

Induktives Schreib-Lese Identssystem CIS3(A)

**Handbuch
für das
Nur-Lesekopf CIT3(A)PL..
Schreib-Lesekopf CIT(A)3SX..**

Ident. Nr. 071652

EUCHNER

EUCHNER GmbH + Co.
Industrie-Elektrik und -Elektronik

Kohlhammerstraße 16
D-70771 Leinfelden-Echterdingen

Telefon: 07 11 / 75 97-0
Telefax: 07 11 / 75 33 16

www.euchner.de
info@euchner.de

Inhaltsverzeichnis

1. Übersicht.....	3
1.1. Kabel- und Klemmenbelegungen	3
1.1.1 Nur-Lesekopf CIT3PL.....	3
1.1.2 Schreib-Lesekopf CIT3SX.....	4
1.2 Anschlußbeispiel für Schreib-Lesekopf CIT3SX.....	4
2. Installationshinweise	4
3. Impulsdiagramm für Nur-Lesekopf CIT3PL.....	5
4. Besondere Punkte, die bei der Programmierung des Nur-Lesekopfes CIT3PL beachtet werden sollten.....	6
4.1 Das Einschaltverhalten des Nur-Lesekopfes CIT3PL.....	6
4.2 Das Timing.....	6
4.3 Die Speicherfunktion	6
4.4 Das Auslesen der 32 Dezimalziffern	7
4.5 Verhalten des Nur-Lesekopfes CIT3PL... im Fehlerfalle.....	7
4.6 Definiertes Zurücksetzen (Reset) des Nur-Lesekopfes CIT3PL.....	8
4.7 Besonderheiten der Datenträger-Programmierung	8
5. Grundsätzlicher Telegrammaufbau ohne Reaktionstelegramm.....	9
5.1 Grundsätzlicher Kommandoaufbau	9
5.2 Besonderheiten des Datenübertragungsprotokolls 3964-R /1/	10
5.2.1 Prinzipielles zu Datenübertragungsverfahren mit Protokoll /1/.....	10
5.2.2 Das Übertragungsprotokoll 3964-R /1/	10
5.2.2.1 Die Steuerung sendet /1/.....	11
5.2.2.2 Die Steuerung empfängt /1/	12
5.2.3 Zusammenfassung der wichtigsten Punkte	12
6. Kommandos zum Schreiben und Lesen des Datenträgers	13
6.1 Initialisierung auf verschiedene Datenträger-Typen mit dem Kommandotelegramm „TU“	13
6.1.1 Der PCF-Modus (Default).....	15
6.1.2 Der NOVA-Modus	15
6.1.3 Der UNIQUE-Modus.....	15
6.2 Schreibvorgang	16
6.2.1 Datenträgerprogrammierung im „NOVA“-Modus	17
6.3 Lesevorgang	18
6.4 Befehlsübersicht	20
7. Fehlermeldungen:.....	20
8. PC-Beispielprogramm in Q-BASIC unter MS-DOS.....	21
9. maximal zulässiger Mittenversatz des CIT3PL... Nur-Lesekopfes.....	26
10. maximale Relativgeschwindigkeit des CIT3PL... Nur-Lesekopfes	27
11. Einbauanleitung für CIS3-Datenträger.....	28
12. Technische Daten.....	29
Literaturhinweis.....	30

1. Übersicht

In diesem Handbuch wird im wesentlichen die Kommunikation zwischen SPS- bzw. CNC-Steuerung und dem Schreib-Lesekopf CIT3SX... beschrieben.
Die Übertragungsprozedur Protokoll R dient zum Anschluß des Schreib-Lesekopfes CIT3SX... an SPS- bzw. NC-Steuerungen .

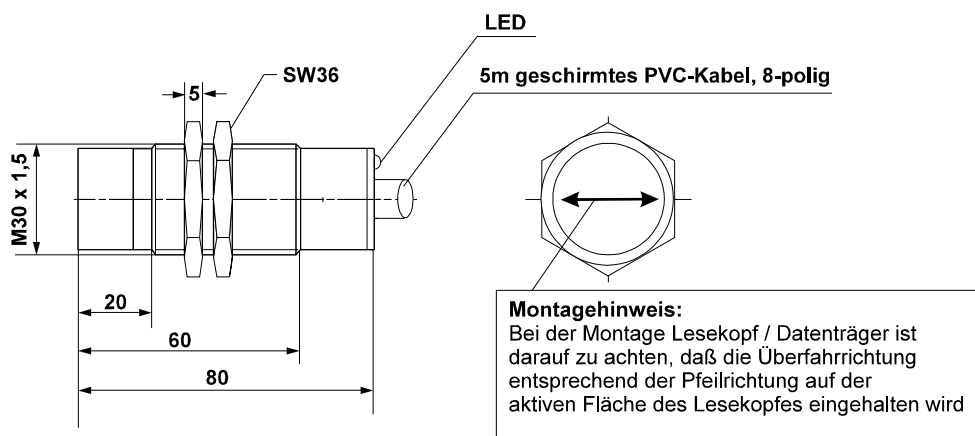
Die Übertragungstelegramme für die Kommandos

- Datenträger programmieren (schreiben)
- Datenträger lesen

basieren auf der Übertragungsprozedur 3964-R /1/

1.1. Kabel- und Klemmenbelegungen

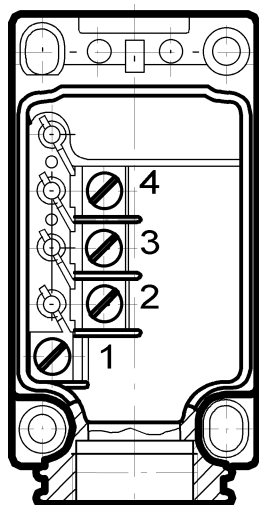
1.1.1 Nur-Lesekopf CIT3PL...



Kabelanschlußfarbe	weiß	braun	grün	gelb	grau	pink	blau	rot	transp.
Anschluß / Signal	0V/GND	24V/UB	A	B	C	D	SKIP	STROBE	Schirm

Abbildung 1: Adernbelegung Nur-Lesekopf CIT3PL...

1.1.2 Schreib-Lesekopf CIT3SX...



Klemme

- 4 TxD
- 3 GND
- 2 RxD
- 1 24V

Abbildung 2: Klemmenbelegung Schreib-Lesekopf CIT3SX1R1G05...
(Basisgerät bei geöffnetem Klemmendeckel)

1.2 Anschlußbeispiel für Schreib-Lesekopf CIT3SX...

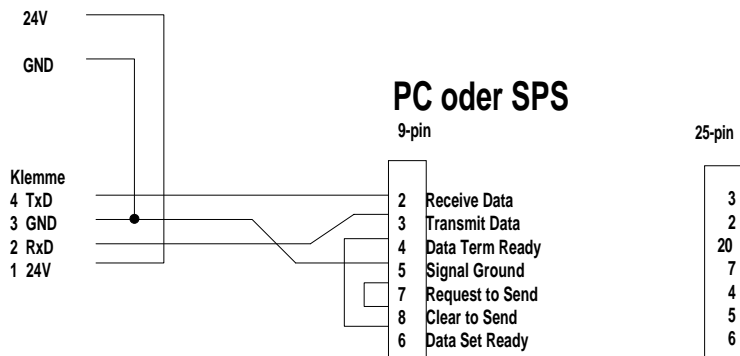
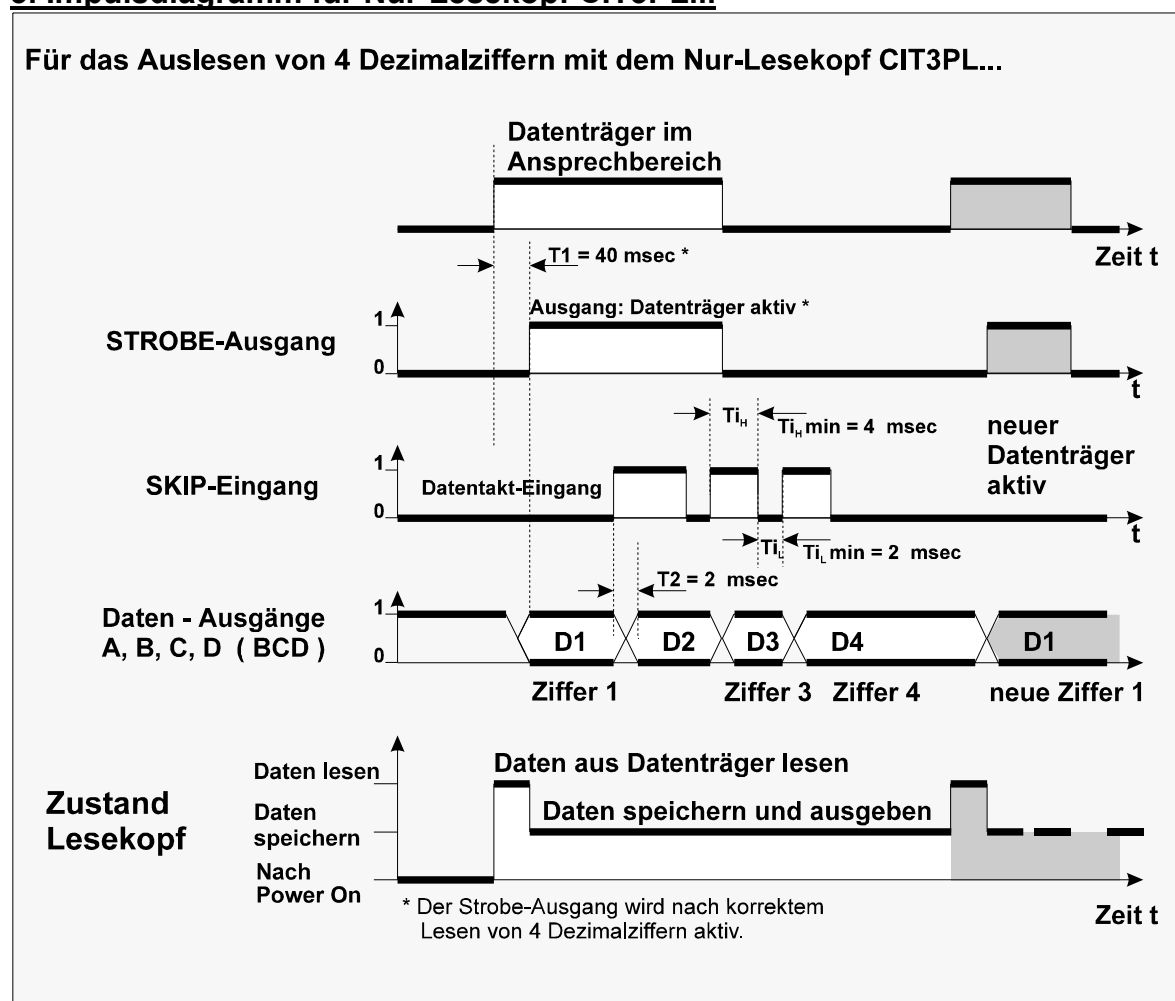


Abbildung 3:

2. Installationshinweise

- Der Nur-Lesekopf CIT3PL..., sowie der Schreib-Lesekopf CIT3SX... dürfen **nicht unter Spannung** angeschlossen werden.
- Der Nur-Lesekopf CIT3PL... sollte **isoliert** montiert werden, um Störungen zu vermeiden.
- Bei der Montage der Anlage ist darauf zu achten, daß die Verfahrrichtung der Datenträger mit dem Pfeil auf dem Nur-Lese- bzw. Schreib-Lesekopf übereinstimmt.

3. Impulsdiagramm für Nur-Lesekopf CIT3PL...



Das obige Impulsdiagramm zeigt das Zeitverhalten der Ein- und Ausgangssignale vom Nur-Lesekopf zur übergeordneten Steuerung.

Sobald ein Datenträger eingelesen wurde, steht die erste Ziffer an den Daten-Ausgängen zur Verfügung.

Um die nachfolgenden Ziffern an den Ausgang zu schalten, muß der Skip-Eingang einen HIGH-Impuls erhalten.

Mit jedem HIGH-Impuls wird um eine Ziffer weitergeschaltet.

Der Nur-Lesekopf speichert eine vom Datenträger gelesene Code-Information so lange, bis ein neuer Datenträger gelesen wurde.

Die Speicherfunktion bietet folgenden Vorteil:

Bei hohen Relativgeschwindigkeiten zwischen Lesekopf und Datenträger ist die Verweildauer eines Datenträgers im Abfragebereich des Lesekopfes unter Umständen kürzer als die Zykluszeit der übergeordneten Steuerung. Auf Grund der Speichereigenschaften des Lesekopfes kann der Code trotzdem von der übergeordneten Steuerung gelesen werden.

Der zusätzliche Ausgang STROBE zeigt mit HIGH-Pegel an, daß sich ein Datenträger im Abfragebereich des Lesekopfes befindet.

Dieser Ausgang wird auf HIGH gesetzt, sobald der Lesekopf 4 Dezimalstellen lesen konnte. Werden mehr als 4 Ziffern benötigt, so kann mit dem STROBE-Ausgang nicht überprüft werden, ob alle benötigten Ziffern eingelesen wurden.

Konnten aus irgendwelchen Gründen (z. B. zu hohe Relativgeschwindigkeit) nicht alle Ziffern gelesen werden, so wird bei der Datenausgabe ab der nicht mehr gelesenen Stelle Fhex (Fehlermeldung) ausgegeben.

4. Besondere Punkte, die bei der Programmierung des Nur-Lesekopfes CIT3PL... beachtet werden sollten

4.1 Das Einschaltverhalten des Nur-Lesekopfes CIT3PL...

- Beim Einschalten des Nur-Lesekopfes CIT3PL... werden alle Datenausgänge A, B, C, D auf HIGH (log. 1) gesetzt.
Dies dient zur Kabelbruchkontrolle und weiterhin zur Kontrolle der Ausgangsstufen des Nur-Lesekopfes CIT3PL...
- Das STROBE-Signal wird beim Einschalten *nicht* auf HIGH (log. 1) gesetzt, um eine Irritation der SPS-Steuerung zu umgehen.
Dieses Signal dient weiterhin als Trigger- (Start-) Signal für die SPS-Steuerung.

4.2 Das Timing

- Zwischen der Erkennung des STROBE-Signals und dem Einlesen der Daten an den Datenausgängen A, B, C, D sollte mind. 1 SPS-Zyklus gewartet werden, um sicherzustellen, daß die Daten auch sicher und stabil an den Ausgängen anliegen.
- Zwischen der positiven Anstiegsflanke des STROBE-Signals und der Ausgabe des ersten SKIP-Signals durch die SPS-Steuerung sollten mindestens 60msec. liegen.
- Das SKIP-Signal wird im Nur-Lesekopf CIT3PL... auf den positiven Flankenanstieg hin untersucht.
Nach der Ausgabe des SKIP-Signals durch die SPS-Steuerung muß mindestens 2 msec. gewartet werden, um sicherzustellen, daß die Daten auch sicher und stabil an den Ausgängen anliegen.

4.3 Die Speicherfunktion

- Wird ein Datenträger am Nur-Lesekopf CIT3PL... vorbeigeführt, so wird der komplette Speicherinhalt des Datenträgers (alle 32 Dezimalziffern) in den Speicher des Nur-Lesekopfes CIT3PL... übernommen.
Der Datenträger muß sich *nicht* während des gesamten Auslesevorganges im aktiven Bereich des Lesekopfes befinden.
- Der Speicher des Nur-Lesekopfes CIT3PL... kann allerdings nur den Datensatz *eines* Datenträgers speichern.
Wird erneut ein Datenträger eingelesen, so wird der Speicherinhalt des Nur-Lesekopfes CIT3PL... mit dem Datum (32 Dezimalziffern) des neuen Datenträgers überschrieben.
Es muß somit sichergestellt werden, daß vor dem Einlesen eines neuen Datenträgers die benötigten Informationen bereits aus dem Nur-Lesekopf CIT3PL... ausgelesen worden sind.
- Der Speicher des Nur-Lesekopfes CIT3PL... ist als Ringspeicher aufgebaut, d.h. wird nach dem Auslesen der 32. Dezimalziffer nochmals ein SKIP-Signal von der SPS-Steuerung an den Nur-Lesekopf CIT3PL... ausgegeben, so wird erneut die erste Ziffer ausgegeben.
Dies kann z.B. zum Überprüfen des bereits eingelesenen Codes genutzt werden (doppeltes Lesen).

4.4 Das Auslesen der 32 Dezimalziffern

- Das STROBE-Signal sollte generell als Trigger- (Start-) Signal für die SPS-Steuerung dienen.
- Der Ausgang STROBE zeigt mit HIGH-Pegel an, daß sich ein Datenträger im Abfragebereich des Nur-Lesekopfes CIT3PL... befindet. Dieser Ausgang wird auf HIGH gesetzt, sobald der Nur-Lesekopfes CIT3PL... 4 Dezimalstellen korrekt lesen konnte.
- Es kann mit dem Ausgang STROBE *nicht* überprüft werden, ob mehr als 4 Ziffern korrekt eingelesen wurden
- Gleichzeitig mit der Ausgabe des STROBE-Signals liegt das Datum der ersten Dezimalziffer an den Datenausgängen A, B, C, D an.
- Nach der Ausgabe des SKIP-Signals durch die SPS-Steuerung, liegt das Datum der zweiten Dezimalziffer an den Datenausgängen A, B, C, D an.
- usw.
- Es müssen allerdings nicht alle 32 Stellen ausgelesen werden. Man kann nur so viele Stellen auslesen, wie benötigt werden, um daraufhin auf einen neuen Datenträger und somit auch auf ein neues STROBE-Signal zu warten.

4.5 Verhalten des Nur-Lesekopfes CIT3PL... im Fehlerfalle

- Der Ausgang STROBE wird auf HIGH gesetzt, sobald der Nur-Lesekopfes CIT3PL... 4 Dezimalstellen korrekt lesen konnte.
- Es kann mit dem Ausgang STROBE *nicht* überprüft werden, ob mehr als 4 Ziffern korrekt eingelesen wurden.
- Der Nur-Lesekopfes CIT3PL...versucht, so viele Ziffern wie möglich einzulesen. Ist es, z.B. aufgrund einer zu hohen Relativgeschwindigkeit für den Nur-Lesekopf CIT3PL... nicht möglich alle 32 Dezimalziffern auszulesen, so werden die Stellen, die nicht mehr korrekt gelesen werden konnten, mit F_{hex} aufgefüllt.
- F_{hex} ist generell eine, vom Nur-Lesekopfes CIT3PL... generierte Fehlermeldung, die somit niemals auf einen Datenträger programmiert werden sollte.

4.6 Definiertes Zurücksetzen (Reset) des Nur-Lesekopfes CIT3PL...

- Aufgrund der Kompaktheit des Nur-Lesekopfes CIT3PL... besteht keine Möglichkeit von extern ein Reset-Signal auf den Nur-Lesekopfes CIT3PL... zu geben, um, z.B. zu Kontrollzwecken den selben Datenträger nochmals erneut einzulesen.
- Dies kann dadurch umgangen werden, daß die Spannungsversorgung des Nur-Lesekopfes CIT3PL... über einen SPS-Ausgang ein- bzw. ausgeschaltet wird. Der Nur-Lesekopfes CIT3PL... trägt durch dieses mehrfaches ein- und ausschalten keinerlei Schäden davon !
- Bleibt ein Datenträger im aktiven Lesebereich des Nur-Lesekopfes CIT3PL... stehen, und wird der Lesekopf dann aus- und wieder eingeschaltet, so wird der Datenträger erneut eingelesen.
Somit ist ein definierter Reset möglich.

4.7 Besonderheiten der Datenträger-Programmierung

- Sollte der Datenträger nicht vollständig mit einer 32-stelligen Dezimalzahl programmiert sein, so werden generell die restlichen Speicherstellen mit einem Füllzeichen beschrieben.
Als Standard hierfür dient E_{hex} , um eine Differenzierung zu allen anderen möglichen Zeichen und Zahlen zu erhalten.
In diesem Falle kann das Zeichen E_{hex} als String-Ende-Zeichen Verwendung finden, um einen Lesevorgang definiert zu beenden.
- Wird an eine Stelle des Datenträger-Speichers eine „0“ programmiert, so muß beachtet werden, daß beim Einlesen der Dateninformationen an den Datenausgängen A, B, C, D alle Ausgänge auf LOW (log. 0) sind, also kein Signal ansteht.
- Dies ist vor allem dann wichtig, wenn an der ersten Stelle des Datenträger-Speichers eine „0“ programmiert ist, da dann beim Auslesen dieses Datenträgers an den Datenausgängen A, B, C, D, die gleichzeitig mit dem STROBE-Signal ausgegeben werden, durchgehend ein LOW (log. 0) anliegt.

5. Grundsätzlicher Telegrammaufbau ohne Reaktionstelegramm

Jedes Kommando und evtl. zugehörige Datenblöcke werden in einem Telegrammkern innerhalb des Telegrammrahmens gemäß der Prozedur 3964-R von und zum Schreib-Lesekopf übertragen (Abbildung 4).

Bei dem Protokoll 3964-R quittiert der jeweilige Empfänger das empfangene Telegramm Durch Rücksendung eines Quittungszeichen (DLE). Bei negativer Quittierung (NAK) wird das komplette Protokoll wiederholt. Kann das Protokoll nach insgesamt sechs Versuchen nicht fehlerfrei übertragen werden, wird der Vorgang abgebrochen.

5.1 Grundsätzlicher Kommandoaufbau

	Beschreibung	Byte-Nr.	I N H A L T (ASCII)	Quittung d. Empfängers + -
Verbindungsaufbau	3964-R Prozedur - Beginn		STX	
				DLE NAK
Telegrammdaten Max. 128 Byte (Telegramm-Kern)	Anzahl der Telegrammbytes	0		
	Kommando-Identifikation	1 2	T oder R Kommando	
	<i>Kopf Adresse</i> *)	3	01h	
	Nutzdatenbeschreibung	4	Startadresse	
		5 6	Startadresse Anzahl Daten	
Nutzdaten	7 bis n			
Verbindungsabbau	3964-R Prozedur - Abschluß		DLE ETX BCC	
				DLE NAK

Abbildung 4: Grundsätzlicher Kommandoaufbau

*) *Zur Abwärtskompatibilität mit dem Identsystem CIS2 .
Bei CIS3 muß der Inhalt der Kopf Adresse immer 1 sein.*

5.2 Besonderheiten des Datenübertragungsprotokolls 3964-R /1/

Bei dem Datenübertragungsprotokoll 3964-R handelt es sich um ein vergleichsweise sicheres Programm für den elektronischen Datenaustausch zwischen einer Steuerung und einem angeschlossenen Peripheriegerät, weil die Datenübertragung mit einem standardisierten Protokoll abgewickelt wird.

Bei Steuerungen mit integriertem 3964-R Treiber (s. z.B. /1/) ist es für den Anwender **nicht** notwendig, sich um die Details des Verbindungsauf- und -abbaus bzw. um die Datensicherung zu kümmern. Hier genügt es, per Programm den Telegrammkern an den 3964-R Treiber zu übergeben.

Bei Steuerungen ohne 3964-R Treiber oder beim Anschluß des Schreib-Lesekopfes an PCs muß der Anwender jedoch auch den Verbindungsauf- und -abbau sowie die Wiederholungsversuche programmieren.

5.2.1 Prinzipielles zu Datenübertragungsverfahren mit Protokoll /1/

Für ein Datenübertragungsverfahren müssen zahlreiche Vereinbarungen getroffen werden; Codes, Betriebsarten, Übertragungsgeschwindigkeiten und der algorithmische Ablauf der Übertragung. Die Festlegung dieses algorithmischen Ablaufs bezeichnet man als **Übertragungsprotokoll** (kurz: *Protokoll*). Ein Übertragungsprotokoll legt im allgemeinen folgende Phasen der Datenübertragung fest:

- Aufforderung von A an B zum Datenaustausch
- Datenaustausch
- Beendigung des Datenaustauschs

Das Übertragungsprotokoll ist im wesentlichen eine Angelegenheit der Steuerung. Das heißt, sie wickelt die Datenübertragung selbständig nach diesem Protokoll ab.

5.2.2 Das Übertragungsprotokoll 3964-R /1/

Im Gegensatz zu protokollfreien Datenübertragungsverfahren handelt es sich beim 3964-R um eine Datenübertragung mit Protokoll. Das bedeutet, daß die eigentlichen Daten, die übertragen werden sollen, in bestimmte Steuerzeichen eingeschlossen werden. Der 3964-R-Treiber erlaubt eine vergleichsweise sichere Datenübertragung dadurch, daß der Empfänger dem Sender seine Empfangsbereitschaft erst signalisieren muß (Verbindungsaufbau) und nach erfolgtem Datenaustausch den richtigen Empfang quittiert. Beim Übertragungsprotokoll 3964-R wird die Datensicherheit durch ein zusätzliches Blockprüfzeichen erhöht.

Der 3964-R-Treiber interpretiert folgende Steuerzeichen:

- DLE (10_{hex}) Datenübertragungsumschaltung (**Data Link Escape**)
- STX (02_{hex}) Anfang des Textes (**Start of Text**)
- NAK (15_{hex}) Negative Rückmeldung (**Negative Acknowledgement**)
- ETX (03_{hex}) Ende des Textes (**End of Text**)

Am Ende eines jeden Datenblocks wird beim Übertragungsprotokoll 3964-R zur Datensicherung ein **Blockprüfzeichen** (kurz: **BCC**) gesendet. Das Blockprüfzeichen BCC ist die **gerade Längsparität** (**EXOR-Verknüpfung aller Datenbytes**) eines gesendeten bzw. empfangenen Blocks. Die Bildung **beginnt** mit dem **ersten Nutzdatenbyte** (**1.Byte des Telegramms**) nach dem Verbindungsaufbau und **endet nach** den Zeichen **DLE** und **ETX** beim Verbindungsabbau.

5.2.2.1 Die Steuerung sendet /1/

Zum Aufbau der Verbindung sendet die Steuerung das Steuerzeichen STX. Antwortet das Peripheriegerät vor Ablauf der Quittungsverzugszeit (QVZ, typisch: 2sec.) mit dem Steuerzeichen DLE, dann geht das Übertragungsprotokoll in den Sendebetrieb über. Antwortet das Peripheriegerät mit dem Steuerzeichen NAK, einem beliebigen anderen Zeichen (außer DLE) oder die Quittungsverzugszeit verstreicht ohne Reaktion, ist der Verbindungsaufbau gescheitert. Nach insgesamt 6 vergeblichen Versuchen (Spezifikation des 3964-R-Protokolls) wird das Verfahren abgebrochen.

Gelingt der Verbindungsaufbau, so werden die im Sendepuffer der Steuerung enthaltenen Nutzinformatiionszeichen mit der gewählten Übertragungsgeschwindigkeit an das Peripheriegerät gesendet. Dieses überwacht den zeitlichen Abstand der ankommenden Zeichen. Der Abstand zwischen zwei Zeichen darf nicht mehr als die Zeichenverzugszeit (ZVZ, typisch: 100ms) betragen.

Jedes in den Nutzinformatiionen enthaltene Steuerzeichen DLE (10_{hex}) **muß zweimal** gesendet werden, damit der Kommunikationspartner erkennt, daß es sich hierbei um Nutzdaten und nicht um das Steuerzeichen DLE handelt. (**DLE-Verdopplung**).

Nach dem Senden der Nutzdaten fügt die Steuerung folgende Zeichen als **Endekennung** an: DLE, ETX, BCC

Danach wartet die Steuerung auf ein Quittungszeichen vom Peripheriegerät. Sendet das Peripheriegerät innerhalb der Quittungsverzugszeit (QVZ, typ.: 2sec.) das Steuerzeichen DLE, so wurde der Datenblock fehlerfrei übernommen.

Antwortet das Peripheriegerät hingegen mit dem Steuerzeichen NAK, einem beliebigen anderem Zeichen oder die Quittungsverzugszeit verstreicht ohne Reaktion, beginnt die Steuerung das Senden erneut mit dem Verbindungsaufbau STX. Nach insgesamt 6 vergeblichen Versuchen (Spezifikation des 3964-R-Protokolls), den Datenblock zu senden, wird das Verfahren abgebrochen und die Steuerung sendet das Steuerzeichen NAK an das Peripheriegerät.

Sendet das Peripheriegerät während einer laufenden Sendung das Steuerzeichen NAK, so bricht die Steuerung den Block ab und wiederholt ihn in der oben beschriebenen Weise. Bei einem anderen Zeichen wartet die Steuerung zunächst auf den Ablauf der Zeichenverzugszeit (ZVZ) und sendet anschließend das Steuerzeichen NAK, um das Peripheriegerät in den Ruhezustand zu bringen. Danach beginnt die Steuerung das Senden erneut mit dem Verbindungsaufbau STX.

5.2.2.2 Die Steuerung empfängt /1/

Empfängt die Steuerung im Ruhezustand das Steuerzeichen STX vom Peripheriegerät, dann antwortet sie mit DLE.

Empfängt die Steuerung im Ruhezustand ein anderes Zeichen (außer STX), so wartet sie auf den Ablauf der Zeichenverzugszeit (ZVZ, typisch: 100ms) und sendet dann das Steuerzeichen NAK. Nach jedem Zeichen wird während der Zeichenverzugszeit (ZVZ) auf das nächste Zeichen gewartet. Verstreicht die Zeichenverzugszeit ohne Empfang, so wird das Steuerzeichen NAK an das Peripheriegerät gesendet.

Erkennt die Steuerung die Zeichenfolge DLE ETX BCC, so beendet sie den Empfang. Sie vergleicht das empfangene Blockprüfzeichen BCC mit der intern gebildeten Längsparität. Ist das Blockprüfzeichen korrekt und kein anderer Empfangsfehler aufgetreten, so sendet die Steuerung das Steuerzeichen DLE. Bei fehlerhaftem BCC wird das Steuerzeichen NAK an das Peripheriegerät gesendet. Anschließend wird eine Wiederholung erwartet. Kann der Block auch nach insgesamt 6 Versuchen (Spezifikation des 3964-R-Protokolls) nicht fehlerfrei empfangen werden oder wird die Wiederholung vom Peripheriegerät nicht innerhalb der Blockwartezeit von 4sec. gestartet, so bricht die Steuerung den Empfang ab.

Treten während des Empfangens Übertragungsfehler auf (verlorenes Zeichen, Rahmenfehler, Paritätsfehler), so wird bis zum Verbindungsabbau weiterempfangen und dann das Steuerzeichen NAK an das Peripheriegerät gesendet. Danach wird eine Wiederholung in der oben beschriebenen Weise erwartet.

5.2.3 Zusammenfassung der wichtigsten Punkte

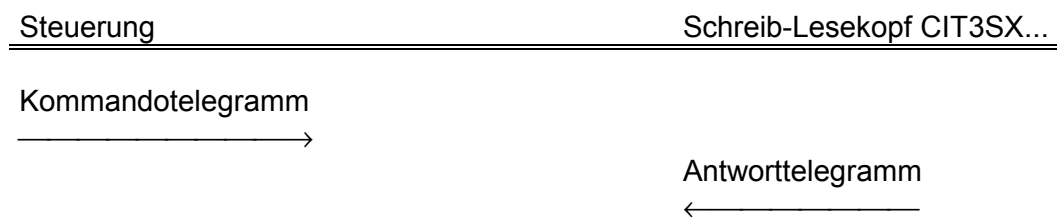
- **Die DLE-Verdopplung:**
Damit die Steuerung zwischen dem Steuerzeichen DLE und einem zufällig vorkommenden DLE als Nutzinformatiionszeichen unterscheiden kann, **muß bei einem DLE als Nutzinformatiionszeichen nochmals ein DLE gesendet werden**. Das heißt, wenn innerhalb des Telegrammkerns ein Byte mit dem ASCII-Wert DLE (10_{HEX}) auftritt , muß dieses Zeichen nochmals übertragen werden damit es von der Gegenstelle nicht als Steuerzeichen für den Verbindungsabbau interpretiert wird.
- **Das Blockprüfzeichen (BCC -Block-Check-Charakter-):**
Am Ende eines jeden Datenblocks wird zur Datensicherung ein Blockprüfzeichen gesendet. Das Blockprüfzeichen BCC ist die **gerade Längsparität (EXOR-Verknüpfung aller Datenbytes)** eines gesendeten bzw. empfangenen Blocks. Die Bildung **beginnt** mit dem **ersten Nutzdatenbyte (1.Byte des Telegramms)** nach dem Verbindungsaufbau und **endet nach** den Zeichen **DLE** und **ETX** beim Verbindungsabbau.
- **Wiederholversuche bei Fehlern:**
Tritt aus irgendwelchen Gründen bei der Datenübertragung ein Fehler auf, so werden **insgesamt 6 Versuche** unternommen, eine korrekte Datenübertragung durchzuführen.

Weitere Details können dem Programmbeispiel im Anhang dieses Handbuches entnommen werden.

6. Kommandos zum Schreiben und Lesen des Datenträgers

Schreib- und Lesevorgänge werden grundsätzlich von der übergeordneten Steuerung (NC, SPS) mit einem "Kommandotelegramm" eingeleitet.

Danach sendet der Schreib-Lesekopf ein Antworttelegramm an die Steuerung.



6.1 Initialisierung auf verschiedene Datenträger-Typen mit dem Kommandotelegramm „TU“

Mit dem Kommandotelegramm „TU“ kann der Schreib-Lesekopf in verschiedene Modis umgeschaltet werden.

Bitte beachten:

Die Initialisierung des Schreib-Lesekopfes muß **einmalig**, nach **jedem** Einschalten des Schreib-Lesekopfes erfolgen.

Zur Initialisierung muß sich **kein Datenträger** vor dem Schreib-Lesekopf befinden.

Die Initialisierung erfolgt mit dem

Kommandotelegramm: (Telegrammkern, SPS → CIT3SX , s. auch Abbildung 5):

TU(S/L-Kopf-Adr.)(0,0)(1)(M*)

- * M = 1 entspricht dem „PCF-Modus“
für EUCHNER CIS3- bzw. CIS3A-Datenträger der 1. Generation
(Default-Wert nach dem Einschalten des Gerätes)
- M = 3 entspricht dem „NOVA-Modus“
für EUCHNER CIS3- bzw. CIS3A-Datenträger der 2. Generation
- M = 8 entspricht dem „UNIQUE-Modus“
(für zukünftige Anwendungen)

Antworttelegramm: (Telegrammkern, CIT3SX → SPS, s. auch Abbildung 6):

RF(S/L-Kopf-Adr.)(0,0)(Fehler-Nr.)

Byte Nr.	Beschreibung	I N H A L T		
		ASCII	HEX	Dezimal
0	Anzahl der Telegrammbytes			8
1	Kommando-Identifikation	T	54	84
2		U	55	80
3	S/L-Kopf Adresse *)		01	01
4	Start-Adresse		00	00
5			00	00
6	Byte - Anzahl		01	01
6	Modus		M *	M *

Abbildung 5: Kommandotelegramm "Schreib-Lesekopf formatieren" (Telegrammkern)

*) **Zur Abwärtskompatibilität mit dem Identsystem CIS2 .
Bei CIS3 muß der Inhalt der Kopf Adresse immer 1 sein.**

* M = 01 entspricht dem „PCF-Modus“
für EUCHNER CIS3- bzw. CIS3A-Datenträger der 1. Generation
(Default-Wert nach dem Einschalten des Gerätes)
M = 03 entspricht dem „NOVA-Modus“
für EUCHNER CIS3- bzw. CIS3A-Datenträger der 2. Generation
M = 08 entspricht dem „UNIQUE-Modus“
(für zukünftige Anwendungen)

Byte Nr.	Beschreibung	I N H A L T		
		ASCII	HEX	Dezimal
0	Anzahl der Telegrammbytes			7
1	Kommando-Identifikation	R	52	82
2		F	46	70
3	S/L-Kopf Adresse *)		01	01
4	Fülldaten		00	00
5			00	00
6	Fehlernummer		Fehlernr. **)	

Abbildung 6: Antworttelegramm "Schreib-Lesekopf formatieren" (Telegrammkern)

*) **Zur Abwärtskompatibilität mit dem Identsystem CIS2 .
Bei CIS3 muß der Inhalt der Kopf Adresse immer 1 sein.**

**) Fehlernummer f = 00: ohne Fehler

Sollte der Initialisierungsvorgang im Schreib-Lesekopf nicht ordnungsgemäß abgeschlossen worden sein, so sendet der Schreib-Lesekopf generell kein Antworttelegramm zurück.

6.1.1 Der PCF-Modus (Default)

Die Grundeinstellung nach dem Einschalten des Schreib-Lesekopfes ist der „PCF-Modus“ (M = 1), d.h. der Modus für die CIS3-Standard-Datenträger, Generation 1.

Dies sind mit dem Schreib-Lesekopf, Typ: CIT3SX1R1G05KS (Ident-Nr.: 040091) folgende Datenträger:

- CIS3P35X16SH01KH (Ident-Nr.: 040045, 071745)
- CIS3P35X16SH01KH (Ident-Nr.: 040045, 071745)
- CIS3P16D08KH01K (Ident-Nr.: 071865, 071867)

sowie mit dem Schreib-Lesekopf, Typ: CIT3AX1R1G05KR (Ident-Nr.: 040090) folgende Datenträger:

- CIS3AP50X50SH01K (Ident-Nr.: 071711, 071783)

Generell ist dieser Modus für Datenträger der Fa. Philips, Typ: PCF7930 und PCF7931, sowie für Datenträger der Fa. Sokymat, Typ „TITAN“ geeignet.

6.1.2 Der NOVA-Modus

Der „Nova-Modus“ (M = 3) für CIS3-Datenträger der Generation 2.

Dies sind mit dem Schreib-Lesekopf, Typ: CIT3AX1R1G05KR (Ident-Nr.: 040090) folgende Datenträger:

- CIS3AP50X50SH01K (Ident-Nr.: 071909, 071910)

Generell ist dieser Modus für Datenträger der Fa. Sokymat, Typ „NOVA“ geeignet.

6.1.3 Der UNIQUE-Modus

Der „UNIQUE-Modus“ (M = 8)

Für zukünftige Anwendungen

Momentan gibt es hierfür noch keinen EUCHNER-Standard-Datenträger.

Generell ist dieser Modus für Datenträger der Fa. Sokymat, Typ „UNIQUE“ geeignet.

6.2 Schreibvorgang

Der Datenträger muß sich bei diesem Kommando vor dem Schreib-Lese-Kopf befinden und darf erst nach Empfang des Antworttelegramms aus dem Ansprechbereich entfernt werden.

Kommandotelegramm (Telegrammkern, SPS → CIT3SX , s. auch Abbildung 7):

TP (S/L-Kopf-Adr.) (Start-Adr.) (Byte-Anz Nutzdaten) (Nutzdaten)

Antworttelegramm (Telegrammkern, CIT3SX → SPS, s. auch Abbildung 8):

RF (S/L-Kopf-Adr.) (0,0) (Fehler-Nr.)

Byte Nr.	Beschreibung	Inhalt bzw. möglicher Wertebereich		
		ASCII	HEX	Dezimal
0	Anzahl der Telegrammbytes			8 ... 23
1	Kommando-Identifikation	T	54	84
2		P	50	80
3	S/L-Kopf Adresse *)		01	01
4	Start-Adresse d. Nutzdaten **)		00	00
5			0 ... 5Fh **)	0 ... 95 **)
6	Byte - Anzahl der Nutzdaten **)		1 ... 10h **)	1 ... 16 **)
7 ... 22	Nutzdaten	ASCII oder HEX bzw. BCD (Codetransparent) **)		

Abbildung 7: Kommandotelegramm "Datenträger schreiben" (Telegrammkern)

- *) **Zur Abwärtskompatibilität mit dem Identsystem CIS2 .
Bei CIS3 muß der Inhalt der Kopf Adresse immer 1 sein.**
- ***) **Der gültige, beschreibbare Speicherbereich der CIS3-Datenträger reicht vom Adressbereich 0_{hex} bis 10_{hex} (16 Byte Nutzdaten).
Siehe Hierzu auch Kapitel „6.2.1 Datenträgerprogrammierung im „NOVA“-Modus“**
- ***) **Wird der Schreib-Lesekopf CIT3SX... zusammen mit dem Lesekopf CIT3PL eingesetzt, ist darauf zu achten, daß sich bei Programmierung der Nutzdaten ein Byte aus 2 BCD-Digits (0 ... 9) zusammensetzt.**

Byte Nr.	Beschreibung	I N H A L T		
		ASCII	HEX	Dezimal
0	Anzahl der Telegrammbytes			7
1	Kommando-Identifikation	R	52	82
2		F	46	70
3	S/L-Kopf Adresse *)		01	01
4	Fülldaten		00	00
5			00	00
6	Fehlernummer		Fehlernr. **)	

Abbildung 8: Antworttelegramm "Datenträger schreiben" (Telegrammkern)

*) Zur Abwärtskompatibilität mit dem Identsystem CIS2 .
Bei CIS3 muß der Inhalt der Kopf Adresse immer 1 sein.

**) Fehlernummer f =

- 00_{hex}: ohne Fehler
- 02_{hex}: Datenträger nicht im Ansprechbereich
- 04_{hex}: Fehler beim Programmieren bzw. Kontrolllesen des Datenträgers
- 05_{hex}: Schreibvorgang abgebrochen, da der Datenträger aus dem Ansprechbereich weggefahren ist
- 06_{hex}: Schreibvorgang abgebrochen.
Initialisierungs-Modus „M“ des Schreib-Lesekopfes paßt nicht mit dem, zu beschreibenden Datenträger zusammen.
- 08_{hex}: Schreibvorgang abgebrochen.
Initialisierungs-Modus „M“ des Schreib-Lesekopfes paßt nicht mit dem, zu beschreibenden Datenträger zusammen.
- 16_{hex}: Nutzdatenlänge ist größer als 16 Byte

6.2.1 Datenträgerprogrammier im „NOVA“-Modus

WICHTIG !!! BITTE BEACHTEN !!! WICHTIG !!! BITTE BEACHTEN !!! WICHTIG !!!

Bei CIS3-Datenträger, die im „NOVA“-Modus programmiert werden, muß folgendes unbedingt beachtet werden:

- Die „Start-Adresse der Nutzdaten“ (Byte 4 und 5 im Telegrammkern des Kommandotelegramms "Datenträger schreiben", siehe hierzu auch Abbildung 7) darf immer nur geradzahlig sein (Zum Beispiel: 0, 2, 4, 6...).
- Die „Byte-Anzahl der Nutzdaten“ (Byte 6 im Telegrammkern des Kommandotelegramms "Datenträger schreiben", siehe hierzu auch Abbildung 7) darf immer nur geradzahlig sein (Zum Beispiel: 2, 4, 6...).

6.3 Lesevorgang

Kommandotelegramm (Telegrammkern, SPS → CIT3SX , s. auch Abbildung 9):

TL (S/L-Kopf-Adr.) (Start-Adr.) (Byte-Anz Nutzdaten)

Antworttelegramm (Telegrammkern, CIT3SX → SPS, s. auch Abbildung 10 oder Abbildung 11):

Bei diesem Kommando gibt es zwei verschiedene Möglichkeiten einer Antwort

1. RL (S/L-Kopf-Adr.) (Start-Adr.) (Byte-Anz Nutzdaten) (Nutzdaten)

oder

2. RF (S/L-Kopf-Adr.) (0,0) (Fehler-Nr.)

Das Antworttelegramm RL (siehe auch Abbildung 10) steht für fehlerfreien Empfang der Daten.

Wenn ein Datenträger nicht gelesen werden kann, bekommt man ein RF Antworttelegramm (siehe auch Abbildung 11). Die Fehlernummer zeigt dann die Fehlerursache an.

Byte Nr.	Beschreibung	INHALT		
		ASCII	HEX	Dezimal
0	Anzahl der Telegrammbytes			7
1	Kommando-Identifikation	T	54	84
2		L	4C	76
3	S/L-Kopf Adresse *)		01	01
4	Start-Adresse d.		00	00
5	Nutzdaten		0 ... Fh	0 ... 15
6	Byte - Anzahl der Nutzdaten		1 ... 10h	1 ... 16

Abbildung 9: Kommandotelegramm "Datenträger lesen" (Telegrammkern)

Byte Nr.	Beschreibung	INHALT		
		ASCII	HEX	Dezimal
0	Anzahl der Telegrammbytes			8 ... 23
1	Kommando-Identifikation	R	52	82
2		L	4C	76
3	S/L-Kopf Adresse *)		01	01
4	Start-Adresse d.		00	00
5	Nutzdaten		0 ... Fh	0 ... 15
6	Byte - Anzahl der Nutzdaten		1 ... 10h	1 ... 16
7 ... 22	Nutzdaten	ASCII oder HEX bzw. BCD (Codetransparent)		

Abbildung 10: Antworttelegramm "Datenträger lesen" (Telegrammkern)

Byte Nr.	Beschreibung	I N H A L T		
		ASCII	HEX	Dezimal
0	Anzahl der Telegrammbytes			7
1	Kommando- Identifikation	R	52	82
2		F	46	70
3	S/L-Kopf Adresse *)		01	01
4	Fülldaten		00	00
5			00	00
6	Fehlernummer		Fehlernr. **)	

Abbildung 11: Antworttelegramm "Datenträger lesen" (Telegrammkern)

*) **Zur Abwärtskompatibilität mit dem Identsystem CIS2 .
Bei CIS3 muß der Inhalt der Kopf Adresse immer 1 sein.**

) Fehlernummer f = **00_{hex}: ohne Fehler
 02_{hex}: Datenträger nicht im Ansprechbereich
 03_{hex}: Lesevorgang abgebrochen

6.4 Befehlsübersicht

Beschreibung	Kommandotelegramm	Antworttelegramm
Daten programmieren	(Anz. Tele-Bytes) TP (S/L-Kopf-Adr. *) (Start-Adr.) (Byte-Anz Nutzdaten) (Nutzdaten.....)	(Anz. Tele-Bytes) RF (S/L-Kopf-Adr. *) (Fehler-Nr.)
Datenträger Im Ansprechbereich Lesen	(Anz. Tele-Bytes) TL (S/L-Kopf-Adr. *) (Start-Adr.) (Byte-Anz. Nutzdaten)	(Anz. Tele-Bytes) RL (S/L-Kopf-Adr. *) (Start-Adr.) (Byte-Anz. Nutzda.) (Nutzda.) <i>oder</i> (Anz. Tele-Bytes) RF (S/L-Kopf-Adr. *) (Fehler-Nr.)

7. Fehlermeldungen:

- 00_{hex}**: ohne Fehler
- 02_{hex}**: Datenträger nicht im Ansprechbereich
- 03_{hex}**: Lesevorgang abgebrochen
- 04_{hex}**: Fehler beim Programmieren bzw. Kontrolllesen des Datenträgers
- 05_{hex}**: Schreibvorgang abgebrochen, Datenträger aus dem Ansprechbereich weggefahren
- 06_{hex}**: Schreibvorgang abgebrochen.
Initialisierungs-Modus „M“ des Schreib-Lesekopfes paßt nicht mit dem, zu beschreibenden Datenträger zusammen.
- 08_{hex}**: Schreibvorgang abgebrochen.
Initialisierungs-Modus „M“ des Schreib-Lesekopfes paßt nicht mit dem, zu beschreibenden Datenträger zusammen.
- 16_{hex}**: Datenlänge ist größer als 16 Byte

8. PC-Beispielprogramm in Q-BASIC unter MS-DOS

```
' ***** Demoprogramm für EUCHNER S/L System *****'  
DECLARE FUNCTION Lesebefehl$ (Start!, Anzahl%, Kopfnr%)  
DECLARE FUNCTION Schreibbefehl! (Start!, Anzahl%, Kopfnr%, Daten$)  
DECLARE FUNCTION Telegrammsenden! (Telegramm$, AnVersu!)  
DECLARE FUNCTION Antworttelegramm$ (AnVersu!)  
DECLARE FUNCTION BCCermittlung$ (Telegramm$)  
DECLARE FUNCTION Datenauswerten$ (A$)  
DECLARE SUB Fehleranzeigen (A$)  
  
REM ***** Einstellung fuer COM1 *****  
REM OPEN "COM1:9600,N,8,1,CD,CS,DS,OP0,RS,TB0,RB0" FOR RANDOM AS #1  
REM OUT &H3FB, &H1B '8 Datenbit 1 Stopbit Even Parity für COM1  
  
REM ***** Einstellung fuer COM2 *****  
OPEN "COM2:9600,N,8,1,CD,CS,DS,OP0,RS,TB0,RB0" FOR RANDOM AS #1  
OUT &H2FB, &H1B '8 Datenbit 1 Stopbit Even Parity für COM2  
  
REM ***** Verbindungsabbau *****  
END$ = CHR$(&H10) + CHR$(&H3)  
  
DO  
CLS  
PRINT "Datenträger lesen          :1"  
PRINT "Datenträger schreiben      :2"  
PRINT "Daten ausdrucken           :3"  
PRINT "Beenden                     :4"  
PRINT " "  
PRINT " "  
INPUT "Auswahl :?", A  
  
SELECT CASE A  
  
REM ***** Datentraeger lesen *****  
CASE 1  
INPUT "Datenträger Startadresse    :", A!  
INPUT "Anzahl der Daten 0 bis 16   :", B%  
INPUT "Kopfadresse                 :", C%  
INPUT "Schleife J/N                :", D$  
SCHLEIFE:  
Returnwert$ = Lesebefehl(A!, B%, C%)  
IF Returnwert$ = "0" THEN  
PRINT "Keine gültigen Daten eingelesen"  
ELSE  
PRINT Returnwert$  
END IF  
  
FOR i = 0 TO 10  
NEXT i  
  
IF D$ = "J" THEN GOTO SCHLEIFE  
INPUT H  
CASE 2
```

CASE 2

```
REM ***** Datentraeger schreiben *****
INPUT "Datentraeger Startadresse      :", A!
INPUT "Daten max. 16 Bytes           :", D$
INPUT "Kopfadresse                   :", C%
B% = 16
REM B% = LEN(D$)
Returnwert = Schreibbefehl(A!, B%, C%, D$)
```

```
REM ** Datentraegerinhalt nach lesen ausdrucken **
```

CASE 3

```
IF Returnwert$ = "0" THEN
  PRINT "Keine gultigen Daten eingelesen"
ELSE
  OPEN "LPT1:" FOR OUTPUT AS #2
  PRINT #2, Returnwert$
  INPUT H
  CLOSE #2
END IF
```

CASE 4

```
REM ***** Programmende *****
END
CASE ELSE
  INPUT "Fehlerhafte Eingabe", H
END SELECT
```

```
LOOP UNTIL 0
```

```
CLOSE #1
```

```
END
```

```
/*
```

```
***** 3964R Antworttelegramm einlesen *****
```

```
/*
```

```
FUNCTION Antworttelegramm$ (AnVersu)
```

```
END$ = CHR$(&H10) + CHR$(&H3)
```

```
STX$ = CHR$(2)
```

```
DLE$ = CHR$(&H10)
```

```
NAK$ = CHR$(&H15)
```

```
Wiederholzaehler = 0
```

```
Startantworttegr:
```

```
A$ = INPUT$(1, #1)          'STX von Bussystem einlesen
```

```
PRINT #1, DLE$;           'DLE senden
```

```
Wiederholzaehler = Wiederholzaehler + 1
```

```
A$ = INPUT$(1, #1)          'Telegrammlänge einlesen
```

```
B = ASC(A$) - 1
```

```
BCCWE = ASC(A$)
```

```
IF A$ = DLE$ THEN          'DLE Vedopplung nach 3964R
```

```
  D$ = INPUT$(1, #1)
```

```
  BCCWE = BCCWE XOR ASC(D$)
```

```
END IF
```

```
FOR i = 1 TO B              'Telegramm einlesen
```

```
  C$ = INPUT$(1, #1)
```

BCCWE = BCCWE XOR ASC(C\$)

```

IF C$ = DLE$ THEN          'DLE Vedopplung nach 3964R
  D$ = INPUT$(1, #1)
  BCCWE = BCCWE XOR ASC(D$)
END IF
A$ = A$ + C$
NEXT i
C$ = INPUT$(1, #1)        'DLE einlesen
BCCWE = BCCWE XOR ASC(C$)
C$ = INPUT$(1, #1)        'ETX einlesen
BCCWE = BCCWE XOR ASC(C$)
C$ = INPUT$(1, #1)        'BCC einlesen
IF Wiederholzaehler = AnVersu THEN 'nach mehrmaligen ungültigen Versuch, Verbindung abbrechen
  PRINT "Kein gültiges Antworttelegramm"
  PRINT #1, NAK$
  GOTO Abbruch
END IF
IF BCCWE <> ASC(C$) THEN      'Vergleich ob Telegramm-BCC richtig ist
  PRINT "BCC-Fehler", BCCWE, ASC(C$)
  PRINT #1, NAK$
  GOTO Startantworttegr      'Erwartet Protokollwiederholung
END IF
PRINT #1, DLE$;            'positiv quittieren

```

Antworttelegramm\$ = A\$

Abbruch:

END FUNCTION

'*

***** Ermittlung des BCC *****

'*

FUNCTION BCCermittlung\$ (Telegramm\$) STATIC

BCC = 0

FOR i = 1 TO LEN(Telegramm\$)

BCC = BCC XOR ASC(MID\$(Telegramm\$, i, 1))

NEXT i

BCCermittlung\$ = CHR\$(BCC)

END FUNCTION

'*

***** Daten aus dem Antworttelegramm ausfiltern *****

'*

FUNCTION Datenauswerten\$ (A\$)

IF MID\$(A\$, 3, 1) = "K" THEN PRINT "Daten korrigiert"

Z = ASC(MID\$(A\$, 7, 1))

B\$ = MID\$(A\$, 8, Z)

Datenauswerten\$ = B\$

END FUNCTION

'*

***** Fehler oder Statusermittlung *****

'*

SUB Fehleranzeigen (A\$)

PRINT "Fehler Nr. : ";

B\$ = MID\$(A\$, 7, 1)

PRINT ASC(B\$)

INPUT X

END SUB

```

'
*
***** Datenträger lesen *****
'*
FUNCTION Lesebefehl$ (Start!, Anzahl%, Kopfnr%)

Starthi% = Start \ 256          'Startadresse in high und low Teil aufteilen
Startlo% = Start MOD 256

Telegramm$ = CHR$(7) + "TL" + CHR$(Kopfnr%) + CHR$(Starthi%) + CHR$(Startlo%) + CHR$(Anzahl%)

IF Telegrammsenden(Telegramm$, 6) <> 0 THEN GOTO ende1

A$ = Antworttelegramm(6)      'Antworttelegramm einlesen

PRINT "Antworttelegramm auswerten"
          'Antworttelegramm auswerten
IF MID$(A$, 3, 1) = "F" THEN Fehleranzeigen (A$): Lesebefehl$ = "0"
IF MID$(A$, 3, 1) = "L" THEN Lesebefehl$ = Datenauswerten(A$)
IF MID$(A$, 3, 1) = "K" THEN Lesebefehl$ = Datenauswerten(A$)

ende1:
END FUNCTION

*
***** Daten auf Datenträger schreiben *****
'*
FUNCTION Schreibbefehl (Start!, Anzahl%, Kopfnr%, Daten$)

Starthi% = Start \ 256          'Startadresse in high und low Teil aufteilen
Startlo% = Start MOD 256

Telegramm$ = CHR$(LEN(Daten$) + 7) + "TP" + CHR$(Kopfnr%) + CHR$(Starthi%) + CHR$(Startlo%) +
CHR$(Anzahl%) + Daten$

A = Telegrammsenden(Telegramm$, 6)
IF A <> 0 THEN GOTO ende

PRINT "Programmiertelegramm gesendet"

A$ = Antworttelegramm(6)      'Antworttelegramm einlesen

PRINT "Antworttelegramm auswerten"
          'Antworttelegramm auswerten
IF MID$(A$, 3, 1) = "F" THEN Fehleranzeigen (A$)

Schreibbefehl = 0
ende:

END FUNCTION

*
***** 3964R Telegramm senden *****
'*
FUNCTION Telegrammsenden (Telegramm$, AnVersu!)
END$ = CHR$(&H10) + CHR$(&H3)
STX$ = CHR$(2)
DLE$ = CHR$(&H10)

```


Wiederholzaehler = 0

Protokolstart:

```
PRINT #1, STX$;           'STX ausgeben
A$ = INPUT$(1, #1)       'DLE einlesen
```

```
IF Wiederholzaehler = AnVersu! THEN
  PRINT "Kein Verbindungsaufbau": INPUT " ", Q
  Telegrammsenden = -1
  GOTO ENDE2
END IF
```

```
IF A$ <> DLE$ THEN
  Wiederholzaehler = Wiederholzaehler + 1
  GOTO Protokolstart
END IF
```

```
telegraus$ = ""
FOR i = 1 TO LEN(Telegramm$)
  C$ = MID$(Telegramm$, i, 1)
```

```
  IF C$ = DLE$ THEN telegraus$ = telegraus$ + C$ 'DLE Verdopplung nach 3964R
  telegraus$ = telegraus$ + C$
NEXT i
```

' Kommandotelegramm senden

```
PRINT #1, telegraus$ + END$ + BCCermittlung$(telegraus$ + END$);
```

```
A$ = INPUT$(1, #1)           'DLE von Bussystem einlesen
IF Wiederholzaehler = AnVersu! THEN PRINT "negative Quittierung": INPUT " ", Q: Telegrammsenden = -1:
GOTO ENDE2
IF A$ <> DLE$ THEN Wiederholzaehler = Wiederholzaehler + 1: GOTO Protokolstart
PRINT "Anforderung gesendet"
Telesenden = 0:
ENDE2:
```

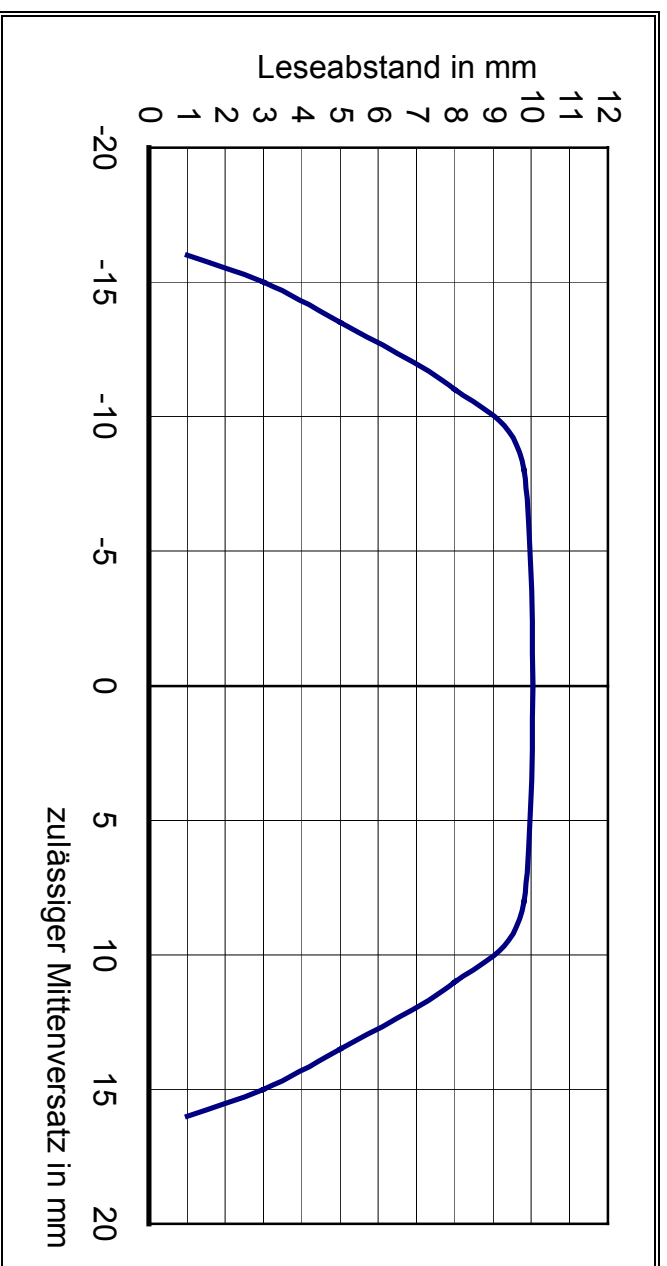
```
END FUNCTION
```

9. maximal zulässiger Mittenversatz des CIT3PL... Nur-Lesekopfes

Testbedingungen: Umgebungstemperatur: 25°C

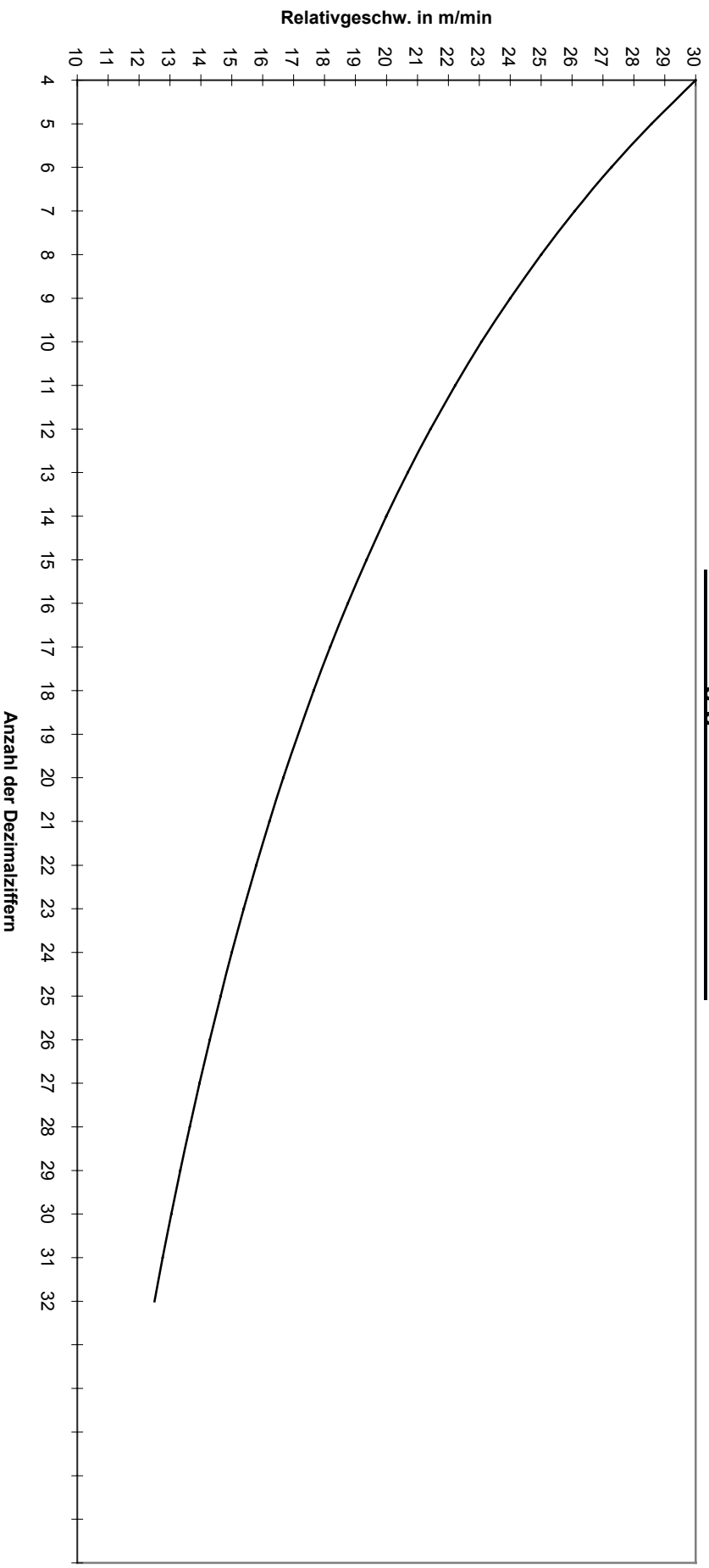
Verwendete Geräte: Nur-Lesekopf CIT3PL1N30-5000 (Ident-Nr. 040085)

Datenträger CIS3P35X16SH01KH (Ident-Nr. 040045)



10. maximale Relativgeschwindigkeit des CIT3PL... Nur-Lesekopfes

max. Lese-Relativgeschwindigkeit
in Abhängigkeit der Zifferanzahl

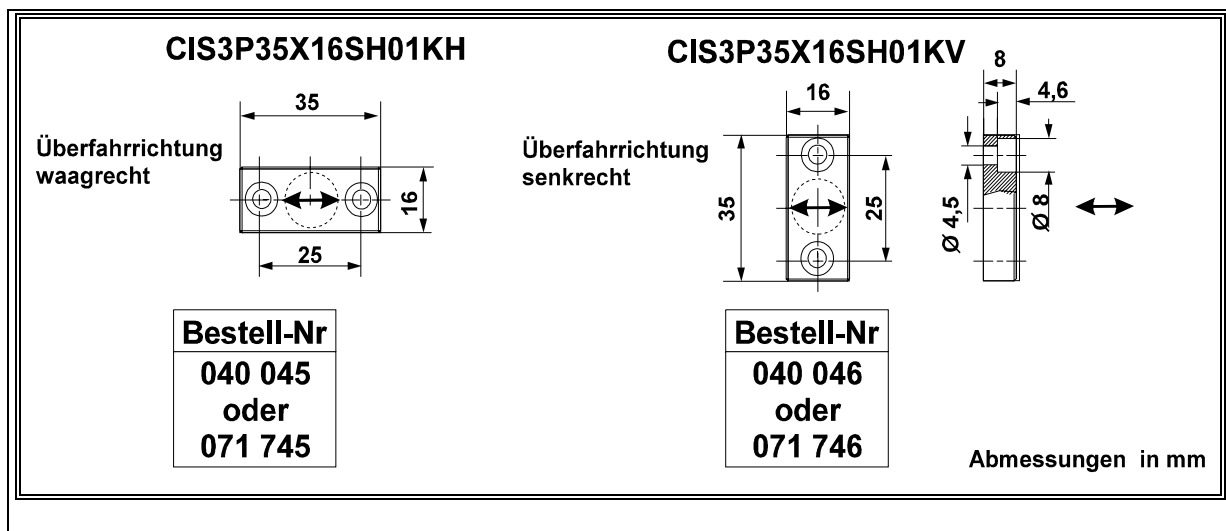


Anmerkung: Da *mindestens* 4 Dezimalziffern gelesen werden müssen, beginnt auch das Diagramm erst bei der 4. Dezimalziffer

11. Einbauanleitung für CIS3-Datenträger

Um die in den Datenblätter für das Identsystem CIS3 enthaltenen technischen Spezifikationen auch bei **bündigem Einbau in Metall** einhalten zu können, müssen die folgenden Punkte für den Einbau der Datenträger beachtet werden:

- Beim Datenträger, Typ CIS3P35X16SH01KH (Bestell-Nr. 040045 bzw. 071745), **muß** an der Längsseite (35mm-Seite) ein **Mindestabstand von 5mm je Seite** zum umgebenden Metall eingehalten werden.
Die Querseite (16mm-Seite) kann direkt an das umgebende Metall angrenzen.
- Der Datenträger, Typ CIS3P35X16SH01KV (Bestell-Nr. 040046 bzw. 071746)), kann ohne weiteres bündig in Metall eingebaut werden.
Es muß auf keiner Seite ein Mindestabstand zum umgebenden Metall eingehalten werden.



12. Technische Daten

CIT3PL1N30-5000

Betriebsspannung:	15 - 28 V
Stromaufnahme:	100 mA (max.)
Laststrom pro Ausgang:	30 mA (max.)
Lagertemperatur	0 - 80°C
Umgebungstemp	0 - 50°C
Leseabstand:	0 - 12 mm (siehe Seite 18)
Mittensversatz:	nom. ± 10 mm (siehe Seite 18)
Ansprechzeit (für 4 Dezimalziffern):	40 msec. (max.)
Relativ-Geschwindigkeit:	max. 30m/min. (siehe Seite 19)

CIT3SX1R1G05KS:

Betriebsspannung:	15 - 25 V
Stromaufnahme:	140 mA (max.)
Lagertemperatur	0 - 80°C
Umgebungstemp	0 - 50°C
Anschlußart:	Schraubklemmen
Leitungslänge (RS232):	5 m (max.)
Leseabstand:	0 - 12 mm
Schreibabstand (statisches Schreiben):	0 - 6 mm
Schreibdauer	
4 Dezimalziffern:	230 ms (max.)
32 Dezimalziffern:	420 ms (max.)

Für die serielle Schnittstelle (V24, RS 232) sind folgende Werte eingestellt:

1 Startbit
8 Datenbit
1 Paritybit (gerade Parität)
1 Stopbit

Übertragungsrate: 9600 Baud

Literaturhinweis

/1/ SIEMENS

Handbücher :

Anschlußkomponenten für S5 Steuerungen:

Kommunikations-Prozessor CP 521 SI*Bezeichnung :*

Gerätehandbuch CP 521 SI

SIEMENS-Bestell-Nr.

GES5 998 - 1 UD 11

Kommunikations-Prozessor CP 523*Bezeichnung :*

Gerätehandbuch CP 523(d/e/f/s/i)

SIEMENS-Bestell-Nr.

GES5 998 - 0 DD d 1

Kommunikations-Prozessor CP 544*Bezeichnung :*

Gerätehandbuch CP 544(d/e/f)

SIEMENS-Bestell-Nr.

GES5 998 - 2 DB d 1

/2/ Link, W.

Codierung und Codesicherung bei programmierbaren
Datenträgern
IDENT'88 , Sindelfingen