NMRA Standard
Layout Command Control® (LCC) **Memory Configuration**

Feb 17, 2015	9.7.4.2-S

Adopted as a NMRA Standard

The OpenLCB Standard document appended to this cover sheet has been formally adopted as a NMRA Standard by the NMRA Board of Directors on the date shown in the *Adopted* column in the *Version History* table below.

5 ## Version History

Date	Adopted	Summary of Changes
Feb 17, 2015		Initial version

OpenLCB Standard	
Memory Configuration	
Feb 17, 2015	**Adopted**

1 Introduction (Informative)

This Standard defines a protocol for configuring OpenLCB nodes by directly accessing their configuration memory.

2 Intended Use (Informative)

5   Intended to be used to configure self-contained nodes over their OpenLCB links.

3 References and Context (Normative)

This Standard is in the context of the following OpenLCB Standards:

- OpenLCB Configuration Description Information (CDI), which defines fixed information to allow a Configuration Tool (CT) to configure a node.

10
- OpenLCB Datagram Transport Protocol, which defines the protocol for transporting 0-72 byte datagrams from node to node.

- OpenLCB Message Network, which defines message interchange for OpenLCB, specifically how this protocol appears in the Protocol Support Reply message and Error Codes.

15
- The OpenLCB Stream Transport Standard, which defines the protocol for transporting data from node to node.

4 Message Formats (Normative)

The tables below show available configuration message formats. All other formats are reserved. Note that Byte 6 is sometimes specified as optional. When Byte 6 is specified as optional, and
20   not used, it is not present in the message, and the remaining bytes directly follow Byte 5 with no Byte 6 padding in between.

4.1 Address Space Size

Configuration memory addresses are 32 bits. The addressable quantity is the byte. There are 256 unique address spaces, which are specified using an address space selectors. Address spaces may
25   be wholly designated as read-only or write-only by the implementing node. An address space may not be partially designated as read-only or write-only.

4.2 Address Space Selection

Of the 256 unique address spaces, those that follow are required to be implemented by any node that implements this Standard (these may or may not have content on a particular node); the following
30   address space numbers can only be used for the specified purpose, and if the corresponding information is available, it must be accessible by these address space selectors (in addition to any others the designer might provide):

- (space=0xFF, control bits=0b0000011) Configuration definition – A read-only space containing the Configuration Description Information for this node, if any.

35
- (space=0xFE, control bits=0b00000010) All memory – This special space provides access to all memory in the device, where "all" is defined by the designer.

- (space=0xFD, control bits=0b00000001) Configuration – A writable basic configuration space, with the structure of the 32-bit space defined by the designer.

These three spaces, inclusive, can (but do not have to) be addressed without an extra byte in the
40   datagram using control bits in the flag byte (byte 1). All others must be specified as a byte value in byte 6 of the datagram payload (see Section 4.3).

4.3 Generic Error Handling

The Error Codes returned by failed operations, either in the Datagram Rejected message, or the Failed response datagrams, shall conform to the general Error Code specification as defined by the OpenLCB
45   Message Network Standard.

An unknown command in byte 1 shall return a datagram rejected reply. Nodes may, but are not required to use the error code "Not implemented, subcommand unknown"..

Additionally, the following protocol specific enumerated error code extensions (logically or'd into bits 0-3 of the base error code as defined in the OpenLCB Message Network Standard) are defined for use
50   with commands that are supported and known to the node:

- "Permanent error, invalid argument" | 0x0001 – Address space is not known.

- "Permanent error, invalid argument" | 0x0002 – Out of bounds, the address space is valid, however the address within the space is not implemented.

- "Permanent error, invalid argument" | 0x0003 – Write access to a read-only address space.

55
4.4 Read Command

Byte 0	Byte 1	Byte 2-5	Byte 6 (optional)	Byte 7/6
0x20	0x40 Space in byte 6 0x41 Space 0xFD 0x42 Space 0xFE 0x43 Space 0xFF	Starting Address	Address Space	Read Count (1 – 64 bytes) Upper bit reserved, must be ignored

In reply to the datagram containing a Read command, the receiving node shall set the Reply Pending bit in the Datagram Received OK message. The receiving node may, but is not required to, include a specific timeout interval in the Datagram Received OK message.

60   If the interval is provided and has elapsed without a Read Reply message being returned, the node requesting the Read operation may, but is not required to, repeat the request.

4.5 Read Reply

Byte 0	Byte 1	Byte 2-5	Byte 6 (optional)	Remaining Bytes (in network order, top to bottom)
0x20	OK status: 0x50 Space in byte 6 0x51 Space 0xFD 0x52 Space 0xFE 0x53 Space 0xFF	Starting Address	Address Space	Data (1 – 64 bytes)
	Failure status: 0x58 Space in byte 6 0x59 Space 0xFD 0x5A Space 0xFE 0x5B Space 0xFF			• Error Code (two bytes) • optional zero-terminated message string (0 – 63 bytes)

In general, a read reply may provide less than the requested data, but always at least one byte if it's a
65   valid read. The maximum read request is 64 bytes when reading via datagrams. When reading via streams, any length up to 0xFF,FF,FF,FF (4GB-1) can be requested; a length of 0x0 means "read forever" or until the end of the address space.

Reading at least one byte, but less than the requested amount, due to the length of the address space is not considered an error.

70   If the full number of bytes cannot be read for any reason other than the size of the address space, or if no bytes can be read, the Read Reply shall have the Fail bit set and include an error code instead of the requested data.

4.6 Read Stream Command

Byte 0	Byte 1	Bytes 2-5	Byte 6 (optional)	Remaining Bytes (in network order, top to bottom)
0x20	0x60 Space in byte 6 0x61 Space 0xFD 0x62 Space 0xFE 0x63 Space 0xFF	Starting Address	Address Space	• Read Count (4 bytes) • Destination Stream ID (1 byte)

75   Very similar to a Read Command, except that a stream is created to carry the read information back to the requesting node. The stream must be established before the Read Reply datagram is sent, so that error codes can be returned via the Read Reply in the stream initialization fails.

The Destination Stream ID provided must be used as the suggested Destination Stream ID in the subsequent OpenLCB Stream Transport protocol Stream Initiate Request.

4.7 Read Stream Reply

Byte 0	Byte 1	Bytes 2-5	Byte 6 (optional)	Remaining Bytes (in network order, top to bottom)
0x20	OK status: 0x70 Space in byte 6 0x71 Space 0xFD 0x72 Space 0xFE 0x73 Space 0xFF	Starting Address	Address Space	• Read Count (4 bytes) • Destination Stream ID (1 byte) • Source Stream ID (1 byte)
	Failure status: 0x78 Space in byte 6 0x79 Space 0xFD 0x7A Space 0xFE 0x7BSpace 0xFF			• Error Code (2 bytes) • optional zero-terminated message string (0 – 63 bytes)

The Destination Stream ID must match the Destination Stream ID used in both the Read Stream Command and the subsequent stream that was created as a result of the Read Stream Command. The Source Stream ID must match the Source Stream ID used in the the subsequent stream that was a result of the Read Stream Command. The Read Stream Reply shall be sent after the stream has been established, but before any data has been sent using the stream. When the stream successfully closes, after having read 1 or more bytes, the transaction is considered complete. If the stream closes with an error, this transaction is considered to be in error.

Should an error occur prior to or during the establishment of the stream, the Read Stream Reply shall indicate a failure with the error code set appropriately.

4.8 Write Command

Byte 0	Byte 1	Byte 2-5	Byte 6 (optional)	Remaining Bytes
0x20	0x00 Space in byte 6 0x01 Space 0xFD 0x02 Space 0xFE 0x03 Space 0xFF	Starting Address	Address Space	Data (1 – 64 bytes)

If the write operation can be done immediately and succeeds, only the Datagram Received OK message reply to the Write Command datagram is returned. The Reply Pending bit is not set in the Datagram Received OK reply.

If the write operation takes time, or fails immediately, the Datagram Received OK message reply to the Write Command datagram shall carry the Reply Pending bit set. The receiving node may, but is not required to, include a specific timeout interval in the Datagram Received OK message. This is followed

100 later by the receiving node sending a Write Reply datagram with either OK or Fail set. If Fail is set, the error code shall be included. The optional message string may be, but is not required to be, included.

If a timeout interval was provided and has elapsed without a Write Reply message being returned, the node requesting the Write operation may, but is not required to, repeat the request.

4.9 Write Reply

Byte 0	Byte 1	Byte 2-5	Byte 6 (optional)	Remaining Bytes (in network order, top to bottom)
0x20	OK status: 0x10 Space in byte 6 0x11 Space 0xFD 0x12 Space 0xFE 0x13 Space 0xFF	Starting Address	Address Space	(None)
	Failure status: 0x18 Space in byte 6 0x19 Space 0xFD 0x1A Space 0xFE 0x1B Space 0xFF			• Error Code (two bytes) • optional zero-terminated message string (0 – 63 bytes)

105 This message is sent in only if the Reply Pending bit in the Datagram OK message reply to a previous Write Command is set. If the write fails, a Failure Status reply is sent with the Error Code set appropriately.

4.10 Write Under Mask Command

Byte 0	Byte 1	Byte 2-5	Byte 6 (optional)	Remainder
0x20	0x08 Space in byte 6 0x09 Space 0xFD 0x0A Space 0xFE 0x0B Space 0xFF	Starting Address	Address Space	Data pairs (2-64 bytes)

The data pairs are (Mask, Data) where each of Mask and Data are single bytes. Bits in Data where the
110 corresponding bit in Mask is a 1 will be written to the memory location. Bits in memory where the corresponding bit in Mask is a 0 will be left unchanged. The first data pair refers to memory at the Starting Address. Each following data pair refers to a successive byte in memory.

The Write Under Mask Command requires a response consistent with that of a standard Write Command whereby the Reply Pending bit in the Datagram OK reply determines if a subsequent Write
115 Reply will occur.

4.11 Write Stream Command

Byte 0	Byte 1	Byte 2-5	Byte 6 (optional)	Remainder (in network order, top to bottom)
0x20	0x20 Space in byte 6 0x21 Space 0xFD 0x22 Space 0xFE 0x23 Space 0xFF	Starting Address	Address Space	• send as 0xFF, ignore on receipt (1 byte) • Source Stream ID (1 byte)

After sending this command, and receiving Datagram OK, an OpenLCB Stream Transport protocol Stream Initiate Request will be sent. The destination node is not required to accept the subsequent stream, however, once the Write Stream Command has received a Datagram OK reply, the destination node has committed itself to sending a Write Stream Reply with the appropriate success or failure with error code. If the Datagram OK reply has the Reply Pending bit set, it may use any timeout value (if provided) to wait for the Write Stream Reply. Any timeout shall be from the time of the last stream shutdown message and is provided for the receiving node to commit data to memory.

The destination node may reject the operation immediately with a Datagram Rejected response and the error code set appropriately. In this case, no stream will be established, and a Write Stream Reply is not required or expected.

4.12 Write Stream Reply

Byte 0	Byte 1	Byte 2-5	Byte 6 (optional)	Remaining Bytes (in network order, top to bottom)
0x20	OK status: 0x30 Space in byte 6 0x31 Space 0xFD 0x32 Space 0xFE 0x33 Space 0xFF	Starting Address	Address Space	• Destination Stream ID (1 byte) • Source Stream ID (1 byte)
	Failure status: 0x38 Space in byte 6 0x39 Space 0xFD 0x3A Space 0xFE 0x3B Space 0xFF			• Error Code (two bytes) • optional zero-terminated message string (0 – 63 bytes)

The Destination Stream ID must match the Destination Stream ID used in both the Read Stream Command and the subsequent stream that was created as a result of the Read Stream Command. The Source Stream ID must match the Source Stream ID used in the the subsequent stream that was a result of the Read Stream Command. The Read Stream Reply shall be sent after the stream has been established, but before any data has been sent using the stream. When the stream successfully closes, after having read 1 or more bytes, the transaction is considered complete. If the stream closes with an error, this transaction is considered to be in error.

Should an error occur prior to or during the establishment of the stream, the Read Stream Reply shall indicate a failure with the error code set appropriately.

4.13 Get Configuration Options Command

Byte 0	Byte 1
0x20	0x80

140

4.14 Get Configuration Options Reply

Byte 0	Byte 1	Byte 2-3	Byte 4	Byte 5	Byte 6 (Optional)	Remainder (Optional)
0x20	0x82	Available Commands	Write Lengths	Highest Address Space	Lowest Address Space	Name String

- Available operations mask (2 bytes, bit coded): Indicate which operations are available so the using software can know whether convenience operations (which are not possible on some hardware) are available.

145

- 0x8000 Write under mask supported

- 0x4000 Unaligned reads supported. If not set, reads have to start on an address with the low bits, as given by the data size, all zero. For example a 4-byte write must have the low two address bits zero.

150
- 0x2000 Unaligned writes supported. If not set, reads have to start on an address with the low bits, as given by the data size, all zero. For example a 4-byte write must have the low two address bits zero.

- 0x0800 Read from address space 0xFC available (this is the manufacturer part of Abbreviated CDI)

155
- 0x0400 Read from address space 0xFB available (this is the user-entered part of Abbreviated CDI)

- 0x0200 Write to address space 0xFB available (this is the user-entered part of Abbreviated CDI)

- Others reserved, must be ignored on receipt and sent as zero.

160
- Write lengths supported (One byte, bit coded): (provided for devices that can only write certain sizes to memory) (at least one bit must be set)

- 0x80 1 byte write

- 0x40 2 byte write

- 0x20 4 byte writes

165
- 0x10 64 byte writes (full datagram, but not 63 bytes or arbitrary length, just exactly 64)

- 0x02 arbitrary writes of any length OK

- 0x01 stream writes supported (stream support will identify buffer size)

- Others reserved, must be ignored on receipt and sent as zero.

- Highest Address Space (byte): Highest number space available. Not all up to that need be
170   available, but sparse allocation will slow down the process as "Get Address Space Information"
       is needed to determine whether they are present.

- Lowest Address spaces (byte): Lowest number space available. Note that spaces 0xFD, 0xFE
  and 0xFF are assumed to be included even if the low space ↔ high space range doesn't include
  them. (also 0xFC, 0xFB of Abbreviated Default CDI if bits indicate they're available)

175 A node that only has the high spaces could have Highest Address Space = 255, Lowest Address Space
     = 253 or 251.

A node that has additional low address spaces, e.g. to make more memory available with a 39-bit
address, could have Highest Address Space = 127, Lowest Address Space = 0 and leave the top spaces
assumed.

180 ## *4.15 Get Address Space Information Command*

Byte 0	Byte 1	Byte 2
0x20	0x84	Address Space

4.16 Get Address Space Information Reply

Byte 0	Byte 1	Byte 2	Byte 3-6	Byte 7	Byte 8-11	Remaining
0x20	0x86 0x87	Address Space	Highest Address	Flags	Lowest Address	Description

- Command (Byte 1)

185
- Byte 1 – 0x87: present, else if 0x86: not present

- Address Space (Byte 2) – provided to identify request this reply is in response to

- Highest Address (Bytes 3 - 6)

- Flags (Byte 7)

- Bits 7..2 – Reserved

190
- Bit 1 – 0b0: low address is zero, else if 0b1: low address specified in next four bytes

- Bit 0 – 0b0: writable, else if 0b1: read-only

- Lowest Address (Bytes 8 - 11) – optional, omit if zero, as that will let reply fit in single CAN frame; if present, "non-zero lowest address" bit in prior byte must be 1.

- Description (Bytes 12+, variable length) – optional null-terminated string giving the user-readable name of this space

195

4.17 Lock/Reserve Command

Byte 0	Byte 1	Byte 2-7
0x20	0x88	Reserving Node ID

At the start of configuration, a configuring node sends a Lock message with its NodeID. If no node has locked this node, indicated by zero content in the lock memory, the incoming NodeID is placed in the 200 lock memory. If a node has locked this node, the non-zero NodeID in the lock memory is not changed. In either case, the content of the lock memory is returned in the reply. This acts as a test and set operation, and informs the requesting node whether it successfully reserved the node. To release the node, repeat the lock operation with a zero Node ID. The lock memory is set to zero when the node is reset. Note that this is a voluntary protocol in the configuring nodes only; the node being configured 205 does not change it's response to configuration operations when locked or unlocked.

4.18 Lock/Reserve Reply

Byte 0	Byte 1	Byte 2-7
0x20	0x8A	Reserved Node ID

4.19 Get Unique ID Command

Byte 0	Byte 1	Byte 2
0x20	0x8C	Number requested (1-8, 3 bits); upper bits reserved and must be ignored

210   Nodes maintain a list of unique Event IDs for use in configuration. These are allocated based on the node's unique Node ID. This command allows a configuration tool to get new unique Event IDs from the node's pool, for example to interact with the Blue/Gold configuration process. Each request must provide a different Event ID, without repeat, even through node resets and factory resets.

4.20 Get Unique ID Reply

Byte 0	Byte 1	Remainder
0x20	0x8D	1-8 requested Event IDs, a total of 8-64 bytes

215

4.21 Update Complete Command

Byte 0	Byte 1
0x20	0xA8

This command notifies the target node that the configuration update operation sequence has completed. Nodes do not have to require this operation, but receiving it must be permitted. Configuration tools
220 should send it at the end of operations. Nodes may, but are not required to, reset after sending the reply to this message.

4.22 Reset/Reboot Command

Byte 0	Byte 1
0x20	0xA9

After reset successfully processing this command, the receiving node shall reset/reboot into its power
225 on reset state, as if it has been put through a power cycle.

4.23 Reinitialize/Factory Reset Command

Byte 0	Byte 1	Byte 2-7
0x20	0xAA	Node ID

"Reinitialize/Factory Reset" restores the node's configuration as if factory reset. Node ID shall carry the destination Node ID for redundancy.

230 ### *4.24 Enter Bootloader Command*

Byte 0	Byte 1
0x20	0xAB

This command is reserved for the future specification of a firmware upgrade protocol.

Table of Contents