Command Manager Multiple Concurrent vSP Connections

BL652

Application Note *v1.0*

INTRODUCTION

The goal of this document is to use command manager (*cmd.manager.sb*) to demonstrate multiple concurrent vSP connections with a Laird BL652.

OVERVIEW

Command Manager provides a command interface over the UART for many common BL652 Bluetooth operations. It allows you to configure, pair, and connect with other Bluetooth devices using Laird's Virtual Serial Port (vSP) service for Bluetooth Low Energy (BLE). Command Manager allows the BL652 to bridge the radio to the UART and it allows data arriving over the radio to be presented to the UART via unsolicited messages identified by connection handle. Data from the host intended for the radio is presented by way of text-based commands. This allows the demonstration of multiple concurrent vSP connections.

In this document, we assume the BL652 is the master and that it discovers, pairs, and connects with multiple remote devices. The following screen shots show the commands used on the BL652 master. We assume you are familiar with the configuration of the remote devices. You may need to use various types of devices to achieve multiple concurrent connections. To prepare for this application note, we used a mixture of devices including BL652 development kits and BL600 development kits. You can also use additional BT900 development kits, if available.

Note: Laird provides a library of sample *smart*BASIC applications including Command Manager (*cmd.manager.sb*) to provide an easy-to-use guide for implementing a range of different functionality within your applications. The sample application library on GitHub is never intended to be a completely robust, end-customer application for use in real world applications.

REQUIREMENTS

The following equipment is required:

- BL652 development kit for master
- Two or more Bluetooth Laird vSP such as another BL652 development kit or a BL600 development kit
- PC with enough USB ports for all the above
- UWTerminalX - https://github.com/LairdCP/UwTerminalX/releases
- Cmd.manager.sb sample app - https://github.com/LairdCP/BL652-Applications/tree/master/Applications

Americas: +1-800-492-2320
Europe: +44-1628-858-940
Hong Kong: +852 2923 0610

BL652 Master Development Kit Setup

To set up the development kit, follow these steps:

1. Use UWTerminalX to return the BL652 dev board to factory defaults using the command **AT&F*** as shown in Figure 1.

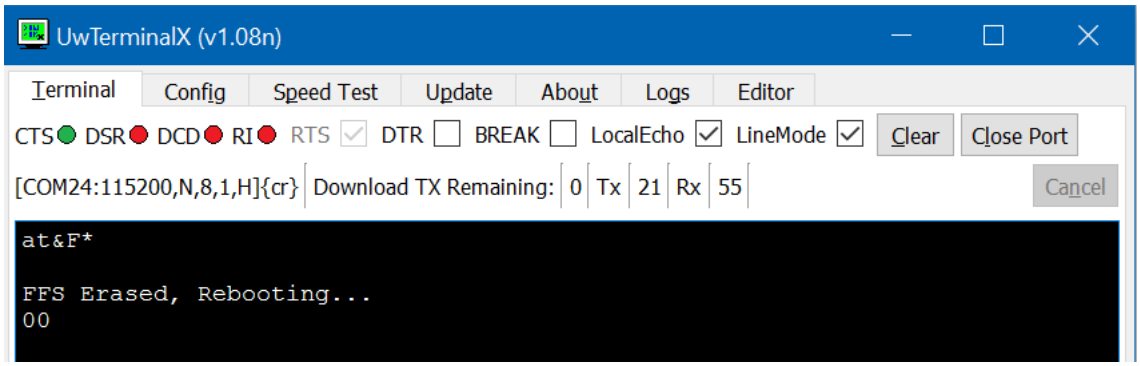

Figure 1: Factory default

2. Load Command Manager – use the right-click menu to select Xcompile+load (Figure 2).

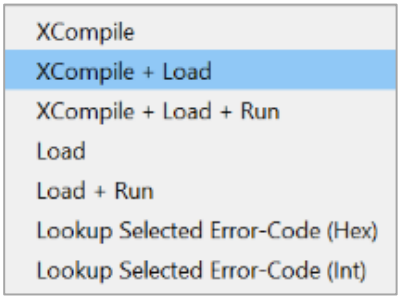

Figure 2: XCompile + Load

3. Select the *cmd.manager.sb* file.
4. Wait for the command manager program to load; this should take approximately 25 seconds. Command Manager can now be run by typing ***cmd*** followed by a return.

Note: A complete list of commands available through Command manager can be found in the source code file.

Americas: +1-800-492-2320
Europe: +44-1628-858-940
Hong Kong: +852 2923 0600

BLE (vSP) Scan

The vSP service UUID is **569a1101-b87f-490c-92cb-11ba5ea5167c**. Look for adverts which contain that UUID (Figure 3). Note that the UUID appears in reverse.

```
scan start 1000 0

OK
>
ADV:01D226C0A4B96D 02010609094C545F555041535303030A1811077C16A55EBA11CB920C497FB801119A56 0 -26
ADV:032EA5B59B3AE4 1EFF060001092000206CFD93FED9B033F75D6572FE7B337B823BD91E1B2F05 0 -39
ADV:01D226C0A4B96D 02010609094C545F555041535303030A1811077C16A55EBA11CB920C497FB801119A56 0 -29
ADV:032EA5B59B3AE4 1EFF060001092000206CFD93FED9B033F75D6572FE7B337B823BD91E1B2F05 0 -45
ADV:01C09C35256377 02010609094C545F555041535303030A1811077C16A55EBA11CB920C497FB801119A56 0 -31
ADV:0078BDBC6BDB31 1BFF750042040180207 8BDBC6BDB317ABDBC6BDB3001000000000000 0 -88
ADV:032EA5B59B3AE4 1EFF060001092000206CFD93FED9B033F75D6572FE7B337B823BD91E1B2F05 0 -39
ADV:01D226C0A4B96D 02010609094C545F555041535303030A1811077C16A55EBA11CB920C497FB801119A56 0 -27
ADV:01C09C35256377 02010609094C545F555041535303030A1811077C16A55EBA11CB920C497FB801119A56 0 -30
ADV:032EA5B59B3AE4 1EFF060001092000206CFD93FED9B033F75D6572FE7B337B823BD91E1B2F05 0 -38
ADV:01D226C0A4B96D 02010609094C545F555041535303030A1811077C16A55EBA11CB920C497FB801119A56 0 -26
ADV:01C09C35256377 02010609094C545F555041535303030A1811077C16A55EBA11CB920C497FB801119A56 0 -29
ADV:01D226C0A4B96D 02010609094C545F555041535303030A1811077C16A55EBA11CB920C497FB801119A56 0 -29
ADV:01C09C35256377 02010609094C545F555041535303030A1811077C16A55EBA11CB920C497FB801119A56 0 -32
ADV:0078BDBC6BDB31 1BFF750042040180207 8BDBC6BDB317ABDBC6BDB3001000000000000 0 -87
ADV:01D226C0A4B96D 02010609094C545F555041535303030A1811077C16A55EBA11CB920C497FB801119A56 0 -30

Scanning stopped via timeout
```

Figure 3: BLE Advert Scan Results

scan start 1000 0	Begins the BLE scanning process with the specified timeout (ms) and whitelist filter.

The resulting scan data shows two BLE devices advertising with the vSP service – 01D226C0A4B96D and 01C09C35256377. You may need to set the timeout longer depending on the advertising interval set for the vSP peripheral devices.

Note: Actual connection handles may vary depending on how many vSP devices are connected. Do not assume that your connection handle is the same as those shown in the screenshots contained within this document. The screenshots only show what type of interaction to expect.

vSP Connection

```
>connect 01D226C0A4B96D 10000 7500 10000 32000000

OK
>
 --- Connect: (0002FE01) handle=2
Conn Interval 10000
Conn Supervision Timeout 32000000
Conn Slave Latency 0
```

Figure 4: BLE vSP Connection

Connect 01D226C0A4B96D 10000 7500 10000 32000000	Initiates a connection to the specified device with the connection parameters provided – connection timeout (ms), minimum connection interval (us), maximum connection interval (us), and supervision timeout (us).

After establishing one vSP connection you can make further vSP connections, if required. Note that each connection has a unique connection handle.

Note: We assume you are familiar with vSP and have read the following document: *Application Note – Using VSP with smartBASIC*

ENABLING GATT CLIENT

Once connected to a vSP device, you must enable the GATT client.

```
>gattc open 0 0

OK
>
```

Figure 5: Open GATT Client

gattc open 0 0	Opens the GATT client with the ring buffer size for notify/indicate buffer and the flags set to ***0*** for automatic indication confirmations and ***1*** to disable automatic indication confirmations.

BLE (VSP) GATT TABLE

Once connected to a vSP device and with the GATT client open, you can query the GATT table using the following:

```
>gattc tablemap 1

S:1 ,(9) ,FE011800
 C:3 ,0000000A ,FE012A00 ,0
 C:5 ,00000002 ,FE012A01 ,0
 C:7 ,00000002 ,FE012A04 ,0
 C:9 ,00000002 ,FE012AA6 ,0
S:10 ,(13) ,FE011801
 C:12 ,00000020 ,FE012A05 ,0
  D:13 ,FE012902
S:14 ,(26) ,FE01180A
 C:16 ,00000002 ,FE012A29 ,0
 C:18 ,00000002 ,FE012A24 ,0
 C:20 ,00000002 ,FE012A25 ,0
 C:22 ,00000002 ,FE012A27 ,0
 C:24 ,00000002 ,FE012A26 ,0
 C:26 ,00000002 ,FE012A28 ,0
S:27 ,(65535) ,FD021101
 C:29 ,00000010 ,FD022000 ,0
  D:30 ,FE012902
 C:32 ,0000000C ,FD022001 ,0
 C:34 ,00000010 ,FD022002 ,0
  D:35 ,FE012902
 C:37 ,0000000C ,FD022003 ,0
OK
>
```

```
>gattc tablemap 2

S:1 ,(7) ,FE011800
 C:3 ,0000000A ,FE012A00 ,0
 C:5 ,00000002 ,FE012A01 ,0
 C:7 ,00000002 ,FE012A04 ,0
S:8 ,(11) ,FE011801
 C:10 ,00000020 ,FE012A05 ,0
  D:11 ,FE012902
S:12 ,(24) ,FE01180A
 C:14 ,00000002 ,FE012A29 ,0
 C:16 ,00000002 ,FE012A24 ,0
 C:18 ,00000002 ,FE012A25 ,0
 C:20 ,00000002 ,FE012A27 ,0
 C:22 ,00000002 ,FE012A26 ,0
 C:24 ,00000002 ,FE012A28 ,0
S:25 ,(65535) ,FD021101
 C:27 ,00000010 ,FD022000 ,0
  D:28 ,FE012902
 C:30 ,0000000C ,FD022001 ,0
 C:32 ,00000010 ,FD022002 ,0
  D:33 ,FE012902
 C:35 ,0000000C ,FD022003 ,0
OK
>
```

Figure 6: GATT tables for connection 1 and connection 2

gattc tablemap 1	***1*** is the connection handle of the device for which you want the GATT table.

Note: Both tables in Figure 6 contain the vSP service (0x1101); however, they are located at different attribute handle locations for connection 1 versus connection 2. This can occur when you are testing with multiple vSP device types.

Americas: +1-800-492-2320
Europe: +44-1628-858-940
Hong Kong: +852 2923 0600

For the tables in Figure 6, the following is used to show Service, Characteristics, and Descriptors:

Table 1: Service, characteristics, and descriptors

S:n	Service:handle
C:n	Characteristic:handle
D:n	Descriptor:handle

BLE (vSP) Read/Write

Note: The GATT handles shown in Table 1 may differ from the GATT handles in the table for your device. Be sure to match the vSP 16-bit offset UUIDs to make sure you have selected the right handle.

You can also read back individual attributes from the GATT table (Figure 6). Here we read back the value of handle 3 from the GATT table of connection handle 1. Looking at the GATT table (Figure 6), we can see that the UUID for this characteristic is 0x2A00 which is the device name, in this case being LT_UPASS.

```
>gattc read 1 3 0

OK
>
EVATTRREAD(hConn=130816,handle=3,status=0)
 >BleGattcReadData(data=4C545F5550415353,offset=0)
                  (data=LT_UPASS)
```

Figure 7: GATT client read

gattc read 1 3 0 | ***1*** is the connection handle, ***3*** is the attribute handle, and ***0*** is the offset.

To enable notifications for the vSP service you must identify the descriptor from the GATT table. 0x2000 is the 16-bit offset for the vSP TX characteristic; you can see its descriptor handle is 30 from the GATT table above. We use the writecmd to write 0x0001 (written in little endien) into the descriptor to enable notifications with an EVNOTIFYBUF, if successful. You can also enable notifications for modemout characteristics descriptor located at handle 35.

```
>gattc writecmd 1 30 0100

OK
>
EVNOTIFYBUF

>gattc writecmd 1 35 0100

OK
>
EVNOTIFYBUF
```

Figure 8: BLE GATT client Write command to enable notifications for characteristics

Americas: +1-800-492-2320
Europe: +44-1628-858-940
Hong Kong: +852 2923 0600

gattc writecmd 1 30 0100	***1*** is the connection handle, ***30*** is the attribute handle for the CCCD descriptor, and ***0100*** is the value in little endien.
gattc writecmd 1 35 0100	***1*** is the connection handle, ***35*** is the attribute handle for the CCCD descriptor, and ***0100*** is the value in little endien.

Whenever a notification is received from the GATT server, we receive a notification event identifying the connection handle on which it was received. We also receive the attribute handle and the data received.

Figure 9 shows data being received from connection handle 2 with the attribute handle being 27 (29 in Figure 10), vSP TX FiFo Characteristic.

```
EVATTRNOTIFY()
 >BleGattcNotifyRead(hConn=0002FE01,handle=27,Dumped=0,data=0D)
EVATTRNOTIFY()
 >BleGattcNotifyRead(hConn=0002FE01,handle=27,Dumped=0,data=686F77)
EVATTRNOTIFY()
 >BleGattcNotifyRead(hConn=0002FE01,handle=27,Dumped=0,data=647920706172)
EVATTRNOTIFY()
 >BleGattcNotifyRead(hConn=0002FE01,handle=27,Dumped=0,data=746E6572210D)
```

Figure 9: BLE vSP Notification Event at GATT Client from vSP Server, Connection handle 2

The next screenshot (Figure 10) shows data being received from connection handle 1.

```
EVATTRNOTIFY()
 >BleGattcNotifyRead(hConn=0001FF00,handle=29,Dumped=0,data=48)
EVATTRNOTIFY()
 >BleGattcNotifyRead(hConn=0001FF00,handle=29,Dumped=0,data=656C6C6F204920616D2066726F6D204561727
468)
EVATTRNOTIFY()
 >BleGattcNotifyRead(hConn=0001FF00,handle=29,Dumped=0,data=0D)
```

Figure 10: BLE vSP Notification Event at GATT Client from vSP Server, Connection handle 1

To write data to a vSP server from the BL652 master GATT client, use the **gattc write** and **gattc write$** commands to write data into the RX FiFo Characteristic (0x2001) attribute value.

```
>gattc write$ 1 32 "Hello how are you?"

OK
>
EVATTRWRITE(hConn=0001FF00,handle=32,status=0)
```

Figure 11: BLE vSP Write String Data to Connection 1, Attribute 32 (RX FiFo Characteristic 0x2001)

```
>gattc write 2 30 444546

OK
>
EVATTRWRITE(hConn=0002FE01,handle=30,status=0)
```

Figure 12: BLE vSP Write Hex Data to Connection 2, Attribute 30 (RX FiFo Characteristic 0x2001)

Americas: +1-800-492-2320
Europe: +44-1628-858-940
Hong Kong: +852 2923 0600

BLE vSP Disconnection

The following command can be used to disconnect any existing connection using its connection handle (Figure 13):

```
>disconnect 1

OK
>
 --- Disconnect: (0001FF00) handle=1
>disconnect 2

OK
>
 --- Disconnect: (0002FE01) handle=2
>
```

Figure 13: BLE vSP Disconnection

disconnect n	Causes the BLE connection where ***n*** is the connection handle to be terminated

Revision History

Version	Date	Notes	Approver
1.0	18 Oct 2017	Initial Release	Jonathan Kaye

Americas: +1-800-492-2320
Europe: +44-1628-858-940
Hong Kong: +852 2923 0600