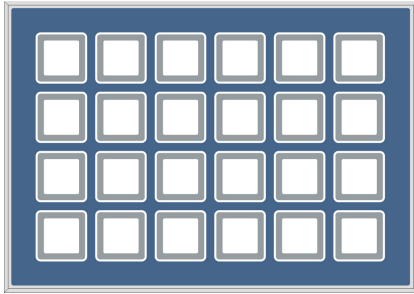


## Beschreibung:



- **Bedienfeldbaugruppe mit beleuchteten Tastern**
  - **24 Taster mit Druckpunkt ohne Beschriftung, über Einschubstreifen individuell beschriftbar**
  - **Hintergrundbeleuchtung der Taster einzeln schaltbar**
- Optionale Erweiterung BFB.790.31:**
- **4 Transistorausgänge 24 VDC davon 1 Ausgang konfigurierbar als Relais (Wechsler) und 1 Ausgang konfigurierbar als Relais (Schließer)**
  - **4 universelle Eingänge Digital 24 VDC oder Analog 0...10 V konfigurierbar**

## BFB.790.30/31 D1

Bedienfeldbaugruppe mit 24 beleuchteten Druckpunktastern als CAN-Slave oder Modbus RTU-Slave.

Je 4 Taster ein Einschub (senkrecht) für Beschriftungsstreifen zur individuellen Beschriftung. Die Taster werden einzeln durch eine Hintergrundbeleuchtung vollflächig ausgeleuchtet, Farbe: rot, grün, gelb.

### Optionale Erweiterung BFB.790.31:

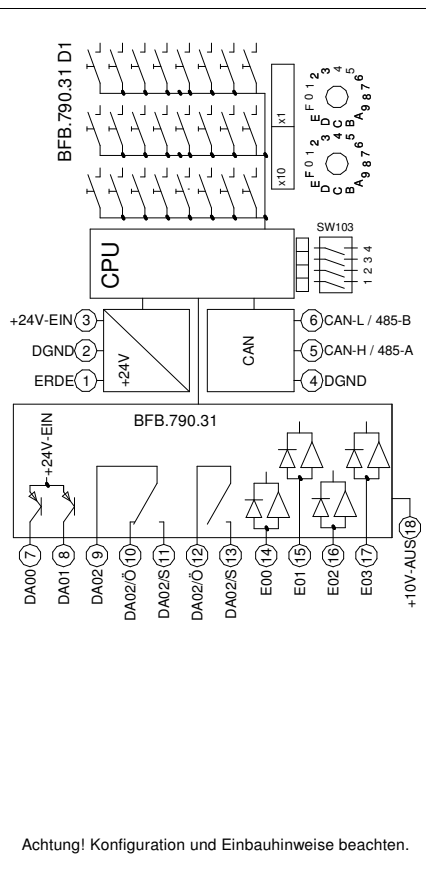
4 Digitalausgänge Transistor 24 VDC, davon 1 Ausgang als Relais (Wechsler) und 1 Ausgang als Relais (Schließer) konfigurierbar

4 universelle Eingänge Digital 24 VDC oder Analog 0...10 V konfigurierbar.

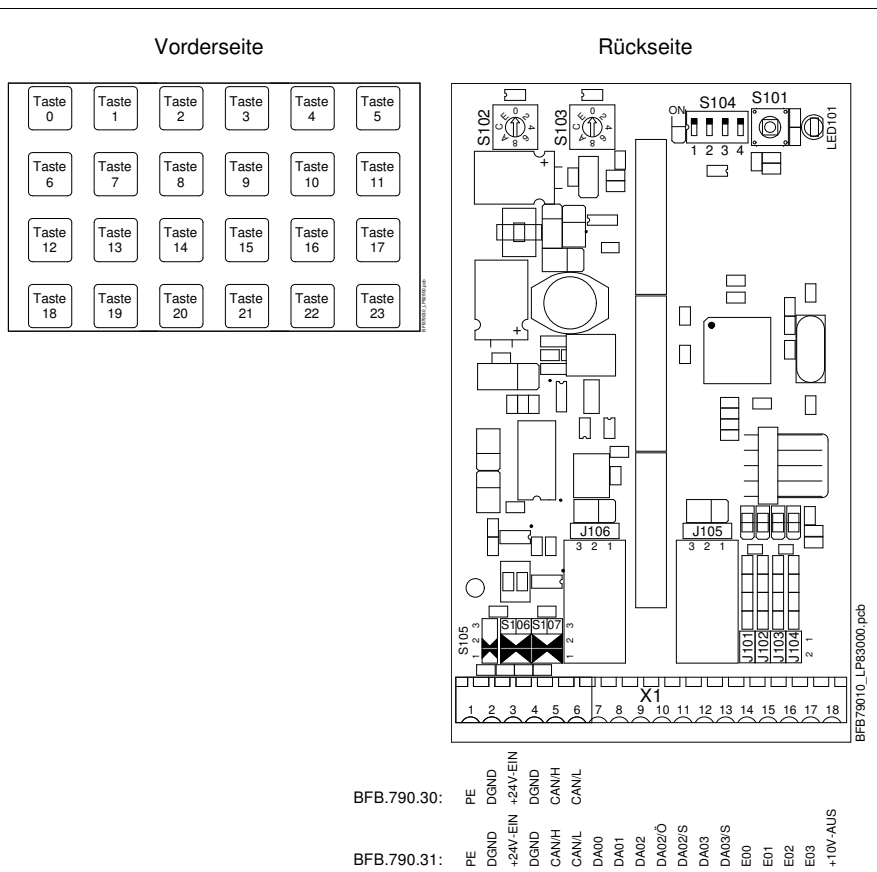
Ausführung der Baugruppe zur Fronttafelmontage mit Aluminiumplatte und Dekorfolie.

Schraubsteckklemmen und Abdeckhaube zur Montage auf der Rückseite

## Blockschema:



## Anschlussplan:



## Bestellbezeichnung:

Bedienfeldbaugruppe mit 24 beleuchteten Tastern  
 Bedienfeldbaugruppe mit 24 beleuchteten Tastern und Erweiterung Ein-/Ausgänge

## Artikel-Nr.:

BFB.790.30  
 BFB.790.31

## Konfiguration:

**BFB.790.30/31 D1**

### Konfiguration Schnittstellen

Terminierung CAN-Bus	S105		EIN Terminierung mit 120 Ohm AUS keine Terminierung
Terminierung RS485/Modbus	S106		EIN Akt. Terminierung mit 150 Ohm AUS keine Terminierung
Umschaltung Klemmen X1:5 und X1:6 CAN-Bus oder RS485/Modbus	S107		Durch Schalter S107 können die Klemmen X2:5 und X2:6 entweder mit CAN-Bus oder RS485/Modbus belegt werden. Gemeinsamer Betrieb ist nicht möglich!

### Konfiguration Digitalausgänge (nur BFB.790.31)

Digitalausgang DA02	J105		X1:9 nicht verwendet X1:10 nicht verwendet X1:11 Transistorausg. (High-schaltend)
	J105		X1:9 Relais COM X1:10 Relais (Öffner) X1:11 Relais (Schließer)
Digitalausgang DA03	J106		X1:12 nicht verwendet X1:13 Transistorausg. (High-schaltend)
	J106		X1:12 Relais COM X1:13 Relais (Schließer)

### Konfiguration universelle Eingänge (nur BFB.790.31)

Universeller Eingang E00	J101		offen: Digitaleingang 0/24 VDC gesteckt: Analogeingang 0...10 V
Universeller Eingang E01	J102		offen: Digitaleingang 0/24 VDC gesteckt: Analogeingang 0...10 V
Universeller Eingang E02	J103		offen: Digitaleingang 0/24 VDC gesteckt: Analogeingang 0...10 V
Universeller Eingang E03	J104		offen: Digitaleingang 0/24 VDC gesteckt: Analogeingang 0...10 V

## Konfiguration CAN-Bus:

### S107 in Stellung "CAN" (s.o.)

#### Adressanwahl CAN-Bus über S102 u. S103

S102 ⇒ Adresse High (hex)  
 S103 ⇒ Adresse Low (hex)

Stellung 00, 01 ⇒ nicht erlaubt  
 02...7F ⇒ gültige CAN-ID 2...127  
 80...FF ⇒ nicht erlaubt

### Einstellung Baudrate über S104

S104:1	S104:2	S104:3	Baudrate
OFF	OFF	OFF	(n.v.)
ON	OFF	OFF	20 kBaud
OFF	ON	OFF	50 kBaud
ON	ON	OFF	100 kBaud
OFF	OFF	ON	125 kBaud
ON	OFF	ON	250 kBaud
OFF	ON	ON	500 kBaud
ON	ON	ON	1000 kBaud

## Konfiguration Modbus RTU:

### S107 in Stellung "RS485" (s.o.)

#### Adressanwahl Modbus RTU über S102 u. S103

S102 ⇒ Adresse High (hex)  
 S103 ⇒ Adresse Low (hex)

Stellung 00 ⇒ nicht erlaubt  
 01...F7 ⇒ gültige Slave-ID 1...247  
 F8...FF ⇒ nicht erlaubt

### Einstellung Protokoll und Baudrate über S104

S104:1 = OFF ⇒ Protokoll 8 – E – 1  
 S104:1 = ON ⇒ Protokoll 8 – N – 1

S104:2	S104:3	Baudrate
OFF	OFF	1200 Baud
ON	OFF	9600 Baud
OFF	ON	19200 Baud
ON	ON	57600 Baud

## Testmodus:

**BFB.790.30/31 D1**

Über **S104:4** = ON ist ein Testmodus anwählbar, bei dem die Taster und Leuchten folgende Funktion haben:

- Taster 0, 6, 12, 18 (linke Spalte): Alle Leuchten aus / alle rot / alle grün / alle gelb
- Alle anderen Taster: Bei Drücken leuchtet die zugehörige Leuchte gelb

## Anzeigen / Servicetaster S101:

### Status LED rot

blinkend  
dauer

Ungültige CAN-Adresse eingestellt  
Firmware-Bootloader ist aktiv

### Status LED gelb

blinkend

Modul wartet auf Initialisierung durch PLM-Master

### Status LED grün

langsam blinkend  
schnell blinkend  
dauer

Modul betriebsbereit, aber noch nicht vom PLM-Master gestartet  
Modul betriebsbereit, aber Kontakt zum PLM-Master verloren oder vom PLM-Master gestoppt  
Modul betriebsbereit und gestartet

### Servicetaster S101

Beim Einschalten gedrückt  
Drücken im Betrieb

Der Firmware-Bootloader wird gestartet (nur in Verbindung mit Konfigurations-Software nutzbar)  
Keine Funktion

## Technische Daten:

### Anschlussdaten

Spannungsversorgung 24 VDC  $\pm$  10%,  
Stromaufnahme max. 450 mA

### Ausgänge

4 Transistorausgänge, 24 VDC  $\pm$  15 %,  
davon 1 Ausgang konfigurierbar als Relais  
(Wechsler) 24 VDC / 1 A und 1 Ausgang  
konfigurierbar als Relais (Schließer)  
24 VDC / 1 A

### Eingänge

4 universelle Eingänge,  
einzeln konfigurierbar als Digitaleingang  
24 VDC oder Analogeingang 0...10 V

### Kommunikationsschnittstelle

Umschaltbar zwischen CAN-Bus und  
RS485/Modbus RTU

### Mechanische Daten

Baugruppe mit Abdeckhaube  
zur Montage auf Terminal-Rückseite  
Anschlüsse über Schraubsteckklemmen  
Maße BxHxT: 203 x 143 x 32 mm  
Gewicht ca. 440 g

### Schutzart

IP 54 frontseitig / IP 20 rückseitig  
nach IEC 529

### Klimatische Bedingungen

Lagertemperatur -10...+70 °C  
Umgebungstemperatur +5...+40 °C  
Luftfeuchtigkeit bis 85 % ohne Betauung  
nach VDE 0160, EN 50178, Klasse 3K3

### Installationshinweise

Es sind die gesonderten Hinweise zum  
EMV-gerechten Einbau der Hardware im  
Systemhandbuch der SABO Elektronik  
GmbH zu beachten!

Downloadmöglichkeit unter [www.sabo.de](http://www.sabo.de)

## Installationshinweise:

### Konfiguration

Achtung! Beachten Sie vor dem Einbau des Moduls die interne  
Konfiguration, den Software-Stand und die Einbauhinweise.

### Spannungsversorgung

Nach dem Anreihen von 10 Modulen ist die Spannungsversorgung  
neu anzulegen

### Aufbau

Das Feldbusmodul darf nicht unter Spannung gesteckt werden, da  
sonst Schäden am Modul bzw. Datenverlust möglich sind.

### CAN-Bus Terminierung

Bei Standardterminierung sollten das Mikroprozessormodul und das  
letzte Feldbusmodul terminiert werden. Maximal 2 Terminierungen  
sind zulässig

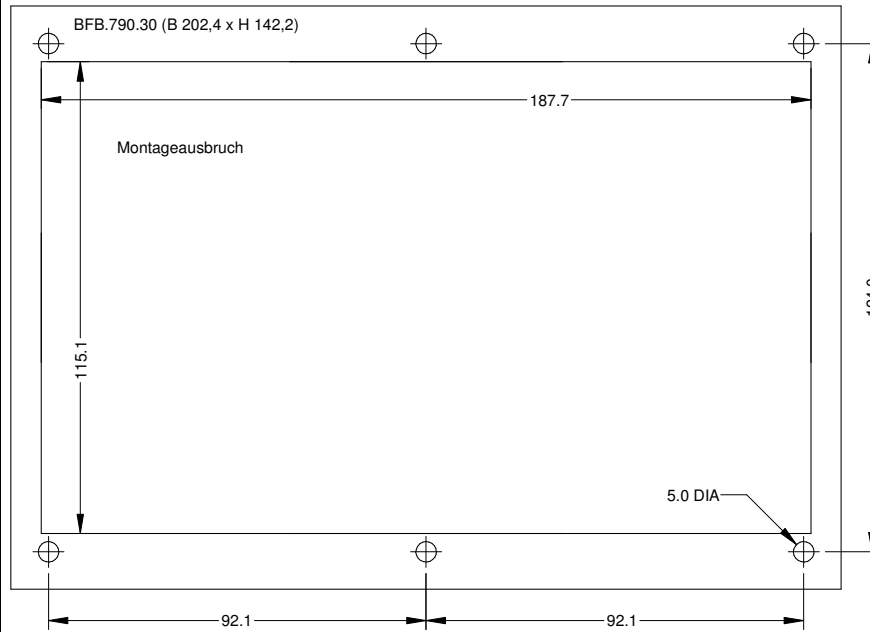
### Installationshinweise

Es sind die gesonderten Produkthinweise im Systemhandbuch der  
SABO Elektronik GmbH zu beachten.

Downloadmöglichkeit unter [www.sabo.de](http://www.sabo.de)

## Ausschnittmaße für Fronttafeleinbau:

## BFB.790.30/31 D1



**Bei Schaltschrankfronttafelmontage sind folgende Maße einzuhalten:**

Stanzmaß 115,1 mm x 187,7 mm

6 Bohrungen Ø 5 mm

für 6 Gewindebolzen M 4

## Programmierhinweise:

## BFB.790.30/31 D1

### Gerätebeschreibungsdatei

Zur Verwendung des Moduls muss in CoDeSys die aktuelle Gerätebeschreibungsdatei (EDS-Datei) *BFB.790.30\_v2.EDS* bzw. *BFB.790.31\_v2.EDS* verwendet werden (Downloadmöglichkeit unter [www.sabo.de](http://www.sabo.de)). Anschließend wird das Modul unter *Ressourcen* → *Steuerungskonfiguration* eingefügt. Beispiel für BFB.790.31:

```

Can 0 Master[VAR]
  BFB.790.31_v2 (EDS) [VAR]
    %QB1.0 Can-Output
      AT %QB1.0.0: USINT; (* LED 0...7 state [COBID=0x202] *)
      AT %QB1.0.1: USINT; (* LED 8...15 state [COBID=0x202] *)
      AT %QB1.0.2: USINT; (* LED 16...23 state [COBID=0x202] *)
      AT %QB1.0.3: USINT; (* Digital out 0...3, Lampstest [COBID=0x202] *)
      AT %QB1.0.4: USINT; (* LED 0 color (0=default, 1=red, 2=green, 3=orange, 4=blink) [COBID=0x302] *)
      AT %QB1.0.5: USINT; (* LED 8 color (0=default, 1=red, 2=green, 3=orange, 4=blink) [COBID=0x302] *)
      AT %QB1.0.6: USINT; (* LED 2 color (0=default, 1=red, 2=green, 3=orange, 4=blink) [COBID=0x302] *)
      AT %QB1.0.7: USINT; (* LED 3 color (0=default, 1=red, 2=green, 3=orange, 4=blink) [COBID=0x302] *)
      AT %QB1.0.8: USINT; (* LED 4 color (0=default, 1=red, 2=green, 3=orange, 4=blink) [COBID=0x302] *)
      AT %QB1.0.9: USINT; (* LED 5 color (0=default, 1=red, 2=green, 3=orange, 4=blink) [COBID=0x302] *)
      AT %QB1.0.10: USINT; (* LED 6 color (0=default, 1=red, 2=green, 3=orange, 4=blink) [COBID=0x302] *)
      AT %QB1.0.11: USINT; (* LED 7 color (0=default, 1=red, 2=green, 3=orange, 4=blink) [COBID=0x302] *)
      AT %QB1.0.12: USINT; (* LED 8 color (0=default, 1=red, 2=green, 3=orange, 4=blink) [COBID=0x402] *)
      AT %QB1.0.13: USINT; (* LED 9 color (0=default, 1=red, 2=green, 3=orange, 4=blink) [COBID=0x402] *)
      AT %QB1.0.14: USINT; (* LED 10 color (0=default, 1=red, 2=green, 3=orange, 4=blink) [COBID=0x402] *)
      AT %QB1.0.15: USINT; (* LED 11 color (0=default, 1=red, 2=green, 3=orange, 4=blink) [COBID=0x402] *)
      AT %QB1.0.16: USINT; (* LED 12 color (0=default, 1=red, 2=green, 3=orange, 4=blink) [COBID=0x402] *)
      AT %QB1.0.17: USINT; (* LED 13 color (0=default, 1=red, 2=green, 3=orange, 4=blink) [COBID=0x402] *)
      AT %QB1.0.18: USINT; (* LED 14 color (0=default, 1=red, 2=green, 3=orange, 4=blink) [COBID=0x402] *)
      AT %QB1.0.19: USINT; (* LED 15 color (0=default, 1=red, 2=green, 3=orange, 4=blink) [COBID=0x402] *)
      AT %QB1.0.20: USINT; (* LED 16 color (0=default, 1=red, 2=green, 3=orange, 4=blink) [COBID=0x502] *)
      AT %QB1.0.21: USINT; (* LED 17 color (0=default, 1=red, 2=green, 3=orange, 4=blink) [COBID=0x502] *)
      AT %QB1.0.22: USINT; (* LED 18 color (0=default, 1=red, 2=green, 3=orange, 4=blink) [COBID=0x502] *)
      AT %QB1.0.23: USINT; (* LED 19 color (0=default, 1=red, 2=green, 3=orange, 4=blink) [COBID=0x502] *)
      AT %QB1.0.24: USINT; (* LED 20 color (0=default, 1=red, 2=green, 3=orange, 4=blink) [COBID=0x502] *)
      AT %QB1.0.25: USINT; (* LED 21 color (0=default, 1=red, 2=green, 3=orange, 4=blink) [COBID=0x502] *)
      AT %QB1.0.26: USINT; (* LED 22 color (0=default, 1=red, 2=green, 3=orange, 4=blink) [COBID=0x502] *)
      AT %QB1.0.27: USINT; (* LED 23 color (0=default, 1=red, 2=green, 3=orange, 4=blink) [COBID=0x502] *)
    %IB1.0 Can-Input
      AT %IB1.0.0: USINT; (* Button 0...7 state [COBID=0x182] *)
      AT %IB1.0.1: USINT; (* Button 8...15 state [COBID=0x182] *)
      AT %IB1.0.2: USINT; (* Button 16...23 state [COBID=0x182] *)
      AT %IB1.0.3: USINT; (* Input 0...3 (digital) [COBID=0x182] *)
      AT %IB1.0.4: USINT; (* Input 0 (analog) [COBID=0x282] *)
      AT %IB1.0.5: USINT; (* Input 1 (analog) [COBID=0x282] *)
      AT %IB1.0.6: USINT; (* Input 2 (analog) [COBID=0x282] *)
      AT %IB1.0.7: USINT; (* Input 3 (analog) [COBID=0x282] *)
  
```

### Digitaleingänge

Die Bits in *Button1*, *Button2* und *Button3* sind wie folgt belegt:

Button1 Bit							
7	6	5	4	3	2	1	0
T7	T6	T5	T4	T3	T2	T1	T0

T0...T7 ⇒ Zustand der Taster 0...7

Button2 Bit							
7	6	5	4	3	2	1	0
T15	T14	T13	T12	T11	T10	T9	T8

T8...T15 ⇒ Zustand der Taster 8...15

Button3 Bit							
7	6	5	4	3	2	1	0
T23	T22	T21	T20	T19	T18	T17	T16

T16...T23 ⇒ Zustand der Taster 16...23

Die Bits in *DigIn* sind wie folgt belegt (nur BFB.790.31):

Bit							
7	6	5	4	3	2	1	0
-	-	-	-	E03	E02	E01	E00

E00...E03 ⇒ Zustand der als Digitaleingang konfigurierten Universaleingänge

### Digitalausgänge

Die Bits in *LedState1*, *LedState2* und *LedState3* sind wie folgt belegt:

LedState1 Bit							
7	6	5	4	3	2	1	0
L7	L6	L5	L4	L3	L2	L1	L0

L0...L7 ⇒ Lampen der Taster 0...7

LedState2 Bit							
7	6	5	4	3	2	1	0
L15	L14	L13	L12	L11	L10	L9	L8

L8...L15 ⇒ Lampen der Taster 8...15

LedState3 Bit							
7	6	5	4	3	2	1	0
L23	L22	L21	L20	L19	L18	L17	L16

L16...L23 ⇒ Lampen der Taster 16...23

Die Bits in *DigOut* sind wie folgt belegt:

Bit							
7	6	5	4	3	2	1	0
LT	-	-	-	DA03	DA02	DA01	DA00

DA00...DA03 ⇒ Digitalausgänge (nur BFB.790.31)

LT ⇒ Lampentest, alle LEDs an

### Ausgänge LED Color

Die Leuchtfarbe der Tasten wird zunächst vom Wert des Parameters *0x2500 LED default color* bestimmt. Durch Beschreiben der Ausgänge *LED color* kann die Farbe pro Taste einzeln festgelegt werden. Folgende Werte sind möglich:

- 0 ⇒ Default-Farbe (*LED default color*)
- 1 ⇒ rot
- 2 ⇒ grün
- 3 ⇒ gelb
- + 4 ⇒ wie 1, 2 oder 3, aber Beleuchtung blinkt

### Universelle Eingänge E00...E03 (nur BFB.790.31)

Die universellen Eingänge E00...E03 können durch Jumper-Konfiguration (s.o.) als Analog- oder Digitaleingänge genutzt werden. Eine Software-Konfiguration ist nicht erforderlich.

Bei Verwendung als Analogeingang erscheint der Wandlerwert in den Variablen *AnaIn0...AnaIn3*. Der A/D-Wandler hat eine Auflösung von 10 Bit, die Wandlerwerte bewegen sich daher im Bereich 0...1023 und werden als vorzeichenlose 16-Bit-Zahl vom Typ WORD übertragen.

Bei Verwendung als Digitaleingang entsprechen die Eingänge E00...E03 den Bits 0...3 in der Variablen *DigIn*.

### Verwendung im IEC-Programm

Die Verwendung im IEC-Programm erfolgt zweckmäßigerweise durch Anlegen von Globalen Variablen, die mit einer AT-Deklaration an die Adresse des jeweiligen Übertragungswerts gebunden werden.

Beispiel BFB.790.31:

```

VAR_GLOBAL
  (* Output *)
  LedState1 AT %QB1.0.0: BYTE;
  LedState2 AT %QB1.0.1: BYTE;
  LedState3 AT %QB1.0.2: BYTE;
  DigOut AT %QB1.0.3: BYTE;
  (* Input *)
  Button1 AT %IB1.0.0: BYTE;
  Button2 AT %IB1.0.1: BYTE;
  Button3 AT %IB1.0.2: BYTE;
  DigIn AT %IB1.0.3: BYTE;
  AnaIn0 AT %IB1.0.4: WORD;
  AnaIn1 AT %IB1.0.5: WORD;
  AnaIn2 AT %IB1.0.6: WORD;
  AnaIn3 AT %IB1.0.7: WORD;
END_VAR
  
```

## Programmierhinweise:

**BFB.790.30/31 D1**

### Einstellen der Modulparameter

In CoDeSys: *Ressourcen* → *Steuerungskonfiguration* → *BFB.790.31\_v2* → *Service Data Objects* (s.u.). Die in der Spalte *Wert* eingetragenen Parameter werden beim Start der Steuerung an das Modul übertragen (Initialisierung).

Index	Name	Wert	Typ	Default
2503sub1	Button 0 mode (0=button, 1=toggle, +2=LED shows button state)	0	Unsigned8	0
2503sub2	Button 1 mode (0=button, 1=toggle, +2=LED shows button state)	0	Unsigned8	0
2503sub3	Button 2 mode (0=button, 1=toggle, +2=LED shows button state)	0	Unsigned8	0
2503sub4	Button 3 mode (0=button, 1=toggle, +2=LED shows button state)	0	Unsigned8	0
2503sub5	Button 4 mode (0=button, 1=toggle, +2=LED shows button state)	0	Unsigned8	0
2503sub6	Button 5 mode (0=button, 1=toggle, +2=LED shows button state)	0	Unsigned8	0
2503sub7	Button 6 mode (0=button, 1=toggle, +2=LED shows button state)	0	Unsigned8	0
2503sub8	Button 7 mode (0=button, 1=toggle, +2=LED shows button state)	0	Unsigned8	0
2503sub9	Button 8 mode (0=button, 1=toggle, +2=LED shows button state)	0	Unsigned8	0
2503suba	Button 9 mode (0=button, 1=toggle, +2=LED shows button state)	0	Unsigned8	0
2503subb	Button 10 mode (0=button, 1=toggle, +2=LED shows button state)	0	Unsigned8	0
2503subc	Button 11 mode (0=button, 1=toggle, +2=LED shows button state)	0	Unsigned8	0
2503subd	Button 12 mode (0=button, 1=toggle, +2=LED shows button state)	0	Unsigned8	0
2503sube	Button 13 mode (0=button, 1=toggle, +2=LED shows button state)	0	Unsigned8	0
2503subf	Button 14 mode (0=button, 1=toggle, +2=LED shows button state)	0	Unsigned8	0
2503sub10	Button 15 mode (0=button, 1=toggle, +2=LED shows button state)	0	Unsigned8	0
2503sub11	Button 16 mode (0=button, 1=toggle, +2=LED shows button state)	0	Unsigned8	0
2503sub12	Button 17 mode (0=button, 1=toggle, +2=LED shows button state)	0	Unsigned8	0
2503sub13	Button 18 mode (0=button, 1=toggle, +2=LED shows button state)	0	Unsigned8	0
2503sub14	Button 19 mode (0=button, 1=toggle, +2=LED shows button state)	0	Unsigned8	0
2503sub15	Button 20 mode (0=button, 1=toggle, +2=LED shows button state)	0	Unsigned8	0
2503sub16	Button 21 mode (0=button, 1=toggle, +2=LED shows button state)	0	Unsigned8	0
2503sub17	Button 22 mode (0=button, 1=toggle, +2=LED shows button state)	0	Unsigned8	0
2503sub18	Button 23 mode (0=button, 1=toggle, +2=LED shows button state)	0	Unsigned8	0
2504	Button debounce time (ms)	50	Unsigned16	50
2500	LED default color (1=red, 2=green, 3=yellow, +4=blink)	3	Unsigned8	3
2502	LED blink cycle period (ms)	500	Unsigned16	500
2506	LED brightness (0..100%)	100	Unsigned8	100
6426sub1	Input 0 (analog) send threshold	1	Unsigned16	1
6426sub2	Input 1 (analog) send threshold	1	Unsigned16	1
6426sub3	Input 2 (analog) send threshold	1	Unsigned16	1
6426sub4	Input 3 (analog) send threshold	1	Unsigned16	1
6206	Error Mode (0=keep state, 1=clear)	1	Unsigned8	1
2100	Periodic Datatransfer (ms, 0=off)	2000	Unsigned16	2000
2180	Send Inhibit Time (ms)	20	Unsigned16	20

### Parameter 0x6426 *Input (analog) send threshold*

Nur BFB.790.31: Unterdrückt das Übertragen neuer Messwerte, bis der angegebene Schwellwert überschritten wird. Dadurch wird die Belastung des CAN-Busses verringert. Angabe in Digits. 0 ⇒ Abschalten der Funktion.

### Parameter 0x6206 *Error Mode*

Nur BFB.790.31: Legt das Verhalten der LEDs und Digitalausgänge bei Stop oder Fehler wie folgt fest:

- 0 ⇒ letzter Wert wird beibehalten
- 1 ⇒ abschalten

### Parameter 0x2100 *Periodic Datatransfer*

Veranlasst das regelmäßige Übertragen der aktuellen Prozessdaten an die Steuerung, auch wenn keine Änderung der Digitaleingänge stattgefunden hat und bei Analogeingängen die bei *Input Send Threshold* eingestellte Schwelle nicht überschritten wurde. Angabe in ms, 0 ⇒ Abschalten der Funktion.

### Parameter 0x2180 *Send Inhibit Time*

Verhindert das Übertragen von neuen Prozessdaten an die Steuerung vor Ablauf der angegebenen Zeit. Angabe in ms, 0 ⇒ Abschalten der Funktion.

### Parameter 0x2503 *Button mode*

Legt die Betriebsart einer Taste fest:

- 0 ⇒ Taste sendet 1 solange gedrückt, sonst 0
- 1 ⇒ Taste schaltet bei Drücken abwechselnd zwischen 1 und 0 um
- 2 ⇒ wie 0, aber Beleuchtung entspricht Tastenzustand
- 3 ⇒ wie 1, aber Beleuchtung entspricht Tastenzustand

In den Betriebsarten 0 und 1 wird der Zustand der Tastenbeleuchtung durch die Variablen *LedState1*, *LedState2* und *LedState3* gesteuert.

### Parameter 0x2504 *Button debounce time*

Legt die Entprellzeit der Tasten fest. Angabe in ms.

### Parameter 0x2500 *LED default color*

Voreingestellte Farbe für die Tastenbeleuchtung. Folgende Werte sind möglich:

- 1 ⇒ rot
- 2 ⇒ grün
- 3 ⇒ gelb
- + 4 ⇒ wie 1, 2 oder 3, aber Beleuchtung blinkt

Die Farbe jeder Tastenbeleuchtung kann mit den Variablen *LED x color* einzeln eingestellt werden. Bei 0 wird der hier eingestellte Default-Wert verwendet.

### Parameter 0x2502 *LED blink cycle*

Blinkperiode bei aktiviertem Blinken der Tastenbeleuchtung. Angabe in ms.

### Parameter 0x2505 *LED brightness* (ab Firmware v2.14)

Helligkeit der Tastenbeleuchtung. Angabe in Prozent (0..100) vom Maximalwert.

## Modbus RTU Slave:

**BFB.790.30/31 D1**

### Function-Codes und Datentypen

Bei Verwendung des Modbus-Interfaces stehen alle Konfigurationsdaten sowie die Input- und Output-Werte des Moduls als 16-Bit-Register zur Verfügung.

Der Zugriff kann mit folgenden Function-Codes (FC) erfolgen:

FC 3	⇒	Read Holding Registers
FC 4	⇒	Read Input Registers
FC 6	⇒	Write Single Register
FC 16	⇒	Write Multiple Registers

Alle unten aufgeführten Adressen bezeichnen die im Modbus-Telegramm verwendeten Registerindizes.

Die Bedeutung der Register ist oben unter *Programmierhinweise* dokumentiert.

Zusätzlich stellt das Modbus-Interface fünf Spezialregister zur Verfügung, die der Überwachung des Modbus-Interfaces dienen (s.u.).

Folgende Datentypen werden verwendet (Datenübertragung erfolgt immer als 16-Bit-Register) :

Datentyp	Beschreibung	Wertebereich
UINT8	8 Bit Integer unsigned	0 ... 255
UINT16	16 Bit Integer unsigned	0 ... 65535
INT16	16 Bit Integer mit Vorz.	-32768 ... 32767

Als Zugriffstypen werden verwendet:

Zugriffstyp	Beschreibung	Function-Codes
RO	Read-Only	FC 3, 4
WO	Write-Only	FC 6, 10
RW	Read / Write	FC 3, 4, 6, 10

### Konfigurationsregister

Die Bedeutung der Register ist unter *Programmierhinweise* dokumentiert (s.o.).

Adr.	Datentyp	Zugriff	Name
100	UINT8	RW	Button Mode Taster 0
101	UINT8	RW	Button Mode Taster 1
102	UINT8	RW	Button Mode Taster 2
103	UINT8	RW	Button Mode Taster 3
104	UINT8	RW	Button Mode Taster 4
105	UINT8	RW	Button Mode Taster 5
106	UINT8	RW	Button Mode Taster 6
107	UINT8	RW	Button Mode Taster 7
108	UINT8	RW	Button Mode Taster 8
109	UINT8	RW	Button Mode Taster 9
110	UINT8	RW	Button Mode Taster 10
111	UINT8	RW	Button Mode Taster 11
112	UINT8	RW	Button Mode Taster 12
113	UINT8	RW	Button Mode Taster 13
114	UINT8	RW	Button Mode Taster 14
115	UINT8	RW	Button Mode Taster 15
116	UINT8	RW	Button Mode Taster 16
117	UINT8	RW	Button Mode Taster 17
118	UINT8	RW	Button Mode Taster 18
119	UINT8	RW	Button Mode Taster 19
120	UINT8	RW	Button Mode Taster 20
121	UINT8	RW	Button Mode Taster 21
122	UINT8	RW	Button Mode Taster 22
123	UINT8	RW	Button Mode Taster 23
124	UINT8	RW	LED Default Color
125	UINT8	RW	LED Brightness
126	UINT16	RW	Button Debounce Time (ms)
127	UINT16	RW	LED Blink Cycle (ms)

### Input-Register

Die Bedeutung der Register ist unter *Programmierhinweise* dokumentiert (s.o.).

BFB.790.30:

Adr.	Datentyp	Zugriff	Name
200	UINT8	RO	Button1, Zustand der Taster 0...7
201	UINT8	RO	Button2, Zustand der Taster 8...15
202	UINT8	RO	Button3, Zustand der Taster 16...23

BFB.790.31:

Adr.	Datentyp	Zugriff	Name
200	UINT8	RO	Button1, Zustand der Taster 0...7
201	UINT8	RO	Button2, Zustand der Taster 8...15
202	UINT8	RO	Button3, Zustand der Taster 16...23
203	UINT8	RO	DigIn, Zustand der Digitaleingänge
204	UINT16	RO	Analogwert E00
205	UINT16	RO	Analogwert E01
206	UINT16	RO	Analogwert E02
207	UINT16	RO	Analogwert E03

### Output-Register

Die Bedeutung der Register ist unter *Programmierhinweise* dokumentiert (s.o.).

Adr.	Datentyp	Zugriff	Name
300	UINT8	RW	LedState1, Ansteuerung der Lampen 0...7
301	UINT8	RW	LedState2, Ansteuerung der Lampen 8...15
302	UINT8	RW	LedState3, Ansteuerung der Lampen 16...23
303	UINT8	RW	DigOut, Ansteuerung der Digitalausgänge (nur BFB.790.31) und Lampentest
304	UINT8	RW	LED Color Taster 0
305	UINT8	RW	LED Color Taster 1
306	UINT8	RW	LED Color Taster 2
307	UINT8	RW	LED Color Taster 3
308	UINT8	RW	LED Color Taster 4
309	UINT8	RW	LED Color Taster 5
310	UINT8	RW	LED Color Taster 6
311	UINT8	RW	LED Color Taster 7
312	UINT8	RW	LED Color Taster 8
313	UINT8	RW	LED Color Taster 9
314	UINT8	RW	LED Color Taster 10
315	UINT8	RW	LED Color Taster 11
316	UINT8	RW	LED Color Taster 12
317	UINT8	RW	LED Color Taster 13
318	UINT8	RW	LED Color Taster 14
319	UINT8	RW	LED Color Taster 15
320	UINT8	RW	LED Color Taster 16
321	UINT8	RW	LED Color Taster 17
322	UINT8	RW	LED Color Taster 18
323	UINT8	RW	LED Color Taster 19
324	UINT8	RW	LED Color Taster 20
325	UINT8	RW	LED Color Taster 21
326	UINT8	RW	LED Color Taster 22
327	UINT8	RW	LED Color Taster 23

## Modbus RTU Slave:

BFB.790.30/31 D1

### Spezialregister

Die Spezialregister dienen der Überwachung des Modbus-Interfaces und des Modul-Status'. Sie ermöglichen eine systematische Ausfallerkennung des Moduls bzw. des Modbus' sowie eine einmalige Initialisierung von Modulregistern, wenn diese Initialisierung für den Betrieb des Moduls Voraussetzung ist.

Die Verwendung der Spezialregister ist optional, d.h. sie stellen zusätzliche Funktionen zur Verfügung, werden jedoch für eine korrekte Funktion des Modbus' nicht benötigt.

Adr.	Datentyp	Zugriff	Name
195	UINT16	RO	Module Type
196	UINT16	RO	Firmware Version
197	UINT16	RW	Response Delay
198	UINT16	RW	Timeout Value
199	UINT16	RW	Status

- Module Type:* ⇒ Feste Modultypkennung (BFB.790.30/31 = 243/244)
- Firmware Version:* ⇒ High Byte = Major Revision  
Low Byte = Minor Revision
- Response Delay:* ⇒ Verzögerung zwischen Empfang einer Modbus-Messung und Versenden der Antwort, Angabe in ms, z.B. 20 ⇒ Response Delay 20 ms (Default)
- Timeout Value:* ⇒ Timeoutzeit, Angabe in 10 ms, 0 = Abschalten der Funktion (Default), z.B. 300 ⇒ Timeout 3 Sekunden
- Status:* ⇒ High-Byte = ReadStatusCount (s.u.),  
Low-Byte = div. Bits (s.u.),  
Schreibzugriff: Sonderfunktion (s.u.)

Das Register *Status* ermöglicht dem Modbus-Master eine Überwachung des Moduls. Ein Lesezugriff auf *Status* liefert folgende Informationen:

Status Bit																					
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0						
ReadStatusCount														-	-	-	-	-	Tim	Ini	Wr

Der Wert von *ReadStatusCount* wird bei jedem Lesezugriff auf das Register *Status* um eins erhöht. Damit kann der Modbus-Master feststellen, ob das Modul tatsächlich Anfragen beantwortet. *ReadStatusCount* wird durch einen beliebigen Schreibzugriff auf das Register *Status* auf 0 zurückgesetzt.

Das Bit *Wr* ist nach dem Einschalten zunächst 0 (FALSE) und wird 1 (TRUE) nach dem ersten Schreibzugriff auf ein beliebiges Register (auch bel. Schreibzugriff auf Register *Status*). Der Wert 0 (FALSE) kann so interpretiert werden, dass das Modul neu gestartet wurde und evtl. eine Neuinitialisierung notwendig ist.

Das Bit *Ini* ist nach dem Einschalten zunächst 0 (FALSE). Es wird durch einen beliebigen Schreibzugriff auf das Register *Status* auf 1 (TRUE) gesetzt und verbleibt in diesem Zustand bis zum Ausschalten. Damit kann der Modbus-Master vermerken, dass die Modulinitialisierung abgeschlossen und das Modul betriebsbereit ist.

Das Bit *Tim* ist 0 (FALSE), solange kein Timeout aufgetreten ist. Es wird 1 (TRUE), sobald ein Timeout aufgetreten ist. Es kann durch einen beliebigen Schreibzugriff auf das Register *Status* oder auf das Register *Timeout Value* wieder auf 0 (FALSE) zurückgesetzt werden.

Ein beliebiger Schreibzugriff auf das Register *Status* bewirkt das Setzen von *Wr* und *Ini* auf 1 (TRUE), das Rücksetzen von *Tim* auf 0 (FALSE) und das Rücksetzen von *ReadStatusCount* auf 0. Der Datenwert selbst wird dabei ignoriert.

### Timeout-Überwachung

Das Register *Timeout Value* ermöglicht das Aktivieren eines Watchdog-Timers, der den Datenaustausch über den Modbus überwacht. Wenn innerhalb der vorgegebenen Zeit keine Modbus-Zugriffe erfolgen (Read oder Write), werden die Modulausgänge in einen vorgegebenen Zustand gebracht und die Slave-Status-LED beginnt zu blinken. Durch ein beliebiges Modbus-Telegramm wird der Zustand beendet und der Timeout-Zyklus beginnt von vorne.

Die Timeout-Überwachung ist zunächst ausgeschaltet. Sie wird durch Schreiben eines Werts größer Null in das Register *Timeout Value* aktiviert. Der Registerwert ist die Timeoutzeit in 10 ms-Einheiten (z.B. 300 ⇒ Timeout 3 Sekunden).

Bei einem Timeout werden die Modulausgänge entsprechend der Parameter *Error Mode* und *Error Value* (s.o.) gesetzt sowie das Bit *Tim* im Register *Status* auf 1 (TRUE) gesetzt. Das Bit *Tim* bleibt solange gesetzt, bis entweder ein Schreibzugriff auf das Register *Timeout Value* erfolgt oder ein beliebiger Schreibzugriff auf das Register *Status*.

Im normalen Betrieb empfehlen sich Timeout-Zeiten im Bereich mehrerer Sekunden, abhängig von der Geschwindigkeit des Modbus' und der Anzahl der Slaves.

### Response Delay

Das Register *Response Delay* ermöglicht das Einstellen der Zeit zwischen Ende der Master-Anfrage (Read- oder Write-Request) und Beginn der Slave-Antwort (Response). Der Registerwert ist in Millisekunden anzugeben. Nach dem Einschalten ist *Response Delay* auf 20 ms eingestellt.

Durch Eintragen einer kürzeren Zeit erfolgen die Slave-Antworten schneller. Allerdings muss der Modbus-Master schnell genug den RS485-Bus freigeben. Der minimale Wert von *Response Delay* ist 0; in diesem Fall ist die Verzögerung mind. 3,5 Byte-Zeiten gemäß Modbus-Spezifikation.

Große Werte von *Response Delay* sind ggf. in der Timeout-Überwachung zu berücksichtigen.