Application note

DA1468x Bluetooth© Direct Test Mode

AN-B-043

Abstract

This application note explains how to setup RF testing for the DA1468x BLE SoC using various BT testers.

Contents

Figures

1 Terms and definitions

DTM	Direct Test Mode
BLE	Bluetooth Low Energy (or Bluetooth Smart)
RF	Radio Frequency
DUT	Device Under Test (also called EUT: Equipment Under Test)
GPIO	General Purpose Input Output
GUI	Graphical User Interface
PCB	Printed Circuit Board
PER	Packet Error Rate
SoC	System on Chip
SWD	Serial Wire Debug
UART	Universal Asynchronous Receiver/Transmitter

2 References

[1] DA14680, DA14681, DA14682, DA14683 Datasheets, Dialog Semiconductor

[2] DA1468x hci_customer_production_test_core_commands_v3.xlsx

3 Introduction

This document describes how to measure the RF performance of the DA14680, DA14681, DA14682 or DA14683 BLE device using a BT tester like R&S CBT, R&S CMW270 or Anritsu MT8852B using the Bluetooth Low Energy (BLE) Direct Test Mode (DTM).
Other equipment that do not support signalling mode are also discussed, like the LitePoint IQ tester.

The DA14680, DA14681, DA14682 and DA14683 (DA1468x) are Bluetooth® Smart SoC devices, working with extremely low power while providing world-class RF performance, a small footprint and flexible peripheral configurations for a wide range of applications like wearables.

The DA1468x SoC supports Direct Test Mode (DTM) for RF PHY as specified by the Bluetooth SIG. The Device Under Test (DUT) communicates with the Bluetooth tester over a 2-wire HCI UART. Details are explained in section 4.
See Figure 1 through Figure 4 for descriptions of the test setups.

4 Setting up Direct Test Mode

4.1 Software Setup

For Bluetooth Direct Test Mode to work, the DTM test software must be loaded in the DA1468x SRAM memory. For the DA1468x this software is provided in the SDK, in the *\DA1468x_SDK_BTLE_v_1.0. . \projects\dk_apps\reference_designs\plt_fw* folder.
The DTM binary can also be requested from the Dialog Semiconductor LPC Customer Support team. The compiled PLT_FW.bin software can be loaded in SRAM via UART or SWD (Jlink).
Loading the DTM software can be done using e.g. the Booter or RF-Master tools in SmartSnippets Toolbox. Read the SmartSnippets Toolbox User Manual (UM-B-083) included in the install for details.
Alternatively the project can be compiled for QSPI Flash, and programmed in the QSPI Flash. The latter option is convenient when many RF tests are planned: no need to load the binary every time.
By default the UART baudrate is set to 115200 and the default UART GPIO's for signalling between the BT tester and the DUT are P1_3 and P2_3, respectively the DA1468x UART Tx and UART Rx.

4.2 Hardware Setup

The measurements on the RF PHY for instance can be performed using the R&S CBT or Anritsu MT8852B in local mode by controlling the buttons on the equipment, or in remote mode under control of a PC tool running a test script in R&S CBTgo or CMWRUN for the CMW270, Anritsu's Bluetooth Low Energy Measurement Software or Anritsu's Combi Test program.
Signalling can be executed directly from the tester's serial COM- or EUT-port to the DA1468x UART pins as indicated in Figure 1, or as in Figure 2 via the PC using a Comm Tunnel tool.
The Figure 1 and Figure 2 setups in general can be used for all BLE DTM tests using signalling, e.g R&S CBT, Anritsu MT8852B, Tescom TC-3000C or similar Bluetooth Test equipment.

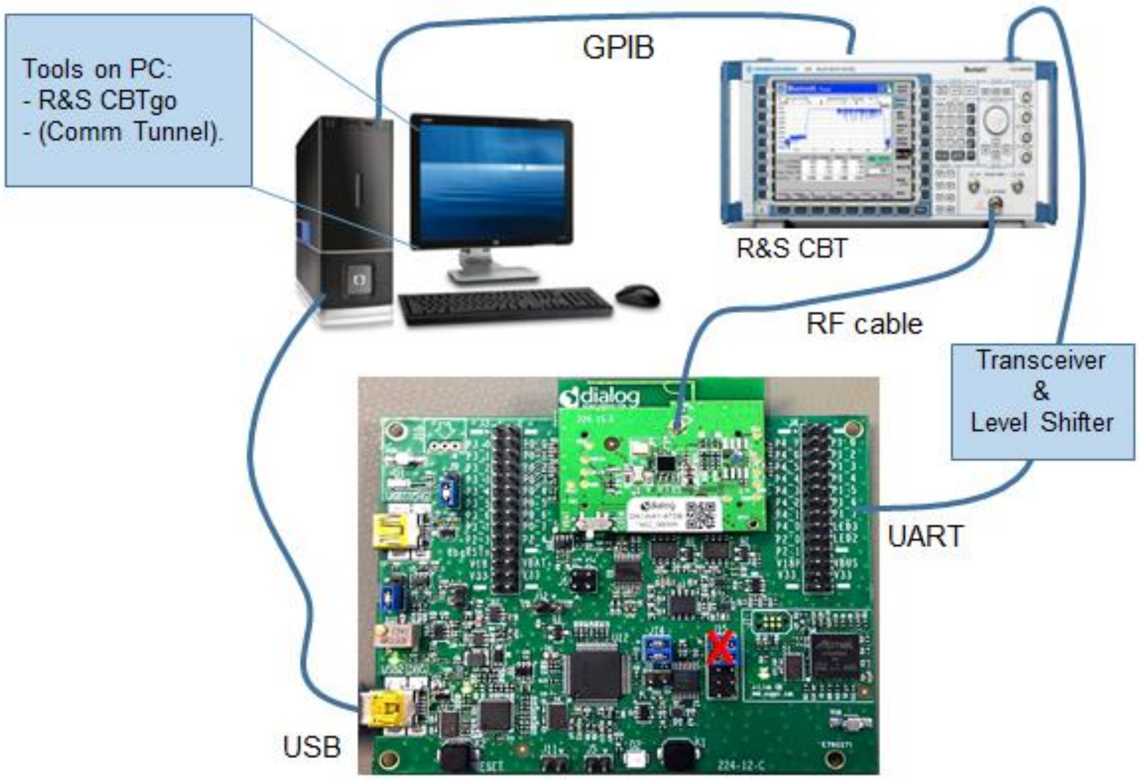


Figure 1: Setup for Bluetooth Direct Test Mode using R&S CBT.

The Litepoint IQxel-M tester connections are given in Figure 4, and a signalling over USB method is shown in Figure 3, using the R&S CMW270 as example.

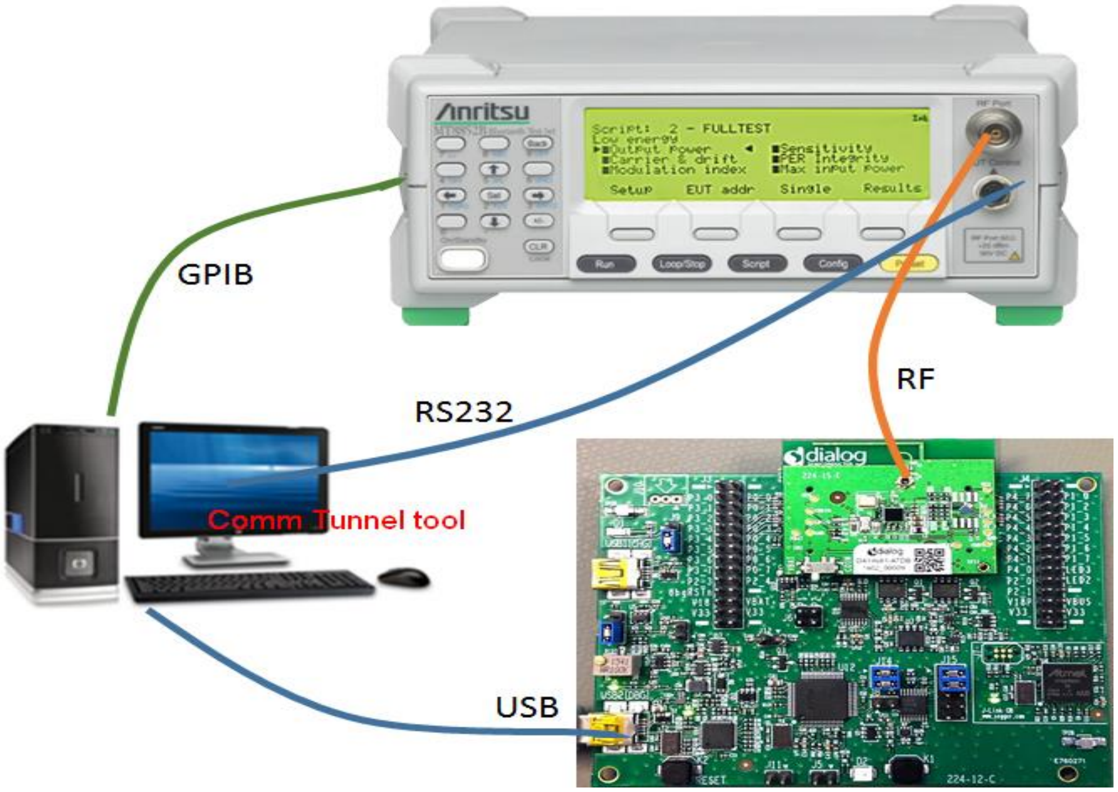


Figure 2: DTM RF testing using Anritsu tester and a serial Comm Tunnel tool.

In **Figure 3** the connections are given for the R&S CMW270 and a DA1468x kit, signalling is done via the CMW's USB port. The CMW270's USB port is providing the supply for the Pro-DK as well.

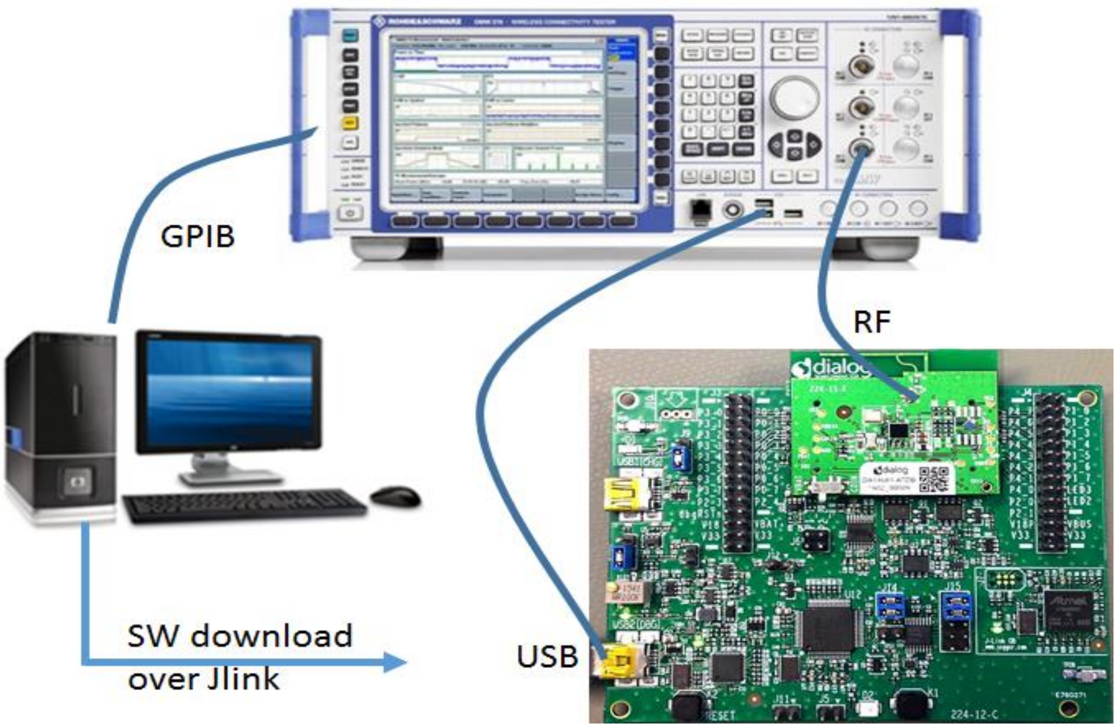


Figure 3: USB Port signalling when using the R&S CMW270.

In this non signalling LitePoint example in Figure 4, the PC controls the IQxel-M tester (loading waveforms and e.g. start Tx) and controls the DUT telling it to e.g. receive packets from the IQ tester, or sending packets to the tester in packeted Tx mode. Since this is a non-signalling setup, the DA1468x must be controlled by a PC based tool, e.g. a serial port terminal program or RF-Master in SmartSnippets ToolBox. Examples of these are given in chapters 4.3 and 4.4.

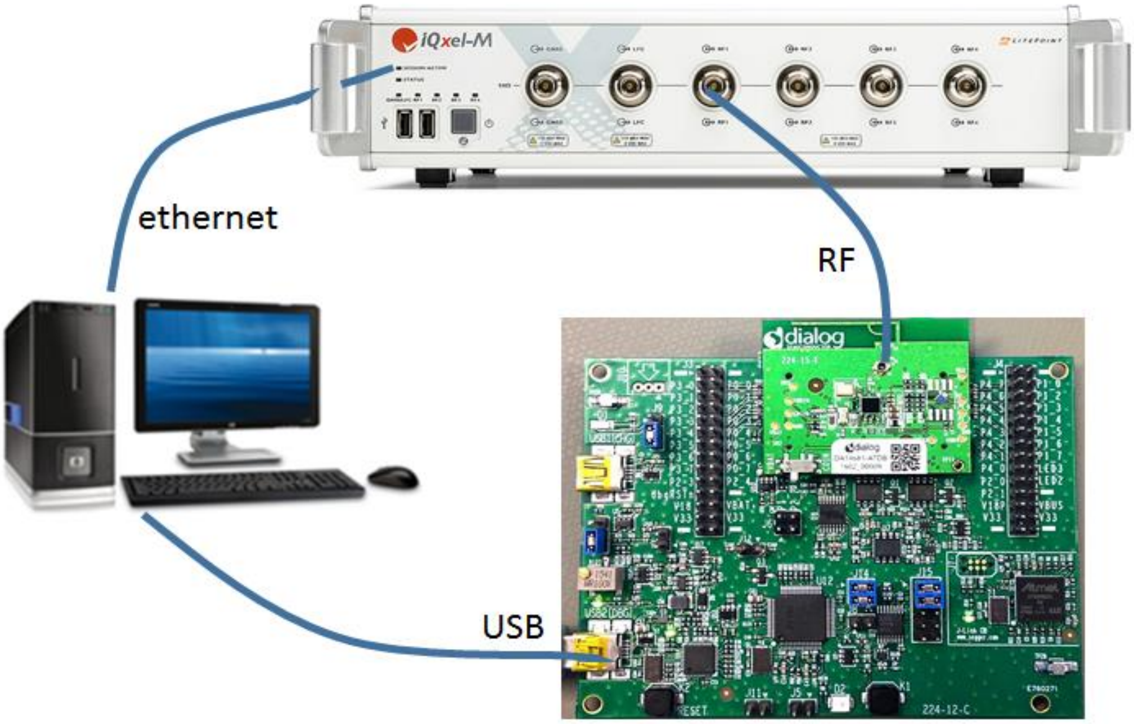


Figure 4: Non DTM mode example using a LitePoint IQxel-M tester.

4.2.1 Setting up DTM using DA1468x UART using R&S CBT

For the complete setup, please refer to Figure 1. Please remove the jumpers at J15.
The CBT's COM port must be connected to the DA1468x UART ports P1_3 (UTx) and P2_3 (URx) via a level shifter. The default baudrate is 115200 baud. In case the DA1468x Pro-DK board is used, connect the UART wires to P1_3 (UTx) and to P2_3 (URx) of the board's J15 connector, pin #2 for UTx and pin #4 for URx. Please see Figure 5. Alternatively, a COM Tunnel tool as described in section 4.2.2 can be used. For tunneling make sure the upper two jumpers at J15 are placed, since the UART signals now come from the on-board FTDI chip connected to pins #1 and #3 of J15.

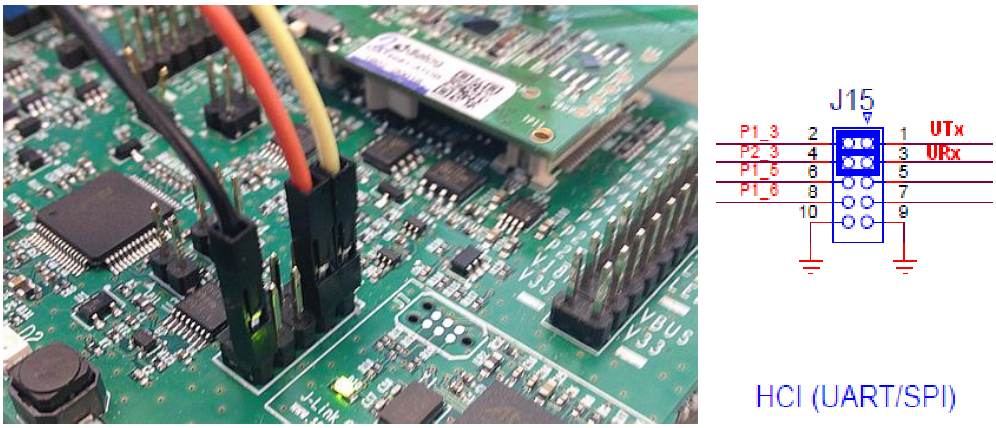


Figure 5: UART connection at Pro-DK J15

4.2.2 Setting up DTM via Comm Tunnel tool using Anritsu MT8852B

Please refer to Figure 2 for the setup.
The MT8852B EUT cable connects to the physical or a virtual COM port of the PC or Laptop.
One of the PC's USB ports is connected to the DA1468x Pro-DK board's USB port. In all other cases an USB-to-RS232 cable can be used, e.g. the TTL-232R-3V3. In case the DA1468x Pro-DK is used, the FTDI cable can be connected to the board's J15 connector. The yellow RXD wire connects to UTx at pin #2 (P1_3), the orange TXD wire connects to URx at pin #4 (P2_3). The black wire is the ground connection and will be connected to J15 pin #10. Please refer to Figure 6.

Note: when the FTDI 232R 3V3 cable is plugged onto the J15 connector, the J15 #6 and #8 pins connect to the cable's VCC (red wire) and CTS# (brown wire) respectively. This has no effect on the operation of the DA1468x device in DTM mode. The default state of the ports P1_5 and P1_6 at J15 pins # 6 and #8 are input pull-up to 3.3V. The cable's RTS# (green wire) does not connect to any pin.

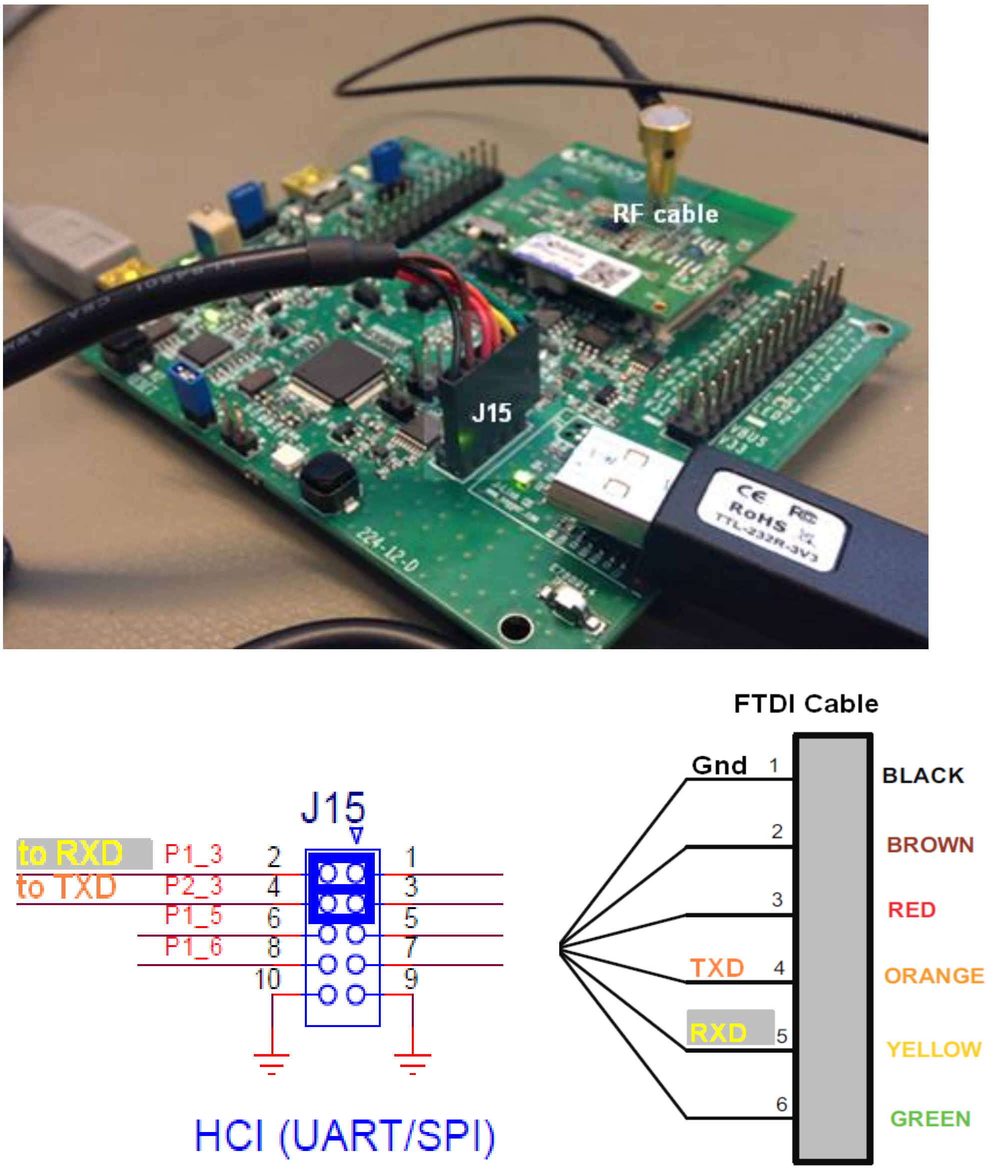


Figure 6: FTDI cable connections to the Pro-DK board.

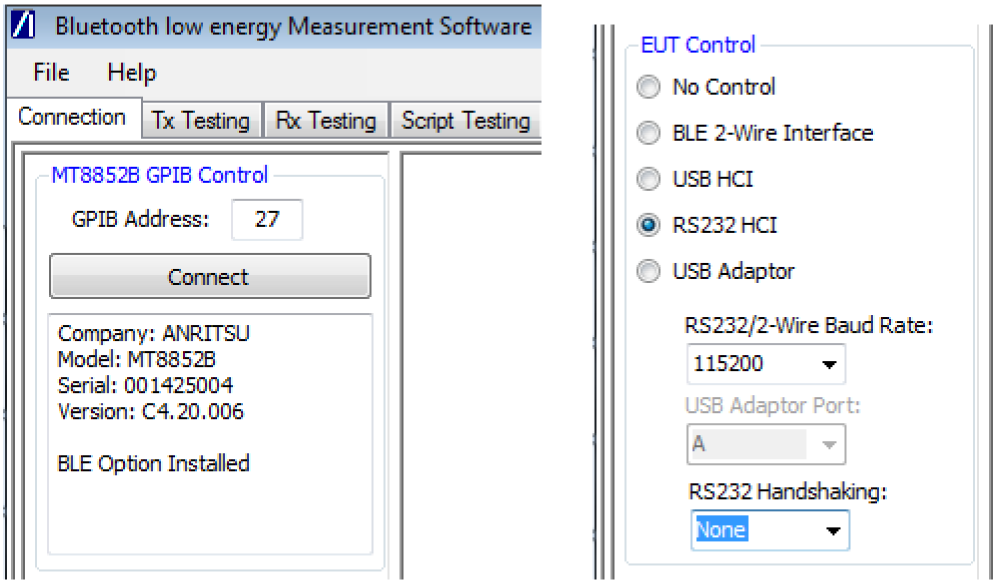


Figure 7: Anritsu BLE Measurement Software Connection Settings

Here below in Figure 8 the Comm Tunnel Tool settings: In this example COM24 is the PC's Virtual serial port at which the BT Tester is connected, and COM 28 is the DA1468x Pro-Kit board virtual COM port for UART. Set the Baudrate for both Endpoints to 115200 baud, and flow control to none.

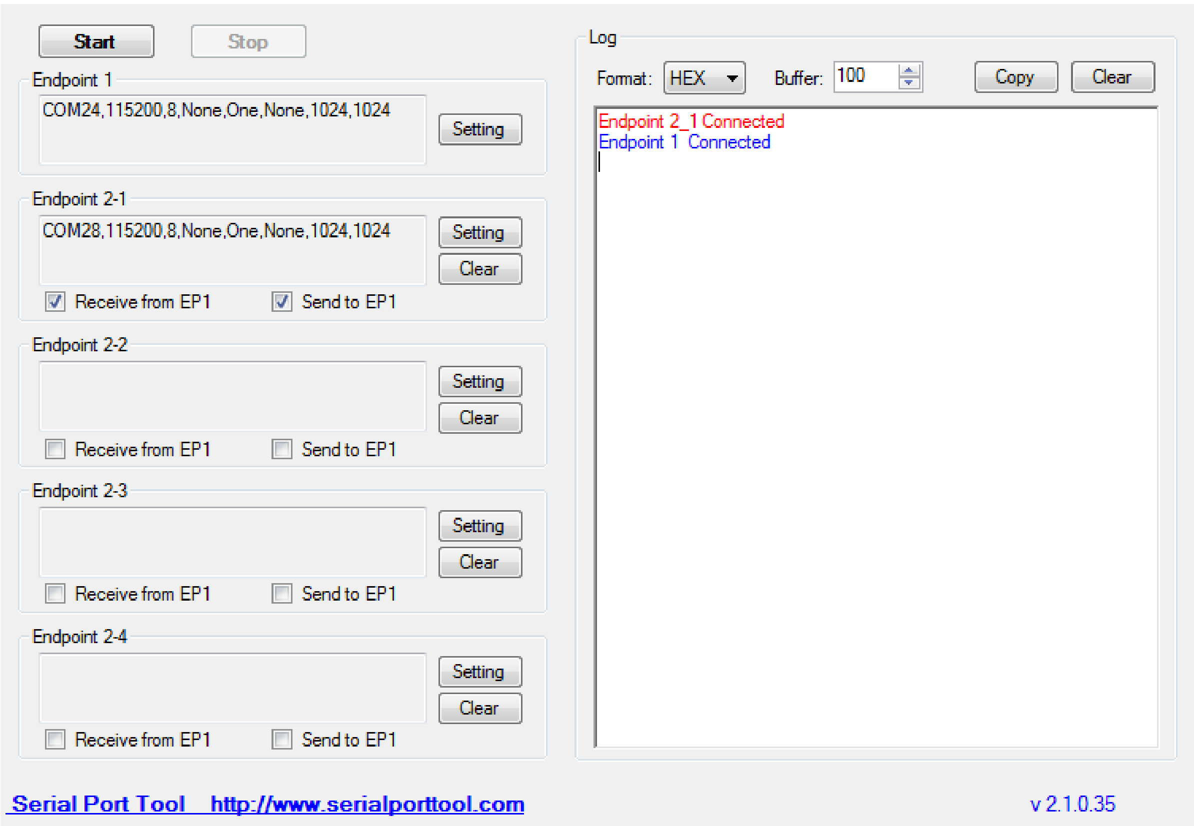


Figure 8: Comm Tunnel Tool Settings

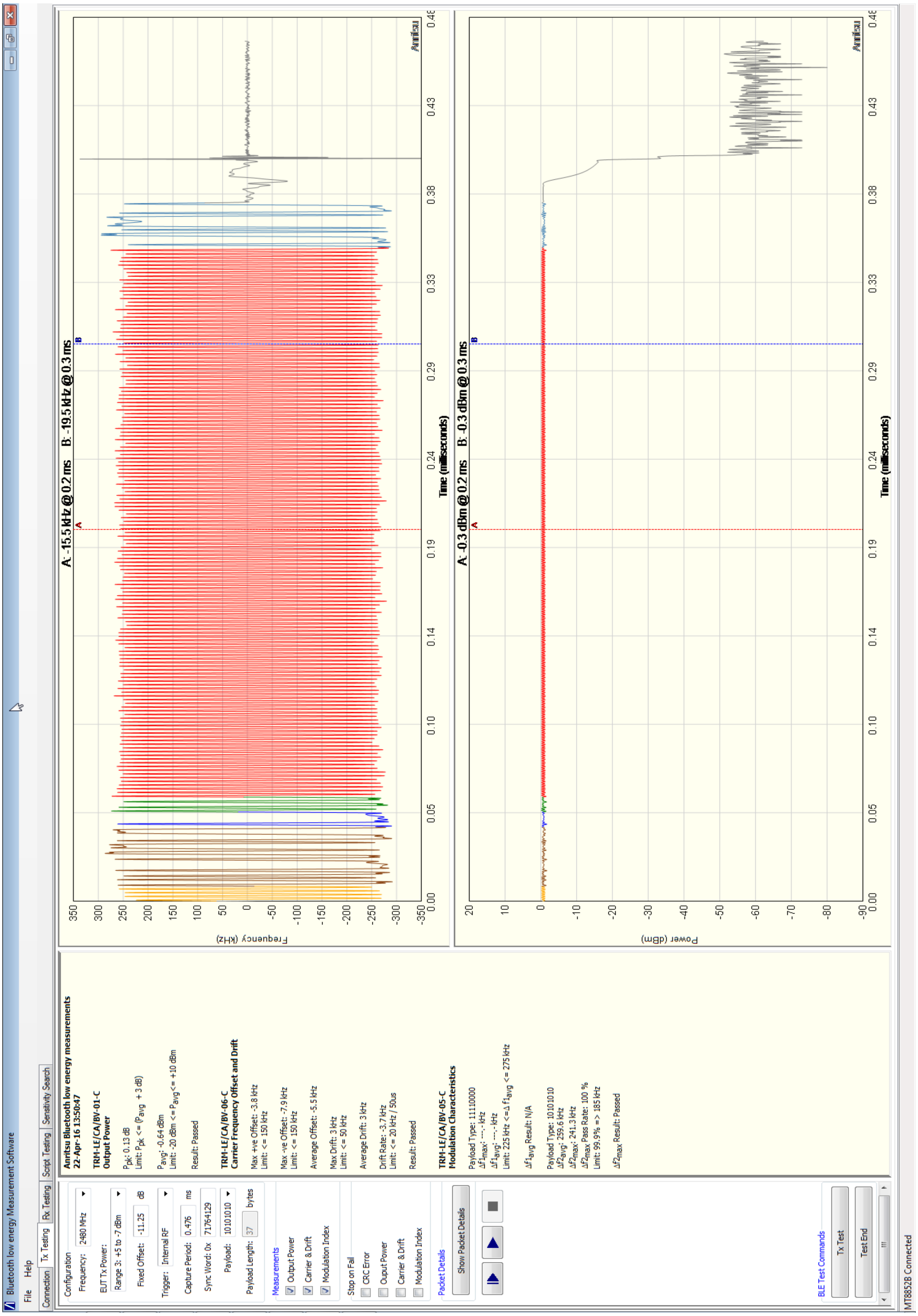


Figure 9: Anritsu Tx Testing results

4.2.3 Setting up DTM via the tester's USB port using CMW270

An R&S CMW270 Communication Tester having BLE license and BLE Signalling option (KS611) is taken as example. For the hardware setup, please refer to Figure 3.
In order to be able to make use of the DA1468x Pro-Kit on-board FTDI chip, make sure the FTDI USB-to-Serial driver for the virtual com port is installed on the CMW270. Then the signalling commands from the CMW270 can enter the DA1468x UART pins over USB and the FTDI chip, and vice versa of course for the data sent back by the DA1468x to the CMW270.
Check whether the driver is installed properly by opening the Windows OS Device Manager, as shown in Figure 10. For BLE DTM signalling, the first listed USB Serial COM port of the installed pair must be used in the BTLE connection settings. In below example, COM3 should be selected.

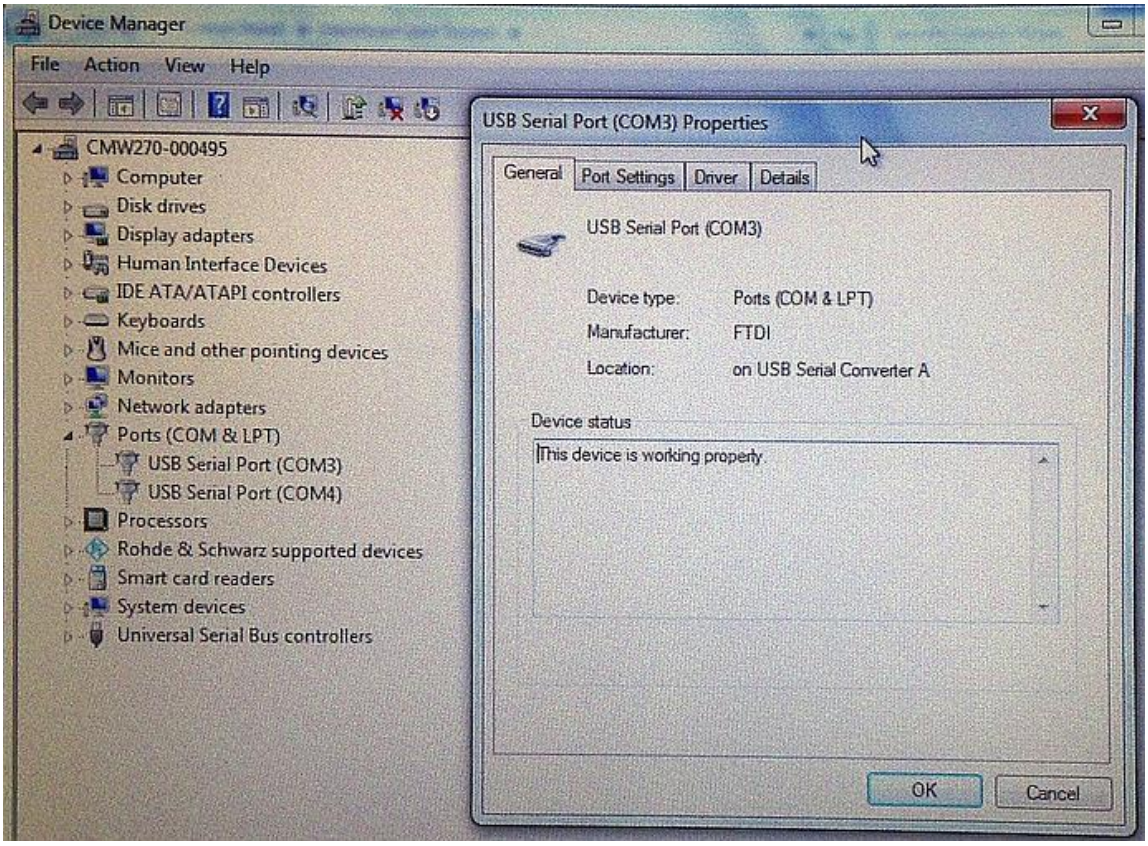


Figure 10: CMW270 installed virtual COM port.

In Figure 11 Connection menu following must be set in order to communicate over a virtual USB COM port with the DA1468x device:

- USB to RS232 adapter as HW Interface.
- HCI as EUT Communication Protocol.
- The right Virtual COM Port: COM3 in this example.
- Baudrate: 115200
- No flow control: set Protocol to None. No Parity, 1 Stop bit.
- Select Reset EUT.

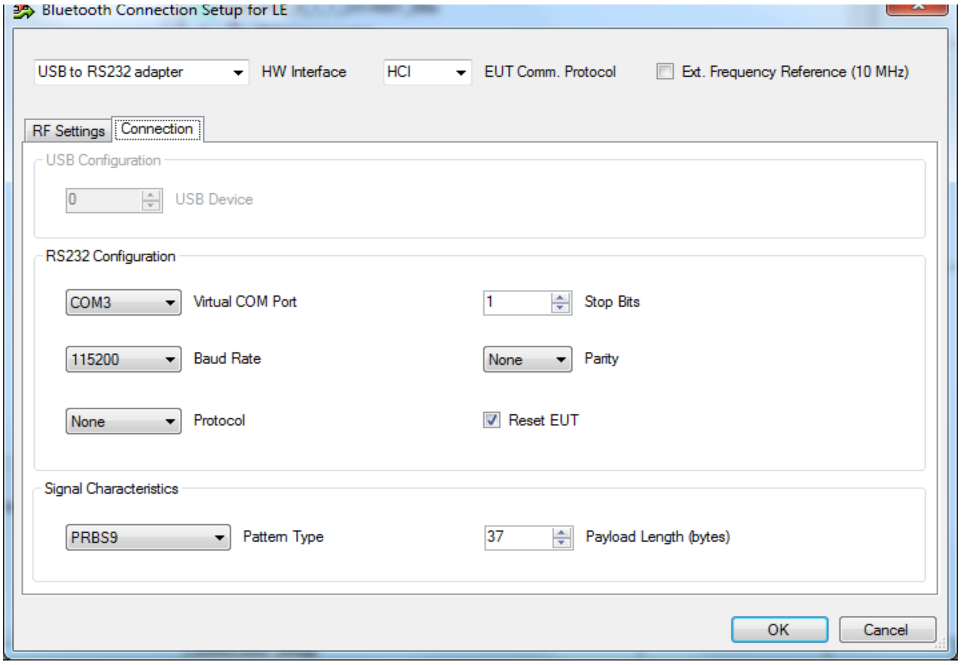


Figure 11: CMW270 BTLE Connection Settings

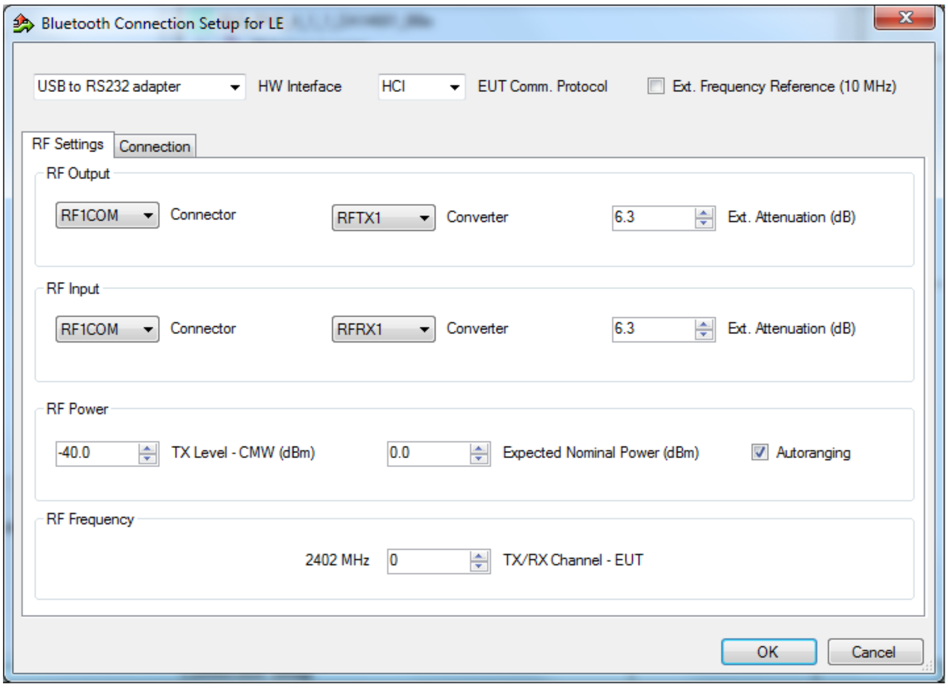


Figure 12: CMW270 RF-settings

In Figure 12 RF Settings menu, the used RF output must be set, and the amount of external signal attenuation can be given. This attenuation can consist of an applied RF-attenuator and RF-cable losses. After connecting the DA1468x Pro-Kit board to the CMW270, and downloading the test software (PLT_FW.bin or PLT_FW.hex) to the DA14568x via UART or JLINK, the testing can begin. Downloading the test SW from the CMW270 via USB/UART should be possible as well. Alternatively, the PLT_FW project can be compiled for QSPI Flash usage and programmed in the Flash. After selecting the Tx and the Rx tests to be executed, the CMW270 starts to check the BTLE connection and then starts executing the selected tests. See below Figures 13, 14 and 15.

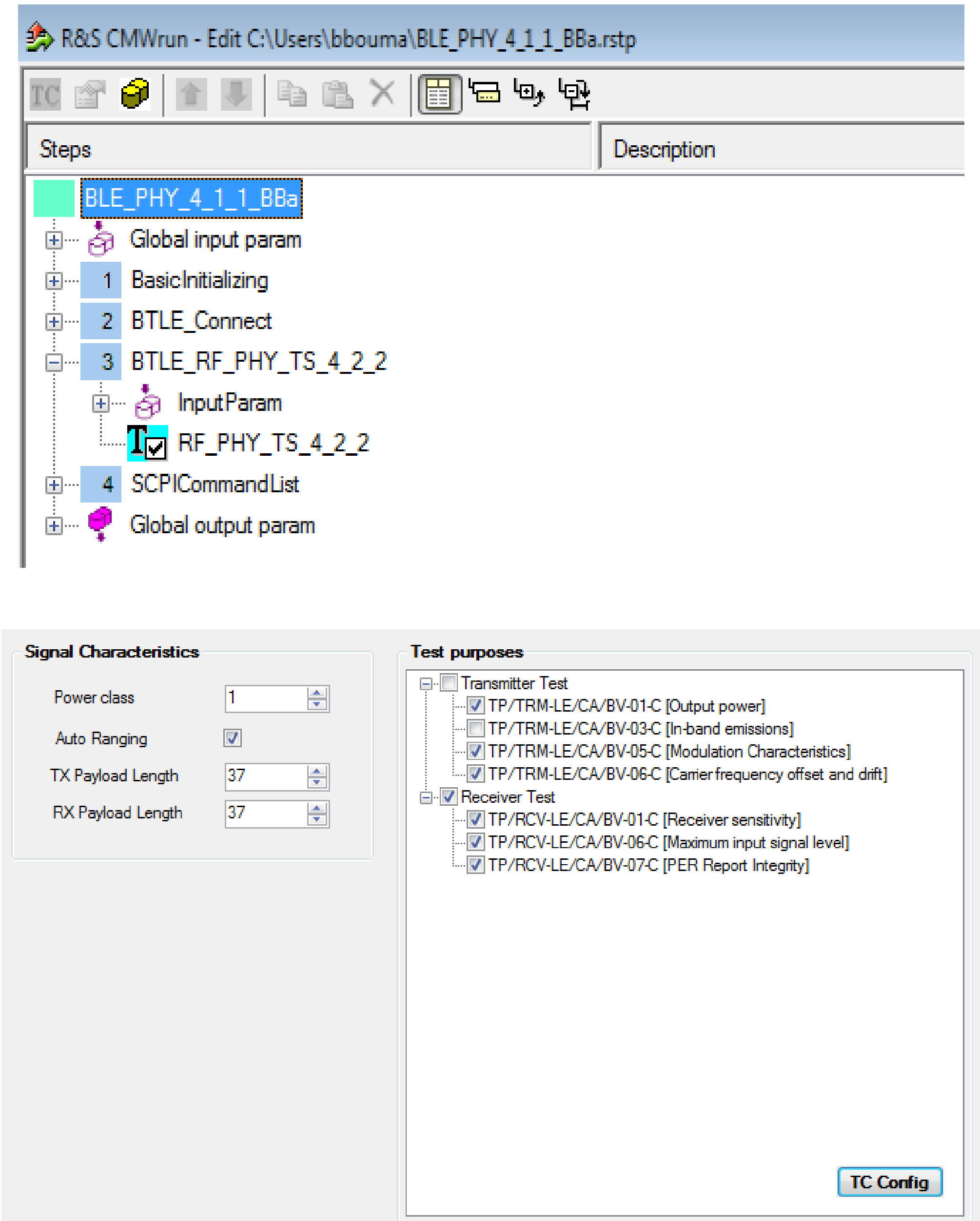
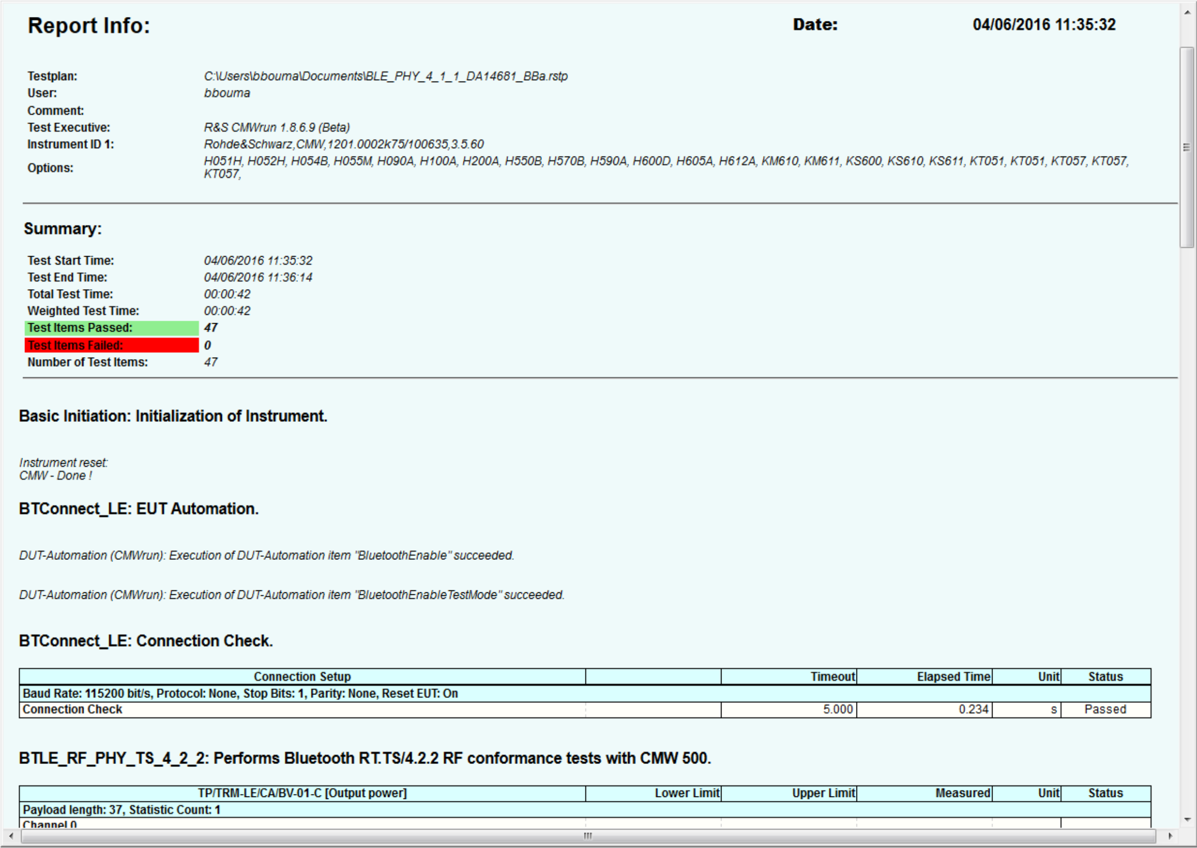


Figure 13: Selected Tx and Rx tests in BTLE_RF_PHY_TS_4_2_2.



Report Info:

Date: 04/06/2016 11:35:32

Testplan:	C:\Users\bbouma\Documents\BLE_PHY_4_1_1_DA14681_BBa.rstp
User:	bbouma
Comment:	
Test Executive:	R&S CMWrun 1.8.6.9 (Beta)
Instrument ID 1:	Rohde&Schwarz,CMW,1201.0002k75/100635,3.5.60
Options:	H051H, H052H, H054B, H055M, H090A, H100A, H200A, H550B, H570B, H590A, H600D, H605A, H612A, KM610, KM611, KS600, KS610, KS611, KT051, KT057, KT057, KT057,

Summary:

Test Start Time:	04/06/2016 11:35:32
Test End Time:	04/06/2016 11:36:14
Total Test Time:	00:00:42
Weighted Test Time:	00:00:42
Test Items Passed:	47
Test Items Failed:	0
Number of Test Items:	47

Basic Initiation: Initialization of Instrument.

Instrument reset:
CMW - Done !

BTConnect_LE: EUT Automation.

DUT-Automation (CMWrun): Execution of DUT-Automation item "BluetoothEnable" succeeded.

DUT-Automation (CMWrun): Execution of DUT-Automation item "BluetoothEnableTestMode" succeeded.

BTConnect_LE: Connection Check.

Connection Setup		Timeout	Elapsed Time	Unit	Status
Baud Rate: 115200 bit/s, Protocol: None, Stop Bits: 1, Parity: None, Reset EUT: On					
Connection Check		5.000	0.234	s	Passed

BTLE_RF_PHY_TS_4_2_2: Performs Bluetooth RT.TS/4.2.2 RF conformance tests with CMW 500.

TP/TRM-LE/CA/BV-01-C [Output power]	Lower Limit	Upper Limit	Measured	Unit	Status
Payload length: 37, Statistic Count: 1					
Channel 0					

Figure 14: CMW270 Reporting

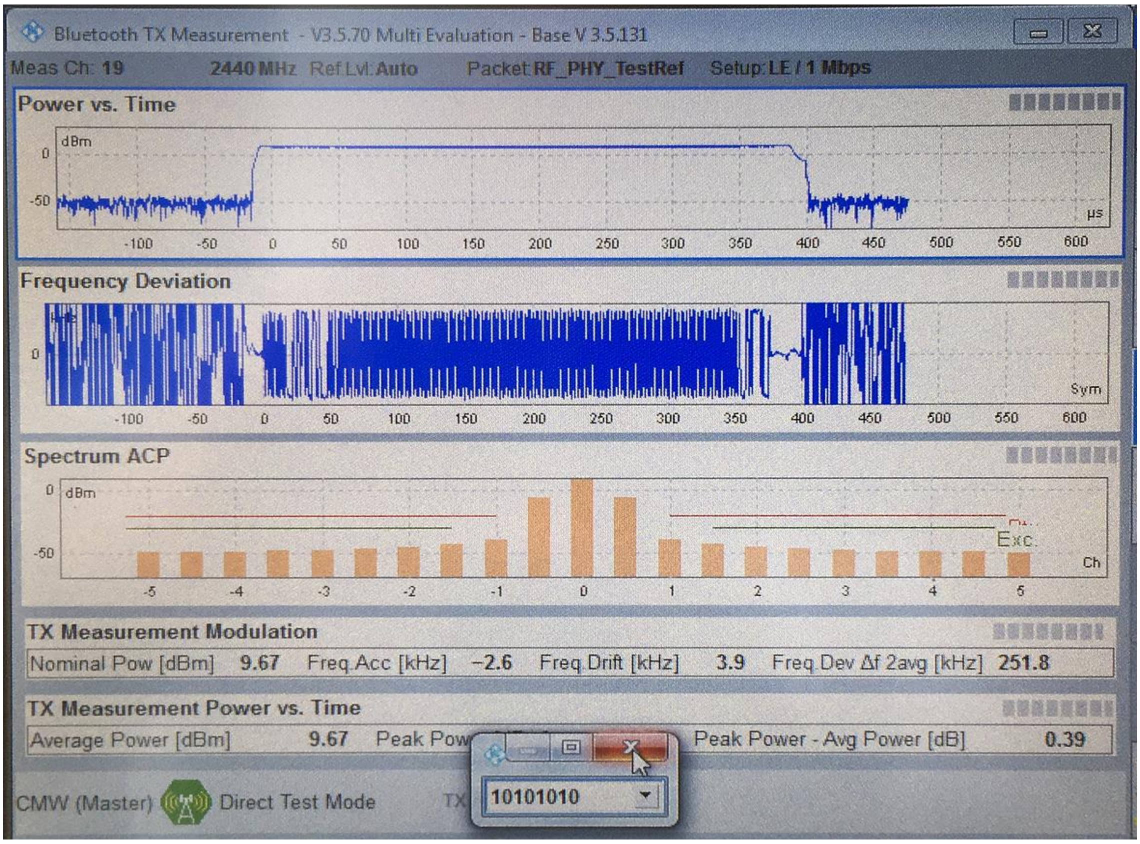


Figure 15: Tx measurement on the R&S CMW270's display

4.2.4 Setting up LitePoint IQxel-M BLE Tester

The LitePoint IQxel-M universal testers are targeted for fast production line testing.
The setup is simple: a RF-output (e.g. Port RF1) of the LitePoint tester is connected to the DA1468x RF port, e.g. to the small RF-connector on the BLE chip's daughterboard.
Please use a good quality coaxial cable for this. Software running on the PC controls the LitePoint IQxel-M tester and the DUT: the DA1468x application.

The DA1468x BLE device must be set into Direct Test Mode (DTM). For this DTM mode, the appropriate DTM software must be loaded into the DUT: *PLT_FW.bin,* which can be created from *\projects\dk_apps\reference_designs\plt_fw* found in the *DA1468x_SDK_BTLE* folder. The DTM binary can also be requested from the Dialog Semiconductor LPC Customer Support team.

The testing procedure works as follows: for DA1468x Rx testing the IQxel-M tester sends out a known number of packets, the Terminal program or SmartSnippets ToolBox RF-Master reports the number of packets correctly received by the DA1468x and then a packet error rate (PER) calculation can be executed.
For Tx testing the DUT is set to transmit a modulated Tx signal, which then is analysed and displayed by the IQxel-M tester.

Unlike the R&S CBT or the Anritsu MT8852B tester, there is no direct signalling between the IQxel-M tester and the DUT. The DUT must be controlled by the PC using LitePoint automated IQfact+ application, SmartSnippets ToolBox RF-Master or a Terminal program. See also Section 4.3 and Section 4.4.

4.2.4.1 IQxel-M Vector Signal Generator

In this example the DA1468x is controlled using SmartSnippets ToolBox RF-Master or Terminal program and the IQxel-M tester is controlled with the on-tester Web GUI and a browser on the PC.
For the RF1 Port to be used as BLE RF output, as shown in Figure 4, it must be configured as follows in the *Port Configuration* menu of the *IQxel* program: Port RF1 => VSG. See Figure 16.

In the Vector Signal Generator (VSG) tab, the desired generator waveform file can be selected and loaded using the Load button in the Waveforms section. Once the waveform is loaded, the generator can run continuously or for a certain number of waveforms or packets. The frequency/channel and the RF output level of the tester can be set in the VSG1 Settings section. See Figure 17.

In Figure 17 the VSG window is shown. In the Waveforms section one can switch the RF on and off by the RF(on) and RF(off) buttons. The waveform can be started clicking the Play button.
Ending the waveform can be either done manually (Stop) or after a given number of counts (Count).

The IQxel-M analyser can be used to analyse waveforms that the VSG will transmit. These waveforms are loaded into the analyser by <right-clicking> on the black arrow in the white square located in the top bar, and selecting the "LOAD" option. If measurement windows are enabled with VSA1->Results tab, the waveform will automatically be analysed and the results shown.
This waveform window is not automatically updated when selecting another waveform file for the VSG, so the new selected waveform must be re-loaded by the user if it's wanted to display what will be sent to the DUT.

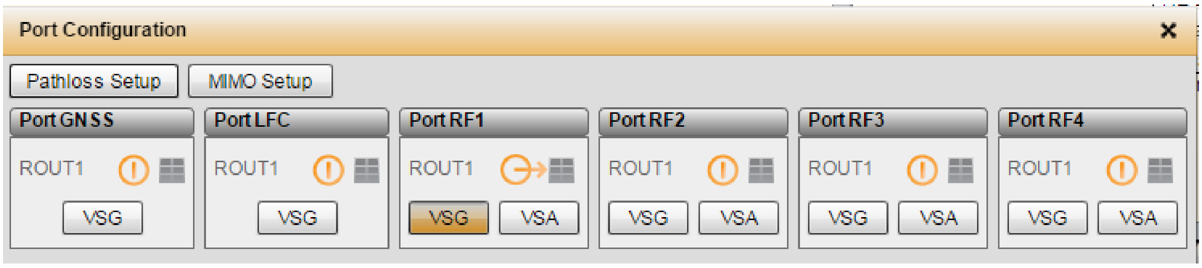


Figure 16: IQxel Port Configuration – using Port RF1 for VSG

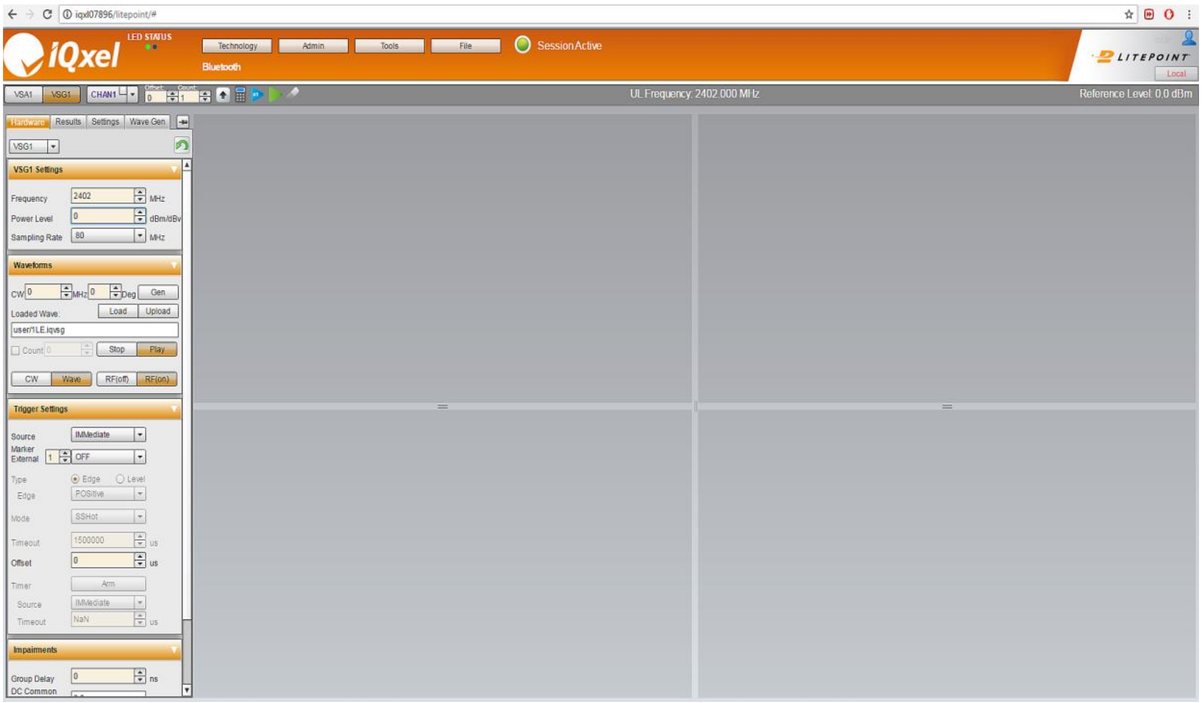


Figure 17: IQxel Vector Signal Generator window

Transmitting the selected waveform starts when the *RF(on)* button is pressed, and when a number is selected in the *Count* section, it stops transmitting when this number of waveforms have been sent.

Procedure:

1. Send the *start_pkt_rx* or *start_pkt_rx_stats* HCI command to the DUT.
2. Press the RF (on) button in the IQxel program.
3. After finishing, send the *stoptest* or the *stop_pkt_rx_stats* HCI command to the DUT.

The transmitted number of packets is known. The received number of correct packets can be read from the DUT using for instance a terminal program and sending the Rx with Statistics HCI command. Instead of sending HCI commands using a terminal program, RF-Master in the SmartSnippets ToolBox can be used to start and stop receiving.
Please, refer to sections 4.3 or 4.4 and the Appendix.

4.2.4.2 IQxel-M Vector Signal Analyzer

The Vector Signal Analyzer (VSA) can be used to analyse the Tx output signal of the DUT. For the RF1 Port to be used as RF input, as shown in Figure 4, it must be configured as follows in the *Port Configuration* menu of the *IQxel* program: Port RF1=> VSA. See Figure 18.

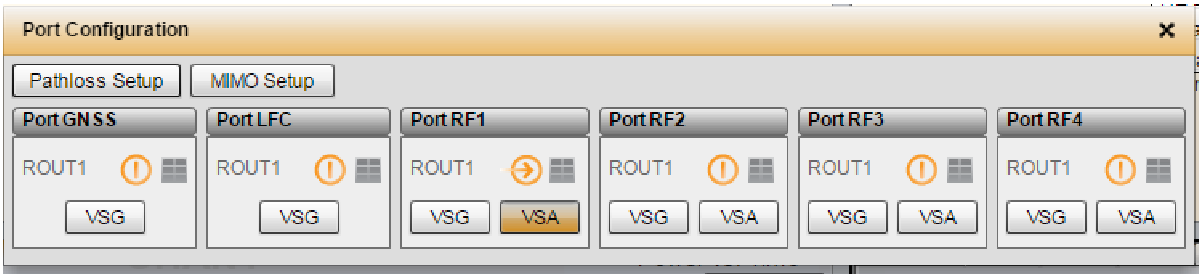


Figure 18: IQxel Port Configuration – using Port RF1 for VSA

The DUT (DA1468x) again is in Direct Test Mode, but now as transmitter, transmitting either continuously or a certain number of packets. This can be set by sending the *cont_pkt_tx* or *pkt_tx* HCI command. Use these HCI commands also for setting the desired frequency and payload type.

The VSA screen reports the Peak Power, Average Power, Frequency Offset, Delta F1, Delta F2 modulation figures, etc. There are also a 'Spectral Mask' window, and e.g. a 'LE Delta F2 Max vs. Time' can be displayed. For Delta F1 analysis to be done, the DUT's packet pattern must be set to '11110000', for Delta F2 the packet pattern '10101010' must be applied as shown in the analyser's left-hand bottom window in Figure 19.

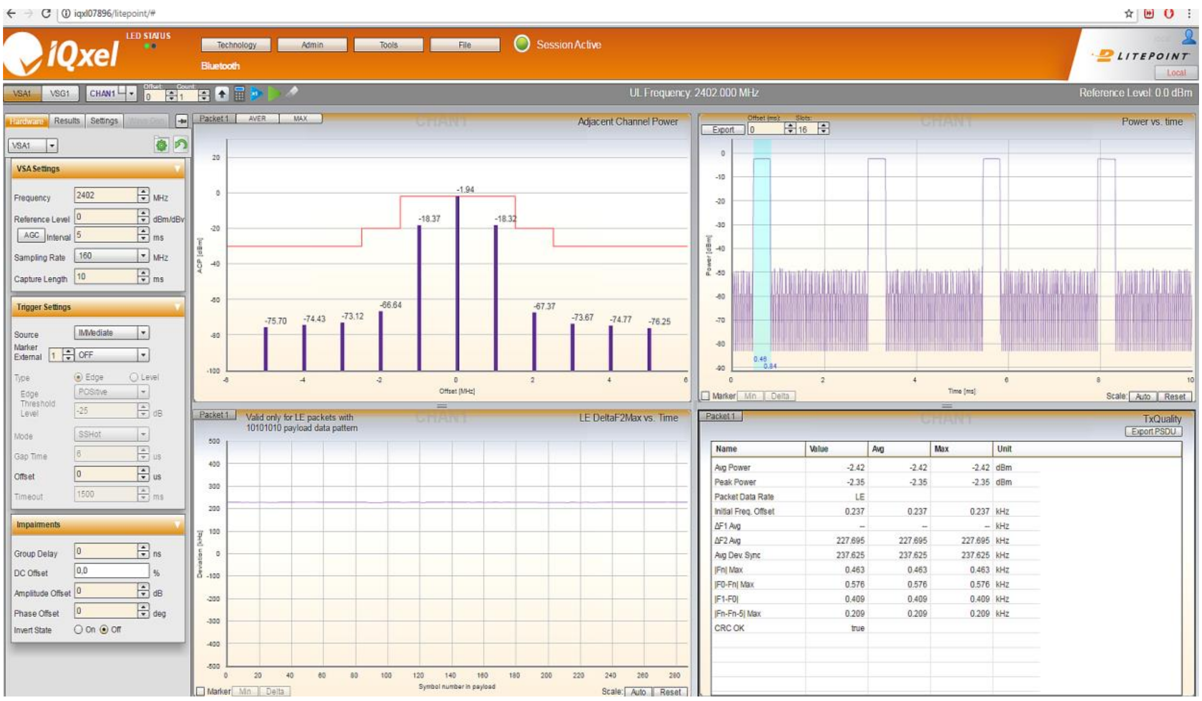


Figure 19: Vector Signal Analyser - analysing DA1468x's TX output signal

The received signal as shown in the right-hand top window can be exported as a waveform file.

4.2.5 LitePoint Bluetooth Advanced Measurement Solution

In some cases, it is preferable to use over-the-air measurement method instead of DTM. The LitePoint Bluetooth Advanced solution provides a Bluetooth Low Energy over-the-air measurement solution for BLE peripheral and beacon devices. No wiring or cables are needed. There is no COM port required on the DUT and the RF signal can be radiated between the tester and the DUT using antennas as shown below, or the RF signal can be conducted with a RF coax cable. The DA1468x SoC is configured with standard commercial firmware, not DTM, and the DUT is set to start advertising. The base equipment for such a BLE measurement solution is LitePoint's IQxel-M tester with LitePoint BT Advanced measurement application. Please refer to Figure 20.

Figure 20: Bluetooth Advanced Measurement System

Next to the IQxel-M, LitePoint's IQfact+ test software and a RF-enclosure, a shielded box, will be needed.

Measurement setup: the system is setup as shown. The test application is LitePoint's BT Advanced IQfact+ which provides parametric TX and RX testing. The application is fully automated and allows user to configure test flows and define pass/fail limits. Measurement results are shown on-screen as well as logged to text and .csv files. The DUT (DA1468x) is configured with commercial firmware and set to start advertising. To minimize test time, the advertising period should be set as small as possible. 80ms to 100ms advertising is optimal.

Measurement Methodology: RF measurements are performed on an advertising BLE device, on channels 37, 38 and 39. These channels have a frequency of respectively 2402 MHz, 2426 MHz and 2480 MHz. Because other advertising BLE devices use these very same channels, a shielded RF enclosure is needed for making sure the measured DUT is not interfered by those other advertising BLE devices.

What can be measured:
- TX power
- TX modulation, frequency offset and frequency drift
- TX adjacent channel power
- RX packet error rate
- RX sensitivity

For beacon devices which transmit, but do not receive, the test application can be configured to perform the TX tests as shown above. For peripheral devices which transmit and receive, the test application can be configured to perform both TX and RX tests. In manufacturing a test flow with TX Power, TX Quality, and RX PER can be run in as little as 3-5 seconds and provides parametric data with pass/fail limit checking on all 3 advertising channel.
For design verification, a complete flow which includes all the TX and RX tests, including RX sensitivity can be performed. Please contact LitePoint for more details on this.

4.3 Sending HCI commands over UART to the DUT

For putting the DUT in DTM Tx or DTM Rx mode, HCI commands as given in [2] and shown in the Appendix must be sent to the BLE device. The PLT_FW DTM software must be loaded in the device.

Two examples will be given, one for Un-modulated CW Tx and for Rx with Statistics (reporting the number of received packets).

In the examples the Serial Capture Tool RealTerm v.2.0 is used. The ‘Display as’ in the Display tab must be set to ‘Hex (space)’, and sending must be done by pressing ‘Send Numbers’. Please refer to Figure 21. The port settings must be set as follows:

Baudrate: 115200
Parity: none
Data Bits: 8
Stop Bits: 1
Hardware Flow Control: None.

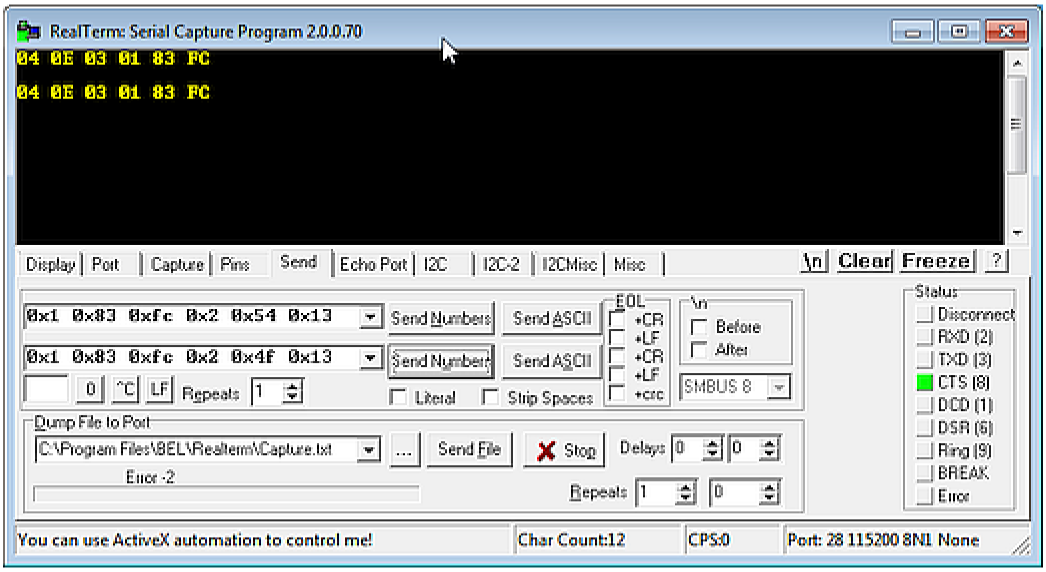


Figure 21: RealTerm 'Send' and ‘Receive’ window.

Tx un-modulated CW command:

Byte Description	Value
HCI Command Packet	0x01
Command Opcode LSB	0x83
Command Opcode MSB	0xFC
Parameter Length	0x02
Operation	0x4F: OFF 0x54: unmodulated TX 0x52: unmodulated RX
Frequency	= (F – 2402) / 2, where F ranges from 2402 MHz to 2480 MHz. Range: 0x00 – 0x27.

The command for Tx un-modulated at Channel 19 (2440 MHz) is: 0x1 0x83 0xFC 0x2 0x54 0x13.
For stopping Tx un-modulated this command must be given: 0x1 0x83 0xFC 0x2 0x4F 0x13.
In both cases the BLE device responds with: 0x4 0x0E 0x03 0x1 0x83 0xFC.

Note: stopping the test is required before any new test command can be given!

Start_Pkt_Rx_Stats command:

Byte Description	Value
HCI Command Packet	0x01
Command Opcode LSB	0x81
Command Opcode MSB	0xFC
Parameter Length	0x01
Frequency	= (F – 2402) / 2, where F ranges from 2402 MHz to 2480 MHz. Range: 0x00 – 0x27.

The command for Rx with Statistics at Channel 0 (2402 MHz) is: 0x01 0x81 0xFC 0x01 0x00.
The DUT will respond with: 0x04 0x0E 0x03 0x01 0x81 0xFC.

The 'Stoptest' command will return only the number of received packets, whereas the 'Stop_Pkt_Rx_Stats' command will return the number of received packets, the number of packets with Sync errors, the number of packets with CRC errors and the RSSi value.
The latter command will be shown in below example.

Stop_Pkt_Rx_Stats command: 0x01 0x82 0xFC 0x00

Byte Description	Value
HCI Command Packet	0x01
Command Opcode LSB	0x82
Command Opcode MSB	0xFC
Parameter Length	0x00

Return Message from DUT:

Byte Description	Value
HCI Event Packet	0x04
Event Code	0x0E
Parameter Length	0xb
Num_HCI_Command_Packets	0x01
Command_Opcode LSB	0x82
Command_Opcode MSB	0xFC
Number of **total received** packets LSB	0xXX
Number of **total received** packets MSB	0xXX
Number of received packets with sync errors LSB	0xXX
Number of received packets with sync errors MSB	0xXX
Number of received packets with CRC errors LSB	0xXX
Number of received packets with CRC errors MSB	0xXX
RSSI LSB	0xXX
RSSI MSB	0xXX

In this example the BT Tester sent 1500 packets at 2402 MHz, -92 dBm, Dirty Transmitter On.

The received data from the BLE device:
Number of total received packets is: 0x05 0xD3 (1491)
Number of Packets with Sync errors: 0x00 0x00 (0)
Number of packets with CRC errors: 0x00 0x90 (144)
Please refer to Figure 22.

The total number of received correct packets equals 1491 – 144 = 1347.
The packet error rate can be calculated from these values: in this 1500 packets example it is:
100% * (1500 – 1347) / 1500 = 10.2 %.

The reported RSSi value is: 0x00 0x24: 36 or about -94 dBm.

Please note that for all reported data the LSB is transferred first, before the MSB.

Received message in RealTerm:

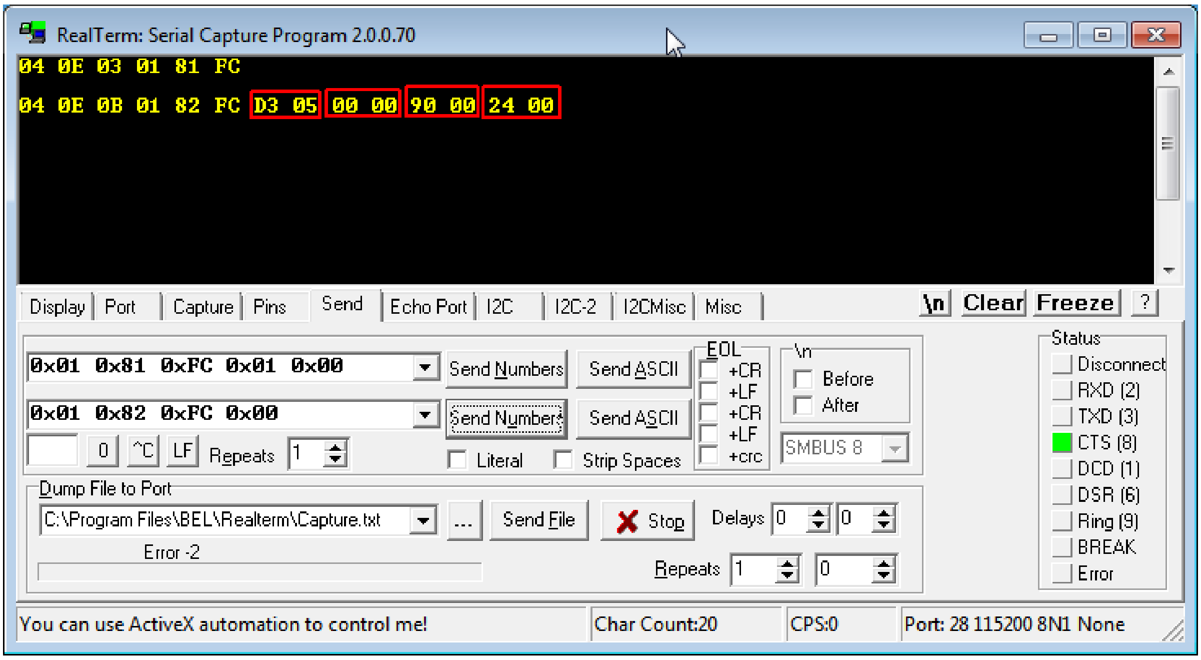


Figure 22: Received Message for RX with Statistics

4.4 Using SmartSnippets ToolBox RF-Master to control the DA1468x

RF-Master in SmartSnippets Toolbox can be used to set the DA1468x in TX mode, and can also be used to read the number of received packets sent by e.g. a LitePoint tester or another DA1468x Pro-Kit or application.

The Tx modes include e.g. Continuous Tx and Un-modulated Tx. The Rx mode includes reading the RX statistics, e.g. the number of received packets and the RSSI value. The latter is shown in below Figure 23: received number of packets: 1482, packets having CRC error: 215. Rssi = -94.86 dBm.

Procedure: Press Start button in RF-Master, send the desired number of packets, press Stop button.

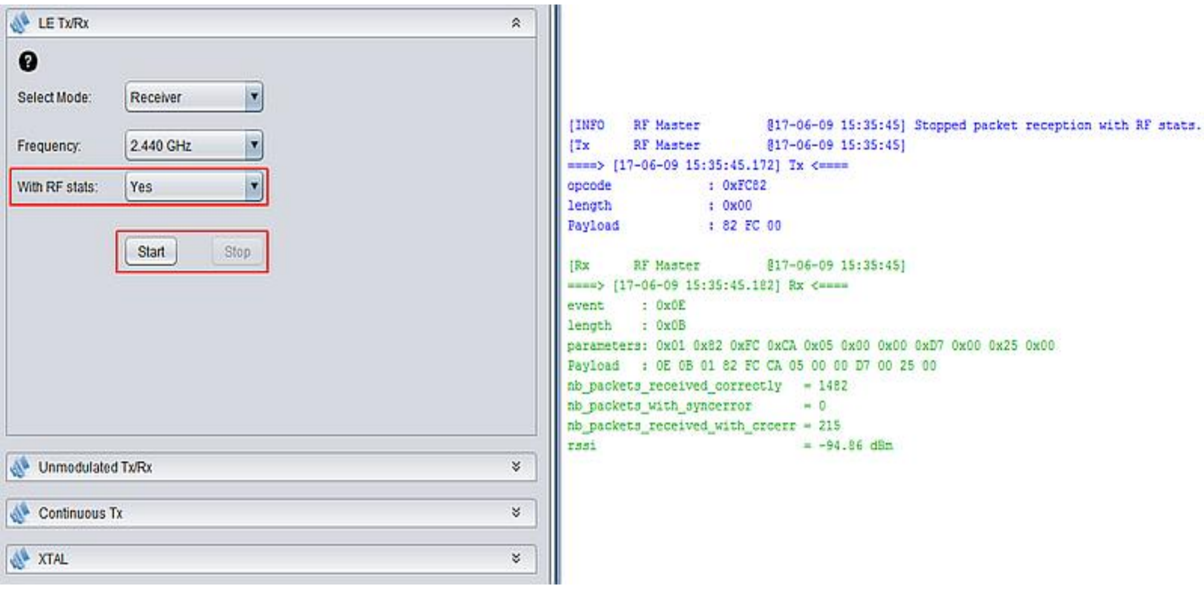


Figure 23: SmartSnippets RF-Master RX Statistics

Appendix: listing of most common HCI commands for RF testing

cont_pkt_tx

	Byte Description	Value
Command Format	**Byte Description**	**Value**
	HCI Command Packet	0x01
	Command Opcode LSB	0x1E
	Command Opcode MSB	0x20
	Parameter Length	0x03
	Frequency	= (F – 2402) / 2, where F ranges from 2402 MHz to 2480 MHz. Range: 0x00 – 0x27.
	Data Length	0x01-0x25: Length in bytes of payload data in each packet
	Payload Type	0x00: Pseudo-Random bit sequence 9 0x01: Pattern of alternating bits '11110000' 0x02: Pattern of alternating bits '10101010' 0x03: Pseudo-Random bit sequence 15 0x04: Pattern of All '1' bits 0x05: Pattern of All '0' bits 0x06: Pattern of alternating bits '00001111' 0x07: Pattern of alternating bits '0101'
Return Message	**Byte Description**	**Value**
	HCI Event Packet	0x04
	Event Code	0x0E
	Parameter Length	0x04
	Num_HCI_Command_Packets	0x01
	Command_Opcode LSB	0x1E
	Command_Opcode MSB	0x20
	Status	0x00: command succeeded. 0x01 – 0xFF: command failed. See Volume 2, Part D -Error Codes in Bluetooth 4.0 specification for a list of error codes and descriptions.

pkt_tx

	Byte Description	Value
Command Format	**Byte Description**	**Value**
	HCI Command Packet	0x01
	Command Opcode LSB	0x80
	Command Opcode MSB	0xFC
	Parameter Length	0x05
	Frequency	= (F – 2402) / 2, where F ranges from 2402 MHz to 2480 MHz. Range: 0x00 – 0x27.
	Data Length	0x01-0x25 Length in bytes of payload data in each packet
	Payload Type	0x00: Pseudo-Random bit sequence 9 0x01: Pattern of alternating bits '11110000' 0x02: Pattern of alternating bits '10101010' 0x03: Pseudo-Random bit sequence 15 0x04: Pattern of All '1' bits 0x05: Pattern of All '0' bits 0x06: Pattern of alternating bits '00001111' 0x07: Pattern of alternating bits '0101'
	Number of packets to be sent LSB	0xXX
	Number of packets to be sent MSB	0xXX
Return Message	**Byte Description**	**Value**
	HCI Event Packet	0x04
	Event Code	0x0F
	Parameter Length	0x03
	Num_HCI_Command_Packets	0x01
	Command_Opcode LSB	0x80
	Command_Opcode MSB	0xFC

unmodulated OFF / TX / RX

	Byte Description	Value
Command Format	**Byte Description**	**Value**
	HCI Command Packet	0x01
	Command Opcode LSB	0x83
	Command Opcode MSB	0xFC
	Parameter Length	0x02
	Operation	0x4F: OFF 0x54: unmodulated TX 0x52: unmodulated RX
	Frequency	= (F – 2402) / 2, where F ranges from 2402 MHz to 2480 MHz. Range: 0x00 – 0x27.
Return Message	**Byte Description**	**Value**
	HCI Event Packet	0x04
	Event Code	0x0E
	Parameter Length	0x03
	Num_HCI_Command_Packets	0x01
	Command_Opcode LSB	0x83
	Command_Opcode MSB	0xFC

start_pkt_rx		
Command Format	Byte Description	Value
	HCI Command Packet	0x01
	Command Opcode LSB	0x1D
	Command Opcode MSB	0x20
	Parameter Length	0x01
	Frequency	= (F – 2402) / 2, where F ranges from 2402 MHz to 2480 MHz. Range: 0x00 – 0x27.
Return Message	Byte Description	Value
	HCI Event Packet	0x04
	Event Code	0x0E
	Parameter Length	0x04
	Num_HCI_Command_Packets	0x01
	Command_Opcode LSB	0x1D
	Command_Opcode MSB	0x20
	Status	0x00: Command succeeded. 0x01 – 0xFF: Command failed. See Volume 2, Part D -Error Codes in Bluetooth 4.0 specification for a list of error codes and descriptions.

start_pkt_rx_stats		
Command Format	Byte Description	Value
	HCI Command Packet	0x01
	Command Opcode LSB	0x81
	Command Opcode MSB	0xFC
	Parameter Length	0x01
	Frequency	= (F – 2402) / 2, where F ranges from 2402 MHz to 2480 MHz. Range: 0x00 – 0x27.
Return Message	Byte Description	Value
	HCI Event Packet	0x04
	Event Code	0x0E
	Parameter Length	0x03
	Num_HCI_Command_Packets	0x01
	Command_Opcode LSB	0x81
	Command_Opcode MSB	0xFC

stop_pkt_rx_stats		
Command Format	Byte Description	Value
	HCI Command Packet	0x01
	Command Opcode LSB	0x82
	Command Opcode MSB	0xFC
	Parameter Length	0x00
Return Message	Byte Description	Value
	HCI Event Packet	0x04
	Event Code	0x0E
	Parameter Length	0xb
	Num_HCI_Command_Packets	0x01
	Command_Opcode LSB	0x82
	Command_Opcode MSB	0xFC
	Number of received packets LSB	0xXX
	Number of received packets MSB	0xXX
	Number of received packets with sync errors LSB	0xXX
	Number of received packets with sync errors MSB	0xXX
	Number of received packets with CRC errors LSB	0xXX
	Number of received packets with CRC errors MSB	0xXX
	RSSI LSB	0xXX
	RSSI MSB	0xXX

stoptest

	Byte Description	Value
Command Format	**Byte Description**	**Value**
	HCI Command Packet	0x01
	Command Opcode LSB	0x1F
	Command Opcode MSB	0x20
	Parameter Length	0x00
Return Message	**Byte Description**	**Value**
	HCI Event Packet	0x04
	Event Code	0x0E
	Parameter Length	0x06
	Num_HCI_Command_Packets	0x01
	Command_Opcode LSB	0x1F
	Command_Opcode MSB	0x20
	Status	0x00: Command succeeded. 0x01-0xFF: Command failed. See Volume 2, Part D -Error Codes in Bluetooth 4.0 specification for a list of error codes and descriptions.
	Number of packets received LSB	0xXX
	Number of packets received MSB	0xXX

start_cont_tx

	Byte Description	Value
Command Format	**Byte Description**	**Value**
	HCI Command Packet	0x01
	Command Opcode LSB	0x84
	Command Opcode MSB	0xFC
	Parameter Length	0x02
	Frequency	= (F – 2402) / 2, where F ranges from 2402 MHz to 2480 MHz. Range: 0x00 – 0x27.
	Payload Type	0x00: Pseudo-Random bit sequence 9 0x01: Pattern of alternating bits '11110000' 0x02: Pattern of alternating bits '10101010' 0x03: Pseudo-Random bit sequence 15 0x04: Pattern of All '1' bits 0x05: Pattern of All '0' bits 0x06: Pattern of alternating bits '00001111' 0x07: Pattern of alternating bits '0101'
Return Message	**Byte Description**	**Value**
	HCI Event Packet	0x04
	Event Code	0x0E
	Parameter Length	0x03
	Num_HCI_Command_Packets	0x01
	Command_Opcode LSB	0x84
	Command_Opcode MSB	0xFC

Revision history

Revision	Date	Description
1.0	22-Apr-2016	Initial version.
1.1	22-May-2016	Ch 4.3: Return Message description for Stoptest and Stop_pkt_rx_stats adapted to make clearer what has been read.
1.2	01-Sep-2016	Updated the PLT_FW path in the DA1468x SDK in section 4.1
1.3	26-Jun-2017	Updated Figure 1 Replaced LitePoint IQ2015 with IQxel-M tester in section 4.2 & 4.2.4 Added LitePoint BT Advanced solution in section 4.2.5 Adding SmartSnippets Toolbox RF-master example in section 4.4
1.4	10-Aug-2017	Completed the list of HCI commands in the appendix.
1.5	14-Sep-2017	Updated for DA14682-00/DA14683-00. Figure 4 corrected.
1.6	17-Feb-2022	Updated logo, disclaimer, copyright.

Status definitions

Status	Definition
DRAFT	The content of this document is under review and subject to formal approval, which may result in modifications or additions.
APPROVED or unmarked	The content of this document has been approved for publication.

RoHS Compliance

Dialog Semiconductor complies to European Directive 2001/95/EC and from 2 January 2013 onwards to European Directive 2011/65/EU concerning Restriction of Hazardous Substances (RoHS/RoHS2).
Dialog Semiconductor's statement on RoHS can be found on the customer portal https://support.diasemi.com/. RoHS certificates from our suppliers are available on request.