

Norm	RCN-211 DCC-Protokoll Paketstruktur, Adressbereiche und globale Befehle	
Ausgabe 23.07.2023		RailCommunity – Verband der Hersteller Digitaler Modellbahnprodukte e.V.

Inhalt

1 Allgemeines	1
1.1 Zweck der Norm	1
1.2 Anforderungen	2
1.3 Erläuterungen	2
2 Aufbau des DCC-Datenpakets	3
3 Adressbereiche	4
4 Sonderpakete	5
4.1 Rücksetzpaket	5
4.2 Leerlaufpaket	6
5 Globale Befehle	6
5.1 Zeitbefehl	6
5.2 Systemzeit	7
5.3 Zentralen-Eigenschaftenkennung	8
6 Wiederholung der DCC-Pakete	9
Anhang A: Verweise auf andere Normen	10
A.1 Normative Verweise	10
A.2 Informative Verweise	10
Anhang B: Historie	10

1 Allgemeines

1.1 Zweck der Norm

Diese Norm beschreibt die allgemeine Paketstruktur beim DCC-Protokoll, basierend auf [RCN-210]. Die auf dieser Struktur aufbauenden Pakete werden in den Normen [RCN-212], [RCN-213] und [RCN-214] beschrieben. Ausnahmen hiervon bilden die beiden Spezialpakete Rücksetzpaket und Leerlaufpaket.

Diese Norm entspricht im Wesentlichen [S-9.2] bzw. [NEM 671], wobei aber das Basispaket nach [RCN-212] ausgelagert wurde und die Abschnitte A und B von [S-9.2.1] hinzugefügt wurden.

1.2 Anforderungen

Um diese Norm zu erfüllen müssen alle von einer Zentrale für das DCC-Protokoll gesendeten Pakete dieser Struktur inklusive der geforderten Zahl an Synchronbits entsprechen, wobei die Bit-Übertragung entsprechend [RCN-210] zu erfolgen hat. Ein Decoder muss in der Lage sein, die auf dieser Struktur aufgebauten und in [RCN-212] bzw. [RCN-213] als Minimum definierten Befehle zu empfangen und alle direkten Forderungen an den Decoder in dieser Norm erfüllen.

Eine Zentrale darf nur DCC-Pakete senden, die in einer RCN oder einer Norm der NMRA definiert sind, oder von der RailCommunity explizit für eine gesonderte Verwendung freigegeben wurden. Als DCC-Paket ist dabei jedes Signal zu verstehen, dass der [RCN-210] entspricht, mit einer Folge von 11 oder mehr Einsbits beginnt und Gruppen von jeweils 8 Bits untereinander und gegenüber den Einsbits am Anfang durch jeweils ein Nullbit trennt und am Paketende durch ein Einsbit abgeschlossen ist.

Eine Zentrale sollte alle Fahrzeugadressen bis einschließlich 127 als kurze Adresse senden.

Ein Decoder, der einen Adressbereich unterstützt, muss immer den gesamten Adressbereich unterstützen. Ein Fahrzeugdecoder muss immer beide Adressbereiche unterstützen.

Decoder müssen alle Befehle, die sie **nicht** unterstützen, ignorieren und diese Befehle dürfen nicht zu einer Fehlfunktion des Decoders führen. Ein Paket, dessen Signalform nicht [RCN-210] entspricht oder nicht der im Abschnitt "Aufbau des DCC-Datenpakets" beschriebenen Struktur entspricht, darf nicht als DCC-Paket interpretiert werden.

1.3 Erläuterungen

- Ein DCC-Datenpaket ist eine definierte Folge von Bits, die als Gleissignal in [RCN-210] beschrieben sind.
- Als Bytes werden Bitgruppen aus je acht Bits bezeichnet.
- Jedes Bit im Byte hat eine von seiner Position abhängende Wertigkeit, das erste gesendete, in der Darstellung linke Bit, hat die höchste Wertigkeit und heißt MSB "most significant bit". Die Bits eines Bytes werden von links mit 7 beginnend nach rechts fallend bis 0 nummeriert. Das zuletzt gesendete, in der Darstellung rechte Bit heißt LSB "least significant bit".
- Folgende Zeichen werden zur Kennzeichnung der Bedeutung eines Bits verwendet:

0 Bitwert 0

1 Bitwert 1

A Adressbit

D Datenbit

P Prüfbit – die acht Bits des letzten Bytes eines Paketes

x Platzhalter für ein Bit, dessen Wert von der Art des Pakets und des Befehls abhängt und an der Stelle nicht näher betrachtet wird.

Weitere Zeichen werden bei den einzelnen globalen Befehlen definiert.

2 Aufbau des DCC-Datenpakets

Die nachfolgend beschriebenen Bestandteile des Datenpakets aus Bits und Bytes definieren den allgemein gültigen Aufbau des DCC-Datenpakets. Die Teile 4 und 5 kommen jeweils paarweise ein- oder mehrmals vor.

1. Synchronbits

Die Kennzeichnung eines DCC-Datenpaketes und die Synchronisation auf die Bytegrenzen erfolgt mit einer Folge von Einsbits.

Eine Zentrale muss mindestens 17 Synchronbits senden.

Ein Decoder muss ein Paket mit 12 oder mehr Synchronbits empfangen können.

Ein Decoder darf ein Paket mit weniger als 10 Synchronbits nicht als gültig anerkennen.

2. Paket-Startbit

Das Paket-Startbit ist ein Nullbit, das den Synchronbits unmittelbar folgt. Das Startbit schließt die Synchronisation ab und zeigt dem Decoder an, dass die folgenden Bits zum ersten Byte des Paket es gehören.

3. Erstes Byte

Das Erste Byte eines Paket es ist im Betriebsmodus ein Adressbyte oder im Programmiermodus ein Befehlsbyte.

4. Datenbyte-Startbit

Dieses Nullbit leitet das folgende Datenbyte ein.

5. Datenbyte

Die in jedem Datenbyte enthaltenen 8 Bits werden benutzt für Adressen, Instruktionen (Steuerbefehle), Daten oder zur Übertragungsfehlererkennung als Prüfbyte.

Das letzte Byte eines DCC-Paket es ist immer das Prüfbyte.

6. Paket-Endebit

Das Paket-Endebit ist ein Einsbit und markiert das Ende des Datenpaket es.

Folgt ohne Unterbrechung das nächste DCC-Paket, darf dieses Bit bei den Synchronbits des folgenden Paket es mitgezählt werden.

Folgt auf das Paket ein anders Format oder eine Unterbrechung, so muss die DCC-Bitfolge für mindestens 26 µs nach dem Paket-Endebit weitergeführt werden, d.h. es darf in dieser Zeit kein Polaritätswechsel oder ein Abschalten der Spannung erfolgen.

Das Prüfbyte wird gebildet, indem alle vorhergehenden Bytes bitweise logisch per "Exklusiv-Oder (EXOR)" verknüpft werden. Es kann geprüft werden, indem alle Bytes einschließlich des Prüfbytes bitweise logisch EXOR-verknüpft werden. Das Ergebnis sollte dabei Null sein. Decoder müssen das Prüfbyte testen und bei einem Fehler das Paket ignorieren.

Ein DCC-Paket ist mindestens drei Byte lang. Damit ergibt sich als Bitfolge für ein Paket mit drei Bytes:

11111111111111111111	0	xxxxxxxx	0	xxxxxxxx	0	PPPPPPP	1
Synchronbits & Start		Byte 1		Byte 2		Byte 3 = Prüfbyte und Paket-Endebit	

Für das Beispiel eines Basis Geschwindigkeits- und Richtungsbefehls für die 7-Bit-Adresse 55 vorwärts mit Fahrstufe 14 wäre die Bitfolge:

111111111111111111 0 00110111 0 01100111 0 01010000 1

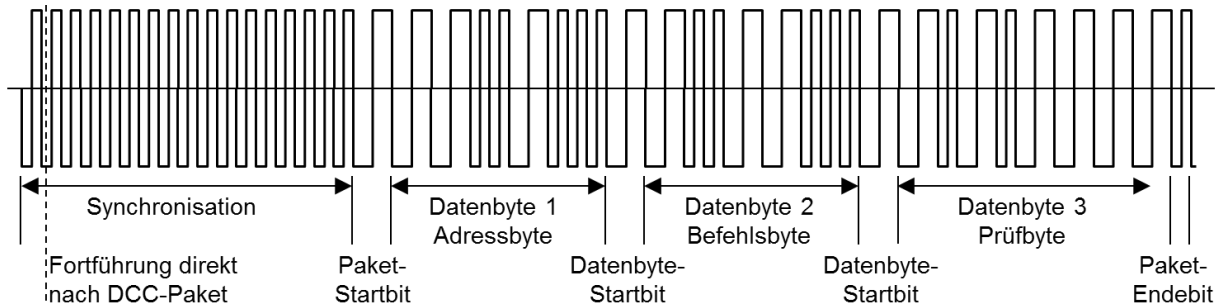


Bild 1: Beispiel eines DCC-Paketes mit drei Bytes (1 Adressbyte, 1 Befehlsbyte, 1 Prüfbyte), kodiert für Adresse 55 und Vorwärtsfahrt mit Fahrstufe 14. Es sind die minimal zu sendenden 17 Synchronbits dargestellt.

Das Bild 1 stellt exakt ein Paket dar. Davor und dahinter kann ein beliebiges Signal anliegen. Es beginnt mit 17 Synchronbits und endet die geforderten 26 µs nach dem Paket-Endebit. Wenn zwei DCC-Pakete ohne Lücke aufeinander folgen, kann – wie schon vorher festgelegt – das Paket-Endebit des ersten Paketes als erstes der 17 Synchronbits des zweiten Paketes betrachtet werden. Daher beginnt dann der Kurvenzug für das zweite Paket an der gestrichelten Linie.

Die Bitfolge für ein sechs Byte langes Paket ist:

{Synchronbits} 0 xxxxxxxx 0 xxxxxxxx 0 xxxxxxxx 0 xxxxxxxx 0 xxxxxxxx 0 xxxxxxxx 0 P P P P P P P P 1

Die maximale Länge eines Paketes ist nicht explizit festgelegt, sondern ergibt sich indirekt aus der im Abschnitt "Wiederholung der DCC-Pakete" festgelegten Zeit zwischen zwei Startbits.

3 Adressbereiche

Das erste Byte des DCC Paketformats enthält im Betriebsmodus die primäre Adresse. Um verschiedene Arten von Decodern zuzulassen, ist diese primäre Adresse wie folgt in feste Blöcke unterteilt.

- Adresse **0000-0000 (0)**:
Broadcast Adresse für Nachrichten an alle Fahrzeugdecoder.
- Adressen **0000-0001 bis 0111-1111 (1-127)(inklusive)**:
Fahrzeugdecoder mit 7 Bit Adressen **0AAA-AAAA**
- Adressen **1000-0000 bis 1011-1111 (128-191)(inklusive)**:
Einfache Zubehördecoder mit 11 Bit Adressen **10AA-AAAA 1AAA-DAAR**,
Erweiterte Zubehördecoder mit 11-Bit Adressen **10AA-AAAA 0AAA-0AA1** und
NOP-Befehl für beide Arten von Zubehördecodern **10AA-AAAA 0AAA-1AAT**

- Adressen **1100-0000** bis **1110-0111** (192-231)(inklusive):
Fahrzeugdecoder mit 14 Bit Adressen **11AA-AAAA AAAA-AAAA**
- Adressen **1110-1000** bis **1111-1100** (232-252)(inklusive):
Reserviert für zukünftige Anwendungen
- Adresse **1111-1101** (253): Erweiterte Befehle nach [S-9.2.1.1]
- Adresse **1111-1110** (254): Automatische Anmeldung nach [RCN-218]
- Adresse **1111-1111** (255): Leerlauf oder auch Idle Paket

An die Broadcast Adresse **0000-0000** adressierte Befehle müssen von allen Fahrzeugdecodern ausgeführt werden. Das Rücksetzpaket – das ist ein Rücksetzen Befehl **0000-0000** an die Broadcast Adresse **0000-0000** – muss auch von den Zubehördecodern beachtet werden.

Bei den Adressen im Bereich **0111-0000** bis **0111-1111** (112-127) gibt es eine Überschneidung mit den Befehlen für den Programmiermodus. Beim Programmiermodus wird ein einzelner Decoder auf einem isoliertem Gleisabschnitt konfiguriert und daher haben die entsprechenden Pakete keine Adresse sondern beginnen direkt mit den Befehlsbytes. Nach einem Rücksetzpaket dürfen die nächsten zehn Pakete für den Normalbetrieb nicht an eine dieser Adressen gesendet werden um zu verhindern, dass ein Decoder ungewollt in den Programmiermodus wechselt. Die genauen Regeln für den Übergang zwischen den Betriebsarten sind in [RCN-216] "DCC Programmierumgebung" festgelegt.

Der Aufbau der Adressen für Zubehördecoder und die Wertigkeit der einzelnen Adressbits in diesem Adressbereich sind in [RCN-213] "DCC Betriebsbefehle für Zubehördecoder" festgelegt.

Bei den Adressen für die Fahrzeugdecoder mit 14 Bit Adressen enthält das erste Adressbyte die sechs höherwertigen Adressbits. Das zweite Adressbyte liefert die acht niederwertigen Adressbits. Damit ist das höchstwertigste Bit der 14 Bit Adresse das Bit 5 des ersten Adressbytes und das niederwertigste Bit der 14 Bit Adresse das Bit 0 des zweiten Adressbytes.

4 Sonderpakete

4.1 Rücksetzpaket

Die Bitfolge dieses Befehls ist:

{Synchronbits} 0 0000000 0 0000000 0 0000000 1

Das Datenpaket, in dessen Bytes alle Bits den Wert "0" besitzen, ist das Rücksetzpaket. Es löscht alle flüchtigen Speicher aller Decoder einschließlich der Geschwindigkeits- und Fahrtrichtungsdaten. Die Decoder kehren nach dem Empfang des Rücksetzpakets in den Einschaltzustand zurück, in Bewegung befindliche Fahrzeuge haben einen Nothalt auszuführen. Zur Vorbereitung auf den Programmiermodus sind alle Verbraucher abzuschalten. Dieser Befehl gilt sowohl für Fahrzeug- als auch für Zubehördecoder.

Folgt einem Rücksetzpaket ein DCC-Paket, dessen erstes Byte im Bereich zwischen **0111-0000** (Adresse 112) und **0111-1111** (Adresse 127) liegt, so schalten Decoder entsprechend [RCN-216] in den Programmiermodus, sofern sie diesen unterstützen.

4.2 Leerlaufpaket

Die Bitfolge dieses Befehls ist:

{Synchronbits} 0 11111111 0 00000000 0 11111111 1

Das Datenpaket, dessen erstes und drittes Byte 8 Einsbits und dessen zweites Byte 8 Nullbits enthält, ist das allgemeine Leerlaufpaket.

Ein Decoder darf aufgrund dieses Befehls keine Aktion auslösen und muss sich verhalten, als würde ein normales, an andere Decoder adressiertes Datenpaket gesendet. Die Zentrale sendet dieses Paket wenn keine anderen Daten zu senden sind oder um z.B. zwischen zwei Befehlen an einen Decoder eine Lücke von mehr als 5 ms zu erzeugen (siehe folgenden Abschnitt). Das Leerlaufpaket dauert inklusive der 17 Synchronbits etwa 6 ms.

RailComPlus® verwendet die Adresse 255 (0xFF) für andere Pakete mit größerer Länge, die aber weder von der RailCommunity noch der NMRA genormt wurden.

5 Globale Befehle

Diese Befehle werden ausschließlich an die Broadcast-Adresse 0 gesendet und beginnen immer mit **{Synchronbits} 0 0000-0000 ...** und enden mit der Prüfsumme ... **PPPPPPP 1**. Daher werden im Folgenden nur die Bytes der Befehle und nicht das ganze Paket dargestellt. Die Befehle können von Fahrzeug- und Zubehördecodern genutzt werden

5.1 Zeitbefehl

Dieser Befehl ist vier Byte lang und hat das Format:

1100-0001 CCxx-xxxx xxxx-xxxxx xxxx-xxxx

CC gibt an, welche Daten in dem Paket übertragen werden:

CC = 00 Modellzeit:

1100-0001 00MM-MMMM WWWH-HHHH U0BB-BBBB mit:

MMMMMM = Minuten, Wertebereich: 0..59

WWW = Wochentag, Wertebereich: 0 = Montag, 1 = Dienstag, 2 = Mittwoch, 3 = Donnerstag, 4 = Freitag, 5 = Samstag, 6 = Sonntag, 7 = Wochentag wird nicht unterstützt.

HHHHH = Stunden, Wertebereich: 0..23

U = Update, d.h. die Zeit hat sich sprunghaft geändert, z.B. um einen neuen Fahrplan zu starten. Pro sprunghafter Änderung können bis zu 4 Befehle so markiert werden.

BBBBBB = Beschleunigungsfaktor, Wertebereich 0..63. Ein Beschleunigungsfaktor von 0 bedeutet die Modelluhr wurde angehalten, ein Faktor von 1 entspricht Echtzeit, bei 2 läuft die Uhr doppelt so schnell, bei drei dreimal so schnell wie die Echtzeit, usw.

CC = 01 Datum:
1100-0001 010T-TTTT MMM-YYYY YYYY-YYYY mit:

TTTT = Tag im Monat, Wertebereich: 1..31
MMM = Monat, Wertebereich: 1..12
YYYYYYYYYYYY = Jahr, Wertebereich: 0..4095

CC = 10 Zeitmaßstab:
1100-0001 1011-1111 SEEE-EEMM MMM-MMMM mit:

Im dritten und vierten Byte wird der Beschleunigungsfaktor als 16 Bit float dargestellt, wie er von vielen Controllern verarbeitet werden kann.

S = Vorzeichen (sign)
EEEE = 5 Bit Exponent
MMMMMMMM = 10 Bit Mantisse

Damit können beliebige Zeitmaßstäbe auch mit langsamer Uhr eingestellt werden. Wird dieses Paket gesendet, so ist der Beschleunigungsfaktor in dem Paket für die Modellzeit zu ignorieren.

CC = 11 reserviert

Die Zentrale sollte die Modellzeit maximal jede (Modell-)Minute einmal aussenden. Die Zentrale darf Zeitausgaben auslassen, z.B. bei großem Bandbreitenbedarf sonstiger Pakete. Übersprungene Zeitpunkte müssen von den Decodern toleriert werden und können fallweise anhand des Beschleunigungsfaktors ersetzt werden.

Das Datum und der Zeitmaßstab werden nur bei Änderung, ggf. mehrfach, übertragen.

5.2 Systemzeit

Der Befehl für die Systemzeit ist drei Byte lang hat das Format:

1100-0010 MMMM-MMMM MMMM-MMMM

In den mit **M** gekennzeichneten Bits werden die Millisekunden seit dem Systemstart übertragen. Der Maximalwert ist 0xFFFF = 65536 und entspricht etwa 65,5 Sekunden. Das zweite Byte enthält die höchstwertigsten Bits, das dritte Byte die niederwertigsten Bits. Nach Erreichen des Maximalwertes beginnt der Zähler wieder bei 0. Bei der Bestimmung von relativen Zeiten bis zu einer Minute kann man einfach mit einem 16 Bit Integer arbeiten ohne einen Überlauf zu beachten.

Der Zeitstempel bezieht sich auf den Anfang des Startbits. Der Befehl sollte alle 30 Sekunden wiederholt werden, um eine ausreichende Synchronisation sicherzustellen. Da diese Pakete im

Wesentlichen zur Erzeugung von Zeitmarken durch stationäre Melder dienen, ist ein mehrfaches Senden des Befehls nicht erforderlich.

5.3 Zentralen-Eigenschaftenkennung

Der Befehl für die Zentralen-Funktionskennung ist vier Byte lang und hat das Format:

1100-0011 1111-IIII DDDD-DDDD DDDD-DDDD

Die mit **I** gekennzeichneten Index-Bits wählen eine von 16 Bitmasken aus. In jeder Bitmaske **DDDD-DDDD DDDD-DDDD** werden jeweils 16 Eigenschaften als unterstützt (Bit = 1) oder nicht unterstützt (Bit = 0) markiert. Die beiden Bytes werden als ein 16-Bit Wort betrachtet und das höherwertige Byte wird zuerst gesendet (drittes Bytes des Befehls)

Der Befehl sollte alle 5 Sekunden wiederholt werden

IIII = 1111 Befehle für Fahrzeugdecoder:

- Bit 0 Adressen 100 bis 127 werden als lange (14 Bit) Adresse gesendet.
- Bit 1 Unterstützt lange Lokadressen 10000 bis 10239
- Bit 2 Unterstützt 128 Fahrstufen
- Bit 3 Unterstützt Kombibefehl für Geschwindigkeit, Richtung und Funktion
- Bit 4 Unterstützt POM-Befehle (schreiben)
- Bit 5 Unterstützt XPOM-Befehle (schreiben)
- Bit 6 bis Bit 7 reserviert
- Bit 8 Unterstützt Lokfunktionen 13 bis 28
- Bit 9 Unterstützt Lokfunktionen 29 bis 68
- Bit 10 Unterstützt kurze Binärzustände
- Bit 11 Unterstützt lange Binärzustände
- Bit 12 Unterstützt Analogfunktionen
- Bit 13 Unterstützt Sonderbetriebsarten-Befehl
- Bit 14 und Bit 15 reserviert.

IIII = 1110 Befehle für Zubehördecoder und Broadcastbefehle

- Bit 0 Abweichende Adresstabelle (Versatz um 4)
- Bit 1 Unterstützt Extended Zubehöradresse
- Bit 2 reserviert
- Bit 3 Unterstützt POM-Befehle (schreiben)
- Bit 4 bis Bit 7 reserviert
- Bit 8 Unterstützt Modellzeit
- Bit 9 Unterstützt Datum
- Bit 10 Unterstützt Zeitmaßstab
- Bit 11 Unterstützt Systemzeit
- Bit 12 bis Bit 15 reserviert

IIII = 1101 Befehle für RailCom & gewünschte Meldungen

- Bit 0 Unterstützt RailCom nach [RCN-217]
- Bit 1 Unterstützt DCC-A nach [RCN-218]
- Bit 2 Unterstützt NOP für Schaltdecoder

- Bit 3 Unterstützt POM-Befehle (lesen)
- Bit 4 Unterstützt XPOM-Befehle (lesen)
- Bit 5 bis Bit 7 reserviert
- Bit 8 Behälterstände (ID 7 DYN, Sub-Index 8 bis 19)
- Bit 9 Betriebsparameter (ID 7 DYN, Sub-Indices 0, 1, 7, 20, 22 und 26)
- Bit 10 Testbetrieb: Spannung am Gleis (ID 7 DYN, Sub-Index 46)
- Bit 11 bis Bit 14 reserviert
- Bit 15 Unterstützt RailComPlus®

IIII = 0000 Testbefehlsgruppe neue Funktionen

Bit 0 bis 15 noch undefinierte Testfunktionen

(Soll nicht in Serie genutzt werden, wegen abweichender Belegungen)

Alle nicht definierten Bits sind reserviert. Nur die Bits in der Testbefehlsgruppe dürfen frei verwendet werden.

6 Wiederholung der DCC-Pakete

Die zu Decodern gesendeten DCC-Pakete sollen so oft wie möglich wiederholt werden, weil sie durch Störungen oder aufgrund schlechter elektrischer Leitfähigkeit zwischen Schienen, Rädern und Stromabnehmern Informationsverluste erleiden können. Zwischen dem Paket-Endebit und den Synchronbits des folgenden DCC-Paketes kann die Gleisspannung abgeschaltet werden, um andere Steuerformate zuzulassen. Alternative Datenformate zwischen dem Paket-Endebit und den nächsten Synchronbits sind explizit erlaubt.

Ein Decoder muss auf alle an ihn adressierte Pakete reagieren können, wenn die Zeit zwischen Paket-Endebit des einen Paketes und dem Startbit des folgenden Paketes mindestens 5 ms beträgt. Falls ein Decoder eine Bitfolge mit einem fehlenden oder ungültigen Null-Bit vor einem Byte, einem fehlenden oder ungültigen Paket-Endebit oder einem fehlerhaften Prüfbyte empfängt, muss er die nächsten Synchronbits als Beginn eines neuen Pakets erkennen.

Eine automatische Umschaltung auf andere Protokolle oder Betriebsarten ist zulässig. Weitere Regeln zur Umschaltung auf andere Protokolle oder Betriebsarten sind in [RCN-200] und [RCN-216] festgelegt.

Zentralen müssen so konfiguriert werden können, dass sie wenigstens ein vollständiges Paket alle 30 ms senden, wobei die Zeit zwischen den Paket-Startbits gemessen wird.

Zusammenfassung:

Mindestzeitabstand zwischen 2 DCC-Datenpaketen an einen Decoder: $t_D > 5 \text{ ms}$ Distanzzeit

Beharrungszeit der Decoder im DCC-Modus: $t_W \geq 30 \text{ ms}$ Wartezeit

Minimale DCC-Datenpaketrate: $t_R \leq 30 \text{ ms}$ Wiederholzeit

Anhang A: Verweise auf andere Normen

A.1 Normative Verweise

Die hier aufgeführten Normen sind ganz oder in dem beim Zitat angegebenen Rahmen einzuhalten, um diese Norm zu erfüllen.

- [RCN-210] [RCN-210](#) DCC Bit-Übertragung
- [RCN-212] [RCN-212](#) DCC Betriebsbefehle für Fahrzeugdecoder
- [RCN-213] [RCN-213](#) DCC Betriebsbefehle für Zubehördecoder

A.2 Informative Verweise

Die hier aufgeführten Normen und Dokumente haben rein informativen Charakter und sind nicht Bestandteil dieser Norm.

- [RCN-200] [RCN-200](#) Multiprotokollbetrieb - Decoder
- [RCN-214] [RCN-214](#) DCC Konfigurationsbefehle
- [RCN-216] [RCN-216](#) DCC Programmierumgebung
- [RCN-217] [RCN-217](#) DCC Rückmeldeprotokoll RailCom
- [RCN-218] [RCN-218](#) DCC-A – Automatische Anmeldung
- [S-9.2] NMRA: [S-9.2](#) DCC Communications Standard
- [S-9.2.1] NMRA: [S-9.2.1](#) DCC Extended Packet Formats
- [S-9.2.1.1] NMRA: [S-9.2.1.1](#) Advanced Extended Packet Formats
- [NEM 671] MOROP: [NEM 671](#) Digitales Steuersignal DCC Basis-Datenpakete

Anhang B: Historie

Datum	Kapitel	Änderungen gegenüber der jeweils vorhergehenden Version
23.07.2023	4.1	Abschalten aller Ausgänge beim Rücksetzbefehl, expliziter Hinweis, dass dieser Befehl auch für Zubehördecoder gilt.
31.07.2022	Titel 4.3 5 5.1 5.2 5.3	Titel in „Paketstruktur, Adressbereiche und globale Befehle“ geändert. Systembefehle entfernt, Hinweis auf RailComPlus® in 4.2 Abschnitt „Globale Befehle“ ergänzt. Zeitbefehl aus RCN-212 übernommen Systemzeit aus RCN-212 übernommen Zentralen-Eigenschaftenkennung hinzugefügt
10.09.2021	3 3, 5	Ergänzung Adressbereiche (Zubehör-NOP, 253 und 254) Verweise auf RCN-215 ersetzt durch Verweise auf RCN-216
02.12.2018	4.3	Redaktionelle Änderung: Systembefehle in RailComPlus-Spezifikation definiert

08.05.2015	1.2	Redaktionelle Änderung ehemaliger Absatz 2 jetzt Absatz 4
	2	Änderung der Zahl der Synchronbits von 16 auf 17
	2	Zwei zusätzliche Absätze zur Erläuterung unter Bild und am Ende
	4.3	Zusätzliche Erklärungen zu den Systemkommandos
	5	Möglichkeit ein anderes Protokoll zu verwenden, auch wenn DCC gesendet wird
09.05.2014	3	Anpassung an RCN-213
03.05.2013	Alle	Erste Version

Copyright 2023 RailCommunity – Verband der Hersteller Digitaler Modellbahnprodukte e.V.