



Befehlsbeschreibung
XpressNet V3.6
mit
Interface LI101F

Achtung:
In dieser Doku sind die Befehle der Versionen
1.x und 2.x, die ab der Version 3 durch neue
ersetzt wurden, nicht mehr gelistet

Inhalt

1	Allgemeines	1-1
1.1	<i>XpressNet-Aufbau.....</i>	1-2
1.2	<i>Ungefragte Informationen</i>	1-3
1.3	<i>Lokadressen im Digital plus-System.....</i>	1-3
1.4	<i>Programmierhinweise</i>	1-4
1.5	<i>Antworten des LI101F.....</i>	1-7
1.6	<i>Versionsnummer des LI101F feststellen</i>	1-8
1.7	<i>Geräteadresse des LI101F feststellen und ändern.....</i>	1-9
1.8	<i>Baudrate des LI101F feststellen und ändern.....</i>	1-10
1.9	<i>Break-Signal und Reset des LI101F.....</i>	1-10
2	Datenverkehr Zentrale an PC.....	2-1
2.1	<i>Broadcast.....</i>	2-1
2.1.1	<i>BC „Alles An“.....</i>	2-1
2.1.2	<i>BC „Alles Aus“ (Notaus).....</i>	2-2
2.1.3	<i>BC „Alle Loks Aus“ (Nothalt).....</i>	2-2
2.1.4	<i>BC „Programmiermode“</i>	2-3
2.1.5	<i>BC „Rückmeldung“.....</i>	2-3
2.2	<i>Programmierinformationen</i>	2-4
2.2.1	<i>Programmierinfo „Kurzschluß“.....</i>	2-4
2.2.2	<i>Programmierinfo „Daten nicht gefunden“.....</i>	2-4
2.2.3	<i>Programmierinfo „Zentrale Busy“.....</i>	2-5
2.2.4	<i>Programmierinfo „Zentrale Bereit“</i>	2-5
2.2.5	<i>Programmierinfo „Daten 3-Byte-Format“</i>	2-5
2.2.6	<i>Programmierinfo „Daten 4-Byte-Format“ CV 1-255 und CV1024 (neu ab Version 3.6; ersetzt gleichen Befehl bis Version 3)</i>	2-6
2.2.7	<i>Programmierinfo „Daten 4-Byte-Format“ CV256 bis CV511 (neu ab Version 3.6).....</i>	2-6
2.2.8	<i>Programmierinfo „Daten 4-Byte-Format“ CV512 bis CV767 (neu ab Version 3.6).....</i>	2-7
2.2.9	<i>Programmierinfo „Daten 4-Byte-Format“ CV768 bis CV1023 (neu ab Version 3.6).....</i>	2-8
2.3	<i>Softwareversion Zentrale</i>	2-8
2.3.1	<i>Softwareversion bis LZ-Version 2.3.....</i>	2-8
2.3.2	<i>Softwareversion ab LZ-Version 3.0</i>	2-9
2.4	<i>Status Zentrale.....</i>	2-9
2.5	<i>Übertragungsfehler</i>	2-10
2.6	<i>Zentrale Busy.....</i>	2-10
2.7	<i>Befehl in Zentrale nicht vorhanden</i>	2-11
2.8	<i>Schaltinformation</i>	2-11
2.9	<i>Lokinformationen ab Zentralen-Version 3.0.....</i>	2-13
2.9.1	<i>Lokinformation normale Lok.....</i>	2-14
2.10	<i>Funktionszustand F13 bis F28 der angefragten Lok (ab Version 3.6).....</i>	2-15
2.10.1	<i>Lokinformation Lok befindet sich in einer Mehrfachtraktion</i>	2-16
2.10.2	<i>Lokinformation Lokadresse ist die Basisadresse einer Mehrfachtraktion</i>	2-16
2.10.3	<i>Lokinformation Lok befindet sich in einer Doppeltraktion</i>	2-17
2.11	<i>Lok besetzt ab Zentralen-Version 3.0.....</i>	2-17
2.12	<i>Funktionsstatus der angefragten Lok ab Zentralen-Version 3.0.....</i>	2-18
2.13	<i>Funktionsstatus F13 bis F28 der angefragten Lok (ab Version 3.6).....</i>	2-18
2.14	<i>Lokinformation bei Adress-Suchanfragen ab Zentralen-Version 3.0.....</i>	2-19

2.15	<i>Fehlermeldungen ab Zentralen-Version 3.0</i>	2-20
3	Datenverkehr PC an Zentrale	3-21
3.1	<i>Alles An</i>	3-21
3.2	<i>Alles Aus (Notaus)</i>	3-21
3.3	<i>Alle Loks anhalten (Nothalt)</i>	3-22
3.4	<i>Eine Lok anhalten (Nothalt für eine Lok)</i>	3-22
3.5	<i>Leseanfrage Programmieren 3-Byte-Format (Registermode)</i>	3-22
3.6	<i>Leseanfrage Programmieren 4-Byte-Format (CV-Mode, CV 1-256)</i>	3-23
3.7	<i>Leseanfrage Programmieren 4-Byte-Format (CV 1-255 und CV1024) (neu ab V3.6)</i>	3-24
3.8	<i>Leseanfrage Programmieren 4-Byte-Format (CV 256-511) (neu ab V3.6)</i>	3-24
3.9	<i>Leseanfrage Programmieren 4-Byte-Format (CV 512-767) (neu ab V3.6)</i>	3-25
3.10	<i>Leseanfrage Programmieren 4-Byte-Format (CV 768-1023) (neu ab V3.6)</i>	3-25
3.11	<i>Leseanfrage Programmieren 3-Byte-Format (Pagemode)</i>	3-26
3.12	<i>Programmiererergebnis anfordern</i>	3-26
3.13	<i>Schreibbefehl Programmieren 3-Byte-Format (Register-Mode)</i>	3-27
3.14	<i>Schreibbefehl Programmieren 4-Byte-Format (CV-Mode, CV 1-256)</i>	3-27
3.15	<i>Schreibbefehl Programmieren 4-Byte-Format (CV-Mode, CV 1-255 und CV1024) (neu ab Version 3.6)</i>	3-28
3.16	<i>Schreibbefehl Programmieren 4-Byte-Format (CV-Mode, CV 256-511) (neu ab Version 3.6)</i>	3-28
3.17	<i>Schreibbefehl Programmieren 4-Byte-Format (CV-Mode, CV 512-767) (neu ab Version 3.6)</i>	3-29
3.18	<i>Schreibbefehl Programmieren 4-Byte-Format (CV-Mode, CV 768-1023) (neu ab Version 3.6)</i>	3-29
3.19	<i>Schreibbefehl Programmieren 3-Byte-Format (Page-Mode)</i>	3-30
3.20	<i>Softwareversion der Zentrale anfordern</i>	3-30
3.21	<i>Status der Zentrale anfordern</i>	3-31
3.22	<i>Zentralen-Startmode setzen</i>	3-31
3.23	<i>Schaltinformationen anfordern</i>	3-32
3.24	<i>Schaltbefehl</i>	3-32
3.25	<i>Lokinformationen anfordern</i>	3-33
3.25.1	<i>Funktionsstatus anfordern</i>	3-33
3.25.2	<i>Funktionsstatus anfordern F13 – F28 (neu ab Zentralen-Version 3.6)</i>	3-34
3.25.3	<i>Funktionszustand anfordern F13 – F28 (neu ab Zentralen-Version 3.6)</i>	3-34
3.26	<i>Lok steuern</i>	3-35
3.26.1	<i>Fahrbefehle</i>	3-35
3.26.2	<i>Funktionsbefehle ab Zentralen-Version 3.0 / Version 3.6</i>	3-36
3.26.3	<i>Funktionsstatus setzen ab Zentralen-Version 3.0 / Version 3.6</i>	3-37
3.26.4	<i>Funktionsrefresh-Modus setzen ab Zentralen-Version 3.6</i>	3-39
3.27	<i>Doppeltraktionen</i>	3-40
3.27.1	<i>Doppeltraktion montieren</i>	3-40
3.27.2	<i>Doppeltraktion auflösen</i>	3-41
3.28	<i>Programming on Main</i>	3-41
3.28.1	<i>Programming on Main Byte schreiben</i>	3-41
3.28.2	<i>Programming on Main Byte lesen (ab Version 3.6)</i>	3-42
3.28.3	<i>Programming on Main Bit schreiben</i>	3-42
3.29	<i>Mehrfachtraktionen</i>	3-43
3.29.1	<i>Lok zu einer Mehrfachtraktion hinzufügen oder MTR erzeugen</i>	3-43

3.29.2	Lok aus einer Mehrfachtraktion entfernen oder MTR löschen.....	3-44
3.30	<i>Adress-Suchbefehle</i>	3-44
3.30.1	Adressanfrage Mitglied einer Mehrfachtraktion	3-44
3.30.2	Adressanfrage Mehrfachtraktion.....	3-45
3.30.3	Adressanfrage Lok in Zentralenstack	3-45
3.31	<i>Lok aus Zentralenstack löschen</i>	3-46
4	Befehlsübersicht Zentrale an PC	4-1
5	Befehlsübersicht PC an Zentrale	5-1
6	Unterschiede zu LI100F	6-1

1 Allgemeines

Die vorliegende Dokumentation enthält die Befehlsbeschreibung des XpressNet für die Benutzung mit einem Computer über das Interface LI101F. Bei Verwendung des LI100 und LI100F beachten Sie bitte die Beschreibung, die diesen Geräten beiliegt. Neben den neuen XpressNet-Befehlen sind hier auch die Befehle enthalten, die bis einschließlich Version 3.5 der Zentralen verwendet wurden, sowie zukünftige Erweiterungen, die zum Teil schon in Zentralen oder XpressNet-Geräten (im folgenden auch Slaves genannt) vorgesehen sind, jedoch erst bei späteren Versionen benutzt werden. Unterschieden wird bei den Zentralen nach Versionen bis einschließlich 1.5, von Version 1.5 bis einschließlich Version 2.3 und ab Version 3.0 oder höher.

Da die Struktur des XpressNet ein genau definiertes Timing der Teilnehmer erfordert und die Übertragung von Daten nicht den üblichen PC-Gegebenheiten entspricht, wurde eine einfachere Steuermöglichkeit der Modellbahnanlage vom PC aus über ein Interface LI101F vorgesehen. Hierbei ist das LI101F der eigentliche XpressNet-Teilnehmer, der für das erforderliche Timing sorgt und die Kommunikation mit einem PC ermöglicht, welcher dann asynchron Daten mit dem Interface austauscht. Das Interface reicht hierbei die vom PC erhaltenen Befehle unverändert an die Zentrale weiter und gibt die Antworten der Zentrale an den PC, sobald sie verfügbar sind. Da die Zentrale einer Modellbahnanlage das zeitbestimmende System ist, muß das LI101F zum einen in der Lage sein, Befehle vom PC nur dann zuzulassen, wenn dies gerade möglich ist und zum anderen Informationen, die es von der Zentrale bekommen hat, unverzüglich zum PC weiterzureichen, damit nichts verlorenggeht. Dies wird dadurch gewährleistet, daß das Interface über eine serielle Schnittstelle mit dem PC verbunden wird und folgende Übertragungsdaten festgelegt werden:

- Baudraten: 19200, 38400, 57600, 115200 per Software einstellbar
- 8 Datenbits, 1 Startbit, 1 Stopbit, kein Paritybit
- Hardware-Handshake

Der Hardware-Handshake wird so vereinbart, daß das Interface mit dem CTS-Signal die Datenübertragung vom PC zulassen oder stoppen kann. Das RTS-Signal vom PC wird nicht beachtet, damit jedes Zentralenkommando an den PC gesendet wird.

Der PC muß sich also trotz seiner hohen Rechenleistung in der Verarbeitungsgeschwindigkeit der Modellbahn anpassen. Dies bedeutet, daß gegebenenfalls niedrigere Baudrateneinstellungen zu wählen sind.

In Bezug auf „USB auf Seriell“-Adapter ist zu sagen, daß diese nicht den Anforderungen genügen, die eine echte RS232-Schnittstelle auszeichnen. Dies betrifft zum einen die Spannungspegel, mit denen oftmals gearbeitet wird und die nicht ausreichen können, die Versorgung des LI101F sicherzustellen. Zum anderen werden die Handshakeleitungen per Software emuliert, so daß der erforderliche byteweise Handshake nicht garantiert ist. Sollte ein Adapter augenscheinlich funktionieren, so ist dies keine Gewähr, daß dies auch in jeder Betriebssituation so ist. Das LI101F ist für den ausschließlichen Betrieb an einer echten RS232-Schnittstelle konzipiert.

1.1 XpressNet-Aufbau

Auch wenn zum Betrieb der Modellbahn mit dem PC über das Interface LI101F die Struktur des XpressNet nicht bekannt sein muß, ist etwas Hintergrundwissen dazu nützlich.

Über das XpressNet werden die Eingabegeräte des Digital plus Systems mit der Zentrale verbunden. Das XpressNet ist ein serieller Bus nach RS485-Standard im Half-Duplex-Betrieb mit differentieller Signalübertragung. Übertragen werden:

1 Startbit (0), 9 Datenbits, 1 Stopbit (1), kein Paritybit

Baudrate : 62,5 kbaud

Ein Paritybit wird nur im Rufbyte von der Zentrale an ein Gerät eingefügt (das Rufbyte erscheint nicht am PC und ist in dieser Beschreibung auch nicht enthalten, es ist dasjenige, welches dem Headerbyte vorausgeht). Dies ist das Bit 7. Bit 7 wird so ergänzt, daß eine gerade Anzahl Einsen im Rufbyte steht (gerade Parität). Das 9. Bit wird hardwareseitig als Adressbit (1) oder Datenbit (0) hinzugefügt. Für 8051-Derivate: 9-Bit-UART, Mode 2 mit Adress-/Datenkennung der seriellen Schnittstelle. Die Datenübertragung eines XpressNet-Gerätes an die Zentrale erfolgt immer so, daß das 9. Datenbit vom Gerät auf „0“ zu setzen ist. Auch das LI101F ist ein XpressNet-Gerät welches dieses Datenbit mit 0 überträgt.

Pinbelegung des XpressNet :

L – Versorgung der Busteilnehmer (12VDC)

M – Masse

A – Receive/Transmit nicht invertierend

B – Receive/Transmit invertierend

Es ist auf korrekten Anschluß der Geräte an die A- und B-Leitung zu achten. Ein Vertauschen führt dazu, daß keine Kommunikation zustande kommt. Aufgrund der Datenübertragungsrate von 62,5kbaud kann die Verbindung zwischen den Teilnehmern bis zu 1000m lang sein. Auch ist es nicht notwendig, eine bestimmte Busstruktur einzuhalten. So sind z.B. Stichleitungen, stern- und ringförmige Verdrahtungen möglich. Bei Übertragungsproblemen bei sehr weitläufigen Verdrahtungen oder unter ungünstigen Umgebungsbedingungen kann es nötig sein, den Bus verdrillt zu verlegen und mit einem Widerstand von 120Ω abzuschließen. Die Zentrale LZ100 enthält bereits einen Abschlußwiderstand.

Die Kommunikation auf dem XpressNet erfolgt so, daß die Zentrale als einziger Master-Teilnehmer alle angeschlossenen Geräte (Slaves) zyklisch adressiert und diese dann Nachrichten mit der Zentrale austauschen können. Auch das Interface LI101F wird so in bestimmten Zeitabschnitten angesprochen.

Ist ein solcher Austausch abgeschlossen, wird das nächste Gerät adressiert. Zuvor wird die aktuelle Befehlssequenz abgearbeitet. Hierzu gehört z.B. auch, daß die Zentrale einen Notaus-Befehl bearbeitet und dann eine Broadcast an alle Teilnehmer schickt und erst danach weiter adressiert. Dies kann u. U. lange dauern, je nach den vom Slave gewünschten Befehlen (z.B. Programmierung von Lokempfängern).

Abweichend von diesem Frage-Antwort-Verfahren sendet die Zentrale den Teilnehmern auch Informationen, die diese nicht explizit angefragt haben, auf die sie aber gegebenenfalls reagieren müssen (z.B. Anzeige, Lokübernahme etc.).

Zur Fehlerkontrolle wird jedem Datenpaket ein X-Or-Byte hinzugefügt, welches der Empfänger prüfen muß.

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-Fahrbehle) 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)
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 Lokadressen im Digital plus-System

Mit der Version 3.0 der Digital plus-Zentralen ist als eine neue Eigenschaft eine größere Anzahl von Lokadressen realisiert worden. Dies hat u.a. Konsequenzen für die Art und Weise der Datenübertragung auf dem XpressNet. Die bisher benutzten Lokadressen 0 bis 99 lassen sich bekanntermaßen in einem Byte übertragen, welches 256 verschiedene Werte (8 Bit) annehmen kann. Dies reicht aber für Lokadressen von 0 bis 9999 nicht aus, so daß hierfür zwei Byte benutzt werden müssen. Die folgenden Betrachtungen und Berechnungen werden hexadezimal angegeben, da dies eine bessere Darstellung erlaubt.

Nach NMRA-Standard werden kurze Lokadressen und lange Lokadressen unterschieden. Die kurzen Adressen (ein Byte) laufen von 0 bis 127dez., die langen Lokadressen (2 byte) von 0 bis 16383 dez. Die kurzen Adressen bestehen also aus 7 Bit, die langen Adressen aus 14 Bit. Um die Benutzung für den Anwender nicht zu kompliziert zu machen, haben wir für das Digital plus-System die Vereinbarung getroffen, daß die Lokadressen von 0 bis 99 kurze Adressen sind und die Lokadressen von 100 bis 9999 lange Adressen. Es wird also bewußt auf eine (erlaubte) Überschneidung der Bereiche verzichtet.

Die Darstellung der kurzen Adressen ist einfach: 0 bis 99dez. ergibt 0x00 bis 0x63 hex.

Bei den langen Adressen werden die beiden obersten Bits des Highbytes zu 1 gesetzt. D.h. die Lokadresse mit 14 Bit Länge erhält zusätzlich einen Offset von 0xC000 hex. Mit dem begrenzten Wertebereich ergibt sich also: Lokadresse 100 bis 9999 ist 0xC064 bis 0xE70F.

Bei den neuen Befehlen der Version 3 sind grundsätzlich zwei Byte für Lokadressen reserviert, so daß also auch zwei Byte für eine kurze Adresse übertragen werden müssen.

Wenn in den folgenden Beschreibungen eine zwei-Byte-Lokadresse erforderlich ist (gekennzeichnet durch AH und AL für High- und Lowbyte der Lokadresse), so muß vom PC also immer folgendes gesendet werden:

für Lokadressen von 0 bis 99: 0x0000 bis 0x0063

für Lokadressen 100 bis 9999: 0xC064 bis 0xE70F

(=0x0064 + 0xC000 bis 0x270F + 0xC000)

1.4 Programmierhinweise

Die Modellbahn wird digital dadurch gesteuert, daß eine Zentrale das von der NMRA genormte Gleissignal erzeugt, welches Lokdecoder etc. auswerten. Im Falle des Digital plus-Systems ist diese Zentrale immer auf der Anlage präsent, wird also nicht durch die Software eines PC ersetzt.

Die verschiedenen Eingabegeräte wie Handregler, Stellwerk und auch Interface sind über einen eigenen Bus, dem XpressNet, mit der Zentrale verbunden. Über das XpressNet wird der Zentrale mitgeteilt, welche Lok z.B. eine neue Geschwindigkeit erhalten soll, welche Weiche zu schalten ist etc. Die Zentrale setzt diese Informationen dann auf das Gleisformat um und sendet sie damit an den Empfänger, ohne allerdings zu wissen, ob dieser auch vorhanden ist. Im wesentlichen bedeutet dies, daß der Zentrale einmalig ein Befehl über das XpressNet mitgeteilt wird (z.B. Lok Nr. 3 auf Geschwindigkeit 10) und die Zentrale diese Information dann zyklisch auf das Gleis legt, damit der Empfänger diese Nachricht auch irgendwann empfangen kann. Dazu ist in der Zentrale ein Datensatz für diese Lok abgelegt, der immer wieder ausgesendet wird. Vom XpressNet-Gerät aus wird dann, vereinfacht gesagt, nur dieser Datensatz verändert. Umgekehrt kann ein XpressNet-Gerät diesen Datensatz (oder Teile davon) auch bei der Zentrale erfragen, um z.B. die aktuelle (Soll-)Geschwindigkeit einer Lok anzuzeigen oder deren Fahrtrichtung. Es wäre ein falscher Ansatz, wenn man per Programm versucht, das zyklische Aussenden von Information durch die Zentrale auf das Gleis nachzubilden. Man schränkt dadurch nur den möglichen Datendurchsatz des XpressNet ein, ohne einen Nutzen zu erzielen. Ein PC-Programm sollte ereignisgesteuert reagieren und nicht zyklusgesteuert.

Die Kommunikation auf dem XpressNet-Bus erfordert ein festgelegtes Timing, welches von einem PC üblicherweise nicht so einfach eingehalten werden kann, weil man selten direkten Zugriff auf die Schnittstellenbausteine hat. Damit eine Kommunikation trotzdem einigermaßen einfach bewerkstelligt werden kann, dient das Interface LI101F als Zwischenstation. Das LI101F übernimmt den zeitkritischen Teil des Bustimings und ermöglicht über einen Hardware-Handshake eine asynchrone Kommunikation mit dem PC über eine serielle Schnittstelle, die von jeder üblichen PC-Programmiersoftware

unterstützt wird. Dies bedeutet allerdings keineswegs, daß die Modellbahnanlage sich auch so bedienen läßt, wie es eine gerade gängige PC-Ausstattung erlaubt. Da der zeitbestimmende und -vorgebende Teil die Zentrale und oftmals eben die Gleisausgabe selbst ist, muß sich alles und damit auch der PC nach deren Eigenschaften richten. Es würde also nicht funktionieren, einen großen Datensatz anzulegen, bei dem viele Loks unterschiedliche Daten erhalten und diesen wie bei einem Drucker als Job loszuschicken und nicht mehr zu beachten. Vielmehr muß, da es sich um die Steuerung eines Echtzeitsystems handelt, ein erteilter Befehl auf seine Akzeptierung bzw. Ausführung geprüft werden, damit keine Inkonsistenzen entstehen, wenn nachfolgende Befehle einen bestimmten Zustand voraussetzen, der nicht eintrat. Damit dies erfolgen kann, hat das LI101F die Möglichkeit, die Datenübertragung des PC über den Hardware-Handshake anzuhalten und wieder freizugeben. Dies auch zu beachten liegt allerdings in der Verantwortung des Programmierers.

Wie bei einigen neuen Befehlen zu erkennen ist, beinhaltet die Antwort der Zentrale z.B. auf eine Lokanfrage nicht mehr die Adresse dieser Lok. Im vorliegenden System ist aber klar, daß es sich nur um die soeben angefragte Lok handeln kann, da die Zentrale zwischen Anfrage und Antwort kein anderes Busgerät bedient, welches die Daten verändern könnte. Diese Zwangsläufigkeit muss auch in einem PC-Programm beachtet werden. Ansonsten ist nicht mit einem sicheren Betrieb zu rechnen. Es muß also z.B. vermieden werden, daß in einem Ausgabebereich eines Programmes für eine Lokadresse verschiedene Daten gesendet werden ohne einen Eingabebereich zu beachten, in dem vielleicht erkennbar wäre, daß diese Lok von einem anderen Busgerät übernommen wurde und deren Daten dann verändert sein können.

Das LI101F dient dazu, die Daten des XpressNet auf ein geringfügig anderes, PC-gängigeres Format umzusetzen. Es wird dabei nicht auf den Inhalt der Bytes geachtet. Somit wird der Bus transparent dem PC-Programm zugänglich gemacht.

Das LI100 wird bisher mit 9600 baud betrieben, für das LI101F sind verschiedene Baudraten von 19200 bis 57600 baud per Setup-Software einstellbar. Die Baudrateneinstellung ist unkritisch, da die Gleisausgabe als hauptsächlich durchsatzbestimmender Systemteil im Mittel nur eine Übertragungsrate von etwa 4500 baud erreicht. Es kann durchaus sinnvoll sein, nicht die höchstmögliche Übertragungsrate zwischen PC und LI101F zu wählen, damit die zeitlichen Anforderungen an das Programm nicht zu streng werden.

Jeder Befehl besteht aus einem Header-Byte, einem oder mehreren Datenbytes und einem Prüfbyte. Das Headerbyte besteht aus zwei Teilen zu je 4 Bit (Nibble). Im oberen Nibble ist die auszuführende Funktion enthalten, im unteren Nibble die Anzahl der Datenbytes in diesem Befehl. Hierbei werden das Headerbyte selbst und das Prüfbyte nicht mitgezählt. Es können also bis zu 15 Datenbytes übertragen werden.

Die Datenbytes vervollständigen den mit dem Header begonnenen Befehl. Die Bedeutung ist in der folgenden Beschreibung nachzulesen.

Den Abschluß der Datenübertragung bildet das Prüfbyte. Es ist eine X-Or-Verknüpfung aller vorhergehenden Bytes inclusive des Headerbytes. Beim Empfang muß geprüft werden, ob die X-Or-Verknüpfung aller empfangenen Bytes inclusive des X-Or-Bytes aufgeht, d.h. 0 ergibt. Wenn nicht, ist das ein Kennzeichen dafür, daß bei der Übertragung ein Fehler auftrat.

Um einen Befehl sicher vom PC an das LI101F zu senden, d.h. den Befehl vollständig in den PC-Sendepuffer zu schreiben, ist es sinnvoll, vor dem Schreiben der Befehlesbytes zu prüfen, ob das CTS-Signal aktiv ist und genügend Platz im Sendepuffer vorhanden ist. Um beim Empfang niemals einen Puffer-Überlauf zu

erhalten, muß zu jeder Zeit dafür gesorgt werden, daß empfangene Bytes des Schnittstellenbausteines abgeholt und z.B. in einen Software-Fifo übertragen werden. Der eigentliche Empfangspuffer des UART-Bausteines ist nur wenige Byte groß. Ein Überlauf entsteht insbesondere dann schnell, wenn ungefragte Informationen eintreffen, die aufgrund der realisierten PC-Programmstruktur nicht abgeholt werden.

Für die Erarbeitung eines PC-Programmkonzeptes ist von folgenden Gegebenheiten auszugehen:

Es gibt Befehle, die eine Zentralenantwort zur Folge haben (z.B. Anfrage Zentralenstatus).

Es gibt Befehle, die keine Zentralenantwort zur Folge haben (z.B. Lok-Fahrbehl). Diese werden vom LI101F mit einer Befehlsbestätigung versehen (1/4/5)

Es gibt Zentralenreaktionen, denen kein Befehl des PC vorausgegangen ist (weil ein anderes Gerät dies ausgelöst hat). Beispiel: Broadcast „Alles Aus“ oder Information „Lok besetzt“.

Es kann jedoch auch bei jedem Befehl, der an das LI101F gesendet wird, eine Antwort zurückkommen, falls der Befehl nicht akzeptiert oder empfangen werden konnte. In diesem Fall wird anstelle der Bestätigung 1/4/5 des LI101F bzw. der erwarteten Antwort (z.B. Lokinfos) eben die zugehörige Fehlerantwort (z.B. „Zentrale busy“ oder „Übertragungsfehler“) gesendet.

Eine weitere Besonderheit besteht in der Generierung der Befehlsbestätigung (1/4/5) des LI101F. Diese wird immer dann erzeugt, wenn der PC-Befehl den das LI101F an die Zentrale gesendet hat, keine Antwort im Empfangspuffer des LI101F ergab, sondern das LI101F feststellt, daß die Zentrale das nächste Busgerät adressiert. Wird aber direkt nachdem das LI101F einen Befehl an die Zentrale gesendet hat, der keine Antwort zur Folge hat, durch ein Ereignis auf der Anlage ein Broadcast, z.B. Rückmeldung, ausgelöst, unterbleibt die Befehlsbestätigung, da das LI101F nicht feststellen kann, ob dieser Broadcast die Reaktion auf den vorangegangenen Befehl ist oder nicht.. Diese Struktur ist grundsätzlich immer zu beachten. Es ist bei dem vorliegenden System nicht möglich, sich darauf zu beschränken, einen Befehl zu senden und einzig auf die damit verbundene Antwort zu warten. Gegebenenfalls ist ein Befehl zu wiederholen. Ein Befehl gilt als erfolgreich gesendet, wenn die Quittierung des LI101F dazu eintrifft, die erwartete Antwort oder eine Fehlermeldung.

Aufgrund der Systemstruktur kann aus Sicht des PC-Programmes folgende Situation entstehen, die im Programm durch eine besondere Behandlung (z.B. „Ausortieren“) von Broadcasts in der Empfangsroutine abgefangen werden muß:

PC sendet Befehl an LZ100:	33 / 33 / 00	(Anfrage Softwareversion)
PC erhält folgende Antworten:	66 / 05 / 74 / 13	(Rückmelde-Broadcast)
	66 / 05 / 75 / 12	(Rückmelde-Broadcast)
	99 / 33 / 53 / 0 / 119	(gewünschte Antwort)

Diese Situation kann dann entstehen, wenn ein hohes Rückmelde-Aufkommen herrscht, oder diese Rückmeldungen zu „unglücklichen“ Zeitpunkten entstehen. In der Zeit vom ersten Byte, welches der PC sendet, bis das LI101F von der Zentrale adressiert wurde, um den kompletten Befehl an diese zu senden, trafen hier sogar zwei Rückmeldungen ein. Dieser Antwortversatz läßt sich aufgrund der stets vorhandenen, nur durch einen Timeout von ca. 250ms begrenzten, Übertragungszeit vom PC zum LI101F, sowie der Zeit, bis das LI101F nach komplettem PC-

Befehlsverlust von der Zentrale adressiert wird, nicht vermeiden. Das LI101F kann einen Befehl erst an die Zentrale weiterreichen, wenn es selbst diesen komplett vom PC empfangen hat und eventuell sogar eine Wartezeit von wenigen ms vergangen ist, bis es von der Zentrale adressiert wird.

1.5 Antworten des LI101F

Die Befehle, die der PC über das LI101F 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 Zentralantwort zur Folge haben (z.B. Fahrbefehl erteilen). Damit aber im PC-Programm eine Zuordnung zwischen gesendeten und empfangenen Daten möglich wird, wird nach einem Befehl des PC an diesen eine Antwort zurückgeschickt.

Folgende Meldungen werden vom LI101F als Antwort an den PC gesendet, falls keine anderen Zentraldaten verfügbar sind:

Headerbyte	Meldung	X-Or	Bedeutung
01	01	00	Fehler zwischen Interface und PC (ca. 250ms Timeout bei Datenübertragung PC an LI)
01	02	03	Fehler zwischen Interface und Zentrale (ca. 250ms Timeout bei Übertragung LI an Zentrale)
01	03	02	unbekannter Fehler (Zentrale adressierte LI101F mit Quittierungsaufforderung)
01	04	05	Befehl ist an Zentrale geschickt oder Zentrale adressiert LI101F wieder
01	05	04	Zentrale adressiert LI101F nicht mehr
01	06	07	Puffer-Überlauf im LI101F

Hierbei bedeutet:

01 / 01 / 00:

Beim ersten Byte, das der PC an das LI101F 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.

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.

01 / 03 / 02:

Kommt es während einer Datenübertragung an die Zentrale zu einem Fehler, so wird das gerade bediente XpressNet-Gerät (LI101F) von dieser noch einmal angesprochen und muß dann eine Quittierung senden. Der Befehl muß vom PC-Programm dann wiederholt werden. Der PC erhält hierüber diese Information. Tritt dies wiederholt auf, so ist zunächst die Verkabelung zu prüfen.

01 / 04 / 05:

Wird immer dann an den PC gesendet, wenn ein Befehl, den das LI101F an die Zentrale geleitet hat, keine Antwort zur Folge hatte (z.B. ein Lok-Fahrbefehl). D.h. wenn nach Senden eines Befehles die Zentrale das Interface wieder adressiert hat. Die Meldung kommt auch als Bestätigung, daß die Zentrale das LI101F 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 LI101F keine Befehlsbestätigung gesendet.

Es ist zu beachten, daß seitens des PC-Programmes (und auch des LI101F) 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.

01 / 05 / 04:

Die Zentrale adressiert angeschlossene Geräte in einem bestimmten Zeitintervall. Passiert das nicht, wird diese Meldung an den PC gesendet. Das CTS-Signal wird vom LI101F dann auch weggelassen. Kehrt die Zentrale dann zur normalen Adressierungsrate zurück, so wird dies an den PC mit dem vorher beschriebenen Befehl bestätigt und der normale Betrieb kann wieder aufgenommen werden.

01 / 06 / 07:

Damit keine Informationen der Zentrale an das LI101F verlorengehen, wird dort ein Puffer unterhalten. Kann dieser nicht geleert werden, so wird diese Meldung an den PC gesendet. Es ist dann davon auszugehen, daß ein oder mehr Datensätze verlorengegangen sind. Gegebenenfalls müssen vom PC aus dann Informationen erneut angefordert werden.

Die hier beschriebenen Antworten unterscheiden sich zum Teil von den bisher verwendeten. Gegebenenfalls sind bestehende PC-Programme diesem Antwortschema anzupassen.

1.6 Versionsnummer des LI101F feststellen

Das Auslesen der Versionsnummer des LI101F ist eine Aktion, die nur zwischen PC und LI101F stattfindet. Der Befehlsaufbau und die zugehörige LI101F-Antwort entsprechen aber trotzdem dem in Kapitel 2 beschriebenen Format.

Befehl zum Auslesen der Versions- und Codenummer:

Headerbyte X-Or-Byte

Binär :	1111 0000	1111 0000
Hex :	0xF0	0xF0
Dez :	240	240

Antwort des LI101F:

	Headerbyte	Daten 1	Daten 2	X-Or-Byte
Binär :	0000 0010	VVVV VVVV	CCCC CCCC	X-Or-Byte
Hex :	0x02	VV	CC	X-Or-Byte
Dez :	2	VV	CC	X-Or-Byte

Beschreibung:

VV gibt die Versionsnummer des LI101F hexadezimal in BCD-Darstellung an. CC gibt die Codenummer des LI101F hexadezimal in BCD-Darstellung an. Beispiel:

Antwort = 0x02 0x10 0x01 0x13

Versionsnummer 1.0, Codenummer 01

Besonderheiten:

Keine.

1.7 Geräteadresse des LI101F feststellen und ändern

Dies ist eine Aktion, die nur zwischen PC und LI101F stattfindet. Der Befehlsaufbau und die zugehörige LI101F-Antwort entsprechen aber trotzdem dem in Kapitel 2 beschriebenen Format.

Befehl zur Übernahme einer neuen Geräteadresse:

	Headerbyte	Daten 1	Daten 2	X-Or-Byte
Binär :	1111 0010	0000 0001	ADR	X-Or-Byte
Hex :	0xF2	0x01	ADR	X-Or-Byte
Dez :	242	1	ADR	X-Or-Byte

Antwort des LI101F:

	Headerbyte	Daten 1	Daten 2	X-Or-Byte
Binär :	1111 0010	0000 0001	ADR	X-Or-Byte
Hex :	0xF2	0x01	ADR	X-Or-Byte
Dez :	242	1	ADR	X-Or-Byte

Beschreibung:

ADR gibt die XpressNet-Geräteadresse an, die das LI101F verwenden soll. Der erlaubte Bereich liegt zwischen 1 und 31 dez. In der Antwort des LI101F 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 LI101F die Geräteadresse nicht im Bereich von 1 bis 31 angegeben, so antwortet das LI101F mit seiner aktuell eingestellten Adresse. Damit kann die Adresse festgestellt werden, ohne sie zu ändern.

1.8 Baudrate des LI101F feststellen und ändern

Dies ist eine Aktion, die nur zwischen PC und LI101F stattfindet. Der Befehlsaufbau und die zugehörige LI101F-Antwort entsprechen aber trotzdem dem in Kapitel 2 beschriebenen Format.

Befehl zum Setzen einer neuen Übertragungs-Baudrate:

	Headerbyte	Daten 1	Daten 2	X-Or-Byte
Binär :	1111 0010	0000 0010	BAUD	X-Or-Byte
Hex :	0xF2	0x02	BAUD	X-Or-Byte
Dez :	242	2	BAUD	X-Or-Byte

Antwort des LI101F:

	Headerbyte	Daten 1	Daten 2	X-Or-Byte
Binär :	1111 0010	0000 0010	BAUD	X-Or-Byte
Hex :	0xF2	0x02	BAUD	X-Or-Byte
Dez :	242	2	BAUD	X-Or-Byte

Beschreibung:

BAUD gibt codiert die neue Übertragungs-Baudrate der seriellen Schnittstelle an, die das LI101F verwenden soll. Die Codierung lautet wie folgt:

BAUD = 1 19200 baud (Standardeinstellung)

BAUD = 2 38400 baud

BAUD = 3 57600 baud

BAUD = 4 115200 baud

Es ist in der Gerätebeschreibung des LI101F nachzusehen, ob eventuell eine hierzu eingeschränkte oder erweiterte Codierung im Gerät verwendet wird.

Besonderheiten:

Liegt die Baudratencodierung im Bereich 1 bis 4, so sendet das LI101F die Antwort noch mit der alten Baudrate und schaltet erst danach auf die neue um. Ein Wert außerhalb dieses Bereiches ist nicht definiert und führt zu nicht vorhersagbaren Ergebnissen. Siehe hierzu die dem Gerät beiliegende Beschreibung.

1.9 Break-Signal und Reset des LI101F

Wenn der Fall eintritt, daß PC und LI101F mit verschiedenen Baudraten arbeiten, so führt dies zu Kommunikationsschwierigkeiten. Um diese Situation definiert beenden zu können, wird das Senden eines Break-Signales durch den PC benutzt. Das Break-Signal muß für eine Dauer von mindestens 200 Millisekunden gesendet werden. Es ist unabhängig von der Baudrate.

Das LI101F empfängt in diesem Fall den Bytewert 0 mit einem Framing-Error und nach einer weiteren Prüfzeit wird dann ein Gerätereset durchgeführt. Zusätzlich wird die Übertragungsbaudrate im LI101F auf 19200baud eingestellt, so daß das PC-Programm direkt im Anschluß an das Break-Signal sicher eine Kommunikation beginnen kann.

Empfohlene Vorgehensweise bei Start des PC-Programmes:

- Break-Signal senden (damit wird das LI101F definiert auf 19200baud gesetzt)

- LI101F Versionsnummer und Adresse auslesen (mit 19200baud)
- Baudrate umschalten auf die gewünschte Geschwindigkeit (falls ungleich 19200), die
- Antwort des LI101F abwarten, sie erfolgt noch mit der alten Baudrate
- Zentralenstatus und -version anfragen etc. (mit neuer Baudrate)

2 Datenverkehr 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. 2.7 , Befehl nicht vorhanden).

2.1 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 BC „Alles An“

Format:

	Headerbyte	Daten 1	X-Or-Byte
Binär :	0110 0001	0000 0001	0110 0000
Hex :	0x61	0x01	0x60
Dez :	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 auf-gehoben 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.

2.1.2 BC „Alles Aus“ (Notaus)

Format:

	Headerbyte	Daten 1	X-Or-Byte
Binär :	0110 0001	0000 0000	0110 0001
Hex :	0x61	0x00	0x61
Dez :	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.

2.1.3 BC „Alle Loks Aus“ (Nothalt)

Format:

	Headerbyte	Daten 1	X-Or-Byte
Binär :	1000 0001	0000 0000	0110 0001
Hex :	0x81	0x00	0x81
Dez :	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.

2.1.4 BC „Programmiermode“

Format:

	Headerbyte	Daten 1	X-Or-Byte
Binär :	0110 0001	0000 0010	0110 0011
Hex :	0x61	0x02	0x63
Dez :	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 LI101F kommunizieren. Hat ein anderes Gerät den Programmiermode ausgelöst, so kann kein Befehl an das LI101F 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.

2.1.5 BC „Rückmeldung“

Format:

	Headerbyte	Daten 1	Daten 2	Daten 3	Daten 4	usw.	X-Or-Byte
Binär :	0100 NNNN	ADR_1	DAT_1	ADR_2	DAT2	usw.	X-Or-Byte
Hex :	0x40 + N						X-Or-Byte
Dez :	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.

2.2 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.

2.2.1 Programmierinfo „Kurzschluß“

Format:

	Headerbyte	Daten 1	X-Or-Byte
Binär :	0110 0001	0001 0010	X-Or-Byte
Hex :	0x61	0x12	X-Or-Byte
Dez :	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.

2.2.2 Programmierinfo „Daten nicht gefunden“

Format:

	Headerbyte	Daten 1	X-Or-Byte
Binär :	0110 0001	0001 0011	X-Or-Byte
Hex :	0x61	0x13	X-Or-Byte
Dez :	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.

2.2.3 Programmierinfo „Zentrale Busy“

Format:

	Headerbyte	Daten 1	X-Or-Byte
Binär :	0110 0001	0001 1111	X-Or-Byte
Hex :	0x61	0x1f	X-Or-Byte
Dez :	97	31	X-Or-Byte

Beschreibung:

Dieser Befehl wird bis einschließlich Zentralen-Version 3.5 noch nicht verwendet.

Besonderheiten:

Keine.

2.2.4 Programmierinfo „Zentrale Bereit“

Format:

	Headerbyte	Daten 1	X-Or-Byte
Binär :	0110 0001	0001 0001	X-Or-Byte
Hex :	0x61	0x11	X-Or-Byte
Dez :	97	17	X-Or-Byte

Beschreibung:

Dieser Befehl wird bis einschließlich Zentralen-Version 3.5 noch nicht verwendet.

Besonderheiten:

Keine.

2.2.5 Programmierinfo „Daten 3-Byte-Format“

Format:

	Headerbyte	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	0110 0011	0001 0000	EEEE EEEE	DDDD DDDD	X-Or-Byte
Hex :	0x63	0x10	E	D	X-Or-Byte
Dez :	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.

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

Format:

	Headerbyte	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	0110 0011	0001 0100	CCCC CCCC	DDDD DDDD	X-Or-Byte
Hex :	0x63	0x14	C	DAT	X-Or-Byte
Dez :	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 2.2.5.

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

Format:

	Headerbyte	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	0110 0011	0001 0101	CCCC CCCC	DDDD DDDD	X-Or-Byte
Hex :	0x63	0x15	C	DAT	X-Or-Byte
Dez :	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 2.2.6.

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

Format:

	Headerbyte	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	0110 0011	0001 0110	CCCC CCCC	DDDD DDDD	X-Or-Byte
Hex :	0x63	0x16	C	DAT	X-Or-Byte
Dez :	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 <=> CV – Adressen:

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

Besonderheiten:

Siehe 2.2.6.

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

Format:

	Headerbyte	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	0110 0011	0001 0111	CCCC CCCC	DDDD DDDD	X-Or-Byte
Hex :	0x63	0x17	C	DAT	X-Or-Byte
Dez :	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 2.2.6.

2.3 Softwareversion Zentrale

Bis einschließlich Zentralenversion 2.3 wird auf die Anfrage nach der Zentralen-Software-version ein Byte zurückgesendet. Ab Version 3.0 wird zusätzlich ein zweites Byte gesendet, welches die Zentralen-Kennung enthält. Damit kann ein Slave entscheiden, z.B. welche Fahrbefehle die Zentrale versteht, ob Mehrfachtraktionen möglich sind etc. Die Anforderung des Slave nach der Zentralen-Software ist in beiden Fällen der Antwort identisch.

2.3.1 Softwareversion bis LZ-Version 2.3

Format:

	Headerbyte	Daten 1	Daten 2	X-Or-Byte
Binär :	0110 0010	0010 0001	OOOO UUUU	X-Or-Byte
Hex :	0x62	0x21	O + U	X-Or-Byte
Dez :	98	33	O + U	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 = 0010 0011 = 0x23 : Version 2.3

Besonderheiten:

Keine.

2.3.2 Softwareversion ab LZ-Version 3.0**Format:**

	Headerbyte	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	0110 0011	0010 0001	OOOO UUUU	IIII IIII	X-Or-Byte
Hex :	0x63	0x21	O + U	ID	X-Or-Byte
Dez :	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 0010 = 0x32 : Version 3.2.

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)
- ID = 0x03: Control Plus – Zentrale

Besonderheiten:

Keine.

2.4 Status Zentrale**Format:**

	Headerbyte	Daten 1	Daten 2	X-Or-Byte
Binär :	0110 0010	0010 0010	SSSS SSSS	X-Or-Byte
Hex :	0x62	0x22	S	X-Or-Byte
Dez :	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 Gleisausgabe. 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.

2.5 Übertragungsfehler

Format:

	Headerbyte	Daten 1	X-Or-Byte
Binär :	0110 0001	1000 0000	1110 0001
Hex :	0x61	0x80	0xE1
Dez :	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 Send- und Empfangs- FiFo). In aller Regel hat ein Übertragungsfehler noch weitere Fehlermeldungen (Timeout PC-LI101F) zur Folge. Auch ein Befehl, der normalerweise keine Antwort zur Folge hat und mit 01/04/05 quittiert würde, kann stattdessen diese Antwort liefern.

2.6 Zentrale Busy

Format:

	Headerbyte	Daten 1	X-Or-Byte
Binär :	0110 0001	1000 0001	X-Or-Byte
Hex :	0x61	0x81	X-Or-Byte
Dez :	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.

2.7 Befehl in Zentrale nicht vorhanden

Format:

	Headerbyte	Daten 1	X-Or-Byte
Binär :	0110 0001	1000 0010	X-Or-Byte
Hex :	0x61	0x82	X-Or-Byte
Dez :	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.

2.8 Schaltinformation

Format:

	Headerbyte	Daten 1	Daten 2	X-Or-Byte
Binär :	0100 0010	AAAA AAAA	ITTN ZZZZ	X-Or-Byte
Hex :	0x42	ADR	ITNZ	X-Or-Byte
Dez :	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.

	<p>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.</p> <p>Für einen Rückmeldebaustein kann die Adresse im Bereich 0..127 liegen (7 Bit Adresse). Dies ist direkt die Adresse des Bausteins.</p> <p>Die Adressinformation wird so in dieser Info zurückgeschickt, wie sie in der Anfrage nach Schaltinformationen der Zentrale gesendet wurde.</p>
Daten 2: I	<p>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.</p>
Daten 2: TT	<p>Diese beiden Bits stellen die Kennung der angefragten Adresse dar. Es gilt:</p> <p>TT = 0 0 : Adresse ist Schaltempfänger ohne Rückmeldung</p> <p>TT = 0 1 : Adresse ist Schaltempfänger mit Rückmeldung</p> <p>TT = 1 0 : Adresse ist ein Rückmeldebaustein</p> <p>TT = 1 1 : reserviert für zukünftige Anwendungen</p>
Daten 2: N	<p>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 Rückmeldebausteins zu erfassen, ist eine Anfrage an das untere Nibble und eine zweite Anfrage an das obere Nibble der Rückmeldeadresse zu richten.</p> <p>Achtung: Das Nibble-Bit stimmt nur dann, wenn die Weiche schon einmal geschaltet wurde!</p>

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.

2.9 Lokinformationen ab Zentralen-Version 3.0

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 2.2). 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.

2.9.1 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:

	Headerbyte	Kennung	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	1110 0100	0000 BFFF	RVVV VVVV	000F FFFF	FFFF FFFF	X-Or-Byte
Hex :	0xE4	Kennung	Speed	F0	F1	X-Or-Byte
Dez :	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.

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

Format:

	Headerbyte	Kennung	Daten 1	Daten 2	X-Or-Byte
Binär :	1110 0011	0101 0010	FFFF FFFF	FFFF FFFF	X-Or-Byte
Hex :	0xE3	0x52			X-Or-Byte
Dez :	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.

2.10.1 Lokinformation Lok befindet sich in einer Mehrfachtraktion

Format:

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

Beschreibung:

Kennung: Die Bits 3 bis 0 sind wie unter 2.9.1 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 2.9.1 beschrieben codiert. Die Geschwindigkeit gibt die Geschwindigkeit der angefragten Lok an!

F0, F1: Codiert wie unter 2.9.1 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.

2.10.2 Lokinformation Lokadresse ist die Basisadresse einer Mehrfachtraktion

Format:

	Headerbyte	Kennung	Daten 1	X-Or-Byte
Binär :	1110 0010	0010 BFFF	RVVV VVVV	X-Or-Byte
Hex :	0xE2	Kennung	Speed	X-Or-Byte
Dez :	226	Kennung	Speed	X-Or-Byte

Beschreibung:

Kennung: Die Bits 3 bis 0 sind wie unter 2.9.1 beschrieben codiert. Die Fahrstufenzahl gibt die Fahrstufenzahl der Mehrfachtraktion an.

Speed: Das Geschwindigkeits-Byte ist wie unter 2.9.1 beschrieben codiert. Die Geschwindigkeit gibt die Geschwindigkeit der MTR an.

Besonderheiten:

An die Basisadresse einer MTR sollen keine Funktionsbefehle gesendet werden.

2.10.3 Lokinformation Lok befindet sich in einer Doppeltraktion

Format:

	Headerbyt e	Kennung	Daten 1	Daten 2	Daten 3	Daten 4	Daten 5	X-Or-Byte
Binär :	1110 0110	0110 BFFF	RVVV VVVV	000F FFFF	FFFF FFFF	Adr High	Adr Low	X-Or-Byte
Hex :	0xE6	Kennung	Speed	F0	F1	AH	AL	X-Or-Byte
Dez :	230	Kennung	Speed	F0	F1	AH	AL	X-Or-Byte

Beschreibung:

Kennung: Die Bits 3 bis 0 sind wie unter 2.9.1 beschrieben codiert.

Speed: Das Geschwindigkeits-Byte ist wie unter 2.9.1 beschrieben codiert.

F0, F1: Codiert wie unter 2.9.1 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 3.27).

2.11 Lok besetzt ab Zentralen-Version 3.0

Format:

	Headerbyte	Kennung	Daten 1	Daten 2	X-Or-Byte
Binär :	1110 0011	0100 0000	Adresse High	Adresse Low	X-Or-Byte
Hex :	0xE3	0x40	AH	AL	X-Or-Byte
Dez :	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.

2.12 Funktionsstatus der angefragten Lok ab Zentralen-Version 3.0

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:

	Headerbyte	Kennung	Daten 1	Daten 2	X-Or-Byte
Binär :	1110 0011	0101 0000	000S SSSS	SSSS SSSS	X-Or-Byte
Hex :	0xE3	0x50	S0	S1	X-Or-Byte
Dez :	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.

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

	Rufbyte	Headerbyte	Kennung	Daten 1	Daten 2	X-Or-Byte
Binär :	P11A AAAA	1110 0100	0101 0001	SSSS SSSS	SSSS SSSS	X-Or-Byte
Hex :	P+0x60+GA	0xE4	0x51			X-Or-Byte
Dez :	P+96+GA	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.

2.14 Lokinformation bei Adress-Suchanfragen ab Zentralen-Version 3.0

Format:

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

Beschreibung:

Diese Antwort wird gesendet, wenn das XpressNet-Gerät eine der Suchanfragen aus 3.30 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 2.11 beschrieben.

Besonderheiten:

Keine.

2.15 Fehlermeldungen ab Zentralen-Version 3.0

Ab Zentralen-Version 3.0 sind verschiedene Fehler in einer Fehlerantwort zusammengefasst. Der Zusammenhang ergibt sich aus dem vorausgehend erteilten Befehl an die Zentrale, auf den sich die Fehlermeldung bezieht.

Format:

	Headerbyte	Kennung	X-Or-Byte
Binär :	1110 0001	1000 FFFF	X-Or-Byte
Hex :	0xE1	0x80 + F	X-Or-Byte
Dez :	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.

3 Datenverkehr PC an Zentrale

Nachdem das LI101F durch das CTS-Signal dem PC die Sendefreigabe erteilt hat (und nur dann), kann dieser eine Kommunikation beginnen und z.B. Daten anfordern, Loks steuern oder Programmieraktionen durchführen. Je nach gewünschter Aktion gibt die Zentrale bzw. das LI101F dem PC dann die entsprechende Antwort. Es ist wichtig, daß vom PC aus der Hardware-Handshake der seriellen Schnittstelle benutzt wird, denn nur dadurch kann sichergestellt werden, daß der PC nur dann sendet, wenn das LI101F dies erlaubt.

3.1 Alles An

Format:

	Headerbyte	Daten 1	X-Or-Byte
Binär :	0010 0001	1000 0001	1010 0000
Hex :	0x21	0x81	0xA0
Dez :	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.

3.2 Alles Aus (Notaus)

Format:

	Headerbyte	Daten 1	X-Or-Byte
Binär :	0010 0001	1000 0000	1010 0001
Hex :	0x21	0x80	0xA1
Dez :	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.

3.3 Alle Loks anhalten (Nothalt)

Format:

	Headerbyte	X-Or-Byte
Binär :	1000 0000	1000 0000
Hex :	0x80	0x80
Dez :	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.

3.4 Eine Lok anhalten (Nothalt für eine Lok)

Format:

	Headerbyte	Daten 1	Daten 2	X-Or-Byte
Binär :	1001 0010	Adresse High	Adresse Low	X-Or-Byte
Hex :	0x92	AH	AL	X-Or-Byte
Dez :	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 2.11 angegeben.

3.5 Leseanfrage Programmieren 3-Byte-Format (Registermode)

Format:

	Headerbyte	Daten 1	Daten 2	X-Or-Byte
Binär :	0010 0010	0001 0001	0000 RRRR	X-Or-Byte
Hex :	0x22	0x11	R	X-Or-Byte
Dez :	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 weiter Befehle an die Zentrale senden, das den Programmiermode ausgelöst hat.

3.6 Leseanfrage Programmieren 4-Byte-Format (CV-Mode, CV 1-256)

Format:

	Headerbyte	Daten 1	Daten 2	X-Or-Byte
Binär :	0010 0010	0001 0101	CCCC CCCC	X-Or-Byte
Hex :	0x22	0x15	CV	X-Or-Byte
Dez :	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 3.7 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 3.7 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.

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

Format:

	Headerbyte	Daten 1	Daten 2	X-Or-Byte
Binär :	0010 0010	0001 1000	CCCC CCCC	X-Or-Byte
Hex :	0x22	0x18	CV	X-Or-Byte
Dez :	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 3.6 beschrieben.

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

Format:

	Headerbyte	Daten 1	Daten 2	X-Or-Byte
Binär :	0010 0010	0001 1001	CCCC CCCC	X-Or-Byte
Hex :	0x22	0x19	CV	X-Or-Byte
Dez :	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 3.6 beschrieben.

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

Format:

	Headerbyte	Daten 1	Daten 2	X-Or-Byte
Binär :	0010 0010	0001 1010	CCCC CCCC	X-Or-Byte
Hex :	0x22	0x1A	CV	X-Or-Byte
Dez :	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 3.6 beschrieben.

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

Format:

	Headerbyte	Daten 1	Daten 2	X-Or-Byte
Binär :	0010 0010	0001 1011	CCCC CCCC	X-Or-Byte
Hex :	0x22	0x1B	CV	X-Or-Byte
Dez :	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 3.6 beschrieben.

3.11 Leseanfrage Programmieren 3-Byte-Format (Pagemode)

Format:

	Headerbyte	Daten 1	Daten 2	X-Or-Byte
Binär :	0010 0010	0001 0100	CCCC CCCC	X-Or-Byte
Hex :	0x22	0x14	CV	X-Or-Byte
Dez :	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 „Programmiererergebnis 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.

3.12 Programmiererergebnis anfordern

Format:

	Headerbyte	Daten 1	X-Or-Byte
Binär :	0010 0001	0001 0000	0011 0001
Hex :	0x21	0x10	0x31
Dez :	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 2.2 beschriebenen Möglichkeiten.

Besonderheiten:

Keine.

3.13 Schreibbefehl Programmieren 3-Byte-Format (Register-Mode)

Format:

	Headerbyte	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	0010 0011	0001 0010	0000 RRRR	Daten	X-Or-Byte
Hex :	0x23	0x12	R	Daten	X-Or-Byte
Dez :	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.

3.14 Schreibbefehl Programmieren 4-Byte-Format (CV-Mode, CV 1-256)

Format:

	Headerbyte	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	0010 0011	0001 0110	CCCC CCCC	Daten	X-Or-Byte
Hex :	0x23	0x16	CV	Daten	X-Or-Byte
Dez :	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 256, wobei CV256 als 0x00 gesendet werden muß.

Besonderheiten:

Dieser Befehl existiert zusätzlich zum Befehl wie in 3.15 beschrieben. Wird eine Zentrale ab Version 3.6 verwendet, so wird die CV1024 statt der CV256 beschrieben. Daher empfehlen wir die Verwendung des in 3.15 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.

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

Format:

	Headerbyte	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	0010 0011	0001 1100	CCCC CCCC	Daten	X-Or-Byte
Hex :	0x23	0x1C	CV	Daten	X-Or-Byte
Dez :	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.

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

Format:

	Headerbyte	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	0010 0011	0001 1101	CCCC CCCC	Daten	X-Or-Byte
Hex :	0x23	0x1D	CV	Daten	X-Or-Byte
Dez :	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 3.15.

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

Format:

	Headerbyte	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	0010 0011	0001 1110	CCCC CCCC	Daten	X-Or-Byte
Hex :	0x23	0x1E	CV	Daten	X-Or-Byte
Dez :	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 3.15.

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

Format:

	Headerbyte	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	0010 0011	0001 1111	CCCC CCCC	Daten	X-Or-Byte
Hex :	0x23	0x1F	CV	Daten	X-Or-Byte
Dez :	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 3.15.

3.19 Schreibbefehl Programmieren 3-Byte-Format (Page-Mode)

Format:

	Headerbyte	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	0010 0011	0001 0111	CCCC CCCC	Daten	X-Or-Byte
Hex :	0x23	0x17	CV	Daten	X-Or-Byte
Dez :	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.

3.20 Softwareversion der Zentrale anfordern

Format:

	Headerbyte	Daten 1	X-Or-Byte
Binär :	0010 0001	0010 0001	0000 0000
Hex :	0x21	0x21	0x00
Dez :	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 2.3 beschrieben möglich.

Besonderheiten:

Keine.

3.21 Status der Zentrale anfordern

Format:

	Headerbyte	Daten 1	X-Or-Byte
Binär :	0010 0001	0010 0100	0000 0101
Hex :	0x21	0x24	0x05
Dez :	33	36	5

Beschreibung:

Die Anfrage nach dem Zentralenstatus ergibt die unter 2.4 beschriebene Antwort.

Besonderheiten:

Keine.

3.22 Zentralen-Startmode setzen

Format:

	Headerbyte	Daten 1	Daten 2	X-Or-Byte
Binär :	0010 0010	0010 0010	0000 0M00	X-Or-Byte
Hex :	0x22	0x22	M	X-Or-Byte
Dez :	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.

3.23 Schaltinformationen anfordern

Format:

	Headerbyte	Daten 1	Daten 2	X-Or-Byte
Binär :	0100 0010	AAAA AAAA	1000 000N	X-Or-Byte
Hex :	0x42	Adresse	0x80 + N	X-Or-Byte
Dez :	66	Adresse	128 + N	X-Or-Byte

Beschreibung:

Aufgrund dieses Kommandos sendet die Zentrale die unter 2.8 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: $620 \bmod 4 = 155$, dh. Weiche 620 liegt in Weichengruppe 155. In Weichengruppe 155 liegen die Weichen 620, 621, 622, 623. Damit ist das Nibblebit 1 (oberes Nibble).

3.24 Schaltbefehl

Format:

	Headerbyte	Daten 1	Daten 2	X-Or-Byte
Binär :	0101 0010	AAAA AAAA	1000 DBBD	X-Or-Byte
Hex :	0x52	Adresse	0x80 + DBBD	X-Or-Byte
Dez :	82	Adresse	128 + DBBD	X-Or-Byte

Beschreibung:

Schaltbefehle können nur an Schaltempfänger erteilt werden. Die Adresse ist damit die Weichenummer / 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 3.23.

3.25 Lokinformationen anfordern

Format:

	Headerbyte	Kennung	Daten 1	Daten 2	X-Or-Byte
Binär :	1110 0011	0000 0000	Adresse High	Adresse Low	X-Or-Byte
Hex :	0xE3	0x00	AH	AL	X-Or-Byte
Dez :	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 2.11 angegeben.

Die möglichen Antworten sind unter 2.9 beschrieben.

Besonderheiten:

Keine.

3.25.1 Funktionsstatus anfordern

Format:

	Headerbyte	Kennung	Daten 1	Daten 2	X-Or-Byte
Binär :	1110 0011	0000 0111	Adresse High	Adresse Low	X-Or-Byte
Hex :	0xE3	0x07	AH	AL	X-Or-Byte
Dez :	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 2.11 angegeben.

Die möglichen Antworten sind unter 2.9 beschrieben.

Besonderheiten:

Keine.

3.25.2 Funktionsstatus anfordern F13 – F28 (neu ab Zentralen-Version 3.6)**Format:**

	Headerbyte	Kennung	Daten 1	Daten 2	X-Or-Byte
Binär :	1110 0011	0000 1000	Adresse High	Adresse Low	X-Or-Byte
Hex :	0xE3	0x08	AH	AL	X-Or-Byte
Dez :	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 2.11 angegeben.

Die möglichen Antworten sind unter 2.13 beschrieben.

Besonderheiten:

Keine.

3.25.3 Funktionszustand anfordern F13 – F28 (neu ab Zentralen-Version 3.6)**Format:**

	Headerbyte	Kennung	Daten 1	Daten 2	X-Or-Byte
Binär :	1110 0011	0000 1001	Adresse High	Adresse Low	X-Or-Byte
Hex :	0xE3	0x09	AH	AL	X-Or-Byte
Dez :	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 2.11 angegeben.

Die möglichen Antworten sind unter 2.10 beschrieben.

Besonderheiten:

Keine.

3.26 Lok steuern

3.26.1 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 2.9 beschrieben codiert. Die Geschwindigkeit für 128 Fahrstufen wie unter 2.9.1.

Fahrbefehl 14 Fahrstufen:

Format:

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

Fahrbefehl 27 Fahrstufen:

Format:

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

Fahrbefehl 28 Fahrstufen:

Format:

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

Fahrbefehl 128 Fahrstufen:

Format:

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

Beschreibung:

Der Fahrbefehl 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 2.11 angegeben.

Besonderheiten:

Keine.

3.26.2 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) und der 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.

Funktionsbefehl Gruppe 1:

Format:

	Headerbyte	Kennung	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	1110 0100	0010 0000	Adresse High	Adresse Low	000F FFFF	X-Or-Byte
Hex :	0xE4	0x20	AH	AL	Gruppe 1	X-Or-Byte
Dez :	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:

	Headerbyte	Kennung	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	1110 0100	0010 0001	Adresse High	Adresse Low	0000 FFFF	X-Or-Byte
Hex :	0xE4	0x21	AH	AL	Gruppe 2	X-Or-Byte
Dez :	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:

	Headerbyte	Kennung	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	1110 0100	0010 0010	Adresse High	Adresse Low	0000 FFFF	X-Or-Byte

Hex :	0xE4	0x22	AH	AL	Gruppe 3	X-Or-Byte
Dez :	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:

	Headerbyte	Kennung	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	1110 0100	0010 0011	Adresse High	Adresse Low	0000 FFFF	X-Or-Byte
Hex :	0xE4	0x23	AH	AL	Gruppe 4	X-Or-Byte
Dez :	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:

	Headerbyte	Kennung	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	1110 0100	0010 1000	Adresse High	Adresse Low	0000 FFFF	X-Or-Byte
Hex :	0xE4	0x28	AH	AL	Gruppe 5	X-Or-Byte
Dez :	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:

Es können die Loks 0 bis 9999 angesprochen werden.

Die Lokadresse AH/AL berechnet sich wie unter 2.11 angegeben.

Besonderheiten:

Keine.

3.26.3 [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 z.B. ihre Bedienoberfläche entsprechend gestalten. Diese Funktionalität ist vor allem für Geräusche interessant.

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:

	Headerbyte	Kennung	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	1110 0100	0010 0100	Adresse High	Adresse Low	000S SSSS	X-Or-Byte
Hex :	0xE4	0x24	AH	AL	Gruppe 1	X-Or-Byte
Dez :	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:

	Headerbyte	Kennung	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	1110 0100	0010 0101	Adresse High	Adresse Low	0000 SSSS	X-Or-Byte
Hex :	0xE4	0x25	AH	AL	Gruppe 2	X-Or-Byte
Dez :	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:

	Headerbyte	Kennung	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	1110 0100	0010 0110	Adresse High	Adresse Low	0000 SSSS	X-Or-Byte
Hex :	0xE4	0x26	AH	AL	Gruppe 3	X-Or-Byte
Dez :	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:

	Headerbyte	Kennung	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	1110 0100	0010 0111	Adresse High	Adresse Low	0000 SSSS	X-Or-Byte
Hex :	0xE4	0x27	AH	AL	Gruppe 4	X-Or-Byte
Dez :	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:

	Headerbyte	Kennung	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	1110 0100	0010 1100	Adresse High	Adresse Low	0000 SSSS	X-Or-Byte
Hex :	0xE4	0x2C	AH	AL	Gruppe 5	X-Or-Byte
Dez :	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.

Beschreibung:

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 2.11 angegeben.

Besonderheiten:

Keine.

3.26.4 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:

	Headerbyte	Kennung	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	1110 0101	0010 1111	Adresse High	Adresse Low	Refresh-Modus	X-Or-Byte
Hex :	0xE5	0x2F	AH	AL	RF	X-Or-Byte
Dez :	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 2.11 angegeben.

Besonderheiten:

Keine.

3.27 Doppeltraktionen

3.27.1 Doppeltraktion montieren

Format:

	Headerbyte	Kennung	Daten 1	Daten 2	Daten 3	Daten 4	X-Or-Byte
Binär :	1110 0101	0100 0011	Adr High 1	Adr Low 1	Adr. High 2	Adr Low 2	X-Or-Byte
Hex :	0xE5	0x43	AH1	AL1	AH 2	AL 2	X-Or-Byte
Dez :	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 2.11 angegeben.

Gelingt die Montage nicht, so sendet die Zentrale eine der unter 2.15 beschriebenen Fehlermeldungen.

Besonderheiten:

Der Befehl ersetzt die alten Doppeltraktionsbefehle der Versionen 1.xx und 2.xx, die in späteren Zentralenversionen nicht mehr unterstützt werden.

3.27.2 Doppeltraktion auflösen

Format:

	Headerbyte	Kennung	Daten 1	Daten 2	Daten 3	Daten 4	X-Or-Byte
Binär :	1110 0101	0100 0011	Adr High 1	Adr Low 1	0000 0000	0000 0000	X-Or-Byte
Hex :	0xE5	0x43	AH1	AL1	0x00	0x00	X-Or-Byte
Dez :	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 2.15 beschriebenen Fehlermeldungen.

Besonderheiten:

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

3.28 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).

3.28.1 Programming on Main Byte schreiben

Format:

	Headerbyte	Kennung	Daten 1	Daten 2	Daten 3	Daten 4	Daten 5	X-Or-Byte
Binär :	1110 0110	0011 0000	Adresse High	Adresse Low	1110 11CC	CCCC CCCC	DDDD DDDD	X-Or-Byte
Hex :	0xE6	0x30	AH	AL	0xEC + C	CV	D	X-Or-Byte
Dez :	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 2.11 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 decremientiert.

Besonderheiten:

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

3.28.2 Programming on Main Byte lesen (ab Version 3.6)

Format:

	Headerbyte	Kennung	Daten 1	Daten 2	Daten 3	Daten 4	Daten 5	X-Or-Byte
Binär :	1110 0110	0011 0000	Adresse High	Adresse Low	1110 01CC	CCCC CCCC	DDDD DDDD	X-Or-Byte
Hex :	0xE6	0x30	AH	AL	0xEA + C	CV	D	X-Or-Byte
Dez :	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 2.11 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 decremientiert.

Besonderheiten:

Keine

3.28.3 Programming on Main Bit schreiben

Format:

	Headerbyte	Kennung	Daten 1	Daten 2	Daten 3	Daten 4	Daten 5	X-Or-Byte
Binär :	1110 0110	0011 0000	Adresse High	Adresse Low	1110 10CC	CCCC CCCC	1111 WBBB	X-Or-Byte
Hex :	0xE6	0x30	AH	AL	0xE8 + C	CV	WB	X-Or-Byte
Dez :	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 2.11 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.

3.29 Mehrfachtraktionen

3.29.1 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:

	Headerbyte	Kennung	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	1110 0100	0100 000R	Adresse High	Adresse Low	MTR	X-Or-Byte
Hex :	0xE4	0x40 + R	AH	AL	MTR	X-Or-Byte
Dez :	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 2.11 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.

3.29.2 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:

	Headerbyte	Kennung	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	1110 0100	0100 0010	Adresse High	Adresse Low	MTR	X-Or-Byte
Hex :	0xE4	0x42	AH	AL	MTR	X-Or-Byte
Dez :	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 2.11 angegeben.

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

Mehrfachtraktion gefahren werden kann.

Besonderheiten:

Keine.

3.30 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.

3.30.1 Adressanfrage Mitglied einer Mehrfachtraktion

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

Format:

	Headerbyte	Kennung	Daten 1	Daten 2	Daten 3	X-Or-Byte
Binär :	1110 0100	0000 00RR	MTR	Adresse High	Adresse Low	X-Or-Byte
Hex :	0xE4	0x01 + R	MTR	AH	AL	X-Or-Byte
Dez :	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 2.11 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 2.14 beschrieben gesendet.

Besonderheiten:

Keine.

3.30.2 Adressanfrage Mehrfachtraktion

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

Format:

Headerbyt Kennung Daten 1 X-Or-Byte
e

Binär :	1110 0010	0000 0RRR	MTR	X-Or-Byte
Hex :	0xE2	0x03 + R	MTR	X-Or-Byte
Dez :	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 2.14 beschrieben gesendet.

Besonderheiten:

Keine.

3.30.3 Adressanfrage Lok in Zentralenstack

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

Format:

	Headerbyte	Kennung	Daten 1	Daten 2	X-Or-Byte
Binär :	1110 0011	0000 01RR	Adresse High	Adresse Low	X-Or-Byte
Hex :	0xE3	0x05 + R	AH	AL	X-Or-Byte
Dez :	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 2.11 angegeben.

Das Ergebnis der Suche wird an das Gerät in der Antwort wie unter 2.14 beschrieben gesendet.

Besonderheiten:

Keine.

3.31 Lok aus Zentralenstack löschen

Format:

	Headerbyte	Kennung	Daten 1	Daten 2	X-Or-Byte
Binär :	1110 0011	0100 0100	Adresse High	Adresse Low	X-Or-Byte
Hex :	0xE3	0x44	AH	AL	X-Or-Byte
Dez :	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 2.11 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.

4 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.

V3	Befehl	Header	Daten1	Daten2	Daten3	Daten4	Daten5	Daten6	Daten7
x	BC Alles An	0x61	0x01	0x60					
x	BC Alles Aus	0x61	0x00	0x61					
x	BC Alle Loks Aus	0x81	0x00	0x81					
x	BC Programmiermode	0x61	0x02	0x63					
x	BC Rückmeldung	0x40 + N	ADR_1	DAT_1	ADR_2	DAT2	usw.	usw.	X-Or
x	P-Info Kurzschluß	0x61	0x12	X-Or					
x	P-Info Keine Daten	0x61	0x13	X-Or					
x	P-Info Busy	0x61	0x1f	X-Or					
x	P-Info bereit	0x61	0x11	X-Or					
x	P-Info Daten 3 Byte	0x63	0x10	EE	DAT	X-Or			
x	P-Info Daten 4 Byte	0x63	0x14	CV	DAT	X-Or			
V3	P-Info Daten 4 Byte	0x63	0x14	CV	DAT	X-Or			
V3.6	P-Info CV1-255 u. 1024	0x63	0x14	CV	DAT	X-Or			
V3.6	P-Info CV256 - 511	0x63	0x15	CV	DAT	X-Or			
V3.6	P-Info CV512 - 767	0x63	0x16	CV	DAT	X-Or			
V3.6	P-Info CV768 - 1023	0x63	0x17	CV	DAT	X-Or			
x	Software LZ	0x63	0x21	DAT1	DAT2	X-Or			
x	Status Zentrale	0x62	0x22	DAT	X-Or				
x	Übertragungsfehler	0x61	0x80	X-Or					
x	Zentrale Busy	0x61	0x81	X-Or					
x	Befehl nicht vorhanden	0x61	0x82	X-Or					
x	Schaltinformationen	0x42	ADR	DAT	X-Or				
x	Normale Lokinfo	0xE4	Kennung	Speed	FKT0	FKT1	X-Or		
V3.6	F-Zustand F13 ... F28	0xE3	0x52	F 13-20	F 21-28	X-Or			
x	MTR-Mitglied	0xE5	Kennung	Speed	FKT0	FKT1	MTR	X-Or	
x	MTR-Basisadresse	0xE2	Kennung	Speed	X-Or				
x	Lok ist in DTR	0xE6	Kennung	Speed	FKT0	FKT1	ADR High	ADR Low	X-Or
x	Lok besetzt	0xE3	0x40	ADR High	ADR Low	X-Or			
x	Funktionsstatus	0xE3	0x50	STAT 0	STAT 1	X-Or			
V3.6	F-status F13 bis 28	0xE4	0x51	STAT 2	STAT 3	X-Or			
x	Lok-Suchergebnis ab	0xE3	0x30 + K	ADR High	ADR Low	X-Or			
x	DTR-Fehler	0x61	0x80 + F	X-Or					
x	Fehlermeldung	0xE1	0x80 + F	X-Or					

5 Befehlsübersicht PC an Zentrale

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

V3	Befehl	Header	Kennung	Daten1	Daten2	Daten3	Daten4	Daten5	Daten6
x	Alles An	0x21	0x81	0xA0					
x	Alles Aus	0x21	0x80	0xA1					
x	Alle Loks anhalten	0x80	0x80						
x	Eine Lok anhalten	0x92	ADR High	ADR Low	X-Or				
x	Prog.-Lesen Register	0x22	0x11	REG	X-Or				
x	Prog.-Lesen CV	0x22	0x15	CV	X-Or				
V3.6	Prog.-Lesen CV1-255; 1024	0x22	0x18	CV low	X-Or				
V3.6	Prog.-Lesen CV256-511	0x22	0x19	CV low	X-Or				
V3.6	Prog.-Lesen CV512-767	0x22	0x1A	CV low	X-Or				
V3.6	Prog.-Lesen CV768-1023	0x22	0x1B	CV low	X-Or				
x	Prog.-Lesen Paging	0x22	0x14	CV	X-Or				
x	Prog.-Ergebnis anfordern	0x21	0x10	0x31					
x	Prog.-Schreiben Register	0x23	0x12	REG	DAT	X-Or			
x	Prog.-Schreiben CV	0x23	0x16	CV	DAT	X-Or			
V3.6	Prog.-Schr. CV1-255; 1024	0x23	0x1C	CV low	DAT	X-Or			
V3.6	Prog.-Schr. CV256-511	0x23	0x1D	CV low	DAT	X-Or			
V3.6	Prog.-Schr. CV512-767	0x23	0x1E	CV low	DAT	X-Or			
V3.6	Prog.-Schr. CV768-1023	0x23	0x1F	CV low	DAT	X-Or			
x	Prog.-Schreiben Paging	0x23	0x17	CV	DAT	X-Or			
x	Softwareversion anfordern	0x21	0x21	0x00					
x	Status Zentrale anfordern	0x21	0x24	0x05					
	Startmode setzen	0x22	0x22	0000M00	X-Or				
x	Schaltinformation anfordern	0x42	ADR	Nibble	X-Or				
x	Schaltbefehl	0x52	ADR	DAT	X-Or				
x	Lokdaten anfordern	0xE3	0x00	ADR High	ADR Low	X-Or			
x	Fkt-Status anfordern	0xE3	0x07	ADR High	ADR Low	X-Or			
V3.6	Fkt-Status anf. F13-F28	0xE3	0x08	ADR High	ADR Low	X-Or			
V3.6	Fkt-Zustand anf. F13-F28	0xE3	0x09	ADR High	ADR Low	X-Or			
x	Lok Fahrbefehl	0xE4	Kennung	ADR High	ADR Low	Speed	X-Or		
x	Lok Funktionsbefehl	0xE4	Kennung	ADR High	ADR Low	Gruppe	X-Or		
x	Funktionsstatus setzen	0xE4	Kennung	ADR High	ADR Low	Gruppe	X-Or		
V3.6	Lok Funktionsbefehl	0xE4	Kennung	ADR High	ADR Low	Gruppe	X-Or		
V3.6	Funktionsstatus setzen	0xE4	Kennung	ADR High	ADR Low	Gruppe	X-Or		
V3.6	Func.refresh-Modus setzen	0xE5	0x2F	ADR High	ADR Low	Modus	X-Or		
x	DTR-Befehle ab V3	0xE5	0x43	ADR1 H	ADR1 L	ADR2 H	ADR2 L	X-Or	
x	Prog. on Main Byte schreiben	0xE6	0x30	ADR High	ADR Low	0xEC + C	CV	DAT	X-Or
V3.6	Prog. on Main Byte lesen	0xE6	0x30	ADR High	ADR Low	0xEA + C	CV	DAT	X-Or
x	Prog. on Main Bit	0xE6	0x30	ADR High	ADR Low	0xE8 + C	CV	DAT	X-Or
x	Lok zu MTR hinzufügen	0xE4	0x40 + R	ADR High	ADR Low	MTR	X-Or		
x	Lok aus MTR entfernen	0xE4	0x42	ADR High	ADR Low	MTR	X-Or		
x	Adresssuche Lok in Mtr	0xE4	0x01 + R	MTR	ADR High	ADR Low	X-Or		

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

6 Unterschiede zu LI100F

Hinsichtlich der XpressNet-Befehle sind keine Änderungen vorgenommen worden. Es werden alle Befehle transparent, d.h. ohne jede weitere Prüfung, zwischen PC und Zentrale weitergeleitet.

Das LI101F enthält ebenfalls eine galvanische Trennung, so daß auch hier nur echte RS232-Schnittstellen benutzt werden können, die die genormten Spannungspegel und Handshake-leitungen aufweisen.

USB-auf-seriell-Adapter genügen diesen Anforderungen nicht und sollten deshalb nicht verwendet werden, wie auch schon beim LI100F.

Das LI101F kann per PC-Programm auf eine XpressNet-Geräteadresse von 1..31 eingestellt werden.

Die Baudrate ist ebenfalls softwareseitig einstellbar auf 19200, 38400 oder 57600 baud. Jeweils 8N1 mit Hardware-Handshake. 9600baud werden nicht mehr unterstützt. 115200baud sind möglich, erfordern aber eventuell einen zu hohen Zeitbedarf bei der Bedienung der seriellen Schnittstelle, so daß eine Einstellempfehlung bei 57600baud liegt.

Für den Fall unterschiedlicher Baudraten, deren Folge Framing-Fehler sind, kann das LI101F durch Senden eines Break-Signales auf 19200baud rückgesetzt werden. Das Break-Signal muß für mindesetens 200 Millisekunden angelegt werden.