

This document is under formal review of the NMRANET Working Group. It has not been approved by the WG, reviewed by the Manager, S&C Dept., nor approved by the NMRA Board of Directors.

NMRA STANDARD	
S 9.6.2C	
CAN Frame Transfer	
All Scales	
(date)	(number)

1 Introduction (Informative)

This specification describes the mechanism for sending OpenLCB-CAN information via frames on a CAN segment.

2 References and Context (Normative)

- 5 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
 - The OpenLCB Node Identifier Specification, which specifies that every OpenLCB node has a unique 6-byte identifier

- 10 “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.

15 3 Frame Format (Normative)

OpenLCB-CAN frames are 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.

- 20 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.

OpenLCB-CAN nodes shall not transmit overload frames. Nodes shall operate properly when the CAN segment carries proper overload frames.

- 25 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.

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

The last twelve bits (least significant) are the Source Node ID Alias value for this node, see below.

Bit 0	Bits 1	Bits 2-16	Bits 17-28
Reserved: Send as 1, ignore upon receipt	Frame Type: 1: OpenLCB Message 0: CAN Control Message	Variable Field	Source NID Alias
0x1000,0000	0x0800,0000	0x07FF,F000	0x0000,0FFF
Solo top bit	Top bit of 6 th nibble from right	3 bits, then three nibbles	Right-most three nibbles

- 35 After the header, the frame may contain from zero to eight bytes of data. Length and content are defined with specific message definitions.

3.1 States

A node has two states:

- Inhibited
- 40 • Permitted

Nodes shall start in the Inhibited state.

A node may transmit Check ID Message and Reserved ID Message frames while in the Inhibited state. A node shall not transmit any other frame while in Inhibited state.

Nodes shall transition to the Permitted state after assignment of a valid Node ID alias.

- 45 Nodes may transmit any frames except Check ID Message and Reserved ID Message frames while in Permitted state. A node shall not transmit Check ID Message and Reserved ID Message frames while in Permitted state.

4 CAN-specific Control Messages and Interactions (Normative)

4.1 Control Message Format

50 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 0x4 through 0x7 NNNN,NNNN,NNNN is the 12-bit Node ID section being checked	None
Reserved ID (RIM) Message	0x0700	None
Mapping Reset (MR) Message	0x0701	None
Mapping Enquiry Request (MRQ) Message	0x0702	Full Node ID
Mapping Enquiry Reply (MRR) Message	0x0703	Full Node ID
Reserved; may not be sent, and must be ignored upon receipt	All others	To be defined

4.2 Interactions

This section describes the interactions which use the above messages.

4.2.1 ID Alias assignment

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

55 Calculates a new tentative source Node ID alias value

Transmits 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 tentative source Node ID alias value in the Source NID Alias field.

60 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.

(What to do when something goes wrong)

- If error, step the algorithm in a standard way and repeat until success.

65 Aliases for nodes on the local segment are only valid until an "Initialization Complete" is seen from that alias. Initialization Complete indicates that a node has restarted, and may have another local alias. *(Can't use Initialization Complete, it's the wrong layer, need something at the CAN level. Need something like RIM/CIM)*

70 4.2.2 ID Alias validation

A node receiving a Mapping Enquiry Request 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 Mapping Enquiry Reply Message carrying the node's full Node ID in the data segment of the frame.

75 A node receiving a Mapping Enquiry Request Message with no data content shall reply with a Mapping Enquiry Reply Message carrying the node's full Node ID in the data segment of the frame.

4.2.3 ID Alias termination

A node receiving a Mapping Reset Message must drop the

80 5 OpenLCB Message Format

OpenLCB common messages are carried in frames with a 1 in the Frame Type field. They 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 is used in two forms:

- 85
- Unaddressed messages – messages that don't have a destination address put the low 12 bits of the MTI in the variable field

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

- Addressed messages – messages that have a specific destination address put it in the variable field, and carry the MTI in the payload. This allows filtering.

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

90

5.1 Node ID Alias Generation

Aliases must be created from the node ID (give separate starting points)

95 May, but need not, be saved to be the starting point for the next time

When a second is needed, it needs to be different.

Aliases must not be zero

100 (algorithm is recommended in tech note, but not required; see also discussion of what makes a good algorithm)

Table of Contents

1 Introduction (Informative)..... 1

2 References and Context (Normative)..... 1

3 Frame Format (Normative)..... 1

 3.1 States..... 2

4 CAN-specific Control Messages and Interactions (Normative)..... 2

 4.1 Control Message Format..... 2

 4.2 Interactions..... 3

 4.2.1 ID Alias assignment..... 3

 4.2.2 ID Alias validation..... 4

 4.2.3 ID Alias termination..... 4

5 OpenLCB Message Format.....	4
5.1 Node ID Alias Generation.....	6

DRAFT