

## Software-/Bausteinbeschreibung

romutec® 19“-Handbedienebene mit Notfunktion  
und  
MOD RTU BUS-Anbindung

# TÜREINBAUSYSTEM *BUS-tec MOD RTU*



romutec®  
Steuer- u. Regelsysteme GmbH  
Jochsberger Straße 39  
D-91592 Buch am Wald  
Telefon: +49 (0) 98 67/ 97 90-0  
Telefax: +49 (0) 98 67/ 97 90-90  
E-Mail: [info@romutec.de](mailto:info@romutec.de)  
Home: [www.romutec.de](http://www.romutec.de)

## Allgemeine Information

### Hinweise zur Bedienungsanleitung

Um alle Vorteile Ihrer neuen Handbedienebene umfassend nutzen zu können, sollten Sie alle Kapitel dieser Bedienungsanleitung lesen, um die Merkmale der Geräte kennenzulernen und den sicheren Umgang mit dem System zu erlernen.

### Sicherheitshinweise

Bevor Sie Ihr Gerät benutzen, sollten Sie die folgenden Sicherheitshinweise sorgfältig lesen. Dies gilt auch, falls zu einem späteren Zeitpunkt Fragen auftreten sollten.

#### **Bestimmungsgemäße Verwendung:**

Das Gerät ist ausschließlich für die in dieser Dokumentation vorgegebenen Bestimmungen und Leistungsmerkmale einzusetzen. Bei nicht bestimmungsgemäßer Benutzung übernimmt der Hersteller keine Haftungs- und Gewährleistungsansprüche.

- Beachten Sie alle am Gerät angebrachten oder in der technischen Dokumentation aufgeführten Hinweise und Warnungen
- Betreiben Sie das Gerät nur in den dafür vorgesehenen Halterungen oder Einbaurahmen
- Die Module sollten nicht in unmittelbarer Umgebung von Frequenzumrichtern und Transformatoren eingebaut werden
- Frequenzumrichter sind mit sämtlichen Schutzmaßnahmen zu beschalten, dass die geforderten Vorschriften und Richtlinien eingehalten werden (z.B. Netzfilter etc.)
- Betreiben Sie das Gerät nicht in der Nähe von Wasser oder anderen Flüssigkeiten, die zu Beschädigungen der elektronischen Bauteile führen können
- Die Anschlußspannung muß den Angaben in der Dokumentation entsprechen
- Die an den Geräten befindlichen Anschlußklemmen sollten ausschließlich von autorisiertem und unterwiesenem Fachpersonal verdrahtet werden
- Führen Sie keine Verdrahtungsarbeiten unter Spannung durch. Es besteht die Gefahr eines elektrischen Schlags, da einige Klemmen 230 V führen können
- Das Verbinden und Lösen von Steckverbindungen unter Spannung ist zu vermeiden. Die Geräte können dadurch zerstört werden !
- Achten Sie darauf, dass keine Gegenstände, z.B. Schrauben oder anderes Befestigungsmaterial, in das Gerät gelangen
- Vermeiden Sie die Installation an Orten mit extremen Temperaturschwankungen. Die im Datenblatt angegebenen Temperaturbereiche für Lagerung und Betrieb sind einzuhalten, um einen störungsfreien Betrieb zu gewährleisten.

Sollten dennoch einmal Störungen auftreten, versuchen Sie niemals, Ihr Gerät selbst zu reparieren. Zerlegen Sie Ihr Gerät nicht, da sonst Teile im Inneren des Gerätes freigelegt und bei Berührung beschädigt werden können. Wenden Sie sich bei Problemen grundsätzlich an den Hersteller.

### Copyright

Copyright © 2016 romutec® Steuer- und Regelsysteme GmbH. Alle Rechte vorbehalten. Ohne die ausdrückliche schriftliche Genehmigung darf diese Anleitung weder als Ganzes noch in Teilen reproduziert, übertragen, umgeschrieben, in Datenerfassungssystemen gespeichert oder in andere Landes- bzw. Computersprachen übersetzt werden. Dies gilt für jede Form und jedes Mittel, sei es elektronisch, mechanisch, magnetisch, optisch, manuell oder auf andere Art und Weise

Modbus® ist ein eingetragenes Warenzeichen von Schneider Electric, lizenziert an die Modbus Organization, Inc.

## Romutec Handbedienebene mit Modbus RTU-Anbindung

### Allgemein

Das System kann an allen Systemen mit Modbus-Protokoll über eine RS485 Schnittstelle betrieben werden. Das verwendete Modbus Protokoll ist Modbus RTU.

Es können mehrere Handbedienebenen mit unterschiedlichen Stationsadressen auf dem gleichen Bus betrieben werden.

Die Klemmenbelegung an der Zentralbaugruppe ist wie folgt:

<b>RS485</b>	<b>Rx-Tx</b>	Klemme:	<b>5</b>
	<b>/Rx-/Tx</b>		<b>4</b>
	<b>GND</b>		<b>3</b>

### Signalisierung

LED I<sup>2</sup>C      **Aus**      Kein Datenaustausch  
                 **Kurz** (ca. 50 msec)      Schreiben von Daten auf den I<sup>2</sup>C-Bus  
                 **An**, Dauerlicht      Busaktivitäts-Überwachung (MODBus) hat ausgelöst

LED MOD-BUS      **Aus**      Kein Datenaustausch  
                 **Kurz** (ca. 100 msec)      Leuchtet während der Telegrammübertragung,  
                 wenn die eigene Stationsnummer angesprochen wird

Im Konfigurationsmenü signalisieren die LEDs die Werte der einzustellenden Parameter.

### MOD-BUS Einstellungen

Datenbits:      8  
Stopbits:      1  
Parität:      keine  
Handshake:      aus

### Busaktivitäts-Überwachung

Die Busaktivitäts-Überwachung setzt alle Ausgänge der Handbedienebene zurück, wenn innerhalb der eingestellten Zeit kein gültiges Telegramm empfangen wurde. Jedes korrekt empfangene Telegramm startet den Timer der Busaktivitäts-Überwachung neu. Die Einstellung der Zeit erfolgt über Register 42 und 43:

	<b>Modbus Typ</b>	Registeradresse	Wertebereich
Busaktivitäts-Überwachung	Word	42 43 <sup>*)</sup>	1...250 Sekunden andere Werte deaktivieren die Überwachung

<sup>\*)</sup> Um eine Änderung beabsichtigt und korrekt ausführen zu können, ist in dem mit \* gekennzeichneten Word das Komplement des vorherigen Word einzustellen, z.B.:

Zeit = 160 Sek. →      Word 42 = 00 A0 h  
   Word 43 = FF 5F h

## Konfiguration

- **Hardware**

### Maximale Anzahl der Erweiterungsmodule pro Zentralmodul

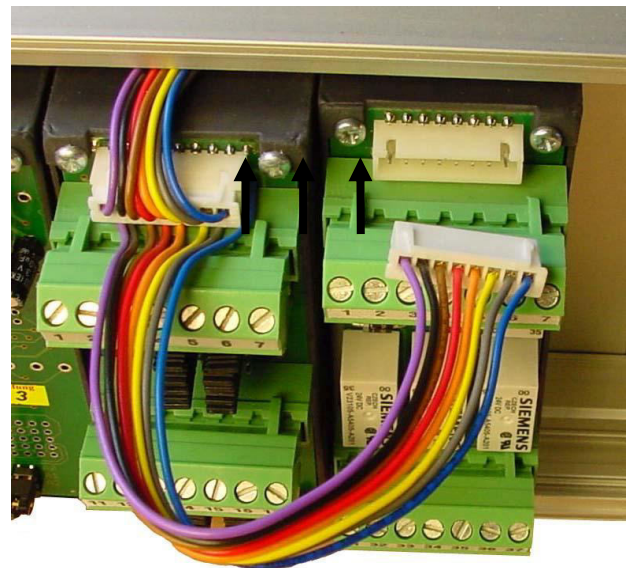
An ein Zentral-/Kommunikationsmodul BZK1000MOD können bis zu 9 Erweiterungsmodule angeschlossen werden, wobei die maximale Anzahl von 8 digitalen oder 7 analogen Modulen gleichzeitig nicht überschritten werden darf. Jede Kombination ist möglich, bezüglich der Einbaureihenfolge gibt es keine Einschränkungen. Es sind lediglich die Regeln für die Adressierung (siehe unten) zu beachten.

### Montage und Busverbindungen

Für die Montage der Steuerkarten stehen 19"-Baugruppenträger (3 HE) zur Verfügung, die Platz für bis zu 10 Module bieten. Die Busverbindungen zwischen den Modulen (I<sup>2</sup>C-/Systembus) sind steckbar ausgeführt und werden mit den an jedem Modul fest angebrachten Busleitungen hergestellt (siehe rechtes Bild).

**Achtung: Stecken oder Lösen der Verbindungen im eingeschalteten Zustand führt zur Zerstörung der Module!**

Über diese Busverbindungen erfolgt außerdem die Spannungsversorgung der digitalen Erweiterungsmodule. Die Spannungsversorgung der Analogmodule ist direkt über Klemmen anzuschließen.



### Adressierung

Die Einstellung der Adressen, unter denen die Erweiterungsmodule vom Zentralmodul angesprochen werden, sind mit jeweils 3 Steckbrücken (Jumpers) einzustellen (binäres System). Damit ergibt sich ein möglicher Wertebereich von 0...7, wie folgende Tabelle zeigt:

Jumper (0=offen, 1=gesteckt)			Adresswert
J1	J2	J3	
0	0	0	0
1	0	0	1
0	1	0	2
1	1	0	3
0	0	1	4
1	0	1	5
0	1	1	6
1	1	1	7

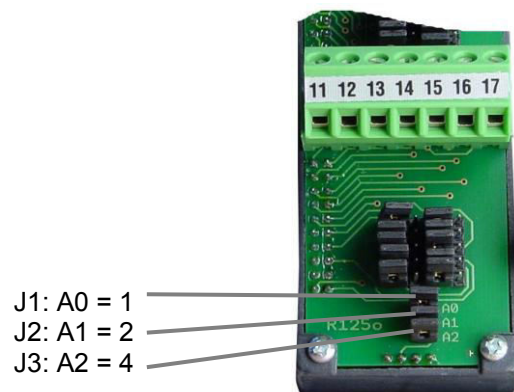


Abb.: Die dargestellte Adresse ist 7 (=1+2+4)

Der Jumper mit der geringsten Wertigkeit (1) ist der obere, der mit der höchsten Wertigkeit (4) der untere, wie die Abbildung 2 zeigt.

Folgendes ist bei der Adressierung zu beachten:

- Es ist keine Doppeladressierung zulässig. Jede Adresse darf pro BZK-Linie nur einmal vergeben werden.
- Jeweils ein digitales und ein analoges Modul an einem BZK1000MOD können dagegen die gleiche Jumperadresse haben, da sich Digital- und Analogmodule durch ein internes Adressbit unterscheiden.

- **Busanbindung**

## Defaulteinstellung

Jedes BZK1000MOD wird mit einer werkseitig konfigurierten Defaulteinstellung (Werte siehe unten) ausgeliefert, mit der die Inbetriebnahme in der Regel sofort möglich ist. Die Defaultwerte können jedoch jederzeit geändert werden, was z.B. nötig ist, wenn mehrere romutec®-Handbedienebenen (mit unterschiedlichen Stationsnummern) an einem Bus betrieben werden sollen. Zum Ändern der Defaultwerte stehen zwei Wege zur Verfügung, nämlich einerseits die Änderung direkt über die Taster am BZK, andererseits über eine bereits funktionierende Modbus-Kommunikation.

## Die Konfiguration über die Taster des Moduls BZK1000 MOD (Verfahren für Geräte ab Version V108)

Werden während des normalen Betriebs die beiden Taster „**Entsperren**“ und „**Hupe quittieren**“ **gleichzeitig** mindestens etwa sieben Sekunden lang gedrückt gehalten, erfolgt der Einsprung in das Konfigurationsmenü. Dabei signalisiert die obere LED (I<sup>2</sup>C) den angewählten nun veränderbaren Menüpunkt, während die untere LED (MOD) den eingestellten Wert anzeigt.

Mit dem Taster „**Entsperren**“ können anschließend die verschiedenen Menüpunkte angewählt werden. Nach dem letzten Menüpunkt folgt wieder der erste Menüpunkt.

Die untere LED (Modbus-Status) zeigt währenddessen die für die Menüpunkte gespeicherten Werte durch Impulsserien mit einer festen Pause an. Die Anzeige von angewähltem Menüpunkt (LED I<sup>2</sup>C) und eingestelltem Wert (LED MOD) erfolgt jeweils im Wechsel.

Mit dem Taster „**Hupe quittieren**“ kann der Wert jeweils um einen Zähler erhöht werden, bis die gewünschte Einstellung erreicht ist.

Mit dem Taster „**Lampen prüfen**“ wird das Konfigurationsmenü wieder verlassen und die Einstellungen dabei gespeichert.

Durch langes Drücken des Tasters auf der Rückseite des Zentralmoduls (ungefähr sieben Sekunden, Signalisierung durch Aufleuchten der vorderen LEDs) werden die Default-Werte wieder hergestellt. Ein vorheriges Aktivieren des Konfigurationsmenüs ist dazu **nicht** erforderlich! Damit werden folgende Einstellungen vorgenommen, die auch die Werkseinstellung sind:

<b>Stationsnummer</b>	<b>160</b>
<b>Modus</b>	<b>Modbus RTU</b>
<b>Schnittstelle</b>	<b>RS 485</b>
<b>Baudrate</b>	<b>9600 / Auto-Bauding</b>



Abb.: Die nebenstehend beschriebenen Bedienelemente

### Menüpunkt, signalisiert durch die LED I<sup>2</sup>C-Status

Punkt 1: „**Einer**“ der Stationsnummer  
Punkt 2: „**Zehner**“ der Stationsnummer  
Punkt 3: „**Hunderter**“ der Stationsnummer

Punkt 4: „**Modus**“                    7 = Modbus RTU

Punkt 5: „**Schnittstelle**“        4 = RS485

Punkt 6: „**Baudrate**“            6 = 9600 Baud  
    7 = 19200 Baud  
    8 = 38400 Baud  
    9 = Auto-Bauding (funktioniert parallel immer)

## Anzeige des konfigurierten Schnittstellentyps

Die **grüne LED** auf der Rückseite des Zentralmoduls zeigt bei nicht gedrücktem Taster auch im normalen Betriebsmodus durch zwei Blinkfrequenzen an, ob die Schnittstelle als RS232 oder RS485 konfiguriert ist:

RS485                      Blinken der LED, ca. 2 Hz  
RS232                      ab Version V18 nicht mehr verfügbar

Taster für  
Defaultwerte

LED zur  
Signalisierung des  
Schnittstellentyps

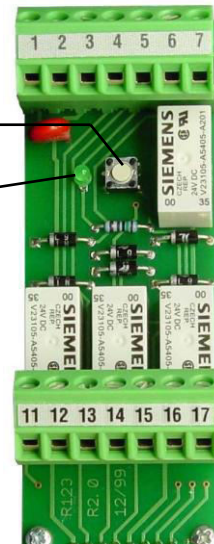


Abb.: LED zur Anzeige des konfigurierten Schnittstellentyps und Taster zum Einstellen der Defaultwerte

## Adressierung bei Schreib- und Lesezugriffen

Die Modbus-Kommunikation mit der Romutec Handbedienebene erfolgt über die Standard Modbus Elemente Input, Coil und Word. Hierbei stehen drei verschiedene Möglichkeiten der Adressierung zur Verfügung.

### Adressierung digitaler Ein-/Ausgänge über Bit-Funktionen (modulweise)

Die Digitalmodule bestehen aus bis zu 4 Gruppen mit je 8 Ein- oder Ausgängen (4 x 8 Bit). Diese werden je nach Modultyp über die Modbus-Befehle **Read Input Status (Function Code FC02)** und **Force multiple Coils (FC15)** angesprochen. Dabei ist zu beachten, dass immer alle Gruppen eines Moduls mit einem einzigen Befehl gelesen oder geschrieben werden, eine Bearbeitung von Teilmengen (z.B. Lesen nur der Gruppe 2 von 3) ist nicht möglich.

Die Analogmodule bestehen aus einer Gruppe mit digitalen Eingängen (für die Schalterstellungs-Rückmeldung) und je einer analogen Eingangs- und Ausgangsgruppe mit je 4 Analogwerten. Diese Gruppen werden mittels der Modbus-Befehle **Read Input Status (FC02)** für die Abfrage der Schalterstellungen, **Read Input Registers (FC04)** und **Force multiple Registers (FC16)** für die Analogwerte angesprochen. Auch hierbei sind alle Datenpunkte eines Moduls mit einem einzigen Befehl zu lesen oder zu schreiben.

Die Startadressen für Schreib- und Lesezugriffe auf die Erweiterungsmodule errechnen sich aus deren gejumpter Hardwareadresse. Dabei ist darauf zu achten, dass die nachfolgend beschriebenen Regeln hinsichtlich Anzahl der Elemente und Startadresse exakt eingehalten werden. Das Bearbeiten von Teilmengen oder ungültige Startadressen werden mit einer Fehlermeldung zurück gewiesen.

## Adressierung digitaler Ein-/Ausgänge über Wort-Funktionen (modulweise)

Ab der Firmware-Version 3.0 des BZK1000 MOD können alle digitalen Ein- und Ausgänge auch mit den Wortfunktionen **Read Input Registers (FC04)** bzw. **Force Multiple Registers (FC16)** angesprochen werden. Auch bei Verwendung der Wortfunktionen müssen jedoch stets alle Register eines Erweiterungsmoduls mit einem einzigen Befehl gelesen bzw. geschrieben werden. Ein Zugriff auf einzelne Datenpunkte ist nicht möglich. **Achtung:** Bei Lesezugriffen auf Register, auf die nur schreibend zugegriffen werden darf, kann es zu hardwarebedingten Störungen kommen, z.B. Schlagen der Relais. Die Startadressen für Schreib- und Lesezugriffe auf die Erweiterungsmodule errechnen sich ähnlich wie bei den Bit-Funktionen aus den gejumperten Hardwareadressen. Genaue Informationen hierüber sind im Anhang in den MODBus-Crosslisten enthalten.

## Adressierung einzelner Datenpunkte sowie modulübergreifende Adressierung

Falls die DDC nicht die Möglichkeit bietet, die bei den beiden oben genannten Adressierungs-Varianten geforderte Anzahl von Datenpunkten mit einem einzigen Befehl anzusprechen, steht bei den BZK1000 MOD ab der Firmware-Version 3.121 ein weiterer Adressbereich zur Verfügung, mit dem sämtliche Datenpunkte einzeln und u.U. sogar modulübergreifend angesprochen werden können.

Um diesen Adressbereich zu nutzen, müssen jedoch im Rahmen der Inbetriebnahme einmalig Konfigurationsregister beschrieben werden. In diesen Konfigurationsregistern werden dem BZK1000 MOD die angeschlossenen Erweiterungsmodule sowie deren gejumperte Moduladresse mitgeteilt. Das BZK1000 kommuniziert dann ständig mit den Erweiterungsmodulen und hält ein Abbild der Eingänge bereit. Hinweis: Aufgrund der Prozessorlast für die ständig im Hintergrund laufende Abfrage der Datenpunkte auf den Erweiterungsmodulen kann diese Art der Adressierung langsamer sein als die beiden erstgenannten Möglichkeiten der Adressierung.

Die Register-, Input- und Coil-Adressen für die Schreib- und Lesezugriffe auf die Erweiterungsmodule errechnen sich ähnlich wie bei den Bit- oder Wort-Funktionen. Allerdings dienen hier nicht die gejumperten Hardwareadressen als Grundlage, sondern der Index der Konfigurationsregister.

In den Konfigurationsregistern R2000 ... R2015 (R2000 + i [mit Index i = 0 ... 15]) wird der Typ der angeschlossenen Erweiterungsmodule mit deren gejumpeter Adresse eingetragen und gespeichert. Das höherwertige Byte enthält den Modultyp, während im niederwertigen Byte die mittels Jumper eingestellte Adresse eingetragen wird.

Genaue Informationen zur Adressierung der Datenpunkte sind dem Anhang „MODBus-Crosslisten für die Adressierung von Einzeldatenpunkten“ zu entnehmen.

Modultyp	Typ-Nr.	Wert (HEX)	Wert (Dez.)
		[High-/Low-Byte]	
BLM1000	0	00 XX	0
BLM1001	1	01 XX	256
BDH1400	2	02 XX	512
BDH1401	3	03 XX	768
BDH2200	4	04 XX	1024
BDH2201	5	05 XX	1280
BDH4800	6	06 XX	1536
BAH400x	8	08 XX	2048
<leer>	<15>	0F FF	4095

Tabelle: Modultypen

## Beispiel

Ein BZK1000 MOD soll mit den aufgeführten Erweiterungsmodulen betrieben werden:

- BDH1401 (Adr. 0)
- BLM1001 (Adr. 1)
- BDH1400 (Adr. 2)
- BAH4000 (Adr. 7)

Dann sind in die Konfigurationsregister die folgenden Werte einzutragen:

Register	Index	Modultyp /Adr.	Wert (Dez.)	Wert (HEX)
			[Modultyp + Adr.]	[High-/Low-Byte]
R 2000	0	BDH1401 Adr. 0	768 + 0 = 768	03 00
R 2001	1	BLM1001 Adr. 1	256 + 1 = 257	01 01
R 2002	2	BDH1400 Adr. 2	512 + 2 = 514	02 02
R 2003	3	BAH4000 Adr. 7	2048 + 7 = 2055	08 07
R 2004	4	-	4095	0F FF
R 2005	5	-	4095	0F FF
R 2006	6	-	4095	0F FF
R 2007	7	-	4095	0F FF
R 2008	8	-	4095	0F FF
R 2009	9	-	4095	0F FF
R 2010	10	-	4095	0F FF
R 2011	11	-	4095	0F FF
R 2012	12	-	4095	0F FF
R 2013	13	-	4095	0F FF
R 2014	14	-	4095	0F FF
R 2015	15	-	4095	0F FF

Tabelle R2000 + Index i [i = 0 ... 15]

**Hinweis:** Für die **Berechnung der Datenpunkt-Adressen** aus den im Anhang befindlichen Cross-Listen ist nach der Konfiguration der Register R2000 ... R2015 nur noch der **Index** entscheidend, nicht mehr die gejumperte Moduladresse, da diese nun ja in den Konfigurationsregistern enthalten ist.

## Lampentest bei Analogmodulen BAH400x

Ab der Version 3.121 des BZK1000MOD ist es möglich, durch Setzen des **Konfigurations-Registers R52** im BZK1000MOD auch bei den BAH400x-Modulen einen „Pseudo-Lampentest“ auszuführen, bei dem die Analogausgänge kurzzeitig auf 10 Volt gesetzt werden. Die Schalter müssen sich dabei in der Automatikstellung befinden. Für die Aktivierung dieser Funktion sind die Bits des R52 im BZK1000MOD auf 1 zu konfigurieren, entsprechend den Stellen, an denen Analogmodule für den Lampentest angeschlossen sind (siehe auch S.53 oder S.62 in der MODBus Crossliste).



Die Aktivierung des Lampentests bei den Analogausgangsmodulen steuert die Ausgänge auf 10 Volt an, solange der Lampentest-Taster gedrückt wird. Bei empfindlichen Aktoren darf diese Funktion nicht aktiviert werden.



## Modul BLM1001

- **Statusabfrage der LEDs 1-10**

Zu verwendender Befehl: **MODBus [02] → READ INPUT STATUS**

**Beispiel: Lesen der Eingänge (Gruppe 0 bis 3) von Moduladresse „0“ an Stationsadresse 160**

Query Message:

Stations- adresse	Modbus- Kommando	Startadresse (High Byte)	Startadresse (Low Byte)	Anzahl Bits (High Byte)	Anzahl Bits (Low Byte)	CRC	CRC
A0 h	<b>02 h</b>	00 h	00 h	<b>00 h</b>	<b>20 h</b>	XX	XX

Werte aus dem Beispiel:

A0 h = BZK-Stationsadresse 160

**02 h** = Kommando READ INPUT STATUS

00 00 h = Startadresse 0 (Moduladresse 0 x 32 + 0 = 0)

**00 20 h** = Anzahl Bits = 32

Response Message:

Stations- adresse	Modbus- Kommando	Byte Count	Data (Byte 1)	Data (Byte 2)	Data (Byte 3)	Data (Byte 4)	CRC	CRC
A0 h	<b>02 h</b>	<b>04 h</b>	3E h	3D h	15 h	3C h	XX	XX

Werte aus dem Beispiel:

A0 h = Adresse 160

**02 h** = Kommando READ INPUT STATUS

**04 h** = Anzahl empfangener Datenbytes = 4

3E h = Datenbyte 1 (xx**11 110**) → LED 1 grün = EIN (Bits 6 + 7 = xx werden nicht berücksichtigt)

3D h = Datenbyte 2 (xx**11 101**) → LED 4 rot = EIN (Bits 6 + 7 = xx werden nicht berücksichtigt)

15 h = Datenbyte 3 (xx**01 010**) → LED 7 - 9 rot = EIN (Bits 6 + 7 = xx werden nicht berücksichtigt)

3C h = Datenbyte 4 (xxxx xx**00**) → LED 10 rot+gn = EIN (Bits 2...7 = xxxx xx werden nicht berücksichtigt)

Detaillierte Informationen, welche LED in welcher Farbe welches Bit setzt, sind im Anhang aus der Modbus-Crossliste ersichtlich.

Anmerkung: Beim BLM1001 werden aktive LEDs mit einer „0“ signalisiert, eine „1“ bedeutet, dass die LED nicht angesteuert ist!

**Wichtig: Es müssen alle 4 Datenbytes zusammen abgefragt werden. Das Auslesen eines einzelnen Bytes (z.B. nur Datenbyte 2) führt zu Fehlfunktionen.**

**Die fett gedruckten Werte im obigen Beispiel sind unveränderlich.**

## Modul BLM1000

- **Ansteuerung der LEDs 1-10**

Zu verwendender Befehl: **MODBus [0F] → FORCE MULTIPLE COILS**

**Beispiel: Setzen der LEDs (=Ausgänge, Gruppe 0 bis 3) von Moduladresse „1“ an Stationsadresse 160**

Query Message:

Stations- adresse	Modbus- Kommando	Startadresse		Anzahl Bits		Byte Count	Data Byte 1	Data Byte 2	Data Byte 3	Data Byte 4	CRC	CRC
		High Byte	Low Byte	High Byte	Low Byte							
A0 h	<b>0F h</b>	00 h	20 h	<b>00 h</b>	<b>20 h</b>	<b>04 h</b>	01 h	02 h	2A h	03 h	XX	XX

Werte aus dem Beispiel:

A0 h = BZK-Stationsadresse 160  
**0F h** = Kommando FORCE MULTIPLE COILS  
 00 20 h = Startadresse 32 (Moduladresse 1 x 32 + 0 = 32)  
**00 20 h** = Anzahl Bits = 32  
**04 h** = Anzahl zu sendender Datenbytes = 4  
 01 h = Datenbyte 1 (xx**00 0001**) → LED 1 grün = EIN (Bits 6 + 7 = xx werden nicht berücksichtigt)  
 02 h = Datenbyte 2 (xx**00 0010**) → LED 4 rot = EIN (Bits 6 + 7 = xx werden nicht berücksichtigt)  
 2A h = Datenbyte 3 (xx**10 1010**) → LED 7 - 9 rot = EIN (Bits 6 + 7 = xx werden nicht berücksichtigt)  
 03 h = Datenbyte 4 (xxxx xx**11**) → LED 10 rot+gn = EIN (Bits 2...7 = xxxx xx werden nicht berücksichtigt)

Response Message:

Stations- adresse	Modbus- Kommando	Startadresse (High Byte)	Startadresse (Low Byte)	Anzahl Bits (High Byte)	Anzahl Bits (Low Byte)	CRC	CRC
A0 h	<b>0F h</b>	00 h	20 h	<b>00 h</b>	<b>20 h</b>	XX	XX

Werte aus dem Beispiel:

A0 h = BZK-Stationsadresse 160  
**0F h** = Kommando FORCE MULTIPLE COILS  
 00 20 h = Startadresse 32 (Moduladresse 1 x 32 + 0 = 32)  
**00 20 h** = Anzahl Bits = 32

Detaillierte Informationen, welche LED in welcher Farbe von welchen Bits angesteuert wird sind im Anhang aus der Modbus-Crossliste ersichtlich.

Anmerkung: Beim BLM1000 werden aktive LEDs mit einer „1“ angesteuert, eine „0“ bedeutet, dass die LED nicht angesteuert wird.

**Wichtig: Es müssen alle 4 Datenbytes zusammen geschrieben werden. Das Schreiben eines einzelnen Bytes (z.B. nur Datenbyte 2) führt zu Fehlfunktionen.**

**Die fett gedruckten Werte im obigen Beispiel sind unveränderlich.**

## Modul BDH1401

- **Statusabfrage der Schalterstellungen und der LEDs**

Zu verwendender Befehl: **MODBus [02] → READ INPUT STATUS**

**Beispiel: Lesen der Eingänge (Gruppe 0 bis 2) von Moduladresse „2“ an Stationsadresse 160**

Query Message:

Stations- adresse	Modbus- Kommando	Startadresse (High Byte)	Startadresse (Low Byte)	Anzahl Bits (High Byte)	Anzahl Bits (Low Byte)	CRC	CRC
A0 h	<b>02 h</b>	00 h	40 h	<b>00 h</b>	<b>18 h</b>	XX	XX

Werte aus dem Beispiel:

A0 h = BZK-Stationsadresse 160

**02 h** = Kommando READ INPUT STATUS

00 40 h = Startadresse 64 (Moduladresse 2 x 32 + 0 = 64)

**00 18 h** = Anzahl Bits = 24

Response Message:

Stations- adresse	Modbus- Kommando	Byte Count	Data (Byte 1)	Data (Byte 2)	Data (Byte 3)	CRC	CRC
A0 h	<b>02 h</b>	<b>03 h</b>	39 h	2A h	3D h	XX	XX

Werte aus dem Beispiel:

A0 h = Adresse 160

**02 h** = Kommando READ INPUT STATUS

**03 h** = Anzahl empfangener Datenbytes = 3

39 h = Datenbyte 1 (xxxx 1001) → Schalter 1 + 4 = Hand (Bits 4...7 = xxxx werden nicht berücksichtigt)

2A h = Datenbyte 2 (xx**10 1010**) → LED 1, 3, 5 rot = EIN (Bits 6 + 7 = xx werden nicht berücksichtigt)

3D h = Datenbyte 3 (xxxx xx**01**) → LED 8 grün = EIN (Bits 2...7 = xxxx xx werden nicht berücksichtigt)

Detaillierte Informationen, welcher Schalter und welche LED welches Bit beeinflussen, sind im Anhang aus der Modbus-Crossliste ersichtlich.

Anmerkung: Beim BDH1401 werden aktive LEDs mit einer „0“ signalisiert, eine „1“ bedeutet, dass die LED nicht angesteuert ist!

Schalter in Automatikstellung setzen ihr zugehöriges Bit auf „0“. Wechselt dieses auf „1“ bedeutet das, dass der Schalter in Stellung Hand (Aus oder Ein) gedreht wurde.

**Wichtig: Es müssen alle 3 Datenbytes zusammen abgefragt werden. Das Auslesen eines einzelnen Bytes (z.B. nur Datenbyte 2) führt zu Fehlfunktionen.**

**Die fett gedruckten Werte im obigen Beispiel sind unveränderlich.**

## Modul BDH1401

- **Ansteuerung der Relais**

Zu verwendender Befehl: **MODBus [0F] → FORCE MULTIPLE COILS**

**Beispiel: Setzen der Relais (=Ausgänge, Gruppe 0) von Moduladresse „2“ an Stationsadresse 160**

Query Message:

Stations- adresse	Modbus- Kommando	Startadresse		Anzahl Bits		Byte Count	Data Byte	CRC	CRC
		High Byte	Low Byte	High Byte	Low Byte				
A0 h	<b>0F h</b>	00 h	40 h	<b>00 h</b>	<b>08 h</b>	<b>01 h</b>	07 h	XX	XX

Werte aus dem Beispiel:

A0 h = BZK-Stationsadresse 160

**0F h** = Kommando FORCE MULTIPLE COILS

00 40 h = Startadresse 64 (Moduladresse 2 x 32 + 0 = 64)

**00 08 h** = Anzahl Bits = 8

**01 h** = Anzahl zu sendender Datenbytes = 1

07 h = Datenbyte (xxxx 0111) → Relais 1 bis 3 = EIN (Bits 4...7 = xxxx werden nicht berücksichtigt)

Response Message:

Stations- adresse	Modbus- Kommando	Startadresse (High Byte)	Startadresse (Low Byte)	Anzahl Bits (High Byte)	Anzahl Bits (Low Byte)	CRC	CRC
A0 h	<b>0F h</b>	00 h	40 h	<b>00 h</b>	<b>08 h</b>	XX	XX

Werte aus dem Beispiel:

A0 h = BZK-Stationsadresse 160

**0F h** = Kommando FORCE MULTIPLE COILS

00 40 h = Startadresse 64 (Moduladresse 2 x 32 + 0 = 64)

**00 08 h** = Anzahl Bits = 8

Detaillierte Informationen, welches Relais von welchem Bit angesteuert wird sind im Anhang aus der Modbus-Crossliste ersichtlich.

Anmerkung: Beim BDH1401 werden die Relais mit einer „1“ angesteuert, eine „0“ bedeutet, dass das Relais nicht angesteuert wird.

**Die fett gedruckten Werte im obigen Beispiel sind unveränderlich.**

## Modul BDH1400

- **Statusabfrage der Schalterstellungen**

Zu verwendender Befehl: **MODBus [02] → READ INPUT STATUS**

**Beispiel: Lesen der Eingänge (Gruppe 0) von Moduladresse „3“ an Stationsadresse 160**

Query Message:

Stations- adresse	Modbus- Kommando	Startadresse (High Byte)	Startadresse (Low Byte)	Anzahl Bits (High Byte)	Anzahl Bits (Low Byte)	CRC	CRC
A0 h	<b>02 h</b>	00 h	60 h	<b>00 h</b>	<b>08 h</b>	XX	XX

Werte aus dem Beispiel:

A0 h = BZK-Stationsadresse 160

**02 h** = Kommando READ INPUT STATUS

00 60 h = Startadresse 96 (Moduladresse 3 x 32 + 0 = 96)

**00 08 h** = Anzahl Bits = 8

Response Message:

Stations- adresse	Modbus- Kommando	Byte Count	Data Byte	CRC	CRC
A0 h	<b>02 h</b>	<b>01 h</b>	06 h	XX	XX

Werte aus dem Beispiel:

A0 h = Adresse 160

**02 h** = Kommando READ INPUT STATUS

**01 h** = Anzahl empfangener Datenbytes = 1

06 h = Datenbyte (xxxx 0110) → Schalter 2 + 3 = Hand (Bits 4...7 = xxxx werden nicht berücksichtigt)

Detaillierte Informationen, welcher Schalter welches Bit beeinflusst, sind im Anhang aus der Modbus-Crossliste ersichtlich.

Anmerkung: Beim BDH1400 setzen Schalter in Automatikstellung ihr zugehöriges Bit auf „0“. Wechselt dieses dagegen auf „1“ bedeutet das, dass der Schalter in Stellung Hand (Aus oder Ein) gedreht wurde.

**Die fett gedruckten Werte im obigen Beispiel sind unveränderlich.**

## Modul BDH1400

- **Ansteuerung der Relais und LEDs**

Zu verwendender Befehl: **MODBus [0F] → FORCE MULTIPLE COILS**

**Beispiel: Setzen der Relais und LEDs (=Ausgänge, Gruppen 0 bis 2) von Moduladresse „3“ an Stationsadresse 160**

Query Message:

Stations- adresse	Modbus- Kommando	Startadresse		Anzahl Bits		Byte Count	Data Byte 1	Data Byte 2	Data Byte 3	CRC	CRC
		High Byte	Low Byte	High Byte	Low Byte						
A0 h	<b>0F h</b>	00 h	60 h	<b>00 h</b>	<b>18 h</b>	<b>03 h</b>	0E h	20 h	06 h	XX	XX

Werte aus dem Beispiel:

A0 h = BZK-Stationsadresse 160

**0F h** = Kommando FORCE MULTIPLE COILS

00 60 h = Startadresse 96 (Moduladresse 3 x 32 + 0 = 96)

**00 18 h** = Anzahl Bits = 24

**03 h** = Anzahl zu sendender Datenbytes = 3

0E h = Datenbyte 1 (xx**00 1110**) → LED 1+2 grün, 3 rot (Bits 6 + 7 = xx werden nicht berücksichtigt)

20 h = Datenbyte 2 (xx**10 0000**) → LED 8 grün = EIN (Bits 6 + 7 = xx werden nicht berücksichtigt)

06 h = Datenbyte 3 (xxxx **0110**) → Relais 2 und 3 = EIN (Bits 4...7 = xxxx werden nicht berücksichtigt)

Response Message:

Stations- adresse	Modbus- Kommando	Startadresse (High Byte)	Startadresse (Low Byte)	Anzahl Bits (High Byte)	Anzahl Bits (Low Byte)	CRC	CRC
A0 h	<b>0F h</b>	00 h	60 h	<b>00 h</b>	<b>18 h</b>	XX	XX

Werte aus dem Beispiel:

A0 h = BZK-Stationsadresse 160

**0F h** = Kommando FORCE MULTIPLE COILS

00 60 h = Startadresse 96 (Moduladresse 3 x 32 + 0 = 96)

**00 18 h** = Anzahl Bits = 24

Detaillierte Informationen, welches Relais von welchem Bit angesteuert wird und welche LEDs in welcher Farbe von welchen Bits beeinflusst werden, sind im Anhang aus der Modbus-Crossliste ersichtlich.

Anmerkung: Beim BDH1400 werden die Relais und LEDs mit einer „1“ angesteuert, eine „0“ bedeutet, dass das Relais bzw. die LED nicht angesteuert wird.

**Die fett gedruckten Werte im obigen Beispiel sind unveränderlich.**

## Modul BDH2201

- **Statusabfrage der Schalterstellungen und der LEDs**

Zu verwendender Befehl: **MODBus [02] → READ INPUT STATUS**

**Beispiel: Lesen der Eingänge (Gruppe 0 bis 2) von Moduladresse „4“ an Stationsadresse 160**

Query Message:

Stations- adresse	Modbus- Kommando	Startadresse (High Byte)	Startadresse (Low Byte)	Anzahl Bits (High Byte)	Anzahl Bits (Low Byte)	CRC	CRC
A0 h	<b>02 h</b>	00 h	80 h	<b>00 h</b>	<b>18 h</b>	XX	XX

Werte aus dem Beispiel:

- A0 h = BZK-Stationsadresse 160
- 02 h** = Kommando READ INPUT STATUS
- 00 80 h = Startadresse 128 (Moduladresse 4 x 32 + 0 = 128)
- 00 18 h** = Anzahl Bits = 24

Response Message:

Stations- adresse	Modbus- Kommando	Byte Count	Data (Byte 1)	Data (Byte 2)	Data (Byte 3)	CRC	CRC
A0 h	<b>02 h</b>	<b>03 h</b>	31 h	1E h	3F h	XX	XX

Werte aus dem Beispiel:

- A0 h = Adresse 160
- 02 h** = Kommando READ INPUT STATUS
- 03 h** = Anzahl empfangener Datenbytes = 3
- 31 h = Datenbyte 1 (xxxx xx01) → Schalter 1 = Hand (Bits 2...7 = xxxx xx werden nicht berücksichtigt)
- 1E h = Datenbyte 2 (xx**01 1110**) → LED 1 und 6 = EIN (Bits 6 + 7 = xx werden nicht berücksichtigt)
- 3F h = Datenbyte 3 (xxxx xxxx) → (Byte 3 wird nicht berücksichtigt, muss jedoch mit abgefragt werden)

Detaillierte Informationen, welcher Schalter und welche LED welches Bit beeinflussen, sind im Anhang aus der Modbus-Crossliste ersichtlich.

Anmerkung: Beim BDH2201 werden aktive LEDs mit einer „0“ signalisiert, eine „1“ bedeutet, dass die LED nicht angesteuert ist!

Schalter in Automatikstellung setzen ihr zugehöriges Bit auf „0“. Wechselt dieses auf „1“ bedeutet das, dass der Schalter in Stellung Hand (Aus oder Ein) gedreht wurde.

**Wichtig: Es müssen alle 3 Datenbytes zusammen abgefragt werden. Das Auslesen eines einzelnen Bytes (z.B. nur Datenbyte 2) führt zu Fehlfunktionen.**

**Die fett gedruckten Werte im obigen Beispiel sind unveränderlich.**

## Modul BDH2201

- **Ansteuerung der Relais**

Zu verwendender Befehl: **MODBus [0F] → FORCE MULTIPLE COILS**

**Beispiel: Setzen der Relais (=Ausgänge, Gruppe 0) von Moduladresse „4“ an Stationsadresse 160**

Query Message:

Stations- adresse	Modbus- Kommando	Startadresse		Anzahl Bits		Byte Count	Data Byte	CRC	CRC
		High Byte	Low Byte	High Byte	Low Byte				
A0 h	<b>0F h</b>	00 h	80 h	<b>00 h</b>	<b>08 h</b>	<b>01 h</b>	0A h	XX	XX

Werte aus dem Beispiel:

A0 h = BZK-Stationsadresse 160

**0F h** = Kommando FORCE MULTIPLE COILS

00 80 h = Startadresse 128 (Moduladresse 4 x 32 + 0 = 128)

**00 08 h** = Anzahl Bits = 8

**01 h** = Anzahl zu sendender Datenbytes = 1

0A h = Datenbyte (xxxx 1010) → Relais 2 und 4 = EIN (Bits 4...7 = xxxx werden nicht berücksichtigt)

Response Message:

Stations- adresse	Modbus- Kommando	Startadresse (High Byte)	Startadresse (Low Byte)	Anzahl Bits (High Byte)	Anzahl Bits (Low Byte)	CRC	CRC
A0 h	<b>0F h</b>	00 h	80 h	<b>00 h</b>	<b>08 h</b>	XX	XX

Werte aus dem Beispiel:

A0 h = BZK-Stationsadresse 160

**0F h** = Kommando FORCE MULTIPLE COILS

00 80 h = Startadresse 128 (Moduladresse 4 x 32 + 0 = 128)

**00 08 h** = Anzahl Bits = 8

Detaillierte Informationen, welches Relais von welchem Bit angesteuert wird sind im Anhang aus der Modbus-Crossliste ersichtlich.

Anmerkung: Beim BDH2201 werden die Relais mit einer „1“ angesteuert, eine „0“ bedeutet, dass das Relais nicht angesteuert wird.

**Die fett gedruckten Werte im obigen Beispiel sind unveränderlich.**



## Modul BDH2200

- **Statusabfrage der Schalterstellungen**

Zu verwendender Befehl: **MODBus [02] → READ INPUT STATUS**

**Beispiel: Lesen der Eingänge (Gruppe 0) von Moduladresse „6“ an Stationsadresse 160**

Query Message:

Stations- adresse	Modbus- Kommando	Startadresse (High Byte)	Startadresse (Low Byte)	Anzahl Bits (High Byte)	Anzahl Bits (Low Byte)	CRC	CRC
A0 h	<b>02 h</b>	00 h	C0 h	<b>00 h</b>	<b>08 h</b>	XX	XX

Werte aus dem Beispiel:

A0 h = BZK-Stationsadresse 160

**02 h** = Kommando READ INPUT STATUS

00 C0 h = Startadresse 192 (Moduladresse 6 x 32 + 0 = 192)

**00 08 h** = Anzahl Bits = 8

Response Message:

Stations- adresse	Modbus- Kommando	Byte Count	Data Byte	CRC	CRC
A0 h	<b>02 h</b>	<b>01 h</b>	03 h	XX	XX

Werte aus dem Beispiel:

A0 h = Adresse 160

**02 h** = Kommando READ INPUT STATUS

**01 h** = Anzahl empfangener Datenbytes = 1

03 h = Datenbyte (xxxx xx11) → beide Schalter 1 + 2 = Hand (Bits 2...7 = xxxx xx werden nicht berücksichtigt)

Detaillierte Informationen, welcher Schalter welches Bit beeinflusst, sind im Anhang aus der Modbus-Crossliste ersichtlich.

Anmerkung: Beim BDH2200 setzen Schalter in Automatikstellung ihr zugehöriges Bit auf „0“. Wechselt dieses dagegen auf „1“ bedeutet das, dass der Schalter in Stellung Hand (Aus oder Ein) gedreht wurde.

**Die fett gedruckten Werte im obigen Beispiel sind unveränderlich.**

## Modul BDH2200

- **Ansteuerung der Relais und LEDs**

Zu verwendender Befehl: **MODBus [0F] → FORCE MULTIPLE COILS**

**Beispiel: Setzen der Relais und LEDs (=Ausgänge, Gruppen 0 bis 2) von Moduladresse „6“ an Stationsadresse 160**

Query Message:

Stations- adresse	Modbus- Kommando	Startadresse		Anzahl Bits		Byte Count	Data Byte 1	Data Byte 2	Data Byte 3	CRC	CRC
		High Byte	Low Byte	High Byte	Low Byte						
A0 h	<b>0F h</b>	00 h	C0 h	<b>00 h</b>	<b>18 h</b>	<b>03 h</b>	01 h	03 h	06 h	XX	XX

Werte aus dem Beispiel:

A0 h = BZK-Stationsadresse 160

**0F h** = Kommando FORCE MULTIPLE COILS

00 C0 h = Startadresse 192 (Moduladresse 6 x 32 + 0 = 192)

**00 18 h** = Anzahl Bits = 24

**03 h** = Anzahl zu sendender Datenbytes = 3

01 h = Datenbyte 1 (xx**00 0001**) → LED 1 rot = EIN

(Bits 6 + 7 = xx werden nicht berücksichtigt)

03 h = Datenbyte 2 (xxxx xx**11**) → LED 5 + 6 grün = EIN

(Bits 2...7 = xxxx xx werden nicht berücksichtigt)

06 h = Datenbyte 3 (xxxx 0**110**) → Relais 2 und 3 = EIN

(Bits 4...7 = xxxx werden nicht berücksichtigt)

Response Message:

Stations- adresse	Modbus- Kommando	Startadresse (High Byte)	Startadresse (Low Byte)	Anzahl Bits (High Byte)	Anzahl Bits (Low Byte)	CRC	CRC
A0 h	<b>0F h</b>	00 h	C0 h	<b>00 h</b>	<b>18 h</b>	XX	XX

Werte aus dem Beispiel:

A0 h = BZK-Stationsadresse 160

**0F h** = Kommando FORCE MULTIPLE COILS

00 C0 h = Startadresse 192 (Moduladresse 6 x 32 + 0 = 192)

**00 18 h** = Anzahl Bits = 24

Detaillierte Informationen, welches Relais von welchem Bit angesteuert wird und welche LEDs in welcher Farbe von welchen Bits beeinflusst werden, sind im Anhang aus der Modbus-Crossliste ersichtlich.

Anmerkung: Beim BDH2200 werden die Relais und LEDs mit einer „1“ angesteuert, eine „0“ bedeutet, dass das Relais bzw. die LED nicht angesteuert wird.

**Die fett gedruckten Werte im obigen Beispiel sind unveränderlich.**

## Modul BDH4800

- **Statusabfrage der Taster**

Zu verwendender Befehl: **MODBus [02] → READ INPUT STATUS**

**Beispiel: Lesen der Eingänge (Gruppe 0) von Moduladresse „5“ an Stationsadresse 160**

Query Message:

Stations- adresse	Modbus- Kommando	Startadresse (High Byte)	Startadresse (Low Byte)	Anzahl Bits (High Byte)	Anzahl Bits (Low Byte)	CRC	CRC
A0 h	<b>02 h</b>	00 h	A0 h	<b>00 h</b>	<b>08 h</b>	XX	XX

Werte aus dem Beispiel:

A0 h = BZK-Stationsadresse 160

**02 h** = Kommando READ INPUT STATUS

00 A0 h = Startadresse 160 (Moduladresse 5 x 32 + 0 = 160)

**00 08 h** = Anzahl Bits = 8

Response Message:

Stations- adresse	Modbus- Kommando	Byte Count	Data Byte	CRC	CRC
A0 h	<b>02 h</b>	<b>01 h</b>	04 h	XX	XX

Werte aus dem Beispiel:

A0 h = Adresse 160

**02 h** = Kommando READ INPUT STATUS

**01 h** = Anzahl empfangener Datenbytes = 1

04 h = Datenbyte (xxxx 0100) → Taster 3 = gedrückt (Bits 4...7 = xxxx werden nicht berücksichtigt)

Detaillierte Informationen, welcher Taster welches Bit beeinflusst, sind im Anhang aus der Modbus-Crossliste ersichtlich.

Anmerkung: Beim BDH4800 setzen nicht betätigte Taster ihr zugehöriges Bit auf „0“. Wechselt dieses dagegen auf „1“ bedeutet das, dass der Taster gerade gedrückt gehalten wird.

**Die fett gedruckten Werte im obigen Beispiel sind unveränderlich.**

## Modul BDH4800

- Ansteuerung der Relais und LEDs**

Zu verwendender Befehl: **MODBus [0F] → FORCE MULTIPLE COILS**

**Beispiel: Setzen der Relais und LEDs (=Ausgänge, Gruppen 0 bis 2) von Moduladresse „5“ an Stationsadresse 160**

Query Message:

Stations- adresse	Modbus- Kommando	Startadresse		Anzahl Bits		Byte Count	Data Byte 1	Data Byte 2	Data Byte 3	CRC	CRC
		High Byte	Low Byte	High Byte	Low Byte						
A0 h	<b>0F h</b>	00 h	A0 h	<b>00 h</b>	<b>18 h</b>	<b>03 h</b>	02 h	11 h	00 h	XX	XX

Werte aus dem Beispiel:

- A0 h = BZK-Stationsadresse 160
- 0F h** = Kommando FORCE MULTIPLE COILS
- 00 A0 h = Startadresse 160 (Moduladresse 5 x 32 + 0 = 160)
- 00 18 h** = Anzahl Bits = 24
- 03 h** = Anzahl zu sender Datenbytes = 3
- 02 h = Datenbyte 1 (xx00 0010) → LED 1 (!) = EIN (Bits 6 + 7 = xx werden nicht berücksichtigt)
- 11 h = Datenbyte 2 (xx01 0001) → LED 7 + 10 = EIN (Bits 6 + 7 = xx werden nicht berücksichtigt)
- 00 h = Datenbyte 3 (xxxx 0000) → alle Relais = AUS (Bits 4...7 = xxxx werden nicht berücksichtigt)

Response Message:

Stations- adresse	Modbus- Kommando	Startadresse (High Byte)	Startadresse (Low Byte)	Anzahl Bits (High Byte)	Anzahl Bits (Low Byte)	CRC	CRC
A0 h	<b>0F h</b>	00 h	A0 h	<b>00 h</b>	<b>18 h</b>	XX	XX

Werte aus dem Beispiel:

- A0 h = BZK-Stationsadresse 160
- 0F h** = Kommando FORCE MULTIPLE COILS
- 00 A0 h = Startadresse 160 (Moduladresse 5 x 32 + 0 = 160)
- 00 18 h** = Anzahl Bits = 24

Detaillierte Informationen, welches Relais und welche LED von welchem Bit angesteuert wird sind im Anhang aus der Modbus-Crossliste ersichtlich.

Anmerkung: Beim BDH4800 werden die Relais und LEDs mit einer „1“ angesteuert, eine „0“ bedeutet, dass das Relais bzw. die LED nicht angesteuert wird.

**Die fett gedruckten Werte im obigen Beispiel sind unveränderlich.**

## Modul BAH4000

- **Statusabfrage der Schalterstellungen**

Zu verwendender Befehl: **MODBus [02] → READ INPUT STATUS**

**Beispiel: Lesen der Eingänge (Gruppe 0) von Moduladresse „3“ an Stationsadresse 160**

Query Message:

Stations- adresse	Modbus- Kommando	Startadresse (High Byte)	Startadresse (Low Byte)	Anzahl Bits (High Byte)	Anzahl Bits (Low Byte)	CRC	CRC
A0 h	<b>02 h</b>	01 h	18 h	<b>00 h</b>	<b>08 h</b>	XX	XX

Werte aus dem Beispiel:

A0 h = BZK-Stationsadresse 160

**02 h** = Kommando READ INPUT STATUS

01 18 h = Startadresse 280 (Moduladresse 3 x 8 + 256 = 280)

**00 08 h** = Anzahl Bits = 8

Response Message:

Stations- adresse	Modbus- Kommando	Byte Count	Data Byte	CRC	CRC
A0 h	<b>02 h</b>	<b>01 h</b>	04 h	XX	XX

Werte aus dem Beispiel:

A0 h = Adresse 160

**02 h** = Kommando READ INPUT STATUS

**01 h** = Anzahl empfangener Datenbytes = 1

04 h = Datenbyte (xxxx 0100) → Schalter 3 = Hand (Bits 4...7 = xxxx werden nicht berücksichtigt)

Detaillierte Informationen, welcher Schalter welches Bit beeinflusst, sind im Anhang aus der Modbus-Crossliste ersichtlich.

Anmerkung: Beim BAH4000 setzen Schalter in Automatikstellung ihr zugehöriges Bit auf „0“. Wechselt dieses dagegen auf „1“ bedeutet das, dass der Schalter in Stellung Hand (Aus oder Ein) gedreht wurde.

**Die fett gedruckten Werte im obigen Beispiel sind unveränderlich.**

## Modul BAH4000

- **Abfrage der Analogeingänge (Rückmeldung des Istwertes der Analogausgänge)**

Zu verwendender Befehl: **MODBus [04] → READ INPUT REGISTERS**

**Beispiel: lesen der Analogeingänge 1 bis 4 mit der Moduladresse „3“**

Query Message:

Stations- adresse	Modbus- Kommando	Startadresse (High Byte)	Startadresse (Low Byte)	Anzahl Word (High Byte)	Anzahl Word (Low Byte)	CRC	CRC
A0 h	<b>04 h</b>	00 h	0C h	<b>00 h</b>	<b>04 h</b>	XX	XX

Werte aus dem Beispiel:

A0 h = BZK-Stationsadresse 160

**04 h** = Kommando READ INPUT REGISTERS

00 0C h = Startadresse 12 (Moduladresse 3 x 4 + 0 = 12)

**00 04 h** = Anzahl Word, die zu empfangen sind = 4

Response Message:

Stations- adresse	Modbus- Kommando	Byte Count	Word AI 1		Word AI 2		Word AI 3		Word AI 4		CRC	CRC
			High Byte	Low Byte	High Byte	Low Byte	High Byte	Low Byte	High Byte	Low Byte		
A0 h	<b>04 h</b>	<b>08 h</b>	<b>00 h</b>	FF h	<b>00 h</b>	7F h	<b>00 h</b>	3F h	<b>00 h</b>	00 h	XX	XX

Werte aus dem Beispiel:

A0 h = BZK-Stationsadresse 160

**04 h** = Kommando READ INPUT REGISTERS

**08 h** = Anzahl empfangener Datenbytes = 8

FF h = AI 1 (FF h ≡ 255) → 100% ≈ 10,0 Volt

(High Byte = 00 wird nicht genutzt)

7F h = AI 2 (7F h ≡ 127) → 50% ≈ 5,0 Volt

(High Byte = 00 wird nicht genutzt)

3F h = AI 3 (3F h ≡ 63) → 25% ≈ 2,5 Volt

(High Byte = 00 wird nicht genutzt)

00 h = AI 4 (00 h ≡ 0) → 0% ≈ 0,0 Volt

(High Byte = 00 wird nicht genutzt)

**Die fett gedruckten Werte im obigen Beispiel sind unveränderlich.**

## Modul BAH4000

- **Setzen der Analogausgänge**

Zu verwendender Befehl: **MODBus [10] → PRESET MULTIPLE REGISTERS**

**Beispiel: schreiben der Analogausgänge 1 bis 4 mit der Moduladresse „3“**

Query Message:

Stations- adresse	Modbus- Befehl	Start- adresse		Anzahl Word		Byte Count	Word AO 1		Word AO 2		Word AO 3		Word AO 4		CRC	CRC
		High Byte	Low Byte	High Byte	Low Byte		High Byte	Low Byte	High Byte	Low Byte	High Byte	Low Byte	High Byte	Low Byte		
A0 h	<b>10 h</b>	00 h	0C h	<b>00 h</b>	<b>04 h</b>	<b>08 h</b>	<b>00 h</b>	FF h	<b>00 h</b>	7F h	<b>00 h</b>	3F h	<b>00 h</b>	00 h	XX	XX

Werte aus dem Beispiel:

A0 h = BZK-Stationsadresse 160

**10 h** = Kommando PRESET MULTIPLE REGISTERS

00 0C h = Startadresse 12 (Moduladresse 3 x 4 + 0 = 12)

**00 04 h** = Anzahl Word, die zu senden sind = 4

**08 h** = Anzahl zu sendender Datenbytes = 8

FF h = AO 1 (FF h ≡ 255) → 100% ≈ 10,0 Volt

(High Byte = 00 wird nicht genutzt)

7F h = AO 2 (7F h ≡ 127) → 50% ≈ 5,0 Volt

(High Byte = 00 wird nicht genutzt)

3F h = AO 3 (3F h ≡ 63) → 25% ≈ 2,5 Volt

(High Byte = 00 wird nicht genutzt)

00 h = AO 4 (00 h ≡ 0) → 0% ≈ 0,0 Volt

(High Byte = 00 wird nicht genutzt)

Response Message:

Stations- adresse	Modbus- Kommando	Startadresse (High Byte)	Startadresse (Low Byte)	Anzahl Word (High Byte)	Anzahl Word (Low Byte)	CRC	CRC
A0 h	<b>10 h</b>	00 h	0C h	<b>00 h</b>	<b>04 h</b>	XX	XX

Werte aus dem Beispiel:

A0 h = BZK-Stationsadresse 160

**10 h** = Kommando PRESET MULTIPLE REGISTERS

00 0C h = Startadresse 12 (Moduladresse 3 x 4 + 0 = 12)

**00 04 h** = Anzahl Word, die empfangen wurden = 4

**Die fett gedruckten Werte im obigen Beispiel sind unveränderlich.**

## Modul BZK1000MOD

### Zentrales Buskopplermodul mit den Zusatzfunktionen

- **Lampentest**
- **Sammelstörmeldung**
- **externer Hupenkontakt**
- **Taster für Hupe quittieren**
- **Taster für externe Resetfunktion**

- **Statusabfrage der Taster**

Zu verwendender Befehl: **MODBus [02] → READ INPUT STATUS**

**Beispiel: Lesen der Eingänge vom Zentralmodul mit der Stationsadresse 160**

Query Message:

Stations- adresse	Modbus- Kommando	Startadresse (High Byte)	Startadresse (Low Byte)	Anzahl Bits (High Byte)	Anzahl Bits (Low Byte)	CRC	CRC
A0 h	<b>02 h</b>	<b>01 h</b>	<b>40 h</b>	<b>00 h</b>	<b>03 h</b>	XX	XX

Werte aus dem Beispiel:

A0 h = BZK-Stationsadresse 160

**02 h** = Kommando READ INPUT STATUS

**01 40 h** = Startadresse 320

**00 03 h** = Anzahl Bits = 3

Response Message:

Stations- adresse	Modbus- Kommando	Byte Count	Data Byte	CRC	CRC
A0 h	<b>02 h</b>	<b>01 h</b>	04 h	XX	XX

Werte aus dem Beispiel:

A0 h = Adresse 160

**02 h** = Kommando READ INPUT STATUS

**01 h** = Anzahl empfangener Datenbytes = 1

04 h = Datenbyte (xxxx x100) → Taster Lampentest gedrückt (Bits 3...7 = xxxx x werden nicht berücksichtigt)

Detaillierte Informationen, welcher Taster welches Bit beeinflusst, sind im Anhang aus der Modbus-Crossliste ersichtlich.

Anmerkung: Beim BZK1000MOD setzen nicht betätigte Taster ihr zugehöriges Bit auf „0“. Wechselt dieses dagegen auf „1“ bedeutet das, dass der Taster gerade gedrückt gehalten wird.

**Die fett gedruckten Werte im obigen Beispiel sind unveränderlich.**



## Modul BZK1000MOD

- **Ansteuerung der Relais**

**Zu verwendender Befehl:      MODBus [0F] → FORCE MULTIPLE COILS**

**Beispiel: Setzen der Ausgänge vom Zentralmodul mit der Stationsadresse 160**

Query Message:

Stations- adresse	Modbus- Kommando	Startadresse		Anzahl Bits		Byte Count	Data Byte	CRC	CRC
		High Byte	Low Byte	High Byte	Low Byte				
A0 h	<b>0F h</b>	01 h	48 h	<b>00 h</b>	<b>04 h</b>	<b>01 h</b>	08 h	XX	XX

Werte aus dem Beispiel:

A0 h = BZK-Stationsadresse 160

**0F h** = Kommando FORCE MULTIPLE COILS

**01 48 h** = Startadresse 328

**00 04 h** = Anzahl Bits = 4

**01 h** = Anzahl zu sendender Datenbytes = 1

08 h = Datenbyte (xxxx 1000) → Relais Lampentest = EIN (Bits 4...7 = xxxx werden nicht berücksichtigt)

Response Message:

Stations- adresse	Modbus- Kommando	Startadresse (High Byte)	Startadresse (Low Byte)	Anzahl Bits (High Byte)	Anzahl Bits (Low Byte)	CRC	CRC
A0 h	<b>0F h</b>	01 h	48 h	<b>00 h</b>	<b>04 h</b>	XX	XX

Werte aus dem Beispiel:

A0 h = BZK-Stationsadresse 160

**0F h** = Kommando FORCE MULTIPLE COILS

**01 48 h** = Startadresse 328

**00 04 h** = Anzahl Bits = 4

Detaillierte Informationen, welches Relais von welchem Bit angesteuert wird sind im Anhang aus der Modbus-Crossliste ersichtlich.

Anmerkung: Beim BZK1000MOD werden die Relais mit einer „1“ angesteuert, eine „0“ bedeutet, dass das Relais nicht angesteuert wird.

**Die fett gedruckten Werte im obigen Beispiel sind unveränderlich.**

## Modul BZK1000MOD

- Konfiguration des Zentralmoduls über Modbus**

**Zu verwendender Befehl: MODBus [10] → PRESET MULTIPLE REGISTERS**

Die änderbaren Parameter sind in folgender Tabelle aufgeführt:

	<b>Modbus Typ</b>	Registeradresse	Wertebereich
Stationsnummer	Word	32 33 <sup>*)</sup>	0...249  default: [160]
Baudrate	Word	34 35 <sup>*)</sup>	6...9 entsprechend 9600/19200/38400/Auto-Baud  default: [6] = 9600 Baud
Modus	Word	36 37 <sup>*)</sup>	7 = Modbus RTU  default: [7] = Modbus RTU
Schnittstelle	Word	38 39 <sup>*)</sup>	4 = RS485  default: [4] = RS485

<sup>\*)</sup> Um eine Änderung beabsichtigt und korrekt ausführen zu können, ist in dem mit \* gekennzeichneten Word das Komplement des vorherigen Word einzustellen, z.B.:

Busadresse = 160 → Word 32 = 00 A0 h  
 Word 33 = FF 5F h

**Beispiel: Schreiben der neuen Konfiguration (Station 150, Baudrate 9600, Modus = Modbus RTU, Schnittstelle RS485) in das Modul mit der bisherigen (noch aktuellen) Moduladresse „160“.**

**Wichtig: Es müssen alle vier Parameter zusammen mit einem einzigen Befehl geschrieben werden, auch wenn nur ein einziger Wert geändert werden soll!**

Query Message:

Stations- adresse	Modbus- Kommando	Startadresse		Anzahl Word		Byte Count
		High Byte	Low Byte	High Byte	Low Byte	
A0 h	10 h	00 h	20 h	00 h	08 h	10 h

Data 1		NOT Data 1		Data 2		NOT Data 2		Data 3		NOT Data 3		Data 4		NOT Data 4		CRC	CRC
High Byte	Low Byte	High Byte	Low Byte	High Byte	Low Byte	High Byte	Low Byte	High Byte	Low Byte	High Byte	Low Byte	High Byte	Low Byte	High Byte	Low Byte		
00 h	96 h	FF h	69 h	00 h	06 h	FF h	F9 h	00 h	07 h	FF h	F8 h	00 h	04 h	FF h	FB h	XX	XX

Response Message:

Stations- adresse	Modbus- Kommando	Startadresse (High Byte)	Startadresse (Low Byte)	Anzahl Word (High Byte)	Anzahl Word (Low Byte)	CRC	CRC
A0 h	10 h	00 h	20 h	00 h	08 h	XX	XX

BLM 1000			Zieladresse	Type	Modbus-Funktion	Status	Ansteuerung einer LED *)	Bitmuster
LED 1 grün	DO 1	Out 00	Moduladresse x 32 + 0	Coil	15	1 = Ein	01 00 00 00	xx00 0001
LED 1 rot	DO 2	Out 01	Moduladresse x 32 + 1	Coil		1 = Ein	02 00 00 00	xx00 0010
LED 2 grün	DO 3	Out 02	Moduladresse x 32 + 2	Coil	Force	1 = Ein	04 00 00 00	xx00 0100
LED 2 rot	DO 4	Out 03	Moduladresse x 32 + 3	Coil	Multiple	1 = Ein	08 00 00 00	xx00 1000
LED 3 grün	DO 5	Out 04	Moduladresse x 32 + 4	Coil	Coils	1 = Ein	10 00 00 00	xx01 0000
LED 3 rot	DO 6	Out 05	Moduladresse x 32 + 5	Coil		1 = Ein	20 00 00 00	xx10 0000
Reserve	n.c.	Out 06	Moduladresse x 32 + 6	Coil				
Reserve	n.c.	Out 07	Moduladresse x 32 + 7	Coil				
LED 4 grün	DO 7	Out 08	Moduladresse x 32 + 8	Coil	[15]	1 = Ein	00 01 00 00	xx00 0001
LED 4 rot	DO 8	Out 09	Moduladresse x 32 + 9	Coil		1 = Ein	00 02 00 00	xx00 0010
LED 5 grün	DO 9	Out 10	Moduladresse x 32 + 10	Coil		1 = Ein	00 04 00 00	xx00 0100
LED 5 rot	DO 10	Out 11	Moduladresse x 32 + 11	Coil		1 = Ein	00 08 00 00	xx00 1000
LED 6 grün	DO 11	Out 12	Moduladresse x 32 + 12	Coil		1 = Ein	00 10 00 00	xx01 0000
LED 6 rot	DO 12	Out 13	Moduladresse x 32 + 13	Coil		1 = Ein	00 20 00 00	xx10 0000
Reserve	n.c.	Out 14	Moduladresse x 32 + 14	Coil				
Reserve	n.c.	Out 15	Moduladresse x 32 + 15	Coil				
LED 7 grün	DO 13	Out 16	Moduladresse x 32 + 16	Coil	[15]	1 = Ein	00 00 01 00	xx00 0001
LED 7 rot	DO 14	Out 17	Moduladresse x 32 + 17	Coil		1 = Ein	00 00 02 00	xx00 0010
LED 8 grün	DO 15	Out 18	Moduladresse x 32 + 18	Coil		1 = Ein	00 00 04 00	xx00 0100
LED 8 rot	DO 16	Out 19	Moduladresse x 32 + 19	Coil		1 = Ein	00 00 08 00	xx00 1000
LED 9 grün	DO 17	Out 20	Moduladresse x 32 + 20	Coil		1 = Ein	00 00 10 00	xx01 0000
LED 9 rot	DO 18	Out 21	Moduladresse x 32 + 21	Coil		1 = Ein	00 00 20 00	xx10 0000
Reserve	n.c.	Out 22	Moduladresse x 32 + 22	Coil				
Reserve	n.c.	Out 23	Moduladresse x 32 + 23	Coil				
LED 10 grün	DO 19	Out 24	Moduladresse x 32 + 24	Coil	[15]	1 = Ein	00 00 00 01	xxxx xx01
LED 10 rot	DO 20	Out 25	Moduladresse x 32 + 25	Coil		1 = Ein	00 00 00 02	xxxx xx10
Reserve	n.c.	Out 26	Moduladresse x 32 + 26	Coil				
Reserve	n.c.	Out 27	Moduladresse x 32 + 27	Coil				
Reserve	n.c.	Out 28	Moduladresse x 32 + 28	Coil				
Reserve	n.c.	Out 29	Moduladresse x 32 + 29	Coil				
Reserve	n.c.	Out 30	Moduladresse x 32 + 30	Coil				
Reserve	n.c.	Out 31	Moduladresse x 32 + 31	Coil				
*) Die Ansteuerung mehrerer LEDs erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.								

BLM 1001			Quelladresse	Type	Modbus-Funktion	Status	Meldung einer LED *)	Bitmuster
LED 1 grün	DI 1	Inp 00	Moduladresse x 32 + 0	Input	2	0 = EIN	3E 3F 3F 3F	xx11 1110
LED 1 rot	DI 2	Inp 01	Moduladresse x 32 + 1	Input		0 = EIN	3D 3F 3F 3F	xx11 1101
LED 2 grün	DI 3	Inp 02	Moduladresse x 32 + 2	Input	Read	0 = EIN	3B 3F 3F 3F	xx11 1011
LED 2 rot	DI 4	Inp 03	Moduladresse x 32 + 3	Input	Input	0 = EIN	37 3F 3F 3F	xx11 0111
LED 3 grün	DI 5	Inp 04	Moduladresse x 32 + 4	Input	Status	0 = EIN	2F 3F 3F 3F	xx10 1111
LED 3 rot	DI 6	Inp 05	Moduladresse x 32 + 5	Input		0 = EIN	1F 3F 3F 3F	xx01 1111
Reserve	n.c.	Inp 06	Moduladresse x 32 + 6	Input				
Reserve	n.c.	Inp 07	Moduladresse x 32 + 7	Input				
LED 4 grün	DI 7	Inp 08	Moduladresse x 32 + 8	Input	[2]	0 = EIN	3F 3E 3F 3F	xx11 1110
LED 4 rot	DI 8	Inp 09	Moduladresse x 32 + 9	Input		0 = EIN	3F 3D 3F 3F	xx11 1101
LED 5 grün	DI 9	Inp 10	Moduladresse x 32 + 10	Input		0 = EIN	3F 3B 3F 3F	xx11 1011
LED 5 rot	DI 10	Inp 11	Moduladresse x 32 + 11	Input		0 = EIN	3F 37 3F 3F	xx11 0111
LED 6 grün	DI 11	Inp 12	Moduladresse x 32 + 12	Input		0 = EIN	3F 2F 3F 3F	xx10 1111
LED 6 rot	DI 12	Inp 13	Moduladresse x 32 + 13	Input		0 = EIN	3F 1F 3F 3F	xx01 1111
Reserve	n.c.	Inp 14	Moduladresse x 32 + 14	Input				
Reserve	n.c.	Inp 15	Moduladresse x 32 + 15	Input				
LED 7 grün	DI 13	Inp 16	Moduladresse x 32 + 16	Input	[2]	0 = EIN	3F 3F 3E 3F	xx11 1110
LED 7 rot	DI 14	Inp 17	Moduladresse x 32 + 17	Input		0 = EIN	3F 3F 3D 3F	xx11 1101
LED 8 grün	DI 15	Inp 18	Moduladresse x 32 + 18	Input		0 = EIN	3F 3F 3B 3F	xx11 1011
LED 8 rot	DI 16	Inp 19	Moduladresse x 32 + 19	Input		0 = EIN	3F 3F 37 3F	xx11 0111
LED 9 grün	DI 17	Inp 20	Moduladresse x 32 + 20	Input		0 = EIN	3F 3F 2F 3F	xx10 1111
LED 9 rot	DI 18	Inp 21	Moduladresse x 32 + 21	Input		0 = EIN	3F 3F 1F 3F	xx01 1111
Reserve	n.c.	Inp 22	Moduladresse x 32 + 22	Input				
Reserve	n.c.	Inp 23	Moduladresse x 32 + 23	Input				
LED 10 grün	DI 19	Inp 24	Moduladresse x 32 + 24	Input	[2]	0 = EIN	3F 3F 3F 3E	xxxx xx10
LED 10 rot	DI 20	Inp 25	Moduladresse x 32 + 25	Input		0 = EIN	3F 3F 3F 3D	xxxx xx01
Reserve	n.c.	Inp 26	Moduladresse x 32 + 26	Input				
Reserve	n.c.	Inp 27	Moduladresse x 32 + 27	Input				
Reserve	n.c.	Inp 28	Moduladresse x 32 + 28	Input				
Reserve	n.c.	Inp 29	Moduladresse x 32 + 29	Input				
Reserve	n.c.	Inp 30	Moduladresse x 32 + 30	Input				
Reserve	n.c.	Inp 31	Moduladresse x 32 + 31	Input				
*) Die Statusmeldung mehrerer LEDs erfolgt durch Überlagerung der relevanten Bits durch binäres UND.								

BDH 1400			Ziel-/Quelladresse	Type	Funktion	Status	Meldung eines Schalters*)	Bitmuster
Schalter 1	DI 1	Inp 00	Moduladresse x 32 + 0	Input	2	1 = Hand	01	xxxx 0001
Schalter 2	DI 2	Inp 01	Moduladresse x 32 + 1	Input		1 = Hand	02	xxxx 0010
Schalter 3	DI 3	Inp 02	Moduladresse x 32 + 2	Input	Read	1 = Hand	04	xxxx 0100
Schalter 4	DI 4	Inp 03	Moduladresse x 32 + 3	Input	Input	1 = Hand	08	xxxx 1000
Reserve	n.c.	Inp 04	Moduladresse x 32 + 4	Input	Status			
Reserve	n.c.	Inp 05	Moduladresse x 32 + 5	Input				
Reserve	n.c.	Inp 06	Moduladresse x 32 + 6	Input				
Reserve	n.c.	Inp 07	Moduladresse x 32 + 7	Input				
*) Die Statusmeldung mehrerer Schalter erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.								
BDH 1400			Ziel-/Quelladresse	Type	Funktion	Status	Setzen eines Parameters*)	Bitmuster
LED 1 rot	DO 7	Out 08	Moduladresse x 32 + 0	Coil	15	1 = Ein	01 00 00	xx00 0001
LED 1 grün	DO 8	Out 09	Moduladresse x 32 + 1	Coil		1 = Ein	02 00 00	xx00 0010
LED 2 grün	DO 9	Out 10	Moduladresse x 32 + 2	Coil	Force	1 = Ein	04 00 00	xx00 0100
LED 3 rot	DO 10	Out 11	Moduladresse x 32 + 3	Coil	Multiple	1 = Ein	08 00 00	xx00 1000
LED 3 grün	DO 11	Out 12	Moduladresse x 32 + 4	Coil	Coils	1 = Ein	10 00 00	xx01 0000
LED 4 grün	DO 12	Out 13	Moduladresse x 32 + 5	Coil		1 = Ein	20 00 00	xx10 0000
Reserve	n.c.	Out 14	Moduladresse x 32 + 6	Coil				
Reserve	n.c.	Out 15	Moduladresse x 32 + 7	Coil				
LED 5 rot	DO 13	Out 16	Moduladresse x 32 + 8	Coil	[15]	1 = Ein	00 01 00	xx00 0001
LED 5 grün	DO 14	Out 17	Moduladresse x 32 + 9	Coil		1 = Ein	00 02 00	xx00 0010
LED 6 grün	DO 15	Out 18	Moduladresse x 32 + 10	Coil		1 = Ein	00 04 00	xx00 0100
LED 7 rot	DO 16	Out 19	Moduladresse x 32 + 11	Coil		1 = Ein	00 08 00	xx00 1000
LED 7 grün	DO 17	Out 20	Moduladresse x 32 + 12	Coil		1 = Ein	00 10 00	xx01 0000
LED 8 grün	DO 18	Out 21	Moduladresse x 32 + 13	Coil		1 = Ein	00 20 00	xx10 0000
Reserve	n.c.	Out 22	Moduladresse x 32 + 14	Coil				
Reserve	n.c.	Out 23	Moduladresse x 32 + 15	Coil				
Relais 1	DO 1	Out 00	Moduladresse x 32 + 16	Coil	[15]	1 = Ein	00 00 01	xxxx 0001
Relais 2	DO 2	Out 01	Moduladresse x 32 + 17	Coil		1 = Ein	00 00 02	xxxx 0010
Relais 3	DO 3	Out 02	Moduladresse x 32 + 18	Coil		1 = Ein	00 00 04	xxxx 0100
Relais 4	DO 4	Out 03	Moduladresse x 32 + 19	Coil		1 = Ein	00 00 08	xxxx 1000
Reserve	n.c.	Out 04	Moduladresse x 32 + 20	Coil				
Reserve	n.c.	Out 05	Moduladresse x 32 + 21	Coil				
Reserve	n.c.	Out 06	Moduladresse x 32 + 22	Coil				
Reserve	n.c.	Out 07	Moduladresse x 32 + 23	Coil				
*) Die Ansteuerung mehrerer LEDs/Relais erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.								

BDH 1401			Ziel-/Quelladresse	Type	Funktion	Status	Meldung eines Parameters*)	Bitmuster
Schalter 1	DI 1	Inp 00	Moduladresse x 32 + 0	Input	<b>2</b>	1 = Hand	<b>31</b> 3F 3F	xxxx 0001
Schalter 2	DI 2	Inp 01	Moduladresse x 32 + 1	Input		1 = Hand	<b>32</b> 3F 3F	xxxx 0010
Schalter 3	DI 3	Inp 02	Moduladresse x 32 + 2	Input	<b>Read</b>	1 = Hand	<b>34</b> 3F 3F	xxxx 0100
Schalter 4	DI 4	Inp 03	Moduladresse x 32 + 3	Input	<b>Input</b>	1 = Hand	<b>38</b> 3F 3F	xxxx 1000
Reserve	n.c.	Inp 04	Moduladresse x 32 + 4	Input	<b>Status</b>			
Reserve	n.c.	Inp 05	Moduladresse x 32 + 5	Input				
Reserve	n.c.	Inp 06	Moduladresse x 32 + 6	Input				
Reserve	n.c.	Inp 07	Moduladresse x 32 + 7	Input				
LED 1 rot	DI 7	Inp 08	Moduladresse x 32 + 8	Input	[2]	0 = EIN	30 <b>3E</b> 3F	xx11 1110
LED 2 grün	DI 8	Inp 09	Moduladresse x 32 + 9	Input		0 = EIN	30 <b>3D</b> 3F	xx11 1101
LED 3 rot	DI 9	Inp 10	Moduladresse x 32 + 10	Input		0 = EIN	30 <b>3B</b> 3F	xx11 1011
LED 4 grün	DI 10	Inp 11	Moduladresse x 32 + 11	Input		0 = EIN	30 <b>37</b> 3F	xx11 0111
LED 5 rot	DI 11	Inp 12	Moduladresse x 32 + 12	Input		0 = EIN	30 <b>2F</b> 3F	xx10 1111
LED 6 grün	DI 12	Inp 13	Moduladresse x 32 + 13	Input		0 = EIN	30 <b>1F</b> 3F	xx01 1111
Reserve	n.c.	Inp 14	Moduladresse x 32 + 14	Input				
Reserve	n.c.	Inp 15	Moduladresse x 32 + 15	Input				
LED 7 rot	DI 13	Inp 16	Moduladresse x 32 + 16	Input	[2]	0 = EIN	30 3F <b>3E</b>	xxxx xx10
LED 8 grün	DI 14	Inp 17	Moduladresse x 32 + 17	Input		0 = EIN	30 3F <b>3D</b>	xxxx xx01
Reserve	n.c.	Inp 18	Moduladresse x 32 + 18	Input				
Reserve	n.c.	Inp 19	Moduladresse x 32 + 19	Input				
Reserve	n.c.	Inp 20	Moduladresse x 32 + 20	Input				
Reserve	n.c.	Inp 21	Moduladresse x 32 + 21	Input				
Reserve	n.c.	Inp 22	Moduladresse x 32 + 22	Input				
Reserve	n.c.	Inp 23	Moduladresse x 32 + 23	Input				
*) Die Statusmeldung mehrerer Schalter/LEDs erfolgt durch Überlagerung der relevanten Bits durch binäres UND.								
BDH 1401			Ziel-/Quelladresse	Type	Funktion	Status	Ansteuerung eines Relais*)	Bitmuster
Relais 1	DO 1	Out 00	Moduladresse x 32 + 0	Coil	<b>15</b>	1 = EIN	<b>01</b>	xxxx 0001
Relais 2	DO 2	Out 01	Moduladresse x 32 + 1	Coil		1 = EIN	<b>02</b>	xxxx 0010
Relais 3	DO 3	Out 02	Moduladresse x 32 + 2	Coil	<b>Force</b>	1 = EIN	<b>04</b>	xxxx 0100
Relais 4	DO 4	Out 03	Moduladresse x 32 + 3	Coil	<b>Multiple</b>	1 = EIN	<b>08</b>	xxxx 1000
Reserve	n.c.	Out 04	Moduladresse x 32 + 4	Coil	<b>Coils</b>			
Reserve	n.c.	Out 05	Moduladresse x 32 + 5	Coil				
Reserve	n.c.	Out 06	Moduladresse x 32 + 6	Coil				
Reserve	n.c.	Out 07	Moduladresse x 32 + 7	Coil				
*) Die Ansteuerung mehrerer Relais erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.								

BDH 2200								Meldung eines	Bitmuster
			Ziel-/Quelladresse	Type	Funktion	Status	Schalters*)		
Schalter 1	DI 1	Inp 00	Moduladresse x 32 + 0	Input	2	1 = Hand	01	xxxx xx01	
Schalter 2	DI 2	Inp 01	Moduladresse x 32 + 1	Input		1 = Hand	02	xxxx xx10	
Reserve	n.c.	Inp 02	Moduladresse x 32 + 2	Input	Read				
Reserve	n.c.	Inp 03	Moduladresse x 32 + 3	Input	Input				
Reserve	n.c.	Inp 04	Moduladresse x 32 + 4	Input	Status				
Reserve	n.c.	Inp 05	Moduladresse x 32 + 5	Input					
Reserve	n.c.	Inp 06	Moduladresse x 32 + 6	Input					
Reserve	n.c.	Inp 07	Moduladresse x 32 + 7	Input					
*) Die Statusmeldung mehrerer Schalter erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.									
BDH 2200								Setzen eines	Bitmuster
			Ziel-/Quelladresse	Type	Funktion	Status	Parmeters*)		
LED 1 rot	DO 7	Out 08	Moduladresse x 32 + 0	Coil	15	1 = Ein	01 00 00	xx00 0001	
LED 1 grün	DO 8	Out 09	Moduladresse x 32 + 1	Coil		1 = Ein	02 00 00	xx00 0010	
LED 2 grün	DO 9	Out 10	Moduladresse x 32 + 2	Coil	Force	1 = Ein	04 00 00	xx00 0100	
LED 3 grün	DO 10	Out 11	Moduladresse x 32 + 3	Coil	Multiple	1 = Ein	08 00 00	xx00 1000	
LED 4 rot	DO 11	Out 12	Moduladresse x 32 + 4	Coil	Coils	1 = Ein	10 00 00	xx01 0000	
LED 4 grün	DO 12	Out 13	Moduladresse x 32 + 5	Coil		1 = Ein	20 00 00	xx10 0000	
Reserve	n.c.	Out 14	Moduladresse x 32 + 6	Coil					
Reserve	n.c.	Out 15	Moduladresse x 32 + 7	Coil					
LED 5 grün	DO 13	Out 16	Moduladresse x 32 + 8	Coil	[15]	1 = Ein	00 01 00	xxxx xx01	
LED 6 grün	DO 14	Out 17	Moduladresse x 32 + 9	Coil		1 = Ein	00 02 00	xxxx xx10	
Reserve	n.c.	Out 18	Moduladresse x 32 + 10	Coil					
Reserve	n.c.	Out 19	Moduladresse x 32 + 11	Coil					
Reserve	n.c.	Out 20	Moduladresse x 32 + 12	Coil					
Reserve	n.c.	Out 21	Moduladresse x 32 + 13	Coil					
Reserve	n.c.	Out 22	Moduladresse x 32 + 14	Coil					
Reserve	n.c.	Out 23	Moduladresse x 32 + 15	Coil					
Relais 1	DO 1	Out 00	Moduladresse x 32 + 16	Coil	[15]	1 = Ein	00 00 01	xxxx 0001	
Relais 2	DO 2	Out 01	Moduladresse x 32 + 17	Coil		1 = Ein	00 00 02	xxxx 0010	
Relais 3	DO 3	Out 02	Moduladresse x 32 + 18	Coil		1 = Ein	00 00 04	xxxx 0100	
Relais 4	DO 4	Out 03	Moduladresse x 32 + 19	Coil		1 = Ein	00 00 08	xxxx 1000	
Reserve	n.c.	Out 04	Moduladresse x 32 + 20	Coil					
Reserve	n.c.	Out 05	Moduladresse x 32 + 21	Coil					
Reserve	n.c.	Out 06	Moduladresse x 32 + 22	Coil					
Reserve	n.c.	Out 07	Moduladresse x 32 + 23	Coil					
*) Die Ansteuerung mehrerer LEDs/Relais erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.									

BDH 2201			Ziel-/Quelladresse	Type	Funktion	Status	Meldung eines Parameters*)	Bitmuster
Schalter 1	DI 1	Inp 00	Moduladresse x 32 + 0	Input	2	1 = Hand	31 3F 3F	xxxx xx01
Schalter 2	DI 2	Inp 01	Moduladresse x 32 + 1	Input		1 = Hand	32 3F 3F	xxxx xx10
Reserve	n.c.	Inp 02	Moduladresse x 32 + 2	Input	Read			
Reserve	n.c.	Inp 03	Moduladresse x 32 + 3	Input	Input			
Reserve	n.c.	Inp 04	Moduladresse x 32 + 4	Input	Status			
Reserve	n.c.	Inp 05	Moduladresse x 32 + 5	Input				
Reserve	n.c.	Inp 06	Moduladresse x 32 + 6	Input				
Reserve	n.c.	Inp 07	Moduladresse x 32 + 7	Input				
LED 1 rot	DI 7	Inp 08	Moduladresse x 32 + 8	Input	[2]	0 = EIN	30 3E 3F	xx11 1110
LED 2 grün	DI 8	Inp 09	Moduladresse x 32 + 9	Input		0 = EIN	30 3D 3F	xx11 1101
LED 3 grün	DI 9	Inp 10	Moduladresse x 32 + 10	Input		0 = EIN	30 3B 3F	xx11 1011
LED 4 rot	DI 10	Inp 11	Moduladresse x 32 + 11	Input		0 = EIN	30 37 3F	xx11 0111
LED 5 grün	DI 11	Inp 12	Moduladresse x 32 + 12	Input		0 = EIN	30 2F 3F	xx10 1111
LED 6 grün	DI 12	Inp 13	Moduladresse x 32 + 13	Input		0 = EIN	30 1F 3F	xx01 1111
Reserve	n.c.	Inp 14	Moduladresse x 32 + 14	Input				
Reserve	n.c.	Inp 15	Moduladresse x 32 + 15	Input				
Reserve	n.c.	Inp 16	Moduladresse x 32 + 16	Input	[2]		30 3F XX**)	xxxx xxxx
Reserve	n.c.	Inp 17	Moduladresse x 32 + 17	Input				
Reserve	n.c.	Inp 18	Moduladresse x 32 + 18	Input				
Reserve	n.c.	Inp 19	Moduladresse x 32 + 19	Input				
Reserve	n.c.	Inp 20	Moduladresse x 32 + 20	Input				
Reserve	n.c.	Inp 21	Moduladresse x 32 + 21	Input				
Reserve	n.c.	Inp 22	Moduladresse x 32 + 22	Input				
Reserve	n.c.	Inp 23	Moduladresse x 32 + 23	Input				
*) Die Statusmeldung mehrerer Schalter/LEDs erfolgt durch Überlagerung der relevanten Bits durch binäres UND.								
**) Datenbyte Nr. 3 muss mit ausgelesen werden!								
BDH 2201			Ziel-/Quelladresse	Type	Funktion	Status	Ansteuerung eines Relais*)	Bitmuster
Relais 1	DO 1	Out 00	Moduladresse x 32 + 0	Coil	15	1 = EIN	01	xxxx 0001
Relais 2	DO 2	Out 01	Moduladresse x 32 + 1	Coil		1 = EIN	02	xxxx 0010
Relais 3	DO 3	Out 02	Moduladresse x 32 + 2	Coil	Force	1 = EIN	04	xxxx 0100
Relais 4	DO 4	Out 03	Moduladresse x 32 + 3	Coil	Multiple Coils	1 = EIN	08	xxxx 1000
Reserve	n.c.	Out 04	Moduladresse x 32 + 4	Coil				
Reserve	n.c.	Out 05	Moduladresse x 32 + 5	Coil				
Reserve	n.c.	Out 06	Moduladresse x 32 + 6	Coil				
Reserve	n.c.	Out 07	Moduladresse x 32 + 7	Coil				
*) Die Ansteuerung mehrerer Relais erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.								



BDH 4800			Ziel-/Quelladresse	Type	Funktion	Status	Meldung eines Tasters*)	Bitmuster
Taster 1	DI 1	Inp 00	Moduladresse x 32 + 0	Input	2	1 = betät.	01	xxxx 0001
Taster 2	DI 2	Inp 01	Moduladresse x 32 + 1	Input		1 = betät.	02	xxxx 0010
Taster 3	DI 3	Inp 02	Moduladresse x 32 + 2	Input	Read	1 = betät.	04	xxxx 0100
Taster 4	DI 4	Inp 03	Moduladresse x 32 + 3	Input	Input	1 = betät.	08	xxxx 1000
Reserve	n.c.	Inp 04	Moduladresse x 32 + 4	Input	Status			
Reserve	n.c.	Inp 05	Moduladresse x 32 + 5	Input				
Reserve	n.c.	Inp 06	Moduladresse x 32 + 6	Input				
Reserve	n.c.	Inp 07	Moduladresse x 32 + 7	Input				
*) Die Statusmeldung mehrerer Taster erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.								
BDH 4800			Ziel-/Quelladresse	Type	Funktion	Status	Setzen eines Parameters*)	Bitmuster
LED 1 orange	DO 7	Out 08	Moduladresse x 32 + 0	Coil	15	1 = Ein	02 00 00	xx00 0010
LED 2 rot-blink.	DO 8	Out 09	Moduladresse x 32 + 1	Coil		1 = Ein	01 00 00	xx00 0001
LED 3 rot	DO 9	Out 10	Moduladresse x 32 + 2	Coil	Force	1 = Ein	04 00 00	xx00 0100
LED 4 orange	DO 10	Out 11	Moduladresse x 32 + 3	Coil	Multiple	1 = Ein	08 00 00	xx00 1000
LED 5 rot-blink.	DO 11	Out 12	Moduladresse x 32 + 4	Coil	Coils	1 = Ein	10 00 00	xx01 0000
LED 6 rot	DO 12	Out 13	Moduladresse x 32 + 5	Coil		1 = Ein	20 00 00	xx10 0000
Reserve	n.c.	Out 14	Moduladresse x 32 + 6	Coil				
Reserve	n.c.	Out 15	Moduladresse x 32 + 7	Coil				
LED 7 orange	DO 13	Out 16	Moduladresse x 32 + 8	Coil	[15]	1 = Ein	00 01 00	xx00 0001
LED 8 rot-blink.	DO 14	Out 17	Moduladresse x 32 + 9	Coil		1 = Ein	00 02 00	xx00 0010
LED 9 rot	DO 15	Out 18	Moduladresse x 32 + 10	Coil		1 = Ein	00 04 00	xx00 0100
LED 10 orange	DO 16	Out 19	Moduladresse x 32 + 11	Coil		1 = Ein	00 10 00	xx01 0000
LED 11 rot-blink.	DO 17	Out 20	Moduladresse x 32 + 12	Coil		1 = Ein	00 08 00	xx00 1000
LED 12 rot	DO 18	Out 21	Moduladresse x 32 + 13	Coil		1 = Ein	00 20 00	xx10 0000
Reserve	n.c.	Out 22	Moduladresse x 32 + 14	Coil				
Reserve	n.c.	Out 23	Moduladresse x 32 + 15	Coil				
Relais 1	DO 1	Out 00	Moduladresse x 32 + 16	Coil	[15]	1 = Ein	00 00 01	xxxx 0001
Relais 2	DO 2	Out 01	Moduladresse x 32 + 17	Coil		1 = Ein	00 00 02	xxxx 0010
Relais 3	DO 3	Out 02	Moduladresse x 32 + 18	Coil		1 = Ein	00 00 04	xxxx 0100
Relais 4	DO 4	Out 03	Moduladresse x 32 + 19	Coil		1 = Ein	00 00 08	xxxx 1000
Reserve	n.c.	Out 04	Moduladresse x 32 + 20	Coil				
Reserve	n.c.	Out 05	Moduladresse x 32 + 21	Coil				
Reserve	n.c.	Out 06	Moduladresse x 32 + 22	Coil				
Reserve	n.c.	Out 07	Moduladresse x 32 + 23	Coil				
*) Die Ansteuerung mehrerer LEDs/Relais erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.								

BAH 4000			Ziel-/Quelladresse	Type	Funktion	Status	Meldung eines Schalters*)	Bitmuster
<b>Digitaleingänge</b>								
Input 1	DI 1	Inp 00	Moduladresse x 8 + 256	Input	<b>2</b>	1 = Hand	<b>01</b>	xxxx 0001
Input 2	DI 2	Inp 01	Moduladresse x 8 + 257	Input		1 = Hand	<b>02</b>	xxxx 0010
Input 3	DI 3	Inp 02	Moduladresse x 8 + 258	Input	<b>Read</b>	1 = Hand	<b>04</b>	xxxx 0100
Input 4	DI 4	Inp 03	Moduladresse x 8 + 259	Input	<b>Input</b>	1 = Hand	<b>08</b>	xxxx 1000
Reserve	n.c.	Inp 04	Moduladresse x 8 + 260	Input	<b>Status</b>			
Reserve	n.c.	Inp 05	Moduladresse x 8 + 261	Input				
Reserve	n.c.	Inp 06	Moduladresse x 8 + 262	Input				
Reserve	n.c.	Inp 07	Moduladresse x 8 + 263	Input				
*) Die Statusmeldung mehrerer Schalter erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.								
BAH 4000			Ziel-/Quelladresse	Type	Funktion	Status		
<b>Analogeingänge</b>								
Input 1			Moduladresse x 4 + 0	Word	<b>4</b>	0 ... 255	(Low-Byte, High-Byte = 00)	
Input 2			Moduladresse x 4 + 1	Word	<b>Read</b>	0 ... 255	(Low-Byte, High-Byte = 00)	
Input 3			Moduladresse x 4 + 2	Word	<b>Input</b>	0 ... 255	(Low-Byte, High-Byte = 00)	
Input 4			Moduladresse x 4 + 3	Word	<b>Registers</b>	0 ... 255	(Low-Byte, High-Byte = 00)	
BAH 4000			Ziel-/Quelladresse	Type	Funktion	Status		
<b>Analogausgänge</b>								
Output 1			Moduladresse x 4 + 0	Word	<b>16</b>	0 ... 255	(Low-Byte, High-Byte = 00)	
Output 2			Moduladresse x 4 + 1	Word	<b>Preset</b>	0 ... 255	(Low-Byte, High-Byte = 00)	
Output 3			Moduladresse x 4 + 2	Word	<b>Multiple</b>	0 ... 255	(Low-Byte, High-Byte = 00)	
Output 4			Moduladresse x 4 + 3	Word	<b>Registers</b>	0 ... 255	(Low-Byte, High-Byte = 00)	

BZK 1000			Ziel-/Quelladresse	Type	Funktion	Status	Meldung eines Tasters*)	Bitmuster
<b>Digitaleingänge</b>								
Entsperren	DI 1	Inp 00	320	Input	<b>2</b>	1 = betät.	<b>01</b>	xxxx x001
Hupe quittieren	DI 2	Inp 01	321	Input	<b>Read</b>	1 = betät.	<b>02</b>	xxxx x010
Lampentest	DI 3	Inp 02	322	Input	<b>Input Status</b>	1 = betät.	<b>04</b>	xxxx x100
*) Die Statusmeldung mehrerer Taster erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.								
BZK 1000			Ziel-/Quelladresse	Type	Funktion	Status	Ansteuerung eines Relais*)	Bitmuster
<b>Digitalausgänge</b>								
Reset	DO 1	Out 00	328	Coil	<b>15</b>	1 = Ein	<b>01</b>	xxxx 0001
Sammelalarm	DO 2	Out 01	329	Coil	<b>Force</b>	1 = Ein	<b>02</b>	xxxx 0010
Hupenkontakt	DO 3	Out 02	330	Coil	<b>Multiple</b>	1 = Ein	<b>04</b>	xxxx 0100
Lampentest	DO 4	Out 03	331	Coil	<b>Coils</b>	1 = Ein	<b>08</b>	xxxx 1000
*) Die Ansteuerung mehrerer Relais erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.								
<b>Allgemeine</b>			<b>Defaultwerte</b>					
<b>Konfiguration</b>	<b>Wertbereich</b>	<b>Zieladresse</b>	<b>Type</b>	<b>Funktion</b>	<b>dezimal</b>	<b>hexadezimal</b>		
<b>Stationsnummer</b>	0..249	32: <Station> 33: NOT <Station>	Word	<b>16</b>	<b>d 160</b>	00 A0 FF 5F		
<b>Baudrate</b>	6 = 9600 Baud 7 = 19200 Baud 8 = 38400 Baud	34: <Baud> 35: NOT <Baud>	Word	[16]	<b>d 6</b>	00 06 FF F9		
<b>Modus</b>	7=Modbus RTU	36: <Modus> 37: NOT <Modus>	Word	[16]	<b>d 7</b>	00 07 FF F8		
<b>Schnittstelle</b>	4 = RS485	38: <Uart> 39: NOT <Uart>	Word	[16]	<b>d 4</b>	00 04 FF FB		
<b>Wichtig: Es müssen alle vier Parameter (R32...R39) zusammen mit einem einzigen Befehl geschrieben werden, auch wenn nur ein einziger Wert geändert werden soll!</b>								
<b>Busaktivitäts-Überwachung</b>	Zeit in Sek.	42: <Zeit> 43: NOT <Zeit>	Word	<b>16</b>	<b>d 10</b>	00 0A FF F5		

BLM 1000			Zieladresse (Register)	Type	Modbus-Funktion	Status	Ansteuerung einer LED *)	Bitmuster	
LED 1 grün	DO 1	Out 00	Moduladresse x 4 + 80	Word	16	1 = Ein	h 01 00 00 00	xx00 0001	
LED 1 rot	DO 2	Out 01				1 = Ein	h 02 00 00 00	xx00 0010	
LED 2 grün	DO 3	Out 02				Preset Multiple Registers	1 = Ein	h 04 00 00 00	xx00 0100
LED 2 rot	DO 4	Out 03					1 = Ein	h 08 00 00 00	xx00 1000
LED 3 grün	DO 5	Out 04					1 = Ein	h 10 00 00 00	xx01 0000
LED 3 rot	DO 6	Out 05					1 = Ein	h 20 00 00 00	xx10 0000
Reserve	n.c.	Out 06							
Reserve	n.c.	Out 07							
LED 4 grün	DO 7	Out 08	Moduladresse x 4 + 81	Word	[16]	1 = Ein	h 00 01 00 00	xx00 0001	
LED 4 rot	DO 8	Out 09				1 = Ein	h 00 02 00 00	xx00 0010	
LED 5 grün	DO 9	Out 10				1 = Ein	h 00 04 00 00	xx00 0100	
LED 5 rot	DO 10	Out 11				1 = Ein	h 00 08 00 00	xx00 1000	
LED 6 grün	DO 11	Out 12				1 = Ein	h 00 10 00 00	xx01 0000	
LED 6 rot	DO 12	Out 13				1 = Ein	h 00 20 00 00	xx10 0000	
Reserve	n.c.	Out 14							
Reserve	n.c.	Out 15							
LED 7 grün	DO 13	Out 16	Moduladresse x 4 + 82	Word	[16]	1 = Ein	h 00 00 01 00	xx00 0001	
LED 7 rot	DO 14	Out 17				1 = Ein	h 00 00 02 00	xx00 0010	
LED 8 grün	DO 15	Out 18				1 = Ein	h 00 00 04 00	xx00 0100	
LED 8 rot	DO 16	Out 19				1 = Ein	h 00 00 08 00	xx00 1000	
LED 9 grün	DO 17	Out 20				1 = Ein	h 00 00 10 00	xx01 0000	
LED 9 rot	DO 18	Out 21				1 = Ein	h 00 00 20 00	xx10 0000	
Reserve	n.c.	Out 22							
Reserve	n.c.	Out 23							
LED 10 grün	DO 19	Out 24	Moduladresse x 4 + 83	Word	[16]	1 = Ein	h 00 00 00 01	xxxx xx01	
LED 10 rot	DO 20	Out 25				1 = Ein	h 00 00 00 02	xxxx xx10	
Reserve	n.c.	Out 26							
Reserve	n.c.	Out 27							
Reserve	n.c.	Out 28							
Reserve	n.c.	Out 29							
Reserve	n.c.	Out 30							
Reserve	n.c.	Out 31							
*) Die Ansteuerung mehrerer LEDs erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.									

BLM 1001			Quelladresse (Register)	Type	Modbus-Funktion	Status	Meldung einer LED *)	Bitmuster	
LED 1 grün	DI 1	Inp 00	Moduladresse x 4 + 80	Word	4	1 = Ein	h 01 00 00 00	xx00 0001	
LED 1 rot	DI 2	Inp 01				1 = Ein	h 02 00 00 00	xx00 0010	
LED 2 grün	DI 3	Inp 02				1 = Ein	h 04 00 00 00	xx00 0100	
LED 2 rot	DI 4	Inp 03				1 = Ein	h 08 00 00 00	xx00 1000	
LED 3 grün	DI 5	Inp 04				1 = Ein	h 10 00 00 00	xx01 0000	
LED 3 rot	DI 6	Inp 05				1 = Ein	h 20 00 00 00	xx10 0000	
Reserve	n.c.	Inp 06							
Reserve	n.c.	Inp 07							
LED 4 grün	DI 7	Inp 08	Moduladresse x 4 + 81	Word	[4]	1 = Ein	h 00 01 00 00	xx00 0001	
LED 4 rot	DI 8	Inp 09				1 = Ein	h 00 02 00 00	xx00 0010	
LED 5 grün	DI 9	Inp 10				1 = Ein	h 00 04 00 00	xx00 0100	
LED 5 rot	DI 10	Inp 11				1 = Ein	h 00 08 00 00	xx00 1000	
LED 6 grün	DI 11	Inp 12				1 = Ein	h 00 10 00 00	xx01 0000	
LED 6 rot	DI 12	Inp 13				1 = Ein	h 00 20 00 00	xx10 0000	
Reserve	n.c.	Inp 14							
Reserve	n.c.	Inp 15							
LED 7 grün	DI 13	Inp 16	Moduladresse x 4 + 82	Word	[4]	1 = Ein	h 00 00 01 00	xx00 0001	
LED 7 rot	DI 14	Inp 17				1 = Ein	h 00 00 02 00	xx00 0010	
LED 8 grün	DI 15	Inp 18				1 = Ein	h 00 00 04 00	xx00 0100	
LED 8 rot	DI 16	Inp 19				1 = Ein	h 00 00 08 00	xx00 1000	
LED 9 grün	DI 17	Inp 20				1 = Ein	h 00 00 10 00	xx01 0000	
LED 9 rot	DI 18	Inp 21				1 = Ein	h 00 00 20 00	xx10 0000	
Reserve	n.c.	Inp 22							
Reserve	n.c.	Inp 23							
LED 10 grün	DI 19	Inp 24	Moduladresse x 4 + 83	Word	[4]	1 = Ein	h 00 00 00 01	xxxx xx01	
LED 10 rot	DI 20	Inp 25				1 = Ein	h 00 00 00 02	xxxx xx10	
Reserve	n.c.	Inp 26							
Reserve	n.c.	Inp 27							
Reserve	n.c.	Inp 28							
Reserve	n.c.	Inp 29							
Reserve	n.c.	Inp 30							
Reserve	n.c.	Inp 31							
*) Die Statusmeldung mehrerer LEDs erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.									

BDH 1400			Ziel-/Quelladresse	Type	Funktion	Status	Meldung eines Schalters*)	Bitmuster	
Schalter 1	DI 1	Inp 00	Moduladresse x 4 + 80	Word	4	1 = Auto	<b>h 01</b>	xxxx 0001	
Schalter 2	DI 2	Inp 01				1 = Auto	<b>h 02</b>	xxxx 0010	
Schalter 3	DI 3	Inp 02				Read	1 = Auto	<b>h 04</b>	xxxx 0100
Schalter 4	DI 4	Inp 03				Input	1 = Auto	<b>h 08</b>	xxxx 1000
Reserve	n.c.	Inp 04			Registers				
Reserve	n.c.	Inp 05							
Reserve	n.c.	Inp 06							
Reserve	n.c.	Inp 07							
*) Die Statusmeldung mehrerer Schalter erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.									
BDH 1400			Ziel-/Quelladresse	Type	Funktion	Status	Setzen eines Parameters*)	Bitmuster	
LED 1 rot	DO 7	Out 08	Moduladresse x 4 + 80	Word	16	1 = Ein	<b>h 01 00 00</b>	xx00 0001	
LED 1 grün	DO 8	Out 09				1 = Ein	<b>h 02 00 00</b>	xx00 0010	
LED 2 grün	DO 9	Out 10				Preset	1 = Ein	<b>h 04 00 00</b>	xx00 0100
LED 3 rot	DO 10	Out 11				Multiple	1 = Ein	<b>h 08 00 00</b>	xx00 1000
LED 3 grün	DO 11	Out 12		Registers	1 = Ein	<b>h 10 00 00</b>	xx01 0000		
LED 4 grün	DO 12	Out 13			1 = Ein	<b>h 20 00 00</b>	xx10 0000		
Reserve	n.c.	Out 14							
Reserve	n.c.	Out 15							
LED 5 rot	DO 13	Out 16	Moduladresse x 4 + 81	Word	[16]	1 = Ein	<b>h 00 01 00</b>	xx00 0001	
LED 5 grün	DO 14	Out 17				1 = Ein	<b>h 00 02 00</b>	xx00 0010	
LED 6 grün	DO 15	Out 18				1 = Ein	<b>h 00 04 00</b>	xx00 0100	
LED 7 rot	DO 16	Out 19				1 = Ein	<b>h 00 08 00</b>	xx00 1000	
LED 7 grün	DO 17	Out 20			1 = Ein	<b>h 00 10 00</b>	xx01 0000		
LED 8 grün	DO 18	Out 21			1 = Ein	<b>h 00 20 00</b>	xx10 0000		
Reserve	n.c.	Out 22							
Reserve	n.c.	Out 23							
Relais 1	DO 1	Out 00	Moduladresse x 4 + 82	Word	[16]	1 = Ein	<b>h 00 00 01</b>	xxxx 0001	
Relais 2	DO 2	Out 01				1 = Ein	<b>h 00 00 02</b>	xxxx 0010	
Relais 3	DO 3	Out 02				1 = Ein	<b>h 00 00 04</b>	xxxx 0100	
Relais 4	DO 4	Out 03				1 = Ein	<b>h 00 00 08</b>	xxxx 1000	
Reserve	n.c.	Out 04							
Reserve	n.c.	Out 05							
Reserve	n.c.	Out 06							
Reserve	n.c.	Out 07							
*) Die Ansteuerung mehrerer LEDs/Relais erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.									

BDH 1401			Ziel-/Quelladresse	Type	Funktion	Status	Meldung eines Parameters*)	Bitmuster		
Schalter 1	DI 1	Inp 00	Moduladresse x 4 + 80	Word	4	1 = Auto	<b>h 01 00 00</b>	xxxx 0001		
Schalter 2	DI 2	Inp 01				1 = Auto	<b>h 02 00 00</b>	xxxx 0010		
Schalter 3	DI 3	Inp 02				Read Input	1 = Auto	<b>h 04 00 00</b>	xxxx 0100	
Schalter 4	DI 4	Inp 03					1 = Auto	<b>h 08 00 00</b>	xxxx 1000	
Reserve	n.c.	Inp 04		Registers						
Reserve	n.c.	Inp 05								
Reserve	n.c.	Inp 06								
Reserve	n.c.	Inp 07								
LED 1 rot	DI 7	Inp 08	Moduladresse x 4 + 81	Word	[4]	1 = Ein	h 00 <b>01</b> 00	xx00 000 <b>1</b>		
LED 2 grün	DI 8	Inp 09				1 = Ein	h 00 <b>02</b> 00	xx00 0010		
LED 3 rot	DI 9	Inp 10				1 = Ein	h 00 <b>04</b> 00	xx00 0 <b>1</b> 00		
LED 4 grün	DI 10	Inp 11				1 = Ein	h 00 <b>08</b> 00	xx00 1000		
LED 5 rot	DI 11	Inp 12				1 = Ein	h 00 <b>10</b> 00	xx <b>01</b> 0000		
LED 6 grün	DI 12	Inp 13				1 = Ein	h 00 <b>20</b> 00	xx10 0000		
Reserve	n.c.	Inp 14								
Reserve	n.c.	Inp 15								
LED 7 rot	DI 13	Inp 16	Moduladresse x 4 + 82	Word	[4]	1 = Ein	h 00 00 <b>01</b>	xx00 000 <b>1</b>		
LED 8 grün	DI 14	Inp 17				1 = Ein	h 00 00 <b>02</b>	xx00 0010		
Reserve	n.c.	Inp 18								
Reserve	n.c.	Inp 19								
Reserve	n.c.	Inp 20								
Reserve	n.c.	Inp 21								
Reserve	n.c.	Inp 22								
Reserve	n.c.	Inp 23								
*) Die Statusmeldung mehrerer Schalter/LEDs erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.										
BDH 1401			Ziel-/Quelladresse	Type	Funktion	Status	Ansteuerung eines Relais*)	Bitmuster		
Relais 1	DO 1	Out 00	Moduladresse x 4 + 80	Word	16	1 = EIN	<b>h 01</b>	xxxx 0001		
Relais 2	DO 2	Out 01				1 = EIN	<b>h 02</b>	xxxx 0010		
Relais 3	DO 3	Out 02				Preset Multiple	1 = EIN	<b>h 04</b>	xxxx 0100	
Relais 4	DO 4	Out 03					1 = EIN	<b>h 08</b>	xxxx 1000	
Reserve	n.c.	Out 04		Registers						
Reserve	n.c.	Out 05								
Reserve	n.c.	Out 06								
Reserve	n.c.	Out 07								
*) Die Ansteuerung mehrerer Relais erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.										

BDH 2200			Ziel-/Quelladresse	Type	Funktion	Status	Meldung eines Schalters*)	Bitmuster	
Schalter 1	DI 1	Inp 00	Moduladresse x 4 + 80	Word	4	1 = Auto	<b>h 01</b>	xxxx xx01	
Schalter 2	DI 2	Inp 01				1 = Auto	<b>h 02</b>	xxxx xx10	
Reserve	n.c.	Inp 02				Read Input Registers			
Reserve	n.c.	Inp 03							
Reserve	n.c.	Inp 04							
Reserve	n.c.	Inp 05							
Reserve	n.c.	Inp 06							
Reserve	n.c.	Inp 07							
*) Die Statusmeldung mehrerer Schalter erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.									
BDH 2200			Ziel-/Quelladresse	Type	Funktion	Status	Setzen eines Parameters*)	Bitmuster	
LED 1 rot	DO 7	Out 08	Moduladresse x 4 + 80	Word	16	1 = Ein	<b>h 01 00 00</b>	xx00 000 <b>1</b>	
LED 1 grün	DO 8	Out 09				1 = Ein	<b>h 02 00 00</b>	xx00 0010	
LED 2 grün	DO 9	Out 10				Preset Multiple Registers	1 = Ein	<b>h 04 00 00</b>	xx00 0100
LED 3 grün	DO 10	Out 11					1 = Ein	<b>h 08 00 00</b>	xx00 1000
LED 4 rot	DO 11	Out 12					1 = Ein	<b>h 10 00 00</b>	xx <b>01</b> 0000
LED 4 grün	DO 12	Out 13					1 = Ein	<b>h 20 00 00</b>	xx10 0000
Reserve	n.c.	Out 14							
Reserve	n.c.	Out 15							
LED 5 grün	DO 13	Out 16	Moduladresse x 4 + 81	Word	[16]	1 = Ein	<b>h 00 01 00</b>	xxxx xx0 <b>1</b>	
LED 6 grün	DO 14	Out 17				1 = Ein	<b>h 00 02 00</b>	xxxx xx10	
Reserve	n.c.	Out 18							
Reserve	n.c.	Out 19							
Reserve	n.c.	Out 20							
Reserve	n.c.	Out 21							
Reserve	n.c.	Out 22							
Reserve	n.c.	Out 23							
Relais 1	DO 1	Out 00	Moduladresse x 4 + 82	Word	[16]	1 = Ein	<b>h 00 00 01</b>	xxxx 0001	
Relais 2	DO 2	Out 01				1 = Ein	<b>h 00 00 02</b>	xxxx 0010	
Relais 3	DO 3	Out 02				1 = Ein	<b>h 00 00 04</b>	xxxx 0100	
Relais 4	DO 4	Out 03				1 = Ein	<b>h 00 00 08</b>	xxxx 1000	
Reserve	n.c.	Out 04							
Reserve	n.c.	Out 05							
Reserve	n.c.	Out 06							
Reserve	n.c.	Out 07							
*) Die Ansteuerung mehrerer LEDs/Relais erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.									



BDH 2201			Ziel-/Quelladresse	Type	Funktion	Status	Meldung eines Parameters*)	Bitmuster	
Schalter 1	DI 1	Inp 00	Moduladresse x 4 + 80	Word	4	1 = Auto	<b>h 01 00 00</b>	xxxx xx01	
Schalter 2	DI 2	Inp 01				1 = Auto	<b>h 02 00 00</b>	xxxx xx10	
Reserve	n.c.	Inp 02				Read Input Registers			
Reserve	n.c.	Inp 03							
Reserve	n.c.	Inp 04							
Reserve	n.c.	Inp 05							
Reserve	n.c.	Inp 06							
Reserve	n.c.	Inp 07							
LED 1 rot	DI 7	Inp 08	Moduladresse x 4 + 81	Word	[4]	1 = Ein	<b>h 00 01 00</b>	xx00 000 <b>1</b>	
LED 2 grün	DI 8	Inp 09				1 = Ein	<b>h 00 02 00</b>	xx00 0010	
LED 3 grün	DI 9	Inp 10				1 = Ein	<b>h 00 04 00</b>	xx00 0100	
LED 4 rot	DI 10	Inp 11				1 = Ein	<b>h 00 08 00</b>	xx00 <b>1</b> 000	
LED 5 grün	DI 11	Inp 12				1 = Ein	<b>h 00 10 00</b>	xx01 0000	
LED 6 grün	DI 12	Inp 13				1 = Ein	<b>h 00 20 00</b>	xx10 0000	
Reserve	n.c.	Inp 14							
Reserve	n.c.	Inp 15							
Reserve	n.c.	Inp 16	Moduladresse x 4 + 82	Word	[4]		<b>h 00 00 XX**)</b>	xxxx xxxx	
Reserve	n.c.	Inp 17							
Reserve	n.c.	Inp 18							
Reserve	n.c.	Inp 19							
Reserve	n.c.	Inp 20							
Reserve	n.c.	Inp 21							
Reserve	n.c.	Inp 22							
Reserve	n.c.	Inp 23							
*) Die Statusmeldung mehrerer Schalter/LEDs erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.									
**) Datenbyte Nr. 3 muss mit ausgelesen werden!									
BDH 2201			Ziel-/Quelladresse	Type	Funktion	Status	Ansteuerung eines Relais*)	Bitmuster	
Relais 1	DO 1	Out 00	Moduladresse x 4 + 80	Word	16	1 = EIN	<b>h 01</b>	xxxx 0001	
Relais 2	DO 2	Out 01				1 = EIN	<b>h 02</b>	xxxx 0010	
Relais 3	DO 3	Out 02				Preset Multiple Registers	1 = EIN	<b>h 04</b>	xxxx 0100
Relais 4	DO 4	Out 03					1 = EIN	<b>h 08</b>	xxxx 1000
Reserve	n.c.	Out 04							
Reserve	n.c.	Out 05							
Reserve	n.c.	Out 06							
Reserve	n.c.	Out 07							
*) Die Ansteuerung mehrerer Relais erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.									

BDH 4800			Ziel-/Quelladresse	Type	Funktion	Status	Meldung eines Tasters*)	Bitmuster
Taster 1	DI 1	Inp 00	Moduladresse x 4 + 80	Word	4	1 = Auto (= nicht betätigt)	h 01	xxxx 0001
Taster 2	DI 2	Inp 01					h 02	xxxx 0010
Taster 3	DI 3	Inp 02					h 04	xxxx 0100
Taster 4	DI 4	Inp 03					h 08	xxxx 1000
Reserve	n.c.	Inp 04			Registers			
Reserve	n.c.	Inp 05						
Reserve	n.c.	Inp 06						
Reserve	n.c.	Inp 07						
*) Die Statusmeldung mehrerer Taster erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.								
BDH 4800			Ziel-/Quelladresse	Type	Funktion	Status	Setzen eines Parameters*)	Bitmuster
LED 1 orange	DO 7	Out 08	Moduladresse x 4 + 80	Word	16	1 = Ein	h 02 00 00	xx00 0010
LED 2 rot-blink.	DO 8	Out 09					h 01 00 00	xx00 0001
LED 3 rot	DO 9	Out 10					h 04 00 00	xx00 0100
LED 4 orange	DO 10	Out 11					h 08 00 00	xx00 1000
LED 5 rot-blink.	DO 11	Out 12			Registers	1 = Ein	h 10 00 00	xx01 0000
LED 6 rot	DO 12	Out 13				1 = Ein	h 20 00 00	xx10 0000
Reserve	n.c.	Out 14						
Reserve	n.c.	Out 15						
LED 7 orange	DO 13	Out 16	Moduladresse x 4 + 81	Word	[16]	1 = Ein	h 00 01 00	xx00 0001
LED 8 rot-blink.	DO 14	Out 17					h 00 02 00	xx00 0010
LED 9 rot	DO 15	Out 18					h 00 04 00	xx00 0100
LED 10 orange	DO 16	Out 19					h 00 10 00	xx01 0000
LED 11 rot-blink.	DO 17	Out 20				1 = Ein	h 00 08 00	xx00 1000
LED 12 rot	DO 18	Out 21				1 = Ein	h 00 20 00	xx10 0000
Reserve	n.c.	Out 22						
Reserve	n.c.	Out 23						
Relais 1	DO 1	Out 00	Moduladresse x 4 + 82	Word	[16]	1 = Ein	h 00 00 01	xxxx 0001
Relais 2	DO 2	Out 01					h 00 00 02	xxxx 0010
Relais 3	DO 3	Out 02					h 00 00 04	xxxx 0100
Relais 4	DO 4	Out 03					h 00 00 08	xxxx 1000
Reserve	n.c.	Out 04						
Reserve	n.c.	Out 05						
Reserve	n.c.	Out 06						
Reserve	n.c.	Out 07						
*) Die Ansteuerung mehrerer LEDs/Relais erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.								

BAH 4000			Ziel-/Quelladresse	Type	Funktion	Status	Meldung eines Schalters*)	Bitmuster	
<b>Digitaleingänge</b>									
Input 1	DI 1	Inp 00	Moduladresse + 112	Word	<b>4</b>	1 = Auto	<b>h 01</b>	xxxx 0001	
Input 2	DI 2	Inp 01				1 = Auto	<b>h 02</b>	xxxx 0010	
Input 3	DI 3	Inp 02			<b>Read</b>	1 = Auto	<b>h 04</b>	xxxx 0100	
Input 4	DI 4	Inp 03			<b>Input</b>	1 = Auto	<b>h 08</b>	xxxx 1000	
Reserve	n.c.	Inp 04			<b>Registers</b>				
Reserve	n.c.	Inp 05							
Reserve	n.c.	Inp 06							
Reserve	n.c.	Inp 07							
			*) Die Statusmeldung mehrerer Schalter erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.						
<b>BAH 4000</b>									
<b>Analogeingänge</b>									
			Ziel-/Quelladresse	Type	Funktion	Status			
Input 1			Moduladresse x 4 + 0	Word	<b>4</b>	0 ... 255	(Low-Byte, High-Byte = 00)		
Input 2			Moduladresse x 4 + 1		<b>Read</b>	0 ... 255	(Low-Byte, High-Byte = 00)		
Input 3			Moduladresse x 4 + 2		<b>Input</b>	0 ... 255	(Low-Byte, High-Byte = 00)		
Input 4			Moduladresse x 4 + 3		<b>Registers</b>	0 ... 255	(Low-Byte, High-Byte = 00)		
<b>BAH 4000</b>									
<b>Analogausgänge</b>									
			Ziel-/Quelladresse	Type	Funktion	Status			
Output 1			Moduladresse x 4 + 0	Word	<b>16</b>	0 ... 255	(Low-Byte, High-Byte = 00)		
Output 2			Moduladresse x 4 + 1		<b>Preset</b>	0 ... 255	(Low-Byte, High-Byte = 00)		
Output 3			Moduladresse x 4 + 2		<b>Multiple</b>	0 ... 255	(Low-Byte, High-Byte = 00)		
Output 4			Moduladresse x 4 + 3		<b>Registers</b>	0 ... 255	(Low-Byte, High-Byte = 00)		

BZK 1000			Ziel-/Quelladresse	Type	Funktion	Status	Meldung eines Tasters*)	Bitmuster
<b>Digitaleingänge</b>								
<b>Entsperren</b>	DI 1	Inp 00	R 70	Word	<b>4</b>	1 = betät.	<b>h 01</b>	xxxx x001
<b>Hupe quittieren</b>	DI 2	Inp 01			<b>Read</b>	1 = betät.	<b>h 02</b>	xxxx x010
<b>Lampentest</b>	DI 3	Inp 02			<b>Input Registers</b>	1 = betät.	<b>h 04</b>	xxxx x100
*) Die Statusmeldung mehrerer Taster erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.								
BZK 1000			Ziel-/Quelladresse	Type	Funktion	Status	Ansteuerung eines Relais*)	Bitmuster
<b>Digitalausgänge</b>								
<b>Reset</b>	DO 1	Out 00	R 70	Word	<b>16</b>	1 = Ein	<b>h 01</b>	xxxx 0001
<b>Sammelalarm</b>	DO 2	Out 01			<b>Preset</b>	1 = Ein	<b>h 02</b>	xxxx 0010
<b>Hupenkontakt</b>	DO 3	Out 02			<b>Multiple</b>	1 = Ein	<b>h 04</b>	xxxx 0100
<b>Lampentest</b>	DO 4	Out 03			<b>Registers</b>	1 = Ein	<b>h 08</b>	xxxx 1000
*) Die Ansteuerung mehrerer Relais erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.								
<b>Allgemeine</b>			<b>Defaultwerte</b>					
<b>Konfiguration</b>	<b>Wertbereich</b>	<b>Zieladresse</b>	<b>Type</b>	<b>Funktion</b>	<b>dezimal</b>	<b>hexadezimal</b>		
<b>Stationsnummer</b>	0..249	32: <Station> 33: NOT <Station>	Word	<b>16</b>	<b>d 160</b>	h 00 A0 h FF 5F		
<b>Baudrate</b>	6 = 9600 Baud 7 = 19200 Baud 8 = 38400 Baud	34: <Baud> 35: NOT <Baud>	Word	[16]	<b>d 6</b>	h 00 06 h FF F9		
<b>Modus</b>	7=Modbus RTU	36: <Modus> 37: NOT <Modus>	Word	[16]	<b>d 7</b>	h 00 07 h FF F8		
<b>Schnittstelle</b>	4 = RS485	38: <Uart> 39: NOT <Uart>	Word	[16]	<b>d 4</b>	h 00 04 h FF FB		
<b>Wichtig: Es müssen alle vier Parameter (R32...R39) zusammen mit einem einzigen Befehl geschrieben werden, auch wenn nur ein einziger Wert geändert werden soll!</b>								
<b>Busaktivitäts-Überwachung</b>	Zeit in Sek.	42: <Zeit> 43: NOT <Zeit>	Word	<b>16</b>	<b>d 10</b>	h 00 0A h FF F5		

**MODBus Crossliste,**  
**Adressierung modulübergreifend**  
**und Einzeldatenpunkt-Abfrage**  
**mittels Bit-Funktionen**

BLM 1000			Zieladresse	Type	Modbus-Funktion	Status	Ansteuerung einer LED *)	Bitmuster
LED 1 grün	DO 1	Out 00	Index [i] x 32 + 1000	Coil	<b>15</b>	1 = Ein	<b>01</b>	xx00 0001
LED 1 rot	DO 2	Out 01	Index [i] x 32 + 1001	Coil		1 = Ein	<b>02</b>	xx00 0010
LED 2 grün	DO 3	Out 02	Index [i] x 32 + 1002	Coil	<b>Force</b>	1 = Ein	<b>04</b>	xx00 0100
LED 2 rot	DO 4	Out 03	Index [i] x 32 + 1003	Coil	<b>Multiple</b>	1 = Ein	<b>08</b>	xx00 1000
LED 3 grün	DO 5	Out 04	Index [i] x 32 + 1004	Coil	<b>Coils</b>	1 = Ein	<b>10</b>	xx01 0000
LED 3 rot	DO 6	Out 05	Index [i] x 32 + 1005	Coil		1 = Ein	<b>20</b>	xx10 0000
Reserve	n.c.	Out 06	Index [i] x 32 + 1006	Coil				
Reserve	n.c.	Out 07	Index [i] x 32 + 1007	Coil				
LED 4 grün	DO 7	Out 08	Index [i] x 32 + 1008	Coil	[15]	1 = Ein	<b>01</b>	xx00 0001
LED 4 rot	DO 8	Out 09	Index [i] x 32 + 1009	Coil		1 = Ein	<b>02</b>	xx00 0010
LED 5 grün	DO 9	Out 10	Index [i] x 32 + 1010	Coil		1 = Ein	<b>04</b>	xx00 0100
LED 5 rot	DO 10	Out 11	Index [i] x 32 + 1011	Coil		1 = Ein	<b>08</b>	xx00 1000
LED 6 grün	DO 11	Out 12	Index [i] x 32 + 1012	Coil		1 = Ein	<b>10</b>	xx01 0000
LED 6 rot	DO 12	Out 13	Index [i] x 32 + 1013	Coil		1 = Ein	<b>20</b>	xx10 0000
Reserve	n.c.	Out 14	Index [i] x 32 + 1014	Coil				
Reserve	n.c.	Out 15	Index [i] x 32 + 1015	Coil				
LED 7 grün	DO 13	Out 16	Index [i] x 32 + 1016	Coil	[15]	1 = Ein	<b>01</b>	xx00 0001
LED 7 rot	DO 14	Out 17	Index [i] x 32 + 1017	Coil		1 = Ein	<b>02</b>	xx00 0010
LED 8 grün	DO 15	Out 18	Index [i] x 32 + 1018	Coil		1 = Ein	<b>04</b>	xx00 0100
LED 8 rot	DO 16	Out 19	Index [i] x 32 + 1019	Coil		1 = Ein	<b>08</b>	xx00 1000
LED 9 grün	DO 17	Out 20	Index [i] x 32 + 1020	Coil		1 = Ein	<b>10</b>	xx01 0000
LED 9 rot	DO 18	Out 21	Index [i] x 32 + 1021	Coil		1 = Ein	<b>20</b>	xx10 0000
Reserve	n.c.	Out 22	Index [i] x 32 + 1022	Coil				
Reserve	n.c.	Out 23	Index [i] x 32 + 1023	Coil				
LED 10 grün	DO 19	Out 24	Index [i] x 32 + 1024	Coil	[15]	1 = Ein	<b>01</b>	xxxx xx01
LED 10 rot	DO 20	Out 25	Index [i] x 32 + 1025	Coil		1 = Ein	<b>02</b>	xxxx xx10
Reserve	n.c.	Out 26	Index [i] x 32 + 1026	Coil				
Reserve	n.c.	Out 27	Index [i] x 32 + 1027	Coil				
Reserve	n.c.	Out 28	Index [i] x 32 + 1028	Coil				
Reserve	n.c.	Out 29	Index [i] x 32 + 1029	Coil				
Reserve	n.c.	Out 30	Index [i] x 32 + 1030	Coil				
Reserve	n.c.	Out 31	Index [i] x 32 + 1031	Coil				
*) Die Ansteuerung mehrerer LEDs erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.								

**MODBus Crossliste,**  
**Adressierung modulübergreifend**  
**und Einzeldatenpunkt-Abfrage**  
**mittels Bit-Funktionen**

BLM 1001			Quelladresse	Type	Modbus-Funktion	Status	Meldung einer LED *)	Bitmuster
LED 1 grün	DI 1	Inp 00	Index [i] x 32 + 1000	Input	2	0 = EIN	3E	xx11 1110
LED 1 rot	DI 2	Inp 01	Index [i] x 32 + 1001	Input		0 = EIN	3D	xx11 1101
LED 2 grün	DI 3	Inp 02	Index [i] x 32 + 1002	Input	Read	0 = EIN	3B	xx11 1011
LED 2 rot	DI 4	Inp 03	Index [i] x 32 + 1003	Input	Input	0 = EIN	37	xx11 0111
LED 3 grün	DI 5	Inp 04	Index [i] x 32 + 1004	Input	Status	0 = EIN	2F	xx10 1111
LED 3 rot	DI 6	Inp 05	Index [i] x 32 + 1005	Input		0 = EIN	1F	xx01 1111
Reserve	n.c.	Inp 06	Index [i] x 32 + 1006	Input				
Reserve	n.c.	Inp 07	Index [i] x 32 + 1007	Input				
LED 4 grün	DI 7	Inp 08	Index [i] x 32 + 1008	Input	[2]	0 = EIN	3E	xx11 1110
LED 4 rot	DI 8	Inp 09	Index [i] x 32 + 1009	Input		0 = EIN	3D	xx11 1101
LED 5 grün	DI 9	Inp 10	Index [i] x 32 + 1010	Input		0 = EIN	3B	xx11 1011
LED 5 rot	DI 10	Inp 11	Index [i] x 32 + 1011	Input		0 = EIN	37	xx11 0111
LED 6 grün	DI 11	Inp 12	Index [i] x 32 + 1012	Input		0 = EIN	2F	xx10 1111
LED 6 rot	DI 12	Inp 13	Index [i] x 32 + 1013	Input		0 = EIN	1F	xx01 1111
Reserve	n.c.	Inp 14	Index [i] x 32 + 1014	Input				
Reserve	n.c.	Inp 15	Index [i] x 32 + 1015	Input				
LED 7 grün	DI 13	Inp 16	Index [i] x 32 + 1016	Input	[2]	0 = EIN	3E	xx11 1110
LED 7 rot	DI 14	Inp 17	Index [i] x 32 + 1017	Input		0 = EIN	3D	xx11 1101
LED 8 grün	DI 15	Inp 18	Index [i] x 32 + 1018	Input		0 = EIN	3B	xx11 1011
LED 8 rot	DI 16	Inp 19	Index [i] x 32 + 1019	Input		0 = EIN	37	xx11 0111
LED 9 grün	DI 17	Inp 20	Index [i] x 32 + 1020	Input		0 = EIN	2F	xx10 1111
LED 9 rot	DI 18	Inp 21	Index [i] x 32 + 1021	Input		0 = EIN	1F	xx01 1111
Reserve	n.c.	Inp 22	Index [i] x 32 + 1022	Input				
Reserve	n.c.	Inp 23	Index [i] x 32 + 1023	Input				
LED 10 grün	DI 19	Inp 24	Index [i] x 32 + 1024	Input	[2]	0 = EIN	0E	xxxx xx10
LED 10 rot	DI 20	Inp 25	Index [i] x 32 + 1025	Input		0 = EIN	0D	xxxx xx01
Reserve	n.c.	Inp 26	Index [i] x 32 + 1026	Input				
Reserve	n.c.	Inp 27	Index [i] x 32 + 1027	Input				
Reserve	n.c.	Inp 28	Index [i] x 32 + 1028	Input				
Reserve	n.c.	Inp 29	Index [i] x 32 + 1029	Input				
Reserve	n.c.	Inp 30	Index [i] x 32 + 1030	Input				
Reserve	n.c.	Inp 31	Index [i] x 32 + 1031	Input				
*) Die Statusmeldung mehrerer LEDs erfolgt durch Überlagerung der relevanten Bits durch binäres UND.								

MODBus Crossliste,  
Adressierung *modulübergreifend*  
und *Einzeldatenpunkt-Abfrage*  
mittels **Bit-Funktionen**

BDH 1400							Meldung eines	Bitmuster
			Ziel-/Quelladresse	Type	Funktion	Status	Schalters*)	
Schalter 1	DI 1	Inp 00	Index [i] x 32 + 1024	Input	2	1 = Hand	01	xxxx 0001
Schalter 2	DI 2	Inp 01	Index [i] x 32 + 1025	Input		1 = Hand	02	xxxx 0010
Schalter 3	DI 3	Inp 02	Index [i] x 32 + 1026	Input	Read	1 = Hand	04	xxxx 0100
Schalter 4	DI 4	Inp 03	Index [i] x 32 + 1027	Input	Input	1 = Hand	08	xxxx 1000
Reserve	n.c.	Inp 04	Index [i] x 32 + 1028	Input	Status			
Reserve	n.c.	Inp 05	Index [i] x 32 + 1029	Input				
Reserve	n.c.	Inp 06	Index [i] x 32 + 1030	Input				
Reserve	n.c.	Inp 07	Index [i] x 32 + 1031	Input				
*) Die Statusmeldung mehrerer Schalter erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.								
BDH 1400							Setzen eines	Bitmuster
			Ziel-/Quelladresse	Type	Funktion	Status	Parameters*)	
LED 1 rot	DO 7	Out 08	Index [i] x 32 + 1000	Coil	15	1 = Ein	01	xx00 0001
LED 1 grün	DO 8	Out 09	Index [i] x 32 + 1001	Coil		1 = Ein	02	xx00 0010
LED 2 grün	DO 9	Out 10	Index [i] x 32 + 1002	Coil	Force	1 = Ein	04	xx00 0100
LED 3 rot	DO 10	Out 11	Index [i] x 32 + 1003	Coil	Multiple	1 = Ein	08	xx00 1000
LED 3 grün	DO 11	Out 12	Index [i] x 32 + 1004	Coil	Coils	1 = Ein	10	xx01 0000
LED 4 grün	DO 12	Out 13	Index [i] x 32 + 1005	Coil		1 = Ein	20	xx10 0000
Reserve	n.c.	Out 14	Index [i] x 32 + 1006	Coil				
Reserve	n.c.	Out 15	Index [i] x 32 + 1007	Coil				
LED 5 rot	DO 13	Out 16	Index [i] x 32 + 1008	Coil	[15]	1 = Ein	01	xx00 0001
LED 5 grün	DO 14	Out 17	Index [i] x 32 + 1009	Coil		1 = Ein	02	xx00 0010
LED 6 grün	DO 15	Out 18	Index [i] x 32 + 1010	Coil		1 = Ein	04	xx00 0100
LED 7 rot	DO 16	Out 19	Index [i] x 32 + 1011	Coil		1 = Ein	08	xx00 1000
LED 7 grün	DO 17	Out 20	Index [i] x 32 + 1012	Coil		1 = Ein	10	xx01 0000
LED 8 grün	DO 18	Out 21	Index [i] x 32 + 1013	Coil		1 = Ein	20	xx10 0000
Reserve	n.c.	Out 22	Index [i] x 32 + 1014	Coil				
Reserve	n.c.	Out 23	Index [i] x 32 + 1015	Coil				
Relais 1	DO 1	Out 00	Index [i] x 32 + 1016	Coil	[15]	1 = Ein	01	xxxx 0001
Relais 2	DO 2	Out 01	Index [i] x 32 + 1017	Coil		1 = Ein	02	xxxx 0010
Relais 3	DO 3	Out 02	Index [i] x 32 + 1018	Coil		1 = Ein	04	xxxx 0100
Relais 4	DO 4	Out 03	Index [i] x 32 + 1019	Coil		1 = Ein	08	xxxx 1000
Reserve	n.c.	Out 04	Index [i] x 32 + 1020	Coil				
Reserve	n.c.	Out 05	Index [i] x 32 + 1021	Coil				
Reserve	n.c.	Out 06	Index [i] x 32 + 1022	Coil				
Reserve	n.c.	Out 07	Index [i] x 32 + 1023	Coil				
*) Die Ansteuerung mehrerer LEDs/Relais erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.								

**MODBus Crossliste,**  
**Adressierung modulübergreifend**  
**und Einzeldatenpunkt-Abfrage**  
**mittels Bit-Funktionen**

BDH 1401			Ziel-/Quelladresse	Type	Funktion	Status	Meldung eines Parameters*)	Bitmuster
Schalter 1	DI 1	Inp 00	Index [i] x 32 + 1008	Input	<b>2</b>	1 = Hand	<b>01</b>	xxxx 0001
Schalter 2	DI 2	Inp 01	Index [i] x 32 + 1009	Input		1 = Hand	<b>02</b>	xxxx 0010
Schalter 3	DI 3	Inp 02	Index [i] x 32 + 1010	Input	<b>Read</b>	1 = Hand	<b>04</b>	xxxx 0100
Schalter 4	DI 4	Inp 03	Index [i] x 32 + 1011	Input	<b>Input</b>	1 = Hand	<b>08</b>	xxxx 1000
Reserve	n.c.	Inp 04	Index [i] x 32 + 1012	Input	<b>Status</b>			
Reserve	n.c.	Inp 05	Index [i] x 32 + 1013	Input				
Reserve	n.c.	Inp 06	Index [i] x 32 + 1014	Input				
Reserve	n.c.	Inp 07	Index [i] x 32 + 1015	Input				
LED 1 rot	DI 7	Inp 08	Index [i] x 32 + 1016	Input	[2]	0 = EIN	<b>3E</b>	xx11 1110
LED 2 grün	DI 8	Inp 09	Index [i] x 32 + 1017	Input		0 = EIN	<b>3D</b>	xx11 1101
LED 3 rot	DI 9	Inp 10	Index [i] x 32 + 1018	Input		0 = EIN	<b>3B</b>	xx11 1011
LED 4 grün	DI 10	Inp 11	Index [i] x 32 + 1019	Input		0 = EIN	<b>37</b>	xx11 0111
LED 5 rot	DI 11	Inp 12	Index [i] x 32 + 1020	Input		0 = EIN	<b>2F</b>	xx10 1111
LED 6 grün	DI 12	Inp 13	Index [i] x 32 + 1021	Input		0 = EIN	<b>1F</b>	xx01 1111
Reserve	n.c.	Inp 14	Index [i] x 32 + 1022	Input				
Reserve	n.c.	Inp 15	Index [i] x 32 + 1023	Input				
LED 7 rot	DI 13	Inp 16	Index [i] x 32 + 1024	Input	[2]	0 = EIN	<b>0E</b>	xxxx xx10
LED 8 grün	DI 14	Inp 17	Index [i] x 32 + 1025	Input		0 = EIN	<b>0D</b>	xxxx xx01
Reserve	n.c.	Inp 18	Index [i] x 32 + 1026	Input				
Reserve	n.c.	Inp 19	Index [i] x 32 + 1027	Input				
Reserve	n.c.	Inp 20	Index [i] x 32 + 1028	Input				
Reserve	n.c.	Inp 21	Index [i] x 32 + 1029	Input				
Reserve	n.c.	Inp 22	Index [i] x 32 + 1030	Input				
Reserve	n.c.	Inp 23	Index [i] x 32 + 1031	Input				
*) Die Statusmeldung mehrerer Schalter/LEDs erfolgt durch Überlagerung der relevanten Bits durch binäres UND.								
BDH 1401			Ziel-/Quelladresse	Type	Funktion	Status	Ansteuerung eines Relais*)	Bitmuster
Relais 1	DO 1	Out 00	Index [i] x 32 + 1000	Coil	<b>15</b>	1 = EIN	<b>01</b>	xxxx 0001
Relais 2	DO 2	Out 01	Index [i] x 32 + 1001	Coil		1 = EIN	<b>02</b>	xxxx 0010
Relais 3	DO 3	Out 02	Index [i] x 32 + 1002	Coil	<b>Force</b>	1 = EIN	<b>04</b>	xxxx 0100
Relais 4	DO 4	Out 03	Index [i] x 32 + 1003	Coil	<b>Multiple</b>	1 = EIN	<b>08</b>	xxxx 1000
Reserve	n.c.	Out 04	Index [i] x 32 + 1004	Coil	<b>Coils</b>			
Reserve	n.c.	Out 05	Index [i] x 32 + 1005	Coil				
Reserve	n.c.	Out 06	Index [i] x 32 + 1006	Coil				
Reserve	n.c.	Out 07	Index [i] x 32 + 1007	Coil				
*) Die Ansteuerung mehrerer Relais erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.								



MODBus Crossliste,  
Adressierung *modulübergreifend*  
und *Einzeldatenpunkt-Abfrage*  
mittels **Bit-Funktionen**

BDH 2200								Meldung eines	Bitmuster
			Ziel-/Quelladresse	Type	Funktion	Status	Schalters*)		
Schalter 1	DI 1	Inp 00	Index [i] x 32 + 1024	Input	2	1 = Hand	01	xxxx xx01	
Schalter 2	DI 2	Inp 01	Index [i] x 32 + 1025	Input		1 = Hand	02	xxxx xx10	
Reserve	n.c.	Inp 02	Index [i] x 32 + 1026	Input	Read				
Reserve	n.c.	Inp 03	Index [i] x 32 + 1027	Input	Input				
Reserve	n.c.	Inp 04	Index [i] x 32 + 1028	Input	Status				
Reserve	n.c.	Inp 05	Index [i] x 32 + 1029	Input					
Reserve	n.c.	Inp 06	Index [i] x 32 + 1030	Input					
Reserve	n.c.	Inp 07	Index [i] x 32 + 1031	Input					
*) Die Statusmeldung mehrerer Schalter erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.									
BDH 2200								Setzen eines	Bitmuster
			Ziel-/Quelladresse	Type	Funktion	Status	Parameters*)		
LED 1 rot	DO 7	Out 08	Index [i] x 32 + 1000	Coil	15	1 = Ein	01	xx00 0001	
LED 1 grün	DO 8	Out 09	Index [i] x 32 + 1001	Coil		1 = Ein	02	xx00 0010	
LED 2 grün	DO 9	Out 10	Index [i] x 32 + 1002	Coil	Force	1 = Ein	04	xx00 0100	
LED 3 grün	DO 10	Out 11	Index [i] x 32 + 1003	Coil	Multiple	1 = Ein	08	xx00 1000	
LED 4 rot	DO 11	Out 12	Index [i] x 32 + 1004	Coil	Coils	1 = Ein	10	xx01 0000	
LED 4 grün	DO 12	Out 13	Index [i] x 32 + 1005	Coil		1 = Ein	20	xx10 0000	
Reserve	n.c.	Out 14	Index [i] x 32 + 1006	Coil					
Reserve	n.c.	Out 15	Index [i] x 32 + 1007	Coil					
LED 5 grün	DO 13	Out 16	Index [i] x 32 + 1008	Coil	[15]	1 = Ein	01	xxxx xx01	
LED 6 grün	DO 14	Out 17	Index [i] x 32 + 1009	Coil		1 = Ein	02	xxxx xx10	
Reserve	n.c.	Out 18	Index [i] x 32 + 1010	Coil					
Reserve	n.c.	Out 19	Index [i] x 32 + 1011	Coil					
Reserve	n.c.	Out 20	Index [i] x 32 + 1012	Coil					
Reserve	n.c.	Out 21	Index [i] x 32 + 1013	Coil					
Reserve	n.c.	Out 22	Index [i] x 32 + 1014	Coil					
Reserve	n.c.	Out 23	Index [i] x 32 + 1015	Coil					
Relais 1	DO 1	Out 00	Index [i] x 32 + 1016	Coil	[15]	1 = Ein	01	xxxx 0001	
Relais 2	DO 2	Out 01	Index [i] x 32 + 1017	Coil		1 = Ein	02	xxxx 0010	
Relais 3	DO 3	Out 02	Index [i] x 32 + 1018	Coil		1 = Ein	04	xxxx 0100	
Relais 4	DO 4	Out 03	Index [i] x 32 + 1019	Coil		1 = Ein	08	xxxx 1000	
Reserve	n.c.	Out 04	Index [i] x 32 + 1020	Coil					
Reserve	n.c.	Out 05	Index [i] x 32 + 1021	Coil					
Reserve	n.c.	Out 06	Index [i] x 32 + 1022	Coil					
Reserve	n.c.	Out 07	Index [i] x 32 + 1023	Coil					
*) Die Ansteuerung mehrerer LEDs/Relais erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.									

**MODBus Crossliste,**  
**Adressierung modulübergreifend**  
**und Einzeldatenpunkt-Abfrage**  
**mittels Bit-Funktionen**

BDH 2201								Meldung eines	Bitmuster
			Ziel-/Quelladresse	Type	Funktion	Status	Parameters*)		
Schalter 1	DI 1	Inp 00	Index [i] x 32 + 1008	Input	<b>2</b>	1 = Hand	<b>01</b>	xxxx xx01	
Schalter 2	DI 2	Inp 01	Index [i] x 32 + 1009	Input		1 = Hand	<b>02</b>	xxxx xx10	
Reserve	n.c.	Inp 02	Index [i] x 32 + 1010	Input	<b>Read</b>				
Reserve	n.c.	Inp 03	Index [i] x 32 + 1011	Input	<b>Input</b>				
Reserve	n.c.	Inp 04	Index [i] x 32 + 1012	Input	<b>Status</b>				
Reserve	n.c.	Inp 05	Index [i] x 32 + 1013	Input					
Reserve	n.c.	Inp 06	Index [i] x 32 + 1014	Input					
Reserve	n.c.	Inp 07	Index [i] x 32 + 1015	Input					
LED 1 rot	DI 7	Inp 08	Index [i] x 32 + 1016	Input	[2]	0 = EIN	<b>3E</b>	xx11 1110	
LED 2 grün	DI 8	Inp 09	Index [i] x 32 + 1017	Input		0 = EIN	<b>3D</b>	xx11 1101	
LED 3 grün	DI 9	Inp 10	Index [i] x 32 + 1018	Input		0 = EIN	<b>3B</b>	xx11 1011	
LED 4 rot	DI 10	Inp 11	Index [i] x 32 + 1019	Input		0 = EIN	<b>37</b>	xx11 0111	
LED 5 grün	DI 11	Inp 12	Index [i] x 32 + 1020	Input		0 = EIN	<b>2F</b>	xx10 1111	
LED 6 grün	DI 12	Inp 13	Index [i] x 32 + 1021	Input		0 = EIN	<b>1F</b>	xx01 1111	
Reserve	n.c.	Inp 14	Index [i] x 32 + 1022	Input					
Reserve	n.c.	Inp 15	Index [i] x 32 + 1023	Input					
Reserve	n.c.	Inp 16	Index [i] x 32 + 1024	Input	[2]		<b>XX</b>	xxxx xxxx	
Reserve	n.c.	Inp 17	Index [i] x 32 + 1025	Input					
Reserve	n.c.	Inp 18	Index [i] x 32 + 1026	Input					
Reserve	n.c.	Inp 19	Index [i] x 32 + 1027	Input					
Reserve	n.c.	Inp 20	Index [i] x 32 + 1028	Input					
Reserve	n.c.	Inp 21	Index [i] x 32 + 1029	Input					
Reserve	n.c.	Inp 22	Index [i] x 32 + 1030	Input					
Reserve	n.c.	Inp 23	Index [i] x 32 + 1031	Input					
*) Die Statusmeldung mehrerer Schalter/LEDs erfolgt durch Überlagerung der relevanten Bits durch binäres UND.									
BDH 2201								Ansteuerung	Bitmuster
			Ziel-/Quelladresse	Type	Funktion	Status	eines Relais*)		
Relais 1	DO 1	Out 00	Index [i] x 32 + 1000	Coil	<b>15</b>	1 = EIN	<b>01</b>	xxxx 0001	
Relais 2	DO 2	Out 01	Index [i] x 32 + 1001	Coil		1 = EIN	<b>02</b>	xxxx 0010	
Relais 3	DO 3	Out 02	Index [i] x 32 + 1002	Coil	<b>Force</b>	1 = EIN	<b>04</b>	xxxx 0100	
Relais 4	DO 4	Out 03	Index [i] x 32 + 1003	Coil	<b>Multiple</b>	1 = EIN	<b>08</b>	xxxx 1000	
Reserve	n.c.	Out 04	Index [i] x 32 + 1004	Coil	<b>Coils</b>				
Reserve	n.c.	Out 05	Index [i] x 32 + 1005	Coil					
Reserve	n.c.	Out 06	Index [i] x 32 + 1006	Coil					
Reserve	n.c.	Out 07	Index [i] x 32 + 1007	Coil					
*) Die Ansteuerung mehrerer Relais erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.									

MODBus Crossliste,  
Adressierung *modulübergreifend*  
und *Einzeldatenpunkt-Abfrage*  
mittels Bit-Funktionen

BDH 4800			Ziel-/Quelladresse	Type	Funktion	Status	Meldung eines Tasters*)	Bitmuster
Taster 1	DI 1	Inp 00	Index [i] x 32 + 1024	Input	2	1 = betät.	01	xxxx 0001
Taster 2	DI 2	Inp 01	Index [i] x 32 + 1025	Input		1 = betät.	02	xxxx 0010
Taster 3	DI 3	Inp 02	Index [i] x 32 + 1026	Input	Read	1 = betät.	04	xxxx 0100
Taster 4	DI 4	Inp 03	Index [i] x 32 + 1027	Input	Input	1 = betät.	08	xxxx 1000
Reserve	n.c.	Inp 04	Index [i] x 32 + 1028	Input	Status			
Reserve	n.c.	Inp 05	Index [i] x 32 + 1029	Input				
Reserve	n.c.	Inp 06	Index [i] x 32 + 1030	Input				
Reserve	n.c.	Inp 07	Index [i] x 32 + 1031	Input				
*) Die Statusmeldung mehrerer Taster erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.								
BDH 4800			Ziel-/Quelladresse	Type	Funktion	Status	Setzen eines Parameters*)	Bitmuster
LED 1 orange	DO 7	Out 08	Index [i] x 32 + 1000	Coil	15	1 = Ein	02	xx00 0010
LED 2 rot-blink.	DO 8	Out 09	Index [i] x 32 + 1001	Coil		1 = Ein	01	xx00 0001
LED 3 rot	DO 9	Out 10	Index [i] x 32 + 1002	Coil	Force	1 = Ein	04	xx00 0100
LED 4 orange	DO 10	Out 11	Index [i] x 32 + 1003	Coil	Multiple	1 = Ein	08	xx00 1000
LED 5 rot-blink.	DO 11	Out 12	Index [i] x 32 + 1004	Coil	Coils	1 = Ein	10	xx01 0000
LED 6 rot	DO 12	Out 13	Index [i] x 32 + 1005	Coil		1 = Ein	20	xx10 0000
Reserve	n.c.	Out 14	Index [i] x 32 + 1006	Coil				
Reserve	n.c.	Out 15	Index [i] x 32 + 1007	Coil				
LED 7 orange	DO 13	Out 16	Index [i] x 32 + 1008	Coil	[15]	1 = Ein	01	xx00 0001
LED 8 rot-blink.	DO 14	Out 17	Index [i] x 32 + 1009	Coil		1 = Ein	02	xx00 0010
LED 9 rot	DO 15	Out 18	Index [i] x 32 + 1010	Coil		1 = Ein	04	xx00 0100
LED 10 orange	DO 16	Out 19	Index [i] x 32 + 1011	Coil		1 = Ein	10	xx01 0000
LED 11 rot-blink.	DO 17	Out 20	Index [i] x 32 + 1012	Coil		1 = Ein	08	xx00 1000
LED 12 rot	DO 18	Out 21	Index [i] x 32 + 1013	Coil		1 = Ein	20	xx10 0000
Reserve	n.c.	Out 22	Index [i] x 32 + 1014	Coil				
Reserve	n.c.	Out 23	Index [i] x 32 + 1015	Coil				
Relais 1	DO 1	Out 00	Index [i] x 32 + 1016	Coil	[15]	1 = Ein	01	xxxx 0001
Relais 2	DO 2	Out 01	Index [i] x 32 + 1017	Coil		1 = Ein	02	xxxx 0010
Relais 3	DO 3	Out 02	Index [i] x 32 + 1018	Coil		1 = Ein	04	xxxx 0100
Relais 4	DO 4	Out 03	Index [i] x 32 + 1019	Coil		1 = Ein	08	xxxx 1000
Reserve	n.c.	Out 04	Index [i] x 32 + 1020	Coil				
Reserve	n.c.	Out 05	Index [i] x 32 + 1021	Coil				
Reserve	n.c.	Out 06	Index [i] x 32 + 1022	Coil				
Reserve	n.c.	Out 07	Index [i] x 32 + 1023	Coil				
*) Die Ansteuerung mehrerer LEDs/Relais erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.								

**MODBus Crossliste,**  
**Adressierung modulübergreifend**  
**und Einzeldatenpunkt-Abfrage**  
**mittels Bit-Funktionen**

BAH 4000			Ziel-/Quelladresse	Type	Funktion	Status	Meldung eines Schalters*)	Bitmuster
<b>Digitaleingänge</b>								
Input 1	DI 1	Inp 00	Index [i] x 32 + 1000	Input	<b>2</b>	1 = Hand	<b>01</b>	xxxx 0001
Input 2	DI 2	Inp 01	Index [i] x 32 + 1001	Input		1 = Hand	<b>02</b>	xxxx 0010
Input 3	DI 3	Inp 02	Index [i] x 32 + 1002	Input	<b>Read</b>	1 = Hand	<b>04</b>	xxxx 0100
Input 4	DI 4	Inp 03	Index [i] x 32 + 1003	Input	<b>Input</b>	1 = Hand	<b>08</b>	xxxx 1000
Reserve	n.c.	Inp 04	Index [i] x 32 + 1004	Input	<b>Status</b>			
Reserve	n.c.	Inp 05	Index [i] x 32 + 1005	Input				
Reserve	n.c.	Inp 06	Index [i] x 32 + 1006	Input				
Reserve	n.c.	Inp 07	Index [i] x 32 + 1007	Input				
*) Die Statusmeldung mehrerer Schalter erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.								
BAH 4000			Ziel-/Quelladresse	Type	Funktion	Status		
<b>Analogeingänge</b>								
Input 1			Index [i] x 8 + 1064	Word	<b>4</b>	0 ... 255	(Low-Byte, High-Byte = 00)	
Input 2			Index [i] x 8 + 1065	Word	<b>Read</b>	0 ... 255	(Low-Byte, High-Byte = 00)	
Input 3			Index [i] x 8 + 1066	Word	<b>Input</b>	0 ... 255	(Low-Byte, High-Byte = 00)	
Input 4			Index [i] x 8 + 1067	Word	<b>Registers</b>	0 ... 255	(Low-Byte, High-Byte = 00)	
BAH 4000			Ziel-/Quelladresse	Type	Funktion	Status		
<b>Analogausgänge</b>								
Output 1			Index [i] x 8 + 1068	Word	<b>16</b>	0 ... 255	(Low-Byte, High-Byte = 00)	
Output 2			Index [i] x 8 + 1069	Word	<b>Preset</b>	0 ... 255	(Low-Byte, High-Byte = 00)	
Output 3			Index [i] x 8 + 1070	Word	<b>Multiple</b>	0 ... 255	(Low-Byte, High-Byte = 00)	
Output 4			Index [i] x 8 + 1071	Word	<b>Registers</b>	0 ... 255	(Low-Byte, High-Byte = 00)	

**MODBus Crossliste,**  
**Adressierung modulübergreifend**  
**und Einzeldatenpunkt-Abfrage**  
**mittels Bit-Funktionen**

BZK 1000			Ziel-/Quelladresse	Type	Funktion	Status	Meldung eines Tasters*)	Bitmuster
<b>Digitaleingänge</b>								
<b>Entsperren</b>	DI 1	Inp 00	320	Input	<b>2</b>	1 = betät.	<b>01</b>	xxxx x001
<b>Hupe quittieren</b>	DI 2	Inp 01	321	Input	<b>Read</b>	1 = betät.	<b>02</b>	xxxx x010
<b>Lampentest</b>	DI 3	Inp 02	322	Input	<b>Input Status</b>	1 = betät.	<b>04</b>	xxxx x100
*) Die Statusmeldung mehrerer Taster erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.								
BZK 1000			Ziel-/Quelladresse	Type	Funktion	Status	Ansteuerung eines Relais*)	Bitmuster
<b>Digitalausgänge</b>								
<b>Reset</b>	DO 1	Out 00	328	Coil	<b>15</b>	1 = Ein	<b>01</b>	xxxx 0001
<b>Sammelalarm</b>	DO 2	Out 01	329	Coil	<b>Force</b>	1 = Ein	<b>02</b>	xxxx 0010
<b>Hupenkontakt</b>	DO 3	Out 02	330	Coil	<b>Multiple</b>	1 = Ein	<b>04</b>	xxxx 0100
<b>Lampentest</b>	DO 4	Out 03	331	Coil	<b>Coils</b>	1 = Ein	<b>08</b>	xxxx 1000
*) Die Ansteuerung mehrerer Relais erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.								
BZK 1000			Ziel-/Quelladresse	Type	Funktion	Status	Ansteuerung des BAH Adr.x	Bitmuster
<b>Aktivierung</b>								
<b>Lampentest für BAH400x</b>	BAH Adr. 0		R 52	Word	<b>6</b>	1 = Ein	<b>h 01</b>	0000 0001
	...						...	...
	BAH Adr. 7					1 = Ein	<b>h 80</b>	1000 0000
	alle BAH Adr. 0..7					1 = Ein	<b>h FF</b>	1111 1111
<b>Allgemeine</b>							<b>Defaultwerte</b>	
<b>Konfiguration</b>	<b>Wertbereich</b>	<b>Zieladresse</b>		<b>Type</b>	<b>Funktion</b>	<b>dezimal</b>	<b>hexadezimal</b>	
<b>Stationsnummer</b>	0..249	32: <Station> 33: NOT <Station>		Word	<b>16</b>	<b>d 160</b>	h 00 A0 h FF 5F	
<b>Baudrate</b>	6 = 9600 Baud 7 = 19200 Baud 8 = 38400 Baud	34: <Baud> 35: NOT <Baud>		Word	[16]	<b>d 6</b>	h 00 06 h FF F9	
<b>Modus</b>	7=Modbus RTU	36: <Modus> 37: NOT <Modus>		Word	[16]	<b>d 7</b>	h 00 07 h FF F8	
<b>Schnittstelle</b>	4 = RS485	38: <Uart> 39: NOT <Uart>		Word	[16]	<b>d 4</b>	h 00 04 h FF FB	
<b>Wichtig: Es müssen alle vier Parameter (R32...R39) zusammen mit einem einzigen Befehl geschrieben werden, auch wenn nur ein einziger Wert geändert werden soll!</b>								
<b>Busaktivitäts-Überwachung</b>	Zeit in Sek.	42: <Zeit> 43: NOT <Zeit>		Word	<b>16</b>	<b>d 10</b>	h 00 0A h FF F5	

**MODBus Crossliste,**  
**Adressierung modulübergreifend**  
**und Einzeldatenpunkt-Abfrage**  
**mittels Word-Funktionen**

BLM 1000			Zieladresse (Register)	Type	Modbus-Funktion	Status	Ansteuerung einer LED *)	Bitmuster		
LED 1 grün	DO 1	Out 00	Index [i] x 4 + 1000	Word	16	1 = Ein	<b>h 01</b>	xx00 0001		
LED 1 rot	DO 2	Out 01				1 = Ein	<b>h 02</b>	xx00 0010		
LED 2 grün	DO 3	Out 02				<b>Preset Multiple Registers</b>	1 = Ein	<b>h 04</b>	xx00 0100	
LED 2 rot	DO 4	Out 03					1 = Ein	<b>h 08</b>	xx00 1000	
LED 3 grün	DO 5	Out 04					1 = Ein	<b>h 10</b>	xx01 0000	
LED 3 rot	DO 6	Out 05					1 = Ein	<b>h 20</b>	xx10 0000	
Reserve	n.c.	Out 06								
Reserve	n.c.	Out 07								
LED 4 grün	DO 7	Out 08	Index [i] x 4 + 1001	Word	[16]	1 = Ein	<b>h 01</b>	xx00 0001		
LED 4 rot	DO 8	Out 09				1 = Ein	<b>h 02</b>	xx00 0010		
LED 5 grün	DO 9	Out 10				1 = Ein	<b>h 04</b>	xx00 0100		
LED 5 rot	DO 10	Out 11				1 = Ein	<b>h 08</b>	xx00 1000		
LED 6 grün	DO 11	Out 12				1 = Ein	<b>h 10</b>	xx01 0000		
LED 6 rot	DO 12	Out 13				1 = Ein	<b>h 20</b>	xx10 0000		
Reserve	n.c.	Out 14								
Reserve	n.c.	Out 15								
LED 7 grün	DO 13	Out 16	Index [i] x 4 + 1002	Word	[16]	1 = Ein	<b>h 01</b>	xx00 0001		
LED 7 rot	DO 14	Out 17				1 = Ein	<b>h 02</b>	xx00 0010		
LED 8 grün	DO 15	Out 18				1 = Ein	<b>h 04</b>	xx00 0100		
LED 8 rot	DO 16	Out 19				1 = Ein	<b>h 08</b>	xx00 1000		
LED 9 grün	DO 17	Out 20				1 = Ein	<b>h 10</b>	xx01 0000		
LED 9 rot	DO 18	Out 21				1 = Ein	<b>h 20</b>	xx10 0000		
Reserve	n.c.	Out 22								
Reserve	n.c.	Out 23								
LED 10 grün	DO 19	Out 24	Index [i] x 4 + 1003	Word	[16]	1 = Ein	<b>h 01</b>	xxxx xx01		
LED 10 rot	DO 20	Out 25				1 = Ein	<b>h 02</b>	xxxx xx10		
Reserve	n.c.	Out 26								
Reserve	n.c.	Out 27								
Reserve	n.c.	Out 28								
Reserve	n.c.	Out 29								
Reserve	n.c.	Out 30								
Reserve	n.c.	Out 31								
*) Die Ansteuerung mehrerer LEDs erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.										

MODBus Crossliste,  
Adressierung modulübergreifend  
und Einzeldatenpunkt-Abfrage  
mittels Word-Funktionen

BLM 1001			Quelladresse (Register)	Type	Modbus-Funktion	Status	Meldung einer LED *)	Bitmuster
LED 1 grün	DI 1	Inp 00	Index [i] x 4 + 1000	Word	4	0 = EIN	h 3E	xx11 1110
LED 1 rot	DI 2	Inp 01				0 = EIN	h 3D	xx11 1101
LED 2 grün	DI 3	Inp 02			Read	0 = EIN	h 3B	xx11 1011
LED 2 rot	DI 4	Inp 03			Input	0 = EIN	h 37	xx11 0111
LED 3 grün	DI 5	Inp 04			Registers	0 = EIN	h 2F	xx10 1111
LED 3 rot	DI 6	Inp 05				0 = EIN	h 1F	xx01 1111
Reserve	n.c.	Inp 06						
Reserve	n.c.	Inp 07						
LED 4 grün	DI 7	Inp 08	Index [i] x 4 + 1001	Word	[4]	0 = EIN	h 3E	xx11 1110
LED 4 rot	DI 8	Inp 09				0 = EIN	h 3D	xx11 1101
LED 5 grün	DI 9	Inp 10				0 = EIN	h 3B	xx11 1011
LED 5 rot	DI 10	Inp 11				0 = EIN	h 37	xx11 0111
LED 6 grün	DI 11	Inp 12				0 = EIN	h 2F	xx10 1111
LED 6 rot	DI 12	Inp 13				0 = EIN	h 1F	xx01 1111
Reserve	n.c.	Inp 14						
Reserve	n.c.	Inp 15						
LED 7 grün	DI 13	Inp 16	Index [i] x 4 + 1002	Word	[4]	0 = EIN	h 3E	xx11 1110
LED 7 rot	DI 14	Inp 17				0 = EIN	h 3D	xx11 1101
LED 8 grün	DI 15	Inp 18				0 = EIN	h 3B	xx11 1011
LED 8 rot	DI 16	Inp 19				0 = EIN	h 37	xx11 0111
LED 9 grün	DI 17	Inp 20				0 = EIN	h 2F	xx10 1111
LED 9 rot	DI 18	Inp 21				0 = EIN	h 1F	xx01 1111
Reserve	n.c.	Inp 22						
Reserve	n.c.	Inp 23						
LED 10 grün	DI 19	Inp 24	Index [i] x 4 + 1003	Word	[4]	0 = EIN	h 3E	xxxx xx10
LED 10 rot	DI 20	Inp 25				0 = EIN	h 3D	xxxx xx01
Reserve	n.c.	Inp 26						
Reserve	n.c.	Inp 27						
Reserve	n.c.	Inp 28						
Reserve	n.c.	Inp 29						
Reserve	n.c.	Inp 30						
Reserve	n.c.	Inp 31						
*) Die Statusmeldung mehrerer LEDs erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.								

MODBus Crossliste,  
Adressierung modulübergreifend  
und Einzeldatenpunkt-Abfrage  
mittels Word-Funktionen

BDH 1400			Ziel-/Quelladresse	Type	Funktion	Status	Meldung eines Schalters*)	Bitmuster	
Schalter 1	DI 1	Inp 00	Index [i] x 4 + 1003	Word	4	1 = Hand	<b>h 01</b>	xxxx 0001	
Schalter 2	DI 2	Inp 01				1 = Hand	<b>h 02</b>	xxxx 0010	
Schalter 3	DI 3	Inp 02				<b>Read</b>	1 = Hand	<b>h 04</b>	xxxx 0100
Schalter 4	DI 4	Inp 03				<b>Input</b>	1 = Hand	<b>h 08</b>	xxxx 1000
Reserve	n.c.	Inp 04			<b>Registers</b>				
Reserve	n.c.	Inp 05							
Reserve	n.c.	Inp 06							
Reserve	n.c.	Inp 07							
*) Die Statusmeldung mehrerer Schalter erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.									
BDH 1400			Ziel-/Quelladresse	Type	Funktion	Status	Setzen eines Parameters*)	Bitmuster	
LED 1 rot	DO 7	Out 08	Index [i] x 4 + 1000	Word	16	1 = Ein	<b>h 01</b>	xx00 0001	
LED 1 grün	DO 8	Out 09				1 = Ein	<b>h 02</b>	xx00 0010	
LED 2 grün	DO 9	Out 10				<b>Preset</b>	1 = Ein	<b>h 04</b>	xx00 0100
LED 3 rot	DO 10	Out 11				<b>Multiple</b>	1 = Ein	<b>h 08</b>	xx00 1000
LED 3 grün	DO 11	Out 12		<b>Registers</b>	1 = Ein	<b>h 10</b>	xx01 0000		
LED 4 grün	DO 12	Out 13			1 = Ein	<b>h 20</b>	xx10 0000		
Reserve	n.c.	Out 14							
Reserve	n.c.	Out 15							
LED 5 rot	DO 13	Out 16	Index [i] x 4 + 1001	Word	[16]	1 = Ein	<b>h 01</b>	xx00 0001	
LED 5 grün	DO 14	Out 17				1 = Ein	<b>h 02</b>	xx00 0010	
LED 6 grün	DO 15	Out 18				1 = Ein	<b>h 04</b>	xx00 0100	
LED 7 rot	DO 16	Out 19				1 = Ein	<b>h 08</b>	xx00 1000	
LED 7 grün	DO 17	Out 20			1 = Ein	<b>h 10</b>	xx01 0000		
LED 8 grün	DO 18	Out 21			1 = Ein	<b>h 20</b>	xx10 0000		
Reserve	n.c.	Out 22							
Reserve	n.c.	Out 23							
Relais 1	DO 1	Out 00	Index [i] x 4 + 1002	Word	[16]	1 = Ein	<b>h 01</b>	xxxx 0001	
Relais 2	DO 2	Out 01				1 = Ein	<b>h 02</b>	xxxx 0010	
Relais 3	DO 3	Out 02				1 = Ein	<b>h 04</b>	xxxx 0100	
Relais 4	DO 4	Out 03				1 = Ein	<b>h 08</b>	xxxx 1000	
Reserve	n.c.	Out 04							
Reserve	n.c.	Out 05							
Reserve	n.c.	Out 06							
Reserve	n.c.	Out 07							
*) Die Ansteuerung mehrerer LEDs/Relais erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.									



**MODBus Crossliste,**  
**Adressierung modulübergreifend**  
**und Einzeldatenpunkt-Abfrage**  
**mittels Word-Funktionen**

BDH 1401			Ziel-/Quelladresse	Type	Funktion	Status	Meldung eines Parameters*)	Bitmuster		
Schalter 1	DI 1	Inp 00	Index [i] x 4 + 1001	Word	4	1 = Hand	<b>h 01</b>	xxxx 0001		
Schalter 2	DI 2	Inp 01				1 = Hand	<b>h 02</b>	xxxx 0010		
Schalter 3	DI 3	Inp 02				Read	1 = Hand	<b>h 04</b>	xxxx 0100	
Schalter 4	DI 4	Inp 03				Input	1 = Hand	<b>h 08</b>	xxxx 1000	
Reserve	n.c.	Inp 04		Registers						
Reserve	n.c.	Inp 05								
Reserve	n.c.	Inp 06								
Reserve	n.c.	Inp 07								
LED 1 rot	DI 7	Inp 08	Index [i] x 4 + 1002	Word	[4]	0 = EIN	<b>h 3E</b>	xx11 1110		
LED 2 grün	DI 8	Inp 09				0 = EIN	<b>h 3D</b>	xx11 1101		
LED 3 rot	DI 9	Inp 10				0 = EIN	<b>h 3B</b>	xx11 1011		
LED 4 grün	DI 10	Inp 11				0 = EIN	<b>h 37</b>	xx11 0111		
LED 5 rot	DI 11	Inp 12				0 = EIN	<b>h 2F</b>	xx10 1111		
LED 6 grün	DI 12	Inp 13				0 = EIN	<b>h 1F</b>	xx01 1111		
Reserve	n.c.	Inp 14								
Reserve	n.c.	Inp 15								
LED 7 rot	DI 13	Inp 16	Index [i] x 4 + 1003	Word	[4]	0 = EIN	<b>h 3E</b>	xxxx xx10		
LED 8 grün	DI 14	Inp 17				0 = EIN	<b>h 3D</b>	xxxx xx01		
Reserve	n.c.	Inp 18								
Reserve	n.c.	Inp 19								
Reserve	n.c.	Inp 20								
Reserve	n.c.	Inp 21								
Reserve	n.c.	Inp 22								
Reserve	n.c.	Inp 23								
*) Die Statusmeldung mehrerer Schalter/LEDs erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.										
BDH 1401			Ziel-/Quelladresse	Type	Funktion	Status	Ansteuerung eines Relais*)	Bitmuster		
Relais 1	DO 1	Out 00	Index [i] x 4 + 1000	Word	16	1 = EIN	<b>h 01</b>	xxxx 0001		
Relais 2	DO 2	Out 01				1 = EIN	<b>h 02</b>	xxxx 0010		
Relais 3	DO 3	Out 02				Preset	1 = EIN	<b>h 04</b>	xxxx 0100	
Relais 4	DO 4	Out 03				Multiple	1 = EIN	<b>h 08</b>	xxxx 1000	
Reserve	n.c.	Out 04		Registers						
Reserve	n.c.	Out 05								
Reserve	n.c.	Out 06								
Reserve	n.c.	Out 07								
*) Die Ansteuerung mehrerer Relais erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.										

MODBus Crossliste,  
Adressierung *modulübergreifend*  
und *Einzeldatenpunkt-Abfrage*  
mittels Word-Funktionen

BDH 2200			Ziel-/Quelladresse	Type	Funktion	Status	Meldung eines Schalters*)	Bitmuster	
Schalter 1	DI 1	Inp 00	Index [i] x 4 + 1003	Word	4	1 = Hand	<b>h 01</b>	xxxx xx01	
Schalter 2	DI 2	Inp 01				1 = Hand	<b>h 02</b>	xxxx xx10	
Reserve	n.c.	Inp 02				Read Input Registers			
Reserve	n.c.	Inp 03							
Reserve	n.c.	Inp 04							
Reserve	n.c.	Inp 05							
Reserve	n.c.	Inp 06							
Reserve	n.c.	Inp 07							
*) Die Statusmeldung mehrerer Schalter erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.									
BDH 2200			Ziel-/Quelladresse	Type	Funktion	Status	Setzen eines Parameters*)	Bitmuster	
LED 1 rot	DO 7	Out 08	Index [i] x 4 + 1000	Word	16	1 = Ein	<b>h 01</b>	xx00 0001	
LED 1 grün	DO 8	Out 09				1 = Ein	<b>h 02</b>	xx00 0010	
LED 2 grün	DO 9	Out 10				Preset Multiple Registers	1 = Ein	<b>h 04</b>	xx00 0100
LED 3 grün	DO 10	Out 11					1 = Ein	<b>h 08</b>	xx00 1000
LED 4 rot	DO 11	Out 12					1 = Ein	<b>h 10</b>	xx01 0000
LED 4 grün	DO 12	Out 13					1 = Ein	<b>h 20</b>	xx10 0000
Reserve	n.c.	Out 14							
Reserve	n.c.	Out 15							
LED 5 grün	DO 13	Out 16	Index [i] x 4 + 1001	Word	[16]	1 = Ein	<b>h 01</b>	xxxx xx01	
LED 6 grün	DO 14	Out 17				1 = Ein	<b>h 02</b>	xxxx xx10	
Reserve	n.c.	Out 18							
Reserve	n.c.	Out 19							
Reserve	n.c.	Out 20							
Reserve	n.c.	Out 21							
Reserve	n.c.	Out 22							
Reserve	n.c.	Out 23							
Relais 1	DO 1	Out 00	Index [i] x 4 + 1002	Word	[16]	1 = Ein	<b>h 01</b>	xxxx 0001	
Relais 2	DO 2	Out 01				1 = Ein	<b>h 02</b>	xxxx 0010	
Relais 3	DO 3	Out 02				1 = Ein	<b>h 04</b>	xxxx 0100	
Relais 4	DO 4	Out 03				1 = Ein	<b>h 08</b>	xxxx 1000	
Reserve	n.c.	Out 04							
Reserve	n.c.	Out 05							
Reserve	n.c.	Out 06							
Reserve	n.c.	Out 07							
*) Die Ansteuerung mehrerer LEDs/Relais erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.									

**MODBus Crossliste,**  
**Adressierung modulübergreifend**  
**und Einzeldatenpunkt-Abfrage**  
**mittels Word-Funktionen**

BDH 2201			Ziel-/Quelladresse	Type	Funktion	Status	Meldung eines Parameters*)	Bitmuster	
Schalter 1	DI 1	Inp 00	Index [i] x 4 + 1001	Word	4	1 = Hand	<b>h 01</b>	xxxx xx01	
Schalter 2	DI 2	Inp 01				1 = Hand	<b>h 02</b>	xxxx xx10	
Reserve	n.c.	Inp 02				Read Input Registers			
Reserve	n.c.	Inp 03							
Reserve	n.c.	Inp 04							
Reserve	n.c.	Inp 05							
Reserve	n.c.	Inp 06							
Reserve	n.c.	Inp 07							
LED 1 rot	DI 7	Inp 08	Index [i] x 4 + 1002	Word	[4]	0 = EIN	<b>h 3E</b>	xx11 1110	
LED 2 grün	DI 8	Inp 09				0 = EIN	<b>h 3D</b>	xx11 1101	
LED 3 grün	DI 9	Inp 10				0 = EIN	<b>h 3B</b>	xx11 1011	
LED 4 rot	DI 10	Inp 11				0 = EIN	<b>h 37</b>	xx11 0111	
LED 5 grün	DI 11	Inp 12				0 = EIN	<b>h 2F</b>	xx10 1111	
LED 6 grün	DI 12	Inp 13				0 = EIN	<b>h 1F</b>	xx01 1111	
Reserve	n.c.	Inp 14							
Reserve	n.c.	Inp 15							
Reserve	n.c.	Inp 16	Index [i] x 4 + 1003	Word	[4]		<b>h XX</b>	xxxx xxxx	
Reserve	n.c.	Inp 17							
Reserve	n.c.	Inp 18							
Reserve	n.c.	Inp 19							
Reserve	n.c.	Inp 20							
Reserve	n.c.	Inp 21							
Reserve	n.c.	Inp 22							
Reserve	n.c.	Inp 23							
*) Die Statusmeldung mehrerer Schalter/LEDs erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.									
BDH 2201			Ziel-/Quelladresse	Type	Funktion	Status	Ansteuerung eines Relais*)	Bitmuster	
Relais 1	DO 1	Out 00	Index [i] x 4 + 1000	Word	16	1 = EIN	<b>h 01</b>	xxxx 0001	
Relais 2	DO 2	Out 01				1 = EIN	<b>h 02</b>	xxxx 0010	
Relais 3	DO 3	Out 02				Preset Multiple Registers	1 = EIN	<b>h 04</b>	xxxx 0100
Relais 4	DO 4	Out 03					1 = EIN	<b>h 08</b>	xxxx 1000
Reserve	n.c.	Out 04							
Reserve	n.c.	Out 05							
Reserve	n.c.	Out 06							
Reserve	n.c.	Out 07							
*) Die Ansteuerung mehrerer Relais erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.									

MODBus Crossliste,  
Adressierung modulübergreifend  
und Einzeldatenpunkt-Abfrage  
mittels Word-Funktionen

BDH 4800			Ziel-/Quelladresse	Type	Funktion	Status	Meldung eines Tasters*)	Bitmuster	
Taster 1	DI 1	Inp 00	Index [i] x 4 + 1003	Word	4	1 = betät.	<b>h 01</b>	xxxx 0001	
Taster 2	DI 2	Inp 01				1 = betät.	<b>h 02</b>	xxxx 0010	
Taster 3	DI 3	Inp 02				<b>Read</b>	1 = betät.	<b>h 04</b>	xxxx 0100
Taster 4	DI 4	Inp 03				<b>Input</b>	1 = betät.	<b>h 08</b>	xxxx 1000
Reserve	n.c.	Inp 04			<b>Registers</b>				
Reserve	n.c.	Inp 05							
Reserve	n.c.	Inp 06							
Reserve	n.c.	Inp 07							
			*) Die Statusmeldung mehrerer Taster erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.						
BDH 4800			Ziel-/Quelladresse	Type	Funktion	Status	Setzen eines Parameters*)	Bitmuster	
LED 1 orange	DO 7	Out 08	Index [i] x 4 + 1000	Word	16	1 = Ein	<b>h 02</b>	xx00 0010	
LED 2 rot-blink.	DO 8	Out 09				1 = Ein	<b>h 01</b>	xx00 0001	
LED 3 rot	DO 9	Out 10				<b>Preset</b>	1 = Ein	<b>h 04</b>	xx00 0100
LED 4 orange	DO 10	Out 11				<b>Multiple</b>	1 = Ein	<b>h 08</b>	xx00 1000
LED 5 rot-blink.	DO 11	Out 12			<b>Registers</b>	1 = Ein	<b>h 10</b>	xx01 0000	
LED 6 rot	DO 12	Out 13				1 = Ein	<b>h 20</b>	xx10 0000	
Reserve	n.c.	Out 14							
Reserve	n.c.	Out 15							
LED 7 orange	DO 13	Out 16	Index [i] x 4 + 1001	Word	[16]	1 = Ein	<b>h 01</b>	xx00 0001	
LED 8 rot-blink.	DO 14	Out 17				1 = Ein	<b>h 02</b>	xx00 0010	
LED 9 rot	DO 15	Out 18				1 = Ein	<b>h 04</b>	xx00 0100	
LED 10 orange	DO 16	Out 19				1 = Ein	<b>h 10</b>	xx01 0000	
LED 11 rot-blink.	DO 17	Out 20				1 = Ein	<b>h 08</b>	xx00 1000	
LED 12 rot	DO 18	Out 21				1 = Ein	<b>h 20</b>	xx10 0000	
Reserve	n.c.	Out 22							
Reserve	n.c.	Out 23							
Relais 1	DO 1	Out 00	Index [i] x 4 + 1002	Word	[16]	1 = Ein	<b>h 01</b>	xxxx 0001	
Relais 2	DO 2	Out 01				1 = Ein	<b>h 02</b>	xxxx 0010	
Relais 3	DO 3	Out 02				1 = Ein	<b>h 04</b>	xxxx 0100	
Relais 4	DO 4	Out 03				1 = Ein	<b>h 08</b>	xxxx 1000	
Reserve	n.c.	Out 04							
Reserve	n.c.	Out 05							
Reserve	n.c.	Out 06							
Reserve	n.c.	Out 07							
			*) Die Ansteuerung mehrerer LEDs/Relais erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.						

**MODBus Crossliste,**  
**Adressierung modulübergreifend**  
**und Einzeldatenpunkt-Abfrage**  
**mittels Word-Funktionen**

BAH 4000			Ziel-/Quelladresse	Type	Funktion	Status	Meldung eines Schalters*)	Bitmuster
<b>Digitaleingänge</b>								
Input 1	DI 1	Inp 00	Index [i] x 4 + 1000	Word	<b>4</b>	1 = Hand	<b>h 01</b>	xxxx 0001
Input 2	DI 2	Inp 01				1 = Hand	<b>h 02</b>	xxxx 0010
Input 3	DI 3	Inp 02			<b>Read</b>	1 = Hand	<b>h 04</b>	xxxx 0100
Input 4	DI 4	Inp 03			<b>Input</b>	1 = Hand	<b>h 08</b>	xxxx 1000
Reserve	n.c.	Inp 04			<b>Registers</b>			
Reserve	n.c.	Inp 05						
Reserve	n.c.	Inp 06						
Reserve	n.c.	Inp 07						
*) Die Statusmeldung mehrerer Schalter erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.								
<b>BAH 4000</b>			<b>Ziel-/Quelladresse</b>	<b>Type</b>	<b>Funktion</b>	<b>Status</b>		
<b>Analogeingänge</b>								
Input 1			Index [i] x 8 + 1064	Word	<b>4</b>	0 ... 255	(Low-Byte, High-Byte = 00)	
Input 2			Index [i] x 8 + 1065		<b>Read</b>	0 ... 255	(Low-Byte, High-Byte = 00)	
Input 3			Index [i] x 8 + 1066		<b>Input</b>	0 ... 255	(Low-Byte, High-Byte = 00)	
Input 4			Index [i] x 8 + 1067		<b>Registers</b>	0 ... 255	(Low-Byte, High-Byte = 00)	
<b>BAH 4000</b>			<b>Ziel-/Quelladresse</b>	<b>Type</b>	<b>Funktion</b>	<b>Status</b>		
<b>Analogausgänge</b>								
Output 1			Index [i] x 8 + 1068	Word	<b>16</b>	0 ... 255	(Low-Byte, High-Byte = 00)	
Output 2			Index [i] x 8 + 1069		<b>Preset</b>	0 ... 255	(Low-Byte, High-Byte = 00)	
Output 3			Index [i] x 8 + 1070		<b>Multiple</b>	0 ... 255	(Low-Byte, High-Byte = 00)	
Output 4			Index [i] x 8 + 1071		<b>Registers</b>	0 ... 255	(Low-Byte, High-Byte = 00)	

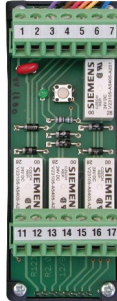
**MODBus Crossliste,**  
**Adressierung modulübergreifend**  
**und Einzeldatenpunkt-Abfrage**  
**mittels Word-Funktionen**

BZK 1000			Ziel-/Quelladresse	Type	Funktion	Status	Meldung eines Tasters*)	Bitmuster
<b>Digitaleingänge</b>								
Entsperren	DI 1	Inp 00	R 70	Word	4	1 = betät.	h 01	xxxx x001
Hupe quittieren	DI 2	Inp 01			Read	1 = betät.	h 02	xxxx x010
Lampentest	DI 3	Inp 02			Input	1 = betät.	h 04	xxxx x100
					Registers			
*) Die Statusmeldung mehrerer Taster erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.								
BZK 1000			Ziel-/Quelladresse	Type	Funktion	Status	Ansteuerung eines Relais*)	Bitmuster
<b>Digitalausgänge</b>								
Reset	DO 1	Out 00	R 70	Word	16	1 = Ein	h 01	xxxx 0001
Sammelalarm	DO 2	Out 01			Preset	1 = Ein	h 02	xxxx 0010
Hupenkontakt	DO 3	Out 02			Multiple	1 = Ein	h 04	xxxx 0100
Lampentest	DO 4	Out 03			Registers	1 = Ein	h 08	xxxx 1000
*) Die Ansteuerung mehrerer Relais erfolgt durch Überlagerung der relevanten Bits durch binäres ODER.								
BZK 1000			Ziel-/Quelladresse	Type	Funktion	Status	Ansteuerung des BAH Adr.x	Bitmuster
<b>Aktivierung</b>								
Lampentest für BAH400x	BAH Adr. 0		R 52	Word	6	1 = Ein	h 01	0000 0001
	...						...	...
	BAH Adr. 7					1 = Ein	h 80	1000 0000
	alle BAH Adr. 0..7					1 = Ein	h FF	1111 1111
<b>Allgemeine</b>							<b>Defaultwerte</b>	
<b>Konfiguration</b>	<b>Wertbereich</b>	<b>Zieladresse</b>		<b>Type</b>	<b>Funktion</b>	<b>dezimal</b>	<b>hexadezimal</b>	
Stationsnummer	0..249	32: <Station> 33: NOT <Station>		Word	16	d 160	h 00 A0 h FF 5F	
Baudrate	6 = 9600 Baud 7 = 19200 Baud 8 = 38400 Baud	34: <Baud> 35: NOT <Baud>		Word	[16]	d 6	h 00 06 h FF F9	
Modus	7=Modbus RTU	36: <Modus> 37: NOT <Modus>		Word	[16]	d 7	h 00 07 h FF F8	
Schnittstelle	4 = RS485	38: <Uart> 39: NOT <Uart>		Word	[16]	d 4	h 00 04 h FF FB	
<b>Wichtig: Es müssen alle vier Parameter (R32...R39) zusammen mit einem einzigen Befehl geschrieben werden, auch wenn nur ein einziger Wert geändert werden soll!</b>								
<b>Busaktivitäts-Überwachung</b>	Zeit in Sek.	42: <Zeit> 43: NOT <Zeit>		Word	16	d 10	h 00 0A h FF F5	

Die Anbindung einer romutec-Handbedienebene an diverse SPS-Systeme wird mit einem Modul des Typs BZK1000MOD (Protokoll MOD-RTU) realisiert.

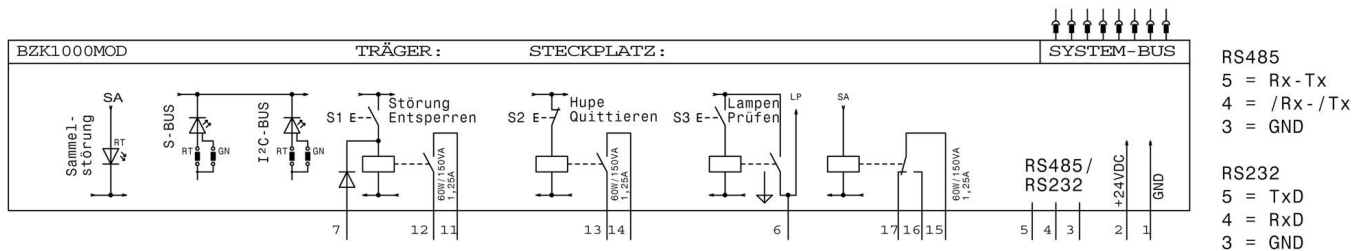
Dieses übernimmt folgende Funktionen:

- Herstellung der Buskommunikation zwischen SPS und der rt-Handbedienebene (es können bis zu 8 digitale oder analoge Module an ein BZK angeschlossen werden; maximale Gesamtzahl der anschließbaren Erweiterungsmodule = 9)
- Überwachung und Statusanzeige der Buskommunikation (I2C- und MOD-BUS)
- Lampentest aller angeschlossenen Erweiterungsmodule
- Signalisierung einer im System gebildeten Sammelstörung mittels rot blinkender LED und potenzialfreiem Relaiskontakt (Wechsler)
- Quittierung und Entsperrn von externen Störungen (keine Speicherung)
- Spannungsversorgung der angeschlossenen digitalen Erweiterungsmodule



Bedeutung der beiden Status-LED's zur Signalisierung der Buskommunikation: LED I2C: Aus: Kein Datenaustausch; Kurz (ca. 50 msec): Datenaustausch über den I2C-Bus (rt-BUS o.k.); Lang (ca. 300 msec): Baugruppe nicht vorhanden oder illeale Parameter im Telegramm (Fehler rt-BUS); LED MOD-BUS: Blinken, langsam: Betriebsbereitschaft; Kurz (ca. 100 msec): Leuchtet während der Telegrammübertragung, wenn die eigene Stationsnummer angesprochen wird.

Bezüglich der Anlagenkonfiguration (Adressierung, maximale Anzahl von Erweiterungsmodulen an einem BZK, Montage, Busverbindungen etc.) sind die allgemeinen Hinweise in der Systembeschreibung im Kapitel Konfiguration zu beachten. Siehe auch Systembeschreibung BUSStecMOD-RTU !



Versorgungsspannung 24V DC, ±10%

Stromaufnahme max. 150 mA

Prozessor Typ P89C660

Taktfrequenz 14,7456 MHz Bus-Schnittstelle MOD RTU --> I2C

Speicher µPC-intern Protokoll MOD RTU-BUS

potenzialgebundene digitale Ein-/Ausgänge: 24V DC

Relaisdaten: Schaltspannung max.: 250 VAC / 30 VDC  
Schaltstrom max. (Resistiv): 5 A  
Schaltleistung: 625 VA / 150 W

Nennlast (Resistiv): 2,5 A / 250 VAC; 5 A / 30 VDC

Lebensdauer  
bei Nennlast 50.000 Schaltspiele  
ohne Last 5.000.000 Schaltspiele

Umgebungsbedingungen  
Betriebstemperatur 0...50°C  
Transport- und Lagertemperatur -20...+70°C  
Relative Feuchte 5...95%, nicht kondensierend

Schutzart: IP 40, durch Abdeckung mit Sichtfenster bis IP 54

Abmessungen: 19"-Geräte  
Breite 8 TE (40.5 mm), Höhe 3 HE (129 mm), Einbautiefe < 80 mm

Anschlußklemmen: 2,5mm<sup>2</sup> steckbar, Schraubanschluß

CE-Konformität  
EN 61000-4-2 / IEC 801-2 Elektrostatische Entladung ESD  
Kontaktentladung 8 kV / Luftentladung 8 kV  
EN 61000-4-5 / IEC 801-5 Surge-Prüfung  
Versorgungspg. AC 4 kV, DC 0,5 kV  
Signalleitungen 2 kV  
EN 61000-4-4 / IEC 801-4 Burst-Prüfung  
Eingänge - Ausgänge +/- 2 kV  
Versorgungsspannung AC / DC +/- 2 kV

00002239

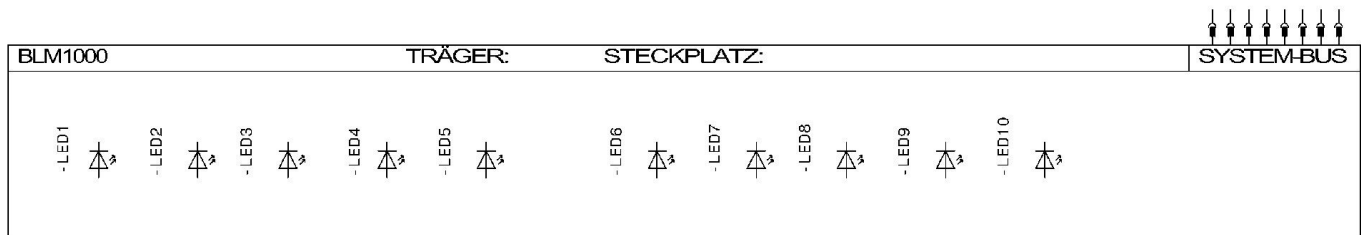
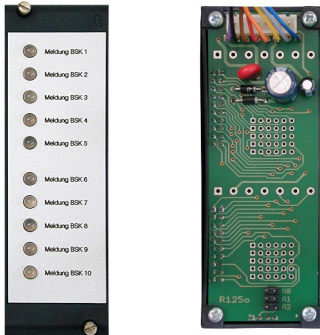
BLM1000

Das Lampen-Melde-Modul BLM 1000 dient zur Signalisierung von 10 Meldungen. Das BLM 1000 findet damit hauptsächlich Anwendung zur Signalisierung von softwaremäßig erfaßten Meldungen wie z.B. im System gebildeten Gruppenmeldungen oder der Statusanzeige.

Blinkende Störmeldungen werden an das Buszentralmodul übergeben, wo eine Sammelstörmeldung aller angeschlossenen Module gebildet wird.

Bezüglich der Anlagenkonfiguration (Adressierung, maximale Anzahl von Erweiterungsmodulen an einem BZK, Montage, Busverbindungen etc.) sind die allgemeinen Hinweise im Kapitel Konfiguration zu beachten.

Modul ist in Kombination mit dem BZK1000MOD (Modbus RTU) sowie SBZK1000 (S-Bus) verwendbar.



Versorgungsspannung 24V DC,  $\pm 10\%$

Stromaufnahme max. 120 mA

Umgebungsbedingungen

Betriebstemperatur 0...50°C

Transport- und Lagertemperatur -20...+70°C

Relative Feuchte 5...95%, nicht kondensierend

Schutzart: IP 40, durch Abdeckung mit Sichtfenster bis IP 54

Abmessungen: 19"-Geräte

Breite 8 TE (40.5 mm), Höhe 3 HE (129 mm), Einbautiefe < 80 mm

CE-Konformität

EN 61000-4-2 / IEC 801-2 Elektrostatistische Entladung ESD

Kontaktentladung 8 kV / Luftentladung 8 kV

EN 61000-4-5 / IEC 801-5 Surge-Prüfung

Versorgungspg. AC 4 kV, DC 0,5 kV

Signalleitungen 2 kV

EN 61000-4-4 / IEC 801-4 Burst-Prüfung

Eingänge - Ausgänge +/- 2 kV

Versorgungsspannung AC / DC +/- 2 kV



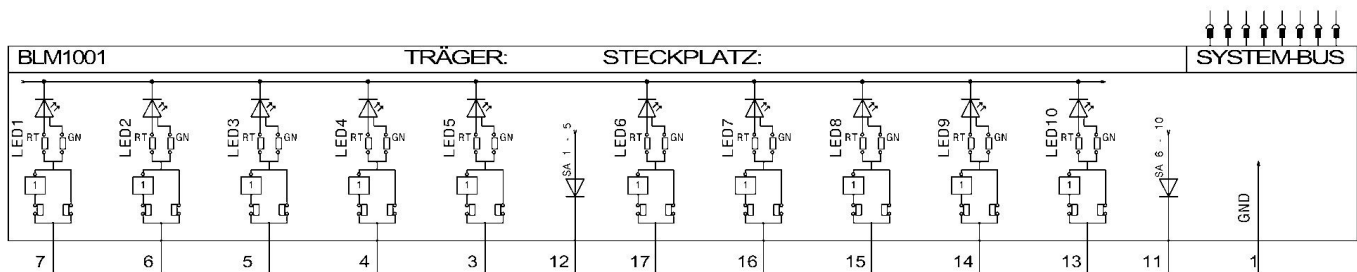
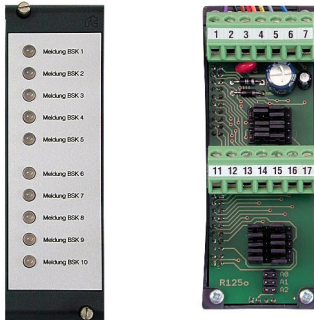
Das Lampen-Melde-Modul BLM 1001 dient zur Signalisierung von bis zu 10 Meldungen. Dazu zählen Betriebsmeldungen, Störmeldungen wie Frost, Filter oder Keilriemen sowie Statusmeldungen wie z.B. Klappenstellungen.

Die Ansteuerung der LEDs erfolgt mit +24 V DC durch externe potenzialfreie Kontakte, die über abziehbare Schraubklemmen auf die Karte aufgeschaltet werden.

Über Jumper kann für jeden einzelnen Eingang Arbeits- oder Ruhestromprinzip gewählt werden. Die Farbe jeder der 10 LED's ist ebenfalls über Jumper wählbar, entweder auf Rot (blinkend) oder Grün (dauerleuchtend). Blinkende Störmeldungen (Rot) werden dabei an das Buszentralmodul BZK übergeben, wo von der Hardware eine Sammelstörmeldung aller angeschlossenen Module gebildet wird.

Aus Störmeldungen der Eingänge 1-5 sowie 6-10 werden 2 Gruppenmeldungen gebildet, die über Klemmen abgreifbar sind (potenzialgebunden, +24 V).

Modul ist in Kombination mit dem BZK1000MOD (Modbus RTU) und SBZK1000 (S-Bus) verwendbar.



Versorgungsspannung 24V DC, ±10%

Stromaufnahme max. 120 mA

Ansteuerung Eingänge 24V DC

Gruppenausgänge 24V DC

Umgebungsbedingungen

Betriebstemperatur 0...50°C

Transport- und Lagertemperatur -20...+70°C

Relative Feuchte 5...95%, nicht kondensierend

Schutzart: IP 40, durch Abdeckung mit Sichtfenster bis IP 54

Abmessungen: 19"-Geräte

Breite 8 TE (40.5 mm), Höhe 3 HE (129 mm), Einbautiefe < 80 mm

Anschlußklemmen: 2,5mm 2 steckbar, Schraubanschluß

CE-Konformität

EN 61000-4-2 / IEC 801-2 Elektrostatische Entladung ESD

Kontaktentladung 8 kV / Luftentladung 8 kV

EN 61000-4-5 / IEC 801-5 Surge-Prüfung

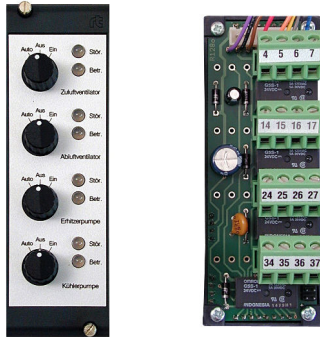
Versorgungspg. AC 4 kV, DC 0,5 kV

Signalleitungen 2 kV

EN 61000-4-4 / IEC 801-4 Burst-Prüfung

Eingänge - Ausgänge +/- 2 kV

Versorgungsspannung AC / DC +/- 2 kV



Die Karte BDH 1400 dient als Handbedienebene mit Notfunktion zur Ansteuerung von vier 1-stufigen Motoren (Auto-Aus-Hand) bzw. Klappen (2-Punktsteuerung, Auto-Zu-Auf). Jeder der 4 Kanäle verfügt über einen Relaisausgang (Wechsler) zur Ansteuerung von Leistungsschützen sowie über zwei LEDs zur Signalisierung von Stör-, Betriebs- oder Statusmeldungen.

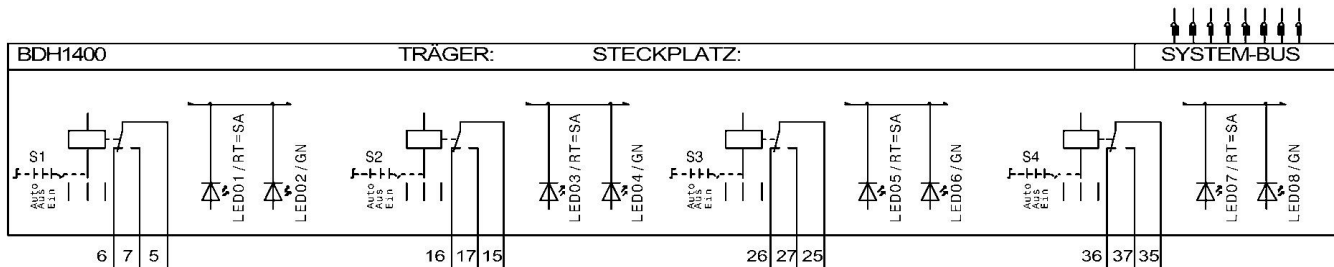
Die Ansteuerung der LEDs erfolgt über den Bus. Die LEDs 1, 3, 5 und 7 können sowohl rot/blinkend als auch grün/dauerleuchtend bzw. gelb-grün/blinkend angesteuert werden. Im Fall einer rot oder gelb leuchtenden LED wird am Zentralmodul eine Sammelstörmeldung ausgegeben; die LEDs 2, 4, 6 und 8 sind grün/dauerleuchtend ausgeführt. Die Information zur Ansteuerung der LEDs 1 - 8 wird in den Bytes 2 - 9 der Netzwerk-Eingangs-Variablen nvi (SNVT\_str\_int) an die Steuerkarte übertragen.

Die Information zur Relais-Ansteuerung wird in den Bytes 10 bis 13 der Netzwerk-Eingangs-Variablen nvi (SNVT\_str\_int) an das BDH 1400 übergeben.

Die Schalterstellung (Auto bzw. Aus/Hand oder Zu/Auf) wird in den Bytes 2 bis 5 der Netzwerk-Ausgangs-Variable nvo (SNVT\_str\_int) an den LON-Knoten übergeben.

Bezüglich der Anlagenkonfiguration (Adressierung, maximale Anzahl von Erweiterungsmodulen an einem BZK, Montage, Busverbindungen etc.) sind die allgemeinen Hinweise im Kapitel Konfiguration zu beachten.

Modul ist auch in Kombination mit dem BZK1000MOD (Modbus RTU) verwendbar.



Versorgungsspannung 24V DC, ±10%

Stromaufnahme max. 155 mA

Relaisdaten:

Schaltspannung, max.: 250 VAC / 30 VDC

Schaltstrom, max. (Resistiv): 5 A

Schaltleistung: 625 VA / 150 W

Nennlast (Resistiv): 2,5 A / 250 VAC; 5 A / 30 VDC

Lebensdauer

bei Nennlast 50.000 Schaltspiele

ohne Last 5.000.000 Schaltspiele

Umgebungsbedingungen

Betriebstemperatur 0...50°C

Transport- und Lagertemperatur -20...+70°C

Relative Feuchte 5...95%, nicht kondensierend

Schutzart: IP 40, durch Abdeckung mit Sichtfenster bis IP 54

Abmessungen: 19"-Geräte

Breite 8 TE (40.5 mm), Höhe 3 HE (129 mm), Einbautiefe < 80 mm

Anschlussklemmen: 2,5mm<sup>2</sup> steckbar, Schraubanschluß

CE-Konformität

EN 61000-4-2 / IEC 801-2 Elektrostatische Entladung ESD

Kontaktentladung 8 kV / Luftentladung 8 kV

EN 61000-4-5 / IEC 801-5 Surge-Prüfung

Versorgungspg. AC 4 kV, DC 0,5 kV

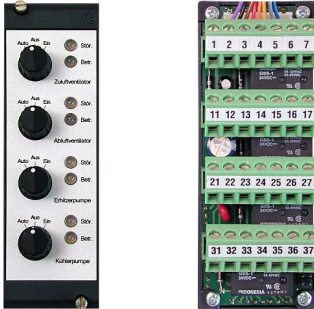
Signalleitungen 2 kV

EN 61000-4-4 / IEC 801-4 Burst-Prüfung

Eingänge - Ausgänge +/- 2 kV

Versorgungsspannung AC / DC +/- 2 kV

Die Karte BDH 1401 dient als Handbedienebene mit Notfunktion zur Ansteuerung von vier 1-stufigen Motoren. Jeder der 4 Kanäle verfügt über einen Relaisausgang (Wechsler) zur Ansteuerung von Leistungsschützen sowie über zwei LEDs zur Signalisierung von Stör- und Betriebsmeldungen. Die LEDs 1, 3, 5 und 7 sind rot/blinkend ausgeführt und bewirken bei Ansteuerung eine Sammelstörmeldung auf dem Zentralmodul; die LEDs 2, 4, 6 und 8 sind grün/dauerleuchtend ausgeführt.

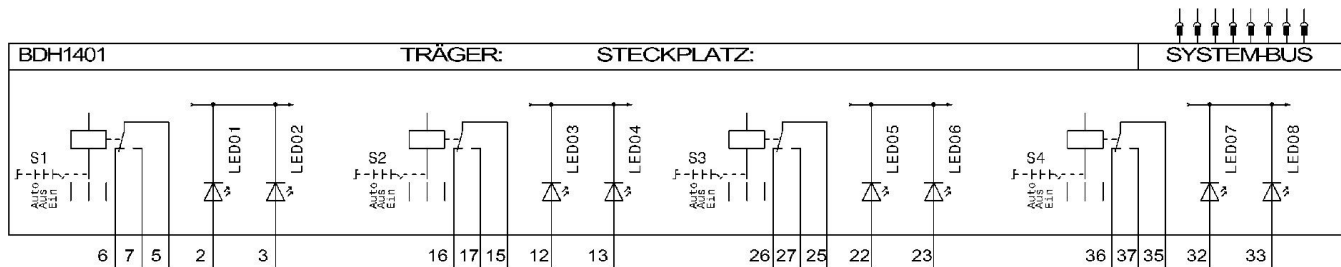


Die Ansteuerung der LEDs erfolgt mit +24 V DC durch externe potenzialfreie Kontakte, die über Klemmen auf die Karte aufgeschaltet werden. Der Status der LEDs sowie die Schalterstellung (Auto oder Aus/Hand) werden über die Netzwerk-Ausgangs-Variable nvo (SNVT\_str\_int) an den LON-Knoten übergeben.

Die Information zur Relais-Ansteuerung wird in den Bytes 2 bis 5 der Netzwerk-Eingangs-Variablen nvi (SNVT\_str\_int) an das BDH 1401 übergeben.

Bezüglich der Anlagenkonfiguration (Adressierung, maximale Anzahl von Erweiterungsmodulen an einem BZK, Montage, Busverbindungen etc.) sind die allgemeinen Hinweise im Kapitel Konfiguration zu beachten.

Modul ist auch in Kombination mit dem BZK1000MOD (Modbus RTU) verwendbar.



Versorgungsspannung 24V DC,  $\pm 10\%$

Stromaufnahme max. 155 mA

Ansteuerung Eingänge 24V DC

Relaisdaten:

Schaltspannung, max.: 250 VAC / 30 VDC

Schaltstrom, max. (Resistiv): 5 A

Schaltleistung: 625 VA / 150 W

Nennlast (Resistiv): 2,5 A / 250 VAC; 5 A / 30 VDC

Lebensdauer

bei Nennlast 50.000 Schaltspiele

ohne Last 5.000.000 Schaltspiele

Umgebungsbedingungen

Betriebstemperatur 0...50°C

Transport- und Lagertemperatur -20...+70°C

Relative Feuchte 5...95%, nicht kondensierend

Schutzart: IP 40, durch Abdeckung mit Sichtfenster bis IP 54

Abmessungen: 19"-Geräte

Breite 8 TE (40.5 mm), Höhe 3 HE (129 mm), Einbautiefe < 80 mm

Anschlußklemmen: 2,5mm2 steckbar, Schraubanschluß

CE-Konformität

EN 61000-4-2 / IEC 801-2 Elektrostatische Entladung ESD

Kontaktentladung 8 kV / Luftentladung 8 kV

EN 61000-4-5 / IEC 801-5 Surge-Prüfung

Versorgungspg. AC 4 kV, DC 0,5 kV

Signalleitungen 2 kV

EN 61000-4-4 / IEC 801-4 Burst-Prüfung

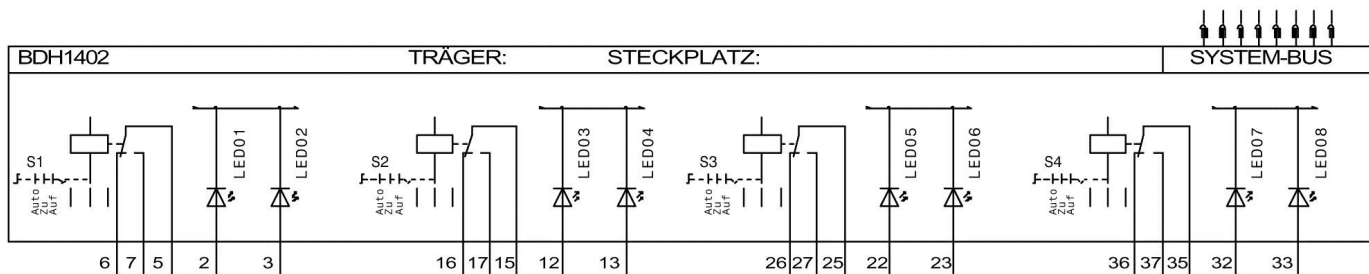
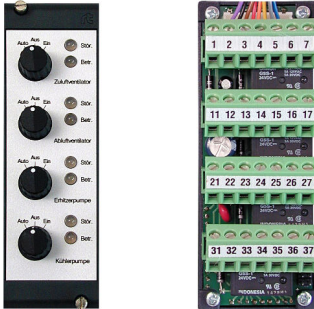
Eingänge - Ausgänge +/- 2 kV

Versorgungsspannung AC / DC +/- 2 kV

Die Karte BDH1402 dient als Handbedienebene mit Notfunktion zur Ansteuerung von vier Klappenantrieben. Jeder der 4 Kanäle verfügt über einen Relaisausgang (Wechsler) zur Ansteuerung von Leistungsschützen sowie über zwei LEDs zur Signalisierung von Auf- und ZUMeldungen. Die LEDs SIND grün/dauerleuchtend ausgeführt.

Die Ansteuerung der LEDs erfolgt mit +24 V DC durch externe potenzialfreie Kontakte, die über Klemmen auf die Karte aufgeschaltet werden. Der Status der LEDs sowie die Schalterstellung (Auto oder Aus/Hand) werden über den Bus und das Zentralmodul BZK1000LON an das LON Netzwerk übergeben, wo sie zur weiteren Verarbeitung zur Verfügung stehen.

Bezüglich der Anlagenkonfiguration (Adressierung, maximale Anzahl von Erweiterungsmodulen an einem BZK, Montage, Busverbindungen etc.) sind die allgemeinen Hinweise im Kapitel Konfiguration zu beachten.



Versorgungsspannung 24V DC, ±10%

Stromaufnahme max. 155 mA

Ansteuerung Eingänge 24V DC

Relaisdaten:

Schaltspannung, max.: 250 VAC / 30 VDC

Schaltstrom, max. (Resistiv): 5 A

Schaltleistung: 625 VA / 150 W

Nennlast (Resistiv): 2,5 A / 250 VAC; 5 A / 30 VDC

Lebensdauer

bei Nennlast 50.000 Schaltspiele

ohne Last 5.000.000 Schaltspiele

Umgebungsbedingungen

Betriebstemperatur 0...50°C

Transport- und Lagertemperatur -20...+70°C

Relative Feuchte 5...95%, nicht kondensierend

Schutzart: IP 40, durch Abdeckung mit Sichtfenster bis IP 54

Abmessungen: 19"-Geräte

Breite 8 TE (40.5 mm), Höhe 3 HE (129 mm), Einbautiefe < 80 mm

Anschlußklemmen: 2,5mm<sup>2</sup> steckbar, Schraubanschluß

CE-Konformität

EN 61000-4-2 / IEC 801-2 Elektrostatische Entladung ESD

Kontaktentladung 8 kV / Luftentladung 8 kV

EN 61000-4-5 / IEC 801-5 Surge-Prüfung

Versorgungspg. AC 4 kV, DC 0,5 kV

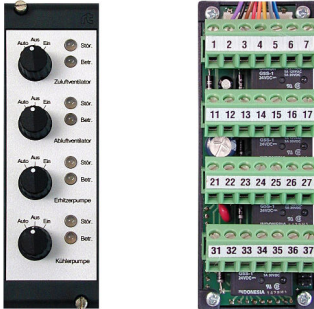
Signalleitungen 2 kV

EN 61000-4-4 / IEC 801-4 Burst-Prüfung

Eingänge - Ausgänge +/- 2 kV

Versorgungsspannung AC / DC +/- 2 kV

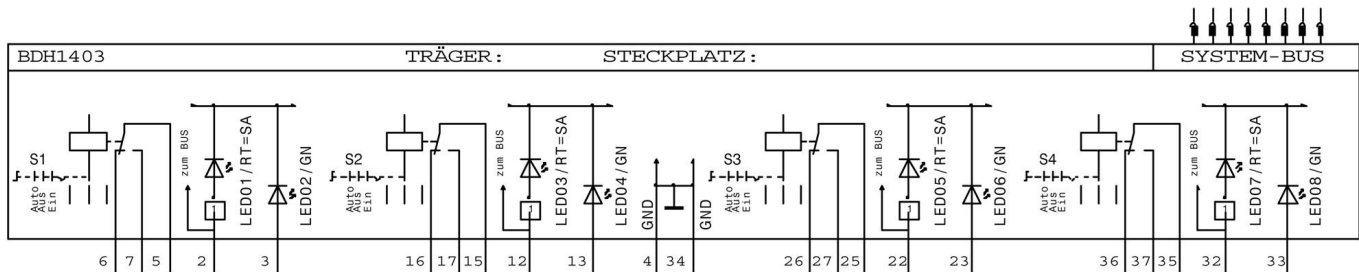
Die Karte BDH 1403 dient als Handbedienebene mit Notfunktion zur Ansteuerung von vier 1-stufigen Motoren. Jeder der 4 Kanäle verfügt über einen Relaisausgang (Wechsler) zur Ansteuerung von Leistungsschützen sowie über zwei LEDs zur Signalisierung von Stör- und Betriebsmeldungen. Die LEDs 1, 3, 5 und 7 sind rot/blinkend ausgeführt und bewirken bei Ansteuerung eine Sammelstörmeldung auf dem Zentralmodul; die LEDs 2, 4, 6 und 8 sind grün/dauerleuchtend ausgeführt.



Die Ansteuerung der LEDs erfolgt mit +24 V DC durch externe potenzialfreie Kontakte mit Ruhestromschaltung, die über Klemmen auf die Karte aufgeschaltet werden. Der Status der LEDs sowie die Schalterstellung (Auto oder Aus/Hand) werden über die Netzwerk-Ausgangs-Variable nvo (SNVT\_str\_int) an den LON-Knoten übergeben. Die Information zur Relais-Ansteuerung wird in den Bytes 2 bis 5 der Netzwerk-Eingangs-Variablen nvi (SNVT\_str\_int) an das BDH 1401 übergeben.

Bezüglich der Anlagenkonfiguration (Adressierung, maximale Anzahl von Erweiterungsmodulen an einem BZK, Montage, Busverbindungen etc.) sind die allgemeinen Hinweise im Kapitel Konfiguration zu beachten.

Modul ist auch in Kombination mit dem BZK1000MOD (Modbus RTU) verwendbar.



Versorgungsspannung 24V DC,  $\pm 10\%$

Stromaufnahme max. 155 mA

Ansteuerung Eingänge 24V DC

Relaisdaten:

Schaltspannung, max.: 250 VAC / 30 VDC

Schaltstrom, max. (Resistiv): 5 A

Schaltleistung: 625 VA / 150 W

Nennlast (Resistiv): 2,5 A / 250 VAC; 5 A / 30 VDC

Lebensdauer

bei Nennlast 50.000 Schaltspiele

ohne Last 5.000.000 Schaltspiele

Umgebungsbedingungen

Betriebstemperatur 0...50°C

Transport- und Lagertemperatur -20...+70°C

Relative Feuchte 5...95%, nicht kondensierend

Schutzart: IP 40, durch Abdeckung mit Sichtfenster bis IP 54

Abmessungen: 19"-Geräte

Breite 8 TE (40.5 mm), Höhe 3 HE (129 mm), Einbautiefe < 80 mm

Anschlußklemmen: 2,5mm<sup>2</sup> steckbar, Schraubanschluß

CE-Konformität

EN 61000-4-2 / IEC 801-2 Elektrostatische Entladung ESD

Kontaktentladung 8 kV / Luftentladung 8 kV

EN 61000-4-5 / IEC 801-5 Surge-Prüfung

Versorgungspg. AC 4 kV, DC 0,5 kV

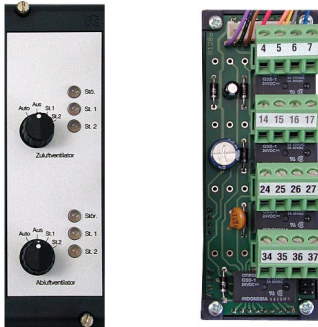
Signalleitungen 2 kV

EN 61000-4-4 / IEC 801-4 Burst-Prüfung

Eingänge - Ausgänge +/- 2 kV

Versorgungsspannung AC / DC +/- 2 kV

Die Karte BDH 2200 dient als Handbedienebene mit Notfunktion zur Ansteuerung von zwei 2-stufigen Motoren (Auto-Aus-Stufe1-Stufe2) bzw. Klappen (3-Punktsteuerung, Auto-Halt-Auf-Zu). Die beiden Kanäle verfügen über je zwei Relaisausgänge (Wechsler) zur Ansteuerung von Leistungsschützen sowie über drei LEDs zur Signalisierung von Stör-, Betriebs- oder Statusmeldungen.



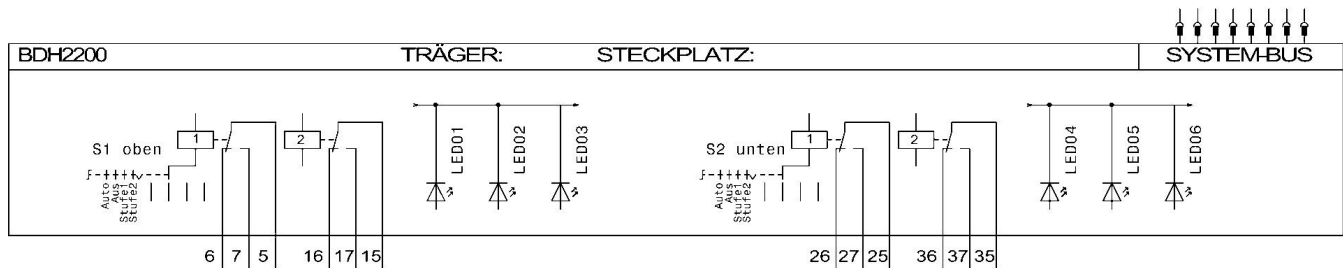
Die Ansteuerung der LEDs erfolgt über den Bus. Die LEDs 1 und 4 können sowohl rot/blinkend als auch grün/dauerleuchtend bzw. gelb-grün/blinkend angesteuert werden. Im Fall einer rot oder gelb leuchtenden LED wird am Zentralmodul eine Sammelstörungsmeldung ausgegeben; die LEDs 2, 3, 5 und 6 sind grün/dauerleuchtend ausgeführt. Die Information zur Ansteuerung der LEDs 1 - 6 wird in den Bytes 2 - 7 der Netzwerk-Eingangs-Variablen nvi (SNVT\_str\_int) an die Steuerkarte übertragen.

Die Information zur Relais-Ansteuerung wird in den Bytes 8 bis 11 der Netzwerk-Eingangs-Variablen nvi (SNVT\_str\_int) an das BDH 2200 übergeben.

Die Schalterstellung (Auto bzw. Aus/St.1/St.2 oder Halt/Auf/Zu) wird in den Bytes 2 und 3 der Netzwerk-Ausgangs-Variable nvo (SNVT\_str\_int) an den LON-Knoten übergeben.

Bezüglich der Anlagenkonfiguration (Adressierung, maximale Anzahl von Erweiterungsmodulen an einem BZK, Montage, Busverbindungen etc.) sind die allgemeinen Hinweise im Kapitel Konfiguration zu beachten.

Modul ist auch in Kombination mit dem BZK1000MOD (Modbus RTU) verwendbar.



Versorgungsspannung 24V DC,  $\pm 10\%$

Stromaufnahme max. 155 mA

Relaisdaten:

Schaltspannung, max.: 250 VAC / 30 VDC

Schaltstrom, max. (Resistiv): 5 A

Schaltleistung: 625 VA / 150 W

Nennlast (Resistiv): 2,5 A / 250 VAC; 5 A / 30 VDC

Umgebungsbedingungen

Betriebstemperatur 0...50°C

Transport- und Lagertemperatur -20...+70°C

Relative Feuchte 5...95%, nicht kondensierend

Schutzart: IP 40, durch Abdeckung mit Sichtfenster bis IP 54

Abmessungen: 19"-Geräte

Breite 8 TE (40.5 mm), Höhe 3 HE (129 mm), Einbautiefe < 80 mm

Anschlussklemmen: 2,5mm2 steckbar, Schraubanschluß

CE-Konformität

EN 61000-4-2 / IEC 801-2 Elektrostatische Entladung ESD

Kontaktentladung 8 kV / Luftentladung 8 kV

EN 61000-4-5 / IEC 801-5 Surge-Prüfung

Versorgungspg. AC 4 kV, DC 0,5 kV

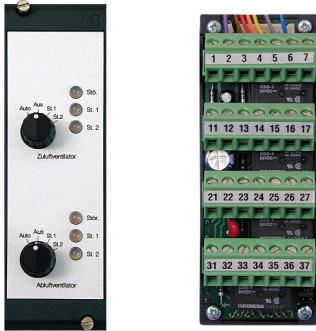
Signalleitungen 2 kV

EN 61000-4-4 / IEC 801-4 Burst-Prüfung

Eingänge - Ausgänge +/- 2 kV

Versorgungsspannung AC / DC +/- 2 kV

Die Karte BDH 2201 dient als Handbedienebene mit Notfunktion zur Ansteuerung von zwei 2-stufigen Motoren. Beide Kanäle verfügen über jeweils zwei Relaisausgänge (Wechsler) zur Ansteuerung von Leistungsschützen sowie über drei LEDs zur Signalisierung von Stör- und Betriebsmeldungen. Die LEDs 1 und 4 sind rot/blinkend ausgeführt und bewirken bei Ansteuerung eine Sammelstörmeldung auf dem Zentralmodul; die LEDs 2, 3, 5 und 6 sind grün/dauerleuchtend ausgeführt.

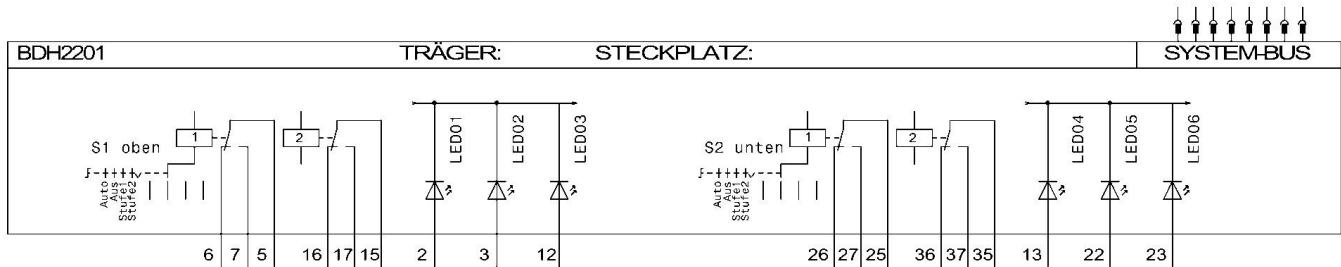


Die Ansteuerung der LEDs erfolgt mit +24 V DC durch externe potenzialfreie Kontakte, die über Klemmen auf die Karte aufgeschaltet werden. Der Status der LEDs sowie die Schalterstellung (Auto bzw. Aus/Hand) werden über die Netzwerk-Ausgangs-Variable nvo (SNVT\_str\_int) an den LON-Knoten übergeben.

Die Information zur Relais-Ansteuerung wird in den Bytes 2 bis 5 der Netzwerk-Eingangs-Variablen nvi (SNVT\_str\_int) an das BDH 2201 übergeben.

Bezüglich der Anlagenkonfiguration (Adressierung, maximale Anzahl von Erweiterungsmodulen an einem BZK, Montage, Busverbindungen etc.) sind die allgemeinen Hinweise im Kapitel Konfiguration zu beachten.

Modul ist auch in Kombination mit dem BZK1000MOD (Modbus RTU) verwendbar.



Versorgungsspannung 24V DC,  $\pm 10\%$

Stromaufnahme max. 155 mA

Ansteuerung Eingänge 24V DC

Relaisdaten:

Schaltspannung, max.: 250 VAC / 30 VDC

Schaltstrom, max. (Resistiv): 5 A

Schaltleistung: 625 VA / 150 W

Nennlast (Resistiv): 2,5 A / 250 VAC; 5 A / 30 VDC

Lebensdauer

bei Nennlast 50.000 Schaltspiele

ohne Last 5.000.000 Schaltspiele

Umgebungsbedingungen

Betriebstemperatur 0...50°C

Transport- und Lagertemperatur -20...+70°C

Relative Feuchte 5...95%, nicht kondensierend

Schutzart: IP 40, durch Abdeckung mit Sichtfenster bis IP 54

Abmessungen: 19"-Geräte

Breite 8 TE (40.5 mm), Höhe 3 HE (129 mm), Einbautiefe < 80 mm

Anschlußklemmen: 2,5mm<sup>2</sup> steckbar, Schraubanschluß

CE-Konformität

EN 61000-4-2 / IEC 801-2 Elektrostatische Entladung ESD

Kontaktentladung 8 kV / Luftentladung 8 kV

EN 61000-4-5 / IEC 801-5 Surge-Prüfung

Versorgungspg. AC 4 kV, DC 0,5 kV

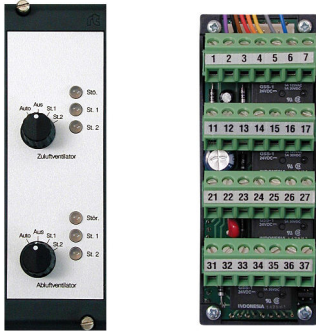
Signalleitungen 2 kV

EN 61000-4-4 / IEC 801-4 Burst-Prüfung

Eingänge - Ausgänge +/- 2 kV

Versorgungsspannung AC / DC +/- 2 kV

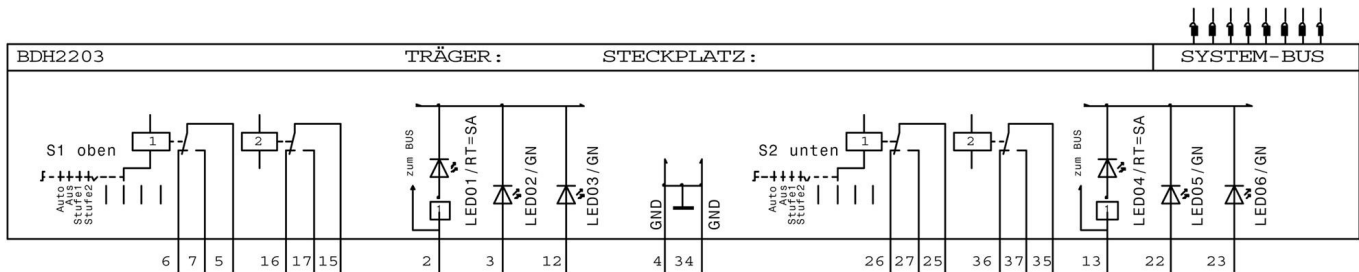
Die Karte BDH 2203 dient als Handbedienebene mit Notfunktion zur Ansteuerung von zwei 2-stufigen Motoren. Beide Kanäle verfügen über jeweils zwei Relaisausgänge (Wechsler) zur Ansteuerung von Leistungsschützen sowie über drei LEDs zur Signalisierung von Stör- und Betriebsmeldungen. Die LEDs 1 und 4 sind rot/blinkend ausgeführt und bewirken bei Ansteuerung eine Sammelstörungsmeldung auf dem Zentralmodul; die LEDs 2, 3, 5 und 6 sind grün/dauerleuchtend ausgeführt.



Die Ansteuerung der LEDs erfolgt mit +24 V DC durch externe potenzialfreie Kontakte mit Ruhestromschaltung, die über Klemmen auf die Karte aufgeschaltet werden. Der Status der LEDs sowie die Schalterstellung (Auto bzw. Aus/Hand) werden über die Netzwerk-Ausgangs-Variablen nvo (SNVT\_str\_int) an den LON-Knoten übergeben. Die Information zur Relais-Ansteuerung wird in den Bytes 2 bis 5 der Netzwerk-Eingangs-Variablen nvi (SNVT\_str\_int) an das BDH 2201 übergeben.

Bezüglich der Anlagenkonfiguration (Adressierung, maximale Anzahl von Erweiterungsmodulen an einem BZK, Montage, Busverbindungen etc.) sind die allgemeinen Hinweise im Kapitel Konfiguration zu beachten.

Modul ist auch in Kombination mit dem BZK1000MOD (Modbus RTU) verwendbar.



Versorgungsspannung 24V DC, ±10%

Stromaufnahme max. 155 mA

Ansteuerung Eingänge 24V DC

Relaisdaten:

Schaltspannung, max.: 250 VAC / 30 VDC

Schaltstrom, max. (Resistiv): 5 A

Schaltleistung: 625 VA / 150 W

Nennlast (Resistiv): 2,5 A / 250 VAC; 5 A / 30 VDC

Lebensdauer

bei Nennlast 50.000 Schaltspiele

ohne Last 5.000.000 Schaltspiele

Umgebungsbedingungen

Betriebstemperatur 0...50°C

Transport- und Lagertemperatur -20...+70°C

Relative Feuchte 5...95%, nicht kondensierend

Schutzart: IP 40, durch Abdeckung mit Sichtfenster bis IP 54

Abmessungen: 19"-Geräte

Breite 8 TE (40.5 mm), Höhe 3 HE (129 mm), Einbautiefe < 80 mm

Anschlussklemmen: 2,5mm<sup>2</sup> steckbar, Schraubanschluß

CE-Konformität

EN 61000-4-2 / IEC 801-2 Elektrostatische Entladung ESD

Kontaktentladung 8 kV / Luftentladung 8 kV

EN 61000-4-5 / IEC 801-5 Surge-Prüfung

Versorgungspg. AC 4 kV, DC 0,5 kV

Signalleitungen 2 kV

EN 61000-4-4 / IEC 801-4 Burst-Prüfung

Eingänge - Ausgänge +/- 2 kV

Versorgungsspannung AC / DC +/- 2 kV



Die Analoggeberkarte BAH 4000 kommt dort zum Einsatz, wo Analogausgänge von Reglern oder DDC-Systemen ggf. durch Handeingriff übersteuert werden müssen. Typischer Einsatz in der Heizungs-, Lüftungs- und Klimatechnik sind z.B. Heizventile, Klappen und Frequenzumrichter.

Das Analoggebermodul BAH 4000 besitzt 4 Kanäle, wovon jeder mit 10 mA belastbar ist. Weiterhin enthält die Karte einen ebenfalls mit 10 mA belastbaren 10 Volt-Festspannungsausgang. Jeder Kanal besitzt einen Schalter, mit dem das Automatiksignal übersteuert werden kann (Auto-0V-10V-Poti), ein Poti zur stufenlosen Einstellung der Ausgangsspannung von Hand sowie eine LED in Hell-/Dunkelschaltung zur optischen Kontrolle der Ausgangsspannung.

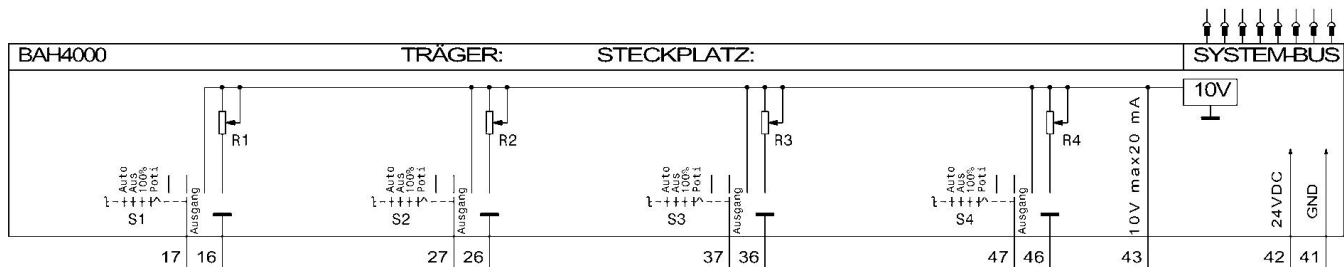
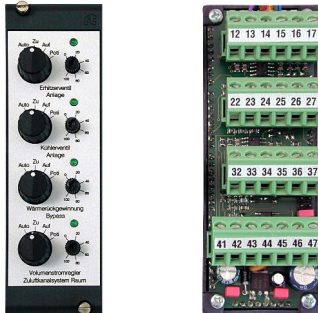
Der Sollwert der Ausgangsspannung im Automatikbetrieb wird über den Bus mittels der Netzwerk-Eingangs-Variablen nvi (SNVT\_str\_int) an die Karte übergeben. Den Kanälen 1 - 4 entsprechen die Bytes 2 - 5, der Wertebereich erstreckt sich von 0 bis 255 (entspricht 0 bis 100%).

Die Schalterstellungen (Auto oder 0V/10V/Poti) und die tatsächlichen Werte der Ausgangsspannungen (sowohl in der Automatik- als auch in den Örtlich-Betriebsarten) werden über die Netzwerk-Ausgangs-Variable nvo (SNVT\_str\_int) an den LON-Knoten übergeben.

Die Spannungsversorgung (24 V DC) ist beim Analoggebermodul BAH 4000 über Klemmen (41, 42) zu verdrahten (siehe Anschlußplan).

Bezüglich der Anlagenkonfiguration (Adressierung, maximale Anzahl von Erweiterungsmodulen an einem BZK, Montage, Busverbindungen etc.) sind die allgemeinen Hinweise im Kapitel Konfiguration zu beachten.

Modul ist auch in Kombination mit dem BZK1000MOD (Modbus RTU) verwendbar.



Versorgungsspannung 24V DC, ±10%

Stromaufnahme max. 220 mA

Belastbarkeit je Analogausgang: 10 mA

Umgebungsbedingungen

Betriebstemperatur 0...50°C

Transport- und Lagertemperatur -20...+70°C

Relative Feuchte 5...95%, nicht kondensierend

Schutzart: IP 40, durch Abdeckung mit Sichtfenster bis IP 54

Abmessungen: 19"-Geräte

Breite 8 TE (40.5 mm), Höhe 3 HE (129 mm), Einbautiefe < 80 mm

Anschlußklemmen: 2,5mm<sup>2</sup> steckbar, Schraubanschluß

CE-Konformität

EN 61000-4-2 / IEC 801-2 Elektrostatische Entladung ESD

Kontaktentladung 8 kV / Luftentladung 8 kV

EN 61000-4-5 / IEC 801-5 Surge-Prüfung

Versorgungspg. AC 4 kV, DC 0,5 kV

Signalleitungen 2 kV

EN 61000-4-4 / IEC 801-4 Burst-Prüfung

Eingänge - Ausgänge +/- 2 kV

Versorgungsspannung AC / DC +/- 2 kV

Die Analoggeberkarte BAH 4001 kommt dort zum Einsatz, wo Analogausgänge von Reglern oder DDC-Systemen ggf. durch Handeingriff übersteuert werden müssen. Typischer Einsatz in der Heizungs-, Lüftungs- und Klimatechnik sind z.B. Heizventile, Klappen und Frequenzumrichter.

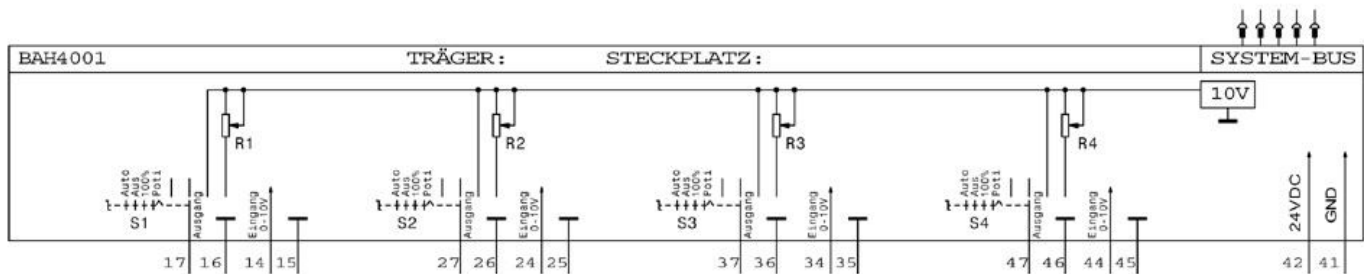
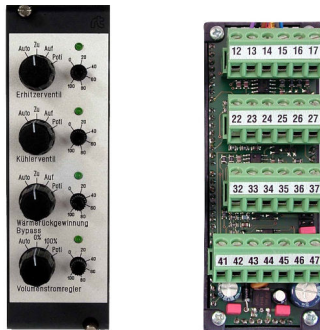
Das Analoggebermodul BAH 4001 besitzt 4 Kanäle, wovon jeder mit 10 mA belastbar ist. Weiterhin enthält die Karte einen ebenfalls mit 10 mA belastbaren 10 Volt-Festspannungsausgang. Jeder Kanal besitzt einen Schalter, mit dem das Automatiksignal übersteuert werden kann (Auto-0V-10V-Poti), ein Poti zur stufenlosen Einstellung der Ausgangsspannung von Hand sowie eine LED in Hell-/Dunkelschaltung zur optischen Kontrolle der Ausgangsspannung.

Der Sollwert der Ausgangsspannung im Automatikbetrieb wird über den Bus an die Zentral-Karte übergeben. Der Wertebereich der Kanäle 1 - 4 erstreckt sich von 0 bis 255 (entspricht 0 bis 100%). Die Schalterstellungen (Auto oder 0V/10V/Poti) und die Werte der an den Analogeingangsklemmen anstehenden Spannungen 0-10V (sowohl in der Automatik- als auch in den Örtlich-Betriebsarten) werden über das MOD-Bus / S-Bus Netzwerk übergeben.

Die Spannungsversorgung (24 V DC) ist beim Analoggebermodul BAH 4001 über Klemmen (41, 42) zu verdrahten (siehe Anschlußplan).

Bezüglich der Anlagenkonfiguration (Adressierung, maximale Anzahl von Erweiterungsmodulen an einem BZK, Montage, Busverbindungen etc.) sind die allgemeinen Hinweise im Kapitel Konfiguration zu beachten.

Modul ist in Kombination mit dem BZK1000MOD (Modbus RTU) und SBZK1000 (S-Bus) verwendbar.



Versorgungsspannung 24V DC,  $\pm 10\%$

Stromaufnahme max. 220 mA

Analogausgänge: 0-10V

Belastbarkeit je Analogausgang: 10 mA

Analogeingänge: 0-10V

Umgebungsbedingungen

Betriebstemperatur 0...50°C

Transport- und Lagertemperatur -20...+70°C

Relative Feuchte 5...95%, nicht kondensierend

Schutzart: IP 40, durch Abdeckung mit Sichtfenster bis IP 54

Abmessungen: 19"-Geräte

Breite 8 TE (40.5 mm), Höhe 3 HE (129 mm), Einbautiefe < 80 mm

Anschlußklemmen: 2,5mm<sup>2</sup> steckbar, Schraubanschluß

CE-Konformität

EN 61000-4-2 / IEC 801-2 Elektrostatische Entladung ESD

Kontaktentladung 8 kV / Luftentladung 8 kV

EN 61000-4-5 / IEC 801-5 Surge-Prüfung

Versorgungspg. AC 4 kV, DC 0,5 kV

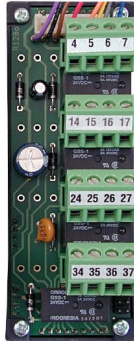
Signalleitungen 2 kV

EN 61000-4-4 / IEC 801-4 Burst-Prüfung

Eingänge - Ausgänge +/- 2 kV

Versorgungsspannung AC / DC +/- 2 kV

Die Taster-Lampen-Karte BDH4800 enthält 4 entprellte Taster, denen je drei LEDs zugeordnet sind:  
LED 1: Gelb, dauerleuchtend, keine Auslösung von Sammelstörung  
LED 2: Rot, blinkend, keine Auslösung von Sammelstörung  
LED 3: Rot, dauerleuchtend, Auslösung von Sammelstörung auf Zentralmodul  
(LEDs 4 - 6, 7 - 9, 10 - 12 entsprechend)



Die Information zur Ansteuerung der LEDs 1 - 12 wird in den Bytes 2 - 9 der Netzwerk-Eingangs-Variablen (nvi) an die Steuerkarte übertragen.

Die Information zur Relais-Ansteuerung wird in den Bytes 10 bis 13 der Netzwerk-Eingangs-Variablen nvi (SNVT\_str\_int) an das BDH 4800 übergeben.

Gültige Werte hierfür sind:

00 Aus

01 Ein

Eine Betätigung der Taster 1 - 4 wird über die Netzwerk-Ausgangs-Variable (nvo) an den LON-Knoten übergeben.

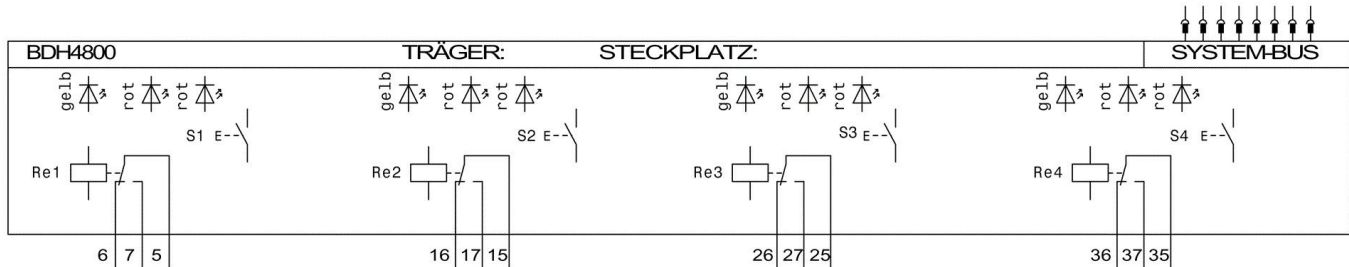
Den Tastern 1 - 4 entsprechen die nvo-Bytes 2 - 5, wobei gilt:

00 Taster gedrückt

01 Taster nicht betätigt

Bezüglich der Anlagenkonfiguration (Moduladressierung etc.) sind die allgemeinen Hinweise über das romutec-LON-Modulsystem zu beachten.

Das Modul ist auch in Kombination mit dem BZK1000MOD (Modbus RTU) verwendbar.



Versorgungsspannung 24V DC,  $\pm 10\%$

Stromaufnahme max. 155 mA

Relaisdaten:

Schaltspannung, max.: 250 VAC / 30 VDC

Schaltstrom, max. (Resistiv): 5 A

Schaltleistung: 625 VA / 150 W

Nennlast (Resistiv): 2,5 A / 250 VAC; 5 A / 30 VDC

Lebensdauer

bei Nennlast 50.000 Schaltspiele

ohne Last 5.000.000 Schaltspiele

Umgebungsbedingungen

Betriebstemperatur 0...50°C

Transport- und Lagertemperatur -20...+70°C

Relative Feuchte 5...95%, nicht kondensierend

Schutzart: IP 40, durch Abdeckung mit Sichtfenster bis IP 54

Abmessungen: 19"-Geräte

Breite 8 TE (40.5 mm), Höhe 3 HE (129 mm), Einbautiefe < 80 mm

Anschlußklemmen: 2,5mm<sup>2</sup> steckbar, Schraubanschluß

CE-Konformität

EN 61000-4-2 / IEC 801-2 Elektrostatische Entladung ESD

Kontaktentladung 8 kV / Luftentladung 8 kV

EN 61000-4-5 / IEC 801-5 Surge-Prüfung

Versorgungspg. AC 4 kV, DC 0,5 kV

Signalleitungen 2 kV

EN 61000-4-4 / IEC 801-4 Burst-Prüfung

Eingänge - Ausgänge +/- 2 kV

Versorgungsspannung AC / DC +/- 2 kV



Die Taster-Lampen-Karte BDH4800-100 enthält 4 entprellte Taster, denen je drei LEDs zugeordnet sind:

LED 1: Gelb, dauerleuchtend, keine Auslösung von Sammelstörung

LED 2: Rot, blinkend, keine Auslösung von Sammelstörung

LED 3: Rot, dauerleuchtend, Auslösung von Sammelstörung auf Zentralmodul (LEDs 4 - 6, 7 - 9, 10 - 12 entsprechend)

Die Information zur Ansteuerung der LEDs 1 - 12 wird in den Bytes 2 - 9 der Netzwerk-Eingangs-Variablen (nvi) an die Steuerkarte übertragen.

Eine Betätigung der Taster 1 - 4 wird über die Netzwerk-Ausgangs-Variable (nvo) an den LON-Knoten übergeben.

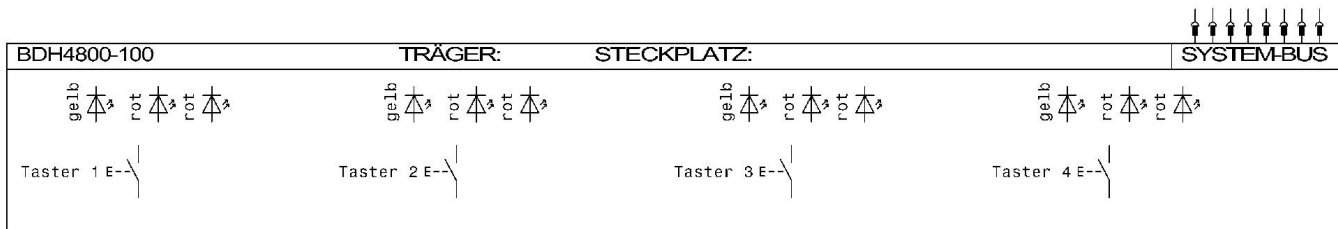
Den Tastern 1 - 4 entsprechen die nvo-Bytes 2 - 5, wobei gilt:

00 Taster gedrückt

01 Taster nicht betätigt

Bezüglich der Anlagenkonfiguration (Moduladressierung etc.) sind die allgemeinen Hinweise über das romutec-LON-Modulsystem zu beachten.

Modul ist auch in Kombination mit dem BZK1000MOD (Modbus RTU) verwendbar.



Versorgungsspannung 24V DC, ±10%

Stromaufnahme max. 155 mA

Umgebungsbedingungen

Betriebstemperatur 0...50°C

Transport- und Lagertemperatur -20...+70°C

Relative Feuchte 5...95%, nicht kondensierend

Schutzart: IP 40, durch Abdeckung mit Sichtfenster bis IP 54

Abmessungen: 19"-Geräte

Breite 8 TE (40.5 mm), Höhe 3 HE (129 mm), Einbautiefe < 80 mm

CE-Konformität

EN 61000-4-2 / IEC 801-2 Elektrostatische Entladung ESD

Kontaktentladung 8 kV / Luftentladung 8 kV

EN 61000-4-5 / IEC 801-5 Surge-Prüfung

Versorgungspg. AC 4 kV, DC 0,5 kV

Signalleitungen 2 kV

EN 61000-4-4 / IEC 801-4 Burst-Prüfung

Eingänge - Ausgänge +/- 2 kV

Versorgungsspannung AC / DC +/- 2 kV



Die Taster-Lampen-Karte BDH4800-101 enthält 4 entprellte Taster, denen je drei LEDs zugeordnet sind:

LED 1: Gelb, dauerleuchtend, keine Auslösung von Sammelstörung

LED 2: Rot, blinkend, keine Auslösung von Sammelstörung

LED 3: Grün, dauerleuchtend,  
(LEDs 4 - 6, 7 - 9, 10 - 12 entsprechend)

Die Information zur Ansteuerung der LEDs 1 - 12 wird in den Bytes 2 - 9 der Netzwerk-Eingangs-Variablen (nvi) an die Steuerkarte übertragen.

Eine Betätigung der Taster 1 - 4 wird über die Netzwerk-Ausgangs-Variable (nvo) an den LON-Knoten übergeben.

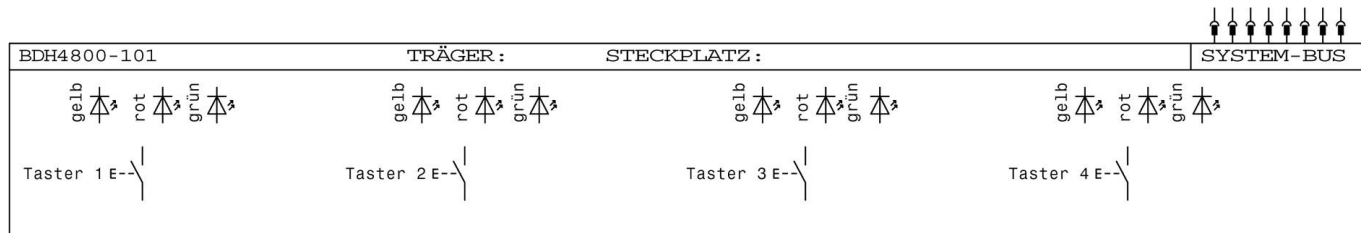
Den Tastern 1 - 4 entsprechen die nvo-Bytes 2 - 5, wobei gilt:

00 Taster gedrückt

01 Taster nicht betätigt

Bezüglich der Anlagenkonfiguration (Moduladressierung etc.) sind die allgemeinen Hinweise über das romutec-LON Modulsystem zu beachten.

Modul ist auch in Kombination mit dem BZK1000MOD (Modbus RTU) verwendbar.



Versorgungsspannung: 24V DC, ±10%

Stromaufnahme: max. 155mA

Umgebungsbedingungen

Betriebstemperatur 0...50°C

Transport- und Lagertemperatur -20...+70°C

Relative Feuchte 5...95%, nicht kondensierend

Schutzart: IP 40, durch Abdeckung mit Sichtfenster bis IP 54

Abmessungen

19"-Geräte Breite 8 TE (40.5 mm), Höhe 3 HE (129 mm),

Einbautiefe < 80 mm

Montage: Im 19-Zoll-Rahmen

CE-Konformität

EN 61000-4-2 / IEC 801-2 Elektrostatische Entladung ESD

Kontaktentladung 8 kV / Luftentladung 8 kV

EN 61000-4-5 / IEC 801-5 Surge-Prüfung

Versorgungspg. AC 4 kV, DC 0,5 kV

Signalleitungen 2 kV

EN 61000-4-4 / IEC 801-4 Burst-Prüfung

Eingänge - Ausgänge +/- 2 kV

Versorgungsspannung AC / DC +/- 2 kV

Die Taster-Lampen-Karte BDH4800-102 enthält 4 entprellte Taster, denen je zwei LEDs zugeordnet sind:

LED 1: Gelb, dauerleuchtend, keine Auslösung von Sammelstörung

LED 2: Rot, dauerleuchtend, keine Auslösung von Sammelstörung  
(LEDs 3 - 8 entsprechend)



Die Information zur Ansteuerung der LEDs 1 - 8 wird in den Bytes 2 - 9 der Netzwerk-Eingangs-Variablen (nvi) an die Steuerkarte übertragen.

Eine Betätigung der Taster 1 - 4 wird über die Netzwerk-Ausgangs-Variable (nvo) an den LON-Knoten übergeben.

Den Tastern 1 - 4 entsprechen die nvo-Bytes 2 - 5, wobei gilt:

00 Taster gedrückt

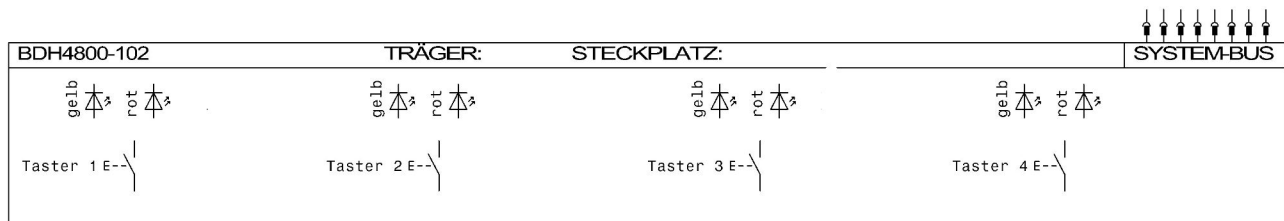
01 Taster nicht betätigt

Bezüglich der Anlagenkonfiguration (Moduladressierung etc.) sind die allgemeinen Hinweise über das romutec-LONModulsystem zu beachten.

Modul ist auch in Kombination mit dem BZK1000MOD (Modbus RTU) verwendbar.

Wichtige technische Daten:

Spannungsversorgung: +24 V DC über den 8-poligen Systembus



Versorgungsspannung: 24V DC, ±10%

Stromaufnahme: max. 155mA

Umgebungsbedingungen

Betriebstemperatur 0...50°C

Transport- und Lagertemperatur -20...+70°C

Relative Feuchte 5...95%, nicht kondensierend

Schutzart: IP 40, durch Abdeckung mit Sichtfenster bis IP 54

Abmessungen

19"-Geräte Breite 8 TE (40.5 mm), Höhe 3 HE (129 mm),

Einbautiefe < 80 mm

Montage: Im 19-Zoll-Rahmen

CE-Konformität

EN 61000-4-2 / IEC 801-2 Elektrostatische Entladung ESD

Kontaktentladung 8 kV / Luftentladung 8 kV

EN 61000-4-5 / IEC 801-5 Surge-Prüfung

Versorgungspg. AC 4 kV, DC 0,5 kV

Signalleitungen 2 kV

EN 61000-4-4 / IEC 801-4 Burst-Prüfung

Eingänge - Ausgänge +/- 2 kV

Versorgungsspannung AC / DC +/- 2 kV