



# Lokomander II DCC-Decoder für Lokomotive

Bedienungsanleitung  
- Version 0.1.22-



by **TEHNO**  
**LOGISTIC**



© Copyright 2018 Tehnologic SRL

**Alle Rechte vorbehalten**

Keiner Teil dieser Publikation darf in irgendeiner Form oder auf irgendeine Weise, sei es elektronisch oder mechanisch, einschließlich Fotokopieren, reproduziert oder übertragen werden, ohne die vorherige schriftliche Zustimmung von Tehnologic SRL zu beantragen.



Bitte lesen Sie diese Anleitung sorgfältig durch, bevor Sie mit der Installation fortfahren! Obwohl unsere Produkte sehr robust sind, kann ein falscher Anschluss das Modul zerstören!



Überschreiten Sie bitte die angegebenen technischen Parameter nicht! Überprüfen Sie die Umgebung, in der das Gerät installiert werden soll. Es sollte zu keinen direkten Sonneneinstrahlung und Feuchtigkeit ausgesetzt werden.



Während der Installation kann ein elektrischer LötKolben erforderlich sein. Ihre Verwendung benötigt die Anwendung spezieller Sicherheitsmaßnahmen.



Stellen Sie sicher, dass die Unterseite des Geräts keine metallischen (leitfähigen) Oberflächen berührt.



## Inhaltsverzeichnis

Inhaltsverzeichnis .....	3
1. Wichtige Informationen.....	4
2. Verwendete Abkürzungen .....	5
3. Was beinhaltet diese Bedienungsanleitung?.....	6
4. Hauptmerkmale.....	7
5. Technische Spezifikationen .....	8
6. Allgemeine Beschreibung der Lokommander II Decoder .....	9
7. Installation des Decoders .....	14
8. Inbetriebnahme .....	16
9. Die Adresse des Decoders .....	18
10. Beschreibung der Charaktereigenschaften beim Fahren .....	21
10.1. Lineare Geschwindigkeitseinstellung an 3 Punkten .....	21
10.2. Tabellarische Geschwindigkeitseinstellung in 28 Stufen (CV67 ... CV94) .....	22
11. Motorsteuerung.....	24
12. Kontrollierte Stopps, gezielte Anhalten.....	31
12.1. Stopp bei konstantem Distanz (CBD).....	31
12.1.1. Stopp bei fester Verzögerung.....	31
12.1.2. Stopp bei variabler Verzögerung .....	31
12.2. Erkennung des assymetrischen DCC-Signals (Lenz ABC) .....	32
12.3. Pendelsteuerung (push-pull Funktion) .....	33
12.3.1. Ohne Zwischenstopps .....	34
12.3.2. Mit Zwischenstopps .....	34
13. Funktionsausgänge .....	35
14. Decoder in Analog (DC) verwenden .....	40
14.1. Analogmodus 1 .....	40
14.2. Analogmodus 2 .....	41
14.3. Kontrollierte Stopp im DC Sektor.....	42
15. Bidirektionale Kommunikation (RailCom) .....	43
16. Spezielle Funktionen .....	44
17. Automatische Entkupplung .....	47
18. SUSI/ Locowire Schnittstelle .....	51

18.1. Programmierung der SUSI-Module .....	51
19. Verwendung externer Kondensatoren oder von eines power pack.....	53
20. Rücksetzung des Decoders/Reset .....	56
21. Sekundäre Adresse (blockiert das Programmieren des Decoders) .....	57
22. Firmware aktualisieren .....	58
23. Spezielle Firmware-Version für 3V-Motor .....	59
24. Ergänzendes .....	60
25. Technischer Unterstützung .....	61
26. Tabelle der CVs des Decoders.....	62
27. Anhang Bits und Bytes .....	98

## 1. Wichtige Informationen



Bitte lesen Sie diese primären Kapitel sorgfältig durch!

- Die Lokommander Decoders sind ausschließlich zur Verwendung für elektronische Lokomotive gemacht. Jede andere Verwendung ist streng verboten!
  - Vermeiden Sie es, Schläge oder mechanischen Druck auf den Decoder auszuüben.
  - Entfernen Sie nicht den Schrumpfschlauch am Decoder (bei Modellen mit Schutzmanschette).
  - Stellen Sie sicher, dass weder der Lokommander II-Decoder noch die nicht verwendeten Kabel sich in elektrischen Kontakt mit dem Fahrgestell der Lokomotive (Kurzschlussgefahr) befinden. Isolieren sie die Enden der unbenutzten Kabel.
  - Verlöten Sie keine Verlängerungskabel auf der Decoderplatine, es sei denn, dies ist unbedingt erforderlich (Anschlüsse an Soundmodule, Netzteil / power pack).



- Es ist verboten, den Decoder in ein Isolierband zu wickeln, da dies zu Überhitzung führen kann.
- Führen Sie die Verdrahtung zu und von externen Komponenten, wie in dieser Anleitung empfohlen. Eine falsche Verdrahtung / Verbindung kann den Lokommander II-Decoder beschädigen.
- Stellen Sie sicher, dass sich beim Zusammenbau der Lokomotive keine Kabel im Getriebesystem der Lokomotive verfangen.
- Jede verwendete Stromquelle muss durch eine Sicherung oder eine elektronische Sicherheitsvorrichtung geschützt werden, um mögliche Gefahren im Kurzschlussfall vermeiden zu können. Verwenden Sie nur solche Transformatoren oder Verteiler, die speziell für elektrische Züge vorgesehen sind.
- Lassen Sie nicht zu, dass Kinder den Lokommander II-Decoder unbeaufsichtigt verwenden. Lokommander II-Decoder sind kein Spielzeug.
- Verwenden Sie Lokommander II-Decoder nicht in feuchter Umgebung

## 2. Verwendete Abkürzungen

- ABC - automatische Kontrolle der Bremsen (Automatic Brake Control)
- BEMF- erzeugte Gegen EMK des Motors (Back Electro-Motive Force)
- CBD- Konstanter Bremsweg (Constant Braking Distance)
- CV - Konfigurationsvariablen (Configuration Variable)
- DCC- Digitale Befehlssteuerung (Digital Command Control)
- DC - Gleichstrom (Direct Current)
- FL - Licht vorne (Front Light)
- RL - Licht hinten (Rear Light)
- MSB - höchstwertige Bit oder Bytes (Most Significant Byte)
- LSB - niederwertigstes Bit oder Bytes (Least Significant Byte)
- NMRA- National Model RailRoad Association



- NEM- Normen Europäischer Modellbahnen (MOROP)  
PT - Programmierschiene(Programming Track)  
PoM - Programmierung auf dem Hauptgleis (Programming on the Main)  
RCN/RailCommunity - Verband der Hersteller Digitaler Modellbahnprodukte e.V.  
SPP - Smart Power Pack (unterbrechungsfreie Stromquelle)  
n. v. - nicht verbunden

### **3. Was beinhaltet diese Bedienungsanleitung?**

Herzlichen Glückwunsch zum Kauf eines Lokommander II-Decoders. Diese Gebrauchsanleitung wurde in mehrere Kapiteln untergeteilt, die Ihnen Schritt für Schritt zeigen werden, wie Sie einen Lokommander II-Decoder installieren und anpassen können. In den vierten und fünften Kapiteln schaffen wir einen Überblick über die Funktionen und Parameter des Decoders. Kapitel 6 enthält die allgemeine Beschreibung des Gerätes. Kapitel 7 beschreibt detailliert den Einbau des Decoders in die Lokomotiven. Machen Sie sich mit dem Motortyp und der Art der Schnittstelle in der Lokomotive vertraut, bevor Sie dieses Kapitel durcharbeiten. Die Lokommander II-Decoder können mit den meisten Steuerungssystemen betrieben werden, die für elektrische Zugmodelle üblich auf dem Markt produziert werden. Kapitel 8 gibt einen Überblick über die digitalen und analogen Systeme, in denen der Decoder betrieben werden kann, und erläutert die möglichen Besonderheiten. Sie können die Standardeinstellungen Ihres Lokommander II nach eigenem Wunsch ändern. In den Kapiteln 9-17 werden die konfigurierbaren Parameter erläutert und wie sie nach Ihrem Wunsch angepasst werden können. Wir empfehlen Ihnen, die 9-11 Kapitel zu lesen, um die Adress- und Motorsteuerungsparameter zu konfigurieren, und um den Decoder für Ihre Lokomotive anpassen zu können. Kapitel 26 enthält alle Konfigurationsvariable und kann als Referenz verwendet werden.



## 4. Hauptmerkmale

- Generischer DCC-Mobildecoder, NMRA, NEM und RCN RailCommunity Normen kompatibel
- PT oder PoM Programmiermodus
- analog Funktionsweise (DC), Konfiguration der aktiven Funktionen in für DC
- Kurze (1-127) und lange (128-9999) konfigurierbare Adressen
- 14, 28/128 Fahrstufen
- Maximaler Motorstrom von 1000 mA, Lastausgleich und BEMF
- Geschwindigkeit kann man auf 3 Punkte (Vmin, Vmid, Vmax) oder in Tabellenform eingestellt werden
- Rangiergang (umschaltbar von F3, CV114)
- Anfahr- / Bremsverzögerung (umschaltbar von F4, CV115)
- Bremsen mit konstantem Abstand, aktiviert bei ABC oder DC, bei Nullgeschwindigkeit
- Fahrt mit einem niedrigem Geschwindigkeit auf ABC Slow Speed- Abschnitt
- Pendelsteuerung (push-pull Funktion)
- bis zu 10 regulierbare Hilfsausgänge, maximaler Strom 300mA
- Ausgänge auf Funktionen abbilden
- Kurzschluss- und Überstromschutz für Motorausgang und Hilfsausgänge
- RAILCOM Bidirektionale Kommunikation
- SUSI© und LocoWire© Schnittstelle
- Ausgänge für Smart Power Pack (SPP ©)
- Funktion zur Steuerung von elektromagnetischen Kupplungen (Decoupling, Entkupplung)
- Sperrfunktion der Konfigurationsvariablen (sekundäre Adresse)
- Software, die vom Programmierer und mit dem an der Lokomotive angebrachten Decoder nachrüstbar ist
- Die kleine Größen ermöglichen den Einsatz auf der Skala H0, TT (auch bei N)

## 5. Technische Spezifikationen

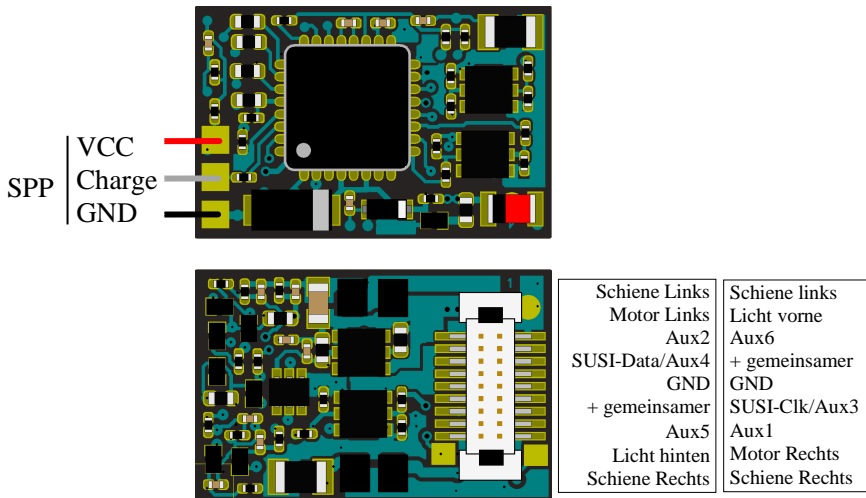
- Versorgungsspannung: 4-24 V, Spannung von den Schienen (DCC)
- Leistungsaufnahme ohne Ausgänge: <10 mA
- Maximalstrom für jeden Ausgang: 200 mA
- Minimalstrom für den Decoder: 400 mA
- Größe (Ohne Kable und Stecker):
  - NEM651: 14x9x3,3mm
  - NEM652,PLUX12/16: 19,5x11x3mm
  - NEXT18: 14,2x9,2x3mm
  - MTC21: 20x15,3x5mm
  - PLUX22: 20,5x15x3,5mm
- Gewicht: TBD 4-6 g
- Schutzklasse: IP00
- Betriebstemperatur: 0 ÷ +60 °C
- Lagertemperatur: -20 ÷ +60 °C
- Luftfeuchtigkeit: max 85%



## 6. Allgemeine Beschreibung der Lokommander II Decoder

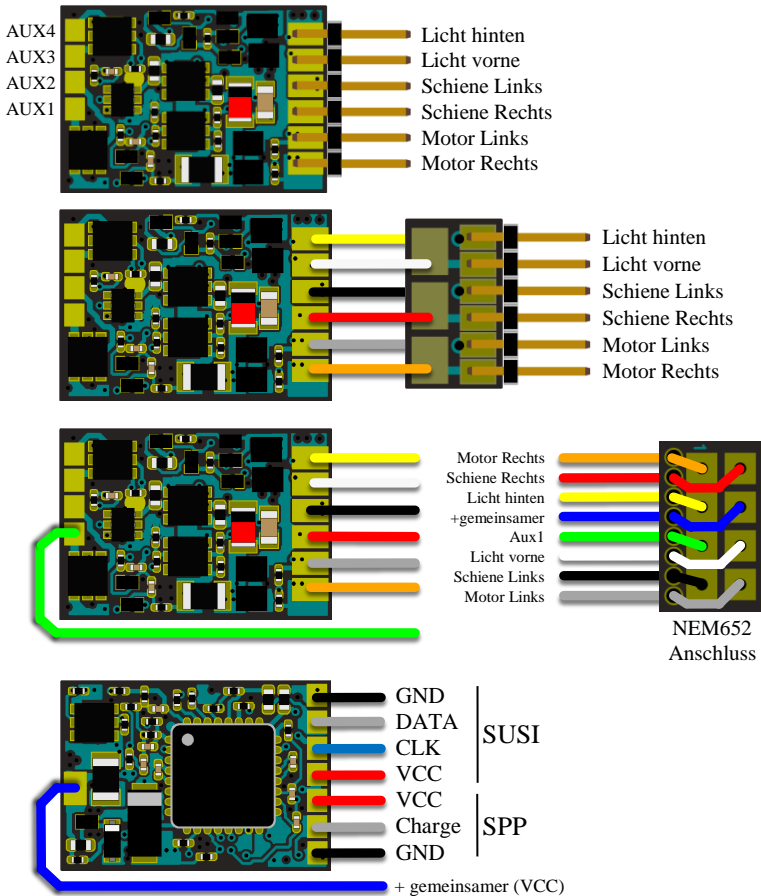
Lokommander II-Decoder sind für die Verwendung bei Eisenbahnmodellen in den Größen N, TT, H0, H0e konzipiert. Die verschiedenen Modelle unterscheiden sich durch ihre Größe, den Steckverbinderotyp, die Stromzufuhr zum Motor und die Anzahl der verfügbaren Hilfsausgänge. Was die Funktion und die Programme betrifft, sie sind identisch.

- Die NEXT18-Version ist 14,2x9,2x3mm.



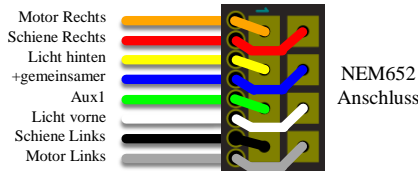
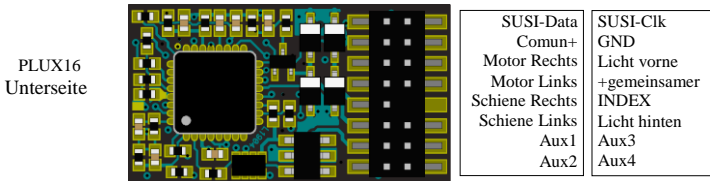
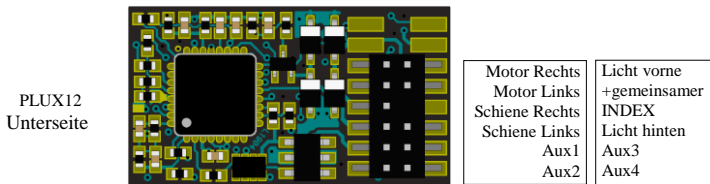
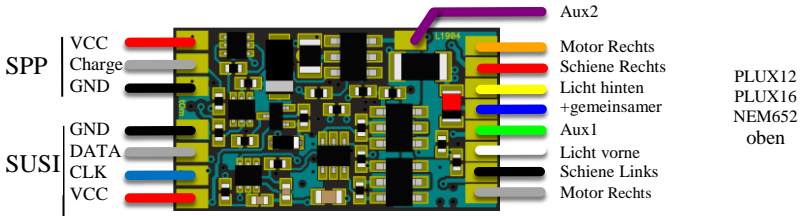
Lokommander II mit NEXT18-Anschluss

- Die MICRO-Version hat eine Größe von 14 x 9 x 3,3 mm und kann mit einem 6-poligen, geraden oder winkligen NEM651-Stecker, oder NEM651-Stecker oder 8-poligem NEM652-Stecker mit angeschlossenen Drähten geliefert werden.



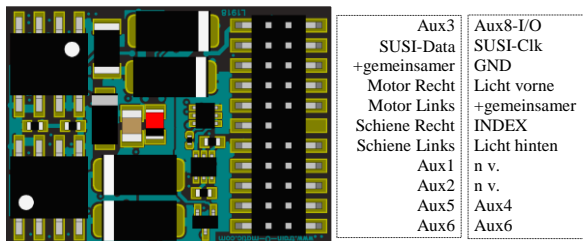
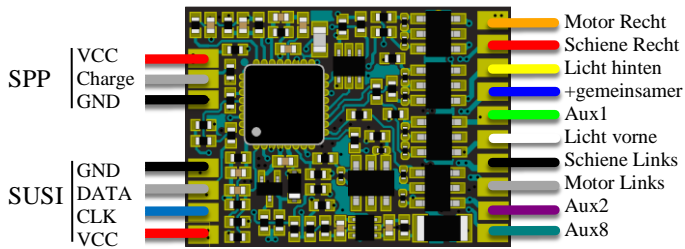
## Lokommander II MICRO

- Die PLUX16-Version ist 19,5 x 11 x 3 mm und kann mit einem 15-poligen PLUX16-Stecker mit PLUX12-Stecker oder einem 8-poligen NEM652-Stecker geliefert werden.



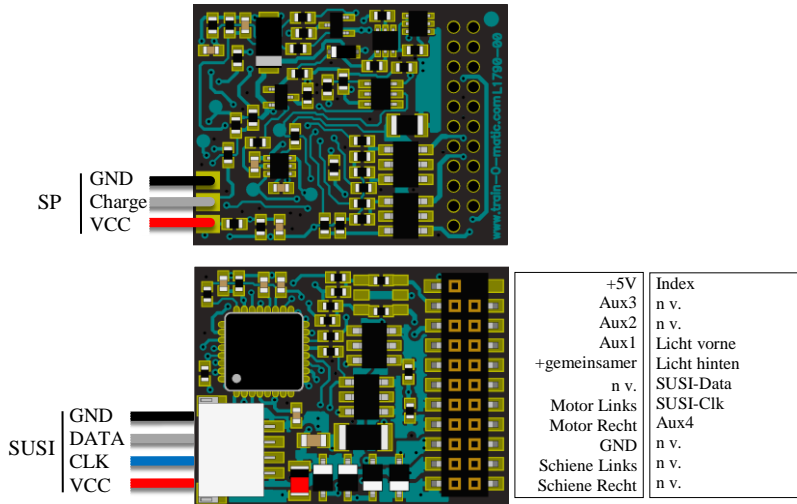
Lokommander II mit PLUX12/16 oder NEM652 -Anschluss

- Die PLUX22-Version ist 14,2x9,2x3mm und kann mit 21-poligen PLUX22- oder 8-poligen NEM652-Pin-Steckverbindern geliefert werden.



Lokommander II mit PLUX22-Anschluss

- Die MTC21-Version ist 20x15,3x5mm. Es kann mit oder ohne 4-poligen SUSI-Anschluss geliefert werden (die Anschlussversion hat das Suffix S, MTC21S).



Lokommander II mit MTC21-

In der folgenden Tabelle finden Sie den Identifikationscode für jede Lokommander II Variante.

Format	Anschlüsse	tOm Bestellcode	Abmessungen ohne Stecker
MICRO	NEM651 gerade	02010220	14x9x3,3mm
MICRO	NEM651 Winkel	02010221	14x9x3,3mm
MICRO	Drähte + NEM651	02010222	14x9x3,3mm
MICRO	Drähte + NEM652	02010223	14x9x3,3mm
NEXT18	NEXT18	02010216	14,2x9,2x3mm
PLUX22	PLUX22	02010217	14,2x9,2x3mm
PLUX22	Drähte + NEM652	02010218	14,2x9,2x3mm
PLUX16	LPUX16	02010211	19,5x11x3mm
PLUX16	PLUX12	02010210	19,5x11x3mm
PLUX16	Drähte + NEM652	02010212	19,5x11x3mm
MTC21	MTC21	02010208	20x15,3x5mm
MTC21	MTC21 + SUSI	02010209	20x15,3x5mm

Tabelle 1.

## 7. Installation des Decoders

Vor der Installation eines Digitaldecoders, insbesondere bei älteren Modellen, ist es ratsam, sich selbst davon zu überzeugen, dass die Lokomotive in Gleichstrom ordnungsgemäß funktioniert. Es wird empfohlen, die folgenden Vorgänge auszuführen:

- Reinigung von Rädern und Stromabnehmerkontakten
- Überprüfung des Motorstatus, Messung des Leerlaufstroms des mit 5-10 V gespeisten Motors, der 200-300 mA nicht überschreiten darf, ggf. Bürsten- und Kollektorreinigung.
- Überprüfung des Getriebes, wenn nötig Achsen und Zahnräder reinigen und schmieren
- Wenn die Lokomotive mit einer Glühlampenbeleuchtung ausgestattet ist, überprüfen Sie, ob sie der Spannung von 16 V standhält, und ersetzen Sie gegebenenfalls die Glühlampen

Bei zur Digitalisierung vorbereiteten Lokomotiven wird die Installation der mit dem Steckverbinder (PLUX, MTC, NEXT18, MICRO-6, NEM652) ausgestatteten Decoder durch den Austausch des Motherboard-Moduls "Dummy" aus dem Mutterplattenanschluss gemacht. In dem auf diese Weise freigegebenen Verbinder wird der tOm Decoder durch Folgen des Schlüssels (INDEX) an den PLUX und den MTC oder, falls vorhanden, die mit der Lokomotive erhaltenen Anweisungen eingefügt.

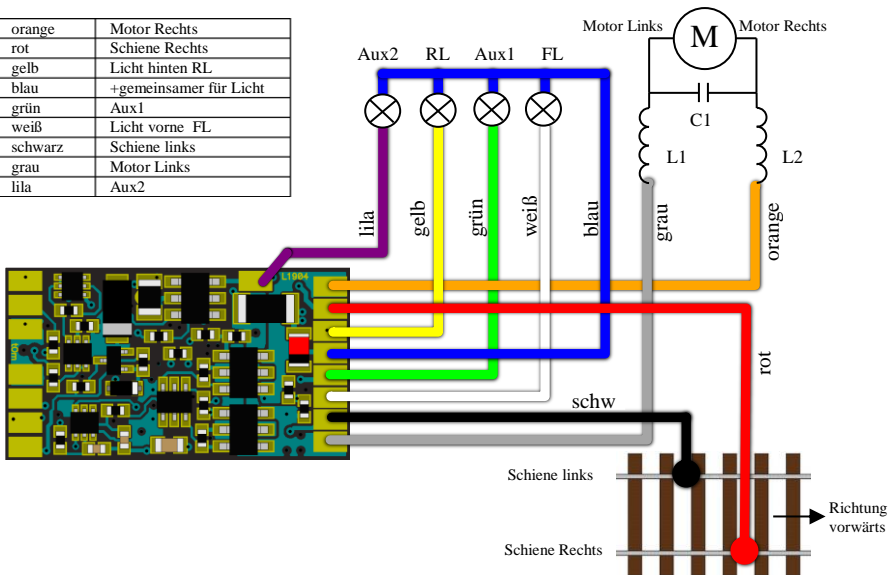
Der NEXT18-Stecker hat die Möglichkeit der Umkehrung. In diesem Fall werden die Fahrtrichtung und die Richtungslichter umgekehrt, ohne dass der Decoder beschädigt werden kann.

Der Decoder mit NEM651- Anschluss ist zwar dieser Umkehrung tolerant, aber funktioniert damit überhaupt nicht.

Bei einem NEM652- Anschluss muss der orangefarbene Draht mit Pin 1 des Steckverbinders der Lokomotiven Basisplatte übereinstimmen. Der Reverse- Einsatz beschädigt den Decoder nicht, der Motor dreht sich anders mherum und die Beleuchtung wird nicht funktionieren.

Bei älteren Lokomotiven, die nicht für die Digitalisierung bereit sind, kann man einen Decoder ohne einen Steckverbinder installieren, der nur Kabel enthält. Die Farbe der Litzen hat eine wichtige Bedeutung, die Verbindungen müssen, wie in der folgenden Zeichnung dargestellt, genau eingehalten angeschlossen werden.

orange	Motor Rechts
rot	Schiene Rechts
gelb	Licht hinten RL
blau	+gemeinsamer für Licht
grün	Aux1
weiß	Licht vorne FL
schwarz	Schiene links
grau	Motor Links
lila	Aux2



Bestimmen Sie zuerst die Vorwärtsrichtung der Lokomotive. Die Verbindungen zwischen dem Motor und den Kollektoren werden gelöst, wodurch die Korrelation zwischen den Motorklemmen und der linken / rechten Schiene erhalten bleibt. Wenn vorhanden, darf

der Kondensator C1 an den Motorklemmen 47nF nicht überschreiten. Wenn keine Entstörspulen vorhanden sind, entfällt auch der Kondensator C1, und wenn zwischen den Motorklemmen und dem metallischen Chassis des Motors Kondensatoren angebracht sind, entfallen auch diese. Löten Sie die roten / schwarzen Drähte an den Stromabnehmer an, und passen Sie auf, dass die orange / grauen Drähte an den Motorklemmen sich nach vorne (rechts / links) entsprechend der Fahrtrichtung eines konventionellen Traforeglers richten.

Beleuchtung oder andere Hilfsverbraucher verbinden die blaue (Common +) - Leitung und die Litzen, den dem gewünschten Ausgang entspricht (FL, RL, Aux1, Aux2 usw.). Wenn diese Verbraucher polarisiert sind (z. B. LEDs), achten Sie auf die Polarität: Die positive Klemme (Anode) wird am Common + angekoppelt und die negative Klemme (Catod) am gewünschten Ausgang. Die LEDs werden, notwendigerweise, mit einem Widerstand von 1-33K verbunden, der die Strombegrenzung je nach gewünschter maximaler Lichtstärke vornimmt..

## 8. Inbetriebnahme

Überprüfen Sie vor dem Anschließen an die digitalen Steuerstation die Richtigkeit der Anschlüsse der Decoder-Installation. Stellen Sie sicher, dass keine Kurzschlüsse oder fehlerhaften Verbindungen vorhanden sind. Stellen Sie sicher, dass die Verbindungsdrähte nicht mit den Zahnrädern oder anderen beweglichen Teilen in Kontakt kommen.

Bei der ersten Inbetriebnahme des Decoders wird empfohlen, einen Reset durchzuführen, indem Sie in CV8 den Wert 8 eingeben, um sicherzustellen, dass Sie mit den Werkseinstellungen der Konfigurationsvariablen beginnen.





Stellen Sie die Adresse der in CV1 gespeicherten Lokomotive ein, wobei der Werksstandardwert 3 ist, oder legen Sie eine Erweiterung, wie es in Kapitel 9 beschrieben ist, fest. Wenn Sie eine solche Adresse verwenden möchten, sollten Sie diese Adresse erst eingeben, nachdem alle anderen Parameter festgelegt wurden (endgültig) und testen Sie den Decoder zunächst mit der primären Adresse.

## 9. Die Adresse des Decoders

Die Primäre (Kurzadresse) des Decoders befindet sich in CV1. Sie kann die Werte im Bereich 1-127 annehmen. Der Werksstandardwert ist 3 (Kurzer Adressierungsmodus, Bit 5 von CV29 ist 0).



Wenn wir CV1 schreiben, wird die ursprüngliche Adresse automatisch gelöscht und die erweiterte Adresse wird automatisch deaktiviert.

Wenn 128 Kurzadressen nicht ausreichend sind, kann man eine erweiterte Adresse in CV17 (MSB) und CV18 (LSB) verwenden.



Der Decoder antwortet nur auf eine der Adressen (Primäre/Kurze oder erweiterte Adresse).

Wenn eine erweiterte Adressierung erforderlich ist, muss der Adressierungsmodus in der Konfiguration CV des Decoders geändert werden (Bit 5 von CV29). Wenn Sie den Bit Wert von CV29 in 1 ändern, wird der Modus für erweiterte Adressierung aktiviert, und der Decoder reagiert auf die in CV17 und CV18 angegebene erweiterte Adresse.

CV17 kann Werte im Bereich von 192 bis 231 annehmen und CV18 im Bereich von 0 bis 255. Die erweiterte Adresse erhält man mit der Formel:

Erweiterte Adresse:  $(\text{Wert von CV17}-192)*256 + \text{Wert von CV18}$

Die erweiterte Adressen werden mit dem folgenden Algorithmus berechnet (in unserem Beispiel betrachten wir die lange Adresse 2000):

- teilen Sie die gewünschte erweiterte Adresse mit 256 (in unserem Beispiel  $2000/256 = 7$ , Rest = 208)
- addiere 192 zum Ergebnis und programmiere es in CV17 ( $7 + 192 = 199$ , programmiere den Wert von 199 in CV17)
- programmieren Sie den Wert der restlichen Division in CV18 (programmieren Sie den Wert von 208 in CV18).

Nach dem Programmieren von CV29, CV17 und CV18 auf die genannten Werte kann auf den Decoder mit der erweiterte Adresse 2000 zugegriffen werden. Um wieder auf Kurzadressierung umzuschalten, muss das Bit5 von CV29 deaktiviert werden.

Somit können wir aus 10239 (1-10239) verschiedenen Adressen wählen.

Für Züge, die mehrere Decoder enthalten, kann man auch eine Konsist (Mehrfachtraktions) Adresse herstellen. Diese Konsist (Mehrfachtraktions) Adresse, die aus CV19 besteht und auf die alle Decoder antworten werden, wird für den Zug identisch sein. Somit werden die Steuer - und Geschwindigkeitsbefehle an alle Decoder gesendet, die dieselbe Konsist (Mehrfachtraktions) Adresse haben.

Wenn dieser Konsist (Mehrfachtraktions) Modus aktiviert ist (CV19 enthält einen anderen Wert als Null), führt der Decoder die in CV21 (F1-F8) und CV22 (F0F, F0R, F12-F9) definierten Funktionen (Die genaue Bitkorrespondenz entnehmen Sie bitte der CV-Tabelle) nur in dem Fall aus, wenn diese an die Konsist (Mehrfachtraktions) Adresse gesendet worden sind. Alle anderen Funktionen werden nur ausgeführt, wenn sie an die Basisadresse (kurz oder lang) gesendet werden.

Die in CV21 und CV22 deklarierten Funktionen, aber auch die Steuer- und Geschwindigkeitsbefehle, werden nicht ausgeführt, wenn sie an die Basisadresse übertragen werden, wenn der Konsist Modus aktiviert ist.



Im Konsist (Mehrfachtraktions) Modus können nur die Funktionen F0, F1-F12 verwendet werden. Die Anzahl der Fahrstufen muß im Decoder der zu steuernden Loks in CV 29 mit der Anzahl in der Zentrale festgelegten Fahrstufen sowohl für die Basis- als auch die Konsist (Mehrfachtraktions) Adresse übereinstimmen.

Die Konsist (Mehrfachtraktions) Adresse der CV19 kann Werte im Bereich von 1-127 für die Normalrichtung oder 128-255 für die Rückwärtsrichtung annehmen. Bit7 / CV19 repräsentiert somit die Richtung in Konsist (Mehrfachtraktions) Modus. Wenn CV19 = 0 ist, wird die Konsist (Mehrfachtraktions) Adresse deaktiviert.

Aus CV21 (F1-F8) und CV22 (F0F, F0R, F12-F9) können die Funktionen bestimmt werden, die für die Konsist (Mehrfachtraktions) Adresse aktiv sind. Für den Wert 0 eines einzelnen Bits ist die Funktion nur mit der individuellen Adresse aktiv, für den Wert 1 ist die Funktion nur mit der Konsist (Mehrfachtraktions) Adresse aktiv.

Wenn wir beispielsweise F0 in beiden Richtungen und F3, F4 mit der Konsist (Mehrfachtraktions) Adresse verwenden möchten, schreiben wir in CV21 = 12 (Dezimal) (oder 00001100 in binär) und in CV22 = 3 (Dezimal) (oder 00000011 in binär).



Für die Dezimal-Binär-Umwandlung und erweiterte Adressberechnung lesen Sie bitte Kapitel 27 Anhang Bits und Bytes.

---

## 10. Beschreibung der Charaktereigenschaften beim Fahren

In diesem Kapitel werden die Grundlagen für die Festlegung von Mindest-, Durchschnitt- und Höchstgeschwindigkeit und die Merkmale der Beschleunigung und Verzögerung der Lokomotive beschrieben.

CV2-Motordrehzahl bei niedrigster Drehzahlstufe.

CV5 Motordrehzahl bei höchster Drehzahlstufe.

CV6 Motordrehzahl bei mittlerer Drehzahl

CV3-Beschleunigungsrate

CV4-Verzögerungsrate

Für eine optimale Motorsteuerung wird empfohlen, Decoder mit 128 Geschwindigkeiten (in DCC) zu benutzen. Wenn dies nicht möglich ist, akzeptiert der Decoder auch DCC-Befehle mit 28- oder 14 Geschwindigkeiten, aber die Geschwindigkeitseinstellung erfolgt in größeren Sprüngen und wird damit gröber.

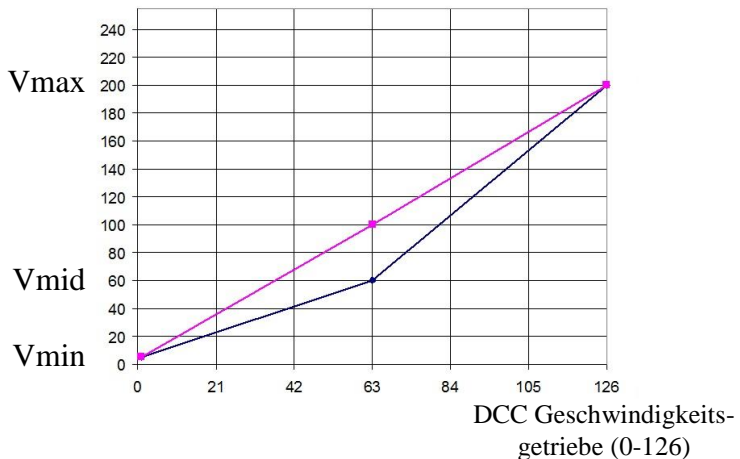
Die Motordrehzahl ist auf 255 innere Geschwindigkeitsstufen eingestellt. Die Korrelation zwischen der DCC-Stufe und den inneren Geschwindigkeitsstufe kann man auf zwei Arten herstellen.

### 10.1. Lineare Geschwindigkeitseinstellung an 3 Punkten

Die minimale CV2-Geschwindigkeit und die maximale CV5-Geschwindigkeit sind die Grenzen, zwischen denen die Motordrehzahl zwischen der ersten und dem letzten DCC-Fahrstufe eingestellt werden kann. Die Motordrehzahl wird entlang zweier durch  $V_{min}$ - $V_{mid}$  bzw.  $V_{mid}$ - $V_{max}$  abgegrenzter Linien linear bestimmt. Normalerweise wird das erste gerade Segment mit einer geringeren Steigung ausgewählt, um eine Feinabstimmung der niedrigeren Geschwindigkeiten zu erhalten. Dies erfolgt, indem für

Vmid ein Wert gewählt wird, der unter den Durchschnittswert der minimalen und maximalen Geschwindigkeiten liegt. Spezieller linearer Modus: wenn Vmid auf 0 gesetzt ist, die Geschwindigkeit wird zwischen Vmin und Vmax linear sein, ohne einen Bremspunkt in den Kennlinien.

innere Geschwindigkeits-  
getriebe 0-255



## 10.2. Tabellarische Geschwindigkeitseinstellung in 28 Stufen (CV67 ... CV94)

Die Einstellung der Motordrehzahl basiert auf der im CV-Bereich enthaltenen Tabelle. Der Startdrehzahl entspricht der dem Wert aus CV67 und höchste Drehzahl dem Wert aus CV94. Durch die Auswahl der Werte aus der Tabelle kann eine beliebige Form für die Kennlinie der Motorgeschwindigkeit bestimmt werden..



Um ein realistisches Verhalten des Eisenbahnmodelles zu erreichen, haben wir die Möglichkeit, Beschleunigungs- und Verzögerungsraten einzustellen. In CV3 können wir die Beschleunigung und in CV4 die Motorverzögerung ändern.

Wenn die Beschleunigung oder Verzögerung je nach Fahrtrichtung unterschiedlich ist, kann CV152-153 für die umgekehrte Fahrtrichtung verwendet werden.

Solange CV152 / CV153 einen Nullwert hat, werden CV3 und CV4 für beide Fahrtrichtungen verwendet.

Wenn wir beispielsweise in CV152 einen anderen Wert als Null eingeben, wird dies für die Rückwärtsbeschleunigung verwendet. CV3 wird weiterhin für die Vorwärtsbeschleunigung verwendet.

CV153 hat das gleiche Verhalten für die Rückwärtsverzögerung.

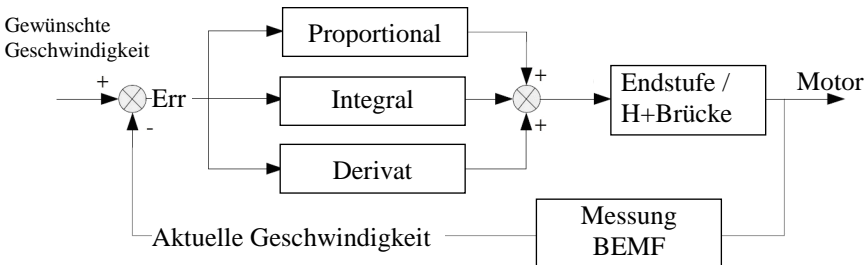
## 11. Motorsteuerung

Die Decoder der Lokommander II - Familie implementierten eine PID-Einstellschleife des Motors, die die gemessene Gegen-EMK des Motors verwendet.

Diese Funktion wird allgemein als "Lastkompensation" bezeichnet und kann ab Bit0 von CV60 aktiviert oder deaktiviert werden (Werkswert Bit0 = 1, d.h. PID-Regler ist aktiv). Der Motor ist in einer der Diagonalen einer H-Brücke (bestehend aus 4 FET-Transistoren) eingebunden, die Regelung erfolgt über die andere Diagonale. Der Regelbefehl wird vom Mikrocontroller im Decoder bereitgestellt, wobei die Impulsfrequenzmodulation mit fester Frequenz (PWM= pulse width modulation) und variabler Füllfaktor verwendet wird.

Die PWM-Signalfrequenz beträgt 16/32 kHz und kann in Bit 7 des CV60 eingestellt werden. Die Werkeinstellung des Wertes ist bit7 = 0, dh dass es den Frequenz von 32kHz entspricht. Der Motor wird in PWM-Impulsen gesteuert, unabhängig davon, ob der PID-Regler aktiviert ist oder nicht.

Der PID-Regler ist gemäß dem folgenden Block-Schema implementiert:

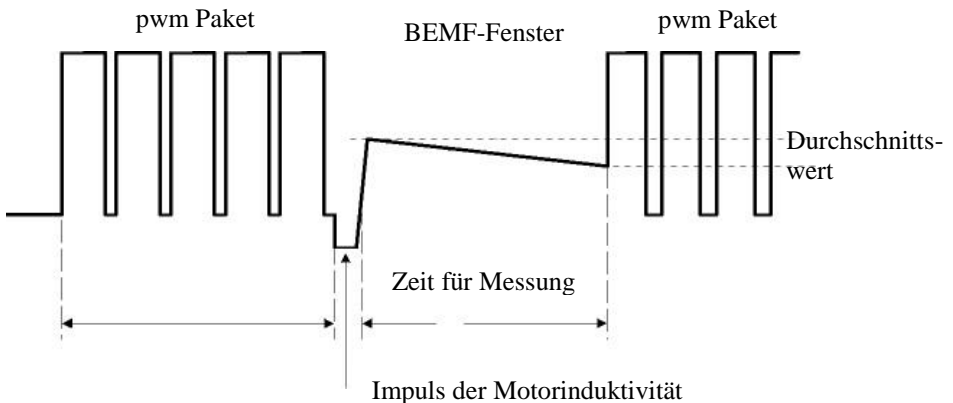




Das Referenzsignal (gewünschte Geschwindigkeit) wird permanent mit der aktuellen Geschwindigkeit verglichen und das resultierende Fehlersignal (Err) wird vom PID-Regler verarbeitet, der auf die Motorsteuerbasis wirkt und den PWM-Signalfüllfaktor so modifiziert, dass der Fehler, die Differenz zwischen den gewünschten- und den aktuellen Geschwindigkeit, minimal ist.

Um die aktuelle Geschwindigkeit zu bestimmen, wird für kurze Zeit die Versorgung des Motors unterbrochen (zu diesem Zeitpunkt ist er ein Generator) und die erzeugte elektromotorische Spannung (BEMF) wird gemessen. Diese Spannung ist direkt proportional zur Drehzahl des Motors und wird mit der gewünschten Drehzahl verglichen, um das Fehlersignal zu erhalten.

Die Periode in dem der Motor nicht versorgt wird, wird als BEMF-Fenster bezeichnet. Die zu häufige Verwendung eines BEMF-Fensters hat auch Nachteile, der Motor verliert in seiner Leistung. Aus diesem Gesichtspunkt ist es daher wünschenswert, so wenig wie möglich und umso schneller den BEMF des Motors zu messen. Der Motoraufbau erfordert jedoch einige Zeit für das BEMF-Fenster, das kann man nicht beliebig minimieren.





Wenn die Stromversorgung aufgrund der Induktivität des Motors unterbrochen wird, wird an seinen Klemmen ein Impuls entstehen, der die Messung der BEMF Spannung beeinträchtigt. Daher wird die Messung nach einer Verzögerung durchgeführt (BEMF delay). Die Breite dieses Impulses (implizit die erforderliche Wartezeit) hängt von der Konstruktion des Motors ab: Die Leistungsmotoren (5 oder mehr Pole) haben eine relativ kleine Impulsbreite im Vergleich zu Motoren der älteren Generation (mit 3 Polen).

Während der BEMF-Messung wird der Motor nicht mit elektrischer Energie versorgt und verliert aufgrund von mechanischer Belastung (Zahnräder, Masse der Lokomotiven, Anzahl der Waggone der Lokomotive usw.) an Geschwindigkeit, wie die Abwärtsneigung in der vorherigen Illustration zeigt. Um einen korrekten BEMF-Wert (oder die aktuelle Geschwindigkeit) zu erhalten, sollten die Messungen mehrmals durchgeführt und vermittelt werden.

Folglich ist die Anzahl der PWM-Pakete, nach denen ein BEMF-Fenster eingefügt wird, ebenfalls variabel.

Die Werkseinstellungen gewährleisten in den meisten Anwendungen einen ordnungsgemäßen Betrieb. Für den optimalen Betrieb in eine bestimmte Lokomotive werden jedoch die unten beschriebenen Einstellungen empfohlen.

Der Algorithmus, der zu den Motorsteuerung verwendet wird kann aus CV9 ausgewählt werden. Der Standardwert ist 3. Mit diesem Wert arbeitet der Decoder bei den meisten Lokomotivenmodellen optimal und gewährleistet einen reibungslosen Lauf ohne Sprünge für alle Geschwindigkeitsstufen (dabei wird die Verwendung von 128 Geschwindigkeitsstufen) empfohlen, um optimale BEMF/Lastkompensationsergebnisse zu erzielen. Standardwerte (CV9 = 0 bis 8) entsprechen einer Reihe von internen Steuerungsparametern, die dem Benutzer im Standardmodus



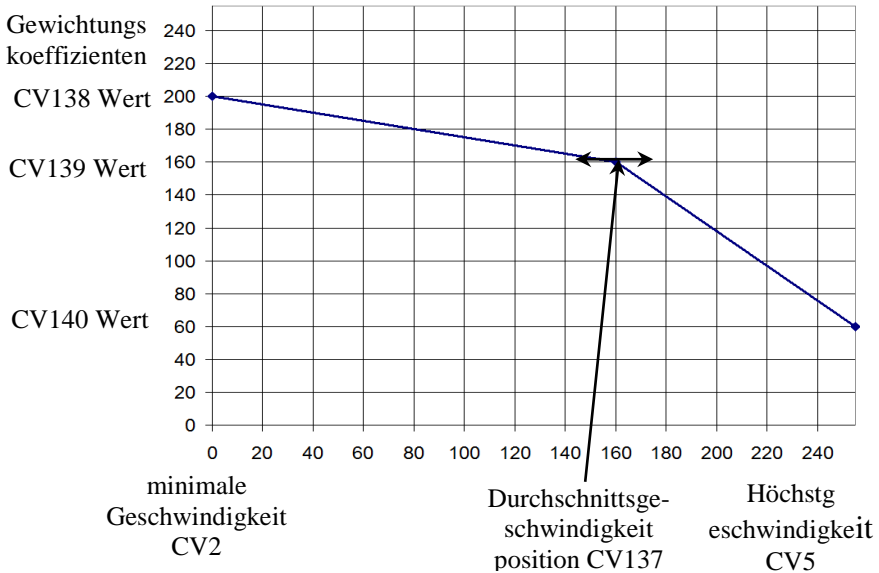
nicht zur Verfügung stehen. Für geringe Trägheit (Faulhaber-Motoren, kleine Lokomotiven usw.) werden Werte von 0,1,2 empfohlen. Die Werte 6,7 und 8 können bei Lokomotiven mit hoher Trägheit (schwere Lokomotiven, große Motoren) verwendet werden.

Für allgemeine Motortypen werden Durchschnittswerte von 3, 4 und 5 verwendet: Durch Auswahl eines Standardsatzes kann der Benutzer nun auf die PID-Reglerkoeffizienten (CV61,62,63) und einen neuen Parametersatz zugreifen, die in der zweiten Generation eingeführt werden: Kompensationskoeffizienten der Lastkompensation (CV137,138,139,140). In der Praxis können diese Gewichtungskoeffizienten abhängig von der Geschwindigkeit der Lokomotive bestimmen, wie stark die Lastkompensation ist. Die Charakteristik der Lastkompensation wird durch zwei Segmente mit einer negativen Neigung bestimmt, das erste zwischen  $V_{min}$  (CV2) und  $V_{mid}$  (CV138), das zweite zwischen  $V_{mid}$  (CV138) und  $V_{max}$  (CV5).

CV138 setzt den Lastkompensationskoeffizienten auf die minimale Geschwindigkeit (in CV2 definierte Geschwindigkeit) und CV140 auf die maximale Geschwindigkeit (in CV5 definierte Geschwindigkeit). Die Durchschnittsgeschwindigkeit ist in CV137 definiert, sie kann von der in CV6 definierten Durchschnittsgeschwindigkeit abweichen, und der Wert des Lastkompensationskoeffizienten bei der in CV137 definierten Geschwindigkeit wird in CV139 eingestellt. Das Maximalgewicht kann man bei 255 der CVs 138/139/140 erhalten. Praktisch bei hohen Geschwindigkeiten spielt die Lastkompensation nichtmehr so eine bedeutende Rolle wie bei niedrigen (und extrem niedrigen), sodass die CV140 ohne Motorlaufprobleme senkenden Tendenz aufweisen kann.

Das Experimentieren mit den Koeffizienten der Gewichtskompensation CV138 bei Mindestgeschwindigkeit und das Ändern der Durchschnittsgeschwindigkeit (CV137) und des CV 139-

Werts führt zu sehr guten Ergebnissen, auch für Motoren mit Problemen, die die Standardalgorithmen für Einstellung verwenden (zB 3, CV9 = 3) ohne der PID-Parametern (CV61,62,63) zu verändern.



Wenn Sie manuell auf den Parametersatz des Setup-Algorithmus zugreifen möchten, legen Sie CV9 = 9 fest. Auf diese Weise haben wir Zugriff auf die folgenden CVs:

CV128: Anzahl der PWM-Pakete, nach denen ein BEMF-Fenster eingefügt wird. Der Fabrikwert ist 1, sein Wachstum ist nur für größere Motoren mit hohem Trägheitsmoment gerechtfertigt. Der Bereich der numerischen Werte ist auf maximal 4 begrenzt.

CV130 BEMF-Messlücke: Ziel ist es, die BEMF-Messung um den Impuls zu verzögern, der von der Induktivität des Motors nach dem Trennen der Stromversorgung erzeugt wird. Der werkseitige Wert ist 6. Ein zu niedriger Wert hat katastrophale Auswirkungen auf die BEMF-Messung und wird durch den vom Motor erzeugten Impuls "überflutet". Bei mehrpoligen Leistungsmotoren (wie beispielsweise dem Typ Faulhaber oder Maxon), bei denen die Induktivität der Rotoren klein ist, kann dieser Spalt verringert werden. Motoren von schlechter Qualität (wie die Piko Modelle mit 3-Pole der Hobby-Lokomotiven Kategorie) erfordern einen höheren Offset, um BEMF-Messungen an einem stabilen / sauberen Teil der erzeugten Spannung durchzuführen.

CV129 Anzahl der Vermittler bei der BEMF-Messung. Der werkseitige Wert ist 6. Bei Motoren mit besserer Qualität und mehreren Polen kann die Anzahl der Vermittler sinken. Über 10 Vermittlern gibt es praktisch keinen Wertzuwachs.

CV95 PID-Fehlerlimitation sorgt für vollständige Zeitbegrenzung im PID-Kreis, ohne dessen Antwortzeit zu verkürzen. Werte sind in eine Skala von 1-10 möglich. Ein zu niedriger Wert führt zum Verlust der Motorleistung und zum ruckartigen Fahren, aber zu viel kann zu Instabilität der Einstellungsschleife und zu ausgeprägtem Motorgeräusch führen.

Im Folgenden präsentieren wir die Tabelle mit den Werten des Parametersatzes entsprechend dem CV9-Wert. In der Tabelle in Zeile 9 sind die für die Verwendung von Modus 9 akzeptierten numerischen Intervallen angegeben: Die Einstellung von Außenwerten verursacht keine Probleme, der Decoder begrenzt die in den CVs eingegebenen Werte nur in den angegebenen Intervallen.



CV9	CV128 Nummer der pwm Pakete	CV129 Nummer der BEMF Vermittler	CV130 offset, BEMF Messung	Notiz
0	1	4	1	
1	1	4	2	
2	1	6	2	
3	1	6	2	Default
4	2	4	1	
5	2	4	2	
6	2	6	2	
7	2	6	4	
8	2	8	6	
9	1-4	1-10	1-12	

Tabelle 2

## **12. Kontrollierte Stopps, gezielte Anhalten**

### **12.1. Stopp bei konstantem Distanz (CBD)**

Durch das gezielte Anhalten kann die Lokomotive angehalten werden, wenn ein Befehl für eine festgelegte Strecke unabhängig von der Fahrgeschwindigkeit empfangen wird. Das Stoppen kann durch 3 Faktoren ausgelöst werden:

- Eintritt in den Sektor mit asymmetrischer DCC (ABC) Empfang -Semantik Cap.13.2.
- Sektoreingang mit DC-Signal Cap. 15.3.
- Empfangen eines Befehls bei der Nullgeschwindigkeit

Das Stoppen der CBD bei Empfang eines Nulldrehzahlbefehls wird von Bit7-CV27 = 1 aktiviert. Es gibt zwei Möglichkeiten um die Stopps auf ein kontrollierte Distanz machen zu können:

#### **12.1.1. Stopp bei fester Verzögerung**

Nachdem Erhalt der Stopfbefehle fährt die Lokomotive eine, mit der aktuellen Geschwindigkeit berechnete, Strecke und stoppt dann mit der in CV64 (CV150) eingestellten Verzögerung. Die Laufzeit bei Anfangsgeschwindigkeit kann mit eine in CV65 (CV151) eingestellte variable Verzögerung durch die Verzögerungsformel =  $CV65 * 8ms$  ergänzt werden.

#### **12.1.2. Stopp bei variabler Verzögerung**

Nachdem Erhalt des Stopfbefehls stoppt die Lokomotive mit der berechneten Verzögerung, der anhand der Geschwindigkeits ausgerechnet wird, mit der die Lokomotive in dem Zeitpunkt des

Empfangens des Stoppbefehls gelaufen war, und des mit CV153 (CV161) eingestellten Bremsweges.

Wenn der CV153 den Nullwert beinhaltet (Anfangswert), verwenden Sie den festen Verzögerungsstopp in CV64. In dem Fall, wenn CV64 auch einen Nullwert beinhaltet, wird der Offline-Abstand deaktiviert. Wenn beide Werte andere Werte als Null haben, hat der in CV153 (CV161) eingestellte Variable, die bei variabler Verzögerung verwendet wurde, Priorität.

Alle Parameter eines Stopps können nach Fahrtrichtung unterschiedlich dargestellt werden. Es sind zwei CVs, eine für jede Richtung. Wenn die umgekehrte CV Null ist, wird der nach vorne gerichtete Wert für beide Richtungen verwendet.

Das Stoppen auf einer kontrollierter Distanz wird durch "Shunting" (F3) oder CBD-OFF (F5) verhindert.

## **12.2. Erkennung des asymmetrischen DCC-Signals (Lenz ABC)**

Das asymmetrische DCC-Signal ermöglicht das genaue Anhalten vor Signalen oder in Stationen und das Schalten in die entgegengesetzte Richtung. Über die BM1- und BM2-Module, die das Bremsen vor dem Signal erzeugen, erhält der Decoder der Lokomotive, der Fahrtrichtung entsprechend, Informationen über den Zustand des Signals. Es können zwei verschiedene Informationen übermittelt werden: "Stop" oder "Slow Approach"("langsame Annäherung").

Beim Empfang des "Stopp" -Befehls leitet die Lokomotive die kontrollierte Abstandskontrolle (Kapitel 12.1) ein, oder wenn sie deaktiviert ist, stoppt sie mit der CV4(CV149)-Verzögerung.



Beim Empfang des Befehls "Langsame Annäherung" ("Slow Approach") wird die Geschwindigkeit auf den in CV143 (CV163) eingestellten Wert verringert.

Die Aktivierung von ABC erfolgt aus der CV27:

- Bit0 = 1 Ermöglicht die Erkennung des ABC-Signals, wenn die rechte Schiene positiver ist (geht vorwärts)
- Bit1 = 1 Ermöglicht die Erkennung des ABC-Signals, wenn die linke Schiene positiver ist (Rückwärtsgang)

ABC funktioniert normalerweise nur in eine Richtung, die Aktivierung für beide Richtungen ist jedoch möglich (außer "Push-Pull").

Die Empfindlichkeit der Erkennung der Spannungsdifferenzen zwischen den beiden Schienen kann von CV141 aus verändert werden. Wenn der Anfangswert keine guten Ergebnisse für die ABC-Erkennung liefert, kann der optimale Wert innerhalb des empfohlenen Bereichs von 8 bis 16 experimentell Neubestimmt werden. Ein zu niedriger Wert führt zu einer unerwünschten fehlerhaften Erkennung und ein zu hoher Wert macht die Erkennung umständlich oder sogar unmöglich.

## 12.3. Pendelsteuerung (push-pull Funktion)

Mit der "Push-Pull" -Funktion können Sie mit ihre Lokomotive wiederholt eine Route zwischen zwei Endgeräten durchlaufen. Das Anhalten und Ändern der Fahrtrichtung erfolgt, wenn ABC-Befehle an Endstationen empfangen werden.

Von DCC-Befehlen werden nur die Fahrgeschwindigkeit und eventuell die aktiven Funktionen eingestellt. Es gibt zwei Möglichkeiten was man wählen kann:

### **12.3.1. Ohne Zwischenstopps**

Die "Push-Pull" -Funktion ohne Zwischenstopps erfordert zwei separate Abschnitte am Ende der Route, die ein ABC-" Stopp " Signal erzeugen, das der Richtung entspricht, in der sich die Lokomotive nähert (die positivere rechte Schiene). Die im Terminal angekommene Lokomotive wird anhalten, kehrt die Richtung (einschließlich Richtungslichter) um und startet nach der Wartezeit in die neue Richtung. Die Aktivierung erfolgt über Bit4 (CV122) = 1. In dem CV142 können Sie die Zeit des Stillstandes ändern. Es gibt möglicherweise ABC-Sektoren "Langsame Annäherung" auf dem Weg, wo es langsamer wird.

### **12.3.2. Mit Zwischenstopps**

Die "Push-Pull" Funktion mit Zwischenstopps erfordert zwei getrennte Abschnitte am Ende der Route, die ein ABC-Signal "Slow Approach"/"Langsame Annäherung" entsprechend der Richtung erzeugen, in der sich die Lokomotive nähert. In den Zwischensektoren, in denen das Anhalten vorgesehen ist, wird ein ABC "Stop" Signal aktiviert, das natürlich der Richtung entspricht, in der sich die Lokomotive nähert. Der Dauer des Zwischenstopps wird bis dahin gerechnet, bis das ABC "Stop" Signal aufhört aktiv zu sein.

Die Aktivierung erfolgt aus dem Bit5 (CV122) = 1. Ab CV142 können Sie die Dauer der Standzeit (in Endstationen) nach Ihrem Wunsch ändern.

Für den "Push-Pull" -Betrieb ist die Erkennung des ABC-Signals in CV27 für eine der Richtungen erforderlich (Kapitel 12.2).



Es ist nicht erlaubt, ABC für eine beliebige Richtung zu aktivieren, denn dies führt zu einem fehlerhaften Betrieb des "Push-Pull" -Modus! Die gleichzeitige Aktivierung von Bit4, Bit5 (CV122) ist nicht zulässig!

Es wird empfohlen, eine der " Stopp bei konstantem Distanz (CBD)" Methoden zu aktivieren, um sicherzustellen, dass sich die Lokomotive, unabhängig von der Fahrgeschwindigkeit, immer in derselben Position befinden wird (Kapitel 12.1).

## 13.Funktionsausgänge

Funktionsausgänge können verschiedene Verbraucher wie LEDs, Glühlampen, Rauchgenerator, elektromagnetische Koppler usw. steuern.

Die tOm Decoder verfügen über zwei Arten von Leistungs- oder Logikausgängen. Leistungsausgänge verfügen über einen Transistor, der zum Zeitpunkt der Aktivierung an den (-) - Ausgang durchschaltet. Somit verbinden Sie zu aktivierende Verbraucher zwischen Ausgang und + Vcc. Logikausgänge haben eine Spannung von + 5 V, wenn sie aktiv und 0 V, wenn sie inaktiv sind. Die Logikausgänge dürfen den maximalen Strom von 5 mA nicht überschreiten. Andernfalls besteht die Gefahr, dass der Decoder beschädigt wird. Ein Logikausgang kann um mit 1-2 LEDs die Steuerung für Strombegrenzungswiderstand verwendet werden oder mit einen externen Transistor, um größere Lasten zu steuern.

Um die Anzahl der Ausgänge zu erhöhen, können Sie die SUSI-Schnittstelle löschen und als 2 logische Ausgänge verwenden. Sie sind werkseitig als logische Ausgänge konfiguriert. Um sie für das SUSI-Schnittstelle zu verwenden, müssen Bit 0 und 1 von CV122 auf 1 gesetzt werden.

Einige Decoder verfügen über eine größere Anzahl von Ausgängen als die über den Anschluss verfügbaren. Für diese müssen Sie zusätzliche Drähte an den (in den Zeichnungen markierten) Pads auflöten.

In der Tabelle 3. haben wir die Anzahl und den Typ der verfügbaren Ausgänge hervorgehoben, die für verschiedene Decodertypen geeignet sind.

	FL	RL	AUX1	AUX2	AUX3	AUX4	AUX5	AUX6	AUX7	AUX8	AUX9	AUX10
MICRO	P	P	O,P	O,P	O,P	O,P	O,L,S	O,L,S				
NEXT18	P	P	P	P	L,S	L,S						
MTC21	P	P	P	P	L	L	L,S	L,S				
Plux12	P	P	P	P	O,L,S	O,L,S						
Plux16	P	P	P	P	L,S	L,S						
Plux22	P	P	P	P	P	P	P	P	P	P	L,S	L,S

Tabelle 3.

P – Leistungsausgang

L – Logikausgang

O – optionelle Ausgang, die durch Anhängen eines zusätzlichen Drahtes erreichbar wird

S – gemeinsamer Ausgang mit SUSI

Für Decoder mit maximal 8 Ausgängen verwenden wir ein vereinfachtes Nachabbilden (Mapping), das sich leicht vom NMRA-Standard unterscheidet, der eine größere Flexibilität bietet (jeder Funktion kann jeden Ausgang ein Befehlen ausgeben).



	CV nr.	Anfangswert	AUX 6	AUX 5	AUX 4	AUX 3	AUX 2	AUX 1	FR	FL
F0f	33	1	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F0r	34	2	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F1f	35	1	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F1r	47	1	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F2	36	2	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F3	37	4	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F4	38	8	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F5	39	16	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F6	40	32	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F7	41	64	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F8	42	128	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F9	43	0	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F10	44	0	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F11	45	0	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F12	46	0	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)

Tabelle 4.

Für PLUX22-Decoder mit 10 Ausgängen wurde ein NMRA-Standardmapping verwendet.

	CV nr.	Anfangswert	AUX 10	AUX 9	AUX 8	AUX 7	AUX 6	AUX 5	AUX 4	AUX 3	AUX 2	AUX 1	FR	FL
F0f	33	1					Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F0r	34	2					Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F1f	35	4					Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F2	36	8					Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F3	37	16					Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F4	38	4		Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)			
F5	39	8		Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)			
F6	40	16		Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)			
F7	41	32		Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)			
F8	42	64		Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)			
F9	43	16	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)						
F10	44	32	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)						
F11	45	64	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)						
F12	46	126	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)						
F1r	47	4					Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)

Tabelle 5.



Mapping aus dem grauen Feldern sind inaktiv / unmöglich. Einstellungen sind nur innerhalb des weißen Feldes möglich (Werkseinstellungen sind mit Grün markiert).

Die Funktionen F0 (f=forward, vorwärts, r=reverse, rückwärts), F1 (f=forward, vorwärts, r=reverse, rückwärts), F2 und F3 können nur die FL-, FR- und AUX-Ausgänge 1, 2, 3, 4, 5, 6 steuern. Für NMRA-Kompatibilität wird die CV35 F1 Konfiguration für Vorwärts (vorwärts) und CV47 für Rückwärts (rückwärts) benutzt. Die Funktionen F4,5,6,7,8 können nur die Ausgänge AUX2,3,4,5,6,7,8 und 9 steuern, die Funktionen F9,10,11,12 können nur die AUX 5,6,7,8, 9 und 10 Ausgänge koordinieren.

Die PWM der 12 Ausgänge wird in CV48-59 eingestellt, wie es im Kapitel 26. dargestellt wird.

In CV112 können wir die Anstiegszeit beim Start einstellen. In CV113 kann man die Abfallzeit des an die Ausgänge angelegten PWM-Signals einstellen. ( FadeIN und FadeOUT). Diese Zeiten können in Schritten von 8 ms eingestellt werden und repräsentieren die Zeit, zu der der Ausgabe-PWM-Füllfaktor von 0 bis 255 erreicht oder umgekehrt.

Wenn durch CV48-59 ein Füllfaktor festlegt wird, der niedriger als der Maximalwert 255 ist, nehmen die Anstiegs- und Abfallzeiten proportional ab. Diese beiden Parameter sind für alle Ausgänge geltend. Diese Funktion ist nützlich, wenn Sie die langsame Anschaltung von Glühlampen simulieren möchten.

Wenn ein Ausgang mit einem Dauersignal (ohne PWM mit variablem Füllfaktor) in CV117 befohlen werden soll, setzen wir den Wert von 1 Bit entsprechend dem gewünschten Ausgang bzw. Ausgänge. Bei einem Lokommander II mit mehr als 8 Ausgängen kann der Befehl für die Ausgänge 9-12 über das CV185-Bit 0-3 eingestellt werden.



In der aktuellen Softwareversion können die Funktionen F0 (f / r), F1 (f / r) und F2-F12 auch einem oder mehreren der Ausgänge FL, FR, AUX1, ... AUX6 zugeordnet werden.

Wenn eine Funktion den Betrieb eines der Ausgänge unterbrechen soll, muss gemäß der Tabelle 6. das entsprechende Ausgangsbit in der, der Funktion entsprechenden, CV auf 1 gesetzt werden. Die Funktionen F0 und F1 können je nach Fahrtrichtung die Ausgänge FL, FR, AUX1, ... AUX6 sperren. Die CVs 166/168 stellen die Sperrung einiger Ausgänge ein, wenn sich die Lokomotive nach vorwärts bewegt, bzw. in den CVs 166/180 die Sperrung einiger Ausgänge, wenn sich die Lokomotive in Rückwärtsrichtung bewegt.

	CV nr.	Anfangswert	AUX 6	AUX 5	AUX 4	AUX 3	AUX 2	AUX 1	FR	FL
F0f	166	0	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F0r	167	0	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F1f	168	0	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F2	169	0	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F3	170	0	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F4	171	0	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F5	172	0	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F6	173	0	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F7	174	0	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F8	175	0	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F9	176	0	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F10	177	0	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F11	178	0	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F12	179	0	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)
F1r	180	0	Bit7 (128)	Bit6 (64)	Bit5 (32)	Bit4 (16)	Bit3 (8)	Bit2 (4)	Bit1 (2)	Bit0 (1)

Tabelle 6.

## 14. Decoder in Analog (DC) verwenden

Der Decoder ermöglicht den Betrieb der Lokomotive auch mit klassischen Geschwindigkeitsregler, die Gleichstrom liefern. Diese können zwei Arten haben: gefilterte und gepulste (PWM) Spannung. Um den Betrieb bei Gleichstrom zu ermöglichen, muss in Bit2 / CV29 der Wert "1" eingegeben werden. Mit CV13 und CV14 können wir einstellen, welche Funktion aktiviert werden sollen, wenn wir den Decoder mit Gleichstrom versorgen.

In der Tabelle 7. finden Sie die Bedeutung jedes Bits der beiden CVs. Wenn das Bit den Wert "1" hat, ist diese Funktion im Analogbetrieb aktiv.

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
CV13	F8	F7	F6	F5	F4	F3	F2	F1
CV14	F14	F13	F12	F11	F10	F9	RL	FL

Tabelle 7.

Es gibt zwei Arten im Falle der Gleichstromversorgung:

### 14.1. Analogmodus 1

Modus 1 kann mit Reglern verwendet werden, die eine gefilterte Dauerspannung liefern. Die Höhe der Spannung in den Schienen wird die gewünschte Geschwindigkeit einstellen und die Motorsteuerung wird über die PID-Schleife sichergestellt. Es wird auch bei niedrigen Geschwindigkeiten. Wenn die Maximalspannung plötzlich angelegt wird, erreicht der Motor mit der Beschleunigungsverhältnis in CV3 (CV148) die Höchstgeschwindigkeit.





Der Zusammenhang zwischen Schienenspannung und Fahrgeschwindigkeit wird linear in Abhängigkeit von 3 CVs gemacht.

CV145: Startschwelle. Wenn dieser Wert erreicht ist, wird der Motor gestartet

CV146: Standschwelle. Dieser Wert ist niedriger als die Startschwelle, bei dieser Spannung steht das Fahrzeug garantiert

CV147 maximale Geschwindigkeit, bei dieser Spannungswert in den Schienen erreicht es die maximale Geschwindigkeit

Der in diesen CVs eingeschriebene Wert wird, durch Multiplizieren der gewünschten Spannung mit 10, berechnet.

Bei einer maximalen Spannung von 14 V wird in CV146 beispielsweise 140 eingegeben.

Dieser Modus funktioniert nicht richtig mit PWM (Pulsvarianten) Reglern !

Um Modus 1 zu wählen, in CV164 muss man den Nullwert eingeben.

## **14.2. Analogmodus 2**

In diesem Modus wird der Motor durch Pulsvarianten(PWM) mit hohen Spannungswert von Frequenzimpuls gesteuert. Der PWM-Füllfaktor wird durch CV164 festgelegt. Bei einem Maximalwert von 255 wird praktisch die gesamte Schienenspannung auf den Motor angewendet.

Wenn ein niedrigerer Wert eingestellt wird, ist die an den Motor angelegte Spannung geringer als die in den Schienen (erlaubt den Einsatz von Motoren mit niedrigerer Nennspannung). Um Modus 2 auszuwählen, wird in CV164 in ein anderen Wert als Null eingegeben. Nur dieser Modus kann mit PWM verwendet werden.

### 14.3. Kontrollierte Stopp im DC Sektor

Dauerstrom kann auch in Kombination mit DCC verwendet werden, um die DC-Bremssektoren anzutreiben. Wenn also eine DCC-betriebene Lokomotive einen DC-Sektor (mit Gleichstrom versorgt) erreicht, stoppt sie, wenn die folgenden Bedingungen erfüllt sind:

Bit4 oder Bit5 in CV27 haben den Wert "1", Bit2-CV29 = 0 und die Schienenspannung ist höher als die in CV162 eingestellte Schwelle.

Die in CV162 eingestellte Schwelle (Standardwert 100 => 10V) ist nützlich, wenn Sie ein power-pack und die DC-Bremsfunktion verwenden möchten.

Befindet sich die Versorgungsleitung unterhalb der Schwelle, befinden wir uns im SPP-Modus und die Lokomotive stoppt nach Ablauf der in den CV123 eingestellten Zeit. Wenn die Spannung den Schwellenwert überschreitet, wird die Gleichstrombremsfunktion aktiviert und die Lokomotive hält in kontrollierter Entfernung an (siehe im Kapitel 12.1).

---

## 15. Bidirektionale Kommunikation (RailCom)

"Bidirektional" bedeutet, dass die Übertragung von Informationen unter dem DCC-Protokoll nicht nur für den Decoder gilt, sondern auch für die entgegengesetzte Richtung. So kann der Decoder Nachrichten senden, z. B. eine Bestätigung des Eingangs von Befehlen, die eigene Adresse, tatsächliche Geschwindigkeit, Innentemperatur, Last und andere Statusinformationen.

Im Prinzip der Funktion von RailCom basiert sich auf der Einführung durch die Leitstelle eines Fensters ("Unterbrechung") durch die Steuerstation am Ende jedes DCC-Pakets, bei der die Stromversorgung unterbrochen und die beiden Leitungen kurzgeschlossen werden. In diesen Fenstern senden die Decoder einige Bytes von Daten, die von den mit der Schaltung verbundenen Detektoren oder der Steuerstation empfangen werden.

Das Datenpaket ist in zwei Kanäle aufgeteilt. Auf dem ersten Kanal wird die Adresse (kurz, lang oder zusammengesetzt) des Decoders übertragen. Auf dem zweiten Kanal werden POM-Antworten geliefert, um CVs zu bearbeiten (Lese-, Schreibergebnis).

Die RailCom-Kommunikation kann über CV29-Bit3 deaktiviert werden (0 - RailCom inaktiv, 1 - RailCom aktiv). Die Kanäle 1 und 2 sind in CV28 Bit0 und Bit1 aktiviert.

## 16. Spezielle Funktionen

Wenn Sie die speziellen Funktionen nutzen möchten, erhalten Sie Informationen über:

- der Wert der internen Decodertemperatur
- die Qualität des empfangenen DCC-Signals
- die Anzahl der Stunden und Minuten des Betriebs
- die Zeitmarke (Uhrzeit), zu der die letzte Lokomotivenwartung durchgeführt wurde

Um die Werte dieser Parameter zu speichern, muss der nichtflüchtige Speicher (eeprom) des Decoders im für den Benutzer zugänglichen CV-Bereich aktiviert sein. Bit7 / CV122 aktiviert oder deaktiviert die Speicherfunktion (Bit7 = 0, Speicherfunktion deaktiviert, Bit7 = 1, Speicherfunktion aktiviert). Das Speichern der Momentanwerte erfolgt durch Aufruf der F5-Funktion von der Leitstelle (oder vom tOm-Programmiergerät aus).



Ohne den Aufruf von F5 (Ein und Aus) werden die Werte in den entsprechenden CVs nicht aktualisiert !!!

Die interne (gespeicherte) Temperatur des Decoders kann aus CV133 abgelesen werden. Die Temperatur wird in Grad Celsius angegeben.

Das Qualitätssignal des DCC-Signalempfangs (QoS = Quality of Signal, Signlasqualität) wird aus CV135 gelesen. Der Wert wird in Prozent (im Bereich 0-100%) angezeigt. Den minimalen QoS-Wert, der bei dem letzten Lesen des Decoders berechnet wurde, finden Sie

in CV136. Um den Mindestwert zurückzusetzen, geben Sie den CV136-Wert 100 [%] ein. (Rufen Sie vor dem Lesen die Speicherfunktion mit F5 On, F5 Off auf).

Die Anzahl der Stunden und Minuten des Betriebs werden aus den CV156, 157 und 158 wie folgt gelesen:

- Die Anzahl der im Betrieb verbrachten Minuten ist der aus CV156 gelesene Wert.
- Die Anzahl der Betriebsstunden ist die Summe der abgelesenen Werte aus CV157 und  $256 * \text{der abgelesenen Werte aus CV158}$ . (Rufen Sie vor dem Lesen die Speicherfunktion mit F5 On, F5 Off auf).

### **Wartungszeit:**

Der Decoder kann die Zeitsmarke der Lokomotivenwartung beibehalten und kann anzeigen, dass seit der letzten Wartung einige Stunden vergangen sind.

Die Aktivierung und Konfiguration dieser Funktion kann in CV154 wie folgt durchgeführt werden:

- |       |  |
|-------|--|
| Bit 0 | Wert 0, Wartungsfunktion ist nicht aktiv<br>Wert 1 ist die Wartungsfunktion aktiv  |
| Bit 1 | Wert 0, die Überschreitung vom Wartungsintervall wird in CV30 nicht gemeldet<br>Wert 1, der das Wartungsintervall überschreitet, wird in Bit 3 des CV30 signalisiert   |
| Bit 2 | Wert 0, Überschreiben des Wartungsintervalls wird durch die FL / RL-Leuchten nicht signalisiert<br>Wert 1, der das Wartungsintervall überschreitet, wird durch Ändern der FL / RL-Leuchten bei niedriger Frequenz signalisiert |

- Bit 3      Wert 0, 50% Wartungsüberschreibung wird nicht von FL /RL- Leuchten gemeldet  
Wert 1, der das Wartungsintervall um 50% überschreitet, wird durch Ändern der FL / RL-Leuchten mit einer höheren Frequenz signalisiert

Das Wartungsintervall wird in Stunden in CV155 angegeben. Der werkseitige Wert beträgt 40 Stunden. Der Wert kann vom Benutzer in einem Intervall zwischen 0-255 geändert werden. Nach dem Zurücksetzen ist der Decoderwert von CV155 40 (Stunden).

Der Zeitpunkt, zu dem die letzte Wartung durchgeführt wurde, kann aus den CV159 und 160 wie folgt abgelesen werden:

Die Anzahl der Wartungsstunden (bestätigt) ist die Summe des von CV159 Lesewert \* 256 und des von CV160 gelesenen Wert.

$$\text{Anz. Stun.} = (\text{Werts CV159}) + 256 * (\text{Werts CV160})$$

Zur Bestätigung der Wartung wird die sogenannte Pseudo-Programmierung verwendet: In CV8 wird der Wert 128 eingegeben (nicht äquivalent zu einem Decoder-Reset). Als Ergebnis dieser Operation wird die Wartungszeitmarke gespeichert und das neue Wartungsintervall wird aus diesem Zeitsmarke berechnet.



Wenn das Wartungsintervall mit dem Bit 3 von CV30 überschritten wurde, muss nach Bestätigung der Messung auch CV30 (auf 0) zurückgesetzt werden. Nur durch das Wartungsbestätigungsverfahren wird das CV30-Bit 3 nicht gelöscht.

## 17. Automatische Entkupplung

Der Lokommander II-Decoder ermöglicht die Verwendung jedes physikalischen Ausgangs zum Betrieb von elektromagnetischen Kopplern. Wenn ein Logikausgang gewählt wird, ist es notwendig, einen externen Transistor zu verwenden, da dessen Ausgang nicht genügend Strom liefert, um den Kupplung zu versorgen. Die Krois ® - und Roco ® -Kupplungen erfordern ein PWM-Signal mit einer hoher Frequenz, um eine Verbrennung der Spulen in der Konstruktion der Kupplung zu vermeiden. Die automatische Entkupplungsfunktion des Decoders liefert dieses Befehlssignal.



Die automatische Entkupplungsfunktion kann nur bei stehender Lokomotive aktiviert werden (nachdem die Lok angehalten wurde. Fahrstufe =0).

Die automatische Entkupplungsfunktion ist eine physische Funktion (nicht logisch, wie beim Regeln der Geschwindigkeit, Beschleunigung und Verlangsamung inaktivieren usw.), weil die Konfiguration folgendermaßen läuft:

Eine F-Funktion wird ausgewählt, die für die automatische Entkopplungsfunktion verwendet werden soll (dies kann eine Funktion sein, die auch für andere Befehle verwendet werden kann, z. B. für Ton/Sound).

Für die ausgewählte Funktion, CVs 33-47, wird die Zuordnung (mapping) des physikalischen Ausgangs gemacht, an den der elektrischer Kuppler angeschlossen ist (Wenn man beispielsweise die Funktion F8 für die automatische Entkupplung auswählt und die Elektromagnete des Kupplers an den physikalischen Ausgang Aux2 / der violetten Litze angeschlossen sind, gibt man in der CV43 den



Wert 8 ein, Folglich, wenn die Funktion F8 aktiviert ist, wird der physische Ausgang Aux2 damit aktiviert).

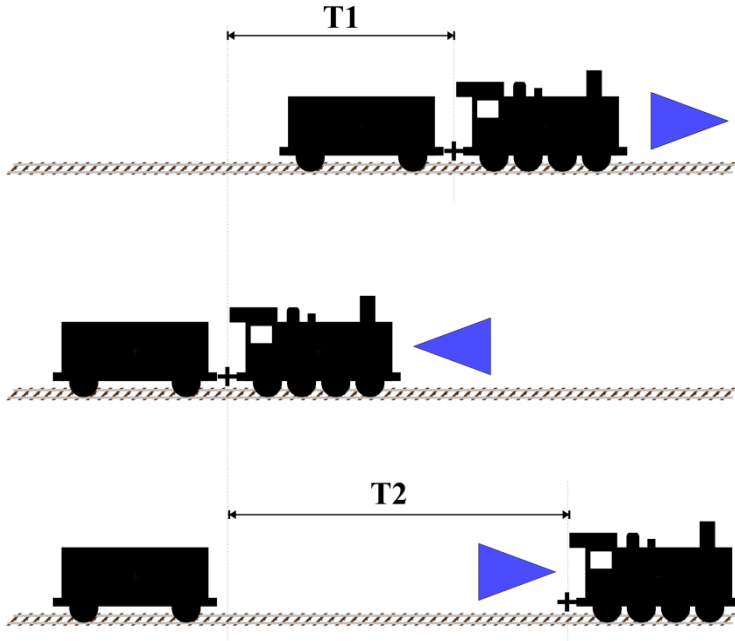
Um die elektromagnetische Entkuppung durchzuführen zu können, muss man es in CV118 angeben, denn auf diese wird die Entkuppungsfunktion für den Ausgang angewendet.

Wenn wir bei Aux2 sind, wir werden in CV118 den Wert 4 schreiben (in CV118 wird die Ausgangsnummer angegeben: für FL den Wert 1, für RL den Wert 2, für Aux1 den Wert 3 und für Aux2 den Wert 4).

Unter CV124 können wir die Art der verwendeten Kopplung (DC oder PWM) wie folgt auswählen: Bit0-0 Ausgang HF PWM; Bit0-1 Dauerausgang.

Von Bit1-CV124 können wir den Steuerungsmodus des Motors während des Abkoppelns auswählen. Somit wird mit bit1-0 der Motor sofort, mit eine plötzliche Bewegung, angesteuert; Bei Bit1-1 wird der Motor durch die PID-Schleife mit Verzögerungs- bzw. Beschleunigungswerten gemäß CV3-4 gesteuert.





Sobald wir die F-Logikfunktion auf den physischen Ausgang abgebildet haben (mapping), können wir die Konfigurationsvariablen ändern, um die automatische Entkopplungsfunktion zu optimieren. Für die stationäre Lokomotive (nach dem Stoppen) wird die Richtung gemäß den oben beschriebenen Abbildung eingestellt. Die maximale Fahrgeschwindigkeit während der automatischen Entkopplungsfunktion wird in CV121 eingestellt (Wert 0 bedeutet keine Bewegung, nur die elektromagnetische Kopplung wird funktionieren).

Beim Aufruf der Funktion wird die Lokomotive den Elektrokoppler aktivieren und fährt eine Zeitspanne T1 in die entgegengesetzte Richtung zur eingestellten Richtung, bevor die Funktion gewählt wird. Die Länge des Fahrtes kann durch die

Fahrgeschwindigkeit (CV121) und die Dauer des Rückwärtsfahrens (CV119) gesteuert werden. Nach dieser Bewegung stoppt die Lokomotive, ändert die Fahrtrichtung (geht damit in die gleiche Richtung wie vor dem Aufruf der Funktion) und fährt für eine Dauer von T2, wonach die Lokomotive die elektrischen Kupplungen abschaltet und deaktiviert. Um die Länge der Bewegung im direkten Sinne zu bestimmen, haben wir auch 2 Parameter, die Verfahrgeschwindigkeit (CV121) und die Zeit T2 (CV120). Mit diesen 2 Parametern können Sie die zurückgelegte Distanz direkt verringern oder erhöhen. Funktionen, die vor dem Wählen der Entkopplungsfunktion aktiviert werden, bleiben während der Entkopplung aktiv.

Die Entkopplungsfunktion wird aufgerufen, wenn die Funktion (EIN/ON) aktiviert ist und nach einem vollständigen Zyklus ausgeschaltet wird, auch wenn die F-Funktion nicht deaktiviert wurde.

Um die Funktion wieder einzuschalten, wird der AUS/OFF-Befehl und dann erneut ein EIN/ON-Befehl gesendet.



Die Polarität der elektromagnetischen Kopplungsdrähte ist wichtig. Wenn sie nicht richtig angeschlossen sind, wird die Bewegung (Hebung) umgekehrt verlaufen!

## 18. SUSI/ Locowire Schnittstelle

Sie können an die SUSI /Locowire Schnittstelle jeden Soundmodul oder Funktionsdecoder anschließen, der den Schnittstelle spezifikationen entspricht.

Für den Anschluss befinden sich 4 Stecker auf der Oberseite des Decoders (siehe Abbildung unten). Bei Versionen mit Plux16-, Plux22-, MTC21- und NEXT18-Steckverbindern sind diese Verbindungspunkte auf den Pins des Steckverbinders zu finden, und daher ist es keine zusätzliche Verdrahtung erforderlich. Diese Kontakte berücksichtigen die Reihenfolge / Bedeutung der SUSI- oder Lokowire-Schnittstelle. Wir empfehlen die Verwendung bestimmter Farben für die Litzen.



**Achtung!** Ein falscher Anschluss des SUSI / Lokowire-Moduls kann zu Schäden an diesem Modul führen.

### 18.1. Programmierung der SUSI-Module

Genauso wie die Lokomotivedecoder, können SUSI-Soundmodule durch Ändern einiger Betriebsparameter angepasst werden. Die Werte dieser Parameter werden in Konfigurationsvariablen (CVs) zwischen CV897 und CV1024 gespeichert. Das SUSI-Soundmodul wird über den Lokommander II-Decoder programmiert. Abhängig von der CV-Nummer erkennt der Lokommander II-Decoder, ob dieser CV von einem an das Decoder-Schnittstelle angeschlossenen SUSI-Modul geschrieben oder gelesen werden soll. Um die verschiedenen Konfigurationsvariablen des SUSI-Moduls zu programmieren, lesen Sie bitte das Konfigurationshandbuch des SUSI-Moduls.

Das Schreiben von CVs von SUSI-Modulen kann im PT-Modus und im PoM-Modus erfolgt werden. Dank einiger digitaler Systeme können CVs im Bereich 1-255 geschrieben und gelesen werden, deshalb wurde im Lokommander II-Decoder ein Mechanismus für diese digitalen Systeme implementiert, womit, mit Hilfe von zwei CVs, der Zugriff auf die CVs der SUSI-Module ermöglicht wird. CV126 wird als Anzeige (Index) und CV127 als Transport-CV verwendet.

In CV126 schreiben wir also die Differenz zwischen der Adresse der CVs, auf den wir zugreifen wollen, und 800. Beim Lesen oder Schreiben von CV127, werden wir die CV mit der Adresse  $800 + CV126$  lesen oder schreiben.

Beispiele:

- Wenn Sie in CV897 des SUSI-Moduls den Wert 1 eingeben möchten, müssen Sie in CV126 den Wert 97 ( $897-800 = 97$ ) und in CV127 den Wert 1 eingeben. Der Lokommander II-Decoder sendet nach Eingabe des Werts 1 in CV127 den Schreibbefehl mit dem Wert 1 an CV 897 über das SUSI-Schnittstelle an das Soundmodul (oder an den Funktionsdecoder).
- Wenn Sie den Inhalt von CV 902 von dem an die Lokommander II Decoder Schnittstelle angeschlossenen SUSI-Modul lesen möchten, geben Sie in CV126 den Wert 102 ( $902-800 = 102$ ) ein und lesen Sie den Wert von CV127. Dieser Wert entspricht dem in CV 902 des Soundmoduls (oder Funktionsdecoders) enthaltenen Wert, der an den Lokommander II-Decoder angeschlossen ist.

Das Lokowire-Schnittstelle erfordert keine Programmierung der Konfigurationsvariablen. Der Lokommander II-Decoder wird ab Werk mit der für SUSI-konfigurierten Schnittstelle ( $CV122 = 1$ ) ausgeliefert. Um die Lokowire-Schnittstelle zu aktivieren, muss man den Wert 0 ( $CV122 = 0$ ) eingeben.

## 19. Verwendung externer Kondensatoren oder von eines power pack

Bei einigen Gleisanlagen ist der Kontakt der Radsätze oder Schleifer mit der Schiene, aufgrund der Abnutzung der Schienen und Ablagerungen aus unterschiedlichen Materialien, unsicher. Sie verursachen Stromausfälle, die unstetes Fahren verursachen, insbesondere auf Weichenpassagen und bei niedriger Geschwindigkeit. Diese Nachteile können durch die Verwendung von Pufferkondensatoren (mit der Kapazität 220 uF / 25V oder für bessere Ergebnisse mit höheren Wert, über 2200 uF) oder unterbrechungsfreien SPP-Quellen behoben werden.

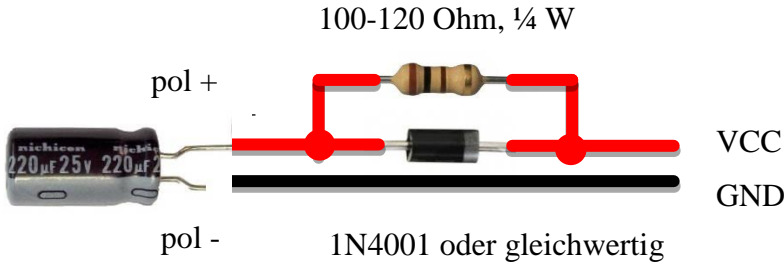
Zum Anschließen dieser Geräte verfügt der Lokommander II-Decoder über 3 Kontakte auf einer Seite des Verdrahtung. Die Position der 3 Kontakte, an denen die Verbindungsdrähte angebracht werden, kann aus den Bildern der verschiedenen Decodertypen in Kap. 6 bestimmt werden.



Die Installation dieser Geräte erfordert hochwertige Lötgeräte und Erfahrung. Unsere Garantie erstreckt sich nicht auf Fehler, die auf unsachgemäße Eingriffe und Lötfehler zurückzuführen sind.

Die Kondensatoren sind mit einer Reihe von Widerständen von 100 Ohm beschaltet, wodurch der Ladestrom begrenzt wird, sodass die digitalen Steuersysteme den Ladestrom des Kondensators nicht als Kurzschluss-Situation interpretieren. Die Dioden sind so ausgelegt, dass sie den internen Schaltungen des Decoders die erforderliche Leistung/Energie oder den maximalen Strom zur Verfügung stellen, wenn keine Stromversorgung zur Verfügung steht. Diode und Widerstand sind externe Komponenten, die nicht im Lokommander enthalten sind. Beim Erstellen von Verbindungen folgen Sie den Bildern in Kapitel 6. Die roten und schwarzen Kabeln

werden an den GND- und Vcc-Punkten gelötet, den beiden externen Punkten der SPP-Gruppe des 3poligen Lötanschlusses. Nachdem wir die Verbindung hergestellt haben, können wir einen Schrumpfschlauch oder ein Isolierband zur Absicherung verwenden.



Trennen Sie die Puffer kondensatoren, bevor Sie die Decoder programmieren (sei es entweder in DCC-Digitalsystemen oder mit dem tOm Programmer programmiert). Die Verwendung von Puffer kondensatoren erschwert die Programmierung von CVs der Decodierer wegen der zusätzlichen Stromaufnahme.

Ein unterbrechungsfreies SPP Ressource (Smart Power Pack oder andere gleichwertige) beseitigt diese Unannehmlichkeit, da sowohl Schreib- als auch Lesevorgänge der CVs auf den üblichen Weise fortgesetzt werden können, ohne dass sie auseinandergenommen / getrennt werden müssen). Das Ausschalten der SPP-Quelle während der Programmierung wird vom Lokommander über das dritte Kabel (Charge) automatisch durchgeführt.

Verwenden Sie zum Anschließen der SPP-Module die 3 Stecker auf einer der Lokommander II-Seiten. Zusätzliche Informationen zum Anschluss finden Sie in den Handbüchern der jeweiligen Quellen bzw. auf den, in Kapitel. 6 gezeigten Bildern.

SPP-Quellen arbeiten nur im digitalen Modus, im analogen Modus sind sie deaktiviert (siehe Konfiguration des CV29). Um einen sehr hohen Konsum durch gleichzeitiges Laden unterbrechungsfreier Stromversorgungen zu vermeiden, gibt es eine Startverzögerung für die Einspeisung mehrerer mit SPO ausgestatteter Lokomotiven. In CV152 kann also die Zeit eingestellt werden, nach der das SPP-Modul ab dem Zeitpunkt der Stromversorgung gestartet wird. Bei mehreren gleichzeitig verwendeten Decodern wird diese Zeit in unterschiedlichen Modellen unterschiedlich eingestellt, um das gleichzeitige Starten aller SPPs zu vermeiden. Mit den SPP-Modulen können Lokomotiven bis zu 4 Sekunden ohne DCC-Versorgung aus den Schienen laufen (wenn sie vollgeladet sind, abhängig vom Verbrauch der Lokomotive). Diese Dauer wird in CV123 eingestellt (die Standardwert beträgt:  $16 \cdot CVV * 16\text{ms} = 0,25$  Sekunden). Nach der Ablaufzeit dieses Prozesses, wenn das DCC-Signal nicht mehr vorhanden ist, führt die Lokomotive nach ihrem Ablauf einen Nothalt (als Sicherheitsmaßnahme) durch, gefolgt von der Wiederaufnahme des DCC-Signals nach dem DCC-Signal erneut empfangen wird.



Beachten Sie, dass beim Laden der SPP-Module beim unterbrechungsfreien Laden der Kondensatoren ein Strom von etwa 300 mA für maximal 2 Minuten nach dem Start des Ladevorgangs verbraucht werden kann. Weitere Informationen finden Sie im SPP-Handbuch für unterbrechungsfreie Stromversorgung.

## 20. Rücksetzung des Decoders/Reset

Sie können den Decoder jederzeit auf die Standardeinstellungen zurücksetzen.

Bei Verwendung einer beliebigen DCC-Steuerstation reicht es aus, einen beliebigen numerischen Wert in CV8 einzugeben (außer dem Wert 128). Als Ergebnis dieses Zurücksetzens werden alle CVs wieder den Standardwert haben (siehe Spalte Standardwert in der CV-Tabelle). Decoders können auch mithilfe von tOm Programmer zurückgesetzt werden. Drücken Sie für dasselbe Ergebnis in der Firmware-TAB die Option Reset CV-s. Es gibt 2 CVs, die eine Ausnahme darstellen. Ihr Inhalt wird bei einem Reset nicht gelöscht. Dies sind CV105 und CV106, die benutzerspezifische Informationen (Seriennummer, Kennzeichnung, Inventarnummer usw.) speichern sollen. Ihr Inhalt wird durch direktes Schreiben geändert. Durch das Zurücksetzen des Decoders wird der Inhalt dieser CVs nicht geändert.



**Achtung!** Bei einem Firmware-Upgrade werden die Standardwerte in diese CVs eingegeben, um ihren Wert zu erhalten, bevor Sie Ihre CVs aktualisieren, machen sie einen back-up, speichern Sie Ihr CVs (mithilfe des tOm-Programmiergeräts).



Durch das Zurücksetzen des Lokommander II-Decoders werden die CVs der angeschlossenen SUSI-Module nicht zurückgesetzt.



## 21. Sekundäre Adresse (blockiert das Programmieren des Decoders)

Bei der Verwendung mehrerer Decoder in demselben Modell oder Zugverband ist es sinnvoll, eine sekundäre Adresse zu verwenden, damit der Decoder während der Programmierung ausgewählt werden kann. Auf diese Weise kann jeder Decoder, der sich im selben Modell befindet, auf der Programmierschiene programmiert werden, ohne dass er aus dem Modell demontiert werden muß. Die verschiedenen Sekundäradressen werden in CV16 programmiert, bevor die Decoder im Modell montiert wird. Der Wertebereich der sekundären Adressen beträgt 1 bis 7 (der Wert 0 bedeutet, dass keine sekundäre Adresse verwendet wird). Dies ermöglicht den Einsatz von bis zu 7 Decodern im selben Modell oder Verband, was mehr als ausreichend ist. Durch Zuweisen einer anderen sekundären Adresse zu jedem Decoder, der dem Zugverband angehört, wird nur der Decoder programmiert, für den CV15 = CV16 programmiert wird. Auf diese Weise können wir, indem Sie jede der einzelnen Adressen in CV15 nacheinander schreiben, mehrere Decoder unabhängig voneinander programmieren, auch wenn sie gleichzeitig auf der Programmierschienen sind. Die Decoder, für die der CV15  $\neq$  CV16 werden alle CV Änderungen oder Lesungen ignorieren.



Selbst CV16 kann nur programmiert werden, wenn der richtige Wert in CV15 programmiert ist.

Bei der Verwendung der sekundären Adressen ist es wichtig zu wissen, dass die einzige CV der gelesen oder geschrieben werden kann, ohne dass die sekundäre Adresse kennen würde, CV15 ist.



Wenn Sie den Decoder versehentlich durch Eingabe eines unbekanntes Werts in CV16 blockieren, müssen Sie die Werte 1-7 in CV15 eingeben, um den korrekten Wert heraus zu finden womit der CV16-Wert gelöscht wird

Diese Art des Zugriffs /Programmieren für die CVs des Decoders ist bei fest verbundenen Waggonen oder mit ständig konnektierte Sets hilfreich, die mit mehrere Decoders ausgestattet sind und es wäre sehr unpraktisch, sie auf eine traditionelle Weise zu programmieren (auf der Programmierungsschiene wären alle vorhandene Decoder mit den gleichen CV-Werten programmiert, was eher eine unerwünschte Sache ist).

## 22. Firmware aktualisieren

Sie können die Betriebssoftware des Lokommander II-Decoders (Firmware genannt) jederzeit aktualisieren. Neue Firmware-Versionen dienen dazu, Fehler (bug) beim Betrieb von Decodern zu beseitigen oder neue Funktionen zu implementieren.

Dieses Update kann von Ihnen über den tOm Programmierer durchgeführt werden, ohne den Decoder von der Lokomotive entfernen zu müssen.

Die Betriebssystemsoftware und Aktualisierungsdateien für den Firmware des tOm Programmiers können von der train-O-matic-Site heruntergeladen werden. Informationen zur Durchführung des Upgrades finden Sie im Handbuch des tOm Programmiers.

Die aktuelle Firmware-Version kann aus den folgenden CV-Dateien abgelesen werden:

CV253	Firmware Version
CV254	Firmware Subversion
CV254	Build-Nummer, höchstwertiges Byte
CV256	Build-Nummer, niederwertigstes Byte

---

## 23. Spezielle Firmware-Version für 3V-Motor

Diese Option ist bei Eisenbahnmodellen hilfreich, die aus verschiedenen Gründen (Größe, Abmessungen) die Verwendung normaler Motoren nicht zulassen, sondern nur Motoren mit niedriger Betriebsspannung.

Um die Steuerung der 3V-Motoren sicherzustellen, wurden Änderungen an der Lokomander II-Firmware und an der Verwendung zusätzlicher Hardware vorgenommen. Wenn Sie Kapitel 11 lesen, können wir das Prinzip der Motorsteuerung verstehen. Es fällt auf, dass die Messung der elektromotorischen Spannung (BEMF) in regelmäßigen Abständen erfolgt. Für den 3V-Motor ist diese Spannung viel niedriger. Um dieses Problem Unannehmlichkeiten zu überwinden, haben wir einen externen Verstärker verwendet. Das Lesen der verstärkten Spannung erfolgt an einem der Logikausgänge (AUX6) des Decoders, dessen Software für den Analogeingang konfiguriert ist. Der Befehl des BEMF-Spannungsteilers wird auch mit einem anderen Logikausgang (AUX5) ausgeführt.

Die 3V-Motorsteuerung erfolgt wie beim regulären Motor mit PWM-Impulsen mit einem variablen Füllfaktor von 12-16 V. Die Firmware begrenzt den Füllfaktor, um ein Verbrennen von 3V-Motoren zu verhindern. Um die Spitzenströme zu begrenzen, die beim Anlegen des PWM-Signals am 3-V-Motor auftreten können, werden zwei an den Motor angeschlossene Widerstände verwendet.

Für den SPP-Befehl (oder ein externes Kondensatorpaket) wird der AUX2-Ausgang verwendet.

Zur Steuerung der Leuchten und Zusatzverbraucher stehen 3 Ausgänge weniger zur Verfügung. Beim NEXT18-Decoder bleiben die Ausgänge FL, RL, AUX1 und AUX3/AUX4 (nur verfügbar,

wenn die SUSI-Schnittstelle nicht verwendet wird) für die Leuchten verfügbar.

Diese spezielle Firmware kann mit Lokommander II mit einer der folgenden Schnittstellen verwendet werden: NEXT18, MTC21, PLUX22.

Mit dieser speziellen Firmware ausgestattete Decoder können nur für die 3V-Motor Steuerung und nur mit den zugehörigen Hardwareschaltungen verwendet werden. Durch ein erneutes Schreiben der Firmware können Sie die üblichen 12-16V-Motoren verwenden.

## 24. Ergänzendes

- *tOm SPP* kann optional an Decoder angeschlossen werden und bietet bis zu 4 Sekunden Energieunabhängigkeit
- Der *tOm Programmer* ist eine PC-Schnittstelle für die Programmierung von DCC- Lokdecodern.
- *shine FDT*, *shine LT*, *shine MICRO* sind LED-Module für die Beleuchtung von Lokomotiven und Waggon
- *shine mini/midi/maxi digi/ana* sind LED-Sticks für die Innenbeleuchtung von Waggon
- *TD Maxi*, *TD Roco* sind Schaltdecoder für Weichen

Einzelheiten zum Zubehör und eine vollständige Liste der Produkte für Eisenbahnmodelle finden Sie unter: [www.train-o-matic.com](http://www.train-o-matic.com)



## 25. Technischer Unterstützung

Wenn Sie Fragen oder Anregungen zu train-O-matic-Produkten haben, können Sie uns schreiben an: [support@train-o-matic.com](mailto:support@train-o-matic.com)

Jede positive oder negative Kritik ist willkommen. Wir arbeiten ständig daran, die Firmware zu optimieren, indem wir neue Funktionen hinzufügen und eventuell noch vorhandene Fehler korrigieren.



## 26. Tabelle der CVs des Decoders

In der Tabelle auf den folgenden Seiten haben wir alle CV-s der Lokommander II Decoder aufgelistet. Wir empfehlen, dass Sie die CV-s nur dann ändern, wenn Sie sich ihrer Funktion und der Wirkung Ihrer Maßnahmen sicher sind. Falsche CV-Einstellungen können sich negativ auf die Leistung des Decoders auswirken oder zu falschen Antworten auf Befehle führen, die an den Decoder gesendet werden. Die CV-Spalte enthält die CVs Nummer, die Default-Spalte enthält die Werkseinstellung der CVs (nach dem Zurücksetzen des Decoders haben alle CVs den entsprechenden Wert in dieser Spalte), Spalte Der Wertebereich enthält den Bereich der verwendbaren Werte für jeden CV, und die Spalte Beschreibung enthält den Namen (wenn ein festgelegter Name vorhanden ist) und Informationen zur CV Funktion sowie den Verweis auf das entsprechende Kapitel.

CV	Standardwert	Der Wertebereich	Beschreibung
1	3	0-127	Hauptadresse des Decoders, 7 Bit
2	1	1-255	Vmin , Minimale Geschwindigkeit
3	5	0-63	Beschleunigungsrate : 0 = schnellste Beschleunigung
4	7	0-63	Verzögerungsrate : 0 = schnellste Verzögerung
5	200	1-255	Vmax, Maximale Geschwindigkeit
6	68	0-255	Vmid, Mittengeschwindigkeit, [25%-75%]Vmax
7	105	-	Softwareversion (nur lesbar)
8	78	-	Hersteller-ID, Decoderreset, Bei jedem geschriebenen Wert wird der Decoder auf die Werkseinstellung



			zurückgesetzt, mit Ausnahme des Werts 128, der die Wartung kennzeichnet, sofern aktiviert (siehe Kapitel. 16)
9	3	0-9	Motorsteuerungsalgorithmus , 0-8 benutzerdefinierten = 9 ( siehe Kapitel . 11 und CV-s 128-130)
13	0	0-255	Funktionstasten F1-F8 im Analogbetrieb aktivieren Bit 0 = 0(0): F1 im Analogbetrieb ausgeschaltet = 1(1): F1 im Analogbetrieb eingeschaltet Bit 1 = 0(0): F2 im Analogbetrieb ausgeschaltet = 1(2): F2 im Analogbetrieb eingeschaltet Bit 2 = 0(0): F3 im Analogbetrieb ausgeschaltet = 1(4): F3 im Analogbetrieb eingeschaltet Bit 3 = 0(0): F4 im Analogbetrieb ausgeschaltet = 1(8): F4 im Analogbetrieb eingeschaltet Bit 4 = 0(0): F5 im Analogbetrieb ausgeschaltet = 1(16): F5 im Analogbetrieb eingeschaltet Bit 5 = 0(0): F6 im Analogbetrieb ausgeschaltet = 1(32): F6 im Analogbetrieb eingeschaltet Bit 6 = 0(0): F7 im Analogbetrieb ausgeschaltet = 1(64): F7 im Analogbetrieb eingeschaltet Bit 7 = 0(0): F8 im Analogbetrieb ausgeschaltet = 1(128):F8 im Analogbetrieb eingeschaltet
14	3=	0-255	Funktionstasten F0f, F0r, F9-F14 im Analogbetrieb aktivieren Bit 0 = 0(0): F1 im Analogbetrieb ausgeschaltet = 1(1): F1 im Analogbetrieb eingeschaltet



	1+ 2		Bit 1 = 0(0): F2 im Analogbetrieb ausgeschaltet = 1(2): F2 im Analogbetrieb eingeschaltet Bit 2 = 0(0): F3 im Analogbetrieb ausgeschaltet = 1(4): F3 im Analogbetrieb eingeschaltet Bit 3 = 0(0): F4 im Analogbetrieb ausgeschaltet = 1(8): F4 im Analogbetrieb eingeschaltet Bit 4 = 0(0): F5 im Analogbetrieb ausgeschaltet = 1(16): F5 im Analogbetrieb eingeschaltet Bit 5 = 0(0): F6 im Analogbetrieb ausgeschaltet = 1(32): F6 im Analogbetrieb eingeschaltet Bit 6 = 0(0): F7 im Analogbetrieb ausgeschaltet = 1(64): F7 im Analogbetrieb eingeschaltet Bit 7 = 0(0): F8 im Analogbetrieb ausgeschaltet = 1(128):F8 im Analogbetrieb eingeschaltet
15	0	0-7	LockCV (sekundäre Address): Die Programmierung des Decoders (CV-Änderung) ist nur zulässig, wenn CV15 = CV16 ist. CV15 kann in jeder Situation geschrieben werden
16	0	0-7	LockID: Verwenden Sie eindeutige Identifizierungsnummern für jeden Decoder, um zu verhindern, dass mehrere Decoder versehentlich gleichzeitig programmiert werden. (siehe Kapitel. 21)
17	192	192-255	Erweiterte (Lange) Lokadresse, Höherwertiges Byte
18	3	0-255	Erweiterte (Lange) Lokadresse, Niederwertiges Byte
19	0	0-127	Konsist (Mehrfachtraktions) Adresse Wenn CV19> 0: Geschwindigkeit und Richtung werden von dieser Adresse





			bestimmt (nicht von der Haupt- oder erweiterten individuellen Adresse). Funktionen werden entweder von der Konsist (Mehrfachtraktions) Adresse oder der individuellen Adresse gesteuert (siehe CV21,22 und Kapitel 9)
21	0	0-255	Funktion gesteuert von der Consist Adresse: Bit 0 = 0(0): F1 gesteuert von der individuellen Adresse = 1(1): F1 gesteuert von der consist Adresse Bit 1 = 0(0): F2 gesteuert von der individuellen Adresse = 1(2): F2 gesteuert von der consist Adresse Bit 2 = 0(0): F3 gesteuert von der individuellen Adresse = 1(4): F3 gesteuert von der consist Adresse Bit 3 = 0(0): F4 gesteuert von der individuellen Adresse = 1(8): F4 gesteuert von der consist Adresse Bit 4 = 0(0): F5 gesteuert von der individuellen Adresse = 1(16):F5 gesteuert von der consist Adresse Bit 5 = 0(0): F6 gesteuert von der individuellen Adresse = 1(32):F6 gesteuert von der consist Adresse Bit 6 = 0(0): F7 gesteuert von der individuellen Adresse = 1(64):F7 gesteuert von der consist Adresse Bit 7 = 0(0): F8 gesteuert von der individuellen Adresse = 1(255):F8 gesteuert von der consist Adresse
22	0	0-63	Funktion gesteuert von der Consist Adresse: Bit 0 = 0(0): F0f gesteuert von der individuellen Adresse = 1(1): F0f gesteuert von der consist Adresse Bit 1 = 0 (0): F0r gesteuert von der individuellen Adresse



			<p>= 1(2): F0r gesteuert von der consist Adresse Bit 2 = 0(0): F9 gesteuert von der individuellen Adresse = 1(4): F9 gesteuert von der consist Adresse Bit 3 = 0(0): F10 gesteuert von der individuellen Adresse = 1(8): F10 gesteuert von der consist Adresse Bit 4 = 0(0): F11 gesteuert von der individuellen Adresse = 1(16):F11 gesteuert von der consist Adresse Bit 5 = 0(0): F12 gesteuert von der individuellen Adresse = 1(32):F12 gesteuert von der consist Adresse</p>
27	0	0-7	<p>Automatische Bremse Steuerung Einstellungen (ABC) – (siehe Kapitel. 12) Bit0 = 0(0): Erkennung deaktiviert, wenn die rechte Schiene positiver ist = 1(1): Erkennung des ABC-Signals, wenn die rechte Schiene positiver Bit1 = 0(0): Erkennung deaktiviert, wenn die linke Schiene positiver ist = 1(2): Erkennung des ABC-Signals, wenn die linke Schiene positiver Bit4 = 0(0): keine Aktion wenn DC mit Fahrtrichtung entgegengesetzt = 1(16): automatisches Halten wenn DC mit Fahrtrichtung entgegengesetzt Bit5 = 0(0): keine Aktion wenn DC mit Fahrtrichtung gleich = 1(32): automatisches Halten wenn DC mit Fahrtrichtung gleich Bit7 = 0(0): automatisches Halten bei Nullgeschwindigkeit deaktiviert = 1(128):automatisches Halten bei Nullgeschwindigkeit aktiviert</p>
28	3	0-3	<p>RailCom® Konfiguration : Bit 0 = 0(0): Kanal 1 aus = 1(1): Kanal 1 ein Bit 1 = 0 (0): Kanal 2 aus</p>



			= 1(2): Kanal 2 ein
29	10(2+8)	0-63	<p>Decoder Konfiguration:</p> <p>Bit 0 = 0(0): Normale Fahrtrichtung = 1(1): Entgegengesetzte Fahrtrichtung</p> <p>Bit 1 = 0(0): 14 Fahrstufen (F0f wird von Bit4 in den DCC-Anweisungen für Geschwindigkeit und Richtung gesteuert) = 1(2): 28/128 Fahrstufen (F0f wird von Bit4 in der DCC-Gruppe 1-Anweisung gesteuert)</p> <p>Bit 2 = 0(0): Nur Digitalbetrieb = 1(4): Automatische Analog/Digitalumschaltung</p> <p>Bit 3 = 0(0): RailCom® ausgeschaltet = 1(8): RailCom® eingeschaltet</p> <p>Bit 4 = 0(0): Fahrstufen in 3 Punkten über CV2, CV5 und CV6 = 1(16): Fahrstufen verwenden die Kurve von den Tabelle CV 66-95</p> <p>Bit 5 = 0(0): Kurze Adresse (CV 1) = 1(32): Lange Adresse (CV 17/18)</p>
30	0	0-7	<p>Fehlerspeicher für Funktionsausgänge, Motor und Temperaturüberwachung:</p> <p>0 - Es wurde kein Fehler aufgezeichnet</p> <p>1 (Bit0) - Fehler (Überstrom) Motorausgänge</p> <p>2 (Bit1) - Fehler (Überstrom) Funktionsausgänge</p> <p>4 (Bit2) - Fehler Übertemperatur</p> <p>8 (Bit3) - Warnung, das Wartungsintervall überschreitet</p> <p>Ein gespeicherter Fehler kann durch Programmierung des Wertes 0 in der CV30 gelöscht werden.</p>



33	1	0-255	<p>Function mapping für F0 vorwärts</p> <p>Bit 0 = 0(0): FL inaktiv für F0 ( vorwärts ) = 1(1): FL aktiv für F0 ( vorwärts )</p> <p>Bit 1 = 0(0): FR inaktiv für F0 ( vorwärts ) = 1(2): FR aktiv für F0 ( vorwärts )</p> <p>Bit 2 = 0(0): AUX1 inaktiv für F0 ( vorwärts ) = 1(4): AUX1 aktiv für F0 ( vorwärts )</p> <p>Bit 3 = 0(0): AUX2 inaktiv für F0 ( vorwärts ) = 1(8): AUX2 aktiv für F0 ( vorwärts )</p> <p>Bit 4 = 0(0): AUX3 inaktiv für F0 ( vorwärts ) = 1(16):AUX3 aktiv für F0 ( vorwärts )</p> <p>Bit 5 = 0(0): AUX4 inaktiv für F0 ( vorwärts ) = 1(32):AUX4 aktiv für F0 ( vorwärts )</p> <p>Bit 6 = 0(0): AUX5 inaktiv für F0 ( vorwärts ) = 1(64):AUX5 aktiv für F0 ( vorwärts )</p> <p>Bit 7 = 0(0): AUX6 inaktiv für F0 ( vorwärts ) = 1(128):AUX6 aktiv für F0 ( vorwärts )</p>
34	2	0-255	<p>Function mapping für F0 rückwärts</p> <p>Bit 0 = 0(0): FL inaktiv für F0 ( rückwärts ) = 1(1): FL aktiv für F0 ( rückwärts )</p> <p>Bit 1 = 0(0): FR inaktiv für F0 ( rückwärts ) = 1(2): FR aktiv für F0 ( rückwärts )</p> <p>Bit 2 = 0(0): AUX1 inaktiv für F0 ( rückwärts ) = 1(4): AUX1 aktiv für F0 (rückwärts)</p>



## Lokommander II

Bedienungsanleitung firmware 3.5.191

Version

0.1.22

			<p>Bit 3 = 0(0): AUX2 inaktiv für F0 (rückwärts) = 1(8): AUX2 aktiv für F0 (rückwärts)</p> <p>Bit 4 = 0(0): AUX3 inaktiv für F0 (rückwärts) = 1(16):AUX3 aktiv für F0 (rückwärts)</p> <p>Bit 5 = 0(0): AUX4 inaktiv für F0 (rückwärts) = 1(32):AUX4 aktiv für F0 (rückwärts)</p> <p>Bit 6 = 0(0): AUX5 inaktiv für F0 (rückwärts) = 1(64):AUX5 aktiv für F0 (rückwärts)</p> <p>Bit 7 = 0(0): AUX6 inaktiv für F0 (rückwärts) = 1(128):AUX6 aktiv für F0 (rückwärts)</p>	
35	1 4 für PLUX22	0-255	<p>Function mapping für F1 (vorwärts)</p> <p>Bit 0 = 0(0): FL inactiv für F1 (vorwärts) = 1(1): FL activ für F1 (vorwärts)</p> <p>Bit 1 = 0(0): FR inactiv für F1 (vorwärts) = 1(2): FR activ für F1 (vorwärts)</p> <p>Bit 2 = 0(0): Aux1 inactiv für F1 (vorwärts) = 1(4): Aux1 activ für F1 (vorwärts)</p> <p>Bit 3 = 0(0): Aux2 inactiv für F1 (vorwärts) = 1(8): Aux2 activ für F1 (vorwärts)</p> <p>Bit 4 = 0(0): Aux3 inactiv für F1 (vorwärts) = 1(16):Aux3 activ für F1 (vorwärts)</p> <p>Bit 5 = 0(0): Aux4 inactiv für F1 (vorwärts) = 1(32):Aux4 activ für F1 (vorwärts)</p> <p>Bit 6 = 0(0): Aux5 inactiv für F1 (vorwärts)</p>	<p>Function mapping für F1 für Plux22</p> <p>Bit 0 = 0(0): FL inactiv für F1 = 1(1): FL activ für F1</p> <p>Bit 1 = 0(0): FR inactiv für F1 = 1(2): FR activ für F1</p> <p>Bit 2 = 0(0): Aux1 inactiv für F1 = 1(4): Aux1 activ für F1</p> <p>Bit 3 = 0(0): Aux2 inactiv für F1 = 1(8): Aux2 activ für F1</p> <p>Bit 4 = 0(0): Aux3 inactiv für F1 = 1(16):Aux3 activ für F1</p> <p>Bit 5 = 0(0): Aux4 inactiv für F1 = 1(32):Aux4 activ für F1</p> <p>Bit 6 = 0(0): Aux5 inactiv für F1</p>



# Lokommander II

Bedienungsanleitung firmware 3.5.191

Version

0.1.22

			<p>= 1(64):Aux5 activ für F1 (vorwärts)          Bit 7 = 0(0): Aux6 inactiv für F1 (vorwärts)          = 1(128):Aux6 activ für F1 (vorwärts)</p>	<p>= 1(64):Aux5 activ für F1          Bit 7 = 0(0): Aux6 inactiv für F1          = 1(128):Aux6 activ für F1</p>
36	1 8 für PLUX22	0-255	<p>Function mapping für F1 (rückwärts)          Bit 0 = 0(0): FL inactiv für F1 (rückwärts)          = 1(1): FL activ für F1 (rückwärts)          Bit 1 = 0(0): FR inactiv für F1 (rückwärts)          = 1(2): FR activ für F1 (rückwärts)          Bit 2 = 0(0): Aux1 inactiv für F1(rückwärts)          = 1(4): Aux1 activ für F1 (rückwärts)          Bit 3 = 0(0): Aux2 inactiv für F1(rückwärts)          = 1(8): Aux2 activ für F1 (rückwärts)          Bit 4 = 0(0): Aux3 inactiv für F1(rückwärts)          = 1(16):Aux3 activ für F1 (rückwärts)          Bit 5 = 0(0): Aux4 inactiv für F1(rückwärts)          = 1(32):Aux4 activ für F1 (rückwärts)          Bit 6 = 0(0): Aux5 inactiv für F1(rückwärts)          = 1(64):Aux5 activ für F1 (rückwärts)          Bit 7 = 0(0): Aux6 inactiv für F1(rückwärts)          = 1(128):Aux6 activ für F1(rückwärts)</p>	<p>Function mapping für F2 für Plux22          Bit 0 = 0(0): FL inactiv für F2          = 1(1): FL activ für F2          Bit 1 = 0(0): FR inactiv für F2          = 1(2): FR activ für F2          Bit 2 = 0(0): Aux1 inactiv für F2          = 1(4): Aux1 activ für F2          Bit 3 = 0(0): Aux2 inactiv für F2          = 1(8): Aux2 activ für F2          Bit 4 = 0(0): Aux3 inactiv für F2          = 1(16):Aux3 activ für F2          Bit 5 = 0(0): Aux4 inactiv für F2          = 1(32):Aux4 activ für F2          Bit 6 = 0(0): Aux5 inactiv für F2          = 1(64):Aux5 activ für F2          Bit 7 = 0(0): Aux6 inactiv für F2          = 1(128):Aux6 activ für F2</p>
37	2 16 für PLUX22	0-255	<p>Function mapping für F2:          Bit 0 = 0(0): FL inactiv für F2          = 1(1): FL activ für F2          Bit 1 = 0(0): FR inactiv für F2</p>	<p>Function mapping für F3 für PLUX22          Bit 0 = 0(0): FL inactiv für F3          = 1(1): FL activ für F3          Bit 1 = 0(0): FR inactiv für F3</p>



## Lokommander II

Bedienungsanleitung firmware 3.5.191

Version

0.1.22

			<p>= 1(2): FR activ für F2 Bit 2 = 0(0): Aux1 inactiv für F2 = 1(4): Aux1 activ für F2 Bit 3 = 0(0): Aux2 inactiv für F2 = 1(8): Aux2 activ für F2 Bit 4 = 0(0): Aux3 inactiv für F2 = 1(16):Aux3 activ für F2 Bit 5 = 0(0): Aux4 inactiv für F2 = 1(32):Aux4 activ für F2 Bit 6 = 0(0): Aux5 inactiv für F2 = 1(64):Aux5 activ für F2 Bit 7 = 0(0): Aux6 inactiv für F2 = 1(128):Aux6 activ für F2</p>	<p>= 1(2): FR activ für F3 Bit 2 = 0(0): Aux1 inactiv für F3 = 1(4): Aux1 activ für F3 Bit 3 = 0(0): Aux2 inactiv für F3 = 1(8): Aux2 activ für F3 Bit 4 = 0(0): Aux3 inactiv für F3 = 1(16):Aux3 activ für F3 Bit 5 = 0(0): Aux4 inactiv für F3 = 1(32):Aux4 activ für F3 Bit 6 = 0(0): Aux5 inactiv für F3 = 1(64):Aux5 activ für F3 Bit 7 = 0(0): Aux6 inactiv für F3 = 1(128):Aux6 activ für F3</p>
38	4 4 für PLUX22	0-255	<p>Function mapping für F3 Bit 0 = 0(0): FL inactiv für F3 = 1(1): FL activ für F3 Bit 1 = 0(0): FR inactiv für F3 = 1(2): FR activ für F3 Bit 2 = 0(0): Aux1 inactiv für F3 = 1(4): Aux1 activ für F3 Bit 3 = 0(0): Aux2 inactiv für F3 = 1(8): Aux2 activ für F3 Bit 4 = 0(0): Aux3 inactiv für F3 = 1(16):Aux3 activ für F3</p>	<p>Function mapping für F4 für PLUX22 Bit 0 = 0(0): Aux2 inactiv für F4 = 1(1): Aux2 activ für F4 Bit 1 = 0(0): Aux3 inactiv für F4 = 1(2): Aux3 activ für F4 Bit 2 = 0(0): Aux4 inactiv für F4 = 1(4): Aux4 activ für F4 Bit 3 = 0(0): Aux5 inactiv für F4 = 1(8): Aux5 activ für F4 Bit 4 = 0(0): Aux6 inactiv für F4 = 1(16):Aux6 activ für F4</p>



# Lokommander II

Bedienungsanleitung firmware 3.5.191

Version

0.1.22

			<p>Bit 5 = 0(0): Aux4 inactiv für F3 = 1(32):Aux4 activ für F3</p> <p>Bit 6 = 0(0): Aux5 inactiv für F3 = 1(64):Aux5 activ für F3</p> <p>Bit 7 = 0(0): Aux6 inactiv für F3 = 1(128):Aux6 activ für F3</p>	<p>Bit 5 = 0(0): Aux7 inactiv für F4 = 1(32):Aux7 activ für F4</p> <p>Bit 6 = 0(0): Aux8 inactiv für F4 = 1(64):Aux8 activ für F4</p> <p>Bit 7 = 0(0): Aux9 inactiv für F4 = 1(128):Aux9 activ für F4</p>
39	8 8 für PLUX22	0-255	<p>Function mapping für F4</p> <p>Bit 0 = 0(0): FL inactiv für F4 = 1(1): FL activ für F4</p> <p>Bit 1 = 0(0): FR inactiv für F4 = 1(2): FR activ für F4</p> <p>Bit 2 = 0(0): Aux1 inactiv für F4 = 1(4): Aux1 activ für F4</p> <p>Bit 3 = 0(0): Aux2 inactiv für F4 = 1(8): Aux2 activ für F4</p> <p>Bit 4 = 0(0): Aux3 inactiv für F4 = 1(16):Aux3 activ für F4</p> <p>Bit 5 = 0(0): Aux4 inactiv für F4 = 1(32):Aux4 activ für F4</p> <p>Bit 6 = 0(0): Aux5 inactiv für F4 = 1(64):Aux5 activ für F4</p> <p>Bit 7 = 0(0): Aux6 inactiv für F4 = 1(128):Aux6 activ für F4</p>	<p>Function mapping für F5 für PLUX22</p> <p>Bit 0 = 0(0): Aux2 inactiv für F5 = 1(1): Aux2 activ für F5</p> <p>Bit 1 = 0(0): Aux3 inactiv für F5 = 1(2): Aux3 activ für F5</p> <p>Bit 2 = 0(0): Aux4 inactiv für F5 = 1(4): Aux4 activ für F5</p> <p>Bit 3 = 0(0): Aux5 inactiv für F5 = 1(8): Aux5 activ für F5</p> <p>Bit 4 = 0(0): Aux6 inactiv für F5 = 1(16):Aux6 activ für F5</p> <p>Bit 5 = 0(0): Aux7 inactiv für F5 = 1(32):Aux7 activ für F5</p> <p>Bit 6 = 0(0): Aux8 inactiv für F5 = 1(64):Aux8 activ für F5</p> <p>Bit 7 = 0(0): Aux9 inactiv für F5 = 1(128):Aux9 activ für F5</p>
40	16	0-255	Function mapping für F5	Function mapping für F6 für PLUX22





	16 für PLUX22		Bit 0 = 0(0): FL inactiv für F5 = 1(1): FL activ für F5 Bit 1 = 0(0): FR inactiv für F5 = 1(2): FR activ für F5 Bit 2 = 0(0): Aux1 inactiv für F5 = 1(4): Aux1 activ für F5 Bit 3 = 0(0): Aux2 inactiv für F5 = 1(8): Aux2 activ für F5 Bit 4 = 0(0): Aux3 inactiv für F5 = 1(16):Aux3 activ für F5 Bit 5 = 0(0): Aux4 inactiv für F5 = 1(32):Aux4 activ für F5 Bit 6 = 0(0): Aux5 inactiv für F5 = 1(64):Aux5 activ für F5 Bit 7 = 0(0): Aux6 inactiv für F5 = 1(128):Aux6 activ für F5	Bit 0 = 0(0): Aux2 inactiv für F6 = 1(1): Aux2 activ für F6 Bit 1 = 0(0): Aux3 inactiv für F6 = 1(2): Aux3 activ für F6 Bit 2 = 0(0): Aux4 inactiv für F6 = 1(4): Aux4 activ für F6 Bit 3 = 0(0): Aux5 inactiv für F6 = 1(8): Aux5 activ für F6 Bit 4 = 0(0): Aux6 inactiv für F6 = 1(16):Aux6 activ für F6 Bit 5 = 0(0): Aux7 inactiv für F6 = 1(32):Aux7 activ für F6 Bit 6 = 0(0): Aux8 inactiv für F6 = 1(64):Aux8 activ für F6 Bit 7 = 0(0): Aux9 inactiv für F6 = 1(128):Aux9 activ für F6
41	32 32 für PLUX22	0-255	Function mapping für F6 Bit 0 = 0(0): FL inactiv für F6 = 1(1): FL activ für F6 Bit 1 = 0(0): FR inactiv für F6 = 1(2): FR activ für F6 Bit 2 = 0(0): Aux1 inactiv für F6 = 1(4): Aux1 activ für F6 Bit 3 = 0(0): Aux2 inactiv für F6	Function mapping für F7 für PLUX22 Bit 0 = 0(0): Aux2 inactiv für F7 = 1(1): Aux2 activ für F7 Bit 1 = 0(0): Aux3 inactiv für F7 = 1(2): Aux3 activ für F7 Bit 2 = 0(0): Aux4 inactiv für F7 = 1(4): Aux4 activ für F7 Bit 3 = 0(0): Aux5 inactiv für F7



## Lokommander II

Bedienungsanleitung firmware 3.5.191

Version

0.1.22

			<p>= 1(8): Aux2 activ für F6 Bit 4 = 0(0): Aux3 inactiv für F6 = 1(16):Aux3 activ für F6 Bit 5 = 0(0): Aux4 inactiv für F6 = 1(32):Aux4 activ für F6 Bit 6 = 0(0): Aux5 inactiv für F6 = 1(64):Aux5 activ für F6 Bit 7 = 0(0): Aux6 inactiv für F6 = 1(128):Aux6 activ für F6</p>	<p>= 1(8): Aux5 activ für F7 Bit 4 = 0(0): Aux6 inactiv für F7 = 1(16):Aux6 activ für F7 Bit 5 = 0(0): Aux7 inactiv für F7 = 1(32):Aux7 activ für F7 Bit 6 = 0(0): Aux8 inactiv für F7 = 1(64):Aux8 activ für F7 Bit 7 = 0(0): Aux9 inactiv für F7 = 1(128):Aux9 activ für F7</p>
42	64 64 für PLUX22	0-255	<p>Function mapping für F7 Bit 0 = 0(0): FL inactiv für F7 = 1(1): FL activ für F7 Bit 1 = 0(0): FR inactiv für F7 = 1(2): FR activ für F7 Bit 2 = 0(0): Aux1 inactiv für F7 = 1(4): Aux1 activ für F7 Bit 3 = 0(0): Aux2 inactiv für F7 = 1(8): Aux2 activ für F7 Bit 4 = 0(0): Aux3 inactiv für F7 = 1(16):Aux3 activ für F7 Bit 5 = 0(0): Aux4 inactiv für F7 = 1(32):Aux4 activ für F7 Bit 6 = 0(0): Aux5 inactiv für F7 = 1(64):Aux5 activ für F7</p>	<p>Function mapping für F8 für PLUX22 Bit 0 = 0(0): Aux2 inactiv für F8 = 1(1): Aux2 activ für F8 Bit 1 = 0(0): Aux3 inactiv für F8 = 1(2): Aux3 activ für F8 Bit 2 = 0(0): Aux4 inactiv für F8 = 1(4): Aux4 activ für F8 Bit 3 = 0(0): Aux5 inactiv für F8 = 1(8): Aux5 activ für F8 Bit 4 = 0(0): Aux6 inactiv für F8 = 1(16):Aux6 activ für F8 Bit 5 = 0(0): Aux7 inactiv für F8 = 1(32):Aux7 activ für F8 Bit 6 = 0(0): Aux8 inactiv für F8 = 1(64):Aux8 activ für F8</p>



			Bit 7 = 0(0): Aux6 inactiv für F7 = 1(128):Aux6 activ für F7	Bit 7 = 0(0): Aux9 inactiv für F8 = 1(128):Aux9 activ für F8
43	128 16 für PLUX22	0-255	Function mapping für F8 Bit 0 = 0(0): FL inactiv für F8 = 1(1): FL activ für F8 Bit 1 = 0(0): FR inactiv für F8 = 1(2): FR activ für F8 Bit 2 = 0(0): Aux1 inactiv für F8 = 1(4): Aux1 activ für F8 Bit 3 = 0(0): Aux2 inactiv für F8 = 1(8): Aux2 activ für F8 Bit 4 = 0(0): Aux3 inactiv für F8 = 1(16):Aux3 activ für F8 Bit 5 = 0(0): Aux4 inactiv für F8 = 1(32):Aux4 activ für F8 Bit 6 = 0(0): Aux5 inactiv für F8 = 1(64):Aux5 activ für F8 Bit 7 = 0(0): Aux6 inactiv für F8 = 1(128):Aux6 activ für F8	Function mapping für F9 für PLUX22 Bit 0 = 0(0): Aux5 inactiv für F9 = 1(1): Aux5 activ für F9 Bit 1 = 0(0): Aux6 inactiv für F9 = 1(2): Aux6 activ für F9 Bit 2 = 0(0): Aux7 inactiv für F9 = 1(4): Aux7 activ für F9 Bit 3 = 0(0): Aux8 inactiv für F9 = 1(8): Aux8 activ für F9 Bit 4 = 0(0): Aux9 inactiv für F9 = 1(16):Aux9 activ für F9 Bit 5 = 0(0): Aux10 inactiv für F9 = 1(32):Aux10 activ für F9
44	0 32 für PLUX22	0-255	Function mapping für F9 Bit 0 = 0(0): FL inactiv für F9 = 1(1): FL activ für F9 Bit 1 = 0(0): FR inactiv für F9 = 1(2): FR activ für F9	Function mapping für F10 für PLUX22 Bit 0 = 0(0): Aux5 inactiv für F10 = 1(1): Aux5 activ für F10 Bit 1 = 0(0): Aux6 inactiv für F10 = 1(2): Aux6 activ für F10



			<p>Bit 2 = 0(0): Aux1 inactiv für F9 = 1(4): Aux1 activ für F9</p> <p>Bit 3 = 0(0): Aux2 inactiv für F9 = 1(8): Aux2 activ für F9</p> <p>Bit 4 = 0(0): Aux3 inactiv für F9 = 1(16):Aux3 activ für F9</p> <p>Bit 5 = 0(0): Aux4 inactiv für F9 = 1(32):Aux4 activ für F9</p> <p>Bit 6 = 0(0): Aux5 inactiv für F9 = 1(64):Aux5 activ für F9</p> <p>Bit 7 = 0(0): Aux6 inactiv für F9 = 1(128):Aux6 activ für F9</p>	<p>Bit 2 = 0(0): Aux7 inactiv für F10 = 1(4): Aux7 activ für F10</p> <p>Bit 3 = 0(0): Aux8 inactiv für F10 = 1(8): Aux8 activ für F10</p> <p>Bit 4 = 0(0): Aux9 inactiv für F10 = 1(16):Aux9 activ für F10</p> <p>Bit 5 = 0(0): Aux10 inactiv für F10 = 1(32):Aux10 activ für F10</p>
45	0 64 für PLUX22	0-255	<p>Function mapping für F10</p> <p>Bit 0 = 0(0): FL inactiv für F10 = 1(1): FL activ für F10</p> <p>Bit 1 = 0(0): FR inactiv für F10 = 1(2): FR activ für F10</p> <p>Bit 2 = 0(0): Aux1 inactiv für F10 = 1(4): Aux1 activ für F10</p> <p>Bit 3 = 0(0): Aux2 inactiv für F10 = 1(8): Aux2 activ für F10</p> <p>Bit 4 = 0(0): Aux3 inactiv für F10 = 1(16):Aux3 activ für F10</p> <p>Bit 5 = 0(0): Aux4 inactiv für F10</p>	<p>Function mapping für F11 für PLUX22</p> <p>Bit 0 = 0(0): Aux5 inactiv für F11 = 1(1): Aux5 activ für F11</p> <p>Bit 1 = 0(0): Aux6 inactiv für F11 = 1(2): Aux6 activ für F11</p> <p>Bit 2 = 0(0): Aux7 inactiv für F11 = 1(4): Aux7 activ für F11</p> <p>Bit 3 = 0(0): Aux8 inactiv für F11 = 1(8): Aux8 activ für F11</p> <p>Bit 4 = 0(0): Aux9 inactiv für F11 = 1(16):Aux9 activ für F11</p> <p>Bit 5 = 0(0): Aux10 inactiv für F11</p>



# Lokommander II

Bedienungsanleitung firmware 3.5.191

Version

0.1.22

			<p>= 1(32):Aux4 activ für F10          Bit 6 = 0(0): Aux5 inactiv für F10          = 1(64):Aux5 activ für F10          Bit 7 = 0(0): Aux6 inactiv für F10          = 1(128):Aux6 activ für F10</p>	<p>= 1(32):Aux10 activ für F11</p>
46	0 128 für PLUX22	0-255	<p>Function mapping für F11          Bit 0 = 0(0): FL inactiv für F11          = 1(1): FL activ für F11          Bit 1 = 0(0): FR inactiv für F11          = 1(2): FR activ für F11          Bit 2 = 0(0): Aux1 inactiv für F11          = 1(4): Aux1 activ für F11          Bit 3 = 0(0): Aux2 inactiv für F11          = 1(8): Aux2 activ für F11          Bit 4 = 0(0): Aux3 inactiv für F11          = 1(16):Aux3 activ für F11          Bit 5 = 0(0): Aux4 inactiv für F11          = 1(32):Aux4 activ für F11          Bit 6 = 0(0): Aux5 inactiv für F11          = 1(64):Aux5 activ für F11          Bit 7 = 0(0): Aux6 inactiv für F11          = 1(128):Aux6 activ für F11</p>	<p>Function mapping für F12 für PLUX22          Bit 0 = 0(0): Aux5 inactiv für F12          = 1(1): Aux5 activ für F12          Bit 1 = 0(0): Aux6 inactiv für F12          = 1(2): Aux6 activ für F12          Bit 2 = 0(0): Aux7 inactiv für F12          = 1(4): Aux7 activ für F12          Bit 3 = 0(0): Aux8 inactiv für F12          = 1(8): Aux8 activ für F12          Bit 4 = 0(0): Aux9 inactiv für F12          = 1(16):Aux9 activ für F12          Bit 5 = 0(0): Aux10 inactiv für F12          = 1(32):Aux10 activ für F12</p>
47	0 4 für PLUX22	0-255	<p>Function mapping für F12          Bit 0 = 0(0): FL inactiv für F12</p>	<p>Function mapping für F1 (rückwärts) für          Plux22</p>



# Lokommander II

Bedienungsanleitung firmware 3.5.191

Version

0.1.22

			= 1(1): FL activ für F12 Bit 1 = 0(0): FR inactiv für F12 = 1(2): FR activ für F12 Bit 2 = 0(0): Aux1 inactiv für F12 = 1(4): Aux1 activ für F12 Bit 3 = 0(0): Aux2 inactiv für F12 = 1(8): Aux2 activ für F12 Bit 4 = 0(0): Aux3 inactiv für F12 = 1(16):Aux3 activ für F12 Bit 5 = 0(0): Aux4 inactiv für F12 = 1(32):Aux4 activ für F12 Bit 6 = 0(0): Aux5 inactiv für F12 = 1(64):Aux5 activ für F12 Bit 7 = 0(0): Aux6 inactiv für F12 = 1(128):Aux6 activ für F12	Bit 0 = 0(0): FL inactiv für F1 (rückwärts) = 1(1): FL activ für F1 (rückwärts) Bit 1 = 0(0): FR inactiv für F1 (rückwärts) = 1(2): FR activ für F1 (rückwärts) Bit 2 = 0(0): Aux1 inactiv für F1 (rückwärts) = 1(4): Aux1 activ für F1 (rückwärts) Bit 3 = 0(0): Aux2 inactiv für F1 (rückwärts) = 1(8): Aux2 activ für F1 (rückwärts) Bit 4 = 0(0): Aux3 inactiv für F1 (rückwärts) = 1(16):Aux3 activ für F1 (rückwärts) Bit 5 = 0(0): Aux4 inactiv für F1 (rückwärts) = 1(32):Aux4 activ für F1 (rückwärts) Bit 6 = 0(0): Aux5 inactiv für F1 (rückwärts) = 1(64):Aux5 activ für F1 (rückwärts) Bit 7 = 0(0): Aux6 inactiv für F1 (rückwärts) = 1(128):Aux6 activ für F1 (rückwärts)
48	255	0-255	Lichtintensität FL ( PWM-Faktor ) [1-255]	
49	255	0-255	Lichtintensität FR ( PWM-Faktor ) [1-255]	
50	255	0-255	Lichtintensität Aux1 ( PWM-Faktor ) [1-255]	
51	255	0-255	Lichtintensität Aux2 ( PWM-Faktor ) [1-255]	
52	255	0-255	Lichtintensität Aux3 ( PWM-Faktor ) [1-255]	
53	255	0-255	Lichtintensität Aux4 ( PWM-Faktor ) [1-255]	
54	255	0-255	Lichtintensität Aux5 ( PWM-Faktor ) [1-255]	



55	255	0-255	Lichtintensität Aux6 ( PWM-Faktor ) [1-255]
56	255	0-255	Lichtintensität Aux7 ( PWM-Faktor ) [1-255]
57	255	0-255	Lichtintensität Aux8 ( PWM-Faktor ) [1-255]
58	255	0-255	Lichtintensität Aux9 ( PWM-Faktor ) [1-255]
59	255	0-255	Lichtintensität Aux10 ( PWM-Faktor ) [1-255]
60	3	0,1,128, 129	PID-Motorsteuerung und PWM-Frequenz des Motor Bit 0 = 0(0): Inaktiver PID-Regler = 1(1): Aktiver PID-Regler Bit7 = 0(0): PWM-Frequenz des Motor = 32kHz = 1(128): PWM-Frequenz des Motor = 16kHz
61	80	0-255	PID P-Konstante
62	120	0-255	PID I-Konstante
63	40	0-255	PID P-Konstante
64	1	0-10	Bremsweg im CBD-Modus – siehe Kapitel. 12.1 0: Konstanter Bremsweg deaktiviert, 1-15: Bremsweg = (wert CV64) x ( Mindestbremsweg )
65	0	0-255	Bremsverzögerung im CBD-Modus (mit fester Verzögerung) 0: ohne Verzögerung Um den Bremsweg in kleinen Mengen zu erhöhen, erhöhen Sie den Wert dieser CV. Das Bremsen wird um (CV65-Wert) * 8ms verzögert
67	2	1-255	Geschwindigkeitstabelle, Fahrstufen 1
:			



94	240	1-255	Geschwindigkeitstabelle, Fahrstufen 28
95	1	1-10	PID-Fehlerlimitation
105	0	0-255	CV Benutzerdaten 1
106	0	0-255	CV Benutzerdaten 2
112	15	1-127	Aufblendzeit (fade-in) für ausgänge FR, FL, AUX1..9 1=8ms, 15=120ms, 125=1000ms
113	3	1-127	Abblendzeit (fade-out) für ausgänge FR, FL, AUX1..9 1=8ms, 15=120ms, 125=1000ms
114	4	0-255	Function mapping zum deaktivieren der Rangiergeschwindigkeit 1 - F1, 2 - F2, 4 - F3 ... 128 - F8 ( Werkseinstellung F3)
115	8	0-255	Function mapping zum deaktivieren der Beschleunigung / Verzögerung 1 - F1, 2 - F2, 4 - F3 ... 128 - F8 (Werkseinstellung F4)
116	16	0-255	Function mapping zum deaktivieren des Konstanter Bremsweg 1 - F1, 2 - F2, 4 - F3 ... 128 - F8 ( Werkseinstellung F5)
117	0	0-255	PWM (Dimmen) an den Ausgängen ein/ausschalten (FL, FR, Aux1-Aux6): Bit 0 = 0(0): FL PWM signal am Ausgang mit Dimmen = 1(1): FL Dauersignal am Ausgang ohne Dimmen Bit 1 = 0(0): FR PWM signal am Ausgang mit Dimmen = 1(2): FR Dauersignal am Ausgang ohne Dimmen Bit 2 = 0(0): Aux1 PWM signal am Ausgang mit Dimmen = 1(4): Aux1 Dauersignal am Ausgang ohne Dimmen Bit 3 = 0(0): Aux2 PWM signal am Ausgang mit Dimmen = 1(8): Aux2 Dauersignal am Ausgang ohne Dimmen





			<p>Bit 4 = 0(0): Aux3 PWM signal am Ausgang mit Dimmen = 1(16): Aux3 Dauersignal am Ausgang ohne Dimmen</p> <p>Bit 5 = 0(0): Aux4 PWM signal am Ausgang mit Dimmen = 1(32): Aux4 Dauersignal am Ausgang ohne Dimmen</p> <p>Bit 6 = 0(0): Aux5 PWM signal am Ausgang mit Dimmen = 1(64): Aux5 Dauersignal am Ausgang ohne Dimmen</p> <p>Bit 7 = 0(0): Aux6 PWM signal am Ausgang mit Dimmen = 1(128):Aux6 Dauersignal am Ausgang ohne Dimmen</p>
118	0	0-11	<p>Konfigurieren der automatischen Entkupplung für Ausgänge (FL, FR, Aux1..10). Für den automatischen Entkupplungs kann nur einer der Ausgänge ausgewählt werden.</p> <p>CV118 = 0, Kein Ausgang für automatische Entkupplung konfiguriert .</p> <p>CV118 = 1, FL in automatischer Entkupplung konfiguriert.</p> <p>CV118 = 2, FR in automatischer Entkupplung konfiguriert.</p> <p>CV118 = 3, Aux1 in automatischer Entkupplung konfiguriert.</p> <p>CV118 = 4, Aux2 in automatischer Entkupplung konfiguriert.</p> <p>....</p> <p>CV118 = 11, Aux9 in automatischer Entkupplung konfiguriert.</p> <p>CV118 = 12, Aux10 in automatischer Entkupplung konfiguriert.</p>
119	50	0-255	<p>Entkupplung wartezeit T1 ( siehe Kapitel. 17)</p> <p><math>T1 = (\text{Wert CV119}) * 8\text{ms}</math></p> <p>z.B: <math>400\text{ms} = 50 * 8\text{ms}</math></p>
120	50	0-255	<p>Entkupplung wartezeit T2 ( siehe Kapitel. 17)</p> <p><math>T2 = (\text{Wert CV120}) * 8\text{ms}</math></p>



			z.B: 400ms=50*8ms
121	50	0-255	Entkupplung Fahrgeschwindigkeit der Lok ( siehe Kapitel. 17)
122	71	0-255	zweite Konfigurationsvariable : Bit 0 = 0(0): SUSI-Pins werden als auxiliare ausgänge verwendet (siehe Tab. 3 ) = 1(1): SUSI-Pins werden als SUSI-CLK/DATA oder Locowire verwendet Bit 1 = 0(0): Locowire-Schnittstelle aktiv = 1(2): SUSI-Schnittstelle aktiv Bit 2 = 0(0): Motorlast werden nicht über SUSI übertragen = 1(4): Motorlast werden über SUSI übertragen Bit 3 = 0(0): Motor PWM-Gewichtung aus = 1(8): Motor-PWM-Gewichtung mit Gleisspannung auf Bit 4,5 = 00(0): Pendelzugsteuerung deaktivieren – siehe Kapitel. 12.3 = 10(16) Pendeln ohne Zwischenhalt aktivieren = 01(32) Pendeln mit Zwischenhalt aktivieren = 11(48) Eine inakzeptable Kombination sollte vermieden werden !! Bit 6 = 0(0): FL/RL inaktiv, während die Firmware aktualisiert wird = 1(64): FL/RL leuchtet während des Firmware-Updates abwechselnd auf Bit 7 = 0(0): Speichern Sie QoS und Temperatur nicht im EEPROM = 1(128):Aktivieren Sie die QoS und die Temperatur Speicherung im EEPROM ( siehe Kapitel. 16)
123	16	0-255	SPP lauf zeit (Smart Power Pack) in 16ms-Schritten – siehe Kapitel. 19 Dauer =(wert CV123) * 16ms ; z.B: SPP lauf zeit = 16*16 ms =256ms
124 Old	0	0-1	Konfigurieren des Ausgangs bei automatischem Entkupplung: CV124 = 0, Der ausgewählte Ausgang (in CV118) empfängt ein PWM-Signal



			CV124 = 1, Der ausgewählte Ausgang (in CV118) empfängt ein Dauersignal
126	102	0-255	SUSI CV Lese/Schreib Offset , angesprochen CV = 800 + (wert CV126)
127		0-255	SUSI CV Daten lesen/schreiben, von 800+CV126 adressiert
128	1		Die Anzahl der PWM-Pakete, nach denen ein BEMF-Fenster eingegeben wird (nur wenn CV9 = 9) – siehe Kapitel. 11
129	6		Mediationsnummer für die BEMF-Messung (nur wenn CV9 = 9)
130	6		Verzögerung der BEMF-Messung (nur wenn CV9 = 9)
133			Decoder-Temperatur (Nur speichern, wenn F5 aktiviert/deaktiviert ist )
134	85	60-120	Maximal zulässige Temperatur am Decoder in Celsiusgraden
135	0-100 %		Aktueller Wert von QoS (Servicequalität) aktiviert durch CV122 bit 7 = 1 (Der Wert ist nur verfügbar, nachdem F5 ein-/ausgeschaltet wurde)
136	0-100 %		Minimalwert von QoS (Servicequalität), aktiviert durch CV122 bit 7 = 1 (Der Wert ist nur verfügbar, nachdem F5 ein-/ausgeschaltet wurde)
137	0-255		Durchschnittsgeschwindigkeit für Lastkompensation (unterscheidet sich von CV6)
138	0-255		Lastkompensation Koeffizient auf die minimale Geschwindigkeit (CV2)
139	0-255		Lastkompensation Koeffizient am Durchschnittsgeschwindigkeit von CV137
140	0-255		Lastkompensationskoeffizienten auf die maximale Geschwindigkeit (CV5)
141	14	0-255	ABC Detektionsempfindlichkeit – Setzt die ABC-Erkennungsschwelle in Schritten von 0,1 V z.B: wert = 14 => Schwellenwerterkennung ABC = 1,4V
142	10	0-255	Abschaltzeit im Pendelmodus (Push-Pull) in Sekunden – siehe Kapitel. 12.3
143	15	0-255	Die interne Fahrstufe in der ABC - Langsamfahrstrecke



144	10	0-255	SPP Startverzögerung nach dem Einschalten, in Schritten von 1 sek.
145	85	0-255	Die Spannungsschwelle in den Schienen, bei der Motor in den analogen Modus 1 startet, in Schritten von 0,1 V – siehe Kapitel. 14.1 z.B: wert CV161=85 => schwelle=8.5V
146	65	0-255	Die Spannungsschwelle in den Schienen, bei der Motor in analogen Modus 1 anhält, in Schritten von 0,1 V – siehe Kapitel. 14.1 z.B: wert CV161=65 => schwelle=6.5V
147	160	0-160	Die Spannungsschwelle in den Schienen, bei der Motor im Analogmodus1 die Höchstgeschwindigkeit erreicht, in Schritten von 0,1 V – siehe Kapitel. 14.1 z.B: wert CV147=160 => schwelle=16V
148	0	1-63	Beschleunigungsrate rückwärts (komplementär zu CV3) = 0: der Beschleunigungsrate von CV3 wird in beide Richtungen verwendet
149	0	0-63	Verzögerungsrate rückwärts (komplementär zu CV4) = 0: der Verzögerungsrate von CV4 wird in beide Richtungen verwendet
150	0	0-15	Verzögerungsrate rückwärts im CBD-Modus mit fester Verzögerung, siehe Kapitel. 12.1.1 =0: Die Verzögerungsrate vorwärts in CV64 wird verwendet =1-15: Bremsweg rückwärts = (wert CV150) x ( Mindestbremsweg )
151	0	0-255	Bremsweg Verzögerung rückwärts im CBD-Modus (mit fester Verzögerung) = 0: Die Bremsweg Verzögerung vorwärts in CV65 wird verwendet Um den Bremsweg in kleinen Mengen zu erhöhen, erhöhen Sie den Wert dieser CV. Das Bremsen rückwärts wird um (CV151-Wert) * 8ms verzögert
152	10	0-255	Smart Power Pack Einschaltverzögerung in sekunden



153	0	0-255	Bremsweg vorwärts im CBD-Modus mit variable Verzögerung = 0: CBD-Modus mit fester Verzögerung wird verwendet
154	0	0-15	Wartungsmodus konfigurieren (siehe Kapitel. 16): Bit0 = 0 (0): Die Wartungsfunktion ist nicht aktiv = 1 (1): Die Wartungsfunktion ist aktiv Bit1 = 0 (0): Überwindung der Wartungsintervall (IM) wird in CV30, bit3 nicht gemeldet = 1 (2): Überwindung der IM wird durch CV30, bit3 signalisiert Bit2 = 0 (0): Überwindung der IM wird nicht von FL / RL signalisiert = 1 (4): Überwindung der IM wird durch eine niederfrequente Wechselfrequenz von FL / RL signalisiert Bit3 = 0 (0): Überwindung der IM um 50% wird von FL / RL nicht signalisiert = 1(8): Überwindung der IM wird durch eine höherer Wechselfrequenz von FL / RL signalisiert
155	40	0-255	Wartungsintervall (IM), ausgedrückt in Stunden
156		0-59	Anzahl der Minuten seit der letzten bestätigten Wartung
157		0-255	Betriebsstunden, niedrigeres Byte (siehe Kapitel. 16)
158		0-255	Betriebsstunden, oberes Byte (siehe Kapitel. 16)
159		0-255	Anzahl der Stunden, die letzte Wartung bestätigt wurde, das niedrigere Byte
160		0-255	Anzahl der Stunden, die letzte Wartung bestätigt wurde, das höhere Byte
161	0	0-255	Bremsweg rückwärts im CBD-Modus mit variable Verzögerung (kompl. zu CV153) = 0: Die Bremsweg vorwärts in CV153 wird verwendet
162	100	0-255	Der Schwellenwert für die CBD-Auslösung im DC-Sektor bei Verwendung von



			SPP, in Schritten von 0,1 V – siehe Kapitel. 14.3 Spannung in den Schienen > Schwellenwert => kontrollierte Gleichstrombremsung Spannung in den Schienen < Schwellenwert => SPP lauf zeit
163	15	0-255	Die interne Fahrstufe in der ABC - Langsamfahrstrecke
164	255	0-255	Geschwindigkeit im Analogmodus 2 (siehe Kapitel. 14): = 0: Analogmodus 1 ausgewählt = 1-255: Analogmodus 2 ausgewählt
165	255	0-255	Mapping der bestätigen Impuls an den Ausgängen
166 (* )	0	0-255	Ausgänge mit F0f gesperrt (F0 vorwärts) – siehe Kapitel. 13 Bit 0 = 0(0): FL durch F0 (vorwärts) nicht gesperrt = 1(1): FL durch F0 (vorwärts) gesperrt Bit 1 = 0(0): FR durch F0 (vorwärts) nicht gesperrt = 1(2): FR durch F0 (vorwärts) gesperrt Bit 2 = 0(0): Aux1 durch F0 (vorwärts) nicht gesperrt = 1(4): Aux1 durch F0 (vorwärts) gesperrt Bit 3 = 0(0): Aux2 durch F0 (vorwärts) nicht gesperrt = 1(8): Aux2 durch F0 (vorwärts) gesperrt Bit 4 = 0(0): Aux3 durch F0 (vorwärts) nicht gesperrt = 1(16): Aux3 durch F0 (vorwärts) gesperrt Bit 5 = 0(0): Aux4 durch F0 (vorwärts) nicht gesperrt = 1(32): Aux4 durch F0 (vorwärts) gesperrt Bit 6 = 0(0): Aux5 durch F0 (vorwärts) nicht gesperrt = 1(64): Aux5 durch F0 (vorwärts) gesperrt Bit 7 = 0(0): Aux6 durch F0 (vorwärts) nicht gesperrt



			= 1(128): Aux6 durch F0 (vorwärts) gesperrt
167 (* )	0	0-255	Ausgänge mit F0r gesperrt (F0 rückwärts) Bit 0 = 0(0): FL durch F0 (rückwärts) nicht gesperrt = 1(1): FL durch F0 (rückwärts) gesperrt Bit 1 = 0(0): FR durch F0 (rückwärts) nicht gesperrt = 1(2): FR durch F0 (rückwärts) gesperrt Bit 2 = 0(0): Aux1 durch F0 (rückwärts) nicht gesperrt = 1(4): Aux1 durch F0 (rückwärts) gesperrt Bit 3 = 0(0): Aux2 durch F0 (rückwärts) nicht gesperrt = 1(8): Aux2 durch F0 (rückwärts) gesperrt Bit 4 = 0(0): Aux3 durch F0 (rückwärts) nicht gesperrt = 1(16): Aux3 durch F0 (rückwärts) gesperrt Bit 5 = 0(0): Aux4 durch F0 (rückwärts) nicht gesperrt = 1(32): Aux4 durch F0 (rückwärts) gesperrt Bit 6 = 0(0): Aux5 durch F0 (rückwärts) nicht gesperrt = 1(64): Aux5 durch F0 (rückwärts) gesperrt Bit 7 = 0(0): Aux6 durch F0 (rückwärts) nicht gesperrt = 1(128): Aux6 durch F0 (rückwärts) gesperrt
168 (* )	0	0-255	Ausgänge mit F1f gesperrt (F1 vorwärts) Bit 0 = 0(0): FL durch F1 (vorwärts) nicht gesperrt = 1(1): FL durch F1 (vorwärts) gesperrt Bit 1 = 0(0): FR durch F1 (vorwärts) nicht gesperrt = 1(2): FR durch F1 (vorwärts) gesperrt Bit 2 = 0(0): Aux1 durch F1 (vorwärts) nicht gesperrt



			<p>= 1(4): Aux1 durch F1 (vorwärts) gesperrt Bit 3 = 0(0): Aux2 durch F1 (vorwärts) nicht gesperrt = 1(8): Aux2 durch F1 (vorwärts) gesperrt Bit 4 = 0(0): Aux3 durch F1 (vorwärts) nicht gesperrt = 1(16): Aux3 durch F1 (vorwärts) gesperrt Bit 5 = 0(0): Aux4 durch F1 (vorwärts) nicht gesperrt = 1(32): Aux4 durch F1 (vorwärts) gesperrt Bit 6 = 0(0): Aux5 durch F1 (vorwärts) nicht gesperrt = 1(64): Aux5 durch F1 (vorwärts) gesperrt Bit 7 = 0(0): Aux6 durch F1 (vorwärts) nicht gesperrt = 1(128): Aux6 durch F1 (vorwärts) gesperrt</p>
169 (* )	0	0-255	<p>Ausgänge mit F2 gesperrt Bit 0 = 0(0): FL durch F2 nicht gesperrt = 1(1): FL durch F2 gesperrt Bit 1 = 0(0): FR durch F2 nicht gesperrt = 1(2): FR durch F2 gesperrt Bit 2 = 0(0): Aux1 durch F2 nicht gesperrt = 1(4): Aux1 durch F2 gesperrt Bit 3 = 0(0): Aux2 durch F2 nicht gesperrt = 1(8): Aux2 durch F2 gesperrt Bit 4 = 0(0): Aux3 durch F2 nicht gesperrt = 1(16): Aux3 durch F2 gesperrt Bit 5 = 0(0): Aux4 durch F2 nicht gesperrt = 1(32): Aux4 durch F2 gesperrt</p>





			Bit 6 = 0(0): Aux5 durch F2 nicht gesperrt = 1(64): Aux5 durch F2 gesperrt Bit 7 = 0(0): Aux6 durch F2 nicht gesperrt = 1(128): Aux6 durch F2 gesperrt
170 (* )	0	0-255	Ausgänge mit F3 gesperrt Bit 0 = 0(0): FL durch F3 nicht gesperrt = 1(1): FL durch F3 gesperrt Bit 1 = 0(0): FR durch F3 nicht gesperrt = 1(2): FR durch F3 gesperrt Bit 2 = 0(0): Aux1 durch F3 nicht gesperrt = 1(4): Aux1 durch F3 gesperrt Bit 3 = 0(0): Aux2 durch F3 nicht gesperrt = 1(8): Aux2 durch F3 gesperrt Bit 4 = 0(0): Aux3 durch F3 nicht gesperrt = 1(16): Aux3 durch F3 gesperrt Bit 5 = 0(0): Aux4 durch F3 nicht gesperrt = 1(32): Aux4 durch F3 gesperrt Bit 6 = 0(0): Aux5 durch F3 nicht gesperrt = 1(64): Aux5 durch F3 gesperrt Bit 7 = 0(0): Aux6 durch F3 nicht gesperrt = 1(128): Aux6 durch F3 gesperrt
171 (* )	0	0-255	Ausgänge mit F4 gesperrt Bit 0 = 0(0): FL durch F4 nicht gesperrt = 1(1): FL durch F4 gesperrt



			<p>Bit 1 = 0(0): FR durch F4 nicht gesperrt = 1(2): FR durch F4 gesperrt</p> <p>Bit 2 = 0(0): Aux1 durch F4 nicht gesperrt = 1(4): Aux1 durch F4 gesperrt</p> <p>Bit 3 = 0(0): Aux2 durch F4 nicht gesperrt = 1(8): Aux2 durch F4 gesperrt</p> <p>Bit 4 = 0(0): Aux3 durch F4 nicht gesperrt = 1(16): Aux3 durch F4 gesperrt</p> <p>Bit 5 = 0(0): Aux4 durch F4 nicht gesperrt = 1(32): Aux4 durch F4 gesperrt</p> <p>Bit 6 = 0(0): Aux5 durch F4 nicht gesperrt = 1(64): Aux5 durch F4 gesperrt</p> <p>Bit 7 = 0(0): Aux6 durch F4 nicht gesperrt = 1(128): Aux6 durch F4 gesperrt</p>
172 (* )	0	0-255	<p>Ausgänge mit F5 gesperrt</p> <p>Bit 0 = 0(0): FL durch F5 nicht gesperrt = 1(1): FL durch F5 gesperrt</p> <p>Bit 1 = 0(0): FR durch F5 nicht gesperrt = 1(2): FR durch F5 gesperrt</p> <p>Bit 2 = 0(0): Aux1 durch F5 nicht gesperrt = 1(4): Aux1 durch F5 gesperrt</p> <p>Bit 3 = 0(0): Aux2 durch F5 nicht gesperrt = 1(8): Aux2 durch F5 gesperrt</p> <p>Bit 4 = 0(0): Aux3 durch F5 nicht gesperrt</p>



			<p>= 1(16): Aux3 durch F5 gesperrt Bit 5 = 0(0): Aux4 durch F5 nicht gesperrt = 1(32): Aux4 durch F5 gesperrt Bit 6 = 0(0): Aux5 durch F5 nicht gesperrt = 1(64): Aux5 durch F5 gesperrt Bit 7 = 0(0): Aux6 durch F5 nicht gesperrt = 1(128): Aux6 durch F5 gesperrt</p>
173 (* )	0	0-255	<p>Ausgänge mit F6 gesperrt Bit 0 = 0(0): FL durch F6 nicht gesperrt = 1(1): FL durch F6 gesperrt Bit 1 = 0(0): FR durch F6 nicht gesperrt = 1(2): FR durch F6 gesperrt Bit 2 = 0(0): Aux1 durch F6 nicht gesperrt = 1(4): Aux1 durch F6 gesperrt Bit 3 = 0(0): Aux2 durch F6 nicht gesperrt = 1(8): Aux2 durch F6 gesperrt Bit 4 = 0(0): Aux3 durch F6 nicht gesperrt = 1(16): Aux3 durch F6 gesperrt Bit 5 = 0(0): Aux4 durch F6 nicht gesperrt = 1(32): Aux4 durch F6 gesperrt Bit 6 = 0(0): Aux5 durch F6 nicht gesperrt = 1(64): Aux5 durch F6 gesperrt Bit 7 = 0(0): Aux6 durch F6 nicht gesperrt = 1(128): Aux6 durch F6 gesperrt</p>



174 (* )	0	0-255	<p>Ausgänge mit F7 gesperrt</p> <p>Bit 0 = 0(0): FL durch F7 nicht gesperrt = 1(1): FL durch F7 gesperrt</p> <p>Bit 1 = 0(0): FR durch F7 nicht gesperrt = 1(2): FR durch F7 gesperrt</p> <p>Bit 2 = 0(0): Aux1 durch F7 nicht gesperrt = 1(4): Aux1 durch F7 gesperrt</p> <p>Bit 3 = 0(0): Aux2 durch F7 nicht gesperrt = 1(8): Aux2 durch F7 gesperrt</p> <p>Bit 4 = 0(0): Aux3 durch F7 nicht gesperrt = 1(16): Aux3 durch F7 gesperrt</p> <p>Bit 5 = 0(0): Aux4 durch F7 nicht gesperrt = 1(32): Aux4 durch F7 gesperrt</p> <p>Bit 6 = 0(0): Aux5 durch F7 nicht gesperrt = 1(64): Aux5 durch F7 gesperrt</p> <p>Bit 7 = 0(0): Aux6 durch F7 nicht gesperrt = 1(128): Aux6 durch F7 gesperrt</p>
175 (* )	0	0-255	<p>Ausgänge mit F8 gesperrt</p> <p>Bit 0 = 0(0): FL nu e durch F8 gesperrt = 1(1): FL durch F8 gesperrt</p> <p>Bit 1 = 0(0): FR nu e durch F8 gesperrt = 1(2): FR durch F8 gesperrt</p> <p>Bit 2 = 0(0): Aux1 nu e durch F8 gesperrt = 1(4): Aux1 durch F8 gesperrt</p>



			<p>Bit 3 = 0(0): Aux2 nu e durch F8 gesperrt = 1(8): Aux2 durch F8 gesperrt</p> <p>Bit 4 = 0(0): Aux3 nu e durch F8 gesperrt = 1(16): Aux3 durch F8 gesperrt</p> <p>Bit 5 = 0(0): Aux4 nu e durch F8 gesperrt = 1(32): Aux4 durch F8 gesperrt</p> <p>Bit 6 = 0(0): Aux5 nu e durch F8 gesperrt = 1(64): Aux5 durch F8 gesperrt</p> <p>Bit 7 = 0(0): Aux6 nu e durch F8 gesperrt = 1(128): Aux6 durch F8 gesperrt</p>
176 (* )	0	0-255	<p>Ausgänge mit F8 gesperrt</p> <p>Bit 0 = 0(0): FL durch F9 nicht gesperrt = 1(1): FL durch F9 gesperrt</p> <p>Bit 1 = 0(0): FR durch F9 nicht gesperrt = 1(2): FR durch F9 gesperrt</p> <p>Bit 2 = 0(0): Aux1 durch F9 nicht gesperrt = 1(4): Aux1 durch F9 gesperrt</p> <p>Bit 3 = 0(0): Aux2 durch F9 nicht gesperrt = 1(8): Aux2 durch F9 gesperrt</p> <p>Bit 4 = 0(0): Aux3 durch F9 nicht gesperrt = 1(16): Aux3 durch F9 gesperrt</p> <p>Bit 5 = 0(0): Aux4 durch F9 nicht gesperrt = 1(32): Aux4 durch F9 gesperrt</p> <p>Bit 6 = 0(0): Aux5 durch F9 nicht gesperrt</p>



			<p>= 1(64): Aux5 durch F9 gesperrt Bit 7 = 0(0): Aux6 durch F9 nicht gesperrt = 1(128): Aux6 durch F9 gesperrt</p>
177 (* )	0	0-255	<p>Ausgänge mit F10 gesperrt Bit 0 = 0(0): FL durch F10 nicht gesperrt = 1(1): FL durch F10 gesperrt Bit 1 = 0(0): FR durch F10 nicht gesperrt = 1(2): FR durch F10 gesperrt Bit 2 = 0(0): Aux1 durch F10 nicht gesperrt = 1(4): Aux1 durch F10 gesperrt Bit 3 = 0(0): Aux2 durch F10 nicht gesperrt = 1(8): Aux2 durch F10 gesperrt Bit 4 = 0(0): Aux3 durch F10 nicht gesperrt = 1(16): Aux3 durch F10 gesperrt Bit 5 = 0(0): Aux4 durch F10 nicht gesperrt = 1(32): Aux4 durch F10 gesperrt Bit 6 = 0(0): Aux5 durch F10 nicht gesperrt = 1(64): Aux5 durch F10 gesperrt Bit 7 = 0(0): Aux6 durch F10 nicht gesperrt = 1(128): Aux6 durch F10 gesperrt</p>
178 (* )	0	0-255	<p>Ausgänge mit F11 gesperrt Bit 0 = 0(0): FL durch F11 nicht gesperrt = 1(1): FL durch F11 gesperrt Bit 1 = 0(0): FR durch F11 nicht gesperrt</p>



			<p>= 1(2): FR durch F11 gesperrt Bit 2 = 0(0): Aux1 durch F11 nicht gesperrt = 1(4): Aux1 durch F11 gesperrt Bit 3 = 0(0): Aux2 durch F11 nicht gesperrt = 1(8): Aux2 durch F11 gesperrt Bit 4 = 0(0): Aux3 durch F11 nicht gesperrt = 1(16):Aux3 durch F11 gesperrt Bit 5 = 0(0): Aux4 durch F11 nicht gesperrt = 1(32):Aux4 durch F11 gesperrt Bit 6 = 0(0): Aux5 durch F11 nicht gesperrt = 1(64):Aux5 durch F11 gesperrt Bit 7 = 0(0): Aux6 durch F11 nicht gesperrt = 1(128):Aux6 durch F11 gesperrt</p>
179 (* )	0	0-255	<p>Ausgänge mit F12 gesperrt Bit 0 = 0(0): FL durch F12 nicht gesperrt = 1(1): FL durch F12 gesperrt Bit 1 = 0(0): FR durch F12 nicht gesperrt = 1(2): FR durch F12 gesperrt Bit 2 = 0(0): Aux1 durch F12 nicht gesperrt = 1(4): Aux1 durch F12 gesperrt Bit 3 = 0(0): Aux2 durch F12 nicht gesperrt = 1(8): Aux2 durch F12 gesperrt Bit 4 = 0(0): Aux3 durch F12 nicht gesperrt = 1(16): Aux3 durch F12 gesperrt</p>



			Bit 5 = 0(0): Aux4 durch F12 nicht gesperrt = 1(32): Aux4 durch F12 gesperrt Bit 6 = 0(0): Aux5 durch F12 nicht gesperrt = 1(64): Aux5 durch F12 gesperrt Bit 7 = 0(0): Aux6 durch F12 nicht gesperrt = 1(128): Aux6 durch F12 gesperrt
180 (* )	0	0-255	Ausgänge mit F1r gesperrt (F1 rückwärts) Bit 0 = 0(0): FL durch F1 (rückwärts) nicht gesperrt = 1(1): FL durch F1 (rückwärts) gesperrt Bit 1 = 0(0): FR nu e durch F1 (rückwärts) nicht gesperrt = 1(2): FR durch F1 (rückwärts) gesperrt Bit 2 = 0(0): Aux1 durch F1 (rückwärts) nicht gesperrt = 1(4): Aux1 durch F1 (rückwärts) gesperrt Bit 3 = 0(0): Aux2 durch F1 (rückwärts) nicht gesperrt = 1(8): Aux2 durch F1 (rückwärts) gesperrt Bit 4 = 0(0): Aux3 durch F1 (rückwärts) nicht gesperrt = 1(16): Aux3 durch F1 (rückwärts) gesperrt Bit 5 = 0(0): Aux4 durch F1 (rückwärts) nicht gesperrt = 1(32): Aux4 durch F1 (rückwärts) gesperrt Bit 6 = 0(0): Aux5 durch F1 (rückwärts) nicht gesperrt = 1(64): Aux5 durch F1 (rückwärts) gesperrt Bit 7 = 0(0): Aux6 durch F1 (rückwärts) nicht gesperrt = 1(128): Aux6 durch F1 (rückwärts) gesperrt
181	0	0,1	Letzter Status speichern = 1: Speichern; = 0: Nicht speichern





(*)			
182 (*)			Letzter Status FL, FR
183 (*)			Letzter Status F1 - F8
184 (*)			Letzter Status F9 - F16
185 (*)	0	0-15	PWM (Dimmen) an den Ausgängen ein/ausschalten (Aux7 – Aux10): Bit 0 = 0(0): Aux7 PWM signal am Ausgang mit Dimmen = 1(1): Aux7 Dauersignal am Ausgang ohne Dimmen Bit 1 = 0(0): Aux8 PWM signal am Ausgang mit Dimmen = 1(2): Aux8 Dauersignal am Ausgang ohne Dimmen Bit 2 = 0(0): Aux9 PWM signal am Ausgang mit Dimmen = 1(4): Aux9 Dauersignal am Ausgang ohne Dimmen Bit 3 = 0(0): Aux10 PWM signal am Ausgang mit Dimmen = 1(8): Aux10 Dauersignal am Ausgang ohne Dimmen



Mit (\*) gekennzeichnete CV-s können nur in Firmware-Version 3.5.207 und höher verwendet werden

## 27. Anhang Bits und Bytes

Wenn Sie die Werte der Konfigurationsvariablen (CV) ändern möchten, sollten Sie einige Überlegungen zur Darstellung von Zahlen im Binärformat beibehalten. Im Binärformat haben wir nur zwei Ziffern 0 und 1. Eine 8-Bit-Gruppe wird Byte genannt und repräsentiert eine Binärzahl von 8 Binärziffern. Die Konfigurationsvariablen CV sind Bytes, die im nichtflüchtigen Speicher der Decoder gespeichert sind. Bits eines Bytes sind von 0 bis 7 nummeriert. Bit 0 ist das niedrigstwertige Bit (LSB) und hat den Dezimalwert 1 und das höchstwertige Bit7 (MSB) den Dezimalwert von 128.

Einige Steuerstationen, die zum Ändern von CVs verwendet werden, zeigen den Wert an und lassen die Eingabe nur im Dezimalformat zu. In diesem Fall ist es gut zu wissen, wie Sie den Wert eines Bits des gelesenen Dezimalwerts finden oder wie Sie den Dezimalwert berechnen, den Sie in den CV eingeben müssen, anhand der gewünschten Bit-Konfiguration.

	MSB							LSB
Bitposition	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Bitwert	128	64	32	16	8	4	2	1

Tabelle 7.

Wenn wir die Bit Konfiguration kennen und den Dezimalwert ermitteln möchten, verwenden wir die folgende Berechnungsformel:

$$\text{Dec. wert} = B7 * 128 + B6 * 64 + B5 * 32 + B4 * 16 + B3 * 8 + B2 * 4 + B1 * 2 + B0$$

wobei B0 ... B7 den Wert des jeweiligen Bits (0 oder 1) darstellt

zum Beispiel: wenn B7=1, B5=1, B2=1, der Rest 0, ergibt das:

$$\text{Dec. wert} = 1 * 128 + 0 * 64 + 1 * 32 + 0 * 16 + 0 * 8 + 1 * 4 + 0 * 2 + 0 =$$



$$=128+32+4=164$$

Wenn wir die Bitkonfiguration aus dem Dezimalwert herausfinden möchten, machen wir das Gegenteil. Wir versuchen vom Dezimalwert abzuziehen die Bitwerte, mit MSB beginnen, und behalten die Differenz für die nächste Abnahme bei, bis wir Null erhalten. Für mögliche Abnahmen (bei positive Differenz) hat das Bit den Wert 1. Für die unmöglichen Abnahme (wenn die Differenz negativ ist) verlassen wir die Operation (der Wert des Bits ist Null) und fahren mit der nächsten Abnahme fort.

zum Beispiel: Wir möchten die Bitkonfiguration für den Dezimalwert 73 finden:

$73 - 128 = -55$	$\Rightarrow \text{Bit7} = 0$
$73 - 64 = 9$	$\Rightarrow \text{Bit6} = 1$
$9 - 32 = -23$	$\Rightarrow \text{Bit5} = 0$
$9 - 16 = -7$	$\Rightarrow \text{Bit4} = 0$
$9 - 8 = 1$	$\Rightarrow \text{Bit3} = 1$
$1 - 4 = -3$	$\Rightarrow \text{Bit2} = 0$
$1 - 2 = -1$	$\Rightarrow \text{Bit1} = 0$
$1 - 1 = 0$	$\Rightarrow \text{Bit0} = 1$

Das Cv-Tool ist ein kleines Hilfsprogramm, um den Wert von Dezimalbits in binäre und umgekehrt zu konvertieren oder um den Wert von erweiterten Adressen zu berechnen.

CV Tool V 1.01.005 ☰

---

CV / Bit Value Calculator

bit 0 , value = 2 to the power of 0 = 1

bit 1 , value = 2 to the power of 1 = 0

bit 2 , value = 2 to the power of 2 = 4

bit 3 , value = 2 to the power of 3 = 0

bit 4 , value = 2 to the power of 4 = 16

bit 5 , value = 2 to the power of 5 = 32

bit 6 , value = 2 to the power of 6 = 0

bit 7 , value = 2 to the power of 7 = 128

CV Value [0-255]

---

Long Address Calculator

Long Address [0-10239]

CV17 Value [192-231]

CV18 Value [0-255]

Es kann von diesem Link heruntergeladen werden:

<https://train-o-matic.com/downloads/software/cvTool.zip>



**Copyright © 2018 Tehnologistic SRL**

**Alle Rechte vorbehalten.**

**Die Informationen in diesem Dokument können sein  
ohne vorherige Ankündigung geändert**

**“train-O-matic” und das  logo sind  
eingetragene Warenzeichen  
der Tehnologistic SRL**

**[www.train-O-matic.com](http://www.train-O-matic.com)**

**ABC Technology und RailCom sind  
eingetragene Warenzeichen  
der Lenz Elektronik**

**<http://www.digital-plus.de>**

**SUSI und das  logo sind eingetragene Warenzeichen  
der DIETZ ELEKTRONIK**

**<http://www.d-i-e-t-z.de>**



**Tehnologic SRL  
Str. Libertatii 35A  
407035 Apahida  
Romania**

