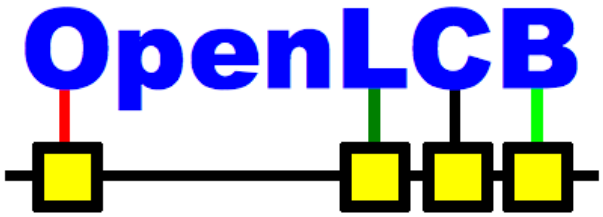

OpenLCB Standard	
OpenLCB-CAN Frame Transfer	
Sep 15, 2012	Draft

1 Introduction (Informative)

This specification describes the mechanism for sending OpenLCB-CAN messages via frames on a CAN segment. It ensures unique headers to prevent CAN arbitration errors and frame loss, message traceability, node addressing and priority management.

2 Intended Use (Informative)

This Standard is intended for use whenever OpenLCB nodes are communicating on a single CAN segment. It is not intended to cover OpenLCB communications over other types of communications links.

3 References and Context (Normative)

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

- The OpenLCB-CAN Physical Layer Standard, which specifies the physical layer for transporting OpenLCB-CAN frames
- The OpenLCB Unique Identifiers Standard, which specifies the mechanism(s) for providing a unique identifier for each node

"CAN" refers to the electrical and protocol specifications as defined in ISO 11898-1:2003 and ISO 11898-2:2003 and their successors.

External certification of parts shall be accepted for conformance to these standards. Conformance with a later version of a standard shall be accepted as conformance with the referenced versions.

Each OpenLCB-CAN node shall have a unique identifier which it shall use as its node identifier (Node ID).

4 Frame Format (Normative)

OpenLCB-CAN frames shall be sent and received using the CAN extended format (29-bit header) only.

OpenLCB-CAN nodes shall operate properly when the CAN segment carries proper standard-format (11-bit header) frames.

OpenLCB-CAN nodes shall not transmit extended-format remote frames (frames with RTR set). Nodes shall operate properly when the CAN segment carries extended-format remote frames.

Nodes shall operate properly when the CAN segment carries overload frames.

The most-significant bit of each OpenLCB-CAN frame is reserved for future use. It shall be transmitted as a 1 bit, and ignored upon receipt.

The second-most-significant bit is the Frame Type indicator. A value of 0 indicates a CAN-specific Control Frame. A value of 1 indicates an OpenLCB Message.

The next 15 bits are the Variable Field. The format and contents of the Variable Field depends on Frame Type and are defined in later sections.

The least significant twelve bits are the Source Node ID Alias value of the source (sending) node.

Bit number[1]:	Bit 28	Bit 27	Bits 26-12	Bits 11-0
Content:	Reserved: Send as 1, ignore upon receipt	Frame Type 1: OpenLCB Message 0: CAN Control Frame	Variable Field	Source NID Alias
Mask:	0x1000,0000	0x0800,0000	0x07FF,F000	0x0000,0FFF
Location:	Solo top bit	Top bit of 6th nibble from right	3 bits, then three nibbles	Right-most three nibbles

Table 1: Frame Format

After the header, the frame shall contain from zero to eight bytes of data. Length and content are defined by specific frame and message definitions elsewhere.

5 States

The frame transfer layer of a node has two states:

- Inhibited
- Permitted

Nodes shall start in the Inhibited state.

A node in the Inhibited state may transmit Check ID, Reserve ID, and Alias Map Definition frames. A node in the Inhibited state shall not transmit any other frame type.

Nodes in Permitted state may transmit any frame type.

[1]See the Common Information OpenLCB Technical Note for detailed conventions on bit and byte numbering. Briefly, the least significant bit of a field is numbered with zero in OpenLCB descriptions, but note that other technologies may use other conventions.

6 CAN-specific Control Frames and Interactions (Normative)

OpenLCB CAN control frames shall be carried in frames with a 0 in the Frame Type field.

6.1 Control Frame Format

The format and contents of CAN-specific Control frames are defined in the following table:

Name	Variable Field	Data Bytes
Check ID (CID) frame	0bMMM,NNNN,NNNN,NNNN MMM is the frame sequence number, with valid values from 0x7 through 0x4 or, for non-OpenLCB protocols, down to 0x1. NNNN,NNNN,NNNN is the 12-bit Node ID section being checked	None
Reserve ID (RID) frame	0x0700	None
Alias Map Definition (AMD) frame	0x0701	Full Node ID
Alias Mapping Enquiry (AME) frame	0x0702	Full Node ID
Alias Map Reset (AMR) frame	0x0703	Full Node ID
Reserved; shall not be sent, and shall be ignored upon receipt	All others	To be defined

Table 2: Control Frame Format

6.2 Interactions

This section describes the interactions which use the above frames.

6.2.1 Reserving a Node ID Alias

To reserve a Node ID alias while in the Inhibited state, a node shall:

- Generate a tentative source Node ID alias value
- Transmit a Check ID frame (CID) with MMM = 0x7, the least significant 12 bits of the full Node ID in the NNNN, NNNN, NNNN remaining twelve bits of the Variable Field, and the tentative source Node ID alias value in the Source NID Alias field.
- Repeat that three more times with MMM = 0x6, x5 and 0x4, respectively, with each frame carrying the next higher 12 bits of the full Node ID value, and the frames carrying the same tentative source Node ID alias value in the Source NID Alias field. (Protocols with a Unique ID

length different from 6 bytes will do this as many times as necessary to carry the entire Unique ID)

- Wait at least 200 milliseconds
- Transmit a Reserve ID frame (RID) with the tentative source Node ID alias value in the Source NID Alias field.

The alias is reserved when that sequence completes without error.

The node shall restart the process at the top if, before completion of the process, a frame is received that carries the source Node ID alias value being testing in its source Node ID alias field.

The node shall restart the process at the top if, before completion of the process, any error is encountered during frame transmission.

6.2.2 Transition to Permitted State

To transition from the Inhibited state to the Permitted state, a node shall, in order:

- Have or obtain a valid reserved Node ID alias
- Transmit an Alias Map Definition (AMD) frame with the node's reserved Node ID alias and Node ID

6.2.3 Node ID Alias validation

A node in Permitted state receiving a Alias Mapping Enquiry frame shall compare the full Node ID in the CAN data segment to the node's own Node ID. If and only if they match in length and content and the receiving node is in Permitted state, the node shall reply with a Alias Map Definition frame carrying the node's full Node ID in the data segment of the frame.

A node in Permitted state receiving an Alias Mapping Enquiry frame with no data content shall reply with an Alias Map Definition frame carrying the node's full Node ID in the data segment of the frame.

A node in Inhibited state shall not reply to a Alias Mapping Enquiry frame.

6.2.4 Transition to Inhibited State

To transition from the Permitted state to the Inhibited state, a node shall successfully transmit an Alias Map Reset frame with the node's reserved Node ID alias and Node ID.

A node shall end use of an alias within 100 milliseconds of receiving an Alias Map Reset (AMR) frame referencing that alias.

6.2.5 Node ID Alias Collision Handling

A node shall compare the source Node ID alias in each received frame against all reserved Node ID aliases it currently holds. In case of a match, the receiving node shall:

- If the frame is a Check ID (CID) frame, send a Reserve ID (RID) frame in response.

- If the frame is not a Check ID (CID) frame, the node is in Permitted state, and the received source Node ID alias is the current Node ID alias of this node, this node shall immediately transition to Inhibited state and release the current Node ID alias.
- If the frame is not a Check ID (CID) frame and the node is not in Permitted state, the node shall immediately release the matching node ID alias.
- If the frame is not a Check ID (CID) frame and the received source Node ID alias is not the current Node ID alias of this node, the node shall immediately release the matching node ID alias.

6.2.6 Duplicate Node ID Handling

Each node shall compare the node ID in each received Alias Map Definition frame with its own Node ID. Should they match, in addition to any other actions that may be required by the incoming message, the node

- may, but is not required to, signal the user that duplicate Node ID values exist using a lamp or other directly-visible indicator
- if in Permitted state, may, but is not required to, emit a Producer-Consumer Event Report (PCER) message with the reserved Event ID “Duplicate Node ID Detected”. If that message is emitted, the node is then required to go offline until reset by the user.

6.3 Node ID Alias Generation

Alias values shall not be zero. Nodes shall not depend on other nodes properly handling zero values in the source and/or destination alias fields.

The first alias values generated by nodes of the same type with node ID values within 255 of each other shall not be identical.

An alias generation algorithm shall ensure that when two different nodes using that alias generation algorithm generate the same alias value at two different points in their sequence, there shall be more than a 99% probability that the next alias values generated by the two nodes are different.

A node may, but need not, save the current alias generation state so that it restarts the sequence at the same point, hence the same alias value, after a reset or power cycle.

Table of Contents

