



**Intelligenter, multifunktionaler**

**B1 Chipkartenleser**

## Inhaltsverzeichnis

	Seite
<b>1 Abkürzungen</b>	<b>4</b>
<b>2 Zum Inhalt</b>	<b>6</b>
<b>3 Einleitung</b>	<b>6</b>
3.1 2.1 PC/SC Workgroup	6
<b>4 Komponenten des Leserkonzeptes</b>	<b>7</b>
<b>5 Anschließen der B1-Chipkartenleser</b>	<b>8</b>
5.1 AT-B1-Chipkartenleser (Auftischversion)	8
5.2 Snuggle AT-B1-Chipkartenleser	9
5.3 EB-B1-Chipkartenleser (Einbauversion 3,5“ und 5,25“)	10
5.4 FB-B1-Chipkartenleser (Flachbaugruppe)	12
5.5 FB-Scenic Pro-B1-Chipkartenleser (Flachbaugruppe für Scenic Pro)	13
5.6 PC Card-B1-Chipkartenleser	14
<b>6 Installation der B1-Chipkartenleser-Software</b>	<b>15</b>
6.1 Dateien der Installationsdiskette	15
6.2 MS-DOS	17
6.3 Windows 3.1x	17
6.4 Windows 95	18
6.5 Windows NT	19
6.6 SINIX/RM	19
6.7 SUN-SOLARIS 2.5	20
6.8 LINUX	21
6.9 UNIX BSD	21
6.10 FLEXOS	21
<b>7 Funktionstest mit dem Testprogramm</b>	<b>22</b>
<b>8 Update der Chipkartenleser-Firmware</b>	<b>22</b>
<b>9 Programmentwicklung für den B1-Chipkartenleser</b>	<b>24</b>
9.1 Ansteuerung des B1-Chipkartenlesers	24

---

9.2 Einrichten der Entwicklungsdateien	24
9.3 Include-Datei CTAPI.H	24
9.4 Allgemeine Hinweise zur Verwendung des CT-API	24
9.5 Schicht 1 Reset-Mechanismus	25
9.6 Zusätzliche Festlegungen zur Schicht 2	26
9.7 Bedienung von mehreren B1-Chipkartenlesern mit CT-API	28
9.8 Entwicklung unter MS-DOS	28
9.9 Entwicklung unter Windows 3.1x	29
9.10 Entwicklung unter WIN32 (Windows 95 und Windows NT)	30
9.11 Entwicklung unter SINIX/RM	31
9.12 Entwicklung unter SUN-SOLARIS 2.5	32
9.13 Entwicklung unter LINUX	32
<b>10 Betrieb mit KV-Anwendungen</b>	<b>32</b>
<b>11 Chipkarten-Protokolle</b>	<b>33</b>
11.1 Automatische Protokollerkennung	33
11.2 Protocol Type Selection (PTS)	33
<b>12 Lieferumfang</b>	<b>33</b>
<b>13 Datenblatt B1-Chipkartenleser-AT (Auftischleser)</b>	<b>34</b>
13.1 Technische Daten	35
<b>14 Datenblatt B1-Chipkartenleser (PC Card Leser)</b>	<b>36</b>
14.1 Technische Daten	37
<b>15 Datenblatt B1-Chipkartenleser-PC (Einbauversion 3,5" / 5.25")</b>	<b>38</b>
15.1 Technische Daten	39
<b>16 Datenblatt B1-Chipkartenleser-FB (Flachbaugruppe)</b>	<b>40</b>
16.1 Technische Daten	41
<b>17 Datenblatt B1-Chipkartenleser-FB (Flachbaugruppe für Scenic Pro)</b>	<b>42</b>
17.1 Technische Daten	43
<b>18 Datenblatt B1-Chipkartenleser Snuggle AT</b>	<b>44</b>
18.1 Technische Daten	45

## 1 Abkürzungen

API	Application Programming Interface
AT	B1 Auftischversion
ATR	Answer to Reset
BGT	Block Guard Time
CRC	Cyclic Redundancy Check
CT	Card Terminal
DSR (‘)	Data Set Ready ( ‘ = Datenstrom vom PD)
DTE (‘)	Data Terminal Equipment ( ‘ = Datenstrom vom PD)
DTR (‘)	Data Terminal Ready ( ‘ = Datenstrom vom PD)
EB	B1 Einbauversion
EDC	Error Detection Code
EMV	Europay Mastercard Visa
FB	B1 Flachbaugruppe
GND	Ground
HBCI	Homebanking-Computer-Interface
HTSI	Host Transport Service Interface
IBWT	Initial Block Waiting Time
ICC	Integrated Circuit Card
ICWT	Initial Character Waiting Time
ID	Identification
IFD	Interface Device
IFS	Information Field Size
ISO/IEC	International Standardization Organization/International Engineering Consortium
KV	Krankenversicherung
LED	Leuchtdiode
LRC	Longitudinal Redundancy Check
NAD	Note Address
PC Card	neue Bezeichnung für PCMCIA (Einschubversion ES)

PC/SC	Personal Computer/Smartcard
PD	Peripheral Device (z.B. B1-Chipkartenleser)
PTS	Protocol Type Selection
RxD (‘)	Received Data ( ‘ = Datenstrom vom PD)
SDK	Software Development Kit
TTL	Transistor-Transistor-Logik
TxD (‘)	Tranferred Data ( ‘ = Datenstrom vom PD)
Vpp	Programmierspannung am PIN

## 2 Zum Inhalt

Dieses Handbuch beschreibt die Inbetriebnahme, Benutzung und die Leistungsmerkmale des B1-Chipkartenlesers mit seriellem Anschluß. Auf Details der Programmierung des B1-Chipkartenlesers wird nur ansatzweise eingegangen. Für Anwendungsprogrammierer steht das Programmierhandbuch (B0/B1 HTSI, Vers. 3.1 E) zur Verfügung.

Die PC Card-Variante des Chipkartenlesers ist in der Dokumentation „B1 Chipkartenleser PC Card“ beschrieben und nicht Gegenstand der folgenden Ausführungen.

## 3 Einleitung

Heute vorhandene Chipkartenanwendungen unterteilen sich in Anwendungen von Speicher-, Mikroprozessorkarten und kombinierte Anwendungen. Es existiert eine große Vielfalt verschiedener Chips/Chipbetriebssysteme. Diese unterscheiden sich im wesentlichen im angebotenen Speicherplatz, den verfügbaren (Grund-) Funktionen und den Schnittstellenprotokollen an den Kontakten. Für die Entwickler einer Chipkartenanwendung ist gerade die Vielfalt der Schnittstellenprotokolle ein Problem. Es ist ein sehr hohes Detailwissen erforderlich, um eine Chipkarte erfolgreich betreiben zu können. Der B1-Chipkartenleser bietet in Verbindung mit dem CT-API eine leicht lernbare Anwendungsschnittstelle, mit der sich verschiedenste Speicher- und Mikroprozessorkartenchips mit EINER Anwendungsschnittstelle und EINEM Chipkartenleser betreiben lassen. Ein Detailwissen über die Schnittstellenprotokolle ist nicht notwendig. Es muß lediglich der Befehlssatz des Chips/Chipbetriebssystems bekannt sein.

Der B1-Chipkartenleser ist kompatibel zu Kartenterminals im Gesundheitswesen und kann daher mit bestehenden KV-Anwendungen betrieben werden.

### 3.1 PC/SC Workgroup

Die PC/SC Workgroup besteht seit Mai 1996 und hat als übergeordnetes Ziel die Definition einer umfassenden und flexiblen Lösung für die Integration von Smartcards in den PC mit dem Resultat einer Spezifikation. Mitgliedsunternehmen sind u.a. Hewlett-Packard, IBM, Microsoft, Siemens Nixdorf und Sun Microsystems.

Anlaß der Gründung war der Standardisierungsbedarf bei Schnittstellen zu IFDs und der Untersuchungsbedarf der kritischen Bereiche (gemeinsame Programmierschnittstellen und Steuerungsmechanismen).

Maßnahmen:

- Standardisierung von Schnittstellen zu IFDs
- weitestgehende Einbindung bestehender Produkte und Anwendungen
- Entwicklung von flächendeckenden Lösungen für die Anforderungen der Industrie
- Plattformunabhängigkeit

Die PC/SC-Architektur ist für Windows 95™ und Windows NT V4.0™ in den PC Card-Chipkartenleser B1 integriert worden. Die Komponenten B1-IFD-Handler in der E-Version und CT-API Service Provider stehen zur Verfügung. Die Microsoft-Testsuites für WIN'95 und WIN NT wurden erfolgreich durchlaufen.

## 4 Komponenten des Leserkonzeptes

Das Konzept des B1-Chipkartenlesers umfaßt eine Reihe unterschiedlicher Bauformen:

### ξ Typ AT-B1 (Auftisch-Version)

Dieses Chipkarten-Terminal wird als externes Gerät über eine serielle Kommunikationsschnittstelle an ein Hostsystem angeschlossen. Die Kommunikationsschnittstelle basiert auf dem Schnittstellenstandard V.24/V.28. Als Übertragungsprotokoll/Transportprotokoll zum Host wird der ISO-Standard 7816-3 Amd.1 „T=1“ verwendet. Zur Stromversorgung wird ein Steckernetzteil mitgeliefert. Für den Anschluß an einen Host gehört ein V.24-Anschlußkabel mit 8-poligem Western Plug und 9-poligem Sub-D zum Lieferumfang.

### ξ Typ Snuggle AT-B1

Bei dieser Version handelt es sich um eine Weiterentwicklung des AT-B1-Chipkarten-Terminals, die die gleiche Funktionalität in einem kompakteren Gehäuse, das auf dem Tisch nur wenig Stellfläche erfordert, zur Verfügung stellt. Zusätzlich kann der Snuggle seitlich am Bildschirm befestigt werden. Der Snuggle entspricht der Schnittstellenbeschreibung für ec-Karten mit Chip (EMV '96) und ist somit für Anwendungen wie z.B. HBCI geeignet. Die Kommunikationsschnittstelle basiert auf dem Schnittstellenstandard V.24/V.28. Als Übertragungsprotokoll/Transportprotokoll zum Host wird der ISO-Standard 7816-3 Amd.1 „T=1“ verwendet. Die Stromversorgung erfolgt über den PS/2-Port (Maus- bzw. Tastaturanschluß des Desktops oder Notebooks). Das hierzu benötigte Kabel sowie ein Standardfuß, ein Klettband und Klebeband gehören zum Lieferumfang.

### ξ Typ EB-B1 (PC-Einbauversion)

In dieser Version wird das Chipkarten-Terminal in einen PC wie ein 3,5“ Diskettenlaufwerk eingebaut. Die Kommunikation erfolgt über einen mitgelieferten seriellen Schnittstellenadapter (V.24-Karte). Wenn kein geeigneter 3,5“- Platz zur Verfügung steht, kann ein mitgelieferter mechanischer Adapter 3,5“/5,25“ benutzt werden. Die notwendigen Anschlußkabel (Daten- und Stromversorgungskabel) sind Bestandteil der Lieferung.

### ξ Typ FB-B1 (Flachbaugruppe)

Diese Version des B1-Chipkartenlesers besteht aus einer Flachbaugruppe und einer Anschlußmöglichkeit für eine Kontaktiereinheit für die Chipkarte. Die Kontaktiereinheit kann über ein Flex-Kabel mit der Flachbaugruppe verbunden werden. Die Flachbaugruppe ist auch ohne Kontaktiereinheit lieferbar. Damit ist diese Bauform für den Einbau in Warenautomaten, Zutrittskontrollleinheiten oder ähnliche Geräte geeignet, die einen Chipkartenleser benötigen. Der Anschluß kann über Schnittstelle V.24 oder TTL erfolgen. Die Spannungsversorgung erfolgt über einen Stecker X3.

### ξ Typ FB-Scenic Pro-B1 (Flachbaugruppe für Scenic Pro)

Diese Version des B1-Chipkartenlesers besteht aus einer Flachbaugruppe, die im Chassis der SNI Scenic Pro PCs eingebaut wird. Die Kontaktiereinheit ist fester Bestandteil der Flachbaugruppe und wird über einen 16poligen Stecker mit +5V vom Motherboard versorgt. Diese Flachbaugruppe wird über eine V.24-Schnittstelle an das Motherboard des PCs angeschlossen.

### ξ Typ PC Card-B1

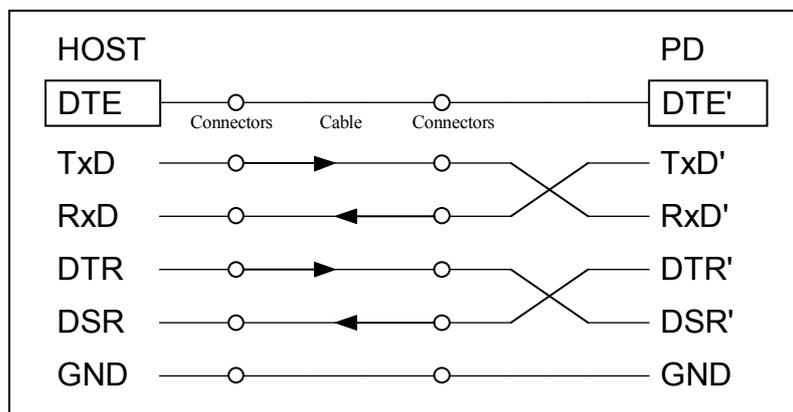
In dieser Version kann das ChipkartenTerminal in einen PC Card-Slot geschoben werden. Als Steuerungs- und Kommunikationsschnittstelle wird die PC Card-Schnittstelle verwendet. Weitere Ausführungen entnehmen Sie bitte der Dokumentation „PC Card-B1-Chipkartenleser“.

Zu allen Leservarianten gehört eine Installationsdiskette mit der B1-Chipkartenleser-Software (incl. relevanter Komponenten für Entwickler), ein Testprogramm und ein Ladeprogramm zum Update der Firmware im Leser. Der Inhalt dieser Diskette steht ebenfalls im Internet unter der Seite „[http://public.sni.com/telecom/ge\\_telecom/-ccsecurity/cc\\_sec2.htm](http://public.sni.com/telecom/ge_telecom/-ccsecurity/cc_sec2.htm)“ zum Download zur Verfügung. Eine Testchipkarte und das Programmierhandbuch (B0/B1 HTSI, Vers. 3.1 E) gehören nicht zum Lieferumfang und müssen separat bestellt werden.

## 5 Anschließen der B1-Chipkartenleser

### 5.1 AT-B1-Chipkartenleser (Auftischversion)

Der AT-B1-Chipkartenleser wird gemäß der folgenden Zeichnung an den Host angeschlossen. Die Leitungen sind im AT-B1-Chipkartenleser gekreuzt verbunden. Für den Anschluß an die V.24-Schnittstelle des Hosts ist dem Leser ein Anschlußkabel beigelegt. Die Stromversorgung erfolgt über ein mitgeliefertes Steckernetzteil an der Rückseite des Lesers.



### LED-Anzeigen

Die zwei LEDs des angeschlossenen und eingeschalteten B1-Chipkartenlesers haben folgende Bedeutung:

LED	Bedeutung
aus	Der B1-Chipkartenleser ist betriebsbereit.
blinkt gelb	Der B1-Chipkartenleser hat beim Selbsttest einen Fehler festgestellt und ist nicht betriebsbereit. Dieser Zustand kann nur direkt nach dem Selbsttest auftreten.
grün	Der B1-Chipkartenleser wurde von einer Anwendung initialisiert. Die Schnittstelle zur ICC ist nicht aktiviert.
gelb & grün	Der B1-Chipkartenleser wurde von einer Anwendung initialisiert. Die Schnittstelle zur ICC ist aktiviert.

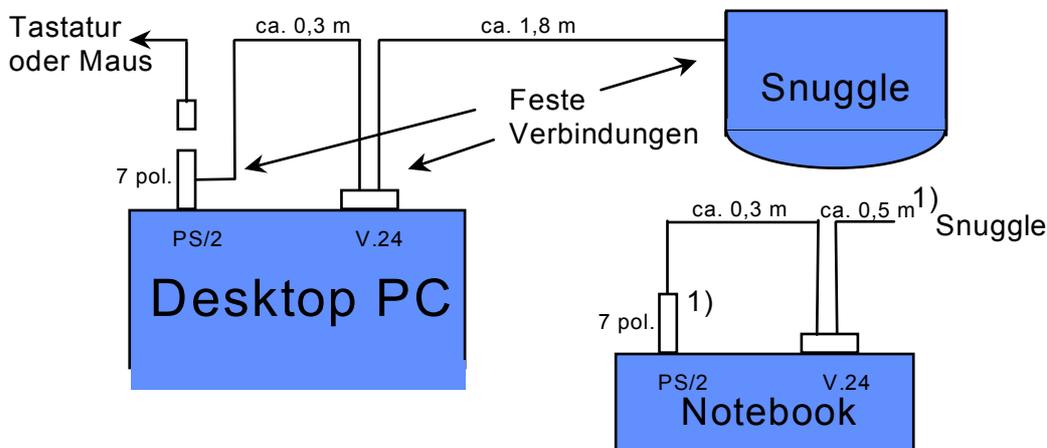
## 5.2 Snuggle AT-B1-Chipkartenleser

Der Snuggle AT-B1-Chipkartenleser wird gemäß der folgenden Zeichnung an den Host angeschlossen. Die Leitungen sind im Snuggle AT-B1-Chipkartenleser gekreuzt verbunden. Die für den Anschluß benötigten Kabel sind dem Leser beigelegt.

Abweichungen des Notebook-Anschlusses zum Desktop-Anschluß:

Notebook	Desktop
PS/2-Stecker ermöglicht keinen weiteren Maus- bzw. Tastaturanschluß	PS/2-Stecker ermöglicht weiterhin den Maus- bzw. Tastaturanschluß
V.24-Anschlußkabel ist 0,5 m lang	V.24-Anschlußkabel ist 1,8 m lang

### Daten- und Stromversorgungskabel



1) Abweichend von Desktop-Anschluß

### LED-Anzeigen

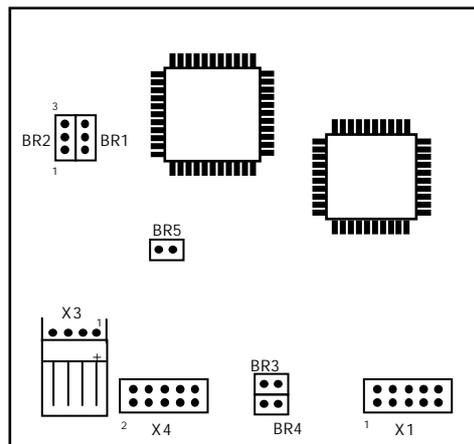
Diese Version des B1-Chipkartenlesers besitzt **eine** 2-farbige LED. Sie hat bei angeschlossenem B1-Chipkartenleser folgende Bedeutung:

LED	Bedeutung
aus	Der B1-Chipkartenleser ist betriebsbereit.
blinkt orange	Der B1-Chipkartenleser hat beim Selbsttest einen Fehler festgestellt und ist nicht betriebsbereit. Dieser Zustand kann nur direkt nach dem Selbsttest auftreten.
grün	Der B1-Chipkartenleser wurde von einer Anwendung initialisiert. Die Schnittstelle zur ICC ist nicht aktiviert.
orange	Der B1-Chipkartenleser wurde von einer Anwendung initialisiert. Die Schnittstelle zur ICC ist aktiviert.

### 5.3 EB-B1-Chipkartenleser (Einbauversion 3,5" und 5,25")

Der Einbau des B1-Chipkartenlesers in der Einbauversion in den PC erfolgt wie bei einem 3,5" Diskettenlaufwerk und sollte von technisch versierten Personen vorgenommen werden. Für den Einbau in einen 5,25"-Schacht steht ein Adapter 3,5"/5,25" zur Verfügung.

Die Kommunikation erfolgt über eine interne serielle Schnittstellenkarte (V.24) des PCs.



#### Spannungsversorgung

Am Stecker **X3** müssen folgende Spannungen angelegt werden:

Pin 1:	+ 5V DC
Pin 2:	Ground
Pin 3:	Ground
Pin 4:	+12 V DC

Diese Steckerbelegung entspricht jener eines 3,5" Diskettenlaufwerkes.

#### Anschlüsse

Der Stecker **X1** stellt eine V.24-Schnittstelle dar. Damit die B1-Einbauversion mit der V.24-Schnittstelle betrieben werden kann, müssen die Brücken **BR1** und **BR2** in der Stellung 1-2 gesteckt werden.

Der Stecker **X4** stellt alternativ eine TTL-Schnittstelle dar. Für den Betrieb der EB-Variante des Chipkartenlesers mit TTL-Schnittstelle siehe Kapitel 5.4.

Die Pinbelegung für den Stecker **X1** (V.24-Schnittstelle):

Pin 1:	frei	Pin 6:	DSR
Pin 2:	TxD	Pin 7:	frei
Pin 3:	RxD	Pin 8:	frei
Pin 4:	DTR	Pin 9:	frei
Pin 5:	Ground	Pin 10:	frei

### ***Funktion der zusätzlichen Brücken***

**BR3:** Benutzerkonfiguriert „Ausgang“

**BR4:** Benutzerkonfiguriert „Eingang“

**BR5:** Verbindung von Vpp an die Kontaktiereinheit.

Auf **keiner** der Brücken BR3-BR5 sollte ein Brückenstecker (Jumper) gesteckt sein.

### ***LED-Anzeigen***

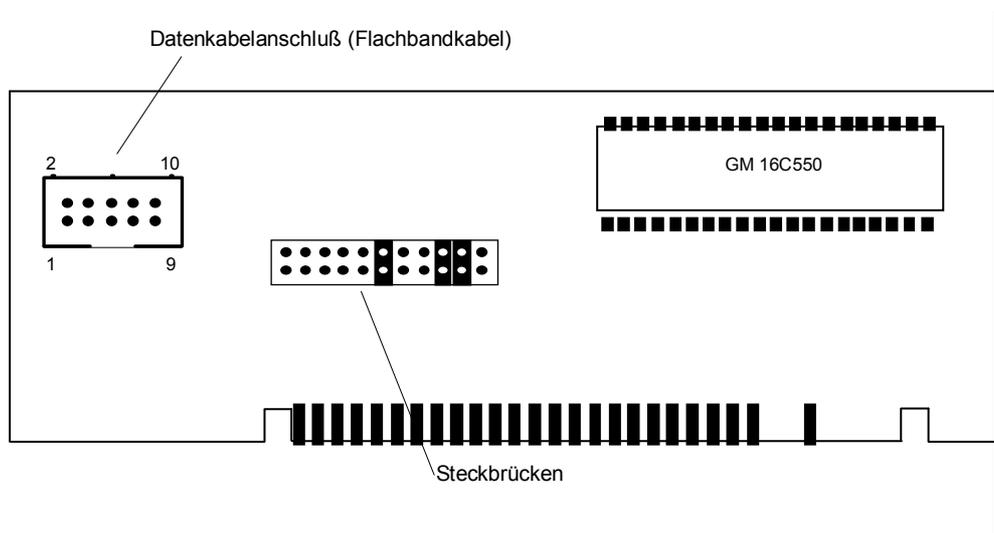
Die Einbauversion des B1-Chipkartenlesers besitzt **eine** 2-farbige LED. Die Bedeutung dieser LED ist dem Kapitel 5.2 zu entnehmen.

### ***Serielle Schnittstellenkarte V.24***

Die Schnittstellenkarte (ISA 16-bit) wurde konzipiert für den Anschluß von Chipkartenlesern des Typs B1 Version „Einbau“ der SBS GmbH und Co. OHG, CC Security.

Kartentyp:	V.24-Telekom, Rel. 1.10
CE- Zeichen:	Vergeben
Abmessungen:	ca. 155 x 50 mm
Schnittstelle:	V.24 intern
Kompatibel:	IBM AT Standard (BIOS)
Adressierung:	COM1 3F8 COM2 2F8 COM3 3E8 *) COM4 2E8
Unterbrechungen:	IRQ3, IRQ4 *), IRQ5, IRQ9, IRQ10, IRQ11, IRQ12. *) Default-Einstellungen
Datenkabel:	0,8 m lang, Schneidklemmensteckverbinder, 1:1 Flachbandleitung, 10-pol.

Stromversorgungskabel: 0,2 m lang, 5,25“ auf 5,25“ und 3,5“ (Netzstecker 4-pol. auf Netzsocket groß und Netzsocket klein)



#### 5.4 FB-B1-Chipkartenleser (Flachbaugruppe)

Die B1-Flachbaugruppe ist technisch identisch mit der Einbauversion des B1-Chipkartenlesers. Diese Variante beinhaltet eine Zweifarben-Leuchtdiode und kann optional mit Kontaktiereinheit geliefert werden.

Als Verbindung zum Hostrechner stehen entweder eine V.24-Schnittstelle oder eine TTL-Schnittstelle zur Verfügung. Für den Betrieb mit der V.24-Schnittstelle siehe Kapitel 5.3. Es sind zwei Protokolle realisiert, T=1 und DIN 66019. Da für jedes Protokoll eine spezielle Firmware existiert, ist bei der Bestellung das später zu verwendende Protokoll anzugeben. Eine Beschreibung des Protokolls nach DIN 66019 kann bei Bedarf zur Verfügung gestellt werden.

Am Stecker **X4** der B1-Flachbaugruppe können die TTL-Signale abgegriffen werden. Dazu müssen die Brücken **BR1** und **BR2** in der Stellung 2-3 gesteckt werden.

Das Anschlussschema des FB-B1 Chipkartenlesers ist der Zeichnung im Kapitel 5.3 zu entnehmen.

### **Spannungsversorgung**

Am Stecker **X3** müssen folgende Spannungen angelegt werden:

Pin 1:	+ 5V DC
Pin 2:	Ground
Pin 3:	Ground
Pin 4:	+12 V DC

Diese Steckerbelegung entspricht jener eines 3,5" Diskettenlaufwerkes.

### **Anschlüsse**

Pinbelegung für den Stecker **X4** (TTL-Schnittstelle):

Pin 1:	frei	Pin 6:	Busy
Pin 2:	TxD	Pin 7:	frei
Pin 3:	RxD	Pin 8:	frei
Pin 4:	Reset	Pin 9:	frei
Pin 5:	Ground	Pin 10:	frei

### **Zusätzliche Kontaktiereinheit**

Am Stecker **X5** kann eine externe Kontaktiereinheit zugesteckt werden.

Pinbelegung des Steckers **X5**:

Pin 1:	Presence-Kontakt	Pin 6:	Kontakt C6
Pin 2:	Presence-Kontakt(+5V)	Pin 7:	Kontakt C3
Pin 3:	Kontakt C1	Pin 8:	Kontakt C7
Pin 4:	Kontakt C5	Pin 9:	Kontakt C4
Pin 5:	Kontakt C2	Pin 10:	Kontakt C8

### **Funktion der zusätzlichen Brücken**

**BR3:** Benutzerkonfiguriert „Ausgang“

**BR4:** Benutzerkonfiguriert „Eingang“

**BR5:** Verbindung von Vpp an die Kontaktiereinheit.

Auf **keiner** der Brücken BR3-BR5 sollte ein Brückenstecker (Jumper) gesteckt sein.

## **5.5 FB-Scenic Pro-B1-Chipkartenleser (Flachbaugruppe für Scenic Pro)**

Die B1-Flachbaugruppe für den Scenic Pro PC besitzt im Gegensatz zu der in Kapitel 5.4 beschriebenen Flachbaugruppe eine Einfarben-Leuchtdiode.

Auf dem 16poligen Stecker **X2** liegen einerseits die Stromversorgung des FB-Scenic Pro-B1-Chipkartenlesers und andererseits die Signale der V.24-Schnittstelle auf. Über ein Flachbandkabel, das der Baugruppe beiliegt, wird die Verbindung zum Motherboard hergestellt.

Die Pinbelegung für den Stecker **X2** (V.24-Schnittstelle und Spannungsversorgung):

Pin 1:	frei	Pin 9:	Ground
Pin 2:	DTR	Pin 10:	frei
Pin 3:	TxD	Pin 11:	frei
Pin 4:	frei	Pin 12:	+ 5V DC
Pin 5:	RxD	Pin 13:	frei
Pin 6:	frei	Pin 14:	Ground
Pin 7:	Reset	Pin 15:	Ground
Pin 8:	frei	Pin 16:	frei

## 5.6 PC Card-B1-Chipkartenleser

Diese Ausführung ist in der Dokumentation „PC Card-B1-Chipkartenleser“ beschrieben.

## 6 Installation der B1-Chipkartenleser-Software

Vor Inbetriebnahme des B1-Chipkartenlesers ist die Installation der benötigten Software-Komponenten erforderlich. Dieser Vorgang muß manuell durchgeführt werden, erfordert aber keine speziellen Systemkenntnisse.

Die mitgelieferte Installationsdiskette enthält die Treiberdateien, Testprogramme, Firmware Update-Programme für die unterstützten Plattformen sowie Dateien für die Anwendungsentwicklung.

Der Installationsvorgang besteht darin, die Dateien von der Installationsdiskette auf das Zielsystem zu übertragen. Für diese Übertragung werden die auf dem Zielbetriebssystem geeigneten Programme/Befehle benutzt.

In den folgenden, nach Zielsystemen gegliederten Punkten sind für alle benötigten Dateien jeweils der Name und die Pfadangabe auf der Installationsdiskette sowie deren Bedeutung angeführt.

### 6.1 Dateien der Installationsdiskette

Auf der Installationsdiskette für den B1-Chipkartenleser finden Sie folgende Dateistruktur:

Verzeichnis	Datei	Bedeutung
\DOS	B1TEST.EXE	Testprogramm
	B1UPDATE.EXE	Firmware Update-Programm
	SETDTR.EXE	B1-HW-Initialisierung für KV-Anwendungen
\DOS\LIB	CTMS15S.LIB	CT-API Bibliothek für Microsoft Compiler (Small Model)
	CTMS15M.LIB	CT-API Bibliothek für Microsoft Compiler (Medium Model)
	CTMS15L.LIB	CT-API Bibliothek für Microsoft Compiler (Large Model)
	CTMS45S.LIB	CT-API Bibliothek für Borland Compiler (Small Model)
	CTMS45M.LIB	CT-API Bibliothek für Borland Compiler (Medium Model)
	CTMS45L.LIB	CT-API Bibliothek für Borland Compiler (Large Model)
	CTAPI.H	C-Include-Datei für CT-API

<b>\WIN16</b>	B1TEST.EXE	Testprogramm
	CT.DLL	Dynamische Bibliothek für CT-Anwendungen
	CTSERVER.EXE	Server-Programm für CT.DLL
<b>\WIN16\LIB</b>	CT.LIB	Import-Bibliothek für CT.DLL
	CTAPI.H	C-Include-Datei für CT-API
<b>\WIN32</b>	B1TEST.EXE	Testprogramm
	B1UPDATE.EXE	Firmware Update-Programm
	CT32.DLL	Dynamische Bibliothek für CT-Anwendungen
<b>\WIN32\LIB</b>	CT32.LIB	Import-Bibliothek für CT.DLL
	CTAPI.H	C-Include-Datei für CT-API
<b>\SINIX</b>	b1test	Testprogramm
	b1htsi.cfg	Musterdatei für die Zuordnung der seriellen Schnittstelle
<b>\SINIX\LIB</b>	libb1ct.a	Bibliothek für CT-API
	ctapi.h	C-Include-Datei für CT-API
<b>\SOLARIS</b>	b1test	Testprogramm
	b1htsi.cfg	Musterdatei für die Zuordnung der seriellen Schnittstelle
<b>\SOLARIS\LIB</b>	libb1ct.a	Bibliothek für CT-API
	ctapi.h	C-Include-Datei für CT-API
<b>\LINUX</b>	b1test	Testprogramm
	b1htsi.cfg	Musterdatei für die Zuordnung der seriellen Schnittstelle
<b>\LINUX\LIB</b>	libb1ct.a	Bibliothek für CT-API
	ctapi.h	C-Include-Datei für CT-API

Im Verzeichnis ...LIB jeder Plattform befinden sich die Dateien zur Entwicklung von Anwendungsprogrammen. Diese werden für den Ablauf von CT-API-Anwendungen nicht benötigt.

## 6.2 MS-DOS

Für die Übertragung der benötigten Dateien empfiehlt sich der DOS-Befehl *copy*.

Installationsdiskette...	Verzeichnis des Zielsystems ...
\DOS\B1TEST.EXE	beliebig
\DOS\B1UPDATE.EXE	beliebig
\DOS\SETDTR.EXE	beliebig

Für den Betrieb des B1-Chipkartenlesers wird kein DOS-Gerätetreiber benötigt. Die Funktionsfähigkeit des B1-Chipkartenlesers kann durch Aufruf des Testprogramms *B1TEST* überprüft werden.

## 6.3 Windows 3.1x

Zum Einbringen der Dateien ins System empfiehlt sich der Windows-Dateimanager.

Installationsdiskette...	Verzeichnis des Zielsystems ...
\WIN16\B1TEST.EXE	beliebig
\WIN16\CTSERVER.EXE	Installationsverzeichnis von Windows (z.B.: C:\WINDOWS)
\WIN16\CT.DLL	Installationsverzeichnis von Windows (z.B.: C:\WINDOWS)

Für den Ablauf von CT-API-Programmen werden die beiden Dateien *CT.DLL* und *CTSERVER.EXE* benötigt. *CTSERVER* ist ein Hintergrundprogramm ohne sichtbare Oberfläche, das automatisch beim Betrieb von CT-API-Anwendungen gestartet und beendet wird.

Nach dem Kopieren der beiden Dateien in das Installationsverzeichnis von Windows können CT-API-Anwendungen (z.B. *B1TEST*) betrieben werden. Änderungen in der Systemkonfiguration sind nicht erforderlich.

Um den Ablauf von CT-API-Anwendungen zu steuern, kann im Windows-Installationsverzeichnis die Datei *B1CTAPI.INI* mit dem Abschnitt *[settings]* angelegt werden, mit deren Hilfe folgende Einstellungen festgelegt werden:

ξ Show=[0|1]

Das Programm *CTSERVER* wird automatisch gestartet, wenn eine Anwendung den B1-Chipkartenleser anspricht. Normalerweise ist der Ablauf von *CTSERVER* für den Benutzer nicht sichtbar. Wenn *Show* mit dem Wert 1 eingestellt ist, zeigt *CTSERVER* sein Sinnbild auf dem Desktop an. Das Sinnbild kann nicht vergrößert werden.

Standardeinstellung: 0

ξ KeepRunning=[0|1]

Um Windows-Systemressourcen zu sparen, wird der *CTSERVER* nur bei Bedarf gestartet, d.h. wenn eine Anwendung mit dem B1-Chipkartenleser kommuniziert. Mit der Einstellung *KeepRunning=1* kann

festgelegt werden, daß CTSERVER nach dem Schließen einer CT-API-Anwendung nicht beendet wird. Diese Einstellung ist dann sinnvoll, wenn aus Performancegründen CTSERVER zum Ablauf von CT-API-Anwendungen nicht immer neu gestartet und beendet werden soll.

Standardeinstellung: 0

ξ **DetectAppExit=[0|1]**

Unter Windows müssen Anwendungen benutzte Systemressourcen nach deren Verwendung bzw. bei Beendigung freigeben, da diese sonst für andere Windows-Anwendungen nicht verfügbar sind. Wenn sich CT-API-Anwendungen nicht ordnungsgemäß abmelden, könnte in dieser Windows-Sitzung keine andere Anwendung mit dem B1-Chipkartenleser kommunizieren. Um dies zu verhindern, überwacht CTSERVER alle CT-API-Anwendungen und beendet die Verbindung zum B1-Chipkartenleser automatisch, wenn sich die Anwendung nicht ordnungsgemäß abmeldet. Diese Überwachung kann mit DetectAppExit=0 abgeschaltet werden.

Standardeinstellung: 1

Das Anlegen der Datei *B1CTAPI.INI* ist für den ordnungsgemäßen Betrieb des B1-Chipkartenlesers nicht notwendig. Existiert diese Datei nicht, sind die oben beschriebenen Standardeinstellungen wirksam.

Eine manuelle Beendigung von CTSERVER ist normalerweise nicht notwendig. Wenn CTSERVER im KeepRunning-Modus läuft oder die CT-API-Anwendung durch einen Programmabsturz beendet wird, kann CTSERVER mit dem Kommandozeilenparameter *-t* beendet werden. Wenn CTSERVER feststellt, daß Anwendungen mit dem B1-Chipkartenleser verbunden sind, erfolgt vor Beendigung eine Benutzerabfrage.

Hinweis: Beim mutwilligen Beenden von CTSERVER werden CT-API-Anwendungen vom B1-Chipkartenleser getrennt und können in weiterer Folge nicht mehr korrekt ablaufen. Das Beenden von CTSERVER sollte daher nur in Notfällen angewandt werden.

## 6.4 Windows 95

Die Dateien können z.B. mit dem Windows-Explorer in das System kopiert werden.

Installationsdiskette...	Verzeichnis des Zielsystems ...
\\WIN32\\B1TEST.EXE	beliebig
\\WIN32\\B1UPDATE.EXE	beliebig
\\WIN32\\CT32.DLL	Systemverzeichnis von Windows 95 (z.B.: C:\\WIN95\\SYSTEM)

Für den Ablauf von CT-API-Anwendungen wird die Datei *CT32.DLL* benötigt.

## 6.5 Windows NT

Die Dateien können z.B. mit dem Windows-Explorer in das System kopiert werden.

Installationsdiskette...	Verzeichnis des Zielsystems ...
WIN32\B1TEST.EXE	beliebig
WIN32\B1UPDATE.EXE	beliebig
WIN32\CT32.DLL	Systemverzeichnis von Windows NT (z.B.: C:\WINNT\SYSTEM32)

Für den Ablauf von CT-API-Anwendungen wird die Datei *CT32.DLL* benötigt.

## 6.6 SINIX/RM

Da die CT-API-Software für alle Betriebssysteme auf einer MS-DOS-Diskette geliefert wird, muß unter SINIX das Einspielen der Dateien in die Verzeichnisse mit dem Systemprogramm *doscpc* vorgenommen werden.

Installationsdiskette...	Verzeichnis des Zielsystems ...
\SINIX\b1test	beliebig
\SINIX\b1htsi.cfg	/etc

Die Datei *b1htsi.cfg* beschreibt die Zuordnung der von einer CT-API-Anwendung verwendeten Portnummer zu dem im Zielsystem vorhandenen Gerätenamen (Devicename).

Bei Erstellung bzw. bei Änderung dieser Datei sind folgende Richtlinien einzuhalten:

- ξ Jede Zeile beinhaltet eine Zuordnung
- ξ Jede Zeile muß das Format `<Portnummer> <Gerätename>` aufweisen. Der Eintrag `<Portnummer>` wird durch eine CT-API-Anwendung vorgegeben. Der zu dieser Portnummer zugehörige Gerätetreiber eintrag wird mit `<Gerätename>` festgelegt (incl. Pfadangabe).

**Beispiel:**

```
1 /dev/term/ser0
2 /dev/term/ser1
3 /dev/term/ser2
4 /dev/term/ser3
```

**Hinweis:** Für einen Hardware-Reset des B1-Chipkartenlesers ist das Ansteuern der V.24-Signalleitungen erforderlich. In älteren SINIX-Versionen (vor SINIX-N 5.42 C20) wird dies nicht unterstützt. In diesem Fall muß die Umgebungsvariable "B1RESETSIMULATION=1" gesetzt werden. Dies bewirkt einen ordnungsgemäßen Reset durch die Software, dessen Ergebnis von der Treibersoftware allerdings nicht zuverlässig festgestellt werden kann.

## 6.7 SUN-SOLARIS 2.5

Da die CT-API-Software für alle Betriebssysteme auf einer MS-DOS-Diskette geliefert wird, muß unter SOLARIS zunächst die Diskette gemountet werden:

```
$ mount -F pcfs /dev/fd0 /mnt
```

Das Einspielen der Dateien in die Verzeichnisse kann mit dem Befehl *cp* vorgenommen werden.

Installationsdiskette...	Verzeichnis des Zielsystems ...
\SOLARIS\b1test	beliebig
\SOLARIS\b1htsi.cfg	/etc

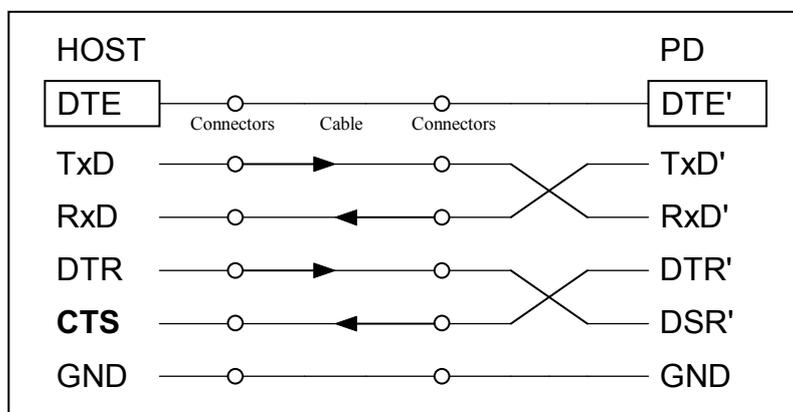
Die Datei *b1htsi.cfg* beschreibt die Zuordnung der von einer CT-API-Anwendung verwendeten Portnummer zu dem im Zielsystem vorhandenen Gerätenamen (Devicename).

Bei Erstellung bzw. bei Änderung dieser Datei sind folgende Richtlinien einzuhalten:

- ξ Jede Zeile beinhaltet eine Zuordnung
- ξ Jede Zeile muß das Format `<Portnummer> <Gerätename>` aufweisen. Der Eintrag `<Portnummer>` wird durch eine CT-API-Anwendung vorgegeben. Der zu dieser Portnummer zugehörige Gerätetreibertrag wird mit `<Gerätename>` festgelegt (incl. Pfadangabe).

Beispiel:  
 1 /dev/term/a  
 2 /dev/term/b

Hinweis: Je nach verwendeter SUN-Workstation sind nicht alle seriellen Ports im vollen Umfang belegt und ein angeschlossener Kartenleser ggf. nicht funktionstüchtig. Für weiterreichende Informationen wenden Sie sich bitte an Ihren Sun-Händler. Aufgrund eines von SUN bestätigten Fehlers im Betriebssystem Solaris wurde eine Modifikation des Anschlußkabels nötig. Das Kabel ist in diesem Fall wie folgt beschaltet:



## 6.8 LINUX

Da die CT-API-Software für alle Betriebssysteme auf einer MS-DOS-Diskette geliefert wird, muß unter LINUX zunächst die Diskette gemountet werden:

```
$ mount -t msdos /dev/fd0 /mnt
```

Das Einspielen der Dateien in die Verzeichnisse kann mit dem Befehl *cp* vorgenommen werden.

Installationsdiskette...	Verzeichnis des Zielsystems ...
\LINUX\b1test	beliebig
\LINUX\b1htsi.cfg	/etc
\LINUX\Lib\ctapi.h	.../include
\LINUX\Lib\libb1ct.a	.../lib

Die Treiber liegen in Form von Objekten vor, die zu den Programmen, welche auf den B1-Chipkartenleser zugreifen, hinzugebunden werden müssen. Es handelt sich somit nicht um Betriebssystemerweiterungen.

Die Datei *b1htsi.cfg* beschreibt die Zuordnung der von einer CT-API-Anwendung verwendeten Portnummer zu dem im Zielsystem vorhandenen Gerätenamen (Devicename).

Bei Erstellung bzw. bei Änderung dieser Datei sind folgende Richtlinien einzuhalten:

- ξ Jede Zeile beinhaltet eine Zuordnung
- ξ Jede Zeile muß das Format `<Portnummer> <Gerätename>` aufweisen. Der Eintrag `<Portnummer>` wird durch eine CT-API-Anwendung vorgegeben. Der zu dieser Portnummer zugehörige Gerätetreibereintrag wird mit `<Gerätename>` festgelegt (incl. Pfadangabe).

Beispiel: 1 /dev/ttyS0  
2 /dev/ttyS1

## 6.9 UNIX BSD

Die Treiber zum Betrieb des B1-Chipkartenlesers unter dem Betriebssystem UNIX BSD gehören nicht zum Standardlieferumfang. Sie sind über Herrn Klaus Bühler, Tel. 0621/429-1942, Email-Adresse [klaus.buehler@rm.sni.de](mailto:klaus.buehler@rm.sni.de) zu beziehen.

## 6.10 FLEXOS

Die Treiber zum Betrieb des B1-Chipkartenlesers unter dem Betriebssystem FLEXOS gehören nicht zum Standardlieferumfang. Sie sind über Herrn Klaus Bühler, Tel. 0621/429-1942, Email-Adresse [klaus.buehler@rm.sni.de](mailto:klaus.buehler@rm.sni.de) zu beziehen.

## 7 Funktionstest mit dem Testprogramm

Auf der Diskette der B1-Chipkartenleser-Software finden Sie für die Microsoft-Betriebssysteme und SINIX/RM das Testprogramm *B1TEST*. Mit diesem Testprogramm kann die Konfiguration und Betriebsbereitschaft der Software-Komponenten und des B1-Chipkartenlesers überprüft werden. Mit einer gesondert erhältlichen Testchipkarte kann die Kommunikation getestet werden.

Voraussetzung für den einwandfreien Betrieb des B1-Chipkartenlesers und des Testprogrammes ist die Installation der Host-Software (siehe Kapitel 6).

Je nach Betriebssystem ist *B1TEST* mit textorientierter (MS-DOS und SINIX/RM) oder grafischer Benutzeroberfläche (Windows-Varianten) ausgestattet. Der Leistungsumfang ist in beiden Varianten gleich und umfaßt folgende Punkte:

### ξ Porteinstellung

Vor dem Start einer Testfunktion muß der Port ausgewählt werden, an dem der zu testende B1-Chipkartenleser angeschlossen ist. Diese Auswahl kann entweder direkt vorgenommen werden, oder aber Sie überlassen das Auffinden vorhandener B1-Chipkartenleser dem Testprogramm (*Automatische Suche*).

Hinweis: Das Testprogramm unterstützt die Ports 1 bis 4. Unter SINIX muß die Datei *b1htsi.cfg* entsprechend vorbereitet werden (siehe Kapitel 6.6).

### ξ B1-Dateisystem lesen

Der B1-Chipkartenleser besitzt ein hierarchisches Dateisystem, dessen Dateien Konfigurations- und Stauseinstellungen beinhalten. Die in den Dateien enthaltene Information ist binär nach dem TLV-Verfahren (*Tag-Length-Value*) abgelegt. *B1TEST* bietet Ihnen die Möglichkeit, den Inhalt dieser Dateien textuell darzustellen.

### ξ Chipkarte testen

Eine weitere Funktion von *B1TEST* gestattet das Speichern von Testdaten auf die Testchipkarte, darauffolgend das Einlesen der geschriebenen Daten und eine abschließende Kontrolle der Testdaten.

## 8 Update der Chipkartenleser-Firmware

Der B1-Chipkartenleser ist mit einer Firmware Update-Funktion ausgestattet. Mit dem dazugehörigen Update-Programm *B1UPDATE* ist es möglich, die bestehende Firmware im B1-Chipkartenleser durch eine neue Version zu ersetzen.

Die neue Firmware wird in Form einer Update-Datei repräsentiert, die vom Update-Programm gelesen und über ein gesichertes Protokoll (T=1) zum B1-Chipkartenleser übertragen wird. Der Inhalt der Update-Datei ist mit dem Data Encryption Standard (DES) Verfahren verschlüsselt, d.h. für einen Firmware Update kann nur eine vom Hersteller erzeugte Update-Datei verwendet werden.

Das Update-Programm ist für die Betriebssysteme MS-DOS sowie für die 32-Bit-Windows-Plattformen Windows NT und Windows 95 verfügbar. Wenn Sie Windows 3.1x einsetzen, bedienen Sie sich bitte des Update-Programms für MS-DOS. Sollten Sie den B1-Chipkartenleser mit einem anderen Betriebssystem einsetzen, schließen

Sie den B1-Chipkartenleser vorübergehend an einen PC mit einem der oben genannten Betriebssysteme an und führen Sie dort den Firmware Update durch.

Ein Firmware Update kann nur bei einem funktionsfähigen B1-Chipkartenleser durchgeführt werden. Die Funktionsfähigkeit des B1-Chipkartenlesers kann mit dem Testprogramm *B1TEST* überprüft werden (siehe Kapitel 7).

Vor dem Start von *B1UPDATE* führen Sie bitte folgende Schritte durch:

- ξ Beenden Sie alle Programme, die mit dem B1-Chipkartenleser arbeiten (bei Windows-Varianten und bei SINIX/RM)
- ξ Entfernen Sie eine eventuell gesteckte Chipkarte
- ξ Schalten Sie den B1-Chipkartenleser ein (AT-Version)
- ξ Halten Sie die Diskette mit der Firmware Update-Datei bereit

Starten Sie das Programm *B1UPDATE* und folgen Sie den Bedienerführungstexten:

- ξ Geben Sie den Namen der Update-Datei ein (z.B.: *a:b1v4v13.upd*)  
*B1UPDATE* überprüft die eingegebene Update-Datei und zeigt deren Namen an.
- ξ Auswahl der seriellen Schnittstelle  
Das Update-Programm benötigt nun die Information, an welcher seriellen Schnittstelle der B1-Chipkartenleser an Ihrem PC angeschlossen ist. Wenn Ihnen dies bekannt ist, geben Sie die Nummer der COM-Schnittstelle ein (z.B. 2 für COM2), anderenfalls kann das Update-Programm selbsttätig nach einem B1-Chipkartenleser suchen. Wenn bei der automatischen Suche mehr als ein B1-Chipkartenleser gefunden wurde, wählen Sie das gewünschte Gerät aus. Das Update-Programm gibt nach einer gültigen Auswahl die Version der aktuellen Firmware aus.
- ξ Starten des Firmware Update-Vorganges  
Nachdem die Update-Datei und die zu verwendende Schnittstelle erfolgreich angegeben worden sind, kann der Firmware Update-Vorgang gestartet werden.

### **WICHTIG:**

Beachten Sie während des Firmware Update-Vorganges unbedingt folgende Punkte:

- ⇒ **Der Firmware Update kann mehrere Minuten dauern.**
- ⇒ **Schalten Sie während des Update-Vorganges keinesfalls den PC oder den B1-Chipkartenleser ab. Dies könnte die Funktionsunfähigkeit des B1-Chipkartenlesers bewirken.**
- ⇒ **Versuchen Sie nicht, das laufende Update-Programm abubrechen.**
- ⇒ **Starten Sie keine anderen Anwendungen, die das B1-Chipkarten-Terminal nutzen.**

*B1UPDATE* informiert Sie abschließend über die Beendigung des Firmware Update-Vorganges und zeigt die Versionsangaben der neuen Firmware an.

Der B1-Chipkartenleser kann sofort nach dem Verlassen des Update-Programmes wieder benutzt werden. Ein Neustart des Systems oder andere Aktivitäten sind nicht erforderlich.

## 9 Programmentwicklung für den B1-Chipkartenleser

Dieses Kapitel richtet sich an Softwareentwickler, die C- oder C++-Programme für den B1-Chipkartenleser auf Basis der CT-API 1.1 entwickeln.

Folgende Voraussetzungen sind für die Programmentwicklung nötig:

- ξ Die Einrichtung der CT-API-Entwicklungsdateien für Ihre Zielsystem.
- ξ CT-API 1.1 Spezifikation
- ξ Das Programmierhandbuch (B0/B1 HTSI) für den B1-Chipkartenleser

### 9.1 Ansteuerung des B1-Chipkartenlesers

Die Aufbereitung der notwendigen Befehle sowie die Auswertung von Fehlermeldungen obliegen dem Anwendungsprogrammierer. Das API des B1-Chipkartenleser ist an ISO 7816-4/5 angelehnt.

Bei asynchronen Chipkarten werden Applikationsbefehle an die Chipkarten weitergeleitet. Die Daten synchroner Chipkarten stellt der B1-Chipkartenleser als Datei dar. Diese Daten kann die Anwendersoftware mit File-Befehlen von der Chipkarte lesen bzw. auf die Chipkarte schreiben.

Zur Zeit werden folgende Baudraten unterstützt:

9.600 Baud
19.200 Baud
28.800 Baud
38.400 Baud
57.600 Baud
115.200 Baud

Für weitere Informationen stehen die CT-API 1.1 Spezifikation und das Programmierhandbuch (B0/B1 HTSI) zur Verfügung.

### 9.2 Einrichten der Entwicklungsdateien

Auf der Diskette der B1-Software finden Sie im Unterverzeichnis '...\LIB' je Hostplattform die dazu benötigten Dateien. Wenn Sie z.B. ein CT-API-Programm für Windows 95 entwickeln, kopieren Sie alle Dateien aus dem Verzeichnis \WIN32\LIB der Diskette in ein frei wählbares Entwicklungsverzeichnis auf Ihrem PC.

### 9.3 Include-Datei CTAPI.H

In dieser Include-Datei sind die Returncodes und die Funktionsprototypen gemäß CT-API 1.1 spezifiziert. Diese Datei muß in alle C-Programmdateien inkludiert werden, die CT-API-Funktionen einsetzen.

### 9.4 Allgemeine Hinweise zur Verwendung des CT-API

Zur Initialisierung der CT-API muß der Funktion `CT_init()` eine sogenannte Card Terminal Number (ctn) und eine Port Number (pn) übergeben werden. Im HTSI für den B1-Chipkartenleser werden diese Parameter wie folgt interpretiert.

### 9.4.1 Card Terminal Number (ctn)

Anhand dieser Nummer identifiziert das HTSI eine CT-API-Sitzung. Die Card Terminal Number ist innerhalb des im CT-API 1.1 festgelegten Wertebereichs frei wählbar. Bei nachfolgenden *CT\_data()*-Aufrufen und bei einem abschließenden *CT\_close()* muß dieselbe Card Terminal Number übergeben werden.

Beim Betrieb des B1-Chipkartenlesers auf Multitasking-Plattformen muß *ctn* innerhalb eines Prozesses eindeutig sein. Wenn also mehrere B1-Chipkartenleser mit einem Programm gleichzeitig betrieben werden, muß für jede Sitzung eine unterschiedliche *ctn* gewählt werden. Werden die einzelnen B1-Chipkartenleser von separaten Programmen bedient, darf die selbe Card Terminal Number mehrfach verwendet werden.

Regel:

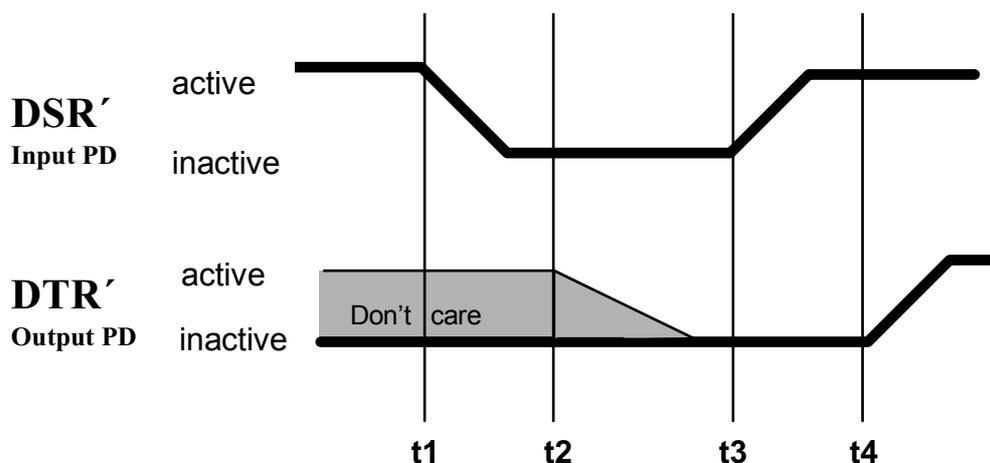
- ξ innerhalb eines Programms muß *ctn* eindeutig sein
- ξ innerhalb des Systems darf die selbe *ctn* mehrfach verwendet werden.

### 9.4.2 Port Number (pn)

Die Port Number bezeichnet die serielle Schnittstelle, an der der B1-Chipkartenleser angeschlossen ist. Für MS-DOS und den Windows-Plattformen ist damit COMx bezeichnet. Wenn ein B1-Chipkartenleser an COM2 angeschlossen ist, muß *pn* entsprechend mit dem Wert 2 definiert werden.

Unter SINIX lokalisiert *pn* einen Eintrag in der Konfigurationsdatei *b1htsi.cfg*, in der der Name für den seriellen Gerätetreiber steht. Weiterführende Informationen entnehmen Sie bitte dem Kapitel 6.6.

## 9.5 Schicht 1 Reset-Mechanismus



Die Reset-Prozedur ist wie folgt definiert:

- ξ Rücksetzen des PD in den Ausgangszustand (power-on)
- ξ implizites Rücksetzen der gesamten PD (Leser)-Umgebung (z.B. incl. ICCs)
- ξ Rücksetzen des PD (Leser)-Treibers am Host in den Ausgangszustand (power-on)
- ξ Informieren der Host-Applikation über die Reset-Prozedur zum Zeitpunkt
  - ξ des Prozedurstarts
  - ξ des Prozedurendes

Die Reset-Prozedur wird einzig durch die DTR- und DSR- Leitung kontrolliert und kann nur vom Host initiiert werden. Wie auf der obenstehenden Abbildung zu sehen startet der Host die Reset-Prozedur des PDs (Leser), indem er seinen DTR-Output deaktiviert. Reset kann ebenfalls initiiert werden, während Host und PD (Leser) im Power-on-Zustand sind. Nachdem der PD (Leser)-Treiber am Host rückgesetzt wurde, stößt der Host den Reset am PD (Leser) an. Die Zeitspanne  $t_1-t_2$  sollte zwischen 20 ms und 50 ms betragen, zwischen  $t_3$  und  $t_4$  sollten nicht mehr als 5 Sekunden verstreichen.

## 9.6 Zusätzliche Festlegungen zur Schicht 2

Im folgenden Kapitel werden Abweichungen des Schicht 2-Protokolls von der ihm zu zugrunde liegenden Norm ISO/IEC 7816-3 Amd. 1, Clause 9) beschrieben.

### ***Rollenverteilung***

Im Host-Leser-Protokoll übernimmt der Host die Rolle des IFD aus ISO/IEC 7816-3 Amd. 1, Clause 9. Der Leser übernimmt die Rolle der ICC.

### ***Defaultwerte für den Zeichen-Rahmen***

Für den Zeichen-Rahmen gelten die folgenden Defaultwerte:

- 9.600 Bit/s
- 8 Datenbits
- 2 Stopbits
- Gerade Parität

Das niederwertigste Bit wird zuerst übertragen. Diese Werte gelten bei Beginn des Blockprotokolls.

### ***Veränderung des Zeichen-Rahmens***

Außer den Defaultwerten können der Host und der Leser zusätzliche Werte für die Baudrate, die Anzahl der Stopbits und die Parität unterstützen. Der Leser zeigt die von ihm zusätzlich unterstützten Werte in der Configuration File des Lesers an. Neue Werte für den Zeichen-Rahmen können durch einen Schicht-7-Befehl vereinbart werden.

## **Block-Rahmen**

Der Block-Rahmen entspricht ISO/IEC 7816-3 Amd. 1, Clause 9.3.

### **Error Detection Code**

Defaultverfahren für die Fehlererkennung ist LRC. Zusätzlich können Host und Leser CRC unterstützen. Der Leser zeigt die von ihm unterstützten EDC-Verfahren in der Configuration File des Lesers an. Die Änderung des verwendeten CRC-Verfahrens kann durch einen Schicht-7-Befehl vereinbart werden.

### **Spezielle Schnittstellen-Parameter**

#### Information Field Size (IFS)

Für den Host gilt als Defaultwert IFSD = 32. Unterstützt der Host ein größeres IFSD, kann er dieses während des Protokolls mit einem S(IFSrequest) einstellen.

Für den Leser gilt als Mindestwert IFSC = 32. Der Leser kann ein größeres IFSC als Defaultwert in der Configuration File des Lesers anzeigen. IFSC kann ebenfalls während des Protokolls durch einen S(IFSrequest) geändert werden.

#### Character Waiting Time (CWT)

Zu Beginn des Protokolls gilt die Initial Character Waiting Time (ICWT). ICWT beträgt 100 ms. Der Leser kann unterschiedliche CWTs unterstützen. Ist dies der Fall, wird es vom Leser in der Configuration File des Lesers angezeigt. Eine Änderung der CWT wird durch einen Schicht-7-Befehl vereinbart.

#### Block Waiting Time (BWT)

Zu Beginn des Protokolls gilt die Initial Block Waiting Time (IBWT). IBWT beträgt 500 ms. Der Leser kann unterschiedliche BWTs unterstützen. Ist dies der Fall, wird es vom Leser in der Configuration File des Lesers angezeigt. Eine Änderung der BWT wird durch einen Schicht-7-Befehl vereinbart.

Hinweis: Während des Protokolls kann BWT für einzelne Kommandos vom Leser durch einen S(WTXrequest) geändert werden.

#### Block Guard Time (BGT)

BGT beträgt 0 ms. Das bedeutet, direkt nach Empfang des letzten Zeichens eines Blocks kann mit dem Senden eines Blocks in die Gegenrichtung begonnen werden.

## **Protokollablauf**

### Beginn des Protokolls

Der Host beginnt das Blockprotokoll direkt nach Abwicklung des Schicht-7-Reset.

### VPP Control

VPP Control (vgl. ISO/IEC 7816-3 Amd. 1, Clause 9.6.1.1) wird nicht unterstützt. Die Bits b8 und b4 im NAD Byte werden beim Senden auf 0 gesetzt, beim Empfang werden diese Bits ignoriert.

## Chaining

Chaining wird unterstützt.

## S-Blöcke

Die VPP Error Response S-Blöcke (vgl. ISO/IEC 7816-3 Amd. 1, Clause 9.6.2.4.3) werden nicht unterstützt. Der Empfang eines solchen Blocks wird wie der Empfang eines ungültigen Blocks behandelt.

## **Verwendung der Adressbytes (NAD)**

### NAD Byte beim Senden

Beim Senden von I-Blöcken enthält das NAD Byte die von der Applikationsschicht generierten Transportadressen. Bei verketteten I-Blöcken ist das NAD Byte in allen I-Blöcken der Kette gleich.

Beim Senden von S- und R-Blöcken werden im NAD Byte die Adressen des zuletzt empfangenen Blocks gespiegelt. War der zuletzt empfangene Block ungültig (Time-out, Paritätsfehler, falsche Prüfsumme, usw.), wird das NAD Byte auf 0 gesetzt.

### NAD Byte beim Empfangen

Beim Empfang von I-Blöcken werden die Adressen des NAD Bytes als Transportadressen interpretiert und an die Applikationsschicht weitergeleitet. Bei verketteten I-Blöcken wird nur das NAD Byte des letzten I-Blockes der Kette ausgewertet.

Beim Empfang von S- und R-Blöcken wird das NAD Byte nicht ausgewertet.

## **9.7 Bedienung von mehreren B1-Chipkartenlesern mit CT-API**

Auf den verschiedenen Plattformen (ausgenommen MS-DOS) können mit Hilfe des CT-API mehrere im System vorhandene B1-Chipkartenleser angesprochen werden. Zu einem Zeitpunkt kann aber nur eine Anwendung die Kommunikation mit einem Chipkartenleser betreiben, d.h. der Chipkartenleser wird exklusiv von einer Anwendung reserviert. Während einer Sitzung werden CT\_init()-Aufrufe von anderen Anwendungen mit derselben Port Number mit ERR\_INVALID abgelehnt.

## **9.8 Entwicklung unter MS-DOS**

Für MS-DOS ist die gesamte Host-Software des B1-Chipkartenlesers in der CT-API-Bibliothek enthalten. Diese wickelt das Protokoll zum B1-Chipkartenleser unter direkter Ansteuerung der COM-Schnittstellen ab. Eine Treiberinstallation ist nicht notwendig, d.h. das erzeugte EXE-Programm kann ohne zusätzliche Software ablaufen.

Abhängig von Compiler und Speichermodell müssen Sie eine Bibliothek zu Ihrem Programm binden.

Compiler: MS=Microsoft (Visual C++), BC=Borland  
Version: Bezeichnet die Compiler-Version, z.B. 15 für Visual C++ 1.5.

Speichermodell: Bezeichnet das Speichermodell, in dem die Bibliothek erstellt ist. Das Speichermodell Ihrer Anwendung und das der CT-API-Bibliothek müssen übereinstimmen.

Die Dateinamen der Bibliotheken sind nach folgender Namenskonvention aufgebaut.  
CT{Compiler}{Version}{Speichermodell}.LIB

Bitte beachten Sie bei der Programmerstellung folgende Hinweise:

### **Vermeiden von Systeminstabilitäten**

Die Kommunikation mit der seriellen Schnittstelle wird innerhalb der Bibliothek interruptbetrieben abgewickelt. Die dazu benötigten Systemressourcen werden mit der CT-API-Funktion *CT\_init()* belegt und mit bei einem *CT\_close()* - Aufruf wieder freigegeben.

Es muß daher darauf geachtet werden, daß vor dem Programmende *CT\_close()* aufgerufen wird, anderenfalls kann dies zu Systeminstabilitäten bzw. Systemabsturz führen.

Um einen undefinierten Programmabbruch mit <Ctrl-C> bzw. <Ctrl-Untbr> zu vermeiden, setzt *CT\_init()* das Signal SIGINT auf SIG\_IGN (siehe dazu Runtime-library-Funktion *signal()*). Wenn dies unerwünscht ist, weil das Programm selbst das Abbruch-Verhalten festlegt, muß nach einem *CT\_init()*-Aufruf der Signal-Handler vom Programm gesetzt werden.

### **Verwendung der CT-API-Bibliothek in Windows-Programmen**

Aus technischer Sicht können die DOS-Bibliotheken auch mit Windows-Programmen gebunden werden. Es wird jedoch empfohlen, bei der Entwicklung von Windows-Programmen die Import-Bibliothek *CT.LIB* gemeinsam mit der dynamischen Bibliothek *CT.DLL* zu verwenden.

## **9.9 Entwicklung unter Windows 3.1x**

Unter Windows ist die Host-Software zur Ansteuerung des B1-Chipkartenlesers in der dynamischen Bibliothek *CT.DLL* untergebracht. Die Nutzung dieser Bibliothek durch den Entwickler erfolgt durch die unter Windows geeigneten Methoden:

- ξ Implizite Bindung mit *LoadLibrary()* und *GetProcAddress()*, siehe Windows SDK
- ξ Explizite Bindung mit der Importbibliothek *CT.LIB*

Für den Einsatz auf dem Zielsystem ist es notwendig, die Dateien *CT.DLL* und *CTSERVER.EXE* zu installieren (siehe Kapitel 6.3).

### **Beachten Sie bitte folgende Hinweise**

Um Systemblockaden bei zeitintensiven *CT\_data()*-Aufrufen zu vermeiden, läuft während der Bearbeitung der *CT\_data()*-Funktion eine Message-Loop. Daher können vom Programm *CT\_data()*-Aufrufe in Serie abgesetzt werden, ohne andere Windows-Anwendungen im System blockieren. Diese Technik gestattet einen linearen Pro-

grammablauf, der keine Rücksicht auf die cooperativ-multitasking-Eigenschaften von Windows nehmen muß.

Damit Rekursionen im CT-API vermieden werden (durch die MessageLoop in *CT\_data()* könnte die rufende Anwendung wieder die Kontrolle erhalten), wird das Window der rufenden Anwendung für Benutzereingaben gesperrt.

Während der Abwicklung eines *CT\_data()*-Aufrufes wird ein *CT\_data()*-Aufruf einer anderen Anwendung mit ERR\_HTSI abgewiesen.

### **CT-Applikationsbefehl Set Interface Parameter**

Mit dem CT-Applikationsbefehl 'Set Interface Parameter' können Änderungen des Schnittstellenprotokolls und der Parameter vorgenommen werden. Das B1 Chipkarten-Terminal kann unter Windows 3.1x die Übertragungsrate von 115200 baud nicht unterstützen.

## **9.10 Entwicklung unter WIN32 (Windows 95 und Windows NT)**

Für die Programmentwicklung unter Windows 95 und Windows NT wird die dynamische Bibliothek *CT32.DLL* zur Laufzeit des Programms gebunden. Wie unter Windows 3.1x bedient sich das Programm einer der beiden folgenden Methoden:

- ξ Implizite Bindung mit *LoadLibrary()* und *GetProcAddress()*, siehe Windows SDK
- ξ Explizite Bindung mit der Importbibliothek *CT.LIB*

Für den Einsatz auf dem Zielsystem ist es notwendig, die Datei *CT32.DLL* zu installieren (siehe Kapitel 6.4 bzw. 6.5).

### **9.10.1 Freeze-Befehl**

Das B1 Chipkarten-Terminal unterstützt den speziellen CT-Applikationsbefehl 'Freeze', welcher es einer Anwendung ermöglicht, auf bestimmte Ereignisse (z.B. die Entnahme der Chipkarte) zu warten. Um ein Freeze-Ereignis in einer Anwendung zu erhalten, wurde der HTSI-Befehl 'Wait Freeze' implementiert (nur WIN32).

Im Gegensatz zu anderen Befehlen blockiert *CT\_data (.. 'Wait Freeze' ..)* den aufrufenden Programmfaden. Besitzt die Anwendung nur einen Faden, so blockiert der gesamte Prozeß. Da dies selten gewünscht ist, sollte *CT\_data (.. 'Wait Freeze' ..)* in einem separaten Faden ausgeführt werden.

Wird in einem Multithread-Prozeß *CT\_close()* aufgerufen, während ein *CT\_data (.. 'Wait Freeze' ..)* in Bearbeitung ist, wird *CT\_data()* mit dem Returncode ERR\_HTSI beendet.

Nachdem ein 'Freeze'-Befehl erfolgreich abgesetzt worden ist, befindet sich das CT im Freeze-Zustand. In weiterer Folge wird von der Anwendung ein 'Wait Freeze'-Befehl gesendet. 'Wait Freeze' kehrt erst wieder zurück, nachdem das mit 'Freeze' angegebene Ereignis eingetreten bzw. ein Fehler aufgetreten ist.

### Kodierung

CLA	'80'
INS	'71'
P1	Timeout in Minuten
P2	Timeout in Sekunden
L <sub>c</sub>	leer
Datenfeld	leer
L <sub>e</sub>	leer

Wird in P1 und P2 der Wert '00' angegeben, wird keine Zeitüberwachung durchgeführt, 'Wait Freeze' kehrt erst nach Eintreffen des Freeze-Ereignisses bzw. bei einem Fehler zurück.

### Spezielle Statusanzeige

SW1-SW2	Bedeutung	mögliche Ursachen
'62F0'	Timeout	Wait Freeze wurde durch ein Timeout beendet, das Freeze-Ereignis wurde nicht empfangen.
'64F0'	DSR bei Aufruf bereits gesetzt	Das CT befindet sich nicht in einem Freeze-Zustand

Statusanzeigen beim 'Wait Freeze'-Befehl

Beim Empfang von SW1-SW2 = '9000' wurde der Wechsel der DSR-Leitung von LO auf HI empfangen. In jedem Fall sollte die Anwendung das Freeze-Status-File bzw. das ICC-Status-File auslesen. Die Signalleitung könnte durch die Aktivität einer anderen Anwendung gesetzt worden sein. In diesem Fall traf das Freeze-Ereignis nicht ein, obwohl DSR gesetzt wurde.

Nach Auslesen der ICC-Status-Files des B1-CT muß die Anwendung ggf. 'Freeze' und 'Wait-Freeze' erneut absetzen.

Mehrere FREEZE-Ereignisse können gleichzeitig aktiv sein.

#### 9.10.2 CT-Applikationsbefehl Set Interface Parameter

Unter Windows NT und Windows 95 ist beim B1 Chipkarten-Terminal die Übertragungsrate von 115.200 Baud nicht implementiert.

#### 9.11 Entwicklung unter SINIX/RM

Unter SINIX muß das CT-API-Programm mit der Bibliothek *libb1ct.a* gebunden werden.

Für die bei dem Befehl *CT\_init()* übergebene Port Number *pn* muß auf dem Zielsystem ein entsprechender Eintrag in der Datei *b1htsi.cfg* vorgenommen werden (siehe Kapitel 6.6). Für die Entwicklung bedeutet dies, daß eine beliebige Port Number im Rahmen von CT-API 1.1 gewählt werden kann. Mit Hilfe der Datei *b1htsi.cfg* wird dieser Port Number der Devicename einer seriellen Schnittstelle zugeordnet.

Unter SINIX/RM ist beim B1 Chipkarten-Terminal die Übertragungsrate von 115.200 Baud nicht implementiert.

### 9.12 Entwicklung unter SUN-SOLARIS 2.5

Unter SOLARIS muß das CT-API-Programm mit der Bibliothek *libb1ct.a* gebunden werden.

Für die bei dem Befehl *CT\_init()* übergebene Port Number *pn* muß auf dem Zielsystem ein entsprechender Eintrag in der Datei *b1htsi.cfg* vorgenommen werden (siehe Kapitel 6.7). Für die Entwicklung bedeutet dies, daß eine beliebige Port Number im Rahmen von CT-API 1.1 gewählt werden kann. Mit Hilfe der Datei *b1htsi.cfg* wird dieser Port Number der Devicename einer seriellen Schnittstelle zugeordnet.

Des weiteren wurde auch hier der HTSI-Befehl 'Wait Freeze' implementiert, wie er bereits in Kapitel 9.10.1 erläutert wurde.

### 9.13 Entwicklung unter LINUX

Unter LINUX muß das CT-API-Programm mit der Bibliothek *libb1ct.a* gebunden werden.

Für die bei dem Befehl *CT\_init()* übergebene Port Number *pn* muß auf dem Zielsystem ein entsprechender Eintrag in der Datei *b1htsi.cfg* vorgenommen werden (siehe Kapitel 6.8). Für die Entwicklung bedeutet dies, daß eine beliebige Port Number im Rahmen von CT-API 1.1 gewählt werden kann. Mit Hilfe der Datei *b1htsi.cfg* wird dieser Port Number der Devicename einer seriellen Schnittstelle zugeordnet.

## 10 Betrieb mit KV-Anwendungen

Der B1-Chipkartenleser ist kompatibel mit den Kartenterminals im Gesundheitswesen und kann daher mit bestehenden KV-Anwendungen betrieben werden. Im Unterschied zu anderen Kartenterminals ist beim B1-Chipkartenleser ein definierter Hardware-Reset erforderlich, um das Terminal betriebsbereit zu schalten. Dies geschieht durch Setzen des Pegels der DTR-Leitung der V.24-Schnittstelle.

Da KV-Anwendungen diese Signalleitung im Normalfall nicht bedienen, muß der B1-Chipkartenleser mittels eines mitgelieferten Hilfsprogrammes vor der Benutzung einer KV-Anwendung initialisiert werden.

Auf der Installationsdiskette finden Sie im Verzeichnis \DOS das Programm *SETDTR.EXE*, welches in ein beliebiges Verzeichnis auf die Festplatte kopiert wird (z.B. C:\). Der Aufruf von *SETDTR* sollte vorzugsweise in der Datei *AUTOEXEC.BAT*

eingetragen werden, damit bei jedem Systemstart der B1-Chipkartenleser automatisch initialisiert wird.

SETDTR muß als Parameter die Nummer der seriellen Schnittstelle übergeben werden, an der der B1-Chipkartenleser angeschlossen ist.

Wenn der B1-Chipkartenleser zum Beispiel an COM2 des PCs angeschlossen ist, muß in der Datei AUTOEXEC.BAT der Eintrag

```
SETDTR 2
```

vorgenommen werden.

## **11 Chipkarten-Protokolle**

### **11.1 Automatische Protokollerkennung**

Beim Reset-Befehl erkennt der B1-Chipkartenleser die Art der Chipkarte und das eingesetzte Protokoll. Voraussetzung dafür ist:

ξ bei asynchronen Chipkarten:

ATR nach ISO 7816-3, auch ATR mit Teilverhältnis 512 wird erkannt.

ξ bei synchronen Chipkarten:

ATR nach ISO 7816-3 oder I<sup>2</sup>C-Bus Karten ATR in den ersten 4 Bytes.

Die Protokollanzeige ist noch nicht von der ISO standardisiert; sie entspricht der Spezifikation der deutschen Krankenversicherten-Karte. Sind die Voraussetzungen nicht erfüllt, muß der Anwender den Protokollparameter selbst einstellen.

### **11.2 Protocol Type Selection (PTS)**

Den PTS nach ISO 7816-3 unterstützt der B1-Chipkartenleser. Insbesondere sind Teilverhältnisse möglich, die Baudraten bis 20 kb/s zulassen.

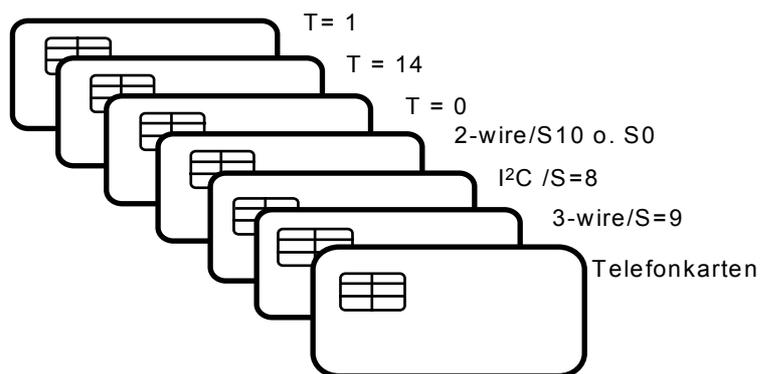
## **12 Lieferumfang**

Der Lieferumfang ist der jeweils gültigen Preisliste zu entnehmen.

### 13 Datenblatt B1-Chipkartenleser-AT (Aufschleser)



Der B1-Chipkartenleser verarbeitet diese Speicher- /Prozessor-Chipkarten



über die einheitliche Applikationsschnittstelle CT-API

### 13.1 Technische Daten

#### Unterstützte Datenübertragungsprotokolle für Chipkarten :

<u>Protokoll</u>	<u>Referenz</u>
<i>Asynchron:</i>	
T=0	ISO 7816-3
T=1	ISO 7816-3
T=14	FTZ 1 TR 15, der gemeinsame Vorschlag der GZS und der Deutschen Telekom AG für ein nationales, asynchrones Blockübertragungsprotokoll
<i>Synchron:</i>	
2-Wire-Protokoll, auch S10 oder S0	Siemens Datenblatt SLE 4432 und SLE 4442, Philips Datenblatt PCB 2032/42
3-Wire-Protokoll	Siemens Datenblatt SLE 4418 und SLE 4428
I <sup>2</sup> C-Bus	Philips Datenblatt PCF 8582E-2
Telefonkarten	Separate Beschreibung notwendig

- Prozessor: 80C32 mit 12 MHz getaktet
  - Smartcard Coupler: Philips TDA 8000
  - Firmware (im Lieferumfang): kryptographisch gesichert fernladbar
  - RAM-Speicher: 32 k \* 8 statisches RAM
  - Programmspeicher: 32 k \* 8 Flash-EPROM
  - Anzeige: 2 LEDs
  - Kontaktiereinheit: Absenkkontakte, 200.000 Steckzyklen
  - Clock zum Betrieb der Karte: 4,9152 MHz
  - Stromversorgung: Steckernetzteil mit 9 VAC/300 mA
  - Ein-/Aus-Schalter: Ja
  - Schnittstelle zum Hostrechner: V.24 mit 8-poligem Western Plug
  - Protokoll zum Hostrechner: T=1 nach ISO 7816-3, Amd. 1 "T=1"
  - Gehäusematerial: ABS-Kunststoff
  - Gehäuse Abmessung: 75 mm (H), Durchmesser ca. 180 mm
  - Gewicht: 0,650 kp
  - Betriebsumgebung: Klasse 3.1, Büroklimatisierung, 0 °-50° C
  - Treiber-SW und CT-API verfügbar für: MS-DOS™, Win 3.11™, Win 95™, Win NT™, SINIX, SUN Solaris™ 2.5.1, LINUX™ 2.029
  - Applikationsschnittstelle: CT-API gemäß Dt. Telekom AG, RWTÜV, TeleTrust und GMD
- alternativ
- PC/SmartCard-Architektur: B1-IFD-Handler, CT-API Service Provider
  - Funktionsbibliothek für: Microsoft, Borland
  - Konformität: ISO 7816-2, -3, B1, Krankenversichertenkarte wird automatisch erkannt

Technische Änderungen vorbehalten

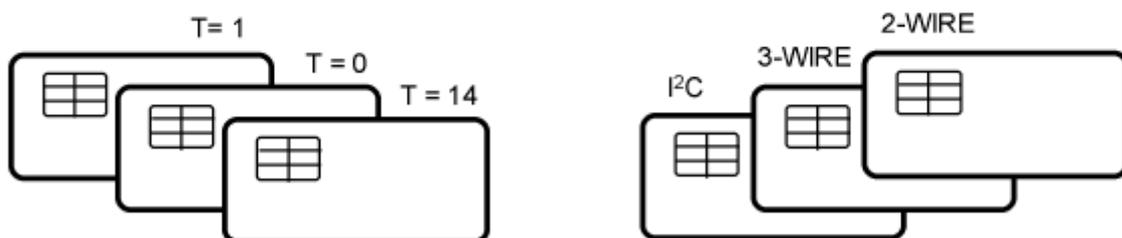
Stand: 01.04.1998

## 14 Datenblatt B1-Chipkartenleser (PC Card Leser)



<b>Produkt Highlights</b>	<ul style="list-style-type: none"><li>• PC Card Type II extended</li><li>• hoch integrierte 2 Chip-Lösung</li><li>• Leser erkennt, wenn eine Chipkarte eingeschoben wird.</li><li>• Flash Memory für Firmware Update</li><li>• Stahlgehäuse</li><li>• CE geprüft</li><li>• kompatibel zu Windows 95™, MS-DOS™, Windows 3.11, WIN NT™ 4.0</li><li>• <b>CT-API Software Interface für Chipkarten-Anwendungen</b></li></ul>
---------------------------	--

Der Chipkarten-Leser B1 in der Ausführung PC Card verarbeitet derzeit diese Protokolle:



über die einheitliche Applikations-Schnittstelle CT-API.

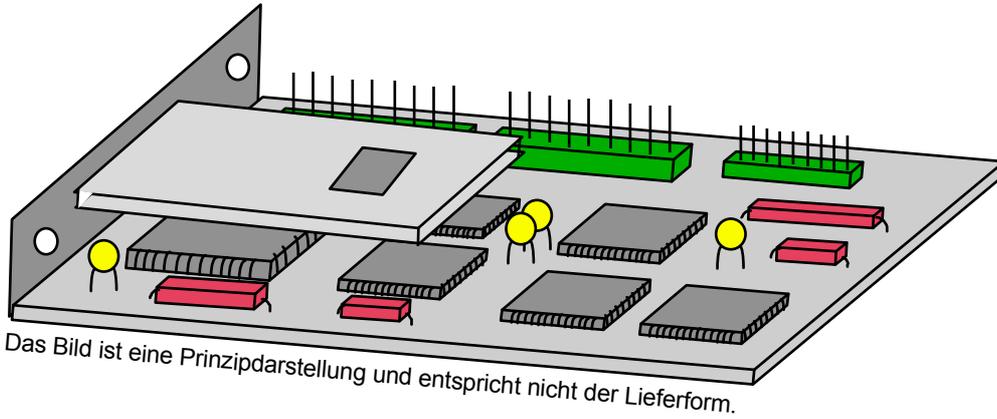
## 14.1 Technische Daten

<b>Hardware</b>	<ul style="list-style-type: none"> <li>• CPU 8-Bit, single-Chip 80C32 Microcontroller</li> <li>• Schnelle Datenübertragung über die PC Card-Schnittstelle (max. Datenübertragungsrate: 3 MBytes/s)</li> <li>• Hochintegrierter ASIC:           <ul style="list-style-type: none"> <li>* Hochleistungsdatenzugriff mit 24 Mbits/sec. zwischen Host und internem Puffer</li> <li>* 2,5 kB RAM</li> <li>* Chipkarten-Interface</li> <li>* Der Leser erkennt, ob die Chipkarte gesteckt oder gezogen ist.</li> <li>* Chipkarten-Takt Generator für verschiedene Chipkartenbetriebsfrequenzen</li> <li>* CIS Information</li> </ul> </li> <li>• Flash Memory</li> <li>• Chipkarten-Kommunikationsprotokolle: ISO 7816</li> <li>• Der Leser unterstützt Chipkarten: ISO 7816</li> <li>• Kompatibel mit geprägten Chipkarten</li> </ul>																																				
<b>Software</b>	<ul style="list-style-type: none"> <li>• Kompatibel zu PC Card Rel 2.1 Card and Socket Services Software</li> <li>• Kompatibel zu Windows 95™, MS-DOS™, Windows 3.1x, WIN NT™ 4.0</li> <li>• Schneller, verschlüsselbarer Software Download</li> <li>• <b>CT-API Software Interface für Chipkarten Applikationen</b></li> </ul>																																				
<b>Elektr. Daten</b>	<p>Leistungsaufnahme: typische Stromaufnahme bei 5 V:</p> <table style="width: 100%; border: none;"> <tr> <td style="width: 80%;">Betrieb</td> <td style="text-align: right;">max. 35 mA</td> </tr> <tr> <td>Niedrig-Energiebetrieb</td> <td style="text-align: right;">150 µA</td> </tr> </table>	Betrieb	max. 35 mA	Niedrig-Energiebetrieb	150 µA																																
Betrieb	max. 35 mA																																				
Niedrig-Energiebetrieb	150 µA																																				
<b>Mechanische Daten</b>	<table style="width: 100%; border: none;"> <tr> <td style="width: 20%;"><u>Größe</u></td> <td style="width: 40%;">Type II extended</td> <td style="width: 20%;"><u>Leser</u></td> <td style="width: 20%;"></td> </tr> <tr> <td>- Länge</td> <td>100.6 mm</td> <td>- Kontakte</td> <td>8+ Endlage-Schalter</td> </tr> <tr> <td>- Breite</td> <td>54 mm</td> <td>- Kontakt-Prinzip</td> <td>schleifende Kontakte</td> </tr> <tr> <td>- Dicke</td> <td>5.0 mm</td> <td>- Material</td> <td>selektiv Au über Ni</td> </tr> <tr> <td>- Anschluß</td> <td>5 V coding</td> <td colspan="2">Verträglich mit den geprägten Teilen von ISO Chipkarten</td> </tr> <tr> <td colspan="4"> </td> </tr> <tr> <td><u>Material</u></td> <td></td> <td></td> <td></td> </tr> <tr> <td>- Gehäuse</td> <td>rostfreier Stahl</td> <td></td> <td></td> </tr> <tr> <td>- Rahmen</td> <td>Kunststoff</td> <td></td> <td></td> </tr> </table>	<u>Größe</u>	Type II extended	<u>Leser</u>		- Länge	100.6 mm	- Kontakte	8+ Endlage-Schalter	- Breite	54 mm	- Kontakt-Prinzip	schleifende Kontakte	- Dicke	5.0 mm	- Material	selektiv Au über Ni	- Anschluß	5 V coding	Verträglich mit den geprägten Teilen von ISO Chipkarten		 				<u>Material</u>				- Gehäuse	rostfreier Stahl			- Rahmen	Kunststoff		
<u>Größe</u>	Type II extended	<u>Leser</u>																																			
- Länge	100.6 mm	- Kontakte	8+ Endlage-Schalter																																		
- Breite	54 mm	- Kontakt-Prinzip	schleifende Kontakte																																		
- Dicke	5.0 mm	- Material	selektiv Au über Ni																																		
- Anschluß	5 V coding	Verträglich mit den geprägten Teilen von ISO Chipkarten																																			
<u>Material</u>																																					
- Gehäuse	rostfreier Stahl																																				
- Rahmen	Kunststoff																																				
<b>Betriebsumgebung</b>	<table style="width: 100%; border: none;"> <tr> <td style="width: 80%;">Betriebstemperaturbereich</td> <td style="text-align: right;">0 to +60 Grad C</td> </tr> <tr> <td>Lagertemperaturbereich</td> <td style="text-align: right;">-20 to +80 Grad C</td> </tr> <tr> <td>Rel. Feuchtigkeit</td> <td style="text-align: right;">max. 95% nicht kondensierend</td> </tr> </table>	Betriebstemperaturbereich	0 to +60 Grad C	Lagertemperaturbereich	-20 to +80 Grad C	Rel. Feuchtigkeit	max. 95% nicht kondensierend																														
Betriebstemperaturbereich	0 to +60 Grad C																																				
Lagertemperaturbereich	-20 to +80 Grad C																																				
Rel. Feuchtigkeit	max. 95% nicht kondensierend																																				
<b>Haltbarkeit</b>	<table style="width: 100%; border: none;"> <tr> <td style="width: 80%;">MTBF (Stunden)</td> <td style="text-align: right;">500.000</td> </tr> <tr> <td>PC Card Anschluß-Steckzyklen</td> <td style="text-align: right;">10.000</td> </tr> <tr> <td>Chipkarten Kontakt-Steckzyklen</td> <td style="text-align: right;">100.000</td> </tr> <tr> <td>Statische Entladung</td> <td style="text-align: right;">CE Kompatibel (8KV)</td> </tr> </table>	MTBF (Stunden)	500.000	PC Card Anschluß-Steckzyklen	10.000	Chipkarten Kontakt-Steckzyklen	100.000	Statische Entladung	CE Kompatibel (8KV)																												
MTBF (Stunden)	500.000																																				
PC Card Anschluß-Steckzyklen	10.000																																				
Chipkarten Kontakt-Steckzyklen	100.000																																				
Statische Entladung	CE Kompatibel (8KV)																																				

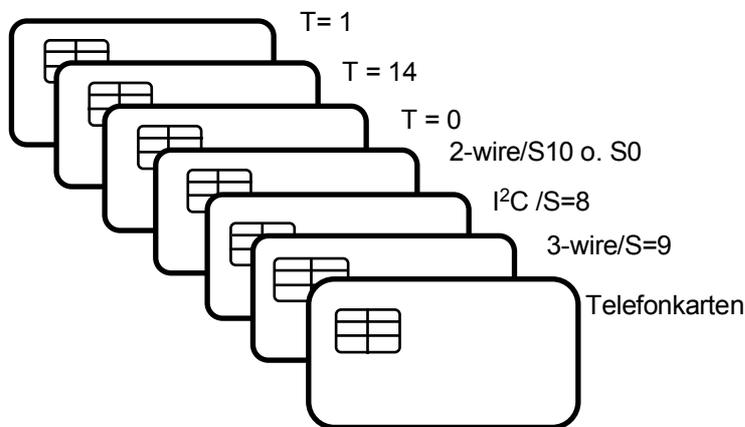
Technische Änderungen vorbehalten

Stand: 01.04.1998

## 15 Datenblatt B1-Chipkartenleser-PC (Einbauversion 3,5" / 5.25")



Der B1-Chipkartenleser verarbeitet diese Speicher- /Prozessor-Chipkarten



über die einheitliche Applikationsschnittstelle CT-API.

## 15.1 Technische Daten

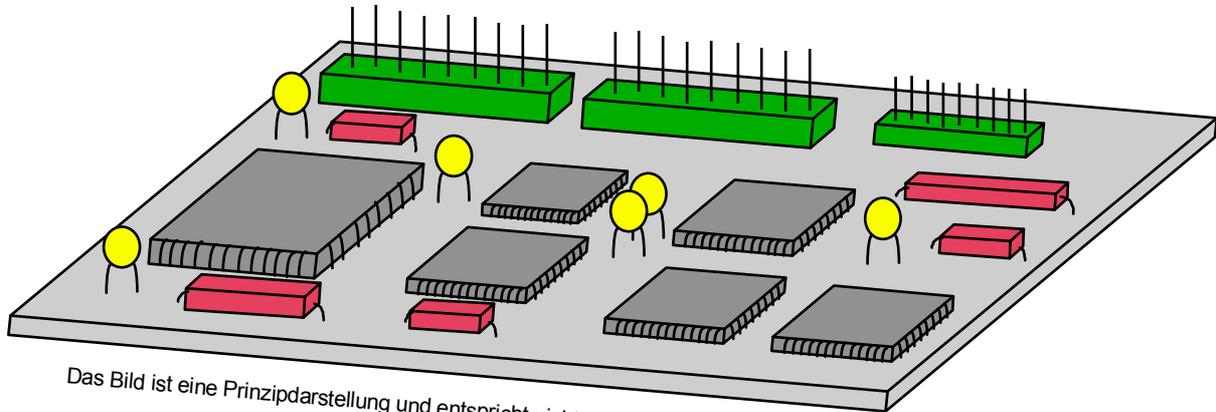
### Unterstützte Datenübertragungsprotokolle für Chipkarten:

<u>Protokoll</u>	<u>Referenz</u>
<i>Asynchron:</i>	
T=0	ISO 7816-3
T=1	ISO 7816-3
T=14	FTZ 1 TR 15, der gemeinsame Vorschlag der GZS und der Deutschen Telekom AG für ein nationales, asynchrones Blockübertragungsprotokoll
<i>Synchron:</i>	
2-Wire-Protokoll, auch S10 oder S0	Siemens Datenblatt SLE 4432 und SLE 4442, Philips Datenblatt PCB 2032/42
3-Wire-Protokoll	Siemens Datenblatt SLE 4418 und SLE 4428
I <sup>2</sup> C-Bus	Philips Datenblatt PCF 8582E-2
Telefonkarten	Separate Beschreibung notwendig

• Prozessor:	80C32 mit 12 MHz getaktet
• Smartcard Coupler:	Philips TDA 8000
• Firmware (im Lieferumfang):	kryptographisch gesichert fernladbar
• RAM-Speicher:	32 k * 8 statisches RAM
• Programmspeicher:	32 k * 8 Flash-EPROM
• Anzeige:	1 LED
• Kontaktiereinheit:	Absenkkontakte, 200.000 Steckzyklen
• Clock zum Betrieb der Karte:	4,9152 MHz
• Stromversorgung:	+ 5 V DC und + 7-18 V DC (intern)
• Schnittstelle zum Hostrechner:	V.24 mit 10poligem Pfostenstecker
• Protokoll zum Hostrechner:	T=1 nach ISO 7816-3, Amd. 1 "T=1"
• V.24-Karte	Incl. Kabel im Lieferumfang enthalten, technische Daten s. Betriebsanleitung
• Betriebsumgebung:	0-70° C Betriebstemperatur bei 20-80% relativer Feuchte
• Einbauschacht:	3,5" und 5,25" Schacht
• Treiber-SW und CT-API verfügbar für:	MS-DOS™, Win 95™, Win 3.11™, Win NT™, SINIX, SUN Solaris™ 2.5.1, LINUX™ 2.029
• Applikationsschnittstelle:	CT-API der Deutschen Telekom AG, TeleTrust, RWTÜV und GMD
alternativ	
• PC/SmartCard-Architektur	B1-IFD-Handler, CT-API Service Provider
• Funktionsbibliotheken für:	Microsoft, Borland
• Konformität:	ISO 7816-2, -3, B1, Krankenversichertenkarte wird automatisch erkannt

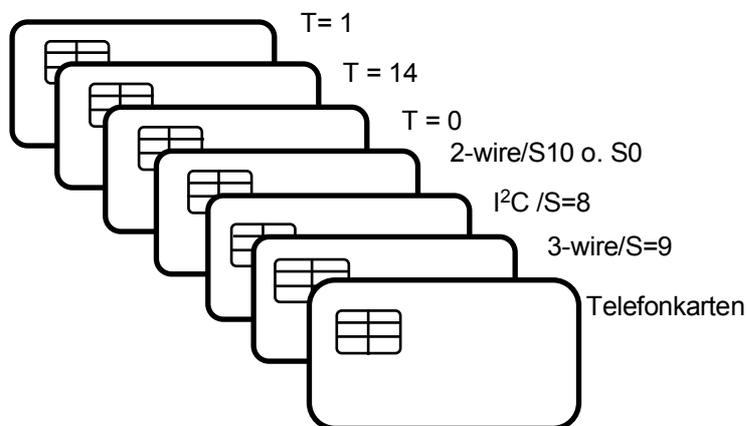
Technische Änderungen vorbehalten, Stand: 01.04.1998

## 16 Datenblatt B1-Chipkartenleser-FB (Flachbaugruppe)



Das Bild ist eine Prinzipdarstellung und entspricht nicht der Lieferform.

Der B1-Chipkartenleser verarbeitet diese Speicher- /Prozessor-Chipkarten



über die einheitliche Applikationsschnittstelle CT-API.

## 16.1 Technische Daten

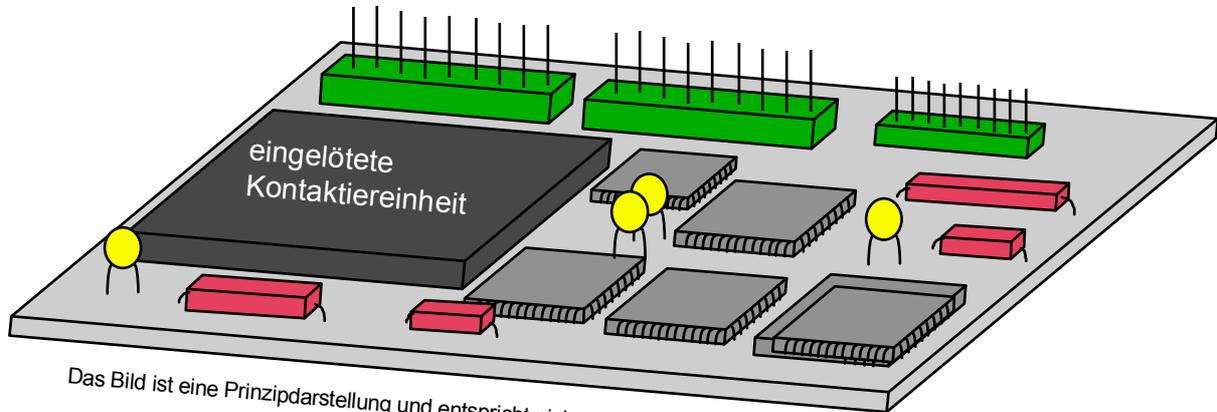
### Unterstützte Datenübertragungsprotokolle für Chipkarten :

<u>Protokoll</u>	<u>Referenz</u>
<i>Asynchron:</i>	
T=0	ISO 7816-3
T=1	ISO 7816-3
T=14	FTZ 1 TR 15, der gemeinsame Vorschlag der GZS und der Deutschen Telekom AG für ein nationales, asynchrones Blockübertragungsprotokoll
<i>Synchron:</i>	
2-Wire-Protokoll, auch S10 oder S0	Siemens Datenblatt SLE 4432 und SLE 4442, Philips Datenblatt PCB 2032/42
3-Wire-Protokoll	Siemens Datenblatt SLE 4418 und SLE 4428
I <sup>2</sup> C-Bus	Philips Datenblatt PCF 8582E-2
Telefonkarten	Separate Beschreibung notwendig

- Prozessor: 80C32 mit 12 MHz getaktet
- Smartcard Coupler: Philips TDA 8000
- Firmware (im Lieferumfang): kryptographisch gesichert fernladbar
- Steuerung des Lesers: über Busy- und Reset-Signal
- RAM-Speicher: 32 k \* 8 statisches RAM
- Programmspeicher: 32 k \* 8 Flash-EPROM
- nicht belegte Input-/Output-Ports: auf Anfrage
- Kontaktiereinheit (**optional**): Kontaktiereinheit mit Senkkontakten (200.000 Steckzyklen) über Flexkabel anschließbar
- Clock zum Betrieb der Karte: 4,9152 MHz
- Stromversorgung: + 5 V DC stabilisiert und + 7-18 V DC
- Schnittstelle zum Hostrechner: V.24-Treiber und TTL-Treiber, einstellbar, 10poliger Pfostenstecker
- Protokoll zum Hostrechner: Softwareabhängig: V.24 mit T=1 oder TTL mit 7-Bit, angenähert an DIN 66019
- Betriebsumgebung: 0-70° C Betr.temperatur bei 20-80% rel. Feuchte
- Größe der Flachbaugruppe: 71(B) x 80(T) x 10(H) mm
- Befestigung der Flachbaugruppe: 3 Löcher, 3,3 mm
- Treiber-SW und CT-API-Verfügbar für: MS-DOS™, Win 95™, Win NT™, SINIX Win 3.11™, SUN Solaris™ 2.5.1, LINUX™ 2.029
- Applikationsschnittstelle: CT-API der Deutschen Telekom AG, TeleTrusT, RWTÜV und GMD
- Funktionsbibliotheken für: Microsoft, Borland
- Konformität: - ISO 7816-2, -3, - B1  
- Krankenversichertenkarte wird automatisch erkannt

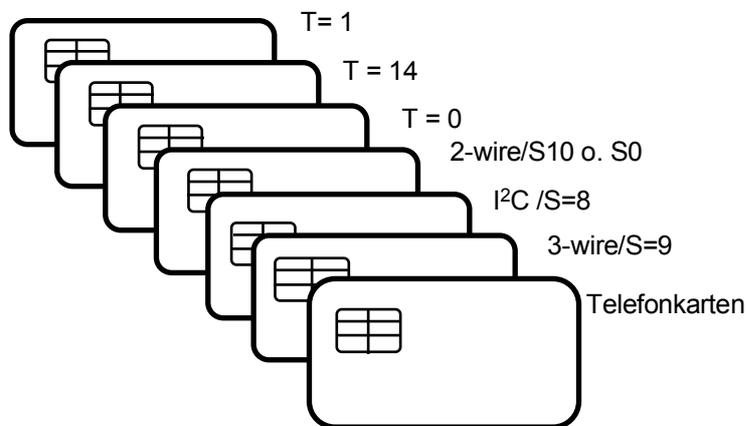
Technische Änderungen vorbehalten, Stand: 01.04.1998

## 17 Datenblatt B1-Chipkartenleser-FB (Flachbaugruppe für Scenic Pro)



Das Bild ist eine Prinzipdarstellung und entspricht nicht der Lieferform.

Der B1-Chipkartenleser verarbeitet diese Speicher- /Prozessor-Chipkarten



über die einheitliche Applikationsschnittstelle CT-API.

## 17.1 Technische Daten

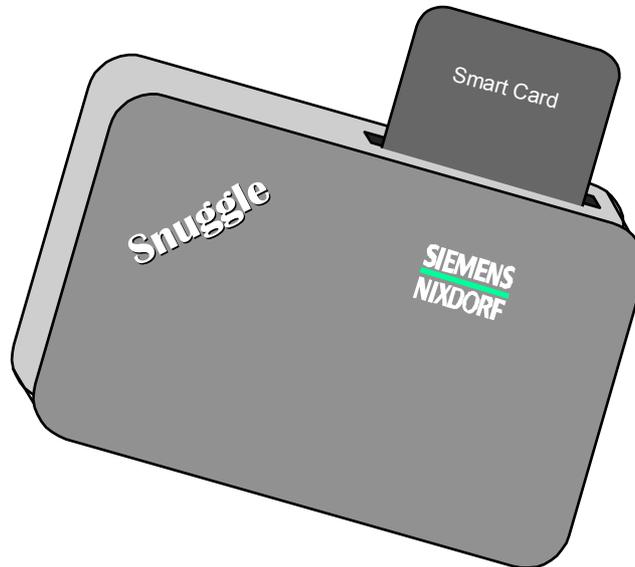
### Unterstützte Datenübertragungsprotokolle für Chipkarten :

<u>Protokoll</u>	<u>Referenz</u>
<i>Asynchron:</i>	
T=0	ISO 7816-3
T=1	ISO 7816-3
T=14	FTZ 1 TR 15, der gemeinsame Vorschlag der GZS und der Deutschen Telekom AG für ein nationales, asynchrones Blockübertragungsprotokoll
<i>Synchron:</i>	
2-Wire-Protokoll, auch S10 oder S0	Siemens Datenblatt SLE 4432 und SLE 4442, Philips Datenblatt PCB 2032/42
3-Wire-Protokoll	Siemens Datenblatt SLE 4418 und SLE 4428
I <sup>2</sup> C-Bus	Philips Datenblatt PCF 8582E-2
Telefonkarten	Separate Beschreibung notwendig

- Prozessor: 80C32 mit 12 MHz getaktet
- Smartcard Coupler: Philips TDA 8002
- Firmware (im Lieferumfang): kryptographisch gesichert fernladbar
- Steuerung des Lesers: über Busy- und Reset-Signal
- RAM-Speicher: 32 k \* 8 statisches RAM
- Programmspeicher: 32 k \* 8 Flash-EPROM
- nicht belegte Input-/Output-Ports: auf Anfrage
- Kontaktiereinheit: fest eingelötet, Senkkontakte 200.000 Steckzyklen
  
- Clock zum Betrieb der Karte: 4,9152 MHz
- Stromversorgung und Schnittstelle zum Hostrechner: über mitgeliefertes Kabel innerhalb Scenic Pro
- Protokoll zum Hostrechner: V.24 mit T=1
- Betriebsumgebung: 0-70° C Betr.temperatur bei 20-80% rel. Feuchte
- Größe der Flachbaugruppe: 100(B) x 71(T) mm
- Befestigung der Flachbaugruppe: 2 Löcher
- Treiber-SW und CT-API-Verfügbar für: MS-DOS™, Win 95™, Win NT™, SINIX Win 3.11™, SUN Solaris™ 2.5.1, LINUX™ 2.029
  
- Applikationsschnittstelle: CT-API der Deutschen Telekom AG, TeleTrusT, RWTÜV und GMD
  
- Funktionsbibliotheken für: Microsoft, Borland
- Konformität: - ISO 7816-2, -3, - B1  
- Krankenversichertenkarte wird automatisch erkannt

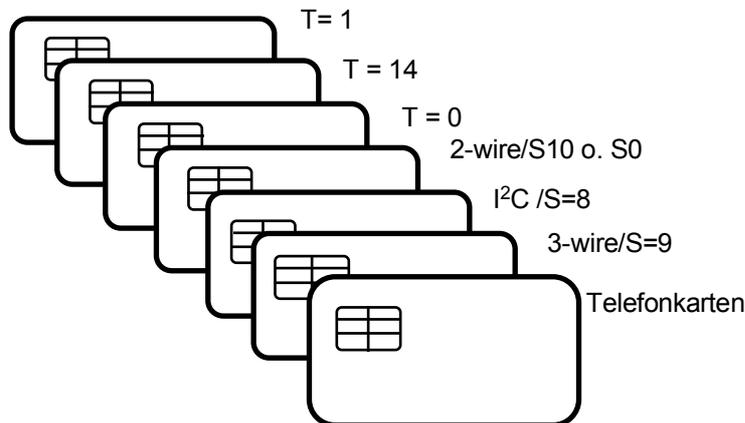
Technische Änderungen vorbehalten, Stand: 01.10.1997

## 18 Datenblatt B1-Chipkartenleser Snuggle AT



Das Bild ist eine Prinzipdarstellung und entspricht nicht der Lieferform.

Der B1-Chipkartenleser verarbeitet diese Speicher- /Prozessor-Chipkarten



über die einheitliche Applikationsschnittstelle CT-API.

## 18.1 Technische Daten

### Unterstützte Datenübertragungsprotokolle für Chipkarten :

<u>Protokoll</u>	<u>Referenz</u>
<i>Asynchron:</i>	
T=0	ISO 7816-3
T=1	ISO 7816-3
T=14	FTZ 1 TR 15, der gemeinsame Vorschlag der GZS und der Deutschen Telekom AG für ein nationales, asynchrones Blockübertragungsprotokoll
<i>Synchron:</i>	
2-Wire-Protokoll, auch S10 oder S0	Siemens Datenblatt SLE 4432 und SLE 4442, Philips Datenblatt PCB 2032/42
3-Wire-Protokoll	Siemens Datenblatt SLE 4418 und SLE 4428
I <sup>2</sup> C-Bus	Philips Datenblatt PCF 8582E-2
Telefonkarten	Separate Beschreibung notwendig

- Prozessor: 80C32 mit 12 MHz getaktet
- Decoder TDA 8002
- Firmware (im Lieferumfang): kryptographisch gesichert fernladbar
- Anzeige: 1 LED zweifarbig
- Kontaktiereinheit: Landkontakte, 100.000 Steckzyklen
- Clock zum Betrieb der Karte: 4,9152 MHz
- Stromversorgung: über PS2-Port, auf Anfrage DIN-Buchse
- Kabellänge: 1,5 m (Desktopversion), 0,5 m (Notebookversion)
- Schnittstelle zum Hostrechner: V.24 mit 9-poligem Sub-D
- Protokoll zum Hostrechner: T=1 nach ISO 7816-3, Amd. 1 "T=1"
- Gehäusematerial: ABS-Kunststoff mit UL-Klassifizierung
- Gehäuse-Abmessung / Gewicht: ca. 120 x 70 x 24 mm / 0,1 kg
- Betriebsumgebung: Klasse 3.1, Büroklimatisierung, 0 °-50° C
- Treiber-SW 6 CT-API verfügbar für: MS-DOS™, Win 3.11™, Win 95™, Win NT™, SINIX™, SUN Solaris™ 2.5.1, LINUX 2.029
- Applikationsschnittstelle: CT-API gemäß Dt. Telekom AG, RWTÜV, TeleTrust und GMD
- alternativ
- PC/SmartCard-Architektur: B1-IFD-Handler, CT-API Service Provider
- Funktionsbibliothek für: Microsoft, Borland
- Konformität: ISO 7816-3, B1, Schnittstellenspezifikation für die ec-Karte mit Chip (EMV '96 Version 3.0), Krankenversichertenkarte wird automatisch erkannt <sup>1)</sup>
- Zubehör: Standfuß, Klettband, Klebeband

<sup>1)</sup> Rezerifizierung nach ITSECE2 ist beantragt

Technische Änderungen vorbehalten, Stand: 01.04.1998

