OpenLCB Standard	
OpenLCB-TCP Segment Transfer	
Jan 27, 2013	Preliminary

1 Introduction (Informative)

The OpenLCB suite of protocols can be used on multiple physical transports; this document describes the specifics that relate to using a CAN bus segment as a transport layer. The CAN protocol is frame based, so OpenLCB messages on a CAN bus segment are expressed as frames. This specification defines unique headers to prevent CAN arbitration errors and frame loss, and to provide message traceability, node addressing and priority management.

2 Intended Use (Informative)

This Standard is intended for use whenever OpenLCB nodes are communicating over a reliable byte-stream link, e.g. a TCP/IP connection. It is not intended to cover OpenLCB communications over other types of communications links.

3 References and Context (Normative)

In this document, the term "OpenLCB-TCP" refers to details of OpenLCB nodes that are specific to using a TCP transport, as opposed to generic OpenLCB protocol or OpenLCB using other transports.

This specification should be interpreted in the the context of the following OpenLCB Standards:

- The OpenLCB Unique Identifiers Standard, which specifies the mechanism(s) for providing a unique identifier for each node

Each OpenLCB node (independent of transport layer in use) shall have a unique identifier. When using a CAN bus transport, that identifier shall be used as its node identifier (NodeID).

For more information on format and presentation, see:

- OpenLCB Common Information Technical Note

4 Message Format (Normative)

Individual messages are preceded by a preamble that contains:

- Flags - 16 bit quantity.
  - ○ The most significant bit indicates whether the remaining message is a common OpenLCB message (1) or a link control message (0).

  - ▪ The 2nd most significant bit is a chaining bit, indicating that another header is contained within this one.
  - ▪ The next 2 bits, the rest of the upper nibble of the flags, are reserved. They should be sent as zero and checked as zero on receipt.
  - ▪ The lower 12 bits are reserved. They are to be sent as zero on and ignored upon receipt.
- Number of bytes in rest of message, a 24-bit quantity. This includes both the rest of the prefix and the actual message content.
- Originating node/gateway Node ID – the Node ID of the node or gateway that sent/forwarded this message onto the link. May or may not be the original source of the message.
- Message capture time at originating gateway - a 48-bit number of monotonic milliseconds since the connection was created or some arbitrary time before; no time synchronization with other nodes implied; no precision required, only monotonicity; could have been a sequence number

The message itself is sent in the order:

- MTI – 16 bits total.
- Source Node ID – the full node ID of the node originating the message
- Destination Node ID (if indicated by the MTI) – the full node ID of the destination node
- Event ID (if indicated by the MTI) – an Event ID
- Other content – in byte order

5 States

6 Interactions

A node transmits a message to another node across a TCP link by first sending a prefix, then the message.

If the node is forwarding a message that was received across another TCP link, it includes all prefixes received with the message. The transmission is then the newest prefix, all prefix prefixes in reverse time order, and then the OpenLCB message.

7 TCP-specific Link Control Transmissions and Interactions (Normative)

7.1 Format

7.2 States

7.3 Interactions

Table of Contents

1 Introduction (Informative)........ 1
2 Intended Use (Informative)........ 1
3 References and Context (Normative)........ 1
4 Message Format (Normative)........ 1
5 States........ 2
6 Interactions........ 2
7 TCP-specific Link Control Transmissions and Interactions (Normative)........ 3
7.1 Format........ 3
7.2 States........ 3
7.3 Interactions........ 3