User manual

DA14580 Production test tool

UM-B-008

Abstract
This document describes the command line interface (CLI) for the production test tool of DA14580.

Contents

Tables

1 Terms and definitions

CLI	Command Line Interface
GUI	Graphical User Interface
HCI	Host Controller Interface
XTAL16M	16 MHz Crystal
XTAL32K	32 KHz Crystal
RC16M	16 MHz RC Oscillator
RC32K	32 KHz RC Oscillator

2 References

1. DA14580, Data sheet, Dialog Semiconductor
2. SPECIFICATION OF THE BLUETOOTH SYSTEM, version 4.0, Bluetooth SIG, 2010
3. Connection Manager, Help Document, Dialog Semiconductor

3 Introduction

This document describes the command line interface (CLI) for the production test tool of DA14580.

The tool is a Microsoft Windows command line program that enables communication over UART with a DA14580 device running the production test firmware.

The production test firmware is a special firmware that supports:

- The Bluetooth SIG standardised receiver and transmitter test HCI commands [2].
- Additional custom test HCI commands.

All test commands are also supported by the Connection Manager GUI application [3].

4 Getting started

4.1 Precompiled binaries

Precompiled binaries are provided for both the production test firmware and tool in the SDK:

- **binaries\da14580\prod_test\cust_prod_test_ES5.hex**
- **binaries\host\windows\prod_test_cmds\prodtest.exe**

4.2 Building the tool

The production test tool source code is placed under the: **tools\prod_test\prod_test_cmds** folder in the SDK.

Open **tools\prod_test\prod_test_cmds\prodtest.sln** in “Microsoft Visual C++ 2010 Express”.

Make sure that the active solution configuration is the Release Configuration.

Build the project by selecting “Build –> Build Solution” in the menu.

The executable **prodtest.exe** is generated under the **tools\prod_test\prod_test_cmds\Release** folder.

4.3 Building the firmware

The required firmware is included in the SDK under the folder:

dk_apps\keil_projects\prod_test\prod_test_ES5

Steps for building the firmware

a. Open **dk_apps\keil_projects\prod_test\prod_test_ES5\prod_es5.uvproj** in Keil µVision IDE.
b. Select menu “Project -> Rebuild all target files” to build the project.
c. Get the generated hex file from:
dk_apps\keil_projects\prod_test\prod_test_ES5\out\cust_prod_test_ES5.hex

5 General description

The general syntax of the tool is:

prodtest <switches> <command> <command-arg-1> … <command-arg-2>

A command always returns a status code and optionally a list of return values. The return status code and values are written to the standard output in a simple format: <*value_name*> = <*value*>.

The return status code is also returned as an exit code.

A zero status code represents the successful execution of a command. A non-zero status code encodes the type of failure.

All other output of the tool is written to stderr (except the help message when –h option is used).

6 Command line switches

6.1 Switch –h

Description: print out all the commands of prodtest on a command line terminal

Usage: prodtest -h

6.2 Switch –p

Description: select the COM port number on a PC. All commands require this switch.

Usage: prodtest –p <COM_PORT_NUMBER> ….

7 Commands

7.1 cont_pkt_tx

Description: this is the Bluetooth SIG standardized [2] HCI_LE_Transmitter_Test command. It continuously transmits packets until the **stoptest** command is executed.

Syntax: prodtest –p <COM_PORT_NUMBER> cont_pkt_tx <FREQUENCY> <DATA_LENGTH> <PAYLOAD_TYPE>

Parameters:

- *FREQUENCY* is an even integer between 2402 and 2480. E.g. integer 2480 corresponds to 2.480 GHz.
- *DATA_LENGTH* is the payload length in bytes (a number between 1 and 37).
- *PAYLOAD_TYPE* must have one of the following values:
  0: Pseudo-Random bit sequence 9
  1: Pattern of alternating bits ‘11110000’
  2: Pattern of alternating bits ‘10101010’
  3: Pseudo-Random bit sequence 15
  4: Pattern of All ‘1’ bits
  5: Pattern of All ‘0’ bits
  6: Pattern of alternating bits ‘00001111’
  7: Pattern of alternating bits ‘0101’

Example:

```
prodtest -p 14 cont_pkt_tx  2402  35  6
```

Output example:

```
status = 0
```

7.2 pkt_tx

Description: transmit the specified number of packets.

Syntax: prodtest –p <COM_PORT_NUMBER> pkt_tx <FREQUENCY> <DATA_LENGTH> <PAYLOAD_TYPE> <NUMBER_OF_PACKETS>

Parameters:

- *FREQUENCY* is an even integer between 2402 and 2480. E.g. 2480 corresponds 2.480 GHz.
- *DATA_LENGTH* is the payload length in bytes (a number between 1 and 37).
- *PAYLOAD_TYPE* must have one of the following values:
  0 : Pseudo-Random bit sequence 9
  1 : Pattern of alternating bits ‘11110000
  2 : Pattern of alternating bits ‘10101010’
  3 : Pseudo-Random bit sequence 15
  4 : Pattern of All ‘1’ bits
  5 : Pattern of All ‘0’ bits
  6 : Pattern of alternating bits ‘00001111
  7 : Pattern of alternating bits ‘0101’
- *NUMBER_OF_PACKETS* is an integer between 1 and 65535.

Example: the following command sends 1000 packets

```
prodtest -p 14 pkt_tx  2402  35  6 1000
```

Output example:

```
status = 0
```

7.3 start_pkt_rx

Description: This is the Bluetooth SIG standardized [2] HCI_LE_Receiver_Test command. It continuously receives packets until the **stoptest** command is executed.

Syntax: prodtest –p <COM_PORT_NUMBER> start_pkt_rx <FREQUENCY>

Parameters:

- *FREQUENCY* is an even integer between 2402 and 2480. E.g. 2480 corresponds 2.480 GHz.

Example:

```
prodtest –p 14 start_pkt_rx  2402
```

Output example:

```
status = 0
```

7.4 start_pkt_rx_stats

Description: starts packet RX with additional statistics. It continuously receives packets until the stop_pkt_rx_stats command is executed.

Syntax: prodtest –p <COM_PORT_NUMBER> start_pkt_rx_stats <FREQUENCY>

Parameters:

- *FREQUENCY* is an even integer between 2402 and 2480. E.g. 2480 corresponds 2.480 GHz.

Example:

```
prodtest –p 14 start_pkt_rx_stats 2402
```

Output example:

```
status = 0
```

7.5 stop_pkt_rx_stats

Description: ends packet RX with additional statistics and reports the following statistics:

- number of packets received correctly (nb_packets_received_correctly)
- number of packets with sync error (nb_packets_with_syncerror)

- number of packets with CRC error (nb_packets_received_with_crcerr)
- RSSI in dBm (rssi)

Syntax: prodtest –p <COM_PORT_NUMBER> stop_pkt_rx_stats

Example:

prodtest –p 14 stop_pkt_rx_stats

Output example:

status = 0

nb_packets_received_correctly = 8529

nb_packets_with_syncerror = 0

nb_packets_received_with_crcerr = 1

rssi = -37.30

7.6 stoptest

Description: This is the Bluetooth SIG standardized [2] HCI_LE_Test_End command. It is used:

- after a cont_pkt_tx command to end the standard packet TX test mode.
- after a start_pkt_rx command to end the standard packet RX mode and report the number of received packets.

Syntax: prodtest –p <COM_PORT_NUMBER> stoptest

Example:

prodtest –p 14 stoptest

Output example (the number of packets is always zero when executed after a cont_pkt_tx):

status = 0

number_of_packets = 0

Output example (when executed after a start_pkt_rx):

status = 0

number_of_packets = 4360

7.7 unmodulated TX

Description: starts a Continuous Wave (CW) or unmodulated TX test.

Syntax: prodtest –p <COM_PORT_NUMBER> unmodulated TX <FREQUENCY>

Parameters:

- *FREQUENCY* is an even integer between 2402 and 2480. E.g. 2480 corresponds 2.480 GHz.

Example:

prodtest -p 14 unmodulated TX 2404

Output example:

status = 0

7.8 unmodulated RX

Description: starts the unmodulated RX test.

Syntax: prodtest –p <COM_PORT_NUMBER> unmodulated RX <FREQUENCY>

Parameters:

- *FREQUENCY* is an even integer between 2402 and 2480. E.g. 2480 corresponds 2.480 GHz.

Example:

prodtest -p 14 unmodulated RX 2404

Output example:

status = 0

7.9 unmodulated OFF

Description: stops unmodulated TX or RX test.

Syntax: prodtest –p <COM_PORT_NUMBER> unmodulated OFF

Example:

prodtest -p 14 unmodulated OFF

Output example:

status = 0

7.10 start_cont_tx

Description: starts the continuous TX test mode. It transmits continuously a modulated signal until the **stop_cont_tx** command is executed.

Syntax: prodtest –p <COM_PORT_NUMBER> start_cont_tx <FREQUENCY> <PAYLOAD_TYPE>

Parameters:

- *FREQUENCY* is an even integer between 2402 and 2480. E.g. 2480 corresponds 2.480 GHz.
- *PAYLOAD_TYPE* must have one of the following values:
  0 : Pseudo-Random bit sequence 9
  1 : Pattern of alternating bits ‘11110000’
  2 : Pattern of alternating bits ‘10101010’
  3 : Pseudo-Random bit sequence 15
  4 : Pattern of All ‘1’ bits
  5 : Pattern of All ‘0’ bits
  6 : Pattern of alternating bits ‘00001111’
  7 : Pattern of alternating bits ‘0101’

Example:

prodtest –p 14 start_cont_tx 2404 4

Output example:

status = 0

7.11 stop_cont_tx

Description: stops the continuous TX test mode.

Syntax: prodtest –p <COM_PORT_NUMBER> stop_cont_tx

Example:

prodtest –p 14 stop_cont_tx

Output example:

status = 0

7.12 reset

Description: this is the Bluetooth SIG standardized [2] HCI_Reset command.

Syntax: prodtest –p <COM_PORT_NUMBER> reset

Example:

prodtest –p 14 reset

Output example:

status = 0

7.13 xtrim rd

Description: reads the value of the XTAL16M trimming register as specified in DA14580 datasheet [1] section 5: CLK_FREQ_TRIM_REG (0x50000002).

The value is reported in decimal format.

Syntax: prodtest -p <COM port number> xtrim rd

Example:

prodtest –p 14 xtrim rd

Output example:

status = 0

trim_value = 500

7.14 xtrim wr

Description: writes a value into the XTAL16M trimming register. The new value is written immediately and it is already effective when the prodtest program finishes executing this command.

Syntax: prodtest -p <COM port number> xtrim wr <trim_value>

Parameters:

- *trim_value* is the unsigned 16-bit decimal value to be written into the XTAL16M trimming register.

Example:

prodtest –p 14 xtrim wr 500

Output example:

status = 0

7.15 xtrim en

Description: enables XTAL16M output on GPIO P0_5. The command takes effect immediately.

Note 1 This command also enables:
XTAL32K output on GPIO P0_6
RC16M output on P0_7
RC32K output on P1_0

Syntax: prodtest -p <COM port number> xtrim en

Example:

prodtest –p 14 xtrim en

Output example:

status = 0

7.16 xtrim dis

Description: disables XTAL16M output on GPIO P0_5. The command takes effect immediately.

Note 2 This command also disables:
XTAL32K output on GPIO P0_6
RC16M output on P0_7
RC32K output on P1_0

Syntax: prodtest -p <COM port number> xtrim dis

Example:

prodtest –p 14 xtrim en

Output example:

status = 0

7.17 xtrim inc

Description: increments the XTAL16M trimming register. The register's new value is already effective when the prodtest program finishes executing this command.

Syntax: prodtest -p <COM port number> xtrim inc <delta>

Parameters:

- *delta* is the unsigned 16-bit decimal value to be added to the XTAL16M trimming register.

Example:

prodtest –p 14 xtrim inc 5

Output example:

status = 0

7.18 xtrim dec

Description: decrements the XTAL16M trimming register. The register's new value is already effective when the prodtest program finishes executing this command.

Syntax: prodtest -p <COM port number> xtrim dec <delta>

Parameters:

- *delta* is the unsigned 16-bit decimal value to be subtracted from the XTAL16M trimming register.

Example:

prodtest –p 14 xtrim dec 5

Output example:

status = 0

7.19 sleep

Description: puts the device to sleep for a specified number of minutes and seconds.

Syntax: prodtest –p <COM_PORT_NUMBER> sleep <mode> <minutes> <seconds>

Parameters:

- *mode* is one of the available sleep modes defined in section 3.5 of the datasheet [1]:
  - **none** = Sleep Mode (no power gating, ARM CPU is idle waiting for an interrupt)
  - **extended** = Extended Sleep Mode
  - **deep** = Deep Sleep Mode
- *minutes* is a number between 0 and 255.

- *seconds* is a number between 0 and 255.

If both minutes and seconds are set to zero then the device sleeps forever.

The active peripherals in each sleep mode are shown in table 1.

Table 1: Power domain states for each sleep mode

Power mode	PD_SYS (AHB, OTP, ROM, Watchdog, SW timer, GPIO multiplexing)	PD_PER (UARTs, SPI, I2C, Keyboard controller, ADC)	PD_DBG	PD_RAD (Radio and BLE core)	PD_RRx (Retention RAM x)	PD_SR (System RAM)	Analog (BandGap, DCDC converter XTAL16M, RC oscillators, ADC, LDOs)
Sleep Mode	ON	ON	ON	ON	ON	ON	ON
Extended Sleep Mode	OFF	Programmed to OFF	OFF	Programmed to OFF	Programmed to ON	ON	OFF
Deep Sleep Mode	OFF	Programmed to OFF	OFF	Programmed to OFF	Programmed to ON	OFF	OFF

Known limitations:

- UART communication is lost when the device wakes up from extended sleep mode.
- The device will not wake up from deep sleep mode.

Example 1:

prodtest –p 14 sleep none 1 0

Example 2:

prodtest –p 14 sleep extended 0 30

Example 3:

prodtest –p 14 sleep deep 0 0

Output example:

status = 0

8 Return status codes

Following table summarizes the return codes of the productions test tool

Table 2: Return codes

Status code	Description
0	SC_NO_ERROR
1	SC_MISSING_COMMAND
2	SC_INVALID_COMMAND
3	SC_WRONG_NUMBER_OF_ARGUMENTS
4	SC_INVALID_COM_PORT_NUMBER
5	SC_INVALID_FREQUENCY_ARG

Status code	Description
6	SC_INVALID_DATA_LENGTH_ARG
7	SC_INVALID_PAYLOAD_TYPE_ARG
8	SC_COM_PORT_INIT_ERROR
9	SC_RX_TIMEOUT
10	SC_UNEXPECTED_EVENT
11	SC_INVALID_NUMBER_OF_PACKETS_ARG
12	SC_INVALID_UNMODULATED_CMD_MODE_ARG
13	SC_COM_PORT_NOT_SPECIFIED
14	SC_INVALID_SLEEP_CMD_MODE_ARG
15	SC_INVALID_SLEEP_CMD_MINUTES_ARG
16	SC_INVALID_SLEEP_CMD_SECONDS_ARG
17	SC_INVALID_XTAL_TRIMMING_CMD_OPERATION_ARG
18	SC_INVALID_XTAL_TRIMMING_CMD_TRIM_VALUE_ARG
1000 – 1063	SC_HCI_STANDARD_ERROR_CODE_BASE (1000) + standard HCI error code

9 Revision history

Revision	Date	Description
1.0	24-Mar-2014	Initial version for DA14580-01

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.

Disclaimer

Information in this document is believed to be accurate and reliable. However, Dialog Semiconductor does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information. Dialog Semiconductor furthermore takes no responsibility whatsoever for the content in this document if provided by any information source outside of Dialog Semiconductor.

Dialog Semiconductor reserves the right to change without notice the information published in this document, including without limitation the specification and the design of the related semiconductor products, software and applications.

Applications, software, and semiconductor products described in this document are for illustrative purposes only. Dialog Semiconductor makes no representation or warranty that such applications, software and semiconductor products will be suitable for the specified use without further testing or modification. Unless otherwise agreed in writing, such testing or modification is the sole responsibility of the customer and Dialog Semiconductor excludes all liability in this respect.

Customer notes that nothing in this document may be construed as a license for customer to use the Dialog Semiconductor products, software and applications referred to in this document. Such license must be separately sought by customer with Dialog Semiconductor.

All use of Dialog Semiconductor products, software and applications referred to in this document are subject to Dialog Semiconductor's Standard Terms and Conditions of Sale, unless otherwise stated.

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.

Contacting Dialog Semiconductor

Germany Headquarters
Dialog Semiconductor GmbH
Phone: +49 7021 805-0

United Kingdom
Dialog Semiconductor (UK) Ltd
Phone: +44 1793 757700

The Netherlands
Dialog Semiconductor B.V.
Phone: +31 73 640 8822

Email:
enquiry@diasemi.com

North America
Dialog Semiconductor Inc.
Phone: +1 408 845 8500

Japan
Dialog Semiconductor K. K.
Phone: +81 3 5425 4567

Taiwan
Dialog Semiconductor Taiwan
Phone: +886 281 786 222

Web site:
www.dialog-semiconductor.com

Singapore
Dialog Semiconductor Singapore
Phone: +65 64 849929

China
Dialog Semiconductor China
Phone: +86 21 5178 2561

Korea
Dialog Semiconductor Korea
Phone: +82 2 3469 8291