NMRA Standard	
Bi-Directional Communication	
Jan 3, 2021	S-9.3.2 Draft

1 General

1.1 Introduction and Intended Use (Informative)

This standard describes the transmission of data from the decoder over the track to a receiver, ie in the opposite direction to the control protocols. The transmission method described here together with the protocol used is called RailCom.

"RailCom" is a German brand registered in the name of Lenz Elektronik for the class 9 "Electronic Controls" under the number 301 16 303 as well as for the classes 21, 23, 26, 36 and 38 "Electronic Controls for Model Railways" in USA it's registered under trademark No. 2,746,080. European Patent 1 380 326 B1 was repealed. RailCom is thus freely usable under consideration of the trademarks.

This specification applies exclusively to the use of RailCom within the DCC data format (protocol). The use of RailCom within other data formats is not permitted.

1.2 References

This standard should be interpreted in the context of the following NMRA Standards, Technical Notes, and Technical Information.

1.2.1 Normative

- S-9.1 Electrical Standards for Digital Command Control
- S-9.2 Communication Standards for Digital Command Control
- S-9.2.2 Configuration Variables for Digital Command Control

1.2.2 Informative

- TN-9.3.2 Bi-Directional Commnication, which provides commentary on Bi-Directional Communication
- RCN-217 RailCom DCC Feedback Protocol, with which S-9.3.2 is intended to be in harmony[i]

1.3 Terminology

Term	Definition
RailCom®	Lenz trademark and common marketing name for DCC bi-directional communication.
Bi-Directional	Communication protocol for DCC decoder transmission. Commonly known by its Lenz trademark and common marketing name of "RailCom®".
Preamble	Series of DCC '1' bits making up the start of a DCC packet.
Cutout	Period of time where the two DCC phases are shorted together during which a Bi-Directional decoder transmission may occur.

1.4 Requirements

In order to comply with this standard, all technical values and protocols defined in this standard must be complied with. Tables 5 and 6 define a minimum set of messages a decoder should support.

1.5 Overview

A DCC data packet is a defined sequence of bits described as a track signal in [RCN210]. Bytes are bit groups of eight bits each. Each bit in the byte has a significance dependent on its position, the first transmitted bit, the leftmost bit in the representation has the highest significance and is called the "most significant bit" (MSB). The bits of a byte are numbered starting from the left beginning with 7, falling right to 0. The last bit transmitted in the representation is called "least significant bit" (LSB).

The following characters are used to indicate the meaning of a bit:

0	Bit value 0
1	Bit value 1
A	Address bit
D	data bit
P	Location information (position)
R	Direction bit
S	Sequence number
T	Type of location information
X	Subindex

The characters used in the DCC commands are not listed here.

The bit combinations for DCC represented in the box in this standard are purely informative and have no normative character. Only the specified RCNs apply here.

The commands from the Command Station to the decoder (►) are noted in each case without the addressing data. The addressing is done according to the DCC standard.

(◄) depicts the sent RailCom data.

Unless otherwise stated, values always refer to an 8-bit field. Binary values are marked according to section 1.3. Hexadecimal values are indicated by a prefix 0x.

2 Physical Layer

The flow of information in the DCC system normally takes place from the central booster (usually a command station) over the track to the decoders. For the decoder to transmit to the Command station using RailCom it is necessary to interrupt the DCC signal. This is done by the boosters, which generate a „RailCom cutout“ at the end of each DCC packet. The cutout created by disconnecting the two track lines from the power supply and short them together. This devices that can create said cutout are "cutout devices". A cutout device could also be run as a separate unit outside the booster. The actual data transmission takes place by means of a current loop. The decoder must store enough power to source the needed current during the cutout window. Figure 1 shows the arrangement of a booster, detector and decoder during the RailCom cutout. Detectors can detect the RailCom signal and display or retransmist the data as needed, for example a detector may display the address of the decoder in its block. Detectors can be standalone devices or part of a booster, command station, etc.

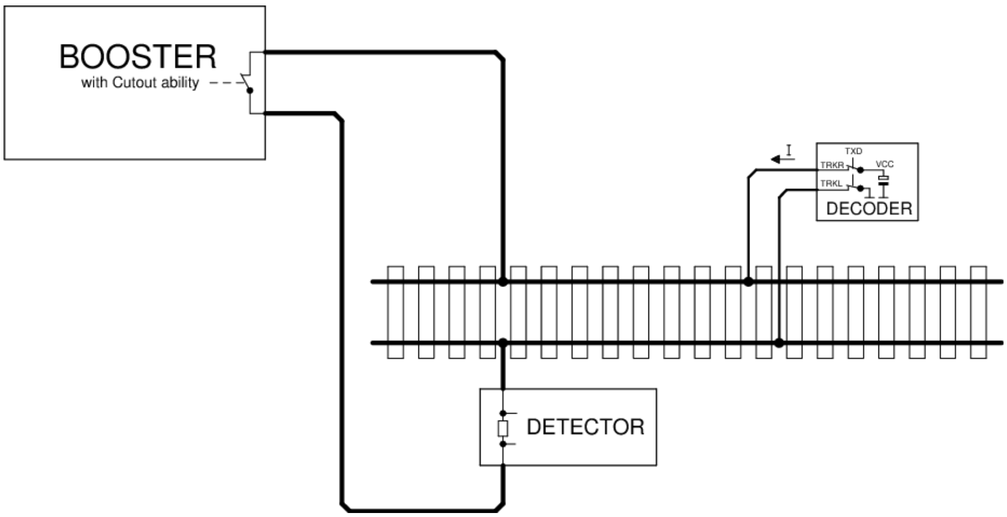

Figure 1: RailCom schematic diagram

The voltage drop across the cutout device must not exceed 10mV at a maximum of 34mA during the cutout.

2.1 RailCom Transmitter in the Decoder

In order to transmit a '0', the decoder must deliver a current I of 30 +4 or -6 mA, with a voltage drop on the track of up to 2.2 V. If high-current RailCom is enabled in CV28, the decoder Current must be within +8mA to -12 mA of 60mA, a voltage drop on the track of up to 2.2 V is allowed. For a '1' the current I may not exceed ± 0.1 mA. The power source of the decoder must be protected against unexpected external voltage on the track during cutout. Figure 2 shows a possible decoder hardware implementation.

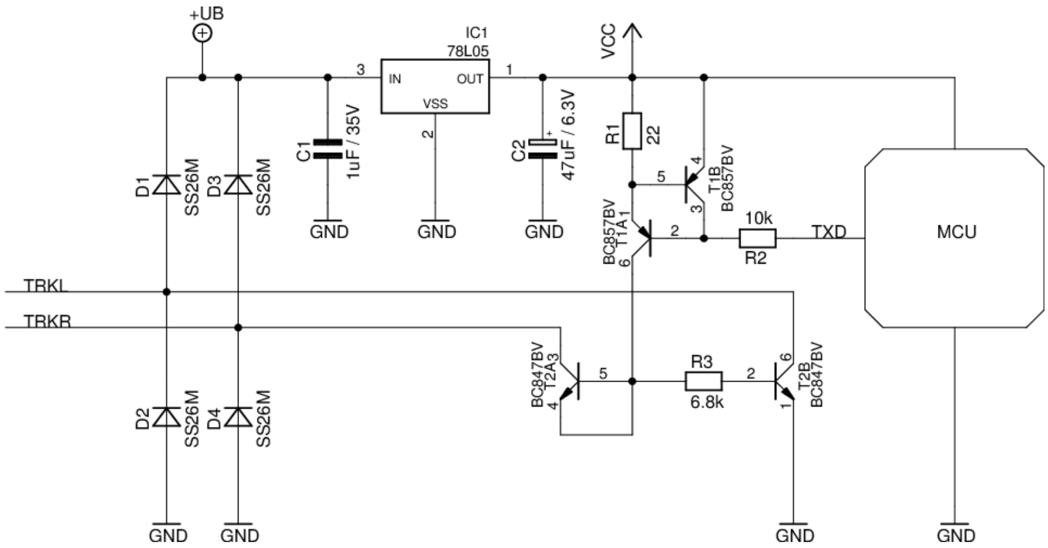

Figure 2: RailCom Decoder Hardware

Explanation of the circuit diagram:

The RailCom circuitry consists only of the resistors R1 to R3 and the transistors T1A to T2B. T1A and T1B form a current source, T2A is diode-connected and protects the current source from positive voltages higher than Vcc.

All other parts of the circuit are normally necessary hardware of a decoder. Note the extremely low hardware costs for the RailCom transmitter.

2.2 RailCom Detector Hardware

A Detector must interpret a current greater than 10mA as '0' during the middle 50% bit time, a current less than 6mA during the middle 50% bit time as '1'. The voltage drop across the detector should not exceed 200mV at a maximum of 34mA during the cutout. Figure 3 shows a possible hardware circuit:

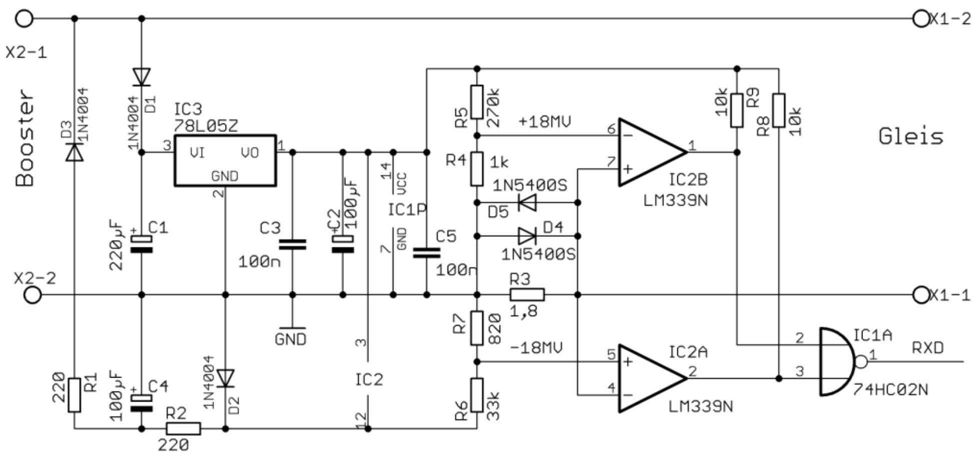

Figure 3: Example of a simple RailCom Detector Hardware

A maximum of two detectors (including the global detector) may be used in series, whereby the local detector should contain a connection for external evaluation of a track occupancy. If this is not the case, externally used occupancy detectors must be specified for RailCom. Explanation:

Comment [JWF1]: I do not know what this is trying to say.

The transmitters and detector circuits were tested on large club facilities up to a distance of 100 meters. This distance was easily bridged. It is not possible to use 5 ohm loads isolated from the track by bridge rectifiers, which are parallel to the measuring resistor of the detector.

The value of 5 ohms corresponds to a current of 3A at a track voltage of 15V.

Incandescent lamps (PTC thermistors) must always be operated via a fast bridge rectifier (<500 ns)..

Comment [JWF2]: I think this comment needs more detail, why was this test done or in what sistuations would those loads be present on a layout?

2.3 Timing

Up to 8 bytes of data can be transferred in one cutout. Each transmitted byte begins with a start bit ('0') followed by the 8 data bits (least significant bit first) and ends with a stop bit ('1'). The transmission rate is 250 kbit / s ± 2%. The rise time (10% → 90%) and fall time (90% → 10%) must not exceed 0.5 µs.

The RailCom Cutout is divided into two channels. In channel 1 two bytes, in channel 2 up to six bytes can be transmitted. Figure 4 shows the timing diagram. All times are related to the zero crossing of the last edge of the packet end bit.

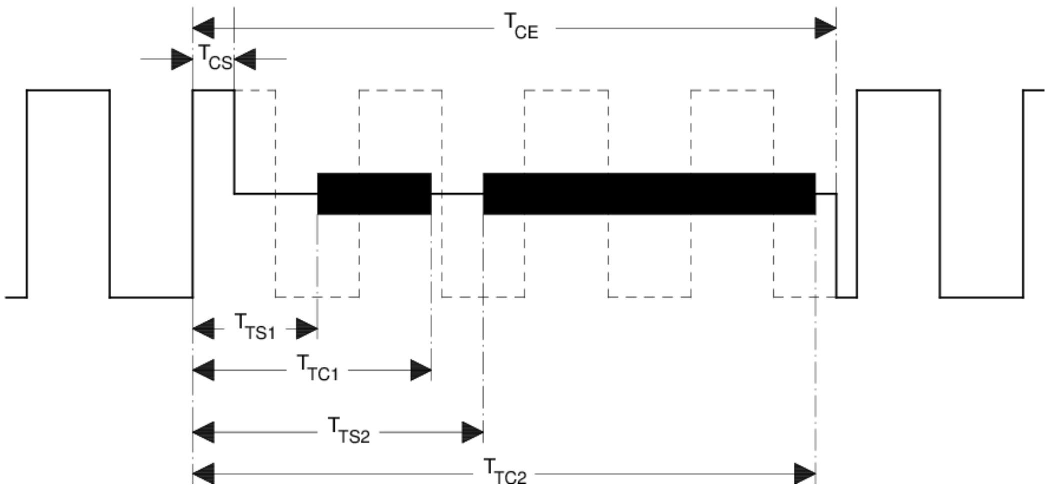

Figure 4: RailCom Timing

Table 1: Timing Parameters

Parameter	**Name**	**Min**	**Max**
Cutout Start	T_{CS}	26µs	32µs
Cutout End	T_{CE}	454µs	488µs
Channel 1 Start	T_{TS1}	80µs	
Channel 1 End	T_{TC1}		177µs
Channel 2 Start	T_{TS2}	193µs	
Channel 2 End	T_{TC2}		454µs

Comment:

The above figure shows the RailCom timing with "1" bits of 2 * 58µs (nominal value of the DCC "1" bit). For shorter "1" bits, it is possible for the cutout to reach the 5th "1" bit. However, this is not a problem since a Command Station must send at least 4 + 12 = 16 sync bits according to [RCN211] (packet end bit of the previous DCC packetbitsbits not counting the), so the decoder has enough sync(at least 11 syncmust be sent) for a recognition of the package by the decoder required) sees.

A cutout time of approx. 450µs must not affect the function of a decoder that does not support RailCom, as current interruptions up to 20ms have been detected on a real model railway layouts, ie a decoder should be able to process at least one power interruption on this scale.

Comment [JWF3]: This line does not make sense

2.4 4/8 Encoding

Transmission integrity is ensured via a 4/8 encoding scheme, ie each transmitted byte contains 4 '1' and 4 '0' bits. If this ratio is violated, there is a transmission error.

There are 70 different bit combinations within a byte that have this ratio 4: 4. Of these, 64 are used for the transmission of 6 useful bits, of the remaining 6 one is used for short special messages, ACK. The remaining five combinations are reservered for future use.

It is possible to transfer up to 12 bits of user data in channel 1 and up to 36 bits in channel 2.

The possible 4/8 encodings are:

Table 2: 4/8 Encoding

Value	4/8 Code	Value	4/8 Code	Value	4/8 Code	Value	4/8 Code
0x00	10101100	0x10	10110010	0x20	01010110	0x30	11000110
0x01	10101010	0x11	10110100	0x21	01001110	0x31	11001100
0x02	10101001	0x12	10111000	0x22	01001101	0x32	01111000
0x03	10100101	0x13	01110100	0x23	01001011	0x33	00010111
0x04	10100011	0x14	01110010	0x24	01000111	0x34	00011011
0x05	10100110	0x15	01101100	0x25	01110001	0x35	00011101
0x06	10011100	0x16	01101010	0x26	11101000	0x36	00011110
0x07	10011010	0x17	01101001	0x27	11100100	0x37	00101110
0x08	10011001	0x18	01100101	0x28	11100010	0x38	00110110
0x09	10010101	0x19	01100011	0x29	11010001	0x39	00111010
0x0A	10010011	0x1A	01100110	0x2A	11001001	0x3A	00100111
0x0B	10010110	0x1B	01011100	0x2B	11000101	0x3B	00101011
0x0C	10001110	0x1C	01011010	0x2C	11011000	0x3C	00101101
0x0D	10001101	0x1D	01011001	0x2D	11010100	0x3D	00110101
0x0E	10001011	0x1E	01010101	0x2E	11010010	0x3E	00111001
0x0F	10110001	0x1F	01010011	0x2F	11001010	0x3F	00110011
ACK	00001111	Command is understood and will be executed.					
ACK	11110000	Command is understood and will be executed.					
Reserved	11100001						
Reserved	11000011						
Reserved	10000111						
NACK	00111100	Command or CV is not supported [optional].					

Detailed explanation of NACK:
This optional response states that a command is not supported. The detection of an unsupported command is restricted to the In POM (regardless of the response to unimplemented instructions), a NACK may also be sent to identify a non-existent CV. To keep from saying that PoM is not supported a NACK must not be the first response when responding to a POM command, ie when

Comment [JF4]: I do not understand this comment

Comment [JF5R4]: 1Input from VITA Szabolcs (Viessmann)
Meanwhile I translated the first part of NACK, where the misunderstanding seems to have come from a mistranslation: "NACK: This optional response can indicate that a command is not supported. Here the detection of a not supported command applies exclusively to the first byte of the command. In case of POM (independently from the answer on not supported commands) it's allowed to send a NACK to indicate a not implemented CV. "

accessing a non-existent CV, an ACK and then a NACK mst be sent. ACK and NACK can be sent together in the same channel 2 cutout.

3 Packet Layer

This chapter describes the structure of RailCom Datagram

RailCom packets (hereinafter referred to as datagrams) have a length of 6, 12, 18, 24 or 36 payload bits, only a 12-bit datagram can be transmitted. In channel 2, any combination of datagrams with a maximum total length of 36 bits can be transmitted.

Optionally, the data channel can be filled with ACK to 36 bits.

Datagrams (except ACK) begin with a 4-bit identifier followed by 8, 14, 20, or 32-bit payload data, as follows:

Table 3: Datagram Structure

Datagram	Bytes
12-bit	ID[3..0]D[7..6] + D[5..0]
18-bit	ID[3..0]D[13..12] + D[11..6] + D[5..0]
24-bit	ID[3..0]D[19..18] + D[17..12] + D[11..6] + D[5..0]
36-bit	ID[3..0]D[31..30] + D[29..24] + D[23..18] + D[17..12] + D[11..6] + D[5..0]

The length of the datagram is determined by the identifier. The identifiers are defined below

Mobile Decoder (vehicle decoder) and stationary decoder (accessory decoder) have different feedback requirements. Accordingly, the channels are used differently for both types of decoders. The meaning of the datagrams therefore depends on the address of the preceding DCC packet. There are also system requirements that all decoders have to. For this purpose, the DCC address 255 is set as "system address". Accordingly, the following RailCom command types MOB (mobile) and STAT (stationary) are distinguished on the basis of the DCC address ranges as defined in [RCN211] Section 3:

Table 4: Command and system

DCC Address		RailCom Command Type
1st Byte	2nd Byte	
1-127		MOB (mobile) Short address
128-191		STAT (accessory)
192-231	ADR low	MOB (mobile) Long Address The long address 3 is also used as a programmingaddress.[1]
255		MOB / STAT 255 is the address of the idle packet (0xFF 0x00 0xFF) The 1st data byte other than 0 indicates the use as the system address. In any case, the packet then consists of more than 3 bytes.

[1] According to the RailCommunity proposal from September 7, 2018.

Decoders are not allowed to send feedback to other addresses or service mode packets.

Comment [JF6]: This line needs more explanation, possible specifiy what channels it can respond to

3.1 RailCom Command MOB Decoder

Channel 1 is used for locating mobile decoders on the layout (see app: adr). To do this, they must send their DCC address after every DCC packet sent to a mobile decoder, which is then received by local detectors on the layout. Mobile decoders must not sent their address in response to programming mode packets (from the time the decoder recognizes the programming mode packets)

Channel 2 may only be used by the addressed decoder and is used to transmit decoder information. An addressed decoder must always send feedback in channel 2 (ACK if necessary) to confirm the error-free reception of the DCC packet.

A response in channel 2 indicates that the decoder received the command without error, but not that the decoder accepts and executes the command.

The following identifiers (datagrams) are defined for mobile decoders:

Table 5: Command type MOB Identifier (Datagrams)

ID	Channel 1	Channel 2	Comment	Command Related
0		app: pom	mandatory, 1 byte	yes
1	app: addr_high	app: addr_high	mandatory in channel 1	yes
2	app: addr_low	app: addr_low	mandatory in channel 1	yes
3		app: ext	optional	yes
4			reserved	
5			reserved	
6			reserved	
7		app: dyn	optional	no
8		app: xpom app: CV-auto	optional optional	yes no
9		app: xpom	optional	yes
10		app: xpom	optional	yes
11		app: xpom	optional	yes
12		app: Test Feature ID	optional	n/a
13		app: block	optional	yes
14			reserved	
15			reserved	

mandatory means full implementation

optional	either full implementation or partial implementation with the conditions under 4.1 related
command related	with "yes" the messages in channel 2 may only be assigned to a corresponding command consequence; with "no" after any command to the decoder. These non-command responses are also considered ACK.

Comment [JF7]: What is the 4.1 its referring to?

Comment [JF8]: I do not understand the „command related" explanation.

> Note:
>
> Older decoders have used different identifiers during a test phase, which are now marked as "reserved". Newer decoders must have entered the RailCom version number in a special CV (see section "RailCom CVs"). This can be used to determine what decoder supports what identifiers. Older decoders without version numbers should be updated.

Comment [JF9]: Is there an approximate date we could add into this note?

3.2 RailCom command type STAT

Comment [JF10]: Will review this when stationary decoder spec is completed

The RailCom specification for accessory decoders is not yet completed. All information is to be regarded as provisional.

Accessory decoders use channel 1 to report service request requests (see app: srq). To do this, you can send your identity (12-bit address) after each DCC packet directed to an accessory decoder (12-bit value without identifier !!) (not for addressing via decoder ID). If several decoders register at the same time, a search must be started.

Channel 2 may only be used by the addressed decoder and is used to transmit decoder information. An addressed decoder must always send a feedback in channel 2 (ACK if necessary) to confirm the error-free reception of the DCC packet.

A response in channel 2 indicates that the decoder received the command without error, but not that the decoder accepts and executes the command.

The following identifiers (datagrams) are defined for accessory decoders:

Table 6: Command type STAT Identifier (datagrams)

ID	Channel 1	Channel 2	Comment	Command Related
	app: srq			
0		app: pom	optional	no
1			mandatory in channel 1	
2			mandatory in channel 1	
3			optional	
4		app: stat1	mandatory	no
5		app: zeit	optional	no
6		app: fehler	mandatory	no
7		app: dyn	optional	yes
8		app: xpom	optional	no
9		app: xpom	optional	no
10		app: xpom	optional	no

11		app: xpom	optional	no
12		app: Test Feature ID	optional	n/a
13		app: block	optional	yes
14			reserved	
15			reserved	

mandatory means full implementation

optional either full implementation or partial implementation with the conditions under 4.1 related

Comment [JF11]: What is the 4.1 its referring to?

command related with "yes" the messages in channel 2 may only be assigned to a corresponding command consequence; with "no" after any command to the decoder. These non-command responses are also considered ACK.

Comment [JF12]: I do not understand the „command related" explanation.

4 RailCom CVs and DCC Commands

4.1 CV28 RailCom Configuration

Table 7: Railcom Configuration

Bit	Function
0	Channel 1 for Address-Broadcast (Section **Error! Reference source not found.**), 1 = enabled, 0 = disabled
1	Channel 2 for Data and Acknowledge, 1 = enabled, 0 = disabled
2	Disable channel 1 automatically (Section 5.2.1) 1 = enabled, 0 = disabled
3	Reserved
4	Programming address 0003 (long address 3)[2], 1 = enabled, 0 = disabled
5	Reserved
6	Enable high-current RailCom (section 2.2) 1 = enabled, 0 = disabled
7	Release RailComPlus, 1 = enabled, 0 = disabled

4.2 CV29

Use According to [**Error! Reference source not found.**].

4.3 CV31, CV32

Use According to [**Error! Reference source not found.**].

4.4 RailCom Block

CV31 = 0 and CV32 = 255 address a block of 256 CVs, which are used for RailCom applications according to the following table:

[2] According to the RailCommunity proposal from September 7, 2018.

Table 8: RailCom Block CV Descriptions

Byte	Purpose	Mandatory/ Optional	Read/ Write
0	Manufacturer ID (from NMRA [S-9.2.2 Appendix A])	m	R
4-7	Product ID (Manufacturer-specific product identifier to distinguish the individual products Little Endian)	o	R
8-11	MUN (Manufacturer Unique Number, Little Endian) 4 byte serial number unique over all devices of one manufacturer.	o	R
12-15	Production(number of seconds since 1.1.2000, Little Endian, Unsigned)	o	R
16-63	Manufacturer-assignment possible	o	R/W
64-127	Dynamic variables according to APP DYN, Section 5.4, Table 11 72 - Container 1 ... 83 - Container 12	o	R/W
128	RailCom Version number "Major revision", binary	m	R
129	RailCom Version number "Minor revision", binary	m	R
130	Feature number for test feature ID (see 5.5)	o	R
131	Reserved		
132	specific consumption, container 1	o	R/W
133	specific consumption, container 2	o	R/W
134	specific consumption, container 3	o	R/W
135	specific consumption, container 4	o	R/W
136	specific consumption, container 5	o	R/W
137	specific consumption, container 6	o	R/W
138	specific consumption, container 7	o	R/W
139	specific consumption, container 8	o	R/W
140	specific consumption, container 9	o	R/W
141	specific consumption, container 10	o	R/W
142	specific consumption, container 11	o	R/W
143	specific consumption, container 12	o	R/W
144	Level of all containers (0 ... 255), a write command to this CV sets the contents of all containers to the specified value.	o	W
145	speedometer scaling in the control panel, value times 2 = maximum speed in km / h or mph	o	R/W

146-255	Reserved		

m = Mandatory, o = Optional, R = Read, W = Write

4.5 DCC Commands

RailCom's enhanced functionality also requires additional DCC commands. These include:

4.5.1 Additional Function

RailCom offers additional functions such as a search function, track search, etc. (see also "Applications (APP) layer for mobile decoder"). Short form binary state control commands are use according to [RCN212] section 2.3.5.

Command: Binary state control as RailCom command	
►	Short binary state control command according to ([RCN212]) Section 2.3.5 1101-1101 DLLL-LLLL

These commands are referred to as XF1 (extra function) through XF127 in this standard.

Of these commands, the first 28 are reserved for special applications such as RailCom in [RCN212]. For RailCom, the first 15 functions are defined.

Table 9: Function Numbers

XF	Function
1	Requesting the location information coresponding to section 5.3.1.
2	Track Search
3-15	Reserved

4.5.2 Advanced Programming

There is a method in [RCN214] Section 3 to write „CV Pairs“ simltanously.

`KKKK` = `0100` = Write CV17 (first data byte) and CV18 (second data byte) (extended address) simultaneously, and set bit 5 in CV29.

Feedback occurs at the old address via two consecutive datagrams with ID0, first the first data byte, then the second data byte. Both datagrams must be sent in the same channel 2, ie 12-bit datagram + 12-bit datagram.

Command: POM	
►	commandWrite CV access from CV17 and CV18 according to [RCN214] section 3 1111-0100
◄	Channel 2: 0000 (ID0) DDDD-DDDD + 0000 (ID0) DDDD-DDDD with D = CV data, first CV17 , then CV18

KKKK = 0101 = write CV31 (first byte of data) and CV32 (second byte of data) (extended range, high byte (31), and low byte (32) pointer value).

Feedback occurs via two consecutive datagrams with ID0, first the first data byte, then the second data byte. Both datagrams must be sent in the same channel 2, ie 12-bit datagram + 12-bit datagram.

Command: POM	
►	commandWrite CV access from CV31 and CV32 according to [RCN214] section 3 1111-0101
◄	Channel 2: 0000 (ID0) DDDD-DDDD + 0000 (ID0) DDDD-DDDD with D = CV data, first CV31 , then CV32decoderpanel

4.5.3 NOP for Accessory Decoder

Accessory decoders will report for a switching command if they want to communicate something to the command station (SRQ).

Normally, switching commands are only sent sporadically, so accessory decoders may only be noticeable just as rarely. Therefore, in Section 2.5 of [RCN213], the NOP command is defined to be sent on a regular basis, but initially does nothing to allow SRQ except accessory decoders.

This must be recognized as invalid by all non-RailCom-compatible simple and extended accessory decoders and thus ignored. On the one hand, it allows accessory decoders to issue an SRQ in the subsequent blanking interval, and on the other hand, it allows for the event that several decoders register simultaneously with an SRQ, as well as the search for the decoders involved.

This is achieved by transmitting an accessory decoder address with the NOP. Thereupon only the decoders whose address is smaller or the same as the ones contained in the NOP are notified. As a result, the decoder with the respective lower address can be found and operated in the case of messages from several decoders by means of successive approximation. The search is repeated until no more decoder announces.

Command: NOP	
►	commandNOP for accessory decoder according to [RCN213] Section 2.5 10AA-AAAA 0AAA-1AATpanel

For regular polling of all accessory decoders, the controlsends a NOP with the highest possible address, so that all decoders are addressed.

As long as only one decoder with an SRQ reports this, it can remain with the highest address in this NOP. Only when several decoders report at the same time, the central office must start a search by suitably varying the address contained in the NOP.

A RailCom-capable center must send a NOP command at certain intervals to query the accessory decoder. The time interval between two NOPs is a compromise of bandwidth limitation on the DCC signal and response time on SRQ messages. An interval of about 0.5 seconds is recommended.

When searching for multiple messages, the NOPs used for the search must of course be sent as quickly as possible in succession.

5 Applications Layer for Mobile Decoder

The following sections describe the commands for the RailCom functionality.

5.1 POM (ID 0)

POM means "Programming On the Main". It is used to read and write configuration variables in the operating mode according to [RCN214] section 2. These commands are respoded to in channel 2 by sending a 12 bit datagram consisting of ID0 = 0000 and 8 data bits. The data bits contain the value of the CV.

Command: POM	
►	Command Access to Bytes or Bits According to [RCN214] Section 2 1110-KKVV VVVV-VVVV DDDD-DDDD 1110-10VV VVVV-VVVV 111K-DBBB
◄	Channel 2: 0000 (ID0) DDDD-DDDD with D = CV-Daten

5.1.1 Byte Read

Command: POM Byte Read	
►	Command byte according to [RCN214] Section 2. 1110-01xx
◄	Channel 2 (12Bit): 0000 (ID0) DDDD-DDDD mit D = CV-Daten.

Comment [JF13]: This is missing some bytes?

The associated response datagram (ID0) does not have to be sent in the cutout immediatly after the command packet but must be sent within 0.5 seconds of the command packet. A Command Station must therefore ensure that the decoder is addressed again and no other programming command is sent to the same address (repeated the command is permitted).

When the read operation is completed, the decoder sends the result to the associated read command. If the decoder does not return the data within 0.5s then the read is considered failed.

5.1.2 Byte Write

Command: POM Byte Write	
►	commandbyte according to [RCN214] Section 2. 1110-11xx
◄	Channel 2 (12 bit): 0000 (ID0) DDDD-DDDD Here D is the value returned = CV data **after** POM Operation is present in the CV.

When writing, the decoder should respond as follows: With ACK, as long as the new value is not written with the new value, if the new value is written successfully, first with ACK and then NACK if the CV is not supported or another one Value if the written value can not be accepted. Instead of an ACK, another non-command message can be used.

Comment [JF14]: I need help determining what the intent of this section is.

If the decoder does not return a value within 0.5s then the write is considered failed. For read only CV's the shall decoder return the current value of the CV.

5.1.3 Bit Write

Command: POM Bit Write

►	Command bit according to [RCN214] Section 2 1110-10xx
◄	Channel 2 (12 bit): 0000 (ID0) DDDD DDDD Here, with D = CV data, the value returned **after** the POM operation in the CV is returned.

The „Bit Read“ command is not implemented because RailCom responds with the whole byte, making a „Bit Read“ not necessary

5.2 ADR (IDs 1 & 2)

With the help of fixed detectors, RailCom can be used for localization.

Vehicle decoders use channel 1 as a broadcast channel for their own address. In the cutout after every DCC packet to a vehicle decoder, they send their active address (base, extended or consist address). For this, the following 12-bit datagrams with ID1 and ID2 are defined:

Table 10: Address Assignment

ADR1 (ID1)	ADR2 (ID2)	Adresse
0 0 0 0 0 0 0 0	0 A6 A5 A4 A3 A2 A1 A0	Base-Address (CV1)
0 1 1 0 0 0 0 0	R A6 A5 A4 A3 A2 A1 A0	Consit Address (CV19)
1 0 A13 A12 A11 A10 A9 A8	A7 A6 A5 A4 A3 A2 A1 A0	Extended Address (CV17 + CV18)

A decoder must be on the ADR1 and ADR2 alternately. The "active address" is the one under which the decoder receives its drive commands.

Command: ADR	
►	commandOperation command to decoder according to [RCN212] Section 2
◄	Channel 1: 0001 (ID1) ADR1 or 0010 (ID2) ADR2

Possible application:

Local Detectors on the system can receive information from decoders and pass that along to a global detector (Command Station, computer, etc.). In this way, the Command Station can learn which decoder is on a particular track section. Figure 5 shows this setup.

This method is suitable for train control.

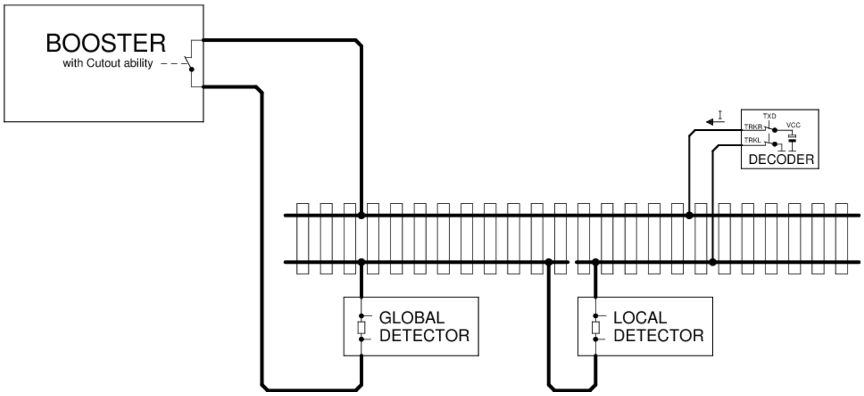

Figure 5: Localized Detector

Decoder locating a decoder in the above pattern can only work if the decoder is on the track section monitored by the Local Detector. However, this is problematic in when multiple locomotives are in a consist. For this it is recommended that only the leading locomotive sends the ADR datagrams on channel 1, while on the following locos this function is deactivated via CV28. This can be done when assembling the consist via POM.

5.2.1 Dynamic Channel 1 Usage

In this case, the sending of the address in channel 1 is automatically switched off in order to reduce collisions. The decoder transmists in channel 1:

- After a restart
- After a change of address
- If it has not been addressed for more than 5s.

If bit 2 is set in CV28, the transmission in channel 1 will be switched off automatically after receiving eight packets addressed to it. This method will allow for detection of a new locomotive and, once addressed by the system, that locomotive no longer interfere with the detection of other locomotives.

5.2.2 Track Search

This is a special application to identify a decoder on the layout. For this purpose, the decoder is temporarily separated from the track power. After the decoder receives power again, it responds for a maximum of 30 seconds to the command "XF2 off" to broadcast address 0 with its address in Channel 2. During this time, the user must trigger this command via the control panel.

Command: EXT	
►	short binary state control command "XF2 off" according to [RCN212] section 2.3.5 to the broadcast address 0: 1101-1101 0000-0010
◄	Channel 2 from the decoder has three 12 bit datagrams 0001 (ID1) DDDD-DDDD : adr_high according to Table 10 0010 (ID2) DDDD-DDDD : adr_low according to Table 10 0000 (ID0) DDDD-DDDD : Time in seconds since the decoder has power again.

Comment [JF15]: I don't fully understand how this feature is to be used or what its purpose is. Is all of the layout powered down? Is just a one block powered down to find out whats in it? What is the command the user must input from the control panel? Is the contol panel a throttle or a switchboard?

Comment [JF16]: How is this data used?

The decoder should not respond to every command "XF2 off" to the broadcast address 0, so that collisions in channel 2 may be reduced. These occur when another vehicle unintentionally had no track power for a short time and therefore might respond. Over time since the decoder has tension again, the desired vehicle should be identifiable.

Decoders should not respond to the command „XF2 off" on broadcast address 0 unless the decoder is powering up, this is to avoid collisions.

Comment [JF17]: This explanation is suitable? But what amount of time determines if the decoder is „just powering up"?

5.3 EXT (ID 3)

This feedback is used to transmit location information. In this way, the location of the locomotive can be determined and location specific operartions can be triggered such as a filling of inventories (coal, fuel, etc).

5.3.1 Sending of the location Information

The location information may be sent by the decoder or the detector, depending on where the information is.

Case 1: Location information is present in the decoder (eg by infrared transmission).

Command: EXT	
►	short binary state control command "XF1 off" according to [RCN212] section 2.3.5: 1101-1101 0000-0001
◄	Channel 2 of the decoder: 0011 (ID3) 00 TTTT-PPPP-PPPP With: TTTT = 0000 - 0111 : Location information (position):0PPP TTTT = 1000 - 1111 : reserved 0PPP-PPPP-PPPP : 11 bit location address (position)

If the location information is present in the decoder, it can also be transmitted spontaneously in channel 2 with the ID DYN. See section 5.4.

Case 2: Location information is available in the Detector.

Command: EXT		
►	short binary state control command "XF1 off" according to [RCN212] section 2.3.5: 1101-1101 0000-0001	
◄	Channel 2 from Decoder: 0011 (ID3) 01	Channel 2 from Detector: TTTT-PPPP-PPPP information TTTT = 1000 : reserved TTTT = 1001 : reserved TTTT = 1 010 : Diesel fuel TTTT = 1011 : coal bunker TTTT = 1100 : Water tower TTTT = 1101 : Sanding station

		TTTT = 1110 : Charging station (battery) TTTT = 1111 : General filling station PPPP-PPPP : 8 bit location address (position)

The Detector works as in Figure 5, but is supplemented by a RailCom transmitter. Since the location information is sent back across the track, a bus system is not needed.

5.3.2 Fill

The filling of a container or all containers takes place with the command "Byte Write" from Section 5.1.2 to the corresponding CVs in the RailCom block (CV31 = 0 & CV32 = 255).

5.4 DYN (ID 7)

This feedback is used to transmit dynamic information from vehicle decoders. Dynamic information refers to dynamic variables (DVs) that change during operation (eg speed, tank contents, etc.). RailCom CVs 64-127 (see Section 4.2.4, Table 8) correspond to DVs that can be changed by programming.

Command: DYN	
►	Operating command to decoder address according to [RCN212] section 2
◄	Channel 2 18Bit [+ 18Bit]: 0111 (ID7) DDDD-DDDD-XXXX-XX [0111 (ID7) DDDD-DDDD-XXXX-XX] with D = DV value and X = subindex for identification of the DV

The transmission of dynamic variables (DV) (eg speed, container, ...) takes place in an 18-bit datagram (ID7), which contains the 8-bit value of the DV (D) as well as a 6-bit subindex (X) of 64 possible DVs selected. The meaning of the DV is determined by the subindex.

2 arbitrary DVs can be transmitted in a feedback frame. Which DVs a decoder sends when he determines himself.

Table 11: Dynamic Informatin Vehicle Decoder

X	Meaning
0	Real speed part 1[3] Values between 0 and 255 are sent back from the decoder. 255 is maximum speed. The figures are in km / h (or mph). At higher speeds than 255 the difference is stored in part 2.
1	Real speed, part 2[3]
2	reserved
3	reserved
4	reserved
5	Flag Register, content yet to be determined
6	Input register, assignment to be determined
7	Reception Statistics: The vehicle decoder keeps track of all received DCC packets and transmits the number of bad packets / total in%. (Value range 0-100).
8	Content of container 1 in% (value range 0-100)
9	Contents of container 2 in% (value range 0-100)
10	Content of container 3 in% (range 0-100)
11	Contents of the container 4 in% (value range 0-100)
12	Contents of the container 5 in% (value range 0-100)
13	Contents of the container 6 in% (value range 0-100)
14	Contents of the container 7 in% (value range 0-100)
15	Contents of the container 8 in% (value range 0-100)
16	Contents of the container 9 in% (value range 0-100)
17	Content of the container 10 in% (value range 0-100)
18	Contents of the container 11 in% (value range 0-100)
19	Contents of the container 12 in% (value range 0-100)
20	Datagram 1: Local address least significant 8 bits Datagram 2: Local address Most significant 3 bits in bits 0 to 2 , bits 3 to 7 are reserved to 0.
21	Status and alarm messages
22	Trip odometer, exact definition yet to be determined

Comment [SB18]: This footnote may be worded better. From JF: if the decoder is calculating the speed I assume it would know its scale and internally be able to do the calculation as part of its „actual speed on the system"
The unit of speed shouldn't be limited to km/h, perhaps it should be configurable in a CV somewhere or set by the manufacturer.

[3] For example, the decoder calculates the actual speed on the system. This is multiplied by the scale, adjustable in the decoder is (for example, the value of H0, 87). Result is the real speed. It is transmitted as km / h. Up to the value 255, the transfer takes place exclusively in ID 7 DYN0. If the speed is higher, only the difference (ie calculated speed in km / h minus 256) is transferred to ID 7 DYN 1. ID 7 DYN 0 will not be transmitted because it no longer contains relevant information.

23	Operating time, exact definition yet to be determined
24-63	reserved

For the location address DV 20, both address parts are to be transmitted separately in 18-bit datagrams within a single channel, transmitting least significant 8 bit datagram first. For addresses <256, a single datagram is sufficient.

The status and alarm messages for DV22 are still to be determined.

The most significant bit determines whether the alarm is active or not.

Comment [JF19]: A value of 1 signify's the alarm is active?

5.5 XPOM (ID 8 to ID 11)

POM means "Programming On the Main", XPOM is an extended format for writing and reading up to four CVs in operating mode according to [RCN214] Section 5, compared to the POM defined in section 5.1. These instructions are used in channel 2 with a 36 bit datagram with the ID 8 = 1000 to ID 11 = 1011 and 32 data bits. The data bits contain the value of four consecutive CVs.

Command: XPOM	
►	Accessing Bytes or Bits According to [RCN214] Section 5 1110-KKSS VVVV-VVVV VVVV-VVVV VVVV-VVVV {DDDD-DDDD {DDDD-DDDD {DDDD-DDDD {DDDD-DDDD}}}
◄	Channel 2 (36-bit): 10SS (ID8 - 11) DDDD-DDDD DDDD-DDDD DDDD-DDDD DDDD-DDDD with D = CV data

Regardless of the number of bytes written and even if only one bit is written, the values of four CVs are always reported back.

The assignment between command and response is carried out by means of sequence number SS and datagram ID as follows:

`SS = 00` - ID 8 `1000`

`SS = 01` - ID 9 `1001`

`SS = 10` - ID 10 `1010`

`SS = 11` - ID 11 `1011`

Comment [JF20]: Put this in a proper table?

The response datagram associated with the XPOM command (ID8 ... ID11) does not have to be sent in the same packet frame.

The decoder must implement a queue for 4 XPOM commands and execute them in sequence. Repeatedly sent XPOM commands with the same sequence number are queued only once. The associated response datagram completes the operation and releases the corresponding entry in the queue. When writing via XPOM, the answer will be given after the actual writing process is completed. In this way, the control panel can synchronize to the writing speed of the decoder. If the response datagram is missing, the last command with the same sequence number must be sent repeatedly. When writing read-only CVs, the decoder returns the current value of the CVs. Whether a write access has been accepted must be determined by the command station by comparing the written value and received value.

The decoder must support fast, block-by-block reading, ie, with consecutive XPOM read commands

the answer of the 1st XPOM read command at the latest in the cutout of the 3. XPOM read command to be sent

the answer of the 2nd XPOM read command at the latest in the cutout of the 4. XPOM read command to be sent

the answer of the 3rd XPOM read command at the latest in the cutout of the 5. XPOM read command sent to,

...

Comment [JF21]: I do not understand this section

In this way is a very quick readout of data is possible.

5.5.1 Byte Read

Command: XPOM Byte Read	
►	Bit Write according to [RCN214] section 5 1110-10SS VVVV-VVVV VVVV-VVVV VVVV-VVVV 1111-DBBB
◄	Channel 2 (36 Bit): 10SS (ID8-11) CV[V+0] CV[V+1] CV[V+2] CV[V+3]

Comment [JF22]: What does V stand for in this case?

At the time of the feedback, the preceding format must be used.

5.5.2 Byte Write

Command: XPOM Bit Write	
►	Byte write according to [RCN214] section 5. 1110-11SS VVVV-VVVV VVVV-VVVV VVVV-VVVV {DDDD-DDDD {DDDD-DDDD {DDDD-DDDD {DDDD-DDDD}}}
◄	Channel 2 (36 Bit): 10SS (ID8-11) CV[V+0] CV[V+1] CV[V+2] CV[V+3]

The CV values must be returned as they were after the write, even if the write operation could not be performed or only partially due to a limited range of CV values for a particular CV.

5.5.3 Bit Write

Response for a bit write command - the Read Bit command is not implemented as the whole byte is always returned.

5.6 Test Feature ID (ID 12)

The test feature ID allows manufacturers the opportunity to develop new Railcom features for testing and implementation in products without creating an incompatibility with existing protocol.

To create a new feature the manufacturer must first contact the RailCommunity and get an ID number, which has to be entered in CV130 of the RailCom block. This number can be used to differentiate between different applications. The default value of CV130 is "0".

The manufacturer must agree to give the RailCommunity a detailed description (as practiced in this specification for the other IDs) at the completion of the development to be added into the Railcom Specification.

Comment [JF23]: Can features become proprietary?

The RailCommunity will publish the number assignment and the details of the feature as it is at the launch of the product. This will allow other manufacturers to take advantage of this new feature. For regular use, however, another RailCom ID according to Table 5 must be used.

5.6.1 List of Previously Issued Test Feature ID’s

This list shows the assigned Test Feature IDs with their intended use.

Table 12: Test Feature IDs

Test Feature ID (decimal)	Purpose	Granted to/on
0	Default	
10	“Layout Sound” – Synchronization of decoders with sound generators on the train layout.	Train Control Systems, March 28th, 2013

6 Application for Accessory Decoder (switches, etc…)

6.1 SRQ – Service Request for Accessory Decoder

Channel 1 of the cutout is used by accessory decoders to cause the panel to communicate. This request is referred to below as SRQ (Service Request).

The SRQ can be sent either to any accessory decoder package (no matter which address is addressed, whether it's a simple or extended accessory control package) or to a NOP if its own address is less than or equal to that in the NOP. (See chapter 4.3.4)

In the case of an SRQ after a NOP , the corresponding message in channel 2 has to be sent in order to save some time. However, if the SRQ is after a regular accessory decoder command, then the message must not be sent so that the messages of the addressed decoder are not obscured.

The SRQ is a 12-bit datagram. In contrast to all other datagrams, the SRQ contains no identifier, but the 12 useful bits are used to transmit the complete accessory address.

In the operation commands for accessories decoders are the address and data are combined so that the address is part of the specified command. Therefore, the entire command packet is shown here (but without sync bits and check byte).

Command: Accessory Control	
►	Simple accessory control according to [RCN213] section 2.1 10AA-AAAA 1AAA-DAAR
►	Extended accessory control according to [RCN213] section 2.3 10AA-AAAA 0AAA-0AA1 DDDD-DDDD
►	NOP command for simple and advanced accessory decoder according to [RCN213] section 2.5 10AA-AAAA 0AAA-1AAT
◄ ◄	Channel 1: 0 A10 A9 A8 A7 A6 A5 A4 A3 A2 A1 A0 (Simple accessory decoder) Channel 1: 1 A10 A9 A8 A7 A6 A5 A4 A3 A2 A1 A0 (Advanced Accessory Decoder)
◄	Channel 2 only in response to NOP: SRQ message, eg. B. Errors

Restricting the address resolution on Simple Accessory Decoders to output pairs, as well as the Extended Accessory Decoders, provides an 11-bit address and can thus distinguish between these two categories with the 12th bit of the SRQ datagram.

If a decoder has issued an SRQ, it must be repeated until it has been treated. The decoder must at this time no to respond to control commands directed.

An SRQ is considered treated if the decoder has received a clear command on its own address. The deletion command is the "coil off" command or the "absolute stop" command. In this state, the commands mentioned are not executed as such, but merely cause the SRQ to stop being sent.

Deletion command for simple accessory decoder: coil off

Format: 10AA-AAAA 1AAA-0AA0

Deletion command for extended accessory decoder: Absolute stop (= Aspect 0)

Format: 10AA-AAAA 0AAA-0AA1 0000-000

Note: A RailCom-enabled center regularly sends a NOP to enable SRQs. Receives an accessory decoder in the first 5 seconds after the first reception of the DCC format no NOPs, he may assume that the central RailCom capable not is , therefore no SRQ can be edited and he does not need to send. In this case , the function of the decoder is not blocked.

6.2 POM (ID 0)

POM means "Programming On the Main", ie programming on the track to drive. As accessory decoder usually fixedly connected to the digital signal for the operation are allow these commands the reading and writing of configuration variables in the operation mode corresponding to [RCN214] Section 2. These commands is in the channel 2 with a 12-bit datagram with ID0 = 0000 and 8 data bits answered. The data bits contain the value of the CV.

The POM commands from the Command Station to the decoder (►) are noted without the addressing data. The addressing is done according to [RCN214] Section 2.1.

Command: POM	
►	Access to Bytes or Bits According to [RCN214] Section 2 1110-KKVV VVVV-VVVV DDDD-DDDD 1110-10VV VVVV-VVVV 111K-DBBB
◄	Channel 2: 0000 (ID0) DDDD-DDDD with D = CV data

6.2.1 Byte Read

Command: POM Byte Read	
►	Read Byte command according to [RCN214] Section 2 1110-01xx
◄	Channel 2 (12Bit): 0000 (ID0) DDDD-DDDD with D = CV data.

The associated response datagram (ID0) does not have to be sent in the same package frame, but can be sent later. A central office must therefore ensure that the decoder is addressed again and no other command is sent to this decoder (the same is allowed).

When the reading process is complete is the decoder sends the result to the associated, again read command. If the decoder does not return the data within 0.5s then the read is considered failed.

6.2.2 Byte Write

Command: POM Byte Write	
►	Command Byte Write according to [RCN214] Section 2 1110-11xx
◄	Channel 2 (12-bit): 0000 (ID0) DDDD-DDDD Here , with D = CV data, the value returned after the POM operation in the CV is returned.

If the decoder does not return the data within 0.5s then the write is considered failed.

For a CV that can only be read, the decoder returns the current value of the CV.

6.2.3 Bit Write

Command: POM Bit Write	
►	Command Write bit according to [RCN214] section 2 1110-10xx
◄	Channel 2 (12-bit): 0000 (ID0) DDDD-DDDD Here, with D = CV data, the value returned after the POM operation in the CV is returned.

Answers like "Byte Write".

6.3 STAT1 (ID 4)

This feedback is used to transmit status messages from accessory decoders,

Part 1

Comment [SB24]: Is there supposed to be a Part 2? I don't see a Part 2 anywhere.

Command: STAT1	
►	Operation command to decoder address according to [RCN213] section 2.1 or 2.3 10AA-AAAA 1AAA-DAAR or 10AA-AAAA 0AAA-0AA1
◄	Channel 2 12Bit: 0100 (ID4) DDDD-DDDD

The status code can be returned as an acknowledgment after accessory decoder commands. An "ACK" can then be dispensed with.

1. Datagram

Table 13: Status Messages, Part 1

Bit	Meaning
11..8	Identifier 0x4

7	Reserved
6	0: Initial state does not match the last received command. 1: Initial state matches the last received command.
5	0: the returned Aspect is the setpoint 1: the returned aspect is the actual value based on real feedback
4..0	Current Aspect. For example, turnout decoders have 2 output states, signal decoder (so-called extended accessory decoder) up to 31

6.4 Time (ID 5)

This confirmation is used to transfer the forecast round trip time.

Command: Time	
►	Operation command to decoder address according to [RCN213] section 2.1 or 2.3 10AA-AAAA 1AAA-DAAR or 10AA-AAAA 0AAA-0AA1
◄	Channel 2 12Bit: 0101 (ID5) DDDD-DDDD

This command confirmation can be returned as an acknowledgment after accessory decoder commands . An "ACK" can then be dispensed with.

The 7 low-order bits of the remaining term characterize the runtime until the end condition of this term is reached (predicted orbital period). The time is given in 1/10 seconds (MSB = 0) or 1 second (MSB = 1) depending on the MSB . A time of 0 means no switching time - eg with signal decoders without light bulb simulation. This results in a value range 0 ... 12 , 7 seconds or 0 ... 127 seconds.

1. Datagram

Table 14: Forecasted Round Trip Time

Bit	Meaning
11..8	Identifier 0x5
7	0: resolution 1/10 second 1: Resolution 1 second
6..0	Forecasted round trip time

6.5 Error (ID 6)

This feedback is used to transmit error information.

Command: ERROR	
►	Operating command to decoder address according to [RCN213] section 2 (including NOP)
◄	Channel 2 12Bit: 0110 (ID6) DDDD-DDDD

The error code may be returned after each instruction addressing the decoder, including NOP. An "ACK" can then be dispensed with.

The error message transfer is cleared by a new switching command. If there is a permanent error, the decoder must not trigger a new SRQ, as long as the same error exists. The central office must therefore address the decoder again after the deletion to determine whether it is a permanent error.

Table 15: Error Messages

Bit	Meaning	
11..8	Identifier 0x6	
7	Reserved	
6	0: There is only the error specified in the following 6 bits. 1: There are other errors besides the one given.	
5..0	error code	meaning
	0x00	No mistake (more)
	0x01	Command could not be executed, unknown command / invalid aspect.
	0x02	Current consumption of the drive too high.
	0x03	Supply voltage too low, the function is not ensured.
	0x04	Fuse defective.
	0x05	Temperature too high.
	0x06	Feedback error (unwanted adjustment detected)
	0x07	Manual adjustment (eg by a button on the decoder)
	0x10	Switch lantern or signal lantern broken
	0x20	Servo defective.
	0x3F	Internal decoder error, eg self-test processor checksum faulty.

6.6 DYN (ID 7)

This feedback is used to transmit dynamic information from accessory decoders. "Dynamic information" refers to CV contents (RailCom CVs) that change during operation.

Command: DYN	
►	Operation command to decoder address according to [RCN213] section 2.1 or 2.3 10AA-AAAA 1AAA-DAAR or 10AA-AAAA 0AAA-0AA1
◄	Channel 2 18Bit+[18Bit]: 0111 (ID7) DDDD-DDDD-XXXX-XX [0111 (ID7) DDDD-DDDD-XXXX-XX]

The transmission of dynamic variables (DV) takes place in an 18-bit datagram (ID7), which contains the 8-bit value of the DV (D) and a 6-bit subindex (X), which selects one of 64 possible DVs. The meaning of the DV is determined by the subindex. Any two DVs can be transmitted in a feedback frame. Which DVs a decoder sends when, he determines himself.

Table 16: Dynamic Information

X	Meaning
0	Flag Register, content yet to be determined
1-63	Reserved

6.7 XPOM (ID 8 to ID 11)

480 These confirmations correspond to those for vehicle decoders. Therefore, reference is made to section 5.5.

6.8 STAT2 (ID 8)

Does not serve for new designs, but is already used by components on the market. In other use of the ID8, the length of the datagram is used as a distinguishing feature.

485 This feedback is used to transmit status messages from accessory decoders

Part 2

Comment [SB25]: Is this the Part 2 corresponding to the Part 1 above?

Command: STAT2	
►	Operation command to decoder address according to [RCN213] section 2.1 or 2.3 10AA-AAAA 1AAA-DAAR or 10AA-AAAA 0AAA-0AA1
◄	Channel 2 12Bit: 1000 (ID8) DDDD-DDDD with D = data according to Table 17, bits 7 ... 0

Specially tailored for mechanical adjustments.

1. Datagram

Table 17: Status Messages, Part 1

Comment [SB26]: Should this be Part 2?

Bit	Meaning
11..8	Identifier 0x8
7..4	Configuration, previously set: 0000 - decoupler 0001 - switch 0010 - three way switch 0011 - Double intersection 1000 - track lock signal 1001 - shape signal Hp0 / Hp1 1010 - Shape signal Hp0 / Hp1 / Hp2 1011 - Advance signal Vr0 / Vr1 1100 - Advance signal Vr0 / Vr1 / Vr2 1101 - Railway barrier
3	0: The state reported in bits 2 ... 0 corresponds to the setpoint value, or"Setting process is still going on". 1: The reported status corresponds to the actual value based on real feedback.
2..0	Current state

7 Document History

Date	Description
Mar 1997	The first version received approval from the NMRA Board of Trustees as a Recommended Practice.
Jul 2002, Jul 2003, Jan 2006	Subsequent changes were adopted.
Jul 2012	Changed from Recommended Practice to a Standard.
Jan 3, 2021	Draft – Updates made to harmonize with RCN-217.

[i] RCN stands for RailCommunity Normen. The direct German to English translation of Normen is Norms and in this context is intended to have an equivalent meaning to Standards. RailCommunity is an organization of manufacturers that creates German language standards for model railway electronics.

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.