

# Premium-, Micro- Steuerungen

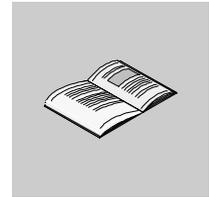
TSX CPP 100/110

07/2008 ger

---

---

# Inhaltsverzeichnis



---

	<b>Über dieses Buch</b> .....	<b>7</b>
<b>Kapitel 1</b>	<b>Allgemeines</b> .....	<b>9</b>
	Auf einen Blick .....	9
	Prinzipien .....	10
	Allgemeine Architektur eines CANopen-Feldbusses .....	11
	Übertragungsgeschwindigkeit und Buslänge .....	12
<b>Kapitel 2</b>	<b>Überblick über die PCMCIA-Karte TSX CPP 100/110</b> .....	<b>15</b>
	Auf einen Blick .....	15
2.1	Beschreibung der Karte TSX CPP 100/110 .....	16
	Auf einen Blick .....	16
	Eigenschaften der Karten TSX CPP 100 und TSX CPP 110 .....	17
	Einbau der Karte TSX CPP 100 .....	19
	Einbau der Karte TSX CPP 110 .....	21
	Anschlusseinheit TSX CPP ACC1 .....	23
2.2	Technische Spezifikationen .....	25
	Auf einen Blick .....	25
	Normen und Kenndaten .....	26
	CANopen-Kenndaten .....	27
	Prozessoren, die die Karte TSX CPP 100/110 unterstützen .....	28
<b>Kapitel 3</b>	<b>Inbetriebnahme der Software</b> .....	<b>31</b>
	Auf einen Blick .....	31
3.1	Allgemeines .....	32
	Auf einen Blick .....	32
	Prinzip der Inbetriebnahme .....	33
	Vorgehensweise bei der Inbetriebnahme .....	35
3.2	Konfiguration .....	37
	Auf einen Blick .....	37
	Zugriff auf das Konfigurationsfenster .....	38
	Konfigurationsbildschirm .....	39
	Beschreibung der E/A Daten und des Busverhaltens beim Starten .....	42
	Beschreibung der Wahlmöglichkeiten für den Konfigurations-Lademodus .....	45
	So laden Sie eine Konfiguration mit X-Way .....	47

	Fenster für die Konfiguration der Slaves .....	50
	Konfiguration einer CANopen-PCMCIA-Karte .....	51
	Konfigurationsdokument der PCMCIA-Karte TSX CPP 100/110.....	54
3.3	Programmierung .....	56
	Auf einen Blick .....	56
	Zugriff auf SDOs von CANopen .....	57
	Zugriff auf die PDUs der Verbindungsschicht CAN .....	62
	Request IDENTIFICATION .....	64
	Diagnosebefehl .....	67
3.4	Debugging .....	69
	Beschreibung des Debug-Fensters .....	69
3.5	Diagnose .....	72
	Auf einen Blick .....	72
	Diagnose mithilfe der Status-LEDs der PCMCIA-Karte TSX CPP 100/110. ....	73
	Die Diagnosedaten .....	75
	So führen Sie eine Diagnose durch .....	78
3.6	Sprachobjekte in Verbindung mit der Karte TSX CPP 100/110.....	82
	Auf einen Blick .....	82
	Sprachobjekte mit implizitem Austausch .....	83
	Sprachobjekte mit explizitem Austausch .....	88
	Verwaltung des expliziten Austauschs .....	89
	Sprachobjekte, der Konfiguration zugeordnet .....	90
	Fehlercodes .....	92
<b>Kapitel 4</b>	<b>Konfigurationsbeispiel mit dem SyCon-Tool .....</b>	<b>99</b>
	Auf einen Blick .....	99
4.1	Standardkonfiguration .....	100
	Auf einen Blick .....	100
	Beispielbeschreibung .....	101
	Deklaration des CANopen Masters .....	102
	Konfiguration des CANopen-Busses .....	104
	Deklaration von Slave 7 .....	105
	Konfiguration des Slave 7 .....	107
	Deklaration von Slave 8 .....	111
	Konfiguration des Slave 8 .....	112
	Überprüfung der durchgeführten Konfiguration .....	113
4.2	Multi-Master-Konfiguration.....	116
	Auf einen Blick .....	116
	Beispielbeschreibung .....	117
	Deklaration des CANopen Masters A und der Slave 7 und 8 .....	119
	Deklaration des CANopen-Masters B .....	120
	CANopen-Buskonfiguration des Masters B .....	122
	Deklaration und Konfiguration der Slave 9 und 10 .....	123
	Deklaration und Konfiguration des Slave 7 für Master B .....	124
	Deklaration und Konfiguration des Slave 8 für Master B .....	129

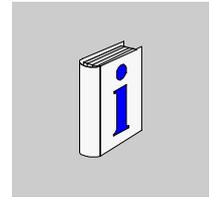
---

<b>Glossar</b>	.....	<b>135</b>
<b>Index</b>	.....	<b>139</b>

---

---

# Über dieses Buch



---

## Auf einen Blick

### Ziel dieses Dokuments

Dieses Handbuch ist für Benutzer gedacht, die eine der folgenden CANopen-PCMCIA-Kommunikationskarten in Betrieb nehmen möchten:

- TSX CPP 100 mit Premium
- TSX CPP 110 mit Premium oder Micro

### Gültigkeitsbereich

Dieses Dokument beschreibt die Inbetriebnahme der PCMCIA-Karte TSX CPP 100 oder TSX CPP 110 unter PL7 V4.5 und SyCon V2.8.

### Weiterführende Dokumentation

Titel	Referenz-Nummer
Dokumentation der gemeinsamen applikationsspezifischen Kommunikation	TLX DSCOM PL7 xxG

### Benutzerkommentar

Ihre Anmerkungen und Hinweise sind uns jederzeit willkommen. Senden Sie sie einfach an unsere E-mail-Adresse: [techpub@schneider-electric.com](mailto:techpub@schneider-electric.com)

---



---

# Allgemeines



# 1

---

## Auf einen Blick

### **Gegenstand dieses Kapitels**

Dieses Kapitel beschreibt die wesentlichen Eigenschaften einer Kommunikation über CANopen.

### **Inhalt dieses Kapitels**

Dieses Kapitel enthält die folgenden Themen:

<b>Thema</b>	<b>Seite</b>
Prinzipien	10
Allgemeine Architektur eines CANopen-Feldbusses	11
Übertragungsgeschwindigkeit und Buslänge	12

## Prinzipien

---

### Einleitung

Nachdem der Kommunikationsbus CAN ursprünglich für die On-Board-Systeme von Autos entwickelt worden ist, wird er inzwischen in zahlreichen Gebieten verwendet, wie etwa:

- Transportwesen,
- fahrbare Geräte,
- medizinische Geräte,
- Bauwesen,
- Industrieüberwachung.

Das CAN-System zeichnet sich durch die folgenden Pluspunkte aus:

- das Zuordnungssystem des Busses,
  - die Fehlererkennung,
  - die Zuverlässigkeit des Datenaustauschs.
- 

### Master-/Slave-Struktur

Der CAN-Bus weist eine Master-/Slave-Struktur für die Verwaltung des Busses auf.

Der Master verwaltet

- die Initialisierung der Slaves,
  - die Kommunikationsfehler,
  - den Status der Slaves.
- 

### Peer-/Peer-Kommunikation

Die Kommunikation auf dem Bus erfolgt von **Peer zu Peer**, jedes Gerät kann jederzeit einen Request an den Bus schicken und die betroffenen Geräte antworten. Die Priorität der Requests auf dem Bus wird bei allen Meldungen mit Hilfe eines Kennzeichens bestimmt.

---

### CAN-Kennungen

Beim expliziten Austausch der CAN-PDUs auf Verbindungsebene (siehe *Zugriff auf die PDUs der Verbindungsschicht CAN, S. 62*) werden die Kennungen im erweiterten 29-Bit-Format verwendet (CAN-Standard V2.0B).

Die 11-Bit-Kennungen (CAN-Standard V2.0A) können beim Senden verwendet werden, aber der Empfang dieses Kennungstyps wird nicht unterstützt.

---

## Allgemeine Architektur eines CANopen-Feldbusses

---

### Auf einen Blick

Eine CANopen-Architektur besteht aus:

- Busmaster (PCMCIA-Karte TSX CPP 1\*\*)
- Slavegeräten

**Hinweis:** es ist auch möglich, mehrere PCMCIA-Karten TSX CPP 1\*\* an den Bus anzuschließen. Dann ist eine Karte der Master, die anderen sind im **Abhörmodus**. Mit Karten im **Abhörmodus** können die zugehörigen Premium-Steuerungen sofort feststellen, in welchem Zustand sich Bus und Slaves am Bus befinden.

**Hinweis:** die PCMCIA-Karte TSX CPP 100 verwaltet zwei Bussegmente (A und B). Hierzu ist das Anschlussgehäuse TSX CPP ACC1 nötig.

Die Baudrate hängt von der Länge und der Art der verwendeten Kabel ab. (siehe *Übertragungsgeschwindigkeit und Buslänge, S. 12*).

---

## Übertragungsgeschwindigkeit und Buslänge

---

**Auf einen Blick** CANopen erlaubt 128 Geräte (Busmaster und 127 dezentrale Slaves). Die Übertragungsgeschwindigkeit hängt stark von der Buslänge und den Eigenschaften der verwendeten Kabel ab. Anhand der beiden folgenden Beispiele lassen sich die zulässigen Werte abschätzen.

---

**Beispiel 1** Die folgende Tabelle beschreibt den Zusammenhang zwischen maximaler Übertragungsgeschwindigkeit und Buslänge in Abhängigkeit von den verwendeten Kabeln. Es werden maximal 32 Slaves und keine Repeater an den Segmenten A oder B der Karte TSX CPP 100 oder dem einzigen Segment der Karte TSX CPP 110 verwendet.

Übertragungsrate	Buslänge	Kabel-Kenndaten
1 Mbit/s	25 m	0,25 mm <sup>2</sup> , AWG 23
800 Kbit/s	50 m	0,34 mm <sup>2</sup> , AWG 22
500 Kbit/s	100 m	0,34 mm <sup>2</sup> , AWG 22
250 Kbit/s	250 m	0,34 mm <sup>2</sup> , AWG 22
125 Kbit/s	500 m	0,5 mm <sup>2</sup> , AWG 20
50 Kbit/s	1.000 m	0,75 mm <sup>2</sup> , AWG 18
20 Kbit/s	1.000 m	0,75 mm <sup>2</sup> , AWG 18
10 Kbit/s	1.000 m	0,75 mm <sup>2</sup> , AWG 18

---

**Beispiel 2**

Die folgende Tabelle beschreibt den Zusammenhang zwischen maximaler Übertragungsgeschwindigkeit und Buslänge in Abhängigkeit von den verwendeten Kabeln. Es werden maximal 100 Slaves und keine Repeater an den Segmenten A oder B der Karte TSX CPP 100 oder dem einzigen Segment der Karte TSX CPP 110 verwendet.

Übertragungsrate	Buslänge	Kabel-Kenndaten
1 Mbit/s	25 m	0,34 mm <sup>2</sup> , AWG 22
800 Kbit/s	50 m	0,6 mm <sup>2</sup> , AWG 20
500 Kbit/s	100 m	0,6 mm <sup>2</sup> , AWG 20
250 Kbit/s	250 m	0,6 mm <sup>2</sup> , AWG 20
125 Kbit/s	500 m	0,75 mm <sup>2</sup> , AWG 18
50 Kbit/s	1.000 m	1 mm <sup>2</sup> , AWG 17
20 Kbit/s	1.000 m	1 mm <sup>2</sup> , AWG 17
10 Kbit/s	1.000 m	1 mm <sup>2</sup> , AWG 17



---

# Überblick über die PCMCIA-Karte TSX CPP 100/110

# 2

---

## Auf einen Blick

### Inhalt des Kapitels

Dieses Kapitel beschreibt die wichtigsten technischen Kenndaten der PCMCIA-Karte TSX CPP 100/110.

### Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Abschnitte:

Abschnitt	Thema	Seite
2.1	Beschreibung der Karte TSX CPP 100/110	16
2.2	Technische Spezifikationen	25

## 2.1 Beschreibung der Karte TSX CPP 100/110

---

### Auf einen Blick

**Inhalt des Abschnitts**

Dieser Abschnitt beschreibt die mechanischen Eigenschaften der PCMCIA-Karte TSX CPP1•• und ihrer Anschlüsse.

---

**Inhalt dieses Abschnitts**

Dieser Abschnitt enthält die folgenden Themen:

Thema	Seite
Eigenschaften der Karten TSX CPP 100 und TSX CPP 110	17
Einbau der Karte TSX CPP 100	19
Einbau der Karte TSX CPP 110	21
Anschlusseinheit TSX CPP ACC1	23

---

## Eigenschaften der Karten TSX CPP 100 und TSX CPP 110

### Auf einen Blick

Mit der CANopen-Kommunikationskarte TSX CPP 1•• lassen sich CANopen-Architekturen errichten. Diese Karte ist der Busmaster und ermöglicht den Anschluss von Geräten, die dem CANopen-Standard entsprechen:

- Impliziter Austausch von **Process Data Objects** über die Wörter %MW
- Expliziter Austausch von **Service Data Objects** über die Funktionsblöcke READ\_VAR und WRITE\_VAR
- Kompatibilität mit für CANopen (2.0A und 2.0B) standardisierten Geräte- und Kommunikationsprofilen

### Physikalische Beschreibung

Die Karte TSX CPP 1•• ist eine PCMCIA-Karte des Typs III, die in den PCMCIA-Kommunikationssteckplatz des Prozessors eingesetzt wird.

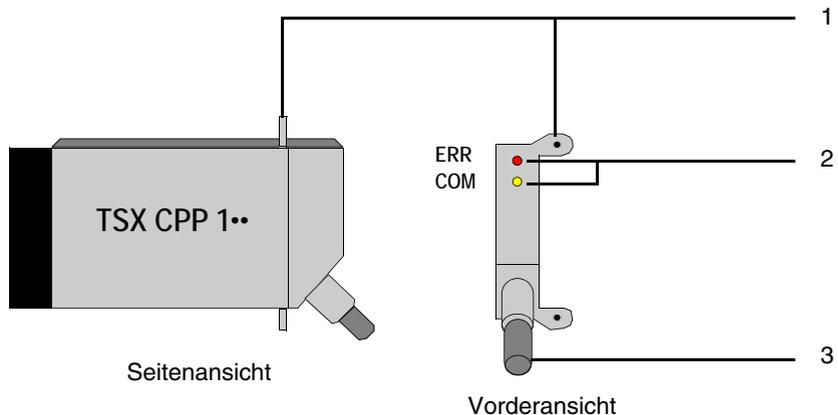
Die CANopen-PC-Karten des Typs III (TSX CPP 1••) funktionieren im PCMCIA-Steckplatz der folgenden Premium-CPU-Typen:

- TSX P57103, TSX P572•3, TSX P572•23, TSX P573•3, TSX P57453, TSX P573623, TSX P574823, T PCX 57203, T PCX 57353.

Die CANopen-PC-Karten des Typs III (TSX CPP 110) funktionieren im PCMCIA-Steckplatz der folgenden Micro-CPU-Typen:

- TSX 3721•••1

Dieses Modul besteht aus den folgenden Elementen:



**Beschreibung**

Diese Tabelle beschreibt die Elemente der vorherigen Abbildung.

Nummer	Beschreibung
1	Befestigungshaken oben und unten an der Karte, mit denen die Karte am Prozessor befestigt werden kann
2	LED-Anzeigen für die Funktionsdiagnose der Kommunikationskarte (siehe <i>Diagnose, S. 74</i> )
3	<ul style="list-style-type: none"><li>● TSX CPP 100: Buskabel mit 60 cm Länge und einem 15-poligen Sub-D-Stecker an einem Ende für den Anschluss an TSX CPP ACC1 (siehe <i>Anschlusseinheit TSX CPP ACC1, S. 23</i>)</li><li>● TSX CPP 110: Buskabel mit 50 cm Länge und einem industriellen CANopen-TAP an einem Ende</li></ul>

---

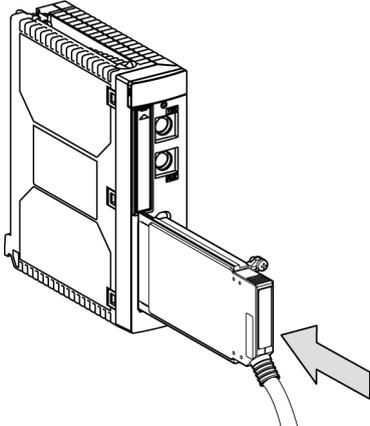
## Einbau der Karte TSX CPP 100

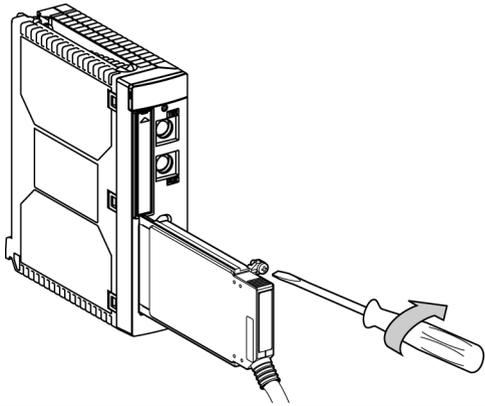
### Auf einen Blick

Diese Karte wird in einen für Kommunikationskarten vorgesehenen Steckplatz gesteckt. Das Anschlusskabel des Busses ist mit der Karte verbunden und wird mit seinem 15-poligen Sub-D-Stecker an das Gehäuse TSX CPP ACC1 (auch TAP genannt) angeschlossen.

	<b>ACHTUNG</b>
	<p>PCMCIA-Karten dürfen nur bei ausgeschalteter Steuerung eingesetzt und entfernt werden.</p> <p><b>Die Nichtbeachtung dieser Anweisungen kann zu Körperverletzungen oder Materialschäden führen.</b></p>

Die folgende Tabelle beschreibt die für die Inbetriebnahme einer Karte TSX CPP 100 auszuführenden Schritte.

Schritt	Aktion	Abbildung der Montage an einer Premium-Steuerung
1	Schalten Sie die Steuerung aus.	
2	Setzen Sie die CANopen-PC-Karte des Typs III in den PCMCIA-Steckplatz der Premium-CPU ein.	

Schritt	Aktion	Abbildung der Montage an einer Premium-Steuerung
3	Schrauben Sie die Karte fest, um Manipulationen an der Karte zu verhindern und um die ordnungsgemäße Funktion zu gewährleisten.	
4	Schließen Sie das Kabel der Karte mit seinem 15-poligen Sub-D-Stecker an das Gehäuse TSX CPP ACC1 an. Vergessen Sie nicht, die Befestigungsschrauben gut anzuziehen und den TAP auf einer DIN-Schiene zu befestigen.	
5	Schalten Sie die Steuerung wieder ein.	

## Einbau der Karte TSX CPP 110

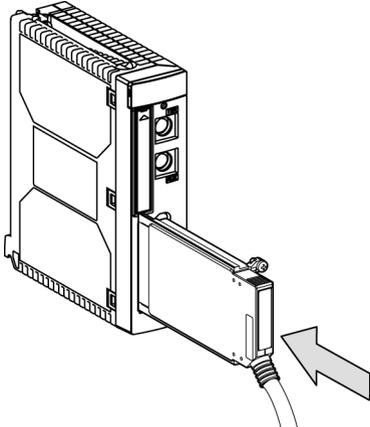
### PCMCIA-Karte TSX CPP 110

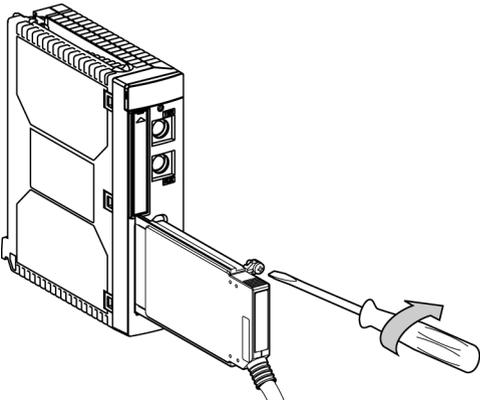
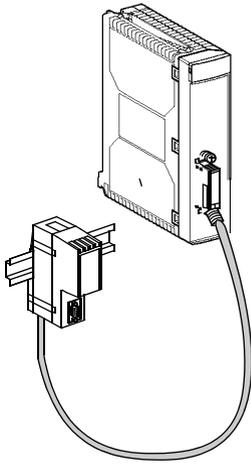
Die PC-Karte TSX CPP 110 mit ihrem industriellen TAP ist das Verbindungselement zwischen einer Premium- oder Micro-CPU und einem CANopen-Netzwerk.

**Hinweis:** das Modbus-Kommunikationsmodul TSX SCY 2160• kann nicht verwendet werden.

	<b>ACHTUNG</b>
	<p>PCMCIA-Karten dürfen nur bei ausgeschalteter Steuerung eingesetzt und entfernt werden.</p> <p><b>Die Nichtbeachtung dieser Anweisungen kann zu Körperverletzungen oder Materialschäden führen.</b></p>

Die folgende Tabelle beschreibt die für die Inbetriebnahme einer Karte TSX CPP 110 auszuführenden Schritte.

Schritt	Aktion	Abbildung der Montage an einer Premium-Steuerung
1	Schalten Sie die Steuerung aus.	
2	Setzen Sie die CANopen-PC-Karte des Typs III in den PCMCIA-Steckplatz der Premium-CPU ein.	

Schritt	Aktion	Abbildung der Montage an einer Premium-Steuerung
3	<p>Schrauben Sie die Karte fest, um Manipulationen an der Karte zu verhindern und um die ordnungsgemäße Funktion zu gewährleisten.</p>	 <p>The diagram illustrates the third step of the installation. It shows a PCMCIA card partially inserted into a slot. A screwdriver is being used to tighten a screw on the side of the card's housing, which is designed to lock the card in place and prevent it from being removed or damaged.</p>
4	<p>Befestigen Sie den TAP auf einer DIN-Schiene.</p>	 <p>The diagram illustrates the fourth step of the installation. It shows the PCMCIA card now fully secured in its housing. A separate component, labeled as a 'TAP', is being connected to a DIN rail. The TAP has a cable that is plugged into the card's connector, and another cable that connects to the DIN rail, effectively linking the card to the control system.</p>
5	<p>Schalten Sie die Steuerung wieder ein.</p>	

## Anschlusseinheit TSX CPP ACC1

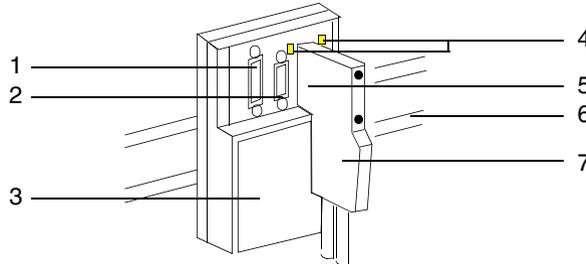
### Auf einen Blick

Die CANopen Karte TSX CPP 100 wird anhand des mit der Karte mitgelieferten Kabels an den Sub-D 15 Anschluss (Stift) der Anschlusseinheit TSX CPP ACC1 angeschlossen. Diese Anschlusseinheit ermöglicht über ihre 2 Sub-D 9 Stecker den Anschluss von zwei (galvanisch) getrennten Segmenten des CANopen Busses (Verdoppelung der maximalen Länge des Busses).

Kabel und anderes Zubehör zum Verbinden stehen in Spezialunternehmen zur Verfügung.

### Anschluss- einheit TSX CPP ACC1

Diese Abbildung stellt eine Anschlusseinheit TSX CPP ACC1 dar.



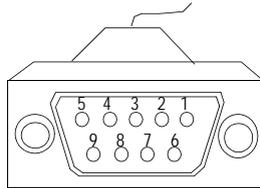
### Beschreibung der Abbildung

Diese Tabelle beschreibt die Elemente der vorherigen Abbildung.

Element	Beschreibung
1	<b>Stecker Sub-D 15 (Stift)</b> : an diesen Stecker wird die TSX CPP 100 Karte angeschlossen.
2	<b>Stecker Sub-D 9 (Stift)</b> : Stecker <b>A</b> für die Verbindung des CANopen Segments <b>A</b> .
3	Gehäuse TSX CPP ACC1.
4	LEDs zur Anzeige des Busbetriebs auf der Anschlusseinheit TSX CPP ACC1 (AUS = keine Kommunikation durch CPP 100).
5	<b>Stecker Sub-D 9 (Stift)</b> : Stecker <b>B</b> für die Verbindung des CANopen Segments <b>B</b> .
6	Hutschiene für die Montage
7	Stecker des CANopen-Bus mit oder ohne Busabschluss.

**Stecker Sub-D 9  
Anschluss-  
punkte**

Diese Abbildung stellt einen A oder B Bus-Stecker, Sub-D 9 (Buchse) für den Anschluss eines Bus-Segmentes an die Anschlusseinheit TSX CPP ACC1 dar.



**Beschreibung  
des Steckers**

Diese Tabelle beschreibt die Steckerbelegung.

Steckerpunkt	Beschreibung
1	reserviert
2	CAN_L
3	CAN_GND
4	reserviert
5	NC
6	NC
7	CAN_H
8	reserviert
9	NC

**Hinweis:** diese Steckerbelegung gilt auch für die CANopen-Schnittstelle des Taps der Karte TSX CPP 110.

---

## 2.2 Technische Spezifikationen

---

### Auf einen Blick

**Inhalt des Abschnitts**

Dieser Abschnitt beschreibt die wichtigsten technischen Kenndaten der Karte TSX CPP 1••.

---

**Inhalt dieses Abschnitts**

Dieser Abschnitt enthält die folgenden Themen:

Thema	Seite
Normen und Kenndaten	26
CANopen-Kenndaten	27
Prozessoren, die die Karte TSX CPP 100/110 unterstützen	28

---

## Normen und Kenndaten

---

### Normen

Die Kommunikationskarte TSX CPP 1•• entspricht den folgenden internationalen Normen und Standards:

Internationale Standards	ISO IS 11898, CAN High Speed Transceiver and Data Link Layer
US-Normen	UL508
Strahlungsstandards	CE-Kennzeichnung, FCC-B (50082-1)

### Elektrische Kenndaten

- Logik-Spannungsversorgung Vcc: 5 V, von der Spannungsversorgung des Racks geliefert
- Leistungsverbrauch: 3 W

### Umgebungsbedingungen

- Lagertemperatur: -25°C bis 70°C
  - Betriebstemperatur: 0°C bis 70°C
  - Lagerfeuchte: 30% bis 95%, ohne Kondensation
  - Betriebsfeuchte: 5% bis 95%, ohne Kondensation
-

## CANopen-Kenndaten

---

### Standards

Die Kommunikationskarte TSX CPP 100 entspricht dem Standard DS301 V3.0.

Die Kommunikationskarte TSX CPP 110 entspricht dem Standard DS301 V4.01.

---

### Besonderheiten

- Der Benutzer kann den PDO-Inhalt entsprechend der Norm DS301 V4.01 mappen.
  - Die Karte TSX CPP 110 unterstützt die Funktion "Heartbeat" (DS 301 V4.01)
  - Die Karte TSX CPP 1•• ist der das Netz verwaltende Master (NMT\_MASTER) am Bus. (Diese Funktion lässt sich mittels SyCon deaktivieren.)
  - Die Karte TSX CPP 1•• erzeugt die Synchronisationsvariable (SYNC). (Diese Funktion lässt sich mittels SyCon deaktivieren.)
  - Die Knotenadresse der Karten TSX CPP 1•• kann nicht für die Datenübertragung genutzt werden. Sie wird bei der Karte TSX CPP 110 lediglich für die Funktion "Heartbeat" genutzt.
-

## Prozessoren, die die Karte TSX CPP 100/110 unterstützen

---

### Auf einen Blick

- Versionsstand der CPUs:
  - Die Verwendung der Karte TSX CPP 110 oder TSX CPP 110 erfordert eine Version  $\geq 5.0$  für eine Premium/Atrium-CPU.
  - Die Verwendung der Karte TSX CPP 110 erfordert eine Version  $\geq 6.0$  für eine Micro-CPU.

Die Inbetriebnahme erfolgt ausgehend von der PL7-Software V4.4 für die TSX CPP 110.

Die allgemeine Konfiguration eines CANopen-Busses erfolgt mit der **Sycon**-Software ab Version 2.630 (TLX LFBCM), TSX CPP 110-Karte ab Version 2.8.

**Hinweis:** die PCMCIA-Karte muss sich unbedingt im Steckplatz des Prozessormoduls befinden. Folglich ist nur ein CANopen-Bus pro Steuerungs-CPU möglich.

---

**Prozessortypen  
und  
Speicherkapa-  
zitäten**

Die folgende Tabelle listet die Prozessoren, die die CANopen-PCMCIA- Karte TSX CPP 100/110 unterstützen, und ihre maximalen Speicherkapazitäten auf.

Prozessor	Maximale Größe der TSX CPP 100/110 im Prozessor befindlichen Konfigurationsdaten	Maximale Größe der Konfigurations-Ein-/Ausgangsdaten für den CANopen-Knoten	
		MAST-Task	FAST-Task
TSX 372***1	0 bis 8 KB	256 %MW (128+128)	32 %MW (16+16)
TSX P57103	0 bis 12 KB	384 %MW (192+192)	48 %MW (24+24)
TSX P57203 T PCX 57 203 TSX P572623 TSX P57253 TSX P572823	16 KB	512 %MW (256+256)	64 %MW (32+32)
TSX P57303 TSX P57353 TSX P573623 T PCX 57353	32 KB	1024 %MW (512+512)	256 %MW (128+128)
TSX P57453 TSX P574823	64 KB	3584 %MW (1792+1792)	512 %MW (256+256)
Legende	(1) : Diese maximale Größe kann überschritten werden, wenn man die Konfiguration mithilfe der Sycon-Software in die Karte lädt (siehe <i>Beschreibung der Wahlmöglichkeiten für den Konfigurations-Lademodus</i> , S. 45). Die aufgrund der Sycon-Software maximal zulässige Größe der Konfigurationsdaten beträgt 128 KB.		

**Hinweis:** die effektive Größe der Konfiguration ist im Wort %KWy.1.2 aufgeführt. (siehe *Sprachobjekte, der Konfiguration zugeordnet*, S. 90).



---

# Inbetriebnahme der Software

# 3

---

## Auf einen Blick

### Gegenstand dieses Kapitels

Diese Kapitel beschreibt die verschiedenen Möglichkeiten in Hinblick auf die Konfiguration, die Steuerung und die Diagnose einer CANopen-Anwendung.

### Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Abschnitte:

Abschnitt	Thema	Seite
3.1	Allgemeines	32
3.2	Konfiguration	37
3.3	Programmierung	56
3.4	Debugging	69
3.5	Diagnose	72
3.6	Sprachobjekte in Verbindung mit der Karte TSX CPP 100/110	82

## 3.1 Allgemeines

---

### Auf einen Blick

**Inhalt des Abschnitts**

Dieser Abschnitt beschreibt die softwaretechnische Inbetriebnahme der PCMCIA-Karte TSX CPP 1\*\*.

---

**Inhalt dieses Abschnitts**

Dieser Abschnitt enthält die folgenden Themen:

Thema	Seite
Prinzip der Inbetriebnahme	33
Vorgehensweise bei der Inbetriebnahme	35

---

## Prinzip der Inbetriebnahme

---

**Auf einen Blick** Bei der Inbetriebnahme eines CANopen-Busses muss festgelegt werden, in welcher hardwaretechnische Umgebung er integriert wird (Rack, Spannungsversorgung, Prozessor, Module oder Geräte usw.), bevor die softwaretechnische Inbetriebnahme erfolgt.

Für die softwaretechnische Inbetriebnahme verwendet man die verschiedenen PL7-Editoren

- entweder im Offline-Modus
- oder im Online-Modus: In diesem Fall können nur bestimmte Parameter geändert werden.

Der Bus wird mit der Sycon-Software konfiguriert.

---

**Prinzip der Inbetriebnahme**

Die folgende Tabelle zeigt die einzelnen Phasen der Inbetriebnahme.

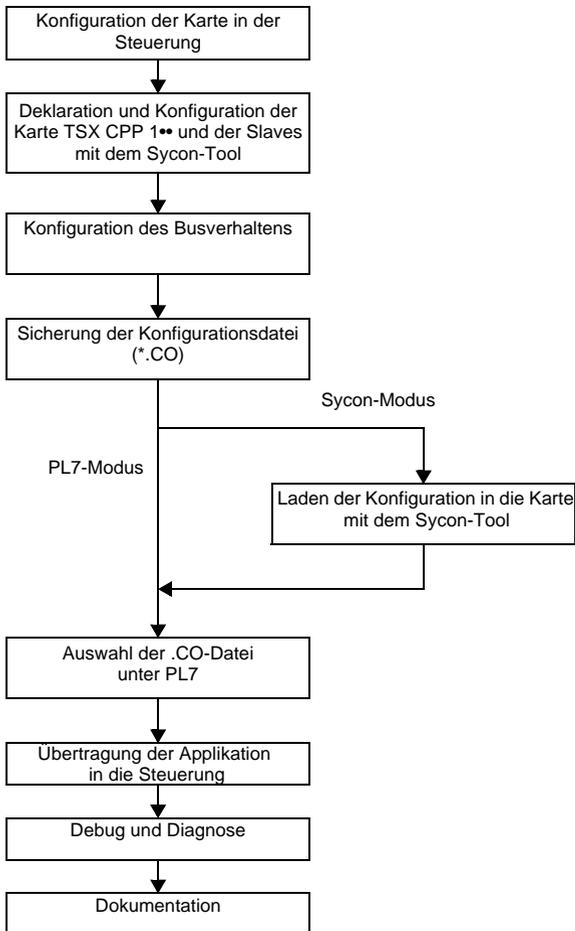
Modus	Phase	Beschreibung
Offline	Deklaration der PCMCIA-Karte TSX CPP 100/110	Die Karte darf nur in den PCMCIA-Steckplatz des Typs III des Prozessors eingesetzt werden.
	Konfiguration	<ul style="list-style-type: none"> <li>● Eingabe der Konfigurationsparameter</li> <li>● Deklaration der Buskonfiguration mittels der <b>Sycon</b>-Software und Erzeugung der Konfigurationsdatei *.CO</li> <li>● Auswahl der Konfigurationsdatei (*.CO) unter PL7</li> </ul>
Offline oder Online	Symbolisierung	Symbolisierung der der CANopen-Karte zugeordneten Variablen
	Programmierung	Programmierung spezieller Funktionen: <ul style="list-style-type: none"> <li>● zugeordnete Bit- und Wortobjekte</li> <li>● spezielle Anweisungen</li> </ul>
Online	Übertragung	Übertragung der Applikation an die Steuerung Übertragung der Applikation an die Steuerung oder Kaltstart der konfigurierten Applikation und Start der Karte TSX CPP 1●●
	Debuggen Diagnose	Für das Debugging der Applikation, die Ansteuerung der Eingänge/Ausgänge und die Fehlerdiagnose gibt es verschiedene Mittel: <ul style="list-style-type: none"> <li>● PL7-Sprachobjekte</li> <li>● Debug-Fenster unter PL7</li> <li>● Anzeige-LEDs</li> </ul>
Offline oder Online	Dokumentation	Gedruckte Informationen bezüglich der Konfiguration der Karte TSX CPP 1●●.

**Hinweis:** die oben angegebene Reihenfolge stellt lediglich einen Anhaltspunkt dar. In der PL7-Software lassen sich die Editoren interaktiv in der von Ihnen gewünschten Reihenfolge verwenden. Man kann jedoch nicht mit dem Daten- oder Programmeditor arbeiten, ohne vorher die Konfiguration durchgeführt zu haben.

## Vorgehensweise bei der Inbetriebnahme

### Übersicht

Der folgende Logikplan zeigt die Vorgehensweise bei der Inbetriebnahme einer Karte TSX CPP 1\*\*.



## **Empfehlungen**

Wenn die Konfiguration des CANopen-Busses zu umfangreich ist, um in den PL7-Modus zu wechseln, empfehlen sich die folgenden Überprüfungen:

- Prüfen Sie, ob Sie im PL7-Modus bleiben können, wenn Sie einen leistungsfähigeren Prozessor (siehe *Prozessortypen und Speicherkapazitäten, S. 29*) verwenden.
  - Kommt ein leistungsfähigerer Prozessor nicht infrage, wechseln Sie in den Sycon-Modus.
-

---

## 3.2 Konfiguration

---

### Auf einen Blick

#### Inhalt des Abschnitts

Dieser Abschnitt beschreibt die Konfiguration der PCMCIA-Karte TSX CPP 1\*\*.

---

#### Inhalt dieses Abschnitts

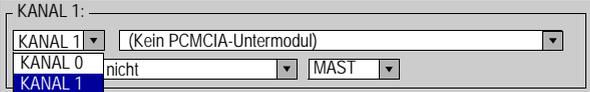
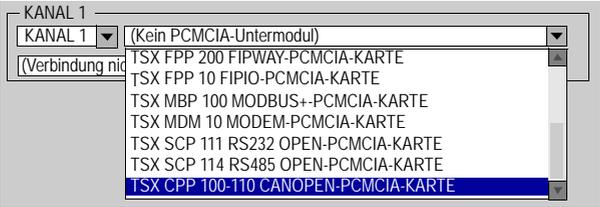
Dieser Abschnitt enthält die folgenden Themen:

Thema	Seite
Zugriff auf das Konfigurationsfenster	38
Konfigurationsbildschirm	39
Beschreibung der E/A Daten und des Busverhaltens beim Starten	42
Beschreibung der Wahlmöglichkeiten für den Konfigurations-Lademodus	45
So laden Sie eine Konfiguration mit X-Way	47
Fenster für die Konfiguration der Slaves	50
Konfiguration einer CANopen-PCMCIA-Karte	51
Konfigurationsdokument der PCMCIA-Karte TSX CPP 100/110	54

---

## Zugriff auf das Konfigurationsfenster

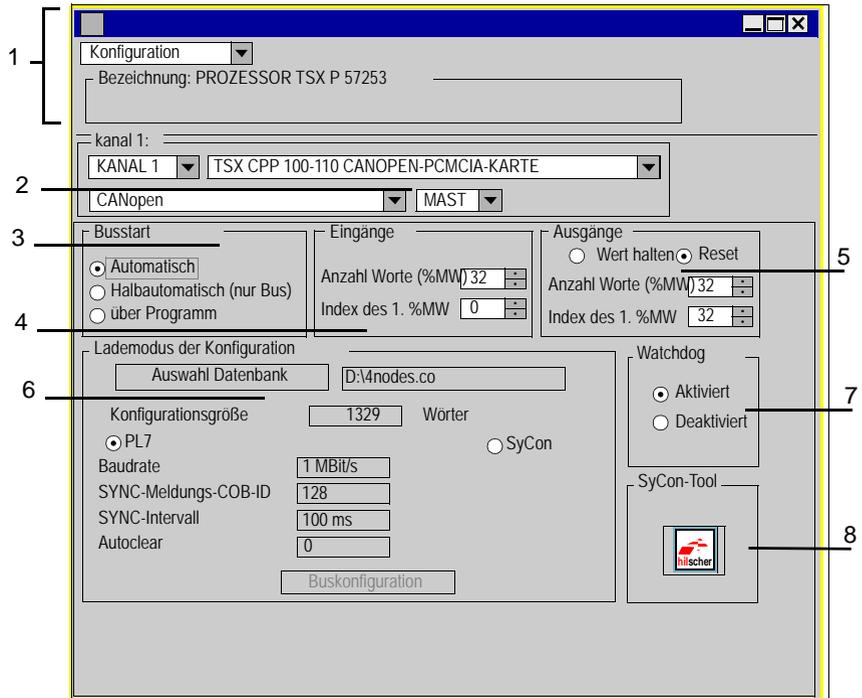
**Vorgehensweise** Mit diesem Verfahren wird eine TSX CPP 100- oder TSX CPP 110-Karte im PCMCIA-Steckplatz des Typs III des Prozessors deklariert. Das Beispiel unten beschreibt die durchzuführenden Schritte.

Schritt	Aktion
1	Öffnen Sie den Hardware-Konfigurationseditor im Applikationsnavigator.
2	Öffnen Sie das Konfigurationsfenster des Prozessor-Kommunikationskanals, indem Sie auf den Bereich <b>Komm</b> des Prozessors doppelklicken.
3	<p>Wählen Sie den Kommunikationskanal <b>KANAL 1</b> im Dropdown-Menü aus.</p> <p><b>Beispiel</b></p> 
4	<p>Wählen Sie im Dropdown-Menü die PCMCIA-Karte TSX CPP 100-110 aus:</p> <p><b>Beispiel</b></p>  <p><b>Ergebnis:</b> Ein Konfigurationsbildschirm für den CANopen-Bus wird angezeigt (siehe <i>Konfigurationsbildschirm, S. 39</i>).</p>

## Konfigurationsbildschirm

**Auf einen Blick** Dieser Bildschirm ermöglicht die Deklaration des Kommunikationskanals und die Konfiguration der Parameter einer CANopen-Verbindung.

**Abbildung** Das für die CANopen-Kommunikation vorgesehene Fenster sieht wie folgt aus:



## Elemente und Funktionen

Diese Tabelle beschreibt die verschiedenen Bereiche des Konfigurationsfensters:

Bereich	Nummer	Funktion
Allen gemeinsam	1	Dieser Bereich umfasst folgende Elemente: <ul style="list-style-type: none"> <li>● eine Titelleiste, die die Bestellreferenz und die Position des Moduls angibt</li> <li>● ein Dropdown-Listefeld, das die Auswahl des <b>Konfigurations-</b> oder <b>Debug-</b> Modus ermöglicht (nur im Online-Modus)</li> <li>● einen Bereich, der die Bezeichnung des ausgewählten Moduls enthält</li> </ul>
Spezifische Informationen	2	Dieses Dropdown-Listefeld besteht aus zwei Optionen, die das Aktualisierungsintervall der mit den E/A verknüpften Speicherbereiche angeben: <ul style="list-style-type: none"> <li>● MAST: Intervall der MAST-Task</li> <li>● FAST: Intervall der FAST-Task</li> </ul>
	3	Dieser Bereich ermöglicht die Auswahl des Bus-Anlaufverhaltens.
	4	Dieser Bereich ermöglicht die Konfiguration der Adresse (interner Speicher der Steuerung), an die regelmäßig die Eingänge der CANopen-Geräte kopiert werden.
	5	Dieser Bereich ermöglicht die Konfiguration des Fehlermodus der Ausgänge der Busgeräte sowie die Adresse (interner Speicher der Steuerung), an der regelmäßig die Ausgänge der CANopen-Geräte gelesen werden.
	6	Dieser Bereich ermöglicht die Konfiguration des Busses: <ul style="list-style-type: none"> <li>● Wahl der Sycon-Konfigurationsdatei (*.CO) (siehe <i>Auswahl einer Konfigurationsdatei</i>, S. 51)</li> <li>● PL7- oder Sycon-Konfiguration (siehe <i>Beschreibung der Wahlmöglichkeiten für den Konfigurations-Lademodus</i>, S. 45)</li> </ul>
	7	Dieser Bereich ermöglicht die Aktivierung oder Deaktivierung des Watchdog des CANopen-Bus. Standardmäßig ist der Watchdog aktiviert. Er wird ausgelöst, wenn die PCMCIA-Karte den Bus nicht mehr richtig verwalten kann. Wenn er ausgelöst wird, setzt er die Ausgänge der Slaves auf Null.
	8	Diese Schaltfläche ermöglicht den Start der <b>Sycon</b> -Software, wenn diese auf dem PC installiert ist.

	<b>WARNUNG</b>
	<p>er wird empfohlen, das Busverhalten sorgfältig zu beobachten, wenn der Watchdog deaktiviert ist.</p> <p><b>Die Nichtbeachtung dieser Anweisungen kann zu Tod, schwerer Körperverletzung oder Materialschäden führen.</b></p>

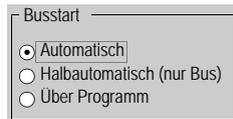
## Beschreibung der E/A Daten und des Busverhaltens beim Starten

---

**Auf einen Blick** Mit dem Konfigurationsfenster kann das Busverhalten beim Start von PL7 konfiguriert werden sowie die Ein- und Ausgänge der Slave-Geräte am Bus.

---

**Busstart** Diese Abbildung erläutert den Konfigurationsbereich beim Busstart.



Der Busstart kann auf drei Wege erfolgen:

- **Automatisch:** Die Buskonfiguration, die Kommunikationsverwaltung und die Aktualisierung der E/A der Slaves werden beim Starten ohne Eingreifen der Applikation gestartet.
- **Halbautomatisch :** Die Buskonfiguration und die Kommunikationsverwaltung werden beim Starten gestartet, die Verwaltung der E/A muss von der Applikation mit Hilfe der zugehörigen Sprachobjekte (siehe *Ausgangs-Wortobjekte*, S. 87) bestätigt werden.
- **mit Programm (Bus und E/A) :** Der Busstart muss komplett von der Applikation mit Hilfe der zugehörigen Sprachobjekte (siehe *Ausgangs-Wortobjekte*, S. 87) verwaltet werden.

**Hinweis:** im automatischen Modus startet der Bus automatisch den Datenaustausch, wenn ein Busfehler auftritt und wieder verschwindet. Die Fehlerinformationen werden mithilfe der Fehlerbits gespeichert (%MWy.1.2:x0 und %ly.MOD.ERR werden auf 1 gesetzt). Die Bestätigung erfolgt durch das Setzen von Bit %QWy.1.0:x3=1 auf 1 per Anwendung oder Animationstabelle.

---

## Eingänge

Diese Abbildung erläutert die Konfigurationsbereiche der Eingänge.

Um die Slave-Eingänge zu konfigurieren, muss der Speicherbereich, in dem diese periodisch kopiert werden, angegeben werden. Um diesen Bereich zu definieren, muss folgendes angegeben werden:

- **Anzahl Wörter:** Entspricht der Anzahl von Eingangswörtern, die der Größe der mit der **Sycon**-Software konfigurierten Eingangsdaten entspricht.
- **Adresse des ersten Wortes** : entspricht der Adresse des ersten Wortes des Speicherbereichs der Eingänge.

**Hinweis:** die %MW enthalten die Werte der Eingänge der Bus-Slaves. Im PL7-Lademodus können die den Slaves zugeordneten %MW angezeigt werden, in dem man auf die Schaltfläche **Buskonfiguration**. Im SyCon-Lademodus ist diese Schaltfläche deaktiviert. Die den Slaves zugeordneten %MW können folgendermaßen ermittelt werden: Index des ersten Merkerwortes + E/A-Adresse des Slaves (siehe SyCon-Software im Bildschirm **Ansicht** → **Adresstabelle ...**).

## Ausgänge

Diese Abbildung erläutert den Ausgangs-Konfigurationsbereich.

Um die Ausgänge zu konfigurieren, muss wie bei den Eingängen die Worttabelle angegeben werden, die den Wert der Busausgänge, aber auch das gewünschte Verhalten im Fehlerfall bei Auftreten eines Fehlers beim Slave beinhaltet:

- Wert halten,
- RESET (wieder auf Null stellen (RAZ))

**Hinweis:** die %MW enthalten die Ausgangswerte der Bus-Slaves. Im PL7-Lademodus können die den Slaves zugeordneten %MW angezeigt werden, in dem man auf die Schaltfläche **Buskonfiguration**. Im SyCon-Lademodus ist diese Schaltfläche deaktiviert. Die mit den Slaves verbundenen %MW können auf folgende Weise erfasst werden: Index des ersten Merkerwortes + E/A-Adresse des Slaves (siehe SyCon-Software im Bildschirm **Ansicht** → **Adresstabelle ...**).

**Hinweis:** die Worttabellen befinden sich im internen Speicher der Steuerung. Beide Bereiche dürfen sich nicht überlappen, da dies zu einem Fehlbetrieb der Anwendung führt.

**Hinweis:** wenn die Anzahl der Wortreferenzen %MW der Eingänge und Ausgänge sich von der Anzahl der für die Konfigurationsdatei bestimmten Anzahl unterscheidet (Namendatei \*.CO), teilt es die Software PL7 bei der Validation der Konfiguration mit.

**Hinweis:** die maximal zulässige Größe des Speicherbereichs, der für die E/A vorgesehenen ist, hängt vom Prozessortyp und der assoziierten Task (siehe *Prozessortypen und Speicherkapazitäten, S. 29*) ab.

**Hinweis:** wird das Wort %SW9 benutzt (Bit %SW9:X0 = 1 für Mast-Task, Bit %SW9:X1 = 1 für Fast-Task), werden die zur jeweiligen Task gehörenden Ausgänge der Module am Bus auf dem letzten Wert eingefroren.

---

---

## Beschreibung der Wahlmöglichkeiten für den Konfigurations-Lademodus

---

**Auf einen Blick** Das Konfigurationsfenster ermöglicht die Auswahl des Konfigurations-Lademodus. Sie haben die zwei folgenden Möglichkeiten:

- Laden per PL7-Software,
- Laden per Sycon-Software.

In beiden Fällen müssen Sie die mit der Sycon-Software erstellte Konfigurationsdatenbank auswählen (siehe *Auswahl einer Konfigurationsdatei*, S. 51).

---

**Abbildung** Die folgende Abbildung zeigt den Bereich des Konfigurationsfensters, in dem Sie den Lademodus der Konfiguration wählen können.

Lademodus der Konfiguration

Auswahl Datenbank: Dateien\Schneider\SyCon\Project\Schneider.co

Konfigurationsgröße: 1329 Wörter

PL7  SyCon

Baudrate: 125 KBits/s

SYNC-Meldungs-COB-ID: 128

SYNC-Intervall: 100 ms

Autoclear: 0

Bus-Konfiguration

---

**Beschreibung**

Die folgende Tabelle beschreibt die verschiedenen Wahlmöglichkeiten.

Bereich	Beschreibung
Auswahl Datenbank	In diesem Bereich wählen Sie die Datenbank aus, die der Konfiguration des von der PCMCIA-Karte TSX CPP 1•• verwalteten Busses entspricht. <b>Hinweis:</b> Die ausgewählte Datenbank (Datei) muss an der angegebenen Position vorhanden bleiben, da ansonsten beim nächsten Öffnen des Konfigurationsfensters eine Fehlermeldung angezeigt wird, die Sie darüber informiert, dass die Datei nicht gefunden werden konnte. Der Zugriff auf das Konfigurationsfenster des Busses wird nicht möglich sein. Diese Konfiguration erfolgt mittels der Sycon-Software, die die Datei *.CO anlegt, die Sie auswählen müssen (siehe <i>Auswahl einer Konfigurationsdatei</i> , S. 51).
PL7	Wird diese Schaltfläche betätigt, wird die Buskonfiguration mit der Applikation PL7 geladen. Ist die Anwendung zu umfangreich (Speichergröße größer als die für den Prozessor zulässige Größe), lässt die Software PL7 die Auswahl nicht zu. Wählen Sie dann <b>Sycon</b> wie unten beschrieben.
Sycon	Wird diese Schaltfläche betätigt, wird davon ausgegangen, dass die Buskonfiguration bereits durch den PC und die Sycon-Software in die PCMCIA-Karte geladen wurde. Die Software PL7 überprüft lediglich, ob die Konfiguration der Karte mit der in der ausgewählten *.CO-Datei übereinstimmt. Dagegen ist keinerlei Änderung der Busparameter in der Software PL7 möglich. Verwenden Sie zur Änderung der Konfiguration die <b>Sycon</b> -Software.
Baudrate	Wenn der PL7-Lademodus gewählt ist, wird in diesem Bereich die in Sycon festgelegte Baudrate auf dem Bus angezeigt.
COB-ID Meldung SYNC	Wenn der PL7-Lademodus gewählt ist, wird in diesem Bereich die in Sycon gewählte COB ID für SYNC angezeigt.
SYNC-Intervall	Wenn der PL7-Lademodus gewählt ist, wird in diesem Bereich die in Sycon ausgewählte Busperiode angezeigt.
Autoclear	Wenn der PL7-Lademodus gewählt ist, wird in diesem Bereich der in Sycon gewählte Modus <b>Autoclear on</b> oder <b>Autoclear off</b> angezeigt.
Buskonfiguration	Wenn der PL7-Lademodus gewählt ist, ermöglicht diese Schaltfläche den Zugriff auf die Konfiguration der an den Bus angeschlossenen Slaves.

## So laden Sie eine Konfiguration mit X-Way

### Allgemeines

Wenn der CANopen-Bus im Modus für das Laden durch Sycon konfiguriert ist, können Sie die Konfiguration der TSX CPP 1••-Karte mit Hilfe des X-Way-Treibers laden.

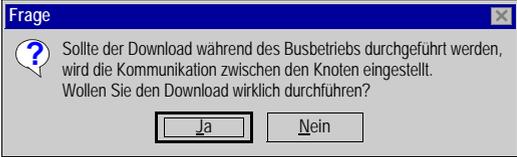
Dieses Laden kann über ein Ethernet-Netzwerk erfolgen oder ganz einfach mit einem Unitelway-Bus.

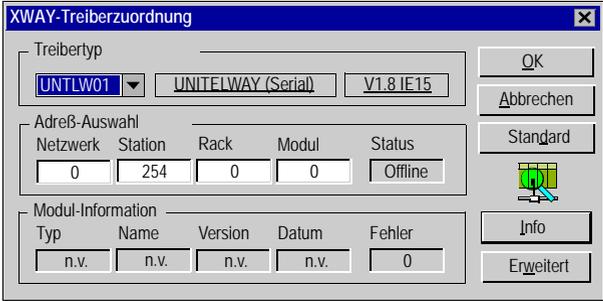
**Hinweis:** ACHTEN SIE darauf, die Steuerung beim Ladevorgang auf STOP zu stellen.

**Hinweis:** wenn Sie die Wahl des X-Way-Kommunikationstreibers bestätigt haben, müssen Sie das Programm Sycon verlassen, um den Treiber zu wechseln.

### Vorgehensweise

Diese Tabelle beschreibt die auszuführenden Schritte zum Laden der Konfiguration einer CANopen-Karte mittels des X-Way-Kommunikationstreibers.

Schritt	Aktion
1	Verbinden Sie sich mit Hilfe der Software PL7 V4 mit der Steuerung, die die Karte TSX CPP 1•• enthält.
2	Schalten Sie den Betriebszustand der Steuerung auf STOP.
3	Starten Sie die Sycon-Software.
4	Laden oder erstellen Sie die gewünschte Konfiguration mit Hilfe von Sycon.
5	<p>Wählen Sie den Befehl <b>Online</b> → <b>Download</b>.</p> <p><b>Ergebnis:</b> Es erscheint eine Meldung, die darauf hinweist, dass beim Laden der Konfiguration die Kommunikation zwischen den Slaves gestoppt wird.</p> 

Schritt	Aktion
6	<p>Klicken Sie auf <b>JA</b> um anzugeben, dass Sie mit dieser Unterbrechung zwischen den Slaves einverstanden sind.</p> <p><b>Ergebnis:</b> Ein Auswahlfenster des <b>X-Way-</b> oder <b>CIF-</b> Treibers erscheint.</p> 
7	<p>Wählen Sie den <b>X-Way-</b> Treiber und klicken Sie auf <b>OK</b>.</p> <p><b>Ergebnis:</b> Das Fenster <b>XWAY-Treiberzuordnung</b> erscheint.</p> 
8	<p>Wählen Sie den gewünschten Treiber (Uni-telway, XIP,...) im Bereich <b>Treiberauswahl</b>.</p>
9	<p>Geben Sie die Adresse der Steuerung ein (Netzwerk, Station, Rack, Modul) und klicken Sie auf <b>OK</b>, um den Ladevorgang zu starten.</p> <p><b>Ergebnis:</b> Solange der Ladevorgang läuft, zeigt ein Fenster den Fortschritt der Datenübertragung an. Am Übertragungsende verschwindet dieses Fenster und an der Stelle erscheint das Hauptfenster der Buskonfiguration.</p>

**SyCon-Befehl  
"Online →  
Firmware /  
Reset"**

Genau wie der SyCon-Befehl "**Online** → **Download**", bewirkt der Online-Befehl "**Reset**", dass alle Slaves vom Bus getrennt werden.

**Hinweis:** nach Empfang des Sycon-Befehls "Reset" reicht es für den Start der Karte TSX CPP 1•• nicht mehr aus, dass die Steuerung in RUN geht: Um die PCMCIA-Karte betriebsbereit zu machen, muss die Steuerung neu initialisiert (RESET-Taste) oder aus- und wieder eingeschaltet werden.

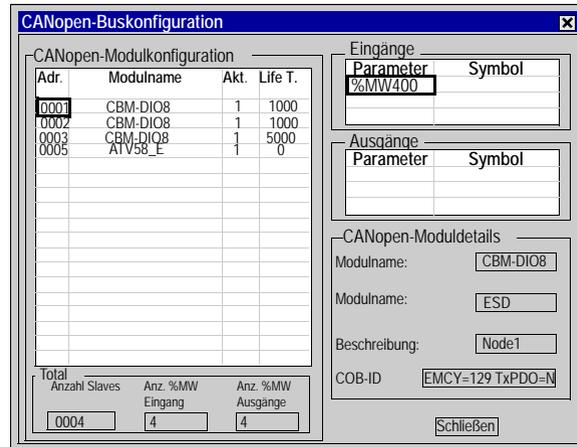
## Fenster für die Konfiguration der Slaves

### Auf einen Blick

Mit dem Programm PL7 können Sie auf die Konfiguration der Bus-Slaves zugreifen. Die im Fenster enthaltenen Informationen sind nahezu identisch mit denen des Debuggingfensters (siehe *Beschreibung des Debug-Fensters*, S. 69).

### Abbildung

Die folgende Abbildung zeigt das Fenster für die Konfiguration der Slaves



### Funktionsweise

- Klicken Sie auf einen Slave aus der Liste der CANopen-Slaves.
- Die Eingangs- und Ausgangsparameter des Slave erscheinen dann im Bereich **Eingänge** und **Ausgänge**.
- Die Informationen über den Slave erscheinen im Bereich **Einzelheiten CANopen-Geräte**.

## Konfiguration einer CANopen-PCMCIA-Karte

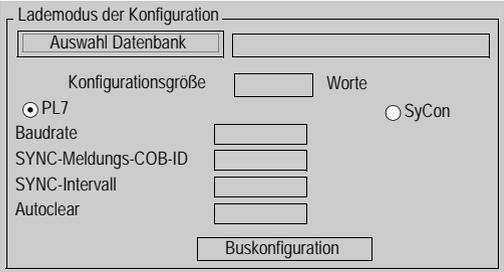
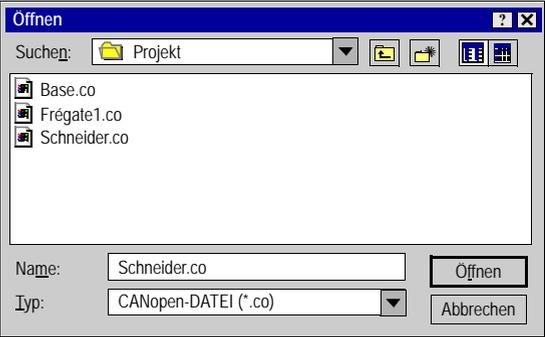
### Auf einen Blick

Beim Konfigurieren einer Karte TSX CPP 1\*\* müssen bestimmte Verfahren eingehalten werden oder es sind spezielle Erklärungen nötig. Diese sind in den nachfolgend aufgeführten Vorgehensweisen beschrieben.

**Hinweis:** wenn die \*.CO-Konfigurationsdatei ausgewählt ist (siehe folgendes Verfahren), muss diese immer vorhanden sein. Befindet sich die angegebene Datei beim nächsten Öffnen des Konfigurationsfensters nicht an der angegebenen Position, wird eine Fehlermeldung angezeigt, die Sie darüber informiert, dass die konfigurierte Datei nicht gefunden werden konnte. Der Zugriff auf das Konfigurationsfenster des Busses wird nicht möglich sein.

### Auswahl einer Konfigurationsdatei

Diese Tabelle beschreibt die für die Auswahl einer CANopen-Konfiguration zu befolgenden Schritte.

Schritt	Aktion
1	<p>Klicken Sie auf die Schaltfläche <b>Auswahl Datenbank:</b></p>  <p><b>Ergebnis:</b> Ein Fenster der folgenden Art erscheint:</p> 

Schritt	Aktion
2	<p>Wählen Sie die gewünschte *.CO-Datei und klicken Sie dann auf <b>Öffnen</b>.  <b>Ergebnis:</b> Wenn die Anzahl der für Eingänge und Ausgänge reservierten Worte mit der gewählten Konfiguration übereinstimmt, erscheint die Konfiguration im PL7-Konfigurationsfenster.</p> <div data-bbox="504 321 1022 592" style="border: 1px solid gray; padding: 5px;"> <p>Lademodus der Konfiguration</p> <p>Auswahl Datenbank: Files\Schneider\SyCon\Project\Schneider.co</p> <p>Konfigurationsgröße: 1329 Wörter</p> <p><input checked="" type="radio"/> PL7 <span style="margin-left: 100px;"><input type="radio"/> SyCon</span></p> <p>Baudrate: 125 KBits/s</p> <p>SYNC-Meldungs-COB-ID: 128</p> <p>SYNC-Intervall: 100 ms</p> <p>Autoclear: 0</p> <p style="text-align: center;">Bus-Konfiguration</p> </div> <p>Ist dies nicht der Fall, informiert Sie eine Warnmeldung darüber, dass die Anzahl der Eingangs- oder Ausgangswörter nicht mit der ausgewählten Datei übereinstimmt.</p> <p><b>Beispiel:</b></p> <div data-bbox="493 727 1086 917" style="border: 1px solid gray; padding: 5px;"> <p style="background-color: #000080; color: white; padding: 2px;">CANOPEN <span style="float: right;">✖</span></p> <p> Sie haben mehr Wörter als nötig reserviert.  Die aktuelle Konfiguration des Kanals TSX CPP 100-110 erfordert vier Ein- und vier Ausgangswörter.</p> <p style="text-align: center;">OK</p> </div> <p>Überschreitet die *.CO-Datei die Maximalkapazität für Konfigurationsdaten (siehe <i>Prozessortypen und Speicherkapazitäten</i>, S. 29), wird eine Fehlermeldung angezeigt. Sie müssten dann entweder einen größeren Premium-Prozessor einsetzen oder den <b>Sycon</b>-Modus aufrufen und die Konfiguration mithilfe der <b>Sycon</b>-Software direkt in die Karte laden.</p>

**Für die  
Konfiguration  
einer CANopen-  
Karte zu  
befolgende  
Schritte**

Die folgende Tabelle beschreibt die für die Konfiguration einer CANopen-PCMCIA-Karte TSX CPP 1\*\* zu befolgenden Schritte.

Schritt	Aktion
1	Wählen Sie den Starttyp des Busses aus.
2	Klicken Sie auf die Schaltfläche <b>Sycon-Tool</b> , um die <b>Sycon</b> -Konfigurationssoftware zu starten.
3	Konfigurieren Sie mit der <b>Sycon</b> -Software Ihren CANopen-Bus abhängig von den für den Bus vorgesehenen Geräten.
4	Notieren Sie die für die Ein- und Ausgänge zu reservierenden Puffergrößen, damit Sie die Steuerungstabellen konfigurieren können.
5	Sichern Sie die Konfiguration in einer Datei mit der Erweiterung "CO".
6	Kehren Sie zu PL7 zurück.
7	Geben Sie die Anzahl der Ein- und Ausgangswörter ein. Verwenden Sie dazu die Werte, die Sie in Punkt 4 notiert haben. <b>Hinweis:</b> Achtung, jedes %MW besteht aus zwei Bytes.
8	Geben Sie die Tabellenanfangsadressen so ein, dass es keine Überschneidung zwischen den Eingängen und Ausgängen gibt.
9	Wählen Sie eine Konfigurationsdatei (siehe <i>Auswahl einer Konfigurationsdatei</i> , S. 51).
10	Wählen Sie das Optionsfeld PL7, wenn Sie die CANopen-Konfiguration in die PL7-Applikation einbetten möchten.
11	Klicken Sie auf die Schaltfläche <b>Aktiviert</b> des Watchdogs.
12	Bestätigen Sie die Konfiguration.

## Konfigurationsdokument der PCMCIA-Karte TSX CPP 100/110

**Auf einen Blick** Mit Hilfe des Prozessormoduls ist im Dokumentationseditor von PL7 ein Dokument verfügbar, das die Konfiguration der Applikation für die PCMCIA-Karte TSX CPP 100/110 beschreibt.

**Abbildung** Dieses Dokument hat folgendes Aussehen:

TSX 57353 [RACK 0 POSITION 0]			
<b>Modulkennung</b>			
<b>Bestellreferenz:</b>	TSX57353	<b>Bezeichnung:</b>	PROZESSOR TSX P 57353
<b>Adresse:</b>	000	<b>Symbol:</b>	
<b>Kanalparameter: 0</b>			
<b>Zuordnung Task/Kanal:</b>	MAST		
Kanaltyp:	PG-Anschluss	Kanalsymbol:	
Applikationsspezifische Funktion:	VERBINDUNG UNI-TELWAY	Kanalsymbol:	
Übertragungsgeschwindigkeit	19200 Bits/s	Verzögerung:	30 ms
Modultyp:	Master	Parität:	ungerade
Slave-Anzahl:	8		
<b>Kanalparameter: 1</b>			
<b>Zuordnung Task/Kanal:</b>	MAST		
Untermodytyp:	TSX CPP 100-110 PCMCIA-KARTE CANOPEN		
Kanaltyp:	PCMCIA-Port	Kanalsymbol:	
Applikationsspezifische Funktion:	CANOPEN		
Eingänge	Adresse des ersten %MW	32	Länge: 424
Ausgänge	Adresse des ersten %MW	1056	Länge: 102
	Fehlermodus der Ausgänge: Reset		
Konfigurationsmodus:	Automatisch	Watchdog TSX CPP 100-110:	Aktiv
Lademodus:	PL7	Konfigurationsdatei CANOPEN:	E:\DB1.CO

Konfig. des CANopen-Busses:	Übertragungsgeschwindigkeit:	1 MBits/s	COB-ID Synchr.-Meldung:	128	
	Auto-Clear:	off	Periode Synchr.-Meldung:	100 ms	
Slave-Konfiguration CANOPEN					
	Adr.	Typ	Akt./Guard poll		
	1	ATV58_F	1 / 1		
	2	Profil 401 EDS-Standard	1 / 0		
	3	CBM-DIO8	1 / 0		
Sprachobjekte CANOPEN-Slave:					
	Adr.	Eingänge	Symbol	Ausgänge	Symbol
	1.	%MW32		%MW1056	
		%MW33			
		%MW34			
		%MW35		%MW1057	
		%MW36		%MW1058	

## 3.3 Programmierung

---

### Auf einen Blick

---

#### Inhalt des Abschnitts

Dieser Abschnitt beschreibt die für Programmierung der Funktionen und Abfrage von Informationen eines CANopen-Busses erhältlichen Tools, der von der PCMCIA-Karte TSX CPP 1\*\* verwaltet wird.

Die CANopen-Busfunktionen können mithilfe von UNI-TE-Requests programmiert werden:

- Senden und Empfangen von SDO-Meldungen über den Bus
- Zugriff auf den Link Layer durch Senden von PDUs

Es ist auch möglich, den Bus und seine Funktion zu überwachen:

- Identifizieren des Masters
- Senden von Diagnose-Requests an die Busgeräte

Diese Requests werden an den CANopen-Master (PCMCIA-Karte TSX CPP 1\*\*) gesendet, der sie verarbeitet.

---

#### Inhalt dieses Abschnitts

Dieser Abschnitt enthält die folgenden Themen:

Thema	Seite
Zugriff auf SDOs von CANopen	57
Zugriff auf die PDUs der Verbindungsschicht CAN	62
Request IDENTIFICATION	64
Diagnosebefehl	67

---

## Zugriff auf SDOs von CANopen

### Auf einen Blick

Die Kommunikationsfunktionen `READ_VAR` und `WRITE_VAR` ermöglichen den Zugriff auf den Transfer der Daten vom Typ SDO CANopen. Die Parameter dieser Funktionen bestimmen die auszuführende Aktion.

Diese Dienste beruhen auf dem standardisierten Nachrichtendienst CMS des CANopen-Standards. Siehe Dokumentation der CANopen-Slaves, wenn Sie mehr über die Formate der verwendeten SDOs erfahren wollen.

### Syntax der Funktion `WRITE_VAR`

Die Syntax der Kommunikationsfunktion sieht wie folgt aus:

```
WRITE_VAR(ADR#y.1.SYS, 'SDO', subindex:index, NodeID, %MWi:L, %MWk:4)
```

Die folgende Tabelle beschreibt die verschiedenen Parameter der Funktion.

Parameter	Beschreibung
ADR#y.1.SYS	Empfängeradresse des Austauschs. <ul style="list-style-type: none"> <li>● y: Position des Prozessors im Rack (0 oder 1)</li> <li>● 1: Kanal (immer 1)</li> <li>● SYS Server UNI-TE der PCMCIA-Karte</li> </ul>
'SDO'	SDO-Objekttyp (immer <b>SDO</b> in Großbuchstaben)
Unterindex:Index	Doppelwort oder Sofortwert, die den Index und den Unterindex SDO CANopen identifizieren: Das MSB im Doppelwort enthält den Unterindex des Index und das LSB den Index.
NodeID	Wort oder Wert, der den Empfänger auf dem CANopen-Bus identifiziert
%MWi:L	Worttabelle, die die zu sendenden Daten enthält (Mindestlänge = 1)
%MWk:4	Parameter zur Austauschverwaltung: vier Wörter, welche die verwendete PL7-Datenadresse zum Steuern der Funktion <code>WRITE_VAR</code> identifizieren.

**Syntax der  
Funktion  
READ\_VAR**

Die Syntax der Kommunikationsfunktion sieht wie folgt aus:

```
READ_VAR(ADR#y.1.SYS, 'SDO', subindex:index, NodeID, %MWi:L,  
%MWk:4)
```

Die folgende Tabelle beschreibt die verschiedenen Parameter der Funktion.

Parameter	Beschreibung
ADR#y.1.SYS	Empfängeradresse des Austauschs. <ul style="list-style-type: none"><li>● y: Position des Prozessors im Rack (0 oder1)</li><li>● 1 : Kanal (immer 1)</li><li>● SYS Server UNI-TE der PCMCIA-Karte</li></ul>
'SDO'	SDO-Objekttyp (immer <b>SDO</b> in Großbuchstaben)
Unterindex:Index	Doppelwort oder Sofortwert, die den Index und den Unterindex SDO CANopen identifizieren: Das MSB im Doppelwort enthält den Unterindex des Index und das LSB den Index.
NodeID	Wort oder Wert, der den Empfänger auf dem CANopen-Bus identifiziert
%MWi:L	Worttabelle, die die zu empfangenden Daten enthält (Mindestlänge =1)
%MWk:4	Parameter zur Austauschverwaltung: vier Wörter, welche die verwendete PL7-Datenadresse zum Steuern der Funktion READ_VAR identifizieren.

**Verwaltungsparameter** In der folgenden Tabelle sind die Wörter %MWk:4 aufgeführt.

Wortnummer	Höchstwertiges Byte	Niederwertiges Byte	Daten verwaltet durch
%MWk	Austauschnummer	Aktivitätsbit	System
%MWk+1	<p>Wert der Operationsrückmeldung: Beinhaltet alle Fehler zu CANopen-Diensten</p> <p>16#00: Positives Ergebnis 16#01: Request nicht verarbeitet 16#02: Antwort ungültig 16#03: Reserviert</p>	<p>Wert der Kommunikationsrückmeldung: Beinhaltet alle Fehler für die Funktionsbausteine 'READ_VAR', 'WRITE_VAR' und 'SEND_REQ'</p> <p>16#00: Korrekter Austausch</p>	
	<p>16#00 S. 'Beispiel für Verwaltungsparameter' weiter unten.</p>	<p>16#01: Abbruch des Austauschs durch Timeout 16#02: Abbruch des Austauschs durch Bedieneranforderung (CANCEL) 16#03: Adressformat ungültig 16#04: Empfängeradresse ungültig 16#05: Verwaltungsparameter-Format nicht korrekt 16#06: Spezifische Parameter ungültig 16#07: Problem beim Senden an Empfänger 16#09: Empfangspuffer zu klein 16#08: Reserviert 16#0A: Sendepuffer zu klein 16#0B: Fehlende Systemressource des Prozessors 16#0C: Ungültige Austauschnummer 16#0D: Kein Telegramm empfangen 16#0E: Ungültige Länge 16#0F: Telegrammdienst nicht konfiguriert 16#10: Buskoppler nicht vorhanden 16#11: Request nicht vorhanden 16#12: Anwendungsserver bereits aktiv 16#13: Transaktionsnr. UNI-TE V2 nicht korrekt</p>	

Wortnummer	Höchstwertiges Byte	Niederwertiges Byte	Daten verwaltet durch
%MWk+1	16#01: Fehlende Ressource zum Prozessor 16#02: Fehlende Leitungsressource 16#04: Leitungsfehler 16#05: Längenfehler 16#06: Kommunikationskanal nicht in Ordnung 16#07: Adressierungsfehler 16#08: Anwendungsfehler 16#0B: Fehlende Systemressource 16#0C: Kommunikationsfunktion nicht aktiv 16#0D: Empfänger nicht vorhanden 16#0F: Routing-Problem zwischen Stationen oder Kanal nicht konfiguriert 16#11: Adressformat nicht verwaltet 16#12: Fehlende Empfängerressource 16#14: Verbindung nicht betriebsbereit (Beispiel : Ethernet TCP/IP) 16#15: Fehlende Ressource auf lokalem Kanal 16#16: Unautorisierter Zugriff (Beispiel : Ethernet TCP/IP) 16#17: Inkonsistente Netzkonfiguration (Beispiel : Ethernet TCP/IP) 16#18: Verbindung vorübergehend nicht verfügbar 16#21: Anwendungsserver angehalten 16#30: Sendefehler	16#FF: Nachricht zurückgewiesen	System
%MWk+2	Timeout		Bediener
%MWk+3	Länge in Byte: <ul style="list-style-type: none"> <li>● für WRITE_VAR initialisieren Sie dieses Wort mit der Anzahl der zu sendenden Bytes.</li> <li>● für READ_VAR enthält dieses Wort nach Abschluss des Requests die Anzahl der empfangenen Zeichen in der Worttabelle der Empfangsdaten.</li> </ul>		

**Hinweis:** Die Funktion kann Parameterfehler vor der Aktivierung des Austauscherkennen. Falls das Aktivitätsbit auf 0 bleibt, wird die Rückmeldung mit den Standardwerten initialisiert.

**Beispiel für  
Verwaltungsparameter**

Eine Rückmeldung der Operation von 16#00 und eine Rückmeldung der Kommunikation von 16#01 (Abbruch des Austauschs durch Timeout) kann auf eine falsch konfigurierte Portadressierung hinweisen (1 anstatt 0). In diesem Fall handelt es sich um Funktionsbaustein-Probleme. Folglich ist es nicht möglich, die CANopen-Dienste zu verarbeiten.

---

## Zugriff auf die PDUs der Verbindungsschicht CAN

**Auf einen Blick** Die Kommunikationsfunktion `SEND_REQ` ermöglicht den Zugriff auf die PDUs der Verbindungsschicht CAN.

### Syntax

Die Syntax der Kommunikationsfunktion lautet wie folgt:

```
SEND_REQ(ADR#y.1.SYS, 16#9F, %MWi:L, %MWj:L, %MWk:4)
```

Folgende Tabelle beschreibt die verschiedenen Parameter der Funktion.

Parameter	Beschreibung
ADR#y.1.SYS	Adresse der Zieleinheit des Austauschs. <ul style="list-style-type: none"> <li>● y: Position des Prozessors im Rack (0 oder 1)</li> <li>● 1: Kanal (immer 1)</li> <li>● SYS Server UNI-TE der PCMCIA-Karte</li> </ul>
16#9F	Funktionscode
%MWi:L	Tabelle mit mindestens 4 Wörtern. <ul style="list-style-type: none"> <li>● %MWi: Aktionsmodus (101: Versenden mit lokaler Bestätigung der Karte)</li> <li>● %MWi+1 und %MWi+2: Kennung auf der Ebene der Verbindungsschicht CANopen, diese beiden Wörter bilden ein Doppelwort, das einen Wert zwischen 0 und 536 870 911 (29 Bits) enthalten muss <b>Bemerkung:</b> auf die Kennungen zwischen 0 und 2047 einschließlich (11Bits) wird nur im Schreibmodus zugegriffen</li> <li>● %MWi+3 bis %MWi+6: PDU-Daten, maximal 8</li> </ul>
%MWj:L	Nicht verwendet (Länge auf 1)
%MWk:4	Parameter der Austauschverwaltung: vier Wörter, die die Adresse der PL7-Daten identifizieren, die für die Steuerung der Funktion verwendet werden

**Verwaltungsparameter** In der folgenden Tabelle sind die Wörter %MWk:4 aufgeführt.

Wortnummer	Hochwertiges Byte	Niederwertiges Byte	Daten verwaltet durch
%MWk	Austauschnummer	Aktivitätsbit	das System
%MWk+1	Rückmeldung der Operation <ul style="list-style-type: none"> <li>● korrekte Antwort: 16#61</li> <li>● falsche Antwort: 16#FD</li> </ul>	Kommunikationsrückmeldung	
%MWk+2	Timeout		Sie
%MWk+3	Länge: Initialisierung mit der Länge der Tabelle %MWi:1 obligatorisch vor dem Versenden der Funktion. Wenn der Request beendet ist, enthält das Wort die Anzahl der empfangenen Bytes.		

## Request IDENTIFICATION

---

**Auf einen Blick** Mit diesem Request lässt sich der CANopen-Bus-Master identifizieren (PCMCIA-Karte TSX CPP 1\*\*).

Der Request wird mithilfe der Kommunikationsfunktion `SEND_REQ` ausgeführt.

---

**Syntax** Die Syntax der Kommunikationsfunktion sieht wie folgt aus:

```
SEND_REQ(ADR#y.1.SYS, 16#0F, %MWi:L, %MWj:L, %MWk:4)
```

Die folgende Tabelle beschreibt die verschiedenen Parameter der Funktion.

Parameter	Beschreibung
ADR#y.1.SYS	Adresse der Zieleinheit des Austauschs <ul style="list-style-type: none"> <li>● y: Steckplatznummer des Prozessors, 0 oder 1</li> <li>● 1 : Kanal (immer 1)</li> <li>● SYS: Zugriff auf den UNI-TE-Server der PCMCIA-Karte</li> </ul>
16#0F	Requestcode
%MWi:L	Von der Funktion IDENTIFICATION nicht verwendet (Länge 1)
%MWj:L	Tabelle der Wörter mit den Identifikationsinformationen der Karte. Die Länge L muss 12 sein.
%MWk:4	Parameter zur Austauschverwaltung: vier Wörter, welche die verwendete PL7-Datenadresse zum Steuern der Funktion IDENTIFICATION identifizieren.

---

**%MWj:L**

In der folgenden Tabelle sind die Wörter %MWj:L aufgeführt.

<b>Wortnummer</b>	<b>Höchstwertiges Byte</b>	<b>Niederwertigstes Byte</b>
%MWj	Produktcode: <ul style="list-style-type: none"> <li>● 16#04 : Micro</li> <li>● 16#05 : Premium</li> </ul>	16#FF
%MWj+1	Länge der Identifikations-Zeichenkette: 16#0C	BCD-codierte Versionsnummer (Version 1.0 ist als 16#10 codiert)
%MWj+2	'S'	'T'
%MWj+3	' '	'X'
%MWj+4	'P'	'C'
%MWj+5	' '	'P'
%MWj+6	'0'	'1'
%MWj+7	16#00	'0'
%MWj+8	Status der LED-Anzeigen. Die LED "COM" ist auf die beiden ersten Bits, die LED "ERR" auf die beiden folgenden Bits gemäß den folgenden Sequenzen codiert: <ul style="list-style-type: none"> <li>● 0, 0 : Aus</li> <li>● 0, 1 : Blinkend</li> <li>● 1, 0 : Permanent leuchtend</li> </ul>	Status der PCMCIA-Karte: <ul style="list-style-type: none"> <li>● 0 : Nicht vorhanden</li> <li>● 1 : Autotest</li> <li>● 2 : Fehler</li> <li>● 3 : Bereit</li> <li>● 4 : Im Wartezustand</li> <li>● 5 : Nicht konfiguriert</li> </ul>
%MWj+9	Produkttyp: 16#02	Funktionsstyp: 16#2E
%MWj+10	Fehlertyp <ul style="list-style-type: none"> <li>● x0: Karte im Test-/Debug-Modus</li> <li>● x1: Busfehler oder Bus nicht aktiv</li> <li>● x2: Kein Anschlussgehäuse vorhanden</li> <li>● x3: Autotest oder Zugriff auf Karte nicht möglich</li> <li>● x4: Reserviert</li> <li>● x5: Karte entspricht nicht der konfigurierten Karte</li> <li>● x6: Keine Karte vorhanden</li> <li>● x7: Fehler mindestens eines Slaves</li> </ul>	Katalogreferenz: 16#01
%MWj+11	-	16#00

**Verwaltungsparameter** In der folgenden Tabelle sind die Wörter %MWk:4 aufgeführt.

Wortnummer	Höchstwertiges Byte	Niederwertigstes Byte	Daten verwaltet durch
%MWk	Austauschnummer	Aktivitätsbit	das System
%MWk+1	Rückmeldung der Operation, positive Rückmeldung 16#3F	Rückmeldung der Kommunikation	
%MWk+2	Timeout		Sie
%MWk+3	Länge: Initialisierung auf 0 vor dem Senden der Funktion erforderlich		

---

## Diagnosebefehl

**Auf einen Blick** Das Senden eines Diagnosebefehls wird mit dem Funktionsblock `SEND_REQ` durchgeführt:

In diesem Fall wird die Funktion `SEND_REQ` für folgendes benutzt:

- Diagnose der Buslave,
- Version einer PCMCIA CANopen Karte,
- um Statusvariablen zu erhalten,
- die bisherigen Busfehler zu erhalten.

### Syntax

Die Syntax der Kommunikationsfunktion ist folgende:

```
SEND_REQ (ADR#y.1.SYS, 16#0031, %MWi:3, %MWj:L, %Mwk:4)
```

Die folgende Tabelle beschreibt die verschiedenen Parameter der Funktion.

Parameter	Beschreibung
ADR#y.1.SYS	Adresse der Zieleinheit des Austauschs. <ul style="list-style-type: none"> <li>● y: Position des Prozessors im Rack (0 oder 1)</li> <li>● 1 : Kanal (immer 1)</li> <li>● SYS Server UNI-TE der PCMCIA-Karte</li> </ul>
16#0031	Anfragecode
%MWi:3	Parameter der Anfrage:: <ul style="list-style-type: none"> <li>● %MWi : Objekttyp der Diagnose: <ul style="list-style-type: none"> <li>● 1 ... 127 : Slave-Diagnose 1 bis 127</li> <li>● 128 : Version der CANopen-Karte</li> <li>● 129 : Status der CANopen-Karte, die Antwort entspricht dem Inhalt der Statuswörter %IWy.li (siehe <i>Eingangswortobjekte</i>, S. 84)</li> <li>● 130 : Historie der Fehlermeldungen</li> </ul> </li> <li>● %MWi+1 : Startadresse in der Diagnosetabelle (Standardwert 0). um ein Teilzugriff auf die Diagnosetabelle durchzuführen, ist es möglich ein Startwort in der Tabelle festzulegen (Start-Offset)</li> <li>● %MWi+2 : Länge in Bytes der zu lesende Diagnose, in der Regel entspricht diese Länge zweimal der Länge der Antworttabelle</li> </ul>
%MWj:L	Empfangstabelle enthält die Diagnosedaten (siehe <i>Die Diagnosedaten</i> , S. 75).
%Mwk:4	Parameter zur Austauschverwaltung: vier Wörter, die die verwendete PL7-Datenadresse zum Steuern der Funktion <code>SEND_REQ</code> identifizieren.

**Verwaltungsparameter** In der folgenden Tabelle sind die Wörter %MWk:4 aufgeführt.

Wortnummer	höchstwertiges Byte	niederwertiges Byte	Daten verwaltet durch
%MWk	Austauschnummer	Aktivitätsbit	das System
%MWk+1	Rückmeldung der Operation: <ul style="list-style-type: none"> <li>● positive Antwort: 16#61</li> <li>● falsche Antwort: 16#FD</li> </ul>	Rückmeldung der Kommunikation	
%MWk+2	Timeout		Anwender
%MWk+3	Länge: Anz. Bytes der Antwort <b>Hinweis</b> Initialisierung erforderlich, um die Funktion mit der Anzahl der gesendeten Bytes, d.h. 6 zu aktivieren.		

---

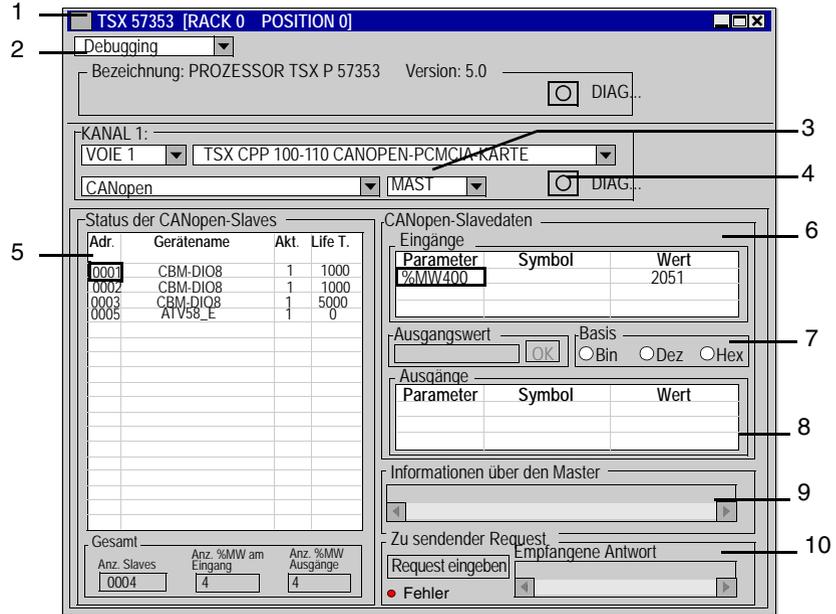
## 3.4 Debugging

### Beschreibung des Debug-Fensters

**Auf einen Blick** Die Debug-Funktion (Auswahl im Konfigurationsfenster oder Doppelklicken auf die PCMCIA-Karte TSX CPP 100-110 in der PL7-Konfiguration PL7) ist nur im Online-Modus verfügbar.

**Abbildung**

Nachfolgende Abbildung ist ein Beispiel eines Debuggingfensters.



**Beschreibung**

Die folgende Tabelle zeigt die einzelnen Elemente des Debug-Fensters:

Nummer	Element	Funktion
1	Titelleiste	Bestellreferenz und Position des Moduls
2	Auswahl der Funktion	<ul style="list-style-type: none"> <li>● Debug (nur im Online-Modus möglich)</li> <li>● Konfiguration</li> </ul>
3	Kanalbereich	Zeigt die Eigenschaften von Kanal 1 an.
4	Schaltfläche <b>DIAG</b>	<p>Wird ein Modulfehler erkannt, erhält man mit dieser Schaltfläche Informationen zum Status des entsprechenden Moduls.</p> <p><b>Hinweis:</b> Wenn die Schaltfläche <b>DIAG</b> aktiv ist (%ly.MOD.ERR ist auf 1 gesetzt) und der Datenaustausch mit den Slaves normal verläuft, zeigt dies die Speicherung eines Busfehlers an einem bestimmten Moment an. Um diesen Fehler zu bestätigen, setzen Sie das Bit %QWy.1.0:x3=1 per Anwendung oder Animationstabelle auf 1.</p>
5	Konfiguration der CANopen-Slaves	<p>Dieser Bereich zeigt alle Slaves des CANopen-Busses an. Ein fehlerhafter Slave wird rot angezeigt. Wenn der Fehler behoben ist, wird er blau angezeigt. In allen anderen Fällen wird er schwarz angezeigt. Bei der Auswahl eines Slaves werden die Bereiche 6, 8, 9 und 10 aktualisiert.</p> <p><b>Aktiv</b> : Zeigt an, ob der Slave in der Sycon-Konfiguration aktiviert wurde (1 aktiviert, 0 gesperrt).</p> <p><b>Life T.</b> : Life Time- oder Heartbeat-Dauer</p> <p><b>Hinweis:</b> Wenn die Konfigurationsdatei (siehe <i>Auswahl einer Konfigurationsdatei</i>, S. 51) nicht gefunden werden konnte, werden die Slaves nicht angezeigt.</p>
6	Eingänge	Ist ein Slave ausgewählt, enthält dieser Bereich die Liste der ihm zugeordneten Eingangswörter.
7	Ausgangswert	Wenn ein Ausgangswort im Bereich 8 gewählt wird, kann sein Wert durch Eingabe eines neuen Wertes und Anklicken der Schaltfläche <b>OK</b> geändert werden.
8	Ausgänge	Ist ein Slave ausgewählt, enthält dieser Bereich die Liste der ihm zugeordneten Ausgangswörter.
9	Informationen über ...	Ist ein Slave ausgewählt (klicken Sie in Bereich 5), enthält dieser Bereich seine letzte Diagnosemeldung. Um Informationen über die Karte TSX CPP 100 zu erhalten, genügt es, auf die Kopfzeile der Tabelle zu klicken.

---

Nummer	Element	Funktion
10	Zu sendender Request	Ist ein Slave gewählt, ermöglicht dieser Bereich das Senden eines SDO-Requests. Die Syntax der Parameter ist identisch mit der, die zum Durchführen von SDO-Transfers durch die Requests READ_VAR und WRITE_VAR (siehe <i>Syntax der Funktion WRITE_VAR, S. 57</i> ) verwendet wird. Die Schaltfläche <b>Request eingeben</b> öffnet die Eingabefenster für den Request. Die LED Fehler zeigt einen Nachrichtenaustauschfehler an.

---

## 3.5 Diagnose

---

### Auf einen Blick

**Inhalt des Abschnitts**

Dieser Abschnitt beschreibt die für die PCMCIA-Karte TSX CPP 1•• erhältlichen Hardware- und Software-Diagnosetools.

---

**Inhalt dieses Abschnitts**

Dieser Abschnitt enthält die folgenden Themen:

Thema	Seite
Diagnose mithilfe der Status-LEDs der PCMCIA-Karte TSX CPP 100/110	73
Die Diagnosedaten	75
So führen Sie eine Diagnose durch	78

---

## Diagnose mithilfe der Status-LEDs der PCMCIA-Karte TSX CPP 100/110

---

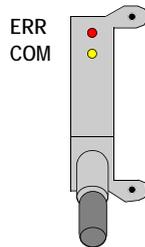
### Auf einen Blick

Der Status der Karte und des CANopen-Busses lässt sich mithilfe von Anzeige-LEDs auf der Karte anzeigen. Im Normalbetrieb ist die LED "ERR" aus. Die LED "COM" leuchtet permanent.

---

### Abbildung

Die folgende Abbildung zeigt die Lage der LEDs "ERR" und "COM".



## Diagnose

Die Diagnose lautet in Abhängigkeit vom Zustand der LEDs:

Status-LEDs		Bedeutung
ERR (rot)	COM (gelb)	
Aus	Aus	Die Karte wird nicht mit Spannung versorgt oder die Konfiguration wird übertragen.
	Blinkend (unregelmäßig)	Keine Konfiguration in der Karte.
	Blinkend (regelmäßig)	Die Karte ist konfiguriert und bereit, der Bus ist nicht aktiviert oder es ist keine CANopen-Firmware vorhanden.
	Ein	Der Bus ist konfiguriert und aktiv, es ist kein Fehler vorhanden.
Permanent leuchtend	Aus	Es wurde ein Fehler erkannt, der Bus-Controller wurde angehalten.
	Blinkend	Die Karte ist konfiguriert und bereit, aber es ist keine Kommunikation mit einem dezentralen Peripheriegerät möglich (z.B. CAN-Buskabel nicht angeschlossen) oder alle konfigurierten Peripheriegeräte melden einen Fehler. Kartenfehler, Konfigurationsfehler oder Synchronisationsfehler zwischen Karte und Steuerung (weitere Informationen finden Sie in den Diagnosedaten des Modulstatus im Handbuch TSXDMCPP100).
	Ein	Der Bus ist konfiguriert und aktiv, mindestens ein Busteilnehmer kann nicht erreicht werden oder meldet einen Fehler.

---

---

## Die Diagnosedaten

---

**Auf einen Blick** Bei einer Diagnose sind die ersten zu verwendenden Daten die Wörter %IW für impliziten Austausch (siehe *Sprachobjekte mit implizitem Austausch*, S. 83).

Es gibt aber auch Diagnosedaten, die Sie ansehen können, indem Sie einige Programmzeilen in die Steuerung schreiben.

Die Kommunikationsfunktion SEND\_REQ ermöglicht das Durchführen dieser Diagnose (siehe *Diagnosebefehl*, S. 67):

- auf dem Slave Ihrer Wahl (1 bis 127, ein Request pro Slave),
- auf der Version der PCMCIA-Karte (128),
- auf den Kartenstatus (129),

und damit können Sie auch die Historie der Fehlermeldungen des Nachrichtendienstes erhalten (130).

**Hinweis:** mit dem Code 129 erhalten Sie dieselben Informationen, die enthalten sind in den Eingangswörtern des impliziten Austauschs (siehe *Eingangswortobjekte*, S. 84).

Die gelieferten Informationen stammen von der PCMCIA-Karte und werden regelmäßig aktualisiert.

Die Empfangstabelle des Request enthält die Informationen, die in den folgenden Abschnitten beschrieben werden.

**Hinweis:** die Informationen werden in Form von Bytetabellen gegeben. Angesichts der Möglichkeit, die gesamte oder einen Teil dieser Tabelle anzufordern, müssen Sie auf die MSB und LSB der Wörter der Tabelle %MW:L. achten.

**Hinweis:** die Diagnoseinformationen berücksichtigen die CAN-Norm, Sie finden hierzu Informationen auf der Seite: <http://www.can-cia.de>.

---

**Diagnose eines Slaves**

Die nachfolgende Tabelle beschreibt die Informationen, die Sie auf einen Diagnose-Request für einen Slave erhalten (Code 1 bis 127).

Rang des Byte	Beschreibung
0	Bits des Gerätestatus: <ul style="list-style-type: none"> <li>● x0: Keine Antwort!</li> <li>● x1: Überlauf der Tabelle mit der Historie der Fehlermeldungen</li> <li>● x2: Parameterfehler</li> <li>● x3: Überwachung des aktiven Geräts</li> <li>● x4 bis x6: reserviert</li> <li>● x7: Gesperrt</li> </ul>
1 und 2:	Zusätzliche Informationen, die beim Busstart auf das Standardobjekt 16#1000 (CAN-Norm) gelesen werden.
3 und 4:	Profilnummer, die beim Busstart auf dem Standardobjekt 16#1000 (CAN-Norm) gelesen wird.
5	Status des Slave: <ul style="list-style-type: none"> <li>● 1: Getrennt</li> <li>● 2: Verbindung läuft</li> <li>● 3: In Vorbereitung</li> <li>● 4: Bereit</li> <li>● 5: In Betrieb</li> <li>● 127: in Vorbereitung oder nicht vorhanden</li> </ul>
6	Fehlercode (siehe <i>Slave-Diagnosecodes</i> , S. 96)(Code des letzten vom Slave erstellten Fehlers)
7	Anzahl dringender Informationsbausteine über Slave (0 bis 5) <b>Bemerkung:</b> diese Bausteine werden nach dieser Tabelle angefügt, ein typischer Baustein wird im folgenden Abschnitt beschrieben

**Beschreibung eines Informationsbausteins über den Slave**

Die folgende Tabelle beschreibt einen typischen Informationsbaustein.

Rang des Byte	Beschreibung
0 und 1:	Fehlercode
2	Wert des Fehlerregisters, Objekt 16#1001 des Slaves (CAN-Norm)
3 bis 6	Wert des spezifischen Statusregisters des Herstellers, Objekt 16#1002 (CAN-Norm)
7	reserviert

### Diagnose auf eine Kartenversion

Die nachfolgende Tabelle beschreibt die Informationen, die Sie auf einen Diagnose-Request für eine PCMCIA-Karte (Code 128) erhalten.

Rang des Byte	Beschreibung
0 bis 7	Version der Firmware (Zeichenkette)
8 bis 10	Erstellungsdatum der Firmware-Version (BCD-codiert, TT.MM.JJ)
11 bis 13	Herstellungsdatum (BCD-codiert, TT.MM.JJ)
14 bis 17	Seriennummer (BCD-codiert)
18 bis 25	Protokollbezeichnung (ASCII, ohne Kettenendzeichen, Beispiel: "CANopen")

### Historie der Fehlermeldungen

Die nachfolgende Tabelle beschreibt die Informationen, die Sie auf einen Diagnose-Request für eine Historie (Code 130) erhalten.

Rang des Byte	Beschreibung
0 und 1:	Anzahl der Fehlerbausteine <b>Bemerkung:</b> diese Bausteine werden nach dieser Tabelle angefügt, ein typischer Baustein wird im folgenden Abschnitt beschrieben
2 bis 49	Inhalt der Fehlerbausteine (maximal acht Bausteine)

### Beschreibung eines Informationsbausteins der Historie

Die folgende Tabelle beschreibt einen typischen Informationsbaustein.

Rang des Byte	Beschreibung
0	Code des Dienstes
1	betroffene ID
2 und 3:	Fehlercode des Nachrichtendienstes (siehe <i>Fehlercodes</i> , S. 92)
4 und 5:	Einzelheit des Fehlercodes (siehe <i>Bisher aufgetretene Codes</i> , S. 96)

## So führen Sie eine Diagnose durch

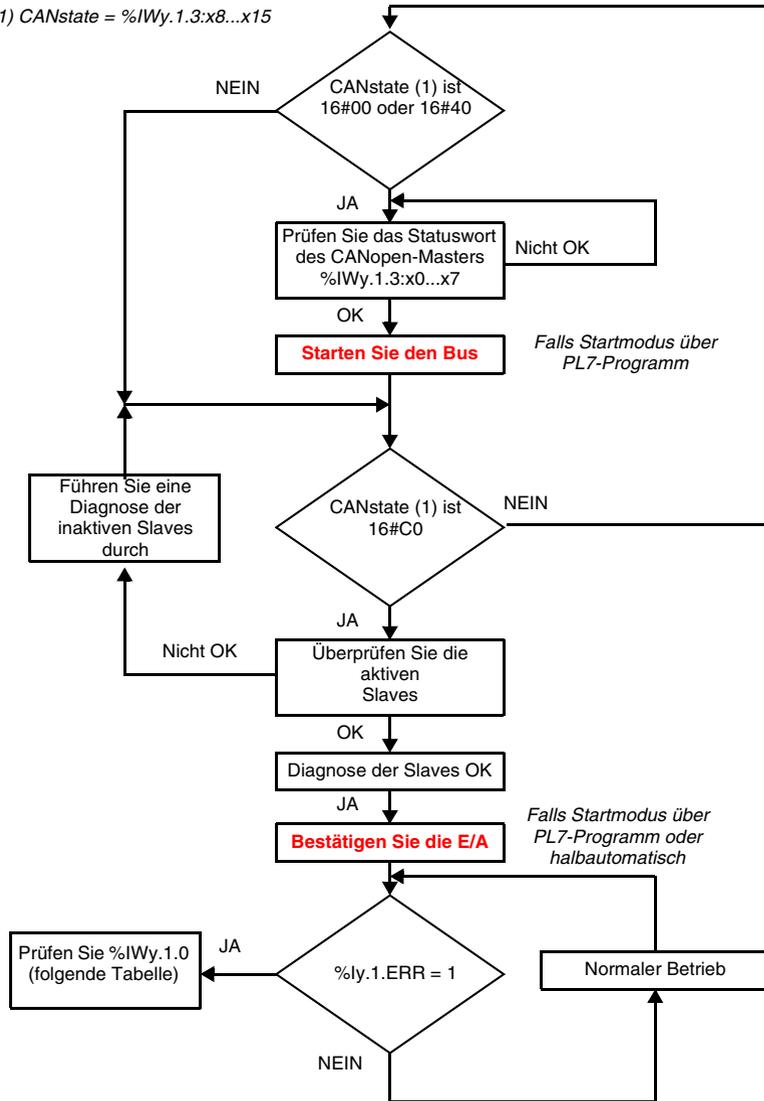
---

**Auf einen Blick** Beachten Sie bei Fehlersuche am CANopen-Bus zuerst die Anzeige-LEDs der PCMCIA-Karte. Danach können Sie folgende Vorgehensweise abarbeiten, die die Verwaltung des Busstarts und die Überprüfungen beschreibt, die sich unter Verwendung der Sprachobjekte (siehe *Sprachobjekte in Verbindung mit der Karte TSX CPP 100/110, S. 82*) der Steuerung durchführen lassen.

---

**Vorgehensweise** Der folgende Plan zeigt die einzelnen Phasen des zu befolgenden Ablaufs.

(1) CANstate = %IWy.1.3:x8...x15



**So überprüfen  
Sie %IWy.1.0**

Diese Tabelle beschreibt die Aktionen, mit denen Sie eine genaue Diagnose mithilfe der Bits x8 bis x15 von %IWy.1.0 erhalten.

Wenn ...	Dann ...
x8 = 1	<p>Konfigurationsfehler</p> <p>Prüfen Sie die Einzelheiten der Fehlercodes (siehe <i>Eingangswortobjekte, S. 84</i>) in den Worten:</p> <ul style="list-style-type: none"> <li>● %IWy.1.1</li> <li>● %IWy.1.2</li> </ul>
x9 = 1	<p>PDO-Übertragungsfehler</p> <p>Setzen Sie sich mit dem technischen Support von Schneider in Verbindung.</p>
x10 = 1	<p>SDO-Übertragungsfehler</p> <p>Prüfen Sie die Einzelheiten der Fehlercodes (siehe <i>Eingangswortobjekte, S. 84</i>) in den Worten:</p> <ul style="list-style-type: none"> <li>● %IWy.1.1</li> <li>● %IWy.1.2</li> </ul> <p>Prüfen Sie die Historie der Fehlermeldungen (siehe <i>Historie der Fehlermeldungen, S. 77</i>).</p>
x11 = 1	<p>Fehler der PCMCIA-Karte</p> <ul style="list-style-type: none"> <li>● Prüfen Sie die Einzelheiten der Fehlercodes in %IWy.1.1.</li> <li>● Prüfen Sie den Inhalt von %IWy.1.3: <ul style="list-style-type: none"> <li>● x0: Parameterfehler, die Fehlerquelle ist in %IWy.1.4 angegeben.</li> <li>● x1: Die Ausgänge sind nach dem Ausfall eines Slave (Autoclear EIN) auf Null gesetzt, die Fehlerquelle ist in %IWy.1.4 angegeben.</li> <li>● x3: Schwerwiegender Fehler, die Karte ist nicht am Bus aktiv.</li> <li>● x7: Fehlerhafte Verbindung zwischen Karte und Anschlussgehäuse TSX CPP ACC1</li> </ul> </li> </ul>
x12 = 1:	<p>Busfehler (Bus nicht gestartet oder erkannter Übertragungsfehler)</p> <ul style="list-style-type: none"> <li>● Prüfen Sie den Busfehler-Zähler %IWy.1.5. Ist er nicht gleich 0, prüfen Sie die Leitung.</li> <li>● Prüfen Sie den Bushalt-Zähler. Zählt er hoch, prüfen Sie die Leitung und starten Sie den Bus neu.</li> </ul> <p><b>Hinweis:</b> Im nichtautomatischen Startmodus ist das Startbit des Busses %QWy.1:x0.</p> <p>Beim TSX CPP 110 kann dieser Fehler auch nach einer Busverbindung auf TAP-Ebene oder nach gleichzeitigem Ausschalten oder Einschalten aller Slaves am Bus auftreten. Der Fehler kann quittiert werden, indem man den das Bit %QWy.1:X2 von 0 auf 1 gehen lässt.</p>

Wenn ...	Dann ...
x13 = 1	<p>Fehler eines Slaves: Kommunikationsfehler oder E/A nicht gültig.</p> <ul style="list-style-type: none"><li>● Bestimmen Sie die letzte Fehlerquelle in %IWy.1.4.</li><li>● Bestimmen Sie alle am Bus aktiven Slaves und sehen Sie in den Busstatus-Worten %IWy.1.16 bis %IWy.1.23 nach.</li><li>● Führen Sie mithilfe eines Diagnose-Requests (siehe <i>Diagnosebefehl</i>, S. 67) eine Diagnose der fehlerhaften Slaves durch.</li></ul> <p><b>Hinweis:</b> Im nichtautomatischen Startmodus ist das Startbit der E/A %QWy.1:x1.</p>
x14 = 1	<p>Ausgangsfehler: Die Ausgänge wurden in den Fehlerzustand gesetzt.</p> <ul style="list-style-type: none"><li>● Prüfen Sie, ob die Steuerung in RUN ist.</li><li>● Prüfen Sie, ob die dem Modul zugeordnete Task aktiv ist.</li><li>● Prüfen Sie das Bit %IWy.1.0:x12 (Busfehler) und das Bit %IWy.1.0:x13 (Slave-Fehler).</li></ul> <p><b>Hinweis:</b> Im nichtautomatischen Startmodus prüfen Sie die Bits %QWy.1:x0 und %QWy.1:x1.</p>
x15 = 1	<p>Es gibt eine neue Diagnose für einen oder mehrere Slaves.</p> <ul style="list-style-type: none"><li>● Bestimmen Sie anhand der Statusworte %IWy.1.16 bis %IWy.1.23 die betreffenden Slaves.</li><li>● Führen Sie eine Diagnose (siehe <i>Diagnosebefehl</i>, S. 67) des oder der betreffenden Slaves durch.</li></ul>

## 3.6 Sprachobjekte in Verbindung mit der Karte TSX CPP 100/110

---

### Auf einen Blick

---

**Inhalt des  
Abschnitts**

Dieser Abschnitt beschreibt die impliziten und expliziten Sprachobjekte in Verbindung mit der Karte TSX CPP 1\*\*.

---

**Inhalt dieses  
Abschnitts**

Dieser Abschnitt enthält die folgenden Themen:

Thema	Seite
Sprachobjekte mit implizitem Austausch	83
Sprachobjekte mit explizitem Austausch	88
Verwaltung des expliziten Austauschs	89
Sprachobjekte, der Konfiguration zugeordnet	90
Fehlercodes	92

---

## Sprachobjekte mit implizitem Austausch

### Auf einen Blick

Auf dieser Seite werden alle Sprachobjekte für den impliziten Austausch (Siehe: PL7 Micro/Junior/Pro; Applikationsspezifische Kommunikation; Teil 1) für eine CANopen-Kommunikation mit der PCMCIA-Karte TSX CPP 1•• beschrieben, die durch das Anwendungsprogramm angezeigt oder geändert werden können.

### Bitobjekte

In der nachfolgenden Tabelle sind die verschiedenen Bitobjekte des impliziten Austauschs aufgeführt.

Objekt (1)	Funktion	Bedeutung
%ly.MOD.ERR	Modulfehler-Bit	Wenn dieses Bit auf 1 gesetzt ist, zeigt es einen Modulfehler an (mindestens einer der Kanäle unterliegt einer Störung).
%ly.1.ERR	Kanalfehler-Bit	Ist dieses Bit 1, liegt ein Fehler der PCMCIA-Karte vor.
Legende		
(1)	Adresse y <ul style="list-style-type: none"> <li>• y: Steckplatznummer des Prozessors im Rack (0 oder 1, je nach verwendeter Spannungsversorgung)</li> </ul>	

**Eingangswort-  
objekte**

In der folgenden Tabelle sind die verschiedenen Eingangs-Wortobjekte des impliziten Austauschs aufgeführt.

Objekt (1)	Funktion	Bedeutung
%IWy.1.0	Statuswort des Kommunikationskanals	Statusbits des Kommunikationskanals: <ul style="list-style-type: none"> <li>● x0 = 1: Kanalfehler, logisches <b>ODER</b> zwischen den folgenden Bits X1 und X14</li> <li>● x1 bis X7: reserviert</li> <li>● x8 = 1: fehlerhafte Konfiguration</li> <li>● x9 = 1: E/A-Austauschfehler (PDO)</li> <li>● x10 = 1: Nachrichten-Austauschfehler (SDO)</li> <li>● x11 = 1: Kartenfehler (keine Karte oder Karte nicht bereit)</li> <li>● x12 = 1: Busfehler (Mindestens ein Ereignis des Typs <b>Busfehler</b> wurde erzeugt.)</li> <li>● x13 = 1: Slave-Fehler (Bei einem oder mehreren Slaves wurden Kommunikationsfehler entdeckt.)</li> <li>● x14 = 1: Ausgangsfehler (Die Ausgänge wurden in den Fehlerzustand gesetzt.)</li> <li>● x15 = 1: Neue Slave-Diagnose liegt vor (Es gibt eine neue Diagnose für einen oder mehrere Slaves.)</li> </ul>
%IWy.1.1	Fehlerwort	Dieses Wort enthält einen Modulfehler-Code (siehe <i>Modul-Fehlercodes</i> , S. 92) (letzter Konfigurations- oder E/A-Fehler).
%IWy.1.2	Fehlerwort	Dieses Wort enthält einen detaillierten Fehlercode des Moduls (siehe <i>Detailcode 806</i> , S. 95) (letzter Konfigurations- oder E/A-Fehler).

Objekt (1)	Funktion	Bedeutung
%IWy.1.3	Statuswort des CANopen-Masters	<p>Statusbits des Masters der CANopen-Kommunikation :</p> <ul style="list-style-type: none"> <li>● x0 = 1: Parameterfehler</li> <li>● x1 = 1: Zeigt an, dass die Ausgänge nach dem Ausfall eines Slaves auf Null gesetzt sind, Autoclear EIN.</li> <li>● x2 = 1: Kein Austausch auf dem Bus (kein Slave kommuniziert)</li> <li>● x3 = 1: Schwerwiegender Fehler, die Karte ist nicht am Bus aktiv</li> <li>● x4 = 1: Eines oder mehrere <b>Busfehler</b>-Ereignisse wurden erkannt</li> <li>● x5 = 1: Der Prozessor hat den Zugriff auf die Karte noch nicht erlaubt.</li> <li>● x6 = 1: Timeout beim Senden von CAN-Meldungen</li> <li>● x7 = 1: Fehlerhafte Verbindung zwischen Karte und Anschlussgehäuse TSX CPP ACC1 (nur TSX CPP 100)</li> <li>● x8 bis x15: Diese Bits bilden ein Byte, dessen Wert die Betriebsart angibt: <ul style="list-style-type: none"> <li>● 16#00 : Offline-Betrieb</li> <li>● 16#40 : Bus auf STOPP</li> <li>● 16#80 : Die Eingänge sind im Sicherheitsmodus (auf Null)</li> <li>● 16#C0: Bus auf RUN</li> </ul> </li> </ul>
%IWy.1.4	Statuswort der Busgeräte	<p>Das niederwertigste Byte (Bits x0 bis x7) enthält die Adresse des Slaves, der den letzten Fehler verursacht hat.</p> <p>Das höchstwertige Byte (Bits x8 bis x15) enthält den letzten Fehlercode.</p>
%IWy.1.5	Statuswort Zähler	Zählt die Fehler am Bus (Software, Frame error)
%IWy.1.6	Statuswort Zähler	Zählt die Anzahl der Busstops (Hardware)
%IWy.1.7	Statuswort Zähler	Zählt, wie oft bei CAN-Meldungen Timeouts aufgetreten sind
%IWy.1.8 bis %IWy.1.15	Statusworte des Busses	Am Bus aktive Geräte, jedes auf 1 gesetzte Bit entspricht einem aktiven Slave am Bus (8 Worte mit 16 Bit wären 128 Bit, also Master und 127 Slaves).
%IWy.1.16 bis %IWy.1.23	Statusworte des Busses	Am Bus verfügbare Diagnose, jedes auf 1 gesetzte Bit entspricht einem Gerät, für das es eine neue Diagnose gibt (8 Worte mit 16 Bit wären 128 Bit, also Master und 127 Slaves).

Objekt (1)	Funktion	Bedeutung
Legende		
	Adresse y	<ul style="list-style-type: none"><li>• y: Steckplatznummer des Prozessors im Rack (0 oder 1, je nach verwendeter Spannungsversorgung)</li></ul>

---

**Ausgangs-  
Wortobjekte**

In der folgenden Tabelle sind die verschiedenen Ausgangs-Wortobjekte des impliziten Austauschs aufgeführt.

Objekt (1)	Funktion	Bedeutung
%QWy.1.0	Befehlswort für die Betriebsarten	<p>Steuerbits:</p> <ul style="list-style-type: none"> <li>● x0 = 1: Aktiviert die Buskonfiguration x0 = 0: Deaktiviert die Buskonfiguration Dieses Bit wird nur verwendet, wenn die Applikation den Busstart verwaltet.</li> <li>● x1 = 1: Aktiviert die Datenübertragung über den Bus x1 = 0: Deaktiviert die Datenübertragung über den Bus Dieses Bit wird bei halbautomatischen Start verwendet, oder wenn die Applikation den Busstart verwaltet.</li> <li>● x2 = 1: Initialisiert die Fehlerbits: <ul style="list-style-type: none"> <li>● E/A- Fehler</li> <li>● Nachrichtenbehandlungsfehler</li> <li>● Bisher aufgetretene Fehler</li> </ul> </li> <li>● x3 = 1: Initialisiert die PCMCIA-Karte; dieses Bit führt einen Kaltstart der Karte durch. Dieses Bit wird in den drei Startmodi verwendet.</li> <li>● x4 bis x15: Diese Bits sind reserviert und auf Null gesetzt.</li> </ul>
%QWy.1.1	reserviert	-
Legende		
	Adresse y	
	<ul style="list-style-type: none"> <li>● y: Steckplatznummer des Prozessors im Rack (0 oder 1, je nach verwendeter Spannungsversorgung)</li> </ul>	

**Hinweis:** diese Ausgangsobjekte für impliziten Austausch werden auch Befehlswoorte genannt. Sie werden von der Applikation gesetzt. Die Bits x2 und x3 des Worts %QWy.0 werden nicht automatisch auf Null zurückgesetzt.

## Sprachobjekte mit explizitem Austausch

### Auf einen Blick

Diese Seite beschreibt Sprachobjekte mit explizitem Austausch (Siehe: PL7 Micro/Junior/Pro; Applikationsspezifische Kommunikation; Teil 1) für die CANopen-Kommunikation mit den PCMCIA-Karten TSX CPP 1••. Im Allgemeinen können diese Objekte durch das Applikationsprogramm angezeigt und geändert werden.

**Hinweis:** die Worte %MWy.1.0 und %MWy.1.1 dienen der Verwaltung des expliziten Austauschs (siehe *Wortobjekte*, S. 89).

### Internes Wort

Die folgende Tabelle beschreibt das Wort %MWy.1.2, das vom Request READ\_STS %CHy.1 aktualisiert wird:

Objekt (1)	Funktion	Bedeutung
%MWy.1.2	Status der PCMCIA-Karte	<ul style="list-style-type: none"> <li>● x0 = 1: Bus im Fehlerzustand oder nicht initialisiert (im Startmodus, von der Applikation verwaltet)</li> <li>● x1 = 1: Slave-Fehler, einer oder mehrere Slaves sind im Fehlerzustand oder nicht in RUN.</li> <li>● x2 = 1: Das Anschlussgehäuse TSX CPP ACC1 ist im Fehlerzustand oder seine Verkabelung ist fehlerhaft (nur TSX CPP 100).</li> <li>● x3 = 1: Die PCMCIA-Karte ist:                         <ul style="list-style-type: none"> <li>● nicht im Steckplatz,</li> <li>● nicht bereit oder</li> <li>● es ist ein schwerwiegender Fehler aufgetreten.</li> </ul> </li> <li>● x4 = 1: Die PCMCIA-Karte ist:                         <ul style="list-style-type: none"> <li>● nicht bereit, weil sie gerade initialisiert wird,</li> <li>● im Fehlerzustand oder</li> <li>● es kann nicht auf sie zugegriffen werden.</li> </ul> </li> <li>● x5 = 1: Der Karten- oder Protokolltyp wird nicht erkannt.</li> <li>● x6 = 1: E/A-Austauschfehler</li> <li>● x7 = 1: Konfigurations- oder Parameterfehler</li> <li>● x8 bis x15: reserviert (Wert 0)</li> </ul>
Legende		
	Adresse y	<ul style="list-style-type: none"> <li>● y: Steckplatznummer des Prozessors im Rack (0 oder 1, je nach verwendeter Spannungsversorgung)</li> </ul>

---

## Verwaltung des expliziten Austauschs

---

**Auf einen Blick** Diese Seite beschreibt alle Sprachobjekte, die den expliziten Austausch verwalten (Siehe: PL7 Micro/Junior/Pro; Applikationsspezifische Kommunikation; Teil 1).

---

**Wortobjekte** In der folgenden Tabelle sind die verschiedenen Wortobjekte für die Verwaltung des expliziten Austauschs aufgeführt.

Objekt (1)	Funktion	Bedeutung
%MWy.1.0	Austausch läuft	● x0 = 1: Austausch läuft
%MWy.1.1	Austauschfehler	● x0 = 1: Fehler beim Austausch
Legende		
	Adresse y	● y: Steckplatznummer des Prozessors im Rack (0 oder 1, je nach verwendeter Spannungsversorgung)

---

## Sprachobjekte, der Konfiguration zugeordnet

### Auf einen Blick

Auf dieser Seite werden alle Konfigurationssprachobjekte für eine CANopen-Kommunikation mit der PCMCIA-Karte TSX CPP 1\*\* beschrieben, die durch das Applikationsprogramm angezeigt werden können.

### Interne Konstanten

In der folgenden Tabelle werden die internen Konstanten beschrieben.

Objekt	Funktion	Bedeutung
%KWy.1.0	Konstanter Wert, vom System verwendet	Niederwertigstes Byte: 16#00 Höchstwertiges Byte: 16#37
%KWy.1.1	Konfigurationsbits	<ul style="list-style-type: none"> <li>● Fehlermodi der Ausgänge beim Wechseln der Steuerung in STOP: x0 = 0: RESET x0 = 1: Wert halten</li> <li>● x1 = 0: Laden der Konfiguration durch das Programmiergerät x1 = 1: Verwenden der Konfiguration im Flash-EEPROM</li> <li>● Bussteuerung beim Start: x2 = 0: Automatisch x2 = 1: Durch die Applikation</li> <li>● E/A-Steuerung beim Start: x3 = 0: Automatisch x3 = 1: Durch die Applikation</li> <li>● Synchronisation des Datenaustauschs x4 = 0: MAST-Task x4 = 1: FAST-Task</li> <li>● x5 reserviert</li> <li>● Watchdog der CANopen-PCMCIA-Karte x6 = 0: Aktiviert x6 = 1: Gesperrt</li> <li>● x7 bis X15: reserviert</li> </ul>
%KWy.1.2	Konfigurationsbits	Speichergröße der Buskonfiguration (in Byte)
%KWy.1.3	Konfigurationsbits	Speichergröße des <b>Eingangsabbilds</b> (in Worten)
%KWy.1.4	Konfigurationsbits	Speichergröße des <b>Ausgangsabbilds</b> (in Worten)
%KWy.1.5	Konfigurationsbits	Anfangsadresse des <b>Eingangsabbilds</b> (%MW)
%KWy.1.6	Konfigurationsbits	Anfangsadresse des <b>Ausgangsabbilds</b> (%MW)
Legende		

---

<b>Objekt</b>	<b>Funktion</b>	<b>Bedeutung</b>
(1)	Adresse y ● y: Steckplatznummer des Prozessors im Rack (0 oder 1, je nach verwendeter Spannungsversorgung)	

---

## Fehlercodes

---

### Auf einen Blick

Die folgenden Tabellen beschreiben die einzelnen Fehlercodes, die in einer CANopen-Konfiguration auftreten können.

Mit den in den jeweiligen Tabellen gegebenen Erklärungen lassen sich Applikationen so programmieren, dass mögliche Fehlfunktionen einfacher erkannt und beseitigt werden können.

### Modul-Fehlercodes

Die folgende Tabelle beschreibt die in den Worten %IWy.1.1 (Beschreibungsspalte) und %IWy.1.2 (Detailspalte) abgelegten Fehlercodes.

Code	Beschreibung	Details
0	Kein Fehler, normale Funktion	
	<b>Standardfehler</b>	
100	Ungültige Adresse (Nullpointer)	-
101	Ungültiger Wert	Wert
102	Ungültiges ID-Objekt	ID
103	Ungültiger Treiberstatus	Status-Code
104	Ungültige Größe des Speicherbereichs beim Lesen	Speichergröße
105	Ungültiger Datenumfang beim Schreiben	Datenumfang
106	Timeout	Timeout beim Übertragen eines SDO oder Ladestatus beim Herunterladen einer Konfiguration
107	Synchronisationsfehler	-
108	Anhalten durch den Benutzer	-
	<b>Ressourcenfehler</b>	
200	Speicherüberlauf	-
201	Speicher voll	-
	<b>Nachrichtenbehandlungsfehler</b>	
300	Ungültige Quelladresse	Wert der Adresse
301	Ungültige Zieladresse	Wert der Adresse
302	Ungültiger Dienst	Dienstcode
303	Ungültige Dienstklasse für ein Segment-ID	Angabe des Werts
304	Ungültige Grundfunktion des Dienstes	Funktionscode

Code	Beschreibung	Details
305	Ungültiges ID des aufgerufenen Dienstes	Angabe des Werts
306	Ungültiger Kommunikationsport	Portnummer
307	Ungültige Busgeräte-ID	Wert der ID
308	Ungültiger SDO-Index	Wert des Index
309	Ungültiger SDO-Unterindex	Wert des Unterindex
310	Dezentraler Fehler bei der Ausführung eines Dienstes	Fehlercode
311	Ungültiges COB-ID	Wert des COB-ID
312	Ungültige Übertragungsart auf dem Link Layer	Code der angeforderten Übertragung: <ul style="list-style-type: none"> <li>● 101 : Senden</li> <li>● 102 : Empfangen</li> <li>● 103 : Senden und empfangen</li> </ul>
<b>PCMCIA-Kartenfehler</b>		
600	Keine Karte	-
601	Andere Karte erkannt als TSX CPP 100 / TSX CPP 110	-
602	Karte nicht kommunikationsbereit	-
603	Karte noch nicht auf RUN	-
<b>Kommunikationsfehler der PCMCIA-Karte</b>		
700	Fehler beim Senden einer Nachricht an die Karte	-
701	Fehler beim Empfang einer Nachricht von der Karte	-
702	Fehler beim Senden eines Ausgangs-PDO an die Karte	-
703	Fehler beim Empfang eines Eingangs-PDO von der Karte	-
<b>Konfigurationsfehler</b>		
800	Falscher Umfang der Bus-Konfigurationsdaten	Umfang der Konfigurationsdaten
801	Ungültiger Datenumfang des Eingangsabbilds	Umfang, in Worten, beim Start der Karte festgelegt
802	Ungültiger Datenumfang des Ausgangsabbilds	Umfang, in Worten, beim Start der Karte festgelegt

Code	Beschreibung	Details
803	Überschneidung der für Eingänge und Ausgänge reservierten Speicherbereiche	Art der Überschneidung: <ul style="list-style-type: none"> <li>● 1 : Der Anfang des Eingangsbereichs überschneidet sich mit dem Ende des Ausgangsbereichs</li> <li>● 2 : Der Anfang des Ausgangsbereichs überschneidet sich mit dem Ende des Eingangsbereichs</li> </ul>
804	Daten-Ladebereich nicht gefunden	Art der des Bereichs: <ul style="list-style-type: none"> <li>● 1 : Globale Daten</li> <li>● 2 : Busparameter</li> <li>● 3 : Synchronisationsmodus</li> </ul>
805	Ungültige Prüfsumme der Konfigurationsdaten (Inkonsistenz zwischen Bus-Konfigurationsdaten im Sycon-Modus)	Prüfsumme der Bus-Konfigurationsdaten im Flash-Speicher der Karte
806	Negative Rückmeldung beim Herunterladen der Konfiguration	Höchstwertiges Byte: Fehlercode der Karte (siehe <i>Detailcode 806, S. 95</i> ) Niederstwertiges Byte: <ul style="list-style-type: none"> <li>● 16#00 : kein laufender Ladevorgang</li> <li>● 16#01 : Lade-Anforderung</li> <li>● 16#02 : Ladevorgang läuft</li> <li>● 16#03 : Ladevorgang beendet</li> <li>● 16#11 : Übertragung zum PC angefordert</li> <li>● 16#12 : Übertragung zum PC läuft</li> </ul>

---

**Detailcode 806**

Die folgende Tabelle beschreibt die im höchstwertigen Byte des Worts %IWy.1.2 (und für den Wert 806 des Worts %IWy.1.1) abgelegten Fehlercodes.

Code	Beschreibung
48	Timeout
52	Unbekannter Bereichscode
53	Überschreiten der maximalen Speichergröße
55	Falscher Parameter
57	Ablauffehler beim Herunterladen
59	Unvollständig heruntergeladene Daten
60	Doppelt eingetragene Adresse
61	PDO-Adresstabelle zu groß
62	Busgeräte-Parameterbereich zu groß
63	Unbekanntes PDO-Übertragungsverfahren
64	PDO-Daten zu umfangreich
65	Unbekannte Übertragungsgeschwindigkeit
66	Grenzwertüberschreitung, COB-ID des Drehmelders
67	Grenzwertüberschreitung, Timer-Preset des Drehmelders
68	Eingangs-Datenumfang + Offset, größer als Maximalgröße des Bereichs für Eingangsabbild
69	Eingangs-Datenumfang + Offset, größer als Maximalgröße des Bereichs für Eingangsabbild
70	Inkonsistenz zwischen PDO-Konfiguration und PDO-Adresstabelle
71	Ungültige Länge der PDO-Adresstabellen
72	Ungültiger Datenumfang beim Herunterladen
73	Grenzwertüberschreitung, COB-ID der Eilmeldungen
74	Grenzwertüberschreitung bei der COB-ID der Busgeräte-Überwachungsmeldungen
75	Grenzwertüberschreitung, PDO-Längenanzeige
76	SDO-Daten zu umfangreich

**Bisher aufgetretene Codes**

Die folgende Tabelle beschreibt die im fünften und sechsten Byte der Tabelle mit der Diagnosehistorie abgelegten Fehlercodes.

Code	Beschreibung
3	Dienst vom Gerät zurückgewiesen
17	Keine Antwort vom Gerät
51	Länge des Empfangs-Speicherbereichs zu groß
53	Länge der fragmentierten Protokolldaten größer als Pufferspeicher
54	Unbekannte Funktion vom Kartentreiber angefordert
55	Grenzwertüberschreitung, Geräteadresse am Bus
57	Ablauffehler bei einer fragmentierten Übertragung, Vorgang abgebrochen
200	Karte nicht konfiguriert

---

**Slave-Diagnosecodes**

Die folgende Tabelle beschreibt die im siebten Byte der Slave-Diagnosetabelle abgelegten Fehlercodes.

Code	Beschreibung
30	Slave-Überwachungsfehler
31	Statuswechsel eines Busgeräts, Gerät nicht verfügbar
32	Ablauffehler des Bus-Überwachungsprotokolls
33	Keine Antwort für ein konfiguriertes PDO
34	Keine Antwort bei der Konfiguration des Geräts
35	Konfiguriertes Geräteprofil entspricht nicht dem Profil des am Bus vorhandenen Geräts
36	Konfigurierter Gerätetyp entspricht nicht dem Typ des am Bus vorhandenen Geräts
37	Unbekannte SDO-Antwort
38	Empfangener Rahmen größer als acht Byte
39	Gerät nicht gescannt oder angehalten (zum Beispiel Autoclear-Modus)

---

## Lade- Fehlercodes, Sycon

Die folgende Tabelle beschreibt die Fehlercodes, die beim Laden der Konfiguration oder Firmware der Karte mithilfe des XWAY-Treibers auftreten können.

Code	Beschreibung
0	Kein Fehler, normale Funktion
<b>Standardfehler</b>	
8001	Treiber nicht aktiv
8002	Treiber hat unbekanntes Ereigniscode geliefert
8003	Befehlscode vom Treiber nicht erkannt
8004	Befehl zurückgewiesen
8005	Ein anderer Befehl ist noch aktiv
8006	Befehl wurde an ein ungültiges Gerät geschickt
<b>Allokationsfehler</b>	
8010	Kein Gerät zugewiesen
8011	Gerät bereits zugewiesen
<b>Kommunikationsfehler</b>	
8020	Senden eines Dienst-Requests, obwohl kein Gerät angeschlossen ist
8021	Initialisieren einer Verbindung, obwohl bereits eine Verbindung besteht
8022	Timeout
8030	Fehler beim Lesen des Treiberstatus
8031	Fehler nach dem Senden eines Requests über das Netz
8032	Ausgangs-Postfach ist immer belegt
8033	Antwortfehler vom Netz
8034	Keine Antwort im Eingangs-Postfach
8035	Übertragungsfehler der Eingangs-/Ausgangsdaten
<b>Initialisierungsfehler des Treibers</b>	
8080	Parameterfehler
8081	Allgemeiner Initialisierungsfehler des Treibers
<b>Multitasking-Fehler</b>	
-1	Keine Arbeitstask erstellt
-2	Task-Zeiger oder Zeiger auf synchronisiertes Objekt ungültig
-3	Kein Synchronisationsereignis erstellt



---

# Konfigurationsbeispiel mit dem SyCon-Tool



# 4

---

## Auf einen Blick

### Inhalt des Kapitels

Dieses Kapitel beschreibt anhand von Beispielen die Softwaredurchführung der CANopen-Buskonfiguration mit Hilfe des SyCon-Tools (V2.8).

### Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Abschnitte:

Abschnitt	Thema	Seite
4.1	Standardkonfiguration	100
4.2	Multi-Master-Konfiguration	116

## 4.1 Standardkonfiguration

---

### Auf einen Blick

**Inhalt des Abschnitts**

Dieser Abschnitt beschreibt anhand von Beispielen die Softwarekonfiguration der CANopen Bus-Konfiguration, die durch einen einzigen Master verwaltet wird.

---

**Inhalt dieses Abschnitts**

Dieser Abschnitt enthält die folgenden Themen:

Thema	Seite
Beispielbeschreibung	101
Deklaration des CANopen Masters	102
Konfiguration des CANopen-Busses	104
Deklaration von Slave 7	105
Konfiguration des Slave 7	107
Deklaration von Slave 8	111
Konfiguration des Slave 8	112
Überprüfung der durchgeführten Konfiguration	113

---

## Beispielbeschreibung

### Auf einen Blick

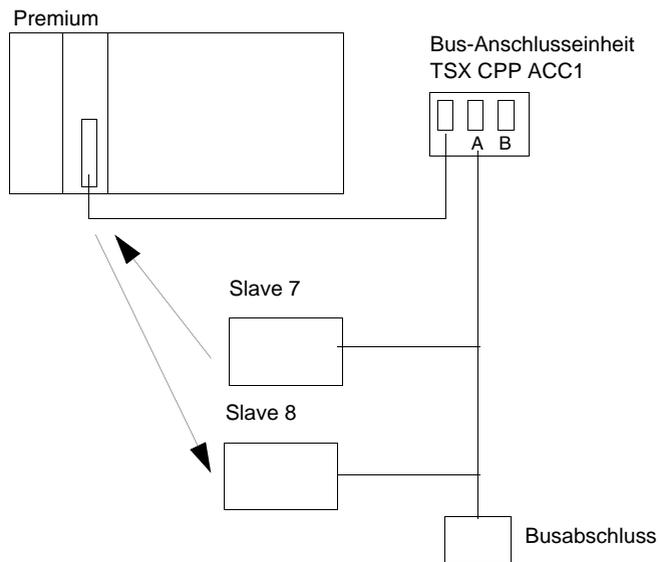
Dieses Beispiel wird aus didaktischen Gründen angegeben. Damit können Sie die verschiedenen Schritte der Konfiguration einer CANopen-Architektur verfolgen, die aus folgendem zusammengesetzt sind:

- ein TSX CPP 100 Mastermodul in einer Premium Steuerung installiert,
- eine Bus-Anschlusseinheit TSX CPP ACC1.
- zwei Slavegeräte:
  - ein Modul mit 8 Eingängen (Slave 7),
  - ein Modul mit 8 Ausgängen (Slave 8),

Die Übertragungsgeschwindigkeit über den Bus ist auf 1 Mbits/s festgesetzt.

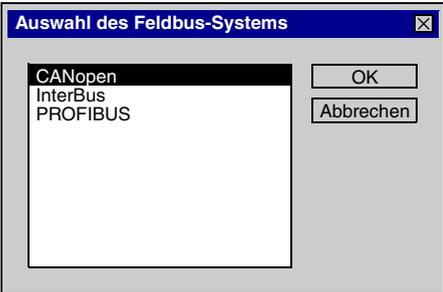
### Abbildung

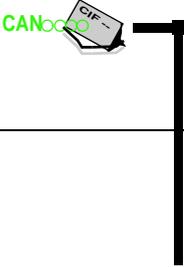
In folgender Darstellung wird die im Beispiel benutzte Struktur schematisiert.



## Deklaration des CANopen Masters

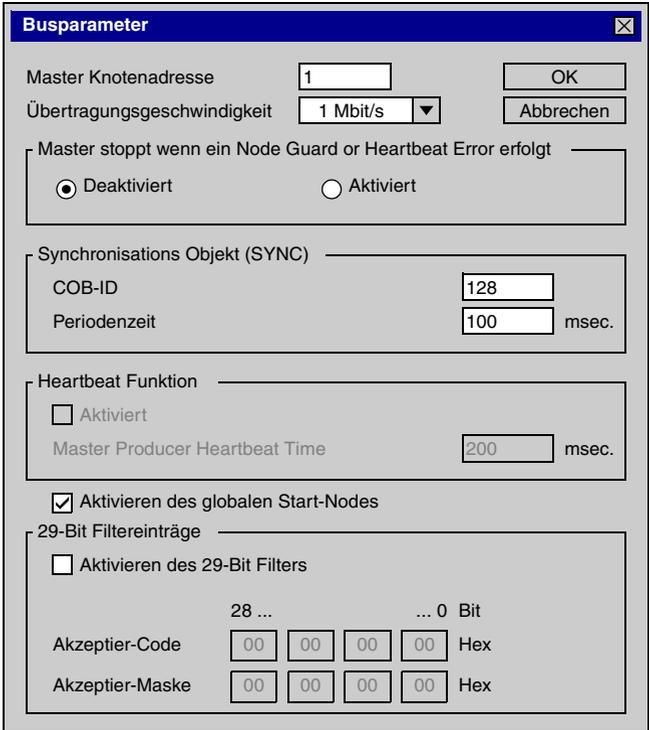
**Vorgehensweise** Folgende Tabelle stellt die verschiedenen Schritte zur Deklaration eines CANopen-Masters dar:

Schritt	Aktion
1	<p>Klicken Sie im Premium PL7 Konfigurationsfenster der TSX CPP100-Karte auf das Symbol  das Symbol <b>Mischer</b></p> <p><b>Ergebnis:</b> Das SyCon-Tool erscheint im Fenster.</p>
2	<p>Wählen Sie den Befehl <b>Datei</b> → <b>Neu</b></p> <p><b>Ergebnis:</b> Folgendes Fenster erscheint:</p> 
3	<p>Wählen Sie <b>CANopen</b> aus und bestätigen Sie mit <b>Ok</b>.</p> <p><b>Ergebnis:</b> Eine leere Struktur erscheint im Fenster.</p>
4	<p>Wählen Sie den Befehl <b>Einfügen</b> → <b>Master</b></p> <p><b>Ergebnis:</b> Folgendes Fenster erscheint:</p> 

Schritt	Aktion
5	<ul style="list-style-type: none"><li>● Wählen Sie <b>TSX CPP 100</b> aus,</li><li>● Klicken Sie <b>Hinzufügen</b>,</li><li>● Geben Sie ein Master-Modulname im Feld <b>Beschreibung</b> ein, <b>Anmerkung:</b> Der Name darf weder Leerzeichen noch hervorgehobene Zeichen beinhalten und ist auf maximal 32 Zeichen begrenzt.</li><li>● bestätigen Sie mit <b>Ok</b>.</li></ul> <p><b>Ergebnis:</b> Folgende Struktur erscheint:</p> <div data-bbox="477 418 1085 558" style="border: 1px solid black; padding: 5px;"><p style="text-align: right;"><b>CANopen_Master</b></p><p><i>Knotenadresse</i>    1 <i>Master</i>            TSX CPP 100</p></div>

## Konfiguration des CANopen-Busses

**Vorgehensweise** Folgende Tabelle stellt die verschiedenen Schritte zur Konfiguration eines CANopen-Busses dar:

Schritt	Aktion
1	<p>Wählen Sie einen Befehl aus <b>Einstellungen</b> → <b>Busparameter</b>.  <b>Ergebnis:</b> Folgendes Fenster erscheint:</p> 
2	<p>Konfigurieren Sie:</p> <ul style="list-style-type: none"> <li>• die Übertragungsgeschwindigkeit mit 1Mbit/s,</li> <li>• den Wert von <b>SYNC COB-ID</b> mit 128 (Standardwert),</li> <li>• die Periodenzeit mit 100 ms.</li> </ul>
3	<p>Wählen Sie <b>Deaktiviert</b> bei <b>Master stoppt wenn ein Node Guard or Heartbeat Error erfolgt</b>.</p>
4	<p>Wählen Sie <b>Aktivieren des globalen Start-Nodes</b>.</p>
5	<p>Bestätigen Sie mit <b>Ok</b>.</p>

## Deklaration von Slave 7

**Vorgehensweise** Folgende Tabelle stellt die verschiedenen Schritte zur Deklaration des Slave 7 dar.

Schritt	Aktion
1	Wählen Sie den Befehl <b>Einfügen</b> → <b>Knoten</b> <b>Ergebnis:</b> Ein Cursor erscheint :
2	Stellen Sie den Cursor auf den Bus außerhalb des Rahmens, der den Master begrenzt. <b>Ergebnis:</b> Folgendes Fenster erscheint:

**Knoten einfügen**
✕

Knotenfilter:

Hersteller  OK

Profil  Abbrechen

Verfügbare Geräte

- ATV58\_E
- ATV58\_F
- ATV58F\_E
- ATV58F\_F
- CBM-AI4
- CBM-AO4
- CBM-DIO8
- CIF104-COS

Gewählte Geräte

Hinzufügen

Alle Hinzuf.

<<

<< Alle Entf.

Knotenadresse

Beschreibung

Herstellername    ESD

Produktnummer   Kein Eintrag

Produktversion    1

Produktrevision   0

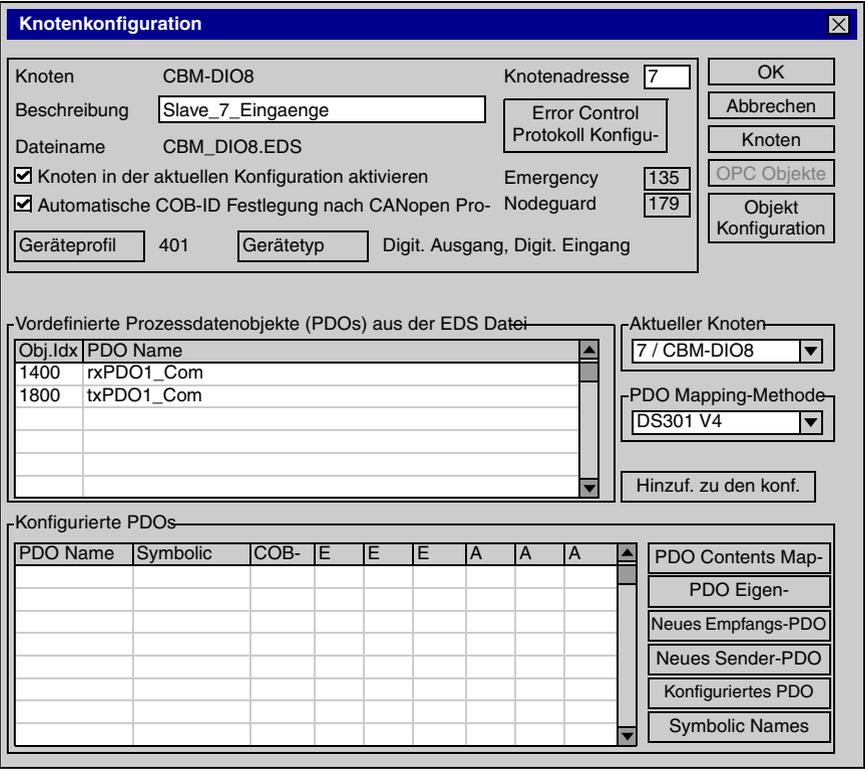
EDS Dateiname    CBM\_DIO8.EDS

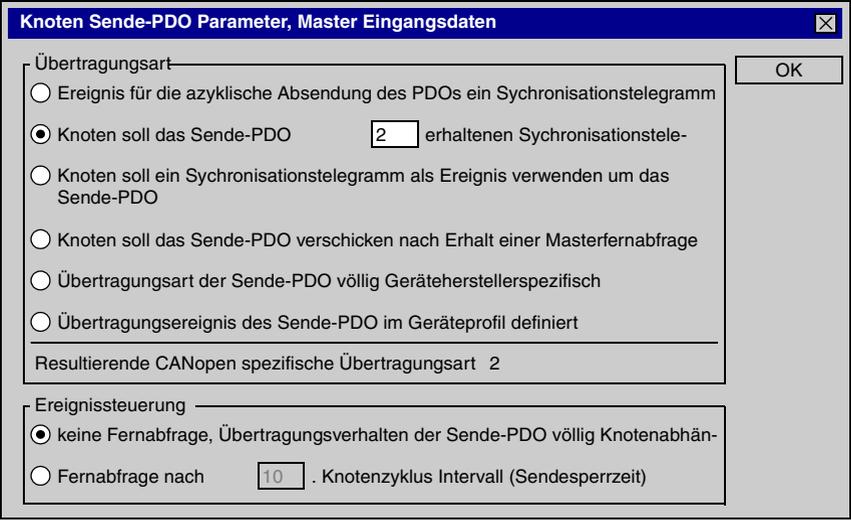
EDS Revision      1

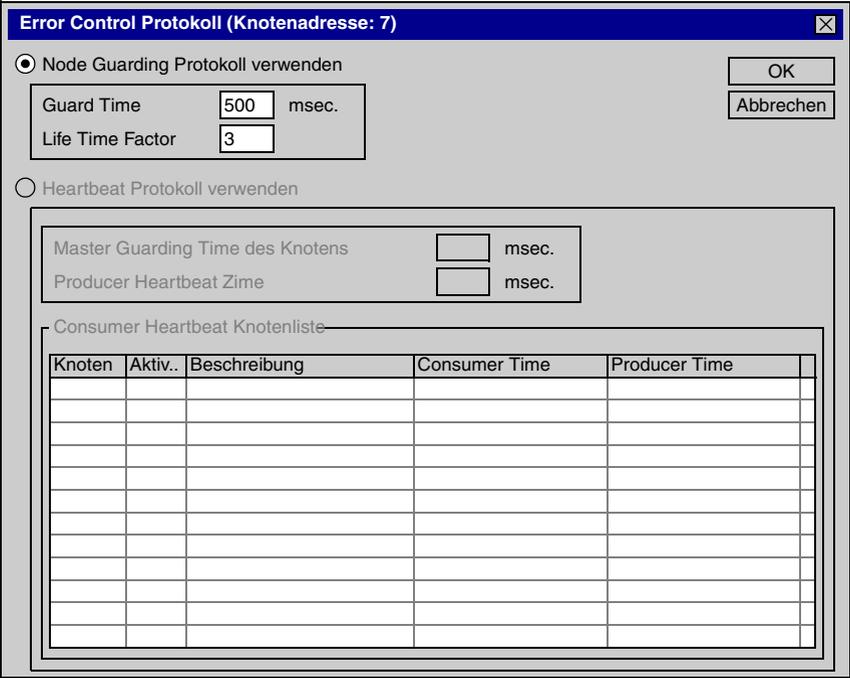
Schritt	Aktion
3	<p>Wählen Sie:</p> <ul style="list-style-type: none"> <li>● den Herstellername (z.B. ESD) :</li> <li>● das Slave-Profil (z.B. 401),</li> <li>● in der Slave-Liste den Slave-Typ (z.B. CBM-DIO8) und klicken Sie auf <b>Hinzufügen</b>,</li> <li>● 7 im Feld <b>Knotenadresse</b> und geben Sie den Slave-Modulnamen in das Feld <b>Beschreibung</b> ein,</li> <li>● bestätigen Sie mit <b>Ok</b>.</li> </ul> <p><b>Ergebnis:</b> Folgende Struktur erscheint:</p> <div style="display: flex; align-items: center;"> <div style="margin-right: 20px;">  </div> <div style="margin-right: 20px;">  </div> <div> <p><b>Master</b></p> <p><i>Knotenadresse</i>    1</p> <p><i>Master</i>            TSX CPP 100</p> </div> </div> <hr/> <div style="display: flex; align-items: center;"> <div style="margin-right: 20px;">  </div> <div> <p><b>Slave_7_Eingaenge</b></p> <p><i>Knotenadresse</i>    7</p> <p><i>Knoten</i>            CBM-DIO8</p> </div> </div>

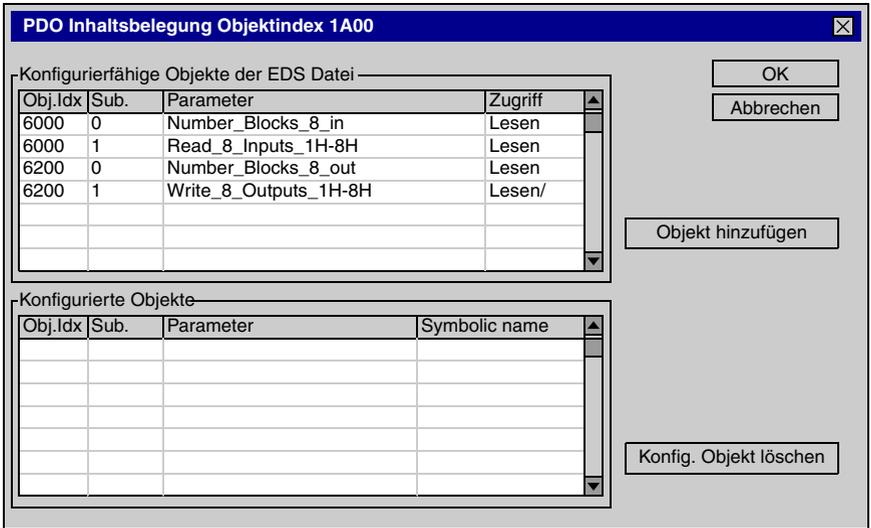
## Konfiguration des Slave 7

**Vorgehensweise** Folgende Tabelle stellt die verschiedenen Schritte zur Konfiguration des Slave 7 dar.

Schritt	Aktion																								
1	<p>Doppelklicken Sie auf Slave 7.  <b>Ergebnis:</b> Folgendes Fenster erscheint:</p>  <p>The screenshot shows the 'Knotenkonfiguration' dialog box with the following details:</p> <ul style="list-style-type: none"> <li><b>Knoten:</b> CBM-DIO8</li> <li><b>Knotenadresse:</b> 7</li> <li><b>Beschreibung:</b> Slave_7_Eingaenge</li> <li><b>Dateiname:</b> CBM_DIO8.EDS</li> <li><input checked="" type="checkbox"/> Knoten in der aktuellen Konfiguration aktivieren</li> <li><input checked="" type="checkbox"/> Automatische COB-ID Festlegung nach CANopen Pro-</li> <li><b>Emergency:</b> 135</li> <li><b>Nodeguard:</b> 179</li> <li><b>Geräteprofil:</b> 401</li> <li><b>Gerätetyp:</b> Digit. Ausgang, Digit. Eingang</li> <li><b>Vordefinierte Prozessdatenobjekte (PDOs) aus der EDS Datei:</b> <table border="1"> <thead> <tr> <th>Obj.Idx</th> <th>PDO Name</th> </tr> </thead> <tbody> <tr> <td>1400</td> <td>rxPDO1_Com</td> </tr> <tr> <td>1800</td> <td>txPDO1_Com</td> </tr> </tbody> </table> </li> <li><b>Aktueller Knoten:</b> 7 / CBM-DIO8</li> <li><b>PDO Mapping-Methode:</b> DS301 V4</li> <li><b>Konfigurierte PDOs:</b> <table border="1"> <thead> <tr> <th>PDO Name</th> <th>Symbolic</th> <th>COB-</th> <th>E</th> <th>E</th> <th>E</th> <th>A</th> <th>A</th> <th>A</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> </tbody> </table> </li> </ul>	Obj.Idx	PDO Name	1400	rxPDO1_Com	1800	txPDO1_Com	PDO Name	Symbolic	COB-	E	E	E	A	A	A									
Obj.Idx	PDO Name																								
1400	rxPDO1_Com																								
1800	txPDO1_Com																								
PDO Name	Symbolic	COB-	E	E	E	A	A	A																	

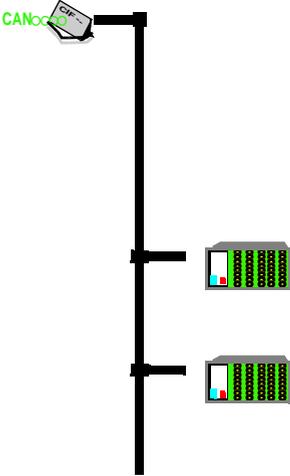
Schritt	Aktion
2	<p>Wählen Sie das Objekt <b>1800 txPDO1_Com</b> aus und klicken Sie auf die Schaltfläche <b>Hinzuf. zu den konf. PDOs</b>.</p> <p><b>Ergebnis:</b> Folgendes Fenster erscheint:</p> 
3	Verändern Sie eventuell die Frequenz beim Senden und bestätigen Sie mit <b>OK</b> .

Schritt	Aktion
4	<p>Klicken Sie auf die Schaltfläche <b>Error Control Protokoll Konfiguration</b>.  <b>Ergebnis:</b> Folgendes Fenster erscheint:</p>  <p>Ändern Sie <b>Life Time Factor</b> auf <b>3</b> und bestätigen Sie mit <b>OK</b>.</p>
<b>Hinweis</b>	<p>Falls einer der Parameter "Guard Time" oder "Life Time Factor" im Konfigurationsfenster des Slave auf den Wert "0" gesetzt wird, gehen die Ausgänge bei Unterbrechung der Verbindung (*) zwischen diesem Slave und der Karte TSX CPP 100 / TSX CPP 110 nicht auf ihren Fehlerwert.</p> <p>(*) Diese Trennung der Verbindung könnte verursacht worden sein durch:</p> <ul style="list-style-type: none"> <li>● Prozessorfehler oder Ausschalten seiner Versorgungsspannung</li> <li>● Trennen das Kabels zwischen PCMCIA-Karte und TAP</li> <li>● Trennen des Slave vom Bus</li> <li>● Defektes Buskabel</li> <li>● "Reset"-Befehl durch Sycon (Menü Online → Firmware / Reset),</li> <li>● Konfigurations-Ladebefehl durch Sycon (Menü Online → Download),</li> <li>● Firmware-Ladebefehl der Karte durch Sycon (Menü Online →Firmware Download),</li> </ul>

Schritt	Aktion																				
5	<p>Wählen Sie <b>PDO_1800</b> im Feld <b>Konfigurierte Datenprozessobjekte (PDO)</b> und klicken Sie auf die Schaltfläche <b>PDO Contents Mapping</b>.  <b>Ergebnis:</b> Folgendes Fenster erscheint:</p>  <p>The screenshot shows a dialog box titled "PDO Inhaltsbelegung Objektindex 1A00". It has two main sections:</p> <ul style="list-style-type: none"> <li><b>Konfigurierfähige Objekte der EDS Datei:</b> A table with columns: Obj.Idx, Sub., Parameter, Zugriff. <table border="1"> <thead> <tr> <th>Obj.Idx</th> <th>Sub.</th> <th>Parameter</th> <th>Zugriff</th> </tr> </thead> <tbody> <tr> <td>6000</td> <td>0</td> <td>Number_Blocks_8_in</td> <td>Lesen</td> </tr> <tr> <td>6000</td> <td>1</td> <td>Read_8_Inputs_1H-8H</td> <td>Lesen</td> </tr> <tr> <td>6200</td> <td>0</td> <td>Number_Blocks_8_out</td> <td>Lesen</td> </tr> <tr> <td>6200</td> <td>1</td> <td>Write_8_Outputs_1H-8H</td> <td>Lesen/</td> </tr> </tbody> </table> </li> <li><b>Konfigurierte Objekte:</b> A table with columns: Obj.Idx, Sub., Parameter, Symbolic name. It is currently empty.</li> </ul> <p>Buttons on the right side of the dialog include: OK, Abbrechen, Objekt hinzufügen, and Konfig. Objekt löschen.</p>	Obj.Idx	Sub.	Parameter	Zugriff	6000	0	Number_Blocks_8_in	Lesen	6000	1	Read_8_Inputs_1H-8H	Lesen	6200	0	Number_Blocks_8_out	Lesen	6200	1	Write_8_Outputs_1H-8H	Lesen/
Obj.Idx	Sub.	Parameter	Zugriff																		
6000	0	Number_Blocks_8_in	Lesen																		
6000	1	Read_8_Inputs_1H-8H	Lesen																		
6200	0	Number_Blocks_8_out	Lesen																		
6200	1	Write_8_Outputs_1H-8H	Lesen/																		
6	<p>Wählen Sie <b>Read_8_Inputs_1H-8H (Idx.Obj. 6000, Sub.Idx. 1)</b>, klicken Sie auf <b>Objekt hinzufügen</b> und bestätigen Sie mit <b>Ok</b>.</p>																				
7	<p>Bestätigen Sie das Konfigurationsfenster in dem Sie auf <b>OK</b> klicken.</p>																				

## Deklaration von Slave 8

**Vorgehensweise** Folgende Tabelle stellt die verschiedenen Schritte zur Deklaration des Slave 8 dar.

Schritt	Aktion
1	Wählen Sie den Befehl <b>Einfügen</b> → <b>Knoten</b>
2	Stellen Sie den Cursor auf den Bus außerhalb des Rahmens, der den Master begrenzt.
3	<p>Wählen Sie:</p> <ul style="list-style-type: none"> <li>den Herstellername (z.B. ESD) :</li> <li>das Slave-Profil (z.B. 401),</li> <li>in der Slave-Liste den Slave-Typ (z.B. CBM-DIO8) und klicken Sie auf <b>Hinzufügen</b>,</li> <li>Geben Sie 8 im Feld <b>Knotenadresse</b> und den Slave-Modulnamen in das Feld <b>Beschreibung</b> ein,</li> <li>bestätigen Sie mit <b>Ok</b>.</li> </ul> <p><b>Ergebnis:</b> Folgende Struktur erscheint:</p>  <p><b>Master</b>  <i>Knotenadresse</i> 1  <i>Master</i> TSX CPP 100</p> <p><b>Slave_7_Eingaenge</b>  <i>Knotenadresse</i> 7  <i>Knoten</i> CBM-DIO8</p> <p><b>Slave_8_Ausgaenge</b>  <i>Knotenadresse</i> 8  <i>Knoten</i> CBM-DIO8</p>

## Konfiguration des Slave 8

**Auf einen Blick** Die Konfiguration des Slave 8 erfolgt genauso wie die Konfiguration des Slave 7 (siehe *Konfiguration des Slave 7, S. 107*).

Der wesentliche Unterschied besteht darin, Ausgänge anstatt Eingänge auszuwählen.

**Vorgehensweise** Folgende Tabelle stellt die verschiedenen Schritte zur Konfiguration des Slave 8 dar.

Schritt	Aktion
1	Doppelklicken Sie auf Slave 8. <b>Ergebnis:</b> Das Konfigurationsfenster erscheint
2	Wählen Sie das Objekt <b>1400 txPD01_Com</b> aus und klicken Sie auf die Schaltfläche <b>Hinzuf. zu den konf. PDOs</b> . <b>Ergebnis:</b> Das Konfigurationsfenster mit den PDO Merkmale erscheint
3	Verändern Sie eventuell die Frequenz beim Senden und bestätigen Sie mit <b>OK</b> .
4	Klicken Sie auf die Schaltfläche <b>Error Control Protokoll Konfiguration</b> . Ändern Sie <b>Life Time Factor</b> auf <b>3</b> und bestätigen Sie mit <b>OK</b> .
<b>Hinweis</b>	Falls einer der Parameter "Guard Time" oder "Life Time Factor" im Konfigurationsfenster des Slave auf den Wert "0" gesetzt wird, gehen die Ausgänge bei Unterbrechung der Verbindung (*) zwischen diesem Slave und der Karte TSX CPP 100 / TSX CPP 110 nicht auf ihren Fehlerwert. (* ) Diese Trennung der Verbindung könnte verursacht worden sein durch: <ul style="list-style-type: none"> <li>● Prozessorfehler oder Ausschalten seiner Versorgungsspannung</li> <li>● Trennen das Kabels zwischen PCMCIA-Karte und TAP</li> <li>● Trennen des Slave vom Bus</li> <li>● Defektes Buskabel</li> <li>● "Reset"-Befehl durch Sycon (Menü Online → Firmware / Reset),</li> <li>● Konfigurations-Ladebefehl durch Sycon (Menü Online → Download),</li> <li>● Firmware-Ladebefehl der Karte durch Sycon (Menü Online → Firmware Download,</li> </ul>
5	Wählen Sie <b>PDO_1400</b> im Feld <b>Konfigurierte PDOs</b> und klicken Sie auf die Schaltfläche <b>PDO Contents Mapping</b> .
6	<ul style="list-style-type: none"> <li>● Wählen Sie <b>Write_8_Ouputs_1H-8H (Obj.Idx. 6200, Sub.Idx. 1)</b>,</li> <li>● klicken Sie auf <b>Objekt hinzufügen</b> und anschließend</li> <li>● bestätigen Sie mit <b>Ok</b>.</li> </ul>
7	Bestätigen Sie das Konfigurationsfenster, indem Sie auf <b>OK</b> klicken.

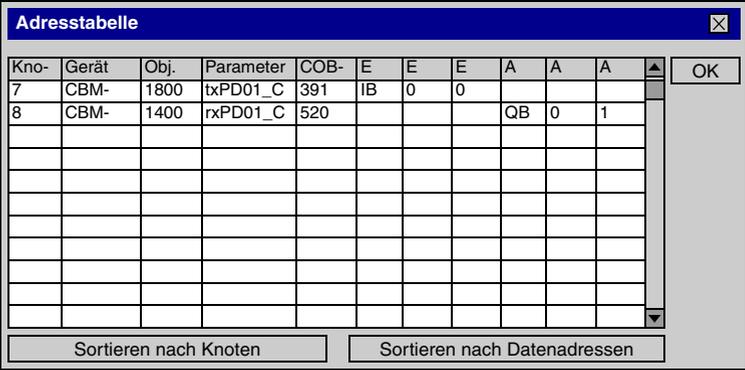
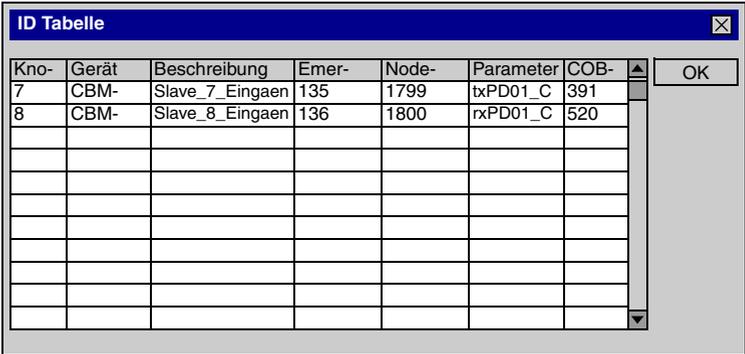
## Überprüfung der durchgeführten Konfiguration

**Auf einen Blick** Mit dem SyCon-Tool kann die in diesem Beispiel ausgeführte Arbeit überprüft werden. Der Zugriff zu den verschiedenen Ansichtstabellen wird hier vorgestellt.

**Hinweis:** nachdem Sie die Konfiguration Ihres Busses überprüft haben, vergessen Sie nicht sie als \*.CO-Datei zu sichern. Diese Datei wird beim Laden von der PL7-Applikation oder beim direkten Laden in die Karte TSX CPP 1•• benutzt.

**Vorgehensweise** Folgende Tabelle stellt die wesentliche Schritte der Bus-Konfigurationsüberprüfung anhand der Visualisierungstabellen dar.

Schritt	Aktion																																																																
1	<p>Wählen Sie den Befehl <b>Ansicht</b> → <b>Gerätetabelle</b></p> <p><b>Ergebnis:</b> Die Liste der konfigurierten Geräte erscheint in folgendem Fenster.</p>  <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>Kno-</th> <th>Gerät</th> <th>Guard Time</th> <th>Live Time</th> <th>Master Guarding</th> <th></th> </tr> </thead> <tbody> <tr> <td>1</td> <td>TSX CPP</td> <td></td> <td></td> <td></td> <td rowspan="3">OK</td> </tr> <tr> <td>7</td> <td>CBM-DIO8</td> <td>500</td> <td>3</td> <td></td> </tr> <tr> <td>8</td> <td>CBM-DIO8</td> <td>500</td> <td>3</td> <td></td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	Kno-	Gerät	Guard Time	Live Time	Master Guarding		1	TSX CPP				OK	7	CBM-DIO8	500	3		8	CBM-DIO8	500	3																																											
Kno-	Gerät	Guard Time	Live Time	Master Guarding																																																													
1	TSX CPP				OK																																																												
7	CBM-DIO8	500	3																																																														
8	CBM-DIO8	500	3																																																														

Schritt	Aktion																																	
2	<p>Wählen Sie den Befehl <b>Ansicht</b> → <b>Adresstabelle</b></p> <p><b>Ergebnis:</b> Die Liste der konfigurierten Slaves und die zugeordneten Dateneigenschaften erscheinen in folgendem Fenster.</p>  <p>The screenshot shows a window titled "Adresstabelle" with a table containing the following data:</p> <table border="1"> <thead> <tr> <th>Kno-</th> <th>Gerät</th> <th>Obj.</th> <th>Parameter</th> <th>COB-</th> <th>E</th> <th>E</th> <th>E</th> <th>A</th> <th>A</th> <th>A</th> </tr> </thead> <tbody> <tr> <td>7</td> <td>CBM-</td> <td>1800</td> <td>txPD01_C</td> <td>391</td> <td>IB</td> <td>0</td> <td>0</td> <td></td> <td></td> <td></td> </tr> <tr> <td>8</td> <td>CBM-</td> <td>1400</td> <td>rxPD01_C</td> <td>520</td> <td></td> <td></td> <td></td> <td>QB</td> <td>0</td> <td>1</td> </tr> </tbody> </table> <p>Buttons at the bottom: "Sortieren nach Knoten" and "Sortieren nach Datenadressen".</p>	Kno-	Gerät	Obj.	Parameter	COB-	E	E	E	A	A	A	7	CBM-	1800	txPD01_C	391	IB	0	0				8	CBM-	1400	rxPD01_C	520				QB	0	1
Kno-	Gerät	Obj.	Parameter	COB-	E	E	E	A	A	A																								
7	CBM-	1800	txPD01_C	391	IB	0	0																											
8	CBM-	1400	rxPD01_C	520				QB	0	1																								
3	<p>Wählen Sie den Befehl <b>Ansicht</b> → <b>ID-Tabelle</b></p> <p><b>Ergebnis:</b> Die Liste der konfigurierten Slaves mit ihren Kennungen erscheint in folgendem Fenster.</p>  <p>The screenshot shows a window titled "ID Tabelle" with a table containing the following data:</p> <table border="1"> <thead> <tr> <th>Kno-</th> <th>Gerät</th> <th>Beschreibung</th> <th>Emer-</th> <th>Node-</th> <th>Parameter</th> <th>COB-</th> </tr> </thead> <tbody> <tr> <td>7</td> <td>CBM-</td> <td>Slave_7_Eingaen</td> <td>135</td> <td>1799</td> <td>txPD01_C</td> <td>391</td> </tr> <tr> <td>8</td> <td>CBM-</td> <td>Slave_8_Eingaen</td> <td>136</td> <td>1800</td> <td>rxPD01_C</td> <td>520</td> </tr> </tbody> </table> <p>Buttons at the bottom: "OK".</p>	Kno-	Gerät	Beschreibung	Emer-	Node-	Parameter	COB-	7	CBM-	Slave_7_Eingaen	135	1799	txPD01_C	391	8	CBM-	Slave_8_Eingaen	136	1800	rxPD01_C	520												
Kno-	Gerät	Beschreibung	Emer-	Node-	Parameter	COB-																												
7	CBM-	Slave_7_Eingaen	135	1799	txPD01_C	391																												
8	CBM-	Slave_8_Eingaen	136	1800	rxPD01_C	520																												

Schritt	Aktion																																																																																																
4	<p>Wählen Sie den Befehl <b>Ansicht</b> → <b>SDO-Tabelle</b></p> <p><b>Ergebnis:</b> Die Liste der konfigurierten Slaves mit den Eigenschaften der SDOs erscheint in folgendem Fenster.</p> <div data-bbox="477 293 1225 776" style="border: 1px solid black; padding: 5px;"> <div style="background-color: #000080; color: white; padding: 2px; display: flex; justify-content: space-between; align-items: center;"> <span><b>SDO Tabelle</b></span> <span>✕</span> </div> <p style="margin-top: 5px;">Konfigurierte Objekte</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>Kno-</th> <th>Obj. Idx.</th> <th>Sub.</th> <th>Parameter</th> <th>Gewählter</th> <th>PDO Dia-</th> </tr> </thead> <tbody> <tr> <td></td> <td>1005</td> <td>0</td> <td>COB-ID Sync</td> <td>80</td> <td></td> </tr> <tr> <td></td> <td>1006</td> <td>0</td> <td>Communication Cycle</td> <td>64</td> <td></td> </tr> <tr> <td>7</td> <td>1800</td> <td>1</td> <td>COB-ID</td> <td>187</td> <td>X</td> </tr> <tr> <td></td> <td></td> <td>2</td> <td>Übertragungsart</td> <td>2</td> <td>X</td> </tr> <tr> <td></td> <td></td> <td>3</td> <td>Sendesperrzeit</td> <td>0</td> <td>X</td> </tr> <tr> <td></td> <td>1A00</td> <td>0</td> <td>Anzahl der gemappten</td> <td>1</td> <td>X</td> </tr> <tr> <td></td> <td></td> <td>1</td> <td>Read_8_Inputs_1H-8H</td> <td>60000108</td> <td>X</td> </tr> <tr> <td>8</td> <td>1400</td> <td>1</td> <td>COB-ID</td> <td>208</td> <td>X</td> </tr> <tr> <td></td> <td></td> <td>2</td> <td>Übertragungsart</td> <td>2</td> <td>X</td> </tr> <tr> <td></td> <td></td> <td>3</td> <td>Sendesperrzeit</td> <td>0</td> <td>X</td> </tr> <tr> <td></td> <td>1600</td> <td>0</td> <td>Anzahl der gemappten</td> <td>1</td> <td>x</td> </tr> <tr> <td></td> <td></td> <td>1</td> <td>Write_8_Outputs_1H-</td> <td>62000108</td> <td>x</td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> </tbody> </table> <div style="display: flex; justify-content: flex-end; margin-top: 5px; gap: 10px;"> <span>OK</span> <span>Dezimal</span> </div> <div style="display: flex; justify-content: space-between; margin-top: 5px; font-size: small;"> <span>Konfigurierte Obj. des PDO Dialogs</span> <span>Konfigurierte Obj. des PDO Dialogs</span> </div> </div>	Kno-	Obj. Idx.	Sub.	Parameter	Gewählter	PDO Dia-		1005	0	COB-ID Sync	80			1006	0	Communication Cycle	64		7	1800	1	COB-ID	187	X			2	Übertragungsart	2	X			3	Sendesperrzeit	0	X		1A00	0	Anzahl der gemappten	1	X			1	Read_8_Inputs_1H-8H	60000108	X	8	1400	1	COB-ID	208	X			2	Übertragungsart	2	X			3	Sendesperrzeit	0	X		1600	0	Anzahl der gemappten	1	x			1	Write_8_Outputs_1H-	62000108	x																		
Kno-	Obj. Idx.	Sub.	Parameter	Gewählter	PDO Dia-																																																																																												
	1005	0	COB-ID Sync	80																																																																																													
	1006	0	Communication Cycle	64																																																																																													
7	1800	1	COB-ID	187	X																																																																																												
		2	Übertragungsart	2	X																																																																																												
		3	Sendesperrzeit	0	X																																																																																												
	1A00	0	Anzahl der gemappten	1	X																																																																																												
		1	Read_8_Inputs_1H-8H	60000108	X																																																																																												
8	1400	1	COB-ID	208	X																																																																																												
		2	Übertragungsart	2	X																																																																																												
		3	Sendesperrzeit	0	X																																																																																												
	1600	0	Anzahl der gemappten	1	x																																																																																												
		1	Write_8_Outputs_1H-	62000108	x																																																																																												

## 4.2 Multi-Master-Konfiguration

---

### Auf einen Blick

**Inhalt des Abschnitts**

Dieser Abschnitt beschreibt anhand von Beispielen die Softwarekonfiguration eines CANopen-Bus auf dem zwei Master nebeneinander bestehen.

Diese Konfiguration wird mit Hilfe de SyCon-Tools durchgeführt.

---

**Inhalt dieses Abschnitts**

Dieser Abschnitt enthält die folgenden Themen:

Thema	Seite
Beispielbeschreibung	117
Deklaration des CANopen Masters A und der Slave 7 und 8	119
Deklaration des CANopen-Masters B	120
CANopen-Buskonfiguration des Masters B	122
Deklaration und Konfiguration der Slave 9 und 10	123
Deklaration und Konfiguration des Slave 7 für Master B	124
Deklaration und Konfiguration des Slave 8 für Master B	129

---

## Beispielbeschreibung

---

### Auf einen Blick

Dieses Beispiel wird aus didaktischen Gründen angegeben. Damit können Sie die verschiedenen Schritte der Konfiguration einer CANopen-Architektur verfolgen, die aus folgendem zusammengesetzt sind:

- ein TSX CPP 100 Mastermodul (auch Master A genannt) in einer Premium Steuerung installiert,
- ein TSX CPP 100 Mastermodul (auch Master B genannt) in einer Premium Steuerung installiert,
- zwei Bus-Anschlusseinheiten TSX CPP ACC1.
- vier Slavegeräte:
  - ein Modul mit 8 Eingängen (Slave 7),
  - ein Modul mit 8 Ausgängen (Slave 8),
  - ein Modul mit 8 Eingängen (Slave 9),
  - ein Modul mit 8 Ausgängen (Slave 10),

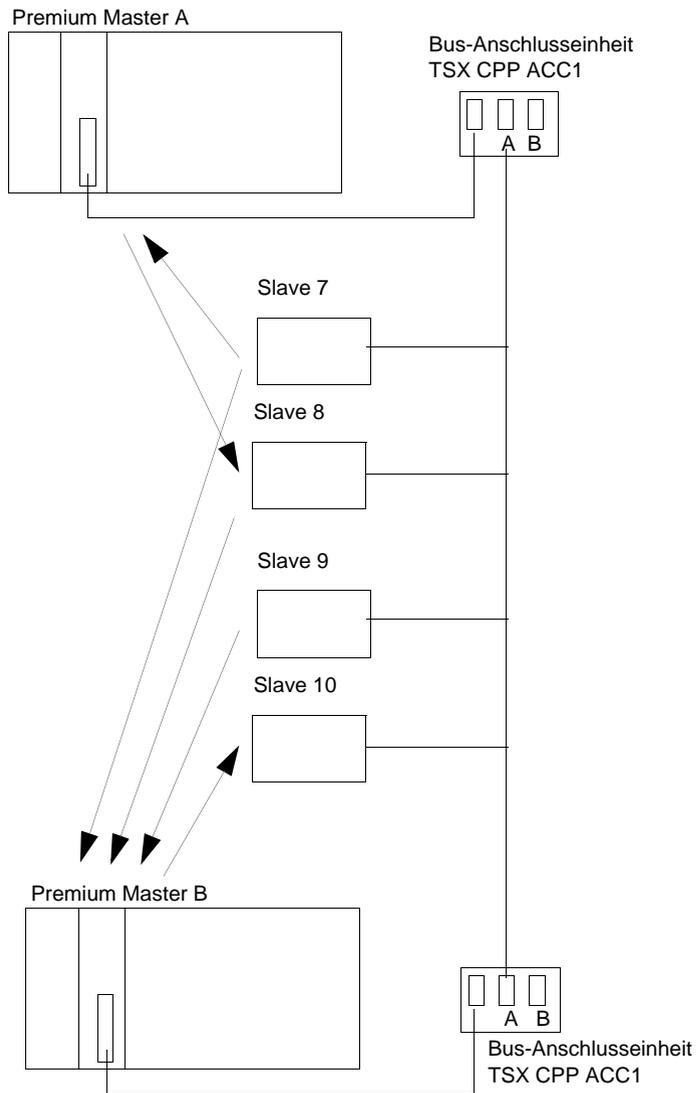
Die Übertragungsgeschwindigkeit über den Bus ist auf 1 Mbits/s festgesetzt.

von Master A und B verlangte Funktionen:

- der Master A verwaltet die Slaves 7 und 8.
  - der Master B verwaltet die Slaves 9 und 10.
  - der Master B ist Mithörer (Listener) bei den Slaves 7 und 8.
-

**Abbildung**

In folgender Darstellung wird die im Beispiel benutzte Struktur schematisiert.



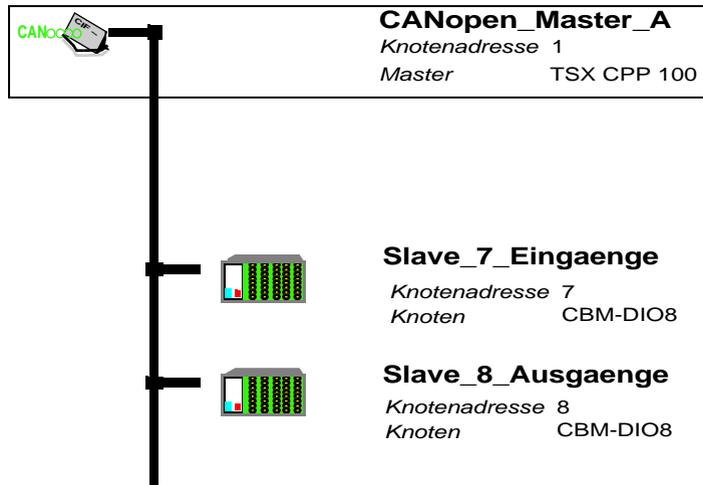
## Deklaration des CANopen Masters A und der Slave 7 und 8

### Auf einen Blick

Die Deklaration und die Konfiguration des CANopen A Busses sowie die Deklaration der Slave 7 und 8 erfolgt genauso wie bei dem Beispiel mit dem Standardbus (siehe *Standardkonfiguration, S. 100*).

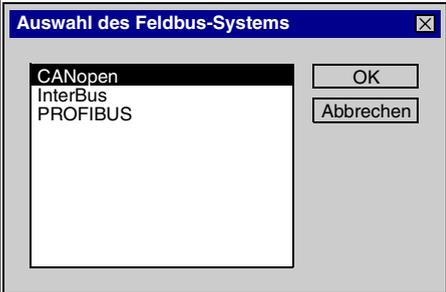
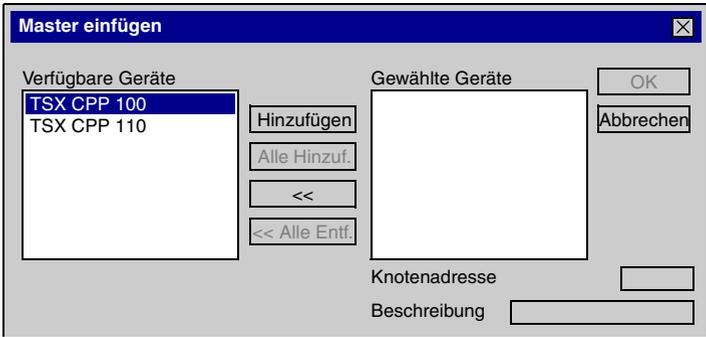
### Ergebnis

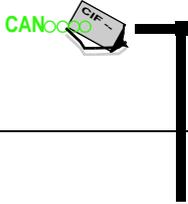
Folgende Abbildung zeigt die Konfiguration des CANopen Masters A:



## Deklaration des CANopen-Masters B

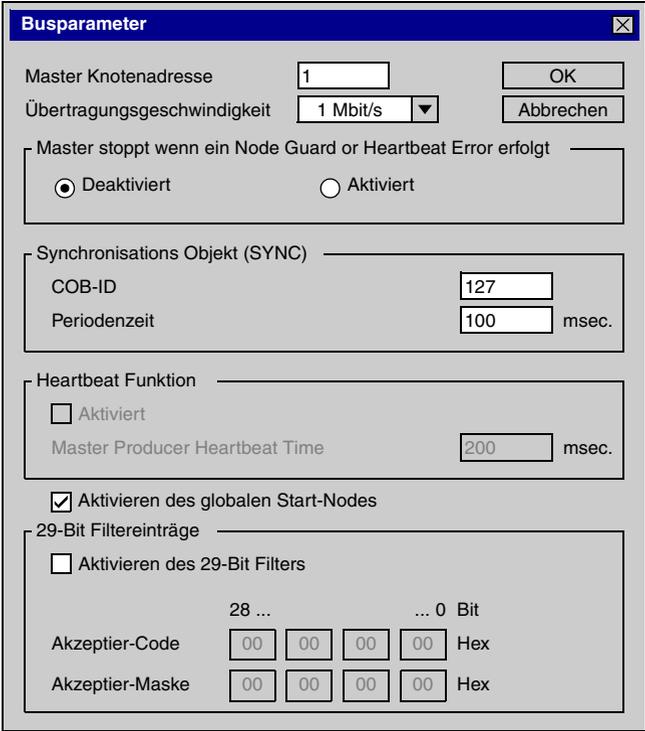
**Vorgehensweise** Folgende Tabelle stellt die verschiedenen Schritte zur Deklaration eines CANopen-Masters dar:

Schritt	Aktion
1	<p>Klicken Sie im Premium PL7 Konfigurationsfenster der TSX CPP100-Karte auf das Symbol  das Symbol <b>Einfügen</b>.</p> <p><b>Ergebnis:</b> Das SyCon-Tool erscheint im Fenster.</p>
2	<p>Wählen Sie den Befehl <b>Datei</b> → <b>Neu</b></p> <p><b>Ergebnis:</b> Folgendes Fenster erscheint:</p> 
3	<p>Wählen Sie <b>CANopen</b> aus und bestätigen Sie mit <b>Ok</b>.</p> <p><b>Ergebnis:</b> Eine leere Struktur erscheint im Fenster.</p>
4	<p>Wählen Sie den Befehl <b>Einfügen</b> → <b>Master</b></p> <p><b>Ergebnis:</b> Folgendes Fenster erscheint:</p> 

Schritt	Aktion
5	<ul style="list-style-type: none"><li>● Wählen Sie <b>TSX CPP 100</b> aus,</li><li>● Klicken Sie <b>Hinzufügen</b>,</li><li>● Geben Sie ein Master-Modulname im Feld <b>Beschreibung</b> ein, <b>Anmerkung:</b> Der Name darf weder Leerzeichen noch hervorgehobene Zeichen beinhalten und ist auf maximal 32 Zeichen begrenzt.</li><li>● bestätigen Sie mit <b>Ok</b>.</li></ul> <p><b>Ergebnis:</b> Folgende Struktur erscheint:</p> <div data-bbox="474 418 1212 558" style="border: 1px solid black; padding: 5px;"><p><b>CANopen</b> <b>CANopen_Master_B_Listener</b></p><p><i>Knotenadresse</i> 1 <i>Master</i> TSX CPP 100</p></div>

## CANopen-Buskonfiguration des Masters B

**Vorgehensweise** Folgende Tabelle stellt die verschiedenen Schritte zur Konfiguration eines CANopen-Busses dar:

Schritt	Aktion
1	<p>Wählen Sie einen Befehl aus <b>Einstellungen</b> → <b>Busparameter</b>.  <b>Ergebnis:</b> Folgendes Fenster erscheint:</p> 
2	<p>Konfigurieren Sie:</p> <ul style="list-style-type: none"> <li>• die Geschwindigkeit mit 1Mbit/s,</li> <li>• den Wert <b>SYNC COB-ID</b> mit 127 (jeder Master am gleichen Bus muss einen unterschiedlichen SYNC COB-ID haben),</li> <li>• die Periodenzeit mit 100 ms.</li> </ul>
3	<p>Wählen Sie <b>Deaktiviert</b> bei <b>Master stoppt wenn ein Node Guard or Heartbeat Error erfolgt</b>.</p>
4	<p>Wählen Sie <b>Enable Global Start Node</b>.</p>
5	<p>Bestätigen Sie mit <b>Ok</b>.</p>

## Deklaration und Konfiguration der Slave 9 und 10

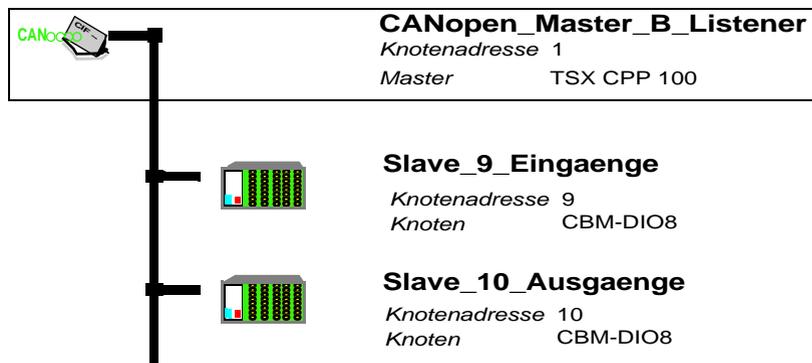
### Auf einen Blick

Die Deklaration und die Konfiguration der Slave 9 und 10 erfolgt genauso wie bei den Slave 7 und 8 des Standardbusses, nur die Knotennummer und die mit jedem Slave assoziierte Beschreibung wechselt:

- Deklaration des Slave 9 (siehe *Deklaration von Slave 7, S. 105*),
- Konfiguration des Slave 9 (siehe *Konfiguration des Slave 7, S. 107*),
- Deklaration des Slave 10 (siehe *Deklaration von Slave 8, S. 111*),
- Konfiguration des Slave 10 (siehe *Konfiguration des Slave 8, S. 112*),

### Ergebnis

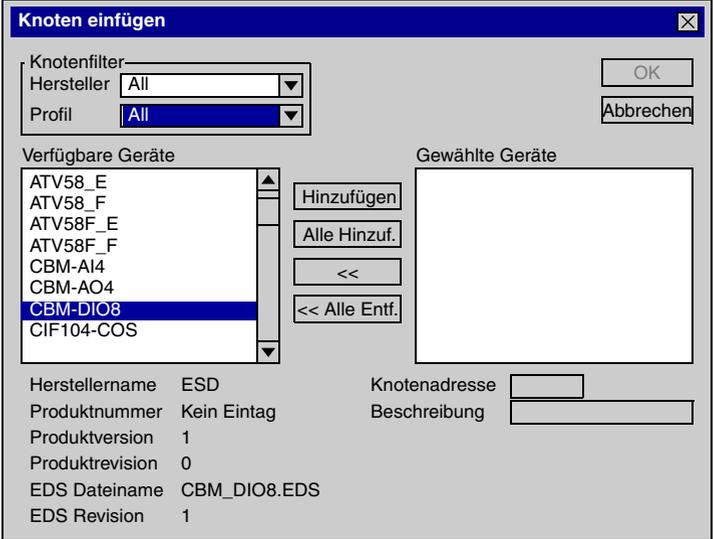
Folgende Struktur zeigt die Konfiguration der Slave 9 und 10 an Master B.



## Deklaration und Konfiguration des Slave 7 für Master B

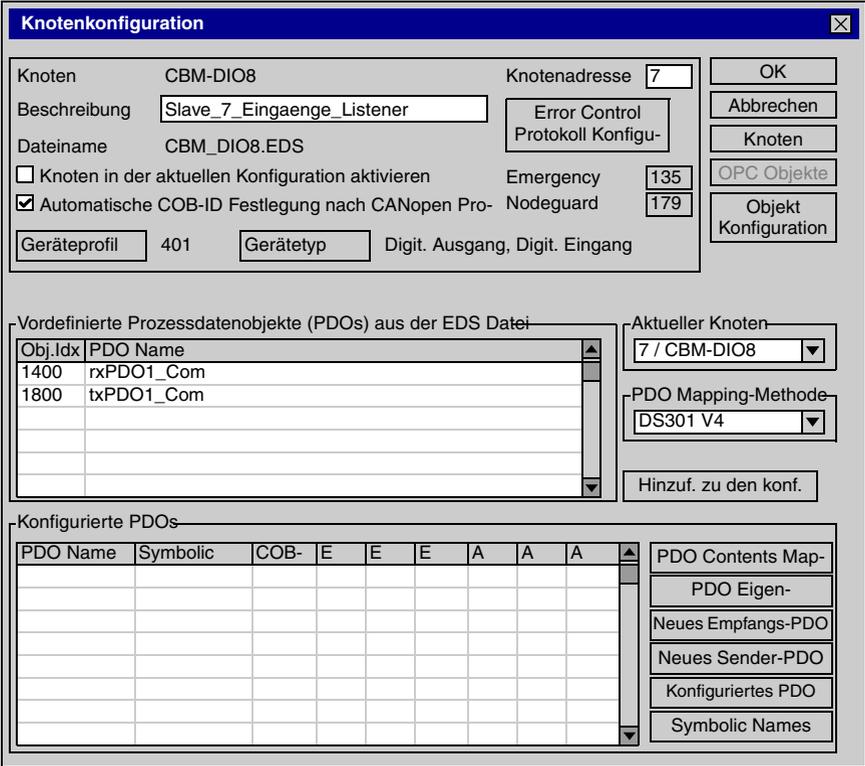
### Wie erfolgt die Deklaration des Slave 7

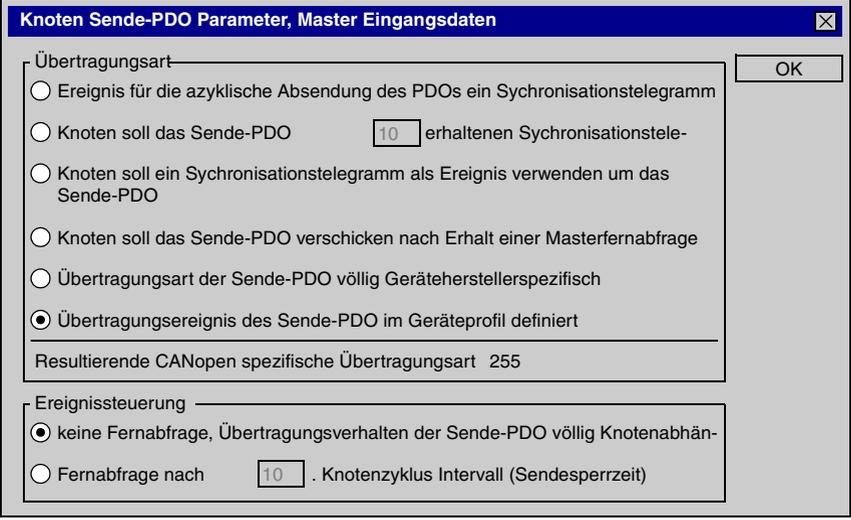
Folgende Tabelle stellt die verschiedenen Schritte zur Deklaration des Slave 7 dar.

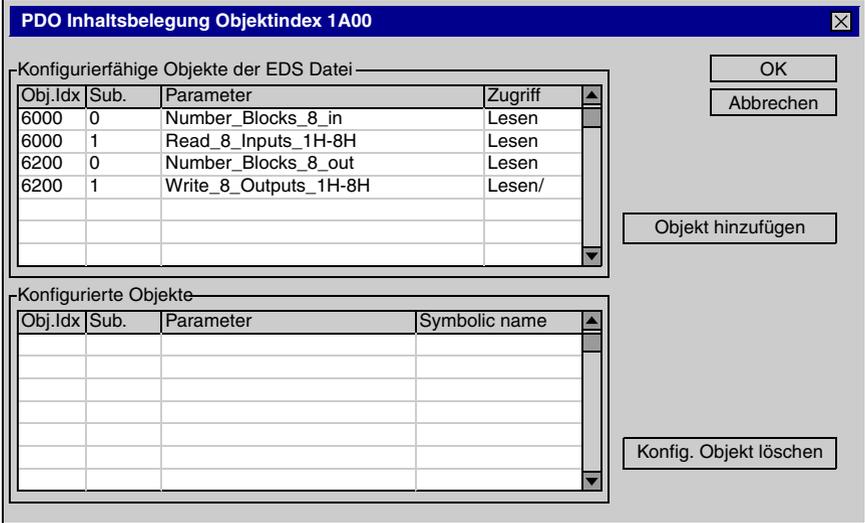
Schritt	Aktion
1	Wählen Sie den Befehl <b>Einfügen</b> → <b>Knoten</b> <b>Ergebnis:</b> Ein Cursor erscheint :
2	Stellen Sie den Cursor auf den Bus außerhalb des Rahmens, der den Master begrenzt. <b>Ergebnis:</b> Folgendes Fenster erscheint: 
3	Wählen Sie: <ul style="list-style-type: none"> <li>• den Herstellername (z.B. ESD) :</li> <li>• das Slave-Profil (z.B. 401),</li> <li>• in der Slave-Liste den Slave-Typ (z.B. CBM-DIO8) und klicken Sie auf <b>Hinzufügen</b>,</li> <li>• 7 im Feld <b>Knotenadresse</b> und geben Sie den Slave-Modulnamen in das Feld <b>Beschreibung</b> ein,</li> <li>• bestätigen Sie mit <b>Ok</b>.</li> </ul>

**Wie wird der Slave 7 konfiguriert**

Folgende Tabelle stellt die verschiedenen Schritte zur Konfiguration des Slave 7 dar.

Schritt	Aktion
1	<p>Führen Sie ein Doppelklick auf Slave 7 aus.  <b>Ergebnis:</b> Folgendes Fenster erscheint:</p> 
2	<p>Deaktivieren Sie das Kontrollkästchen <b>Knoten in der aktuellen Konfiguration aktivieren</b>.</p>

Schritt	Aktion
3	<p>Wählen Sie das Objekt <b>1800 txPDO1_Com</b> aus und klicken Sie auf die Taste <b>Hinzuf. zu den konf. PDOs</b>.</p> <p><b>Ergebnis:</b> Folgendes Fenster erscheint:</p> 
4	<ul style="list-style-type: none"> <li>● Wählen Sie <b>Übertragungsereignis des Sende-PDO im Geräteprofil definiert</b>,</li> <li>● Wählen Sie <b>keine Fernanfrage, Übertragungsverhalten der Sende-PDO völlig Knotenabhängig</b>,</li> <li>● bestätigen Sie mit <b>Ok</b>.</li> </ul>

Schritt	Aktion
5	<p>Wählen Sie <b>PDO_1800</b> im Feld <b>Konfigurierte PDOs</b> und klicken Sie auf die Schaltfläche <b>PDO Contents Mapping</b>.</p> <p><b>Ergebnis:</b> Folgendes Fenster erscheint:</p> 
6	<p>Wählen Sie <b>Read_8_Inputs_1H-8H (Obj.Idx. 6000, Sub.Idx. 1)</b>, klicken Sie auf <b>Objekt hinzufügen</b> und bestätigen Sie mit <b>Ok</b>.</p>

Schritt	Aktion
7	<p>Bestätigen Sie das Konfigurationsfenster in dem Sie auf <b>OK</b> klicken.</p> <p><b>Ergebnis:</b> Folgende Struktur erscheint:</p> <div style="display: flex; align-items: center; gap: 20px;"> <div data-bbox="336 284 637 771" style="flex: 1;"> <p>The diagram shows a vertical backbone with four nodes. At the top is a master node labeled 'CANopen'. Below it are three slave nodes. The first two are active, and the third is crossed out with a large red 'X'.</p> </div> <div data-bbox="713 284 1138 755" style="flex: 2;"> <p><b>CANopen_Master_B_Listener</b>  <i>Knotenadresse</i> 1  <i>Master</i> TSX CPP 100</p> <p><b>Slave_9_Eingaenge</b>  <i>Knotenadresse</i> 9  <i>Knoten</i> CBM-DIO8</p> <p><b>Slave_10_Ausgaenge</b>  <i>Knotenadresse</i> 10  <i>Knoten</i> CBM-DIO8</p> <p><b>Slave_7_Eingänge_Listener</b>  <i>Knotenadresse</i> 7  <i>Knoten</i> CBM-DIO8</p> </div> </div>

## Deklaration und Konfiguration des Slave 8 für Master B

### Slave 8 deklarieren

Folgende Tabelle stellt die verschiedenen Schritte zur Deklaration des Slave 8 dar.

Schritt	Aktion
1	Wählen Sie den Befehl <b>Einfügen</b> → <b>Knoten</b> <b>Ergebnis:</b> Ein Cursor erscheint :
2	Stellen Sie den Cursor auf den Bus außerhalb des Rahmens, der den Master begrenzt. <b>Ergebnis:</b> Folgendes Fenster erscheint:

**Knoten einfügen**✕

Knotenfilter:

Hersteller

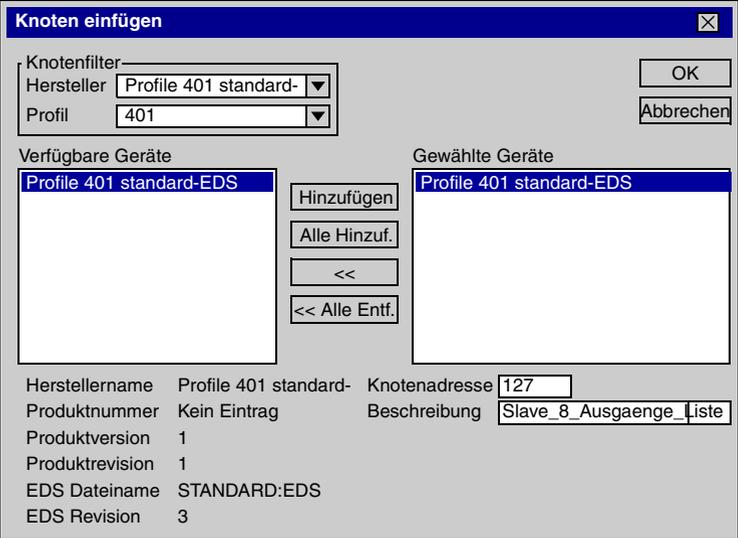
Profil

Verfügbare Geräte

- ATV58\_E
- ATV58\_F
- ATV58F\_E
- ATV58F\_F
- CBM-AI4
- CBM-AO4
- CBM-DIO8
- CIF104-COS

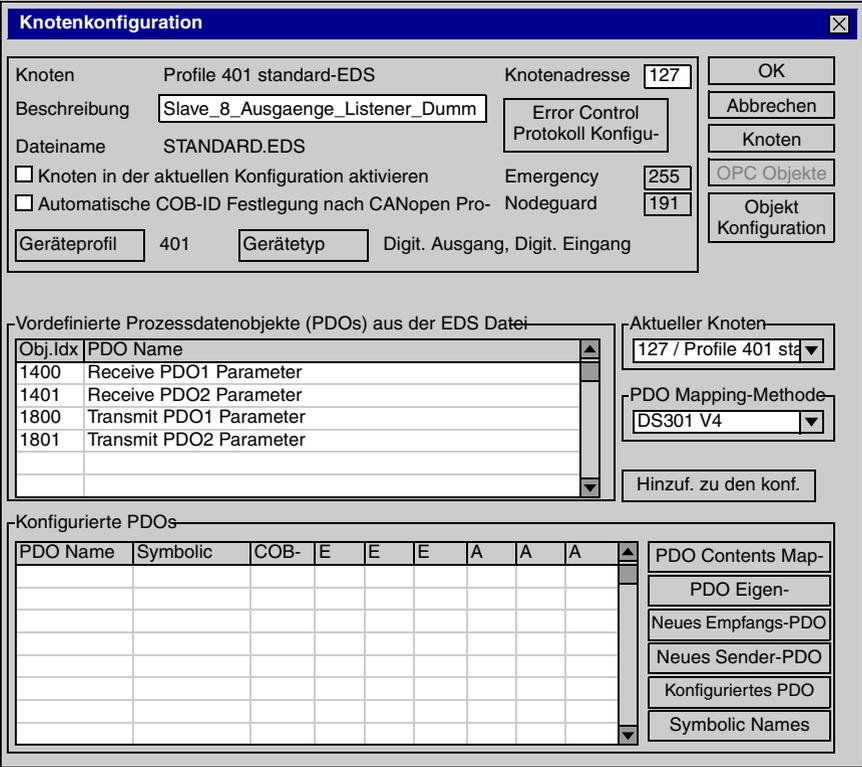
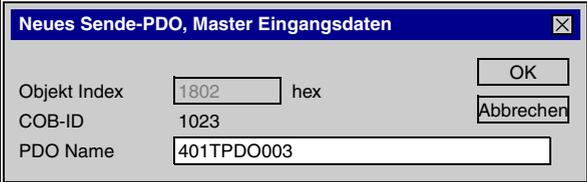
Gewählte Geräte

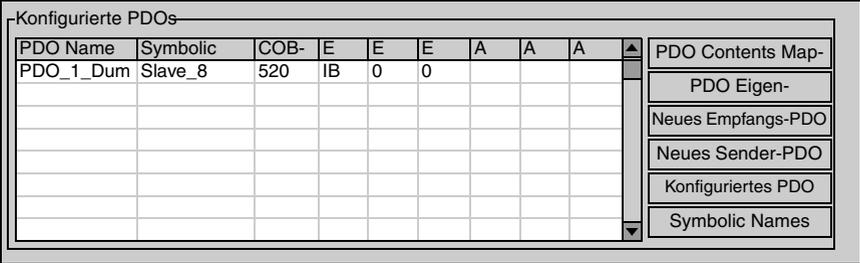
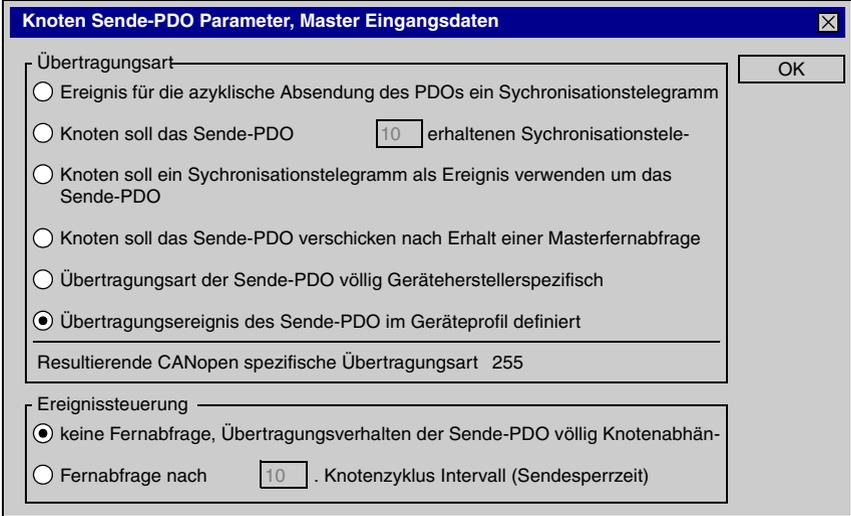
Herstellername	ESD	Knotenadresse	<input type="text"/>
Produktnummer	Kein Eintag	Beschreibung	<input type="text"/>
Produktversion	1		
Produktrevision	0		
EDS Dateiname	CBM_DIO8.EDS		
EDS Revision	1		

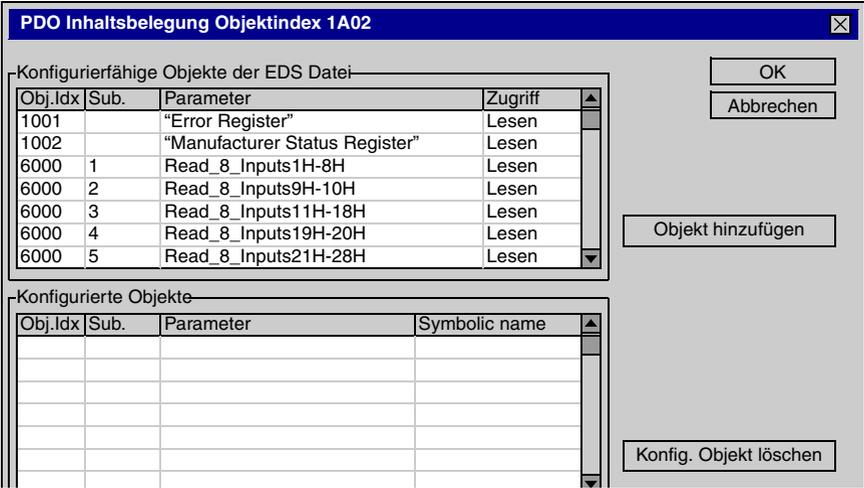
Schritt	Aktion																								
3	<p>Wählen Sie:</p> <ul style="list-style-type: none"> <li>● <b>Profile 401 Standard-EDS</b> im Hersteller-Feld,</li> <li>● <b>401</b> im Profil-Feld,</li> <li>● Klicken Sie <b>Hinzufügen</b>,</li> <li>● Geben Sie <b>127</b> im Feld <b>Knotenadresse</b> und den Slave-Modulnamen in das Feld <b>Beschreibung</b> ein,</li> </ul> <p><b>Ergebnis :</b></p>  <table border="1" data-bbox="518 800 1229 959"> <tr> <td>Herstellername</td> <td>Profile 401 standard-</td> <td>Knotenadresse</td> <td>127</td> </tr> <tr> <td>Produktnummer</td> <td>Kein Eintrag</td> <td>Beschreibung</td> <td>Slave_8_Ausgaenge_Liste</td> </tr> <tr> <td>Produktversion</td> <td>1</td> <td></td> <td></td> </tr> <tr> <td>Produktrevision</td> <td>1</td> <td></td> <td></td> </tr> <tr> <td>EDS Dateiname</td> <td>STANDARD:EDS</td> <td></td> <td></td> </tr> <tr> <td>EDS Revision</td> <td>3</td> <td></td> <td></td> </tr> </table>	Herstellername	Profile 401 standard-	Knotenadresse	127	Produktnummer	Kein Eintrag	Beschreibung	Slave_8_Ausgaenge_Liste	Produktversion	1			Produktrevision	1			EDS Dateiname	STANDARD:EDS			EDS Revision	3		
Herstellername	Profile 401 standard-	Knotenadresse	127																						
Produktnummer	Kein Eintrag	Beschreibung	Slave_8_Ausgaenge_Liste																						
Produktversion	1																								
Produktrevision	1																								
EDS Dateiname	STANDARD:EDS																								
EDS Revision	3																								
4	bestätigen Sie mit <b>Ok</b> .																								

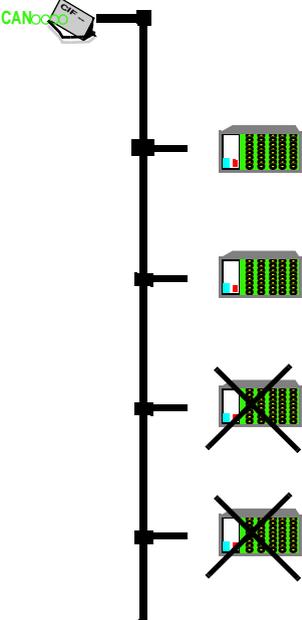
## Konfiguration von Slave 8

Folgende Tabelle stellt die verschiedenen Schritte zur Konfiguration des Slave 8 dar.

Schritt	Aktion
1	<p>Doppelklicken Sie auf den Slave 8.  <b>Ergebnis:</b> Folgendes Fenster erscheint:</p> 
2	Deaktivieren Sie das Kontrollkästchen <b>Knoten in der aktuellen Konfiguration aktivieren</b> .
3	Deaktivieren Sie das Kontrollkästchen <b>Automatische COB-ID Festlegung nach CANopen Profil 301</b> .
4	<p>Klicken Sie auf <b>Neues Sender-PDO def..</b>  <b>Ergebnis:</b> Folgendes Fenster erscheint:</p> 

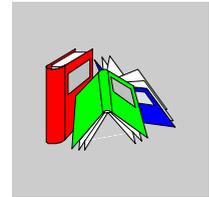
Schritt	Aktion
5	Geben Sie dem PDO einen Namen und bestätigen Sie mit <b>OK</b> .
6	<p>Geben Sie einen symbolischen Namen ein (Spalte <b>Symbolic Name</b>).</p> <p>Geben Sie ein COB-ID, z.B. 520, ein. Vorsicht, diese Nummer muss mit der des Slave 8 am Bus A identisch sein.</p> <p><b>Ergebnis :</b></p> 
7	<p>Klicken Sie auf <b>PDO Eigenschaften</b>.</p> <p><b>Ergebnis:</b> Folgendes Fenster erscheint:</p> 
8	<ul style="list-style-type: none"> <li>• Wählen Sie <b>Übertragungsereignis des Sende-PDO im Geräteprofil definiert</b>,</li> <li>• Wählen Sie <b>keine Fernabfrage, Übertragungsverhalten der Sende PDO völlig Knotenabhängig</b>,</li> <li>• bestätigen Sie mit <b>Ok</b>.</li> </ul>

Schritt	Aktion																																
9	<p>Wählen Sie <b>PDO_1_Dummy</b> im Feld <b>Konfigurierte PDOs</b> und klicken Sie auf die Schaltfläche <b>Mapping Inhalt PDO</b>.</p> <p><b>Ergebnis:</b> Folgendes Fenster erscheint:</p>  <p>The screenshot shows a dialog box titled "PDO Inhaltsbelegung Objektindex 1A02". It contains two tables. The first table, "Konfigurierfähige Objekte der EDS Datei", has the following data:</p> <table border="1"> <thead> <tr> <th>Obj.Idx</th> <th>Sub.</th> <th>Parameter</th> <th>Zugriff</th> </tr> </thead> <tbody> <tr> <td>1001</td> <td></td> <td>"Error Register"</td> <td>Lesen</td> </tr> <tr> <td>1002</td> <td></td> <td>"Manufacturer Status Register"</td> <td>Lesen</td> </tr> <tr> <td>6000</td> <td>1</td> <td>Read_8_Inputs1H-8H</td> <td>Lesen</td> </tr> <tr> <td>6000</td> <td>2</td> <td>Read_8_Inputs9H-10H</td> <td>Lesen</td> </tr> <tr> <td>6000</td> <td>3</td> <td>Read_8_Inputs11H-18H</td> <td>Lesen</td> </tr> <tr> <td>6000</td> <td>4</td> <td>Read_8_Inputs19H-20H</td> <td>Lesen</td> </tr> <tr> <td>6000</td> <td>5</td> <td>Read_8_Inputs21H-28H</td> <td>Lesen</td> </tr> </tbody> </table> <p>The second table, "Konfigurierte Objekte", is empty and has the following columns: "Obj.Idx", "Sub.", "Parameter", and "Symbolic name".</p> <p>Buttons in the dialog include "OK", "Abbrechen", "Objekt hinzufügen", and "Konfig. Objekt löschen".</p>	Obj.Idx	Sub.	Parameter	Zugriff	1001		"Error Register"	Lesen	1002		"Manufacturer Status Register"	Lesen	6000	1	Read_8_Inputs1H-8H	Lesen	6000	2	Read_8_Inputs9H-10H	Lesen	6000	3	Read_8_Inputs11H-18H	Lesen	6000	4	Read_8_Inputs19H-20H	Lesen	6000	5	Read_8_Inputs21H-28H	Lesen
Obj.Idx	Sub.	Parameter	Zugriff																														
1001		"Error Register"	Lesen																														
1002		"Manufacturer Status Register"	Lesen																														
6000	1	Read_8_Inputs1H-8H	Lesen																														
6000	2	Read_8_Inputs9H-10H	Lesen																														
6000	3	Read_8_Inputs11H-18H	Lesen																														
6000	4	Read_8_Inputs19H-20H	Lesen																														
6000	5	Read_8_Inputs21H-28H	Lesen																														
10	<p>Wählen Sie <b>Write_8_Outputs_1H-8H (Obj.Idx. 6200, Sub.1)</b> (benutzen Sie die Bildlaufleiste), klicken Sie auf <b>Objekt hinzufügen</b> und bestätigen Sie mit <b>OK</b>.</p>																																

Schritt	Aktion
11	<p>Bestätigen Sie das Konfigurationsfenster in dem Sie auf <b>OK</b> klicken.</p> <p><b>Ergebnis:</b> Folgende Struktur erscheint:</p> <div style="display: flex; align-items: center;">  <div style="margin-left: 20px;"> <p><b>CANopen_Master_B_Listener</b>  <i>Knotenadresse</i> 1  <i>Master</i> TSX CPP 100</p> <p><b>Slave_9_Eingaenge</b>  <i>Knotenadresse</i> 9  <i>Knoten</i> CBM-DIO8</p> <p><b>Slave_10_Ausgaenge</b>  <i>Knotenadresse</i> 10  <i>Knoten</i> CBM-DIO8</p> <p><b>Slave_7_Eingänge_Listener</b>  <i>Knotenadresse</i> 7  <i>Knoten</i> CBM-DIO8</p> <p><b>Slave_8_Ausgaenge_Listener_Dummy</b>  <i>Knotenadresse</i> 127  <i>Knoten</i> Profil 401 EDS-Standard</p> </div> </div>

---

# Glossar



---

!

**'SDO'**                    **Service Data Object** : Man unterscheidet zwischen den SSDO (Server SDO) und CSDO (Client SDO).

---

C

**CAN**                    **Controller Area Network** : Geländebus ursprünglich für das Auto entwickelt, wird jetzt in mehreren Bereichen benutzt von der Industrie bis im Tertiärsektor.

**CiA**                    **CAN in Automation** : internationale Gruppierung der Benutzer und Hersteller von CAN-Produkten.

**COB**                    **Communication Object** : Transporteinheit über CAN-Bus. Ein COB wird durch ein bestimmtes codiertes Kennzeichen mit 11 Bit, [0, 2047] gekennzeichnet. Ein COB besteht aus maximal 8 Datenbytes. Der Übertragungsvorrang eines COBs erteilt das Kennzeichen, je schwächer das Kennzeichen ist desto vorrangiger ist das gruppierte COB.

**CRC**                    **Cyclic Redundancy Checksum** : Zyklische Redundanz-Prüfsumme zeigt an, dass kein Zeichen während der Übertragung des Frame "verformt" wurde.

**CSMA/CA**            **Carrier Sense, Multiple Access / Collision Avoidance** : Verwaltungsmethode der Kommunikation über ein für den Link Layer charakteristisches Netz.

---

**D**

- DIN**                    **Deutsches Institut für Normung:** Deutsches Institut für Normung.
- DS**                     **Draft Standard:** von der Organisation CIA herausgegebenes Normdokument.
- 

**E**

- EDS**                    **Electronic Data Sheet:** Beschreibungsdatei von jedem CAN-Gerät (wird vom Hersteller geliefert). Um mit der Sycon-Konfigurationssoftware ein CAN-Gerät dem Bus hinzuzufügen, muss das zugehörige EDS ausgewählt werden Die EDS stehen auf der Site <http://www.can-cia.de> zur Verfügung oder bei einem Materiallieferanten.
- 

**L**

- Life Time**            **Life Time = Life Time factor x Guard Time .**
- LLC**                    **Logical Link Control .**
- 

**M**

- MAC**                    **Medium Access Control .**
- MDI**                    **Medium Dependent Interface .**
- MTBF**                    **Mean Time Between Failure :** Durchschnittszeit zwischen zwei Störungen.
-

**O**

**OD**                    **Object Dictionary** : Lexikon der von CAN erkannten Objekte. Ein hexadezimaler Code wird jedem Objekttyp erteilt, das Lexikon fasst die Codes aller Objekte zusammen.

---

**P**

**PCMCIA**            **Personal Computer Memory Card International Association**

**PDO**                    **Process Data Object** : man unterscheidet zwischen den RPDO (Recieve PDO) und TPDO (Transmit PDO).

**PDU**                    **Process Data Unit** : Es gibt die APDU (Application PDU). Eine PDU auf dem Link Layer ist eine von Header und Bytes, die für diese Verbindung charakteristisch sind, eingekapselte APDU.

**PMA**                    **Physical Medium Attachment.**

---

**T**

**TAP**                    **Transmission Access Point** : Anschlussgehäuse des Bus.

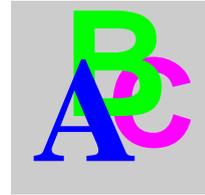
---



---

# Index

---



## A

Allgemeines, 9  
Anzeigen, 73  
Architektur, 11  
Ausgangsworte, 87  
Austauschverwaltung, 89

## B

Buslänge, 12

## C

CANopen, 10  
Configuration  
    Slave-Ausgang, 42

## D

Debug, 69  
Debugging, 69  
Diagnose, 72, 75  
    Befehl  
        , 67  
Dokument, 54

## E

Einbau der PCMCIA-Karte, 19, 21  
Eingangswörter, 84  
Expliziter Austausch, 88

## F

Fehlercodes, 92

## I

IDENTIFICATION, 64  
Impliziter Austausch, 83  
Inbetriebnahme  
    Prinzip, 33

## K

Kenndaten, 26  
Konfiguration, 37  
    CANopen-Bus Verhalten, 42  
    Laden, 45  
    Slave-Eingang, 42  
    Vorgehensweise, 51  
Konfigurationsbildschirm, 39

## L

LEDs, 73

## N

Normen, 26

## P

PDU, 62

Physikalische Beschreibung CPP100/110,  
17  
Prozessoren, 28

## **R**

READ\_VAR, 57  
Rückmeldung, 89

## **S**

Software, 31  
Sprachobjekte, 82  
    Explizite, 88  
    Implizite, 83  
    Konfiguration, 90

## **T**

Topologie, 11  
TSX CPP 100/110, 15  
TSX CPP ACC1, 23

## **U**

Übertragungsgeschwindigkeit, 12

## **V**

Vorgehensweise, 35

## **W**

Wählen Sie eine Konfigurationsdatei aus, 51  
WRITE\_VAR, 57

## **Z**

Zugriff auf das Konfigurationsfenster, 38