

# **KP1**

Temperaturüberwachung

der

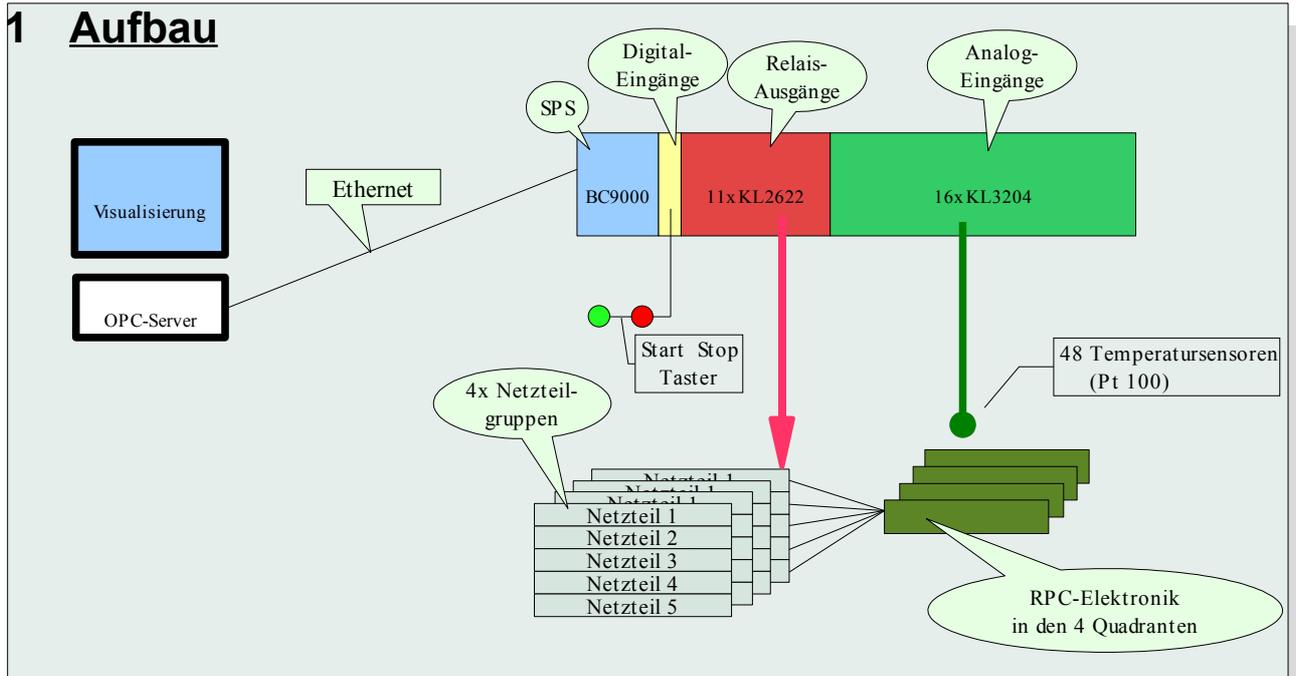
MMRPC-Elektronik

mit einer Beckhoff – SPS

für NOT-Abschaltung der Netzteile

# Inhaltsverzeichnis

1 Aufbau.....	3
2 Realisierung.....	3
2.1 Temperaturmessklemmen.....	4
2.2 Schaltausgänge.....	4
2.3 SPS.....	5
2.4 Konfiguration der Station.....	5
2.4.1 Einloggen.....	5
2.4.2 Stationsanzeige.....	6
2.4.3 Reset.....	6
2.4.4 Einstellungen aktualisieren.....	7
2.4.5 Mapping.....	7
3 Inbetriebnahme.....	8
4 OPC-Konfiguration.....	9
5 OPC Test Client.....	9

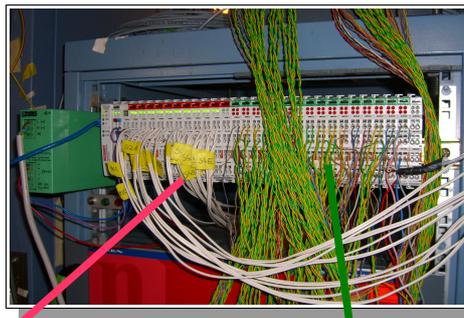


Die SPS misst an der MMRPC-Elektronik mit Pt100-Messfühlern die Temperatur. Wird mindestens eine Temperatur über 60°C gemessen, schalten die Relaisausgänge die Netzteile über Interlock ab. Dabei werden zuerst die negativen Spannungen abgeschaltet, kurz danach die Netzteile für die positiven Spannungen. Eine Abschaltung findet auch bei Störungen der Klemmen statt.

Die Messwerte werden einem OPC-Server zur Verfügung gestellt, der Diese an eine Visualisierung (mit LabVIEW realisiert) weiter gibt.

## 2 Realisierung

SPS mit I/O-Klemmen

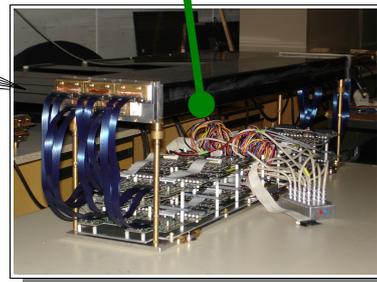


Netzteil-Block

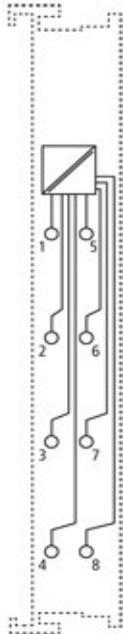


1.

RPC-Elektronik



## 2.1 Temperaturmessklemmen



Als Temperaturmessklemmen kommen die vierkanaligen **KL3204** zum Einsatz.

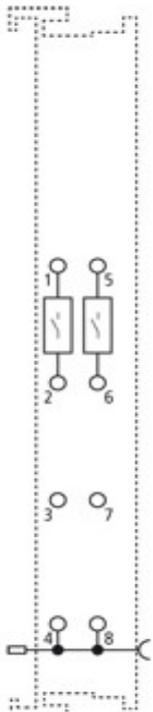
Die Beschaltung der vier Pt100-Widerstände (R1...R4) erfolgt in Zweileitertechnik und ist im nebenstehenden Bild zu sehen.



Die vier roten LEDs signalisieren Drahtbruch oder Kurzschluss der Sensoren.



## 2.2 Schaltausgänge



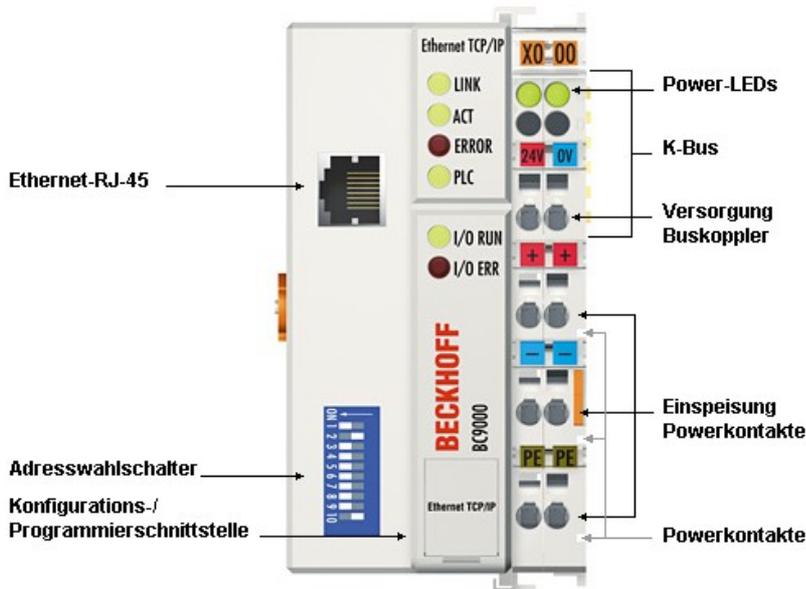
Die Netzteile werden durch die potenzialfreien Kontakte der zweikanaligen Relaisklemme **KL2602** geschaltet.

Die Kontakte liegen für Kanal 1 an Pin 1 und 2 und für Kanal 2 auf Pin 5 und 6

Die beiden grünen LEDs zeigen den Zustand der Relais an.

## 2.3 SPS

Als SPS wird die **BC9000** verwendet, die einen Ethernet-Anschluss besitzt und bei Einschalten der Betriebsspannung (+24V) das im EEPROM gespeicherte Bootprojekt startet.



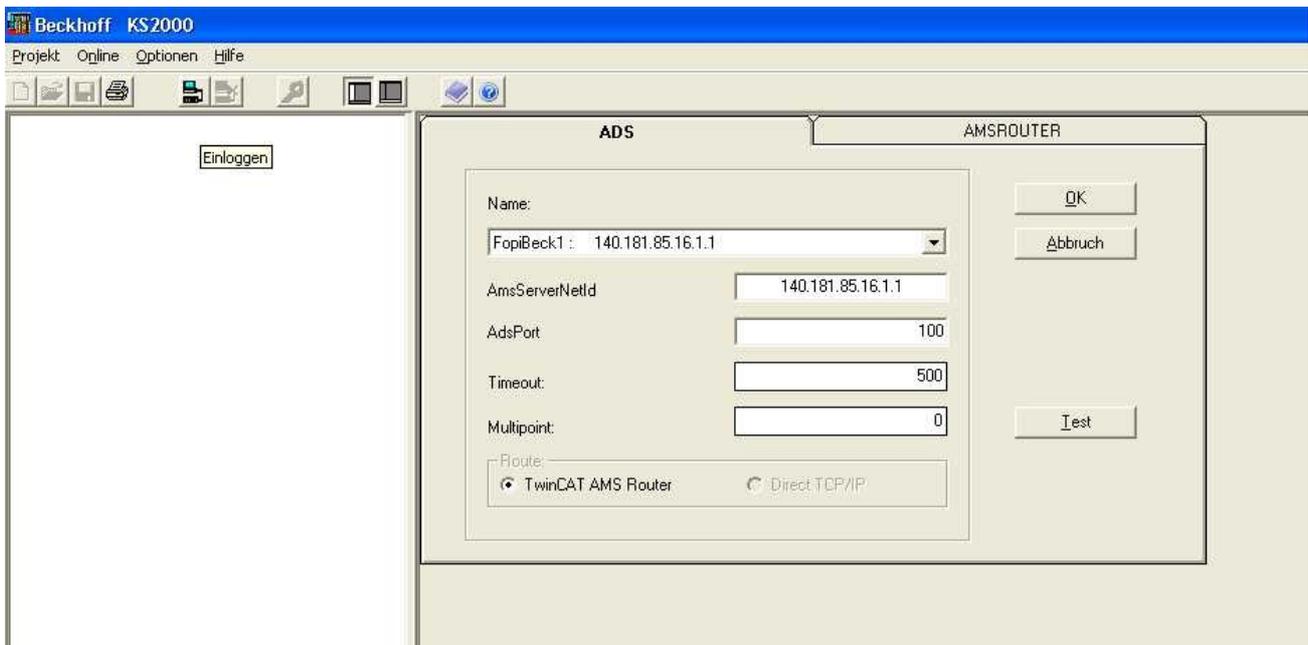
Die Bedeutung der LEDs ist in Anhang 1 beschrieben.

Im Normalbetrieb leuchten die grünen LEDs, die roten LEDs sind aus.

## 2.4 Konfiguration der Station

### 2.4.1 Einloggen

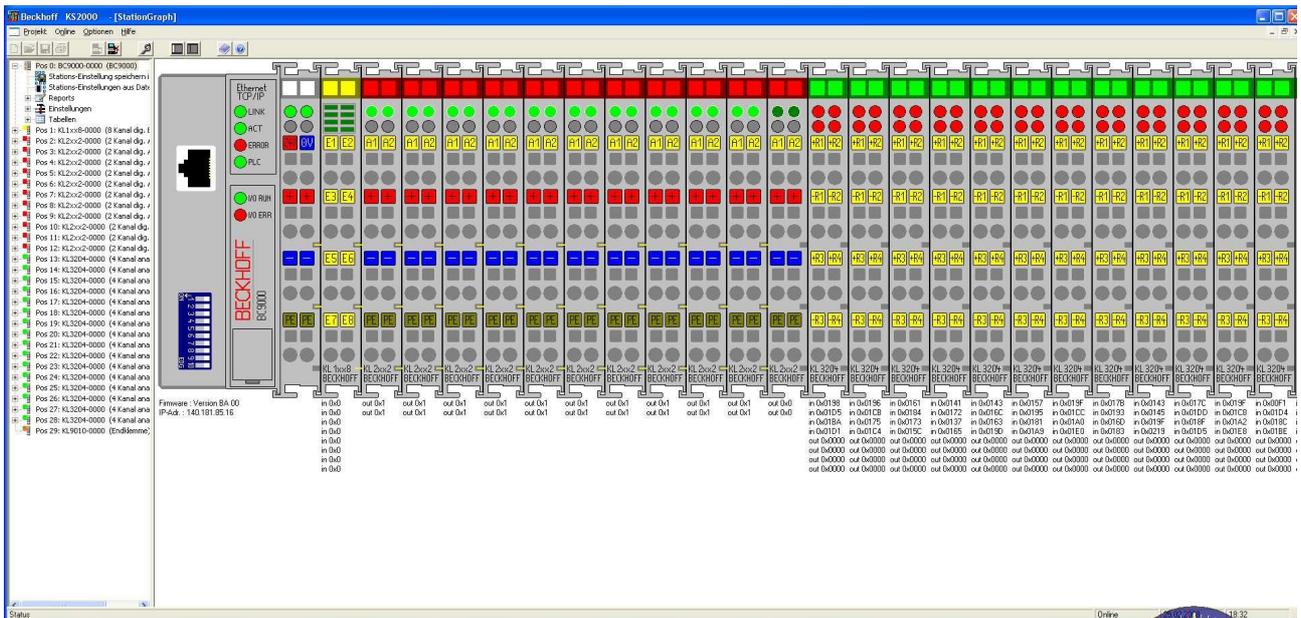
Die Konfiguration der Station geschieht mit der Software **KS2000**. Man kann sich über das Netzwerk über ADS (FopiBeck1: 140.181.85.16.1.1) verbinden:



oder mit einem speziellen seriellen Kabel (von Beckhoff) über Optionen -> seriell direkt mit der BC9000.

## 2.4.2 Stationsanzeige

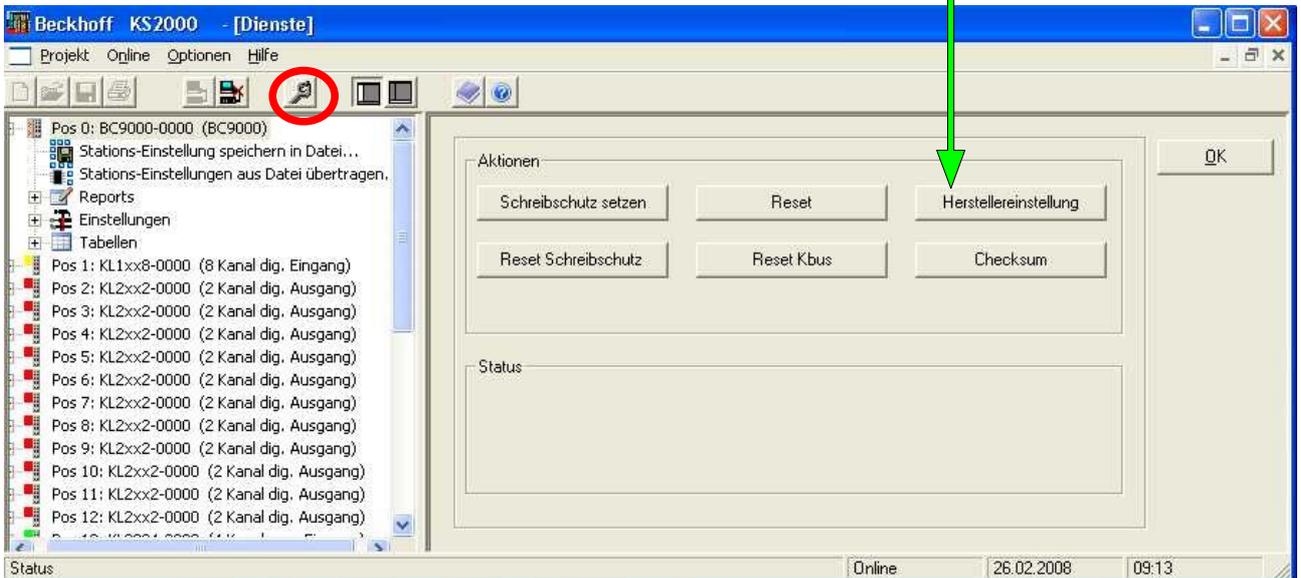
Nach dem Einloggen wird die Station angezeigt:



Unter den Klemmen sieht man auch die eingelesenen Daten (Temperaturen, binäre Werte).

## 2.4.3 Reset

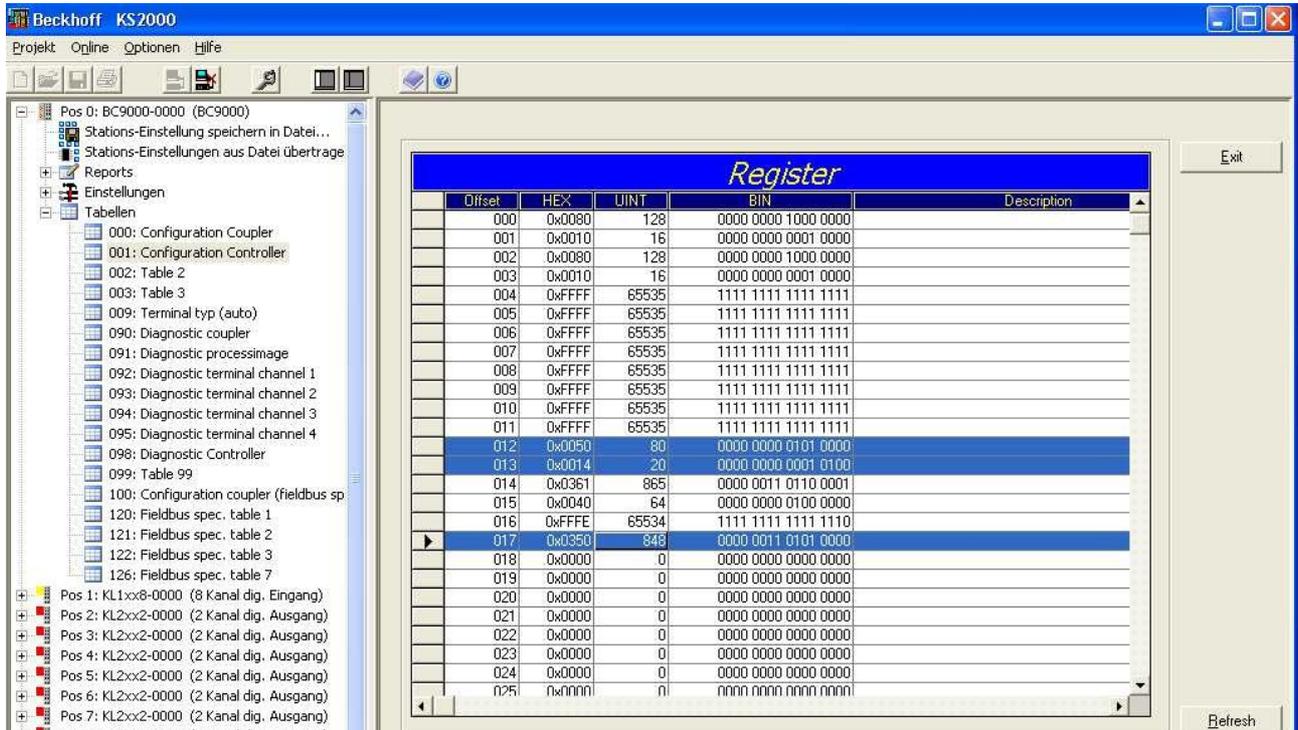
Falls die SPS nicht mehr funktioniert, kann die Station auf Werkseinstellung zurückgesetzt werden:



Danach ist es notwendig, die Einstellungen anzupassen und das Programm neu zu laden.

## 2.4.4 Einstellungen aktualisieren

Nach einem Rücksetzen auf die Werkseinstellungen müssen in der Tabelle 001 einige Werte verändert werden:



The screenshot shows the Beckhoff KS2000 software interface. On the left is a tree view of the configuration for 'Pos 0: BC9000-0000 (BC9000)'. The main window displays a table titled 'Register' with the following columns: Offset, HEX, UINT, BIN, and Description. The table contains 25 rows of data. Three rows are highlighted in blue: row 12 (Offset 012, HEX 0x0050, UINT 80), row 13 (Offset 013, HEX 0x0014, UINT 20), and row 17 (Offset 017, HEX 0x350, UINT 848). The 'Description' column is empty for all rows.

Offset	HEX	UINT	BIN	Description
000	0x0080	128	0000 0000 1000 0000	
001	0x0010	16	0000 0000 0001 0000	
002	0x0080	128	0000 0000 1000 0000	
003	0x0010	16	0000 0000 0001 0000	
004	0xFFFF	65535	1111 1111 1111 1111	
005	0xFFFF	65535	1111 1111 1111 1111	
006	0xFFFF	65535	1111 1111 1111 1111	
007	0xFFFF	65535	1111 1111 1111 1111	
008	0xFFFF	65535	1111 1111 1111 1111	
009	0xFFFF	65535	1111 1111 1111 1111	
010	0xFFFF	65535	1111 1111 1111 1111	
011	0xFFFF	65535	1111 1111 1111 1111	
012	0x0050	80	0000 0000 0101 0000	
013	0x0014	20	0000 0000 0001 0100	
014	0x0361	865	0000 0011 0110 0001	
015	0x0040	64	0000 0000 0100 0000	
016	0xFFFE	65534	1111 1111 1111 1110	
017	0x350	848	0000 0011 0101 0000	
018	0x0000	0	0000 0000 0000 0000	
019	0x0000	0	0000 0000 0000 0000	
020	0x0000	0	0000 0000 0000 0000	
021	0x0000	0	0000 0000 0000 0000	
022	0x0000	0	0000 0000 0000 0000	
023	0x0000	0	0000 0000 0000 0000	
024	0x0000	0	0000 0000 0000 0000	
025	0x0000	0	0000 0000 0000 0000	

Offset 12 muss von 20 auf 80 geändert werden.

Offset 13 muss von 10 auf 20 geändert werden.

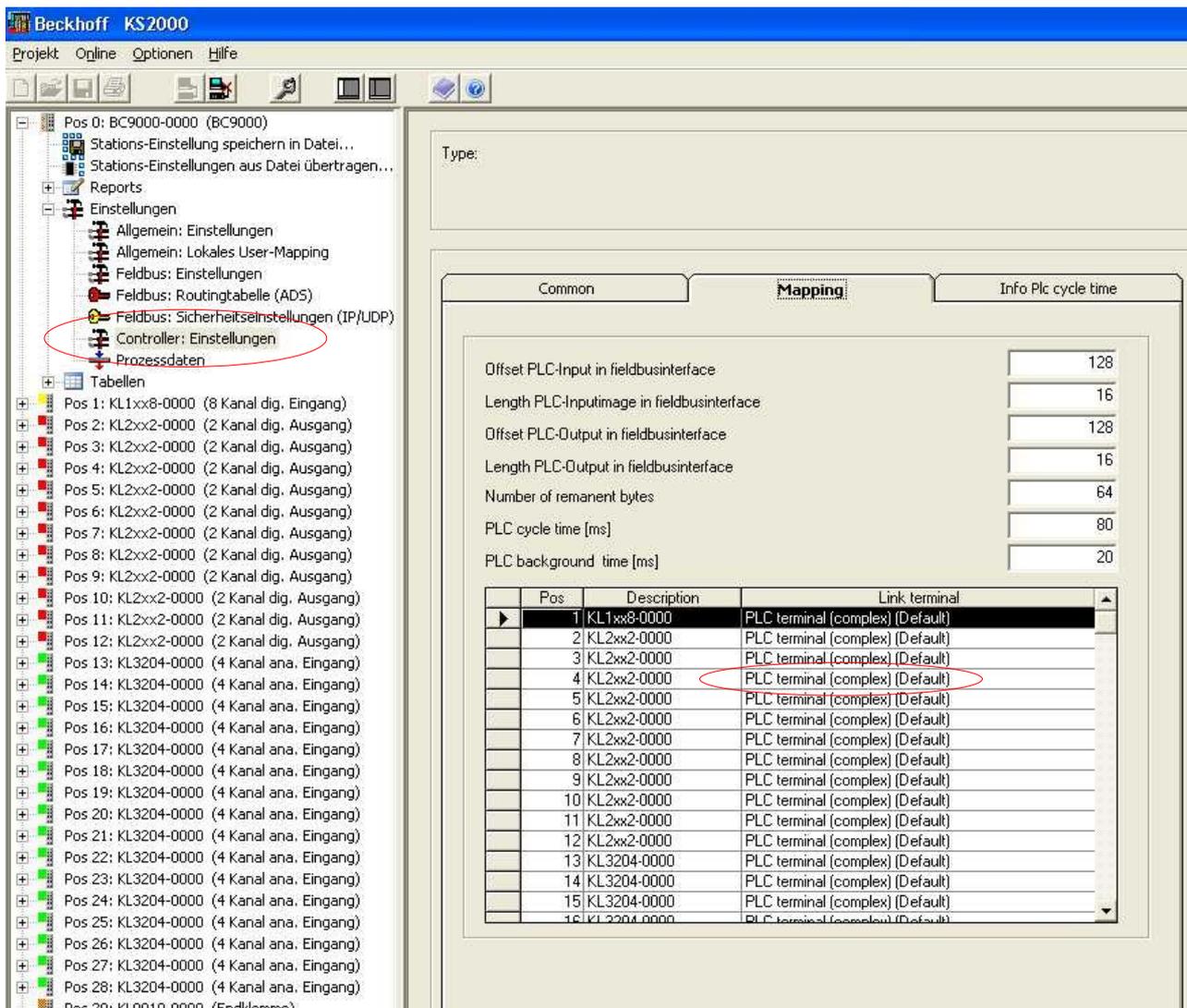
Offset 17 muss in der HEX-Spalte auf 0x350 geändert werden.

## 2.4.5 Mapping

Das Mapping der Klemmen (Speicherbereiche festlegen) hängt davon ab, ob die Klemmen an einer Buskoppler (BKxxxx) oder einem Buscontroller (BCxxxx) betrieben werden.

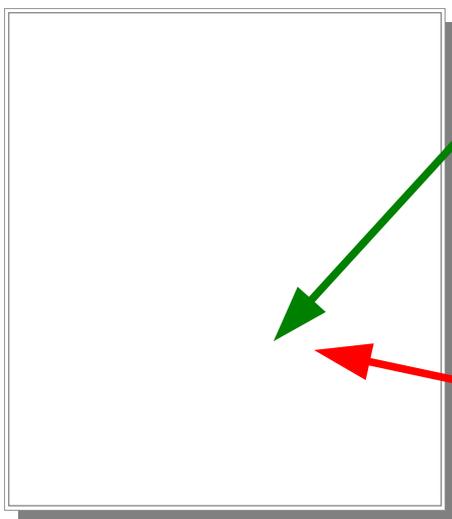
Wird ein Buscontroller (SPS) verwendet, müssen die Klemmen auf „PLC Terminal“ stehen. Dabei gibt es zwei Möglichkeiten „Komplex (default)“ oder „compact“.

Bei Fopi wird die Einstellung **PLC terminal (complex) (default)** benutzt, da mit dieser Einstellung auch der Boxenstatus übermittelt wird.



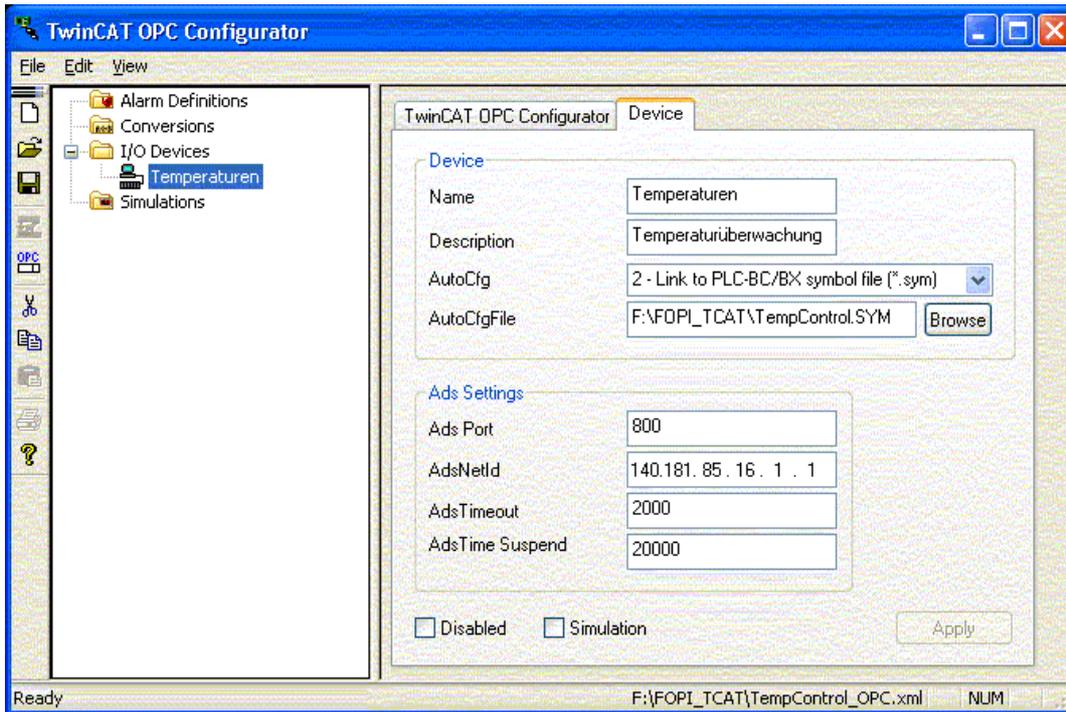
### 3 Inbetriebnahme

Das Einschalten der Netzteile geschieht über den grünen Knopf unterhalb der SPS.



Mit dem roten Knopf kann man die Netzteile ausschalten. Dabei werden zuerst die negativen Spannungen und kurz danach die positiven Spannungen abgeschaltet.

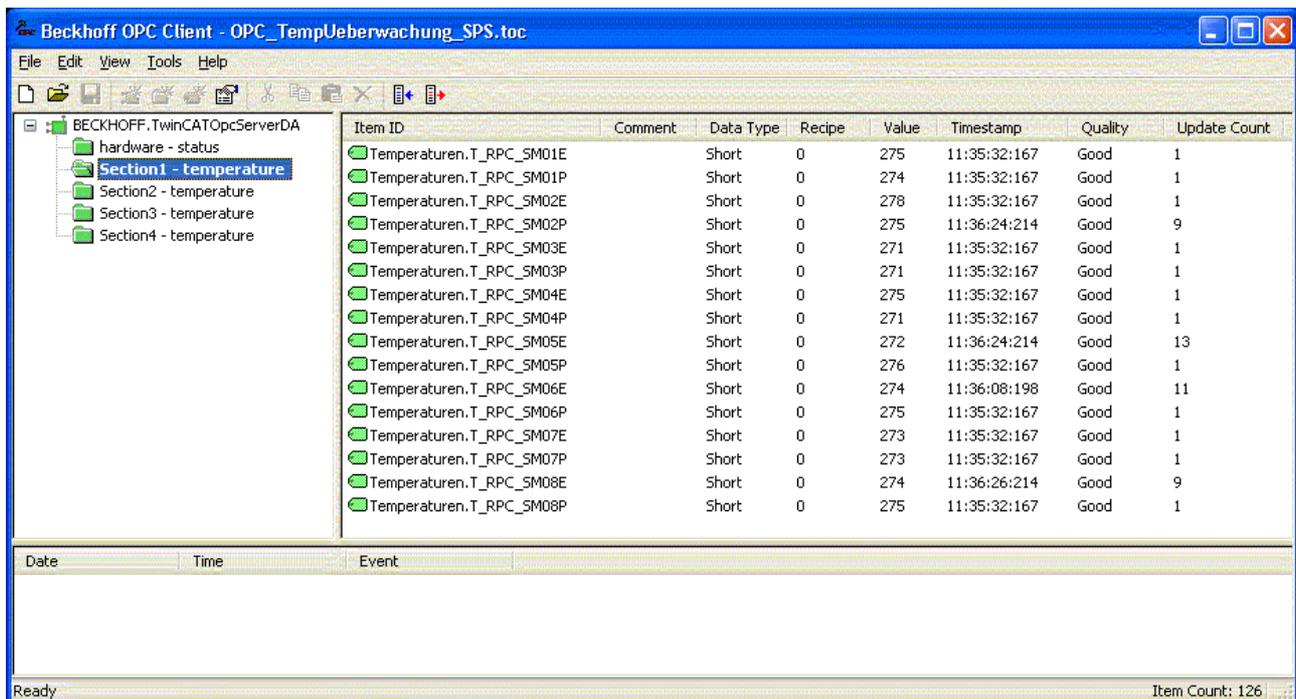
## 4 OPC-Konfiguration



Link über den Symbolfile: **TempControl.SYM**, ADS-Port: **800**

## 5 OPC Test Client

Mit dem TestClient können die Variablen ausgelesen werden. Es existiert schon ein File dafür (F:\FOPI\_TCAT\OPC\_TempUeberwachung\_SPS.toc):



Die Temperaturen werden in 1/10°C übergeben (Wert 275 = 27,5°C).

0001	PROGRAM MAIN
0002	VAR
0003	END_VAR
0001	(* HAUPTPROGRAMM *)
0002	
0003	Status(); (* Auslesen des Systemzustands *)
0004	Messen(); (* Vergleichen der Temp. mit Max. *)
0005	OPC_Transfer(); (* Speichern der Werte in RAM *)
0006	Schalten(); (* Ein- und Ausschalten der NT *)
0007	
0008	(*
0009	Test();
0010	*)
0011	

0001	PROGRAM Status
0002	VAR
0003	END_VAR
0001	<p>Systeminformationen des SPS Controllers auslesen</p> <p>Bit 0 gesetzt    Die Feldbus- und Klemmenbus-Statusinformationen sind vorhanden</p> <p>Bit 1 gesetzt    Der System-Tick ist vorhanden</p> <p>Bit 2 gesetzt    Die SPS-Zykluszeit Watchdog-Überwachung ist aktiviert</p> <div style="display: flex; align-items: center;"> <div style="border: 1px solid black; padding: 2px; margin-right: 10px;">GetSysInfo</div> <div style="margin-right: 10px;">—</div> <div style="margin-right: 10px;">FopiBeck1_SysInfo</div> </div> <p>TRUE—bDummy</p>
0002	<p>Klemmenbus-Status auslesen</p> <p>Bit 0 gesetzt    allgemeiner K-Bus Fehler</p> <p>Bit 1 gesetzt    allgemeiner Konfigurationsfehler</p> <p>Bit 4 gesetzt    SPS Zykluszeitüberwachung</p> <div style="display: flex; align-items: center;"> <div style="border: 1px solid black; padding: 2px; margin-right: 10px;">GetTBusStatus</div> <div style="margin-right: 10px;">—</div> <div style="margin-right: 10px;">FopiBeck1_BusStatus</div> </div> <p>TRUE—bDummy</p>

0001	PROGRAM Messen
0002	VAR
0003	iMaxTemp:                  INT;
0004	END_VAR
0001	iMaxTemp := 600;                  (* entspricht 60° C *)
0002	neu: 700 (am 26.02.2008 auf Wunsch von FOPI geändert)
0003	(* bei einer Fehlermeldung wird das Fehlerbit gesetzt *)
0004	IF (FopiBeck1_ChState_1_1 <> 0)
0005	OR (FopiBeck1_ChState_1_2 <> 0)
0006	OR (FopiBeck1_ChState_1_3 <> 0)
0007	OR (FopiBeck1_ChState_1_4 <> 0)
0008	
0009	OR (FopiBeck1_ChState_2_1 <> 0)
0010	OR (FopiBeck1_ChState_2_2 <> 0)
0011	OR (FopiBeck1_ChState_2_3 <> 0)
0012	OR (FopiBeck1_ChState_2_4 <> 0)
0013	
0014	OR (FopiBeck1_ChState_3_1 <> 0)
0015	OR (FopiBeck1_ChState_3_2 <> 0)
0016	OR (FopiBeck1_ChState_3_3 <> 0)
0017	OR (FopiBeck1_ChState_3_4 <> 0)
0018	
0019	OR (FopiBeck1_ChState_4_1 <> 0)
0020	OR (FopiBeck1_ChState_4_2 <> 0)
0021	OR (FopiBeck1_ChState_4_3 <> 0)
0022	OR (FopiBeck1_ChState_4_4 <> 0)
0023	
0024	OR (FopiBeck1_ChState_5_1 <> 0)
0025	OR (FopiBeck1_ChState_5_2 <> 0)
0026	OR (FopiBeck1_ChState_5_3 <> 0)
0027	OR (FopiBeck1_ChState_5_4 <> 0)
0028	
0029	OR (FopiBeck1_ChState_6_1 <> 0)
0030	OR (FopiBeck1_ChState_6_2 <> 0)
0031	OR (FopiBeck1_ChState_6_3 <> 0)
0032	OR (FopiBeck1_ChState_6_4 <> 0)
0033	
0034	OR (FopiBeck1_ChState_7_1 <> 0)
0035	OR (FopiBeck1_ChState_7_2 <> 0)
0036	OR (FopiBeck1_ChState_7_3 <> 0)
0037	OR (FopiBeck1_ChState_7_4 <> 0)
0038	
0039	OR (FopiBeck1_ChState_8_1 <> 0)
0040	OR (FopiBeck1_ChState_8_2 <> 0)
0041	OR (FopiBeck1_ChState_8_3 <> 0)
0042	OR (FopiBeck1_ChState_8_4 <> 0)
0043	
0044	OR (FopiBeck1_ChState_9_1 <> 0)
0045	OR (FopiBeck1_ChState_9_2 <> 0)
0046	OR (FopiBeck1_ChState_9_3 <> 0)
0047	OR (FopiBeck1_ChState_9_4 <> 0)
0048	
0049	OR (FopiBeck1_ChState_10_1 <> 0)
0050	OR (FopiBeck1_ChState_10_2 <> 0)
0051	OR (FopiBeck1_ChState_10_3 <> 0)
0052	OR (FopiBeck1_ChState_10_4 <> 0)
0053	
0054	OR (FopiBeck1_ChState_11_1 <> 0)
0055	OR (FopiBeck1_ChState_11_2 <> 0)
0056	OR (FopiBeck1_ChState_11_3 <> 0)
0057	OR (FopiBeck1_ChState_11_4 <> 0)
0058	
0059	OR (FopiBeck1_ChState_12_1 <> 0)
0060	OR (FopiBeck1_ChState_12_2 <> 0)
0061	OR (FopiBeck1_ChState_12_3 <> 0)
0062	OR (FopiBeck1_ChState_12_4 <> 0)
0063	
0064	OR (FopiBeck1_ChState_13_1 <> 0)
0065	OR (FopiBeck1_ChState_13_2 <> 0)
0066	OR (FopiBeck1_ChState_13_3 <> 0)
0067	OR (FopiBeck1_ChState_13_4 <> 0)
0068	
0069	OR (FopiBeck1_ChState_14_1 <> 0)
0070	OR (FopiBeck1_ChState_14_2 <> 0)
0071	OR (FopiBeck1_ChState_14_3 <> 0)

```

0072 OR (FopiBeck1_ChState_14_4 <> 0)
0073
0074 OR (FopiBeck1_ChState_15_1 <> 0)
0075 OR (FopiBeck1_ChState_15_2 <> 0)
0076 OR (FopiBeck1_ChState_15_3 <> 0)
0077 OR (FopiBeck1_ChState_15_4 <> 0)
0078
0079 OR (FopiBeck1_ChState_16_1 <> 0)
0080 OR (FopiBeck1_ChState_16_2 <> 0)
0081 OR (FopiBeck1_ChState_16_3 <> 0)
0082 OR (FopiBeck1_ChState_16_4 <> 0)
0083 THEN
0084     bKL3204Error:=TRUE;
0085 ELSE
0086     bKL3204Error:=FALSE;
0087 END_IF
0088
0089 (* bei Überschreiten einer beliebigen Übertemperatur
0090 wird das Übertemperatur-Bit gesetzt *)
0091 IF (T_RPC_SM01E > iMaxTemp) (* 1. Quadrant *)
0092     OR (T_RPC_SM01P > iMaxTemp)
0093     OR (T_RPC_SM02E > iMaxTemp)
0094     OR (T_RPC_SM02P > iMaxTemp)
0095     OR (T_RPC_SM03E > iMaxTemp)
0096     OR (T_RPC_SM03P > iMaxTemp)
0097     OR (T_RPC_SM04E > iMaxTemp)
0098     OR (T_RPC_SM04P > iMaxTemp)
0099     OR (T_RPC_SM05E > iMaxTemp)
0100     OR (T_RPC_SM05P > iMaxTemp)
0101     OR (T_RPC_SM06E > iMaxTemp)
0102     OR (T_RPC_SM06P > iMaxTemp)
0103     OR (T_RPC_SM07E > iMaxTemp)
0104     OR (T_RPC_SM07P > iMaxTemp)
0105     OR (T_RPC_SM08E > iMaxTemp)
0106
0107     OR (T_RPC_SM09E > iMaxTemp) (* 2. Quadrant *)
0108     OR (T_RPC_SM09P > iMaxTemp)
0109     OR (T_RPC_SM10E > iMaxTemp)
0110     OR (T_RPC_SM10P > iMaxTemp)
0111     OR (T_RPC_SM11E > iMaxTemp)
0112     OR (T_RPC_SM11P > iMaxTemp)
0113     OR (T_RPC_SM12E > iMaxTemp)
0114     OR (T_RPC_SM12P > iMaxTemp)
0115     OR (T_RPC_SM13E > iMaxTemp)
0116     OR (T_RPC_SM13P > iMaxTemp)
0117     OR (T_RPC_SM14E > iMaxTemp)
0118     OR (T_RPC_SM14P > iMaxTemp)
0119     OR (T_RPC_SM15E > iMaxTemp)
0120     OR (T_RPC_SM15P > iMaxTemp)
0121     OR (T_RPC_SM16E > iMaxTemp)
0122     OR (T_RPC_SM16P > iMaxTemp)
0123
0124     OR (T_RPC_SM17E > iMaxTemp) (* 3. Quadrant *)
0125     OR (T_RPC_SM17P > iMaxTemp)
0126     OR (T_RPC_SM19E > iMaxTemp)
0127     OR (T_RPC_SM19P > iMaxTemp)
0128     OR (T_RPC_SM20E > iMaxTemp)
0129     OR (T_RPC_SM20P > iMaxTemp)
0130     OR (T_RPC_SM21E > iMaxTemp)
0131     OR (T_RPC_SM21P > iMaxTemp)
0132     OR (T_RPC_SM22E > iMaxTemp)
0133     OR (T_RPC_SM22P > iMaxTemp)
0134     OR (T_RPC_SM23E > iMaxTemp)
0135     OR (T_RPC_SM23P > iMaxTemp)
0136     OR (T_RPC_SM24E > iMaxTemp)
0137     OR (T_RPC_SM24P > iMaxTemp)
0138
0139     OR (T_RPC_SM25E > iMaxTemp) (* 4. Quadrant *)
0140     OR (T_RPC_SM25P > iMaxTemp)
0141     OR (T_RPC_SM26E > iMaxTemp)
0142     OR (T_RPC_SM26P > iMaxTemp)
0143     OR (T_RPC_SM27E > iMaxTemp)
0144     OR (T_RPC_SM27P > iMaxTemp)
0145     OR (T_RPC_SM28E > iMaxTemp)
0146     OR (T_RPC_SM28P > iMaxTemp)

```

```
0147 OR (T_RPC_SM29E > iMaxTemp)
0148 OR (T_RPC_SM29P > iMaxTemp)
0149 OR (T_RPC_SM30E > iMaxTemp)
0150 OR (T_RPC_SM30P > iMaxTemp)
0151 OR (T_RPC_SM32E > iMaxTemp)
0152 OR (T_RPC_SM32P > iMaxTemp)
0153 THEN
0154     bUebertemperatur:=TRUE;
0155 ELSE
0156     bUebertemperatur:=FALSE;
0157 END_IF
0158
0159 bSTOP:= bKL3204Error OR bUebertemperatur OR StopTaste;
```

0001	PROGRAM OPC_Transfer
0002	VAR
0003	END_VAR
0001	(* Übergabe der Temperaturwerte an den Speicher *)
0002	T_RPC_SM01E:= in_T_RPC_SM01E;
0003	T_RPC_SM01P:= in_T_RPC_SM01P;
0004	T_RPC_SM02E:= in_T_RPC_SM02E;
0005	T_RPC_SM02P:= in_T_RPC_SM02P;
0006	
0007	T_RPC_SM03E:= in_T_RPC_SM03E;
0008	T_RPC_SM03P:= in_T_RPC_SM03P;
0009	T_RPC_SM04E:= in_T_RPC_SM04E;
0010	T_RPC_SM04P:= in_T_RPC_SM04P;
0011	
0012	T_RPC_SM05E:= in_T_RPC_SM05E;
0013	T_RPC_SM05P:= in_T_RPC_SM05P;
0014	T_RPC_SM06E:= in_T_RPC_SM06E;
0015	T_RPC_SM06P:= in_T_RPC_SM06P;
0016	
0017	T_RPC_SM07E:= in_T_RPC_SM07E;
0018	T_RPC_SM07P:= in_T_RPC_SM07P;
0019	T_RPC_SM08E:= in_T_RPC_SM08E;
0020	T_RPC_SM08P:= in_T_RPC_SM08P;
0021	
0022	T_RPC_SM09E:= in_T_RPC_SM09E;
0023	T_RPC_SM09P:= in_T_RPC_SM09P;
0024	T_RPC_SM10E:= in_T_RPC_SM10E;
0025	T_RPC_SM10P:= in_T_RPC_SM10P;
0026	
0027	T_RPC_SM11E:= in_T_RPC_SM11E;
0028	T_RPC_SM11P:= in_T_RPC_SM11P;
0029	T_RPC_SM12E:= in_T_RPC_SM12E;
0030	T_RPC_SM12P:= in_T_RPC_SM12P;
0031	
0032	T_RPC_SM13E:= in_T_RPC_SM13E;
0033	T_RPC_SM13P:= in_T_RPC_SM13P;
0034	T_RPC_SM14E:= in_T_RPC_SM14E;
0035	T_RPC_SM14P:= in_T_RPC_SM14P;
0036	
0037	T_RPC_SM15E:= in_T_RPC_SM15E;
0038	T_RPC_SM15P:= in_T_RPC_SM15P;
0039	T_RPC_SM16E:= in_T_RPC_SM16E;
0040	T_RPC_SM16P:= in_T_RPC_SM16P;
0041	
0042	T_RPC_SM17E:= in_T_RPC_SM17E;
0043	T_RPC_SM17P:= in_T_RPC_SM17P;
0044	T_RPC_SM19E:= in_T_RPC_SM19E;
0045	T_RPC_SM19P:= in_T_RPC_SM19P;
0046	
0047	T_RPC_SM20E:= in_T_RPC_SM20E;
0048	T_RPC_SM20P:= in_T_RPC_SM20P;
0049	T_RPC_SM21E:= in_T_RPC_SM21E;
0050	T_RPC_SM21P:= in_T_RPC_SM21P;
0051	
0052	T_RPC_SM22E:= in_T_RPC_SM22E;
0053	T_RPC_SM22P:= in_T_RPC_SM22P;
0054	T_RPC_SM23E:= in_T_RPC_SM23E;
0055	T_RPC_SM23P:= in_T_RPC_SM23P;
0056	
0057	T_RPC_SM24E:= in_T_RPC_SM24E;
0058	T_RPC_SM24P:= in_T_RPC_SM24P;
0059	T_RPC_SM25E:= in_T_RPC_SM25E;
0060	T_RPC_SM25P:= in_T_RPC_SM25P;
0061	
0062	T_RPC_SM26E:= in_T_RPC_SM26E;
0063	T_RPC_SM26P:= in_T_RPC_SM26P;
0064	T_RPC_SM27E:= in_T_RPC_SM27E;
0065	T_RPC_SM27P:= in_T_RPC_SM27P;
0066	
0067	T_RPC_SM28E:= in_T_RPC_SM28E;
0068	T_RPC_SM28P:= in_T_RPC_SM28P;
0069	T_RPC_SM29E:= in_T_RPC_SM29E;
0070	T_RPC_SM29P:= in_T_RPC_SM29P;
0071	
0072	T_RPC_SM30E:= in_T_RPC_SM30E;

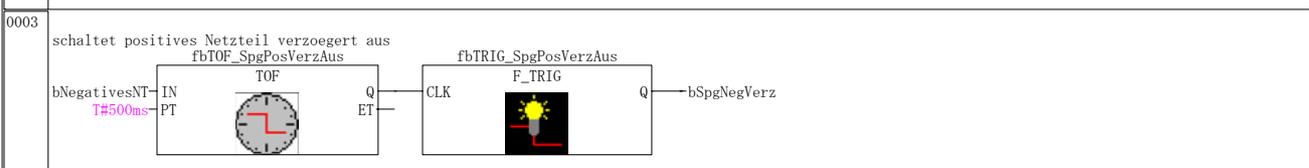
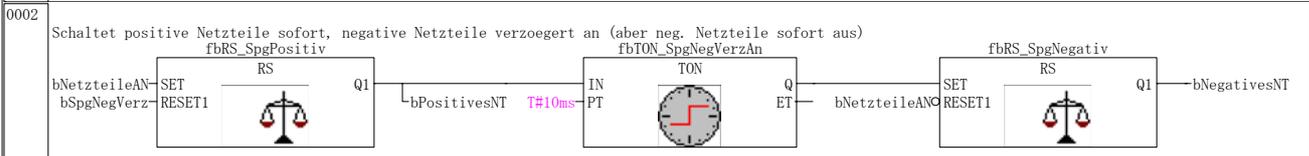
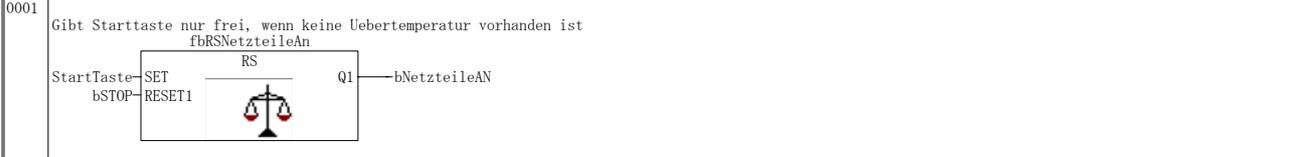
0073	T_RPC_SM30P:=	in_T_RPC_SM30P;
0074	T_RPC_SM32E:=	in_T_RPC_SM32E;
0075	T_RPC_SM32P:=	in_T_RPC_SM32P;
0076		
0077	T_HEL_DN:=	in_T_HEL_DN;
0078	T_K16_K2:=	in_T_K16_K2;
0079	T_HEL_UP:=	in_T_HEL_UP;
0080	T_K16_K4:=	in_T_K16_K4;
0081		
0082	(* Übergabe der Statussignale an den Speicher *)	
0083	FopiBeck1_ChState_1_1:=	in_ChState_1_1;
0084	FopiBeck1_ChState_1_2:=	in_ChState_1_2;
0085	FopiBeck1_ChState_1_3:=	in_ChState_1_3;
0086	FopiBeck1_ChState_1_4:=	in_ChState_1_4;
0087		
0088	FopiBeck1_ChState_2_1:=	in_ChState_2_1;
0089	FopiBeck1_ChState_2_2:=	in_ChState_2_2;
0090	FopiBeck1_ChState_2_3:=	in_ChState_2_3;
0091	FopiBeck1_ChState_2_4:=	in_ChState_2_4;
0092		
0093	FopiBeck1_ChState_3_1:=	in_ChState_3_1;
0094	FopiBeck1_ChState_3_2:=	in_ChState_3_2;
0095	FopiBeck1_ChState_3_3:=	in_ChState_3_3;
0096	FopiBeck1_ChState_3_4:=	in_ChState_3_4;
0097		
0098	FopiBeck1_ChState_4_1:=	in_ChState_4_1;
0099	FopiBeck1_ChState_4_2:=	in_ChState_4_2;
0100	FopiBeck1_ChState_4_3:=	in_ChState_4_3;
0101	FopiBeck1_ChState_4_4:=	in_ChState_4_4;
0102		
0103	FopiBeck1_ChState_5_1:=	in_ChState_5_1;
0104	FopiBeck1_ChState_5_2:=	in_ChState_5_2;
0105	FopiBeck1_ChState_5_3:=	in_ChState_5_3;
0106	FopiBeck1_ChState_5_4:=	in_ChState_5_4;
0107		
0108	FopiBeck1_ChState_6_1:=	in_ChState_6_1;
0109	FopiBeck1_ChState_6_2:=	in_ChState_6_2;
0110	FopiBeck1_ChState_6_3:=	in_ChState_6_3;
0111	FopiBeck1_ChState_6_4:=	in_ChState_6_4;
0112		
0113	FopiBeck1_ChState_7_1:=	in_ChState_7_1;
0114	FopiBeck1_ChState_7_2:=	in_ChState_7_2;
0115	FopiBeck1_ChState_7_3:=	in_ChState_7_3;
0116	FopiBeck1_ChState_7_4:=	in_ChState_7_4;
0117		
0118	FopiBeck1_ChState_8_1:=	in_ChState_8_1;
0119	FopiBeck1_ChState_8_2:=	in_ChState_8_2;
0120	FopiBeck1_ChState_8_3:=	in_ChState_8_3;
0121	FopiBeck1_ChState_8_4:=	in_ChState_8_4;
0122		
0123	FopiBeck1_ChState_9_1:=	in_ChState_9_1;
0124	FopiBeck1_ChState_9_2:=	in_ChState_9_2;
0125	FopiBeck1_ChState_9_3:=	in_ChState_9_3;
0126	FopiBeck1_ChState_9_4:=	in_ChState_9_4;
0127		
0128	FopiBeck1_ChState_10_1:=	in_ChState_10_1;
0129	FopiBeck1_ChState_10_2:=	in_ChState_10_2;
0130	FopiBeck1_ChState_10_3:=	in_ChState_10_3;
0131	FopiBeck1_ChState_10_4:=	in_ChState_10_4;
0132		
0133	FopiBeck1_ChState_11_1:=	in_ChState_11_1;
0134	FopiBeck1_ChState_11_2:=	in_ChState_11_2;
0135	FopiBeck1_ChState_11_3:=	in_ChState_11_3;
0136	FopiBeck1_ChState_11_4:=	in_ChState_11_4;
0137		
0138	FopiBeck1_ChState_12_1:=	in_ChState_12_1;
0139	FopiBeck1_ChState_12_2:=	in_ChState_12_2;
0140	FopiBeck1_ChState_12_3:=	in_ChState_12_3;
0141	FopiBeck1_ChState_12_4:=	in_ChState_12_4;
0142		
0143	FopiBeck1_ChState_13_1:=	in_ChState_13_1;
0144	FopiBeck1_ChState_13_2:=	in_ChState_13_2;
0145	FopiBeck1_ChState_13_3:=	in_ChState_13_3;
0146	FopiBeck1_ChState_13_4:=	in_ChState_13_4;
0147		

0148	FopiBeck1_ChState_14_1:=	in_ChState_14_1;
0149	FopiBeck1_ChState_14_2:=	in_ChState_14_2;
0150	FopiBeck1_ChState_14_3:=	in_ChState_14_3;
0151	FopiBeck1_ChState_14_4:=	in_ChState_14_4;
0152		
0153	FopiBeck1_ChState_15_1:=	in_ChState_15_1;
0154	FopiBeck1_ChState_15_2:=	in_ChState_15_2;
0155	FopiBeck1_ChState_15_3:=	in_ChState_15_3;
0156	FopiBeck1_ChState_15_4:=	in_ChState_15_4;
0157		
0158	FopiBeck1_ChState_16_1:=	in_ChState_16_1;
0159	FopiBeck1_ChState_16_2:=	in_ChState_16_2;
0160	FopiBeck1_ChState_16_3:=	in_ChState_16_3;
0161	FopiBeck1_ChState_16_4:=	in_ChState_16_4;

```

0001 PROGRAM Schalten
0002 VAR
0003     fbRSNetzteileAn:    RS;
0004     fbRS_SpgPositiv:   RS;
0005     fbRS_SpgNegativ:   RS;
0006     fbTON_SpgNegVerzAn: TON;
0007     fbTOF_SpgPosVerzAus: TOF;
0008     fbTRIG_SpgPosVerzAus: F_TRIG;
0009
0010     bSpgNegVerz:        BOOL;
0011     bPositivesNT:       BOOL;
0012     bNegativesNT:      BOOL;
0013 END_VAR

```



0001	VAR_GLOBAL		
0002	(* Temperaturwerte *)		
0003	(* 1. Quadrant *)		
0004	in_T_RPC_SM01E	AT %IB2:	INT; (* K1_KL3204_1_Daten_Ein *)
0005	in_T_RPC_SM01P	AT %IB6:	INT; (* K1_KL3204_2_Daten_Ein *)
0006	in_T_RPC_SM02E	AT %IB10:	INT; (* K1_KL3204_3_Daten_Ein *)
0007	in_T_RPC_SM02P	AT %IB14:	INT; (* K1_KL3204_4_Daten_Ein *)
0008			
0009	in_T_RPC_SM03E	AT %IB18:	INT; (* K2_KL3204_1_Daten_Ein *)
0010	in_T_RPC_SM03P	AT %IB22:	INT; (* : *)
0011	in_T_RPC_SM04E	AT %IB26:	INT; (* : *)
0012	in_T_RPC_SM04P	AT %IB30:	INT; (* : *)
0013			
0014	in_T_RPC_SM05E	AT %IB34:	INT; (* K3_KL3204_1_Daten_Ein *)
0015	in_T_RPC_SM05P	AT %IB38:	INT;
0016	in_T_RPC_SM06E	AT %IB42:	INT;
0017	in_T_RPC_SM06P	AT %IB46:	INT;
0018			
0019	in_T_RPC_SM07E	AT %IB50:	INT; (* K4_KL3204_1_Daten_Ein *)
0020	in_T_RPC_SM07P	AT %IB54:	INT;
0021	in_T_RPC_SM08E	AT %IB58:	INT;
0022	in_T_RPC_SM08P	AT %IB62:	INT;
0023			
0024	(* 2. Quadrant *)		
0025	in_T_RPC_SM09E	AT %IB66:	INT; (* K5_KL3204_1_Daten_Ein *)
0026	in_T_RPC_SM09P	AT %IB70:	INT;
0027	in_T_RPC_SM10E	AT %IB74:	INT;
0028	in_T_RPC_SM10P	AT %IB78:	INT;
0029			
0030	in_T_RPC_SM11E	AT %IB82:	INT; (* K6_KL3204_1_Daten_Ein *)
0031	in_T_RPC_SM11P	AT %IB86:	INT;
0032	in_T_RPC_SM12E	AT %IB90:	INT;
0033	in_T_RPC_SM12P	AT %IB94:	INT;
0034			
0035	in_T_RPC_SM13E	AT %IB98:	INT; (* K7_KL3204_1_Daten_Ein *)
0036	in_T_RPC_SM13P	AT %IB102:	INT;
0037	in_T_RPC_SM14E	AT %IB106:	INT;
0038	in_T_RPC_SM14P	AT %IB110:	INT;
0039			
0040	in_T_RPC_SM15E	AT %IB114:	INT; (* K8_KL3204_1_Daten_Ein *)
0041	in_T_RPC_SM15P	AT %IB118:	INT;
0042	in_T_RPC_SM16E	AT %IB122:	INT;
0043	in_T_RPC_SM16P	AT %IB126:	INT;
0044			
0045	(* 3. Quadrant *)		
0046	in_T_RPC_SM17E	AT %IB130:	INT; (* K9_KL3204_1_Daten_Ein *)
0047	in_T_RPC_SM17P	AT %IB134:	INT;
0048	in_T_RPC_SM19E	AT %IB138:	INT;
0049	in_T_RPC_SM19P	AT %IB142:	INT;
0050			
0051	in_T_RPC_SM20E	AT %IB146:	INT; (* K10_KL3204_1_Daten_Ein *)
0052	in_T_RPC_SM20P	AT %IB150:	INT;
0053	in_T_RPC_SM21E	AT %IB154:	INT;
0054	in_T_RPC_SM21P	AT %IB158:	INT;
0055			
0056	in_T_RPC_SM22E	AT %IB162:	INT; (* K11_KL3204_1_Daten_Ein *)
0057	in_T_RPC_SM22P	AT %IB166:	INT;
0058	in_T_RPC_SM23E	AT %IB170:	INT;
0059	in_T_RPC_SM23P	AT %IB174:	INT;
0060			
0061	in_T_RPC_SM24E	AT %IB178:	INT; (* K12_KL3204_1_Daten_Ein *)
0062	in_T_RPC_SM24P	AT %IB182:	INT;
0063			
0064	(* 4. Quadrant *)		
0065	in_T_RPC_SM25E	AT %IB186:	INT;
0066	in_T_RPC_SM25P	AT %IB190:	INT;
0067			
0068	in_T_RPC_SM26E	AT %IB194:	INT; (* K13_KL3204_1_Daten_Ein *)
0069	in_T_RPC_SM26P	AT %IB198:	INT;
0070	in_T_RPC_SM27E	AT %IB202:	INT;
0071	in_T_RPC_SM27P	AT %IB206:	INT;
0072			
0073	in_T_RPC_SM28E	AT %IB210:	INT; (* K14_KL3204_1_Daten_Ein *)
0074	in_T_RPC_SM28P	AT %IB214:	INT;
0075	in_T_RPC_SM29E	AT %IB218:	INT;

0076	in_T_RPC_SM29P	AT %IB222:	INT;
0077			
0078	in_T_RPC_SM30E	AT %IB226:	INT; (* K15_KL3204_1_Daten_Ein *)
0079	in_T_RPC_SM30P	AT %IB230:	INT;
0080	in_T_RPC_SM32E	AT %IB234:	INT;
0081	in_T_RPC_SM32P	AT %IB238:	INT;
0082			
0083	(* extern *)		
0084	in_T_HEL_DN	AT %IB242:	INT; (* K16_KL3204_1_Daten_Ein *)
0085	in_T_K16_K2	AT %IB246:	INT;
0086	in_T_HEL_UP	AT %IB250:	INT;
0087	in_T_K16_K4	AT %IB254:	INT;
0088			
0089	(* Klemmenstatus *)		
0090	in_ChState_1_1	AT %IB0: USINT;	(* K1_KL3204_1_Status *)
0091	in_ChState_1_2	AT %IB4: USINT;	(* K1_KL3204_2_Status *)
0092	in_ChState_1_3	AT %IB8: USINT;	(* K1_KL3204_3_Status *)
0093	in_ChState_1_4	AT %IB12: USINT;	(* K1_KL3204_4_Status *)
0094			
0095	in_ChState_2_1	AT %IB16: USINT;	(* K2_KL3204_1_Status *)
0096	in_ChState_2_2	AT %IB20: USINT;	(* : *)
0097	in_ChState_2_3	AT %IB24: USINT;	(* : *)
0098	in_ChState_2_4	AT %IB28: USINT;	(* : *)
0099			
0100	in_ChState_3_1	AT %IB32: USINT;	
0101	in_ChState_3_2	AT %IB36: USINT;	
0102	in_ChState_3_3	AT %IB40: USINT;	
0103	in_ChState_3_4	AT %IB44: USINT;	
0104			
0105	in_ChState_4_1	AT %IB48: USINT;	
0106	in_ChState_4_2	AT %IB52: USINT;	
0107	in_ChState_4_3	AT %IB56: USINT;	
0108	in_ChState_4_4	AT %IB60: USINT;	
0109			
0110	in_ChState_5_1	AT %IB64: USINT;	
0111	in_ChState_5_2	AT %IB68: USINT;	
0112	in_ChState_5_3	AT %IB72: USINT;	
0113	in_ChState_5_4	AT %IB76: USINT;	
0114			
0115	in_ChState_6_1	AT %IB80: USINT;	
0116	in_ChState_6_2	AT %IB84: USINT;	
0117	in_ChState_6_3	AT %IB88: USINT;	
0118	in_ChState_6_4	AT %IB92: USINT;	
0119			
0120	in_ChState_7_1	AT %IB96: USINT;	
0121	in_ChState_7_2	AT %IB100: USINT;	
0122	in_ChState_7_3	AT %IB104: USINT;	
0123	in_ChState_7_4	AT %IB108: USINT;	
0124			
0125	in_ChState_8_1	AT %IB112: USINT;	
0126	in_ChState_8_2	AT %IB116: USINT;	
0127	in_ChState_8_3	AT %IB120: USINT;	
0128	in_ChState_8_4	AT %IB124: USINT;	
0129			
0130	in_ChState_9_1	AT %IB128: USINT;	
0131	in_ChState_9_2	AT %IB132: USINT;	
0132	in_ChState_9_3	AT %IB136: USINT;	
0133	in_ChState_9_4	AT %IB140: USINT;	
0134			
0135	in_ChState_10_1	AT %IB144: USINT;	
0136	in_ChState_10_2	AT %IB148: USINT;	
0137	in_ChState_10_3	AT %IB152: USINT;	
0138	in_ChState_10_4	AT %IB156: USINT;	
0139			
0140	in_ChState_11_1	AT %IB160: USINT;	
0141	in_ChState_11_2	AT %IB164: USINT;	
0142	in_ChState_11_3	AT %IB168: USINT;	
0143	in_ChState_11_4	AT %IB172: USINT;	
0144			
0145	in_ChState_12_1	AT %IB176: USINT;	
0146	in_ChState_12_2	AT %IB180: USINT;	
0147	in_ChState_12_3	AT %IB184: USINT;	
0148	in_ChState_12_4	AT %IB188: USINT;	
0149			
0150	in_ChState_13_1	AT %IB192: USINT;	

0151	in_ChState_13_2	AT %IB196:	USINT;
0152	in_ChState_13_3	AT %IB200:	USINT;
0153	in_ChState_13_4	AT %IB204:	USINT;
0154			
0155	in_ChState_14_1	AT %IB208:	USINT;
0156	in_ChState_14_2	AT %IB212:	USINT;
0157	in_ChState_14_3	AT %IB216:	USINT;
0158	in_ChState_14_4	AT %IB220:	USINT;
0159			
0160	in_ChState_15_1	AT %IB224:	USINT;
0161	in_ChState_15_2	AT %IB228:	USINT;
0162	in_ChState_15_3	AT %IB232:	USINT;
0163	in_ChState_15_4	AT %IB236:	USINT;
0164			
0165	in_ChState_16_1	AT %IB240:	USINT;
0166	in_ChState_16_2	AT %IB244:	USINT;
0167	in_ChState_16_3	AT %IB248:	USINT;
0168	in_ChState_16_4	AT %IB252:	USINT;
0169			
0170	StartTaste	AT %IX256.0:	BOOL;
0171	StopTaste	AT %IX256.1:	BOOL;
0172	END_VAR		

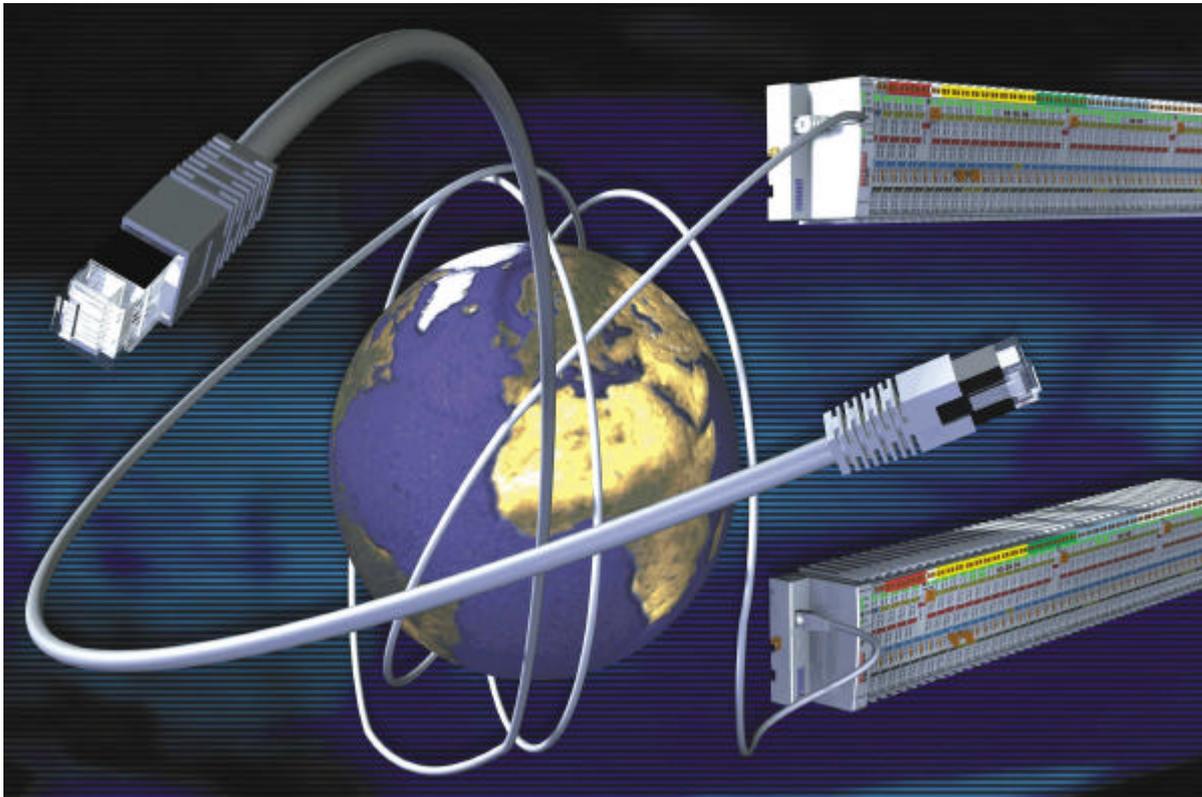
```
0001 VAR_GLOBAL
0002 (* Interlock-Relais für Netzteile *)
0003 Sect1_p5V_VV AT %QX256.0: BOOL:=FALSE; (* Out1_KL2012_1_Ausgang *)
0004 Sect1_m5V_VV AT %QX256.1: BOOL:=FALSE; (* Out1_KL2012_2_Ausgang *)
0005 Sect1_m5V_Analog AT %QX256.2: BOOL:=FALSE; (* Out2_KL2012_1_Ausgang *)
0006 Sect1_p6V AT %QX256.3: BOOL:=FALSE;
0007 Sect1_p5V_Ana_Dig AT %QX256.4: BOOL:=FALSE; (* Out3_KL2012_1_Ausgang *)
0008
0009 Sect2_p5V_VV AT %QX256.5: BOOL:=FALSE;
0010 Sect2_m5V_VV AT %QX256.6: BOOL:=FALSE; (* Out4_KL2012_1_Ausgang *)
0011 Sect2_m5V_Analog AT %QX256.7: BOOL:=FALSE;
0012 Sect2_p6V AT %QX257.0: BOOL:=FALSE; (* Out5_KL2012_1_Ausgang *)
0013 Sect2_p5V_Ana_Dig AT %QX257.1: BOOL:=FALSE;
0014
0015 Sect3_p5V_VV AT %QX257.2: BOOL:=FALSE; (* Out6_KL2012_1_Ausgang *)
0016 Sect3_m5V_VV AT %QX257.3: BOOL:=FALSE;
0017 Sect3_m5V_Analog AT %QX257.4: BOOL:=FALSE; (* Out7_KL2012_1_Ausgang *)
0018 Sect3_p6V AT %QX257.5: BOOL:=FALSE;
0019 Sect3_p5V_Ana_Dig AT %QX257.6: BOOL:=FALSE; (* Out8_KL2012_1_Ausgang *)
0020
0021 Sect4_p5V_VV AT %QX257.7: BOOL:=FALSE;
0022 Sect4_m5V_VV AT %QX258.0: BOOL:=FALSE; (* Out9_KL2012_1_Ausgang *)
0023 Sect4_m5V_Analog AT %QX258.1: BOOL:=FALSE;
0024 Sect4_p6V AT %QX258.2: BOOL:=FALSE; (* Out10_KL2012_1_Ausgang*)
0025 Sect4_p5V_Ana_Dig AT %QX258.3: BOOL:=FALSE;
0026
0027 Out11_KL2012_1_Output AT %QX258.4: BOOL;
0028 Out11_KL2012_2_Output AT %QX258.5: BOOL;
0029
0030 END_VAR
```

0001	VAR_GLOBAL		
0002	(* Hilfs-Variablen *)		
0003	bKL3204Error:	BOOL;	
0004	bUebertemperatur:	BOOL;	
0005	bNetzteileAN:	BOOL;	
0006	bSTOP:	BOOL;	
0007			
0008	(* === für OPC-Transfer === *)		
0009	(* Temperaturen *)		
0010	(* 1. Quadrant *)		
0011	T_RPC_SM01E	AT %MB0: INT;	(* K1_KL3204_Kanal1 *)
0012	T_RPC_SM01P	AT %MB2: INT;	(* K1_KL3204_Kanal2 *)
0013	T_RPC_SM02E	AT %MB4: INT;	(* K1_KL3204_Kanal3 *)
0014	T_RPC_SM02P	AT %MB6: INT;	(* K1_KL3204_Kanal4 *)
0015			
0016	T_RPC_SM03E	AT %MB8: INT;	(* K2_KL3204_Kanal1 *)
0017	T_RPC_SM03P	AT %MB10: INT;	
0018	T_RPC_SM04E	AT %MB12: INT;	
0019	T_RPC_SM04P	AT %MB14: INT;	
0020			
0021	T_RPC_SM05E	AT %MB16: INT;	(* K3_KL3204_Kanal *)
0022	T_RPC_SM05P	AT %MB18: INT;	
0023	T_RPC_SM06E	AT %MB20: INT;	
0024	T_RPC_SM06P	AT %MB22: INT;	
0025			
0026	T_RPC_SM07E	AT %MB24: INT;	(* K4_KL3204_Kanal *)
0027	T_RPC_SM07P	AT %MB26: INT;	
0028	T_RPC_SM08E	AT %MB28: INT;	
0029	T_RPC_SM08P	AT %MB30: INT;	
0030			
0031	(* 2. Quadrant *)		
0032	T_RPC_SM09E	AT %MB32: INT;	(* K5_KL3204_Kanal *)
0033	T_RPC_SM09P	AT %MB34: INT;	
0034	T_RPC_SM10E	AT %MB36: INT;	
0035	T_RPC_SM10P	AT %MB38: INT;	
0036			
0037	T_RPC_SM11E	AT %MB40: INT;	(* K6_KL3204_Kanal *)
0038	T_RPC_SM11P	AT %MB42: INT;	
0039	T_RPC_SM12E	AT %MB44: INT;	
0040	T_RPC_SM12P	AT %MB46: INT;	
0041			
0042	T_RPC_SM13E	AT %MB48: INT;	(* K7_KL3204_Kanal *)
0043	T_RPC_SM13P	AT %MB50: INT;	
0044	T_RPC_SM14E	AT %MB52: INT;	
0045	T_RPC_SM14P	AT %MB54: INT;	
0046			
0047	T_RPC_SM15E	AT %MB56: INT;	(* K8_KL3204_Kanal *)
0048	T_RPC_SM15P	AT %MB58: INT;	
0049	T_RPC_SM16E	AT %MB60: INT;	
0050	T_RPC_SM16P	AT %MB62: INT;	
0051			
0052	(* 3. Quadrant *)		
0053	T_RPC_SM17E	AT %MB64: INT;	(* K9_KL3204_Kanal *)
0054	T_RPC_SM17P	AT %MB66: INT;	
0055	T_RPC_SM19E	AT %MB68: INT;	
0056	T_RPC_SM19P	AT %MB70: INT;	
0057			
0058	T_RPC_SM20E	AT %MB72: INT;	(* K10_KL3204_Kanal *)
0059	T_RPC_SM20P	AT %MB74: INT;	
0060	T_RPC_SM21E	AT %MB76: INT;	
0061	T_RPC_SM21P	AT %MB78: INT;	
0062			
0063	T_RPC_SM22E	AT %MB80: INT;	(* K11_KL3204_Kanal *)
0064	T_RPC_SM22P	AT %MB82: INT;	
0065	T_RPC_SM23E	AT %MB84: INT;	
0066	T_RPC_SM23P	AT %MB86: INT;	
0067			
0068	T_RPC_SM24E	AT %MB88: INT;	(* K12_KL3204_Kanal *)
0069	T_RPC_SM24P	AT %MB90: INT;	
0070			
0071	(* 4. Quadrant *)		
0072	T_RPC_SM25E	AT %MB92: INT;	
0073	T_RPC_SM25P	AT %MB94: INT;	
0074			
0075	T_RPC_SM26E	AT %MB96: INT;	(* K13_KL3204_Kanal *)

0076	T_RPC_SM26P	AT %MB98:	INT;
0077	T_RPC_SM27E	AT %MB100:	INT;
0078	T_RPC_SM27P	AT %MB102:	INT;
0079			
0080	T_RPC_SM28E	AT %MB104:	INT; (* K14_KL3204_Kanal *)
0081	T_RPC_SM28P	AT %MB106:	INT;
0082	T_RPC_SM29E	AT %MB108:	INT;
0083	T_RPC_SM29P	AT %MB110:	INT;
0084			
0085	T_RPC_SM30E	AT %MB112:	INT; (* K15_KL3204_Kanal *)
0086	T_RPC_SM30P	AT %MB114:	INT;
0087	T_RPC_SM32E	AT %MB116:	INT;
0088	T_RPC_SM32P	AT %MB118:	INT;
0089			
0090	(* extern *)		
0091	T_HEL_DN	AT %MB120:	INT; (* K16_KL3204_Kanal *)
0092	T_K16_K2	AT %MB122:	INT;
0093	T_HEL_UP	AT %MB124:	INT;
0094	T_K16_K4	AT %MB126:	INT;
0095			
0096	(* Klemmenstatus *)		
0097	FopiBeck1_ChState_1_1 AT	%MB128:	USINT;
0098	FopiBeck1_ChState_1_2 AT	%MB129:	USINT;
0099	FopiBeck1_ChState_1_3 AT	%MB130:	USINT;
0100	FopiBeck1_ChState_1_4 AT	%MB131:	USINT;
0101			
0102	FopiBeck1_ChState_2_1 AT	%MB132:	USINT;
0103	FopiBeck1_ChState_2_2 AT	%MB133:	USINT;
0104	FopiBeck1_ChState_2_3 AT	%MB134:	USINT;
0105	FopiBeck1_ChState_2_4 AT	%MB135:	USINT;
0106			
0107	FopiBeck1_ChState_3_1 AT	%MB136:	USINT;
0108	FopiBeck1_ChState_3_2 AT	%MB137:	USINT;
0109	FopiBeck1_ChState_3_3 AT	%MB138:	USINT;
0110	FopiBeck1_ChState_3_4 AT	%MB139:	USINT;
0111			
0112	FopiBeck1_ChState_4_1 AT	%MB140:	USINT;
0113	FopiBeck1_ChState_4_2 AT	%MB141:	USINT;
0114	FopiBeck1_ChState_4_3 AT	%MB142:	USINT;
0115	FopiBeck1_ChState_4_4 AT	%MB143:	USINT;
0116			
0117	FopiBeck1_ChState_5_1 AT	%MB144:	USINT;
0118	FopiBeck1_ChState_5_2 AT	%MB145:	USINT;
0119	FopiBeck1_ChState_5_3 AT	%MB146:	USINT;
0120	FopiBeck1_ChState_5_4 AT	%MB147:	USINT;
0121			
0122	FopiBeck1_ChState_6_1 AT	%MB148:	USINT;
0123	FopiBeck1_ChState_6_2 AT	%MB149:	USINT;
0124	FopiBeck1_ChState_6_3 AT	%MB150:	USINT;
0125	FopiBeck1_ChState_6_4 AT	%MB151:	USINT;
0126			
0127	FopiBeck1_ChState_7_1 AT	%MB152:	USINT;
0128	FopiBeck1_ChState_7_2 AT	%MB153:	USINT;
0129	FopiBeck1_ChState_7_3 AT	%MB154:	USINT;
0130	FopiBeck1_ChState_7_4 AT	%MB155:	USINT;
0131			
0132	FopiBeck1_ChState_8_1 AT	%MB156:	USINT;
0133	FopiBeck1_ChState_8_2 AT	%MB157:	USINT;
0134	FopiBeck1_ChState_8_3 AT	%MB158:	USINT;
0135	FopiBeck1_ChState_8_4 AT	%MB159:	USINT;
0136			
0137	FopiBeck1_ChState_9_1 AT	%MB160:	USINT;
0138	FopiBeck1_ChState_9_2 AT	%MB161:	USINT;
0139	FopiBeck1_ChState_9_3 AT	%MB162:	USINT;
0140	FopiBeck1_ChState_9_4 AT	%MB163:	USINT;
0141			
0142	FopiBeck1_ChState_10_1	AT %MB164:	USINT;
0143	FopiBeck1_ChState_10_2	AT %MB165:	USINT;
0144	FopiBeck1_ChState_10_3	AT %MB166:	USINT;
0145	FopiBeck1_ChState_10_4	AT %MB167:	USINT;
0146			
0147	FopiBeck1_ChState_11_1	AT %MB168:	USINT;
0148	FopiBeck1_ChState_11_2	AT %MB169:	USINT;
0149	FopiBeck1_ChState_11_3	AT %MB170:	USINT;
0150	FopiBeck1_ChState_11_4	AT %MB171:	USINT;

0151				
0152	FopiBeck1_ChState_12_1	AT	%MB172:	USINT;
0153	FopiBeck1_ChState_12_2	AT	%MB173:	USINT;
0154	FopiBeck1_ChState_12_3	AT	%MB174:	USINT;
0155	FopiBeck1_ChState_12_4	AT	%MB175:	USINT;
0156				
0157	FopiBeck1_ChState_13_1	AT	%MB176:	USINT;
0158	FopiBeck1_ChState_13_2	AT	%MB177:	USINT;
0159	FopiBeck1_ChState_13_3	AT	%MB178:	USINT;
0160	FopiBeck1_ChState_13_4	AT	%MB179:	USINT;
0161				
0162	FopiBeck1_ChState_14_1	AT	%MB180:	USINT;
0163	FopiBeck1_ChState_14_2	AT	%MB181:	USINT;
0164	FopiBeck1_ChState_14_3	AT	%MB182:	USINT;
0165	FopiBeck1_ChState_14_4	AT	%MB183:	USINT;
0166				
0167	FopiBeck1_ChState_15_1	AT	%MB184:	USINT;
0168	FopiBeck1_ChState_15_2	AT	%MB185:	USINT;
0169	FopiBeck1_ChState_15_3	AT	%MB186:	USINT;
0170	FopiBeck1_ChState_15_4	AT	%MB187:	USINT;
0171				
0172	FopiBeck1_ChState_16_1	AT	%MB188:	USINT;
0173	FopiBeck1_ChState_16_2	AT	%MB189:	USINT;
0174	FopiBeck1_ChState_16_3	AT	%MB190:	USINT;
0175	FopiBeck1_ChState_16_4	AT	%MB191:	USINT;
0176				
0177	(* Systemstatus *)			
0178	FopiBeck1_SysInfo	AT	%MB192:	WORD;
0179	FopiBeck1_BusStatus	AT	%MB194:	WORD;
0180	END_VAR			

## Feldbuskomponenten Vorwort



# Dokumentation zum Beckhoff BC9000 Busklemmen Controller für Ethernet

Version 3.9

Datum: 14.10.02

Feldbuskomponenten: Vorwort

## Hinweise zum Handbuch

Diese Beschreibung wendet sich ausschließlich an ausgebildetes Fachpersonal der Steuerungs- und Automatisierungstechnik, das mit den geltenden nationalen Normen vertraut ist. Zur Installation und Inbetriebnahme der Komponenten ist die Beachtung der nachfolgenden Hinweise und Erklärungen unbedingt notwendig.

### Haftungsbedingungen

Das Fachpersonal hat sicherzustellen, dass die Anwendung bzw. der Einsatz der beschriebenen Produkte alle Sicherheitsanforderungen, einschließlich sämtlicher anwendbaren Gesetze, Vorschriften, Bestimmungen und Normen erfüllt.

Die Dokumentation wurde sorgfältig erstellt. Die beschriebenen Produkte werden jedoch ständig weiterentwickelt. Deshalb ist die Dokumentation nicht in jedem Fall vollständig auf die Übereinstimmung mit den beschriebenen Leistungsdaten, Normen oder sonstigen Merkmalen geprüft und stellt keine Eigenschaftszusicherung im Sinne des § 459, Abs. 2 BGB dar. Falls sie technische Fehler oder Schreibfehler enthält, behalten wir uns das Recht vor, Änderungen jederzeit und ohne Ankündigung durchzuführen. Aus den Angaben, Abbildungen und Beschreibungen in dieser Dokumentation können keine Ansprüche auf Änderung bereits gelieferter Produkte gemacht werden.

© Dieses Handbuch ist urheberrechtlich geschützt. Jede Wiedergabe oder Drittverwendung dieser Publikation, ganz oder auszugsweise, ist ohne schriftliche Erlaubnis der Elektro Beckhoff GmbH verboten.

Feldbuskomponenten: Vorwort

## Sicherheitshinweise

### Sicherheitsbestimmungen

Das Fachpersonal hat sicherzustellen, dass die Anwendung bzw. der Einsatz der beschriebenen Produkte alle Sicherheitsanforderungen, einschließlich sämtlicher anwendbaren Gesetze, Vorschriften, Bestimmungen und Normen erfüllt.

### Auslieferungszustand

Die gesamten Komponenten werden je nach Anwendungsbestimmungen in bestimmten Hard- und Software-Konfigurationen ausgeliefert. Änderungen der Hard-, oder Software-Konfiguration, die über die dokumentierten Möglichkeiten hinausgehen sind unzulässig und bewirken den Haftungsausschluss der Elektro Beckhoff GmbH.

### Qualifikation des Personals

Diese Beschreibung wendet sich ausschließlich an ausgebildetes Fachpersonal der Steuerungs- und Automatisierungstechnik, das mit den geltenden nationalen Normen vertraut ist.

### Erklärung der Sicherheitssymbole

In der vorliegenden Betriebsanleitung werden die folgenden Sicherheitssymbole verwendet. Diese Symbole sollen den Leser vor allem auf den Text des nebenstehenden Sicherheitshinweises aufmerksam machen.



**Gefahr**

Dieses Symbol weist darauf hin, dass Gefahren für Leben und Gesundheit von Personen bestehen.

Dieses Symbol weist darauf hin, dass Gefahren für Maschine, Material oder

**Achtung**

Umwelt bestehen.

**Hinweis**

Dieses Symbol kennzeichnet Informationen, die zum besseren Verständnis beitragen.

Feldbuskomponenten Vorwort

## Ausgabestände der Dokumentation

Version	Änderungen
3.8	HTML-Dokumentation; gültig ab Firmware B8
3.9	HTML-Dokumentation; gültig ab Firmware B9

### Firmware BC9000

Welche Firmware im Auslieferungszustand auf dem Buskoppler ist, erkennen Sie an dem Aufkleber unter dem Koppler (siehe fünfte und sechste Stelle der Produktionsnummer).

Beispiel:

3200B2020000

Die Firmware im Beispiel ist B2.

Zum Update Ihrer Firmware benötigen Sie die Konfigurationssoftware KS2000 und das zu deren Lieferumfang gehörende serielle Kabel oder Sie können ab Software Version B6 Ihren Koppler mit dem TwinCAT 2.8 System Manager updaten. Die Firmware finden Sie unter [www.beckhoff.com](http://www.beckhoff.com).

Firmware	Beschreibung
B0	Vorab Version
B1	Freigegebene Version
B2	- Probleme mit den ASC/CHR-Funktionen behoben - Fehler, das nach einem Neustart von TwinCAT der Fehler-Code 1 erzeugt wird, wurde behoben - Implementierung der PLCSystemBC.lb6
B3	Implementierung von ModbusTCP
B5	- Adressierung über ARP und BootP möglich - Zugriff auf den Merkerbereich über ModbusTCP
B6	- Implementierung der Funktionsbausteine ModbusTCP, ADS, SMTP, IP-Config - Firmware Update über Ethernet möglich - Auslesen der Konfiguration mit System-Manager (ab TwinCAT 2.8) möglich - Zugriffskontrolle ADS über Tabelle 2 implementiert
	- Beim Firmware Update bleiben die Einstellungen, wie IP-Adresse, erhalten - ADS Indication Bausteine Implementiert

B7	- ADS "State" werden unterstützt - DHCP für "Nortel Switch" geändert - Zykluszeit Überwachung - Fast ModbusTCP implementiert
B8	- Interner Software Reset geändert (dies hat keine Auswirkung für den Benutzer)
B9	Fehler der B8 Version behoben, betreffend mapping der KL60xx Busklemmen

## Firmware BK9000

Welche Firmware im Auslieferungszustand auf dem Buskoppler ist, erkennen Sie an dem Aufkleber unter dem Koppler (siehe fünfte und sechste Stelle der Produktionsnummer).

Beispiel:

3200B2020000

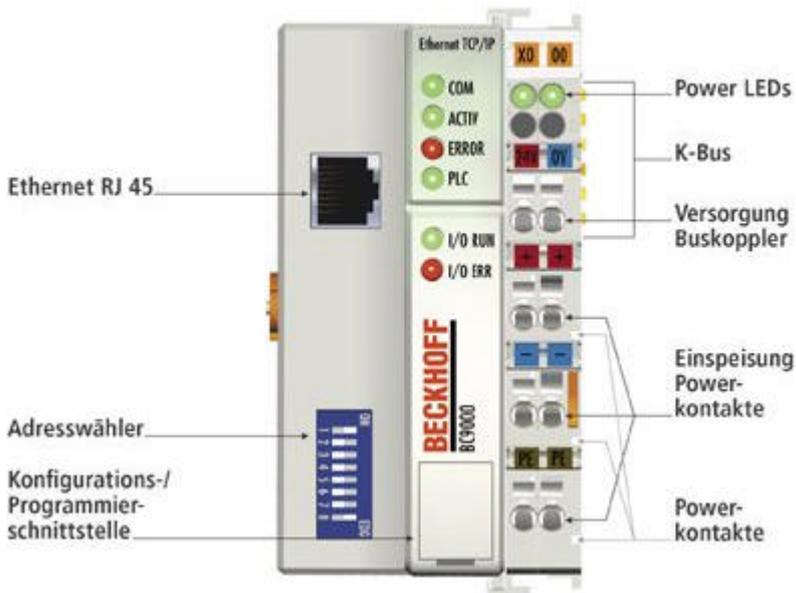
Die Firmware im Beispiel ist B2.

Zum Update Ihrer Firmware benötigen Sie die Konfigurationssoftware KS2000 und das zu deren Lieferumfang gehörende serielle Kabel oder Sie können ab Software Version B6 Ihren Koppler mit dem TwinCAT 2.8 System Manager updaten. Die Firmware finden Sie unter [www.beckhoff.com](http://www.beckhoff.com).

Firmware	Beschreibung
B0	Freigegebene Version
B1	Watchdog LED wird bei ModbusTCP getriggert
B2	- Adressierung über ARP, DHCP und BootP möglich - Zugriff mit FC3 ModbusTCP auf auf Prozesseingänge implementiert
B3	- Firmware Update über Ethernet möglich - Auslesen der Konfiguration mit System-Manager (ab TwinCAT 2.8) möglich - Zugriffskontrolle ADS über Tabelle 2 implementiert
B4	- Beim Firmware Update bleiben die Einstellungen, wie IP-Adresse, erhalten
B5	- Fast ModbusTCP implementiert
B6	- Interner Software Reset geändert (dies hat keine Auswirkung für den Benutzer)
B7	- Fehler der B6 Version behoben, betreffend Mapping der KL60xx Busklemmen

Feldbuskomponenten: Produktübersicht - Technische Daten

## Technische Daten



Systemdaten	Ethernet (BC9000)
Anzahl der E/A-Module	nur durch IP-Adressraum begrenzt
Anzahl der E/A-Punkte	steuerungsabhängig
Übertragungsmedium	4 x 2 Twisted-Pair Kupferkabel; Kategorie 3 (10 MBaud), Kategorie 5 (100 MBaud)
Länge zwischen Modulen	100 m (Verteiler Hub bis BC9000)
Übertragungsrate	10/100 MBaud
Topologie	sternförmige Verkabelung

Technische Daten	BC9000
Anzahl der Busklemmen	64
Digitale Peripheriesignale	256 Ein-/Ausgänge
Analoge Peripheriesignale	128 Ein-/Ausgänge
Protokolle	Beckhoff ADS, ModbusTCP
Konfigurationsmöglichkeit	über die Konfigurationssoftware KS2000 oder die Steuerung (TwinCAT / TwinCAT BC)
maximale Byteanzahl	512 Byte E und 512 Byte A
Busanschluss	RJ 45
Spannungsversorgung	24 V DC (-15%/+20%)
Eingangsstrom	70 mA + (ges. K-Bus Strom)/4
Einschaltstrom	ca. 2,5 x Dauerstrom
Empf. Vorsicherung	≤ 10 A
K-Bus-Stromversorgung bis	1750 mA
Spannung Powerkontakt	24 V DC max.
Stromlast Powerkontakt	10 A max.
Spannungsfestigkeit	500 Veff (Powerkontakt/Versorgungsspannung/Feldbus)
Gewicht ca.	170 g
Betriebstemperatur	0°C ... +55°C
Lagertemperatur	-25°C ... +85°C
Relative Feuchte	95% ohne Betauung
Vibrations-	gemäß IEC 68-2-6 / IEC 68-2-27

/Schockfestigkeit	
EMV-Festigkeit/Aussendung	gemäß EN 50082 (ESD, Burst) / EN 50081
Einbaulage	beliebig
Schutzart	IP20

Feldbuskomponenten: Produktübersicht - Technische Daten

## Technische Daten der SPS

PLC Daten	BC9000
Programmiermöglichkeit	über Programmierschnittstelle (TwinCAT, TwinCAT BC) oder über Ethernet (TwinCAT)
Programmspeicher	64/96 kByte
Datenspeicher	64/128 kByte
Remanente Merker	4 kByte
SPS-Zykluszeit	ca. 1,5 ms für 1000 AWL Befehle (ohne E/A Zyklus)
Programmiersprachen	IEC 6-1131-3 (AWL, KOP, FUP, ST, AS)

Feldbuskomponenten: Produktübersicht - Systemübersicht

## Systemübersicht

### Bis zu 64 Busklemmen mit jeweils 2 E/A-Kanälen für jede Signalform

Das Busklemmen-System ist das universelle Bindeglied zwischen einem Feldbus-System und der Sensor / Aktuator - Ebene. Eine Einheit besteht aus einem Buskoppler als Kopfstation und bis zu 64 elektronischen Reihenklemmen, wovon die letzte eine Endklemme ist. Für jede technische Signalform stehen Klemmen mit jeweils zwei E/A-Kanälen zur Verfügung, die beliebig gemischt werden können. Dabei haben alle Klemmentypen die gleiche Bauform, wodurch der Projektierungsaufwand sehr gering gehalten wird. Bauhöhe und Tiefe sind auf kompakte Klemmenkästen abgestimmt.

### Dezentrale Verdrahtung der E/A-Ebene

Die Feldbustechnik erlaubt den Einsatz kompakter SteuerungsbaufORMen. Die E/A-Ebene muss nicht bis zur Steuerung geführt werden. Die Verdrahtung der Sensoren und Aktuatoren ist dezentral mit minimalen Kabellängen durchführbar. Der Installationsstandort der Steuerung kann im Bereich der Anlage beliebig gewählt werden.

### Industrie-PCs als Steuerung

Durch den Einsatz eines Industrie-PCs als Steuerung lässt sich das Bedien- und Beobachtungselement in der Hardware der Steuerung realisieren. Der Standort der Steuerung kann deshalb ein Bedienpult, eine Leitwarte oder ähnliches sein. Die Busklemmen stellen die dezentrale Ein/Ausgabebene der Steuerung im Schaltschrank und untergeordneten Klemmenkästen dar. Neben der Sensor/Aktuator-Ebene wird auch

der Leistungsteil der Anlage über das Bussystem gesteuert. Die Busklemme ersetzt die konventionelle Reihenklemme als Verdrahtungsebene im Schaltschrank. Der Schaltschrank kann kleiner dimensioniert werden.

## **Buskoppler für alle gängigen Bussysteme**

Das Beckhoff Busklemmen-System vereint die Vorteile eines Bussystems mit den Möglichkeiten der kompakten Reihenklemme. Busklemmen können an allen gängigen Bussystemen betrieben werden und verringern so die Teilevielfalt in der Steuerung. Dabei verhalten sich Busklemmen wie herkömmliche Anschaltungen dieses Bussystems. Alle Leistungsmerkmale des jeweiligen Bussystems werden unterstützt.

## **Montage auf genormten C-Tragschienen**

Die einfache und platzsparende Montage auf einer genormten C-Tragschiene und die direkte Verdrahtung von Aktuatoren und Sensoren ohne Querverbindungen zwischen den Klemmen standardisiert die Installation. Dazu trägt auch das einheitliche Beschriftungskonzept bei.

Die geringe Baugröße und die große Flexibilität des Busklemme-Systems ermöglichen den Einsatz überall dort, wo auch eine Reihenklemme zur Anwendung kommt. Jede Art von Ankopplung, wie analoge, digitale, serielle oder der Direktanschluss von Sensoren kann realisiert werden.

## **Modularität**

Die modulare Zusammenstellung der Klemmleiste mit Busklemmen verschiedener Funktionen begrenzt die Zahl der ungenutzten Kanäle auf maximal einen pro Funktion. Die Anzahl von zwei Kanälen in einer Klemme trifft das Optimum zwischen der Zahl der ungenutzten Kanäle und den Kosten pro Kanal. Auch die Möglichkeit der Potentialtrennung durch Einspeiseklemmen hilft, die Anzahl der ungenutzten Kanäle gering zu halten.

## **Anzeige des Kanalzustands**

Die integrierten Leuchtdioden zeigen in Sensor/Aktuator-Nähe den Zustand des entsprechenden Kanals an.

## **K-Bus**

Der K-Bus ist der Datenweg innerhalb der Klemmleiste. Über sechs Kontakte an den Seitenwänden der Klemmen wird der K-Bus vom Buskoppler durch alle Klemmen geführt. Die Endklemme schließt den K-Bus ab. Der Benutzer muss sich keinerlei Wissen über die Funktion des K-Bus oder die interne Arbeitsweise von Klemmen und Buskoppler aneignen. Viele lieferbare Software-Tools erlauben eine komfortable Projektierung, Konfiguration und Bedienung.

## **Potential-Einspeiseklemmen für potentialgetrennte Gruppen**

Über drei Powerkontakte wird die Betriebsspannung an die nachfolgenden Klemmen weitergegeben. Durch den Einsatz von Potential-Einspeiseklemmen, können Sie die Klemmleiste in beliebige potentialgetrennte Gruppen gliedern. Die Potential-Einspeiseklemmen werden bei der Ansteuerung der Klemmen nicht berücksichtigt, sie dürfen an beliebiger Stelle in die Klemmleiste eingereicht werden.

In einer Klemmleiste können bis zu 64 Klemmen eingesetzt werden. Dabei werden Potential-Einspeiseklemmen mitgezählt, die Endklemme nicht..

## **Buskoppler für verschiedene Feldbus-Systeme**

Verschiedene Buskoppler lassen sich einsetzen, um die elektronische Klemmleiste schnell und einfach an unterschiedliche Feldbus-Systeme anzukoppeln. Auch eine nachträgliche Umrüstung auf ein anderes Feldbus-System ist möglich. Der Buskoppler übernimmt alle Kontroll- und Steuerungsaufgaben, die für den Betrieb der angeschlossenen Busklemmen notwendig sind. Die Bedienung und Konfiguration der Busklemmen wird ausschließlich über den Buskoppler durchgeführt. Die eingestellten Parameter werden jedoch spannungsausfallsicher in den jeweiligen Busklemmen gespeichert. Feldbus, K-Bus und E/A-Ebene sind galvanisch getrennt.

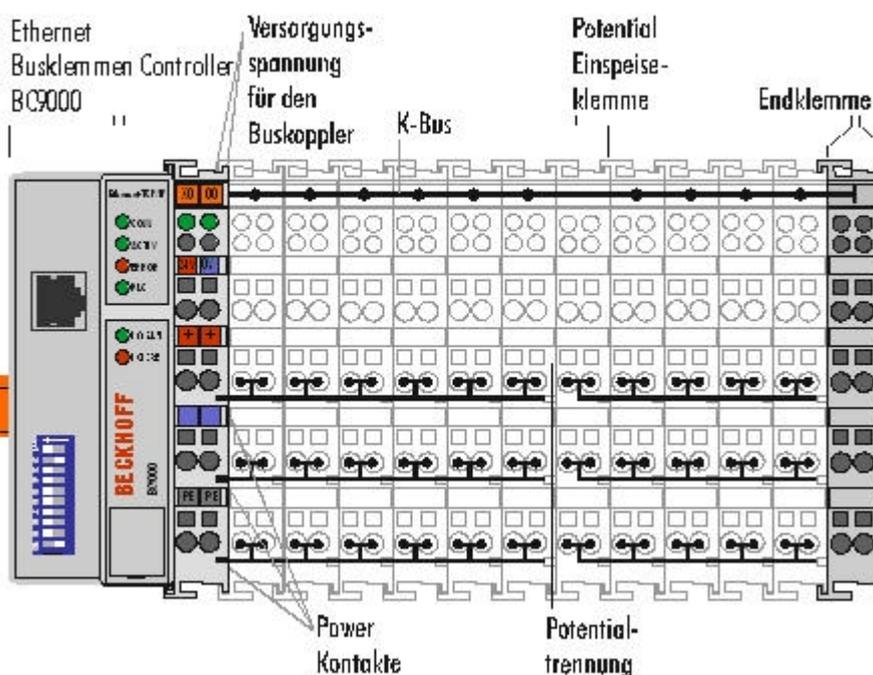
Wenn der Datenaustausch über den Feldbus zeitweise gestört ist oder ausfällt, bleiben Registerinhalte (wie z. B. Zählerstände) erhalten, digitale Ausgänge werden gelöscht und analoge Ausgänge nehmen einen Wert an, der bei der Inbetriebnahme für jeden Ausgang konfigurierbar ist. Die Default-Einstellung der analogen Ausgänge ist 0 V bzw. 0 mA. Digitale Ausgänge fallen in einen inaktiven Zustand zurück. Die Timeout-Zeiten der Buskoppler entsprechen den für das Feldbus-System üblichen Zeiten. Bei der Umstellung auf ein anderes Bussystem beachten Sie im Falle großer Zykluszeiten des Bussystems die Änderung der Timeout-Zeiten.

## Die Schnittstellen

Ein Buskoppler besitzt sechs unterschiedliche Anschlussmöglichkeiten. Diese Schnittstellen sind als Steckverbindungen und Federkraftklemmen ausgelegt.

Feldbuskomponenten: Produktübersicht - Systemübersicht

## Prinzip der Busklemme



Feldbuskomponenten: Produktübersicht - Feldbusübersicht

# Feldbusübersicht

Ethernet wurde ursprünglich von DEC, Intel und Xerox (als DIX-Standard) für die Datenübertragung zwischen Bürogeräten entwickelt. Heute versteht man darunter meist die Spezifikation *IEEE 802.3 CSMA/CD*, die 1985 veröffentlicht wurde. Diese Technologie ist durch ihren weltweiten Einsatz und die hohen Stückzahlen überall erhältlich und sehr preiswert.. Eine Anbindung an vorhandene Netze kann so problemlos realisiert werden.

Mittlerweile gibt es die verschiedensten Übertragungsmedien: Koaxialkabel (10Base5), Lichtwellenleiter (10BaseF) oder verdrehte Zweidrahtleitung (10BaseT) mit Schirmung (STP) oder ohne Schirmung (UTP). Mit Ethernet lassen sich verschiedenen Topologien aufbauen wie Ring, Linie oder Stern.

Ethernet transportiert Ethernet-Pakete von einem Sender zu einem oder mehreren Empfängern. Diese Übertragung verläuft ohne Quittung und ohne Wiederholung von verlorenen Paketen. Für die sichere Daten-Kommunikation stehen Protokolle wie TCP/IP zu Verfügung, die auf Ethernet aufsetzen.

## Grundlagen

### Internet-Protokoll (IP)

Die Grundlage der Datenkommunikation ist das Internet-Protokoll (IP). IP transportiert Datenpakete von einem Teilnehmer zu einem anderen, der sich im gleichen oder in einem anderen Netz befinden kann. IP kümmert sich dabei um das Adress-Management (Finden und Zuordnen der MAC-IDs), die Segmentierung und die Pfadsuche (Routing). Wie das Ethernet-Protokoll gewährleistet auch IP keinen gesicherten Transport der Daten; Datenpakete können verloren gehen oder in ihrer Reihenfolge vertauscht werden.

Für einen standardisierten, gesicherten Informationsaustausch zwischen beliebig vielen verschiedenen Netzwerken wurde TCP/IP entwickelt. Dabei ist TCP/IP weitgehend unabhängig von der verwendeten Hard- und Software. Oftmals als ein Begriff verwendet, handelt es sich hierbei um mehrere aufeinander aufgesetzte Protokolle: z.B. IP, TCP, UDP, ARP und ICMP.

### Transmission Control Protocol (TCP)

Das auf IP aufsetzende Transmission Control Protocol (TCP) ist ein verbindungsorientiertes Transport-Protokoll. Es umfasst Fehlererkennungs- und Behandlungsmechanismen. Verlorene Telegramme werden wiederholt.

### User Datagram Protocol (UDP)

UDP ist ein verbindungsloses Transport-Protokoll. Es gibt keine Kontrollmechanismen beim Datenaustausch zwischen Sender und Empfänger. Dadurch resultiert eine schneller Verarbeitungsgeschwindigkeit als zum Beispiel bei TCP. Eine Prüfung ob das Telegramm angekommen ist muss vom übergeordneten Protokoll durchgeführt werden.

### Internet Control Message Protocol (ICMP)

Dient dem Informationsaustausch der Endgeräte über den aktuellen Status des Internet-Protokolls.

### Address Resolution Protocol (ARP)

Dient der Umwandlung zwischen IP-Adressen und MAC-Adressen.

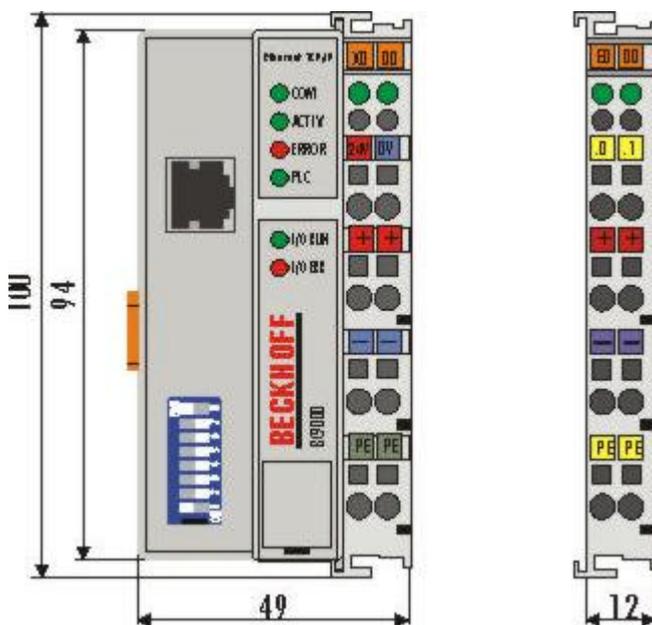
## BootP

Das BootP-Protokoll ermöglicht das Einstellen und ändern der TCP/IP-Adresse indem der Netzwerk-Teilnehmer mit der MAC-ID angesprochen wird.

Feldbuskomponenten: Einbau und Verdrahtung - Mechanischer Einbau

## Abmessungen

Das System der Beckhoff-Busklemmen zeichnet sich durch geringes Bauvolumen und hohe Modularität aus. Für die Projektierung muss ein Buskoppler und eine Anzahl von Busklemmen vorgesehen werden. Die Baumaße der Buskoppler sind unabhängig vom Feldbus-System.



Die Gesamtbreite der Anwendung setzt sich aus der Breite des Buskopplers, der Busendklemme KL9010 und der Breite der verwendeten Busklemmen zusammen. Die Busklemmen sind je nach Funktion 12 mm oder 24 mm breit. Die lichte Höhe von 68 mm wird durch die Frontverdrahtung in Abhängigkeit von der Drahtstärke um ca. 5 mm bis 10 mm überbaut.

Feldbuskomponenten: Busklemmen

## Tragschienenmontage



**Gefahr**

Setzen Sie das Bus-System in einen sicheren, spannungslosen Zustand, bevor Sie mit der Montage, Demontage oder Verdrahtung der Busklemmen beginnen!

## Montage

Die Buskoppler und Busklemmen werden durch leichten Druck auf handelsübliche 35 mm C-Tragschienen (EN 50022) aufgerastet:

1. Stecken Sie zuerst den Feldbuskoppler auf die Tragschiene.
2. Auf der rechten Seite des Feldbuskoppler werden nun die Busklemmen angereiht. Stecken Sie dazu die Komponenten mit Nut und Feder zusammen und schieben Sie die Klemmen gegen die Tragschiene, bis die Verriegelung hörbar auf der Tragschiene einrastet.  
Wenn Sie die Klemmen erst auf die Tragschiene schnappen und dann nebeneinander schieben ohne das Nut und Feder ineinander greifen, wird keine funktionsfähige Verbindung hergestellt! Bei richtiger Montage darf kein nennenswerter Spalt zwischen den Gehäusen zu sehen sein.

Achten Sie bei der Montage der Busklemmen darauf, dass der Verriegelungsmechanismus der Klemmen nicht in Konflikt mit den Befestigungsschrauben der Tragschiene gerät.

## Demontage

Jede Klemme wird durch eine Verriegelung auf der Tragschiene gesichert, die zur Demontage gelöst werden muss:

1. Ziehen Sie vorsichtig die orangefarbige Lasche ca. 1 cm aus der zu demontierenden Klemme heraus, bis die Lasche locker hervorsteht. Jetzt ist für diese Klemme die Verriegelung mit der Tragschiene gelöst und die Klemme kann ohne großen Kraftaufwand von der Tragschiene gezogen werden.
2. Greifen Sie dazu mit Daumen und Zeigefinger die entriegelte Klemme gleichzeitig oben und unten an den geriffelten Gehäuseflächen und ziehen Sie die Klemme von der Tragschiene weg.

## Verbindungen innerhalb eines Busklemmenblocks

Die elektrischen Verbindungen zwischen Buskoppler und Busklemmen werden durch das Zusammenstecken der Komponenten automatisch realisiert:

- Die sechs Federkontakte des K-Bus übernehmen die Übertragung der Daten und die Versorgung der Busklemmenelektronik.
- Die Powerkontakte übertragen die Versorgung für die Feldelektronik und stellen so innerhalb des Busklemmenblocks eine Versorgungsschiene dar. Die Versorgung der Powerkontakte erfolgt über Klemmen auf dem Buskoppler.



**Hinweis**

Beachten Sie bei der Projektierung eines Busklemmenblocks die Kontaktbelegungen der einzelnen Busklemmen, da einige Typen (z.B. analoge Busklemmen oder digitale 4-Kanal-Busklemmen) die Powerkontakte nicht oder nicht vollständig durchschleifen.  
Einspeiseklemmen (KL91xx, KL92xx) unterbrechen die Powerkontakte und stellen so den Anfang einer neuen Versorgungsschiene dar.

## PE-Powerkontakt

Der Powerkontakt mit der Bezeichnung PE kann als Schutz Erde eingesetzt werden. Der Kontakt ist aus Sicherheitsgründen beim Zusammenstecken voreilend und kann Kurzschlussströme bis 125 A ableiten.



**Achtung**

Beachten Sie, dass aus EMV-Gründen die PE-Kontakte kapazitiv mit der Tragschiene verbunden sind. Das kann bei der Isolationsprüfung zu falschen Ergebnissen und auch zur Beschädigung der Klemme führen (z. B. Durchschlag zur PE-Leitung bei der Isolationsprüfung eines Verbrauchers mit 230 V Nennspannung).

Klemmen Sie zur Isolationsprüfung die PE-Zuleitung am Buskoppler bzw. der Einspeiseklemme ab! Um weitere Einspeisestellen für die Prüfung zu

entkoppeln, können Sie diese Einspeiseklemmen entriegeln und mindestens 10 mm aus dem Verbund der übrigen Klemmen herausziehen.

Der PE-Powerkontakt darf nicht für andere Potentiale verwendet werden!

## Verdrahtung

Bis zu acht Anschlüsse ermöglichen den Anschluss von massiven oder feindrähtigen Leitungen an die Busklemmen. Die Klemmen sind in Federkrafttechnik ausgeführt. Schließen Sie die Leitungen folgendermaßen an:

1. Öffnen Sie eine Federkraftklemme, indem Sie mit einem Schraubendreher oder einem Dorn leicht in die viereckige Öffnung über der Klemme drücken.
2. Der Draht kann nun ohne Widerstand in die runde Klemmenöffnung eingeführt werden.
3. Durch Rücknahme des Druckes schließt sich die Klemme automatisch und hält den Draht sicher und dauerhaft fest.

Feldbuskomponenten: Einbau und Verdrahtung - Verdrahtung

## Speisung, Potentialgruppen

### Spannungsversorgung Buskoppler

Die Buskoppler benötigen zum Betrieb eine 24 V Gleichspannung. Der Anschluss findet über die oberen Federkraftklemmen mit der Bezeichnung 24 V und 0 V statt. Die Versorgungsspannung versorgt die Buskopplerelektronik und über den K-Bus die Busklemmen. Die Spannungsversorgung der Buskopplerelektronik und die des K-Bus sind galvanisch von der Spannung der Feldebene getrennt.

### Einspeisung Powerkontakte

Die unteren sechs Anschlüsse mit Federkraftklemmen können zur Einspeisung der Peripherieversorgung benutzt werden. Die Federkraftklemmen sind paarweise mit einem Powerkontakt verbunden. Die Einspeisung zu den Powerkontakten besitzt keine Verbindung zur Spannungsversorgung der Buskoppler. Die Auslegung der Einspeisung lässt Spannungen bis zu 24 V zu. Die paarweise Anordnung und die elektrische Verbindung zwischen den Speiseklemmkontakten ermöglicht das Durchschleifen der Anschlussdrähte zu unterschiedlichen Klemmpunkten. Die Strombelastung über den Powerkontakten darf 10 A nicht dauerhaft überschreiten. Die Strombelastbarkeit zwischen zwei Federkraftklemmen ist mit der Belastbarkeit der Verbindungsdrähte identisch.

### Powerkontakte

An der rechten Seitenfläche des Buskopplers befinden sich drei Federkontakte der Powerkontaktverbindungen. Die Federkontakte sind in Schlitzen verborgen um den Berührungsschutz sicherzustellen. Durch das Anreihen einer Busklemme werden die Messerkontakte auf der linken Seite der Busklemme mit den Federkontakten verbunden. Die Nut/Federführung an der Ober- und Unterseite der Buskoppler und Busklemmen garantiert sichere Führung der Powerkontakte.

### Konfigurations-Schnittstelle

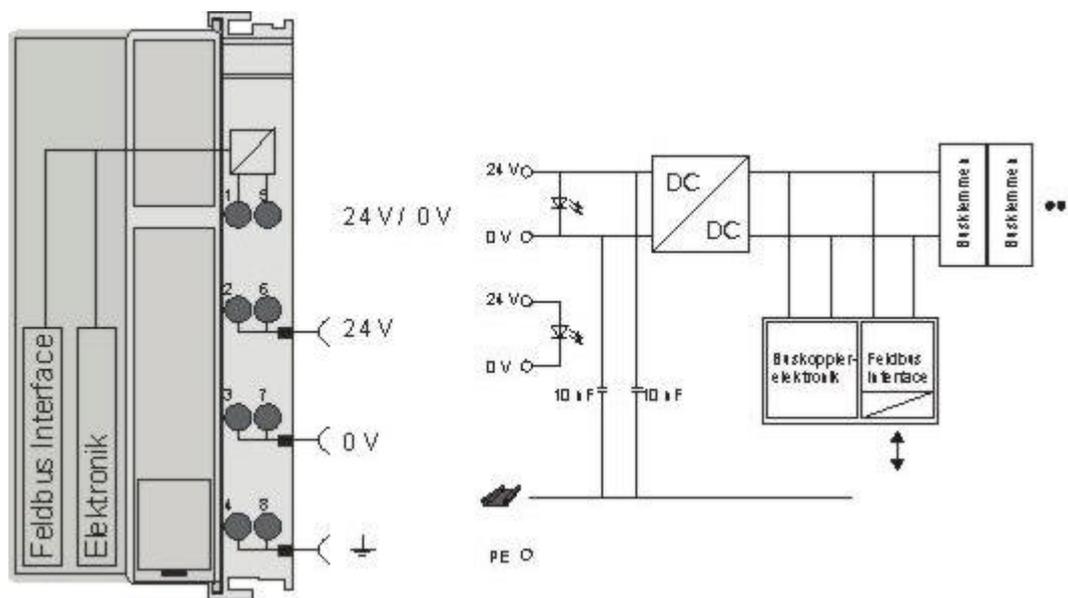
Auf der unteren Seite der Frontfläche sind die Standardbuskoppler mit einer RS232-Schnittstelle ausgerüstet. Der Miniaturstecker kann mit einem Verbindungskabel und der Konfigurationssoftware KS2000 mit einem PC verbunden werden. Die Schnittstelle erlaubt das Konfigurieren der Busklemmen,

wie zum Beispiel das Einstellen des Verstärkungsfaktors der analogen Kanäle. Über die Schnittstelle kann auch die Zuordnungen der Busklemmen- Daten zum Prozessabbild im Buskoppler verändert werden. Die Funktionalität der Konfigurationsschnittstelle ist auch über den Feldbus mit der Stringkommunikation zu erreichen.

## Potentialtrennung

Die Buskoppler arbeiten mit drei unabhängigen Potentialgruppen. Die Versorgungsspannung speist galvanisch getrennt die K-Bus-Elektronik im Buskoppler und den K-Bus selbst. Aus der Versorgungsspannung wird weiter die Betriebsspannung für den Betrieb des Feldbus erzeugt.

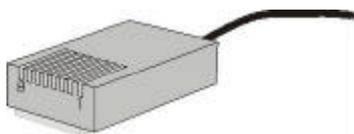
Anmerkung: Alle Busklemmen haben eine galvanische Trennung zum K-Bus. Der K-Bus ist dadurch vollständig galvanisch gekapselt.



Feldbuskomponenten: Einbau und Verdrahtung - Verdrahtung Feldbus

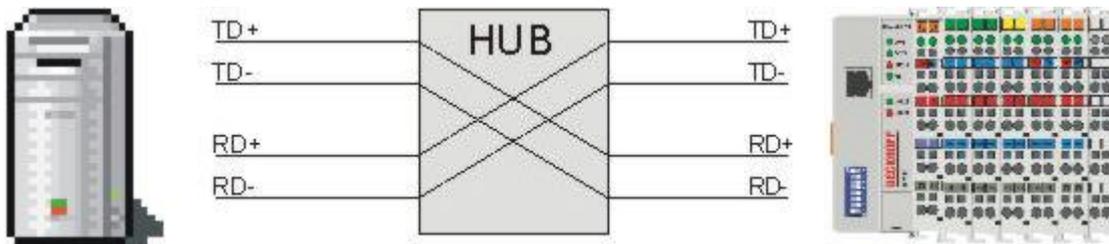
## Ethernet-Anschluss

Der Anschluss an den Ethernet-Bus erfolgt über einen RJ45-Stecker (Westernstecker).



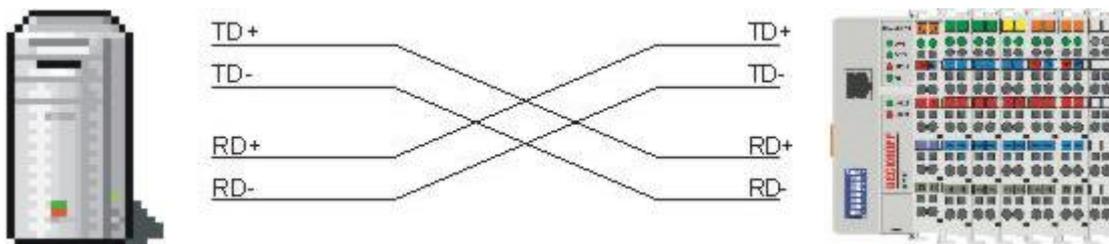
## Verkabelung

### Verbindung über Hub oder Switch



Verbinden Sie die Netzwerkkarte des PCs über ein Standard-Ethernetkabel mit dem Hub und das Hub ebenfalls über ein Standard-Ethernetkabel mit dem Busklemmen Controller. Die Verbindung über einen Switch erfolgt auf die gleiche Weise.

### Direkte Verbindung zwischen PC mit Ethernet-Karte und BC9000



Um den PC direkt mit dem Busklemmen Controller zu verbinden, müssen Sie ein Ethernetkabel mit gekreuzten Adernpaaren (Cross-Over-Kabel) verwenden.

### Belegung des RJ45-Steckers

PIN	Signal	Beschreibung
1	TD +	Transmit +
2	TD -	Transmit -
3	RD +	Receive +
4	-	reserviert
5	-	reserviert
6	RD -	Receive -
7	-	reserviert
8	-	reserviert

Feldbuskomponenten: Ethernet

## Ethernet-Kabel

### Übertragungsstandards

#### 10Base5

Das Übertragungsmedium für 10Base5 ist ein dickes Koaxialkabel (Yellow Cable) mit einer max. Übertragungsgeschwindigkeit von 10 Mbaud und einer Linien-Topologie mit Abzweigen (Drops), an die jeweils ein Teilnehmer angeschlossen wird. Da hier alle Teilnehmer an einem gemeinsamen Übertragungsmedium angeschlossen sind, kommt es bei 10Base5 zwangsläufig häufig zu Kollisionen.

## **10Base2**

10Base2 (Cheapernet) ist eine Weiterentwicklung von 10Base5 und hat den Vorteil dass dieses Koaxialkabel billiger und durch eine höhere Flexibilität einfacher zu verlegen ist. Es können mehrere Geräte an eine 10Base2-Leitung angeschlossen werden. Häufig werden die Abzweige eines 10Base5-Backbones als 10Base2 ausgeführt.

## **10BaseT**

Beschreibt ein Twisted-Pair-Kabel für 10 MBaud. Hierbei wird das Netz sternförmig aufgebaut, so dass nun nicht mehr jeder Teilnehmer am gleichem Medium hängt. Dadurch führt ein Kabelbruch nicht mehr zum Ausfall des gesamten Netzes. Durch den Einsatz von Switches als Sternkoppler können Kollisionen vermindert oder bei Voll-Duplex Verbindungen auch vollständig vermieden werden.

## **100BaseT**

Twisted-Pair-Kabel für 100 MBaud. Für die höhere Datengeschwindigkeit ist eine bessere Kabelqualität und die Verwendung entsprechender Hubs oder Switches erforderlich.

## **10BaseF**

Der Standard 10BaseF beschreibt mehrere Lichtwellenleiter-Varianten.

## **Kurzbezeichnung der Kabeltypen für 10BaseT und 100BaseT**

Twisted-Pair Kupferkabel für sternförmige Topologie, wobei der Abstand zwischen zwei Geräten 100 Meter nicht überschreiten darf.

### **UTP**

Unshielded Twisted-Pair (nicht abgeschirmte, verdrehte Leitung)  
Dieser Kabeltyp gehört zur Kategorie 3 und sind für industrielle Umgebungen nicht empfehlenswert.

### **S/UTP**

Screened/Unshielded Twisted-Pair (mit Kupfergeflecht abgeschirmte, verdrehte Leitung)  
Besitzen einen Gesamtschirm aus einem Kupfergeflecht zur Reduktion der äußeren Störeinflüsse. Dieses Kabel wird zum Einsatz mit dem Buskopplern empfohlen.

### **FTP**

Foilesshielded Twisted-Pair (mit Alufolie abgeschirmte, verdrehte Leitung)  
Dieses Kabel hat eine alukaschierten Kunststoff-Folie-Gesamtschirm.

### **S/FTP**

Screened/Foilesshielded Twisted-Pair (mit Kupfergeflecht und Alufolie abgeschirmte, verdrehte Leitung)  
Besitzt einen alukaschierten Gesamtschirm mit einem darüberliegenden Kupfergeflecht. Solche Kabel können eine Störleistungsunterdrückung bis zu 70dB erreichen.

### **STP**

Shielded Twisted-Pair (abgeschirmte, verdrehte Leitung)

Beschreibt ein Kabel mit Gesamtschirm ohne weitere Angabe der Art der Schirmung.

#### S/STP

Screened/Shielded Twisted-Pair (einzeln abgeschirmte, verdrehte Leitung)

Ein solche Bezeichnung kennzeichnet ein Kabel mit einer Abschirmung für jedes Leitungspaar sowie einen Gesamtschirm.

#### ITP

Industrial Twisted-Pair

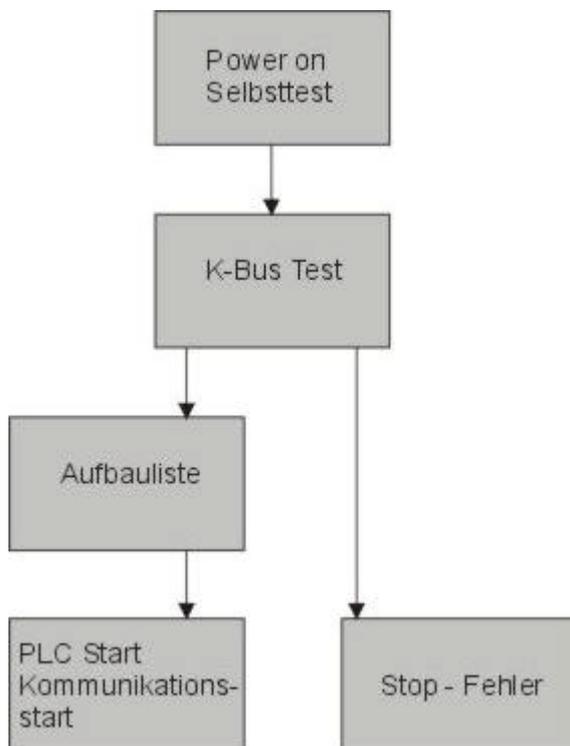
Ist von Aufbau dem S/STP ähnlich, besitzt allerdings im Gegensatz zum S/STP nur 2 Leitungspaare.

Feldbuskomponenten: Parametrierung und Inbetriebnahme

## Anlaufverhalten des Buskopplers

Nach dem Einschalten prüft der Buskoppler seinen Zustand, konfiguriert den K-Bus und erstellt anhand der gesteckten Busklemmen eine Aufbauliste. Wenn der Buskoppler eine dezentrale Steuerung besitzt (BCxxxx) wird nach einer erfolgreichen Aufbauliste die lokale SPS gestartet.

Beim Hochlauf des Buskopplers leuchten und blinken die E/A-LEDs. Im fehlerfreien Zustand sollte nach ca. 2-3 sec keine E/A-LED mehr blinken. Im Fehlerfall hängt es von der Fehlerart ab, welche LED blinkt ([siehe Diagnose-LEDs](#)).



Feldbuskomponenten: Inbetriebnahme

## Parametrierung des Buskopplers über DIP-Schalter

Die folgende Parametrierungen können Sie ohne Konfigurationssoftware mit Hilfe des DIP-Schalters und der Endklemme (KL9010) vornehmen.

Dieser Parametrierungsmodus ist nur dann aktiv, wenn am Buskoppler nur eine Endklemme (KL9010) gesteckt ist. Sonst gelten die normale Einstellungen.

### Wiederherstellen der Hersteller-Einstellungen

- Schalten Sie den Buskoppler aus und stecken Sie nur die Endklemme (KL9010).
- Stellen Sie alle DIP-Schalter auf ON und schalten Sie den Buskoppler wieder ein.
- Nach erfolgreichem Einstellen der Default-Parameter leuchtet die LED *Error* und die LEDs *I/O RUN* und *I/O ERR* blinken abwechselnd.
- Danach können Sie den Buskoppler ausschalten, die Busklemmen anschließen und wie gewohnt fortfahren.

### Löschen des Bootprojektes

- Schalten Sie den Buskoppler aus und stecken Sie nur die Endklemme (KL9010).
- Stellen Sie die DIP-Schalter 1 bis 9 auf ON und den DIP-Schalter 10 auf OFF und schalten Sie den Buskoppler wieder ein.
- Nach erfolgreichem Löschen des Bootprojektes blinken die LEDs *I/O RUN* und *I/O ERR* abwechselnd.
- Danach können Sie den Buskoppler ausschalten, die Busklemmen anschließen und wie gewohnt fortfahren.

### Einstellung der Ethernet-Parameter

- Schalten Sie den Buskoppler aus und stecken Sie nur die Endklemme (KL9010).
- Stellen Sie alle DIP-Schalter auf OFF und schalten Sie den Buskoppler wieder ein.
- Die LEDs *I/O RUN* und *I/O ERR* leuchten konstant.
- Nehmen Sie nun anhand der folgenden Tabelle die gewünschte Einstellung vor.

DIP-Schalter	Parameter	Auswahl	Einstellung	Kommentar
1	Baud-Rate	10 MBaud	OFF (0)	
		100 MBaud	ON (1)	Default
2	Auto-Baud-Rate	Disable	OFF (0)	
		Enable	ON (1)	Default
3	Übertragungsart	Halbduplex	OFF (0)	
		Vollduplex	ON (1)	Default

- Schalten Sie zur um Übernahme der Werte den DIP-Schalter 10 auf ON.  
Der Buskoppler zeigt nun durch blinken der LEDs *I/O RUN* und *I/O ERR*, das er die Parameter übernommen hat.
- Danach können Sie den Buskoppler ausschalten, die Busklemmen anschließen und wie gewohnt fortfahren.

Feldbuskomponenten: Ethernet

## Netzwerkclassen

Man unterscheidet drei verschiedene Adressierungsklassen. Diese legen fest, wie viele Adress-Bits für die Netzwerknummer und wie viele für die Rechnernummer (bzw. Knoten-) reserviert werden. Die Unterscheidung findet in den ersten 3 Bits der IP-Adresse statt. Die Klasse A besitzt 7 Bits für die Netzwerkadresse. Damit sind weltweit 126 Klasse A Netzwerke möglich. Diese Netzwerke können dann 224 Rechner besitzen. Netzwerke der Klasse B können 65536 Rechner beinhalten, wobei 16384 Netzwerke möglich sind. Klasse C Netzwerke haben höchstens 256 Rechner, aber es können 221 Netzwerke aufgebaut werden.

**Achtung**

Eine IP-Adresse muss im gesamten verbundenen Netzwerk einmalig sein!

	Netzwerk	Net-ID	Rechner-Adresse	Host-ID
Klasse A	0	1234567	01234567	01234567

1.xxx.xxx.xxx - 126.xxx.xxx.xxx

	Netzwerk	Net-ID	Rechner-Adresse	Host-ID
Klasse B	01	234567	01234567	01234567

128.0.xxx.xxx - 191.255.xxx.xxx

	Netzwerk	Net-ID	Rechner-Adresse	Host-ID
Klasse C	012	34567	01234567	01234567

192.0.0.xxx - 223.255.255.xxx

**Hinweis**

Die eingestellte IP-Adresse muss bei einer Kommunikation mit einem anderen Ethernet-Teilnehmer die gleiche Netzwerkkategorie besitzen. Beispiel: Ihr PC hat die Adresse 172.16.17.55, damit muss der Buskoppler die Adresse 172.16.xxx.xxx haben (die xxx stehen für eine Zahl von 0..255 wobei die 0 meist für Router reserviert ist).

Um sich die Adresse des eigenen PCs anzusehen, kann man bei Windows NT/2000 im DOS-Fenster den Befehl >>ipconfig<< eingeben.

Feldbuskomponenten: Adressierung

## IP-Adresse

Die Einstellung der IP-Adresse kann mit vier unterschiedlichen Verfahren durchgeführt werden, die im Folgenden genauer beschrieben werden.

Verfahren	Erläuterung	Benötigte Komponenten
KS2000	<a href="#">Adressierung über die Konfigurationssoftware KS2000 und DIP-Schalter</a>	Konfigurationssoftware KS2000 und KS2000-Kabel
ARP	<a href="#">Adressierung über die ARP-Tabelle</a>	PC mit Netzwerk
BootP	<a href="#">Adressierung über BootP-Server</a>	BootP-Server
DHCP	<a href="#">Adressierung über DHCP-Server</a>	DHCP-Server

Feldbuskomponenten: Adressierung KS2000

## Adresseinstellung mit KS2000

Mit der der Konfigurationssoftware KS2000 können Sie die TCP/IP-Adresse per Dialogfenster einstellen (ab Version 3.2.8) oder direkt in die Register schreiben. Die DIP-Schalter 9 und 10 müssen beide vor dem Einschalten auf OFF (0) stehen.

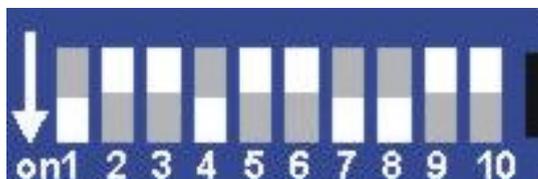
Tabelle 100

Register	High-Byte	Low-Byte
0	IP-Byte 2	IP-Byte 1
1	Nicht benutzt	IP-Byte 3

### Default

Byte	Default-Wert (hex)	Default-Wert (dez)
1	0xAC	172 <sub>dez</sub>
2	0x10	16 <sub>dez</sub>
3	0x11	17 <sub>dez</sub>
4	(DIP-Schalter)	(0 bis 255 <sub>dez</sub> )

### Beispiel:



Schalter-Nr.	1	2	3	4	5	6	7	8	9	10	
Wertigkeit	20	21	22	23	24	25	26	27	-	-	
Im Beispiel	ON	OFF	OFF	ON	OFF	OFF	ON	ON	OFF	OFF	
Wert	1	0	0	8	0	0	64	128	-	-	Summe=201

Feldbuskomponenten: Adressierung ARP

## Einstellung der IP-Adresse über die ARP-Tabelle

Eine einfache Methode die IP-Adresse zu ändern ist die Adresseinstellung über das DOS-Fenster. Sie können so allerdings nur Adressen innerhalb der selben Netzwerkkategorie ändern. Die neu eingestellte IP-Adresse bleibt auch beim Ausschalten des Buskopplers gespeichert.

Vorgehensweise:

- Stellen Sie die DIP-Schalter 9 und 10 auf OFF. Die DIP-Schalter 1-8 haben dann keine Adress-Funktion mehr.
- Öffnen Sie auf Ihrem PC eine DOS-Eingabeaufforderung (DOS-Box).
- Erzeugen Sie mit "ping >IP-Adresse ALT<" einen Eintrag in die ARP-Tabelle.
- Lesen Sie mit "ARP -a" die Tabelle aus.
- Löschen Sie mit "ARP -d >IP-Adresse ALT<" den Buskoppler aus der Tabelle.
- Nehmen Sie mit "ARS -s >IP Adresse NEU< >[MAC-ID](#)<" einen manuellen Eintrag vor.
- Mit "ping -l 123 >IP-Adresse NEU<" ist die neue IP-Adresse gültig

Kurzzeitiges blinken der ERROR LED im Einschaltaugenblick signalisiert, dass der Buskoppler über ARP adressiert wurde und die DIP-Schalter 1-8 keine Aussage über die eingestellte Adresse geben.



### Hinweis

Bei Änderung der IP-Adresse sollten alle dynamischen ARP-Einträge gelöscht werden, da der Buskoppler bei einem Empfang eines ICMP (Internet Control Message Protocol) IP-Telegramms (z.B. ein ping) mit der richtigen MAC-ID und der alten IP-Adresse wieder die alte IP-Adresse zulässt und sich umkonfiguriert. Um dies zu umgehen, wird nur ein ping mit der Länge 123 Byte zur Umkonfiguration der IP-Adresse zugelassen (>ping -l "IP-Adresse<).

Beispiel:

1. C:>ping 172.16.17.255
2. C:>ARP -a  
172.16.17.255 00-01-05-00-11-22
3. C:>arp -d 172.16.17.255
4. C:>arp -s 172.16.44.44 00-01-05-00-11-22
5. C:>ping -l 123 172.16.44.44

Feldbuskomponenten: Adressierung BootP

## Einstellung der IP-Adresse über den Beckhoff BootP-Server

Stellen Sie für die Adresseinstellung mit dem Beckhoff BootP-Server den DIP-Schalter 9 auf ON (1) und den DIP-Schalter 10 auf OFF (0). Die DIP-Schalter 1-8 haben dann keine Adress-Funktion mehr. Ist dies nicht der Fall so meldet der Buskoppler Fehler-Code 6-4. Während der Adressvergabe blinkt die LED *TCP/IP ERROR*.

## Speicherverhalten der IP-Adresse

### DIP-Schalter 1-8 in Stellung ON

Die vom BootP-Server vergebene IP-Adresse wird gespeichert und der BootP-Service wird beim nächsten Kaltstart nicht gestartet.

Die Adresse kann durch Reaktivierung der Herstellereinstellungen (mittels Konfigurationssoftware KS2000 oder [DIP-Schalter und Endklemme](#)) wieder gelöscht werden.

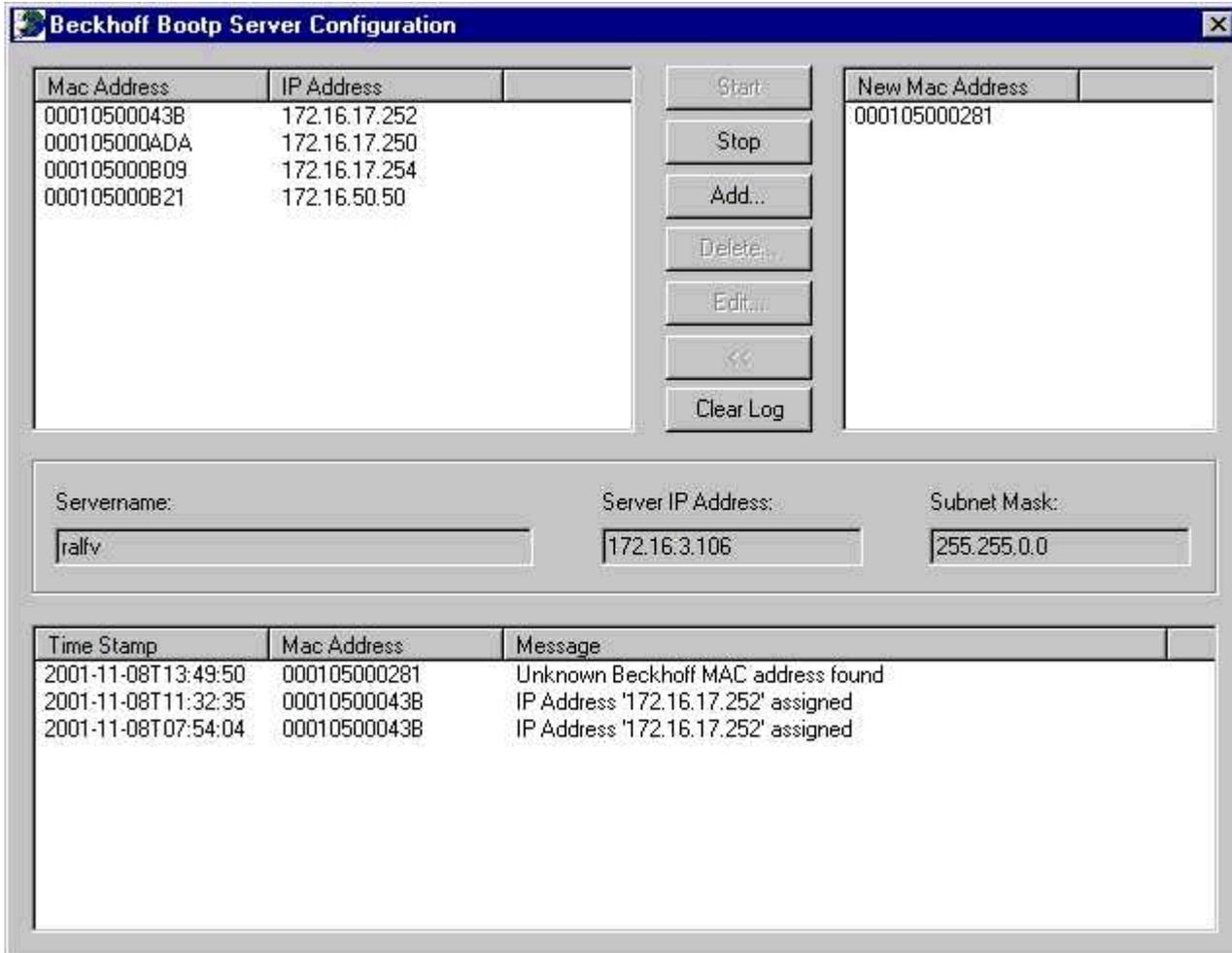
### DIP-Schalter 1-8 in Stellung OFF

Die vom BootP-Server vergebene IP-Adresse ist nur bis zum Ausschalten des Buskopplers gültig. Beim

nächsten Kaltstart muss der BootP-Server dem Buskoppler eine neue IP-Adresse vergeben. Bei einem Software-Reset des Buskopplers bleibt die Adresse allerdings erhalten.

## Beckhoff BootP-Server

Beckhoff bietet einen BootP-Server für Windows 98, ME, NT4.0, NT2000 und XP an.



Sobald der BootP-Server gestartet wird zeigt das Fenster *New MAC Address* alle Beckhoff-Knoten an, die im BootP-Betrieb arbeiten und noch keine IP-Adresse bekommen haben. Die Zuweisung [MAC-ID](#) zu IP-Adresse erfolgt mit dem Button "<<". Eine erfolgreiche Zuweisung wird im Log-Fenster angezeigt. Um den BootP-Server beim Booten Ihres PCs automatisch zu starten reicht eine Verknüpfung im Autostart von Windows. Fügen Sie hierzu im Verknüpfungspfad den Parameter */Start* an (.../TcBootPDIg.exe/start).

Feldbuskomponenten: Adressierung DHCP

## Adresseinstellung über einen DHCP-Server

Stellen Sie für die Adresseinstellung über einen DHCP-Server den DIP-Schalter 9 auf OFF (0) und den DIP-Schalter 10 auf ON (1). In diesem Zustand ist der DHCP-Dienst eingeschaltet und der Buskopplers bekommt automatisch eine IP-Nummer vom DHCP-Server zugewiesen. Der DHCP-Server muss hierfür die [MAC-ID](#) des Buskopplers kennen. Die IP-Adresse sollte statisch eingestellt werden. Während der Adressvergabe blinkt die LED *TCP/IP Error*.

Feldbuskomponenten: Subnet Mask

## Subnetz-Maske

Die Subnetz-Maske unterliegt der Kontrolle des Netzwerkverwalters und legt die Struktur der Subnetze fest.

Kleine Netze ohne Router benötigen keine Subnetz-Maske. Das gleiche gilt, wenn Sie keine registrierten IP-Nummern verwenden. Sie können die Subnetz-Maske dazu verwenden, anstelle des Gebrauchs vieler Netznummern das Netz mit dieser Maske zu unterteilen.

Die Subnetz-Maske ist eine 32-Bit Ziffer:

- Einsen in der Maske kennzeichnen den Subnetz-Nummernteil eines Adressraums.
- Nullen kennzeichnen den Teil des Adressraums, der für die Host-IDs zur Verfügung steht.

Beschreibung	Binäre Darstellung	Dezimale Darstellung
IP-Adresse	10101100.00010000.00010001.11001000	172.16.17.200
Subnetz-Maske	11111111.11111111.00010100.00000000	255.255.20.0
Netz-ID	10101100.00010000.00010000.00000000	172.16.16.0
Host-ID	00000000.00000000.00000001.11001000	0.0.1.200

## Standard Subnetz-Maske

Adressklasse	Standard Subnetz-Maske (dezimal)	Standard Subnetz-Maske (hex)
A	255.0.0.0	FF.00.00.00
B	255.255.0.0	FF.FF.00.00
C	255.255.255.0	FF.FF.FF.00

Die Subnetze 0 und das nur aus nur Einsen bestehende Subnetz dürfen nicht verwendet werden!

Die Host-Nummer 0 und die aus nur Einsen bestehende Host-Nummer dürfen nicht verwendet werden!



### Hinweis

Wenn die IP-Adresse über die Konfigurationssoftware KS2000 eingestellt wurde, muss auch die Subnetz-Maske mit der Konfigurationssoftware KS2000 geändert werden.

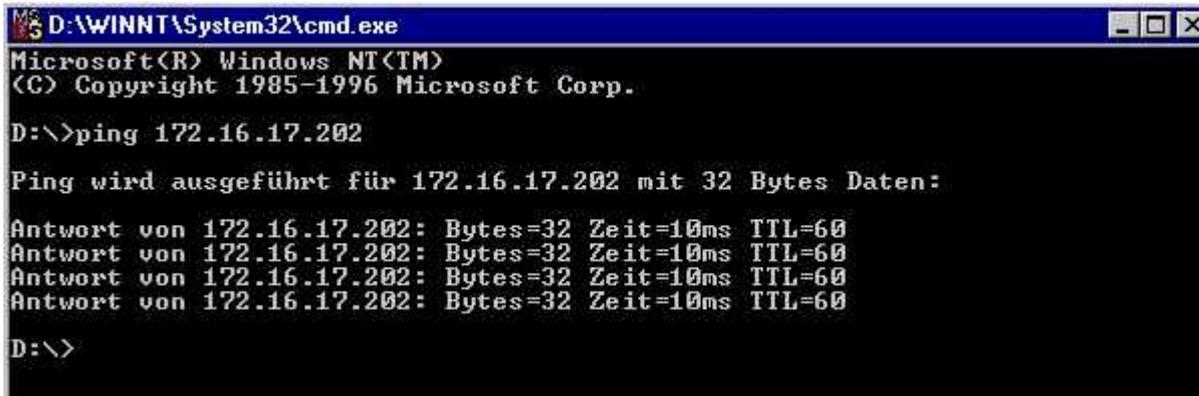
Bei ARP-Adressierung wird anhand der IP-Adresse die dazugehörige Standard Subnetz-Maske eingetragen.

Bei BootP und DHCP wird die Subnetz-Maske mit vom Server übertragen.

Feldbuskomponenten: Testen der IP-Adresse

## Test der IP-Adresse

Zum Testen der IP-Adresse können Sie das Kommando *Ping* benutzen.



```

D:\WINNT\System32\cmd.exe
Microsoft(R) Windows NT(TM)
(C) Copyright 1985-1996 Microsoft Corp.
D:\>ping 172.16.17.202
Ping wird ausgeführt für 172.16.17.202 mit 32 Bytes Daten:
Antwort von 172.16.17.202: Bytes=32 Zeit=10ms TTL=60
D:\>
  
```

Feldbuskomponenten: Ethernet

## Auslesen der MAC-ID

Gehen Sie folgendermaßen vor, um die MAC-ID auszulesen:

- Ändern Sie Ihre IP-Adresse des Buskopplers auf 172.16.x.x.  
Die IP-Adresse ist im Auslieferungszustand 172.16.17.255 (DIP-Schalter 1 bis 8 auf ON).
- Schicken Sie ein Ping auf die IP-Adresse 172.16.17.255
- Lesen Sie die MAC-ID mit *arp -a* aus.

Feldbuskomponenten: TwinCAT

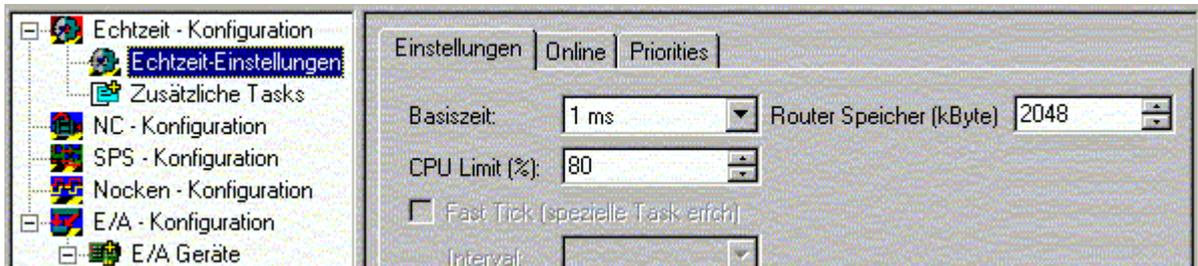
## Konfiguration mit dem System Manager

Tragen Sie im TwinCAT System Manager unter Geräte eine allgemeine Ethernet Karte ein. Sollten die Buskoppler schon am Netzwerk angeschlossen sein und eine IP-Adresse besitzen, können Sie diese auch einlesen. Dabei werden automatisch alle Buskoppler mit Busklemmen und Konfiguration hochgeladen. Diese können Sie dann nach Ihren Bedürfnissen anpassen.



## Inbetriebnahme bei 6 oder mehr Ethernet-Knoten

Wenn Sie im System-Manager mehr als 5 Knoten eintragen, müssen Sie den Router-Speicher erhöhen. Diese Einstellung finden Sie im System-Manager unter *Real Time Settings*. Tragen Sie dort für jeden Buskoppler 350 kByte ein. Dies ergibt z.B. für 10 Buskoppler 3,5 MB (10 x 350 kByte).



Hinweis: Der Router-Speicher ist vom Arbeitsspeicher ihres PCs abhängig und darf nicht beliebig hoch gesetzt werden.

Berechnung des maximal zulässigen Router-Speichers:

Max Routerspeicher [MB] = (1 MB + ((Arbeitsspeicher[MB] - 4 MB) x 0,4))

Beispiel:

Ihr PC hat 128 MB Arbeitsspeicher:

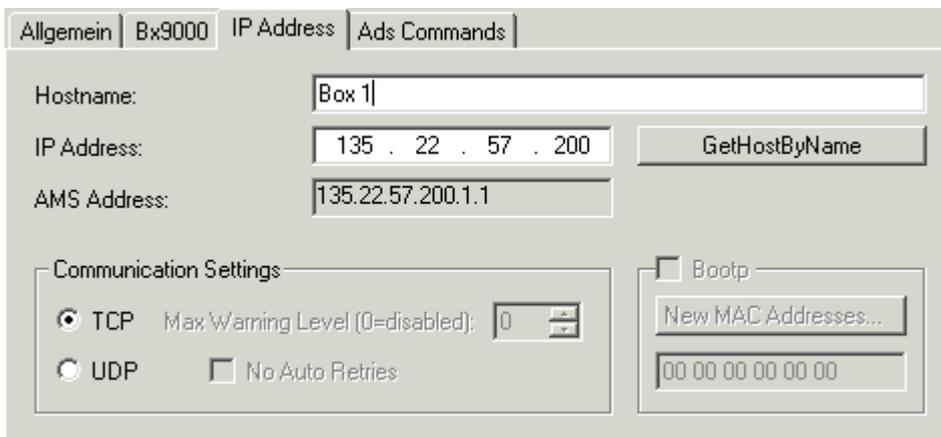
$1\text{MB} + (128\text{MB} - 4\text{MB}) \times 0,4 = 50,6\text{ MB}$

Die maximale Größe des Router-Speichers darf bei 128 MB Arbeitsspeicher 50,6 MB nicht überschreiten.

TwinCAT System-Manager: Reference

## Karteireiter IP Address

Ethernet-Komponenten benötigen im Netzwerk eine eindeutige IP-Adresse. Nachfolgend ist der, für die Konfiguration des Feldbusknoten im TwinCAT-System notwendige, Einstelldialog beschrieben. Der gezeigte Karteireiter erscheint nachdem Sie im System-Manager unter Ihrer Ethernetkarte mit der rechten Maustaste (Box Anfügen) einen BK9000, BC9000 oder eines anderen Ethernet-Feldbusgerätes ausgewählt haben.



### Hostname

Hier kann der Name der Buskopplerstation editiert werden.

**IP-Address**

Tragen Sie hier die IP-Adresse des Buskopplers ein.

**GetHostByName**

Durch Betätigung dieser Schaltfläche kann (ab BK9000 Firmware-Version B2) eine IP-Adresse von einem Windows 2000 DHCP (Dynamic Host Configuration Protocol) Server bezogen werden.

**AMS Address**

Gibt die AMS-Net ID des Buskopplers an. Sie wird automatisch durch hinzufügen von zwei zusätzlichen Bytes (z.B.: ".1.1") aus der IP-Adresse generiert.

**BootP**

Diese Checkbox ist aktivierbar, falls der Beckhoff BootP-Server installiert und gestartet ist (ab TwinCAT Version 2.8)

**New MAC Addresses**

Bei gestartetem [Beckhoff BootP-Server](#) können durch Betätigung dieser Schaltfläche die MAC-Adressen (Media Access Controller) neu angeschalteter Ethernet-Feldbusgerät angezeigt werden. Falls nur ein neues Ethernet-Feldbusgerät angeschlossen ist, wird dementsprechend auch nur eine neue MAC-Adresse angezeigt, der Sie dann im oben beschriebenen Feld *IP Address* die gewünschte IP-Adresse zuweisen können. Bei Anschluss jeder weiteren Ethernet-Feldbuskomponenten, können Sie diesen diesen Vorgang auf dem hier beschriebenen Karteireiter des neuen Geräts wiederholen.

**Communication Settings**

Einstellungen zur Kommunikation über IP (Internet Protocol).

**TCP**

Wenn diese Option aktiv ist (default), wird die Kommunikation über TCP (*Transmission Control Protocol*) abgewickelt. Dies bedeutet, das die Telegrammpakete vom und zum Buskoppler über einen gesicherten (bestätigten) Mechanismus ausgetauscht werden.

**UDP**

Wenn diese Option ausgewählt wurde, wird die Kommunikation über UDP (*User Datagram Protocol*) abgewickelt. Im Gegensatz zum TCP sind hierbei die einzelnen Telegrammpakete nicht gesichert, d.h., sie werden nicht jeweils nach Erhalt vom Empfänger bestätigt. Zerstörte oder in der Reihenfolge vertauschte Telegrammpakete werden nicht erneut gesendet oder sortiert. Der Vorteil des UDP besteht aus kalkulierbaren Telegrammlaufzeiten, da nicht erst auf eine Antwort vom Empfänger gewartet werden muss. Man kann daher bei UDP von einer bedingten Echtzeitfähigkeit reden.

**Max. Warning Level**

Diese Einstelloption ist nur bei *UDP* aktiv. Hier kann der maximale Wert des Fehlerzählers eingetragen werden. Bei Erreichen des eingestellten Maximalwertes wird nicht mehr auf ein Antworttelegramm des Feldbusknotens gewartet. Stattdessen werden nur noch Lesetelegramme, basierend auf einer höheren Zykluszeit, zum Feldbusknoten geschickt.

**No Auto Retries**

Diese Checkbox ist auch nur bei aktivierter Option *UDP* anwählbar. Bei Erreichen des eingestellten Wertes unter *Max. Warning Level*, ist bei aktivierter Checkbox ein Ausführen der Funktion *Online Reset* notwendig.

TwinCAT System-Manager: Konfiguration Busklemmen Controller

## Karteireiter *PLC*

Bei Busklemmen Controllern (BC) können unterhalb von *SPS-Variablen* spezielle Datenaustauschvariablen für die feldbusbasierende Kommunikation zwischen PC und BC definiert werden (siehe [Beispielkonfiguration eines Bus Controllers](#)).

Bei Anwahl eines BCs in der Baumansicht des TwinCAT System Managers, erscheint auf der rechten Seite u.a. der folgende Karteireiter:

### PLC Cycle Time

Die reservierte Zykluszeit der Kleinststeuerung. In dieser Zeitspanne werden vom BC keine Daten an den Feldbus übergeben.

### Background Time

Zeit für die Bearbeitung der Kommunikation über den Feldbus (u.a. auch Programm-Download, Debugging usw.).

### PLC Var. Offs IN

Startoffset der Datenaustauschvariablen im Eingangsbereich des BC-Prozessabbilds

### PLC Var. Offs OUT

Start-Offset der Datenaustauschvariablen im Ausgangsbereich des BC-Prozessabbilds

### AMS Response during download

Kann aus Kompatibilitätsgründen bei alten Busklemmen Controllern (bei BC3100 und BC2000) deaktiviert werden. Neuere Bus Controller Firmware-Stände quittieren den Erhalt einzelner Download-Sequenzen. Daher muss hier die Checkbox aktiviert sein (bei BC9000 in jedem Fall).

### Start with Configuration Error

Auch wenn der konfigurierte BC9000 nicht am Netzwerk angeschlossen ist, lässt sich, sofern diese Checkbox aktiviert ist, das TwinCAT-System starten.

### Retain Data

Store Retain Data: Aktiviert den Support permanenter Daten im Busklemmen Controller.

Size: Größe des permanenten Merkerbereichs innerhalb des Busklemmen Controllers (%Mx0 .. %Mxyy).

### KBus Update

Double Cycle:: Aktiviert die Ausführung von E/A-Zyklen (KBus Update) innerhalb der BC-Station vor und nach jedem SPS-Zyklus.

Before PLC: Vor Ausführung des SPS-Zyklus wird ein KBus Update ausgeführt.

After PLC: Nach Ausführung des SPS-Zyklus wird ein KBus Update ausgeführt.

### Firmware Update (via COMx)

Über diese Funktion kann das Update der BC-Firmware ausgeführt werden.

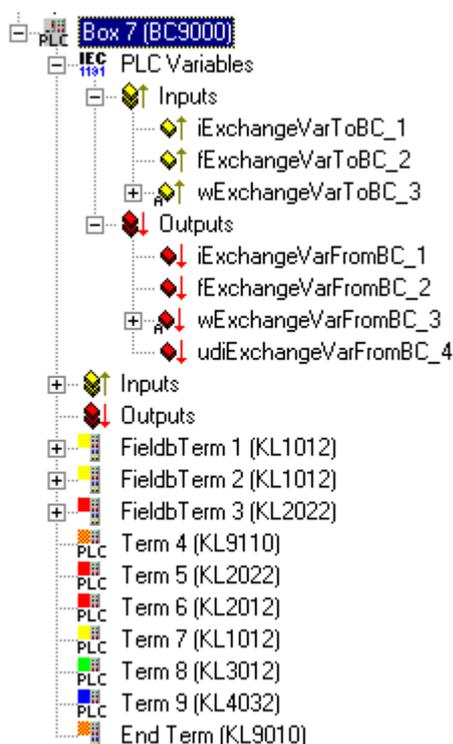
### Karteireiter *Variable*

Nach Anfügen von Variablen unterhalb des Baueintrags *SPS-Variablen*, erscheint auf der rechten Seite der zur Variablen gehörende Dialog, wie in der Rubrik *Variablen Informationen* beschrieben.

Wählen Sie sinnvolle Namen für die Datenaustauschvariablen, da diese bei Export von Variablen-Informationen als Variablen-Namen im lokalen BC-Prozessabbild verwendet werden. Anschließend können Sie, die Datenaustauschvariablen mit vorhandenen Tasks verknüpfen.

### Beispielkonfiguration eines Bus Controllers

Im folgenden Bild sind sowohl Datenaustauschvariablen, wie auch die Feldbus-Variablen dargestellt. Der BC verwaltet die Klemmen 4 bis 9 lokal.



Feldbuskomponenten: ModbusTCP

## Konfiguration für ModbusTCP

Für die Kommunikation über ModbusTCP ist keine besondere Konfiguration notwendig. Die Default-Einstellung beträgt 16 Byte Ein- und Ausgänge (SPS-Variablen). Diese sind über die Modbus-Funktionen erreichbar.

Feldbuskomponenten: Programmierung

## TwinCAT PLC

Das Beckhoff TwinCAT Software-System verwandelt jeden kompatiblen PC in eine Echtzeitsteuerung mit Multi-SPS-System, NC-Achsregelung, Programmierumgebung und Bedienstation. Die Programmierumgebung von TwinCAT wird auch für die Programmierung des BC9000 genutzt. Wenn Sie TwinCAT PLC (WinNT4, Win2000, XP) installiert haben, so können Sie die Ethernet-Verbindung oder die serielle Schnittstelle für Software-Download und Debugging verwenden. Sollten Sie mit TwinCAT BC (auch kompatibel zu Win95/98/ME) programmieren, so erfolgt die Verbindung zum BC9000 ausschließlich über die serielle Schnittstelle.

TwinCAT E/A oder TwinCAT PLC können auch als Ethernet-Master (Host) genutzt werden, um Prozessdaten mit dem Busklemmen-Controller auszutauschen. TwinCAT stellt ihnen hierzu den System-Manager als Konfigurationstool sowie die Treiber und das ADS-Protokoll zu Verfügung.

Feldbuskomponenten: Programmierung

## PLC Cycle Time

Die PLC Cycle Time bestimmt die Wiederholungssequenz des Programms. Diese Zeit ist nicht deterministisch. Das bedeutet, dass die PLC Cycle Time sich je nach Programm vergrößern und über die eingestellte Cycle Time hinweg ansteigen kann. Benötigt das Programm weniger Zeit als eingestellt, wird es mit der eingestellten Cycle Time wiederholt und der Rest der übrigen Zeit mit Background Time ausgefüllt.

In der Background Time läuft die Bearbeitung des Ethernets und der seriellen Schnittstelle. Diese sollte auf ca. 20 bis 50 % der PLC Cycle Time eingestellt sein.

Um das System zu optimieren misst man die *Mittlere Zykluszeit*. Dies finden Sie im PLC Control unter dem Menüpunkt *Online/Koppler*. Der dort ermittelte Zeit gibt man einen Aufschlag von 50% und trägt die Zeit als PLC Cycle Time ein. Stellen Sie die Background Time auf 20 bis 50% der PLC Cycle Time ein.



Die Programmlaufzeit besteht aus der Programmbearbeitung und der K-Bus-Zeit. Bevor das Programm aufgerufen wird, fährt der Busklemmen Controller ein K-Bus-Update und liest die aktuellen Eingänge ein. Nach der Programmbearbeitung wird ein zweites K-Bus-Update um die Ausgänge zu schreiben. Die K-Bus-Zeit ist abhängig von Anzahl und Art der gesteckten Busklemmen.

Man kann die Programmlaufzeit verkürzen indem man nur einen K-Bus-Zyklus fährt und zwar entweder vor oder nach der Programmbearbeitung. Dann werden in einem K-Bus-Zyklus Eingänge gelesen und Ausgänge geschrieben. Sie können diese Einstellungen über die Konfigurationssoftware KS2000 oder per ADS vornehmen.



Feldbuskomponenten: Lokierte Merker

## Remanente und persistente Daten

Im höheren Speicherbereich gibt es spezielle Merker für remanente und persistente Daten.

### Remanente Daten

Die remanenten Daten befinden sich im lokierten Merkerbereich. In der Default-Einstellung sind 64 Byte remanent, d.h. von %MB0 bis %MB63. Dieser Bereich lässt sich maximal bis auf 4 kByte vergrößern. Dabei ist darauf zu achten, dass sich die Task-Zeit mit der Menge der remanenten Daten erhöht (512 Byte ca. 0,5 ms). Die Einstellungen für die remanenten Daten können Sie über den System-Manager (TwinCAT) oder mit der Konfigurationssoftware KS2000 vornehmen (Tabelle 1 Register 15 Default 64

max. 4 kByte).

## Persistente Daten

Die persistenten Daten sind noch eine Stufe stabiler, d.h. diese Daten bleiben sogar bei einem Programmdownload erhalten. Die Anzahl der persistenten Daten muss kleiner oder gleich der Anzahl der remanenten Daten sein! Die persistenten Daten liegen wie die remanenten Daten im lokierten Merkerbereich (%MBxx). Die Einstellungen für die persistenten Daten können Sie mit der Konfigurationssoftware KS2000 vornehmen (Tabelle 1 Register 18 Default 0 max. 4 kByte).

## Beispiel

400 Byte remanente Daten, davon sollen 200 Byte persistent sein

Register 15 400 (%MB200 - %MB399)

Register 18 200 (%MB0 - %MB199)

## Zykluszeitmessung

In dem Merkerbereich befindet sich eine Zeitmessung mit einer Auflösung von einer Millisekunde pro Digit. Der Datentyp ist UDINT (unsigned double integer). Dieser Wert kann vom Programm überschrieben werden, um einen Abgleich zu einer Steuerung zu übernehmen. Der Überlauf geschieht nach ca. 48 Tagen.

Merkerbyte	Bedeutung	
%MB4088-4091	4 Byte	
	Datentyp	UDINT
		0..4,22 Mrd ms / 0.. ca. 48 Tage
	Auflösung	1 ms / Digit

## Diagnose

Es gibt die Möglichkeit, die Diagnosedaten im Busklemmen Controller zu lesen. Diese Information liegen im lokierten Merkerbereich.

Merkerbyte	Bedeutung	
%MB4092-4093	Bit 0	Watchdog ADS Kommunikation
	Bit 1	Watchdog ModbusTCP Kommunikation
	Bit 1-14	reserviert
	Bit 15	Keine Verbindung zum Switch: die LED <i>Link</i> leuchtet nicht (Im Auslieferungszustand ist die Übertragung dieser Fehlerinformation deaktiviert. Setzen Sie zum aktivieren in Tabelle 100, <a href="#">Register 25</a> auf 1 <sub>bin</sub> ).
%MB4094-4095	Bit 0	K-Bus-Fehler
	Bit 1	Konfigurationsfehler
	Bit 2-15	reserviert

Feldbuskomponenten: Programmierung

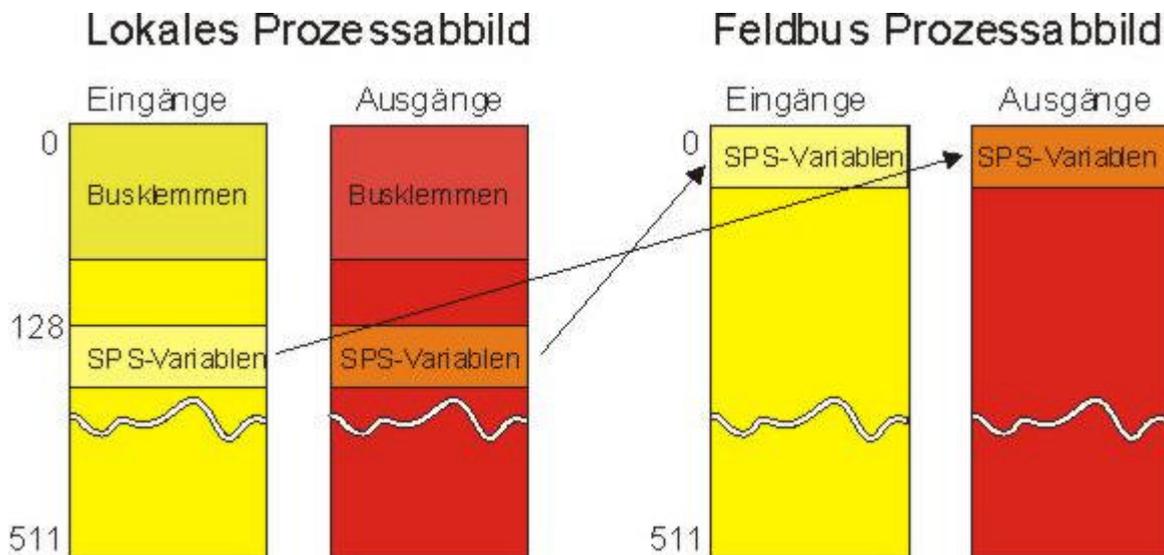
# SPS-Variablen

Die SPS-Variablen dienen als Schnittstelle zwischen dem lokalen Prozessabbild des Bus Controllers und dem Feldbusprozessabbild für eine übergeordnete Steuerung. Als erstes soll dies anhand der Default-Einstellung erläutert werden.

## Busklemmen im lokalen Prozessabbild des BCs

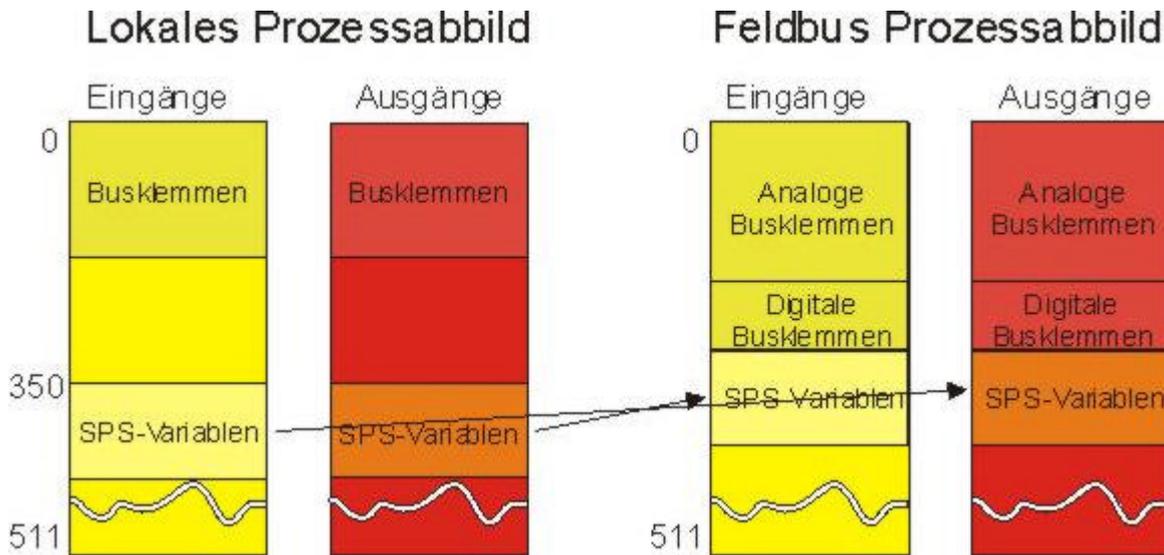
Alle angeschlossenen Busklemmen sind dem lokalen Prozessabbild zugewiesen. Die SPS-Variablen befinden sich ab der Adresse 128. Sie können diese Anfangsadresse sowie die Länge dieser Daten (Default 16 Byte) verändern.

Daten, die von einer überlagerten Steuerung gelesen werden sollen, schreibt man in das Ausgangsprozessabbild. Diese Daten sind für die überlagerte Steuerung Eingangsdaten. Daten, die von der überlagerten Steuerung zum BC übertragen werden, sind für diese Steuerung Ausgangsdaten und für den BC Eingangsdaten. Dies soll die folgende Grafik verdeutlichen.



## Busklemmen der übergeordneten Steuerung zuweisen

Sie können die Busklemmen auch direkt der überlagerten Steuerung zuweisen. Im Feldbus Prozessabbild sieht das ganze dann so aus, dass sich erst die analogen Busklemmen in dieses Prozessabbild mappen. Als nächstes folgen die digitalen Busklemmen und als letztes dann die SPS-Variablen.



Achten Sie darauf, dass alle benutzten Prozessabbilder nicht größer als 512 Byte sein dürfen.

Feldbuskomponenten: Mapping

## Mapping der Busklemmen

Die genaue Belegung der byteorientierten Busklemmen entnehmen Sie bitte der Konfigurations-Anleitung zur jeweiligen Busklemme. Diese Dokumentation finden Sie auf der Beckhoff CD *Products & Solutions* oder im Internet unter [www.beckhoff.com](http://www.beckhoff.com).

byteorientierte Busklemmen	bitorientierte Busklemmen
KL1501	KL10xx, KL11xx, KL12xx, KL17xx
KL25xx	KL20xx, KL21xx, KL22xx, KL26xx, KL27xx
KL3xxx	
KL4xxx	
KL5xxx	
KL6xxx	
KL8xxx	
	KL9110, KL9160, KL9210, KL9260,

Feldbuskomponenten: Lokales Prozessabbild

## Lokales Prozessabbild

Alle angeschlossenen Busklemmen werden in der Default-Einstellung dem lokalen Prozessabbild zugewiesen. Das Mapping im Busklemmen Controller erfolgt nach folgender Gesetzmäßigkeit: Erst alle komplexen Busklemmen, in der Reihenfolge wie diese gesteckt sind, und dann die digitalen Busklemmen, die zu einem Byte aufgefüllt werden. Das Default-Mapping der komplexen Busklemmen ist:

- komplette Auswertung
- Intel-Format

- Wordalignment

Beispiel:

1 x BCxxxx  
 1 1 x KL1012  
 2 1 x KL1104  
 3 1 x KL2012  
 4 1 x KL2034  
 5 1 x KL1502  
 6 1 x KL3002  
 7 1 x KL4002  
 8 1 x KL6001  
 9 1 x KL9010

Busklemme	Position	Eingangsabbild	Ausgangsabbild
KL1502	5	%IB0...%IB5	%QB0...%QB5
KL3002	6	%IB6...%IB13	%QB6...%QB13
KL4002	7	%IB14...%IB21	%QB14...%QB21
KL6001	8	%IB22...%IB29	%QB22...%QB29
KL1012	1	%IX30.0..30.1	-
KL1104	2	%IX30.1..30.5	-
KL2012	3	-	%QX30.0..30.1
KL2034	4	-	%QX30.2..30.5
KL9010	9	-	-

Falls Sie nicht wissen, auf welcher Adresse sich die Busklemmen befinden, die Sie der lokalen SPS (BC9000) zugewiesen haben:

Konfigurieren Sie im System-Manager Ihre Hardware-Konfiguration.

Nachdem Sie alle Busklemmen und SPS-Variablen eingetragen haben klicken

Sie im Hardware-Baum mit der rechten Maustaste auf den BC9000 und wählen Sie den Menü Punkt *Export Variablen Info...*. Es wird eine Datei gespeichert, die Sie im System-Manager unter *Projekt Importieren* einfügen können. Nun haben Sie unter den globalen Variablen den Eintrag

*TwinCAT Import*, indem Sie alle Variablen finden, die Sie der lokalen SPS (BC9000) zugeordnet haben.



**Hinweis**

Feldbuskomponenten: Feldbus Prozessabbild

## Feldbus Prozessabbild

Bei allen BCxxxx sind in der Default-Einstellung 16 Byte Ein- und Ausgänge dem Feldbus zugewiesen. Diese Variablen heißen *SPS-Variablen*. Sie liegen im BCxxxx Prozessabbild in der Default-Einstellung ab Adresse 128. Diese Adresse kann verschoben werden. Die Einstellung kann mit der Konfigurationssoftware KS2000 oder bei einigen BCs per Start-Up-Parameter geändert werden.

Feldbuskomponenten: Feldbus Prozessabbild

# ADS-Prozessabbild

Bei allen BCxxxx sind in der Default-Einstellung 16 Byte Ein- und Ausgänge dem Feldbus zugewiesen. Diese Variablen heißen SPS-Variablen und liegen in der Default-Einstellung des BCxxxx im Prozessabbild ab Adresse 128. Diese Einstellung kann mit der Konfigurationssoftware KS2000 oder bei einigen BCs per Start-UP-Parameter geändert werden.

Feldbuskomponenten: Feldbus Prozessabbild

## ModbusTCP Prozessabbild

Das Prozessabbild des ModbusTCP unterscheidet grundsätzlich digitale und byteorientierte (wortorientierte) Signale. Anhand von Beispielen soll dies verdeutlicht werden.

### Beispiel 1: (Default-Einstellung)

Alle angeschlossenen Busklemmen sind der lokalen PLC zugeordnet. Für die Kommunikation mit dem ModbusTCP Master stehen die 16 Byte SPS-Variablen zur Verfügung. SPS-Variablen zählen zu den byteorientierten Datenpaketen und können deshalb auch nicht mit den digitalen Modbus-Funktionen erfasst werden, auch wenn diese Variablen im BCxxxx digital gemappt sind. Probleme bereitet meist die Sichtweise der Ein - oder Ausgänge. Was für den BCxxxx bei den SPS-Variablen Ausgänge sind, sind für den Modbus Master Eingänge und umgekehrt. Eine Änderung der Default Einstellung kann mit der KS2000 Software geändert werden ([Registerbeschreibung Tabelle1](#)).

Eingänge lokale SPS			Ausgänge lokale SPS		
Modbus-Funktion	Modbus Adresse/Offset	Lokale Variable	Modbus-Funktion	Modbus Adresse/Offset	Lokale Variable
3 lesen	0x800..807	% IB 128..% IB 143	3 lesen	0x000..007	% QB 128..% QB 143
			4 lesen	0x000..007	% QB 128..% QB 143
6 schreiben	0x800..807	% IB 128..% IB 143			
16 schreiben	0x800..807	% IB 128..% IB 143			
23 lesen/schreiben	0x800..807	% IB 128..% IB 143	23 lesen/schreiben	0x000..007	% QB 128..% QB 143

### Beispiel 2:

Zusätzlich können die Busklemmen einmal dem BC zugewiesen werden oder der lokalen SPS. Das verkompliziert das Prozessabbild zusätzlich. Die Regel, wie sich das Prozessabbild zusammenstellt, ist allerdings einfach.

- Eintrag aller byteorientierten Busklemmen (zum Beispiel die analoge Busklemmen)
- Eintrag aller digitalen Busklemmen (auf 16 Bit wird aufgerundet)
- Eintrag der SPS-Variablen

Feldbuskomponenten: Programmierung

## Übertragung über die serielle Schnittstelle

Alles was mit TwinCAT BC programmiert und parametrier wird, kann auch mit TwinCAT geschehen. Die Besonderheit von TwinCAT BC ist, das es sowohl auf Windows 95/98/ME als auch auf NT/2000/XP funktioniert. Mit TwinCAT BC ist allerdings kein Programmdownload über Ethernet möglich.

Jeder Busklemmen-Controller kann über die RS232 Schnittstelle des PC programmiert werden. Verwenden Sie hierzu ein spezielles Kabel, das im Lieferumfang von TwinCAT BC und der Konfigurationssoftware KS2000 enthalten ist.

Wählen Sie im TwinCat PLC-Control die serielle Schnittstelle an.

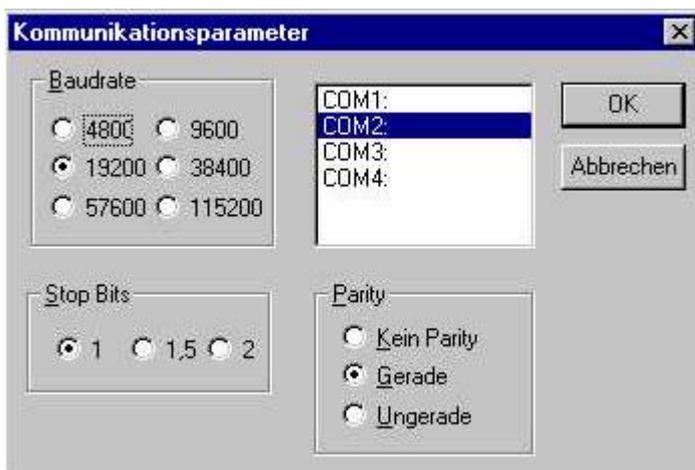


Unter Online/Kommunikationsparameter finden Sie im PLC-Control die Einstellungen zur seriellen Schnittstelle, Portnummer, Baud-Rate usw. Der Busklemmen Controller benötigt folgende Einstellung:

Baud-Rate: 19200

Stop Bits: 1

Parity: Gerade



Feldbuskomponenten: Programmierung

## Übertragung über Ethernet

TwinCAT bietet die Eigenschaft, das Anwenderprogramm über das Ethernet auf den BC9000 zu

übertragen. Im PLC Control kann nach dem Sichern in der Registry und einem Restart des TwinCAT-Systems, als Zielsystem der BC9000 angewählt werden. Erforderlich ist der TwinCAT-Level TwinCAT PLC.

## Initialisieren des BC9000

Um den Koppler im PLC Control auswählen zu können, muss er dem System erst einmal bekannt gemacht werden. Es gibt zwei Möglichkeiten dies zu tun. Welche Möglichkeit Sie nutzen, hängt von Ihren Erfordernissen ab.

### 1. Sie nutzen TwinCAT PLC nur zum Programmieren.

Klicken Sie mit der linken Maustaste auf das TwinCAT-Icon rechts unten in der Bildschirmecke.



Wähle Sie den Menüpunkt Eigenschaften aus.



Tragen Sie den BC9000 unter dem Menüpunkt *AMS Router* als Remote-Computer ein. Klicken Sie dafür auf *Hinzufügen*.

Remote Verbindung hinzufügen

Name:

AMS Net Id:

Adresse:

Transport:   Langsame Verbindung

Tragen Sie den Namen ein, dieser kann frei gewählt werden. Die AMS-Net-ID ist die IP-Adresse ergänzt um ".1.1". Diese Adresse ist die eingestellte IP-Adresse des Buskopplers. Bestätigen Sie mit OK. Nach dem Eintrag der gewünschten Buskoppler müssen Sie das TwinCAT-System Starten (das TwinCAT Icon ist grün). Nun ist der Ethernet-Koppler dem System bekannt gemacht worden und kann jetzt im PLC Control ausgewählt werden.

## 2. Sie nutzen TwinCAT für den zyklischen Datentransport

Unbenannt - TwinCAT System Manager

Datei Bearbeiten Aktionen Ansicht Optionen ?

Allgemein Bx9000 PLC IP Address Ads Commands

Hostname:

IP Address:

AMS Address:

Communication Settings

TCP Max Warning Level:

UDP  No Auto Retries

Bootps

Nummer	Klemmenbezeichnung	Typ	Eing. Größe	Ausg. Größe
1	Klemme 2 (KL1002)	KL1002	(0.2)	(0.0)
2	Klemme 3 (KL1002)	KL1002	(0.2)	(0.0)
3	Klemme 4 (KL1002)	KL1002	(0.2)	(0.0)
4	Klemme 5 (KL2012)	KL2012	(0.0)	(0.2)
5	Klemme 6 (KL2012)	KL2012	(0.0)	(0.2)
6	Klemme 7 (KL2012)	KL2012	(0.0)	(0.2)
7	End-Klemme (KL9010)	KL9010	0.0	0.0

Starten Sie den System-Manager und tragen dort Ihre Konfiguration ein. Im Unterschied zur ersten Methode wird einem Start des TwinCats diese Konfiguration zum Koppler heruntergeschrieben (zum Beispiel Busklemmen Mapping, PLC Zykluszeit, Anzahl der Retain Daten etc.). Andere Einstellungen, die mit der Konfigurationssoftware KS2000 gemacht wurden werden überschrieben.

## PLC Control

Beim Neustart des TwinCAT PLC Control fragt TwinCAT nach der Zielplattform, d.h. auf welchem Gerät später das Anwender-Programm laufen soll. TwinCAT bietet als Steuerung zwei Zielplattformen, den PC oder den Busklemmen-Controller.

Für die Übertragung zum Busklemmen-Controller stehen Ihnen zwei Wege zur Verfügung:

- AMS für die Kommunikation über den Feldbus
- das serielle Kabel für die Kommunikation über die [RS232-Schnittstelle](#) des PCs und die Programmierschnittstelle des Busklemmen-Controllers



Wählen Sie nach Erstellung Ihres Programms unter der Symbolleiste *Online* das Zielsystem aus. Hierzu muss TwinCAT gestartet sein. Im Beispiel ist dies die Ethernet-Karte mit der Box 1 und die Run-Time 1 des Busklemmen-Controllers.



Feldbuskomponenten: Bibliothek

## Bibliotheken

Für die Busklemmen Controller (Buskoppler mit SPS-Funktionalität, Bezeichnung BCxxxx) gibt es verschiedene Bibliotheken ([siehe TwinCAT InfoSys](#)).

### TwinCAT PLC Library: System BC

Die Bibliothek beinhaltet Funktionsbausteine für den Zugriff auf Systemfunktionen der Busklemmen Controller (BCxxxx).

### TwinCAT PLC Library: Utilities BC

Die Bibliothek beinhaltet nützliche Funktionsbausteine für die Busklemmen Controller (BCxxxx). Neben den RTC-Bausteinen beinhaltet die Bibliothek einen Funktionsbaustein für die Dekodierung des DCF-77

Zeitsignals und einige Konvertierungsfunktionen. Intern werden die Systemfunktionen der Busklemmen Controller aufgerufen.

#### **TwinCAT PLC Library: Helper BC**

Die Bibliothek *PlcHelperBC.lib6* beinhaltet einige Funktionen für den direkten Zugriff auf Speicherbereiche des SPS-Laufzeitsystems der Busklemmen Controller (BCxxxx).

#### **TwinCAT PLC Library: ADS BC**

Die Bibliothek beinhaltet Funktionsbausteine für eine azyklische Client-Server-Kommunikation zwischen einem Busklemmen Controller **BC9xxx** und anderen ADS-Geräten im Netzwerk. Die Funktionsweise der Bausteine unterscheidet sich im Wesentlichen nicht von denen für das SPS-Laufzeitsystem auf dem PC.

#### **TwinCAT PLC Library: Ereignisgesteuerte Übertragungsbausteine BC**

Die vorliegenden Funktionsbausteine vereinfachen den ereignisgesteuerten Datenaustausch zwischen Busklemmen Controllern und/oder anderen ADS-Geräten (TwinCAT NC, Busklemmen Controller, ...). Die Funktionsbausteine `FB_WriteXXXOnDelta()` führen einen Schreibvorgang aus, wenn das Eingangssignal einen bestimmten Grenzwert über-/ bzw. unterschritten hat. Wie oft das Eingangssignal überprüft werden soll, kann eingestellt werden. Durch das ereignisgesteuerte Schreiben der Daten wird die Belastung der Feldbusses minimiert. Tritt bei der Übertragung ein Fehler auf, so wird der Vorgang solange wiederholt, bis die Verbindung wieder vorhanden ist. Als Quell- und Zielvariablen sind alle, in der TwinCAT PLC unterstützten Datentypen zugelassen. Symbolnamen werden ebenfalls unterstützt. Zur Überwachung einzelner Kommunikationspartner stehen Watchdog-Bausteine zur Verfügung. Der zu überwachende Teilnehmer versendet zyklisch einen inkrementierenden Zähler. Bei dem Empfänger wird überprüft, ob sich der Zählerstand innerhalb einer bestimmten Zeit ändert.

#### **TwinCAT PLC Library: ModbusTCP BC**

Die Bibliothek beinhaltet Funktionsbausteine die einen Datenaustausch zwischen dem Busklemmen Controller (BC9xxx) und einem Remote-Partner auf dem MODBUS/TCP-Port ermöglichen.

#### **TwinCAT PLC Library: SMTP BC**

Die Bibliothek beinhaltet nützliche Funktionsbausteine für den Busklemmen Controller (BC9xxx) um das SMTP-Protokoll (Simple Mail Transfer Protocol) zu nutzen.

Feldbuskomponenten: Ethernet

## **Ethernet**

Ethernet wurde ursprünglich von DEC, Intel und Xerox (als DIX-Standard) für die Datenübertragung zwischen Bürogeräten entwickelt. Heute versteht man darunter meist die Spezifikation *IEEE 802.3 CSMA/CD*, die 1985 veröffentlicht wurde. Diese Technologie ist durch ihren weltweiten Einsatz und die hohen Stückzahlen überall erhältlich und sehr preiswert.. Eine Anbindung an vorhandene Netze kann so problemlos realisiert werden.

Mittlerweile gibt es die verschiedensten Übertragungsmedien: Koaxialkabel (10Base5), Lichtwellenleiter (10BaseF) oder verdrehte Zweidrahtleitung (10BaseT) mit Schirmung (STP) oder ohne Schirmung (UTP). Mit Ethernet lassen sich verschiedenen Topologien aufbauen wie Ring, Linie oder Stern.

Ethernet transportiert Ethernet-Pakete von einem Sender zu einem oder mehreren Empfängern. Diese

Übertragung verläuft ohne Quittung und ohne Wiederholung von verlorenen Paketen. Für die sichere Daten-Kommunikation stehen Protokolle wie TCP/IP zu Verfügung, die auf Ethernet aufsetzen.

## **MAC-ID**

Sender und Empfänger von Ethernet-Paketen werden über die MAC-ID adressiert. Die MAC-ID ist eine 6 Byte großer Identifikations-Code, der eineindeutig, d.h. für jedes Ethernet-Gerät weltweit unterschiedlich ist. Die MAC-ID besteht aus zwei Teilen. Der erste Teil (d.h. die ersten 3 Byte) ist eine Herstellerkennung. Die Firma Beckhoff hat die Kennung 00 01 05. Die nächsten 3 Byte werden durch den Hersteller vergeben und entsprechen einer eindeutigen Seriennummer. Die MAC-ID kann zum Beispiel beim BootP-Protokoll zum Einstellen der TCP/IP-Nummer verwendet werden. Dafür wird ein Telegramm zum entsprechenden Knoten geschickt, das die Informationen wie Name oder TCP/IP-Nummer beinhaltet. Sie können die MAC-ID mit der Konfigurationssoftware KS2000 auslesen.

## **Internet-Protokoll (IP)**

Die Grundlage der Datenkommunikation ist das Internet-Protokoll (IP). IP transportiert Datenpakete von einem Teilnehmer zu einem anderen, der sich im gleichen oder in einem anderen Netz befinden kann. IP kümmert sich dabei um das Adress-Management (Finden und Zuordnen der MAC-IDs), die Segmentierung und die Pfadsuche (Routing). Wie das Ethernet-Protokoll gewährleistet auch IP keinen gesicherten Transport der Daten; Datenpakete können verloren gehen oder in ihrer Reihenfolge vertauscht werden.

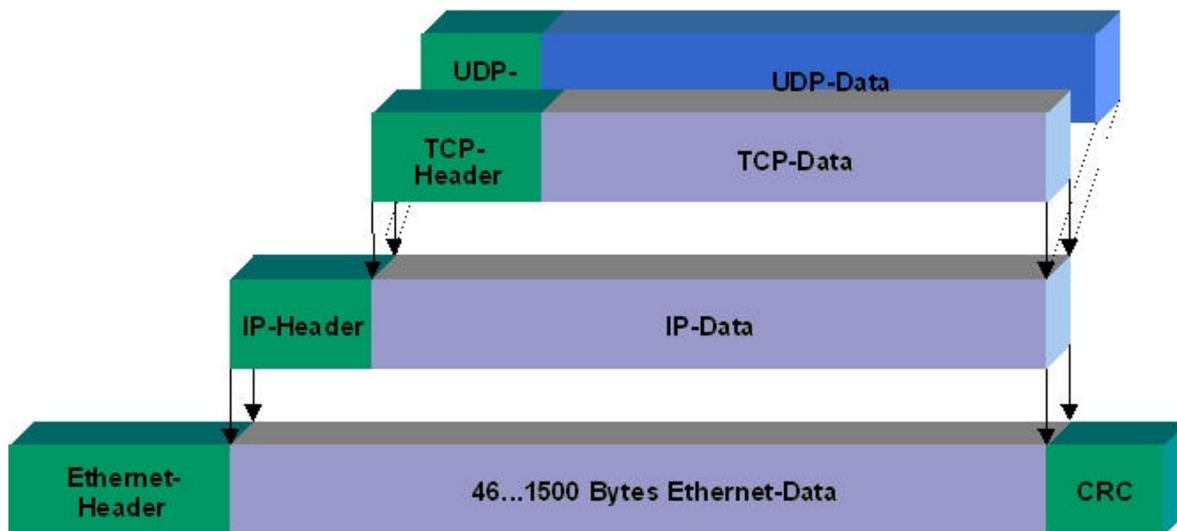
Für einen standardisierten, gesicherten Informationsaustausch zwischen beliebig vielen verschiedenen Netzwerken wurde TCP/IP entwickelt. Dabei ist TCP/IP weitgehend unabhängig von der verwendeten Hard- und Software. Oftmals als ein Begriff verwendet, handelt es sich hierbei um mehrere aufeinander aufgesetzte Protokolle: z.B. IP, TCP, UDP, ARP und ICMP.

## **Transmission Control Protocol (TCP)**

Das auf IP aufsetzende Transmission Control Protocol (TCP) ist ein verbindungsorientiertes Transport-Protokoll. Es umfasst Fehlererkennungs- und Behandlungsmechanismen. Verlorene Telegramme werden wiederholt.

## **User Datagram Protocol (UDP)**

UDP ist ein verbindungsloses Transport-Protokoll. Es gibt keine Kontrollmechanismen beim Datenaustausch zwischen Sender und Empfänger. Dadurch resultiert eine schneller Verarbeitungsgeschwindigkeit als zum Beispiel bei TCP. Eine Prüfung ob das Telegramm angekommen ist muss vom übergeordneten Protokoll durchgeführt werden.

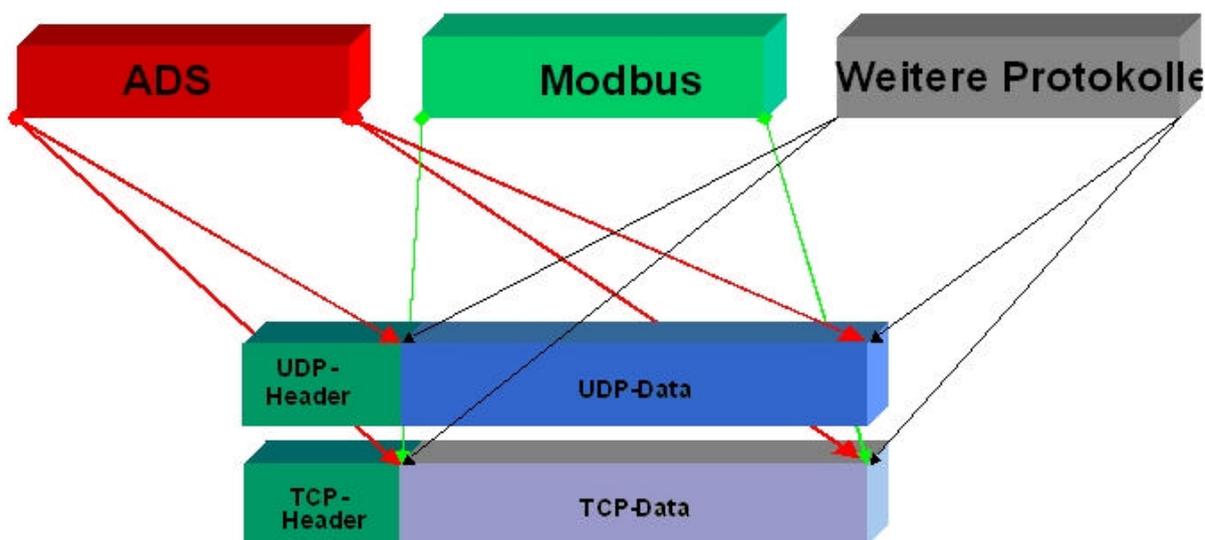


## Auf TCP/IP und UDP/IP aufsetzende Protokolle

Auf TCP/IP bzw. UDP können folgende Protokolle aufsetzen:

- ADS
- ModbusTCP

Beide Protokolle sind parallel auf dem Buskoppler implementiert, so dass für die Aktivierung der Protokolle keine Konfiguration nötig ist.



ADS setzt wahlweise auf TCP oder UDP auf, während ModbusTCP stets auf TCP/IP basiert.

Feldbuskomponenten: Ethernet

## Topologie

Mit 10BaseT und 100BaseT werden mehrere Stationen im Ethernet-Standard sternförmig verbunden.

## Stern-Topologie

Ein Stern-LAN besteht in der einfachsten Netzform aus einzelnen Punkt-zu-Punkt-Verbindungen. Alle Nachrichten laufen über einen zentralen Knoten (Hub oder Switch), der je nach Zieladresse die Informationen an den gewünschten Empfänger weitergibt.

## Baum-Topologie

Eine Baum Topologie besteht aus mehreren verbundenen Stern-Topologien. Sobald mehrere Hubs oder Switches im Netz vorhanden sind, ist eine Baumtopologie vorhanden. Ideal ist es, die Verbindungen zwischen den Sternkopplern besonders breitbandig auszuführen, da diese den meisten Datenverkehr transportieren. Beim Aufbau von Baum-Topologien ist die Repeater-Regel zu beachten, die auch als 5-4-3-Repeater-Regel bezeichnet wird. Maximal zwei Repeater-Paare (bzw. Hub-Paare) dürfen im Übertragungsweg zwischen zwei beliebigen Stationen sein, sofern sie nicht durch Bridges, Switches oder Router getrennt sind. Ein Übertragungsweg kann aus maximal fünf Segmenten und vier Repeater-Sets (zwei Repeater-Paaren) bestehen. Dabei können bis zu drei Segmente Koax-Segmente sein, an denen die Stationen angeschlossen sind, die restlichen Segmente müssen Punkt-zu-Punkt-Verbindungen sein, die auch als IRL-Verbindung (Inter Repeater Link) bezeichnet werden.

## Verkabelungsrichtlinien

Allgemeine Richtlinien für den Netzwerkaufbau eines LAN gibt die *Strukturierte Verkabelung* vor. Darin sind maximal zulässige Kabellängen für die Gelände-, Gebäude- und Etagenverkabelung festgelegt. In den Standards EN 50173, ISO 11801 und TIA 568-A normiert, bildet die *Strukturierte Verkabelung* die Grundlage für eine zukunftsweisende, anwendungsunabhängige und wirtschaftliche Netzwerk-Infrastruktur. Die Verkabelungsstandards definieren einen Geltungsbereich mit einer geographischen Ausdehnung von bis zu 3 km und für eine Bürofläche von bis zu 1 Mio. Quadratmeter mit 50 bis 50.000 Endgeräten. Darüber beschreiben sie Empfehlungen für den Aufbau eines Verkabelungssystems. Abhängig von der gewählten Topologie, den unter Industriebedingungen eingesetzten Übertragungsmedien und Koppelmodulen sowie von dem Einsatz von Komponenten verschiedener Hersteller in einem Netz können sich abweichende Angaben ergeben. Die Angaben verstehen sich hier deshalb lediglich als Empfehlungen.

Feldbuskomponenten: BK9000, BC9000

## Reaktionszeit des BC9000

ADS	Zeit [ms]
TCP	23 bis 50 ms + Task-Zeit auf dem BC9000 (Jitter)
UDP	2 bis 3 ms + Task-Zeit auf dem BC9000 (Jitter)

Modbus	Zeit [ms]
TCP	12 bis 15 ms + Task-Zeit auf dem BC9000 (Jitter) (Default)
Fast TCP*	1 bis 3 ms + Task-Zeit auf dem BC9000 (Jitter)

\* ab Firmware B7, Tabelle 100 Register 29 - "1" Fast TCP / "0" TCP ([Siehe Modbus Interface](#))

## Reaktionszeit BK9000

--	--

ADS	Zeit [ms]
TCP	23 bis 50 ms
UDP	2 bis 3 ms

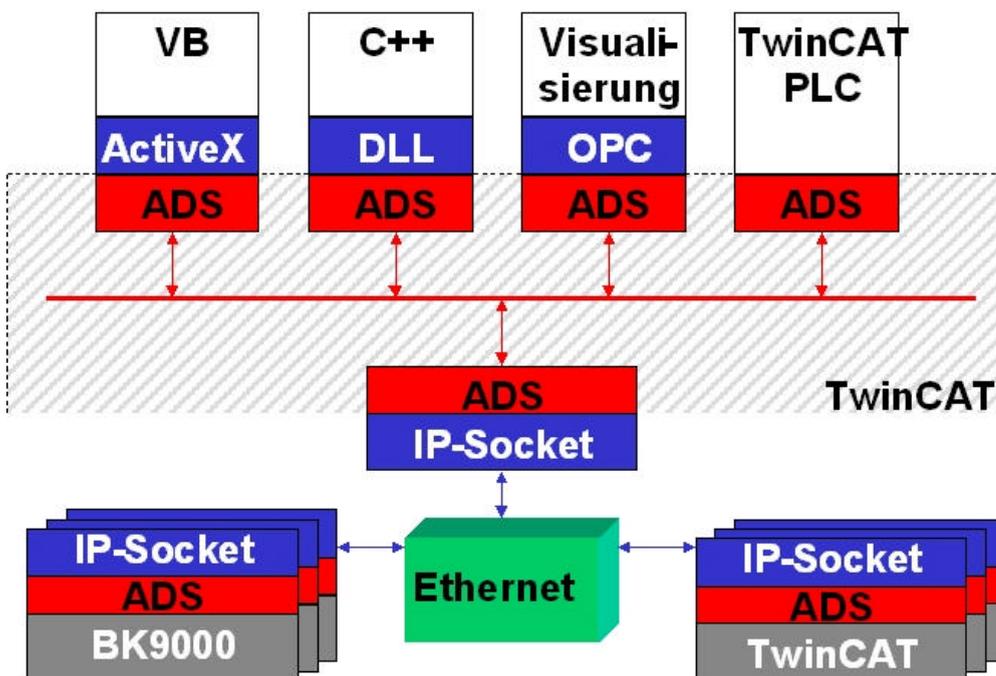
Modbus	Zeit [ms]
TCP	12 bis 15 ms (Default)
Fast TCP*	1 bis 3 ms

\* ab Firmware B5, Tabelle 100 Register 29 - "1" Fast TCP / "0" TCP ([Siehe Modbus Interface](#))

Feldbuskomponenten: ADS

## ADS-Protokoll

Das ADS-Protokoll (ADS: Automation Device Specification) ist eine Transportschicht innerhalb des Beckhoff TwinCAT Systems. Es ist für den Datenaustausch der verschiedenen Software-Module entwickelt worden, zum Beispiel für die Kommunikation zwischen der NC und der PLC. Mit diesem Protokoll hat man die Freiheit von jedem Punkt im TwinCAT mit anderen Tools zu kommunizieren. Wird eine Kommunikation zu anderen PC oder Geräten benötigt setzt das ADS-Protokoll auf TCP/IP auf. Somit ist es in einem vernetzten System möglich, an alle Daten von einem beliebigen Punkt zu kommen.



Das ADS-Protokoll wird auf das TCP/IP- oder UDP/IP-Protokoll aufgesetzt. Es ermöglicht dem Benutzer innerhalb des Beckhoff-Systems über nahezu beliebige Verbindungswege mit allen angeschlossenen Geräten zu kommunizieren und diese zu parametrieren. Außerhalb des Beckhoff-Systems stehen verschiedene Wege offen, um mit anderen Software-Tools Daten auszutauschen.

## Software-Schnittstellen

### ADS-OCX

Das ADS-OCX ist eine Active-X-Komponente und bietet eine Standardschnittstelle zum Beispiel zu Visual Basic, Delphi, u.s.w.

## ADS-DLL

Sie können die ADS-DLL (DLL: Dynamic Link Library) in Ihr C-Programm einbinden.

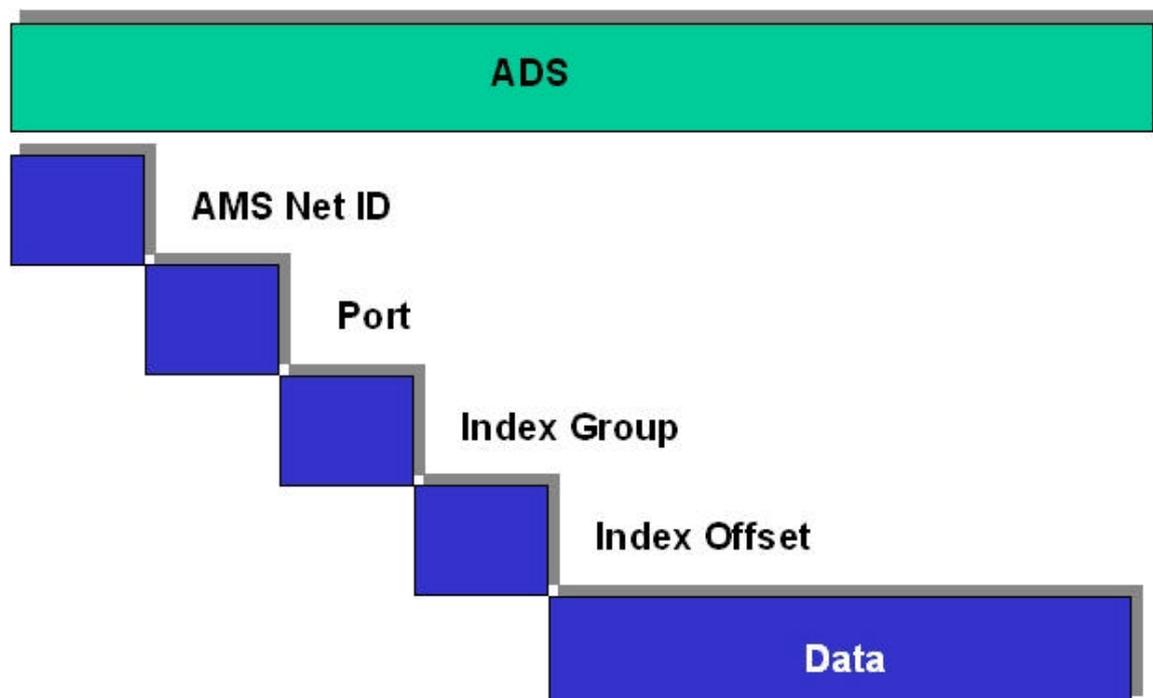
## OPC

Die OPC-Schnittstelle ist eine genormte Standardschnittstelle für die Kommunikation in der Automatisierungstechnik. Beckhoff bietet hierfür einen OPC-Server an.

Feldbuskomponenten: ADS

# ADS-Protokoll

Die ADS-Funktionen bieten die Möglichkeit, direkt vom PC auf Informationen des Buskopplers zuzugreifen. Dafür können ADS-Funktionsbausteine im TwinCAT PLC Control verwendet werden. Die Funktionsbausteine sind in der Bibliothek *PLCSystem.lib* enthalten. Genauso ist es möglich, die ADS-Funktionen von AdsOCX, ADSDLL oder OPC aufzurufen. Über die ADS-Portnummer 300 sind alle Daten und über die ADS-Portnummer 100 Zugriffe auf die Register des Buskopplers und der Klemmen möglich.



## AMSNetID

Die AMSNetID beschreibt das anzusprechende Gerät. Diese wird aus der eingestellten TCP/IP-Adresse und zusätzlichen 2 Byte erstellt. Diese zusätzlichen 2 Byte bestehen aus "1.1" und sind nicht veränderbar.

Beispiel:

IP-Adresse 172.16.17.128

AMS Net ID 172.16.17.128.1.1

## Port-Nummer

Die Portnummer unterscheidet im angeschlossenen Gerät Unterelemente.

Port 100: Registerzugriff

Port 300: Prozessdaten Feldbus

Port 800: lokales Prozessdaten (nur BC90x0)

## Index Group

Die Index Group unterscheidet innerhalb eines Ports verschiedene Daten.

## Index Offset

Gibt den Offset an, ab welchem Byte gelesen oder geschrieben werden soll.

## Len

Gibt die Länge der Daten in Byte an, die gelesen bzw. geschrieben werden sollen.

## TCP-Port-Nummer

Die TCP-Port-Nummer beträgt für das ADS-Protokoll 48898 oder 0xBF02.

Feldbuskomponenten: ADS

# ADS-Dienste

## Prozessdaten-Port 300

Der Zugriff auf die Feldbusdaten erfolgt über die ADS-Portnummer 300. Diese Daten werden von einem Watchdog überwacht. Sollte innerhalb von 1000 ms kein weiteres Telegramm eintreffen, werden die Ausgänge in den sicheren Zustand geschaltet.

Index Group	Bedeutung	Index Offset (Wertebereich)
0xF020	Input - Eingänge	0..511
0xF030	Output - Ausgänge	0..511

## Lokales Prozessabbild Port 800 (nur BC9000)

Im lokalen Prozessabbild können Daten gelesen und geschrieben werden. Sollten Ausgänge geschrieben werden muss darauf geachtet werden, dass diese von der lokalen SPS nicht verwendet werden, da die lokale Steuerung diese Werte überschreibt. Die Daten sind nicht an einen Watchdog gebunden und müssen und dürfen daher nicht für Ausgänge verwendet werden, die im Fehlerfall ausgeschaltet werden müssen.

Index Group	Bedeutung	Index Offset (Wertebereich)
0xF020	Input - Eingänge	0..511
0xF030	Output - Ausgänge	0..511
0x4020	Merker (nur BC9000)	0..4096

## Dienste des ADS

**AdsServerAdsState\*\***

Datentyp (only Read)	Bedeutung
String	Start - die lokale PLC läuft Stop - die lokale PLC ist im Stop

**AdsServerDeviceState\*\***

Datentyp (only Read)	Bedeutung
INT	0 - Start - die lokale PLC läuft 1 - Stop - die lokale PLC ist im Stop

**AdsServerType\*\***

Datentyp (only Read)	Bedeutung
String	Coupler_PLC

**Register Port 100**

Die ADS-Portnummer ist beim BK/BC9000 für die Register-Kommunikation fest vorgegeben und beträgt 100.

Index Group	Index Offset (Wertebereich)		Bedeutung
	Hi-Word	Lo-Word	
0	0..127	0..255	Register des Buskopplers Hi-Word Tabellennummer des Buskopplers Lo-Word Registernummer der Tabelle
1-64	0-3	1-64	Register der Busklemmen Hi-Word Kanalnummer Lo-Word Registernummer der Busklemme

**Hinweis**

Beachten Sie beim Lesen der Register, dass der Time-Out beim ADS-Baustein auf eine Zeit größer 1 sec eingestellt wird.

**Hinweis**

Beachten Sie beim Schreiben auf die Register, dass das Passwort gesetzt wird (siehe Dokumentation zur entsprechenden Busklemme).

**Zugriffskontrolle\***

Die Tabelle *AMS Net-Id* ermöglicht über AMS eine Zugriffskontrolle auf den BC9000. Sobald Einträge in dieser Tabelle vorhanden sind, können nur noch die eingetragenen AMS-Teilnehmer auf den BC9000 zugreifen. Des weiteren wird hier explizit eine Zuordnung der MAC-ID zur IP-Adresse des Knotens gemacht.

Die Tabelle *AMS Net-Id* kann mit ADS Write Kommandos gefüllt werden:  
Es sind maximal 10 Einträge möglich.

#### Aufbau der Struktur

AMS Net ID	Größe
AMS Net ID	6 Byte
IP-Adresse	4 Byte
Reserve	2 Byte
Reserve	4 Byte
Reserve	4 Byte

Zugriff erfolgt überPortnummer: 10.000

Indexgroup: 700

Index Offset (Write)	Bemerkung	Daten
0	Hinzufügen eines Eintrags	Datenstruktur 20 Byte
1	Löschen eines Eintrags	-
2	Löschen aller Einträge	-
10	Speichern der Tabelle im Flash	-

Index Offset (Read)	Bemerkung	Daten
0	Anzahl der Einträge	2 Byte
1..10	Eintrag n (1..10)	Datenstruktur 20 Byte



#### Hinweis

Der erste Eintrag muss der Teilnehmer sein der in die Tabelle schreibt, da diese Einstellungen sofort gültig sind. Achten Sie darauf das die Einstellungen alle Richtig sind. Die Tabelle kann ebenfalls gelöscht werden wenn nur die Endklemme gesteckt ist und die DIP-Schalter 1 bis 7 auf ON gesetzt sind.

\* ab Firmware B6

\*\* ab Firmware B7

Siehe [Beispiel](#)

Feldbuskomponenten: Beispiel

## Beispiel Zugriffskontrolle

```
PROGRAM MAIN
VAR
    i: INT;
```



```

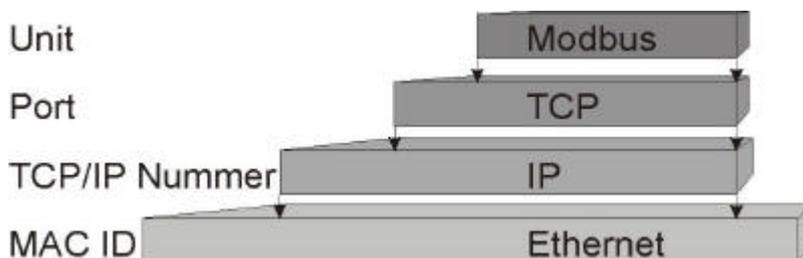
                i:=200;
ELSE
    i:=502;
END_IF
502:  fbADSWRITE1(NETID:='172.16.17.151.1.1' , PORT:=10000 , IDXGRP:=700 , IDXOFFS:
    IF NOT fbADSWrite1.Busy THEN
        i:=i+1;
    END_IF
503:  fbADSWRITE1(Write:=FALSE);
    IF fbADSWRITE1.Err THEN
        i:=200;
    ELSE
        i:=100;
    END_IF
END_CASE

```

Feldbuskomponenten: ModbusTCP

## ModbusTCP-Protokoll

Das Ethernet-Protokoll wird über die MAC-ID adressiert. Der Anwender braucht sich meist um diese Adresse nicht zu kümmern. Die IP-Nummer ist 4 Byte groß und muss vom Anwender auf dem Buskoppler und in der Anwendung parametrisiert werden. Die TCP-Port ist bei ModbusTCP auf 502 festgelegt. Die UNIT ist bei ModbusTCP frei wählbar und braucht vom Anwender nicht konfiguriert werden.



### TCP-Port-Nummer

Die TCP-Port-Nummer von ModbusTCP wurde auf den Wert 502 standardisiert. Diese TCP-Port-Nummer kann auch mit der Konfigurationssoftware KS2000 geändert werden: Tabelle100, Register 26.

### Modbus-Unit

Die Unit ist nur von Bedeutung, wenn von mehreren Stationen auf den BK9000 zugegriffen wird. Die erste Unit, die auf den Buskoppler zugreift, hat einen Schreibzugriff. Alle anderen Teilnehmer können nur lesend auf den BK9000 zugreifen.

## ModbusTCP-Protokoll

Byte	Name	Beschreibung
------	------	--------------

0	Transaction identifier	wird vom Slave zurückgesendet
1	Transaction identifier	wird vom Slave zurückgesendet
2	Protocol identifier	immer 0
3	Protocol identifier	immer 0
4	Length field	0 (wenn die Nachricht kleiner 256 Byte ist)
5	Length field	Anzahl der folgenden Bytes
6	UNIT identifizier	Dies ist wie eine Protokoll Nummer zu betrachten und wird vom Slave zurückgegeben
7	Modbus	es folgt das Modbus-Protokoll beginnend mit der Funktion

Feldbuskomponenten: ModbusTCP

## Modbus TCP-Interface

Adresse		Beschreibung		
0x0000 0x00FF		Prozessdaten-Interface Eingänge		
0x0800 0x08FF		Prozessdaten-Interface Ausgänge		
0x1000 0x1006	Read only	Buskopplerkennung		
0x100A		2 Byte SPS-Interface		
0x100B		Busklemmendiagnose		
0x100C		Buskoppler Status		
0x1010		Prozessabbildlänge in Bit, analoge Ausgänge (ohne SPS Variablen)		
0x1011		Prozessabbildlänge in Bit, analoge Eingänge (ohne SPS Variablen)		
0x1012		Prozessabbildlänge in Bit, digitale Ausgänge		
0x1013		Prozessabbildlänge in Bit, digitale Eingänge		
0x1020		Watchdog, aktuelle Zeit in [ms]		
0x110A		2 Byte SPS Interface		
0x110B		Busklemmendiagnose		
0x1120	Watchdog vordefinierte Zeit in [ms] (Default: 1000)			
0x1121	Watchdog Reset Register			
0x1122	Read/Write	Art des Watchdogs	1	Telegramm Watchdog (Default)
			0	Schreibtelegramm Watchdog
0x1123		ModbusTCP Mode**	1	Fast Modbus
			0	Normal Modbus (Default)
0x4000 0x47FF	Merkerbereich (%MB..)*			

\* nur BC9000

\*\* Firmware BC9000 B7 - BK9000 B5

### Watchdog

Der Watchdog ist im Auslieferungszustand aktiviert. Nach dem ersten Schreibtelegramm wird der Watchdog scharf geschaltet und bei jedem empfangenden Telegramm dieses Teilnehmers getriggert. Andere Teilnehmer haben auf den Watchdog keinen Einfluss. Eine zweite Möglichkeit, die eine schärfere Bedingung des Watchdogs darstellt, ist, dass der Watchdog nur nach jedem Schreibtelegramm getriggert

wird. Dafür Schreiben Sie in das Register 0x1122 eine Null (Default "1").

Der Watchdog kann deaktiviert werden in dem im Offset 0x1120 eine Null geschrieben wird. Das Watchdog Register darf nur dann beschrieben werden, wenn der Watchdog noch nicht aktiv ist. Die Daten in diesem Register bleiben gespeichert.

### Watchdog-Register

Sollte der Watchdog auf Ihren Slave abgelaufen sein können sie diesen durch ein zweimaliges beschreiben des Registers 0x1121 zurücksetzen. Dazu muss folgendes in das Register geschrieben werden: 0xBECF 0xAFFE. Dies kann mit der Funktion 6 oder der Funktion 16 geschehen.

### Statusregister des Buskopplers

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FB	-	-	-	-	-	-	-	-	-	-	-	-	-	CNF	KB

### Legende

Bit	Name	Wert	Beschreibung
15	FB	1 <sub>bin</sub>	Feldbusfehler, Watchdog abgelaufen
14...2	-	-	reserviert
1	CNF	1 <sub>bin</sub>	Buskoppler Konfigurationsfehler
0	KB	1 <sub>bin</sub>	Busklemmenfehler

### ModbusTCP Mode

Der Fast Modbus Mode sollte nur in kleinen lokalen Netzwerken eingesetzt werden. In der Default Einstellung ist der Fast ModbusTCP nicht aktiviert. Werden Probleme mit dieser Kommunikationsart festgestellt, ist der Buskoppler auf die "normale" ModbusTCP Kommunikation umzustellen. Per Modbus Interface Offset 0x1123 ist dieser Mode einzustellen. Nach der Änderung ist ein Reset (zum Beispiel mit der ModbusTCP Funktion 8) des Buskopplers notwendig. Im Fast Modbus Mode ist es nicht gestattet, mehr als ein Modbus Dienst in einem Ethernet Frame zu versenden.

### 2 Byte SPS-Interface

Mit dem 2 Byte SPS-Interface können Register der komplexen Klemmen sowie Register des Busklemmen-Controllers gelesen bzw. beschrieben werden. Die Register der komplexen Klemmen sind in der Dokumentation zur jeweiligen Klemme beschreiben. Über die Register des Buskopplers können z.B. K-Busdiagnosedaten, der Klemmenaufbau oder Zykluszeiten gelesen sowie die programmierte Konfiguration beschrieben werden. Weiterhin kann darüber auch ein manueller K-Bus-Reset durchgeführt werden. Das 2 Byte SPS-Interface benötigt je zwei Bytes in den Ein- und Ausgangsdaten, über die ein spezielles Protokoll abgewickelt wird. Eine Beschreibung des 2 Byte SPS-Interface, der verfügbaren Register im Buskopplers sowie Funktionsbausteine für verschiedene SPS-Systeme, die das 2 Byte SPS-Interface unterstützen, kann auf Anfrage geliefert werden.

### 2 Byte Diagnose-Interface

Die Fehlermeldungen der Klemmen können mit dem 2 Byte Diagnose-Interface gesendet werden. Dazu ist aber die K-Busdiagnose zu aktivieren. Das 2 Byte Diagnose-Interface belegt je zwei Bytes in den Ein- und Ausgangsdaten, über die ein spezielles Protokoll durchgeführt wird. Eine Beschreibung des 2 Byte Diagnose-Interfaces kann auf Anfrage geliefert werden.

Feldbuskomponenten: ModbusTCP

## Fehlerantwort des ModbusTCP Slaves (BK9000, BC9000, IP/IL230x-B/C900)

Wenn der Anwender dem Slave eine Anforderung oder Mitteilung sendet, die der Koppler nicht versteht, antwortet der Slave mit einer Fehlermitteilung. Diese Antwort enthält die Funktion und den Fehler-Code. Der Funktionsrückgabewert wird mit 0x80 addiert.

Code	Name	Bedeutung
1	ILLEGAL FUNKTION	Nicht implementierte Modbus-Funktion
2	ILLEGAL DATA ADDRESS	Ungültige Adresse oder Länge
3	ILLEGAL DATA VALUE	Ungültige Parameter - Diagnosefunktionen - falsches Register
4	SLAVE DEVICE ERROR	Watchdog- oder K-Bus-Fehler
6	SLAVE DEVICE BUSY	Es wird schon von einem anderen IP-Teilnehmer Ausgangs-Daten empfangen

Feldbuskomponenten: ModbusTCP

## ModbusTCP-Funktionen

Die Funktionen entscheiden beim Modbus-Protokoll, ob Daten gelesen oder geschrieben werden und um welche Art von Daten es sich dabei handelt.

Funktion	Code	Beschreibung
<a href="#">Read coil status</a>	1	Lesen digitaler Ausgänge
<a href="#">Read input status</a>	2	Lesen digitaler Eingänge
<a href="#">Read holding register</a>	3	Lesen analoger Aus- und Eingänge / GPR
<a href="#">Read input register</a>	4	Lesen analoger Eingänge / GPR
<a href="#">Force single coil</a>	5	Schreiben eines digitalen Ausgangs
<a href="#">Preset single register</a>	6	Schreiben eines analogen Ausgangs / GPR
<a href="#">Diagnose</a>	8	Diagnose
<a href="#">Force multiple coils</a>	15	Schreiben mehrere digitaler Ausgänge
<a href="#">Preset multiple register</a>	16	Schreiben mehrere analoger Ausgänge / GPRs
<a href="#">Read / write registers</a>	23	Schreiben und lesen mehrere Prozedurdaten Ausgänge / GPRs

GPR (General Preset Register) - Registerstruktur Modbus Interface (siehe Anhang)

Feldbuskomponenten: ModbusTCP

## Read coil status (Funktion 1)

Mit der Funktion *Read coil status* können die gesetzten, digitalen Ausgänge gelesen werden.

In diesem Beispiel werden die ersten 10 digitalen Ausgänge gelesen. Die Start-Adresse ist Null. Im Feld *Start-Adresse* können Sie einen Offset eintragen.

### Anfrage (Query)

Byte Name	Beispiel
Funktions-Code	1
Start-Adresse high	0
Start-Adresse low	0
Anzahl high	0
Anzahl low	10

Der Feldbus-Koppler antwortet mit dem *Byte Count 2*, d.h. 2 Byte Daten kommen zurück. Die Anfrage war 10 Bits, die jetzt auf 2 Byte aufgeteilt werden. Im Ausgangsprozessabbild des BK9000 ist das dritte Bit gesetzt und der Feldbus-Koppler gibt im ersten Datenbyte eine den Wert 4 wieder.

### Antwort (Response)

Byte Name	Beispiel
Funktions-Code	1
Byte Count	2
Daten-Bit 0...7	4
Daten-Bit 8...18	0

Feldbuskomponenten: ModbusTCP

## Read input status (Funktion 2)

Mit der Funktion *Read input status* können die digitalen Eingangsdaten gelesen werden. In diesem Beispiel werden die ersten 10 digitale Eingänge gelesen. Die Start-Adresse ist Null. Im Feld *Start-Adresse* können Sie einen Offset eintragen.

### Anfrage (Query)

Byte Name	Beispiel
Funktions-Code	2
Start-Adresse high	0
Start-Adresse low	0
Anzahl high	0
Anzahl low	10

Der Feldbus-Koppler antwortet mit dem *Byte Count 2*, d.h. es kommen zwei Byte Daten zurück. Die Anfrage war 10 Bits, die jetzt auf zwei Byte aufgeteilt werden. Im Ausgangsprozessabbild des BK9000 ist das dritte Bit gesetzt und der Feldbus-Koppler gibt im ersten Datenbyte eine den Wert 4 wieder.

### Antwort (Response)

Byte Name	Beispiel
Funktions-Code	2

Byte Count	2
Daten-Bits 0..7	1
Daten-Bits 8..18	0

Feldbuskomponenten: ModbusTCP

## Read holding register (Funktion 3)

Mit der Funktion *Read holding register* können die Ein- und Ausgangsworte und die Register gelesen werden. Eingänge ab dem Offset 0 - 0xFF und Ausgänge ab den Offset 0x800 - 0x8FF.

In diesem Beispiel werden die ersten zwei analogen Ausgänge gelesen. Die analogen Ausgänge beginnen beim Offset 0x800. Die Länge bezeichnet die Anzahl, der zu lesenden Kanäle.

### Anfrage (Query)

Byte Name	Beispiel
Funktions-Code	3
Start-Adresse high	8
Start-Adresse low	0
Anzahl high	0
Anzahl low	2

Der Feldbus-Koppler antwortet mit dem Byte Count 4, d.h. es kommen 4 Byte Daten zurück. Die Anfrage waren zwei Analogkanäle, die auf zwei Worte aufgeteilt sind. Im analogen Ausgangsprozessabbild hat der erste Kanal einen Wert von 0x3FFF und der zweite Kanal einen Wert von 0x0.

### Antwort (Response)

Byte Name	Beispiel
Funktions-Code	3
Byte Count	4
Daten 1 High-Byte	63
Daten 1 Low-Byte	255
Daten 2 High-Byte	0
Daten 2 Low-Byte	0

Feldbuskomponenten: ModbusTCP

## Read input register (Funktion 4)

Die Funktion *Read input register* liest die analogen Eingänge aus.

In diesem Beispiel werden die ersten zwei analogen Eingänge des Slaves mit der Nummer 11 gelesen. Die analogen Ausgänge beginnen bei einem Offset von 0x0000. Die Länge bezeichnet die Anzahl der zu lesenden Worte. Eine KL3002 hat zwei Worte Eingangsdaten, daher ist die einzugebende Länge bei *Anzahl low* zwei.

### Anfrage (Query)

--	--

Byte Name	Beispiel
Funktions-Code	4
Start-Adresse high	0
Start-Adresse low	0
Anzahl high	0
Anzahl low	2

Der Feldbus-Koppler antwortet mit dem Byte Count 4, d.h. es kommen vier Byte Daten zurück. Die Anfrage waren zwei analog Kanäle, die jetzt auf zwei Worte aufgeteilt werden. Im analogen Eingangsprozessabbild hat der erste Kanal einen Wert von 0x0038 und der zweite Kanal einen Wert von 0x3F1B.

### Antwort (Response)

Byte Name	Beispiel
Funktions-Code	4
Byte Count	4
Daten 1 High-Byte	0
Daten 1 Low-Byte	56
Daten 2 High-Byte	63
Daten 2 Low-Byte	11

Feldbuskomponenten: ModbusTCP

## Force single coil (Funktion 5)

Mit der Funktion *Force single coil* kann ein digitaler Ausgang beschrieben werden. In diesem Beispiel wird der dritte digitale Ausgang beschrieben. Die digitalen Ausgänge beginnen bei Adresse 0x0000. Im *Daten high* steht der digitale Wert. Um den Ausgang einzuschalten muss im *Daten high* der Wert 0xFF stehen und um den Ausgang wieder auszuschalten der Wert 0x00. Im *Daten low* muss der Wert 0x00 stehen.

### Anfrage (Query)

Byte Name	Beispiel
Funktions-Code	5
Start-Adresse high	0
Start-Adresse low	2
Daten high	255
Daten low	0

Der Koppler antwortet mit dem gleichen Telegramm.

### Antwort (Response)

Byte Name	Beispiel
Funktions-Code	5
Start-Adresse high	0
Start-Adresse low	2
Daten high	255

Daten low	0
-----------	---

Feldbuskomponenten: ModbusTCP

## Preset single register (Funktion 6)

Mit der Funktion *Preset single register* kann auf das Ausgangsprozessabbild und dem Interface zugegriffen werden.

Bei der Funktion 6 wird der erste analoge Ausgang beschrieben. Die analogen Ausgänge beginnen bei einem Offset von 0x0800. Auch hier beschreibt der Offset immer ein Wort. Das heißt, dass der Offset 0x0803, das 4. Wort auf dem Ausgangsprozessabbild ist.

### Anfrage (Query)

Byte Name	Beispiel
Funktions-Code	6
Start-Adresse high	8
Start-Adresse low	0
Daten high	63
Daten low	255

Der Feldbus-Koppler antwortet mit dem gleichen Telegramm und der Bestätigung der empfangenen Werte.

### Antwort (Response)

Byte Name	Beispiel
Funktions-Code	6
Start-Adresse high	8
Start-Adresse low	0
Daten high	63
Daten low	255

Feldbuskomponenten: ModbusTCP

## Diagnose (Funktion 8)

Die Funktion Diagnose stellt eine Reihe von Tests, für die Überprüfung des Übertragungssystems zwischen dem Master und dem Slave oder für die Überprüfung der verschiedenen internen Fehlerzustände innerhalb des Slaves zur Verfügung. Ein Broadcast-Telegramm wird nicht unterstützt.

Die Funktion benutzt in der Anfrage zwei Byte, um mit einem Unterfunktions-Code die Art des durchzuführenden Tests zu definieren. Der Slave gibt in der Antwort den Funktions-Code und den Unterfunktions-Code zurück.

Die Diagnose-Anfragen benutzen ein Datenfeld von zwei Byte, um Diagnosedaten oder Steuerinformationen zum Slave zu schicken.

### Anfrage (Query)

Byte Name	Beispiel
Funktions-Code	8
Subfunktion high	0
Subfunktion low	0
Data high	2
Data low	3

### Antwort (Response)

Byte Name	Beispiel
Funktions-Code	8
Subfunktion high	0
Subfunktion low	0
Data high	2
Data low	3

## Spiegeln einer Anforderung (Unterfunktion 0)

Mit der Unterfunktion 0 werden die Daten, die der Master zum Slave schickt wieder zurückgeschickt.

## Koppler Reset (Unterfunktion 1)

Der BC9000 wird mit der Unterfunktion 1 neu initialisiert, Fehlerzähler werden zurückgesetzt und der Busklemmen-Controller führt einen Selbsttest durch. In der Zeit, in der der Busklemmen-Controller zurückgesetzt wird, werden keine Telegramme empfangen oder gesendet. Der IP-Socket wird geschlossen.



#### Hinweis

Bevor der Busklemmen-Controller neu startet schickt er noch eine Antwort mit der Unterfunktion 1 dann wird der IP-Socket geschlossen.

Unterfunktion	Daten Feld (Query)	Daten Feld (Response)
0x0001	0x0000	0x0000

## Löschen aller Zählerinhalte (Subfunktion 10)

Beim Aufruf dieser Unterfunktion löscht der Busklemmen-Controller alle Fehler Zähler.

Unterfunktion	Daten Feld (Query)	Daten Feld (Response)
0x000B	0x0000	Echo Query Daten

## Buskommunikation Antwort Zähler (Unterfunktion 11)

Gibt die Anzahl der Kommunikationsantworten an.

Unterfunktion	Daten Feld (Query)	Daten Feld (Response)
0x000C	0x0000	Wert des Zählers

## Fehlerantwort Zähler (Unterfunktion 13)

Dieser Zähler gibt die Anzahl der Fehlerantworttelegramme, die der Koppler gesendet hat, an.

Unterfunktion	Daten Feld (Query)	Daten Feld (Response)
0x000D	0x0000	Wert des Zählers

Die folgenden Funktionen beinhalten die Zählerstände für verschiedene Units. D.h. die Modbustelegramme können mit Units unterschieden werden, wenn es sich zum Beispiel um einen Zugriff auf den Buskoppler von verschiedenen Mastern aus handelt.

## Antworten des Slaves (Unterfunktion 14)

Gibt die Anzahl der gesendeten Antworten des Slaves an.

Unterfunktion	Daten Feld (Query)	Daten Feld (Response)
0x000E	0x0000	Wert des Zählers

## Anzahl der nicht geantworteten Telegramme (Unterfunktion 15)

Gibt die Anzahl der nicht gesendeten Antworten des Slaves an.

Unterfunktion	Daten Feld (Query)	Daten Feld (Response)
0x000F	0x0000	Wert des Zählers

## Anzahl der Fehlerantworten (Unterfunktion 16)

Gibt die Anzahl der gesendeten Fehlerantworten des Slaves an.

Unterfunktion	Daten Feld (Query)	Daten Feld (Response)
0x0010	0x0000	Wert des Zählers

Feldbuskomponenten: ModbusTCP

## Force multiple coils (Funktion 15)

Bei der Funktion *Force multiple coils* können gleichzeitig mehrere digitale Ausgänge gesetzt oder zurückgesetzt werden.

In diesem Beispiel werden die ersten 20 digitalen Ausgänge beschrieben. Die digitalen Ausgänge beginnen bei einem Offset von 0x0000. Hier beschreibt der Offset immer ein Bit. Der Offset 0x0003 schreibt ab dem vierten Bit auf das Ausgangsprozessabbild. Die Länge gibt die Anzahl der Bits an und der *Byte Count* setzt sich aus den zu schreibenden Bytes zusammen.

Beispiel: 20 Bits belegen 3 Byte Count (auf Byte-Grenzen aufrunden).

Die Daten-Bytes enthalten die Werte für die einzelnen Bits. In diesem Beispiel werden die Bit 0 bis 15 auf TRUE gesetzt und Bit 16 bis 23 auf FALSE.

### Anfrage (Query)

Byte Name	Beispiel
Funktions-Code	15
Start-Adresse high	0

Start-Adresse low	0
Länge high	0
Länge low	20
Byte Count	3
Daten 1 Bit 0..7	255
Daten 2 Bit 8..15	255
Daten 3 Bit 16..23	0

### Antwort (Response)

Der Buskoppler antwortet mit dem gleichen Telegramm.

Byte Name	Beispiel
Funktions-Code	15
Start-Adresse high	0
Start-Adresse low	0
Länge high	0
Länge low	20

Feldbuskomponenten: ModbusTCP

## Preset multiple register (Funktion 16)

Mit der Funktion *Preset multiple register* können mehrere analoge Ausgänge beschrieben werden. In diesem Beispiel werden die ersten zwei analogen Ausgangsworte beschrieben. Die analogen Ausgänge beginnen bei einem Offset von 0x0800. Hier beschreibt der Offset immer ein Wort. Der Offset 0x0003 schreibt ab dem vierten Wort auf das Ausgangsprozessabbild. Die Länge gibt die Anzahl der Worte an und der *Byte Count* setzt sich aus den zu schreibenden Bytes zusammen.

Beispiel: 4 Worte - entsprechen 8 Byte Count

Die Datenbytes enthalten die Werte für die analogen Ausgänge. In diesem Beispiel sind es zwei Worte, die zu beschreiben sind. Das erste Wort mit dem Wert 0x7FFF und das zweite Wort mit dem Wert 0x3FFF.

### Anfrage (Query)

Byte Name	Beispiel
Funktions-Code	16
Start-Adresse high	8
Start-Adresse low	0
Länge high	0
Länge low	2
Byte Count	4
Daten 1 Byte 1	127
Daten 1 Byte 2	255
Daten 2 Byte 1	63
Daten 2 Byte 2	255

### Antwort (Response)

Der Koppler antwortet mit der Start-Adresse und der Länge der gesendeten Worte.

Byte Name	Beispiel
Funktions-Code	16
Start-Adresse high	8
Start-Adresse low	0
Länge high	0
Länge low	2

Feldbuskomponenten: ModbusTCP

## Read / write registers (Funktion 23)

Mit der Funktion *Read / write registers* können mehrere analoge Ausgänge beschrieben und in einem Telegramm mehrere analoge Eingänge gelesen werden. In diesem Beispiel werden die ersten zwei analogen Ausgangsworte beschrieben und die ersten zwei analogen Eingänge gelesen. Die analogen Ausgänge beginnen beim Offset 0x0800 und die Eingänge ab dem Offset 0x0000. Hier beschreibt der Offset immer ein Wort. Der Offset 0x0003 schreibt ab dem 4. Wort auf das Ausgangsprozessabbild. Die Länge gibt die Anzahl der Worte an und der *Byte Count* setzt sich aus den zu schreibenden Bytes zusammen. Beispiel: 4 Worte - entsprechen 8 Byte Count

Die Datenbytes enthalten die Werte für die analogen Ausgänge. In diesem Beispiel sind es zwei Worte, die zu beschreiben sind. Das erste Wort mit dem Wert 0x3FFF und das zweite Wort mit dem Wert 0x7FFF.

### Anfrage (Query)

Byte Name	Beispiel
Funktions-Code	23
Lesen Start-Adresse high	0
Lesen Start-Adresse low	0
Lesen Länge high	0
Lesen Länge low	2
Schreiben Start-Adresse high	8
Schreiben Start-Adresse low	0
Schreiben Länge high	0
Schreiben Länge low	2
Byte Count	4
Daten 1 high	63
Daten 1 low	255
Daten 2 high	127
Daten 2 low	255

### Antwort (Response)

Der Koppler antwortet mit der Start-Adresse und der Länge der übertragenen Bytes im *Byte Count*. Es folgen die Dateninformationen. In diesem Beispiel steht im ersten Wort eine 0x0038 und im zweiten Wort eine 0x3F0B.

Byte Name	Beispiel
Funktions-Code	23
Byte Count	4
Daten 1 high	0
Daten 1 low	56
Daten 2 high	63
Daten 2 low	11

Feldbuskomponenten: ModbusTCP

## Beispiele zu ModbusTCP

In dieser Hilfe sind Beispiele zu ModbusTCP enthalten.

Einfaches Beispiel: ModbusTCP über VB6.0

[Extract ModbusTCP.zip](#)

Beispiel: ModbusTCP über VB6.0, Set und Reset des Watchdogs:

[Extract BK9000Modbus.zip](#)

Feldbuskomponenten: Register

## Registereinstellungen Tabelle 1

Die Register können mit der Konfigurationssoftware KS2000 per Dialog eingestellt oder direkt in die Register geschrieben werden.

Register	Beschreibung	Default	Busklemmen Controller
0	Byte-Offset SPS-Variablen %I...	128	BC2000, BC3100, BC4000, BC7300, BC8x00, BC9000
1	Länge der SPS-Variablen %I...	16	BC2000, BC3100, BC4000, BC7300, BC8x00, BC9000
2	Byte-Offset SPS-Variablen %Q...	128	BC2000, BC3100, BC4000, BC7300, BC8x00, BC9000
3	Länge der SPS-Variablen %Q...	16	BC2000, BC3100, BC4000, BC7300, BC8x00, BC9000
4-11	Busklemmen Zuordnung		BC2000, BC3100, BC4000, BC7300, BC8x00, BC9000
4.0-4.1	Zuordnung der 1. Busklemme	00 <sub>bin</sub>	BC2000, BC3100, BC4000, BC7300, BC8x00, BC9000
	00 <sub>bin</sub> Lokales Prozessabbild		
	10 <sub>bin</sub> Feldbus Prozessabbild kompakt		
	11 <sub>bin</sub> Feldbus Prozessabbild komplex		
4.2-4.3	Zuordnung der 2. Busklemme	00 <sub>bin</sub>	BC2000, BC3100, BC4000, BC7300, BC8x00, BC9000

m.n- l..m.n	Zuordnung der n. Busklemme	00 <sub>bin</sub>	BC2000, BC3100, BC4000, BC7300, BC8x00, BC9000	
14.2-14.3	Ausführung des K-Bus- Update	00 <sub>bin</sub>	BC2000, BC3100, BC4000, BC7300, BC8x00, BC9000	
	00 <sub>bin</sub> vor und nach der SPS			
	01 <sub>bin</sub> vor der SPS			
	10 <sub>bin</sub> nach der SPS			
12	SPS-Zykluszeit	5 ms (20 ms bei BC9000)	BC2000, BC3100, BC4000, BC7300, BC8x00, BC9000	
13	Background Time	2 ms (10 ms bei BC9000)	BC2000, BC3100, BC4000, BC7300, BC8x00, BC9000	
15	Größe des NOVRAMs (Remante Daten)	64 kB	BC2000, BC3100, BC4000, BC7300, BC8x00, BC9000	
17.0-17.7	Klemmenbus- Autorefresh:  Zyklus in [ms]	0x0000 Disable  (0x005F bei BC9000)	BC2000, BC3100, BC4000, BC7300, BC8x00, BC9000	
	17.8- 17.15	Klemmenbus- Autorefresh: max. Retries		
18	Größe des NOVRAMs Persistente Daten R18<R15	0	BC2000, BC3100, BC4000, BC7300, BC8x00, BC9000	
19	Byte-Offset digitale Eingänge %I..	0	BC2000, BC3100, BC4000, BC7300, BC8x00, BC9000	
20	Byte-Offset digitale Ausgänge %Q..	0	BC2000, BC3100, BC4000, BC7300, BC8x00, BC9000	
27	Byte-Offset KL6xx1 Busklemmen Emulation %Q...	500	BC8x00	
28	Byte-Offset KL6xx1 Busklemmen Emulation %I...	500	BC8x00	
31	Baud-Rate	2	BC8x00	
	0			38400Bd
	1			19200Bd
	2			9600Bd
	3			57600Bd
	4			1200Bd
	5			2400Bd
6	4800Bd			
32	Modus	2	BC8x00	
	0			7 Daten-Bits Even Parity
	1			7 Daten-Bits Odd Parity
	2			8 Daten-Bits No Parity
	3			8 Daten-Bits Even Parity
	8 Daten-Bits Odd			

	4	Parity		
33	Anzahl der Stoppbits		1	BC8x00
	0	ein Stoppbit		
	1	zwei Stoppbits		

Feldbuskomponenten: Register

## Registereinstellungen Tabelle 100

Die Register können mit der Konfigurationssoftware KS2000 per Dialog eingestellt oder direkt in die Register geschrieben werden.

Register	Beschreibung		Default	Busklemmen Controller
0-1	IP-Adresse		0xAC, 0x10, 0x11, 0x00	BC9000, BK9000
2-3	IP-Maske		0xFF, 0xFF, 0x00, 0x00	BC9000, BK9000
4-13	Device Name		BC9000	BC9000, BK9000
14	Watchdog AMS/ADS		1000 ms	BC9000, BK9000
15.0	0 <sub>bin</sub>	enable ModbusTCP	0 <sub>bin</sub>	BC9000, BK9000
	1 <sub>bin</sub>	disable ModbusTCP		
15.1	0 <sub>bin</sub>	enable AMS/ADS	0 <sub>bin</sub>	
	1 <sub>bin</sub>	disable AMS/ADS		
16.8	0 <sub>bin</sub>	Ethernet Modus halbduplex	1 <sub>bin</sub>	
	1 <sub>bin</sub>	Ethernet Modus vollduplex		
16.12	0 <sub>bin</sub>	Autobaud aus	1 <sub>bin</sub>	
	1 <sub>bin</sub>	Autobaud ein		
16.13	0 <sub>bin</sub>	10 MBaud	1 <sub>bin</sub>	
	1 <sub>bin</sub>	100 MBaud		
17-18	Default Gateway		0x00, 0x00, 0x00, 0x00	BC9000, BK9000
24	Watchdog ModbusTCP		1000 ms	BC9000, BK9000
25.0	Eintrag eines Feldbusfehler (Verbindung zum Switch unterbrochen) in die <a href="#">Diagnose des Merkerbereichs</a>		0 <sub>bin</sub>	BC9000
	0	Fehler wird nicht eingetragen		
	1	Fehler wird eingetragen		
26	ModbusTCP Port Nr		502	BC9000, BK9000
27	Lebensdauer ADS -Verbindung		10 s	BC9000, BK9000
28	Lebensdauer Modbus-Verbindung		10 s	BC9000, BK9000
29.0	ModbusTCP Mode		0 <sub>bin</sub>	BC9000, BK9000
	1 <sub>bin</sub>	Fast ModbusTCP		

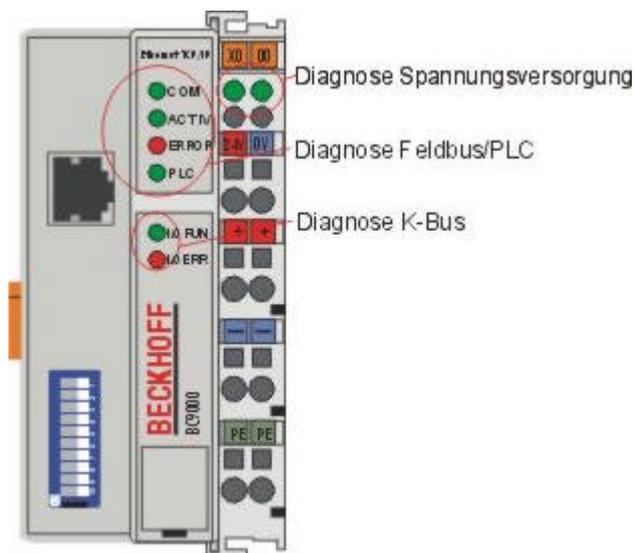
Feldbuskomponenten: Diagnose-LEDs

## Diagnose-LEDs

Nach dem Einschalten überprüft der Buskoppler sofort die angeschlossene Konfiguration. Der fehlerfreie Hochlauf wird durch das Verlöschen roten der LED *E/A ERR* signalisiert. Das Blinken der LED *I/O ERR* zeigt einen Fehler im Bereich der Klemmen an. Durch Frequenz und Anzahl des Blinkens kann der Fehler-Code ermittelt werden. Das ermöglicht eine schnelle Fehlerbeseitigung.

Zur Statusanzeige besitzt der Buskoppler zwei Gruppen von LEDs. Die obere Gruppe mit vier LEDs zeigt den Zustand des jeweiligen Feldbusses an. Die Bedeutung der Feldbusstatus-LEDs wird in den entsprechenden Kapiteln dieses Handbuches erläutert. Sie entspricht den feldbusüblichen Anzeigen.

Auf der rechten oberen Seite des Buskopplers befinden sich zwei weitere grüne LEDs zur Anzeige der Versorgungsspannung. Die linke LED zeigt die 24 V Versorgung des Buskopplers an. Die rechte LED signalisiert die Versorgung der Powerkontakte.



### LEDs zur Diagnose der Spannungsversorgung

LED	Bedeutung
Linke LED aus	Buskoppler hat keine Spannung
Rechte LED aus	Keine Spannungsversorgung 24V DC an den Powerkontakten angeschlossen

### LEDs zur Feldbus-Diagnose

LED	Ein	Aus
LINK	Physikalische Verbindung vorhanden	Keine physikalische Verbindung vorhanden
ACT	Blinkt: Busverkehr vorhanden	kein Busverkehr vorhanden (bus idle)

ERROR	LED blinkt langsam, wenn DHCP oder BootP aktiviert und der Buskoppler noch keine IP-Adresse bekommen hat LED blink schnell (5 mal, nur beim Einschalten) Der Buskoppler wurde mit ARP adressiert. Die Einstellungen auf dem DIP-Schalter sind nicht gültig.	Kein Fehler
PLC	PLC Programm ist im RUN-Modus LED blinkt, während das Bootprojekt gespeichert wird.	PLC Programm ist im Stop-Modus

## LEDs zur K-Bus-Diagnose

Fehler-Code	Fehlerargument	Beschreibung	Abhilfe
Ständiges, konstantes Blinken		EMV Probleme	- Spannungsversorgung auf Unter- oder Überspannungsspitzen kontrollieren - EMV-Maßnahmen ergreifen - Liegt ein K-Bus-Fehler vor, kann durch erneutes Starten (Aus- und Wiedereinschalten des Koppler) der Fehler lokalisiert werden
1 Impuls	0	EEPROM-Prüfsummenfehler	Hestellereinstellung mit der Konfigurationssoftware KS2000 setzen
	1	Überlauf im Code Buffer	Weniger Busklemmen stecken. Bei prog. Konfiguration sind zu viele Einträge in der Tabelle
	2	Unbekannter Datentyp	Software Update des Buskopplers notwendig
2 Impulse	0	Programmierte Konfiguration, falscher Tabelleneintrag	Programmierte Konfiguration auf Richtigkeit überprüfen
	n (n > 0)	Tabellenvergleich (Busklemme n)	Falscher Tabelleneintrag
3 Impulse	0	K-Bus-Kommandofehler	- Keine Busklemme gesteckt - Eine der Busklemmen ist defekt, angehängte Busklemmen halbieren und prüfen ob der Fehler bei den übrigen Busklemmen noch vorhanden ist. Dies weiter durchführen, bis die defekte Busklemme lokalisiert ist.
4 Impulse	0	K-Bus-Datenfehler, Bruchstelle hinter dem Buskoppler	Prüfen ob die n+1 Busklemme richtig gesteckt ist, gegebenenfalls tauschen
	n	Bruchstelle hinter Busklemme n	Kontrollieren ob die Busendklemme 9010 gesteckt ist
5 Impulse	n	K-Bus-Fehler bei Register-Kommunikation mit Busklemme n	n-te Busklemme tauschen

6 Impulse	0	Fehler bei der Initialisierung	Buskoppler tauschen
	1	Interner Datenfehler	Hardware-Reset des Buskopplers (aus - und wieder einschalten)
	2	DIP-Schalter nach einem Software-Reset verändert	Hardware-Reset des Buskopplers (aus - und wieder einschalten)
	4	DIP-Schalter für BootP falsch	1-8 auf <i>on</i> oder <i>off</i> stellen, siehe <a href="#">BootP</a>
	8	Interner Datenfehler	Hardware-Reset des Buskopplers (aus- und wieder einschalten)
	16	Fehler beim IP-Socket	Hardware-Reset des Buskopplers (aus- und wieder einschalten)
7 Impulse	0	Hinweis: Zykluszeit wurde überschritten (siehe Tab. 1 <a href="#">Register 17</a> )	Warning: Die eingestellte Zykluszeit wurde überschritten. Dieser Hinweis (blinken der LEDs) kann nur durch erneutes booten des Buskopplers gelöscht werden. Abhilfe: Zykluszeit erhöhen
9 Impulse	0	Checksummenfehler im Programm-Flash	Programm erneut zum BC übertragen
	1	Falsche oder fehlerhaft Lib implementiert	Entfernen Sie die fehlerhaft Lib
10 Impulse	n	Die Busklemme n stimmt nicht mit der Konfiguration, die beim Erstellen des Boot-Projektes existiert überein	Die n-te Busklemme überprüfen. Sollte eine n-te Busklemme gewollt eingefügt worden sein, muss das Bootprojekt gelöscht werden.
14 Impulse	n	n-te Busklemme hat das falsche Format	Buskoppler erneut Starten, falls der Fehler erneut auftritt die Busklemme tauschen.
15 Impulse	n	Anzahl der Busklemmen stimmt nicht mehr	Buskoppler erneut Starten, falls der Fehler erneut auftritt, Herstellereinstellung mit der Konfigurationssoftware KS2000 setzen
16 Impulse	n	Länge der K-Bus-Daten stimmt nicht mehr	Buskoppler erneut Starten, falls der Fehler erneut auftritt, Herstellereinstellung mit der Konfigurationssoftware KS2000 setzen

Feldbuskomponenten: Fehlerbehandlung und Diagnose

## Allgemeine Fehler

### Kein Datenaustausch nach Tausch eines Buskopplers

Sie haben den Ethernet Buskoppler getauscht und die gleiche TCP/IP-Nummer eingestellt, erhalten danach aber keinen Datenaustausch mehr.

Jedes Ethernet-Gerät hat seine eigene und eindeutige MAC-ID. Diese Nummer wird bei der Verbindung zum Ethernet-Knoten gespeichert und in einer Tabelle hinterlegt. Die Tabelle enthält die Zuordnung zwischen der MAC-ID und der TCP/IP-Nummer. Sie müssen diesen Speicher löschen und das können Sie

tun indem Sie im DOS-Fenster den Befehl >arp -d< und die TCP/IP-Nummer des Buskopplers eingeben.  
 Beispiel: >arp -d 172.16.17.203<  
 Bei aktiviertem DHCP- oder BootP-Protokoll muss auf dem DHCP- oder BootP-Server nach dem Kopplerwechsel die MAC-ID des neuen Buskopplers eingestellt werden.

**Kommunikationsfehler im Online-Zustand**

Im Online-Zustand (eingeloggt über Ethernet/AMS) kommt es nach einiger Zeit immer zu der Meldung *Kommunikationsfehler, es wird ausgeloggt.*

Der Datenverkehr über die Ethernet Schnittstelle stockt. Abhilfe:

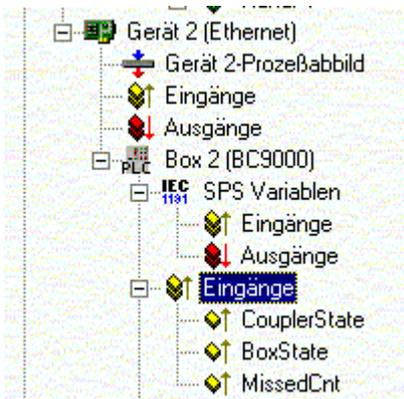
- Verringern Sie die Daten-Kommunikation.
- Halten Sie den zyklischen Datenverkehr an, oder vergrößern Sie die Task-Zeit.
- Verringern Sie die Anzahl der offenen Fenster in der Online-Darstellung.
- Loggen Sie sich über die serielle Schnittstelle ein.

Feldbuskomponenten: Diagnose

**ADS-Diagnose**

**Statuseingänge**

Im System-Manager hat man die Möglichkeit die Kommunikation der Buskoppler BK/BC9000 zu kontrollieren. Jeder Buskoppler verfügt über Status-Eingänge, die man im Hardwarebaum findet.



**Coupler State**

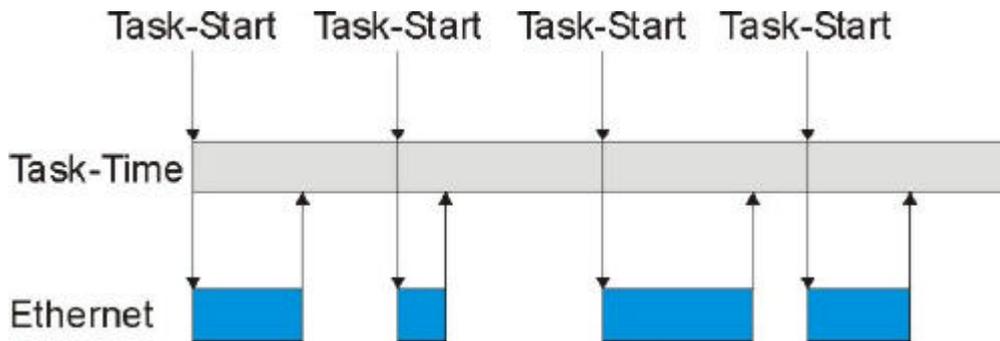
Wert	Bedeutung
0x0000	kein Fehler
0x0001	Busklemmen fehler, es liegt ein K-Bus-Fehler vor
0x0002	Konfigurationsfehler, die parametrierte Konfiguration passt nicht mit der tatsächlichen Konfiguration überein

**MissedCnt**

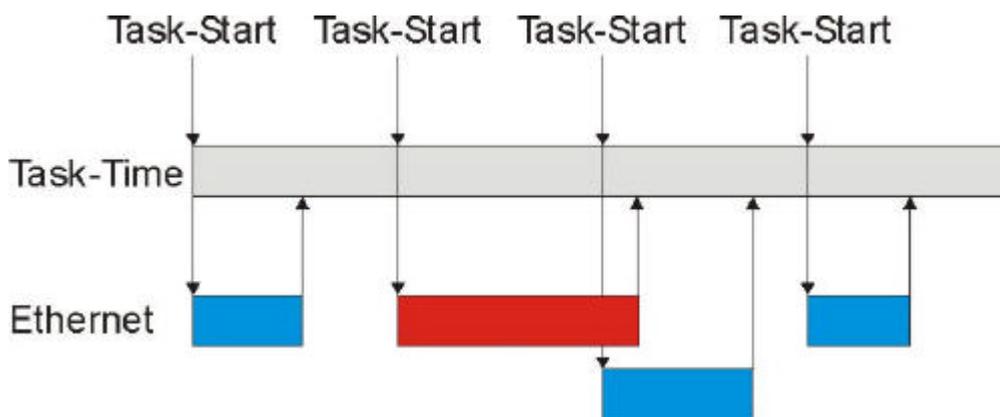
Ideale Konfiguration:

Die Task-Zeit ist immer größer als die Ethernet-Laufzeit. Am Anfang der Task wird ein Ethernet Telegramm weggeschickt und kommt nach einer Zeit  $t_{Eth}$  wieder am PC an. Ist die Zeit  $t_{Eth}$  immer

kleiner als die eingestellte Task-Zeit bleibt der Zähler *MissedCnt* konstant.



Sollte ein Ethernet-Telegramm nach Ablauf der Task-Zeit noch nicht wieder beim PC angekommen sein und erst nach dem Start der nächsten Task eintreffen, so arbeitet TwinCAT mit den alten Eingangsdaten weiter. Zusätzlich wird der Zähler *MissedCnt* um eins erhöht.



Dies kann folgende Ursache haben:

- Task-Zeit zu klein gewählt. Stellen Sie
  - bei TCP 100 ms oder größer ein
  - bei UDP 20 ms oder größer ein
- Zu viele Kollisionen im Netzwerk: setzen Sie Switches anstelle von Hubs ein!
- Die Bus-Auslastung ist zu hoch: wechseln Sie auf 100 MBaud!
- Sie sind beim BC9000 eingeloggt: dieses belegt zusätzliche Prozessorleistung des Busklemmen Controllers, was die Antwortzeit verlängert.

Die beiden folgende Diagnoseeinträge haben je nach Übertragungsart unterschiedliche Bedeutung.

## Diagnose TCP/IP

BoxState

Wert	Bedeutung

0x0000	kein Fehler
0x0001	keine aktuellen Eingänge

MissedCnt

Wert	Bedeutung
0xyyyy	Anzahl der Telegramm, die nicht rechtzeitig zum Task-Anfang wieder angekommen sind. Dieser Wert sollte nahezu konstant sein. Sollte der Wert ständig hochzählen so ist die Task-Zeit zu erhöhen.

## Diagnose UDP/IP

BoxState

Wert	Bedeutung
0x0000	kein Fehler
0x0001	keine aktuellen Eingänge
0x0002	Ausgänge werden zu Null geschrieben
0xxxxzz	xx - Warning Level. Hierbei wird bei jeden nicht rechtzeitigem Empfang der Daten der Wert um eins erhöht. Bei einem fehlerfreien Datenaustausch wird dieser Wert wieder um eins herunter gezählt. Beim Erreichen der max. Warning Level (Default 100) werden die Ausgangsdaten zu Null geschrieben und werden erst wieder für das normale Prozessabbild freigegeben wenn der Warning Level auf Null ist.

MissedCnt

Wert	Bedeutung
0xyyyy	Anzahl der Telegramm, die nicht rechtzeitig zum Task-Anfang wieder angekommen sind. Dieser Wert sollte nahezu konstant sein. Sollte der Wert ständig hochzählen so ist die Task-Zeit zu erhöhen.

Feldbuskomponenten: Diagnose

## ModbusTCP-Diagnose

[siehe Modbus Diagnose-Funktion](#)

[siehe Fehlerantworten ModbusTCP](#)

Feldbuskomponenten: Anhang

## Allgemeine Betriebsbedingungen

Um einen fehlerfreien Betrieb der Komponenten zu erreichen, müssen die nachfolgenden Betriebsbedingungen eingehalten werden.

## Bedingungen an die Umgebung

An folgenden Orten dürfen die Komponenten nicht ohne Zusatzmaßnahmen eingesetzt werden:

- unter erschwerten Betriebsbedingungen, wie z.B. ätzende Dämpfe oder Gase, Staubbildung
- bei hoher ionisierender Strahlung.

Bedingung	zulässiger Bereich
Betriebstemperatur	0°C...+55°C
Vibrationsfestigkeit	gemäß IEC 68-2-6
Schockfestigkeit	gemäß IEC 68-2-27
Einbaulage	beliebig
EMV-Festigkeit	gemäß EN50082 (ESD, Burst)
Aussendung	gemäß EN50081

## Transport und Lagerbedingungen

Bedingung	zulässiger Bereich
Lagertemperatur	-25°C... +85°C
Relative Feuchte	95 %, keine Betauung
Freier Fall	originalverpackt bis 1m

## Schutzklasse und Schutzart

Bedingung	zulässiger Bereich
Schutzklasse nach IEC 536 (VDE 0106, Teil 1)	An der Profilschiene ist ein Schutzleiteranschluss erforderlich!
Schutzart nach IEC 529	IP20 ( Schutz gegen Berührung mit Standard Prüffinger)
Schutz gegen Fremdkörper	kleiner 12 mm im Durchmesser
Schutz gegen Wasser	kein Schutz

## Kennzeichnung der Komponenten

Jede ausgelieferte Komponente enthält einen Aufkleber, mit Informationen über die Zulassung des Produkts. Beispiel für den Buskoppler BK2000:

<b>Lightbus Coupler BK2000</b>	  <small>MWG 28-1+ 53°C max US LISTED Inc. Conf. Co. 2172</small>
Voltage Supply: 24 V DC Baud Rate 2,5 Mbaud	<b>BECKHOFF</b> Eiserstr. 5      D-33415 Verl Phone:      +49-(0)5246/963-0 Fax:      +49-(0)5246/963-149
<b>0 9 0 1 B F 0 6 0 0 0 0</b>	

Auf dem Aufkleber sind folgende Informationen abzulesen:

Aufdruck	Bedeutung für diesen Aufkleber
----------	--------------------------------

genaue Produktbezeichnung	Lightbus Koppler BK2000
Versorgungsspannung	24 V <sub>DC</sub>
Übertragungsrate	2,5 MBaud
Hersteller	Elektro Beckhoff GmbH, Unternehmensbereich Industrie Elektronik
CE-Zeichen	Konformitätskennzeichnung
UL-Zeichen	Kennzeichen für UL-Zulassung. UL steht für Underwriters Laboratories Inc., die führende Zertifizierungsorganisation für Nordamerika mit Sitz in den USA.  C = Kanada, US = USA, LISTED 22ZA (unter diesem Eintrag sind die Prüfergebnisse einsehbar)
Produktionsbezeichnung	Die Zahlenfolge ergibt von links nach rechts die Produktionswoche (2 Ziffern), das Produktionsjahr (2 Ziffern), die Software- (2 Ziffern) sowie die Hardwareversion (2 Ziffern) und Sonderbezeichnungen (4 Ziffern).  In diesem Fall handelt es sich also um einen BK2000 - produziert in der 9. Kalenderwoche - de Jahres 2001 - mit der Firmware-Version BF - in der 6. Hardwareversion - ohne Sonderbezeichnung

Feldbuskomponenten: Anhang

## Zulassungen

UL E172151

## Konformitätskennzeichnung

CE

## Schutzart

IP20 gemäß EN60529

Feldbuskomponenten: Anhang

# Prüfnormen für Geräteprüfung

## EMV

EN 50082-2

EN 50081-2

## Vibrationsfestigkeit

EN 60 68-2-2 Schwingungsprüfung

EN 60 68-2-27 Schockprüfung

Feldbuskomponenten: Anhang

# Literaturverzeichnis

## TCP/IP

TCP/IP (deutsch)

Aufbau und Betrieb eines TCP/IP Netzes  
von Kevin Washburn, Jim Evans

Verlag: ADDISON-WESLEY Longmann Verlag

TCP/IP (englisch)

Illustrated, Volume1 The Protocols  
von W. Richard Stevens

Verlag: ADDISON-WESLEY Longmann Verlag

## Modbus/TCP

<http://www.modicon.com/>

<http://www.modbus.org>

## TwinCAT

TwinCAT Information Dokumentation

<http://tcinfosys.beckhoff.com>

Feldbuskomponenten: Anhang

# Abkürzungsverzeichnis

## ADS

Automation Device Specification

**IP (20)**

Schutzart der Busklemmen

**IPC**

Industrie-PC

**E/A**

Ein- und Ausgänge

**K-Bus**

Klemmen-Bus

**KS2000**

Konfigurationssoftware

**PE**

Der PE-Powerkontakt kann als Schutzerde verwendet werden.

**TwinCAT**

The Windows Control and Automation Technology

Feldbuskomponenten: Anhang

## Support und Service

Beckhoff und seine weltweiten Partnerfirmen bieten einen umfassenden Support und Service, der eine schnelle und kompetente Unterstützung bei allen Fragen zu Beckhoff Produkten und Systemlösungen zur Verfügung stellt.

### Beckhoff Support

Der Support bietet Ihnen einen umfangreichen technischen Support, der Sie nicht nur bei dem Einsatz einzelner Beckhoff Produkte, sondern auch bei weiteren umfassenden Dienstleistungen unterstützt:

- weltweiter Support
- Planung, Programmierung und Inbetriebnahme komplexer Automatisierungssysteme
- umfangreiches Schulungsprogramm für Beckhoff Systemkomponenten

Hotline: +49(0)5246/963-157

Fax: +49(0)5246/963-199

E-Mail: [support@beckhoff.com](mailto:support@beckhoff.com)

### Beckhoff Service

Das Beckhoff Service Center unterstützt Sie rund um den After-Sales-Service:

- Vor-Ort-Service
- Reparaturservice
- Ersatzteilservice
- Hotline-Service

Hotline: +49(0)5246/963-460

Fax: +49(0)5246/963-479

E-Mail: [service@beckhoff.com](mailto:service@beckhoff.com)

Weitere Support- und Service-Adressen entnehmen Sie bitte unseren Internetseiten, unter <http://www.beckhoff.com>. Dort finden Sie auch Dokumentation zu Beckhoff Komponenten.