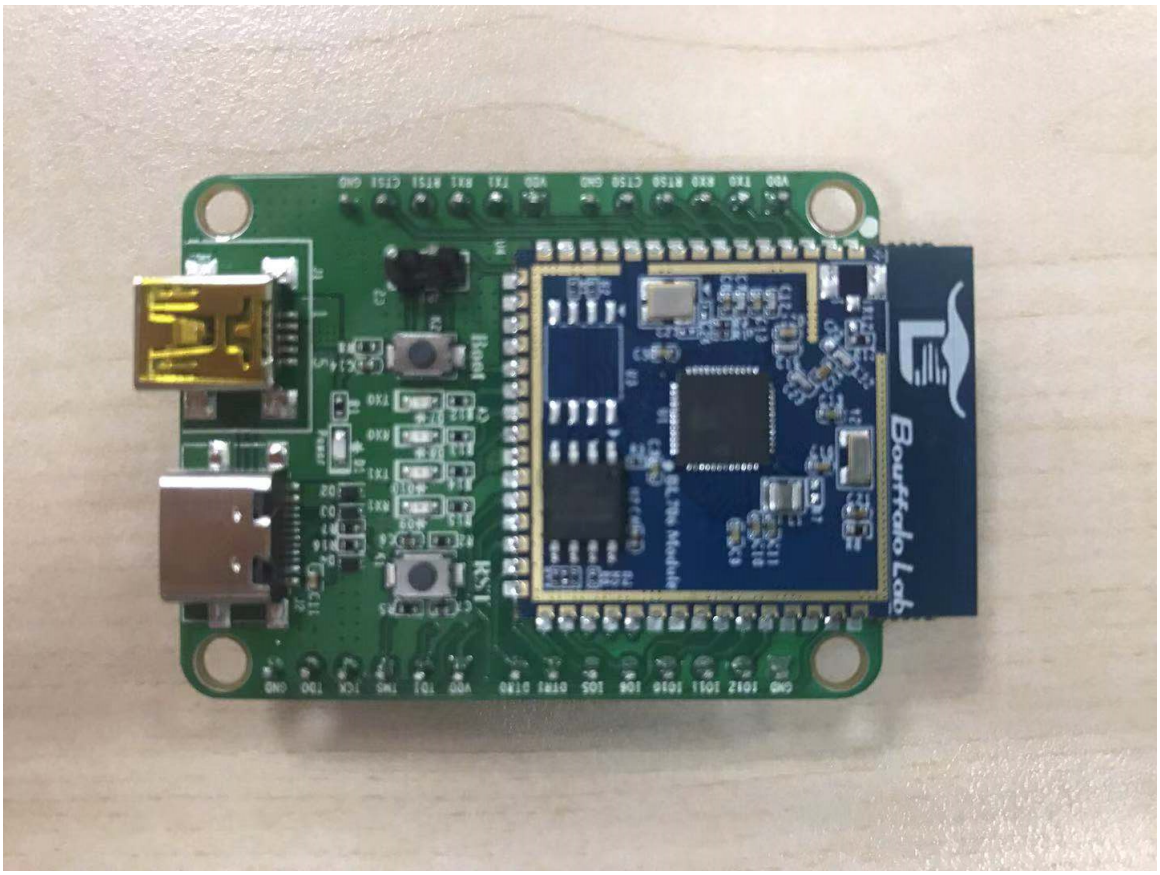


Fig. 4.1: IOT Module Evaluation Kit

The evaluation kit consists of an IOT module and a motherboard. The motherboard uses the USB interface for power supply.

In addition, a USB serial port adapter board is required. The figure below shows one of the adapter boards.

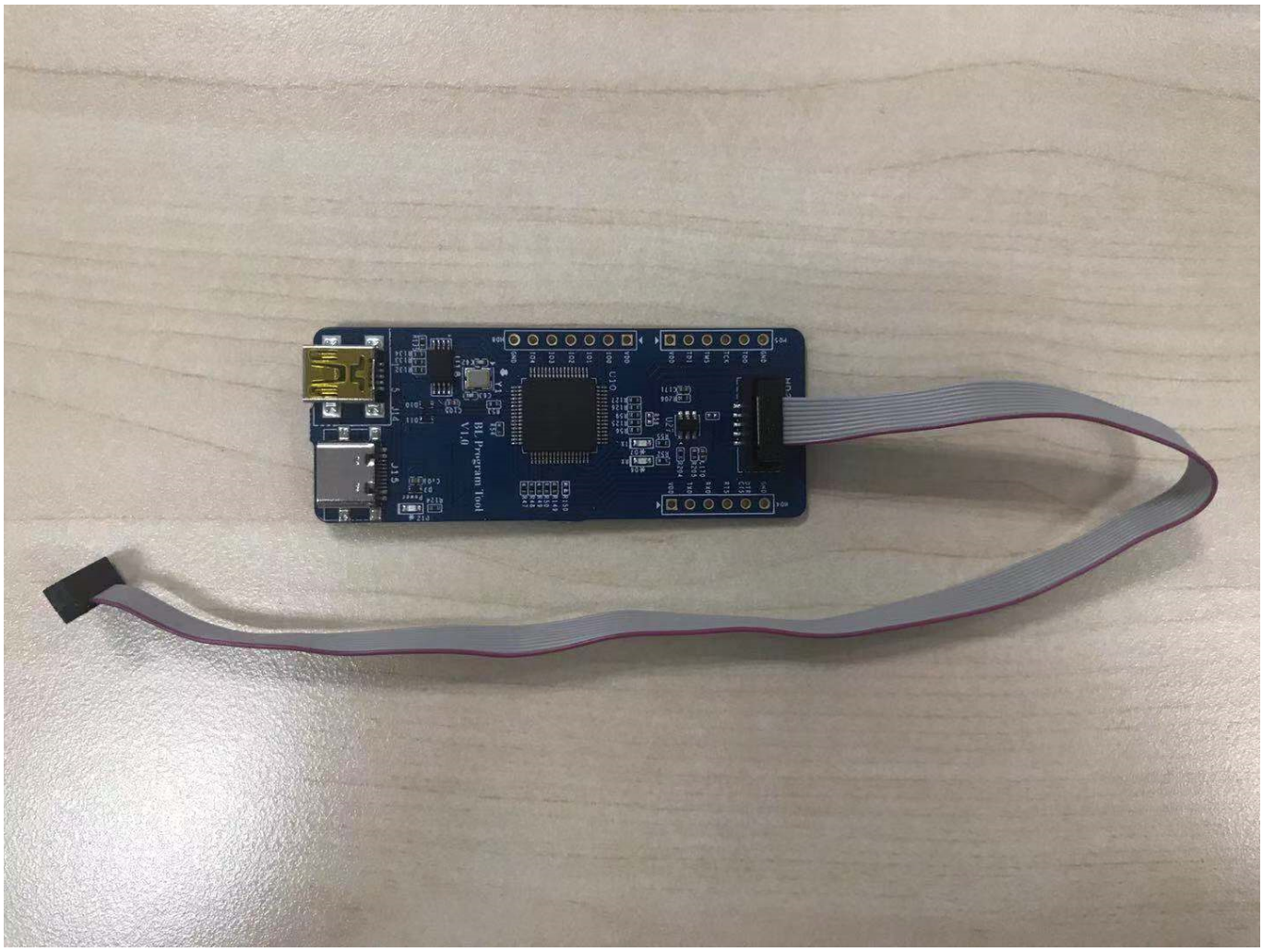

Fig. 4.2: Adapter board

The adapter board uses the USB interface for power supply. After connecting to the evaluation kit, it is used for UART download, UART print and online debug.

When the adapter board is connected to the PC, two serial ports will appear in the PC’s device manager. The two COM numbers are adjacent, and the smaller one is connected to the UART of the chip. If the serial port driver is not automatically installed, please go to https://www.ftdichip.com/Drivers/VCP.htm to download the driver.

After the adapter board is connected to the evaluation kit, run BLDevCube.exe, select BL702/704/706 in Chip Type, then the following programming interface will be shown.

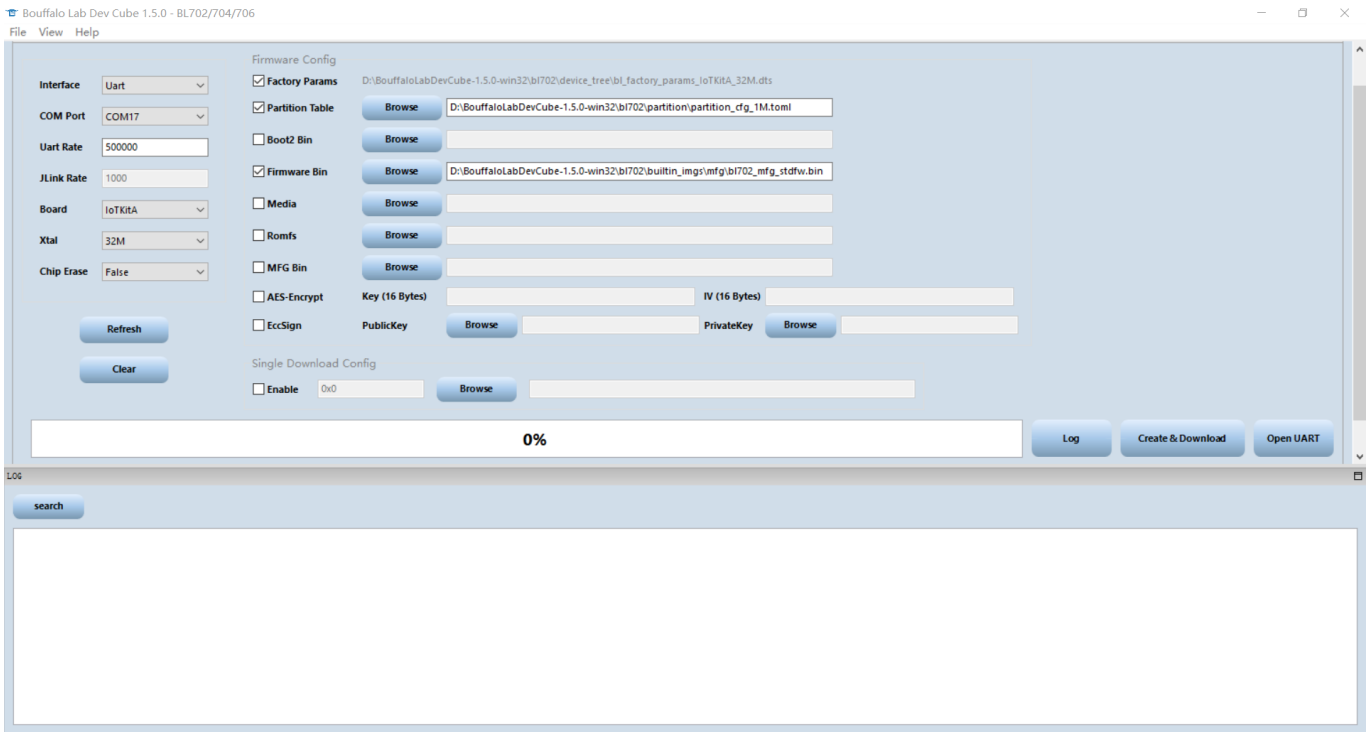


Fig. 4.3: Programming interface

In the communication interface settings on the left:

- Interface: Used to select the communication interface for programming, here select Uart for programming
- COM Port: When selected UART for programming, select the COM port that connects to the chip, and you can click the Refresh button to refresh the COM number
- Uart Rate: When selected UART for programming, fill in the baud rate, 500000 is recommended
- Board: Select the board type you are using, here select IoTKitA
- Xtal: Used to select the crystal type on the board, here select 32M
- Chip Erase: The default setting is False, which means that full chip Flash will not be erased when programming

Use the default configuration for other items.

In the programming settings on the right, select:

- Partition Table: Use the partition table in the partition directory of the corresponding chip type in the programming tool directory. In this example bl702/partition/partition_cfg_1M.toml
- Firmware Bin: Select the corresponding Flash version firmware in the mfg folder under the builtin_imgs directory of the corresponding chip type in the programming tool directory. In this example bl702/builtin_imgs/mfg/bl702_-mfg_stdfw.bin

According to the above configuration, after setting up Dev Cube, configure the chip to UART boot mode, and then

you can start programming.

The method to configure the chip to UART boot mode is as follows:

- Press and hold the Boot button on the module
- Press and release the RST button
- Release the Boot button

After completing the above boot settings, click the Create & Download button to complete the firmware programming. The indication of successful programming is as follows.

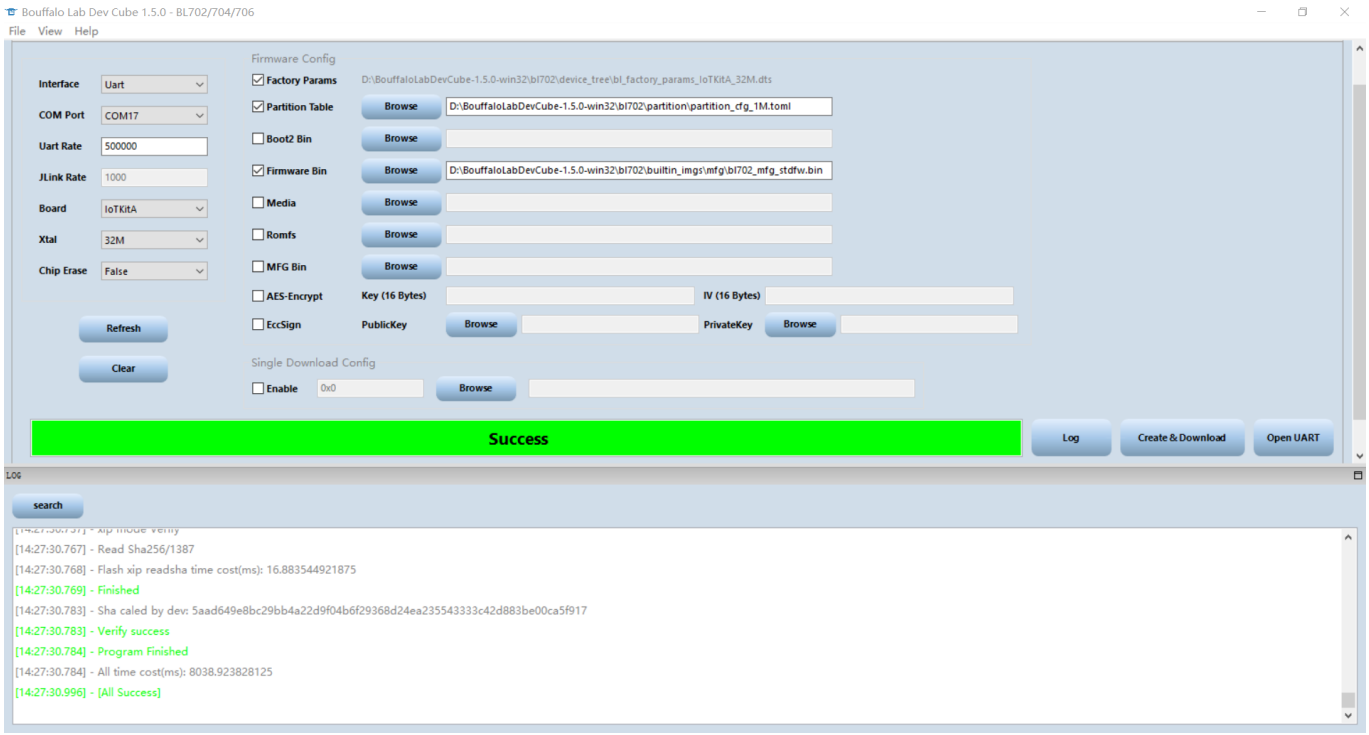


Fig. 4.4: Interface of successful programming

# 5 Run test firmware

For the IOT module (with Flash), after downloading the test firmware, press and release the RST button, and the chip can run the RF test firmware.

On the BLDevCube.exe interface, switch to the RF MFG test interface through View->RF MFG. Select the COM number used and click the Open button to see the log of the test firmware. The example is as follows.

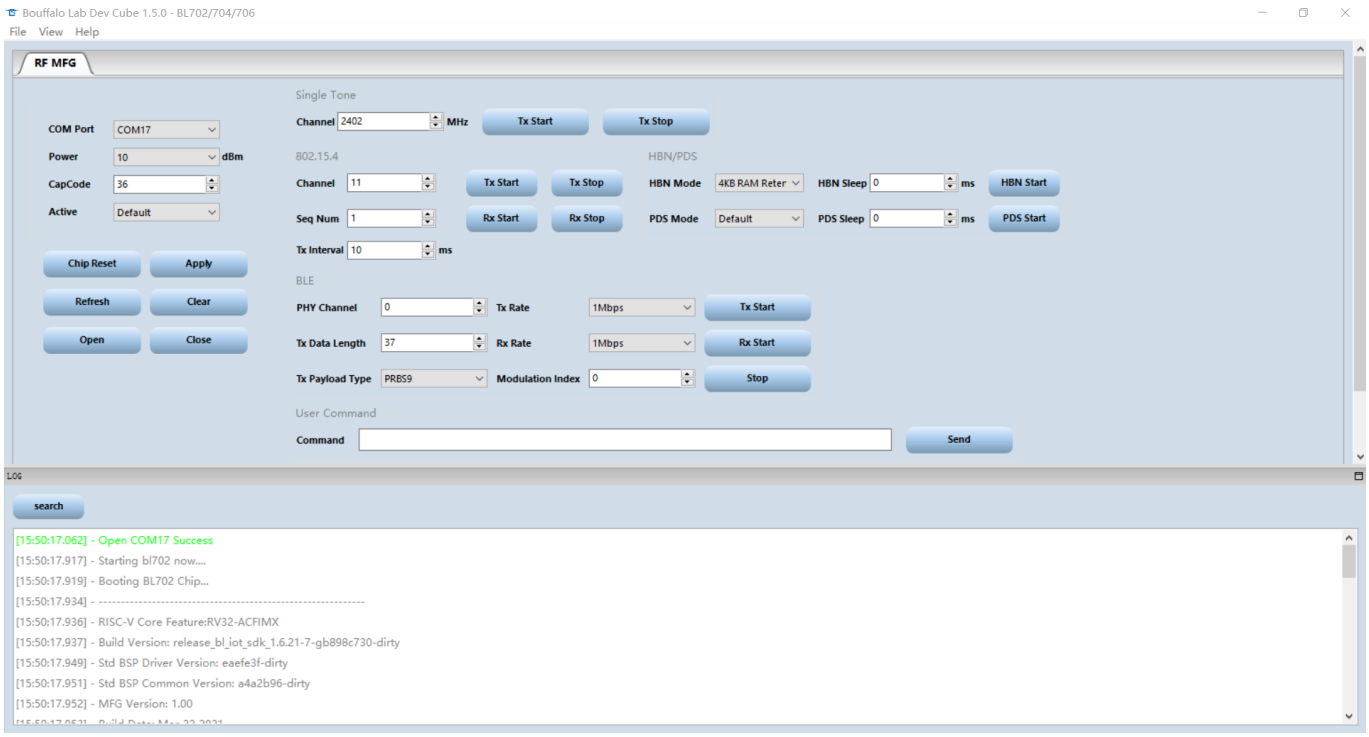


Fig. 5.1: Log of successful operation

The RF test tool running on PC communicates with the test firmware through UART, the baud rate used is 115200, the data bit is 8 bits, and there is no parity.

# 6 Basic configuration

Through the Power drop-down menu, you can set the tx power of the chip, and click the Apply button to take effect. Power can be chosen from 0dBm, 10dBm or 14dBm.

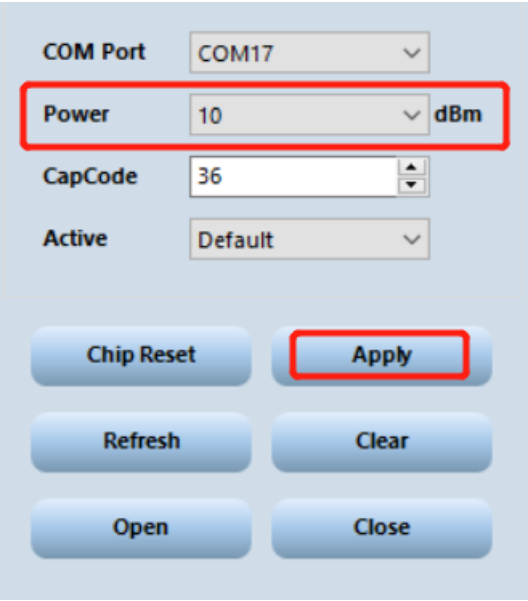


Fig. 6.1: Set tx power

For the load capacitance of the crystal, the BL702 series chip has capacitance compensation inside. Different load capacitance requirements correspond to different capacitance compensation values. The following table provides reference values.

---

**Note:** Actual PCB traces also have certain parasitic capacitance, so the best compensation value is still subject to actual test results.

---

Table 6.1: Capacitance compensation value corresponding to BL702

| XTAL Loading Capacity（pF） | Capacity Code |
|---|---|
| 12 | 32~36 |

The method of use is as follows:

1. Fill in the value to be compensated in the Cap Code.
2. Click the Apply button to update the compensation value.

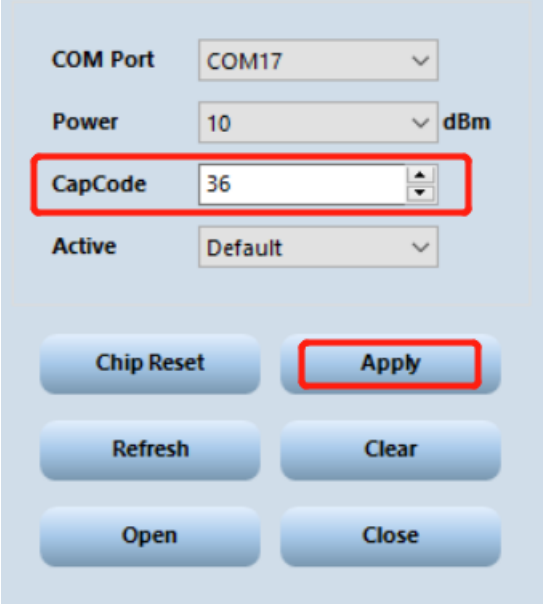


Fig. 6.2: Update compensation value

Through the Active drop-down menu, you can set the active mode of the chip, and click the Apply button to take effect. Among them, Idle mode will turn off more clocks and peripherals to achieve the purpose of reducing active power consumption.

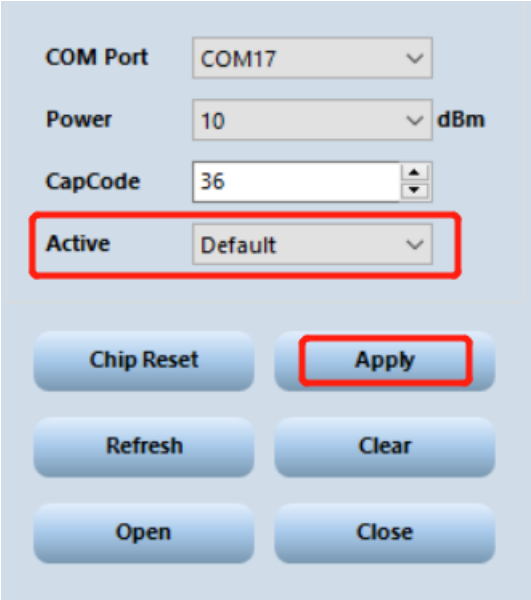


Fig. 6.3: Set active mode

# 7 Single Tone test

MFG supports Single Tone test mode. After setting the Channel, click Tx Start to start the test, and click Tx Stop to stop the test.


Fig. 7.1: Single Tone test

# 8 802.15.4 test

## 8.1 802.15.4 TX test

Set Channel for transmission channel, Seq Num for sequence number of the packet, and Tx Interval for packets tx interval.

Click Tx Start to start the test, and click Tx Stop to stop the test. After the test is over, the LOG area will show how many packets have been sent.

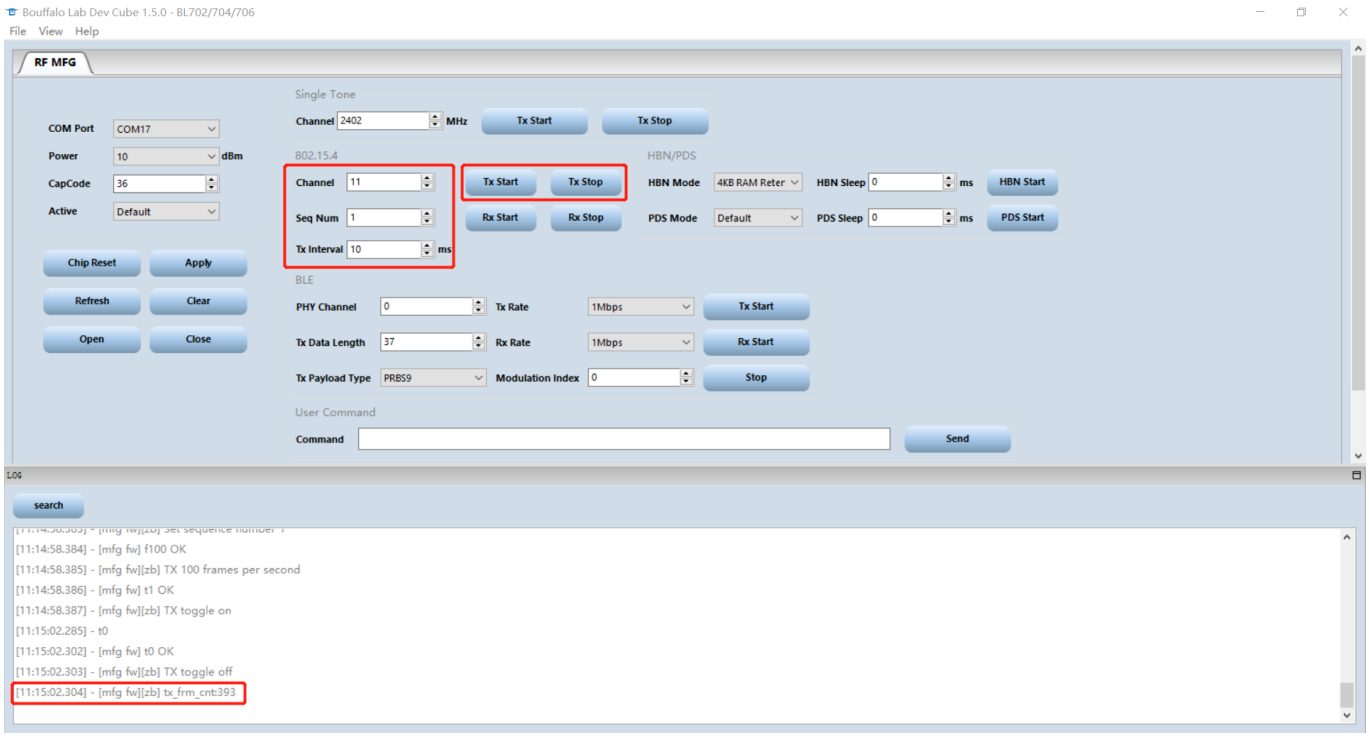


Fig. 8.1: 802.15.4 TX test

## 8.2 802.15.4 RX test

Set Channel for receiving channel, and Seq Num for sequence number of the packet (only packets with matching sequence number will be received).

Click Rx Start to start the test, and click Rx Stop to stop the test. After the test is over, the LOG area will show how many packets have been received, as well as the average rssi, average frequency offset, and average lqi.

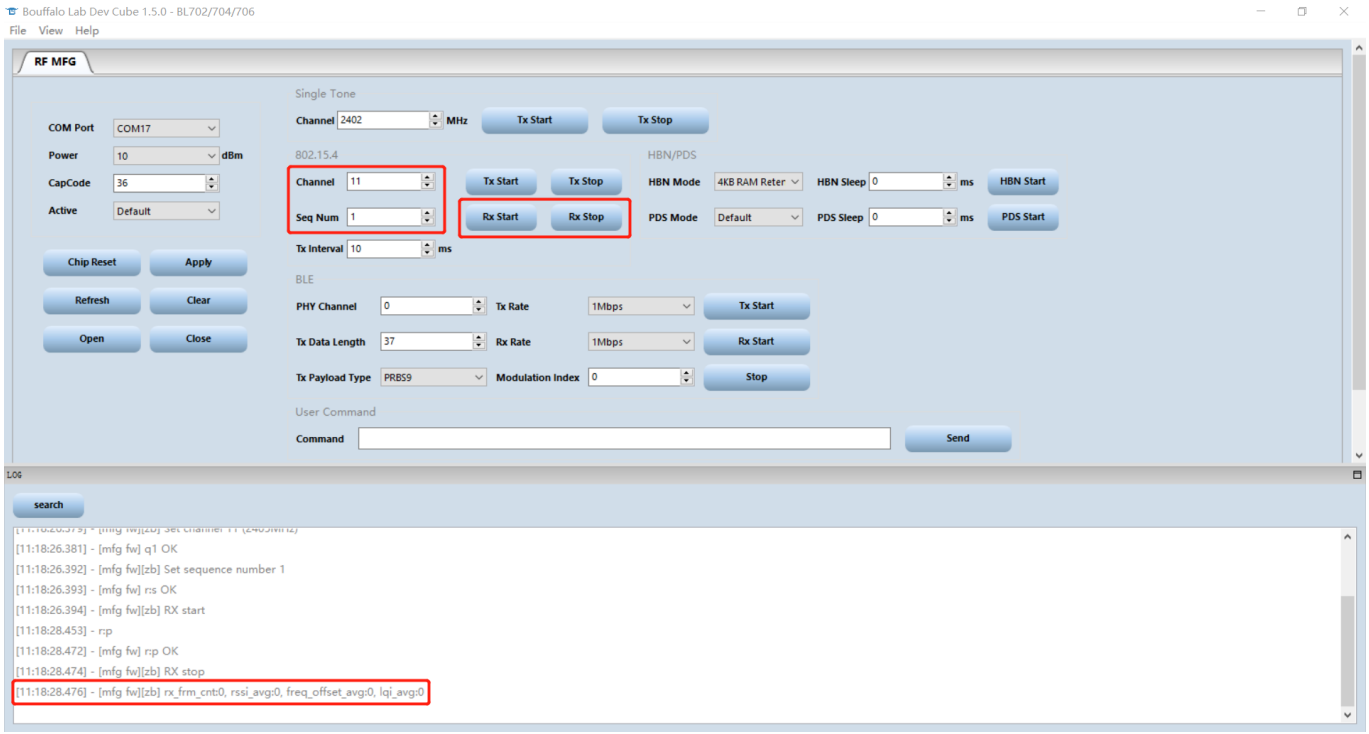


Fig. 8.2: 802.15.4 RX test

# 9 BLE test

RF MFG provides TX and RX testing of BLE.

For TX test, you can set the PHY Channel, Tx Data Length, Tx Payload Type and Tx Rate, and then click the Tx Start button to start the test.

For RX test, you can set the PHY Channel, Rx Rate and Modulation Index, and then click the Rx Start button to start the test.

The test can be stopped by the Stop button.

---

**Note:** The MFG firmware supports standard HCI command. Instruments that supports HCI command (e.g. R&S CMW500) can be connected for test.

---

The Tx Payload Type is shown in the figure below.

| Value | Parameter Description |
|---|---|
| 0x00 | PRBS9 sequence '11111111100000111101…' (in transmission order) as described in [Vol 6] Part F, Section 4.1.5 |
| 0x01 | Repeated '11110000' (in transmission order) sequence as described in [Vol 6] Part F, Section 4.1.5 |
| 0x02 | Repeated '10101010' (in transmission order) sequence as described in [Vol 6] Part F, Section 4.1.5 |
| 0x03 | PRBS15 sequence as described in [Vol 6] Part F, Section 4.1.5 |
| 0x04 | Repeated '11111111' (in transmission order) sequence |
| 0x05 | Repeated '00000000' (in transmission order) sequence |
| 0x06 | Repeated '00001111' (in transmission order) sequence |
| 0x07 | Repeated '01010101' (in transmission order) sequence |

Fig. 9.1: BLE Tx Payload Type

The Tx Rate is shown in the figure below.

*PHY:* *Size: 1 octet*

| Value | Parameter Description |
|---|---|
| 0x01 | Transmitter set to use the LE 1M PHY |
| 0x02 | Transmitter set to use the LE 2M PHY |
| 0x03 | Transmitter set to use the LE Coded PHY with S=8 data coding |
| 0x04 | Transmitter set to use the LE Coded PHY with S=2 data coding |
| All other values | Reserved for future use |

Fig. 9.2: BLE Tx Rate

The Rx Rate is shown in the figure below.

*PHY:* *Size: 1 octet*

| Value | Parameter Description |
|---|---|
| 0x01 | Receiver set to use the LE 1M PHY |
| 0x02 | Receiver set to use the LE 2M PHY |
| 0x03 | Receiver set to use the LE Coded PHY |
| All other values | Reserved for future use |

Fig. 9.3: BLE Rx Rate

The Modulation Index is shown in the figure below.

*Modulation_Index:* *Size: 1 octet*

| Value | Parameter Description |
|---|---|
| 0x00 | Assume transmitter will have a standard modulation index |
| 0x01 | Assume transmitter will have a stable modulation index |
| All other values | Reserved for future use |

Fig. 9.4: BLE Modulation Index

# 10 HBN/PDS test

The HBN/PDS test allows the chip to enter a low power mode. In HBN mode, only a small part of the circuit is kept in working state, while the power supply of other circuits is turned off, thus the power consumption reaches the lowest level.

The chip can be waken up from HBN/PDS mode, and the chip will restart after waken up. The current test tool only supports RTC wakeup.

After setting the HBN/PDS sleep time, click the corresponding Start button to let the chip enter the HBN/PDS mode. After the sleep time expires, the chip will restart.

If the sleep time is set to 0, it means permanent sleep.

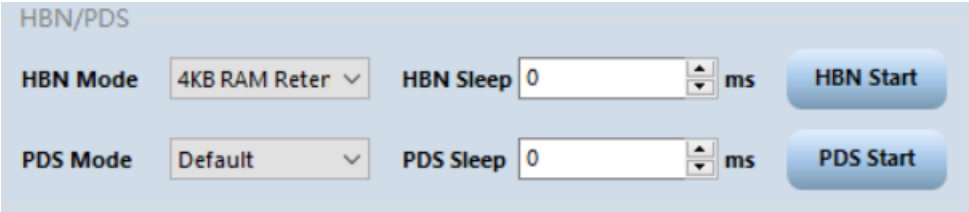


Fig. 10.1: HBN/PDS parameter setting

# 11 Serial communication commands

All commands are of string type.

## 11.1 Shakehand

- Command: `H`
- Return: `mfg`

## 11.2 Set cap code

- Command: `x[cap code]`

cap code: [0, 63]; default: 36

## 11.3 Set power

- Command: `p[power dbm]`

power dbm: 0, 10, 14; default: 10

## 11.4 Set channel

- Command: `c[channel]`

For single tone test, channel is the channel frequency - channel: [2402, 2480]; default: 2402

For 802.15.4 test, channel is the channel index number - channel: [11, 26]; default: 11

## 11.5 TX frequency

- Command: `f[freq]`

freq is the number of packets sent per second

freq: [1, 1000]; default: 100

## 11.6 Set sequence number

- Command: `q[seq num]`

seq num: [0, 255]; default: 1

## 11.7 TX

1. Tx Start: `t1`
2. Tx Stop: `t0`

- Return: `tx_frm_cnt:[tx_frm_cnt]`

## 11.8 RX

1. Rx Start: `r:s`
2. Rx Stop: `r:p`

- Return: `rx_frm_cnt:[rx_frm_cnt], rssi_avg:[rssi_avg], freq_offset_avg:[freq_offset_avg], lqi_avg:[lqi_avg]`

## 11.9 Single Tone

1. Tx Start: `m1`
2. Tx Stop: `m0`

## 11.10 CCA

1. set ed threshold: `C:T[ed threshold]`
2. run cca: `C:M[cca mode]`

- Return: `channel_busy:[channel_busy], rssi:[rssi], ed:[ed]`

ed threshold: [-122, -71]; default: -71

cca mode: [0, 3]; default: 2

## 11.11 Active

1. Default: `i0`
2. Idle: `i1`

## 11.12 HBN

1. set hbn level: `hl[hbn level]`
2. set hbn wakeup pin: `hp[wakeup pin]`
3. enter hbn mode: `ht[sleep time ms]`

hbn level: [0, 3]; default: 0

wakeup pin: 0, 10, 11, 12, 13; default: 0(none)

sleep time ms: [0, 131071999]; default: 0(sleep forever)

## 11.13 PDS

1. set pds level: `sl[pds level]`
2. enter pds mode: `st[sleep time ms]`

pds level: 0, 1, 2, 3, 4, 5, 6, 7, 31; default: 31

sleep time ms: [0, 131071999]; default: 0(sleep forever)

## 11.14 Read Memory

- Command: `RM0x[addr]`
- Return: `Read memory: 0x[addr] = 0x[val]`

## 11.15 Write Memory

- Command: `SM0x[addr]=0x[val]`
- Return: `Write memory: 0x[addr] = 0x[val]`

## 11.16 Get MFG FW version

- Command: `y:v`
- Return: `MFG Version: [version]`

## 11.17 Get MFG FW build infomation

- Command: `y:d`
- Return: `Build Data: [build date] Build Time: [build time]`

## 11.18 Get temperature

- Command: `y:T`
- Return: `Temperature: [temperature]`

## 11.19 Reset MFG FW

- Command: `Reset`

## 11.20 BLE Test

### 11.20.1 BLE TX

- Command: `ETE[channel][tx data length][tx payload type][tx rate]`

All parameters are of hex string type.

Example: ETE00250001

channel: 0x00

tx data length: 0x25

tx payload type: 0x00(PRBS9)

tx rate: 0x01(1Mbps)

### 11.20.2 BLE RX

- Command: `ERE[channel][rx rate][modulation index]`

All parameters are of hex string type.

Example: ERE000100

channel: 0x00

rx rate: 0x01(1Mbps)

modulation index: 0x00

### 11.20.3 BLE test stop

- Command: `EE`