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 message loss, message traceability, node addressing and priority management.

5 For information on format and presentation conventions, see:

- OpenLCB Common Information Technical Note

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
10 communications links.

3 References and Context (Normative)

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

- The OpenLCB-CAN Physical Layer specification, which specifies the physical layer for transporting OpenLCB-CAN frames

15 - The OpenLCB Node Identifier Specification, which specifies that every OpenLCB node has a unique 6-byte identifier

"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.
20 Conformance with a later version of a standard shall be accepted as conformance with the referenced versions.

4 Frame Format (Normative)

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

25 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 proper extended-format remote frames.

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

30   The first (most-significant) bit is reserved for future use. It must be transmitted as a 1 bit, and ignored upon receipt.

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

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

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

Bit number:	Bit 0	Bit 1	Bits 2-16	Bits 17-28
Content:	Reserved: Send as 1, ignore upon receipt	Frame Type: 1: OpenLCB Message 0: CAN Control Message	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

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

5 States

The frame transfer layer of a node has two states:

- Inhibited

- Permitted

45   Nodes shall start in the Inhibited state.

A node in the Inhibited state may transmit Check ID Message, Reserved ID Message, 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.

6 CAN-specific Control Messages and Interactions (Normative)

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

6.1 Control Message Format

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

Name	Variable Field	Data Bytes
Check ID (CIM) Message	MMM,NNNN,NNNN,NNNN MMM is the message sequence number, with valid values from 0x7 through 0x4 NNNN,NNNN,NNNN is the 12-bit Node ID section being checked	None
Reserved ID (RIM) Message	0x0700	None
Alias Map Definition (AMD) Message	0x0701	Full Node ID
Alias Mapping Enquiry (AME) Message	0x0702	Full Node ID
Alias Map Reset (AMR) Message	0x0703	None
Reserved; may not be sent, and must be ignored upon receipt	All others	To be defined

6.2 Interactions

This section describes the interactions which use the above messages.

6.2.1 Reserving a Node ID Alias

55

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

- Generate a new source Node ID alias value

- Transmit a Check ID Message (CIM) 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
60   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 message carrying the next lower 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.

- Wait at least 50 milliseconds

65
- Transmit a Reserve ID Message (RIM) with the tentative source Node ID alias value n the Source NID Alias field.

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
70 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 a Alias Map Definition (AMD) message with the node's reserved Node ID alias and
75 Node ID

6.2.3 Node ID Alias validation

A node in Permitted state receiving a Alias Mapping Enquiry Message 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 Message
80 carrying the node's full Node ID in the data segment of the frame.

A node in Permitted state receiving a Alias Mapping Enquiry Message with no data content shall reply with a Alias Map Definition Message 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 Message.

6.2.4 Transition to Inhibited State

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

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 consider the Node ID alias
90 released and no longer reserved. If that alias was used to transition to Permitted state, the node shall transition to Inhibited state via transmission of an Alias Map Reset Message.

6.3 Node ID Alias Generation

Alias values shall not be zero.

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

When two different nodes generate the same alias value at some point 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.

100 # 7 OpenLCB Message Frame Format

OpenLCB common messages shall be carried in frames with a 1 in the Frame Type field. They shall contain message type information and/or address information in the 15-bit variable field, and zero to eight CAN data bytes.

For OpenLCB messages, the variable field shall be used in one of two forms:

105
- Unaddressed messages – messages that don't have a destination address put the low 12 bits of the MTI in the variable field – shall be formatted:

Variable Field Bit 0 Header Bit 2 0x0400,0000	Variable Field Bits 1-14 Header Bits 3-16 OpenLCB Variable Header Content 0x03FF,F000
0	OpenLCB message information

- Addressed messages – messages that have a specific destination address – shall have the address alias in the low 12 bits of the variable field. Two upper bits can carry part of the OpenLCB

110 message.

Variable Field Bit 0 Header Bit 2 0x0400,0000	Variable Field Bits 1-2 Header Bits 3-4 OpenLCB Variable Header Content 0x0300,0000	Variable Field Bits 3-14 Header Bits 5-16 OpenLCB Variable Header Content 0x00FF,F000
1	OpenLCB message information	Destination Node ID Alias

Table of Contents

115