



SILICON LABS

BLE Interoperability Test and Common Issue Diagnostic

APAC RA



Agenda

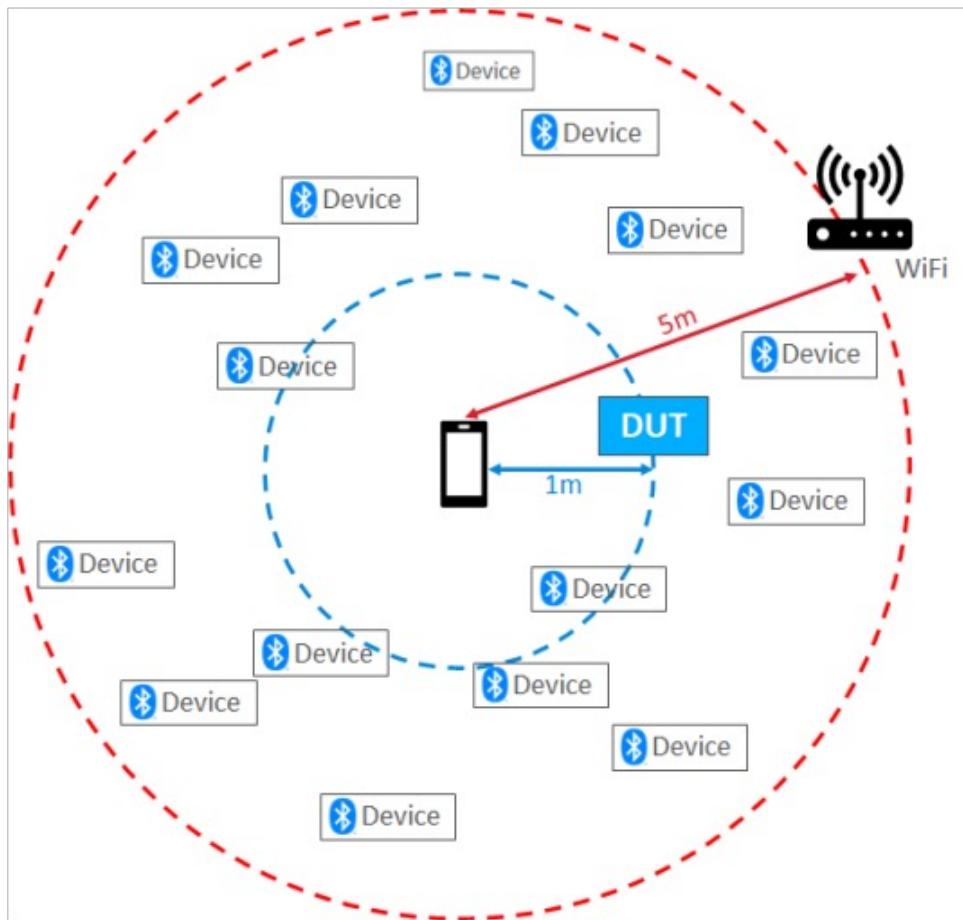
- Overview
- IOP Test
 - Prerequisites
 - Setup IOP Test
 - Running the IOP Test
 - Logging and Sharing Data
- Diagnostic Tools
 - UART Log
 - Network Analyzer
 - Smartphone btsnoop Log
 - Third Party Sniffer
 - Spectrum Analyzer
- Diagnostic Example
 - BLE Device Cannot Be Detected
 - Characteristic Write Cause Disconnect
 - Throughput

Overview



- IOP is a cornerstone of Bluetooth
- It enables end users to mix and match devices between different vendors
- It is essential that Bluetooth solution supplier can provide means to test IOP feature.
- Usually test with smartphones
- Issues that cause by lack of IOP testing
 - Device not found
 - Unreliable connection
 - Incorrect operation
 - Loss of data
- Silicon Labs provides framework to test IOP
- Silicon Labs have tested some phones

Silicon Labs Test Environment



- The Device under test (DUT) was placed about 1 m away from the mobile phone against which the tests were executed.
- The testing was conducted in a generic office environment with
- The closest WiFi access point was about 5 m away.
- About 15 other interferers in the 2.4 GHz spectrum were active during the testing and were randomly located in the 5m range.

Silicon Labs Test Case

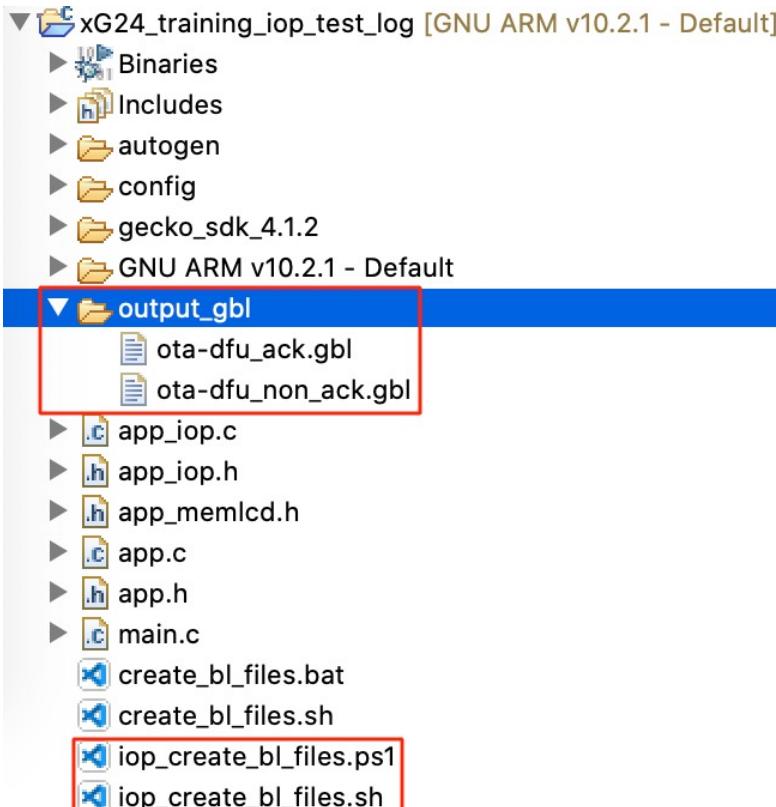
Test ID	Test Type	Test Sub Type
1	Bluetooth LE Scanning	
2	Bluetooth LE Connection	
3	Bluetooth LE Discovery	
4.1	GATT Operations	Read Only Length 1
4.2	GATT Operations	Read Only Length 255
4.3	GATT Operations	Write Only Length 1
4.4	GATT Operations	Write Only Length 255
4.5	GATT Operations	Write Without Response Length 1
4.6	GATT Operations	Write Without Response Length 255
4.7	GATT Operations	Notify length 1
4.8	GATT Operations	Notify length MTU - 3
4.9	GATT Operations	Indicate Length 1
4.10	GATT Operations	Indicate length MTU - 3
5.1	Characteristic	Length 1
5.2	Characteristic	Length 255
5.3	Characteristic	Length Variable 4
5.4	Characteristic	Const Length 1
5.5	Characteristic	Const Length 255
5.6	Characteristic	User Len 1
5.7	Characteristic	User Len 255
5.8	Characteristic	User Len Variable 4
6.1	OTA update	OTA update - Acknowledged write
6.2	OTA update	OTA update - Unacknowledged write
7	Throughput	Throughput - GATT Notification
8.1	Security and Encryption	Security - Pairing
8.2	Security and Encryption	Security - Authentication
8.3	Security and Encryption	Security - Bonding

- Detail test case refer to [AN1309](#)
- Tested phone model and test result also see on [AN1309](#)

Prerequisites

- **Hardware Requirements**
 - Almost any kit that supports Bluetooth
- **Software Requirements**
 - IOP example application is available on Bluetooth SDK 3.3.0 or newer
 - Bluetooth SDK that is part of the GSDK
- **Mobile App Requirements**
 - IOP testing is supported on EFR Connect app, version 2.4 or newer
 - Available for both [Android](#) and [iOS](#)
 - Source(and APK) is available on [GitHub](#)
 - Remove bonded information before start the IOP test
- **Minimum Mobile Operating System Versions**
 - Android™ 9
 - iOS®12

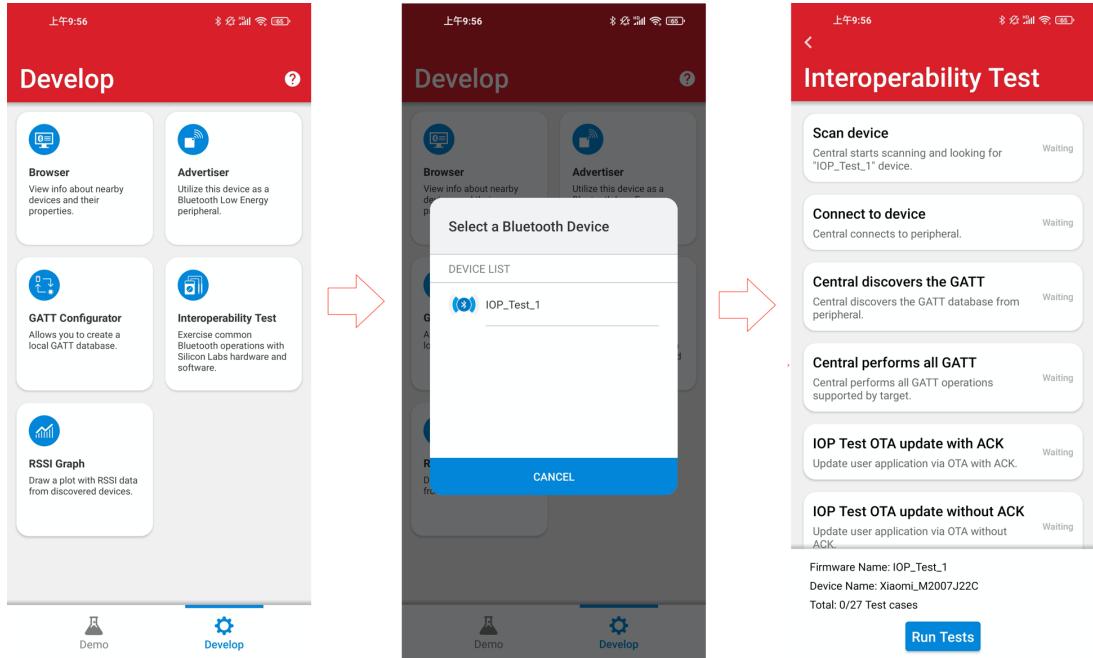
Setup IOP Test on EFR32



```
[I] Stack version: 4.2.0-b321.  
[I] Public device address: 1c:34:f1:de:25:c0.  
[I] All bondings deleted.  
[I] Advertising started.
```

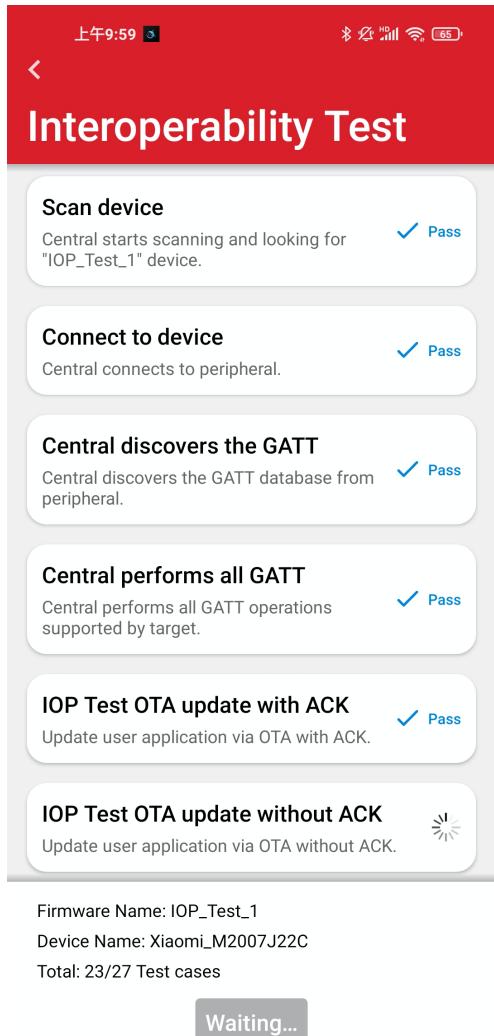
- Create interoperability test example
- Generate GBL file for OTA testing
- Flash hex file, get ready for testing
- SW run on EFR32, Check UART log or LCD

Setup IOP Test on Smartphone



- **Launch EFR Connect App**
- **Navigate to Develop view**
- **Tap “Interoperability Test”, list up nearby boards**
- **Tap the board to test**
- **Tap “Run Tests” for get started**

Running the IOP Test



- **Scrolls through the test case**
- **Indicates Pass/Fail**
- **Run without intervention, except OTA and security**

Logging and Sharing Data #1

```
<timestamp>2022-09-30 15:09:15</timestamp>
<phone_informations>
    <phone_name>Xiaomi_M2007J22C</phone_name>
    <phone_os_version>Android_SDK: 10_29</phone_os_version>
</phone_informations>
<firmware_informations>
    <firmware_original_version>4.2.0</firmware_original_version>
    <firmware_ota_ack_version>4.2.0</firmware_ota_ack_version>
    <firmware_ota_non_ack_version>4.2.0</firmware_ota_non_ack_version>
    <firmware_ic_name>UNKNOWN</firmware_ic_name>
    <firmware_name>IOP_Test_1</firmware_name>
</firmware_informations>
<connection_parameters>
    <mtu_size>247</mtu_size>
    <pdu_size>251</pdu_size>
    <interval>15.0</interval>
    <latency>0</latency>
    <supervision_timeout>20000</supervision_timeout>
</connection_parameters>
<test_results>
    Test case 1,Pass.
    Test case 2,Pass.
    Test case 3,Pass.
    Test case 4.1,Pass.
    Test case 4.2,Pass.
    Test case 4.3,Pass.
    Test case 4.4,Pass.
    Test case 4.5,Pass.
    Test case 4.6,Pass.
    Test case 4.7,Pass, (Testing time: 123ms;Acceptable time: 300ms).
    Test case 4.8,Pass, (Testing time: 130ms;Acceptable time: 300ms).
    Test case 4.9,Pass, (Testing time: 116ms;Acceptable time: 300ms).
    Test case 4.10,Pass, (Testing time: 121ms;Acceptable time: 300ms).
    Test case 5.1,Pass.
    Test case 5.2,Pass.
    Test case 5.3,Pass.
    Test case 5.4,Pass.
    Test case 5.5,Pass.
    Test case 5.6,Pass.
    Test case 5.7,Pass.
    Test case 5.8,Pass.
    Test case 6.1,Pass.
    Test case 6.2,Pass.
    Test case 7.1,N/A, (Throughput: 79381 Bytes/s;Acceptable Throughput: 42293 Bytes/s).
    Test case 7.2,Pass.
    Test case 7.3,Pass.
    Test case 7.4,Pass.
</test_results>
```

- **Test result can be shared in xml format**
- **Contains information about the phone model, OS version, Bluetooth connection parameters, and the result of each test**

Logging and Sharing Data #2

IOP test report

Please help to share your test result

Hi, Eric. When you submit this form, the owner will see your name and email address.

* Required

1. Test result *

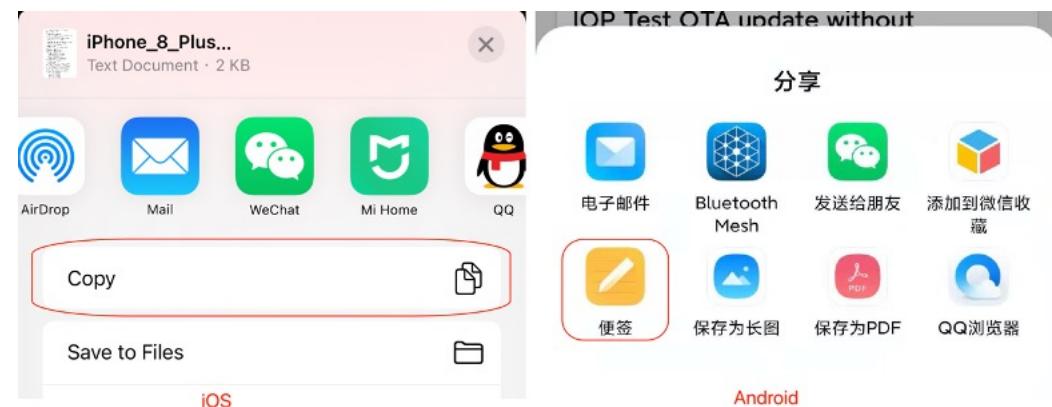
All pass
 Some items fail

2. Log *

Enter your answer

Submit

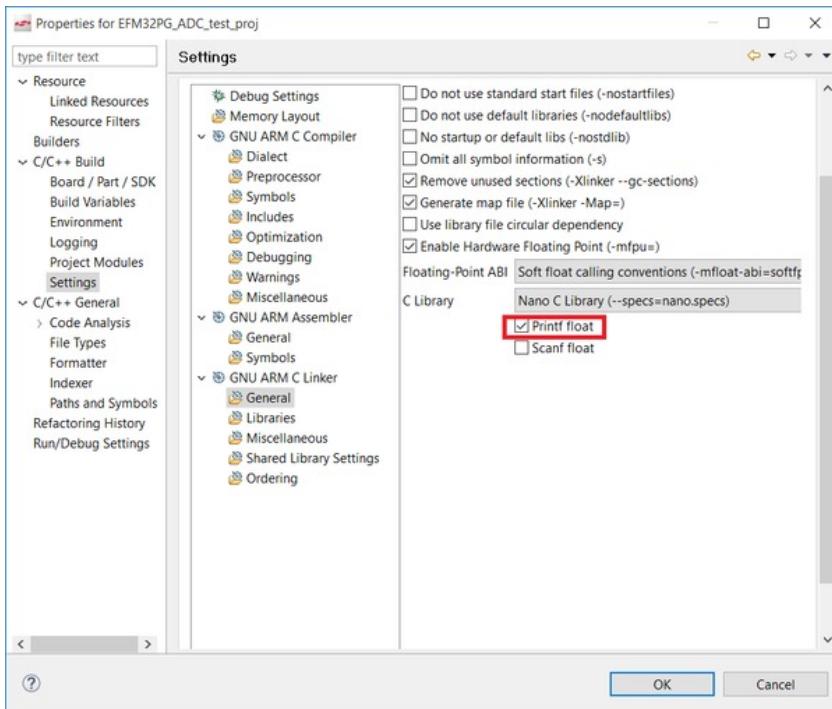
- Share the test result to [IOP test report](#)



IOP Hands On

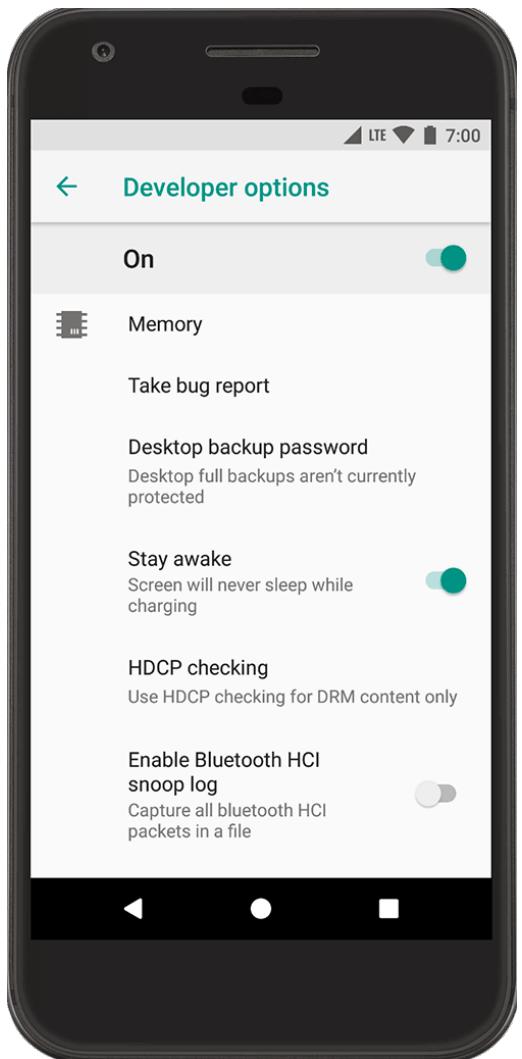
UART Log

[I] Stack version: 4.2.0-b321.
[I] Public device address: 1c:34:f1:de:25:c0.
[I] All bondings deleted.
[I] Advertising started.



- **UART Log, dependence on software components**
 - Services -> IO Stream -> IO Stream: USART
 - Application -> Utility -> Log
- **Enable floating point printf(), refer to [KBA](#).**

BTSNOOP Log #1



■ Smartphone btsnoop Log

- For Android device only
- Different way to get the log for each model
- Redmi Note 9, start and stop the snoop log by input *###5959#*#* on keypad

BTSNOOP Log #2

ComProbe Protocol Analysis System - BT_HCI_2022_0608_111228.cfa

Capture file: C:\Users\Silabs\project\Customer support\00227230_connection\bt_log20220608_111258_disconnect from smartphone\CsLog_2022_0608_111228\BT_HCI_2022_0608_111228.cfa

Frame Display - BT_HCI_2022_0608_111228.cfa

Frame 543: [Host] Len=7

HCI UART

HCI Packet Type: Command Packet

HCI

Packet from: Host

HCI Command

Opcode: 0x0406

Opcode Group (OGF): Link Control command

Command: HCI_Disconnect

Total Length: 3

Connection_Handle: 0x0200

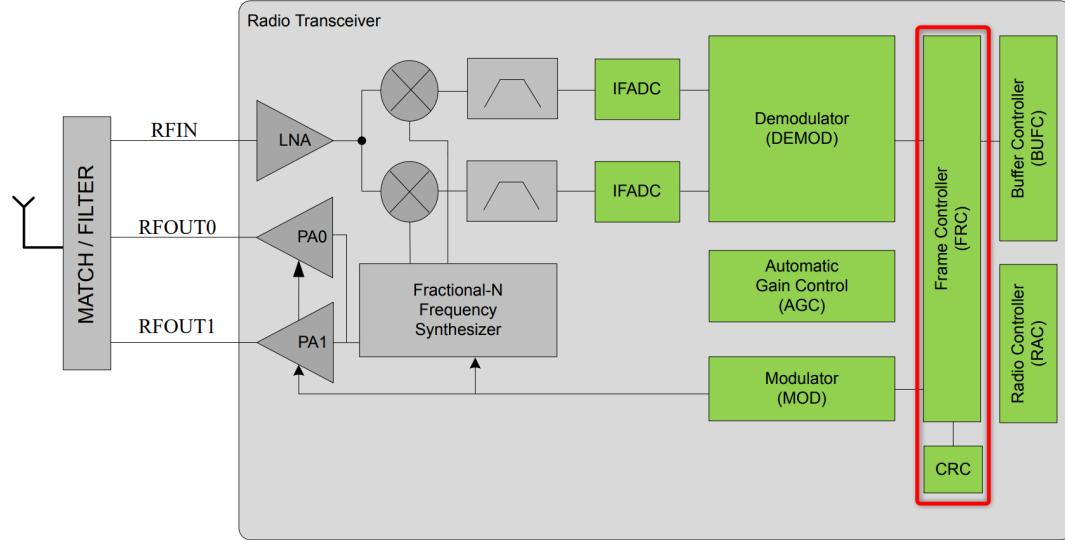
Reason: Remote User Terminated Connection

Frame# Type Opcode Opcode Group Opcode Command Event

Bookmark	Frame#	Type	Opcode	Opcode Group	Opcode Command	Event
	524	Event	0x1405	Status Params	HCI_Read_RSSI	HCI_Corr
	525	Command	0x1405	Status Params	HCI_Read_RSSI	HCI_Corr
	526	Event	0x1405	Status Params	HCI_Read_RSSI	HCI_Corr
	527	Command	0x1405	Status Params	HCI_Read_RSSI	HCI_Corr
	528	Event	0x1405	Status Params	HCI_Read_RSSI	HCI_Corr
	529	Command	0x1405	Status Params	HCI_Read_RSSI	HCI_Corr
	530	Event	0x1405	Status Params	HCI_Read_RSSI	HCI_Corr
	531	Command	0x1405	Status Params	HCI_Read_RSSI	HCI_Corr
	532	Event	0x1405	Status Params	HCI_Read_RSSI	HCI_Corr
	533	Command	0x1405	Status Params	HCI_Read_RSSI	HCI_Corr
	534	Event	0x1405	Status Params	HCI_Read_RSSI	HCI_Corr
	535	Command	0x1405	Status Params	HCI_Read_RSSI	HCI_Corr
	536	Event	0x1405	Status Params	HCI_Read_RSSI	HCI_Corr
	537	Command	0x1405	Status Params	HCI_Read_RSSI	HCI_Corr
	538	Event	0x1405	Status Params	HCI_Read_RSSI	HCI_Corr
	539	Command	0x1405	Status Params	HCI_Read_RSSI	HCI_Corr
	540	Event	0x1405	Status Params	HCI_Read_RSSI	HCI_Corr
	541	Command	0x1405	Status Params	HCI_Read_RSSI	HCI_Corr
	542	Event	0x1405	Status Params	HCI_Read_RSSI	HCI_Corr
	543	Command	0x0406	Link Control	HCI_Disconnect	HCI_Corr
	544	Event	0x0406	Link Control	HCI_Disconnect	HCI_Corr
	545	Event				HCI_Dis

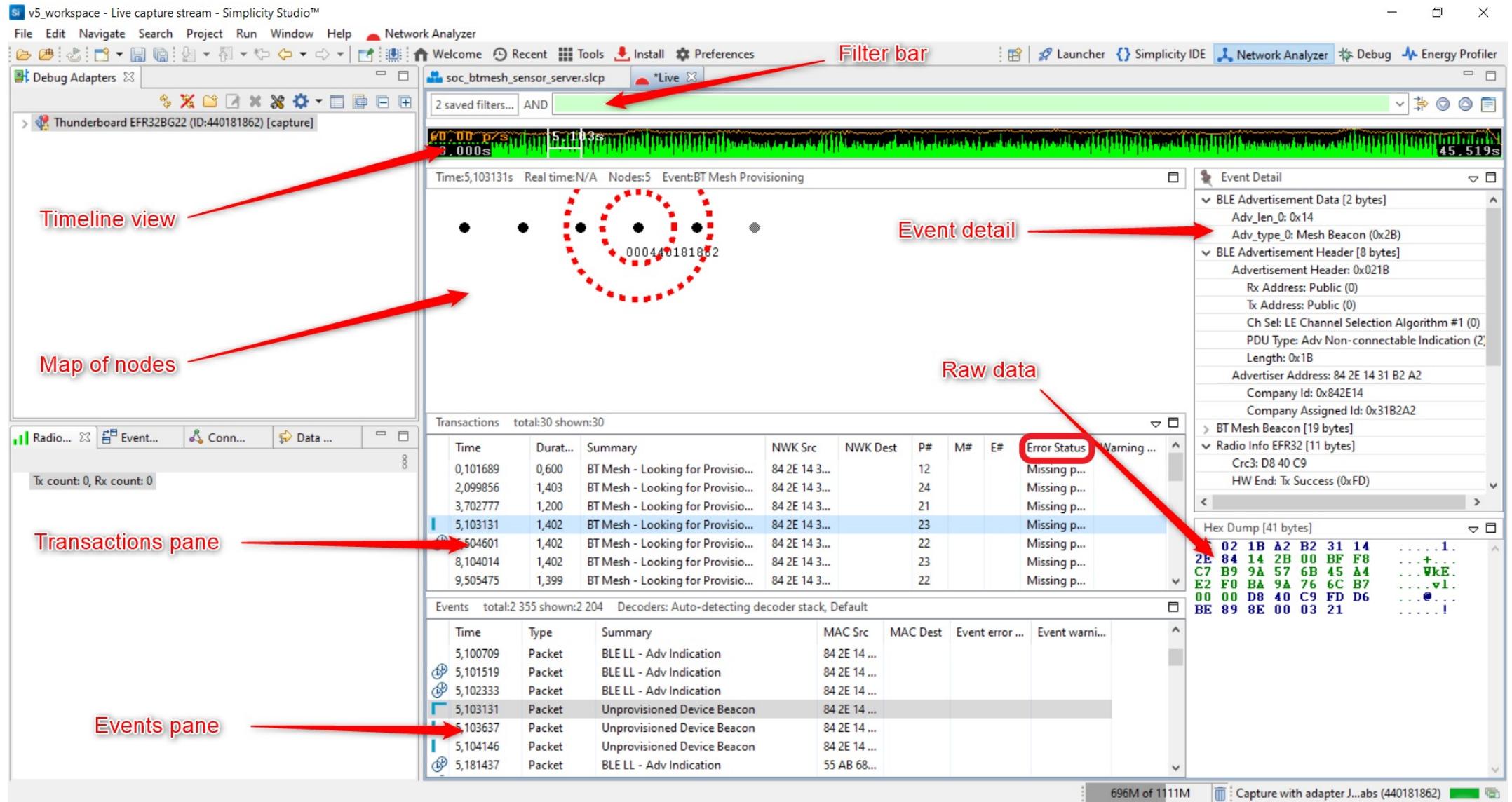
Total Frames: 545 | Frames Filtered In: 545 | Frame #s Selected: 543; (1 total)

Network Analyzer #1



- **Network Analyzer**
 - Tap into the data buffers of the radio transceiver
 - Via Packet Trace Interface (PTI)

Network Analyzer #2

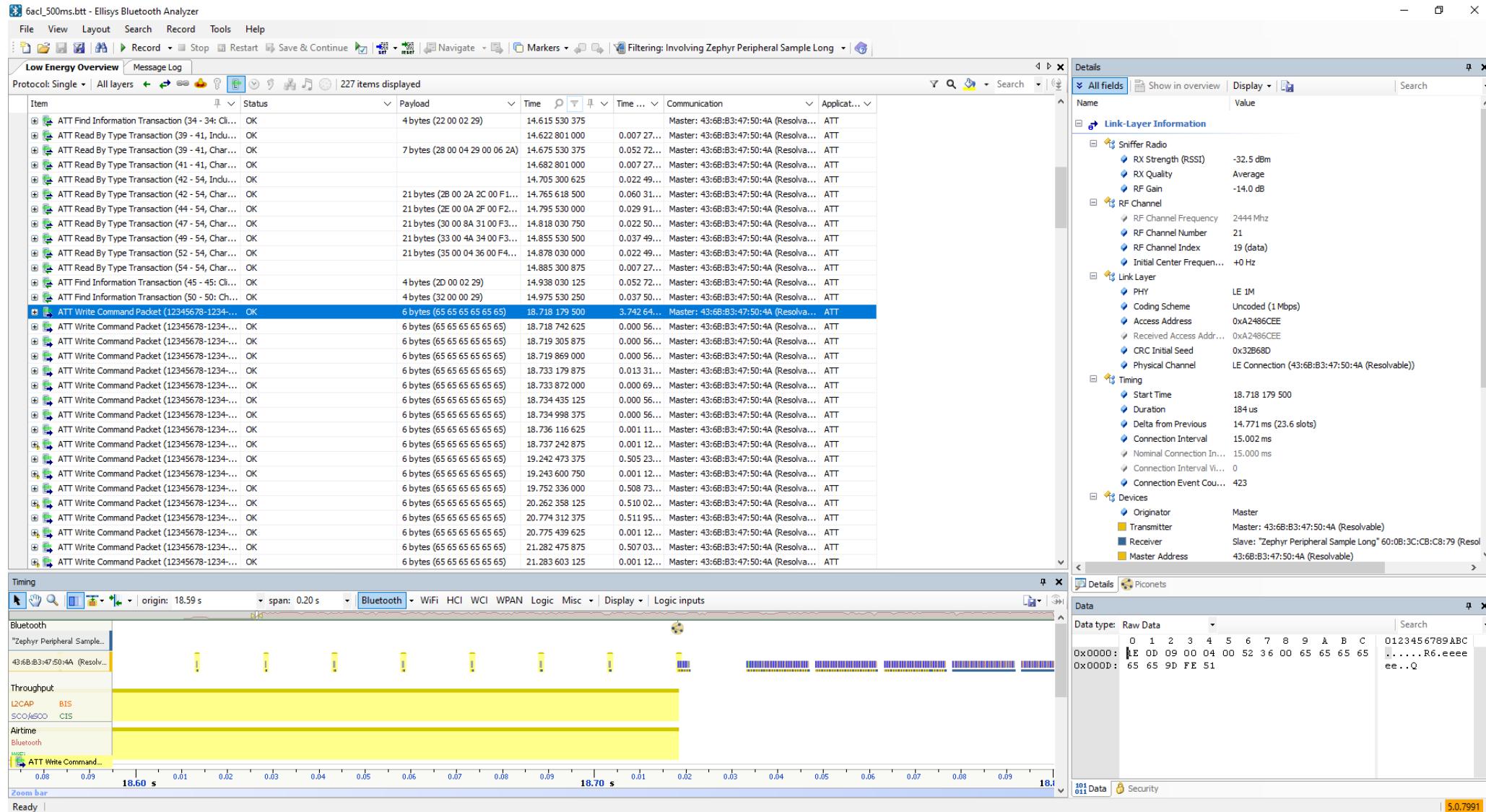


Ellisys Bluetooth tracker #1



- Third Party Sniffer
 - [Ellisys Bluetooth tracker](#), supports BLE air data concurrent capture

Ellisys Bluetooth tracker #2

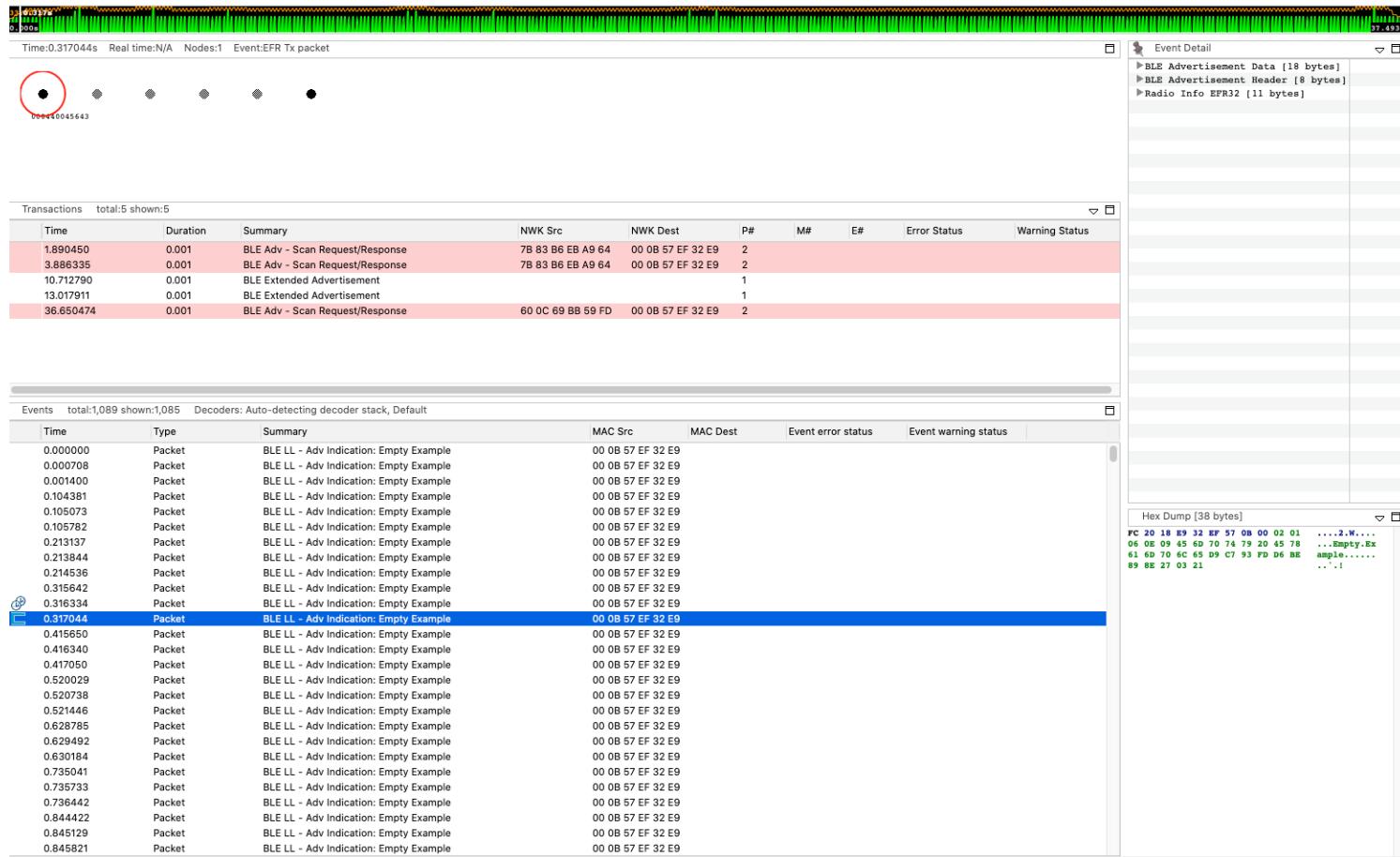


Spectrum Analyzer



- **Spectrum Analyzer**
 - Check the RF feature, TX power, frequency

BLE Device Cannot Be Detected #1



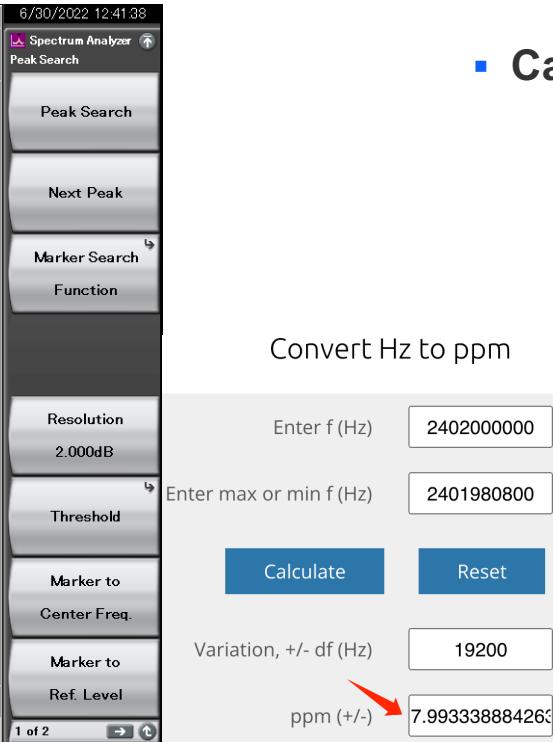
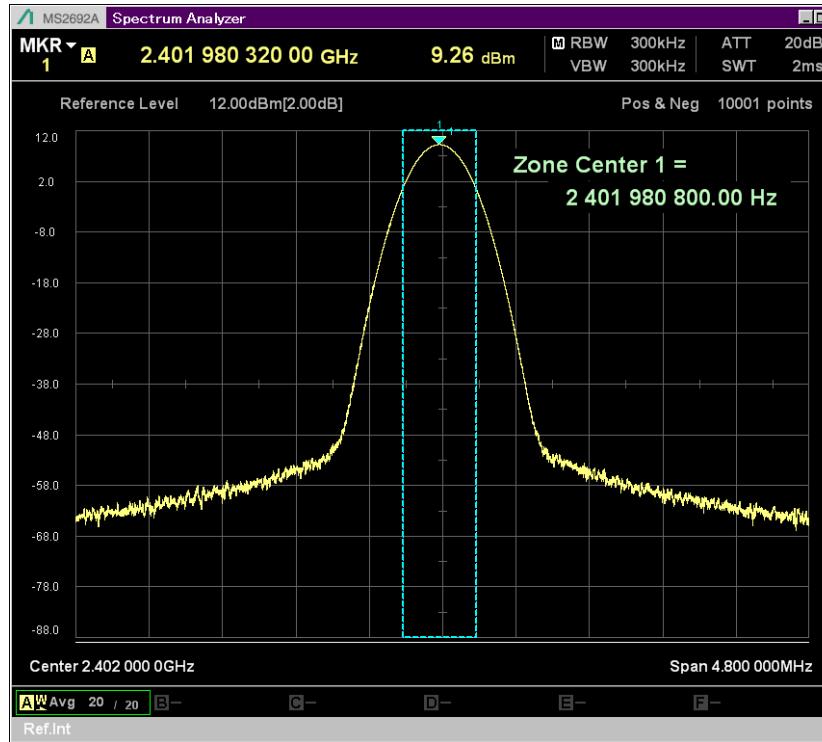
Phenomenon

- UART log indicate device have system boot and advertising
- Network Analyzer can see the advertising
- Just not able to detect by scan device, like smartphone

BLE Device Cannot Be Detected #2

BLE Device Cannot Be Detected #3

BLE Device Cannot Be Detected #4



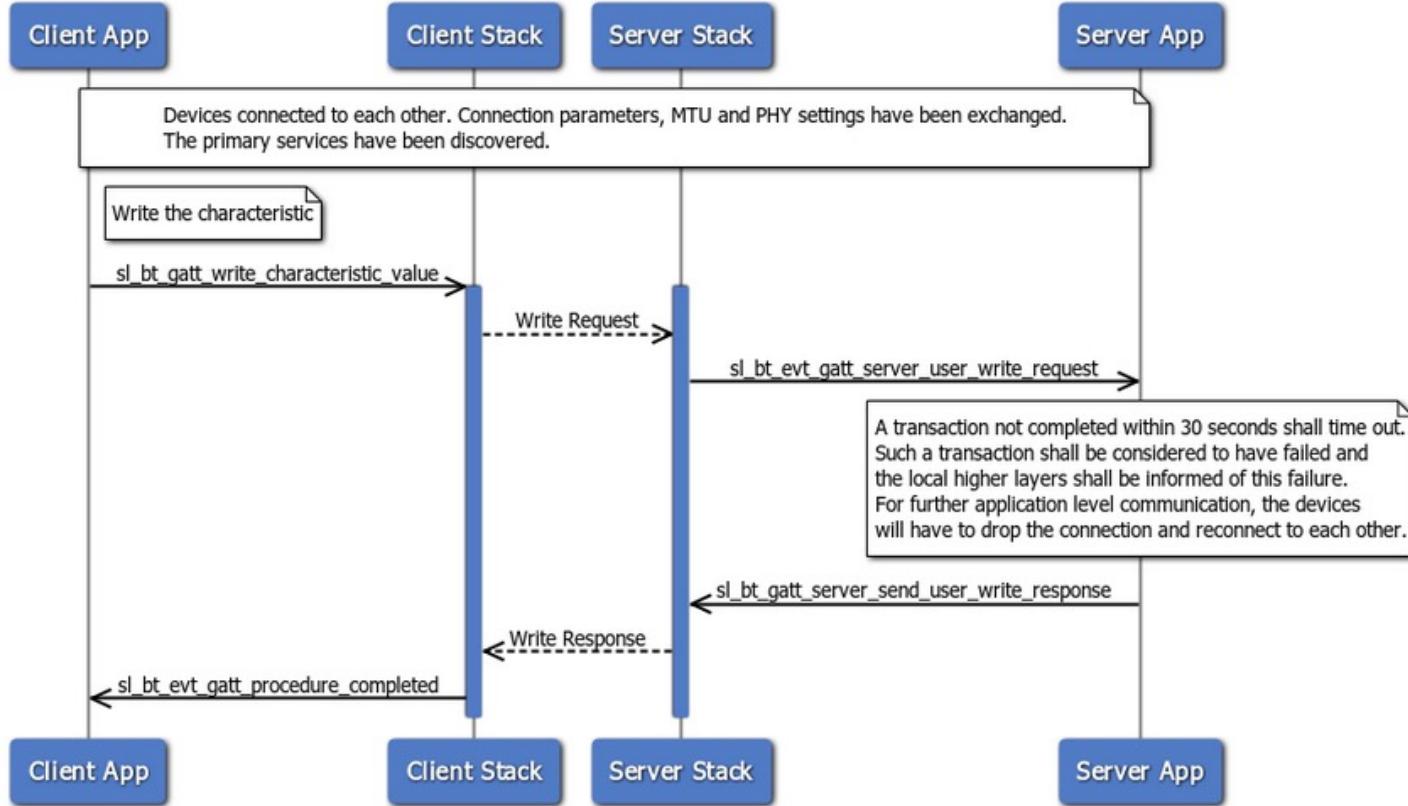
▪ Cause Analyze

- Clock accuracy, the drift must be not more than ± 50 ppm
- Check central frequency
- Finetune if needed
 - ▶ commander ctune get
 - ▶ commander ctune set --value xxx
- Use crystal those list on [AN0016.2](#) for custom board

BLE Device Cannot Be Detected #4

Characteristic Write Cause Disconnect #1

Characteristic Write Cause Disconnect #2



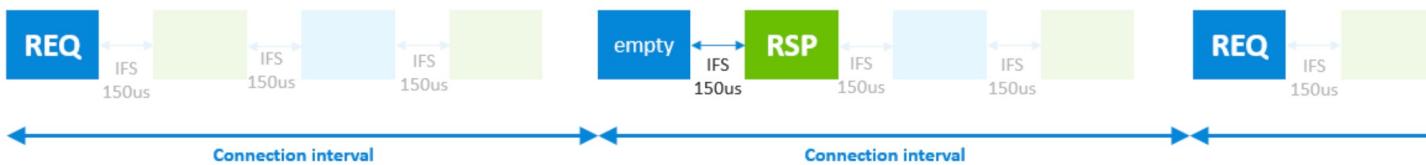
- Usually happen on User Types of Characteristics operation
- User Types means users should be responsible for allocating, maintaining, and freeing a suitable buffer for the characteristic value
- Respond to the GATT write/read requests by sending write/read response back to the peer device
- If not send write /read response to the client, the connection will terminate after 30 seconds

Characteristic Write Cause Disconnect #3

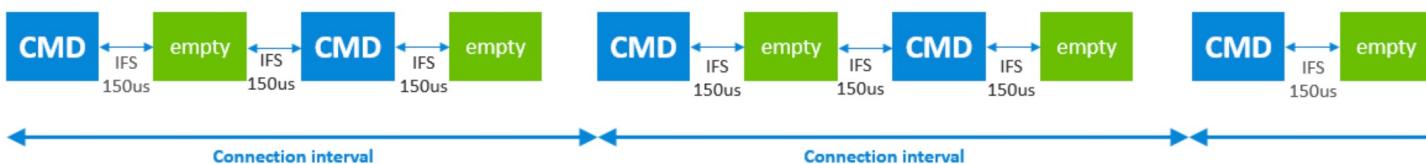
Characteristic Write Cause Disconnect #4

Throughput #1

1. Acknowledged data transfer. In this use case, the reception of all data packets is acknowledged by the receiver. The receiver sends a response for every read/write request in the next connection interval. The connection is reliable but the throughput is low.



2. Unacknowledged data transfer. In this use case, packets can be sent sequentially without waiting for acknowledgment from the other side. This ensures much higher throughput, but a less reliable connection.



$$\text{Throughput} = \frac{1000\text{mSecs} * \text{Number of Packets in a Connection Interval} * \text{Data Per Packet}}{\text{Connection Interval (mSecs)}}$$

- **Acknowledged vs. Unacknowledged**
- **Maximum number of packets per connection event is dependent on the BLE stack/chipsets**
- **Change connection interval can not change above Maximum number of packets**
- **Impact the Throughput performance**

Throughput #2



Thank You!

www.silabs.com

