

NMRA Technical Note	
Layout Command Control™ (LCC)	
Event Identifiers	
July 22, 2024	TN-9.7.0.4

Adopted as a NMRA Technical Note

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.

Version History

Date	Adopted	Summary of Changes
Feb 17, 2015		Initial version submitted for public comment
Feb 6, 2016	Feb 20, 2016	Minor grammatical corrections and readability improvements as well as the following specific changes: <ul style="list-style-type: none"> • Moved 2.5.1.1 MERG CBUS section down to section 2.5.3.2 MERG CBUS • Significantly expanded the explanations for all the subsections of 2.5 Allocation section
Apr 25, 2021	July 2, 2021	Changed LCC logo to include the ® symbol Changed “Layout Command Control” to have the ™ symbol Added the NMRA Legal Disclaimer fine-print Changed the OpenLCB license to “Creative Commons Attribution-ShareAlike 4.0 International” Add section 2.5.2.3 Power Supply Brownout Detected Add section 2.5.3.3 Basic DCC Accessory Decoder Add section 2.5.3.4 Advanced DCC Accessory Decoder
July 22, 2024		Adds Train Search protocol event range Adds 11-bit extended DCC accessory address range Reformat some tables

Important Notices and Disclaimers Concerning NMRA Standards Documents

The Standards (S), Recommended Practices (RP), Technical Note (TN) and Technical Information (TI) documents of the National Model Railroad Association (“NMRA Standards documents”) are made available for use subject to important notices and legal disclaimers. These notices and disclaimers, or a reference to this page, appear in all standards and may be found under the heading “Important Notices and Disclaimers Concerning NMRA Standards Documents.”

Notice and Disclaimer of Liability Concerning the Use of NMRA Standards Documents

NMRA Standards documents are developed within the Standards and Conformance Department of the NMRA in association with certain Working Groups, members, and representatives of manufacturers and sellers. NMRA develops its standards through a consensus development process, which brings together volunteers representing varied viewpoints and interests to achieve the final product. NMRA Standards documents are developed by volunteers with modeling, railroading, engineering, and industry-based expertise. Volunteers are not necessarily members of NMRA, and participate without compensation from NMRA.

NMRA does not warrant or represent the accuracy or completeness of the material contained in NMRA Standards documents, and expressly disclaims all warranties (express, implied and statutory) not included in this or any other document relating to the standard or recommended practice, including, but not limited to, the warranties of: merchantability; fitness for a particular purpose; non-infringement; and quality, accuracy, effectiveness, currency, or completeness of material. In addition, NMRA disclaims any and all conditions relating to results and workmanlike effort. In addition, NMRA does not warrant or represent that the use of the material contained in NMRA Standards documents is free from patent infringement. NMRA Standards documents are supplied “AS IS” and “WITH ALL FAULTS.”

Use of NMRA Standards documents is wholly voluntary. The existence of an NMRA Standard or Recommended Practice does not imply that there are no other ways to produce, test, measure, purchase, market, or provide other goods and services related to the scope of the NMRA Standards documents. Furthermore, the viewpoint expressed at the time that NMRA approves or issues a Standard or Recommended Practice is subject to change brought about through developments in the state of the art and comments received from users of NMRA Standards documents.

In publishing and making its standards available, NMRA is not suggesting or rendering professional or other services for, or on behalf of, any person or entity, nor is NMRA undertaking to perform any duty owed by any other person or entity to another. Any person utilizing any NMRA Standards document, should rely upon their own independent judgment in the exercise of reasonable care in any given circumstances or, as appropriate, seek the advice of a competent professional in determining the appropriateness of a given NMRA Standards documents.

IN NO EVENT SHALL NMRA BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO: THE NEED TO PROCURE SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE PUBLICATION, USE OF, OR RELIANCE UPON ANY STANDARD OR RECOMMENDED PRACTICE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE AND REGARDLESS OF WHETHER SUCH DAMAGE WAS FORESEEABLE.

Translations

NMRA’s development of NMRA Standards documents involves the review of documents in English only. In the event that an NMRA Standards document is translated, only the English version published by NMRA is the approved NMRA Standards document.

Official Statements

A statement, written or oral, that is not processed in accordance with NMRA policies for distribution of NMRA communications, or approved by the Board of Directors, an officer or committee chairperson, shall not be considered or inferred to be the official position of NMRA or any of its committees and shall not be considered to be, nor be relied upon as, a formal position of NMRA.

Comments on Standards

Comments for revision of NMRA Standards documents are welcome from any interested party, regardless of membership. However, **NMRA does not provide interpretations, consulting information, or advice pertaining to NMRA Standards documents.**

Suggestions for changes in documents should be in the form of a proposed change of text, together with appropriate supporting comments. Since NMRA standards represent a consensus of concerned interests, it is important that any responses to comments and questions also receive the concurrence of a balance of interests. For this reason, NMRA, its departments, Working Groups or committees cannot provide an instant response to comments, or questions except in those cases where the matter has previously been addressed. For the same reason, NMRA does not respond to interpretation requests. Any person who would like to participate in evaluating comments or in revisions to NMRA Standards documents may request participation in the relevant NMRA working group.

Laws & Regulations

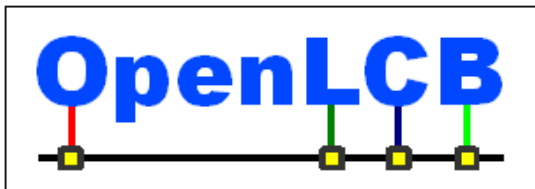
Users of NMRA Standards documents should consult all applicable laws and regulations. Compliance with the provisions of any NMRA Standards document does not constitute compliance to any applicable regulatory requirements. Implementers of the standard are responsible for observing or referring to the applicable regulatory requirements. NMRA does not, by the publication of NMRA Standards documents, intend to urge action that is not in compliance with applicable laws, and NMRA Standards documents may not be construed as doing so.

Copyrights

NMRA Standards documents are copyrighted by NMRA under US and international copyright laws. They are made available by NMRA and are adopted for a wide variety of both public and private uses. These include both use, by reference, in laws and regulations, and use in private self-regulation, standardization, and the promotion of modeling, structural and engineering practices and methods. By making NMRA Standards documents available for use and adoption by public authorities and private users, NMRA does not waive any rights in copyright to the NMRA Standards documents.

IMPORTANT NOTICE

NMRA Standards documents do not guarantee or ensure safety, security, health, or environmental protection, or ensure against interference with or from other systems, devices or networks. NMRA Standards documents development activities consider research and information presented to the standards development group in developing any safety recommendations. Other information about safety practices, changes in technology or technology implementation, or impact by peripheral systems also may be pertinent to safety considerations during implementation of the standard. Implementers and users of NMRA Standards documents are responsible for determining and complying with all appropriate safety, security, environmental, health, and interference protection practices and all applicable laws and regulations.



OpenLCB Technical Note

Event Identifiers

July 22, 2024

Adopted

1 Introduction

This technical note contains informative discussion and background for the corresponding “OpenLCB Event Identifiers Standard”. This explanation is not normative in any way.

2 Annotations to the Standard

- 5 This section provides background information on corresponding sections of the Standard document. It's expected that two documents will be read together.

2.1 Introduction

- 10 Although event identifiers generally take the format of a Unique Node Identifier plus a 16-bit extension, there are some exceptions to this noted in the Event Identifiers Standard which will be detailed in this document.

2.2 Intended Use

- 15 An Event Identifier is intended to be globally unique and for a specific purpose defined either by user configuration or enforced by the Event Identifiers Standard. Any node may be configured to produce or consume any event, regardless of its assigned Node ID. However, logically, a unique Event Identifier is defined to have a unique purpose, whether assigned by user configuration or explicitly defined in the Event Identifier Standard.

- 20 The "globally unique" requirement only refers to the universe of connected nodes; nodes that never need to communicate with each other don't need to have separate Event Identifiers. In general, however, nodes can move: they can be sold or loaned for use on another layout, nodes on modular layouts can be connected to other arbitrary modules, and few assumptions can be made. Therefore, we require global uniqueness for all Event Identifiers.

To ensure uniqueness, the top six bytes of an Event Identifier that the manufacturer or user defines are required to be within a Node ID space that the manufacturer or user controls. The low two bytes can be any number, so long as each value is used for only one event.

- 25 This requirement applies equally to events defined by a hardware node, like a push button, a software node in a computer, or Event Identifiers that are defined by a human writing them on a piece of paper. In each case, the thing doing the definition must ensure it has control over the Node ID corresponding to the top six bytes, so it can ensure that the Event Identifier not be reused.

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
Node ID assigned to you						Unique 16 bit extension	

30

The Node ID part can be from real nodes which a user owns.

A software configuration tool might define events and assign Event Identifiers to them. A board manufacturer may prefer a pushbutton configuration process. A modular club may decide that certain events form the “boundaries” of modules, and need to be assigned Well-Known Event Identifiers to make it easier to create large modular layouts. Whatever the method, the Event Identifiers need to be globally unique, which is ensured by requiring Event Identifiers to be created using Node ID numbers assigned to them (and therefore not assigned to anybody else), plus an additional 16 bits that they are responsible for using only once. This is worth repeating, each Event Identifier can only be used once for a specific state or meaning. It doesn't mean that multiple nodes cannot use it, but rather it should not be used for a different purpose or meaning. This is because the meaning and its use is shared across all of the nodes using it, and unless it can be guaranteed that they all have been changed there will be conflict between them. It is much safer to use a new Event Identifier, and nodes will usually have a mechanism to supply new, “virgin” Event Identifiers.

Note that the Node ID part of an Event Identifier does not have to correspond to any physical node. So long as it is assigned out of a Node ID address space that the assigning body owns, and, by extension, has control of over, it can be used. For example, a fast-clock manufacturer might want to define a range of Event Identifiers so that a different event is emitted every fast minute. For the sake of argument, let's say 24 bits worth of specific events are needed (it's actually smaller, but let's use this as an example). The manufacturer has a large range of Node IDs assigned already, so can use 8-bits worth of that space, plus the extra 16 bits in the Event Identifiers below the Node ID portion, to do this. If a manufacturer had a range of Node IDs that included at least 12.34.56.78.9A.00 through 12.34.56.78.9A.FF (in other words, all possible values of the low byte), they could create a set of Event Identifiers with the low 24 bits used to carry a time value:

Table 2: Example of defining a Set of Events

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
Assigned Node ID						Unique 16 bit extension	
0x12	0x34	0x56	0x78	0x9A	Time Value		

55

This is guaranteed to be unique, no matter where one of these devices is used, because the node ID range is guaranteed to belong to the manufacturer, and they will use it only once for this purpose.

2.3 References and Context

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

- The CAN Physical Layer Standard, which specifies the physical layer for transporting OpenLCB-CAN frames.
- The Message Network Standard, which defines the basic messages and how they interact. Higher-level protocols are based on this message network, but are defined elsewhere.

- The Event Transport Standard, which defines the protocol for transporting events.
- 65 • The Unique Identifiers Standard which defines the format and allocation of unique 48-bit identifiers.

2.4 Format

Event Identifiers are intended to be globally unique 64-bit values.

2.5 Allocation

70 2.5.1 Node ID Based

The majority of Event Identifiers fall into this category. The six most significant bytes are derived from a globally unique six byte Node ID followed by the two least significant bytes which are typically chosen through user configuration.

2.5.1.1

75 2.5.2 Well-Known Automatically-Routed

Unlike Node ID Based Event Identifiers, the Well-Known Automatically Routed Event Identifiers are explicitly defined by the Event Identifiers Standard for a specific purpose. Additionally, gateways are required to route these events to all segments.

80 Note a change that it is not specified whether or not the Well-Known Automatically Routed events participate in producer/consumer identify. The primary purpose of the producer/consumer identify is to pair consumers with produces, especially for the purpose of routing between OpenLCB segments. Since these Well-Known Automatically-Routed events must always be routed, it could be argued that the identify events for these Well-Known Event Identifiers are redundant, but are retained for symmetry.

85 Though not explicitly required, it would be prudent for a manufacture to identify in a product's documentation as to which Well-Known Automatically Routed events it produces and consumes, and what are the resulting actions it takes when these events occur.

2.5.2.1 Emergency Off

90 The Emergency Off Event Identifier (01.00.00.00.00.00.FF.FF) is a request for a node to de-energize all of its outputs. A node receiving this event may continue to remain a powered participant of the OpenLCB bus, but may de-energize any outputs unrelated to maintaining OpenLCB communications. The meaning of de-energize is not prescribed for any given node, it is up to the node manufacturer and/or user to prescribe what, if anything, should happen in the node if it receives this event. For
95 example:

- If the node is a DCC Power Station, it might disable its amplified DCC output.
- If the node is an accessory controlling turnout motors, it might remove power from the motors

A node may revert to its previous energized state following the reception of a Clear Emergency Off event (01.00.00.00.00.00.FF.FE).

100 A node that has recently joined the OpenLCB network is not expected to know about or react in any specifically prescribed way to the current Emergency Off status defined prior to the node joining the network until the next Emergency Off or Clear Emergency Off event is produced.

2.5.2.2 Emergency Stop

105 The Emergency Stop Event Identifier (01.00.00.00.00.00.FF.FD) is a request for a node to command all of its outputs to a safe state. A node receiving this event is not required to de-energize any of its outputs. The meaning of “safe state” is not prescribed for any given node, it is up to the node manufacturer and/or user to prescribe what, if anything, should happen in the node if it receives this event. For example:

- 110
- If the node is acting on behalf of one or more DCC trains, it might send the global emergency stop command onto the DCC signal bus.
 - If the node is an accessory controlling turnout motors, it might do nothing, route its outputs to a manufacture default state, route its outputs to a user defined state, or something else altogether.

A node may revert to its previous non-Emergency Stop state following the reception of a Clear
115 Emergency Stop event (01.00.00.00.00.00.FF.FC). A node that has recently joined the OpenLCB network is not expected to know about or react in any specifically prescribed way to the current Emergency Stop status defined prior to the node joining the network until the next Emergency Stop or Clear Emergency Stop event is produced.

2.5.2.3 Power Supply Brownout Detected

120 The Power Supply Brownout Detected (node) Event Identifier (01.00.00.00.00.00.FF.F1) may optionally be sent by a node in the case where the node detects a low voltage condition that it considers abnormal, or out of its normal operating range. This event is agnostic to any other OpenLCB standard, and is a generic indication that it does not have enough voltage to function properly, regardless as to where that voltage originates. The node is not required to know the exact voltage at which the condition
125 was detected in order to send this event.

The Power Supply Brownout Detected (standard) Event Identifier (01.00.00.00.00.00.FF.F0) may optionally be sent by a node in the case where the node detects a low voltage condition that crosses the threshold required by any relevant OpenLCB Standard. It is implied that the node producing this event is subject to, and subsequently detecting a violation of, one or more OpenLCB Standards. The CAN
130 Physical Layer Standard requirement of nodes operating down to CAN bus voltages of 7.5V is one example.

The exact time at which these events are sent is not specified and left up to the node’s designer. A number of possible options might be considered, which are not necessarily mutually exclusive. For example:

- 135
- The event may be sent immediately. Because the node is in a brownout state, it is reasonable not to assume that the event actually gets sent successfully and observed by other nodes in the network.

140

- The condition which would have resulted in the event may be logged such that it can be sent at a later time when the voltage has recovered to a point where successful transmission and observation by other nodes in the network is likely.
- The condition which would have resulted in the event may be logged such that it can be sent on the next node initialization when the voltage has recovered to a point where successful transmission and observation by other nodes in the network is likely.

145

It is not specified how other nodes in the network are to act upon these two events or that these two events are distinguished in any way when interpreted and presented to a user. The intent is that these two events, when observed, are useful for diagnosing network problems.

150

It is not specified how any given node will produce or inhibit production of these events when reacting to otherwise normal inrush conditions which may temporarily result in a low voltage condition which the node is designed to tolerate and recover from. The trade-off between false positive and false negative detection of brownout conditions is left up to an individual node's designer(s).

2.5.2.4 Other Well-Known Automatically Routed

Other Well-Known Automatically Routed Event Identifiers not discussed here have their uses prescribed and/or discussed elsewhere. No assumptions should be made about the use of these Well-Known Automatically Routed Event Identifiers as prescribed by this document.

155

2.5.3 Well-Known

Unlike Node ID Based Event Identifiers, the Well-Known Event Identifiers are explicitly defined by the Event Identifiers Standard for a specific purpose. Gateways are not required to actively route these events to all segments, and may maintain a static or learned routing table for these events to prevent unnecessary propagation.

160

Though not explicitly required, it would be prudent for a manufacturer to identify in a product's documentation as to which Well-Known events it produces and consumes and what are the resulting actions it takes when these events occur.

165

The following sections provide commentary to individual allocations in this table. Other Well-Known Event Identifiers not discussed here have their uses prescribed and/or discussed elsewhere. No assumptions should be made about the use of these Well-Known Event Identifiers as prescribed by this document.

2.5.3.1 Duplicate Node ID

170

The Duplicate Node ID Detected event would typically be sent by a node that receives a packet from another node containing its own unique Node ID. The production of this event by a node is not explicitly required.

2.5.3.2 MERG CBUS

OpenLCB allocates a Node ID address space specifically for mapping MERG CBUS events into OpenLCB. CBUS events come in two types: long and short. A long event is uniquely identified by the CBUS Node ID and Event ID. A CBUS short event is translated into an OpenLCB event by using

175 0x0000 as the Node ID in the “CBUS Node ID” field of the OpenLCB event identifier. This is allowed because in the CBUS standard, node-id 0x0000 is not permitted.

CBUS events can also come in the form of a request. These requests should be translated into an Identify Producer on OpenLCB in order to solicit the response that the CBUS request is asking for.

180 Two Identify Producer messages will have to be sent, one for the ON state range and another for an OFF state range, the response of which will need to be translated back for CBUS.

2.5.3.3 Basic DCC Accessory Decoder

The Well-Known Event ID space for basic DCC accessory decoders is provided as a suggestion to manufactures on how to translate OpenLCB events to classic DCC accessories. It is possible that there could be future mapping schemes required to implement a use case not covered by this scheme. This scheme is not designed to be mutually exclusive of other possible schemes, and its use is not required

185 when mapping between OpenLCB events and classic DCC accessories.

The format of the DCC packet for a basic DCC Accessory Decoder is as follows (most significant bit first):

Byte 1	Byte 2	Byte 3 (Error Byte)
0b10A ₇ A ₆ A ₅ A ₄ A ₃ A ₂	0b1 $\bar{A}_{10}\bar{A}_9\bar{A}_8$ DA ₁ A ₀ R	0bEEEEEEEE
Bits 5 down to 0 are the middle 6 bits of the address.	Bits 6 down to 4 are the 3 most significant bits of the address in 1's complement form. Bits 2 down to 1 are the 2 least significant bits of the address. Bit 3 is set for active, clear for inactive. Bit 0 is used to distinguish between a pair of outputs.	

190 The Accessory Decoder address is 11 bits in the form of A₁₀..A₀. The pair bit ‘R’ adds a 12th least significant address bit. Though the $\bar{A}_{10}\bar{A}_9\bar{A}_8$ address bits are in 1's complement form under the DCC standards, they are in standard non 1's complement form in OpenLCB format.

By convention, most DCC Accessory Decoders couple two outputs together to form an output pair, and use the least significant address bit ‘R’ to select the direction of travel for the output pair as a whole. In

195 this mode, the active/inactive bit ‘D’ is always sent as set (activate) or ignored altogether by the DCC decoder. The effective address space is cut in half (max of 2048 addresses).

There is some ambiguity between DCC systems in the market in how the 1-2048 addresses that most DCC users are familiar with translate to the bit level addresses sent over the rails. There are generally

200 two schemes that are deployed. The NMRA S-9.2.1 DCC standard defines these as Linear and Non-Linear. It is not prescribed how the OpenLCB user interfaces maps to the bit level events, however the NMRA S-9.2.1 DCC Standard requires the Linear scheme be used in all new designs.

User Address	Linear Addressing Scheme	
	DCC Format	OpenLCB Format
Reference	0b $\bar{A}_{10}\bar{A}_9\bar{A}_8A_7A_6A_5A_4A_3A_2A_1A_0R$	0b $A_{10}A_9A_8A_7A_6A_5A_4A_3A_2A_1A_0R$
1 (reverse)	0b111000001000	0b000000001000
1 (normal)	0b111000001001	0b000000001001
...
2044	0b000111111111R	0b111111111111R
2045	0b11100000000R	0b00000000000R
2046	0b11100000001R	0b00000000001R
2047	0b11100000010R	0b00000000010R
2048	0b11100000011R	0b00000000011R

2.5.3.4 DCC Turnout Feedback

205 These events may be used to report actual turnout state. Some proprietary network protocols may already define messages that can appropriately be mapped to these events. The address mapping of these events to DCC turnout addresses is the same as defined in section 2.5.3.3 above.

2.5.3.5 DCC System Sensor Feedback

Some proprietary network protocols may already define messages for generic sensor feedback, which can then be appropriately mapped to these events. No assumptions are made about address translation.

2.5.3.6 Extended DCC Accessory Decoder

210 The Well-Known Event ID space for basic DCC accessory decoders is provided as a suggestion to manufactures on how to translate OpenLCB events to classic Extended DCC accessories. It is possible that there could be future mapping schemes required to implement a use case not covered by this scheme. This scheme is not designed to be mutually exclusive of other possible schemes, and its use is not required when mapping between OpenLCB events and classic Extended DCC accessories. The format of the DCC packet for an extended DCC Accessory Decoder is as follows (most significant bit first):

215

Byte 1	Byte 2	Byte 3	Byte 4 (Error Byte)
0b10A ₇ A ₆ A ₅ A ₄ A ₃ A ₂	0b0 $\bar{A}_{10}\bar{A}_9\bar{A}_8$ 0A ₁ A ₀ 1	0bXXXXXXXXXX	0bEEEEEEEE

Bits 5 down to 0 are the middle 6 bits of the address.	Bits 6 down to 4 are the 3 most significant bits of the address in 1's complement form. Bits 2 down to 1 are the 2 least significant bits of the address.	8-bit command.	
--	---	----------------	--

220 The Extended Accessory Decoder address format in most significant bit first format is: $0bA_{10}A_9A_8A_7A_6A_5A_4A_3A_2A_1A_0$, and it is expected that this is how the bits map to Bytes 6 and 7 of the Event Identifier. Though the $A_{10}A_9A_8$ address bits are in 1's complement form under the DCC standards, they are in standard non 1's complement form in OpenLCB format.

The reason that the 11-bit extended DCC accessory address falls into Bytes 6 and 7, inconsistent with the use of Bytes 7 and 8 in the basic DCC accessory address, is so that a single producer/consumer identified range message can be used without violating the 50% space utilization rule.

225 Example Event to DCC packet mappings:

User Address	OpenLCB Event ID	DCC Packet
1	01.01.02.00.01.00.04.XX	0b10000001 0b01110001 0bXXXXXXXXXX 0bEEEEEEEE
2	01.01.02.00.01.00.05.XX	0b10000001 0b01110011 0bXXXXXXXXXX 0bEEEEEEEE
3	01.01.02.00.01.00.06.XX	0b10000001 0b01110101 0bXXXXXXXXXX 0bEEEEEEEE
4	01.01.02.00.01.00.07.XX	0b10000001 0b01110111 0bXXXXXXXXXX 0bEEEEEEEE
2043	01.01.02.00.01.07.FE.XX	0b10111111 0b00000101 0bXXXXXXXXXX 0bEEEEEEEE
2044	01.01.02.00.01.07.FF.XX	0b10111111 0b00000111 0bXXXXXXXXXX 0bEEEEEEEE
2045	01.01.02.00.01.00.00.XX	0b10000000 0b01110001 0bXXXXXXXXXX 0bEEEEEEEE
2046	01.01.02.00.01.00.01.XX	0b10000000 0b01110011 0bXXXXXXXXXX 0bEEEEEEEE
2047	01.01.02.00.01.00.02.XX	0b10000000 0b01110101 0bXXXXXXXXXX 0bEEEEEEEE
2048	01.01.02.00.01.00.03.XX	0b10000000 0b01110111 0bXXXXXXXXXX 0bEEEEEEEE

2.5.4 Well-Known Other

230 The general properties of the Well-Known Other Event Identifiers are the same as those for Well-Known Event Identifiers as described in the section above. In general, this section contains events that have not originated as Well-Known, typically in a private Unique Identifier space, but at a later time have been promoted to Well-Known status.

2.5.4.1 Train Search Protocol

235 The Train Search Protocol was developed by Train Control Systems, Inc. (TCS), and placed in a Unique ID range that was assigned to them by the Unique Identifiers Standard. This is a perfectly valid example of how the extensibility of the OpenLCB protocols can be used for new and innovative use-cases. The protocol was shipped in products by TCS, and the documentation was published for the purpose of interoperability with devices and solutions by other manufacturers. When adopting this as an OpenLCB Standard, the allocation was purposefully not changed, to maintain compatibility with devices in the field.

240

Table of Contents

1	Introduction.....	1
2	Annotations to the Standard.....	1
2.1	Introduction.....	1
2.2	Intended Use.....	1
2.3	References and Context.....	2
2.4	Format.....	3
2.5	Allocation.....	3
2.5.1	Node ID Based.....	3
2.5.2	Well-Known Automatically-Routed.....	3
2.5.2.1	Emergency Off.....	3
2.5.2.2	Emergency Stop.....	4
2.5.2.3	Power Supply Brownout Detected.....	4
2.5.2.4	Other Well-Known Automatically Routed.....	5
2.5.3	Well-Known.....	5
2.5.3.1	Duplicate Node ID.....	5
2.5.3.2	MERG CBUS.....	5
2.5.3.3	Basic DCC Accessory Decoder.....	6
2.5.3.4	DCC Turnout Feedback.....	7
2.5.3.5	DCC System Sensor Feedback.....	7
2.5.3.6	Extended DCC Accessory Decoder.....	7
2.5.4	Well-Known Other.....	9
2.5.4.1	Train Search Protocol.....	9