

Installation Guide

Revision-a.3

December 2025

LCC-LoCoNet Gateway

***LCC (Layout Command and Control)
1024 function LCC - LoCoNet translator***

This PDF is designed to be read on screen, two pages at a time. If you want to print a copy, your PDF viewer should have an option for printing two pages on one sheet of paper, but you may need to start with page 2 to get it to print facing pages correctly. (Print this cover page separately.)

Copyright

This document is Copyright © December 2025 by **RR-CirKits, Inc.** You may distribute it under the terms of either the GNU General Public License, version 3 or later (<http://www.gnu.org/licenses/gpl.html>), or the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/3.0/>), version 3.0 or later.

All trademarks within this guide belong to their legitimate owners.

Authors

Dick Bronson

Caleb Davis

Feedback

Please direct any comments or suggestions about this document to:

dick@rr-cirkits.com

caleb@rr-cirkits.com

Contact Information

RR-CirKits, Inc.
7918 Royal Ct.
Waxhaw, NC USA 28173

<https://www.rr-cirkits.com>
sales@rr-cirkits.com
service@rr-cirkits.com
1-704-843-3769
Fax: 1-704-243-4310

Publication date and firmware version

Published December 2025.

Firmware Rev B9b

WARNING:

This product contains a chemical known to the state of California to cause cancer, birth defects or other reproductive harm. Do not ingest.

*You can download an editable version of this document from
<http://www.rr-cirkits.com/manuals/LCC-Loconet-Gateway-manual-a3.odt>*

Contents

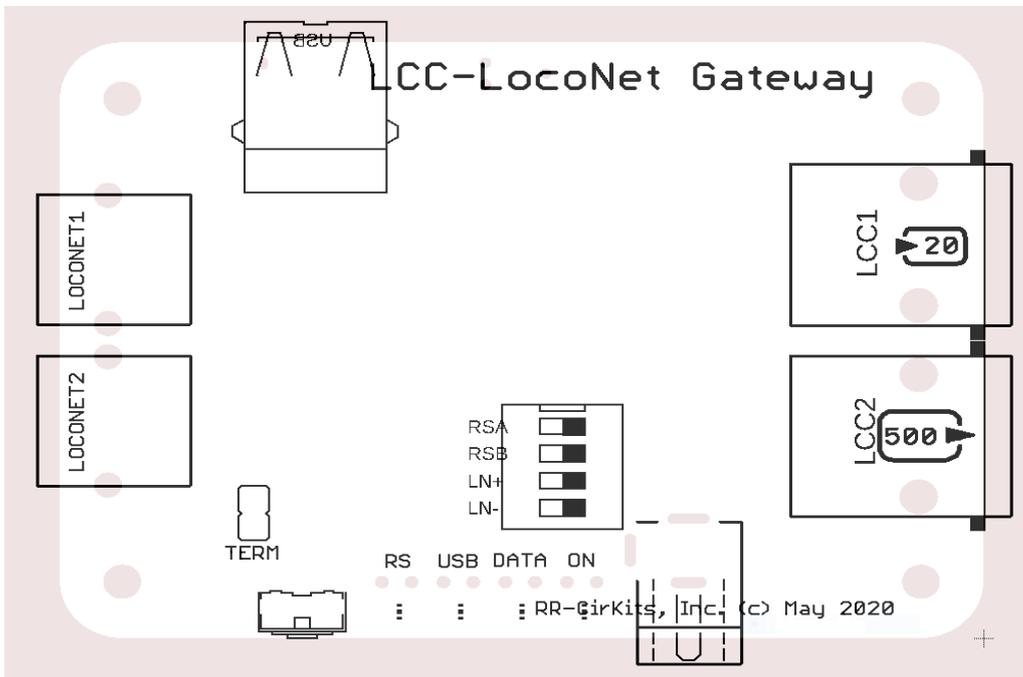
Copyright.....	2
Overview.....	5
1 About LCC.....	6
1.1 Some Definitions.....	7
1.1.1 Event.....	7
1.1.2 EventID.....	7
1.1.3 Well-Known EventID.....	7
1.1.4 Line (channel).....	7
1.1.5 Consumer.....	7
1.1.6 Producer.....	7
2 LCC-LoCoNet Gateway Feature Summary.....	8
2.1 Software Features.....	8
2.2 Hardware Features.....	8
3 Power, Serial Connections, and Indicators.....	9
3.1 Status Indicators.....	9
3.2 Prog Button.....	9
3.3 Power Connections.....	9
3.4 USB.....	9
3.5 CAN LCC® Compatible Connector.....	10
3.6 LoCoNet Connector Wiring.....	11
3.6.1 LoCoNet Configuration Options.....	12
3.6.2 Auto-Switch Model (No Manual Configuration).....	13
4 CDI Basics (in JMRI).....	14
4.1 Configure Nodes.....	14
5 CDI Segments.....	15
5.1 Segment: Identification.....	15
5.2 Segment: NODE ID.....	15
5.3 Segment: Special Options.....	16
5.3.1 Node Power Monitor.....	16
5.3.2 Master Power Control.....	17
5.3.3 Emergency Stop.....	17
5.3.4 Special Interrogate.....	17
5.3.5 Loop Count.....	18
5.3.6 CS Location.....	18
5.3.7 Well-Known Event Automatic Translation (DCC).....	19
5.3.7.1 DCC Accessory Decoders.....	19
5.3.7.2 DCC Turnout Feedback.....	19
5.3.7.3 DCC Sensor Feedback.....	19
5.3.7.4 Well-Known Event Generation.....	19
5.3.8 Fast Clock.....	20
5.3.8.1 Gateway Master Clock Controls.....	21
5.3.8.2 Gateway Resync Time Broadcast Settings.....	22
5.3.8.3 Fast Clock Translation Adjustments.....	23
5.3.8.4 Fast Clock Rate Limitations.....	24
5.4 Segment: Sections, Groups, and Lines.....	25
5.4.1 Section.....	25
5.4.2 Group.....	25
5.4.3 Line (I/O Channel).....	26

5.5 Sensor/Turnout Creation.....	27
6 Troubleshooting.....	28
6.1 Sanity Test.....	28
6.2 Activity Test.....	28
7 Boot Loader.....	29
7.1 Firmware Upgrade.....	29
8 Grounding and Isolation.....	30
9 Warranty Information.....	31
10 FCC Information.....	31
Appendix: Theory of operation.....	32
Case 1: All nodes initialized before link established, event X is sent.....	33
Case 2: Node B2 comes up, after event X has been sent in the example above	33
Case 3: Node A1 comes up, then event X is sent.....	33
Case 4: Node A1 comes up, then event Y is sent.....	34
Case 5: All nodes initialized before link established, event Z is sent.....	34
Case 6: Node A3 comes up, after event Z has been sent in the example above	34
Case 7: Node B1 comes up, then event Z is sent.....	35
Case 8: Special Interrogate Command is sent.....	35

Overview



LCC-LocoNet Gateway Image



LCC-LocoNet Gateway Connectors

The LCC®-LocoNet® Gateway interface provides a simple and easy way to connect between the NMRA LCC® bus and the Digitrax LocoNet bus. The LCC-LocoNet Gateway may be connected at any convenient point on the NMRA LCC® bus. LCC® is a registered trademark of the NMRA. www.nmra.org LocoNet® is a registered trademark of the Digitrax corporation. www.digitrax.com

1 About LCC

The NMRA LCC® is a subset of the OpenLCB specifications created by the OpenLCB group for Layout Command and Control. <http://www.openlcb.org/>

NMRA LCC® devices are controlled by EventIDs. Each EventID has a unique value that will never be repeated by any other LCC® event in use anywhere on your system, nor even on anyone else's system. The only meaning given to any specific EventID is that which you give it. It might represent a turnout command, (Rimrock East Reverse) a block detection message, (Rimrock East Main Occupied) or even a time such as 'Crack of Dawn' or 'High Noon'. Internally the LCC uses a 64 bit number to represent each event (18,446,744,073,709,551,615 possible EventIDs) so we are not planning to run out of EventIDs anytime soon. EventIDs are created by event 'Producers' and used by EventID 'Consumers'. The same EventID may be created by one or more Producers, and may be used by any number of Consumers. (or none at all)

EventIDs happen, they are not states nor the status of indicators. The only memory of EventIDs past exist in hardware. In other words an EventID can tell you to turn a light 'on'. A different EventID can tell you to turn a light 'off'. Many different EventIDs can tell you to turn the same light 'on'. However there is no EventID that tells you that the light is 'on'. That is a state, and only resides in the hardware that controls the light. (actually LCC defines other messages that can query nodes for their state during initialization)

Of course you are free to add another piece of hardware that watches for the same EventIDs and controls its own light to duplicate the original. You could even have a light that turns on when it sees the EventID 'Crack of Dawn' and turns off when it sees the EventID 'Evening'. You could label it "Daylight" and use it to tell you that the room lights should be 'on', or better yet, turn on the room lights itself.

1.1 Some Definitions

1.1.1 Event

Events are the real world changes on our layouts:

- the change in occupancy state of a Block
- the intent (pressing a button) to control the position of a turnout
- the feedback of the change in position of a turnout
- the triggering an animation as another physical thing happens
- the control of various signaling schemes based on sensed track positions and train movements etc.

1.1.2 EventID

Events on our layout are represented in LCC by EventIDs which are the actual messages sent over the system. Because events can represent anything that happens on our layouts there is no presumed context nor meaning given to EventIDs. They are simply globally unique 64 bit numbers. Their only definition is what you give them.

1.1.3 Well-Known EventID

Some EventIDs are given a meaning by the LCC developers for common purposes. This simply means that someone else has defined a particular EventID or group of EventIDs and published what it is. You are free to use it only if you accept its definition for use on your layout. One example might be the time related EventIDs used by system clocks. If a clock manufacturer uses the Well Known time EventIDs, then you can simply look up each EventID in a list and use it to cause something to happen at that time without copying the EventID from the clock node itself. If a second manufacturer decides to build a clock node and uses the same Well Known time EventIDs, then your nodes will work with his clock just as they did before.

1.1.4 Line (channel)

Each LCC-Loconet Gateway contains 1024 I/O lines. (or channels) Each line has the ability to watch for 2 events (consumers) and to send out 2 events. (producers) Each line translates its events into the opposite protocol and queues them for sending out to the appropriate bus.

1.1.5 Consumer

Each consumer event can be configured to control the line's output in one of two ways. These are; 'on' (activate), 'off' (inactivate).

1.1.6 Producer

Each producer event can be configured to trigger in one of 2 ways: These are function 'on'/'off'.

2 LCC-Loconet Gateway Feature Summary

These lists sum up important product features. New features are shown in **bold**.

2.1 Software Features

- Standard LCC node protocols following NMRA and OpenLCB specifications
- Bootloader mode for user-initiated firmware updates over LCC bus or USB
- Configurable options via LCC CDI (Configuration Description Information)
- LN throttle message translation to LCC (if LCC command station set in CDI)
- Manually configurable translation of 1024 Input/Output lines (channels)
- **Automatic DCC accessory and feedback translation** (enabled by default)
- **Automatic fast clock translation** (disabled by default)

2.2 Hardware Features

- Four LED status indicators
- Programming (Prog) button for bootloader mode to reprogram firmware
- Power supplied to unpowered Gateway from a powered LCC bus
- Coaxial power supply which powers the Gateway and LCC Bus
- USB-C port for LCC connection to PC (legacy model uses USB-B instead)
- Two RJ45 ports for one LCC bus (CAN bus implementation from NMRA)
- Two RJ12 ports for one Loconet bus
- Option to create a standalone Loconet for systems without a Digitrax command station (this option bypasses the Loconet isolation)
- Option to connect the Digitrax Railsync lines to the DCC pair of the LCC cable to allow the command station to feed boosters over the LCC cable
- **Automatic switching of last two features** (Auto-Switch model only)
- **Power and DCC fault detection/protection** (Auto-Switch model only)

3 Power, Serial Connections, and Indicators

The LCC-LoCoNet Gateway has six connectors and four status indicators. Two of these connectors are for connections to the LCC bus network. Two are used as connections to the LoCoNet lines. One is a power connection, and the remaining connector is for the USB connection. This section covers the system connections consisting of the CAN bus port connectors, LoCoNet connectors, power connection, USB port connection, and Status indicators.

3.1 Status Indicators

The LCC-LoCoNet Gateway has four status indicators located near to the coaxial power connector. The green **ON** status indicator shows the power status of the LCC-LoCoNet Gateway itself. The yellow **Data** indicator normally shows all data activity related to the LCC-LoCoNet Gateway, and also any activity/error status during power up or a boot loader firmware upgrade. The red **USB** indicator shows activity over the USB connection. The green **Sync** indicator shows power on the RailSync lines. (see 3.6 LoCoNet Connector Wiring)

3.2 Prog Button

The Prog Button may be used to force the unit into boot loader mode for firmware upgrades. Normally this is not required.

3.3 Power Connections

The LCC-LoCoNet Gateway requires a power source of between 7.5 and 27 volts DC connected. This power may be provided through the 5.5mm by 2.1mm coaxial connector and/or through the CAN bus power from another power source, such as an LCC Power-Point, or the RR-CirKits LCC Repeater (requires at least 20mA).

If powered via the coaxial power input, then the LCC-LoCoNet Gateway can act as the power source for a segment of LCC cable. The LCC-LoCoNet Gateway acts as a convenient way to power a small LCC network including other LCC boards over standard RJ45 cables (500mA per LCC connector).

3.4 USB

The LCC-LoCoNet Gateway includes an internal LCC to USB interface. The USB connection is optional and the Gateway function will operate with or without the USB connection. The USB interface is only to the LCC, and does not serve as a LoCoNet interface. The USB interface can serve as the interface for the LCC network instead of an LCC Buffer-USB or other interface device.

3.5 CAN LCC® Compatible Connector

The LCC data connection is made to the LCC-LoCoNet Gateway via an LCC cable connected to either of the two RJ45 connectors LCC1 and LCC2. The LCC wiring passes straight through both connectors. The LCC specification requires a minimum of 1 foot of cable length between connectors.

Pin outs for the CAN LCC RJ45 data connector:

Pin	Description
1	CAN H
2	CAN L
3	GND
4	DCC RailSync
5	DCC RailSync
6	GND
7	GND
8	+Power 12-27V

LCC power is supplied on Pin 7 and Pin 8. Power can be from +12VDC to +27VDC. The RR-CirKits LCC-LoCoNet Gateway delivers approximately 15VDC to the bus when powered by the included power supply.

The LCC connectors accept standard Ethernet style CAT5 (or better) cables. 4 pair cables are required by the LCC-LoCoNet Gateway. For any but the smallest networks it is recommended that you choose AWG 24 CAT-5 wiring. The use of AWG 26 wiring reduces the maximum length of your network to approximately 40% of its specified length. Especially avoid using copper clad aluminum wire or AWG 28 low profile wiring as they have even higher than normal resistance at the relatively low frequencies used by the LCC. This higher resistance shortens the maximum distance for reliable communications even more than using AWG 26 wiring does.

A note on connectors: RJ45 crimp connectors are made with three blade styles. (the end that crimps into/onto the wire) Single 'U', double 'UU', and triple 'VVV' points. Stranded cables may be made with any of the three blade styles because the points crimp into and between the individual wire strands. However if you are using solid wire, then you must only use the three point style of blade. It is designed to trap the solid wire between the three points, two on one side, and the center one on the other side, for a corrosion tight connection. The single or double pointed blades simply press against the side of the solid wire, and will fail in time. (usually the morning of your open house)

Each segment of LCC cable requires a terminator at each end.

For more LCC details, see publicly available documents provided by OpenLCB: <<https://openlcb.org/openlcb-and-lcc-documents/layout-command-control-lcc>>

3.6 LocoNet Connector Wiring

The LocoNet® connection is made to the LCC-LoCoNet Gateway via the provided cable connected to one of the RJ12 (6 pin) modular jacks. (LoCoNet1 or LoCoNet2)

LoCoNet® cables are wired straight through, not reversed like phone cables.

Pin outs for the RJ12 connector:

Pin	Description	Color
1	Rail_Sync-	white
2	Signal Ground	black
3	LoCoNet-	red
4	LoCoNet+	green
5	Signal Ground	yellow
6	Rail_Sync+	blue

LoCoNet® pins 2 and 5, and pins 3 and 4 are connected together internally.

For more LocoNet® details, see publicly available documents provided by Digitrax:

“LoCoNet, the Digitrax Difference”

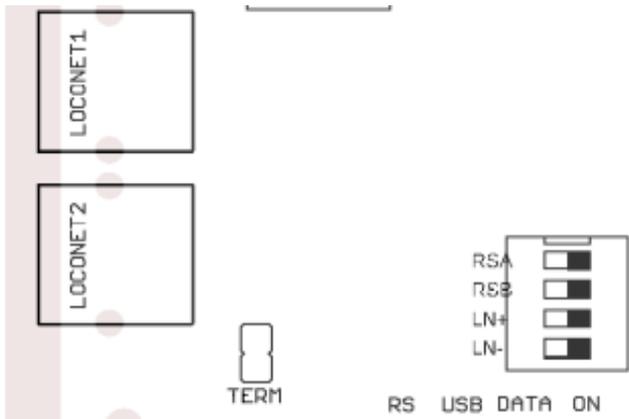
<https://www.digitrax.com/static/apps/cms/media/documents/documentation/LoCoNet_Overview_App_Note.pdf>

“LoCoNet® Personal Use Edition 1.0 SPECIFICATION”

<<https://www.digitrax.com/static/apps/cms/media/documents/loconet/loconetpersonaledition.pdf>>

3.6.1 LocoNet Configuration Options

The LCC-LoCoNet Gateway contains several internal option switches or jumpers related to the LocoNet operation. Terminology: **LEGACY / Jumper Configuration**

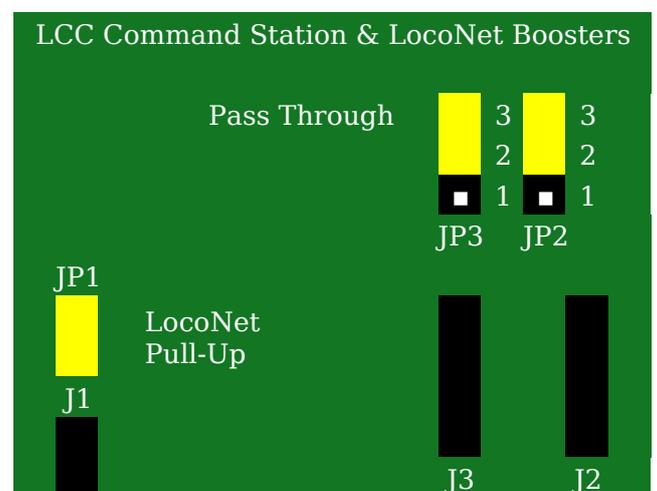
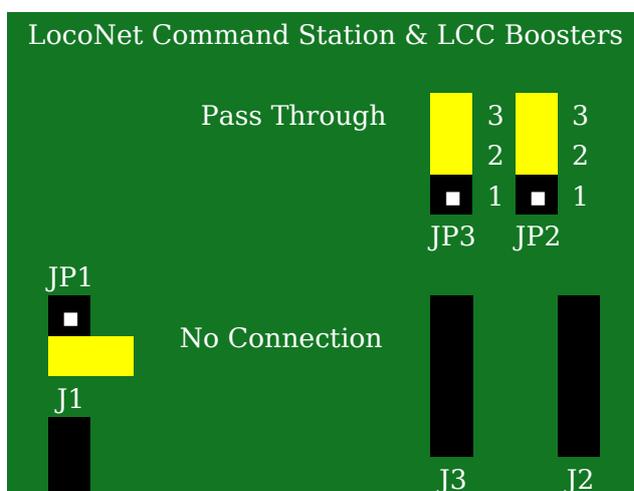
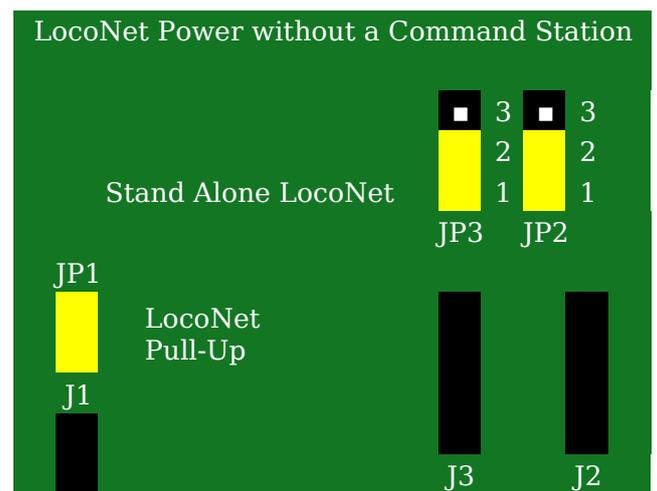
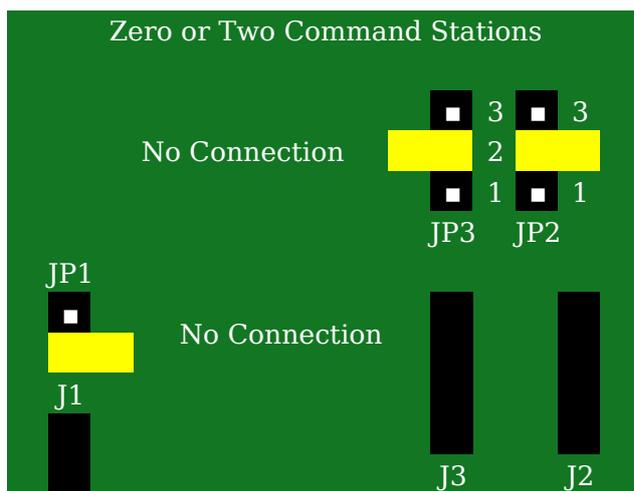


Legacy LocoNet Options with Switches

TERM / LocoNet Pull-Up provides the required LocoNet 15mA pull-up current for its stand alone operation. Do NOT connect this jumper if the LocoNet is being provided by another source.

LN+ & LN- Pair / Stand Alone LocoNet provides DC power to the RailSync lines in order to power LocoNet devices that normally receive their power from a command station. Do NOT activate if the RailSync is being provided by another source, else equipment may be damaged.

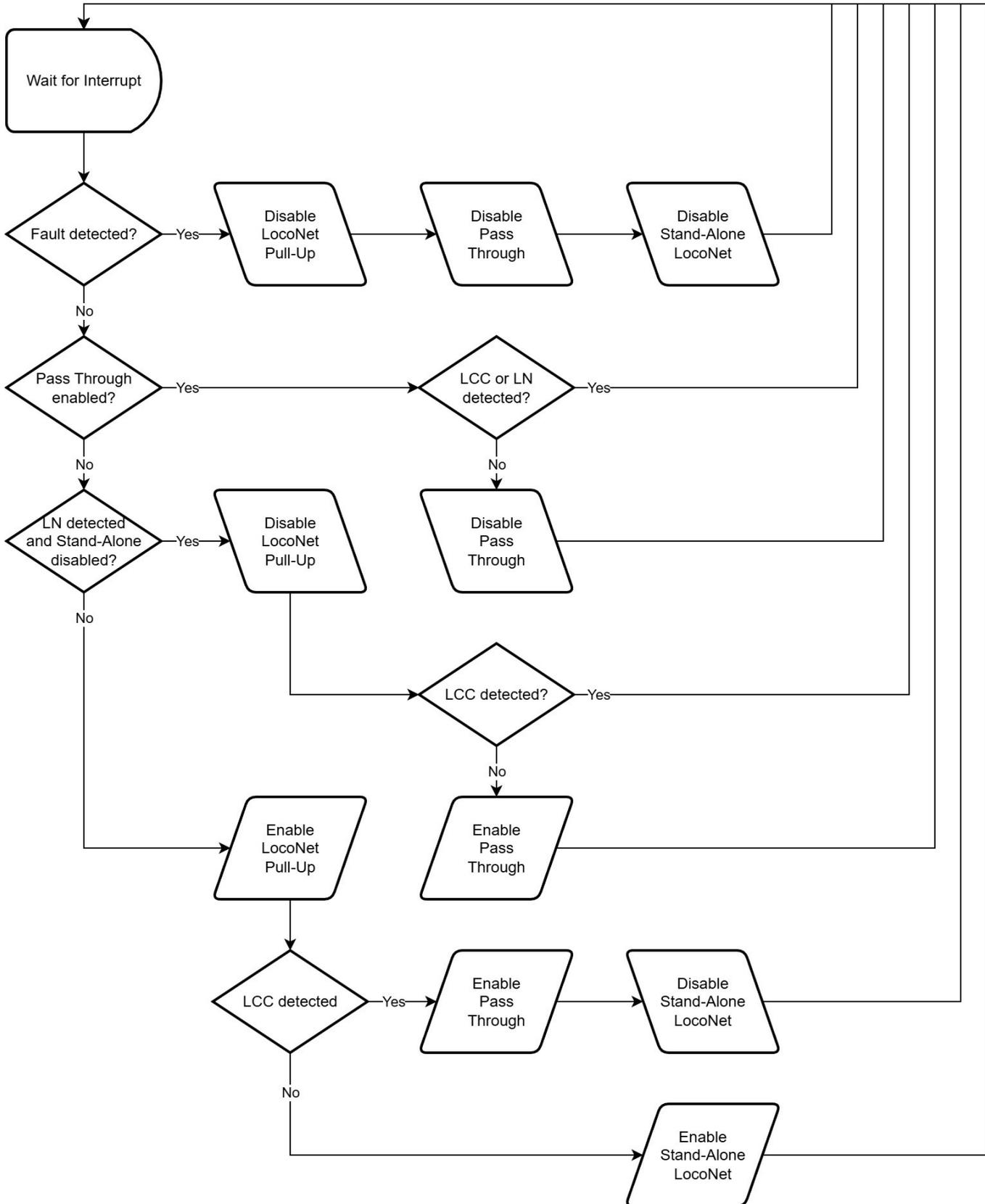
RSA & RSB Pair / Pass Through directly connects the DCC lines of the LCC and LocoNet buses. This allows a command station on one bus to drive boosters on the other bus. If the booster drive connection is not isolated then it may be required to install a common connection between the command station and boosters.



LocoNet Options via Jumper Configuration (J1, J2, J3 headers connect to Gateway)

3.6.2 Auto-Switch Model (No Manual Configuration)

A dedicated processor reconfigures switches automatically based on sensor inputs. Besides conveniently keeping LocoNet devices powered after losing LCC power, the Auto-Switch severs faulty connections before any hardware damage can occur.



4 CDI Basics (in JMRI)

An LCC node's Configuration Description Information (CDI) defines its settings. Instead of relying on printed manuals or volunteer created files to present the various decoder options (like DCC devices have for the many past decades), the LCC specification expects the manufacturer of the LCC node itself to present its capabilities and options in an intuitively standardized manner from an internal file. This allows any LCC configuration tool to be used interchangeably, and not need to be updated to support new hardware or firmware upgrades for a given LCC node.

This manual assumes the user has installed JMRI (5.10+) <<https://www.jmri.org>> (though there are alternatives such as Deepsoft <<https://www.deepsoft.com>>).

4.1 Configure Nodes

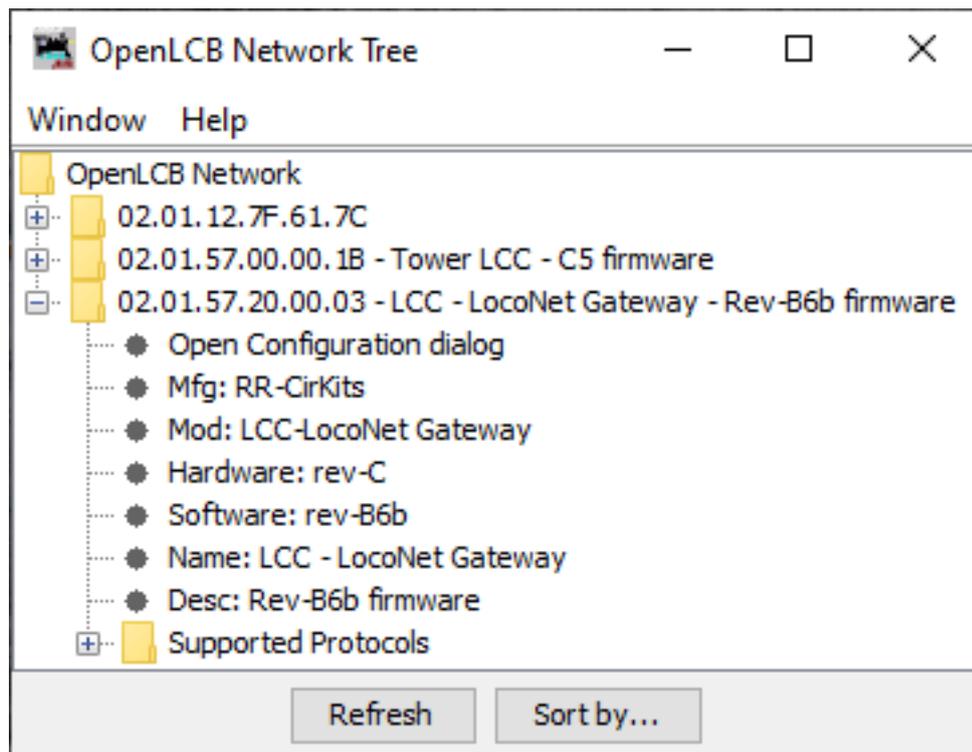
Select the 'LCC' drop down list from your JMRI program (e.g. PanelPro).

Click on 'Configure Nodes'.

Locate the window titled 'OpenLCB Network Tree'.

Expand the node (Check the label on the front of case for LCC ID to identify node).

Click on 'Open Configuration dialog' to open the CDI window for this node.



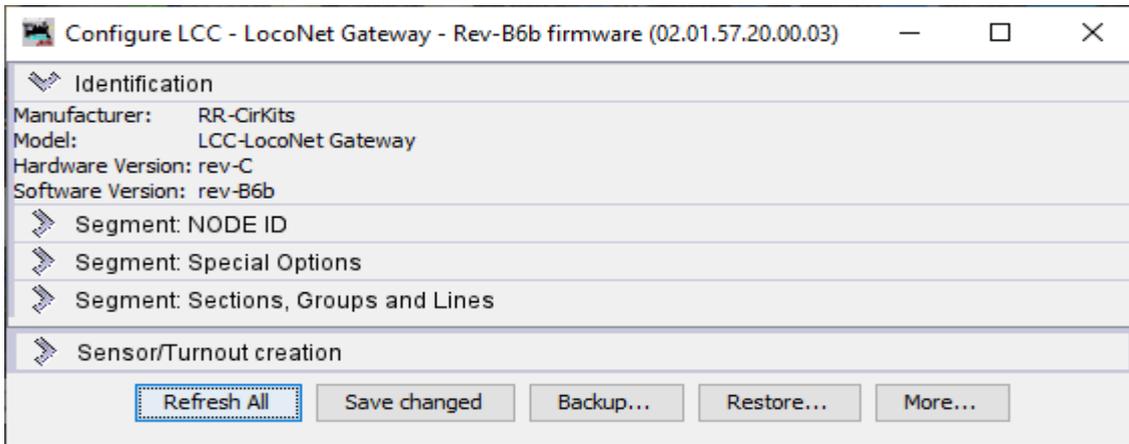
Locate the window titled 'Configure LCC-LoCoNet Gateway (02.01.57.21.XX.XX)'.

The CDI automatically reads in the basic information and default configuration. This information is presented in a tabular format to allow a reasonably compact display but still have easy access to the vast amount of configuration information.

Visit the link below for more details on how to open up the CDI from JMRI:
<<https://www.jmri.org/help/en/package/jmri/jmrix/openlcb/swing/networktree/NetworkTreePane.shtml>>

5 CDI Segments

The CDI is divided up into a number of segments to make it more manageable to configure. The LCC-LoCoNet Gateway CDI consists of four main segments.

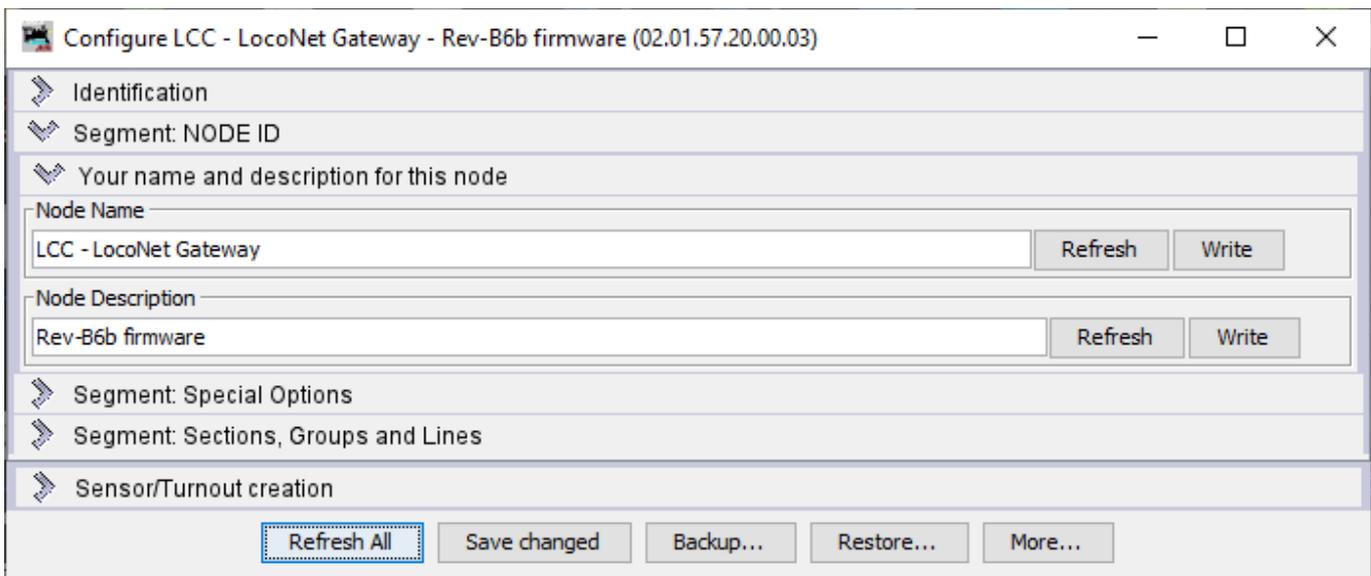


5.1 Segment: Identification

The first segment shown will be the Identification. It includes the manufacturers name and node model plus any version information.

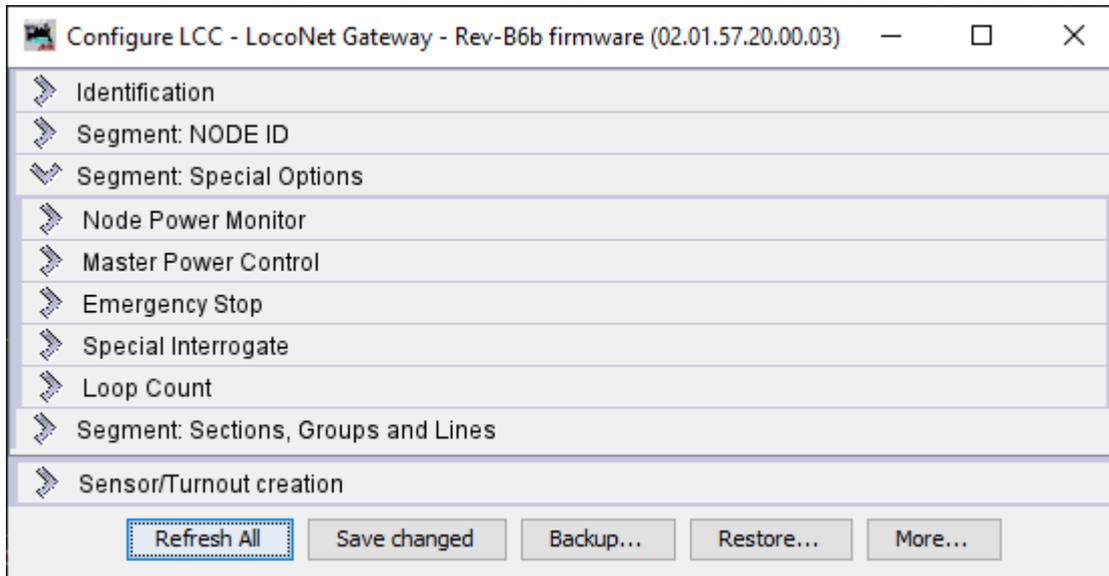
5.2 Segment: NODE ID

The next item is the Node Identification. It contains the Node Name and Node Description that you give to the node. The name of this node is 'LCC - LocoNet Gateway'. (not very original) This name will appear in the node selection window to make it easier to select the correct node for configuration. The Node Description in this example shows the firmware version, but normally would be more related to your layout. There is a 64 character limit to both the node name and description items. These will show next to the node number in the node tree, so short and descriptive is good.



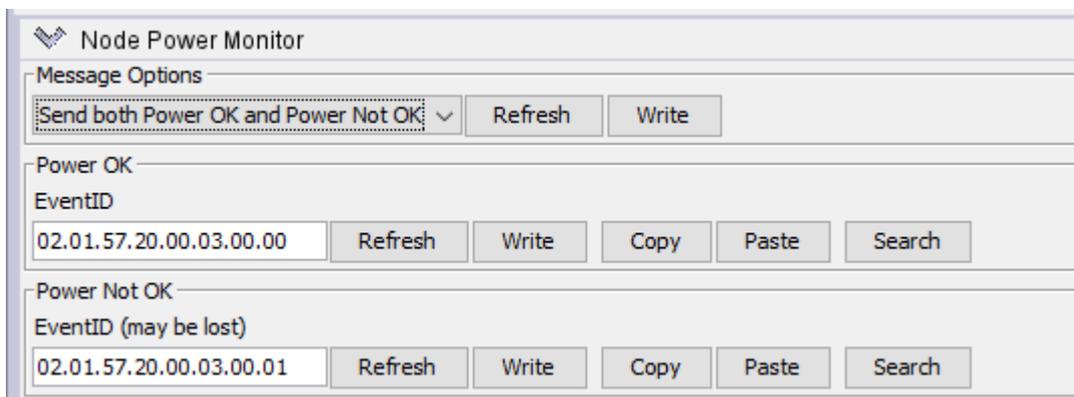
5.3 Segment: Special Options

The LCC-LoCoNet Gateway has a number of supported special options. Some of these are node related options, and others are specific LoCoNet commands that may be useful on the main LCC network.



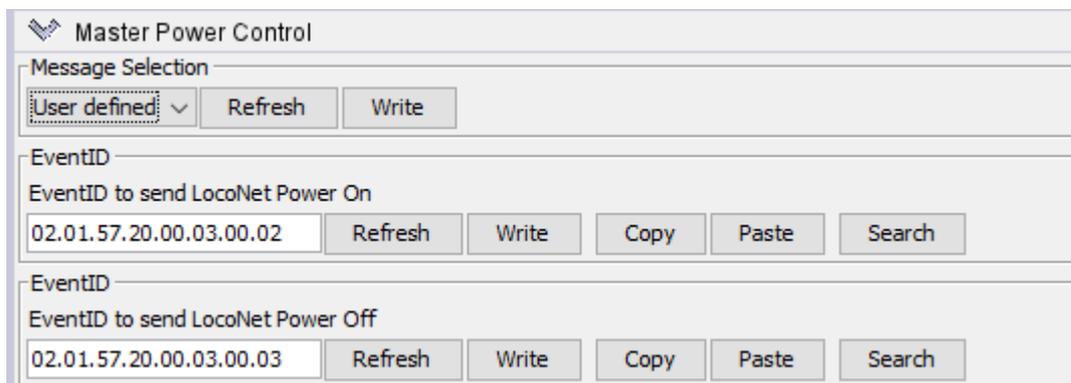
5.3.1 Node Power Monitor

LCC nodes may monitor their own power input and send notifications to the bus if they are in or out of specification. You may choose the EventID or use the defaults.



5.3.2 Master Power Control

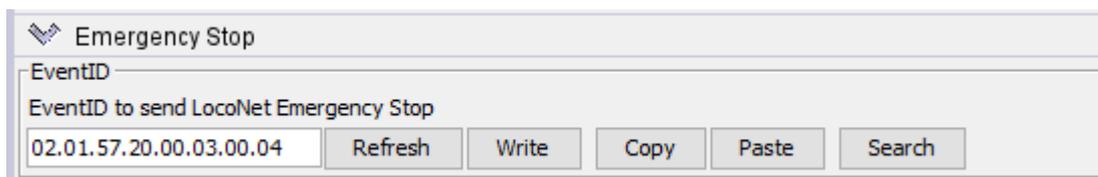
Master Power Control relates to the LocoNet messages GPON and GPOFF. (General Power On, and General Power Off) You can either select the default LCC well known *power on* and *power off* EventIDs, use the node defaults, or define your own.



The screenshot shows the 'Master Power Control' window. It has a 'Message Selection' section with a dropdown menu set to 'User defined' and 'Refresh' and 'Write' buttons. Below this are two 'EventID' sections. The first is for 'EventID to send LocoNet Power On' with a text box containing '02.01.57.20.00.03.00.02' and buttons for 'Refresh', 'Write', 'Copy', 'Paste', and 'Search'. The second is for 'EventID to send LocoNet Power Off' with a text box containing '02.01.57.20.00.03.00.03' and the same set of buttons.

5.3.3 Emergency Stop

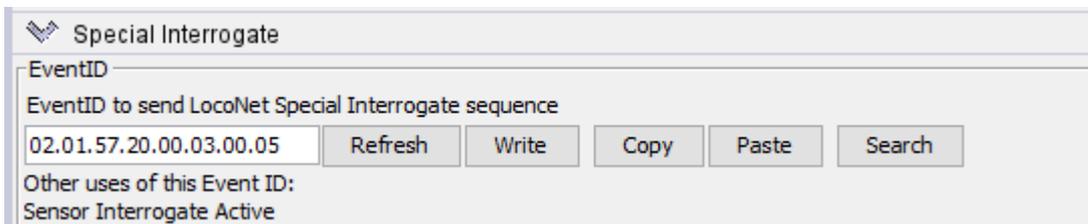
Emergency Stop is simply what it says. It is an Event ID that will send the LocoNet Emergency Stop command. (85 7A Emergency STOP) The command station puts all selected throttles into Emergency Stop. (speed 1)



The screenshot shows the 'Emergency Stop' window. It has an 'EventID' section with a text box for 'EventID to send LocoNet Emergency Stop' containing '02.01.57.20.00.03.00.04' and buttons for 'Refresh', 'Write', 'Copy', 'Paste', and 'Search'.

5.3.4 Special Interrogate

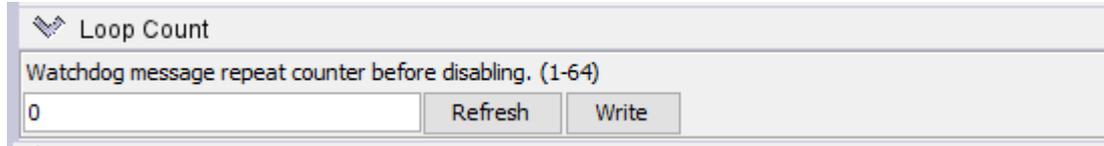
Special Interrogate sends a sequence of queries on the LocoNet bus. DCC devices on LocoNet are expected to respond with messages indicating their current states. For example, you could use a trigger button on a JMRI panel to conveniently poll.



The screenshot shows the 'Special Interrogate' window. It has an 'EventID' section with a text box for 'EventID to send LocoNet Special Interrogate sequence' containing '02.01.57.20.00.03.00.05' and buttons for 'Refresh', 'Write', 'Copy', 'Paste', and 'Search'. Below this, it lists 'Other uses of this Event ID: Sensor Interrogate Active'.

5.3.5 Loop Count

If the LocoNet fails in a manner that causes rapidly repeated messages this watchdog will disable the forwarding of those messages to the LCC network. Any (non zero) loop count value sets the maximum number of any specific LocoNet messages allowed to be translated per second. If this number is exceeded then the line (channel) is disabled until a 1 second timer is reset.

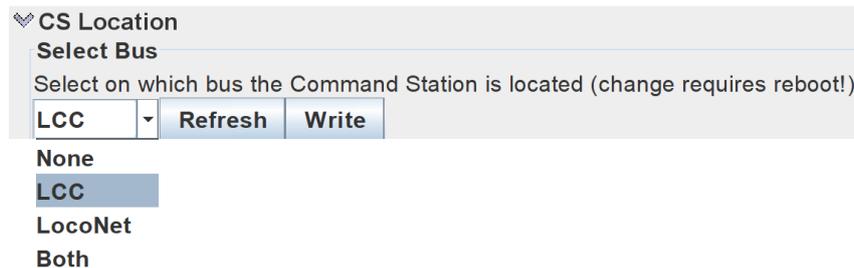


The screenshot shows a configuration window titled "Loop Count". Below the title is a text label: "Watchdog message repeat counter before disabling. (1-64)". There is a text input field containing the number "0". To the right of the input field are two buttons: "Refresh" and "Write".

5.3.6 CS Location

This setting primarily determines whether the Gateway will simulate locomotive slots for LocoNet throttles to control an LCC command station and has no bearing on the hardware configuration (no data passes to/from the Auto-Switch processor).

At this time, the "None" option functions identically as "LocoNet" and similarly "LCC" functions the same as "Both". These are intended to make it clear what to choose depending on the set-up. Note, the "Both" option is there for Auto-Switch Gateways or when there are no jumper connections. There is absolutely no support for using both command stations beyond translating messages between the buses.



The screenshot shows a configuration window titled "CS Location". Below the title is a section labeled "Select Bus" with the instruction "Select on which bus the Command Station is located (change requires reboot!)". There is a dropdown menu currently showing "LCC". To the right of the dropdown are two buttons: "Refresh" and "Write". Below the dropdown menu is a list of options: "None", "LCC", "LocoNet", and "Both". The "LCC" option is highlighted with a blue background.

5.3.7 Well-Known Event Automatic Translation (DCC)

The following options are enabled to quickly set up LocoNet accessory messages to translate into well-known LCC events. Disable if using same events in custom lines.

5.3.7.1 DCC Accessory Decoders

Basic DCC Accessory translates switch requests (commands) between LN and LCC. *Close request*: the last bit is set to high. *Throw request*: the last bit is set to low.

The 11 bits preceding the last bit represent the turnout address with an offset of 4.

Extended DCC Accessory translates signal aspect messages between LN and LCC. In the LCC event, the last byte is the signal aspect with all possible values (0-255). The 11 bits preceding the last byte represent the signal address with an offset of 4.

▼ Well-Known Event Automatic Translation
Automatically translate LCC well-known events (keep a category disabled if translating its well-known events in the lines instead)

DCC Accessory Decoders (Switch Commands and Signal Aspects)
Automatically translate LCC well-known events for basic (and possibly also extended) DCC accessory decoders to/from LocoNet (.0X.XX represents address with an offset and .YY represents signal aspect)

Basic and Extended ▼ Refresh Write

Disabled

Basic (01.01.02.00.00.FF.0X.XX/01.01.02.00.00.FE.0X.XX)

Extended (01.01.02.00.01.0X.XX.YY)

Basic and Extended

5.3.7.2 DCC Turnout Feedback

DCC Turnout Feedback translates switch sensor reports between LN and LCC.

Closed: the last bit is set to high. *Thrown*: the last bit of the fifth byte is high, "FD".

Using two bits allows for two sensors to verify the switch has made proper contact. The 11 bits preceding the last bit represent the turnout address with an offset of 4.

DCC Turnout Feedback
Automatically translate LCC well-known events for DCC turnout feedback to/from LocoNet (.0X.XX represents DCC address with an offset)

Enabled (01.01.02.00.00.FD.0X.XX/01.01.02.00.00.FC.0X.XX) ▼ Refresh Write

5.3.7.3 DCC Sensor Feedback

DCC Sensor Feedback translates general sensor reports between LN and LCC.

Active: the last bit of the fifth byte is high, "FB". *Inactive*: the same bit is low, "FA".

The last 12 bits represent the general sensor address with an offset of 4.

DCC Sensor Feedback
Automatically translate LCC well-known events for DCC sensor feedback to/from LocoNet (.0X.XX represents DCC address with an offset)

Enabled (01.01.02.00.00.FB.0X.XX/01.01.02.00.00.FA.0X.XX) ▼ Refresh Write

5.3.7.4 Well-Known Event Generation

The simplest way to generate a well-known event in JMRI is by right-clicking in an event field (or hitting More button) and selecting from "Insert well-known event".

DCC Accessory Decoder events can be transformed into Turnout Feedback events by changing the "FF" byte to "FD" if Thrown or "FC" if not Thrown (likely Closed).

Extended DCC Accessory events require manual entry (e.g. copied from monitor).

5.3.8 Fast Clock

Enable LCC Clock ID to auto-translate LocoNet Fast Clock messages to/from LCC. The ID options are either from one of four well-known clock events, which any LCC clock can use, or a custom clock event that will be defined in the following setting. Last two event bytes are not included here since they are specific to commands.

Fast Clock

LCC Clock ID

Automatically translate clock events from this node ID to/from LocoNet depending on master clock location (the middle options are well-known IDs and the last option requires user input below)

Custom User Clock (xx.xx.xx.xx.xx.xx) Refresh Write

Disabled

Default Fast Clock (01.01.00.00.01.00)

Default Real-Time Clock (01.01.00.00.01.01)

Alternate Clock 1 (01.01.00.00.01.02)

Alternate Clock 2 (01.01.00.00.01.03)

Custom User Clock (xx.xx.xx.xx.xx.xx)

Custom User Clock ID should be set to the LCC clock ID if option selected above.

Custom User Clock ID

Sets the node ID (first 6 bytes "xx.xx.xx.xx.xx.xx") to translate as clock events when "Custom User" has been selected above (note, all 8 bytes must be filled in but last 2 bytes are ignored ".00.00")

02.02.01.01.51.13.00.00 Refresh Write More... Copy Paste Search

Master Clock Location determines which bus will be treated as master when considering which messages to translate. Both has no filters on what to translate, but may lead to undefined behavior when two master clocks are in active conflict. None will not translate messages between buses, instead it tells the Gateway that it needs to act as the master clock to both buses, i.e. the Gateway Master Clock.

Master Clock Location

Select on which bus the generating clock is located (None defaults Gateway's fast clock as master, which does nothing unless configured below) (Both forwards all clock messages and may result in conflicts)

None (Gateway is master) Refresh Write

None (Gateway is master)

LCC

LocoNet

Both

5.3.8.1 Gateway Master Clock Controls

The Gateway Master Clock Controls sub-segment contains all necessary settings to control the Gateway Master Clock rate, running state, starting time, or initial date. These values are passed on to each bus in a manner dependent on other settings. Remember Master Clock Location must be "None" and JMRI version at least 5.10. Note that the Gateway Master Clock requires another clock to display the time.

Gateway Master Clock Rate can be set to any LCC standard rate (-512 to 511.75). Fast minutes tick this many times per real minute (0.25 is 1 tick every 4 minutes). Positive rates tick forwards, negative rates tick backwards, and zero freezes time.

▼ Gateway Master Clock Controls
Master clock location must be "None" for these settings to have any effect (JMRI version must be at least 5.10)

Gateway Master Clock Rate
Set the rate that fast clocks will increment every real minute (refresh to display stored value, which is a multiple of 0.25 between -512 to 511.75)

60.0 Refresh Write

The toggle button either starts stopped clocks or stops running clocks immediately.

Gateway Master Clock Toggle Running State

Start/Pause

Gateway Master Clock Reset Time and Date allows the time and date to be reset immediately upon pressing the Reset button. Date is only a feature of LCC clocks.

Gateway Master Clock Reset Hour

0 6 12 18 23 Refresh Write

Gateway Master Clock Reset Minute

0 15 30 45 19 Refresh Write

Gateway Master Clock Reset Time and Date

Reset

▼ Gateway Master Clock Date Reset
These values have no effect on LocoNet as it has no concept of a date

Gateway Master Clock Reset Year

0 2000 4000 1970 Refresh Write

Gateway Master Clock Reset Month

January Refresh Write

Gateway Master Clock Reset Day

1 16 31 1 Refresh Write

5.3.8.2 Gateway Resync Time Broadcast Settings

The Gateway Resync Time Broadcast Settings allow for the Gateway to keep clocks synchronized via periodic time messages broadcasting to the designated clocks.

Gateway Resync Period sets how much time passes between each resync message.

Gateway Override is a mode that allows the Gateway to replace timekeeping on the designated clocks with a time message every fast minute that also freezes the rate.

Gateway Resync Time Broadcast Settings
These settings are relevant for users that want the Gateway to send periodic time resynchronization broadcasts or even override fast clock timekeeping altogether

Gateway Resync Period
Set the minimum amount of time between resynchronization messages from the Gateway (may be slightly longer as resync check only occurs once every fast minute)

1 fast minute (Gateway Override) ▾	Refresh	Write
1 fast minute (Gateway Override) ▲	4 real minutes	
15 real seconds	5 real minutes	
30 real seconds	10 real minutes	
45 real seconds	15 real minutes	
1 real minute (Standard)	30 real minutes	
2 real minutes	45 real minutes	
3 real minutes	1 real hour	
4 real minutes	Disabled	

Gateway Resync Location designates which buses will receive resync messages.

Gateway Resync Location
Select which bus will receive a time update every resync period

LocoNet ▾	Refresh	Write
None (Disabled)		
LCC		
LocoNet		
Both		

Gateway Resync Authority designates which clock types receive resync messages.

Gateway Resync Authority
Set which type of clock of will be addressed by resync messages on the selected bus location(s)

Consumer/Slave ▾	Refresh	Write
Consumer/Slave		
Generator/Master		
Both		

5.3.8.3 Fast Clock Translation Adjustments

Fast Clock Translation Adjustments can account for usual clock translation issues.

LocoNet Time Format sets which way the LocoNet fast minute byte is interpreted. Legacy clocks are incompatible with modern clocks, so pick one type per LocoNet.

Fast Clock Translation Adjustments

These settings allow for necessary adjustments in time and rate while translating

LocoNet Time Format

Choose the time format most applicable to your fast clocks on the LocoNet bus (legacy clocks receiving modern formatted messages will be off by a minute and behave strangely when minutes roll over to the next hour)

Legacy (LNFC, DT400+) Refresh Write

Modern (JMRI, DT600+)

Legacy (LNFC, DT400+)

LocoNet Headstart Minute Offset accounts for certain devices ticking immediately.

LocoNet Headstart Minute Offset

Select an offset which adjusts the LocoNet time relative to LCC or Gateway Master (accounts for a fast clock headstart error on certain devices such as Digitrax throttles)

0 (Default, JMRI, LNFC) Refresh Write

0 (Default, JMRI, LNFC)

-1 fast minute (DT400+)

LocoNet Rate Multiplier accounts for clocks on LocoNet running slower than LCC.

LocoNet Rate Multiplier

Select a multiplier which adjusts the LocoNet rate relative to LCC or Gateway Master (allows for more clock rate accuracy without excessive resyncs) (precision depends on rate, so test multiple rates to find a satisfactory multiplier)

+3.7% (LNFC)	Refresh	Write			
+0.0% (No change)	+0.8%	+1.6%	+2.4%	+3.2%	+4.3%
+0.1%	+0.9%	+1.7%	+2.5%	+3.3%	+4.4%
+0.2%	+1.0%	+1.8%	+2.6%	+3.4%	+4.5%
+0.3%	+1.1%	+1.9%	+2.7%	+3.5%	+4.6%
+0.4%	+1.2%	+2.0%	+2.8%	+3.6%	+4.7%
+0.5%	+1.3%	+2.1%	+2.9%	+3.7% (LNFC)	+4.8%
+0.6%	+1.4%	+2.2%	+3.0%	+3.8%	+4.9%
+0.7%	+1.5%	+2.3%	+3.1%	+3.9%	+5.0%

5.3.8.4 Fast Clock Rate Limitations

LocoNet Maximum Rate should be changed for clocks that cannot support 127. Gateway Override applies to LocoNet clocks if the rate goes above this or below 0.

LocoNet Maximum Rate	
Set the maximum rate that devices on the LocoNet bus can handle (rates above this value are considered unsupported, as well as rates below zero, and will result in Gateway Override on the LocoNet bus) (must account for LN Rate Multiplier)	
64 (LNFC)	<input type="button" value="Refresh"/> <input type="button" value="Write"/>
0 (Only Gateway Override)	30
1	60
2	63 (DT600+)
3	64 (LNFC)
4	90
5	100 (JMRI Clock Tool)
10	120
15	127 (Default, DT400+)

LCC Minimum/Maximum Rate default values likely do not need to be changed. Gateway Override applies to LCC clocks if the rate goes above or below this range.

LCC Minimum Rate	
Set the minimum rate that devices on the LCC bus can handle (rates below this value are considered unsupported and will result in Gateway Override on the LCC bus)	
-512.0	<input type="button" value="Refresh"/> <input type="button" value="Write"/>
LCC Maximum Rate	
Set the maximum rate that devices on the LCC bus can handle (rates above this value are considered unsupported and will result in Gateway Override on the LCC bus)	
511.75	<input type="button" value="Refresh"/> <input type="button" value="Write"/>

5.4 Segment: Sections, Groups, and Lines

The LCC-Loconet Gateway has 1024 bi-directional translation lines (IO channels). To prevent presenting 1024 different tab choices we chose to organize them as 4 sections of 256. Each Section is further divided into 16 groups of 16 lines each. With the option to label each section, group, and line it should be possible to configure your gateway in a way that matches your Loconet layout hardware.

The screenshot shows a web-based configuration interface for the LCC-Loconet Gateway. The main window is titled "Segment: Sections, Groups and Lines". It features four tabs labeled "Section 1", "Section 2", "Section 3", and "Section 4". The "Section 1" tab is currently selected. Below the tabs, there are three main configuration sections:

- Section name:** A text input field followed by "Refresh" and "Write" buttons.
- Group:** A row of 16 numbered buttons (1-16) followed by a "Group name" text input field and "Refresh" and "Write" buttons.
- Line (Channel):** A row of 16 numbered buttons (1-16) followed by a "Line name" text input field and "Refresh" and "Write" buttons.

5.4.1 Section

Each section includes 1/4th of the gateway lines. (channels) If it is convenient to divide your Loconet into sections this can help you. If not, then give them some arbitrary names.

5.4.2 Group

Each group includes 16 gateway lines. This conveniently supports the number of lines in some of the Digitrax hardware items. There is no actual requirement that the Loconet addresses be consecutively numbered, but it can be helpful when setting up your gateway to do so. This is especially true if you name your groups to match your hardware.

5.4.3 Line (I/O Channel)

Each line (or channel) controls the translation of one LocoNet item such as a turnout (output) or sensor. (input) The LCC-LoCoNet Gateway also supports the NMRA extended accessory messages that are normally used for controlling signal aspects. Each extended accessory message support as many as 32 possible states.

The screenshot shows the configuration window for a Line (Channel). At the top, there is a grid of 16 channels numbered 1 to 16. Below this is a 'Line name' text box with 'Refresh' and 'Write' buttons. The 'LocoNet (input/output)' section contains an 'Address (1-2048) (1-4096 for sensors)' text box with 'Refresh' and 'Write' buttons. The 'Message type' section features a dropdown menu currently set to 'Switch Request (Turnout) - Send Output(On)-(Off)', with 'Refresh' and 'Write' buttons. The 'LCC' section has two 'EventID' fields. The first is for 'EventID for message (On - Closed - Active - 1st Aspect)' with the value '02.01.57.20.00.03.00.06' and buttons for 'Refresh', 'Write', 'Copy', 'Paste', and 'Search'. The second is for 'EventID for message (Off - Thrown - Inactive - 2nd Aspect)' with the value '02.01.57.20.00.03.00.07' and the same set of buttons.

Configuration Items

- Line name: Your identification for the line being converted.
- LocoNet Address: The address portion of the message (0 to disable).
- Message type: Select a specific type of LocoNet message for this translation.

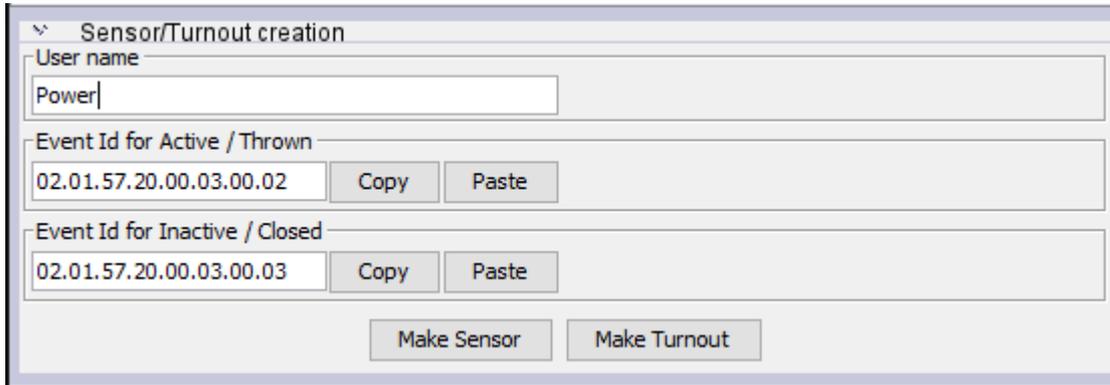
This is a close-up of the 'Message type' dropdown menu. The menu is open, displaying a list of options. The first option, 'Switch Request (Turnout) - Send Output(On)-(Off)', is selected and highlighted in blue. Other visible options include 'Switch Request (Turnout) - Send Output(On) only', 'Turnout Feedback (Output Levels)', 'Turnout Feedback (Switch Sensor Input)', 'Turnout Feedback (Aux Sensor Input)', 'General Sensor Input', 'Signal Aspects 0-1', and 'Signal Aspects 2-3'. To the right of the dropdown, 'Refresh' and 'Write' buttons are visible.

- EventID for message (On...): The first LCC event used for 2-way translation. You may use the globally unique default EventID or enter your own.
- EventID for message (Off...): Second LCC event used for 2-way translation. You may use the globally unique default EventID or enter your own.

For signal aspects you only need to enter aspects that are actually used. However this can cause still create unused items if, for example, you have 'Aspect 0' in your system, but not 'Aspect 1'. Selecting 'Signal Aspects 0-1' gives both translations.

5.5 Sensor/Turnout Creation

Sensor/Turnout Creation is not actually part of the CDI, but is a feature of JMRI to make it easy to enter LCC EventIDs into the JMRI Sensor or Turnout tables.



The screenshot shows a dialog box titled "Sensor/Turnout creation". It contains three input fields and two buttons. The first field is labeled "User name" and contains the text "Power". The second field is labeled "Event Id for Active / Thrown" and contains the text "02.01.57.20.00.03.00.02", with "Copy" and "Paste" buttons to its right. The third field is labeled "Event Id for Inactive / Closed" and contains the text "02.01.57.20.00.03.00.03", also with "Copy" and "Paste" buttons to its right. At the bottom of the dialog are two buttons: "Make Sensor" and "Make Turnout".

Simply use the Copy and Paste features to select the appropriate EventID values from each line and move them into the tool. You may either type a User Name or cut and paste it from the item. (using Ctrl+C and Ctrl+V) Click on the appropriate [Make Sensor] or [Make Turnout] button and a matching entry will automatically be entered in JMRI. Be sure to save panels in order for JMRI to save this information for future use.

6 Troubleshooting

6.1 Sanity Test

To perform a very basic LCC-LoCoNet Gateway sanity test perform the following steps:

- Power up the LCC-LoCoNet Gateway by plugging it into a powered LCC network, or by connecting a power supply to the unit.
- The green power **ON** LED should come on.
- If no other node is present on the network, then the yellow **Data** light will begin to flash a 4 blink error code while the board seeks to establish an alias.
- Alternatively connect the USB jack to your computer. The device should appear in your active USB device list.
- No power light will show, but the same 4 flash error code will begin to appear on the yellow **Data** light.

If the green power LED does not light, be sure that a power supply is connected to the LCC network segment, or provide at least 9V to the LCC-LoCoNet Gateway power jack. The green power LED will initially light at much lower voltages, so it is not a reliable indicator of suitable power.

6.2 Activity Test

The LCC-LoCoNet Gateway's USB circuit and code sends data directly to the unit's processor, so if you send any command to the unit over its USB connection, then it should immediately be seen on the red **USB** LED. This test uses the free software available from the JMRI project to watch the test commands. (www.jmri.org)

Steps:

- Open the JMRI LCC® Monitor window. Using the JMRI turnout control send a command to any configured lines on this LCC-LoCoNet Gateway. The command should appear in the LCC® monitor window and the LCC-LoCoNet Gateway yellow **Data** LED should blink.
- The mapped LoCoNet command should be sent out to the LoCoNet.

If there is activity at the LCC Terminator blue LED, but no activity light at the LCC-LoCoNet Gateway when events are sent, check the LCC wiring. If the command is seen in the LCC® monitor, but not in the command light, be sure that the command you are sending is configured to respond on this LCC-LoCoNet Gateway. If there is no activity shown in the LCC® monitor window, check that you have the correct interface selected in the JMRI preferences, and that you have the correct COM port selected.

7 Boot Loader

7.1 Firmware Upgrade

If an update to your LCC-LoCoNet Gateway firmware is needed, a program such as "Firmware Update" in JMRI version 4.6 or later is required.

To enter Firmware upgrade mode:

- 1) Start JMRI and select "LCC".
- 2) Select 'Firmware Update' from the LCC drop down list.
- 3) Click 'Select' to pick a firmware file.
- 4) From the file menu, select: 'LCC_LN_Gateway_RevD_UPDATE_B9b.hex' or the latest upgrade available. Older devices with USB-B will require 'RevC'.
- 5) Optionally you may check the 'Lock Node' check box to take it off line during the upgrade.
- 6) Click the 'Load' button to initiate the upgrade to LCC-LoCoNet Gateway revision B9b.
- 7) Wait until 'updating device firmware..' is complete.
- 8) Switch back to the OpenLCB Network Tree window.
- 9) It should now show 'Mod: LCC-LoCoNet Gateway' and 'Software: rev-B9b'.
- 10) Any errors will be shown in the lower window ticker tape display.

If the node does not automatically enter boot mode and start the upgrade it may be forced into boot mode by un-powering it, then holding down the 'Gold' button as you power it up again. The gold LED should start flashing to indicate that it is in forced boot mode. This will likely be required after a failed upgrade attempt.

Note: When upgrading from the built in interface you need to provide external power during the entire process. However the USB connection will be reset before and after the firmware upgrade, causing JMRI to loose its connection during the upgrade process. You will need to restart JMRI and continue the process from the point where JMRI disconnected each time.

8 Grounding and Isolation

The LCC-LoCoNet Gateway provides 2,500 VAC electrical isolation between the computer, the LCC bus, and the LoCoNet[®] by using two high speed digital isolators. This prevents possible ground loop problems between the LoCoNet[®], the LCC, and your computer. For example, many systems are installed without properly bonding the booster to power ground nor to each other.

Some interface devices may connect the LoCoNet[®] ground directly to the USB cable ground. This provides a ground path from the power line through your computer and its USB cable to the LoCoNet[®] and then to the booster and all the railroad wiring. At best this indirect grounding causes electrical noise. At worst it could create a fault path via the small gauge LoCoNet[®] wiring to ground.

By isolating the LCC bus ground, the LoCoNet ground, and the USB ground from each other the LCC-LoCoNet Gateway prevents the inadvertent introduction of additional ground loop paths between your different networks.

Note: for the LCC and LoCoNet isolation to be effective you should not use the option for creating a stand alone LoCoNet using the LCC-LoCoNet Gateway, as this will use the same power supply for both networks and connect their grounds. For normal LoCoNet connections to a command station this isolation is maintained. For connecting isolated LoCoNet devices on a stand alone LoCoNet network there is no concern with connecting the LCC and LoCoNet networks.

9 Warranty Information

We offer a one year warranty on the LCC-LoCoNet Gateway. This device contains no user serviceable parts.

If a defect occurs, please contact RR-CirKits at: service@rr-cirkits.com for a replacement.

10 FCC Information

This device complies with part 15 of the FCC Rules. Operation is subject to the following two conditions:

1. This device may not cause harmful interference, and
2. this device must accept any interference received, including interference that may cause undesired operation.

Note: This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/TV technician for help.

Any modifications to this device voids the user's authority to operate under and be in compliance with these regulations. The actual measured radiation from the LCC-LoCoNet Gateway is much lower than the maximum that is permitted by the FCC Rules, so it is unlikely that this device will cause any RFI problems.

RR-CirKits, Inc.
7918 Royal Ct.
Waxhaw, NC USA 28173

<https://www.rr-cirkits.com>
sales@rr-cirkits.com
service@rr-cirkits.com
1-704-843-3769
Fax: 1-704-243-4310

Appendix: Theory of operation

Note:

This information is from the OpenLCB web site and covers the operation of Gateways in general, not the LCC-Loconet Gateway specifically.

Gateways

Gateways can route PCERs only to segments with nodes expressing interest by processing the other event messages, including Consumer and Producer Identified. The automatically-routed event range always has to be routed. This can be done by making a permanent entry in routing tables, or any other implementation method.

Examples

Consider a OpenLCB installation with one CAN segment A and one LOCONET segment B connected via a gateway. In addition, a program is attached via a USB link and LocoBuffer-USB "GB2" to LOCONET segment B.

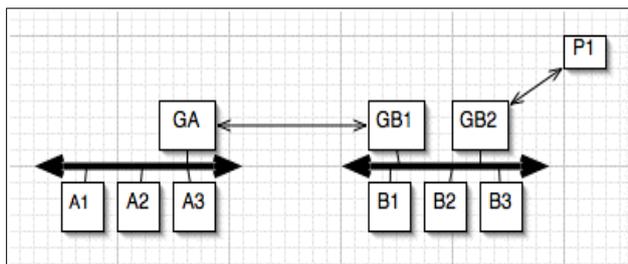


Figure 1 - Example OpenLCB Network

The program has to be able to fully communicate with all OpenLCB nodes on segment A and all Loconet nodes on segment B. These examples shows how the gateway can make that happen. For simplicity, the labels on the figure will be used as if they were the full NID of the node, e.g. "GA" for the gateway on segment A.

Gateway "GB1" on the LOCONET side doesn't physically exist, but lives inside gateway "GA" and is represented by a routing table. Frames and messages on CAN are represented by ["source address", "content", "destination address"] triplets, those on LOCONET by ["opcode", "destination address", "content"]. Not all messages are shown. Note that there are many ways that gateways can manage traffic, and this is just one approach.

Before the cases below, both segments come up, assign the numbers 1,2,3 as aliases to their local nodes. For simplicity, A1 produces event X, which A2 and B2 want to consume. A1 also produces Y, which no other node wants to consume. B1 produces event Z, which A3 wants to consume.

Case 1: All nodes initialized before link established, event X is sent

The combined GA-GB1 gateway initializes its internal tables.

A1 sends X. GA receives it. Because GA hasn't seen X before, X is (internally) forwarded to GB1 where it is looked up in the routing table. If it exists and has a message type assigned to it, X is translated into a LOCONET message of that type.

GB1 puts the translated X message on the LOCONET segment, and B2 consumes it.

Since we have no way of knowing whether message X exists on B2, we have to rely on the entries in the routing table, which are put in by the user.

GB1 notes message sent, forwards to GA, which marks its routing table to continue forwarding these. GA forwards "Consumer of X Identified" on the CAN segment in case there are other listeners for it. If the LOCONET is disconnected or not powered, GB1 cannot send message X and will not forward "Consumer of X Identified" back to GA.

Case 2: Node B2 comes up, after event X has been sent in the example above

The gateway is up, but doesn't know about B2. GA has seen X sent by A1, but has marked it as "do not forward" if it exists and has a message type assigned in the routing table, because there was no consumer for it on B.

B2 comes up, but the gateway still doesn't know about B2. Note that B2 can only come up if the LOCONET segment is connected and powered.

GB1 detects that the LOCONET segment is up, which changes all "do not forward" markers in the routing table into "forward" markers.

GA now remembers that event X must be forwarded across this link if seen.

A1 sends X, GA receives it, forwards it across link.

Case 3: Node A1 comes up, then event X is sent

The gateway is up, but doesn't know about A1 or X.

A1 comes up, gets a NID alias assigned, sends "AMD" and "Initialization Complete".

A1 sends "Producer of X Identified". GA forwards that to GB1.

GB1 notifies GA that event X, if it exists and has a message type assigned in the routing table, currently must be forwarded across this link if/when seen.

A1 sends X. GA receives it. Because GA hasn't seen X before, X is forwarded across the link to GB1.

The rest of this proceeds like Case 1.

Case 4: Node A1 comes up, then event Y is sent

The gateway is up, but doesn't know about A1 or Y.

A1 comes up, gets a NID alias assigned, sends "AMD" and "Initialization Complete".

A1 sends "Producer of Y Identified".

GA remembers that event Y currently must be forwarded across this link if seen.

A1 sends Y. GA receives it. Because GA hasn't seen Y before, Y is forwarded across the link to GB1.

If Y doesn't exist in the routing table or has no message type assigned to it, no "Consumer of Y Identified" message is received back from GB1, so after a short timeout GA marks it's tables to not forward Y. (If a node later comes up that consumes Y, a sequence similar to Case 2 occurs.)

Case 5: All nodes initialized before link established, event Z is sent

The combined GA-GB1 gateway initializes its internal tables.

B1 sends Z. GB1 receives it. Because GB1 hasn't seen Z before, Z is translated and forwarded across the link to GA if Z exists in the routing table.

GA puts the Z message on the CAN segment, and A3 consumes it.

GA puts the "Identify Consumers of Z" message on the CAN segment.

A3 responds with "Consumer of Z Identified".

GA receives "Consumer of Z Identified", forwards to GB1, which marks its routing table to continue forwarding these.

Case 6: Node A3 comes up, after event Z has been sent in the example above

The gateway is up, but doesn't know about A3. GB1 has seen Z sent by B1, but has marked it as "do not forward" because there was no consumer for it on A.

A3 comes up, gets a NID alias assigned, sends "AMD" and "Initialization Complete".

A3 sends "Consumer of X Identified". GA forwards that to GB1.

GB1 remembers that event X must be forwarded across this link if seen.

B1 sends Z, GB1 receives it, forwards it across link.

Case 7: Node B1 comes up, then event Z is sent

The gateway is up, but doesn't know about B1 or Z.

B1 comes up, but the gateway still doesn't know about B1. Note that B1 can only come up if the LOCONET segment is connected and powered.

GB1 detects that the LOCONET segment is up, which changes all "do not forward" markers in the routing table into "forward" markers.

GB1 now remembers that event Z currently must be forwarded across this link if seen.

B1 sends Z. GB1 receives it. Because GB1 hasn't seen Z before, Z is forwarded across the link to GA, followed by an "Identify Consumers of Z" message.

The rest of this proceeds like case 5.

Case 8: Special Interrogate Command is sent

The gateway is up, but doesn't know anything about segment B.

P1 comes up, and sends the Special Interrogate Command (8 Interrogate Switch Requests). The gateway can also issue this command by itself if GA receives the "Special Interrogate" event.

Nodes on segment B respond to this command by sending Turnout Sensor State messages <0xB1>.

GB1 receives each sensor message and proceeds like Case 5.

GB1 receives the first "Interrogate Switch Request", then checks the routing table for all "forward" markers GA → GB1. For each "forward" marker that has message type "Turnout Feedback" or "Sensor Message", a message is generated on segment B. All further "Interrogate Switch Requests" will be ignored. A maximum of 10 messages will be sent every 100 milliseconds, in order to not overload the LOCONET bus.