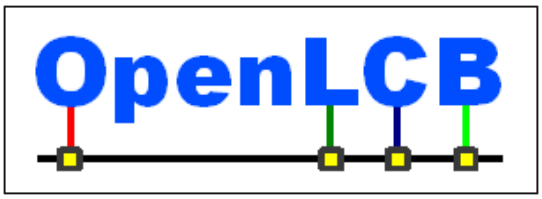

OpenLCB Technical Note	
LocoNet Gateway Protocol	
04/10/2024	Draft

1 Introduction (Informative)

This technical note contains informative discussion and background for the corresponding "OpenLCB LocoNet Gateway Protocol Standard". This explanation is not normative in any way.

The OpenLCB LocoNet Gateway Protocol Standard is intended for use by OpenLCB application nodes that require more extensive interaction with LocoNet than can be provided by a simple message translation LocoNet hardware gateway nodes. In this context the OpenLCB network is a transport mechanism for the LocoNet messages. It allows the host computer of an OpenLCB application node to have a single connection to the OpenLCB network, and the OpenLCB application node to have full access to all the capabilities of LocoNet via one or more LocoNet Gateway nodes connected to the OpenLCB network.

LocoNet is a registered trademark of Digitrax, Inc., Panama City, Florida, USA.

1.1 Served Use Cases

- Virtual LocoNet Command Station

  An OpenLCB application node implements a virtual LocoNet command station. A LocoNet throttle connected to a stand-alone LocoNet segment can control OpenLCB train nodes. The virtual command station translates the commands sent by the throttle into OpenLCB traction control commands, and *vice versa*.

- Virtual OpenLCB Train Nodes

  An OpenLCB application node implements virtual OpenLCB train nodes for trains controlled by a physical LocoNet command station. This allows OpenLCB throttles to control trains on a LocoNet segment as if they were native OpenLCB train nodes. The virtual train node translates the OpenLCB traction and train search commands into LocoNet messages.

- Virtual OpenLCB Nodes for LocoNet devices

  An OpenLCB application node implements virtual OpenLCB nodes that provide wrappers for devices that are connected to a LocoNet segment. The wrapper makes the LocoNet device appear to be a native OpenLCB node. Translating OpenLCB Producer/Consumer Event Report messages into LocoNet messages and *vice versa*. Such nodes may also implement OpenLCB Location Services events for train identification and location detection.

- Configuration Tool

DRAFT

Copyright 2011-2024. All rights reserved. This OpenLCB document is licensed under the Creative Commons Attribution-ShareAlike 4.0 International License (CC BY-SA 4.0). See https://openlcb.org/licensing for more information.

An OpenLCB application node implements virtual OpenLCB nodes that allow an OpenLCB Configuration Tool to configure a device connected to a LocoNet segment in the same way as a native OpenLCB node. The virtual node would implement the necessary CDI and translation of OpenLCB memory configuration commands into the necessary LocoNet messages.

1.2 Un-Served Use Cases

There is anecdotal evidence to suggest that some LocoNet devices also use the LocoNet bus to transport messages that do not conform to the LocoNet message format as defined in the LocoNet Specification. Any such messages are out of scope of this protocol and no provision has been provided to support them.

2 Annotations to the Standard

2.1 Introduction

Note that this section of the Standard is informative, not normative.

2.2 Intended Use

Note that this section of the Standard is informative, not normative.

2.3 References and Context

The LocoNet Specification is the only publicly available document that specifies the LocoNet message format and the details of some basic LocoNet messages. It is known that there are many more LocoNet message types than those detailed in the LocoNet Specification. The OpenLCB LocoNet Gateway Protocol does not require an OpenLCB LocoNet Gateway node to understand the internal format or function of any message that it transports between a LocoNet segment and the OpenLCB network beyond the basic LocoNet message format as defined in the LocoNet Specification.

2.4 Messages

2.4.1 Definitions

A Valid LocoNet Message consists of an opcode byte, zero or more data bytes, followed by a valid checksum byte. The most significant bit (d7) of each byte is the opcode flag. If the opcode flag is 1, then the byte is an opcode byte. The opcode byte is the first byte of a LocoNet message. The opcode flag is 0 for all other bytes in the message including the checksum. The opcode flag may be used to synchronize and detect the start of LocoNet messages. The checksum is calculated as the 1's complement of the byte wise exclusive or of all the bytes in the message except the checksum. The length of the message may be determined by examining bits d6 and d5 of the opcode byte and/or the second byte of the message. See the “Message Format” section of the LocoNet Specification for full details of the LocoNet message format.

2.4.2 Defined Event IDs

After initialization a LocoNet Gateway node should send a Producer Range Identified message with an Event ID Range of 0x01 0x81 0x00 0x00 0x00 0x00 0x00 0x00.

2.4.3 Generic Error Handling

Nothing to add to the Standard.

DRAFT

Copyright 2011-2024. All rights reserved. This OpenLCB document is licensed under the Creative Commons Attribution-ShareAlike 4.0 International License (CC BY-SA 4.0). See https://openlcb.org/licensing for more information.

2.4.4 Transporting Messages from LocoNet to the OpenLCB network

Nodes receiving LocoNet Message Received events may quickly determine if they are interested in them by examining the first two bytes of the space reserved for the Event ID and discard the message if they are not. The OpenLCB LocoNet Gateway Protocol uses the rest of the space in the PCER message reserved for the Event ID to carry either the whole or a portion of a LocoNet message. OpenLCB nodes expect the space reserved for the Event ID to contain 8 bytes. If a LocoNet message is less than 8 bytes long then it must be padded up to the required 8 bytes. The value of 0xFF was chosen as this is an invalid value for a LocoNet message byte in any position other than the first byte of the message. Most common LocoNet messages will fit within the Event ID space and not require the use of PCER with payload.

Encoding Example #1

The LocoNet message 0x83 0x7C is encoded as follows:

Event ID: 0x01 0x81 0x83 0x7C 0xFF 0xFF 0xFF 0xFF

Payload: None

Encoding Example #2

The LocoNet message 0xE7 0x0E 0x7F 0x00 0x0B 0x38 0x42 0x47 0x01 0x00 0x00 0x1C 0x3E 0x7C is encoded as follows:

Event ID: 0x01 0x81 0xE7 0x0E 0x7F 0x00 0x0B 0x38

Payload: 0x42 0x47 0x01 0x00 0x00 0x1C 0x3E 0x7C

If there are multiple LocoNet Gateway nodes attached to the OpenLCB network then an OpenLCB node can determine which LocoNet segment the message originated from by examining the source node ID of the Producer/Consumer Event Report.

2.4.5 Transporting Messages from an OpenLCB node to LocoNet

The LocoNet Gateway node may buffer the LocoNet messages received along with the node ID of the message sender before actual transmission to the LocoNet segment. After the message is actually sent to the LocoNet segment the LocoNet Gateway node will reply to the sender node with either a Send LocoNet Message Reply or a Send LocoNet Message Reply Failure datagram depending upon the success or failure of the transmission.

The LocoNet Standard states that "...that a typical LocoNet implementation allows us to achieve about 98% of network traffic capacity with less than 1% collision". That still allows for collisions, and so it was decided to require collision detection within this protocol. The Standard is silent on how to implement collision detection, but as all messages sent to the LocoNet bus by a device are also received by the device from the bus, one method of determining if a message was successfully transmitted to the LocoNet segment is to compare the messages received after a message is sent, within a timeout, with the message sent and check if one of messages received matches. If none match then the transmission failed. If no message is received within the timeout period then the transmission can also be determined to have failed.

2.4.6 Send LocoNet Message Complete in Datagram Command

The majority of the LocoNet messages are short and will fit within one datagram.

Copyright 2011-2024. All rights reserved. This OpenLCB document is licensed under the Creative Commons Attribution-ShareAlike 4.0 International License (CC BY-SA 4.0). See https://openlcb.org/licensing for more information.

2.4.7 Send LocoNet Message First Part Command

A Valid LocoNet Message could be up to 127 bytes long. It has been reported that some LocoNet devices do indeed use the full 127 bytes for data transfer purposes.

2.4.8 Send LocoNet Message Final Part Command

See Section 2.4.7.

2.4.9 Send LocoNet Message Reply

Nothing to add to the Standard.

2.4.10 Send LocoNet Message Reply Failure

Nothing to add to the Standard.

2.5 LocoNet Gateway Identification

Nothing to add to the Standard.

DRAFT

Copyright 2011-2024. All rights reserved. This OpenLCB document is licensed under the Creative Commons Attribution-ShareAlike 4.0 International License (CC BY-SA 4.0). See https://openlcb.org/licensing for more information.

Table of Contents

DRAFT

Copyright 2011-2024. All rights reserved. This OpenLCB document is licensed under the Creative Commons Attribution-ShareAlike 4.0 International License (CC BY-SA 4.0). See https://openlcb.org/licensing for more information.