



Befehlsbeschreibung  
XpressNet V3.6  
mit  
Interface LI-USB

Kommunikation auf der  
seriellen Schnittstelle

XPressNet Version 3.6

Doku Version 1.01 03/2009

<b>1</b>	<b>Allgemeines .....</b>	<b>4</b>
1.1	<b>Konventionen.....</b>	<b>4</b>
1.2	<b>Ungefragte Informationen .....</b>	<b>5</b>
1.3	<b>Antworten des LI-USB .....</b>	<b>5</b>
1.4	<b>Versionsnummer des LI-USB feststellen.....</b>	<b>7</b>
1.5	<b>Geräteadresse des LI-USB feststellen und ändern.....</b>	<b>8</b>
<b>2</b>	<b>Datenverkehr Zentrale und PC .....</b>	<b>9</b>
2.1	<b>Zentrale an PC .....</b>	<b>9</b>
2.1.1	Broadcast .....	9
2.1.1.1	BC „Alles An“ .....	9
2.1.1.2	BC „Alles Aus“ (Notaus) .....	10
2.1.1.3	BC „Alle Loks Aus“ (Nothalt) .....	10
2.1.1.4	BC „Programmiermode“ .....	11
2.1.1.5	BC „Rückmeldung“ .....	11
2.1.2	Programmierinformationen.....	12
2.1.2.1	Programmierinfo „Kurzschluß“ .....	12
2.1.2.2	Programmierinfo „Daten nicht gefunden“ .....	12
2.1.2.3	Programmierinfo „Zentrale Busy“ .....	13
2.1.2.4	Programmierinfo „Zentrale Bereit“ .....	13
2.1.2.5	Programmierinfo „Daten 3-Byte-Format“ .....	13
2.1.2.6	Programmierinfo „Daten 4-Byte-Format“ CV 1-255 und CV1024 (neu ab Version 3.6; ersetzt gleichen Befehl bis Version 3) .....	14
2.1.2.7	Programmierinfo „Daten 4-Byte-Format“ CV256 bis CV511 (neu ab Version 3.6) .....	15
2.1.2.8	Programmierinfo „Daten 4-Byte-Format“ CV512 bis CV767 (neu ab Version 3.6) .....	15
2.1.2.9	Programmierinfo „Daten 4-Byte-Format“ CV768 bis CV1023 (neu ab Version 3.6) .....	16
2.1.3	Softwareversion Zentrale .....	17
2.1.4	Status Zentrale .....	17
2.1.5	Übertragungsfehler.....	18
2.1.6	Zentrale Busy .....	19
2.1.7	Befehl in Zentrale nicht vorhanden.....	19
2.1.8	Schaltinformation.....	20
2.1.9	Lokinformationen.....	22
2.1.9.1	Lokinformation normale Lok .....	22
2.1.9.2	Funktionszustand F13 bis F28 der angefragten Lok (ab Version 3.6) .....	24
2.1.9.3	Lokinformation Lok befindet sich in einer Mehrfachtraktion .....	24
2.1.9.4	Lokinformation Lokadresse ist die Basisadresse einer Mehrfachtraktion .....	25
2.1.9.5	Lokinformation Lok befindet sich in einer Doppeltraktion .....	25
2.1.10	Lok besetzt ab Zentralen-Version 3.0 .....	26
2.1.11	Funktionsstatus F0 bis F12 der angefragten Lok.....	26
2.1.12	Funktionsstatus F13 bis F28 der angefragten Lok (ab Version 3.6) .....	27
2.1.13	Lokinformation bei Adress-Suchanfragen .....	28
2.1.14	Fehlermeldungen.....	28
<b>2.2</b>	<b>PC an Zentrale .....</b>	<b>29</b>
2.2.1	Alles An .....	29
2.2.2	Alles Aus (Notaus).....	29
2.2.3	Alle Loks anhalten (Nothalt) .....	30
2.2.4	Eine Lok anhalten (Nothalt für eine Lok) .....	30
2.2.5	Leseanfrage Programmieren 3-Byte-Format (Registermode) .....	31
2.2.6	Leseanfrage Programmieren 4-Byte-Format (CV-Mode).....	31
2.2.7	Leseanfrage Programmieren 4-Byte-Format (CV 1-255 und CV1024) (neu ab V3.6).....	32
2.2.8	Leseanfrage Programmieren 4-Byte-Format (CV 256-511) (neu ab V3.6).....	32

2.2.9	Leseanfrage Programmieren 4-Byte-Format (CV 512-767) (neu ab V3.6)	33
2.2.10	Leseanfrage Programmieren 4-Byte-Format (CV 768-1023) (neu ab V3.6)	33
2.2.11	Leseanfrage Programmieren 3-Byte-Format (Pagemode)	34
2.2.12	Programmierergebnis anfordern	35
2.2.13	Schreibbefehl Programmieren 3-Byte-Format (Register-Mode)	35
2.2.14	Schreibbefehl Programmieren 4-Byte-Format (CV-Mode, CV1-256)	35
2.2.15	Schreibbefehl Programmieren 4-Byte-Format (CV-Mode, CV 1-255 und CV1024) (neu ab Version 3.6)	36
2.2.16	Schreibbefehl Programmieren 4-Byte-Format (CV-Mode, CV 256-511) (neu ab Version 3.6)	37
2.2.17	Schreibbefehl Programmieren 4-Byte-Format (CV-Mode, CV 512-767) (neu ab Version 3.6)	37
2.2.18	Schreibbefehl Programmieren 4-Byte-Format (CV-Mode, CV 768-1023) (neu ab Version 3.6)	38
2.2.19	Schreibbefehl Programmieren 3-Byte-Format (Page-Mode)	38
2.2.20	Softwareversion der Zentrale anfordern	39
2.2.21	Status der Zentrale anfordern	39
2.2.22	Zentralen-Startmode setzen	40
2.2.23	Schaltinformationen anfordern	40
2.2.24	Schaltbefehl	41
2.2.25	Lokinformationen anfordern	42
2.2.25.1	Funktionsstatus anfordern	42
2.2.25.2	Funktionsstatus anfordern F13 – F28 (neu ab Zentralen-Version 3.6)	42
2.2.25.3	Funktionszustand anfordern F13 – F28 (neu ab Zentralen-Version 3.6)	43
2.2.26	Lok steuern	43
2.2.26.1	Fahrbefehle	43
2.2.26.2	Funktionsbefehle ab Zentralen-Version 3.0 / Version 3.6	44
2.2.26.3	Funktionsstatus setzen	46
2.2.26.4	Funktionsstatus setzen ab Zentralen-Version 3.0 / Version 3.6	46
2.2.26.5	Funktionsrefresh-Modus setzen ab Zentralen-Version 3.6	48
2.2.27	Doppeltraktionen	49
2.2.27.1	Doppeltraktion montieren	49
2.2.27.2	Doppeltraktion auflösen	49
2.2.28	Programming on Main	50
2.2.28.1	Programming on Main Byte schreiben	50
2.2.28.2	Programming on Main Byte lesen (ab Version 3.6)	51
2.2.28.3	Programming on Main Bit schreiben	51
2.2.29	Mehrfachtraktionen	52
2.2.29.1	Lok zu einer Mehrfachtraktion hinzufügen oder MTR erzeugen	52
2.2.29.2	Lok aus einer Mehrfachtraktion entfernen oder MTR löschen	53
2.2.30	Adress-Suchbefehle	53
2.2.30.1	Adressanfrage Mitglied einer Mehrfachtraktion	53
2.2.30.2	Adressanfrage Mehrfachtraktion	54
2.2.30.3	Adressanfrage Lok in Zentralenstack	55
2.2.31	Lok aus Zentralenstack löschen	55
<b>3</b>	<b>Befehlsübersicht Zentrale an PC</b>	<b>57</b>
<b>4</b>	<b>Befehlsübersicht PC an Zentrale</b>	<b>58</b>

# 1 Allgemeines

Die vorliegende Dokumentation enthält die Befehlsbeschreibung des XpressNet für die Benutzung des Interface über die serielle Schnittstelle.

Dies ist gültig für den Fall, dass das Interface über den Virtuellen Com Port Treiber von FTDI betrieben wird oder für den Fall, dass die Daten per FTDI2XX Blocktreiber ausgewertet werden.

- Baudrate: immer 57600 Bit pro Sekunde
- 8 Datenbits, 1 Startbit, 1 Stopbit, kein Paritybit
- kein Handshake

## 1.1 Konventionen

Jeder Befehl, der an das Interface gesendet wird, muss die Sequenz 0xFF 0xFE vorangestellt bekommen. Diese Sequenz fließt nicht in die Berechnung der Checksumme ein.

Als Beispiel, um den Befehl "Alles an" an das Interface zu senden, müssen folgende Daten übertragen werden :

0xFF 0xFE 0x21 0x81 0xA0

Jeder Befehl an das Interface wird mit einer Antwort quittiert, dies ist entweder die allgemeine Antwort

0xFF 0xFE 0x01 0x04 0x05

die besagt, dass der Befehl erfolgreich an die Zentrale abgesetzt wurde, oder eine Fehlermeldung oder die erwarteten Daten.

Antworten auf einen Befehl werden ebenfalls immer mit dem Header 0xFF 0xFE geschickt.

Achtung: Im Programmiermodus kann es bis zu einer Minute dauern, bis eine Antwort erfolgt.

Solange das Interface einen Befehl nicht quittiert hat, darf kein weiterer Befehl an das Interface gesendet werden, erst nach Ablauf des Timeouts (5 sek im Normalen Betrieb, 1,5 Min im Programmiermodus) darf ein neuer Befehl gesendet werden

Wurde ein Befehl an das Interface gesendet und hat dieses noch keine Antwort zurückgesendet, so ist die nächste Antwort des Interface immer die Befehlsantwort. Zwischenzeitlich im Interface auflaufende Broadcasts werden im Interface gespeichert und erst nach der Befehlsantwort an den Computer gesendet.

Broadcasts sowie unerwartete Meldungen tragen als Header die Kennung 0xFF 0xFD, um diese einwandfrei von Befehlsantworten unterscheiden zu können.

## 1.2 Ungefragte Informationen

Ungefragte Informationen an eines oder alle Geräte, also auch den PC, werden immer dann verschickt, wenn Anlagenzustände allen Geräten bekannt gemacht werden müssen, damit diese in ihrem Verhalten schnellstmöglich korrekt reagieren können. Ungefragte Infos werden entweder als Broadcast verschickt, wenn alle Slaves sie erhalten sollen oder als Antwort formatiert, wenn es nur einen bestimmten Slave betrifft. Kennzeichnend ist immer, daß ein Slave diese Informationen nicht anfragt (also eigentlich nicht mit ihnen rechnet), sie trotzdem zu einem beliebigen Zeitpunkt erhält und richtig bei z.B. Eingaben (Lok-Fahrbefehle) reagieren muß. Ungefragte Infos sind:

Broadcast „Alles An“	(an alle Teilnehmer)
Broadcast „Alles Aus“	(an alle Teilnehmer)
Broadcast „Alle Loks Aus“	(an alle Teilnehmer)
Broadcast „Programmiermode“	(an alle Teilnehmer)
Broadcast „Rückmeldung“	(an alle Teilnehmer)
Broadcast „Railcom-Info“	(an alle Teilnehmer)
Antwort „Lok besetzt“	(an denjenigen Teilnehmer, der die Lok gerade im Zugriff hatte)
Antwort „Doppeltraktion besetzt“	(an denjenigen Teilnehmer, der die Doppeltraktion gerade im Zugriff hatte)

Die Antworten „Übertragungsfehler“, „Zentrale Busy“, „Doppeltraktionsfehler“ und „Befehl nicht vorhanden“ sind keine ungefragten Infos, da diese Antworten grundsätzlich auf Befehle eines Slaves an die Zentrale kommen können. Sie sind also zeitlich an den Befehl an die Zentrale gekoppelt, auch wenn dieser normalerweise keine Antwort zur Folge hat.

## 1.3 Antworten des LI-USB

Die Befehle, die der PC über das LI-USB an die Zentrale schickt, gliedern sich in zwei Bereiche: zum einen Befehle, die eine direkte Reaktion der Zentrale ohne Bearbeitung anderer XpressNet-Geräte zur Folge haben (z.B. Lokanfrage, aber auch vom PC ausgelöste Broadcasts!) und zum anderen in Befehle, die keine Zentralenantwort zur Folge haben (z.B. Fahrbefehl erteilen). Damit aber im PC-Programm eine Zuordnung zwischen gesendeten und empfangenen Daten möglich wird, wird auf jeden Fall nach einem Befehl des PC an diesen eine Antwort zurückgeschickt.

Folgende Meldungen werden vom LI-USB als Antwort an den PC gesendet, falls keine anderen Zentralendaten verfügbar sind (jeweils in dezimaler Darstellung):

Frame 1	Frame 2	Header	Meldung	X- Or	Bedeutung
255	254	01	02	03	Fehler zwischen Interface und Zentrale (Timeout bei Datenübertragung LI an Zentrale)
255	254	01	03	02	unbekannter Fehler (Zentrale adressierte LI-USB mit Quittierungsaufforderung)
255	254	01	04	05	Befehl ist an Zentrale geschickt
255	253	01	05	04	Zentrale adressiert LI-USB nicht mehr (x)
255	254	01	06	07	Puffer-Überlauf im LI-USB
255	253	01	07	06	Zentrale adressiert LI-USB wieder (x)
255	254	01	08	09	Derzeit können keine Befehle an die Zentrale gesendet werden
255	254	01	09	08	Fehler in den Befehlsparametern (z.B. Lokadresse falsch)
255	254	01	10	11	unbekannter Fehler (Zentrale lieferte nicht die erwartete Antwort)

Hierbei bedeutet:

255 / 254 / 01 / 01 / 00:

Beim ersten Byte, das der PC an das LI-USB schickt, wird die Zahl der noch folgenden Bytes festgestellt. Wird diese Anzahl Bytes in einer bestimmten Zeit vom PC nicht gesendet, wird bei Erreichen des Timeouts diese Meldung gesendet. Diese Meldung tritt oft dann auf, wenn zuvor Übertragungsfehler erkannt wurden.

255 / 254 / 01 / 02 / 03:

Die Reaktion der Zentrale auf einen an sie gesendeten Befehl muß ebenfalls innerhalb einer bestimmten Zeit erfolgen, anderenfalls diese Meldung an den PC gesendet wird.

255 / 254 / 01 / 03 / 02:

Kommt es während einer Datenübertragung an die Zentrale zu einem Fehler, so wird das gerade bediente XpressNet-Gerät (LI-USB) noch einmal angesprochen und muß dann eine Quittierung senden. Der PC erhält hierüber diese Information. Tritt dies wiederholt auf, so ist zunächst die Verkabelung zu prüfen. Diese Meldung, daß das LI-USB an die Zentrale eine Quittierung gesendet hat, ist nicht zu verwechseln mit der Meldung „Übertragungsfehler“, die an den PC gesendet wird und eine Antwort zu einem vorangegangenen Befehls darstellt.

255 / 254 / 01 / 04 / 05:

Wird immer dann an den PC gesendet, wenn ein Befehl, den das LI-USB an die Zentrale geleitet hat, keine Antwort zur Folge hatte (z.B. ein Lok-Fahrbehl). D.h. wenn nach Senden eines Befehles die Zentrale das Interface wieder adressiert hat. Die Meldung kommt auch als Bestätigung, daß die Zentrale das LI-USB wieder anspricht, nachdem dies für eine bestimmte Zeit nicht der Fall war (nach einem Timeout beim Adressieren). Hat ein Befehl, der normalerweise keine Zentralenantwort hat, z.B. die Antwort „Übertragungsfehler“ zur Folge, dann wird vom LI-USB keine Befehlsbestätigung gesendet.

Es ist zu beachten, daß seitens des PC-Programmes (und auch des LI-USB) keine Möglichkeit besteht, festzustellen, ob ein Befehl von der Zentrale auch schon auf das Gleis gelegt worden ist. Die Meldung „Befehl verschickt“ besagt lediglich, daß der zugehörige Befehl an die Zentrale gereicht werden konnte. Wird vom PC ein Befehl versendet, der eine Broadcast zur Folge hat (z.B. „Alles An“), so wird dies nicht mit

dieser Meldung bestätigt, weil hier eine direkte Zentralenreaktion aus Sicht des auslösenden Gerätes erfolgt. Beispiel: Der PC sendet „Alles An“ an die Zentrale, aber die Anlage kann von der Zentrale nicht eingeschaltet werden. Vom sendenden Gerät aus gesehen (dem PC) hat der Befehl „Alles An“ die Antwort „Alles Aus“, für alle anderen Geräte kommt dieser Broadcast ungefragt. Diese Zentralenantwort muß vom PC auch auf ihren Inhalt untersucht werden, um den Erfolg oder Mißerfolg des gesendeten Befehles (z.B. „Alles An“) festzustellen.

255 / 253 / 01 / 05 / 04:

Die Zentrale adressiert angeschlossene Geräte in einem bestimmten Zeitintervall. Passiert das nicht, wird diese Meldung an den PC gesendet. Diese Meldung kommt ungefragt.

255 / 254 / 01 / 06 / 07:

Falls zu viele Daten ohne ein Trennframe (0xFF 0xFE) an das Interface gesendet werden, läuft der Eingangspuffer über und diese Meldung wird generiert.

255 / 253 / 01 / 07 / 06:

Die Zentrale adressiert das LI-USB wieder (zum Beispiel nach dem Programmiermodus). Diese Meldung ist üblicherweise auch die erste Meldung, die das Interface aussendet. Diese Meldung kommt ungefragt.

255 / 254 / 01 / 08 / 09:

Diese Fehlermeldung gibt das Interface zurück wenn versucht wird, einen Befehl zu senden, während das Interface nicht von der Zentrale adressiert wird (zum Beispiel, während ein anderer Handregler sich im Programmiermodus befindet)

255 / 254 / 01 / 09 / 08:

Wurden Fehler bei den Parametern gemacht, so antwortet das LI-USB mit diesem Befehl. So zum Beispiel, wenn eine DTR grösser 99 angegeben wurde. Das LI-USB prüft alle Befehle auf Plausibilität.

255 / 254 / 01 / 10 / 11:

Wird diese Fehlermeldung vom LI-USB gesendet, so ist davon auszugehen, dass der letzte Befehl nicht von der Zentrale verstanden wurde. Daher empfiehlt es sich, den letzten Befehl zu wiederholen.

#### 1.4 Versionsnummer des LI-USB feststellen

Das Auslesen der Versionsnummer des LI-USB ist eine Aktion, die nur zwischen PC und LI-USB stattfindet. Der Befehlsaufbau und die zugehörige LI-USB-Antwort entsprechen aber trotzdem dem in Kapitel 2 beschriebenen Format.

##### Befehl zum Auslesen der Versions- und Codenummer:

	Frame1	Frame2	Headerbyte	X-Or-Byte
Binär :	1111 1111	1111 1110	1111 0000	1111 0000
Hex :	0xFF	0xFE	0xF0	0xF0
Dez :	255	254	240	240

**Antwort des LI-USB:**

	Frame1	Frame2	Headerbyte	Daten 1	Daten 2	X-Or-Byte
Binär :	1111 1111	1111 1110	0000 0010	VVVV VVVV	CCCC CCCC	X-Or-Byte
Hex :	0xFF	0xFE	0x02	VV	CC	X-Or-Byte
Dez :	255	254	2	VV	CC	X-Or-Byte

**Beschreibung:**

VV gibt die Versionsnummer des LI-USB hexadezimal in BCD-Darstellung an. CC gibt die Codenummer des LI-USB hexadezimal in BCD-Darstellung an. Beispiel:

Antwort = 0x02 0x30 0x01 0x33

Versionsnummer 3.0, Codenummer 01

**Besonderheiten:**

Dies ist der einzige Befehl, den das LI-USB selbst auswertet. Alles, was nicht diesem Befehl entspricht, wird vom LI-USB an die Zentrale weitergeleitet und inhaltlich nicht geprüft.

**1.5 Geräteadresse des LI-USB feststellen und ändern**

Das Auslesen der Versionsnummer des LI-USB ist eine Aktion, die nur zwischen PC und LI-USB stattfindet. Der Befehlsaufbau und die zugehörige LI-USB-Antwort entsprechen aber trotzdem dem in Kapitel 2 beschriebenen Format.

**Befehl zum Auslesen der Geräteadresse:**

	Frame1	Frame2	Headerbyte	Daten 1	Daten2	X-Or-Byte
Binär :	1111 1111	1111 1110	1111 0010	0000 0001	ADR	X-Or-Byte
Hex :	0xFF	0xFE	0xF2	0x01	ADR	X-Or-Byte
Dez :	255	254	242	1	ADR	X-Or-Byte

**Antwort des LI-USB:**

	Frame1	Frame2	Headerbyte	Daten 1	Daten2	X-Or-Byte
Binär :	1111 1111	1111 1110	1111 0010	0000 0001	ADR	X-Or-Byte
Hex :	0xFF	0xFE	0xF2	0x01	ADR	X-Or-Byte
Dez :	255	254	242	1	ADR	X-Or-Byte

**Beschreibung:**

ADR gibt die XpressNet-Geräteadresse an, die das LI-USB verwenden soll. Der erlaubte Bereich liegt zwischen 1 und 31 dez. In der Antwort des LI-USB ist in Daten 2 die Adresse zu finden, auf die das Gerät eingestellt wurde. Üblicherweise sind also Befehl und Antwort identisch.

**Besonderheiten:**

Wird im Befehl an das LI-USB die Geräteadresse nicht im Bereich von 1 bis 31 angegeben, so antwortet das LI-USB mit seiner aktuell eingestellten Adresse. Damit kann die Adresse festgestellt werden, ohne sie zu ändern.



## 2 Datenverkehr Zentrale und PC

### 2.1 Zentrale an PC

Befehlsaufbau:

Die Zentrale sendet an das Interface Daten, die dieses dann sofort an den PC weiterreicht. Die Daten können vom PC vorher angefordert worden sein oder sie ergaben sich aufgrund von Veränderungen auf der Anlage (ungefragt).

Es wird ein Headerbyte gesendet, ein bis maximal 15 Datenbytes und ein X-Or-Byte.

Im Headerbyte wird im unteren Nibble die Anzahl der noch folgenden Datenbytes eingetragen.

Vereinbarungen für die folgenden Befehlsbeschreibungen:

N = Anzahl der noch folgenden Datenbytes (binär und hexadezimal)

GA = Geräteadresse

Das Befehlsformat wird sowohl binär als auch dezimal und hexadezimal angegeben.

Nicht alle Zentralen unterstützen alle Befehle. Dies muß in einem PC-Programm beachtet werden, um Endlosschleifen zu vermeiden (siehe auch Kap. 2.1.7 , Befehl nicht vorhanden).

#### 2.1.1 Broadcast

Die Rufgruppe „Broadcast“ gibt der Zentrale die Möglichkeit, an alle Slaves gleichzeitig Informationen zu senden. Also auch an den PC. Ein Broadcast wird mehrmals hintereinander ausgesandt, um sicherzustellen, daß jeder Teilnehmer ihn empfangen kann. Einige Befehle an die Zentrale lösen einen solchen Broadcast aus (z.B. „Notaus“). Ein Gerät, welches einen Broadcast auslöst, muß selbst dafür sorgen, daß es zu keinen internen Unstimmigkeiten kommt, wenn es sofort danach diesen Broadcast selbst wieder erhält (wenn z.B. die Zentrale in den Programmiermode gesetzt wurde).

##### 2.1.1.1 BC „Alles An“

**Format:**

	Frame1	Frame2	Headerbyte	Daten 1	X-Or-Byte
Binär :	1111 1111	1111 1101	0110 0001	0000 0001	0110 0000
Hex :	0xFF	0xFD	0x61	0x01	0x60
Dez :	255	253	97	1	96

**Beschreibung:**

Sendet der PC den Befehl „Alles An“ (siehe Abschnitt Gerät an Zentrale), so wird zur Information für alle Teilnehmer der Broadcast „Alles An“ gesendet. Dieser Broadcast entspricht dann dem tatsächlichen Anlagenzustand. Steht z.B. ein Notaus an, der nicht

aufgehoben werden kann und ein Busteilnehmer sendet „Alles An“, so erfolgt der Broadcast „Alles Aus“!

### **Besonderheiten:**

Dieser Ruf wird ohne Anfrage eines XpressNet-Gerätes verschickt. Er ist eine ungefragte Info.

#### **2.1.1.2 BC „Alles Aus“ (Notaus)**

### **Format:**

	Frame1	Frame2	Headerbyte	Daten 1	X-Or-Byte
Binär :	1111 1111	1111 1101	0110 0001	0000 0000	0110 0001
Hex :	0xFF	0xFD	0x61	0x00	0x61
Dez :	255	253	97	0	97

### **Beschreibung:**

Die Zentrale sendet hiermit die Information, daß die Gleisspannung abgeschaltet wurde und deswegen kein Schalt- oder Fahrbefehl mehr verschickt werden kann.

### **Besonderheiten:**

Dieser Ruf wird ohne Anfrage eines XpressNet-Gerätes verschickt. Er ist eine ungefragte Info.

#### **2.1.1.3 BC „Alle Loks Aus“ (Nothalt)**

### **Format:**

	Frame1	Frame2	Headerbyte	Daten 1	X-Or-Byte
Binär :	1111 1111	1111 1101	1000 0001	0000 0000	0110 0001
Hex :	0xFF	0xFD	0x81	0x00	0x81
Dez :	255	253	129	0	129

### **Beschreibung:**

Die Zentrale sendet hiermit die Information, daß alle Loks auf dem Gleis mittels eines Broadcast (gleisseitig) angehalten worden sind. Die Gleisspannung liegt weiterhin an, so daß Schaltbefehle verschickt werden können, jedoch wird keine Lok mehr adressiert, bis alles wieder eingeschaltet wurde.

### **Besonderheiten:**

Dieser Ruf wird ohne Anfrage eines XpressNet-Gerätes verschickt. Er ist eine ungefragte Info.

### 2.1.1.4 BC „Programmiermode“

#### Format:

	Frame1	Frame2	Headerbyte	Daten 1	X-Or-Byte
Binär :	1111 1111	1111 1101	0110 0001	0000 0010	0110 0011
Hex :	0xFF	0xFD	0x61	0x02	0x63
Dez :	255	253	97	2	99

#### Beschreibung:

Wird dieser Ruf an alle Busgeräte (auch den PC) geschickt, so stellt dies eine Information darüber dar, daß jetzt Programmieraktionen laufen. Es wird danach kein XpressNet-Gerät mehr adressiert außer demjenigen, das die Programmieraktion (durch z.B. einen Programmier-Lesebefehl an die Zentrale) ausgelöst hat. D.h. wenn der PC den Programmiermode ausgelöst hat, kann er weiterhin mit dem LI-USB kommunizieren. Hat ein anderes Gerät den Programmiermode ausgelöst, so kann kein Befehl an das LI-USB gesendet werden. Der Programmiermodus kann wieder aufgehoben werden, indem das auslösende Gerät den Befehl „Alles An“ sendet.

#### Besonderheiten:

Dieser Ruf wird ohne Anfrage eines XpressNet-Gerätes verschickt. Er ist eine ungefragte Info.

### 2.1.1.5 BC „Rückmeldung“

#### Format:

	Frame1	Frame2	Headerbyte	Daten 1	Daten 2	Daten 3	Daten 4	usw.	X-Or-Byte
Binär :	1111 1111	1111 1101	0100 NNNN	ADR_1	DAT_1	ADR_2	DAT2	usw.	X-Or-Byte
Hex :	0xFF	0xFD	0x40 + N						X-Or-Byte
Dez :	255	253	64 + N						X-Or-Byte

#### Beschreibung:

Mit diesem Ruf teilt die Zentrale allen Slaves mit, daß sich ein oder mehrere Rückmelde-zustände geändert haben. Nur bei Änderungen wird der Ruf verschickt. In einem Broadcast wird mindestens ein Adresszustand, maximal 7 Zustände übertragen (Je Adresse ein Datenbyte, insgesamt 15 Byte pro Ruf ohne Header und X-Or-Byte). ADR\_x und DAT\_x haben das Format wie unter „Schaltinformationen“ beschrieben. Ein Gerät muß für z.B. eine korrekte Anzeige des Zustandes eines Rückmeldebausteins den gesamten Inhalt der Broadcast auf die gewünschte Adresse untersuchen.

#### Besonderheiten:

Dieser Ruf wird ohne Anfrage eines XpressNet-Gerätes verschickt. Er ist eine ungefragte Info.

### 2.1.2 Programmierinformationen

Nach Erteilung eines Programmier-Lesebefehls wird die Zentrale in den Programmiermodus versetzt. Mit einem sich daran anschließenden Ergebnis-Lesebefehl antwortet die Zentrale mit einer der hier beschriebenen Antworten. Befindet sich die Zentrale nicht im Programmier-modus und ein Ergebnis-Lesebefehl wurde von einem Slave verschickt, so wird als Antwort von der Zentrale „Befehl nicht vorhanden“ gesendet.

#### 2.1.2.1 Programmierinfo „Kurzschluß“

##### **Format:**

	Frame1	Frame2	Headerbyte	Daten 1	X-Or-Byte
Binär :	1111 1111	1111 1110	0110 0001	0001 0010	X-Or-Byte
Hex :	0xFF	0xFE	0x61	0x12	X-Or-Byte
Dez :	255	254	97	18	X-Or-Byte

##### **Beschreibung:**

Bei Auslesen oder Beschreiben eines Empfängers am Programmier-Anschluß der Zentrale ist ein Kurzschluß bzw. ein zu hoher Strom aufgetreten. Es ist davon auszugehen, daß bei einem Schreibbefehl an eine Speicherstelle des Empfängers diese nicht oder falsch beschrieben wurde. Das Programmieren sollte dann mit einer Fehlermeldung abgebrochen werden und falls intern Daten des Empfängers verwendet wurden, diese auf ihre ursprünglichen Werte zurückgesetzt werden.

##### **Besonderheiten:**

Keine.

#### 2.1.2.2 Programmierinfo „Daten nicht gefunden“

##### **Format:**

	Frame1	Frame2	Headerbyte	Daten 1	X-Or-Byte
Binär :	1111 1111	1111 1110	0110 0001	0001 0011	X-Or-Byte
Hex :	0xFF	0xFE	0x61	0x13	X-Or-Byte
Dez :	255	254	97	19	X-Or-Byte

##### **Beschreibung:**

Am Programmieranschluß der Zentrale befindet sich kein Empfänger oder der Empfänger antwortet nicht auf den Leseversuch der Zentrale. Das Programmieren dieses Empfängers sollte abgebrochen oder neu versucht werden.

##### **Besonderheiten:**

Keine.

### 2.1.2.3 Programmierinfo „Zentrale Busy“

#### Format:

	Frame1	Frame2	Headerbyte	Daten 1	X-Or-Byte
Binär :	1111 1111	1111 1110	0110 0001	0001 1111	X-Or-Byte
Hex :	0xFF	0xFE	0x61	0x1f	X-Or-Byte
Dez :	255	254	97	31	X-Or-Byte

#### Beschreibung:

Dieser Befehl wird bis einschließlich Zentralen-Version 3.0 noch nicht verwendet.

#### Besonderheiten:

Keine.

### 2.1.2.4 Programmierinfo „Zentrale Bereit“

#### Format:

	Frame1	Frame2	Headerbyte	Daten 1	X-Or-Byte
Binär :	1111 1111	1111 1110	0110 0001	0001 0001	X-Or-Byte
Hex :	0xFF	0xFE	0x61	0x11	X-Or-Byte
Dez :	255	254	97	17	X-Or-Byte

#### Beschreibung:

Dieser Befehl wird bis einschließlich Zentralen-Version 3.0 noch nicht verwendet.

#### Besonderheiten:

Keine.

### 2.1.2.5 Programmierinfo „Daten 3-Byte-Format“

#### Format:

	Frame1	Frame2	Headerbyte	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	1111 1111	1111 1110	0110 0011	0001 0000	EEEE EEEE	DDDD DDDD	X-Or-Byte
Hex :	0xFF	0xFE	0x63	0x10	E	D	X-Or-Byte
Dez :	255	254	99	16	E	D	X-Or-Byte

#### Beschreibung:

Diese Antwort wird nur auf Anfrage desjenigen Slaves gegeben, der die Zentrale in den Programmiermode versetzt hat. Zurückgeliefert wird die EEPROM-Adresse (E) und die darin gelesenen Daten (D). Nur bei Register- und Pagemode erfolgt diese Antwort!

**Besonderheiten:**

Die Antwort bezieht sich auf Programmieraktionen im Register- und Pagemode. Wurde ein Empfänger jedoch mit CV-Lesen angefragt und man erhält diese Antwort, dann kann der Empfänger mit der CV-Programmierung nicht umgehen (alter Empfänger). Für weitere Programmieraktionen muß man nun Schreib- und Lesebefehle für Register- und Pagemode verwenden.

2.1.2.6 Programmierinfo „Daten 4-Byte-Format“ CV 1-255 und CV1024 (neu ab Version 3.6; ersetzt gleichen Befehl bis Version 3)

**Format:**

	Frame1	Frame2	Headerbyte	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	1111 1111	1111 1110	0110 0011	0001 0100	CCCC CCCC	DDDD DDDD	X-Or-Byte
Hex :	0xFF	0xFE	0x63	0x14	C	DAT	X-Or-Byte
Dez :	255	254	99	20	C	DAT	X-Or-Byte

**Beschreibung:**

Diese Antwort wird nur auf Anfrage desjenigen Slaves gegeben, der die Zentrale in den Programmiermode versetzt hat. Zurückgeliefert wird die CV-Adresse (C) und die darin gelesenen Daten (D). Dies nur bei CV-Programmierung von Empfängern, die diesen Mode beherrschen.

Zuordnung Wert in C  $\Leftrightarrow$  CV – Adressen:

C	CV
0	1024
1 ... 255	1 ... 255

**Besonderheiten:**

Wurde ein Empfänger mit CV-Lesen angefragt und man bekommt diese Antwort, ist alles ok, denn der Empfänger kann damit umgehen. Diese Antwort muß aber nicht zwangsläufig bei CV-Anfrage zurückkommen. Ein Gerät muß dies beachten und dann den Empfänger in Register- oder Pagemode programmieren. Siehe dazu 2.1.2.5.

### 2.1.2.7 Programmierinfo „Daten 4-Byte-Format“ CV256 bis CV511 (neu ab Version 3.6)

#### Format:

	Frame1	Frame2	Headerbyte	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	1111 1111	1111 1110	0110 0011	0001 0101	CCCC CCCC	DDDD DDDD	X-Or-Byte
Hex :	0xFF	0xFE	0x63	0x15	C	DAT	X-Or-Byte
Dez :	255	254	99	21	C	DAT	X-Or-Byte

#### Beschreibung:

Diese Antwort wird nur auf Anfrage desjenigen Slaves gegeben, der die Zentrale in den Programmiermode versetzt hat. Zurückgeliefert wird die CV-Adresse (C) und die darin gelesenen Daten (D). Dies nur bei CV-Programmierung von Empfängern, die diesen Mode beherrschen.

Zuordnung Wert in C <=> CV – Adressen:

C	CV
0 ... 255	256 ... 511

#### Besonderheiten:

Siehe 2.1.2.6.

### 2.1.2.8 Programmierinfo „Daten 4-Byte-Format“ CV512 bis CV767 (neu ab Version 3.6)

#### Format:

	Frame1	Frame2	Headerbyte	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	1111 1111	1111 1110	0110 0011	0001 0110	CCCC CCCC	DDDD DDDD	X-Or-Byte
Hex :	0xFF	0xFE	0x63	0x16	C	DAT	X-Or-Byte
Dez :	255	254	99	22	C	DAT	X-Or-Byte

#### Beschreibung:

Diese Antwort wird nur auf Anfrage desjenigen Slaves gegeben, der die Zentrale in den Programmiermode versetzt hat. Zurückgeliefert wird die CV-Adresse (C) und die darin gelesenen Daten (D). Dies nur bei CV-Programmierung von Empfängern, die diesen Mode beherrschen.

Zuordnung Wert in C  $\Leftrightarrow$  CV – Adressen:

C	CV
0 ... 255	512 ... 767

### Besonderheiten:

Siehe 2.1.2.6.

### 2.1.2.9 Programmierinfo „Daten 4-Byte-Format“ CV768 bis CV1023 (neu ab Version 3.6)

#### Format:

	Frame1	Frame2	Headerbyte	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	1111 1111	1111 1110	0110 0011	0001 0111	CCCC CCCC	DDDD DDDD	X-Or-Byte
Hex :	0xFF	0xFE	0x63	0x17	C	DAT	X-Or-Byte
Dez :	255	254	99	23	C	DAT	X-Or-Byte

#### Beschreibung:

Diese Antwort wird nur auf Anfrage desjenigen Slaves gegeben, der die Zentrale in den Programmiermode versetzt hat. Zurückgeliefert wird die CV-Adresse (C) und die darin gelesenen Daten (D). Dies nur bei CV-Programmierung von Empfängern, die diesen Mode beherrschen.

Zuordnung Wert in C  $\Leftrightarrow$  CV – Adressen:

C	CV
0 ... 255	768 ... 1023

### Besonderheiten:

Siehe 2.1.2.6.



### 2.1.3 Softwareversion Zentrale

#### Format:

	Frame1	Frame2	Headerbyte	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	1111 1111	1111 1110	0110 0011	0010 0001	OOOO UUUU	IIII IIII	X-Or-Byte
Hex :	0xFF	0xFE	0x63	0x21	O + U	ID	X-Or-Byte
Dez :	255	254	99	33	O + U	ID	X-Or-Byte

#### Beschreibung:

Auf die Anfrage nach der Zentralen-Software-Version erhält man diese Antwort. Die Versionsnummer ist in oberes (OOOO) und unteres (UUUU) Nibble hexadezimal kodiert. Beispiel: Daten 2 = 0011 0000 = 0x30 : Version 3.0.

Zusätzlich wird die Zentralen-Kennung gesendet, die folgende Bedeutung hat:

ID = 0x00: LZ 100 – Zentrale

ID = 0x01: LH 200 – Zentrale

ID = 0x02: DPC – Zentrale (Compact und Commander)

#### Besonderheiten:

Keine.

### 2.1.4 Status Zentrale

#### Format:

	Frame1	Frame2	Headerbyte	Daten 1	Daten 2	X-Or-Byte
Binär :	1111 1111	1111 1110	0110 0010	0010 0010	SSSS SSSS	X-Or-Byte
Hex :	0xFF	0xFE	0x62	0x22	S	X-Or-Byte
Dez :	255	254	98	34	S	X-Or-Byte

#### Beschreibung:

Auf die Anfrage nach dem Zentralenstatus bekommt man das Statusbyte zurück. Dieses Byte ist bitweise wie folgt kodiert:

Bit 0: wenn 1, Anlage in Nothalt

Bit 1: wenn 1, Anlage in Notaus

Bit 2: Zentralen-Startmode (0 = manueller Start, 1 = automatischer Start)

Auto-Start : Alle Loks fahren mit ihren Einstellungen sofort los

Manueller Start : Alle Loks haben Geschwindigkeit 0 und Funktionen aus.

Bit 3: wenn 1, dann Programmiermode aktiv

Bit 4: reserviert

Bit 5: reserviert

Bit 6: wenn 1, dann Kaltstart in der Zentrale

Bit 7: wenn 1, dann RAM-Check-Fehler in der Zentrale

**Besonderheiten:**

Nicht alle Bits sind in allen Zentralen vorhanden. Sind Bit 6 und Bit 2 gesetzt, so legt die Zentrale noch keine Daten auf das Gleis. Erst wenn ein Gerät den Startmode auf manuell oder Automatik stellt, beginnt die Gleisabgabe. Zuvor wird von der Zentrale noch der Broadcast „Alles an“ gesendet. Nicht alle Zentralen unterstützen verschiedene Startmodes. Stellt ein Gerät also „Kaltstart“ und „Startmode Auto“ fest, so sollte es in seinem Kontext erst weitergehen (z.B. zum Steuern von Lokomotiven), wenn es den gewünschten Startmode an die Zentrale gesendet hat oder ein Broadcast „Alles an“ empfangen wurde. Dann hat nämlich ein anderes XpressNet-Gerät den Startmode eingestellt. Es ist also sinnvoll, nach einem Gerätereset zunächst den Zentralenstatus und danach erst Lokdaten zu erfragen etc.

**2.1.5      Übertragungsfehler****Format:**

	Frame1	Frame2	Headerbyte	Daten 1	X-Or-Byte
Binär :	1111 1111	1111 1110	0110 0001	1000 0000	1110 0001
Hex :	0xFF	0xFE	0x61	0x80	0xE1
Dez :	255	254	97	128	225

**Beschreibung:**

Die Zentrale sendet an ein Gerät die Antwort „Übertragungsfehler“, wenn aufgrund eines solchen bei der Zentrale das X-Or-Byte nicht aufgeht, d.h. die X-Or-Verknüpfung aller Bytes einschl. des X-Or-Bytes nicht 0 wird.

**Besonderheiten:**

Ein Übertragungsfehler entsteht meist dann, wenn das X-Or-Byte falsch berechnet wurde oder der Hardware-Handshake nicht beachtet wurde. Ursache kann auch ein Puffer-Überlauf der UART-Hardware des PC sein (dies ist nicht der per Treiber-Software eingerichtete Sende- und Empfangs- FiFo). In aller Regel hat ein Übertragungsfehler noch weitere Fehlermeldungen (Timeout PC-LI-USB) zur Folge. Auch ein Befehl, der normalerweise keine Antwort zur Folge hat und mit 255/254/01/04/05 quittiert würde, kann stattdessen diese Antwort liefern.

### 2.1.6 Zentrale Busy

#### Format:

	Frame1	Frame2	Headerbyte	Daten 1	X-Or-Byte
Binär :	1111 1111	1111 1110	0110 0001	1000 0001	X-Or-Byte
Hex :	0xFF	0xFE	0x61	0x81	X-Or-Byte
Dez :	255	254	97	129	X-Or-Byte

#### Beschreibung:

Die Zentrale sendet an ein Gerät als Antwort auf einen Befehl „Busy“, wenn dieser Befehl zur Zeit nicht ausgeführt werden kann. D.h. im wesentlichen, daß der Befehl zur Zeit nicht auf das Gleis gelegt werden kann.

#### Besonderheiten:

Die Antwort „Busy“ wird dann an den PC gesendet, wenn dieser z.B. versucht, möglichst schnell viele Weichen zu schalten, ohne zu warten, ob der Befehl auch tatsächlich von der Zentrale akzeptiert werden konnte. Um die Antwort „Busy“ dann zuordnen zu können, sollte ein PC-Programm sich an der Ablaufstruktur des XpressNet orientieren, um in dieser Situation den richtigen Befehl wiederholen zu können.

### 2.1.7 Befehl in Zentrale nicht vorhanden

#### Format:

	Frame1	Frame2	Headerbyte	Daten 1	X-Or-Byte
Binär :	1111 1111	1111 1110	0110 0001	1000 0010	X-Or-Byte
Hex :	0xFF	0xFE	0x61	0x82	X-Or-Byte
Dez :	255	254	97	130	X-Or-Byte

#### Beschreibung:

Wurde ein Befehl zwar korrekt übertragen, ist aber nicht im Befehlsvorrat der Zentrale enthalten, so sendet die Zentrale diese Antwort zurück. Ebenso, wenn Befehle aus dem aktuellen Kontext heraus nicht möglich sind (Programmiererergebnis lesen, ohne daß die Zentrale im Programmiermode ist).

#### Besonderheiten:

Keine.

### 2.1.8 Schaltinformation

#### Format:

	Frame1	Frame2	Headerbyte	Daten 1	Daten 2	X-Or-Byte
Binär :	1111 1111	1111 1110	0100 0010	AAAA AAAA	ITTN ZZZZ	X-Or-Byte
Hex :	0xFF	0xFE	0x42	ADR	ITNZ	X-Or-Byte
Dez :	255	254	66	ADR	ITNZ	X-Or-Byte

#### Beschreibung:

Die Zentrale sendet als Antwort auf eine Anfrage nach Schalt-/Rückmeldeinformationen diese Info. Es kann als Information enthalten sein der Zustand von rückmeldefähigen oder nicht rückmeldefähigen Weichen oder der Zustand eines Rückmeldeempfängers. Es bedeuten im Einzelnen:

Daten 1: AAAA AAAA Für eine Weiche ist Daten 1 die durch 4 geteilte Adresse einer Weiche aus dem Wertebereich 0..255. Daten 1 hat für Zentralen kleiner Version 3.0 den Wert 0 bis 63 = 6 Bit. Ist ADR z.B. = 0x00, so hat man eine Info über die Weichen 0, 1, 2 oder 3, d.h. über die Weichengruppe 0 (wenn die Kennungsbits TT einen Schaltempfänger kennzeichnen). Für Zentralen ab Version 3.0 werden alle 8 Bit von Daten 1 als Gruppenadresse erlaubt. D.h. es können  $256 \cdot 4 = 1024$  Weichen abgefragt und geschaltet werden.

Für einen Rückmeldebaustein kann die Adresse im Bereich 0..127 liegen (7 Bit Adresse). Dies ist direkt die Adresse des Bausteins.

Die Adressinformation wird so in dieser Info zurückgeschickt, wie sie in der Anfrage nach Schaltinformationen der Zentrale gesendet wurde.

Daten 2: I Ist das Bit = 1, so bedeutet das, daß der Schaltbefehl noch in der Ausführung ist und die Weiche noch keine Endstellung erreicht hat. Nicht definiert für Rückmeldebausteine, da deren Eingänge ja immer 0 oder 1 sind und keinen Zwischenzustand annehmen können.

Daten 2: TT Diese beiden Bits stellen die Kennung der angefragten Adresse dar. Es gilt:

TT = 0 0 : Adresse ist Schaltempfänger ohne Rückmeldung  
 TT = 0 1 : Adresse ist Schaltempfänger mit Rückmeldung  
 TT = 1 0 : Adresse ist ein Rückmeldebaustein  
 TT = 1 1 : reserviert für zukünftige Anwendungen

Daten 2: N Dies ist die Kennung, um welches Nibble einer Weiche oder eines Rückmeldebausteins es sich handelt. N = 0 entspricht dem unteren Nibble, N = 1 entspricht dem oberen Nibble. Für z.B. Weichengruppe 0 bedeutet das

				untere Nibble den Zustand der Weichen 0 und 1 in den 4 Zustandsbits Z. Das obere Nibble den Zustand der Weichen 2 und 3 in den 4 Zustandsbits Z.
				Für einen Rückmeldeempfänger bedeutet das untere Nibble den Zustand der unteren 4 Eingänge in den 4 Zustandsbits Z, das obere Nibble den Zustand der oberen 4 Eingänge in den 4 Zustandsbits Z. Um also alle 8 Eingänge eines Rückmelde-bausteins zu erfassen, ist eine Anfrage an das untere Nibble und eine zweite Anfrage an das obere Nibble der Rückmeldeadresse zu richten.
				Achtung: Das Nibble-Bit stimmt nur dann, wenn die Weiche schon einmal geschaltet wurde!
Daten 3:	Z3 Z2 Z1 Z0			Für den Zustand eines Schaltempfängers gilt:
	Z1 und Z0			stellen den Zustand der ersten Weiche (z.B Weiche Nr. 0 in Weichengruppe 0, Nibble = 0) im Nibble dar,
	Z3 und Z2			Den Zustand der zweiten Weiche im Nibble (Z.B Weiche Nr. 3 in Weichengruppe 0, Nibble = 1).
				Mögliche Kombinationen:
	Z1	Z0		(erste Weiche im Nibble)
	0	0		Weiche seit Zentralenstart noch nicht geschaltet oder bei rückmeldefähigen Weichen ist kein Eingang für Endstellung angeschlossen.
	0	1		Der letzte Schaltbefehl war „0“, die Weiche steht links (das ist natürlich nur relativ).
	1	0		Der letzte Schaltbefehl war „1“, die Weiche steht in der anderen Endstellung (z.B. rechts, relativ).
	1	1		Ungültige Kombination, wenn beide Endschalter einer rückmeldefähigen Weiche aktiv sind.
				->Verdrahtungsfehler?
				Gleiches gilt für Z3 und Z2 (zweite Weiche im Nibble).
				Bei einem Rückmeldeempfänger stellen die 4 Bits Z3..Z0 den Zustand der 4 Eingänge des angefragten Nibbles dar.

**Besonderheiten:**

Keine.

### 2.1.9 Lokinformationen

Als Antwort auf eine allgemeine Lokanfrage im Format für Version 3.0 oder höher (siehe **Fehler! Verweisquelle konnte nicht gefunden werden.**) kann eine der im folgenden beschriebenen vier Antworten kommen. Im Gegensatz zu früheren Versionen ist die „Besetzt“-Information hier enthalten. D.h. die hier beschriebenen Antworten kommen nicht als ungefragte Info. Wird eine Lok von einem anderen XpressNet-Gerät übernommen, so wird dies jetzt über die ungefragte Info „Lok besetzt“ mitgeteilt (siehe 2.1.2). Es wird weiterhin ein zusätzliches Kennungs-Byte nach dem Headerbyte eingefügt, welches zur Unterscheidung der verschiedenen Befehle ab Zentralen-Version 3.0 dient. Bei den hier beschriebenen Antworten ist die Adresse der angefragten Lok nicht enthalten, um nicht unnötig redundante Daten über das XpressNet zu senden.. Aus der Struktur des XpressNet ergibt sich aber die eindeutige Zuordnung dieser Antworten, weil sie direkt auf eine vorhergegangene Anfrage nach Lokdaten kommt.

#### 2.1.9.1 Lokinformation normale Lok

Diese Antwort wird immer dann gesendet, wenn sich die angefragte Lok nicht in einer Mehrfach-/Doppeltraktion befindet und auch nicht die Basisadresse einer Mehrfachtraktion ist.

##### **Format:**

	Frame1	Frame2	Headerbyte	Kennung	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	1111 1111	1111 1110	1110 0100	0000 BFFF	RVVV VVVV	000F FFFF	FFFF FFFF	X-Or-Byte
Hex :	0xFF	0xFE	0xE4	Kennung	Speed	F0	F1	X-Or-Byte
Dez :	255	254	228	Kennung	Speed	F0	F1	X-Or-Byte

##### **Beschreibung:**

Kennung:    Bit3:    B=0:    Lok ist frei  
                               B=1:    Lok ist an anderem Gerät aufgerufen (besetzt)

:

Kennung:    FFF    Bit2 bis Bit0: Kennung der Fahrstufenzahl:

Bit2	Bit1	Bit0	
0	0	0	14 Fahrstufen
0	0	1	27 Fahrstufen
0	1	0	28 Fahrstufen
1	0	0	128 Fahrstufen

Speed:                      Codierung der Geschwindigkeit und Richtung. R=1: vorwärts, R=0: rückwärts.

Bei 14 Fahrstufen: Codierung der Bits 3,2,1,0 für die Geschwindigkeit:

Bit3	Bit2	Bit1	Bit0	
0	0	0	0	Fahrstufe 0
0	0	0	1	Lokspezifischer Nothalt. Die Lok hält ohne die eingestellte Verzögerung sofort an.
0	0	1	0	Fahrstufe 1
.	.	.	.	.
1	1	1	1	Fahrstufe 14

Bei 27 Fahrstufen: Codierung der Bits 4,3,2,1,0 : Man beachte, daß das Bit4 das **LSB** der Fahrstufe ist.

Bit3	Bit2	Bit1	Bit0	Bit4 (!)	
0	0	0	0	0	Fahrstufe 0
0	0	0	0	1	nicht verwendet !
0	0	0	1	0	Lokspezifischer Nothalt. Die Lok hält ohne die eingestellte Verzögerung sofort an.
0	0	0	1	1	nicht verwendet !
0	0	1	0	0	Fahrstufe 1
0	0	1	0	1	Fahrstufe 2
0	0	1	1	0	Fahrstufe 3
.	.	.	.	.	.
1	1	1	1	0	Fahrstufe 27

Bei 28 Fahrstufen: Codierung der Bits 4,3,2,1,0 für die Geschwindigkeit: Man beachte, daß das Bit4 das **LSB** der Fahrstufe ist.

Bit3	Bit2	Bit1	Bit0	Bit4 (!)	
0	0	0	0	0	Fahrstufe 0
0	0	0	0	1	nicht verwendet !
0	0	0	1	0	Lokspezifischer Nothalt. Die Lok hält ohne die eingestellte Verzögerung sofort an.
0	0	0	1	1	nicht verwendet !
0	0	1	0	0	Fahrstufe 1
0	0	1	0	1	Fahrstufe 2
0	0	1	1	0	Fahrstufe 3
.	.	.	.	.	.
1	1	1	1	0	Fahrstufe 27
1	1	1	1	1	Fahrstufe 28

Bei 128 Fahrstufen:

Bit6	B5	B4	B3	B2	B1	B0	
0	0	0	0	0	0	0	Fahrstufe 0
0	0	0	0	0	0	1	Nothalt
0	0	0	0	0	1	0	Fahrstufe 1
0	0	0	0	0	1	1	Fahrstufe 2
.	.	.	.	.	.	.	.
1	1	1	1	1	1	1	Fahrstufe 126

F0: Zustand der Funktionen 0 bis 4. 0 0 0 F0 F4 F3 F2 F1

F1: Zustand der Funktionen 5 bis 12 F12 F11 F10 F9 F8 F7 F6 F5

Eine 1 bedeutet jeweils Funktion ist an.

### Besonderheiten:

Keine.

#### 2.1.9.2 Funktionszustand F13 bis F28 der angefragten Lok (ab Version 3.6)

	Frame1	Frame2	Headerbyte	Kennung	Daten 1	Daten 2	X-Or-Byte
Binär :	1111 1111	1111 1110	1110 0011	0101 0010	FFFF FFFF	FFFF FFFF	X-Or-Byte
Hex :	0xFF	0xFE	0xE3	0x52			X-Or-Byte
Dez :	255	254	227	82			X-Or-Byte

### Beschreibung:

Daten1: Beinhaltet den Zustand der Funktionen 13 bis 20. Fx=1 heißt Funktion ist eingeschaltet.

F2 = F20 F19 F18 F17 F16 F15 F14 F13

Daten2: Beinhaltet den Zustand der Funktionen 21 bis 28. Fx=1 heißt Funktion ist eingeschaltet.

F3 = F28 F27 F26 F25 F24 F23 F22 F21

### Besonderheiten:

Keine.

#### 2.1.9.3 Lokinformation Lok befindet sich in einer Mehrfachtraktion

### Format:

	Frame1	Frame2	Headerbyte	Kennung	Daten 1	Daten 2	Daten 3	Daten 4	X-Or-Byte
Binär :	1111 1111	1111 1110	1110 0101	0001 BFFF	RVVV VVVV	000F FFFF	FFFF FFFF	MTR	X-Or-Byte
Hex :	0xFF	0xFE	0xE5	Kennung	Speed	F0	F1	MTR	X-Or-Byte
Dez :	255	254	229	Kennung	Speed	F0	F1	MTR	X-Or-Byte

### Beschreibung:



**Kennung:** Die Bits 3 bis 0 sind wie unter 2.1.9.1 beschrieben codiert. Die Fahrstufenzahl gibt die Fahrstufenzahl der angefragten Lok an! Diese kann anders sein als die Fahrstufenzahl der Mehrfachtraktion (MTR), in der sich die Lok befindet.

**Speed:** Das Geschwindigkeits-Byte ist wie unter 2.1.9.1 beschrieben codiert. Die Geschwindigkeit gibt die Geschwindigkeit der angefragten Lok an!

**F0, F1:** Codiert wie unter 2.1.14.1. beschrieben.

**MTR:** Dies ist die MTR-Basisadresse der angefragten Lok.

### Besonderheiten:

Lok-Fahrbefehle sind an die Basisadresse zu senden, da nicht alle Zentralen diese Umsetzung vornehmen. Funktionsbefehle sind an die Lokadresse selbst zu senden.

#### 2.1.9.4 Lokinformation Lokadresse ist die Basisadresse einer Mehrfachtraktion

##### Format:

	Frame1	Frame2	Headerbyte	Kennung	Daten 1	X-Or-Byte
Binär :	1111 1111	1111 1110	1110 0010	0010 BFFF	RVVV VVVV	X-Or-Byte
Hex :	0xFF	0xFE	0xE2	Kennung	Speed	X-Or-Byte
Dez :	255	254	226	Kennung	Speed	X-Or-Byte

##### Beschreibung:

**Kennung:** Die Bits 3 bis 0 sind wie unter 2.1.9.1 beschrieben codiert. Die Fahrstufenzahl gibt die Fahrstufenzahl der Mehrfachtraktion an.

**Speed:** Das Geschwindigkeits-Byte ist wie unter 2.1.9.1 beschrieben codiert. Die Geschwindigkeit gibt die Geschwindigkeit der MTR an.

### Besonderheiten:

An die Basisadresse einer MTR sollen keine Funktionsbefehle gesendet werden.

#### 2.1.9.5 Lokinformation Lok befindet sich in einer Doppeltraktion

##### Format:

	Frame1	Frame2	Headerbyte	Kennung	Daten 1	Daten 2	Daten 3	Daten 4	Daten 5	X-Or-Byte
Binär :	1111 1111	1111 1110	1110 0110	0110 BFFF	RVVV VVVV	000F FFFF	FFFF FFFF	Adr High	Adr Low	X-Or-Byte
Hex :	0xFF	0xFE	0xE6	Kennung	Speed	F0	F1	AH	AL	X-Or-Byte
Dez :	255	254	230	Kennung	Speed	F0	F1	AH	AL	X-Or-Byte

##### Beschreibung:

**Kennung:** Die Bits 3 bis 0 sind wie unter 2.1.9.1 beschrieben codiert.

**Speed:** Das Geschwindigkeits-Byte ist wie unter 2.1.9.1 beschrieben codiert.

F0, F1: Codiert wie unter 2.1.9.1 beschrieben.  
 AH: Highbyte der zweiten Lokadresse der Doppeltraktion.  
 AL: Lowbyte der zweiten Lokadresse der Doppeltraktion  
 Für Lokadressen < 100 gilt:  
     Highbyte der Lokadresse ist 0x00  
     Lowbyte der Lokadresse ist 0x00 bis 0x63  
 Für Lokadresse von 100 bis 9999 gilt:  
     Highbyte der Lokadresse ist:  $AH = (ADR \& 0xFF00) + 0xC000$   
     Lowbyte der Lokadresse ist:  $AL = (ADR \& 0x00FF)$

**Besonderheiten:**

Diese Antwort kommt nur dann, wenn die Lok in der DTR mit dem „neuen“ Lokanfragebefehl angefragt wurde (siehe 2.2.27).

**2.1.10 Lok besetzt ab Zentralen-Version 3.0****Format:**

	Frame1	Frame2	Headerbyte	Kennung	Daten 1	Daten 2	X-Or-Byte
Binär :	1111 1111	1111 1101	1110 0011	0100 0000	Adresse High	Adresse Low	X-Or-Byte
Hex :	0xFF	0xFD	0xE3	0x40	AH	AL	X-Or-Byte
Dez :	255	253	227	64	AH	AL	X-Or-Byte

**Beschreibung:**

AH: Highbyte der Lokadresse.  
 AL: Lowbyte der Lokadresse.  
 Für Lokadressen < 100 gilt:  
     Highbyte der Lokadresse ist 0x00  
     Lowbyte der Lokadresse ist 0x00 bis 0x63  
 Für Lokadresse von 100 bis 9999 gilt:  
     Highbyte der Lokadresse ist:  $AH = (ADR \& 0xFF00) + 0xC000$   
     Lowbyte der Lokadresse ist:  $AL = (ADR \& 0x00FF)$

**Besonderheiten:**

Diese Information kommt immer ungefragt, wenn ein anderes XpressNet-Gerät diese Lok übernommen hat.

**2.1.11 Funktionsstatus F0 bis F12 der angefragten Lok**

Ab Version 3.0 der LZ100-Zentrale speichert diese als zusätzliche Information zu einer Lok, ob deren Funktionen tastend oder nicht tastend sein sollen. Die Gleisausgabe hierfür ändert sich jedoch nicht. XpressNet-Geräte können aber ihre Funktionalität in der Bedieneroberfläche erweitern, so daß z.B. für Geräusche eine zugeordnete Funktion nur solange ausgeführt wird, wie eine Taste gedrückt ist. Der Befehl ist in der Zentrale vorgesehen, damit diese Eigenschaft auch bei der Übernahme einer Lok

durch ein anderes XpressNet-Gerät genutzt werden kann. Die zugehörige Lokadresse wird nicht mitgesendet, weil sich die korrekte Zuordnung durch die direkt vorangegangene Anfrage nach dem Funktionsstatus ergibt.

### Format:

	Frame1	Frame2	Headerbyte	Kennung	Daten 1	Daten 2	X-Or-Byte
Binär :	1111 1111	1111 1110	1110 0011	0101 0000	000S SSSS	SSSS SSSS	X-Or-Byte
Hex :	0xFF	0xFE	0xE3	0x50	S0	S1	X-Or-Byte
Dez :	255	254	227	80	S0	S1	X-Or-Byte

### Beschreibung:

S0: Beinhaltet den Status der Funktionen 0 bis 4. Sx=1 heißt Funktion ist tastend.

S0 = 0 0 0 S0 S4 S3 S2 S1

S1: Beinhaltet den Status der Funktionen 5 bis 12. Sx=1 heißt Funktion ist tastend.

S1 = S12 S11 S10 S9 S8 S7 S6 S5

### Besonderheiten:

Keine.

## 2.1.12 Funktionsstatus F13 bis F28 der angefragten Lok (ab Version 3.6)

	Frame1	Frame2	Headerbyte	Kennung	Daten 1	Daten 2	X-Or-Byte
Binär :	1111 1111	1111 1110	1110 0100	0101 0001	SSSS SSSS	SSSS SSSS	X-Or-Byte
Hex :	0xFF	0xFE	0xE4	0x51			X-Or-Byte
Dez :	255	254	228	81			X-Or-Byte

### Beschreibung:

Daten1: Beinhaltet den Status der Funktionen 13 bis 20. Sx=1 heißt Funktion ist tastend.

Bits:

7	6	5	4	3	2	1	0
F20	F19	F18	F17	F16	F15	F14	F13

Daten2: Beinhaltet den Status der Funktionen 21 bis 28. Sx=1 heißt Funktion ist tastend.

Bits:

7	6	5	4	3	2	1	0
F28	F27	F26	F25	F24	F23	F22	F21

### Besonderheiten:

Keine.

### 2.1.13 Lokinformation bei Adress-Suchanfragen

#### Format:

	Frame1	Frame2	Headerbyte	Kennung	Daten 1	Daten 2	X-Or-Byte
Binär :	1111 1111	1111 1110	1110 0011	0011 KKKK	Adresse High	Adresse Low	X-Or-Byte
Hex :	0xFF	0xFE	0xE3	0x30 + K	AH	AL	X-Or-Byte
Dez :	255	254	227	48 + K	AH	AL	X-Or-Byte

#### Beschreibung:

Diese Antwort wird gesendet, wenn das XpressNet-Gerät eine der Suchanfragen aus 2.2.30 benutzt hat. Damit kann z.B. in dem Gerät eine Auswahlliste der gewünschten Loks gezeigt werden (nächste Lok einer Mehrfachtraktion etc.).

Die Kennung enthält den Typ der Lokadresse, die in Adresse High / Adresse Low steht.

Kennung:    KKKK = 0:    Normale Lok in Daten 1/2  
               KKKK = 1:    Doppeltraktionslok in Daten 1/2  
               KKKK = 2:    Mehrfachtraktions-Basisadresse in Daten 1/2  
               KKKK = 3:    Mitglied einer Mehrfachtraktion in Daten 1/2  
               KKKK = 4:    Keine Adresse zur Suchanfrage mehr gefunden. Daten 1/2

= 0x00

AH/AL:        Die Lokadresse wird berechnet, wie unter 2.1.10 beschrieben.

#### Besonderheiten:

Keine.

### 2.1.14 Fehlermeldungen

Der Zusammenhang ergibt sich aus dem vorausgehend erteilten Befehl an die Zentrale, auf den sich die Fehlermeldung bezieht.

#### Format:

	Frame1	Frame2	Headerbyte	Kennung	X-Or-Byte
Binär :	1111 1111	1111 1110	1110 0001	1000 FFFF	X-Or-Byte
Hex :	0xFF	0xFE	0xE1	0x80 + F	X-Or-Byte
Dez :	255	254	225	128 + F	X-Or-Byte

#### Beschreibung:

Die 4 Fehlerbits sind wie folgt codiert:

FFFF = 0001: Eine Lok ist bei Montage einer Mehrfachtraktion nicht durch das montierende Gerät aufgerufen oder die Lok 0 ist gewählt.

FFFF = 0010: Eine der Loks der Mehrfachtraktion ist bei einem anderen Gerät aufgerufen.

FFFF = 0011: Eine der Loks ist schon in einer anderen Mehrfachtraktion oder Doppeltraktion eingebunden.

FFFF = 0100: Die Geschwindigkeit einer der Loks der Mehrfachtraktion ist nicht Null.

FFFF = 0101: Die Lok ist nicht in einer Mehrfachtraktion.

FFFF = 0110 Die Lokadresse ist keine Mehrfachtraktions-Basisadresse.

FFFF = 0111: Lok löschen ist nicht möglich

FFFF = 1000: Der Zentralenstack ist voll

### Besonderheiten:

Keine.

## 2.2 PC an Zentrale

Je nach gewünschter Aktion gibt die Zentrale bzw. das LI-USB dem PC die entsprechende Antwort.

### 2.2.1 Alles An

#### Format:

	Frame1	Frame2	Headerbyte	Daten 1	X-Or-Byte
Binär :	1111 1111	1111 1110	0010 0001	1000 0001	1010 0000
Hex :	0xFF	0xFE	0x21	0x81	0xA0
Dez :	255	254	33	129	160

#### Beschreibung:

Der Befehl veranlasst die Zentrale dazu, die Spannung am Gleis wieder einzuschalten, wenn sie abgeschaltet war und mit der Aussendung von Gleisbefehlen wieder zu beginnen. Damit wird ein Nothalt, ein Notaus oder der Programmierbetrieb auf dem Programmiergleis beendet. Die Zentrale sendet nach erfolgreichem Einschalten die Broadcast „Alles An“. Siehe 2.1.4.1.

### Besonderheiten:

Keine.

### 2.2.2 Alles Aus (Notaus)

#### Format:

	Frame1	Frame2	Headerbyte	Daten 1	X-Or-Byte
Binär :	1111 1111	1111 1110	0010 0001	1000 0000	1010 0001
Hex :	0xFF	0xFE	0x21	0x80	0xA1
Dez :	255	254	33	128	161

#### Beschreibung:

Der Befehl veranlasst die Zentrale dazu, die Spannung am Gleis abzuschalten. Danach sendet die Zentrale mehrmals die Broadcast „Alles Aus“ an alle Busteilnehmer. Auch an denjenigen, der dieses Kommando gegeben hat.

### Besonderheiten:

Keine.

### 2.2.3 Alle Loks anhalten (Nothalt)

#### Format:

	Frame1	Frame2	Headerbyte	X-Or-Byte
Binär :	1111 1111	1111 1110	1000 0000	1000 0000
Hex :	0xFF	0xFE	0x80	0x80
Dez :	255	254	128	128

#### Beschreibung:

Der Befehl veranlasst die Zentrale dazu, alle Loks auf dem Gleis ohne deren eingestellte Verzögerung sofort anzuhalten. Die Spannung am Gleis bleibt jedoch eingeschaltet, so daß z.B. Weichen weiterhin geschaltet werden können.

#### Besonderheiten:

Keine.

### 2.2.4 Eine Lok anhalten (Nothalt für eine Lok)

#### Format:

	Frame1	Frame2	Headerbyte	Daten 1	Daten 2	X-Or-Byte
Binär :	1111 1111	1111 1110	1001 0010	Adresse High	Adresse Low	X-Or-Byte
Hex :	0xFF	0xFE	0x92	AH	AL	X-Or-Byte
Dez :	255	254	146	AH	AL	X-Or-Byte

#### Beschreibung:

Der Befehl veranlasst die Zentrale dazu, nur die gewünschte Lok auf dem Gleis ohne deren eingestellte Verzögerung sofort anzuhalten. Die Spannung am Gleis bleibt eingeschaltet, so daß z.B. Weichen weiterhin geschaltet werden können und auch alle anderen Loks normal weiterfahren.

#### Besonderheiten:

Es können die Loks 0 bis 9999 angehalten werden.

Die Lokadresse AH/AL berechnet sich wie unter 2.1.10 angegeben.

### 2.2.5 Leseanfrage Programmieren 3-Byte-Format (Registermode)

#### Format:

	Frame1	Frame2	Headerbyte	Daten 1	Daten 2	X-Or-Byte
Binär :	1111 1111	1111 1110	0010 0010	0001 0001	0000 RRRR	X-Or-Byte
Hex :	0xFF	0xFE	0x22	0x11	R	X-Or-Byte
Dez :	255	254	34	17	R	X-Or-Byte

#### Beschreibung:

Der Befehl veranlasst die Zentrale dazu, in den Programmiermode zu schalten und den Empfänger, der auf dem Programmiergleis steht, im Registermode zu lesen. Es wird versucht, das Register, welches mit 0000 RRRR angegeben ist, zu lesen. Zulässig ist Register 1..8.

#### Besonderheiten:

Die Leseanforderung hat keine Antwort der Zentrale zur Folge! Diese muß explizit mit dem Befehl „Programmiererergebnis anfordern“ geholt werden. Erst dann kann erkannt werden, ob der Lesebefehl erfolgreich war oder nicht und ob das Ergebnis in der gewünschten Form (Registermode) vorliegt.

Nach dem Erteilen eines Lesebefehls sendet die Zentrale an alle Busteilnehmer die Broadcast „Programmiermode“ und es kann ausschließlich das Gerät weitere Befehle an die Zentrale senden, das den Programmiermode ausgelöst hat.

### 2.2.6 Leseanfrage Programmieren 4-Byte-Format (CV-Mode)

#### Format:

	Frame1	Frame2	Headerbyte	Daten 1	Daten 2	X-Or-Byte
Binär :	1111 1111	1111 1110	0010 0010	0001 0101	CCCC CCCC	X-Or-Byte
Hex :	0xFF	0xFE	0x22	0x15	CV	X-Or-Byte
Dez :	255	254	34	21	CV	X-Or-Byte

#### Beschreibung:

Der Befehl veranlasst die Zentrale dazu, in den Programmiermode zu schalten und den Empfänger, der auf dem Programmiergleis steht, im CV-Mode zu lesen. Es wird versucht, die CV, welche mit CCCC CCCC angegeben ist, zu lesen.

Der Bereich ist von 1 bis 256, wobei CV256 als 00 zu senden ist.

#### Besonderheiten:

Dieser Befehl existiert zusätzlich zum Befehl wie in 2.2.7 beschrieben. Wird eine Zentrale ab Version 3.6 verwendet, so liefert der Befehl den Wert der CV1024 statt der CV256 zurück. Daher empfehlen wir die Verwendung des in 2.2.7 beschriebenen Befehls.

Die Leseanforderung hat keine Antwort der Zentrale zur Folge! Diese muß explizit mit dem Befehl „Programmiererergebnis anfordern“ geholt werden. Erst dann kann erkannt werden, ob der Lesebefehl erfolgreich war oder nicht und ob das Ergebnis in der gewünschten Form (CV-Mode) vorliegt. Konnte der Empfänger nicht im CV-Mode gelesen werden, so versucht es die Zentrale im Registermode. Gelingt diese Leseaktionen, so liegt ein Ergebnis zum Abholen in der Zentrale bereit und das XpressNet-Gerät muß dieses Ergebnis daraufhin prüfen, ob es ein CV-, oder Registerergebnis ist.

Nach dem Erteilen eines Lesebefehls sendet die Zentrale an alle Busteilnehmer die Broadcast „Programmiermode“ und es kann ausschließlich das Gerät weiter Befehle an die Zentrale senden, das den Programmiermode ausgelöst hat.

### 2.2.7 Leseanfrage Programmieren 4-Byte-Format (CV 1-255 und CV1024) (neu ab V3.6)

#### Format:

	Frame1	Frame2	Headerbyte	Daten 1	Daten 2	X-Or-Byte
Binär :	1111 1111	1111 1110	0010 0010	0001 1000	CCCC CCCC	X-Or-Byte
Hex :	0xFF	0xFE	0x22	0x18	CV	X-Or-Byte
Dez :	255	254	34	24	CV	X-Or-Byte

#### Beschreibung:

Der Befehl veranlasst die Zentrale dazu, in den Programmiermode zu schalten und den Empfänger, der auf dem Programmiergleis steht, im CV-Mode zu lesen. Es wird versucht, die CV, welche mit CCCC CCCC angegeben ist, zu lesen.

Zuordnung Wert in "Daten 2" <=> CV – Adressen:

Daten 2	CV
0	1024
1 ... 255	1 ... 255

#### Besonderheiten:

Dieser Befehl sollte an einer Zentrale ab Version 3.6 immer verwendet werden, weitere Besonderheiten gelten wie unter 2.2.6 beschrieben.

### 2.2.8 Leseanfrage Programmieren 4-Byte-Format (CV 256-511) (neu ab V3.6)

#### Format:

	Frame1	Frame2	Headerbyte	Daten 1	Daten 2	X-Or-Byte
Binär :	1111 1111	1111 1110	0010 0010	0001 1001	CCCC CCCC	X-Or-Byte
Hex :	0xFF	0xFE	0x22	0x19	CV	X-Or-Byte
Dez :	255	254	34	25	CV	X-Or-Byte



**Beschreibung:**

Der Befehl veranlasst die Zentrale dazu, in den Programmiermode zu schalten und den Empfänger, der auf dem Programmiergleis steht, im CV-Mode zu lesen. Es wird versucht, die CV, welche mit CCCC CCCC angegeben ist, zu lesen.

Zuordnung Wert in "Daten 2" <=> CV – Adressen:

Daten 2	CV
0 ... 255	256 ... 511

**Besonderheiten:**

Wie unter 2.2.6 beschrieben.

### 2.2.9 Leseanfrage Programmieren 4-Byte-Format (CV 512-767) (neu ab V3.6)

**Format:**

	Frame1	Frame2	Headerbyte	Daten 1	Daten 2	X-Or-Byte
Binär :	1111 1111	1111 1110	0010 0010	0001 1010	CCCC CCCC	X-Or-Byte
Hex :	0xFF	0xFE	0x22	0x1A	CV	X-Or-Byte
Dez :	255	254	34	26	CV	X-Or-Byte

**Beschreibung:**

Der Befehl veranlasst die Zentrale dazu, in den Programmiermode zu schalten und den Empfänger, der auf dem Programmiergleis steht, im CV-Mode zu lesen. Es wird versucht, die CV, welche mit CCCC CCCC angegeben ist, zu lesen.

Zuordnung Wert in "Daten 2" <=> CV – Adressen:

Daten 2	CV
0 ... 255	512 ... 767

**Besonderheiten:**

Wie unter 2.2.6 beschrieben.

### 2.2.10 Leseanfrage Programmieren 4-Byte-Format (CV 768-1023) (neu ab V3.6)

**Format:**

	Frame1	Frame2	Headerbyte	Daten 1	Daten 2	X-Or-Byte
Binär :	1111 1111	1111 1110	0010 0010	0001 1011	CCCC CCCC	X-Or-Byte

Hex :	0xFF	0xFE	0x22	0x1B	CV	X-Or-Byte
Dez :	255	254	34	27	CV	X-Or-Byte

**Beschreibung:**

Der Befehl veranlasst die Zentrale dazu, in den Programmiermode zu schalten und den Empfänger, der auf dem Programmiergleis steht, im CV-Mode zu lesen. Es wird versucht, die CV, welche mit CCCC CCCC angegeben ist, zu lesen.

Zuordnung Wert in "Daten 2" <=> CV – Adressen:

Daten 2	CV
0 ... 255	768 ... 1023

**Besonderheiten:**

Wie unter 2.2.6 beschrieben.

### 2.2.11 Leseanfrage Programmieren 3-Byte-Format (Pagemode)

**Format:**

	Frame1	Frame2	Headerbyte	Daten 1	Daten 2	X-Or-Byte
Binär :	1111 1111	1111 1110	0010 0010	0001 0100	CCCC CCCC	X-Or-Byte
Hex :	0xFF	0xFE	0x22	0x14	CV	X-Or-Byte
Dez :	255	254	34	20	CV	X-Or-Byte

**Beschreibung:**

Der Befehl veranlasst die Zentrale dazu, in den Programmiermode zu schalten und den Empfänger, der auf dem Programmiergleis steht, im Pagemode zu lesen. Es wird versucht, die CV, welche mit CCCC CCCC angegeben ist, zu lesen. Die Zentrale setzt die Pageangaben auf Register um (gleisseitig) und versucht, den Empfänger im Registermode auszulesen.

Der Bereich ist von 1 bis 256, wobei CV256 als 00 zu senden ist.

**Besonderheiten:**

Die Leseanforderung hat keine Antwort der Zentrale zur Folge! Diese muß explizit mit dem Befehl „Programmierergebnis anfordern“ geholt werden. Erst dann kann erkannt werden, ob der Lesebefehl erfolgreich war oder nicht und ob das Ergebnis in der gewünschten Form (Pagemode) vorliegt. Gelingt eine Leseaktionen, so liegt ein Ergebnis zum Abholen in der Zentrale bereit und das XpressNet-Gerät muß dieses Ergebnis auf seinen Inhalt hin untersuchen.

Nach dem Erteilen eines Lesebefehls sendet die Zentrale an alle Busteilnehmer die Broadcast „Programmiermode“ und es kann ausschließlich das Gerät weiter Befehle an die Zentrale senden, das den Programmiermode ausgelöst hat.

### 2.2.12 Programmierergebnis anfordern

#### Format:

	Frame1	Frame2	Headerbyte	Daten 1	X-Or-Byte
Binär :	1111 1111	1111 1110	0010 0001	0001 0000	0011 0001
Hex :	0xFF	0xFE	0x21	0x10	0x31
Dez :	255	254	33	16	49

#### Beschreibung:

Der Befehl veranlasst die Zentrale dazu, das Ergebnis einer vorangegangenen Leseaktion an das XpressNet-Gerät zu senden. Die Antwort ist eine der unter 2.1.2 beschriebenen Möglichkeiten.

#### Besonderheiten:

Keine.

### 2.2.13 Schreibbefehl Programmieren 3-Byte-Format (Register-Mode)

#### Format:

	Frame1	Frame2	Headerbyte	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	1111 1111	1111 1110	0010 0011	0001 0010	0000 RRRR	Daten	X-Or-Byte
Hex :	0xFF	0xFE	0x23	0x12	R	Daten	X-Or-Byte
Dez :	255	254	35	18	R	Daten	X-Or-Byte

#### Beschreibung:

Der Befehl veranlasst die Zentrale dazu, in den Programmiermode zu schalten und den Empfänger, der auf dem Programmiergleis steht, im Register-Mode zu schreiben. Es wird versucht, den Wert, der in Daten 3 steht, in die Register-Adresse in Daten 2 zu schreiben.

Der Bereich ist Register 1 bis 8.

#### Besonderheiten:

Bevor ein Schreibbefehl benutzt wird, sollte die Zentrale durch einen Lesebefehl in den Programmiermode versetzt werden. Es gibt keine Kontrolle seitens des XpressNet-Gerätes darüber, ob der Empfänger die Programmiersequenz auch verstanden hat, außer durch nochmaliges Auslesen.

### 2.2.14 Schreibbefehl Programmieren 4-Byte-Format (CV-Mode, CV1-256)

#### Format:

	Frame1	Frame2	Headerbyte	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	1111 1111	1111 1110	0010 0011	0001 0110	CCCC CCCC	Daten	X-Or-Byte
Hex :	0xFF	0xFE	0x23	0x16	CV	Daten	X-Or-Byte
Dez :	255	254	35	22	CV	Daten	X-Or-Byte

**Beschreibung:**

Der Befehl veranlasst die Zentrale dazu, in den Programmiermode zu schalten und den Empfänger, der auf dem Programmiergleis steht, im CV-Mode zu schreiben. Es wird versucht, den Wert, der in Daten 3 steht, in die CV-Adresse in Daten 2 zu schreiben.

Der Bereich ist CV 1 bis 255.

**Besonderheiten:**

Dieser Befehl existiert zusätzlich zum Befehl wie in 2.2.15 beschrieben. Wird eine Zentrale ab Version 3.6 verwendet, so wird die CV1024 statt der CV256 beschrieben. Daher empfehlen wir die Verwendung des in 2.2.15 beschriebenen Befehls.

Bevor ein Schreibbefehl benutzt wird, sollte die Zentrale durch einen Lesebefehl in den Programmiermode versetzt und geprüft werden, ob der Empfänger sich im CV-Mode programmieren läßt. Es gibt keine Kontrolle seitens des XpressNet-Gerätes darüber, ob der Empfänger die Programmiersequenz auch verstanden hat, außer durch nochmaliges Auslesen.

### 2.2.15 Schreibbefehl Programmieren 4-Byte-Format (CV-Mode, CV 1-255 und CV1024) (neu ab Version 3.6)

**Format:**

	Frame1	Frame2	Headerbyte	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	1111 1111	1111 1110	0010 0011	0001 1100	CCCC CCCC	Daten	X-Or-Byte
Hex :	0xFF	0xFE	0x23	0x1C	CV	Daten	X-Or-Byte
Dez :	255	254	35	28	CV	Daten	X-Or-Byte

**Beschreibung:**

Der Befehl veranlasst die Zentrale dazu, in den Programmiermode zu schalten und den Empfänger, der auf dem Programmiergleis steht, im CV-Mode zu schreiben. Es wird versucht, den Wert, der in Daten 3 steht, in die CV-Adresse in Daten 2 zu schreiben.

Zuordnung Wert in "Daten 2" <=> CV – Adressen:

Daten 2	CV
0	1024
1 ... 255	1 ... 255

**Besonderheiten:**

Bevor ein Schreibbefehl benutzt wird, sollte die Zentrale durch einen Lesebefehl in den Programmiermode versetzt und geprüft werden, ob der Empfänger sich im CV-Mode programmieren läßt. Es gibt keine Kontrolle seitens des XpressNet-Gerätes

darüber, ob der Empfänger die Programmiersequenz auch verstanden hat, außer durch nochmaliges Auslesen.

#### 2.2.16 Schreibbefehl Programmieren 4-Byte-Format (CV-Mode, CV 256-511) (neu ab Version 3.6)

##### Format:

	Frame1	Frame2	Headerbyte	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	1111 1111	1111 1110	0010 0011	0001 1101	CCCC CCCC	Daten	X-Or-Byte
Hex :	0xFF	0xFE	0x23	0x1D	CV	Daten	X-Or-Byte
Dez :	255	254	35	29	CV	Daten	X-Or-Byte

##### Beschreibung:

Der Befehl veranlasst die Zentrale dazu, in den Programmiermode zu schalten und den Empfänger, der auf dem Programmiergleis steht, im CV-Mode zu schreiben. Es wird versucht, den Wert, der in Daten 3 steht, in die CV-Adresse in Daten 2 zu schreiben.

Zuordnung Wert in "Daten 2" <=> CV – Adressen:

Daten 2	CV
0 ... 255	256 ... 511

##### Besonderheiten:

wie 2.2.15.

#### 2.2.17 Schreibbefehl Programmieren 4-Byte-Format (CV-Mode, CV 512-767) (neu ab Version 3.6)

##### Format:

	Frame1	Frame2	Headerbyte	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	1111 1111	1111 1110	0010 0011	0001 1110	CCCC CCCC	Daten	X-Or-Byte
Hex :	0xFF	0xFE	0x23	0x1E	CV	Daten	X-Or-Byte
Dez :	255	254	35	30	CV	Daten	X-Or-Byte

##### Beschreibung:

Der Befehl veranlasst die Zentrale dazu, in den Programmiermode zu schalten und den Empfänger, der auf dem Programmiergleis steht, im CV-Mode zu schreiben. Es wird versucht, den Wert, der in Daten 3 steht, in die CV-Adresse in Daten 2 zu schreiben.

Zuordnung Wert in "Daten 2" <=> CV – Adressen:

Daten 2	CV
0 ... 255	512 ... 767

**Besonderheiten:**

wie 2.2.15.

### 2.2.18 Schreibbefehl Programmieren 4-Byte-Format (CV-Mode, CV 768-1023) (neu ab Version 3.6)

**Format:**

	Frame1	Frame2	Headerbyte	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	1111 1111	1111 1110	0010 0011	0001 1111	CCCC CCCC	Daten	X-Or-Byte
Hex :	0xFF	0xFE	0x23	0x1F	CV	Daten	X-Or-Byte
Dez :	255	254	35	31	CV	Daten	X-Or-Byte

**Beschreibung:**

Der Befehl veranlasst die Zentrale dazu, in den Programmiermode zu schalten und den Empfänger, der auf dem Programmiergleis steht, im CV-Mode zu schreiben. Es wird versucht, den Wert, der in Daten 3 steht, in die CV-Adresse in Daten 2 zu schreiben.

Zuordnung Wert in "Daten 2" <=> CV – Adressen:

Daten 2	CV
0 ... 255	768 ... 1023

**Besonderheiten:**

wie 2.2.15.

### 2.2.19 Schreibbefehl Programmieren 3-Byte-Format (Page-Mode)

**Format:**

	Frame1	Frame2	Headerbyte	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	1111 1111	1111 1110	0010 0011	0001 0111	CCCC CCCC	Daten	X-Or-Byte
Hex :	0xFF	0xFE	0x23	0x17	CV	Daten	X-Or-Byte
Dez :	255	254	35	23	CV	Daten	X-Or-Byte

**Beschreibung:**

Der Befehl veranlasst die Zentrale dazu, in den Programmiermode zu schalten und den Empfänger, der auf dem Programmiergleis steht, im Page-Mode zu schreiben. Es wird versucht, den Wert, der in Daten 3 steht, in die CV-Adresse in Daten 2 zu schreiben, wobei die Zentrale eine entsprechende Umrechnung der CV auf die zu verwendende Page macht und den Empfänger im Registermode programmiert.

Der Bereich ist CV 1 bis 256, wobei CV256 als 0x00 gesendet werden muß.

#### **Besonderheiten:**

Bevor ein Schreibbefehl benutzt wird, sollte die Zentrale durch einen Lesebefehl in den Programmiermode versetzt und geprüft werden, ob der Empfänger sich im Page-Mode programmieren läßt. Es gibt keine Kontrolle seitens des XpressNet-Gerätes darüber, ob der Empfänger die Programmiersequenz auch verstanden hat, außer durch nochmaliges Auslesen.

### **2.2.20     Softwareversion der Zentrale anfordern**

#### **Format:**

	Frame1	Frame2	Headerbyte	Daten 1	X-Or-Byte
Binär :	1111 1111	1111 1110	0010 0001	0010 0001	0000 0000
Hex :	0xFF	0xFE	0x21	0x21	0x00
Dez :	255	254	33	33	0

#### **Beschreibung:**

Mit diesem Befehl wird die Zentrale veranlasst, ihre Softwareversion dem XpressNet-Gerät mitzuteilen. Je nach Zentralenversion sind die Antworten wie unter 2.1.3 beschrieben möglich.

#### **Besonderheiten:**

Keine.

### **2.2.21     Status der Zentrale anfordern**

#### **Format:**

	Frame1	Frame2	Headerbyte	Daten 1	X-Or-Byte
Binär :	1111 1111	1111 1110	0010 0001	0010 0100	0000 0101
Hex :	0xFF	0xFE	0x21	0x24	0x05
Dez :	255	254	33	36	5

#### **Beschreibung:**

Die Anfrage nach dem Zentralenstatus ergibt die unter 2.1.4 beschriebene Antwort.

#### **Besonderheiten:**

Keine.

### 2.2.22 Zentralen-Startmode setzen

#### Format:

	Frame1	Frame2	Headerbyte	Daten 1	Daten 2	X-Or-Byte
Binär :	1111 1111	1111 1110	0010 0010	0010 0010	0000 0M00	X-Or-Byte
Hex :	0xFF	0xFE	0x22	0x22	M	X-Or-Byte
Dez :	255	254	34	34	M	X-Or-Byte

#### Beschreibung:

Setzt den Startmode der Zentrale nach Reset. M=0: Manueller Start aller Loks, M=1: automatischer Start aller Loks mit den letzten Geschwindigkeits- und Funktionseinstellungen.

#### Besonderheiten:

Nicht alle Zentralen unterstützen diesen Befehl.

### 2.2.23 Schaltinformationen anfordern

#### Format:

	Frame1	Frame2	Headerbyte	Daten 1	Daten 2	X-Or-Byte
Binär :	1111 1111	1111 1110	0100 0010	AAAA AAAA	1000 000N	X-Or-Byte
Hex :	0xFF	0xFE	0x42	Adresse	0x80 + N	X-Or-Byte
Dez :	255	254	66	Adresse	128 + N	X-Or-Byte

#### Beschreibung:

Aufgrund dieses Kommandos sendet die Zentrale die unter 2.1.8 beschriebene Antwort.

**Adresse:** Für einen Schaltempfänger ist dies die durch 4 geteilte Adresse des gewünschten Schaltausganges (=Weichengruppe). Damit ergibt sich für die Adresse ein Bereich von 0 bis 63 = 6 Bit für alle Versionen kleiner 3.0.

Ab der Version 3.0 werden für die Weichengruppe alle 8 Bit erlaubt. Damit ergibt sich ein Bereich von 256 (0..255) Weichengruppen. Es können also 1024 Weichen geschaltet werden, wobei die Weichen Nr. 1..512 rückmeldefähig sind, die Weichen Nr. 513 bis 1024 jedoch nicht. Für einen Rückmeldebaustein ist die Adresse im Bereich 0 bis 127 (=7Bit) und gibt direkt den gewünschten Baustein an.

**N:** Kennzeichnung für das gewünschte Nibble.

N=0 ist das untere Nibble, N=1 das obere.

Für Schaltempfänger ist es so, daß in einer Weichengruppe 4 Weichen enthalten sind und das untere Nibble die Weichen 0 und 1 der Weichengruppe bezeichnet und das obere Nibble die Weichen Nr. 2 und 3 der Weichengruppe.



Für einen Rückmeldebaustein bezeichnet das untere Nibble den Zustand der ersten 4 Eingänge des Rückmeldebausteins und das obere Nibble den Zustand der oberen 4 Eingänge.

### Besonderheiten:

- Beispiel 1: Weichenbereich 0..255, der Zustand der Weiche Nr. 21 ist gewünscht.  
 Adresse:  $21 \bmod 4 = 5$ , dh. Weiche 21 liegt in Weichengruppe 5  
 In Weichengruppe 5 liegen die Weichen 20, 21, 22, 23. Damit ist das Nibblebit 0 (unteres Nibble).
- Beispiel 2: Weichenbereich 0..1023, der Zustand der Weiche Nr. 620 ist gewünscht.  
 Adresse:  $623 \bmod 4 = 155$ , dh. Weiche 623 liegt in Weichengruppe 155.  
 In Weichengruppe 155 liegen die Weichen 620, 621, 622, 623. Damit ist das Nibblebit 1 (oberes Nibble).

### 2.2.24 Schaltebefehl

#### Format:

	Frame1	Frame2	Headerbyte	Daten 1	Daten 2	X-Or-Byte
Binär :	1111 1111	1111 1110	0101 0010	AAAA AAAA	1000 DBBD	X-Or-Byte
Hex :	0xFF	0xFE	0x52	Adresse	0x80 + DBBD	X-Or-Byte
Dez :	255	254	82	Adresse	128 + DBBD	X-Or-Byte

#### Beschreibung:

Schaltebefehle können nur an Schaltempfänger erteilt werden. Die Adresse ist damit die Weichennummer / 4 (=Weichengruppe). Es bleibt noch die Definition des Offsets in der Weichengruppe, um die gewünschte Weiche exakt zu definieren, sowie die Auswahl, welcher der beiden Ausgänge dieser Weiche gewünscht ist und ob dieser Ausgang zu aktivieren oder zu deaktivieren ist. Dies geschieht über die 4 Bits D1 B1 B0 D2 in Daten 2.

- B1 B0: Dies sind die beiden LSB's der Weichenadresse, die bei der Division durch 4 weggefallen sind.
- D1: D1 = 0 bedeutet Ausgang deaktivieren.  
 D1 = 1 bedeutet Ausgang aktivieren.
- D2: D2 = 0 bedeutet Ausgang 1 der Weiche gewählt.  
 D2 = 1 bedeutet Ausgang 2 der Weiche gewählt.

#### Besonderheiten:

Für Zentralen kleiner Version 3.0 ist ein Bereich von 0..63 für die Weichengruppe definiert. Damit ist die Gruppenadresse 6 Bit lang. Ab Version 3.0 können auch die Gruppen bis 255 benutzt werden. Siehe auch 2.2.23.

### 2.2.25 Lokinformationen anfordern

#### Format:

	Frame1	Frame2	Headerbyte	Kennung	Daten 1	Daten 2	X-Or-Byte
Binär :	1111 1111	1111 1110	1110 0011	0000 0000	Adresse High	Adresse Low	X-Or-Byte
Hex :	0xFF	0xFE	0xE3	0x00	AH	AL	X-Or-Byte
Dez :	255	254	227	0	AH	AL	X-Or-Byte

#### Beschreibung:

Es können die Loks 0 bis 9999 angefragt werden.

Die Lokadresse AH/AL berechnet sich wie unter 2.1.10 angegeben.

Die möglichen Antworten sind unter 2.1.9 beschrieben.

#### Besonderheiten:

Keine.

### 2.2.25.1 Funktionsstatus anfordern

#### Format:

	Frame1	Frame2	Headerbyte	Kennung	Daten 1	Daten 2	X-Or-Byte
Binär :	1111 1111	1111 1110	1110 0011	0000 0111	Adresse High	Adresse Low	X-Or-Byte
Hex :	0xFF	0xFE	0xE3	0x07	AH	AL	X-Or-Byte
Dez :	255	254	227	7	AH	AL	X-Or-Byte

#### Beschreibung:

Holt den Funktionszustand F0 bis F12 als tastend oder nicht tastend.

Es können die Loks 0 bis 9999 angefragt werden.

Die Lokadresse AH/AL berechnet sich wie unter 2.1.10 angegeben.

Die möglichen Antworten sind unter 2.1.9 beschrieben.

#### Besonderheiten:

Keine.

### 2.2.25.2 Funktionsstatus anfordern F13 – F28 (neu ab Zentralen-Version 3.6)

#### Format:

	Frame1	Frame2	Headerbyte	Kennung	Daten 1	Daten 2	X-Or-Byte
Binär :	1111 1111	1111 1110	1110 0011	0000 1000	Adresse High	Adresse Low	X-Or-Byte
Hex :	0xFF	0xFE	0xE3	0x08	AH	AL	X-Or-Byte
Dez :	255	254	227	8	AH	AL	X-Or-Byte

#### Beschreibung:

Holt den Funktionsstatus F13 bis F28 als tastend oder nicht tastend.  
 Es können die Loks 0 bis 9999 angefragt werden.  
 Die Lokadresse AH/AL berechnet sich wie unter 2.1.10 angegeben.  
 Die möglichen Antworten sind unter 2.1.12 beschrieben.

**Besonderheiten:**

Keine.

**2.2.25.3**     Funktionszustand anfordern F13 – F28 (neu ab Zentralen-Version 3.6)

**Format:**

	Frame1	Frame2	Headerbyte	Kennung	Daten 1	Daten 2	X-Or-Byte
Binär :	1111 1111	1111 1110	1110 0011	0000 1001	Adresse High	Adresse Low	X-Or-Byte
Hex :	0xFF	0xFE	0xE3	0x09	AH	AL	X-Or-Byte
Dez :	255	254	227	9	AH	AL	X-Or-Byte

**Beschreibung:**

Holt den Funktionszustand der Funktionen F13 bis F28.  
 Es können die Loks 0 bis 9999 angefragt werden.  
 Die Lokadresse AH/AL berechnet sich wie unter 2.1.10 angegeben.  
 Die möglichen Antworten sind unter 2.1.9.2 beschrieben.

**Besonderheiten:**

Keine.

**2.2.26**     Lok steuern

**2.2.26.1**     Fahrbefehle

Der Fahrbefehl für eine Lok gliedert sich in 4 verschiedenen Möglichkeiten auf, die der Fahrstufenzahl 14, 27, 28 und 128 zugeordnet sind. Dies wird durch die unterschiedliche Kennung erreicht. Die Geschwindigkeit selbst ist für 14, 27 und 28 Fahrstufen wie unter 2.1.9.1 "Lokinformation normale Lok" beschrieben codiert. Die Geschwindigkeit für 128 Fahrstufen wie unter 2.1.9.1.

Fahrbefehl 14 Fahrstufen:

**Format:**

	Frame1	Frame2	Headerbyte	Kennung	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	1111 1111	1111 1110	1110 0100	0001 0000	Adresse High	Adresse Low	R000 VVVV	X-Or-Byte
Hex :	0xFF	0xFE	0xE4	0x10	AH	AL	RV	X-Or-Byte
Dez :	255	254	228	16	AH	AL	RV	X-Or-Byte

**Fahrbehl 27 Fahrstufen:****Format:**

	Frame1	Frame2	Headerbyte	Kennung	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	1111 1111	1111 1110	1110 0100	0001 0001	Adresse High	Adresse Low	R00V VVVV	X-Or-Byte
Hex :	0xFF	0xFE	0xE4	0x11	AH	AL	RV	X-Or-Byte
Dez :	255	254	228	17	AH	AL	RV	X-Or-Byte

**Fahrbehl 28 Fahrstufen:****Format:**

	Frame1	Frame2	Headerbyte	Kennung	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	1111 1111	1111 1110	1110 0100	0001 0010	Adresse High	Adresse Low	R00V VVVV	X-Or-Byte
Hex :	0xFF	0xFE	0xE4	0x12	AH	AL	RV	X-Or-Byte
Dez :	255	254	228	18	AH	AL	RV	X-Or-Byte

**Fahrbehl 128 Fahrstufen:****Format:**

	Frame1	Frame2	Headerbyte	Kennung	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	1111 1111	1111 1110	1110 0100	0001 0011	Adresse High	Adresse Low	RVVV VVVV	X-Or-Byte
Hex :	0xFF	0xFE	0xE4	0x13	AH	AL	RV	X-Or-Byte
Dez :	255	254	228	19	AH	AL	RV	X-Or-Byte

**Beschreibung:**

Der Fahrbehl für Version 3-Zentralen enthält nur noch die Geschwindigkeits- und Richtungsinformation. Die Funktionen werden separat gesetzt.

Es können die Loks 0 bis 9999 gesteuert werden.

Die Lokadresse AH/AL berechnet sich wie unter 2.1.10 angegeben.

**Besonderheiten:**

Keine.

### [2.2.26.2](#)     [Funktionsbefehle ab Zentralen-Version 3.0 / Version 3.6](#)

Die Funktionsbefehle für eine Lok gliedern sich in 3 verschiedenen Möglichkeiten auf, die den Funktionen der Gruppe 1 (F0..F4), Gruppe 2 (F5..F8), Gruppe 3 (F9..F12), [Gruppe 4 \(F13...F20\) \(ab V3.6\)](#) und der [Gruppe 5 \(F21...F28\) \(ab V3.6\)](#) zugeordnet sind. Dies wird durch die unterschiedliche Kennung erreicht.

**Funktionsbehl Gruppe 1:**

**Format:**

	Frame1	Frame2	Headerbyte	Kennung	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	1111 1111	1111 1110	1110 0100	0010 0000	Adresse High	Adresse Low	000F FFFF	X-Or-Byte
Hex :	0xFF	0xFE	0xE4	0x20	AH	AL	Gruppe 1	X-Or-Byte
Dez :	255	254	228	32	AH	AL	Gruppe 1	X-Or-Byte

Für die Funktionen gilt:

Daten 3: 0 0 0 F0 F4 F3 F2 F1 Ist Fx=1, so ist die Funktion an, sonst aus.

Funktionsbefehl Gruppe 2:

**Format:**

	Frame1	Frame2	Headerbyte	Kennung	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	1111 1111	1111 1110	1110 0100	0010 0001	Adresse High	Adresse Low	0000 FFFF	X-Or-Byte
Hex :	0xFF	0xFE	0xE4	0x21	AH	AL	Gruppe 2	X-Or-Byte
Dez :	255	254	228	33	AH	AL	Gruppe 2	X-Or-Byte

Für die Funktionen gilt:

Daten 3: 0 0 0 0 F8 F7 F6 F5 Ist Fx=1, so ist die Funktion an, sonst aus..

Funktionsbefehl Gruppe 3:

**Format:**

	Frame1	Frame2	Headerbyte	Kennung	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	1111 1111	1111 1110	1110 0100	0010 0010	Adresse High	Adresse Low	0000 FFFF	X-Or-Byte
Hex :	0xFF	0xFE	0xE4	0x22	AH	AL	Gruppe 3	X-Or-Byte
Dez :	255	254	228	34	AH	AL	Gruppe 3	X-Or-Byte

Für die Funktionen gilt:

Daten 3: 0 0 0 0 F12 F11 F10 F9 Ist Fx=1, so ist die Funktion an, sonst aus..

Funktionsbefehl Gruppe 4 (neu ab Version 3.6):

**Format:**

	Frame1	Frame2	Headerbyte	Kennung	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	1111 1111	1111 1110	1110 0100	0010 0011	Adresse High	Adresse Low	0000 FFFF	X-Or-Byte
Hex :	0xFF	0xFE	0xE4	0x23	AH	AL	Gruppe 4	X-Or-Byte
Dez :	255	254	228	35	AH	AL	Gruppe 4	X-Or-Byte

Für die Funktionen gilt:

Daten 3: F20 F19 F18 F17 F16 F15 F14 F13 Ist Fx=1, so ist die Funktion an, sonst aus..

## Funktionsbefehl Gruppe 5 (neu ab Version 3.6):

**Format:**

	Frame1	Frame2	Headerbyte	Kennung	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	1111 1111	1111 1110	1110 0100	0010 1000	Adresse High	Adresse Low	0000 FFFF	X-Or-Byte
Hex :	0xFF	0xFE	0xE4	0x28	AH	AL	Gruppe 5	X-Or-Byte
Dez :	255	254	228	40	AH	AL	Gruppe 5	X-Or-Byte

Für die Funktionen gilt:

Daten 3: F20 F19 F18 F17 F16 F15 F14 F13 Ist Fx=1, so ist die Funktion an, sonst aus..

**Beschreibung:**

Für die Funktionen gilt:

Gruppe 1: 0 0 0 F0 F4 F3 F2 F1 Ist Fx=1, so ist die Funktion an, sonst aus.

Gruppe 2: 0 0 0 0 F8 F7 F6 F5

Gruppe 3: 0 0 0 0 F12 F11 F10 F9

Es können die Loks 0 bis 9999 angesprochen werden.

Die Lokadresse AH/AL berechnet sich wie unter 2.1.10 angegeben.

**Besonderheiten:**

Keine.

**2.2.26.3 Funktionsstatus setzen****2.2.26.4 Funktionsstatus setzen ab Zentralen-Version 3.0 / Version 3.6**

Die LZ100-Zentrale ab Version 3.0 speichert für jede Lokadresse den Zustand ihrer Funktionen als tastend oder nicht tastend. XpressNet-Geräte können diesen Zustand abfragen und ihre Bedienoberfläche entsprechend gestalten. Diese Funktionalität ist vor allem für Geräusche gedacht.

Wie bei den Funktionen wird auch hier die Gruppe 1, Gruppe 2 und Gruppe 3 durch die Kennung unterschieden.

In Version 3.6 sind neu hinzugekommen die Gruppe 4 (F13...20) und Gruppe 5 (F21...F28).

Funktionsstatus setzen Gruppe 1:

**Format:**

	Frame1	Frame2	Headerbyte	Kennung	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	1111 1111	1111 1110	1110 0100	0010 0100	Adresse High	Adresse Low	000S SSSS	X-Or-Byte
Hex :	0xFF	0xFE	0xE4	0x24	AH	AL	Gruppe 1	X-Or-Byte
Dez :	255	254	228	36	AH	AL	Gruppe 1	X-Or-Byte

Für die Funktionen gilt:

Daten 3: 0 0 0 S0 S4 S3 S2 S1 Ist Sx=1, so ist die Funktion tastend, sonst nicht.

Funktionsstatus setzen Gruppe 2:

**Format:**

	Frame1	Frame2	Headerbyte	Kennung	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	1111 1111	1111 1110	1110 0100	0010 0101	Adresse High	Adresse Low	0000 SSSS	X-Or-Byte
Hex :	0xFF	0xFE	0xE4	0x25	AH	AL	Gruppe 2	X-Or-Byte
Dez :	255	254	228	37	AH	AL	Gruppe 2	X-Or-Byte

Für die Funktionen gilt:

Daten 3: 0 0 0 0 S8 S7 S6 S5 Ist Sx=1, so ist die Funktion tastend, sonst nicht.

Funktionsstatus setzen Gruppe 3:

**Format:**

	Frame1	Frame2	Headerbyte	Kennung	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	1111 1111	1111 1110	1110 0100	0010 0110	Adresse High	Adresse Low	0000 SSSS	X-Or-Byte
Hex :	0xFF	0xFE	0xE4	0x26	AH	AL	Gruppe 3	X-Or-Byte
Dez :	255	254	228	38	AH	AL	Gruppe 3	X-Or-Byte

Für die Funktionen gilt:

Daten 3: 0 0 0 0 S12 S11 S10 S9 Ist Sx=1, so ist die Funktion tastend, sonst nicht.

Funktionsstatus setzen Gruppe 4 (ab Version 3.6):

**Format:**

	Frame1	Frame2	Headerbyte	Kennung	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	1111 1111	1111 1110	1110 0100	0010 0111	Adresse High	Adresse Low	0000 SSSS	X-Or-Byte
Hex :	0xFF	0xFE	0xE4	0x27	AH	AL	Gruppe 4	X-Or-Byte
Dez :	255	254	228	39	AH	AL	Gruppe 4	X-Or-Byte

Für die Funktionen gilt:

Daten 3: S20 S19 S18 S17 S16 S15 S14 S13 Ist Sx=1, so ist die Funktion tastend, sonst nicht.

Funktionsstatus setzen Gruppe 5 (ab Version 3.6):

**Format:**

	Frame1	Frame2	Headerbyte	Kennung	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	1111 1111	1111 1110	1110 0100	0010 1100	Adresse High	Adresse Low	0000 SSSS	X-Or-Byte
Hex :	0xFF	0xFE	0xE4	0x2C	AH	AL	Gruppe 5	X-Or-Byte
Dez :	255	254	228	44	AH	AL	Gruppe 5	X-Or-Byte

Für die Funktionen gilt:

Daten 3: S20 S19 S18 S17 S16 S15 S14 S13 Ist Sx=1, so ist die Funktion tastend, sonst nicht.

### 2.2.26.5 Funktionsrefresh-Modus setzen ab Zentralen-Version 3.6

Unter Funktionsrefresh versteht man das zyklische Wiederholen von Funktionsdaten auf dem Gleis. Ab der Zentralenversion 3.6 ist einstellbar, welche Funktionsdaten refreshed werden. Werkseinstellung ist der Refresh der Funktionen 0 bis 8.

**Format:**

	Frame1	Frame2	Headerbyte	Kennung	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	1111 1111	1111 1110	1110 0101	0010 1111	Adresse High	Adresse Low	Refresh- Modus	X-Or-Byte
Hex :	0xFF	0xFE	0xE5	0x2F	AH	AL	RF	X-Or-Byte
Dez :	255	254	228	47	AH	AL	RF	X-Or-Byte

Für Daten 3 (Refresh-Modus) gilt:

Wert	Refresh für
0	F0 ... F4
1	F0 ... F8
3	F0 ... F12
7	F0 ... F20
0xF	F0 ... F28

**Beschreibung:**

Für die Funktionen gilt:

Gruppe 1: 0 0 0 S0 S4 S3 S2 S1 Ist Sx=1, so ist die Funktion tastend, sonst nicht.

Gruppe 2: 0 0 0 0 S8 S7 S6 S5

Gruppe 3: 0 0 0 0 S12 S11 S10 S9

Es können die Loks 0 bis 9999 angesprochen werden.

Die Lokadresse AH/AL berechnet sich wie unter 2.1.10 angegeben.



**Besonderheiten:**

Keine.

**2.2.27 Doppeltraktionen****2.2.27.1 Doppeltraktion montieren****Format:**

	Frame1	Frame2	Headerbyte	Kennung	Daten 1	Daten 2	Daten 3	Daten 4	X-Or-Byte
Binär :	1111 1111	1111 1110	1110 0101	0100 0011	Adr High 1	Adr Low 1	Adr. High 2	Adr Low 2	X-Or-Byte
Hex :	0xFF	0xFE	0xE5	0x43	AH1	AL1	AH 2	AL 2	X-Or-Byte
Dez :	255	254	229	67	AH1	AL1	AH 2	AL 2	X-Or-Byte

**Beschreibung:**

Die Loks in Daten 1/2 und Daten 3/4 werden in der Zentrale zu einer Doppeltraktion zusammengefügt, was bedeutet, daß ein Fahrbefehl an eine der Loks durch die Zentrale auch an die andere gesendet wird.

Die Lokadressen AH/AL berechnen sich wie unter 2.1.10 angegeben.

Gelingt die Montage nicht, so sendet die Zentrale eine der unter 2.1.14 beschriebenen Fehlermeldungen.

**Besonderheiten:**

Der Befehl ersetzt die alten Doppeltraktionsbefehle, die in späteren Zentralenversionen nicht mehr unterstützt werden.

**2.2.27.2 Doppeltraktion auflösen****Format:**

	Frame1	Frame2	Headerbyte	Kennung	Daten 1	Daten 2	Daten 3	Daten 4	X-Or-Byte
Binär :	1111 1111	1111 1110	1110 0101	0100 0011	Adr High 1	Adr Low 1	0000 0000	0000 0000	X-Or-Byte
Hex :	0xFF	0xFE	0xE5	0x43	AH1	AL1	0x00	0x00	X-Or-Byte
Dez :	255	254	229	67	AH1	AL1	0x00	0x00	X-Or-Byte

**Beschreibung:**

Die Lok in Daten 1/2 wird aus der Doppeltraktion, in der sie eingebunden ist, entfernt. Damit wird auch die Doppeltraktion in der Zentrale aufgelöst.

Daß es sich um das Auflösen einer DTR handelt, kann die Zentrale an der zweiten Lokadresse erkennen, die hier 0 ist.

Die Lokadresse AH/AL berechnet sich wie unter 2.1.15. angegeben.

Gelingt das Auflösen nicht, so sendet die Zentrale eine der unter 2.1.14 beschriebenen Fehlermeldungen.

**Besonderheiten:**

keine

**2.2.28 Programming on Main**

Programming on Main bedeutet, daß CV's eines Empfängers geändert werden können, während die Lok auf dem normalen Gleis steht. Ein Programmiergleis ist in diesem Fall nicht nötig. Allerdings kann hiermit nicht die Adresse eines Empfängers geändert werden, da diese im Programmierbefehl benutzt werden muß.

Zentralen, die Programming on Main nicht unterstützen, senden „Befehl nicht vorhanden“ an das XpressNet-Gerät.

Im Gegensatz zum Programmieren auf dem Programmiergleis sind hier die CV's 1..1024 möglich, allerdings sollten XpressNet-Geräte keine CV's zulassen, die eine Adressänderung zur Folge haben, denn dann würde ein Empfänger nicht mehr auf später ausgesandte Datenpakete hören können, falls er die Sendung auswertet (was allerdings nicht erlaubt ist).

**2.2.28.1 Programming on Main Byte schreiben****Format:**

	Frame1	Frame2	Headerbyte	Kennung	Daten 1	Daten 2	Daten 3	Daten 4	Daten 5	X-Or-Byte
Binär :	1111 1111	1111 1110	1110 0110	0011 0000	Adresse High	Adresse Low	1110 11CC	CCCC CCCC	DDDD DDDD	X-Or- Byte
Hex :	0xFF	0xFE	0xE6	0x30	AH	AL	0xEC + C	CV	D	X-Or- Byte
Dez :	255	254	230	48	AH	AL	236 + C	CV	D	X-Or- Byte

**Beschreibung:**

Daten 1 und Daten 2 geben die Lokadresse von 1..9999 an, auf die sich das Byte-Programmieren bezieht.

Die Lokadresse AH/AL berechnet sich wie unter 2.1.10 angegeben.

Da die CV's von 0..1023 möglich sind (=10Bit), werden die oberen 2 Bits (MSB's) nach Daten 3 geschrieben. Der Rest der CV-Adresse (die 8 LSB's) stehen in Daten 4.

Der zu programmierende Wert dieser CV steht in Daten 5.

Die CV-Adresse wird so gesendet, wie sie auf dem Gleis erscheint, d.h. um eins decrementiert.

**Besonderheiten:**

Es sollten keine CV's verwendet werden, die sich auf Empfängeradressen beziehen.

### 2.2.28.2 Programming on Main Byte lesen (ab Version 3.6)

#### Format:

	Frame1	Frame2	Headerbyte	Kennung	Daten 1	Daten 2	Daten 3	Daten 4	Daten 5	X-Or-Byte
Binär :	1111 1111	1111 1110	1110 0110	0011 0000	Adresse High	Adresse Low	1110 01CC	CCCC CCCC	DDDD DDDD	X-Or-Byte
Hex :	0xFF	0xFE	0xE6	0x30	AH	AL	0xEA + C	CV	D	X-Or-Byte
Dez :	255	254	230	48	AH	AL	234 + C	CV	D	X-Or-Byte

#### Beschreibung:

Daten 1 und Daten 2 geben die Lokadresse von 1..9999 an, auf die sich das Byte-Programmieren bezieht.

Die Lokadresse AH/AL berechnet sich wie unter 2.1.10 angegeben.

Da die CV's von 0..1023 möglich sind (=10Bit), werden die oberen 2 Bits (MSB's) nach Daten 3 geschrieben. Der Rest der CV-Adresse (die 8 LSB's) stehen in Daten 4.

Der zu programmierende Wert dieser CV steht in Daten 5.

Die CV-Adresse wird so gesendet, wie sie auf dem Gleis erscheint, d.h. um eins decrementiert.

#### Besonderheiten:

Es sollten keine CV's verwendet werden, die sich auf Empfängeradressen beziehen.

### 2.2.28.3 Programming on Main Bit schreiben

#### Format:

	Frame1	Frame2	Headerbyte	Kennung	Daten 1	Daten 2	Daten 3	Daten 4	Daten 5	X-Or-Byte
Binär :	1111 1111	1111 1110	1110 0110	0011 0000	Adresse High	Adresse Low	1110 10CC	CCCC CCCC	1111 WBBB	X-Or-Byte
Hex :	0xFF	0xFE	0xE6	0x30	AH	AL	0xE8 + C	CV	WB	X-Or-Byte
Dez :	255	254	230	48	AH	AL	232 + C	CV	WB	X-Or-Byte

#### Beschreibung:

Daten 1 und Daten 2 geben die Lokadresse von 1..9999 an, auf die sich das Bit-Programmieren bezieht.

Die Lokadresse AH/AL berechnet sich wie unter 2.1.10 angegeben.

Da die CV's von 0..1023 möglich sind (=10Bit), werden die oberen 2 Bits (MSB's) nach Daten 3 geschrieben. Der Rest der CV-Adresse (die 8 LSB's) stehen in Daten 4.

Der zu programmierende Bitwert steht in Daten 5 und berechnet sich wie folgt:

W ist der Bitwert 0 oder 1.

Die Bits B2, B1, B0 geben die Position des Bits in der CV an (Bitposition 0 bis Bitposition 7).

Die CV-Adresse wird so gesendet, wie sie auf dem Gleis erscheint, d.h. um eins decrementiert.

**Besonderheiten:**

Es sollten keine CV's verwendet werden, die sich auf Empfängeradressen beziehen.

**2.2.29 Mehrfachtraktionen****2.2.29.1 Lok zu einer Mehrfachtraktion hinzufügen oder MTR erzeugen**

Eine Lok kann zu einer Mehrfachtraktion (MTR) hinzugefügt werden, wenn sie noch in keiner anderen MTR enthalten ist. Ist diese Lok die erste Lok, so wird automatisch eine MTR erzeugt.

Zusätzlich kann die Einfügerichtung definiert werden, so daß eine Lok auch „falsch“ herum in einer MTR sitzen kann, aber dennoch in die korrekte Richtung fährt. Dies wird durch ein Bit in der Kennung (R) entschieden.

**Format:**

	Frame1	Frame2	Headerbyte	Kennung	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	1111 1111	1111 1110	1110 0100	0100 000R	Adresse High	Adresse Low	MTR	X-Or-Byte
Hex :	0xFF	0xFE	0xE4	0x40 + R	AH	AL	MTR	X-Or-Byte
Dez :	255	254	228	64 + R	AH	AL	MTR	X-Or-Byte

**Beschreibung:**

R: R = 0 bedeutet, daß die Lok nicht invertiert in die MTR eingefügt wird. D.h. fährt die

MTR vorwärts, fährt auch die Lok vorwärts.

R = 1 bedeutet, daß die Fahrtrichtung der Lok invertiert wird.

Daten 1 und Daten 2 geben die Lokadresse von 1..9999 an, die in die MTR eingefügt werden soll. Die Lokadresse AH/AL berechnet sich wie unter 2.1.10 angegeben.

MTR: Dies ist die MTR-Basisadresse im Bereich von 1 bis 99.

**Besonderheiten:**

Per Definition kann eine Lok nicht in eine Mehrfachtraktion eingefügt werden, die die gleiche Adresse hat.

### 2.2.29.2 Lok aus einer Mehrfachtraktion entfernen oder MTR löschen

Eine Lok kann aus einer MTR entfernt werden, wenn sie Mitglied dieser MTR ist.  
Mit Entfernen der letzten Lok einer MTR wird auch die MTR in der Zentrale gelöscht.

#### Format:

	Frame1	Frame2	Headerbyte	Kennung	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	1111 1111	1111 1110	1110 0100	0100 0010	Adresse High	Adresse Low	MTR	X-Or-Byte
Hex :	0xFF	0xFE	0xE4	0x42	AH	AL	MTR	X-Or-Byte
Dez :	255	254	228	66	AH	AL	MTR	X-Or-Byte

#### Beschreibung:

Daten 1 und Daten 2 geben die Lokadresse von 1..9999 an die aus der MTR entfernt werden soll.

Die Lokadresse AH/AL berechnet sich wie unter 2.1.10 angegeben.

MTR: Dies ist die Basisadresse oder MTR-Adresse im Bereich von 1 bis 99, unter der die

Mehrfachtraktion gefahren werden kann.

#### Besonderheiten:

Keine.

### 2.2.30 Adress-Suchbefehle

Durch die Einführung von Mehrfachtraktionen und einem erweiterten Stack-Handling in den Zentralen ist es nötig geworden, daß XpressNet-Geräte auch Lokadressen suchen müssen, um eine komfortable Bedieneroberfläche zu erzielen.

#### 2.2.30.1 Adressanfrage Mitglied einer Mehrfachtraktion

Die Unterscheidung zwischen Vorwärts- und Rückwärtssuche wird über die Kennung gemacht.

#### Format:

	Frame1	Frame2	Headerbyte	Kennung	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	1111 1111	1111 1110	1110 0100	0000 00RR	MTR	Adresse High	Adresse Low	X-Or-Byte
Hex :	0xFF	0xFE	0xE4	0x01 + R	MTR	AH	AL	X-Or-Byte
Dez :	255	254	228	1 + R	MTR	AH	AL	X-Or-Byte

#### Beschreibung:

Um einen schnellen Zugriff auf die Loks in einer MTR zu haben, um z.B. Funktionen schalten zu können, liefert die Zentrale aufgrund dieser Anfrage die nächste Adresse, die der angefragten folgt (Vorwärtssuche) bzw. vorausgeht (Rückwärtssuche).

Für Zentralen Version 3.x ist nur die Vorwärtssuche definiert.

Kennung = 0x01: (RR=01) bedeutet Vorwärtssuche

Kennung = 0x02: (RR=10) bedeutet Rückwärtssuche

Daten 1 gibt die MTR-Basisadresse im Bereich von 1..99 an, auf die sich die Suche bezieht.

Daten 2 und Daten 3 geben die Lokadresse von 1..9999 an, zu der die folgende bzw. vorausgehende Adresse gesucht werden soll.

Die Lokadresse AH/AL berechnet sich wie unter 2.1.10 angegeben.

MTR: Dies ist die Basisadresse oder MTR-Adresse im Bereich von 1 bis 99, unter der die Mehrfachtraktion gefahren werden kann.

Das Ergebnis der Suche wird an das Gerät in der Antwort wie unter 2.1.13 beschrieben gesendet.

#### **Besonderheiten:**

Keine.

### **2.2.30.2     Adressanfrage Mehrfachtraktion**

Die Unterscheidung zwischen Vorwärts- und Rückwärtssuche wird über die Kennung gemacht.

#### **Format:**

	Frame1	Frame2	Headerbyte	Kennung	Daten 1	X-Or-Byte
Binär :	1111 1111	1111 1110	1110 0010	0000 0RRR	MTR	X-Or-Byte
Hex :	0xFF	0xFE	0xE2	0x03 + R	MTR	X-Or-Byte
Dez :	255	254	226	3 + R	MTR	X-Or-Byte

#### **Beschreibung:**

Dieser Befehl veranlasst die Zentrale dazu, dem XpressNet-Gerät die nächste Basisadresse einer MTR zu senden, die der angefragten MTR folgt (Vorwärtssuche) bzw. vorausgeht (Rückwärtssuche).

Für Zentralen Version 3.x ist nur die Vorwärtssuche definiert.

Kennung = 0x03: (RRR=011) bedeutet Vorwärtssuche

Kennung = 0x04: (RRR=100) bedeutet Rückwärtssuche

MTR: Dies ist die Basisadresse oder MTR-Adresse im Bereich von 1 bis 99, unter der die Mehrfachtraktion gefahren werden kann.

Das Ergebnis der Suche wird an das Gerät in der Antwort wie unter 2.1.13 beschrieben gesendet.

#### **Besonderheiten:**

Keine.

### 2.2.30.3 Adressanfrage Lok in Zentralenstack

Die Unterscheidung zwischen Vorwärts- und Rückwärtssuche wird über die Kennung gemacht.

#### Format:

	Frame1	Frame2	Headerbyte	Kennung	Daten 1	Daten 2	X-Or-Byte
Binär :	1111 1111	1111 1110	1110 0011	0000 01RR	Adresse High	Adresse Low	X-Or-Byte
Hex :	0xFF	0xFE	0xE3	0x05 + R	AH	AL	X-Or-Byte
Dez :	255	254	227	5 + R	AH	AL	X-Or-Byte

#### Beschreibung:

Es wird dem XpressNet-Gerät die Lokadresse gesendet, die im Zentralenstack hinter (Vorwärtssuche) bzw. vor der Lokadresse (Rückwärtssuche) in Daten 1/2 abgelegt ist. Für Zentralen Version 3.x ist nur die Vorwärtssuche definiert.

Kennung = 0x05: (RR=01) bedeutet Vorwärtssuche

Kennung = 0x06: (RR=10) bedeutet Rückwärtssuche

Daten 1 und Daten 2 geben die Lokadresse von 0..9999 an, zu der die folgende bzw. vorausgehende Adresse gesucht werden soll. Die Lokadresse AH/AL berechnet sich wie unter 2.1.10 angegeben.

Das Ergebnis der Suche wird an das Gerät in der Antwort wie unter 2.1.13 beschrieben gesendet.

#### Besonderheiten:

Keine.

### 2.2.31 Lok aus Zentralenstack löschen

#### Format:

	Frame1	Frame2	Headerbyte	Kennung	Daten 1	Daten 2	X-Or-Byte
Binär :	1111 1111	1111 1110	1110 0011	0100 0100	Adresse High	Adresse Low	X-Or-Byte
Hex :	0xFF	0xFE	0xE3	0x44	AH	AL	X-Or-Byte
Dez :	255	254	227	68	AH	AL	X-Or-Byte

#### Beschreibung:

Daten 1 und Daten 2 geben die Lokadresse von 1..9999 an, die im Zentralenstack zu löschen ist.

Die Lokadresse AH/AL berechnet sich wie unter 2.1.10 angegeben.

#### Besonderheiten:

Das Problem, daß der Zentralenstack voll ist, stellt sich vor allem bei Zentralen mit geringer Hardwareausstattung, die dann nicht alle jemals aufgerufenen Loks mit allen

Daten speichern können. Der Zentralenstack dient dazu, die Daten dieser Loks auf das Gleis zu senden.

Das XpressNet-Gerät welches die Lok im Stack gelöscht hat, sollte dafür sorgen, daß es mit einer anderen Lok weiterarbeiten kann, so daß das Löschen auch von Erfolg gekrönt wird. Ansonsten würde man die Lok, die man gerade gelöscht hat, sofort wieder aufnehmen.



### 3 Befehlsübersicht Zentrale an PC

Die exakte Bedeutung der jeweiligen Datenbytes ist in den entsprechenden Kapiteln beschrieben. N im Headerbyte gibt die Zahl der folgenden Datenbytes an.

Befehl	Frame1	Frame2	Header	Daten1	Daten2	Daten3	Daten4	Daten5	Daten6	Daten7
BC Alles An	0xFF	0xFD	0x61	0x01	0x60					
BC Alles Aus	0xFF	0xFD	0x61	0x00	0x61					
BC Alle Loks Aus	0xFF	0xFD	0x81	0x00	0x81					
BC Programmiermode	0xFF	0xFD	0x61	0x02	0x63					
BC Rückmeldung	0xFF	0xFD	0x40 + N	ADR_1	DAT_1	ADR_2	DAT2	usw.	usw.	X-Or
P-Info Kurzschluß	0xFF	0xFE	0x61	0x12	X-Or					
P-Info Keine Daten	0xFF	0xFE	0x61	0x13	X-Or					
P-Info Busy	0xFF	0xFE	0x61	0x1f	X-Or					
P-Info bereit	0xFF	0xFE	0x61	0x11	X-Or					
P-Info Daten 3 Byte	0xFF	0xFE	0x63	0x10	EE	DAT	X-Or			
P-Info CV1-255 u. 1024	0xFF	0xFE	0x63	0x14	CV	DAT	X-Or			
P-Info CV256 - 511	0xFF	0xFE	0x63	0x15	CV	DAT	X-Or			
P-Info CV512 - 767	0xFF	0xFE	0x63	0x16	CV	DAT	X-Or			
P-Info CV768 - 1023	0xFF	0xFE	0x63	0x17	CV	DAT	X-Or			
Softwareversion	0xFF	0xFE	0x63	0x21	DAT1	DAT2	X-Or			
Status Zentrale	0xFF	0xFE	0x62	0x22	DAT	X-Or				
Übertragungsfehler	0xFF	0xFE	0x61	0x80	X-Or					
Zentrale Busy	0xFF	0xFE	0x61	0x81	X-Or					
Befehl nicht vorhanden	0xFF	0xFE	0x61	0x82	X-Or					
Schaltinformationen	0xFF	0xFE	0x42	ADR	DAT	X-Or				
Normale Lokinfo	0xFF	0xFE	0xE4	Kennung	Speed	FKT0	FKT1	X-Or		
F-Zustand F13 ... F28	0xFF	0xFE	0xE3	0x52	F 13-20	F 21-28	X-Or			
MTR-Mitglied	0xFF	0xFE	0xE5	Kennung	Speed	FKT0	FKT1	MTR	X-Or	
MTR-Basisadresse	0xFF	0xFE	0xE2	Kennung	Speed	X-Or				
Lok ist in DTR	0xFF	0xFE	0xE6	Kennung	Speed	FKT0	FKT1	ADR High	ADR Low	X-Or
Lok besetzt	0xFF	0xFD	0xE3	0x40	ADR High	ADR Low	X-Or			
Funktionsstatus	0xFF	0xFE	0xE3	0x50	STAT 0	STAT 1	X-Or			
F-status F13 bis 28	0xFF	0xFE	0xE4	0x51	STAT 2	STAT 3	X-Or			
Lok-Suchergebnis	0xFF	0xFE	0xE3	0x30 + K	ADR High	ADR Low	X-Or			
DTR-Fehler	0xFF	0xFE	0x61	0x80 + F	X-Or					
Fehlermeldung	0xFF	0xFE	0xE1	0x80 + F	X-Or					

## 4 Befehlsübersicht PC an Zentrale

Die Bedeutung der jeweiligen Datenbytes ist in den entsprechenden Kapiteln beschrieben.

Befehl	Frame1	Frame2	Header	Kennun g	Daten1	Daten2	Daten3	Daten4	Daten5	Daten6
Alles An	0xFF	0xFE	0x21	0x81	0xA0					
Alles Aus	0xFF	0xFE	0x21	0x80	0xA1					
Alle Loks anhalten	0xFF	0xFE	0x80	0x80						
Eine Lok anhalten	0xFF	0xFE	0x92	ADR High	ADR Low	X-Or				
Prog.-Lesen Register	0xFF	0xFE	0x22	0x11	REG	X-Or				
Prog.-Lesen CV	0xFF	0xFE	0x22	0x15	CV	X-Or				
Prog.-Lesen CV1-255; 1024	0xFF	0xFE	0x22	0x18	CV low	X-Or				
Prog.-Lesen CV256-511	0xFF	0xFE	0x22	0x19	CV low	X-Or				
Prog.-Lesen CV512-767	0xFF	0xFE	0x22	0x1A	CV low	X-Or				
Prog.-Lesen CV768-1023	0xFF	0xFE	0x22	0x1B	CV low	X-Or				
Prog.-Lesen Paging	0xFF	0xFE	0x22	0x14	CV	X-Or				
Prog.-Ergebnis anfordern	0xFF	0xFE	0x21	0x10	0x31					
Prog.-Schreiben Register	0xFF	0xFE	0x23	0x12	REG	DAT	X-Or			
Prog.-Schreiben CV	0xFF	0xFE	0x23	0x16	CV	DAT	X-Or			
Prog.-Schr. CV1-255; 1024	0xFF	0xFE	0x23	0x1C	CV low	DAT	X-Or			
Prog.-Schr. CV256-511	0xFF	0xFE	0x23	0x1D	CV low	DAT	X-Or			
Prog.-Schr. CV512-767	0xFF	0xFE	0x23	0x1E	CV low	DAT	X-Or			
Prog.-Schr. CV768-1023	0xFF	0xFE	0x23	0x1F	CV low	DAT	X-Or			
Prog.-Schreiben Paging	0xFF	0xFE	0x23	0x17	CV	DAT	X-Or			
Softwareversion anfordern	0xFF	0xFE	0x21	0x21	0x00					
Status Zentrale anfordern	0xFF	0xFE	0x21	0x24	0x05					
Schaltinformation anfordern	0xFF	0xFE	0x42	ADR	Nibble	X-Or				
Schaltbefehl	0xFF	0xFE	0x52	ADR	DAT	X-Or				
Lokdaten anfordern	0xFF	0xFE	0xE3	0x00	ADR High	ADR Low	X-Or			
Fkt-Status anfordern	0xFF	0xFE	0xE3	0x07	ADR High	ADR Low	X-Or			
Fkt-Status anf. F13-F28	0xFF	0xFE	0xE3	0x08	ADR High	ADR Low	X-Or			
Fkt-Zustand anf. F13-F28	0xFF	0xFE	0xE3	0x09	ADR High	ADR Low	X-Or			
Lok Fahrbefehl	0xFF	0xFE	0xE4	Kennung	ADR High	ADR Low	Speed	X-Or		
Lok Funktionsbefehl	0xFF	0xFE	0xE4	Kennung	ADR High	ADR Low	Gruppe	X-Or		
Funktionsstatus setzen	0xFF	0xFE	0xE4	Kennung	ADR High	ADR Low	Gruppe	X-Or		
Func.refresh-Modus setzen	0xFF	0xFE	0xE5	0x2F	ADR High	ADR Low	Modus	X-Or		
Funktionsstatus setzen	0xFF	0xFE	0xE4	Kennung	ADR High	ADR Low	Gruppe	X-Or		
DTR-Befehle	0xFF	0xFE	0xE5	0x43	ADR1 H	ADR1 L	ADR2 H	ADR2 L	X-Or	
Prog. on Main Byte schreiben	0xFF	0xFE	0xE6	0x30	ADR High	ADR Low	0xEC + C	CV	DAT	X-Or
Prog. on Main Byte lesen	0xFF	0xFE	0xE6	0x30	ADR High	ADR Low	0xEA + C	CV	DAT	X-Or
Prog. on Main Bit	0xFF	0xFE	0xE6	0x30	ADR High	ADR Low	0xE8 + C	CV	DAT	X-Or
Lok zu MTR hinzufügen	0xFF	0xFE	0xE4	0x40 + R	ADR High	ADR Low	MTR	X-Or		
Lok aus MTR entfernen	0xFF	0xFE	0xE4	0x42	ADR High	ADR Low	MTR	X-Or		
Adresssuche Lok in Mtr	0xFF	0xFE	0xE4	0x01 + R	MTR	ADR High	ADR Low	X-Or		

Adresssuche MTR	0xFF	0xFE	0xE2	0x03 + R	MTR	X-Or				
Stacksuche Lok	0xFF	0xFE	0xE3	0x05 + R	ADR High	ADR Low	X-Or			
Lok aus Stack löschen	0xFF	0xFE	0xE3	0x44	ADR High	ADR Low	X-Or			