



OpenLCB Standard

BLE Transfer

August 18, 2024

Concept

1 Introduction (Informative)

The OpenLCB suite of protocols can be used on multiple physical transports; this document defines the specifics that relate to using Bluetooth Low Energy (BLE) as a transport layer. The BLE protocol uses broadcast messages and connection based transfer of attribute data. This Standard defines unique broadcast messages called advertisements to identify the presence of OpenLCB nodes supporting BLE and the exchange of data using attributes over connections.

2 Intended Use (Informative)

This Standard is intended for use whenever OpenLCB nodes are communicating using BLE. It is not intended to cover OpenLCB communications over other types of communications links.

3 References and Context

This specification is in the context of the following OpenLCB-CAN Standards:

- The OpenLCB Message Network Standard, which specifies the general OpenLCB message format

For more information about BLE, please refer to the Bluetooth Special Interest Group.

For the purposes of this standard, a BLE “device” implements the BLE Peripheral role and uses BLE Advertisements to both announce itself and accept connections. A “client” implements the BLE Central role and uses BLE scanning to find BLE devices to establish connections to.

4 Message Formats

There exists two possible ways in which a node using BLE transport can join the network.

- The Streaming method utilizes the general OpenLCB message format transported over a pair of BLE attributes. This method requires a full OpenLCB stack implementation on the BLE device.
- The Simplified method assumes that the BLE device implements one or more very specific applications. It uses one or more BLE attributes to exchange information specific to the given application. This method does not require a full OpenLCB stack implementation on the BLE device.

All multi-byte advertising fields are in little-endian format, consistent with BLE conventions.

4.1 BLE UUIDs

30

Version 4 BLE UUIDs are used by OpenLCB services and characteristics. A Version 4 UUID is a 128-bit universally unique identifier that is generated using random numbers.

| <u>OpenLCB BLE UUIDs</u> | |
|--|---|
| <u>Description</u> | <u>UUID</u> |
| <u>Streaming Service</u> | <u>0ff45220-84a9-4daf-83e7-da4c828d1851</u> |
| <u>Streaming Active Characteristic</u> | |
| <u>Streaming Data In Characteristic</u> | <u>b029e1ba-47bc-4d8a-a28e-fbe167fc06cb</u> |
| <u>Streaming Data Out Characteristic</u> | <u>c4fed917-55d9-485d-bf30-2eb3c095a90f</u> |
| <u>Simplified Services</u> | |
| <u>Node Information Service</u> | |
| <u>Node Manufacturer Name</u> | |
| <u>Node Model Name</u> | |
| <u>Node Hardware Version</u> | |
| <u>Node Software Version</u> | |
| <u>Node User Name</u> | |
| <u>Node User Description</u> | |
| <u>emory Configuration Service</u> | |
| <u>emory Configuration Command</u> | |
| <u>emory Configuration Reply</u> | |
| <u>Train Service</u> | |
| <u>Train Incoming Message</u> | |
| <u>Train Outgoing Message</u> | |

4.2 Streaming

If a node supports streaming protocol, it shall implement the following advertising format to announce itself and accept connections.

Legacy Advertising Payload (Connectable Undirected)

| Bytes | Field | Payload |
|-------------|--------------------------|---|
| <u>1</u> | <u>flags length</u> | <u>2</u> |
| <u>1</u> | <u>flags type</u> | |
| <u>1</u> | <u>flags</u> | |
| <u>1</u> | <u>local name length</u> | <u>2-27</u> |
| <u>1</u> | <u>local name type</u> | Use “short” type if the OpenLCB Node user name does not fit. Use “complete” type if the OpenLCB Node user name does fit. |
| <u>1-26</u> | <u>local name</u> | Up to the first 26 bytes of the OpenLCB Node user name, null terminated if length is less than 26 bytes. Shall not contain a ‘:’ or ‘;’ character. ‘:’ or ‘;’ characters shall be replaced by spaces. |

Legacy Scan Response Payload

| Bytes | Field | Payload |
|-----------|--------------------------------|--|
| <u>1</u> | <u>service data 128 length</u> | <u>27</u> |
| <u>1</u> | <u>service data 128 type</u> | |
| <u>16</u> | <u>128-bit UUID</u> | Unique Identifier identifying this as having an OpenLCB Streaming Service (and more). <u>0ff45220-84a9-4daf-83e7-da4c828d1851</u> |
| <u>6</u> | <u>OpenLCB Node ID</u> | <u>48-bit OpenLCB Node ID.</u> |
| <u>4</u> | <u>OpenLCB PIP</u> | If PIP exceeds four bytes, then a connection is required to get the rest. |

OpenLCB BLE Transfer Standard

If a node supports streaming protocol, it shall implement the following BLE GATT Service.

| <u>OpenLCB Streaming Service Attribute Table</u> | | | | |
|--|---|--------------------|---------------------|--|
| <u>UUID Length (bits)</u> | <u>UUID</u> | <u>Permissions</u> | <u>Size (bytes)</u> | <u>Value</u> |
| <u>16</u> | <u>Primary Service</u> | <u>Read</u> | <u>16</u> | <u>Streaming Service UUID</u> <u>0ff45220-84a9-4daf-83e7-da4c828d1851</u> |
| <u>16</u> | <u>Characteristic Declaration</u> | <u>Read</u> | <u>1</u> | <u>Characteristic Property write</u> |
| <u>128</u> | <u>Streaming Active Characteristic</u> | <u>Write</u> | <u>1</u> | <u>0: Streaming connection not active</u> <u>1: Streaming connection active</u> |
| <u>16</u> | <u>Characteristic Declaration</u> | <u>Read</u> | <u>1</u> | <u>Characteristic Property write</u> |
| <u>128</u> | <u>Streaming Data In Characteristic</u> <u>b029e1ba-47bc-4d8a-a28e-fbe167fc06cb</u> | <u>Write</u> | <u>244</u> | <u>OpenLCB message data to the device.</u> |
| <u>16</u> | <u>Characteristic Declaration</u> | <u>Read</u> | <u>1</u> | <u>Characteristic Property read notify ack</u> |
| <u>128</u> | <u>Streaming Data out Characteristic</u> <u>c4fed917-55d9-485d-bf30-2eb3c095a90f</u> | <u>Read</u> | <u>244</u> | <u>OpenLCB message date from the device</u> |

OpenLCB BLE Transfer Standard

| | | | | |
|-----------|---|-------------------|----------|--|
| <u>16</u> | <u>Client Characteristic Configuration Descriptor</u> | <u>Read/Write</u> | <u>2</u> | |
|-----------|---|-------------------|----------|--|

40 | **4.3 Simplified**

If a node supports one or more of the simplified protocols, it shall implement the following advertising format to announce itself and accept connections.

Legacy Advertising Payload (Connectable Undirected)

| <u>Bytes</u> | <u>Field</u> | <u>Payload</u> |
|---------------------|--------------------------|---|
| <u>1</u> | <u>flags length</u> | <u>2</u> |
| <u>1</u> | <u>flags type</u> | |
| <u>1</u> | <u>flags</u> | |
| <u>1</u> | <u>local name length</u> | <u>2-27</u> |
| <u>1</u> | <u>local name type</u> | Use “short” type if the OpenLCB Node user name does not fit. Use “complete” type if the OpenLCB Node user name does fit. |
| <u>1-26</u> | <u>local name</u> | Up to the first 26 bytes of the OpenLCB Node user name, null terminated if length is less than 26 bytes. |

Legacy Scan Response Payload

| <u>Bytes</u> | <u>Field</u> | <u>Payload</u> |
|---------------------|--------------------------------|--|
| <u>1</u> | <u>service data 128 length</u> | <u>27 - 30</u> |
| <u>1</u> | <u>service data 128 type</u> | |
| <u>16</u> | <u>128-bit UUID</u> | Unique Identifier identifying this as having one or more OpenLCB Simplified Services (and more). |
| <u>6</u> | <u>OpenLCB Node ID</u> | 48-bit OpenLCB Node ID. |
| <u>4</u> | <u>OpenLCB PIP</u> | If PIP exceeds four bytes, then a connection is required to get the rest. |

OpenLCB BLE Transfer Standard

- 45 | If a node supports one or more of the simplified protocols, it shall implement one or more of the following BLE GATT Services.

| <u>OpenLCB Node Information Attribute Table</u> | | | | |
|---|---|--------------------|---------------------|--------------------------------------|
| <u>UUID Length (bits)</u> | <u>UUID</u> | <u>Permissions</u> | <u>Size (bytes)</u> | <u>Value</u> |
| <u>16</u> | <u>Primary Service</u> | <u>Read</u> | <u>16</u> | <u>Node Information Service UUID</u> |
| <u>16</u> | <u>Characteristic Declaration</u> | <u>Read</u> | <u>1</u> | <u>Characteristic Property read</u> |
| <u>128</u> | <u>Manufacturer Name</u> | <u>Read</u> | <u>41</u> | <u>Null terminated string</u> |
| <u>16</u> | <u>Characteristic Declaration</u> | <u>Read</u> | <u>1</u> | <u>Characteristic Property read</u> |
| <u>128</u> | <u>Model Name</u> | <u>Read</u> | <u>41</u> | <u>Null terminated string</u> |
| <u>16</u> | <u>Characteristic Declaration</u> | <u>Read</u> | <u>1</u> | <u>Characteristic Property read</u> |
| <u>128</u> | <u>Hardware Revision Characteristic</u> | <u>Read</u> | <u>21</u> | <u>Null terminated string</u> |
| <u>16</u> | <u>Characteristic Declaration</u> | <u>Read</u> | <u>1</u> | <u>Characteristic Property read</u> |
| <u>128</u> | <u>Software Version Characteristic</u> | <u>Read</u> | <u>21</u> | <u>Null terminated string</u> |
| <u>16</u> | <u>Characteristic Declaration</u> | <u>Read</u> | <u>1</u> | <u>Characteristic Property read</u> |
| <u>128</u> | <u>Node User Name Characteristic</u> | <u>Read</u> | <u>63</u> | <u>Null terminated string</u> |

OpenLCB BLE Transfer Standard

| | | | | |
|------------|---|-------------|-----------|-------------------------------------|
| <u>16</u> | <u>Characteristic Declaration</u> | <u>Read</u> | <u>1</u> | <u>Characteristic Property read</u> |
| <u>128</u> | <u>Node User Description Characteristic</u> | <u>Read</u> | <u>64</u> | <u>Null terminated string</u> |

| <u>OpenLCB Memory Configuration Attribute Table</u> | | | | |
|---|---|--------------------|---------------------|--|
| <u>UUID Length (bits)</u> | <u>UUID</u> | <u>Permissions</u> | <u>Size (bytes)</u> | <u>Value</u> |
| <u>16</u> | <u>Primary Service</u> | <u>Read</u> | <u>16</u> | <u>Memory Configuration Service UUID</u> |
| <u>16</u> | <u>Characteristic Declaration</u> | <u>Read</u> | <u>1</u> | <u>Characteristic Property write</u> |
| <u>128</u> | <u>Command</u> | <u>Write</u> | <u>72</u> | |
| <u>16</u> | <u>Characteristic Declaration</u> | <u>Read</u> | <u>1</u> | <u>Characteristic Property read notify ack</u> |
| <u>128</u> | <u>Reply</u> | <u>Read</u> | <u>72</u> | |
| <u>16</u> | <u>Client Characteristic Configuration Descriptor</u> | <u>Read/Write</u> | <u>2</u> | |

| <u>OpenLCB Train Attribute Table</u> | | | | |
|--------------------------------------|--|--|--|--|
|--------------------------------------|--|--|--|--|

OpenLCB BLE Transfer Standard

| <u>UUID Length (bits)</u> | <u>UUID</u> | <u>Permissions</u> | <u>Size (bytes)</u> | <u>Value</u> |
|---------------------------|---|--------------------|---------------------|--|
| <u>16</u> | <u>Primary Service</u> | <u>Read</u> | <u>16</u> | <u>Train Service UUID</u> |
| <u>16</u> | <u>Characteristic Declaration</u> | <u>Read</u> | <u>1</u> | <u>Characteristic Property write</u> |
| <u>128</u> | <u>Incoming Message</u> | <u>Write</u> | <u>11</u> | <u>Train Control Message</u> |
| <u>16</u> | <u>Characteristic Declaration</u> | <u>Read</u> | <u>1</u> | <u>Characteristic Property read notify ack</u> |
| <u>128</u> | <u>Outgoing Message</u> | <u>Read</u> | <u>11</u> | <u>Train Control Message</u> |
| <u>16</u> | <u>Client Characteristic Configuration Descriptor</u> | <u>Read/Write</u> | <u>2</u> | |

5 States

5.1 General

2.5 ±0.5 seconds after a connection is established, the device shall request an update of the connection parameters within the bounds provided in Section 6.2 below. The client shall respond with one of the following:

1. Rejection of connection parameter update.
2. Request to update connection parameters within the bounds requested by the device.
3. Request to update connection parameters outside the bounds requested by the device.
 1. In this case, the connection parameters must still fall within the bounds provided in Section 6.2 below.

The client may skip the connection parameter update request if the initial connection parameters are already acceptable to it.

Upon initiation of a connection, a device shall disable all advertisements and not accept new connections until either:

1. All connections are terminated.
2. At least 5 seconds has past since the last connection initiation.

5.2 Streaming

A connection is not required for basic device interaction. A client may “proxy” information contained in the advertising and scan response payload over the OpenLCB network without an active connection having been established. A client may initiate a connection with a device if:

1. The device is sent an addressed message (routed through the client).

A client is only required to initiate a connection with a device if it has the capacity to establish more BLE connections and one of:

1. The device is sent an addressed message (routed through the client) and the client is previously paired to the device.
2. The client is paired to the device and configured to force connect upon discovery.

The client may automatically terminate connections to devices if a timeout occurs with no OpenLCB messages exchanged between the device and client, and the client is not paired to the device and also configured to force connect upon discovery.

A client shall not proxy information contained in the advertising and scan response payload over the OpenLCB network if does not have the capacity to establish any more BLE connections.

A device shall only support one active streaming connection at any given time. The lifetime of an active streaming connection is as follows:

1. Device starts advertising.
2. Client scans for and receives device advertisement.
- 3.[2.] Client initiates a BLE connection to the device, device stops advertising.
- 85 4. Client exchanges MTU with device.
5. Client discovers OpenLCB Streaming Service handles.
6. Client enables notifications on Streaming Data Out characteristic.
7. Client writes a value of 1 to the Streaming Active characteristic.
- 8.[3.] Device sends Streaming Active characteristic write response.
- 90 9. Device moves node to the initialized state upon write of value 1 to Streaming Active characteristic and begins sending OpenLCB messages over the Streaming Data Out characteristic.
10. Upon receiving Streaming Active characteristic write response from the device, client begins sending OpenLCB messages over the Streaming Data In characteristic.

95 If any errors occur during the above sequence, the connection shall be terminated. The connection may be terminated by either device or client side. The device shall not enable a streaming advertisement and scan response payload as long as it has an active streaming connection.

When an active streaming connection is terminated, the device shall force the Streaming Active characteristic value to 0.

100 If a client requests a write of value 1 to the Streaming Active characteristic when the value is already set to 1, the client connection shall be immediately terminated by the device.

5.3 Simplified

6 Interactions

This section applies to all client/device interactions.

105 **6.1 Scanning and Advertising**

For all device advertising, the recommended interval is 20 milliseconds. The maximum advertising interval is 152.5 milliseconds.

110 There are no specific requirements on client scanning. The recommended scanning interval is 100 milliseconds maximum. The recommended scanning window is 20 milliseconds. Passive and active scanning may be utilized.

The values in this section are “nominal”. Because scanning and advertising are asynchronous activities, many BLE protocol stack implementations will defer and/or skip scanning and advertising activities if a higher priority activity conflicts and preempts a scanning or advertising activity.

All scanning and advertising shall be at 1M PHY rates.

115 | 6.2 Connections

| <u>Parameter</u> | <u>Minimum</u> | <u>Recommended</u> | <u>Maximum</u> |
|--------------------------------|----------------|--------------------|----------------|
| <u>Interval (milliseconds)</u> | <u>30</u> | <u>30 to 60</u> | <u>500</u> |
| <u>Latency (intervals)</u> | <u>0</u> | | <u>30</u> |
| <u>Timeout (seconds)</u> | <u>6</u> | <u>6</u> | <u>10</u> |

The device requests the connection parameters. The client makes the final choice of connection parameters. The device connection parameter request must also meet the following criteria:

1. Interval minimum shall be a multiple of 30 milliseconds.
2. Interval maximum shall be at least 30 milliseconds greater than interval minimum.
3. Interval maximum * (Peripheral Latency + 1) <= 6 seconds
4. Timeout shall be greater than interval maximum * (Peripheral Latency + 1) * 3

The minimum MTU size shall be 27. It is recommended that the client request the maximum supported MTU size supported by both client and device, up to the max of 251 supported by the BLE Data Packet Length Extension.

6.3 Privacy

The device should be able to resolve a Resolvable Private Address in all situations. The device shall use a Random Device Address as defined in the Bluetooth 4.0 specification Volume 3, Part C Section 10.8.

130 | 6.4 Permissions

The accessory should not require special permissions, such as pairing, authentication, or encryption to discover services and characteristics. It may require special permissions only for access to a characteristic value or a descriptor value. See the Bluetooth 4.0 specification, Volume 3, Part G, Section 8.1, fifth paragraph.

135 | 6.5 Pairing

The accessory should not request pairing until an ATT request is rejected using the Insufficient Authentication error code. See the Bluetooth 4.0 specification, Volume 3, Part F, Section 4 for details.

If, for security reasons, the accessory requires a bonded relationship with the Central, the Peripheral should reject the ATT request using the Insufficient Authentication error code, as appropriate. As a result, the device may proceed with the necessary security procedures.

Similarly, if the device acts as a Central and a GATT server, it may reject an ATT request using the Insufficient Authentication error code. The accessory should initiate the security procedure for pairing in response.

145

Pairing may require user authorization depending on device. Once an accessory is paired with a device, the accessory shall retain the distributed keys of both central and peripheral for future use. If the pairing is no longer required, the accessory shall delete both sets of keys.

6.6 PHY Modes

150

All clients and devices shall support the 1M PHY. If a client and device both support 2M, 500K, and/or 125K PHY modes, the PHY may be dynamically upgraded or down graded upon request by the client in order to maximize throughput or maximize range. The PHY selection criteria may be based on Receive Signal Strength Initiator (RSSI) of the link, however other selection criteria may also be used. The device may also initiate a request for a PHY mode change, but the client is not required to honor the request.

Table of Contents

1 Introduction (Informative)..... 1

2 Intended Use (Informative)..... 1

3 References and Context..... 1

4 Message Formats..... 1

 4.1 BLE UUIDs..... 2

 4.2 Streaming..... 3

 4.3 Simplified..... 6

5 States..... 10

 5.1 General..... 10

 5.2 Streaming..... 10

 5.3 Simplified..... 11

6 Interactions..... 11

 6.1 Scanning and Advertising..... 11

 6.2 Connections..... 12

 6.3 Privacy..... 12

 6.4 Permissions..... 12

 6.5 Pairing..... 12

 6.6 PHY Modes..... 13