CyBluetool User Guide (Windows)

Doc. No. 002-24800 Rev. **

Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709
www.cypress.com

Copyrights

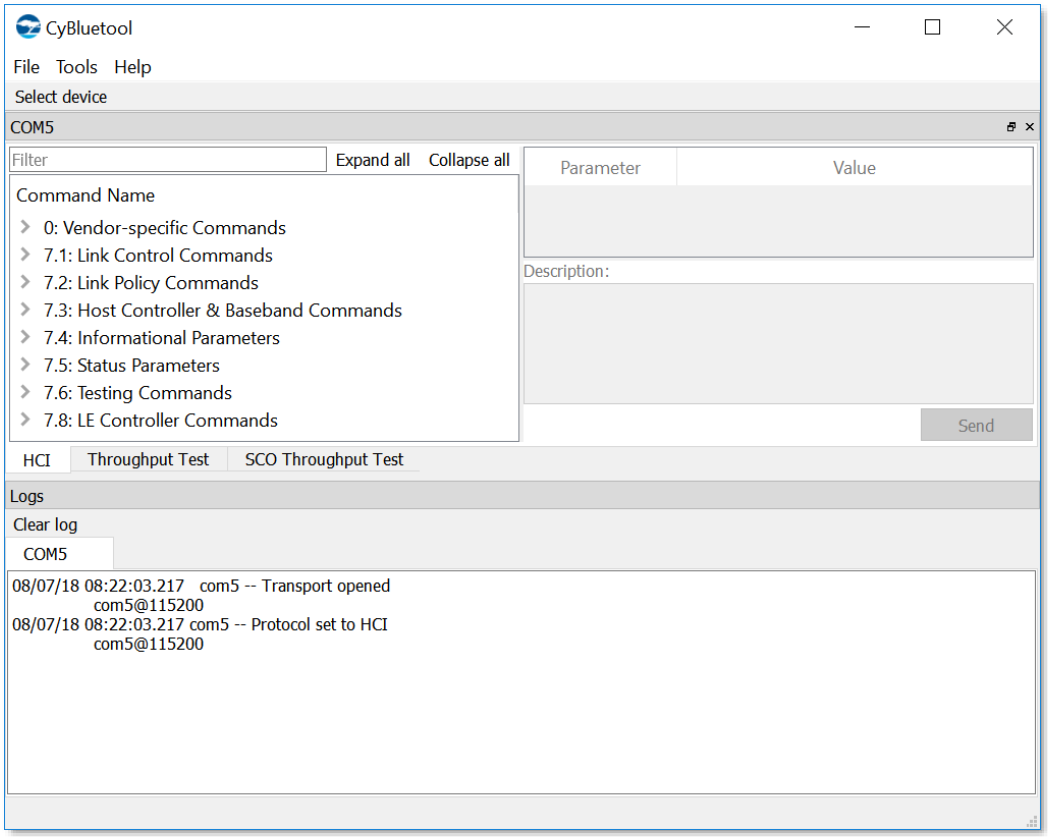
Contents

1. Introduction

CyBluetool is a GUI tool for testing and debugging Cypress Bluetooth devices (CYW920719Q40EVB-01, CYW920735Q60EVB-01, and CYW920706WCDEVAL Evaluation Kits). CyBluetool connects to Bluetooth devices at the Host Controller Interface (HCI) protocol layer; this tool currently supports HCI UART and HCI USB transport interfaces. The tool allows you to send Bluetooth (BT) HCI commands and receive BT HCI events from the BT controller of connected devices.

CyBluetool is supported on 64-bit Windows 7 and higher. Windows 10 (64-bit) is recommended.

Figure 1-1. CyBluetool



CyBluetool features:

- Supports Bluetooth 5.0 HCI commands
- Sends and receives Bluetooth HCI commands and events
- Asynchronous Connection-Less (ACL) and Bluetooth LE throughput testing
- Sending and receiving of WAV files over the Synchronous Connection Oriented (SCO) channel
- Supports HCI UART and HCI USB transports
- Logs HCI commands and events with timestamps
- Firmware download (formats supported: CGS and HCD)

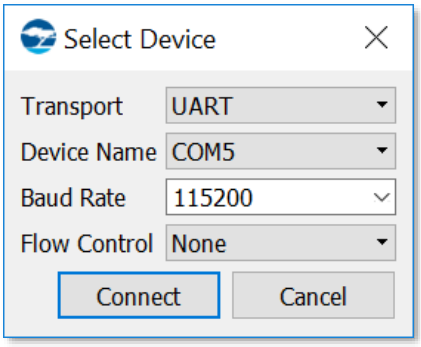
1.1 Installation

1. Double-click the CyBluetool installer for Windows and follow the instructions on the installation wizard.
2. Once installation is complete, in the installation directory, double-click ***cybluetool.exe*** or launch the **CyBluetool** desktop app from the start menu to run the CyBluetool GUI application.

1.2 Getting Started

1. Connect a Cypress Bluetooth device kit to the PC and launch CyBluetool.
2. Click **Select device** on the toolbar. See Figure 1-2.

Figure 1-2. Select Device Dialog



3. Select the following on the Select Device dialog, and click **Connect**:

   - Transport

     From the drop-down, select the HCI transport type supported by the device.

   - Device Name

     In case of UART transport, this field automatically populates all the UART interfaces currently present in the PC. Select the COM port where your kit is connected.

     In case of USB transport, the device name is ‘usb0’, ‘usb1’, and so on, depending on the number of USB devices inserted in the PC.

     Note: Some of the kits have two UART interfaces and the tool will list both the interfaces. Typically, the interface with the smaller number is the HCI transport interface.

   - Baud Rate

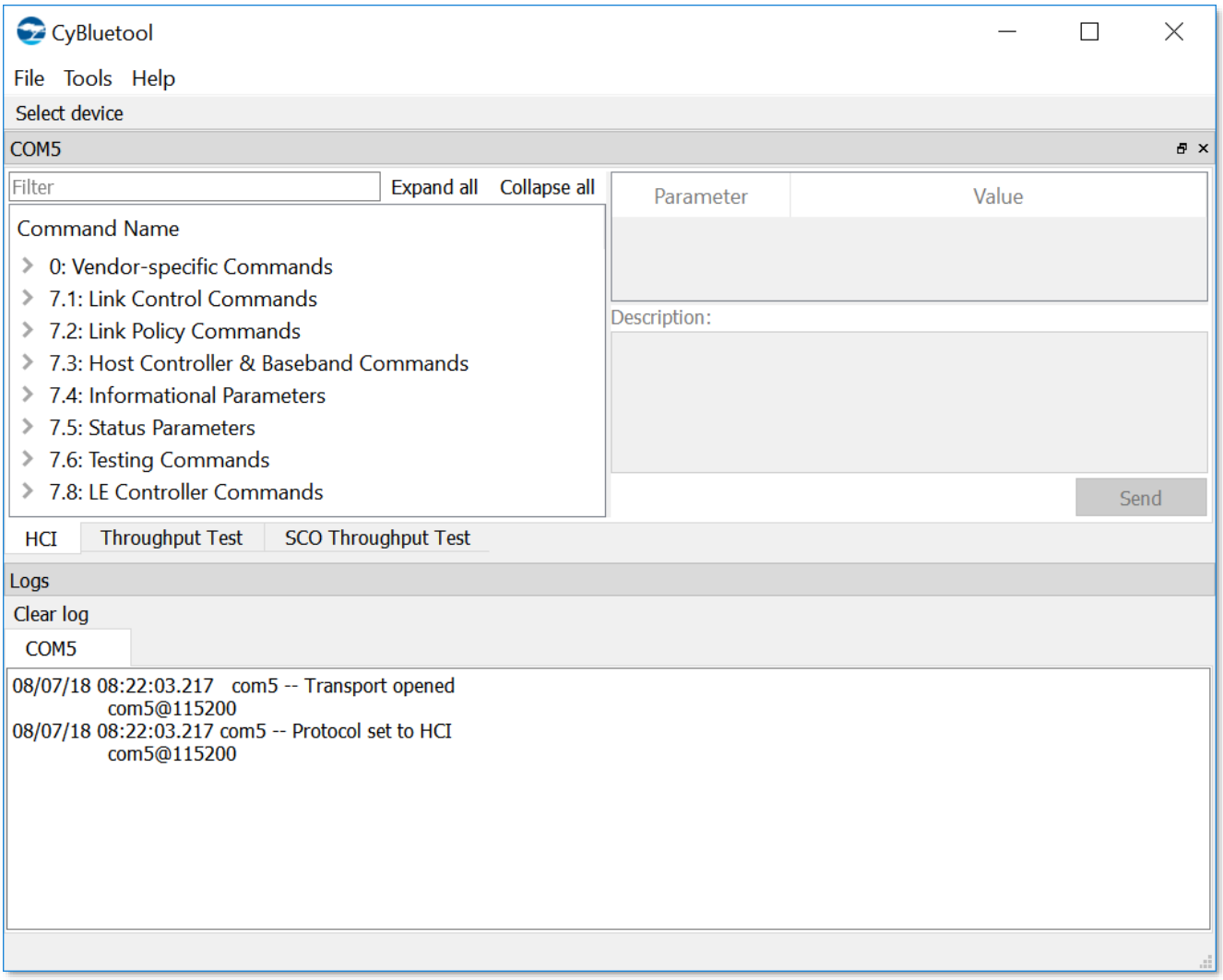
     This field is enabled only for UART transport type. Default baud rate is 115200.

   - Flow control

     This field is enabled only for UART transport type. Default flow control is RTS/CTS.

   If the connection to the device is successful, the tool opens a device view as shown in Figure 1-3. A separate device view is opened for each connected device.

Figure 1-3. Device View



4. Select a command (e.g., 'Reset') from the command tree and click **Send** to send the command to the device. If the selected command has one or more parameters, the parameters will be displayed on the Parameters View. You can modify the parameter values and then click **Send**.

   The 'Logs' window displays the command sent and the received event.

1.3 Exiting the Tool

Choose **File** > **Exit** or click the close button (x) to close the tool.

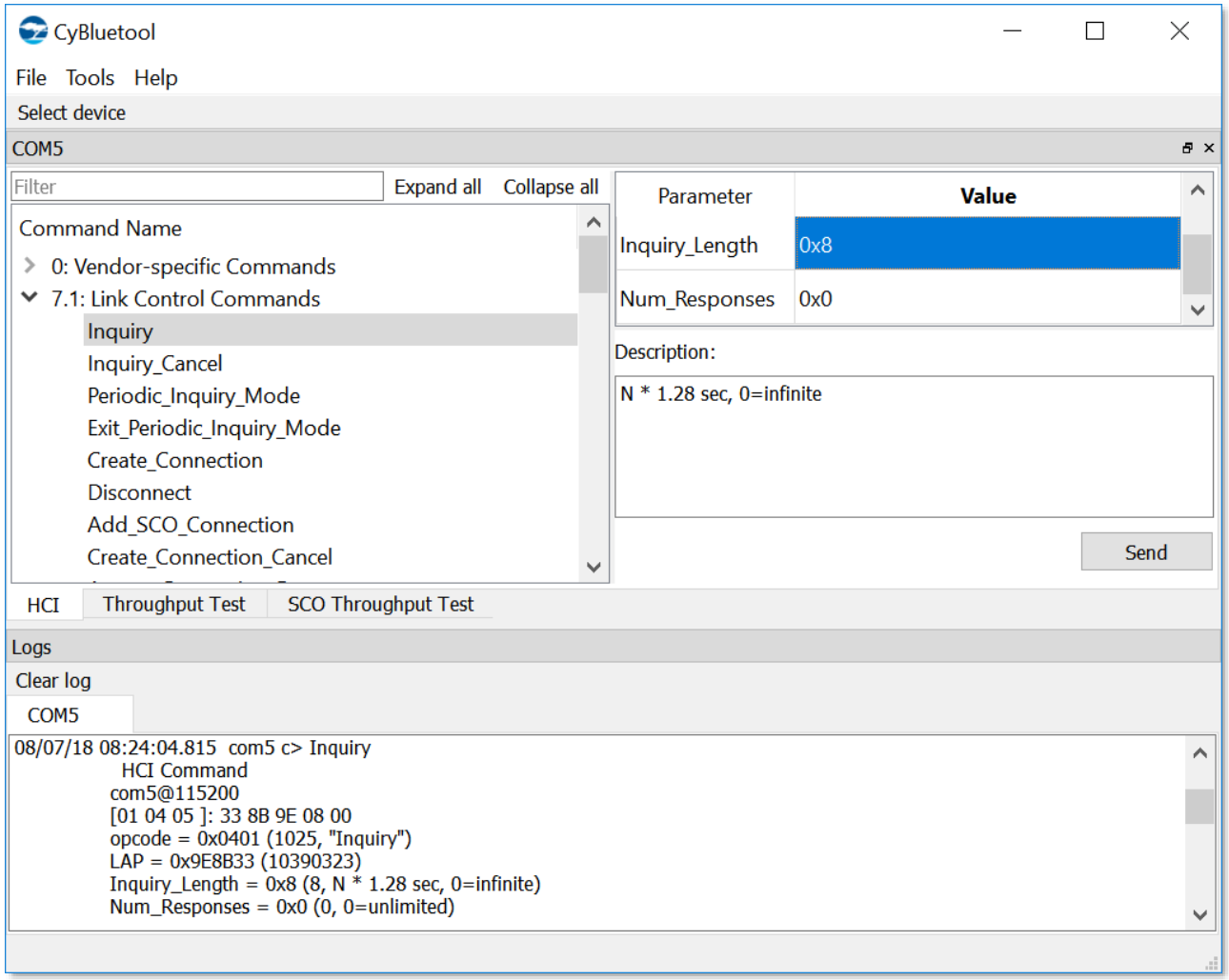
2. Features

2.1 Sending HCI Commands

The HCI command tab in the device view displays all the supported HCI commands in a tree view where they are grouped into categories (as per the Bluetooth 5.0 Core Specification). Use the **Expand all** and **Collapse all** buttons to expand or collapse the categories.

On selecting a command in the tree view, the input parameters for the command, if any, are displayed in a table on the right. If the parameters have default values, these values are displayed. The description box displays any notes or description about a selected command or parameter.

Figure 2-1. Send HCI Commands



After providing all input parameters, click **Send** (or press **Alt**+**S**) to send the command to the connected device. Details of the command sent to the device are displayed in the log window. Any event in response to the command is also displayed along with the event parameters.

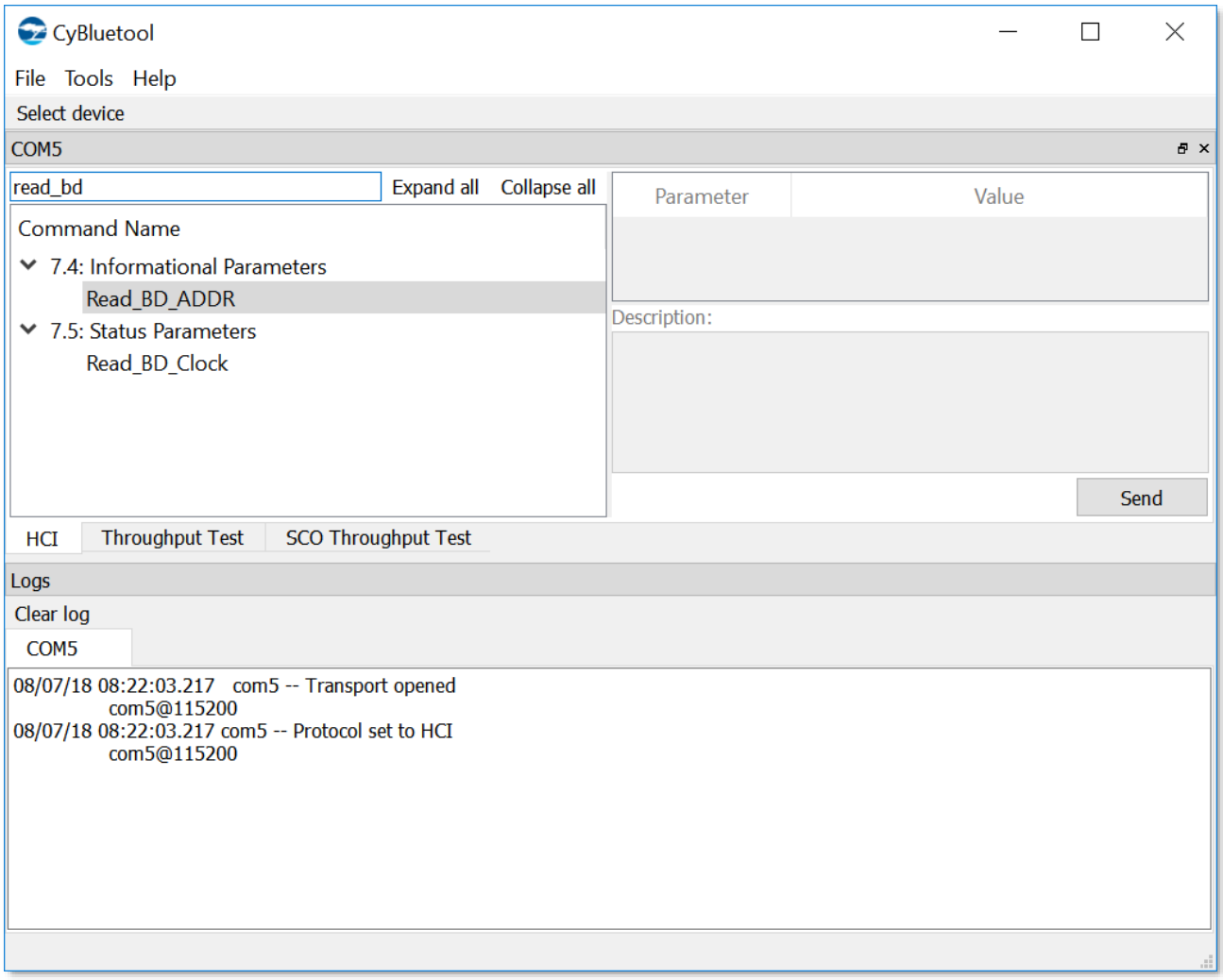
Double-click the command in the tree view (or press **Enter**) to send a command without any parameters.

2.1.1 Find Commands

Click on the 'Filter' text box above the command tree and type the name of the command to find a specific command. The command tree will dynamically update to show only the commands which contains the current filter text.

Alternately, click anywhere on the command tree and start typing to find a command.

Figure 2-2. Find Commands



2.2 Throughput Testing

CyBluetool supports measuring the transmit and receive throughput over ACL and LE channels. To start a throughput test, select the ‘Throughput Test’ tab at the bottom of the device view.

2.2.1 Transmit Test

Use the transmit test to send data to a peer device over an ACL or LE connection and measure the throughput. You can either send a data pattern or a file. If the ASCII option is selected, the input can be any valid string of characters. Here, space

is also considered a character. After providing the data pattern or file, the total bytes to be sent are displayed. This is updated when the count is modified as well.

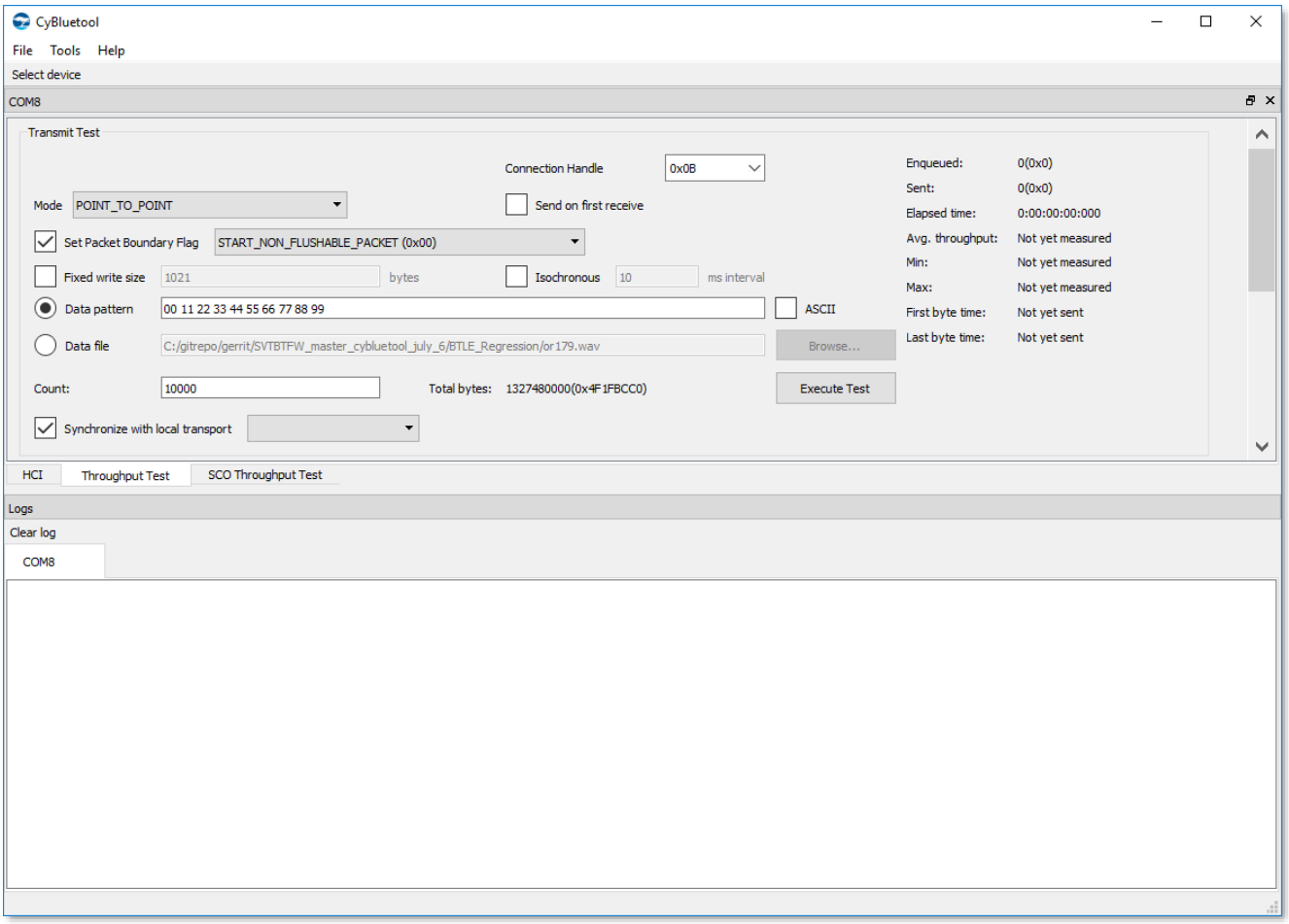
The 'Send on first receive' option starts the TX throughput test when it receives the first byte from the peer device. You can abort test while the device is waiting for the first byte.

Table 2-1 summarizes the TX test parameters.

Table 2-1. Transmit Throughput Test Parameters

Paramter	Description	Values / Valid Range
Mode	Mode of data transfer	Point-to-Point Active Broadcast Slave Broadcast
Set Packet Boundary Flag	You can set the Packet Boundary flag for simple ACL data transfer. When this is selected, only data pattern can be sent. Data file cannot be sent.	Start Non-flushable Packet Continuation Packet Start Packet
Connection handle	Connection handle of the ACL connection established with a peer device.	If an ACL connection is established using the tool, the valid connection handles are populated in a drop-down. You can provide the connection handle in hex or decimal format also.
Fixed Write Size	The size of each write. If the size exceeds ACL buffer size, each write will be fragmented into a start fragment and continuation fragments.	Integer from 1 to 65535.
Interval	The interval at which isochronous data transfer is performed, in millisecond.	Enabled only if the 'Isochronous' option is selected.
Data pattern	Data to be sent - mentioned as bytes	Bytes should be separated by space. Both decimal and hex values are accepted.
Data file	Path to a file from which data is to be read and sent.	Should be a valid path to a file for which you have read permission.
ASCII	Interprets the entered bytes as ASCII code for characters, and displays the characters.	Enabled only if Data Pattern is selected.
Count	Number of times the file content/data pattern is to be transferred.	A positive number above 0.
Send on First Receive	When enabled, it holds the TX data transfer until the device has received one ACL data packet from the peer device.	NA
Synchronize with local transport	When enabled, this option sets the Rx Test values on the selected peer device based on the values entered in the Tx test on current device.	NA
Total bytes	CyBluetool calculates the total number of bytes to be sent based on the data pattern and count, and displays it in this field.	NA

Figure 2-3. Transmit Throughput Test

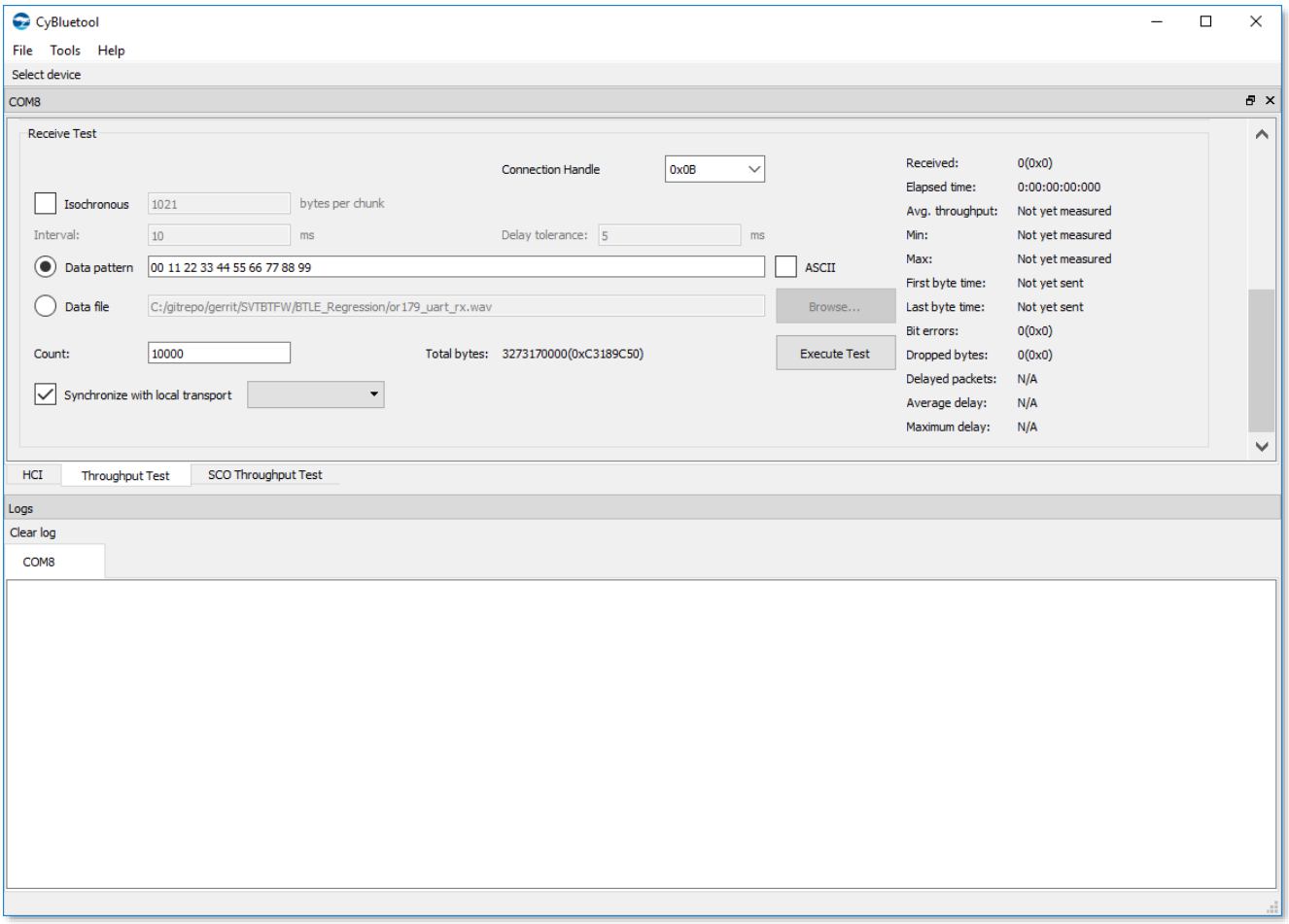


2.2.2 Receive Test

You can use the receive test to receive data from a connected device and measure the throughput. The data pattern or file expected by the receiving device must be specified in the Receive Test form. Similar to the Transmit Test form, the total number of bytes are displayed and updated when pattern/file and/or count is updated.

An easy way to provide the inputs for Receive Test is to use the 'Synchronize with local transport' option. This allows the input values for a receive test (or transmit test) to be copied from the transmit test (or receive test) of the selected transport respectively. When the synchronization option is selected and the receive test is triggered, it automatically starts the transmit test on the other device and vice versa. In both cases, the receive test is started before the transmit test.

Figure 2-4. Receive Throughput Test



2.3 SCO Throughput

CyBluetool supports sending and receiving of data over SCO channels. To start a SCO data transfer, select the **SCO Throughput Test** tab at the bottom of the device view.

2.3.1 SCO Transmit Test

You can use the SCO Transmit test to send a wav file over the SCO channel. The path to the wav file must be selected. 'Count' indicates the number of times the file is to be sent. The number of samples sent and duration of file transfer is displayed on the right.

Table 2-2 summarizes the TX test parameters.

Table 2-2. Transmit SCO Throughput Test Parameters

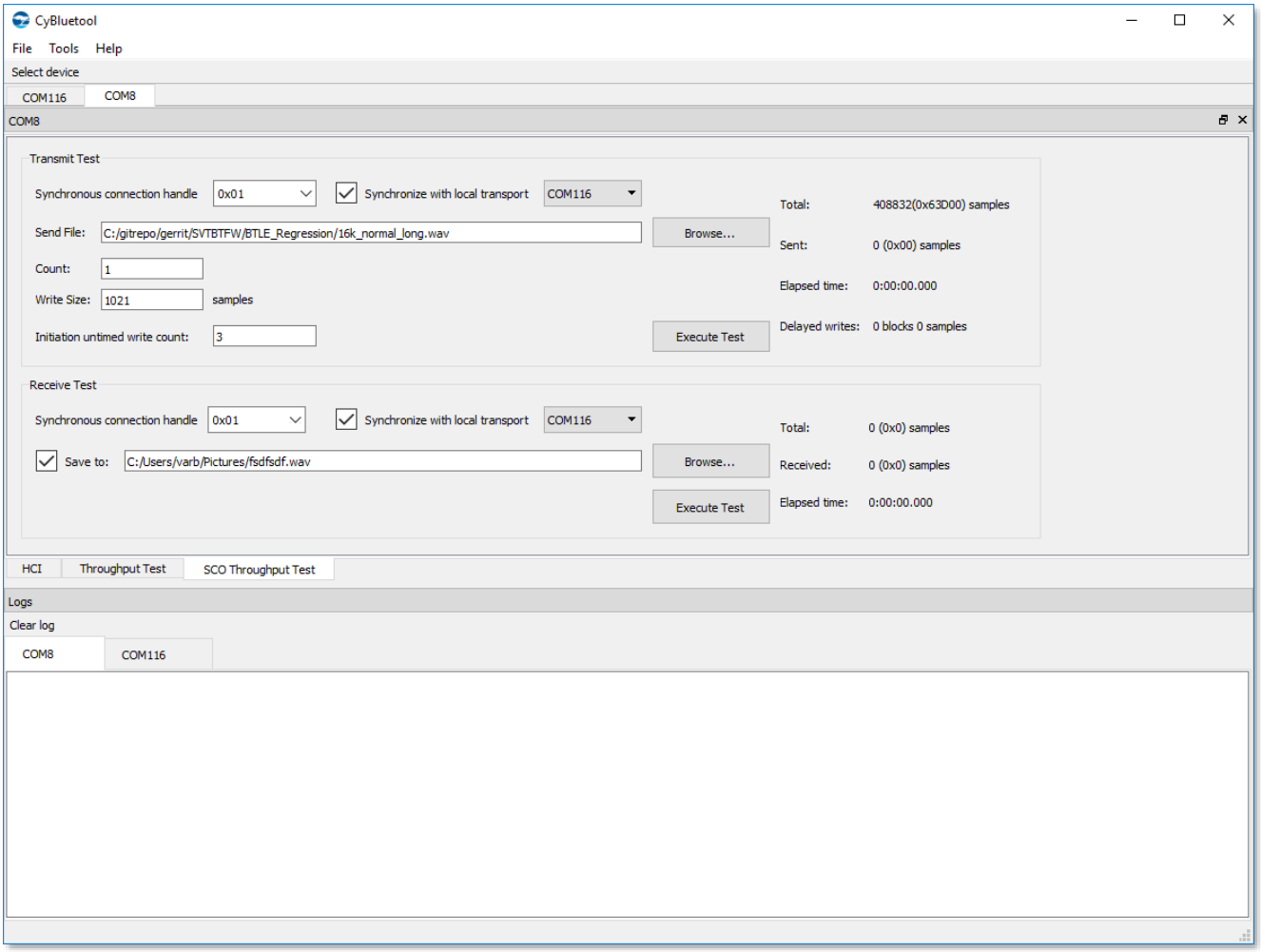
Paramter	Description	Values / Valid Range
Synchronous Connection handle	Connection handle of the SCO connection established with a peer device.	If an ACL connection is established using CyBluetool, valid connection handles are populated in a drop-down. You can provide connection handle in hex or decimal format also.
Send File	Path to a wav file which is to be read and sent.	Should be a valid path to a file for which you have read permission.
Count	Number of times the file is to be transferred.	A positive number above 0.

Paramter	Description	Values / Valid Range
Write Size	Number of samples to send in each SCO buffer.	Integer between 6 and 255.
Initiation untimed write count	Number of blocks of 'Write Size' to write to the device back-to-back, untimed, at the time of test initiation.	Integer between 1 and 255.
Synchronize with local transport	When enabled, this option sets the Rx Test values on the selected peer device based on the values entered in the Tx test on current device.	NA

2.3.2 SCO Receive Test

You can use the SCO Receive test to write the received SCO data and save it as a wav file. The path to the new file must be specified as shown below. The number of samples received, and the elapsed time are displayed on the right.

Figure 2-5. SCO Throughput



2.4 Firmware Download

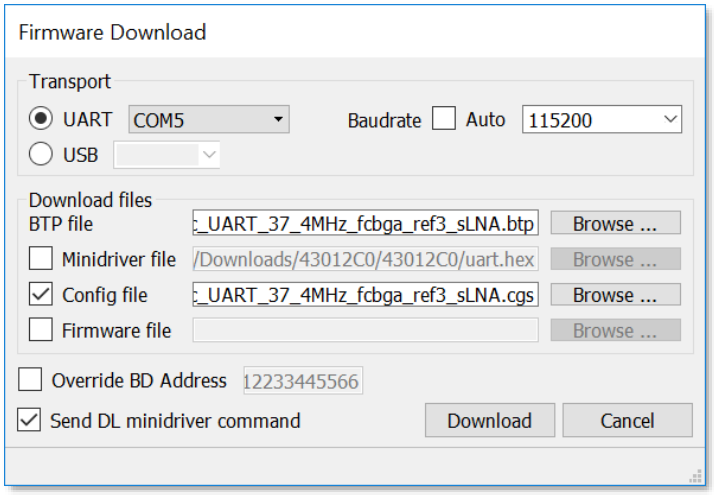
You can use CyBluetool to download firmware onto a device with the **Tools** > **Firmware Download** option.

To download firmware onto a device, the tool need not be connected to the device in prior. Use the options in the Firmware Download interface to select the device and baud rate.

You can select the files required for downloading the firmware onto the device. On selecting a BTP file, the tool looks for information about the paths to the other files (minidriver, config, and firmware files). If paths to these files are provided, the corresponding path fields are automatically populated by the tool. You can deselect these files, but at least one of config file or firmware file must be provided to perform firmware download.

The status of firmware download is displayed on the status bar. Any errors during firmware download are also displayed here.

Figure 2-6. Firmware Download

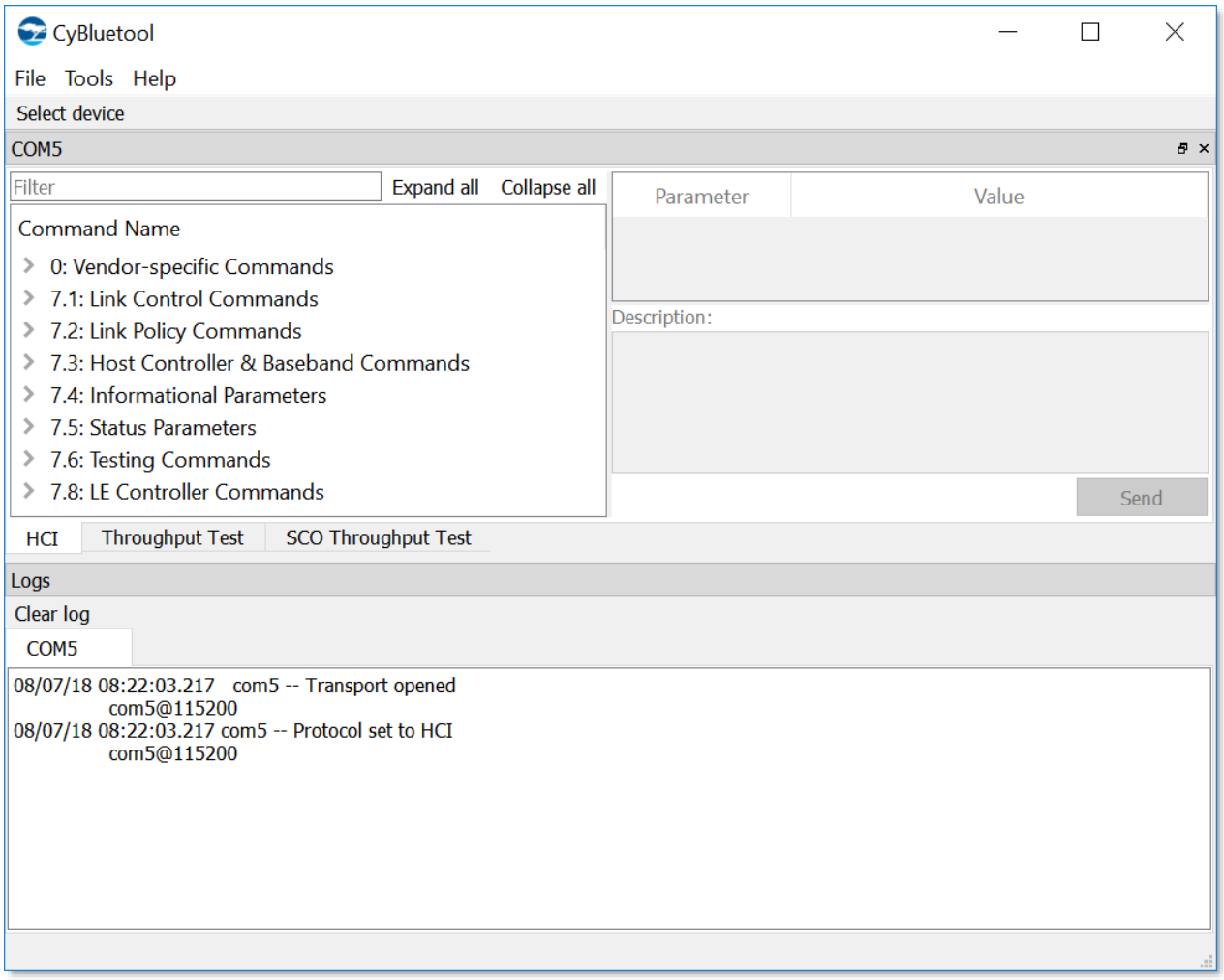


2.5 Log Window

The log window in the tool displays the command, along with parameters, sent over the transport to the device. It also decodes and displays the events and associated event parameter values received from the device. Apart from displaying the logs in the log window, the tool saves the logs to a file in the temp directory of the PC on which it is running. A new log file is created for each session of the tool.

Log for each connected device is displayed on a separate tab in the log window as shown below. You can use the **Clear log** option on the Log window to clear the logs in an active log tab.

Figure 2-7. Log Window

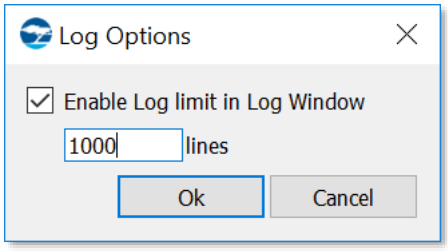


2.5.1 Log Options

Select **Tools** > **Log Options** to set the number of lines of logs in the log window.

Selecting the checkbox enables the limit. By default, no limit is applied.

Figure 2-8. Log Options



Revision History

Document Title: CyBluetool User Guide (Windows) Document Number: 002-24800			
Revision	**Issue Date**	**Origin of Change**	**Description of Change**
**	08/17/2018	VARB	Initial release