UM11801

Manufacturing Software User Manual for RW61x

Rev. 5.0 — 22 November 2024 User manual

Document information

Information	Content
Keywords	Labtool usage, Wi-Fi Labtool commands, Bluetooth Labtool commands, 802.15.4 radio Labtool commands, UL-OFDMA testing
Abstract	Usage of the Labtool commands to test Wi-Fi, Bluetooth RF, and 805.15.4 radios

1 Introduction

This document provides guidance on the usage of RW61x manufacturing software (Labtool) application. Labtool is a software application used to control and run various RF and regulatory compliance tests.

1.1 Purpose and scope

This document explains how to use the Labtool software application.

The Wi-Fi, Bluetooth LE, and 802.15.4 radio Labtool commands are detailed along with some RF test examples.

For details on how to load the manufacturing firmware and launch the Labtool application, refer to [4].

1.2 About Labtool

Labtool enables RF testing for RW61x device. Labtool is used for the following:

- Measurement of RF parameters such as transmit power, error vector magnitude (EVM), and receiver sensitivity
- Regulatory compliance testing (EMC/EMI)

Labtool is part of the Manufacturing Software (MFG SW) package which includes the following components:

- **Labtool application** which runs on a Windows PC and sends commands to the Device under Test (DUT) using the manufacturing bridge application.
- **RW61x manufacturing (MFG) firmware**: specialized radio firmware image that enables manufacturing test commands for the RW61x RF interfaces.
- **Manufacturing bridge application** that runs on the Cortex-M33 core in the RW61x MCU running FreeRTOS. This application enables Labtool application running on the Windows PC to communicate with the MCU over a UART interface. The manufacturing bridge application is needed for the Labtool application. This application is also used when the RF test automation software from the tester vendor runs on a Windows PC.

1.3 Notation conventions

This document employs the following notation conventions:

- Commands and examples of command outputs (return data) are shown in paragraphs with grey color background

```
This is an example of command
```

- Terms related to commands use a `monospace` font:
  - `parameter`
  - `option`
  - `command name`
- File names, directory names and paths are shown in *italics:*
  - *<file name>.<extension>*
  - *<directory>*
  - *path/to/directory/and/file*

2 Software

2.1 Download

Step 1 – Download the manufacturing software package from NXP website.

- Go to RW610 and RW612 product pages on NXP website ([5] and [6]).
- Look for **Design resources** in the top banner

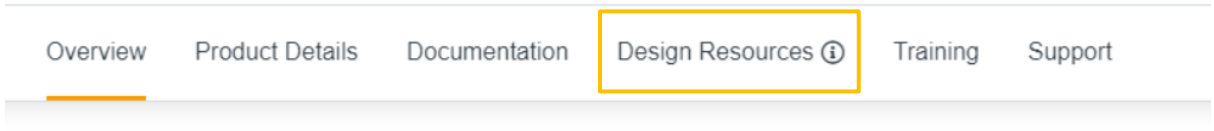

Figure 1. Design Resources link

- Click **Sign in** to access the software files.

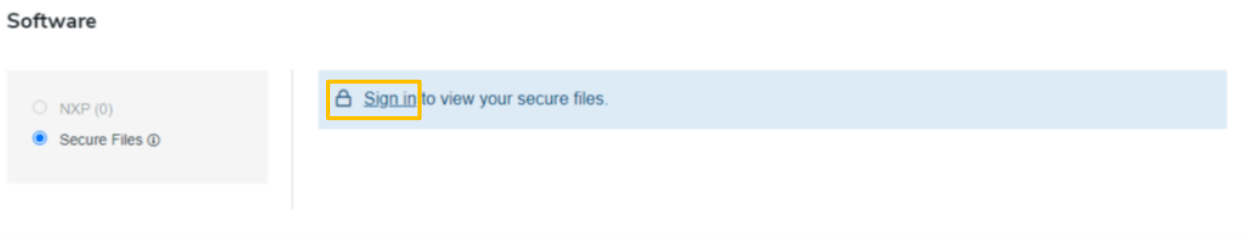

Figure 2. Sign in to access software

- Capture your credentials.
- Select *Software, Debugging and Visualization Tools*.
- Look for the manufacturing software release package.
  For example, *MFG-RW61X-MF-BRG-U16-WIN-X86-1.0.0.9.2-18.80.2.pXX*
- Download the zip file
- Extract the zip file content

2.2 Content

Table 1 shows the content of the manufacturing (MFG) software package.

Table 1. Manufacturing software package content

Type	Name	Description
Directory	Calibration data	Sample calibration files in *.conf* file and *.txt* file formats Driver source code (*.tgz*)
Directory	FWImage	MFG firmware binary files
Directory	Labtool	Precompiled Labtool application • *GenHeader.bin* • *SetUp.ini* • Labtool application *DutApiMimoApApp_LABTOOL_UNIFIED.exe*
PDF file	Release Notes	Release notes
Directory	License files	• *COPYING* • SCR_MFG-RW61X-MF-BRG-U16-WIN-X86-1.0.0.9.2-18.80.2.pX.X

Table 2 shows the content of the native manufacturing (MFG) software package.

Table 2. Manufacturing software package content

Name	Description
prog_flash_AX	Batch file
prog_flash_AX.jlink	J-Link file
uartbridge_AX.bin	MCU firmware and manufacturing bridge binary file
rw610w_mfg_sfw_cpu1_AX.bin	Wi-Fi firmware
rw610w_mfg_sfw_cpu2_AX.bin	Bluetooth LE/802.15.4 firmware

3 Labtool test setup

Figure 3 shows the test set-up for RW61x.

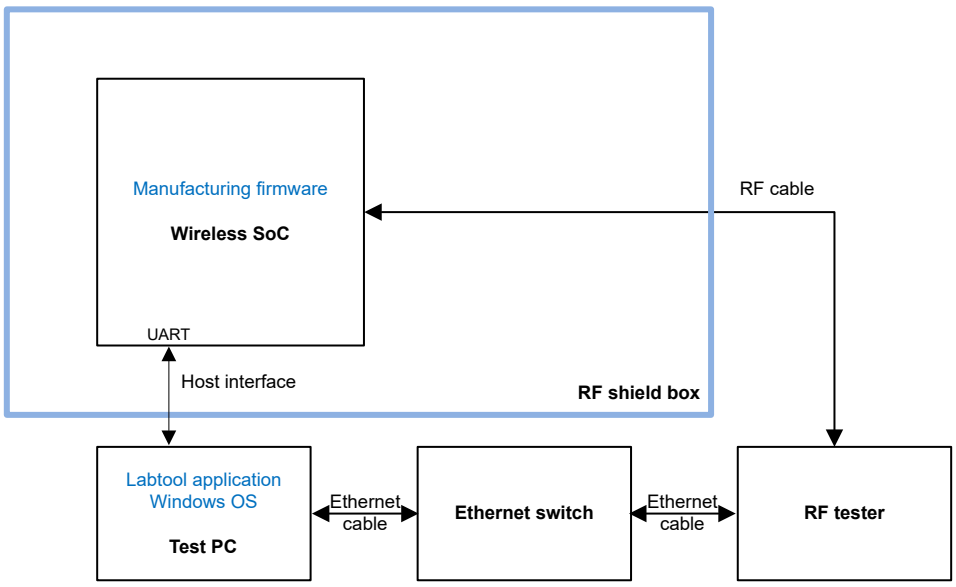

Figure 3. Set-up for RW61x Labtool bridge test mode

Where:

- **Wireless SoC**: RW61x evaluation board.
  Holds the device drivers, manufacturing firmware binaries, and manufacturing bridge application.
- **Test PC**: Windows 10 PC used to run the Labtool application.
- **RF tester**: RF PHY test instrument
- **Test accessories**: Splitter[1], Ethernet switch, AC adapter[1], and cables
- **RF shield box**: provides a shielded RF environment.

[1] Optional equipment used for the set-up

4 Update the manufacturing bridge application and/or firmware

To update the manufacturing bridge application and/or firmware:

Step 1 - Ensure the jumpers are installed on HD12 position 1-2 and 3-4 on NXP RW61x reference design board as shown in Figure 4.

- Set all U38 DIP switches to **OFF** positions – strap **1111** for Flex SPI Boot.

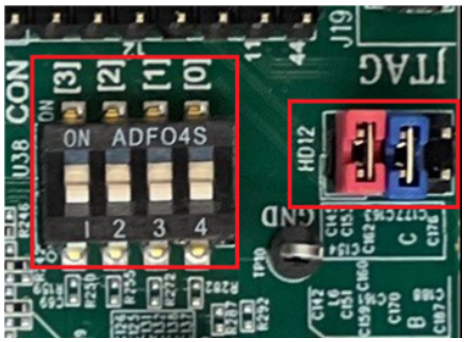

Figure 4. HD12 and U38CON configuration on RW61x reference design

Step 2 - Install a patch on the Windows PC. Refer to the section *Pre-requisites for RW61x image setup* in [4].

Step 3 - Update the manufacturing bridge application and firmware.

- Go to *FwImage/AX* directory in Labtool software package.
- Open the Command Prompt on the Windows PC and execute the commands to flash the board with the new files:

```
prog_flash_AX.bat
C:\>"C:\Program Files\SEGGER\JLINK\JLINK.exe" prog_flash_AX.jlink
```

The *prog_flash_RW610_AX.jlink* file includes:

```
Device RW610
SelectInterface SWD //SelectInterface JTAG
Speed 1000
JTAGConf -1,-1
Connect
Sleep 1000
Halt
loadbin uart_wifi_ble_15d4_bridge.bin,0x08000000
Sleep 2000
loadbin rw61xw_sb_mfg_fw_cpu1_a2.bin,0x08400000
Sleep 1000
loadbin rw61xn_sb_mfg_fw_cpu2_combo_a2.bin,0x085e0000
Sleep 1000
Reset
Exit
```

Note:

- *Flashing the binary files may take a few minutes*
- *The addresses indicated above for the binary files may change in subsequent releases of the firmware and MFG bridge application*

- Wait for the board flashing to complete.
- Power cycle the DUT.

Example of output message during flashing:

```
C:\Labtool\RW61x\MFG-RW61X-MF-BRG-U16-WIN-X86-1.0.0.9.2-18.80.2.p9.2\FwImage
\A0>prog_flash_A0.bat
C:\Labtool\RW61x\MFG-RW61X-MF-BRG-U16-WIN-X86-1.0.0.9.2-18.80.2.p9.2\FwImage\A0>"C:
\Program Files\SEGGER\JLINK\JLINK.exe" prog_flash_A0.jlink
SEGGER J-Link Commander V7.54b (Compiled Sep 14 2021 16:13:05)
DLL version V7.54b, compiled Sep 14 2021 16:11:46
J-Link Command File read successfully.
Processing script file...
J-Link connection not established yet but required for command.
Connecting to J-Link via USB...O.K.
Firmware: J-Link MCU-Link V1 compiled Mar  9 2022 12:25:10
Hardware version: V1.00
S/N: 1063988636
VTref=3.300V
Selecting SWD as current target interface.
Selecting 1000 kHz as target interface speed
Device "RW610" selected.
Connecting to target via SWD
InitTarget() start
*****************************************************
J-Link script: RW610 Cortex-M33 core J-Link script
*****************************************************
InitTarget() end
Found SW-DP with ID 0x6BA02477
DPIDR: 0x6BA02477
AP map detection skipped. Manually configured AP map found.
AP[0]: AHB-AP (IDR: Not set)
AP[1]: AHB-AP (IDR: Not set)
AP[2]: AHB-AP (IDR: Not set)
AP[0]: Core found
AP[0]: AHB-AP ROM base: 0xE00FF000
CPUID register: 0x410FD214. Implementer code: 0x41 (ARM)
Feature set: Mainline
Found Cortex-M33 r0p4, Little endian.
FPUnit: 8 code (BP) slots and 0 literal slots
Security extension: implemented
Secure debug: enabled
CoreSight components:
ROMTbl[0] @ E00FF000
[0][0]: E000E000 CID B105900D PID 000BBD21 DEVARCH 47702A04 DEVTYPE 00 Cortex-M33
[0][1]: E0001000 CID B105900D PID 000BBD21 DEVARCH 47701A02 DEVTYPE 00 DWT
[0][2]: E0002000 CID B105900D PID 000BBD21 DEVARCH 47701A03 DEVTYPE 00 FPB
[0][3]: E0000000 CID B105900D PID 000BBD21 DEVARCH 47701A01 DEVTYPE 43 ITM
[0][5]: E0041000 CID B105900D PID 002BBD21 DEVARCH 47724A13 DEVTYPE 13 ETM
[0][6]: E0042000 CID B105900D PID 000BBD21 DEVARCH 47701A14 DEVTYPE 14 CSS600-CTI
Cortex-M33 identified.
Sleep(1000)
PC = 20002DCC, CycleCnt = 10427375
R0 = 00000001, R1 = 01010101, R2 = 02020202, R3 = 03030303
R4 = 00000000, R5 = 05050505, R6 = 06060606, R7 = 07070707
R8 = 08080808, R9 = 09090909, R10= 10101010, R11= 11111111
R12= 12121212
SP(R13)= 2000EC90, MSP= 2012FF88, PSP= 2000EC90, R14(LR) = 20002D5D
XPSR = 81000000: APSR = Nzcvq, EPSR = 01000000, IPSR = 000 (NoException)
CFBP = 02000000, CONTROL = 02, FAULTMASK = 00, BASEPRI = 00, PRIMASK = 00
```

```
MSPLIM = 2012FC00
PSPLIM = 2000EAA0
Security extension regs:
MSP_S = 2012FF88, MSP_NS = 00000000
MSPLIM_S = 2012FC00, MSPLIM_NS = 00000000
PSP_S = 2000EC90, PSP_NS = FFFFFFFC
PSPLIM_S = 2000EAA0, PSPLIM_NS = 00000000
CONTROL_S  = 02, FAULTMASK_S  = 00, BASEPRI_S  = 00, PRIMASK_S  = 00
CONTROL_NS = 00, FAULTMASK_NS = 00, BASEPRI_NS = 00, PRIMASK_NS = 00
FPS0 = 00000000, FPS1 = 00000000, FPS2 = 00000000, FPS3 = 00000000
FPS4 = 00000000, FPS5 = 00000000, FPS6 = 00000000, FPS7 = 00000000
FPS8 = 00000000, FPS9 = 00000000, FPS10= 00000000, FPS11= 00000000
FPS12= 00000000, FPS13= 00000000, FPS14= 00000000, FPS15= 00000000
FPS16= 00000000, FPS17= 00000000, FPS18= 00000000, FPS19= 00000000
FPS20= 00000000, FPS21= 00000000, FPS22= 00000000, FPS23= 00000000
FPS24= 00000000, FPS25= 00000000, FPS26= 00000000, FPS27= 00000000
FPS28= 00000000, FPS29= 00000000, FPS30= 00000000, FPS31= 00000000
FPSCR= 02000000
Downloading file [uart_bridge_A0.bin]...
J-Link: Flash download: Bank 0 @ 0x08000000: 1 range affected (49152 bytes)
J-Link: Flash download: Total: 9.231s (Prepare: 0.225s, Compare: 2.730s, Erase: 0.637s,
 Program: 4.868s, Verify: 0.693s, Restore: 0.076s)
J-Link: Flash download: Program speed: 10 KB/s
****** Error: Timeout while preparing target, core does not stop. (PC = 0x080031CE, XPSR
 = 0x01000003, SP = 0x2012FF88)!
Failed to initialize RAMCode
O.K.
Sleep(2000)
Downloading file [rw610w_mfg_sfw_cpu1_A0.bin]...
J-Link: Flash download: Bank 0 @ 0x08000000: 1 range affected (487424 bytes)
J-Link: Flash download: Total: 73.645s (Prepare: 0.168s, Compare: 13.076s, Erase: 5.590s,
 Program: 48.215s, Verify: 6.517s, Restore: 0.076s)
J-Link: Flash download: Program speed: 10 KB/s
O.K.
Sleep(1000)
Downloading file [rw610n_mfg_sfw_cpu2_A0.bin]...
J-Link: Flash download: Bank 0 @ 0x08000000: Skipped. Contents already match
O.K.
Sleep(1000)
Reset delay: 0 ms
Reset type NORMAL: Resets core & peripherals via SYSRESETREQ & VECTRESET bit.
Reset: ARMv8M core with Security Extension enabled detected.
Reset: Halt core after reset via DEMCR.VC_CORERESET.
Reset: Reset device via AIRCR.SYSRESETREQ.
AfterResetTarget() start
AfterResetTarget() end
Script processing completed.
```

5 Labtool setup on Windows PC

The Host PC must run 64 bit Windows operating system. Follow these steps to set up the environment.

- Go to the directory where you extracted MFG software release package.
- Copy *.../labtool* directory to C: or D: drive of the host PC to keep the path length less than 255 characters.

Figure 5 shows *Labtool* directory content.

Figure 5. Content of *Labtool* directory

- Power up RW61x Reference Design Board and plug it in to an available USB port of the test PC
- Open Device Manager on the test PC. Look for **JLink CDC UART** port number as shown in Figure 6

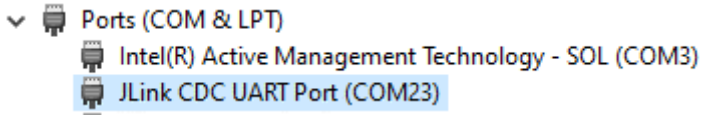

Figure 6. JLink CDC UART port number

- Open *SetUp.ini* file located in the Labtool working directory using a text editor such as Notepad
- Look for `ComNo` in *SetUp.ini* file and set the value to match the JLink CDC UART Port number shown in Device Manager on the test PC

```
ComNo = 23
BaudRate = 115200
byParity = 0
byStopBits = 1
byByteSize = 8
```

- Set `DUT_15_4_If_Type` to `3` (UART-HDLC)

```
[ZBEE_INTF]
;15.4 Interface type: 0 Legacy15.4, 1 SPI, 2 HDLC, 3 UART-HDLC
DUT_15_4_If_Type=3
;ZBEE_PORT = /dev/ttyUSB0
ZBEE_PORT=/dev/spidev1.0
DUT_15_4_GPIO_Pin=12
DUT_15_4_GPIO_Port=/dev/gpiochip5
DUT_15_4_SPI_SPEED=1000000
```

- Run *DutApiMimoApApp_LABTOOL_UNIFIED.exe* application to start Labtool. See Section 7.1

6 Calibration data

The device under test requires calibration data for optimal RF performance. The calibration data can be stored in the on-chip OTP, or in a *.conf* file.

In the *SetUp.ini* file, the parameter `NO_EEPROM` is used to specify the calibration data storage option. If using the configuration file option, place the *.conf* file in the Labtool working directory and name the file as *WlanCalData_ext.conf*.

Table 3. `NO_EEPROM` parameter description

Parameter	Value	Cal data storage selection
NO_EEPROM	1	*.conf* file
	2	OTP

Save the *Setup.ini* file after making the changes.

Note: *If using the configuration file option for the calibration data, make sure to issue the Labtool command "`22` before issuing any RF measurement test sequence. Command `22` loads the calibration data from the .conf file to firmware.*

7 Labtool usage

7.1 Starting Labtool

To start the Labtool application, run the desired Labtool application executable *DutApiMimoApApp_LABTOOL_UNIFIED.exe*.

The Labtool start window opens:

```
Name: Dut labtool Version: 1.0.0.03
Date: Apr 26 2022 (22:57:58)
Note:
1. =========WiFi tool=============
2. =========BT tool=============
3. =========15_4 tool=============
Enter CMD 99 to Exit Enter option:
```

- Enter **1** at the command prompt to start operating the Wi-Fi radio. Enter **2** to start operating Bluetooth LE radio. Enter **3** to start operating 802.15.4 radio.

- For Wi-Fi, enter Wi-Fi menu and issue the command `88` to check the firmware and Labtool versions.
  Refer to the MFG software release notes for the firmware and Labtool versions.
  If cmd `88` returns the same version numbers as in the release notes, start the RF testing.
  If the version numbers are incorrect, check the host, DUT and bridge connections and verify that the Wi-Fi and Bluetooth (RW610) or that the Wi-Fi and Narrowband (Bluetooth + 802.15.4) (RW612) firmware is loaded correctly.

Example of correct response of command `88` execution in Wi-Fi Labtool menu.

```
1. =========WiFi tool=============
2. =========BT   tool=============
3. =========15_4   tool=============

Enter CMD 99 to Exit

Enter option: 1

Name: DutApiClass Interface: EtherNet Version:        1.0.0.3
Date:           Apr 26 2022 (22:57:41)

Note:

DutIf_InitConnection: 0
-----------------------------------------------------
RW610 (802.11a/g/b/n/ac/ax) TEST MENU
-----------------------------------------------------
Enter option: 88
DLL Version : 1.0.0.3
LabTool Version: 1.0.0.3
FW Version:  18.80.1.103
Mfg Version: 2.0.0.63
SFW Version:  0.0.0.00
SHAL Version: 0.0.0.0
SOC OR Version: 0.d
Customer ID:   0
RF OR Version:  0.7
Customer ID:   0
Enter option:
```

- For Bluetooth, enter Bluetooth menu and issue the command `88` to check the firmware and Labtool versions. Refer to the MFG software release notes for the firmware and Labtool versions.
  If cmd `88` returns the same version numbers as in the release notes, start the RF testing.
  If the version numbers are incorrect, check the host, DUT and bridge connections and verify that the Wi-Fi and Narrowband (Bluetooth LE for RW610) (Bluetooth LE + 802.15.4 for RW612) firmware is loaded correctly.

Example of correct response of command `88` execution in Bluetooth Labtool menu.

```
1. =========WiFi tool=============
2. =========BT   tool=============
3. =========15_4   tool=============

Enter CMD 99 to Exit

Enter option: 2
Name: DutApiClass Interface:      Version:        1.0.0.3
Date:           Apr 26 2022 (22:57:38)

Note:

Dut_Bt_OpenDevice: 0x00000000
-------------------------------------------------------------------------
W87xx (BT) TEST MENU
-------------------------------------------------------------------------
Enter option: 88
88

DLL Version : 1.0.0.3
LabTool Version: 1.0.0.3
FW Version:  18.80.1.103
Mfg Version: 1.0.0.11
Enter option:
```

7.1.1 Labtool command response

The console returns a status byte for each Labtool command execution. A successful command execution is returned with a status byte with all 0s.

Example of successful command execution:

```
Enter option: 12 0 1

DutIf_SetRfChannel: 0x00000000
RF Channel: 1 (2412.0 MHz)
Enter option: 11

DutIf_GetRfChannel: 0x00000000
RF Channel: 1 (2412.0 MHz)
Enter option:
```

The return value for a failed command is a non-zero status byte in the range of `0x00000001` to `0xFFFFFFFF`. A failed command returns invalid data.

The possible reasons for failed commands are:

- Communication issue with the DUT
- Incorrect firmware version
- No firmware download
- Wrong version of Labtool/firmware

Example of failed command:

```
Enter option: 12 16

DutIf_SetRfChannel: 0xFFFFFFFF
Error Source : Firmware
```

7.2 Listing Labtool commands

Enter `?` at the Labtool prompt to display the list of all the available commands.

7.3 Using the help menu

To display the help menu listing the supported options for any command, enter the command `number` followed by `?`.

7.4 Keeping default parameter values

When a Labtool command is issued with no value for a Parameter, the default value for this parameter is used.

7.5 Closing Labtool

To close Labtool, issue the command `99`.

Issue the command `99` twice to fully exit Labtool when in WiFi, or Bluetooth.

8 Wi-Fi Labtool commands

8.1 Naming convention

All Labtool commands start with a number followed by parameter values when applicable. The following notation is used to present the commands:

- Each command is described in a subsection
- The heading of the subsection includes the command number followed by a short description of the command.
  For example: **Command 5: Get radio mode index info**

8.2 Labtool command list for Wi-Fi

Table 4. Labtool command list for Wi-Fi

Command description
Get/set commands for general parameters
Command 88: Get the firmware and Labtool versions
Command 5: Get radio mode index info
Command 6: Set radio mode index
Command 9: Get antenna
Command 10: Set antenna
Command 11: Get Wi-Fi RF channel
Command 12: Set Wi-Fi RF channel
Command 111: Get the channel bandwidth
Command 112: Set the channel bandwidth
Commands for Wi-Fi transmit
Command 35: Transmit with SIFS gap
Command 18: Set Wi-Fi CW TX
Command 225: Transmit 802.11ax with SIFS gap
Command 231: Enable uplink OFDMA Tx
Command 235 - Transmit OFDMA packet with configuration file
Commands for Wi-Fi receive
Command 32: Get the received packet count (Stop Rx FER test)
Command 31: Clear the received packet count
Command 198: Start RSSI data collection
Command 199: Stop RSSI data collection and report result
Commands for OTP/calibration data file
Command 44: Get/set the storage type
Command 22: Load calibration data from the configuration file
Command 53: Write calibration data from text files to OTP or .conf file
Command 54: Get calibration data from OTP memory into text file

Table 4. Labtool command list for Wi-Fi*...continued*

Command description
Command 120: Get the thermal sensor reading
Command 121: Get the power detector reading offset
Command 101: Get RF control mode
Command 102: Set RF control mode
Commands for MAC address operation
Command 45: Read the MAC address from OTP memory
Command 46: Write MAC address to OTP memory
Commands for crystal calibration
Command 95: Get RF crystal calibration offset
Command 96: Set RF crystal calibration offset
Command to exit Labtool
Command 99: Exit Labtool

8.3 Command 88: Get the firmware and Labtool versions

This command returns the firmware, and Labtool versions.

Syntax: `88`

Example of return data:

```
88

DLL Version : 1.0.0.8
LabTool Version: 1.0.0.8
FW Version:  18.80.254.23 Mfg Version: 2.0.0.63
SFW Version:  0.0.0.03    SHAL Version: 0.0.0.0
SOC OR Version: 0.9 Customer ID: 0
RF OR Version:  2.5 Customer ID: 0
Enter option:
```

8.4 Command 5: Get radio mode index info

This command reads back the radio mode index set for the Wi-Fi radio. The radio mode index corresponds to the Wi-Fi frequency band set for the Wi-Fi radio.

Syntax: `5`

Example

`5` // Returns the radio mode index set for the Wi-Fi radio. See Table 5.

Table 5. Radio mode indexes

Radio mode index	Description
0	Radio in power down mode
3	Radio in 5 GHz band
11	Radio in 2.4 GHz band

8.5 Command 6: Set radio mode index

This command sets the radio mode index for the Wi-Fi radio.

The Wi-Fi radio can be configured to a desired band (2.4 GHz /5 GHz) using the parameter `radio mode index` in the command.

Syntax: `6 <radio mode index for Wi-Fi radio>`

Where:

Table 6. Command parameters

Parameter	Description
`radio mode index for Wi-Fi radio`	The radio mode index corresponds to a specific frequency band (2.4 GHz/5 GHz). 0 = sets the radio in power-down mode 3 = sets the radio in 5 GHz band 11 = Sets the radio in 2.4 GHz band

Examples

`6 3` // Configures the Wi-Fi radio in 5 GHz band

`6 11` // Configures the Wi-Fi radio in 2.4GHz band

Return data:

- Successful command: all-0 status byte (0x00000000)
- Failed command: non-0 status byte (0x00000001 to 0xFFFFFFFF)

8.6 Command 9: Get antenna

This command reads back the antenna TX/RX path configuration.

Note: *Command 9 is for designs with antenna diversity.*

Syntax: `9`

Example

`9` // Returns the antenna TX/RX path configuration

Return data

Transmit antenna

- 1 = antenna 1
- 2 = antenna 2
- 3 = antenna 3

Receive antenna

- 1 = antenna 1
- 2 = antenna 2
- 3 = antenna 3

Return data example:

```
Enter option: 9
DutIf_GetTxAntenna : Tx Ant 2(10)      0
DutIf_GetRxAntenna : Rx Ant 2(10)      0
Enter option:
```

8.7 Command 10: Set antenna

This command sets the TX/RX antenna path configuration.

Note: *Command 10 is for designs with antenna diversity.*

Syntax: `10 <Radio_ID> <tx_ant> <rx_ant>`

Where:

Table 7. Command parameters

Parameter	Description
`Radio_ID`	Radio identifier set to 0.
`tx_ant`	Transmit antenna 1 = antenna 1 2 = antenna 2 3 = antenna 3
`rx_ant`	Receive antenna 1 = antenna 1 2 = antenna 2 3 = antenna 3

Example

`10 0 2 2` // Set both TX and RX to antenna 2

Return data:

- Successful command: all-0 status byte (0x00000000)
- Failed command: non-0 status byte (0x00000001 to 0xFFFFFFFF)

Return data example:

```
Enter option: 10 0 2 2
DutIf_SetTxAntenna to 2 : status 0
DutIf_SetRxAntenna to 2 : status 0
```

8.8 Command 11: Get Wi-Fi RF channel

This command is used to read/get the 2.4 GHz or 5 GHz RF channel in use. The command requires to set `Radio_ID` value as 0.

Syntax: `11 <Radio_ID>`

Where:

Table 8. Command parameters

Parameter	Description
`Radio_ID`	Radio identifier set to 0

Examples

`11 0` // Gets the current 2.4 GHz/5 GHz channel in use for the Wi-Fi radio

Return data:

- `3` = the 5 GHz channel is in use
- `11` = the 2.4 GHz channel is in use

8.9 Command 12: Set Wi-Fi RF channel

This command sets the 2.4 GHz/5 GHz RF channel for the specified radio. The command requires to set `Radio_ID` value as 0.

Syntax: `12 <Radio_ID> <Channel>`

Where:

Table 9. Command parameters

Parameter	Description
`Radio_ID`	Radio identifier set to 0
`Channel`	Channel identifier. Refer to Section 10 for the list of Wi-Fi channels.

Examples

`12 0 6` // Sets the Wi-Fi radio to 2.4 GHz channel 6

`12 0 36` // Sets the Wi-Fi radio to 5 GHz channel 36

Return data:

- Successful command: all-0 status byte (0x00000000)
- Failed command: non-0 status byte (0x00000001 to 0xFFFFFFFF)

8.10 Command 111: Get the channel bandwidth

This command returns the channel bandwidth in use for the Wi-Fi radio. The command requires to set `Radio_ID` value as 0.

Syntax: `111 <Radio_ID>`

Where:

Table 10. Command parameters

Parameter	Description
`Radio_ID`	Radio identifier value set to 0

Examples

`111 0` // Get the Wi-Fi radio channel

Return values:

- ChBW = 0: 20 MHz

8.11 Command 112: Set the channel bandwidth

This command requires to set `Radio_ID` and `ChBW` values to 0.

Syntax: `112 <Radio_ID> <ChBW>`

Where:

Table 11. Command parameters

Parameter	Description
`Radio_ID`	Radio identifier set to 0
`ChBW`	Channel bandwidth value • 0 = 20 MHz

Examples

`112 0 0` // Sets the 5 GHz channel bandwidth to 20 MHz

Return data:

- Successful command: all-0 status byte (0x00000000)
- Failed command: non-0 status byte (0x00000001 to 0xFFFFFFFF)

8.12 Command 35: Transmit with SIFS gap

This command is used to transmit a modulated signal with SIFS gap. It emulates the packet-based transmission with a packet to packet gap specified by a configurable SIFS interval.

Syntax: `35 <Radio_ID> <Enable> <Rate> <Power Level> <Adjust Tx Burst Gap> <Burst Sifs in us> <Packet length> <Pattern> <Short Preamble> <Short GI> <Adv Coding> <Tx Bf On> <GF Mode> <STBC> <DPD> <Signal BW> <bssid>`

Where:

Table 12. Command parameters

Parameter	Description
`Radio_ID`	Radio identifier set to 0
`Enable`	Enable/disable transmit 0 = disable 1 = enable
`Rate`	Data rate. See Section 10.1 "Data rates".
`Power level`	Target set power in dBm, in 1 dB steps
`Adjust Tx Burst Gap`	Enable/disable adjust Tx burst gap 0 = disable 1 = enable
`Burst Sifs in us`	Burst SIFS gap. Minimum 10 µs, maximum 255 µs.
`Packet length`	Payload length in bytes
`Pattern`	0xB496DEB6 = default value for MFG testing
`Short preamble`	Set long/short preamble 0 = long preamble 1 = short preamble
`Short GI`	Set long/short guard interval (GI) 0 = long GI 1 = short GI
`Adv Coding`	Set signal encoding to BCC 0 = BCC
`Tx Bf On`	This argument is non-functional and should be set to 0
`GF mode`	Enable/disable greenfield (GF) mode 0 = disable 1 = enable
`STBC`	Enable/disable space-time block coding (STBC) 0 = disable 1 = enable
`DPD`	This argument is non-functional and should be set to 0
`Signal BW`	Signal bandwidth. Set to 0.
`BSSID`	Basic service set ID. If not set, all 0xF are used.

Note: *Not all VHT or HE rates are available for all the bandwidth configurations. The error code `0xE` in command `35` response means that the rate for the respective radio configuration is not supported.*

Examples

```
35 0 1 4 12 // Transmit at 11 Mbps at power level = 12 dBm
35 0 0 // Turn off the transmission
```

Return data:

- Successful command: all-0 status byte (0x00000000)
- Failed command: non-0 status byte (0x00000001 to 0xFFFFFFFF)

8.13 Command 225: Transmit 802.11ax with SIFS gap

The command is used to continuously transmit 802.11ax packets with SIFS gap. The command emulates the packet-based transmission with a packet-to-packet gap specified by a configurable SIFS interval.

Syntax: `225 <Radio_ID> <Enable> <Rate> <Power level> <Adjust Tx Burst Gap> <Burst Sifs in us> <Packet length> <Pattern> <Preamble type> <Short GI> <Adv Coding> <Tx Bf> <GF mode> <STBC> <DPD> <Signal BW> <BSSID> <NumPkt> <MaxPktExt> <BeamChange> <DCM><Doppler> <MidamblePeriod> <QNum>`

Where:

Table 13. Command parameters

Parameter	Description
`Radio_ID`	Radio identifier 0 = radio 0
`Enable`	Enable/disable transmit 0 = disable 1 = enable
`Rate`	Data rate. See Section 10.1 "Data rates".
`Power level`	Target set power in dBm
`Adjust Tx Burst Gap`	Enable/disable adjust Tx burst gap 0 = disable (default) 1 = enable
`Burst Sifs in us`	Burst Sifs gap. Minimum 10 µs, maximum 255 µs
`Packet length`	Payload length in bytes
`Pattern`	0xB496DEB6 = default value for MFG testing
`Preamble type`	Set the preamble type Use the default value (-1) unless another value is required For legacy 802.11b: preamble type • 0 = long • 1 = short For legacy 802.11g: not valid For 802.11n: GreenField PPDU indicator: • 0 = HT-mix • 1 = HT-GF For 802.11ac: not valid For 802.11ax: PPDU type: • 0 = HE-SU • 1 = HE-EXT-SU • 2 = HE-MU • 3 = HE-Trigger-based

Table 13. Command parameters*...continued*

Parameter	Description
`Short GI`	Enable short GI. For 802.11ax: 0 = 1xHELTF+GI0.8 us 1 = 2xHELTF+GI0.8 us 2 = 2xHELTF+GI1.6 us 3 = 4xHELTF+GI0.8 us if both DCM and STBC are 1; 4xHELTF+GI3.2 us otherwise For 802.11ax HE MU PPDU: 0 = 4xHELTF+GI0.8 us 1 = 2xHELTF+GI0.8 us 2 = 2xHELTF+GI1.6 us 3 = 4xHELTF+GI3.2 us
`Adv Coding`	Set signal encoding to BCC 0 = BCC 1 = reserved
`Tx Bf`	This argument is non-functional and should be set to 0
`GF mode`	Enable/disable greenfield (GF) mode 0 = disable 1 = enable
`STBC`	Enable/disable space-time block coding (STBC) 0 = disable 1 = enable
`DPD`	This argument is non-functional and should be set to 0
`Signal BW`	Signal BW=-1 (-1 follows the device bandwidth specified in Command 112/Command 111) ***Note:*** *For testing with HE-ER-SU (242 tone RU), set `Signal BW` to 0 and rate to MCS 0, 1, or 2.* *For testing with HE-ER-SU (106 tone RU), set Signal BW to 1 and rate to MCS 0F.*
`BSSID`	Basic service set ID. If not set, all 0xF are used.
`NumPkt`	Set to default value -1
`MaxPktExt`	Set to default value -1
`BeamChange`	Set to default value -1
`DCM`	Set to default value -1
`Doppler`	Set to default value -1
`MidamblePeriod`	Set to default value -1
`QNum`	Transmit queue number that holds the trigger-based response packets. Set to 1 (default value).

Examples

`225 0 1 2100 10` // Set radio 0 to transmit at 802.11ax MCS0 rate at power level = 10 dBm

`225 0 0` // Turn off the transmission of radio 0

Return data:

- Successful command: all-0 status byte (0x00000000)
- Failed command: non-0 status byte (0x00000001 to 0xFFFFFFFF)

8.14 Command 231: Enable uplink OFDMA Tx

This command enables or disables uplink OFDMA TX.

Syntax: `231 <Enable> <Qnum> <AXQ0_MU_Timer> <AID> <TxPwr>`

Where:

Table 14. Command parameters

Parameter	Description
`Enable`	Enable/disable Trigger frame response mode 0 = exit trigger frame response mode (default) 1 = enter trigger frame response mode
`Qnum`	Transmit queue number that holds the trigger-based response packets. Set to 1. 1 = trigger-based test (default)
`AXQ0_MU_Timer`	Arbitrary timer value to ensure SU packets are not transmitted. Units are in 8 ms. Set the value to be larger than the trigger frame interval. Suggest to set the value to 400 => 400 *8 = 3200 ms = 3.2 second.
`AID`	Station ID. Set the value to 5.
`TxPwr`	Transmit power in dBm

Examples

`231 1 1 400 5 9` // Enables trigger frame response testing, configuring station ID to 5 and forcing transmit power of the uplink packet to 9 dBm

`231 0` // Disables trigger frame response testing

Return data:

- Successful command: all-0 status byte (0x00000000)
- Failed command: non-0 status byte (0x00000001 to 0xFFFFFFFF)

8.15 Command 235 - Transmit OFDMA packet with configuration file

This command transmits OFDMA packets with a configuration file. The command emulates the packet-based transmission with a packet-to-packet gap specified by a configurable SFIS interval.

Syntax: `235 <enable> <mode> <count> <fileName> <powerLevel> <Burst Sifs in us> <pattern> <destination ADDR> <BSSID>`

Where:

Table 15. Command parameters

Parameter	Description
`enable`	Enable/disable transmit 0 = disable 1 = enable
`mode`	Trigger frame mode 0 = disable 1 = trigger based UL-OFDMA configuration 2 = standalone UL-OFDMA configuration
`count`	Number of packets to transmit -1 = continuous TX (default)
`fileName`	Trigger frame configuration file. For example *TF_Config_20MHz.txt* ***Note:*** *Trigger frame files are included in the manufacturing software package.*
`powerLevel`	Target set power in dBm
`Burst Sifs in us`	Burst SIFS gap. Minimum 10 µs, maximum 255 µs.
`pattern`	TX data pattern 0xB496DEB6 = default value for MFG testing
`destination ADDR`	destination BSSID If not set, all 0xF are used.
`BSSID`	BSSID od source If not set, all 0xF are used.

Examples

`235 1 2 -1 TF_Config_20MHz.txt` // Configure standalone OFDMA trigger frame

`235 1 1 -1 TF_Config_20MHz.txt` // Configure Trigger TF configuration for trigger-based UL-OFDMA

`235 0 0` // Turn off transmission

Return data:

- Successful command: all-0 status byte (0x00000000)
- Failed command: non-0 status byte (0x00000001 to 0xFFFFFFFF)

8.16 Command 18: Set Wi-Fi CW TX

This command is used to enable Wi-Fi continuous wave (CW) transmit mode.

Syntax: `18 <Radio_ID> <Enable>`

Where:

Table 16. Command parameters

Parameter	Description
`Radio_ID`	Radio identifier value set to 0.
`Enable`	Enable CW transmit mode 0 = disable 1 = enable

Example

`18 0 1` // Enables Wi-Fi CW TX

Return data:

- Successful command: all-0 status byte (0x00000000)
- Failed command: non-0 status byte (0x00000001 to 0xFFFFFFFF)

8.17 Command 32: Get the received packet count (Stop Rx FER test)

This command returns the counts for Rx, multi-cast and error packets, and clear the Rx packet counter. The command requires to set the radio identifier to 0.

Syntax: `32 <Radio_ID>`

Where:

Table 17. Command parameters

Parameter	Description
`Radio_ID`	Radio identifier set to 0

Example

`32 0` // Displays the received multi cast packet count

Return data: The received multi-cast packet count for the identified radio.

Return data example:

```
Enter option: 32 0
DutIf_GetRxPckt: 0x00000000
GetRxPckt:
  Rx Packet 1500
  Multi Cast 1500
  Err Count 0
```

8.18 Command 31: Clear the received packet count

This command clears the packet count register for the Wi-Fi radio. The receiver continuously counts the received multi-cast packets. The command requires to set the radio identifier to 0.

Syntax: `31 <Radio_ID>`

Where:

Table 18. Command parameters

Parameter	Description
`Radio_ID`	Radio identifier set to 0

Example

`31 0` // Clears the received packet count

Return data:

- Successful command: all-0 status byte (0x00000000)
- Failed command: non-0 status byte (0x00000001 to 0xFFFFFFFF)

8.19 Command 198: Start RSSI data collection

This command starts RSSI collection and checks that the DUT is configured to receive.

Syntax: `198 <radio_ID>`

Where:

Table 19. Command parameters

Parameter	Description
`radio_ID`	Radio identifier value set to 0

Example

`198 0` // starts RSSI data collection

Return data:

- Successful command: all-0 status byte (0x00000000)
- Failed command: non-0 status byte (0x00000001 to 0xFFFFFFFF)

Return data example:

```
Enter option: 198 0

Dev_WlanRSSI_Cal_Start
DutIf_WlanRSSI_Cal_Start: 0x00000000
Enter option:
```

8.20 Command 199: Stop RSSI data collection and report result

This command stops RSSI data collection and reports the results.

Syntax: `199 <Radio_ID>`

Where:

Table 20. Command parameters

Parameter	Description
`radio_ID`	Radio identifier value set to 0

Example

`199 0` // stop RSSI data collection and report the result

Return data:

- Successful command: all-0 status byte (0x00000000)
- Failed command: non-0 status byte (0x00000001 to 0xFFFFFFFF)

Return data example:

```
Enter option: 199 0
Rssi_packet_count =2158
Path_0_RSSI_Val:    0x00000D9A
Path_0_Nf_Val:    0x000009FA
>>Path_0_LNA_GAIN:    3(ELNA_HI_ILNA_HI)
>>Path_0_RSSI_Val:    -38.38 dBm
>>Path_0_Nf_Val:    -96.38 dBm
Enter option:
```

8.21 Command 44: Get/set the storage type

This command is used to get or set the storage type for calibration data.

Syntax: `44 <storage type>`

Where:

Table 21. Command parameters

Parameter	Description
`storage type`	0 = EEPROM (not supported) 1 = Configuration file (refer to the calibration data file *WlanCalData_ext.conf*) 2 = OTP memory

Examples

`44` // reads back the storage type

`44 1` // sets the storage type to the configuration file

8.22 Command 22: Load calibration data from the configuration file

This command loads the calibration data from *WlanCalData_ext.conf* configuration file to the manufacturing firmware.

The configuration file name is specified in the *SetUp.ini* file. Make sure that *WlanCalData_ext.conf* configuration file and *SetUp.ini* file are available in Labtool working directory. Verify that `NO_EEPROM` flag is set to **1** (file option) in the *SetUp.ini* file.

Syntax: `22`

Return data:

- Successful command: all-0 status byte (0x00000000)
- Failed command: non-0 status byte (0x00000001 to 0xFFFFFFFF)

8.23 Command 53: Write calibration data from text files to OTP or *.conf* file

This command is used to write the calibration data to NVRAM from the following files:

- *CalBtDataFile.txt*
- *CalWlanDataFile.txt*
- *PwrTble_Otp_Path0.txt*
- *Cal15_4DataFile.txt* (RW612 only)

The calibration text files must be stored in the Labtool working directory.

Before issuing command 53, make sure to configure the cal data storage option “NO_EEPROM” in *SetUp.ini* file as the storage type.

Syntax: `53`

Return data:

- Successful command: all-0 status byte (0x00000000)
- Failed command: non-0 status byte (0x00000001 to 0xFFFFFFFF)

8.24 Command 54: Get calibration data from OTP memory into text file

This command is used to copy the calibration data stored in the OTP memory to the text files:

- *CalWlanDataFile_upload.txt*
- *CalBtDataFile_upload.txt*
- *PwrTble_Otp_Path0_upload.txt*
- *Cal15_4DataFile.txt* (RW612 only)

Syntax: `54`

Return data:

- Successful command: all-0 status byte (0x00000000)
- Failed command: non-0 status byte (0x00000001 to 0xFFFFFFFF)

8.25 Command 101: Get RF control mode

This command is used to get the RF control mode.

Syntax: `101`

Example

`101` // Return the RF control mode

8.26 Command 102: Set RF control mode

This command is used to set the RF control mode.

Syntax: `102 <mode>`

Where:

Table 22. Command parameters

Parameter	Description
`mode`	RF control mode 0 = mode for normal operation 10 = RF calibration mode

Example

`102 10` // Set the control mode to RF calibration

Return data:

- Successful command: all-0 status byte (0x00000000)
- Failed command: non-0 status byte (0x00000001 to 0xFFFFFFFF)

8.27 Command 45: Read the MAC address from OTP memory

This command returns the Wi-Fi MAC addresse stored in the OTP memory. Set `Radio_ID` to 0.

Syntax: `45 <Radio_ID>`

Where:

Table 23. Command parameters

Parameter	Description
`Radio_ID`	Radio identifier set to 0

Example

`45 0` // Returns the stored MAC address

8.28 Command 46: Write MAC address to OTP memory

This command writes the MAC address into the OTP memory. Set `Radio_ID` to 0.

Syntax: `46 <Radio_ID> <MAC address>`

Where:

Table 24. Command parameters

Parameter	Description
`Radio_ID`	Radio identifier set to 0
`MAC address`	Mac address in xx.xx.xx.xx.xx.xx format

Example

`46 0 C0.95.DA.21.12.13` // Writes the WI-Fi MAC address “C0.95.DA.21.12.13” into the OTP memory

Return data:

- Successful command: all-0 status byte (0x00000000)
- Failed command: non-0 status byte (0x00000001 to 0xFFFFFFFF)

8.29 Command 95: Get RF crystal calibration offset

This command is used to get the crystal calibration offset value. The crystal calibration offset is an 8-bit value.

Syntax: `95`

Return data: 8-bit crystal calibration value

```
Enter option: 95
95

Dut_Bt_GetBtXtal_W909X: 0x00000000
XTal: 72
```

8.30 Command 96: Set RF crystal calibration offset

This command sets the 8-bit crystal calibration offset. The value is used to adjust the frequency offset and improve the frequency accurracy.

Syntax: `96 <RF crystal offset value>`

Where:

Table 25. Command parameters

Parameter	Description
`RF crystal offset value`	Offset value for the RF crystal calibration in hex format . The offset range is 0x00 to 0xFF.

Example

`96 A0` // Sets the RF crystal calibration offset value to 0xA0

Return data:

- Successful command: all-0 status byte (0x00000000)
- Failed command: non-0 status byte (0x00000001 to 0xFFFFFFFF)

8.31 Command 120: Get the thermal sensor reading

This command returns the on-chip temperature sensor value measured during calibration, in decimal. The command requires to set `<sensor index>` value as 1.

Note: *For Labtool utility versions 2.0.0.2 or later, the command 120 readback displays the signed temperature as a decimal number. In earlier versions of Labtool utility, the command readback displays the temperature value in hexadecimal.*

Syntax: `120 <sensor index>`

Where:

Table 26. Command parameters

Parameter	Description
`Sensor index`	0 = CAU 1 = Radio0_PathA

Return code: None

Example

`120 1` // Reads the temperature sensor value

Return data: Celsius temperature sensor reading in decimal

Return data example:

```
Enter option: 120 1
DutIf_GetThermalSensorReading_W909X: 0x00000000
SensorReading: 25
```

The temperature reading converted to Celsius is 27 degrees Celsius.

8.32 Command 121: Get the power detector reading offset

This command reads the power detector offset value and represents the measured power detector value at power index 5 during calibration. This parameter is an 8-bit two's complement number converted in decimal. The command requires to set `<Radio_ID>` and `<Tx_Path>` values as 0.

Syntax: `121 <Radio_ID> <Tx_Path> <Power reading>`

Where:

Table 27. Command parameters

Parameter	Description
`Radio_ID`	Radio identifier set to 0
`Tx_Path`	TX path set to 0
`Power reading`	TX power value in dBm from RF Tester. The power resolution supports up to two decimal places.

Example

`121 0 0 11.38` // Gets the power detector offset value when TX power is 11.38 dBm

Return data: power detector offset value

Return data example:

```
Enter option: 121 0 0 11.38
DutIf_GetPwrDetOff: 0x00000000
PwrDet_Offset_in_dB: 1.745000
PwrDet_Offset_Code: 0x1B
```

8.33 Command 99: Exit Labtool

First run command `99` once to exit the Wi-Fi menu.

Run command `99` a second time to exit Labtool and close the command prompt window.

9 Labtool command sequence examples to test Wi-Fi radio

This section provides examples of Labtool command test sequence for various Wi-Fi transmit and receive RF performance tests.

9.1 Labtool command sequence for 802.11b TX/RX

9.1.1 Labtool command sequence for 802.11b TX

- TX on channel 1 at 8 dBm target power with CCK-11 Mbps rate in 2.4 GHz band

```
6 11          // Set radio mode index “11” to test 2.4 GHz band
12 0 1        // Set to 2.4 GHz channel 1
35 0 1 4 8    // 2.4 GHz Tx at 11 Mbps at 8 dBm power level
35 0 0        // Stop Tx
```

9.1.2 Labtool command sequence for 802.11b RX

- RX on channel 1 in 2.4 GHz band

```
6 11           // Set  Radio Mode Index “11” to test 2.4 GHz band
12 0 1         // Set to 2.4 GHz channel 1
31 0           // Clear the received packet count
-Transmit Wi-Fi packets from tester
32 0           // Get Rx packet count and then clear the Rx packet counter
```

9.2 Labtool command sequence for 802.11g TX/RX

9.2.1 Labtool command sequence for 802.11g TX

- TX on channel 13 at 8 dBm with OFDM 54 Mbps data rate in 2.4 GHz band

```
6 11              //Set radio mode index “11” to test 2.4 GHz band
12 0 13           // Set to 2.4 GHz band channel 13
35 0 1 13 8       // Tx at 54 Mbps at 8 dBm power level
35 0 0            // Stop Tx
```

9.2.2 Labtool command sequence for 802.11g RX

- RX on channel 1 in 2.4 GHz band

```
6 11              // Set radio mode index “11” to test 2.4 GHz
12 0 1            // Set to 2.4 GHz channel 1
31 0              // Clear the received packet count
-Transmit 802.11g packets from the tester on channel 1
32 0              // Get Rx Packet Count and then clear the Rx packet counter
```

9.3 Labtool command sequence for 802.11a TX/RX

9.3.1 Labtool command sequence for 802.11a TX

- TX on channel 100 at 8 dBm with OFDM 54 Mbps data rate in 5 GHz band

```
6 3              // Set radio mode index “3” to test 5 GHz
12 0 100         // Set to channel 100
35 0 1 13 8      // Tx at 54 Mbps at 8 dBm power level
35 0 0           // Stop Tx
```

9.3.2 Labtool command sequence for 802.11a RX

- RX channel 100 in 5 GHz band

```
6 3               // Set radio mode index “3” to test 5 GHz
12 0 100          // Set to channel 100
31 0              // Clear the received packet count
-Transmit 802.11a packets from the tester on channel 100
32 0              // Get Rx Packet Count and then clear the Rx packet counter
```

9.4 Labtool command sequences for 802.11n TX/RX

9.4.1 Labtool command sequence for 802.11n TX, 2.4 GHz

- TX on channel 1 at 8 dBm with MCS7 data rate in 2.4 GHz band

```
6 11            // Set radio mode index “11” to test 2.4 GHz band
12 0 1          // Set to channel 1
35 0 1 22 8     // Tx at MCS7 at 8 dBm power level
35 0 0          // Stop Tx
```

9.4.2 Labtool command sequence for 802.11n TX, 5 GHz

- TX on channel 100 at 8 dBm with MCS7 data rate in 5 GHz band

```
6 3               // Set radio mode index “3” to test 5 GHz band
12 0 100          // Set channel 100
35 0 1 22 8       // Tx at MCS7,8 dBm power level
35 0 0            // Stop Tx
```

9.4.3 Labtool command sequence for 802.11n RX, 2.4 GHz

- RX on channel 6 in 2.4 GHz band

```
6 11     // Set radio mode index “11” to test 2.4 GHz band
12 0 6   // Set to channel 6
31 0     // Clear the received packet count
-Transmit 802.11n packets from the tester on channel 6
32 0    // Get Rx Packet Count and then clear the Rx packet counter
```

9.5 Labtool command sequences for 802.11ac TX/RX

9.5.1 Labtool command sequence for 802.11ac TX, 5 GHz

- TX on channel 100 at 8 dBm with VHT MCS8 data rate in 5 GHz band

```
6 3                 // Set radio mode index “3” to test 5 GHz band
12 0 100            // Set to channel 100
35 0 1 1108 8       // Tx at VHT MCS8 at 8 dBm power level.
35 0 0              // Stop Tx
```

9.6 Labtool command sequences for 802.11ax TX/RX

9.6.1 Labtool command sequence for 802.11ax TX, 2.4 GHz

- TX on channel 6 at 8 dBm with HE20 MCS8 data rate in 2.4 GHz band

```
6 11              // Set radio mode index “11” to test 2.4 GHz band
12 0 6            // Set to channel  6
35 0 1 2108 8     // Tx at HE20 MCS8 at 8 dBm power level
35 0 0            // Stop Tx
```

9.6.2 Labtool command sequence for 802.11ax TX, 5 GHz

- TX on channel 36 at 8 dBm with HE20 MCS7 rate in 5 GHz band

```
6 3           // Set radio mode index “3” to test 5 GHz
12 0 36       // Set channel 36
35 0 1 2107 8 // Tx at HE20 MCS7 at 8 dBm power level
35 0 0        // Stop Tx
```

9.6.3 Labtool command sequence for 802.11ax RX, 2.4 GHz

- RX on channel 6 in 2.4 GHz band

```
6 11             // Set radio mode index “11” to test 2.4 GHz band
12 0 6           // Set to channel 6
31 0             // Clear the received packet count
-Transmit 802.11ax packets from the tester on channel 6
32 0             // Get Rx Packet Count and then clear the Rx packet counter
```

9.6.4 Labtool command sequence for 802.11ax RX, 5 GHz

- RX on channel 36 in 5 GHz band

```
6 3      // Set radio mode index “3” to test 5 GHz band
12 0 36  // Set to channel 36
31 0     // Clear the received packet count
-Transmit 802.11ax packets from the tester on channel 36
32 0     // Get Rx Packet Count and then clear the Rx packet
```

9.7 Labtool command sequence for Wi-Fi RSSI measurement

Prior to performing RSSI measurement tests, we recommend to first correct the crystal frequency error. For the crystal frequency calibration procedure, refer to [1]. The following is an example of Labtool command sequence for Wi-Fi RSSI measurement.

- Wi-Fi RSSI measurement on channel 36 in 5 GHz band, 20 MHz bandwidth with 6 Mbps data rate

```
6 3     // Set radio mode index “3” to test 5 GHz band
112 0 0 // Set to 20 MHz bandwidth
12 0 36 // Set to channel 36
198 0   // Start RSSI Data Collection
-Transmit Wi-Fi packets from tester with -50dBm power level OFDM-6 signal
199 0   // Stop RSSI data collection and report result including the count of packet
 received, RSSI and noise floor
```

9.8 Labtool command sequences for Wi-Fi transmit power calibration

- Labtool command sequence for temperature sensor reading for 2.4 GHz band on channel 6:

```
6 11          // Set the band to 2.4GHz
112 0 0       // Set the bandwidth to 20MHz
12 0 6        // Set channel to 6
102 10        // Set the device into power calibration mode
35 0 1 22 0   // Start Tx at MCS7 rate using power index 0
120 1         // Read the temperature sensor value
```

- Example of power detector offset measurement for 2.4 GHz band on channel 6:

```
6 11         // Set the band to 2.4GHz
112 0 0       // Set the bandwidth to 20MHz
12 0 6        // Set channel to 6
35 0 1 22 5   // Start Tx at MCS7 rate using power index 5
<Measure the output power reading. Note the value (assume 11.38 dBm for this case)>
121 0 0 11.38  // Get power detector offset value; input measured power level as last
 parameter
<Power Detector Offset Value displayed in hexadecimal>
35 0 0    // Stop TX
```

9.9 Labtool command sequences for Wi-Fi transmit with HE-EXT-SU

- Labtool command sequence for Wi-Fi transmit with 2.4 GHz 802.11ax HE-EXT-SU (242 tone)

```
6 11      // Set the band to 2.4GHz
112 0 0   // Set the bandwidth to 20MHz
12 0 6    // Set the channel to 6
225 0 1 2101 4 1 20 200 b496deb6 1 3 1 0 0 0 1 0 // TX with 11ax HE-EXT-SU packet
225 0 0   // Stop TX
```

- Labtool command sequence for Wi-Fi transmit with 2.4 GHz 802.11ax HE-ER-SU (106 tone)

```
6 11  // Set the band to 2.4GHz
112 0 0  // Set the bandwidth to 20MHz
12 0 6  // Set the channel to 6
225 0 1 2100 10 1 20 200 B496DEB6 1 3 1 0 0 0 1 1  // TX with 11ax HE-ER-SU packet with
 106 tones
225 0 0  // Stop TX
```

9.10 Labtool command sequence for Wi-Fi continuous wave (CW) transmit

TX on channel 36 at 10 dBm continuous wave.

```
6 3          // Set Wi-Fi in 5GHz band
112 0 0      // Set channel bandwidth to 20MHz
12 0 36      // Set the Wi-Fi radio to 5GHz channel 36
35 0 1 15 10 // Set the TX power to 10dBm
35 0 0       // Stop TX
18 0 1       // Enable Wi-Fi CW TX
18 0 0       // Disable Wi-Fi CW TX
```

10 List of supported data rates and channels for Wi-Fi

This section provides the lists of supported data rates and RF channels.

10.1 Data rates

Table 28 shows 802.11n/a/g/b data rate IDs and data rates.

Table 28. 802.11n/a/g/b data rate ID

Data rate ID	Data rate	Data rate ID	Data rate	Data rate ID	Data rate
1	1 Mbps	6	6 Mbps	15	HT_MCS 0
2	2 Mbps	7	9 Mbps	16	HT_MCS 1
3	5.5 Mbps	8	12 Mbps	17	HT_MCS 2
4	11 Mbps	9	18 Mbps	18	HT_MCS 3
5	Reserved	10	24 Mbps	19	HT_MCS 4
		11	36 Mbps	20	HT_MCS 5
		12	48 Mbps	21	HT_MCS 6
		13	54 Mbps	22	HT_MCS 7
		14	Reserved		

Table 29 shows 802.11ac/802.11ax data rate IDs and data rates.

Table 29. 802.11ac/802.11ax data rate ID

Rate number format : (XYRR) X : 1 - 11ac VHT MCS rates, 2 - 11ax HE MCS rates Y: Number of streams. 1 - SS1 RR : MCS rate number	
Data rate ID **XYRR**	**Data rate**
802.11ac VHT MCS rates[1]	
1100	VHT_SS1_MCS0
1101	VHT_SS1_MCS1
1102	VHT_SS1_MCS2
1103	VHT_SS1_MCS3
1104	VHT_SS1_MCS4
1105	VHT_SS1_MCS5
1106	VHT_SS1_MCS6
1107	VHT_SS1_MCS7
1108	VHT_SS1_MCS8
1109	VHT_SS1_MCS9
802.11ax HE MCS rates	
2100	HE_SS1_MCS0
2101	HE_SS1_MCS1
2102	HE_SS1_MCS2
2103	HE_SS1_MCS3
2104	HE_SS1_MCS4
2105	HE_SS1_MCS5
2106	HE_SS1_MCS6
2107	HE_SS1_MCS7
2108	HE_SS1_MCS8
2109	HE_SS1_MCS9

[1] Not all VHT rates are available for all the bandwidths and Tx antenna configurations. The error code `0xE` means that the rate and hardware combination is not supported.

10.2 RF channels

For the list of supported 2.4 GHz and 5 GHz Wi-Fi channels, refer to [2] and [3].

11 Testing 802.11ax uplink-OFDMA transmit

This section shows how to run UL-OFDMA transmit test by using the manufacturing software and Labtool commands.

11.1 Test setup

To test trigger-based uplink OFDMA, you need two boards and two instances of Labtool application running on the Windows PC:

- One board is the golden unit that transmits the trigger frame. The second board is the DUT that transmits the uplink OFDMA signal.
- One Labtool window is used for the golden unit, and the other Labtool window is used for the DUT.

Figure 7 shows the standard setup for the UL-OFDMA test in the Lab. The Horn antenna receives the UL-OFDMA signal from the DUT and analyzes it.

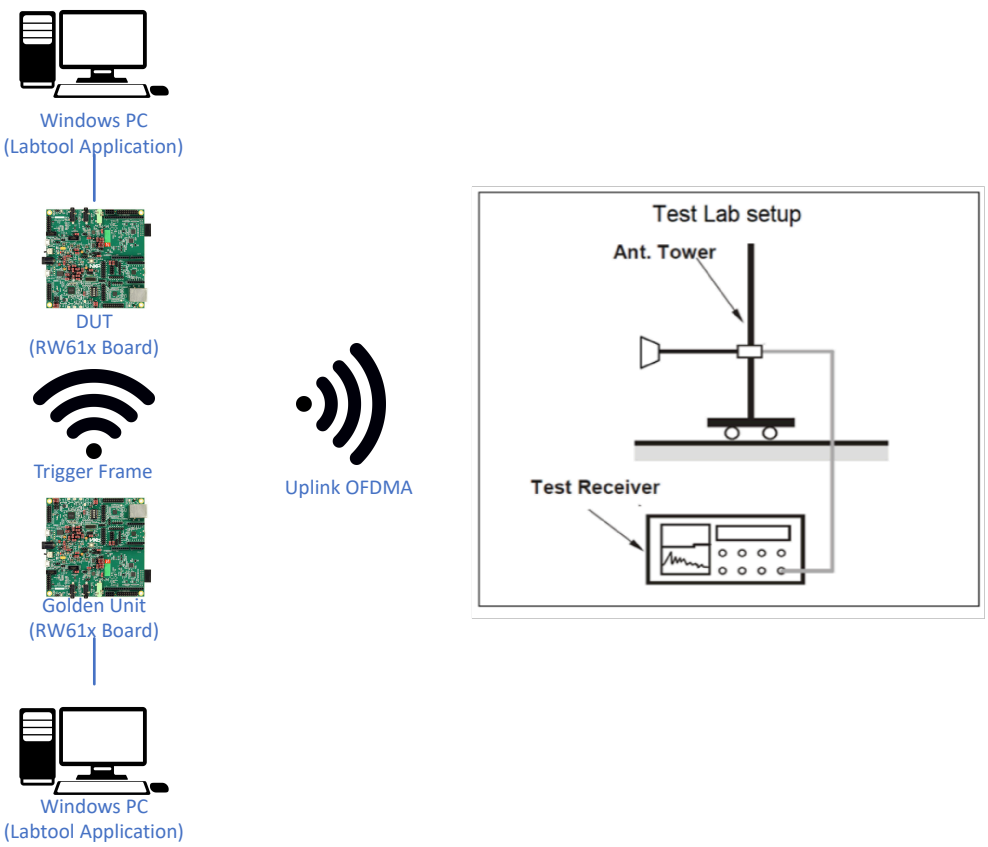

Figure 7. Test setup

11.2 Standalone uplink-OFDMA test

This section describes how to configure an uplink-OFDMA test with one board.

The DUT receives trigger frames from the *.txt* file and responds by transmitting UL-OFDMA.

The RU Index and MCS data rate of the UL-OFDMA transmission are based on the received trigger frame.

- Open the Labtool window of the DUT and enter the commands shown below.

This example shows how to enable UL-OFDMA transmission under the following conditions:

- 5 GHz
- Channel 36
- 20 MHz channel bandwidth

Command sequence to connect the DUT directly to RF tester:

```
6 3   // set radio mode for 5GHz band
112 0 0  // set 20MHz bandwidth
12 0 36  // set channel 36
235 1 2 -1 TF_Config_20MHz.txt //Configure trigger frame settings using config file
225 0 1 2100 9 0 0 40 abababab 3 1 -1 0 0 0 0 -1 ff.ff.ff.ff.ff.ff -1 // Start sending
 trigger frame. The power level is set to 9 dBm and the data rate is set to MCS0 (2100)
225 0   // stop sending trigger frame
235 0   // disable UL-OFDMA configuration
```

Note: *The standalone config files is included in the software release under Labool directory.*

11.3 Edit the configuration file

The configuration file (*TF_Config_ruX_mcsY.txt*) is used to set the data rate and RU are parameters for the golden unit. The file is stored in *Labtool* directory.

Note: *Contact your local NXP FAE if you don't have the file.*

The configuration file has the following format:

```
TrigUserInfoField=5;0;<RU>;1;<MCS data rate>;0;0;80
```

Where:

Table 30. Command parameters

Parameter	Description
`RU`	RU index. Refer to Figure 8 for the values of RU index
`MCS data rate`	MCS data rate

Example

`TrigUserInfoField=5;0;3;1;7;0;0;80` // RU is set to 3 and MCS is set to 7

Figure 8 shows the RU index values for the 20 MHz bandwidth.

Bandwidth (MHz)	20								
RU-Index	0	1	2	3	4	5	6	7	8
RU-Tone	26	26	26	26	26	26	26	26	26
RU-Index	37		38			39		40	
RU-Tone	52		52			52		52	
RU-Index	53					54			
RU-Tone	106					106			
RU-Index	61								
RU-Tone	242								

Figure 8. RU index values for 20 MHz bandwidth

11.4 Configure the golden unit

This section shows how to use the configuration file and Labtool to set the golden unit to transmit a trigger frame.

Step 1 - Edit the configuration.

Set RU index and MCS data rate as detailed in Section 11.3.

Step 2 - Start Labtool on the golden unit.

- Open the Labtool application.
- Use Labtool command `6` to set the transmit path. See Section 8.5 for the command definition.
- Use Labtool command `112` to set the channel bandwidth. See Section 8.11 for the command definition.
- Use Labtool command `12` to set the RF channel. See Section 8.9 for the command definition.
- Use Labtool command `235` to use the configuration file as the trigger frame resource. See Section 8.15 for the command definition.

```
235 1 1 -1 <configuration file name>.txt
```

Step 3 - Transmit packets continuously.

Use the Labtool command `225` to continuously transmit packets. See Section 8.13 for the command definition.

Note: *For UL-OFDMA test, only the parameters* `<Rate>`, `<Power Level>` *and* `<Burst Sifs>` *can be changed for the target power, data rate and packet spacing. The other bits are fixed and must not be changed.*

Example

`225 0 1 2100 10 0 0 400 abababab 3 1 -1 0 0 0 0 -1 ff.ff.ff.ff.ff.ff -1` // Set the golden unit to transmit a trigger frame with a data rate of 2100, power level of 10 dBm, and Burst Sifs of 0 µs

11.5 Configure the DUT

In this section, the DUT is configured to transmit an UL-OFDMA signal.

Step 1 - Start Labtool.

- Go to the location where Labtool executable is installed.
- Start Labtool as explained in Section 7.1.
- Use Labtool command `6` to set the transmit path. See Section 8.5 for the command definition.
- Use Labtool command `112` to set the channel bandwidth. See Section 8.11 for the command definition.
- Use Labtool command `12` to set the channel. See Section 8.9 for the command definition.

Step 2 - Enable UL-OFDMA TX.

Use the Labtool command `231` to transmit an UL-OFDMA signal. See Section 8.14 for the command definition.

Example

`231 1 1 400 5 10` // Enables UL-OFDMA TX, configuring station ID to 5 and forcing transmit power of the uplink packet to 10 dBm

Step 3 - Disable UL-OFDMA TX testing when the test is finished.

Issue `231` Labtool command to disable the trigger-based response testing.

```
231 0
```

Command sequence example :

The following is an example of command sequence to test UL-OFDMA TX:

```
6 1 0
112 0 0
12 0 36
231 1 1 400 5 10
231 0 //Stop
```

Figure 9 shows an example of UL-OFDMA signal transmission captured on a VSA.

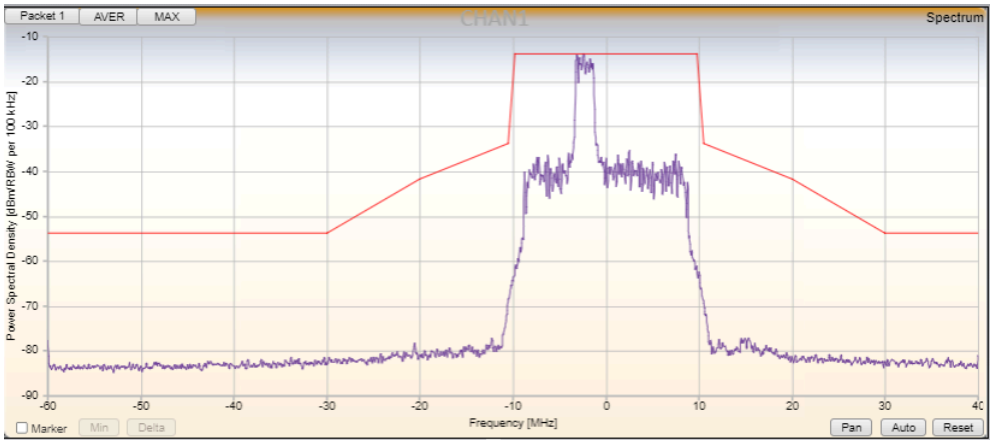

Figure 9. Recording of UL-OFDMA signal

11.6 Command sequence for Wi-Fi 802.11ax trigger-based UL-OFDMA test

- Labtool command sequence to enable sending a trigger frame

```
6 3        // set radio mode for 5GHz band
112 0 0    //set 20MHz bandwidth
12 0 36     // Set channel 36
235 1 1 -1 TF_Config_20MHz.txt // generate trigger frame using config file
225 0 1 2100 10 0 0 400 abababab 3 1 -1 0 0 0 0 -1 ff.ff.ff.ff.ff.ff -1 // Start sending
 trigger frame
225 0    // Stop sending trigger frame
235 0    // disable UL-OFDMA configuration
```

- Command sequence to configure the trigger-based transmit and measure the trigger frame response on VSA

```
6 3       // set radio mode for 5GHz band
112 0 0   //set 20MHz bandwidth
12 36      // Set channel 36
231 1 1 400 5 x   // Enable UL-OFDMA with x TX power in dBm
231 0   / disable UL-OFDMA TX
```

12 Bluetooth LE Labtool commands

This section details the supported Labtool commands for Bluetooth LE radio.

12.1 Labtool command list for Bluetooth LE

Table 31. Bluetooth LE Labtool command list

Command description
Commands for general operation
Command 80: Reset Bluetooth
Command 88: Get the firmware and Labtool versions
Command 44: Get/set the storage type
Command 234: Reload Bluetooth calibration data
Commands for Bluetooth transmit
Command 15: Get Bluetooth TX power level value
Command 16: Set Bluetooth TX power level value
Command 18: Set Bluetooth CW TX
Command 113: Get Bluetooth TX power control class
Command 114: Set Bluetooth TX power control class
Command 115: Get Bluetooth power control status
Command 116: Set Bluetooth power control
Commands for Bluetooth LE transmit
Command 133: Set Bluetooth LE TX power
Command 125: Set Bluetooth LE TX test
Command 126: Get Bluetooth LE TX test packet count
Commands for Bluetooth LE receive
Command 127: Set Bluetooth LE RX test
Command 128: Get Bluetooth LE test packet error
Command 129: Stop Bluetooth LE test
Command for Bluetooth device address
Command 45: Read Bluetooth device (BD) address
Command to exit Labtool
Command 99: Exit Bluetooth menu

12.2 Command 80: Reset Bluetooth

This command resets the Bluetooth RF block.

Example

```
80 0 // Resets the Bluetooth RF block
```

12.3 Command 88: Get the firmware and Labtool versions

This command returns the firmware and Labtool versions.

Syntax: `88`

Example

`88` // Get Bluetooth (RW610) or Narrowband (Bluetooth + 802.15.4) (RW612) firmware version and Labtool version

Return data: Bluetooth (RW610) or Narrowband (Bluetooth + 802.15.4) (RW612) firmware version and Labtool version

12.4 Command 44: Get/set the storage type

This command is used to get or set the storage type for the calibration data.

Syntax: `44 <storage type>`

Where:

Table 32. Command parameters

Parameter	Description
`storage_type`	Type of storage 1 = configuration file (refer to the calibration data file *BtCalData_ex.conf*) 2 = OTP memory

Examples

`44` // reads back the storage type

`44 1` // sets the storage type to the configuration file

12.5 Command 234: Reload Bluetooth calibration data

This command reloads Bluetooth calibration data from *BTCalData_ext.conf* file. This command is also used to update the Bluetooth calibration data in *BTCalData_ext.conf* file.

Syntax: `234`

Example

`234` // Reload Bluetooth calibration data

Return data:

- Successful command: all-0 status byte (0x00000000)
- Failed command: non-0 status byte (0x00000001 to 0xFFFFFFFF)

12.6 Command 15: Get Bluetooth TX power level value

This command returns Bluetooth TX power level value in 0.5 dB steps.

Syntax: `15`

Return data:Target Bluetooth TX power level value in dBm

12.7 Command 16: Set Bluetooth TX power level value

This command sets Bluetooth TX power level value.

Syntax: `16 <pwr> <IsEDR>`

Where:

Table 33. Command parameters

Parameter	Description
`pwr`	Transmit power level in 0.5 dB steps.
`IsEDR`	Flag to specify the data rate. Set to 0. 0 = Bluetooth LE (default) 1 = EDR (not supported)

Example

`16 2.5 0` // Sets Bluetooth LE power as 2.5 dBm

Return data:

- Successful command: all-0 status byte (0x00000000)
- Failed command: non-0 status byte (0x00000001 to 0xFFFFFFFF)

12.8 Command 18: Set Bluetooth CW TX

This command is used to enable Bluetooth continuous wave (CW) transmit mode.

Syntax: `18 <Enable> <Channel option> <Channel No>`

Where:

Table 34. Command parameters

Parameter	Description
`Enable`	Enable CW transmit mode 0 = disable 1 = enable
`Channel option`	Channel option. Set to 1. 0 = Bluetooth Classic (not supported) 1 = Bluetooth LE
`Channel No`	Channel number in the range of 0 to 39 (2402 MHz to 2480 MHz).

Example

`18 1 1 0` // Enable CW TX for Bluetooth LE on channel 0

Return data:

- Successful command: all-0 status byte (0x00000000)
- Failed command: non-0 status byte (0x00000001 to 0xFFFFFFFF)

12.9 Command 113: Get Bluetooth TX power control class

This command returns the setting for Bluetooth TX power control class.

Syntax: `113`

Return data: `mode`

Where:

Table 35. Command parameters

Parameter	Description
`mode`	Value for TX power control class 0 = Class 2 (default) 1 = Class 1.5 3 = Class 1

12.10 Command 114: Set Bluetooth TX power control class

This command sets Bluetooth TX power control class.

Syntax: `114 <mode>`

Where:

Table 36. Command parameters

Parameter	Description
`mode`	TX power control class mode. 0 = Class 2 (with maximum output power of 4 dBm) 1 = Class 1.5 (with maximum output power of 10 dBm) 3 = Class 1 (with maximum output power of 15 dBm)

Example

`114 0` // Sets Bluetooth TX power control class to Class 2

Return data:

- Successful command: all-0 status byte (0x00000000)
- Failed command: non-0 status byte (0x00000001 to 0xFFFFFFFF)

12.11 Command 115: Get Bluetooth power control status

This command returns Bluetooth power control status.

Syntax: `115`

Return data: mode

Where:

Table 37. Command parameters

Parameter	Description
`mode`	Status of Bluetooth power control. Set to 0 0 = enable (default) 1 = disable

12.12 Command 116: Set Bluetooth power control

This command sets Bluetooth power control.

Syntax: `116 <mode>`

Where:

Table 38. Command parameters

Parameter	Description
`mode`	Bluetooth power control mode. Set to 0. 0 = enable (default) 1 = disable

Example

`116 1` // Disable Bluetooth power control

Return data:

- Successful command: all-0 status byte (0x00000000)
- Failed command: non-0 status byte (0x00000001 to 0xFFFFFFFF)

12.13 Command 133: Set Bluetooth LE TX power

This command sets the power level for Bluetooth LE TX test.

Syntax: `133 <pwr> <FEOffsetInDB>`

Where:

Table 39. Command parameters

Parameter	Description
`pwr`	TX power level (dBm) - Default is 0. Step size = 1 dB
`FEOffsetInDB`	Bluetooth front-end loss in dB. Default is 0.[1] Step size = 1 dB

[1] Only takes effect in OTP mode.

Example

`133 6 0` // Sets Bluetooth LE power level as 6 dBm with 0 dB front-end loss

Return data:

- Successful command: all-0 status byte (0x00000000)
- Failed command: non-0 status byte (0x00000001 to 0xFFFFFFFF)

12.14 Command 125: Set Bluetooth LE TX test

This command sets the Bluetooth Low Energy (LE) TX test.

Syntax: `125 <FreqIndex> <Len> <Pattern> <Phy>`

Where:

Table 40. Command parameters

Parameter	Description
`FreqIndex`	Set the channel number for Bluetooth LE TX 0x00 = 2402 MHz 0x01 = 2404 MHz ... 0x39 = 2480 MHz
`Len`	Payload data length Range: 0 to 37 bytes. Default is set to 37 bytes.
`Pattern`	Payload data pattern 0x0 = PN9 0x1 = 0xF0 0x2 = 0xAA 0x3 = PN15 0x4 = all 1 0x5 = all 0 0x6 = 0x0F 0x7 = 0x55
`Phy`	Data rate used for testing 1 = 1 Mbps 2 = 2 Mbps 3 = LR8 (coded PHY) 4 = LR2 (coded PHY)

Example

`125 0 37 0 1` // Set channel 0 (2402 MHz) with payload length of 37 bytes, payload data pattern of PN9, and 1 Mbps data rate

Return data:

- Successful command: all-0 status byte (0x00000000)
- Failed command: non-0 status byte (0x00000001 to 0xFFFFFFFF)

12.15 Command 126: Get Bluetooth LE TX test packet count

This command gets the TX test packet count after the Bluetooth LE TX test.

Syntax: `126`

Return data: TX packet count

12.16 Command 127: Set Bluetooth LE RX test

This command sets Bluetooth receiver for Bluetooth LE RX test.

Syntax: `127 <FreqIndex> <Phy>`

Where:

Table 41. Command parameters

Parameter	Description
`FreqIndex`	Set the channel number for Bluetooth LE RX 0x00 = 2402 MHz 0x01 = 2404 MHz ... 0x39 = 2480 MHz
`Phy`	Data rate used for testing 1 = 1 Mbps 2 = 2 Mbps 3 = LR2 and LR8 (coded PHY)

Example

`127 1 1` // Sets Bluetooth LE RX test on channel 1 (2404 MHz) and 1 Mbps data rate

Return data:

- Successful command: all-0 status byte (0x00000000)
- Failed command: non-0 status byte (0x00000001 to 0xFFFFFFFF)

12.17 Command 128: Get Bluetooth LE test packet error

This command gets the packet error of Bluetooth LE Rx test.

Syntax: `128`

Return data: packet error of Bluetooth LE Rx test

12.18 Command 129: Stop Bluetooth LE test

This command stops the Bluetooth LE Tx/Rx test. It is always used after the Bluetooth LE Tx/Rx test (Section 12.14 "Command 125: Set Bluetooth LE TX test" and Section 12.16 "Command 127: Set Bluetooth LE RX test", respectively).

Syntax: `129`

Return data:

- Successful command: all-0 status byte (0x00000000)
- Failed command: non-0 status byte (0x00000001 to 0xFFFFFFFF)

12.19 Command 45: Read Bluetooth device (BD) address

This command gets the Bluetooth device address from the OTP or *.conf* file.

Syntax: `45`

Return data: Bluetooth device address in xx-xx-xx-xx-xx-xx format

12.20 Command 99: Exit Bluetooth menu

Issue command 99 once to exit from Bluetooth Labtool menu.

Issue command 99 a second time to exit from Labtool and close the command prompt window.

13 Labtool command sequence examples to test Bluetooth LE TX/RX

13.1 Bluetooth LE transmitter

- Bluetooth LE TX on channel 39 at 10 dBm with 1 Mbps data rate

```
80          // Reset the Bluetooth RF block
133 10 0    // Set 10 dBm target output power and 0 dB front end loss
125 39 37 0 1  // Tx on channel 39, packet length: 37, packet pattern: PN9, and 1 Mbps
 data rate
129         // Stop Tx
126         // Review Tx packets count
```

- Bluetooth LE TX on channel 0 at 10 dBm with LR8 data rate

```
80        // Reset the Bluetooth RF block
133 10 0  // Set 10 dBm target output power and 0 dB front end loss
125 0 37 0 3 // Tx on channel 0, packet length: 37, packet pattern: PN9, and LR8 data
 rate
129       // Stop Tx
126       // Review Tx packets count
```

- Bluetooth LE TX on channel 20 at 12 dBm with LR2 data rate

```
80         // Reset the Bluetooth RF block
133 12 0   // Set 12 dBm target output power and 0 dB front end loss
125 20 37 0 4 // Tx on channel 20, packet length: 37, packet pattern: PN9, LR2 data rate
129        // Stop Tx
126        // Review Tx packets count
```

- Bluetooth LE CW TX on channel 0 at 10 dBm

```
80        //Reset the Bluetooth RF block
116 1     //Disable BTU power control
16 10     //Set 10dBm target output power
18 1 1 0  //Enable Bluetooth LE CW Tx on channel 0
18 0      //Disable Bluetooth LE CW Tx
```

13.2 Bluetooth LE receiver

- Bluetooth LE RX on channel 39 with 1 Mbps data rate

```
80          // Reset the Bluetooth RF block
127 39 1    // Set to channel 39 and 1 Mbps
-Configure BT tester to transmit x number of packets
129         // Stop the Rx test and return total Rx Packets count
128         // Check Rx Test Error Report
```

- Bluetooth LE RX on channel 39 with 2 Mbps data rate

```
80          // Reset the Bluetooth RF block
127 39 2    // Set to channel 39 and 2 Mbps
-Configure BT tester to transmit x number of packets
129         // Stop the Rx test and return total Rx Packets count
128         // Check Rx Test Error Report
```

- Bluetooth LE RX on channel 39 with LR2 data rate

```
80          // Reset the Bluetooth RF block
127 39 3    // Set to channel 39 and LR2 data rate
-Configure BT tester to transmit x number of packets
129         // Stop the Rx test and return total Rx Packets count
128         // Check Rx Test Error Report
```

14 List of supported channels for Bluetooth LE radio

The Bluetooth LE radio operates in the 2.4 GHz ISM band at 2400-2483.5 MHz. The Bluetooth LE radio uses 40 RF channels. The center frequency value for these channels is calculated per:

2402 + (k x 2) MHz, where k = 0, ..., 39

Table 42. Operating frequency bands

Regulatory range	RF channels
2400-2483.5 MHz	*f = 2402 + (k x 2)* MHz; k = 0, ..., 39

Table 43. Channel frequencies

Channel number	Frequency (MHz)	Channel number	Frequency (MHz)
0	2402	19	2440
1	2404	20	2442
2	2406	21	2444
3	2408	22	2446
4	2410	23	2448
5	2412	24	2450
6	2414	25	2452
7	2416	26	2454
8	2418	27	2456
9	2420	28	2458
10	2422	29	2460
11	2424	30	2462
12	2426	31	2464
13	2428	32	2466
14	2430	33	2468
15	2432	34	2470
16	2434	35	2472
17	2436	36	2474
18	2438	37	2476
—	—	38	2478
—	—	39	2480

15 802.15.4 Labtool commands

This section details the supported Labtool commands for 802.15.4 radio (RW612 only).

15.1 Labtool command list for 802.15.4 radio

Table 44. Labtool command list for 802.15.4 radio

Command description
Commands for general operation
Command 11: Get the 802.15.4 channel
Command 12: Set the 802.15.4 channel
Command 80: Reset 802.15.4 radio
Commands for 802.15.4 radio transmit
Command 15: Get 802.15.4 TX power level value
Command 16: Set 802.15.4 TX power value
Command 17: Continuous modulated TX
Command 18: Continuous unmodulated TX
Command 20: Get 802.15.4 TX payload size
Command 21: Set 802.15.4 TX packet payload size
Command 23: Get TX power limit
Command 24: Set TX power limit
Command 33: 802.15.4 TX burst packet
Command 35: Set 802.15.4 TX duty cycle
Command 59: Set PHY TX test packet info
Commands for 802.15.4 radio receive
Command 31: Get 802.15.4 RX test result
Command 32: Start 802.15.4 RX test
Commands for MAC address
Command 45: Get EUI 64 MAC address
Command 46: Set EUI 64 MAC address
Command to exit Labtool
Command 99: Exit 802.15.4 menu

15.2 Command 11: Get the 802.15.4 channel

This command returns the 802.15.4 channel.

Syntax: `11`

Example

`11` // Get the 802.15.4 channel

Return data: Currently set channel

15.3 Command 12: Set the 802.15.4 channel

This command sets the channel of 802.15.4 radio.

Syntax: `12 <Channel number>`

Where:

Table 45. Command parameters

Parameter	Description
`Channel number`	Channels 11 to 26 are available. 11 = default channel number, 2405 MHz

Example

`12 20` // Sets channel 20 (2450 MHz) for testing

Return data:

- Successful command: all-0 status byte (0x00000000)
- Failed command: non-0 status byte (0x00000001 to 0xFFFFFFFF)

15.4 Command 80: Reset 802.15.4 radio

This command resets the 802.15.4 RF block.

Syntax: `80 <Radio_ID>`

Where:

Table 46. Command parameters

Parameter	Description
`Radio_ID`	Radio ID set to 0

Example

`80 0` // Resets 802.15.4 radio

Return data

```
Enter option: 80 0
80 0

Dut_15_4_ResetMCU: 0x00000000
```

15.5 Command 15: Get 802.15.4 TX power level value

This command returns the transmit power value in 1 dB steps.

Syntax: `15`

Example

`15` // Get TX power level in dBm

Return data: Target power level value in dBm

15.6 Command 16: Set 802.15.4 TX power value

This command sets 802.15.4 transmit power level value.

Syntax: `16 <Pwr>`

Where:

Table 47. Command parameters

Parameter	Description
`Pwr`	TX power level in 1 dB steps

Example

`16 11` // Sets 802.15.4 transmit power as 11 dBm

Return data:

- Successful command: all-0 status byte (0x00000000)
- Failed command: non-0 status byte (0x00000001 to 0xFFFFFFFF)

15.7 Command 17: Continuous modulated TX

This command is used to transmit the continuous modulated signal.

Syntax: `17 <Enable>`

Where:

Table 48. Command parameters

Parameter	Description
`Enable`	Enable the continuous transmission 0 = off 1 = on

Example

`17 1` // Enables the device to transmit the continuous modulated signal

Return data:

- Successful command: all-0 status byte (0x00000000)
- Failed command: non-0 status byte (0x00000001 to 0xFFFFFFFF)

15.8 Command 18: Continuous unmodulated TX

This command is used to transmit the continuous unmodulated signal.

Syntax: `18 <Enable> <pwr> <Channel No>`

Where:

Table 49. Command parameters

Parameter	Description
`Enable`	Enable continuous unmodulated TX 0 = off 1 = on
`pwr`	TX power level at pin. Values in the range of -20 dBm to 15dBm (in 0.5 dBm step)
`Channel No`	Channel number in the range of 11 to 26 (2405 MHz to 2480 MHz)

Example(s)

`18 1 10 4` // Enable continuous unmodulated transmit at 10 dBm in channel 4

Return data:

- Successful command: all-0 status byte (0x00000000)
- Failed command: non-0 status byte (0x00000001 to 0xFFFFFFFF)

15.9 Command 20: Get 802.15.4 TX payload size

This command is used to get 802.15.4 TX packet payload size (17-116 bytes).

Syntax: `20`

Example

`20` // Get 802.15.4 TX packet payload size in bytes

Return data: 802.15.4 TX packet payload size in bytes

15.10 Command 21: Set 802.15.4 TX packet payload size

This command is used to set 802.15.4 TX packet payload size in bytes (17-116 bytes).

Syntax: `21 <Size>`

Where:

Table 50. Command parameters

Parameter	Description
`Size`	Packet payload size (17 - 116 bytes)

Example

`21 100` // Sets 802.15.4 packet payload size to 100 bytes

Return data:

- Successful command: all-0 status byte (0x00000000)
- Failed command: non-0 status byte (0x00000001 to 0xFFFFFFFF)

15.11 Command 23: Get TX power limit

This command returns the 802.15.4 TX power limit.

Syntax: `23`

Example

`23` // Get the TX power limit

Return data: currently set max TX power in 0.5 dB steps

15.12 Command 24: Set TX power limit

This command sets the 802.15.4 TX power limit.

Syntax: `24 <TX Power>`

Where:

Table 51. Command parameters

Parameter	Description
`TX power`	TX power limit in half dB steps Default = 44 (22 dBm).

Example

`24 10` // Sets the TX power limit 10 dBm

Return data:

- Successful command: all-0 status byte (0x00000000)
- Failed command: non-0 status byte (0x00000001 to 0xFFFFFFFF)

15.13 Command 33: 802.15.4 TX burst packet

This command is used to transmit a specific number of packets with a specific packet gap.

Syntax: `33 <count option> <packet gap>`

Where:

Table 52. Command parameters

Parameter	Description
`count option`	The number of packet(s) 0 = 1 (default) 1 = 25 2 = 100 3 = 500 4 = 1000 5 = 2000 6 = 5000 7 = 10000
`Packet gap`	Minimum packet gap for a given payload size. Refer to Section 15.10 payload size 17 - 44 bytes: 6 ms payload size 45 - 76 bytes: 7 ms payload size 77 - 107 bytes: 8 ms payload size 108- 116 bytes: 9 ms The maximum gap is 20 ms

Example

`33 1 10` // 802.15.4 radio transmits 25 packets in 10 ms

Return data:

- Successful command: all-0 status byte (0x00000000)
- Failed command: non-0 status byte (0x00000001 to 0xFFFFFFFF)

15.14 Command 35: Set 802.15.4 TX duty cycle

This command sets 802.15.4 transmitter in duty cycle TX mode.

Syntax: `35 <Enable>`

Where:

Table 53. Command parameters

Parameter	Description
`Enable`	Mode enable 0 = disable (default) 1 = enables

Example

`35 1` // Enable 802.15.4 TX duty cycle

Return data:

- Successful command: all-0 status byte (0x00000000)
- Failed command: non-0 status byte (0x00000001 to 0xFFFFFFFF)

15.15 Command 59: Set PHY TX test packet info

This command sets 802.15.4 TX packet info.

Syntax: `59 <Seq Number> <PAN ID> <Destination ID> <Source ID>`

Where:

Table 54. Command parameters

Parameter	Description
`Seq Number`	Packet sequence number. Set to 1.
`PAN ID`	PAN ID of the DUT. Set a two-byte value. Default = 0xFFFF
`Destination ID`	Destination ID of the RF tester or companion device Set a two-byte value. Default = 0xFFFF
`Source ID`	Source ID of the DUT Set a two-byte value. Default = 0xFFFF

Example

`59 1 FFFF FFFF FFFF` // Sets the PHY TX test packet info to default.

15.16 Command 31: Get 802.15.4 RX test result

This command returns the extended RX test result report for 802.15.4 radio.

Syntax: `31`

Example)

`31` // Get 802.15.4 RX test result report

Return data: 802.15.4 RX test result

```
Enter option: 31
31

 Dut_15_4_RxTestResult: 0x00000000
Status_Complete0_Abort1         0
TotalPcktCount                500
RxCount                       500
Avg RSSI                     -53
```

15.17 Command 32: Start 802.15.4 RX test

This command is used to start 802.15.4 RX test.

Syntax: `32`

Example

`32` // Start 802.15.4 RX test using VSG

Return data:

- Successful command: all-0 status byte (0x00000000)
- Failed command: non-0 status byte (0x00000001 to 0xFFFFFFFF)

15.18 Command 45: Get EUI 64 MAC address

This commands gets the 802.15.4 EUI 64 MAC address.

Syntax: `45`

Example

`45` // Get EUI 64 MAC address

Return data: MAC Address in the format XX.XX.XX.XX.XX.XX.XX.XX (8 bytes)

Return data example:

```
Dut_15_4_GetEUI64MacAddress: 0
EUI_MAC_ADDRESS: 10.20.30.40.50.60.70.80
```

15.19 Command 46: Set EUI 64 MAC address

This commands sets the 802.15.4 EUI 64 MAC address.

Note: *Once the EUI MAC address is programmed into the OTP, the command* `46` *can NOT update the MAC address at runtime.*

Syntax: `46 <MAC address>`

Where:

Table 55. Command parameters

Parameter	Description
`MAC address`	MAC address in the format XX.XX.XX.XX.XX.XX.XX.XX (8 bytes)

Return code: None

Example

`46 10.20.30.40.50.60.70.80` // Set EUI 64 MAC Address to 10.20.30.40.50.60.70.80

Return data:

- Successful command: all-0 status byte (0x00000000)
- Failed command: non-0 status byte (0x00000001 to 0xFFFFFFFF)

15.20 Command 99: Exit 802.15.4 menu

Issue command 99 once to exit from 802.15.4 Labtool menu.

Issue command 99 a second time to exit from Labtool and close the command prompt window.

16 Labtool command sequence examples to test 802.15.4 radio

This section provides examples of Labtool command test sequence for various 802.15.4 transmit and receive RF performance tests.

16.1 Labtool command sequence to test 802.15.4 radio transmit in duty cycle mode

```
12 20   // Set channel to 20 (2450MHz)
16 4    // Set power to 4dBm
35 1    // Enable Tx in duty cycle mode
35 0    // Stop Tx
```

16.2 Labtool command sequence to test 802.15.4 radio transmit in burst packet mode

```
12 15     // Set channel to 15 (2425MHz)
16 4      // Set power to 4dBm
33 3 7    // Tx 500 packets at 7ms packet gap
```

16.3 Labtool command sequence to test 802.15.4 radio in receive mode

```
12 26    // Set channel to 26 (2480MHz)
-Connect the DUT to 802.15.4 VSG
        // Set Generator File
        // Set 802.15.4 channel is 26 (2480MHz)
        // Set signal level at -90dBm
32      // Start Rx
-Transmit packets from 802.15.4 VSG
31     // Return extended Rx result report
```

16.4 Labtool command sequence to test 802.15.4 radio transmit in continuous unmodulated wave mode

```
80          // Reset 802.15.4 radio
18 1 4 11   // Enable CW unmodulated TX at 4dBm on Channel 11
18 0        // Disable CW unmodulated TX
```

17 List of supported channels for 802.15.4 radio

The 802.15.4 radio operates in the 2.4 GHz ISM band at 2400-2483.5 MHz. The 802.15.4 radio uses 16 RF channels. The center frequency value for these channels is calculated per:

2405 + 5 (k – 11) MHz, where k = 11, 12, …, 26

Table 56. Operating frequency bands

Regulatory range	RF channels
2400 - 2483.5 MHz	*2405 + 5 (k – 11)* MHz; k = 11, 12, …, 26

Table 57. List of supported RF channels for 802.15.4 radio

Channel number	Frequency (MHz)
11	2405
12	2410
13	2415
14	2420
15	2425
16	2430
17	2435
18	2440
19	2445
20	2450
21	2455
22	2460
23	2465
24	2470
25	2475
26	2480

18 Acronyms and abbreviations

Table 58. Abbreviations

Acronym	Description
BSS	Basic service set
BSSID	Basic service set identifier
CW	Continuous wave
DUT	Device under test
EVM	Error vector magnitude
GF	Green field
GI	Guard interval
RU	Resource unit
SIFS	Short inter frame space
STBC	Space time block code
VSA	Vector signal analyzer
VSG	Vector signal generator

19 References

[1] Application note – AN13639: RW61x Calibration Structure
[2] Data sheet – RW610:Wireless MCU with Integrated 1x1 Wi-Fi 6, Bluetooth Low Energy (link)
[3] Data sheet – RW612: Wireless MCU with Integrated 1x1 Wi-Fi 6, Bluetooth Low Energy / 802.15.4 (link)
[4] User manual – UM11798: Getting Started with Wireless on RW61x Evaluation board Running RTOS (link)
[5] Webpage – RW610: Wireless MCU with Integrated 1x1 Wi-Fi® 6 + Bluetooth® Low Energy Radios (link)
[6] Webpage – RW612: Wireless MCU with Integrated Tri-radio: 1x1 Wi-Fi® 6 + Bluetooth® Low Energy / 802.15.4 (link)

20 Note about the source code in the document

The example code shown in this document has the following copyright and BSD-3-Clause license:

Copyright 2022-2024 NXP Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials must be provided with the distribution.
3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

21 Revision history

Table 59. Revision history

Document ID	Release date	Description
UM11801 v.5.0	22 November 2024	**Software** • Section 2 "Software": updated. • Section 2.2 "Content": updated the filename of Labtool application. **Labtool test setup** • Section 3 "Labtool test setup": removed the note about optional RF shield box. **Update the manufacturing bridge application and/or firmware** • Section 4 "Update the manufacturing bridge application and/or firmware": updated the filename and content of the *.jlink* file. **Labtool setup on Windows** • Section 5 "Labtool setup on Windows PC": – Changed the figure showing the content of *Labtool* directory. – Updated the code snippet showing {ZBEE_INTF] – Updated the filename of Labtool application. **Labtool usage** • Section 7.1 "Starting Labtool": updated the filename of Labtool application. **Wi-Fi Labtool commands** • Section 8.2 "Labtool command list for Wi-Fi": added. • Section 8.7 "Command 10: Set antenna": updated. • Section 8.8 "Command 11: Get Wi-Fi RF channel": updated the return data. • Section 8.12 "Command 35: Transmit with SIFS gap": updated the minimum value of `Burst Sifs` parameter. • Section 8.15 "Command 235 - Transmit OFDMA packet with configuration file": updated the minimum value of `Burst Sifs` parameter to 10 µs. • Section 8.13 "Command 225: Transmit 802.11ax with SIFS gap": – Updated the minimum value of `Burst Sifs in us` parameter to 10 µs. – Updated the description of `Adv Coding`. • Section 8.16 "Command 18: Set Wi-Fi CW TX": added. • Section 8.23 "Command 53: Write calibration data from text files to OTP or .conf file": added *Cal15_4DataFile.txt*. • Section 8.24 "Command 54: Get calibration data from OTP memory into text file": added *Cal15_4DataFile.txt*. • Section 8.31 "Command 120: Get the thermal sensor reading ": updated the return value format to decimal and added a note. **Labtool command sequence examples to test Wi-Fi** • Section 9.5 "Labtool command sequences for 802.11ac TX/RX": removed the examples for 2.4 GHz. • Section 9.6 "Labtool command sequences for 802.11ax TX/RX": removed the references to antennas. • Section 9.7 "Labtool command sequence for Wi-Fi RSSI measurement": removed the references to antenna 1. • Section 9.10 "Labtool command sequence for Wi-Fi continuous wave (CW) transmit": added. ——— continues ———

Table 59. Revision history...*continued*

Document ID	Release date	Description
UM11801 v.5.0	22 November 2024	——— continued ——— **Bluetooth LE Labtool commands** • Section 12.1 "Labtool command list for Bluetooth LE": updated. • Section 12.3 "Command 88: Get the firmware and Labtool versions ": added a screen capture showing an example of command return. • Section 12.4 "Command 44: Get/set the storage type ": added. • Section 12.6 "Command 15: Get Bluetooth TX power level value": added. • Section 12.7 "Command 16: Set Bluetooth TX power level value": added. • Section 12.8 "Command 18: Set Bluetooth CW TX ": added. • Section 12.9 "Command 113: Get Bluetooth TX power control class": added. • Section 12.10 "Command 114: Set Bluetooth TX power control class": added. • Section 12.11 "Command 115: Get Bluetooth power control status": added. • Section 12.12 "Command 116: Set Bluetooth power control": added. **Labtool command sequence examples to test Bluetooth LE TX/RX** • Section 13 "Labtool command sequence examples to test Bluetooth LE TX/RX": added the example for Bluetooth LE CW TX. **802.15.4 Labtool commands** • Section 15.1 "Labtool command list for 802.15.4 radio": updated. • Section 15.7 "Command 17: Continuous modulated TX": updated the command description. • Section 15.8 "Command 18: Continuous unmodulated TX": added. **Labtool command sequence examples to test 802.15.4 radio** • Section 16.4 "Labtool command sequence to test 802.15.4 radio transmit in continuous unmodulated wave mode": added.

Table 59. Revision history...*continued*

Document ID	Release date	Description
UM11801 v.4.0	22 April 2024	**Wi-Fi Labtool commands** • Section 4: updated the second code sample in **step 3**. • Section 8.2 "Labtool command list for Wi-Fi": updated. • Section 8.6 "Command 9: Get antenna ": added note about antenna diversity design. • Section 8.7 "Command 10: Set antenna": added note about antenna diversity design. • Section 8.10 "Command 111: Get the channel bandwidth": added. • Section 8.11 "Command 112: Set the channel bandwidth": added. • Section 8.13 "Command 225: Transmit 802.11ax with SIFS gap": added. • Section 8.14 "Command 231: Enable uplink OFDMA Tx": added. • Section 8.15 "Command 235 - Transmit OFDMA packet with configuration file": added. • Section 8.31 "Command 120: Get the thermal sensor reading ": updated. • Section 9.8 "Labtool command sequences for Wi-Fi transmit power calibration": added. • Section 9.9 "Labtool command sequences for Wi-Fi transmit with HE-EXT-SU": added. • Section 10.2 "RF channels": updated. **Testing 802.11ax uplink-OFDMA transmit** • Section 11 "Testing 802.11ax uplink-OFDMA transmit": added. **802.15.4 Labtool commands** • Section 15.1 "Labtool command list for 802.15.4 radio": updated. • Section 15.11 "Command 23: Get TX power limit": added. • Section 15.12 "Command 24: Set TX power limit": added. • Section 15.15 "Command 59: Set PHY TX test packet info": added. • Section 19 "References": updated.
UM11801 v.3.0	6 October 2023	• Section 8.2 "Labtool command list for Wi-Fi": updated. • Section 8.25 "Command 101: Get RF control mode": added. • Section 8.26 "Command 102: Set RF control mode": added. • Section 8.31 "Command 120: Get the thermal sensor reading ": updated. • Section 10.2 "RF channels": updated. • Section 12.16 "Command 127: Set Bluetooth LE RX test": updated `Phy` description for LR2 and LR8. • Section 13 "Labtool command sequence examples to test Bluetooth LE TX/RX": corrected the parameter value for LR2 in the command sequence example for *Bluetooth LE Rx on channel 39 with LR2 data rate*. • Section 20 "Note about the source code in the document": added.

Table 59. Revision history...*continued*

Document ID	Release date	Description
UM11801 v.2.0	17 July 2023	• Section 1.2 "About Labtool": updated • Replace sections *Labtool test setup*, *Device under test setup*, and *Windows PC setup* with: – Section 2 "Software" – Section 3 "Labtool test setup" – Section 4 "Update the manufacturing bridge application and/or firmware" – Section 5 "Labtool setup on Windows PC" – Section 6 "Calibration data" • Section 8.6 "Command 9: Get antenna ": added • Section 8.7 "Command 10: Set antenna": added • Section 8.31 "Command 120: Get the thermal sensor reading ": added • Section 8.32 "Command 121: Get the power detector reading offset": added • Section 12.14 "Command 125: Set Bluetooth LE TX test": updated the definition of `Phy` parameter • Section 13 "Labtool command sequence examples to test Bluetooth LE TX/RX": updated for LR2 and LR8 data rates • Section 15 "802.15.4 Labtool commands": added
UM11801 v.1.0	13 May 2022	Initial version

Legal information

Definitions

Draft — A draft status on a document indicates that the content is still under internal review and subject to formal approval, which may result in modifications or additions. NXP Semiconductors does not give any representations or warranties as to the accuracy or completeness of information included in a draft version of a document and shall have no liability for the consequences of use of such information.

Disclaimers

Limited warranty and liability — Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information. NXP Semiconductors takes no responsibility for the content in this document if provided by an information source outside of NXP Semiconductors.

In no event shall NXP Semiconductors be liable for any indirect, incidental, punitive, special or consequential damages (including - without limitation - lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory.

Notwithstanding any damages that customer might incur for any reason whatsoever, NXP Semiconductors' aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms and conditions of commercial sale of NXP Semiconductors.

Right to make changes — NXP Semiconductors reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

Suitability for use — NXP Semiconductors products are not designed, authorized or warranted to be suitable for use in life support, life-critical or safety-critical systems or equipment, nor in applications where failure or malfunction of an NXP Semiconductors product can reasonably be expected to result in personal injury, death or severe property or environmental damage. NXP Semiconductors and its suppliers accept no liability for inclusion and/or use of NXP Semiconductors products in such equipment or applications and therefore such inclusion and/or use is at the customer's own risk.

Applications — Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

Customers are responsible for the design and operation of their applications and products using NXP Semiconductors products, and NXP Semiconductors accepts no liability for any assistance with applications or customer product design. It is customer's sole responsibility to determine whether the NXP Semiconductors product is suitable and fit for the customer's applications and products planned, as well as for the planned application and use of customer's third party customer(s). Customers should provide appropriate design and operating safeguards to minimize the risks associated with their applications and products.

NXP Semiconductors does not accept any liability related to any default, damage, costs or problem which is based on any weakness or default in the customer's applications or products, or the application or use by customer's third party customer(s). Customer is responsible for doing all necessary testing for the customer's applications and products using NXP Semiconductors products in order to avoid a default of the applications and the products or of the application or use by customer's third party customer(s). NXP does not accept any liability in this respect.

Terms and conditions of commercial sale — NXP Semiconductors products are sold subject to the general terms and conditions of commercial sale, as published at https://www.nxp.com/profile/terms, unless otherwise agreed in a valid written individual agreement. In case an individual agreement is concluded only the terms and conditions of the respective agreement shall apply. NXP Semiconductors hereby expressly objects to applying the customer's general terms and conditions with regard to the purchase of NXP Semiconductors products by customer.

Export control — This document as well as the item(s) described herein may be subject to export control regulations. Export might require a prior authorization from competent authorities.

Suitability for use in non-automotive qualified products — Unless this document expressly states that this specific NXP Semiconductors product is automotive qualified, the product is not suitable for automotive use. It is neither qualified nor tested in accordance with automotive testing or application requirements. NXP Semiconductors accepts no liability for inclusion and/or use of non-automotive qualified products in automotive equipment or applications.

In the event that customer uses the product for design-in and use in automotive applications to automotive specifications and standards, customer (a) shall use the product without NXP Semiconductors' warranty of the product for such automotive applications, use and specifications, and (b) whenever customer uses the product for automotive applications beyond NXP Semiconductors' specifications such use shall be solely at customer's own risk, and (c) customer fully indemnifies NXP Semiconductors for any liability, damages or failed product claims resulting from customer design and use of the product for automotive applications beyond NXP Semiconductors' standard warranty and NXP Semiconductors' product specifications.

HTML publications — An HTML version, if available, of this document is provided as a courtesy. Definitive information is contained in the applicable document in PDF format. If there is a discrepancy between the HTML document and the PDF document, the PDF document has priority.

Translations — A non-English (translated) version of a document, including the legal information in that document, is for reference only. The English version shall prevail in case of any discrepancy between the translated and English versions.

Security — Customer understands that all NXP products may be subject to unidentified vulnerabilities or may support established security standards or specifications with known limitations. Customer is responsible for the design and operation of its applications and products throughout their lifecycles to reduce the effect of these vulnerabilities on customer's applications and products. Customer's responsibility also extends to other open and/or proprietary technologies supported by NXP products for use in customer's applications. NXP accepts no liability for any vulnerability. Customer should regularly check security updates from NXP and follow up appropriately.

Customer shall select products with security features that best meet rules, regulations, and standards of the intended application and make the ultimate design decisions regarding its products and is solely responsible for compliance with all legal, regulatory, and security related requirements concerning its products, regardless of any information or support that may be provided by NXP.

NXP has a Product Security Incident Response Team (PSIRT) (reachable at PSIRT@nxp.com) that manages the investigation, reporting, and solution release to security vulnerabilities of NXP products.

NXP B.V. — NXP B.V. is not an operating company and it does not distribute or sell products.

Trademarks

Notice: All referenced brands, product names, service names, and trademarks are the property of their respective owners.

NXP — wordmark and logo are trademarks of NXP B.V.

Amazon Web Services, AWS, the Powered by AWS logo, and FreeRTOS — are trademarks of Amazon.com, Inc. or its affiliates.

Bluetooth — the Bluetooth wordmark and logos are registered trademarks owned by Bluetooth SIG, Inc. and any use of such marks by NXP Semiconductors is under license.

J-Link — is a trademark of SEGGER Microcontroller GmbH.

Tables

Tab. 1. Manufacturing software package content 4
Tab. 2. Manufacturing software package content 4
Tab. 3. NO_EEPROM parameter description 11
Tab. 4. Labtool command list for Wi-Fi 17
Tab. 5. Radio mode indexes 19
Tab. 6. Command parameters 20
Tab. 7. Command parameters 22
Tab. 8. Command parameters 23
Tab. 9. Command parameters 23
Tab. 10. Command parameters 24
Tab. 11. Command parameters 24
Tab. 12. Command parameters 25
Tab. 13. Command parameters 27
Tab. 14. Command parameters 29
Tab. 15. Command parameters 30
Tab. 16. Command parameters 31
Tab. 17. Command parameters 32
Tab. 18. Command parameters 32
Tab. 19. Command parameters 33
Tab. 20. Command parameters 34
Tab. 21. Command parameters 34
Tab. 22. Command parameters 36
Tab. 23. Command parameters 36
Tab. 24. Command parameters 37
Tab. 25. Command parameters 38
Tab. 26. Command parameters 39
Tab. 27. Command parameters 40
Tab. 28. 802.11n/a/g/b data rate ID 51
Tab. 29. 802.11ac/802.11ax data rate ID 52
Tab. 30. Command parameters 55
Tab. 31. Bluetooth LE Labtool command list 59
Tab. 32. Command parameters 60
Tab. 33. Command parameters 61
Tab. 34. Command parameters 62
Tab. 35. Command parameters 63
Tab. 36. Command parameters 63
Tab. 37. Command parameters 64
Tab. 38. Command parameters 64
Tab. 39. Command parameters 65
Tab. 40. Command parameters 66
Tab. 41. Command parameters 67
Tab. 42. Operating frequency bands 71
Tab. 43. Channel frequencies 71
Tab. 44. Labtool command list for 802.15.4 radio 72
Tab. 45. Command parameters 73
Tab. 46. Command parameters 74
Tab. 47. Command parameters 75
Tab. 48. Command parameters 75
Tab. 49. Command parameters 76
Tab. 50. Command parameters 77
Tab. 51. Command parameters 78
Tab. 52. Command parameters 79
Tab. 53. Command parameters 80
Tab. 54. Command parameters 80
Tab. 55. Command parameters 82
Tab. 56. Operating frequency bands 84
Tab. 57. List of supported RF channels for 802.15.4 radio 84
Tab. 58. Abbreviations 85
Tab. 59. Revision history 88

Figures

Fig. 1. Design Resources link 3
Fig. 2. Sign in to access software 3
Fig. 3. Set-up for RW61x Labtool bridge test mode 5
Fig. 4. HD12 and U38CON configuration on RW61x reference design 6
Fig. 5. Content of Labtool directory 9
Fig. 6. JLink CDC UART port number 9
Fig. 7. Test setup 53
Fig. 8. RU index values for 20 MHz bandwidth 55
Fig. 9. Recording of UL-OFDMA signal 57

Contents

1 Introduction 2
1.1 Purpose and scope 2
1.2 About Labtool 2
1.3 Notation conventions 2
2 Software 3
2.1 Download 3
2.2 Content 4
3 Labtool test setup 5
4 Update the manufacturing bridge application and/or firmware 6
5 Labtool setup on Windows PC 9
6 Calibration data 11
7 Labtool usage 12
7.1 Starting Labtool 12
7.1.1 Labtool command response 15
7.2 Listing Labtool commands 16
7.3 Using the help menu 16
7.4 Keeping default parameter values 16
7.5 Closing Labtool 16
8 Wi-Fi Labtool commands 17
8.1 Naming convention 17
8.2 Labtool command list for Wi-Fi 17
8.3 Command 88: Get the firmware and Labtool versions 18
8.4 Command 5: Get radio mode index info 19
8.5 Command 6: Set radio mode index 20
8.6 Command 9: Get antenna 21
8.7 Command 10: Set antenna 22
8.8 Command 11: Get Wi-Fi RF channel 23
8.9 Command 12: Set Wi-Fi RF channel 23
8.10 Command 111: Get the channel bandwidth 24
8.11 Command 112: Set the channel bandwidth 24
8.12 Command 35: Transmit with SIFS gap 25
8.13 Command 225: Transmit 802.11ax with SIFS gap 27
8.14 Command 231: Enable uplink OFDMA Tx 29
8.15 Command 235 - Transmit OFDMA packet with configuration file 30
8.16 Command 18: Set Wi-Fi CW TX 31
8.17 Command 32: Get the received packet count (Stop Rx FER test) 32
8.18 Command 31: Clear the received packet count 32
8.19 Command 198: Start RSSI data collection 33
8.20 Command 199: Stop RSSI data collection and report result 34
8.21 Command 44: Get/set the storage type 34
8.22 Command 22: Load calibration data from the configuration file 35
8.23 Command 53: Write calibration data from text files to OTP or .conf file 35
8.24 Command 54: Get calibration data from OTP memory into text file 35
8.25 Command 101: Get RF control mode 36
8.26 Command 102: Set RF control mode 36
8.27 Command 45: Read the MAC address from OTP memory 36
8.28 Command 46: Write MAC address to OTP memory 37
8.29 Command 95: Get RF crystal calibration offset 37
8.30 Command 96: Set RF crystal calibration offset 38
8.31 Command 120: Get the thermal sensor reading 39
8.32 Command 121: Get the power detector reading offset 40
8.33 Command 99: Exit Labtool 41
9 Labtool command sequence examples to test Wi-Fi radio 42
9.1 Labtool command sequence for 802.11b TX/RX 42
9.1.1 Labtool command sequence for 802.11b TX 42
9.1.2 Labtool command sequence for 802.11b RX 42
9.2 Labtool command sequence for 802.11g TX/RX 43
9.2.1 Labtool command sequence for 802.11g TX 43
9.2.2 Labtool command sequence for 802.11g RX 43
9.3 Labtool command sequence for 802.11a TX/RX 44
9.3.1 Labtool command sequence for 802.11a TX 44
9.3.2 Labtool command sequence for 802.11a RX 44
9.4 Labtool command sequences for 802.11n TX/RX 45
9.4.1 Labtool command sequence for 802.11n TX, 2.4 GHz 45
9.4.2 Labtool command sequence for 802.11n TX, 5 GHz 45
9.4.3 Labtool command sequence for 802.11n RX, 2.4 GHz 45
9.5 Labtool command sequences for 802.11ac TX/RX 46
9.5.1 Labtool command sequence for 802.11ac TX, 5 GHz 46
9.6 Labtool command sequences for 802.11ax TX/RX 47
9.6.1 Labtool command sequence for 802.11ax TX, 2.4 GHz 47
9.6.2 Labtool command sequence for 802.11ax TX, 5 GHz 47
9.6.3 Labtool command sequence for 802.11ax RX, 2.4 GHz 47
9.6.4 Labtool command sequence for 802.11ax RX, 5 GHz 47
9.7 Labtool command sequence for Wi-Fi RSSI measurement 48
9.8 Labtool command sequences for Wi-Fi transmit power calibration 49

9.9 Labtool command sequences for Wi-Fi transmit with HE-EXT-SU 49
9.10 Labtool command sequence for Wi-Fi continuous wave (CW) transmit 50
10 List of supported data rates and channels for Wi-Fi 51
10.1 Data rates 51
10.2 RF channels 52
11 Testing 802.11ax uplink-OFDMA transmit 53
11.1 Test setup 53
11.2 Standalone uplink-OFDMA test 54
11.3 Edit the configuration file 55
11.4 Configure the golden unit 56
11.5 Configure the DUT 57
11.6 Command sequence for Wi-Fi 802.11ax trigger-based UL-OFDMA test 58
12 Bluetooth LE Labtool commands 59
12.1 Labtool command list for Bluetooth LE 59
12.2 Command 80: Reset Bluetooth 59
12.3 Command 88: Get the firmware and Labtool versions 60
12.4 Command 44: Get/set the storage type 60
12.5 Command 234: Reload Bluetooth calibration data 61
12.6 Command 15: Get Bluetooth TX power level value 61
12.7 Command 16: Set Bluetooth TX power level value 61
12.8 Command 18: Set Bluetooth CW TX 62
12.9 Command 113: Get Bluetooth TX power control class 63
12.10 Command 114: Set Bluetooth TX power control class 63
12.11 Command 115: Get Bluetooth power control status 64
12.12 Command 116: Set Bluetooth power control 64
12.13 Command 133: Set Bluetooth LE TX power 65
12.14 Command 125: Set Bluetooth LE TX test 66
12.15 Command 126: Get Bluetooth LE TX test packet count 66
12.16 Command 127: Set Bluetooth LE RX test 67
12.17 Command 128: Get Bluetooth LE test packet error 67
12.18 Command 129: Stop Bluetooth LE test 67
12.19 Command 45: Read Bluetooth device (BD) address 68
12.20 Command 99: Exit Bluetooth menu 68
13 Labtool command sequence examples to test Bluetooth LE TX/RX 69
13.1 Bluetooth LE transmitter 69
13.2 Bluetooth LE receiver 70
14 List of supported channels for Bluetooth LE radio 71
15 802.15.4 Labtool commands 72
15.1 Labtool command list for 802.15.4 radio 72
15.2 Command 11: Get the 802.15.4 channel 73
15.3 Command 12: Set the 802.15.4 channel 73
15.4 Command 80: Reset 802.15.4 radio 74
15.5 Command 15: Get 802.15.4 TX power level value 74
15.6 Command 16: Set 802.15.4 TX power value 75
15.7 Command 17: Continuous modulated TX 75
15.8 Command 18: Continuous unmodulated TX 76
15.9 Command 20: Get 802.15.4 TX payload size 77
15.10 Command 21: Set 802.15.4 TX packet payload size 77
15.11 Command 23: Get TX power limit 78
15.12 Command 24: Set TX power limit 78
15.13 Command 33: 802.15.4 TX burst packet 79
15.14 Command 35: Set 802.15.4 TX duty cycle 80
15.15 Command 59: Set PHY TX test packet info 80
15.16 Command 31: Get 802.15.4 RX test result 81
15.17 Command 32: Start 802.15.4 RX test 81
15.18 Command 45: Get EUI 64 MAC address 82
15.19 Command 46: Set EUI 64 MAC address 82
15.20 Command 99: Exit 802.15.4 menu 82
16 Labtool command sequence examples to test 802.15.4 radio 83
16.1 Labtool command sequence to test 802.15.4 radio transmit in duty cycle mode 83
16.2 Labtool command sequence to test 802.15.4 radio transmit in burst packet mode 83
16.3 Labtool command sequence to test 802.15.4 radio in receive mode 83
16.4 Labtool command sequence to test 802.15.4 radio transmit in continuous unmodulated wave mode 83
17 List of supported channels for 802.15.4 radio 84
18 Acronyms and abbreviations 85
19 References 86
20 Note about the source code in the document 87
21 Revision history 88
Legal information 92