

KL6021

**Serielle Schnittstelle RS485 (RS422)
Konfigurationsanleitung**

**Version 2.1
24.10.2006**

BECKHOFF

Inhaltsverzeichnis

| | |
|-----------------------------------------|-----------|
| 1. Vorwort | 3 |
| Hinweise zur Dokumentation | 3 |
| Sicherheitshinweise | 4 |
| 2. Technische Daten | 5 |
| 3. Funktionsbeschreibung | 6 |
| 4. Klemmenkonfiguration | 7 |
| 5. Registerbeschreibung | 9 |
| Allgemeine Registerbeschreibung | 9 |
| Klemmenspezifische Registerbeschreibung | 11 |
| Registerkommunikation KL6021 | 14 |
| 6. Datenaustausch, Funktion | 16 |
| 7. Anhang | 18 |
| Mapping im Buskoppler | 18 |
| Registertabelle | 20 |
| Support und Service | 21 |
| Beckhoff Firmenzentrale | 21 |

Vorwort

Hinweise zur Dokumentation

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. Keine der in diesem Handbuch enthaltenen Erklärungen stellt eine Garantie im Sinne von § 443 BGB oder eine Angabe über die nach dem Vertrag vorausgesetzte Verwendung im Sinne von § 434 Abs. 1 Satz 1 Nr. 1 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.

Lieferbedingungen

Es gelten darüber hinaus die allgemeinen Lieferbedingungen der Fa. Beckhoff Automation GmbH.

Copyright

© Diese Dokumentation ist urheberrechtlich geschützt. Jede Wiedergabe oder Drittverwendung dieser Publikation, ganz oder auszugsweise, ist ohne schriftliche Erlaubnis der Beckhoff Automation GmbH verboten.

Sicherheitshinweise

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 Beckhoff Automation GmbH.

Erklärung der Sicherheitssymbole

In der vorliegenden Dokumentation 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.



Achtung

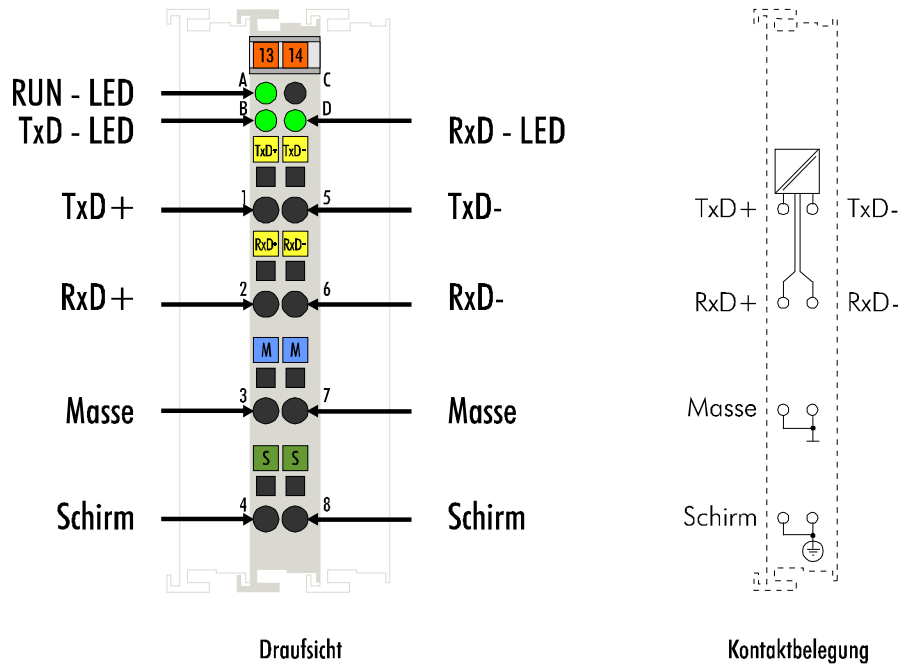
Dieses Symbol weist darauf hin, dass Gefahren für Maschine, Material oder Umwelt bestehen.



Hinweis

Dieses Symbol kennzeichnet Informationen, die zum besseren Verständnis beitragen.

Technische Daten



| Technische Daten | KL6021 |
|-----------------------------|------------------------------------------------------------------------------|
| Übertragungskanäle | TxD und RxD, voll duplex (halbduplex) |
| Übertragungsrage | 9600 Baud (8N1) voreingestellt, (max. 19200 Baud) |
| Bitübertragung | mit Differenzsignal |
| Leitungsimpedanz | 120 Ω |
| Übertragungs - Strecke | ca. 1000 m twisted pair |
| Spannungsversorgung | über den K-Bus |
| Stromaufnahme vom K-Bus | 65 mA typ. |
| Potentialtrennung | 500 Veff (K-Bus / Signalspannung) |
| Datenbuffer | 128 Byte Empfangsbuffer, 16 Byte Sendebuffer |
| Bitbreite im Prozeßabbild | E/A: 3 x 8 Bit Nutzdaten, 1 x 8 Bit Status (bis 5 x 8 Bit Nutzdaten möglich) |
| Konfiguration | keine Adreßeinstellung, Konfigurationseinstellung über den Buskoppler |
| Gewicht ca. | 60 g |
| Betriebstemperatur | 0°C ... +55°C |
| Lagertemperatur | -25°C ... +85°C |
| relative Feuchte | 95% ohne Betauung |
| Vibrations/Schockfestigkeit | gemäß IEC 68-2-6 / IEC 68-2-27 |
| EMV-Festigk. Burst / ESD | gemäß EN 61000-4-4 / EN 61000-4-2 Grenzwerte nach EN 50082-2 |
| Einbaulage | beliebig |
| Schutzart | IP20 |

Funktionsbeschreibung

Die serielle Schnittstellenklemme KL6021 ermöglicht den Anschluß von Geräten mit einer RS485-Schnittstelle. Unabhängig vom überlagerten Bus-system können Daten im Voll-/Halbduplexbetrieb mit der Steuerung ausgetauscht werden. Der Empfangsbuffer ist 128 Byte, der Sendebuffer 16 Byte groß. Der Datentransfer zwischen Klemme und Steuerung wird über einen Handshake im Status und Control-Byte abgewickelt. Die Werkseinstellung der Klemme ist 9600 Baud, 8 Datenbits, 1 Stopbit, no Parity, Voll-Duplex-Modus.

LED Anzeige

Die Run-LED gibt den Betriebszustand der Klemme wieder.
Ein – normaler Betrieb

Aus – Watchdog-Timer Overflow ist aufgetreten. Werden vom Buskoppler 100 ms keine Prozeßdaten übertragen, so erlöscht die grüne LED.

Die TxD und RxD-LEDs geben die Zustände der Signalleitungen wieder.

*Prozeßdaten
Alternatives Ausgabeformat*

Im Alternativ Ausgabeformat werden 4 bzw. 5 Byte (3 Byte Daten und 1 Byte bzw. 2 Byte Control/Status-Byte) im Buskoppler gemappt. Die KL6021 wird im Alternativ Format ausgeliefert. Das Mapping der Klemme im Alternativ Format wird im Kapitel Klemmenkonfiguration näher beschrieben.

Standard Ausgabeformat

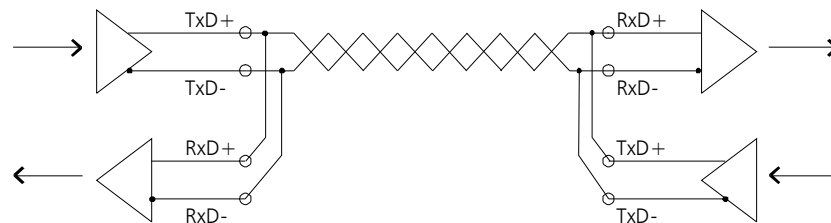
Beim Standard Ausgabeformat werden defaultmäßig 4 Byte (3 Byte Nutzdaten und 1 Byte Control/Status) im Buskoppler gemappt. Durch umparametrieren der KL6021 können bis zu 5 Byte Nutzdaten übertragen werden.

Verweis

Im Anhang befindet sich eine Übersicht über die möglichen Mappingkonfigurationen in Abhängigkeit der einstellbaren Parameter.

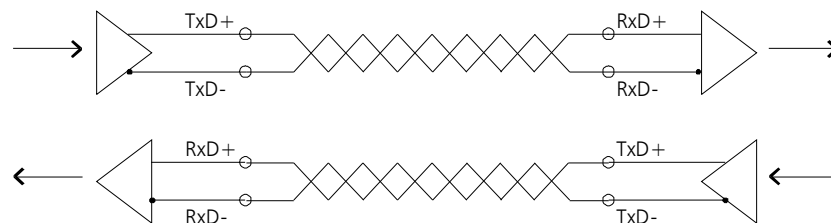
*Anschluß bei RS485
Übertragung*

Bei der RS485 Betriebsart werden die Daten in halbduplex Übertragung ausgetauscht. In dieser Betriebsart kann eine Busstruktur aufgebaut werden.



*Anschluß bei RS422
Übertragung*

Bei der RS422 Betriebsart werden die Daten voll duplex übertragen. Es können nur peer to peer Verbindungen hergestellt werden.



Klemmenkonfiguration

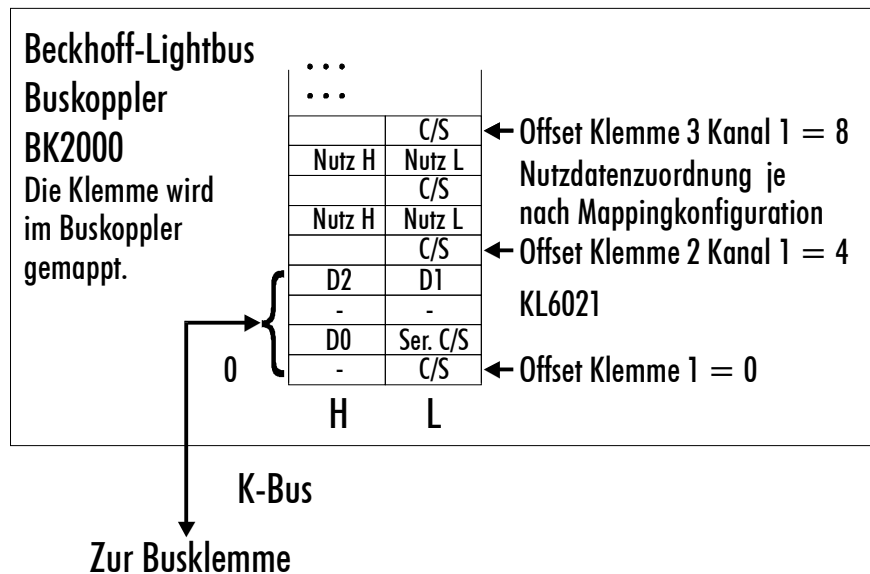
Die Klemme kann über die interne Registerstruktur konfiguriert und parametrisiert werden.

Jeder Klemmenkanal wird im Buskoppler gemappt. In Abhängigkeit vom Typ des Buskopplers und von der eingestellten Mapping-Konfiguration (z.B. Motorola/Intel Format, Wordalignment,...) werden die Daten der Klemme unterschiedlich im Speicher des Buskopplers abgebildet.

Zur Parametrierung einer Klemme ist es erforderlich, das Control-/Statusbyte mit abzubilden.

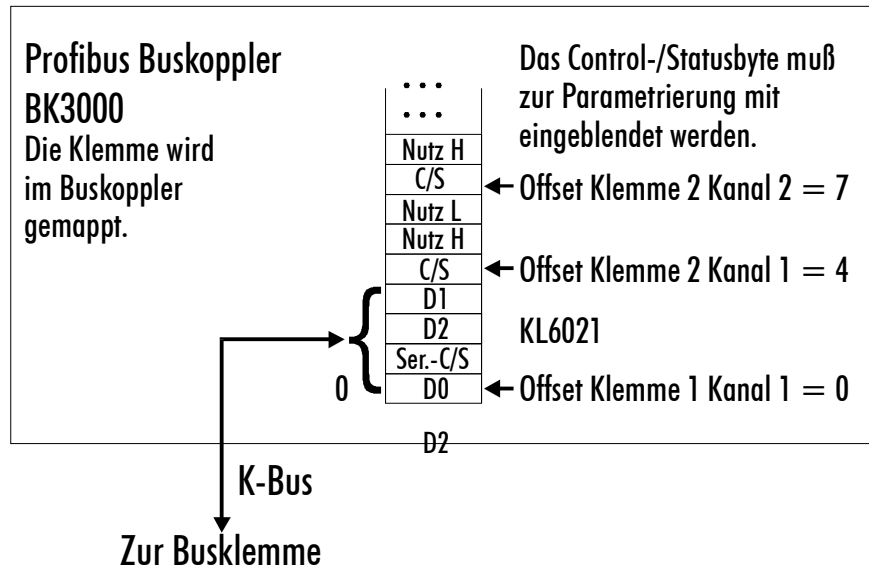
*Beckhoff-Lightbus
Koppler BK2000*

Beim Beckhoff-Lightbus Koppler BK2000 wird neben den Datenbytes auch immer das Control-/Statusbyte gemappt. Dieses liegt stets im Low-Byte auf der Offsetadresse des Klemmenkanals. Bei der KL6021 wird das C/S-Byte ausschließlich im Registermodus genutzt. Das Ser.-C/S-Byte wird für den Protokoll Handshake verwendet.



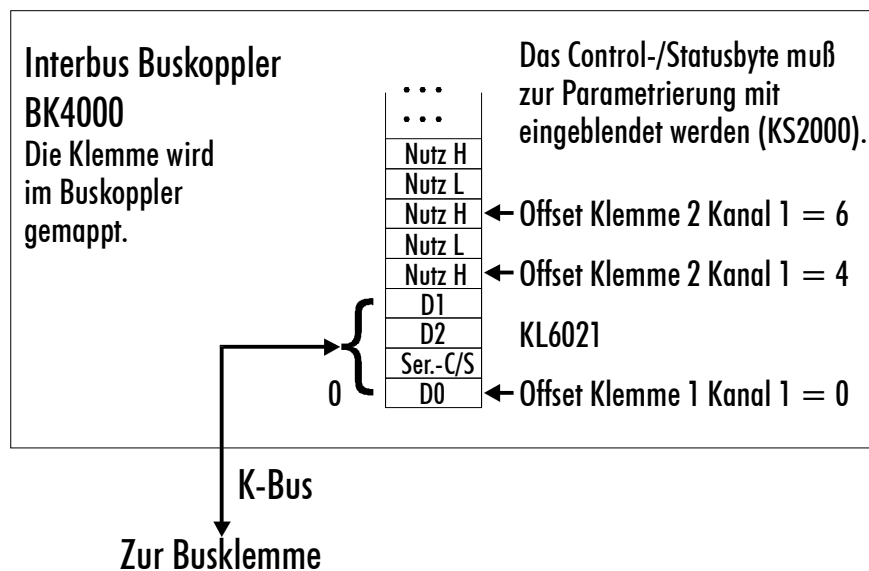
Profibus-Koppler BK3000

Beim Profibus-Koppler BK3000 wird in der Master- Konfiguration Software eingestellt wie sich die KL6021 im Buskoppler mappen soll. Die KL6021 wird im Alternativ Format ausgeliefert. Soll das Standard Format und eine andere Nutzdatenlänge eingestellt werden, beachten Sie bitte die Register 34 und 35. Das Bild zeigt das Mapping für 4 Byte Eingangs- und 4 Byte Ausgangsdaten.



Interbus Koppler BK4000

Der Interbus Koppler BK4000 mappt die KL6021 standardmäßig mit 4 Byte Eingangs- und 4 Byte Ausgangsdaten. Eine Parametrierung über den Feldbus ist nicht möglich. Soll die Klemme umparametriert werden, wird die Software KS2000 benötigt.



andere Buskoppler und weitere Angaben



Parametrierung mit der Software KS2000

Nähere Angaben zur Mappingkonfiguration von Buskopplern finden Sie im jeweiligen Buskoppler-Handbuch im Anhang unter "Konfiguration der Master".

Im Anhang befindet sich eine Übersicht über die möglichen Mappingkonfigurationen in Abhängigkeit der einstellbaren Parameter.

Die Parametrierungen können unabhängig vom Feldbussystem mit der Beckhoff Konfigurationssoftware KS2000 über die serielle Konfigurationschnittstelle im Buskoppler durchgeführt werden.

Registerbeschreibung

Bei den komplexen Klemmen können verschiedene Betriebsarten bzw. Funktionalitäten eingestellt werden. Die „Allgemeine Registerbeschreibung“ erläutert den Inhalt der Register, die bei allen komplexen Klemmen identisch sind.

Die klemmenspezifischen Register werden in dem darauf folgendem Kapitel erklärt.

Der Zugriff auf die internen Register der Klemme wird im Kapitel „Registerkommunikation“ beschrieben.

Allgemeine Registerbeschreibung

Komplexe Klemmen die einen Prozessor besitzen, sind in der Lage mit der übergeordneten Steuerung bidirektional Daten auszutauschen. Diese Klemmen werden im folgenden als intelligente Busklemmen bezeichnet. Zu ihnen zählen die analogen Eingänge (0-10V, -10-10V, 0-20mA, 4-20mA), die analogen Ausgänge (0-10V, -10-10V, 0-20mA, 4-20mA), serielle Schnittstellenklemmen (RS485, RS232, TTY, Datenaustausch-Klemmen), Zähler-Klemmen, Encoder-Interface, SSI-Interface, PWM-Klemme und alle anderen parametrierbaren Klemmen.

Alle intelligenten Klemmen besitzen intern eine in ihren wesentlichen Eigenschaften identisch aufgebaute Datenstruktur. Dieser Datenbereich ist wordweise organisiert und umfaßt 64 Speicherplätze. Über diese Struktur sind die wesentlichen Daten und Parameter der Klemme les- und einstellbar. Zusätzlich sind Funktionsaufrufe mit entsprechenden Parametern möglich. Jeder logische Kanal einer intelligenten Klemme besitzt eine solche Struktur (4-Kanal analog Klemmen besitzen also 4 –Registersätze).

Diese Struktur gliedert sich in folgende Bereiche:
(Eine Liste aller Register finden Sie am Ende dieser Dokumentation.)

| Bereich | Adresse |
|-----------------------------|---------|
| Prozeßvariablen | 0-7 |
| Typ-Register | 8-15 |
| Hersteller Parameter | 16-30 |
| Anwender Parameter | 31-47 |
| Erweiterter Anwenderbereich | 48-63 |

Prozeßvariablen

R0-R7 Register im internen RAM der Klemme:

Die Prozeßvariablen können ergänzend zum eigentlichen Prozeßabbild genutzt werden und sind in ihrer Funktion klemmenspezifisch.

R0-R5: Diese Register besitzen eine vom Klemmen-Typ abhängige Funktion.

R6: Diagnoseregister

Das Diagnoseregister kann zusätzliche Diagnose-Information enthalten. So werden z.B. bei seriellen Schnittstellenklemmen Paritäts-Fehler, die während der Datenübertragung aufgetreten sind, angezeigt.

R7: Kommandoregister

High-Byte_Write = Funktionsparameter
 Low-Byte_Write = Funktionsnummer
 High-Byte_Read = Funktionsergebnis
 Low-Byte_Read = Funktionsnummer

*Typ-Register***R8-R15 Register im internen ROM der Klemme**

Die Typ- und Systemparameter sind fest vom Hersteller programmiert und können vom Anwender nur ausgelesen und nicht verändert werden.

R8: Klemmentype:

Die Klemmentype in Register R8 wird zur Identifizierung der Klemme benötigt.

R9: Softwareversion X.y

Die Software-Version kann als ASCII Zeichenfolge gelesen werden.

R10: Datenlänge

R10 beinhaltet die Anzahl der gemultiplexten Schieberegister und deren Länge in Bit.

Der Buskoppler sieht diese Struktur.

R11: Signalkanäle

Im Vergleich zu R10 steht hier die Anzahl der logisch vorhandenen Kanäle. So kann z.B. ein physikalisch vorhandenes Schieberegister durchaus aus mehreren Signalkanälen bestehen.

R12: Minimale Datenlänge

Das jeweilige Byte enthält die minimal zu übertragene Datenlänge eines Kanals. Ist das MSB gesetzt, so ist das Control/Status-Byte nicht zwingend notwendig für die Funktion der Klemme und wird bei entsprechender Konfiguration des Kopplers nicht zur Steuerung übertragen.

R13: Datentypregister

| Datentypregister | |
|------------------|------------------------------------------------------------------------|
| 0x00 | Klemme ohne gültigen Datentyp |
| 0x01 | Byte-Array |
| 0x02 | Struktur 1 Byte n Bytes |
| 0x03 | Word-Array |
| 0x04 | Struktur 1 Byte n Worte |
| 0x05 | Doppelword-Array |
| 0x06 | Struktur 1 Byte n Doppelworte |
| 0x07 | Struktur 1 Byte 1 Doppelwort |
| 0x08 | Struktur 1 Byte 1 Doppelwort |
| 0x11 | Byte-Array mit variabler logischer Kanallänge |
| 0x12 | Struktur 1 Byte n Bytes mit variabler logischer Kanallänge (z.B. 60xx) |
| 0x13 | Word-Array mit variabler logischer Kanallänge |
| 0x14 | Struktur 1 Byte n Worte mit variabler logischer Kanallänge |
| 0x15 | Doppelword-Array mit variabler logischer Kanallänge |
| 0x16 | Struktur 1 Byte n Doppelworte mit variabler logischer Kanallänge |

R14: nicht benutzt**R15: Alignment-Bits (RAM)**

Mit den Alignment-Bits wird die Analogklemme auf eine Bytegrenze im Klemmenbus gelegt.

*Hersteller Parameter***R16-R30 ist der Bereich der " Hersteller Parameter" (SEEROM)**

Die Herstellerparameter sind spezifisch für jeden Klemmentyp. Sie sind vom Hersteller programmiert, können jedoch auch von der Steuerung geändert werden. Die Herstellerparameter sind spannungsausfallsicher in einem seriellen EERPOM in der Klemme gespeichert.

Diese Register können nur nach dem Setzen eines Codewords in R31 geändert werden.

Anwender Parameter

R31-R47 Bereich "Anwendungs Parameter" (SEEROM)

Die Anwendungsparameter sind spezifisch für jeden Klemmentyp. Sie können vom Programmierer geändert werden. Die Anwendungsparameter sind spannungsausfallsicher in einem seriellen EEPROM in der Klemme gespeichert. Der Anwenderbereich ist über ein Codeword schreibgeschützt.



Hinweis

R31: Codeword-Register im RAM

Damit Parameter im Anwender-Bereich geändert werden können muß hier das Codeword **0x1235** eingetragen werden. Wird ein abweichender Wert in dieses Register eingetragen, so wird der Schreibschutz gesetzt. Bei inaktivem Schreibschutz wird das Codeword beim Lesen des Registers zurückgegeben. Ist der Schreibschutz aktiv, enthält das Register den Wert Null.

R32: Feature-Register

Dieses Register legt die Betriebsarten der Klemme fest. So kann z.B. eine anwenderspezifische Skalierung bei den analogen E/A's aktiviert werden.

R33 - R47

Vom Klemmentyp abhängige Register

Erweiterter Anwendungsbereich

R47-R63

Registererweiterung mit zusätzlichen Funktionen.

Klemmenspezifische Registerbeschreibung

Prozeßvariablen

R0: Anzahl der Datenbytes im Sende-FIFO

Im Low-Byte steht die Anzahl der Daten im Sende-FIFO. Das High-Byte ist nicht benutzt.

R1: Anzahl der Datenbytes im Empfangs-FIFO

Im Low-Byte steht die Anzahl der Daten im Empfangs-FIFO. Das High-Byte ist nicht benutzt.

R2-R5: ohne Funktion

R6: Diagnoseregister

High-Byte: nicht benutzt

Low-Byte: Status des Empfangskanals (Bit 0 –7)

| Bit Nr. | | Bedeutung |
|------------|---|-----------------------------------------------------------------------|
| Bit 0 | 1 | Der Empfangsbuffer ist übergelaufen, ankommende Daten gehen verloren. |
| Bit 1 | 1 | Parity Error ist aufgetreten. |
| Bit 2 | 1 | Framing Error ist aufgetreten. |
| Bit 3 | 1 | Overrun Error ist aufgetreten. |
| Bit 4 | 1 | Buffer ist voll |
| Bit 5 - 15 | - | nicht benutzt, don't change |

Hersteller Parameter

R18: Buffer-Größe

[0x0080]

Das Register R18 legt die Anzahl der Daten im Empfangs-FIFO fest, ab der das BUF_F-Bit im Status-Byte gesetzt wird.

Low-Byte: wird dieser Wert erreicht, so wird BUF_F im Status gesetzt

High-Byte: nicht benutzt

Anwender-Parameter

R32: Baudrate:

[0x0006]

| Bit Nr. | | Baudrate |
|----------------------|-------|-----------------------------|
| Bit 2 Bit 1 Bit 0 | | |
| | 0 1 1 | 1200 Baud |
| | 1 0 0 | 2400 Baud |
| | 1 0 1 | 4800 Baud |
| | 1 1 0 | 9600 Baud [1 1 0] |
| | 1 1 1 | 19200 Baud |
| Bit 3 - 15 | - | nicht benutzt, don't change |

Desweiteren kann die Baudrate nach folgender Gleichung eingestellt werden:

$$\text{Baudrate} = 4 \text{ MHz} / (16 * (\text{HB} + 1))$$

Dabei muß das Low-Byte mit 0xFF beschrieben werden, und das High-Byte (HB) gibt den Operator an.

R33: Datenrahmen

[0x0003]

In diesem Register wird die Einstellung des Datenrahmens vorgenommen.

| Bit Nr. | | Bedeutung |
|----------------------|-------|-------------------------------------|
| Bit 2 Bit 1 Bit 0 | | |
| | 0 0 1 | 7 Datenbits, even-Parity |
| | 0 1 0 | 7 Datenbits, odd-Parity |
| | 0 1 1 | 8 Datenbits, no Parity [0 1 1] |
| | 1 0 0 | 8 Datenbits, even-Parity |
| | 1 0 1 | 8 Datenbits, odd-Parity |
| Bit 3 | 0/1 | 0: 1 Stop-Bit [0] 1: 2 Stop-Bits |
| Bit 4 - 15 | - | nicht benutzt, don't change |

R34: Feature-Register:

[0x0003]

Das Feature-Register legt die Betriebsarten der Klemme fest.

| Feature Bit Nr. | | Beschreibung der Betriebsart |
|-----------------|-----|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Bit 0 | 0/1 | 1: Halbduplex: Der Empfang der von der Klemme gesendeten Daten wird unterdrückt. 0: Vollduplex: Gesendete Daten werden im RS485 Betrieb mitgehört. [0] |
| Bit 1 | 0/1 | 0: Standard Ausgabeformat 1: Alternatives Ausgabeformat [1] |
| Bit 2 | 1 | Das Status-Byte wird von der Klemme einen Zyklus später als die höherwertigen Datenbytes in die Schieberegister des K-Busses kopiert. Dadurch verringert sich die Datenübertragungsrate zur Steuerung. [0] |
| Bit 3 | 1 | Das XON/XOFF-Protokoll wird von der Klemme beim Senden von Daten unterstützt, d.h. die Klemme sendet die von der Steuerung übergebenen Daten, bis sie das Zeichen XOFF (DC3==0x13) vom Partner empfängt. Das Senden wird daraufhin solange unterbunden bis das Zeichen XON (DC1==0x11) empfangen wird. [0] |
| Bit 4 | 1 | Das XON/XOFF-Protokoll wird von der Klemme beim Daten-Empfang unterstützt. Die Klemme sendet das Steuerzeichen XOFF, wenn 118 Zeichen im Buffer der Klemme stehen, XON wird gesendet, wenn vorher XOFF gesendet wurde und die Buffer-Grenze von 18-Byte unterschritten wurde. [0] |
| Bit 5 | 0/1 | 0: Die Klemme wird nach der RS485 Norm in einer Busstruktur genutzt. [0] 1: Die Klemme wird als Punkt zu Punkt Verbindung genutzt (RS422), daraus folgt, daß der Controller die Datenleitung nicht mehr hochohmig schaltet. |
| Bit 6 | 1 | Kontinuierliches Senden der Daten aus dem Fifo. Über die Steuerung wird der Sendebuffer gefüllt (bis zu 16 Byte). Mit steigender Flanke im Control-Byte.3 wird der gefüllte Buffer-Inhalt gesendet. Sind die Daten übertragen, so wird dies durch das Setzen des Bits Status-Byte.2 von der Klemme an die Steuerung quittiert. Status-Byte.2 wird mit Control-Byte.3 zurückgenommen. |
| Bit 7-15 | - | nicht benutzt, don't change |



Hinweis zu Bit6

Dieses Feature ist gültig für alle Klemmen ab Software Version 3x. Die Versionsangabe ist auf der rechten Seitenfläche der Klemme zu finden, in der Seriennummer: xxxx3xxx

Bsp.: 5298**3**A2A ⇒ Software Version **3A**

R35: Anzahl der im Buskoppler gemappten Datenbytes

[0x0003]

Low-Byte: Anzahl der im Buskoppler und zur Steuerung übertragenen Datenbytes. Es können zwischen 1 und 5 Datenbytes übertragen werden. Sollen mehr als 3 Byte Nutzdaten übertragen werden, muß die neue Byte-Anzahl in dieses Register eingetragen werden.

High-Byte = nicht benutzt

Registerkommunikation KL6021

Registerzugriff über den Prozeßdatenaustausch
 Bit 7=1: Registermodus

Bit 6=0: lesen
 Bit 6=1: schreiben

Bit 0 bis 5: Adresse

Control-Byte im Register-Modus

Wenn Bit 7 des Control-Bytes gesetzt wird, werden die ersten zwei Byte der Nutzdaten nicht zum Prozeßdatenaustausch verwendet, sondern in den Registersatz der Klemme geschrieben oder daraus ausgelesen.

In Bit 6 des Control-Bytes legen Sie fest, ob ein Register ausgelesen oder beschrieben werden soll. Wenn das Bit 6 nicht gesetzt ist, wird ein Register ausgelesen, ohne es zu verändern. Der Wert kann dem Eingangs-Prozeßabbild entnommen werden.

Wird das Bit 6 gesetzt, werden die Nutzdaten in ein Register geschrieben. Sobald das Status-Byte im Eingangs-Prozeßabbild eine Quittung geliefert hat, ist der Vorgang abgeschlossen (siehe Bsp.).

In die Bits 0 bis 5 des Control-Bytes wird die Adresse des anzusprechenden Registers eingetragen.

MSB

| | | | | | | | |
|-------|-----|----|----|----|----|----|----|
| REG=1 | W/R | A5 | A4 | A3 | A2 | A1 | A0 |
|-------|-----|----|----|----|----|----|----|

REG = 0 : Prozeßdatenaustausch

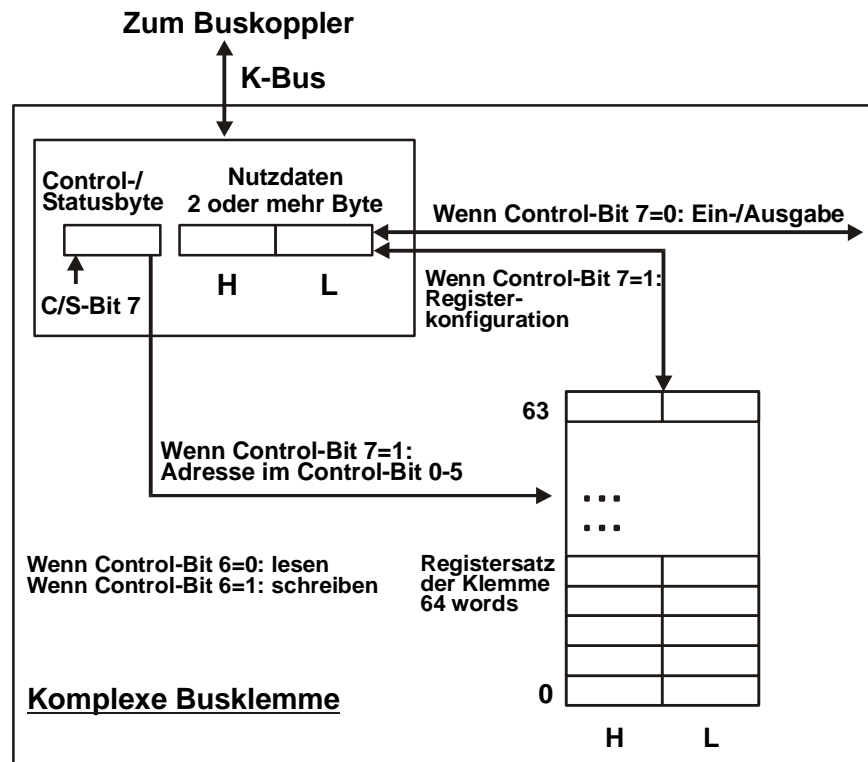
REG = 1 : Zugriff auf Registerstruktur

W/R = 0 : Register lesen

W/R = 1 : Register schreiben

A5..A0 = Registeradresse

Mit Adressen A5...A0 sind insgesamt 64 Register adressierbar.



Das Control- bzw. Status-Byte belegt die niedrigste Adresse eines logischen Kanals. Die entsprechenden Registerwerte befinden sich in den folgenden 2-Datenbytes. (Ausnahme ist der BK2000: hier wird nach dem Control- bzw. Status-Byte ein nicht genutztes Daten-Byte eingeschoben, und somit der Registerwert auf eine Word-Grenze gelegt).

Beispiel

Lesen des Registers 8 im BK2000 mit einer KL3022 und der Endklemme:

Werden die folgenden Bytes von der Steuerung zur Klemme übertragen,

| | | | |
|------------------|------------------------|-------------------------------|------------------------------|
| Byte0 Control | Byte1 Nicht benutzt | Byte2 Daten aus, high byte | Byte3 Daten aus, low byte |
| 0x88 | 0xXX | 0xXX | 0xXX |

so liefert die Klemme die folgende Typ-Bezeichnung zurück (0x0BCE entspricht dem unsigned Integer 3022).

| | | | |
|-----------------|------------------------|-------------------------------|------------------------------|
| Byte0 Status | Byte1 Nicht benutzt | Byte2 Daten ein, high byte | Byte3 Daten ein, low byte |
| 0x88 | 0x00 | 0x0B | 0xCE |

ein weiteres Beispiel

Schreiben des Registers 31 im BK2000 mit einer intelligenten Klemme und der Endklemme:

Werden die folgenden Bytes (Anwender-Codeword) von der Steuerung zur Klemme übertragen,

| | | | |
|------------------|------------------------|-------------------------------|------------------------------|
| Byte0 Control | Byte1 Nicht benutzt | Byte2 Daten aus, high byte | Byte3 Daten aus, low byte |
| 0xDF | 0xXX | 0x12 | 0x35 |

so wird das Anwender-Codeword gesetzt und die Klemme liefert als Quitung die Registeradresse mit dem Bit 7 für Registerzugriff zurück.

| | | | |
|-----------------|------------------------|-------------------------------|------------------------------|
| Byte0 Status | Byte1 Nicht benutzt | Byte2 Daten ein, high byte | Byte3 Daten ein, low byte |
| 0x9F | 0x00 | 0x00 | 0x00 |

Datenaustausch, Funktion

Control-Byte im Prozeßdatenaustausch

Das Control-Byte wird von der Klemme zur Steuerung übertragen. Es kann im Registermodus (REG = 1) oder im Prozeßdatenaustausch (REG = 0) genutzt werden (siehe Anmerkung im Anhang). Zur Abwicklung der Datenübertragung (Handshake) wird das Control- und Status- Byte im Prozeßdatenaustausch benutzt

MSB

| | | | | | | | |
|-------|-----|-----|-----|---|----|----|----|
| REG=0 | OL2 | OL1 | OL0 | 0 | IR | RA | TR |
|-------|-----|-----|-----|---|----|----|----|

Status-Byte im Prozeßdaten-Modus

Das Status-Byte wird von der Klemme zur Steuerung übertragen. Es enthält die für den Handshake erforderlichen Daten.

MSB

| | | | | | | | |
|-------|-----|-----|-----|-------|----|----|----|
| REG=0 | IL2 | IL1 | IL0 | BUF_F | IA | RR | TA |
|-------|-----|-----|-----|-------|----|----|----|

TR/TA:TRANSMIT-REQUEST/ TRANSMIT-ACCEPTED Bits

Der Handshake für das Senden der Daten wird über dieses Bit durchgeführt. Ein Zustandsänderung von TR bewirkt, daß die über OL0-OL2 festgesetzte Anzahl von Daten (maximal 5) in das Sende-FIFO geladen werden. Die Klemme signalisiert über TA die Ausführung dieses Befehls.

Beispiel

| Output Control-Byte | Input Status-Byte | Kommentar |
|-----------------------------------------------------|---------------------------|-----------------------------------------------------------------|
| 00000000 | 0XXXX0X0 | Start Datenübertragung |
| 00100001 Datenbytes: In D0 und D1, Registerdaten | 0XXXX0X0 | Steuerung fordert von der Klemme 2-Daten zu senden |
| | | |
| 00100001 Datenbytes: In D0 und D1, | 0XXXX0X1 | Klemme hat 2 Daten in Sende FIFO geladen, Befehl ist ausgeführt |
| 01010000 Datenbytes In D0 bis D4 | 0XXXX0X1 Datenbytes:DC | Steuerung fordert von der Klemme 5-Daten (D0-D4) zu senden |
| | | |
| 01010000 Datenbytes: In D0 und D1, Registerdaten | 0XXXX0X0 | Klemme hat 5 Daten in Sende FIFO geladen, Befehl ist ausgeführt |

RA/RR:REICEIVE-ACCEPTED/RECEIVE-REQUEST

Über eine Zustandsänderung von RR teilt die Klemme der Steuerung mit, daß sich die in IL0-IL1 angezeigte Anzahl von Daten in D0-D4 befinden. Die Übernahme der Daten wird im Control-Byte mit RA quittiert, erst daraufhin werden neue Daten von der Klemme zur Steuerung übertragen.

Beim ersten Empfang von Daten ist das IL0-Bit gleich 1, auch wenn mehrere Bytes empfangen wurden.

Beispiel

| Output Control-Byte | Input Status-Byte | Kommentar |
|---------------------|-------------------|------------------------------------------------------------------|
| 00000000 | 0XXXX00X | Start Datenübertragung |
| 0XXX000X | 0011X01X | Klemme fordert von der Steuerung 3-Daten von D0-D2 zu übernehmen |
| | | |
| 0XXX001X | 0011X01X | Steuerung hat Daten übernommen |
| 0XXX001X | 0101X00X | Klemme fordert von der Steuerung 5-Daten von D0-D4 zu übernehmen |
| | | |
| 0XXX000X | 0101X00X | Steuerung hat Daten übernommen |

IR/IA:
INIT-REQUEST/INIT-ACCEPTED

Ist IR high, so führt die Klemme eine Initialisierung durch. Die Sende und Empfangsfunktionen werden gesperrt, die FIFO-Zeiger werden zurückgesetzt und die Schnittstelle wird mit den Werten der zuständigen Register (R32-R35,R18) initialisiert. Die Ausführung der Initialisierung wird von der Klemme mit IA quittiert.

Beispiel

| Output Control-Byte | Input Status-Byte | Kommentar |
|---------------------|-------------------|----------------------------------------------|
| 0XXXXXXX | 0XXXXXXX | Start Datenübertragung |
| 00000100 | 0XXXXXXX | Initialisierung wird von Steuerung gefordert |
| | | |
| 00000100 | 00000100 | Klemme hat Initialisierung vollzogen |
| 00000000 | 00000100 | Steuerung fordert Datenaustausch |
| | | |
| 00000000 | 00000000 | Klemme ist bereit |

BUF_F:
BUFFER-FULL_Flag

Der Empfangs-FIFO ist voll. Daten die jetzt empfangen werden, gehen verloren.

Fehlerbehandlung

Tritt ein Parity, Framing oder Overrun Error auf, so geht das betreffende Datum für die Übertragung verloren, es wird nicht in das Empfangs-FIFO der Klemme geladen.

Ist der Buffer voll, so werden die ankommenden Daten ignoriert.

Im Fehlerfall werden die entsprechenden Diagnosebits in R6 gesetzt .

Anhang

Wie bereits im Kapitel Klemmenkonfiguration beschrieben wurde, wird jede Busklemme im Buskoppler gemappt. Dieses Mapping vollzieht sich im Standardfall mit der Defaulteinstellung im Buskoppler / Busklemme. Mit der Beckhoff Konfigurationssoftware KS2000 oder mit einer Master Konfigurationssoftware (z.B. ComProfibus oder TwinCAT System Manager) ist es möglich diese Defaulteinstellung zu verändern. Die folgenden Tabellen geben darüber Auskunft wie sich die KL6021, abhängig von den eingestellten Parametern, im Buskoppler mappt.

Mapping im Buskoppler

Standard Format

Im Standard Format wird die KL6021 defaultmäßig mit 4 Byte (einstellbar : 2 bis 6 Byte über R35) Eingangs- und Ausgangsdaten gemappt.

Anmerkung: Im Standard Format wird das CT/ST-Byte für Register- und Prozeßdaten- Kommunikation genutzt.

| | I/O Offset | High Byte | Low Byte |
|--------------------------|------------|-----------|----------|
| Komplette Auswertung = X | 3 | | |
| MOTOROLA Format = X | 2 | D4(opt.) | D3(opt.) |
| Wordalignment = X | 1 | D2(opt.) | D1(opt.) |
| | 0 | D0 | CT/ST |

Alternatives Format

Im alternativen Format wird die KL6021 mit 4/6 Byte Eingangs- und 4/6 Byte Ausgangsdaten gemappt. Die KL6021 wird in dem Alternativ Format ausgeliefert.

Anmerkung: Im Alternativ Format wird das CT/ST-Byte nur für die Registerkommunikation und das Ser.CT/ST-Byte nur für den Daten-Handshake genutzt.

*Default: CANCEL,
CANopen, RS232,
RS485, ControlNet,
DeviceNet*

| | I/O Offset | High Byte | Low Byte |
|--------------------------|------------|-----------|-----------|
| Komplette Auswertung = 0 | 3 | | |
| MOTOROLA Format = 0 | 2 | | |
| Wordalignment = 0 | 1 | D2 | D1 |
| | 0 | D0 | Ser-CT/ST |

*Default: Interbus,
Profibus*

| | I/O Offset | High Byte | Low Byte |
|--------------------------|------------|-----------|----------|
| Komplette Auswertung = 0 | 3 | | |
| MOTOROLA Format = 1 | 2 | | |
| Wordalignment = 0 | 1 | D1 | D2 |
| | 0 | Ser-CT/ST | D0 |

| | I/O Offset | High Byte | Low Byte |
|--------------------------|------------|-----------|----------|
| Komplette Auswertung = 1 | 3 | | |
| MOTOROLA Format = 0 | 2 | D2 | D1 |
| Wordalignment = 0 | 1 | -- | D0 |
| | 0 | Ser-CT/ST | CT/ST |

| | I/O Offset | High Byte | Low Byte |
|--------------------------|------------|-----------|-----------|
| Komplette Auswertung = 1 | 3 | | |
| MOTOROLA Format = 1 | 2 | D1 | D2 |
| Wordalignment = 0 | 1 | -- | Ser-CT/ST |
| | 0 | D0 | CT/ST |

Default: Lightbus,
Busklemmen Controller
(BCxxxx)

| | I/O Offset | High Byte | Low Byte |
|--------------------------|------------|-----------|-----------|
| Komplette Auswertung = 1 | 3 | D2 | D1 |
| MOTOROLA Format = 0 | 2 | -- | -- |
| Wordalignment = 1 | 1 | D0 | Ser-CT/ST |
| | 0 | -- | CT/ST |

| | I/O Offset | High Byte | Low Byte |
|--------------------------|------------|-----------|----------|
| Komplette Auswertung = 1 | 3 | D1 | D2 |
| MOTOROLA Format = 1 | 2 | -- | -- |
| Wordalignment = 1 | 1 | Ser-CT/ST | D0 |
| | 0 | -- | CT/ST |

Legende

Komplette Auswertung: Die Klemme wird mit Control/ Status-Byte gemappt.
 Motorola Format: Es ist das Motorola oder Intel Format einstellbar.
 Wordalignment: Die Klemme liegt auf einer Wordgrenze im Buskoppler.
 CT: Control- Byte (erscheint im PA der Ausgänge).
 ST: Status- Byte (erscheint im PA der Eingänge).
 Ser.-CT: Control-Byte für den Handshake (erscheint im PA der Ausgänge)
 Ser.-ST: Status-Byte für den Handshake (erscheint im PA der Eingänge)
 D0 – D4: Datenbyte 0 – 4

Registertabelle

Registersatz

| Adresse | Bezeichnung | Defaultwert | R/W | Speichermedium |
|---------|--------------------------------------|-------------|-----|----------------|
| R0 | Anzahl Datenbytes im Sende-Buffer | variabel | R | RAM |
| R1 | Anzahl Datenbytes im Empfangs-Buffer | variabel | R | RAM |
| R2 | nicht benutzt | 0x0000 | R | |
| R3 | nicht benutzt | 0x0000 | R | |
| R4 | nicht benutzt | 0x0000 | R | |
| R5 | nicht benutzt | 0x0000 | R | |
| R6 | Diagnose-Register | variabel | R | RAM |
| R7 | Kommandoregister- nicht benutzt | 0x0000 | R | |
| R8 | Klemmentype | 6021 | R | ROM |
| R9 | Softw. Versionsnummer | 0x???? | R | ROM |
| R10 | Multiplex-Schieberegister | 0x0218 | R | ROM |
| R11 | Signalkanäle | 0x0130 | R | ROM |
| R12 | minimale Datenlänge | 0x3030 | R | ROM |
| R13 | Datenstruktur | 0x0000 | R | ROM |
| R14 | nicht benutzt | 0x0000 | R | |
| R15 | Alignment-Register | variabel | R/W | RAM |
| R16 | Hardware Versionsnummer | spezifisch | R/W | SEEROM |
| R17 | nicht benutzt | 0x0000 | R/W | SEEROM |
| R18 | Buffer voll - Meldung | 0x0080 | R/W | SEEROM |
| R19 | nicht benutzt | 0x0000 | R/W | SEEROM |
| R20 | nicht benutzt | 0x0000 | R/W | SEEROM |
| R21 | nicht benutzt | 0x0000 | R/W | SEEROM |
| R22 | nicht benutzt | 0x0000 | R/W | SEEROM |
| R23 | nicht benutzt | 0x0000 | R/W | SEEROM |
| R24 | nicht benutzt | 0x0000 | R/W | SEEROM |
| R25 | nicht benutzt | 0x0000 | R/W | SEEROM |
| R26 | nicht benutzt | 0x0000 | R/W | SEEROM |
| R27 | nicht benutzt | 0x0000 | R/W | SEEROM |
| R28 | nicht benutzt | 0x0000 | R/W | SEEROM |
| R29 | nicht benutzt | 0x0000 | R/W | SEEROM |
| R30 | nicht benutzt | 0x0000 | R/W | SEEROM |
| R31 | Codeword-Register | variabel | R/W | RAM |
| R32 | Baudrate | 0x0006 | R/W | SEEROM |
| R33 | Datenrahmen | 0x0003 | R/W | SEEROM |
| R34 | Feature-Register | 0x0002 | R/W | SEEROM |
| R35 | Anzahl Datenbytes zum Buskoppler | 0x0003 | R/W | SEEROM |
| R36 | nicht benutzt | 0x0000 | R/W | SEEROM |
| R37 | nicht benutzt | 0x0000 | R/W | SEEROM |
| R38 | nicht benutzt | 0x0000 | R/W | SEEROM |
| R39 | nicht benutzt | 0x0000 | R/W | SEEROM |
| R40 | nicht benutzt | 0x0000 | R/W | SEEROM |
| R41 | nicht benutzt | 0x0000 | R/W | SEEROM |
| R42 | nicht benutzt | 0x0000 | R/W | SEEROM |
| R43 | nicht benutzt | 0x0000 | R/W | SEEROM |
| R44 | nicht benutzt | 0x0000 | R/W | SEEROM |
| R45 | nicht benutzt | 0x0000 | R/W | SEEROM |
| R46 | nicht benutzt | 0x0000 | R/W | SEEROM |
| R47 | nicht benutzt | 0x0000 | R/W | SEEROM |

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) 5246/963-9157
E-Mail: 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

Beckhoff Firmenzentrale

Beckhoff Automation GmbH
Eiserstr. 5
33415 Verl
Germany

Telefon: + 49 (0) 5246/963-0
Fax: + 49 (0) 5246/963-198
E-Mail: info@beckhoff.de
Web: www.beckhoff.de

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