



Guntamatic
Dokumentation

WEB-Schnittstelle
MODBUS-Schnittstelle

ÄNDERUNGSVERZEICHNIS

Index	Version	Beschreibung	Datum / Kz
01		Basisversion	02.04.2014/FS
02		Modbus Beispiele hinzugefügt; Kesselfreigabe	02.04.2014/plt

INHALTSVERZEICHNIS

ÄNDERUNGSVERZEICHNIS	2
INHALTSVERZEICHNIS	3
WEB-Schnittstelle	4
Bussystem „Ethernet“	4
Datenzugriff	4
Externe Befehle	5
Demoseite	6
MODBUS-Schnittstelle	7
Bussystem „Modbus/TCP“	7
Datenzugriff	7
Externe Befehle	8
Fehlerbehandlung	10
Mapping Übersicht	10
Beispiele	11

WEB-Schnittstelle

Bussystem „Ethernet“

Anbindung über Ethernet-Schnittstelle auf MMI 7.0 (Taurus-BCE).
Der Webserver ist über den TCP-Port 80 verfügbar.

Datenzugriff

Ausgabeformat

Daten werden UTF-8 kodiert im JSON-Format ausgegeben:

```
[{"id":Nummer,"name":Bezeichnung,"type":Datentyp,"unit":Einheit},...]
```

Schlüssel

Für die Erstellung des Schlüssels für den Datenzugriff wird der Systemcode der BCE („Detailanzeige – Netzwerk“) benötigt. Dieser Systemcode kann in der Detailanzeige im Screen „Netzwerk“ abgelesen werden, wenn in den Anlageneinstellungen der Parameter „Netzwerk“ auf ‚Ja‘ parametrisiert ist. Auf Anfrage kann mit dem Systemcode durch Guntamatic der entsprechende Schlüssel für die http-Befehle erstellt werden. Der Länge des Datenstrings ist je nach Berechtigungsstufe unterschiedlich.

Berechtigungsstufen

W0: keine Datenausgabe + externe Befehle
W1: Datenausgabe lt. Berechtigungsstufe W1 + externe Befehle
W2: Datenausgabe lt. Berechtigungsstufe W2 + externe Befehle
W3 : Datenausgabe lt. Berechtigungsstufe W3 + externe Befehle

Datenzugriff

Befehl: `http://<ip>/ext/daqdata.cgi?key=<schlüssel>`

Funktion: Ausgabe der aktuellen Analogwerte und Zustände

Ausgabe: z.B.: [25.50,-20.00,...]

Mapping

Befehl: `http://<ip>/ext/daqdesc.cgi?key=<schlüssel>`

Funktion: Ausgabe der aktuellen Beschreibung der ausgegebenen Analogwerte und Zustände

Ausgabe: z.B.: [{"id":3,"name":"Kesseltemperatur","type":"float","unit":"°C"}, {"id":10,"name":"Puffer T5","type":"float","unit":"°C"},...]

Externe Befehle

Einstellen der Kesselfreigabe:

Synonym bei Powerchip/Powercorn/Biocom/Pro: *PK002*.

Synonym bei Therm/Biostar: *K0010*.

Befehl: `http://<ip>/ext/parset.cgi?syn=<Synonym>&value=0&key=<schlüssel>`

Funktion: Kesselfreigabe wird auf AUTO eingestellt

Ausgabe: {"ack": "Bestätigungsnachricht "} oder {"err": "Fehlermeldung"}

Befehl: `http://<ip>/ext/parset.cgi?syn=<Synonym>&value=1&key=<schlüssel>`

Funktion: Kesselfreigabe wird auf AUS eingestellt

Ausgabe: {"ack": "Bestätigungsnachricht "} oder {"err": "Fehlermeldung"}

Befehl: `http://<ip>/ext/parset.cgi?syn=<Synonym>&value=2&key=<schlüssel>`

Funktion: Kesselfreigabe wird auf EIN eingestellt

Ausgabe: {"ack": "Bestätigungsnachricht "} oder {"err": "Fehlermeldung"}

Einstellen des Reglerprogrammes

Befehl: `http://<ip>/ext/parset.cgi?syn=PR001&value=0&key=<schlüssel>`

Funktion: Reglerprogramm wird auf AUS eingestellt

Ausgabe: {"ack": "Bestätigungsnachricht "} oder {"err": "Fehlermeldung"}

Befehl: `http://<ip>/ext/parset.cgi?syn=PR001&value=1&key=<schlüssel>`

Funktion: Reglerprogramm wird auf NORMAL eingestellt

Ausgabe: {"ack": "Bestätigungsnachricht "} oder {"err": "Fehlermeldung"}

Befehl: `http://<ip>/ext/parset.cgi?syn=PR001&value=2&key=<schlüssel>`

Funktion: Reglerprogramm wird auf WARMWASSER eingestellt

Ausgabe: {"ack": "Bestätigungsnachricht "} oder {"err": "Fehlermeldung"}

Nur bei PC/BC/PH/TH/BS/Pro

Befehl: `http://<ip>/ext/parset.cgi?syn=PR001&value=8&key=<schlüssel>`

Funktion: Reglerprogramm wird auf HANDBETRIEB eingestellt

Ausgabe: {"ack": "Bestätigungsnachricht "} oder {"err": "Fehlermeldung"}

Einstellen des Heizkreisprogrammes

Befehl: `http://<ip>/ext/parset.cgi?syn=HKx01&value=0&key=<schlüssel>`

Funktion: Heizkreis x wird auf AUS eingestellt.

Ausgabe: {"ack": "Bestätigungsnachricht "} oder {"err": "Fehlermeldung"}

Befehl: `http://<ip>/ext/parset.cgi?syn=HKx01&value=1&key=<schlüssel>`

Funktion: Heizkreis x wird auf NORMAL (Uhrenprogramm) eingestellt.

Ausgabe: {"ack": "Bestätigungsnachricht "} oder {"err": "Fehlermeldung"}

Befehl: `http://<ip>/ext/parset.cgi?syn=HKx01&value=2&key=<schlüssel>`

Funktion: Heizkreis x wird auf HEIZEN eingestellt.

Ausgabe: {"ack": "Bestätigungsnachricht "} oder {"err": "Fehlermeldung"}

Befehl: `http://<ip>/ext/parset.cgi?syn=HKx01&value=3&key=<schlüssel>`

Funktion: Heizkreis x wird auf ABSENKEN eingestellt.

Ausgabe: {"ack": "Bestätigungsnachricht "} oder {"err": "Fehlermeldung"}

x=0..8 (Heizkreis)

Einstellen der Warmwasser-Nachladung

Befehl: `http://<ip>/ext/parset.cgi?syn=BKx06&value=1&key=<schlüssel>`

Funktion: Warmwasser x wird auf NACHLADEN eingestellt.

Ausgabe: {"ack": "Bestätigungsnachricht "} oder {"err": "Fehlermeldung"}

x=0..2 (Warmwasserkreis)

Einstellen der Zusatz-Warmwasser-Nachladung

Befehl: `http://<ip>/ext/parset.cgi?syn=ZKx06&value=1&key=<schlüssel>`

Funktion: Zusatz-Warmwasser x wird auf NACHLADEN eingestellt.

Ausgabe: {"ack": "Bestätigungsnachricht "} oder {"err": "Fehlermeldung"}

x=0..2 (Zusatz-Warmwasserkreis)

Demoseite

Befehl: <http://<ip>/demo/index.htm>

Funktion: Öffnet eine Demoseite im Internet-Browser

Beispiel

Bezeichnung	Wert
Kesseltemperatur	24.71°C
Puffer T5	-20°C
Puffer T6	-20°C
Puffer T7	-20°C
Pufferladung	0%

MODBUS-Schnittstelle

Bussystem „Modbus/TCP“

Anbindung über Ethernet-Schnittstelle auf MMI 7.0 (Taurus-BCE)

Modbus TCP wird über TCP/IP-Pakete versendet. Die Verbindung wird auf TCP-Port 502 durchgeführt. Es kann immer nur eine Verbindung zu einem Zeitpunkt aufgebaut werden.

Allgemeiner Protokollaufbau

Byte Nr.	Grösse (Byte)	Wert (Hex)	Beschreibung
1-2	2	xx xx	Transaktionsnummer bei mehreren gleichzeitigen Anfragen
3-4	2	00 00	Protokollkennzeichen (immer 0)
5-6	2	xx xx	Anzahl der folgenden Bytes.
7	1	00	Geräteadresse
8	1	xx	Funktionscode
9-n	n		n Byte Daten

Die übertragenen Daten sind oft auf mehrere Register verteilt. Die Daten werden *Big-Endian* übertragen (high-Byte wird zuerst übertragen).

Datenzugriff

Schlüssel und Berechtigungsstufen

siehe Abschnitt *Bussystem „Ethernet“*.

Zugriff freischalten

Um erweiterte Berechtigungen zu freizuschalten muss der Schlüssel mit der Funktion 16 (0x10) in die Register von Adresse 0x0100 bis 0x017F als Text (lt. Ascii) geschrieben werden.

Beispiel:

Schlüssel = „123456“

Adresse [0x0100] = 0x31

Adresse [0x0101] = 0x32

Adresse [0x0102] = 0x33

Adresse [0x0103] = 0x34

Adresse [0x0104] = 0x35

Adresse [0x0105] = 0x36

Adresse [0x0106] = 0x00

Datenzugriff

Der Datenzugriff erfolgt mit der Funktion **04** des Modbus-Protokolls. Pro Wert sind zwei Wörter (2x2 Byte) im Prozessorabbild reserviert. Der Adressbereich umfasst die Adressen von 0x4000 bis 0x47FF. Im Adressbereich 0x4800 bis 0x4FFF sind die Typinformationen jedes Werts anrufbar. Es muss immer der ganze Wert mit einer Abfrage abgerufen werden (2 Register gelesen werden).

Erweiterte Texte

Textausgaben mit bis zu 64 Zeichen können ab Adresse 0x5000 ausgelesen werden. Es muss immer der ganze Text (32 Register) pro Anfrage ausgelesen werden.

Mapping

Befehl: `http://<ip>/mbmap.cgi?key=<schlüssel>`

Funktion: Ausgabe der aktuellen Beschreibung der ausgegebenen Analogwerte und Zustände

Ausgabe: Modbus Mappingtabelle für die freigegebene Berechtigungsstufe

Beispiel:

Modbus Mapping

Id	Adresse	Typ	Einheit	Name
0	0x4000	string		Kesselzustand
1	0x4004	float	%	Leistung
2	0x4008	float	°C	Kesselsolltemp
3	0x400c	float	°C	Kesseltemperatur
4	0x4010	float	°C	Rauchgas

Die einzelnen Werte können in vier unterschiedlichen Datentypen kodiert werden. Welcher Wert welchen Datentyp verwendet ist ebenfalls im Mapping ersichtlich.

1. **int** Ganzzahl mit Vorzeichen – Wertebereich $-2.147.483.648$ bis $2.147.483.647$. Das Vorzeichen wird mithilfe des *Zweierkomplements* gebildet.
2. **float** Gleitkommazahl (32Bit) kodiert nach IEEE-754.
3. **bool** Boolscher Wert – das LSB hat den Wert 0 oder 1.
4. **string** Bis zu 4 Zeichen Text. Wenn der Text kürzer als 4 Zeichen ist, wird dieser mit dem Zeichen 0x00 terminiert. Es wird der Zeichensatz „iso-8859-1“ verwendet.

Externe Befehle

Die Externen Befehle können mit der Funktion 6 (nur einen Wert ändern) oder der Funktion 16 (mehrere Werte ändern) durchgeführt werden. Der aktuelle Wert kann mit der Funktion 3 gelesen werden.

Einstellen der Kesselfreigabe:

Adresse: 0x0202

Wert: 0

Funktion: Kesselfreigabe wird auf AUTO eingestellt

Wert: 1

Funktion: Kesselfreigabe wird auf AUS eingestellt

Wert: 2

Funktion: Kesselfreigabe wird auf EIN eingestellt

Einstellen des Reglerprogrammes

Adresse: 0x0200

Wert: 0
Funktion: Reglerprogramm wird auf AUS eingestellt

Wert: 1
Funktion: Reglerprogramm wird auf NORMAL eingestellt

Wert: 2
Funktion: Reglerprogramm wird auf WARMWASSER eingestellt

Wert: 8
Funktion: Reglerprogramm wird auf HANDBETRIEB eingestellt

Einstellen des Heizkreisprogrammes

Adresse: $0x0210 + x * 2$

Wert: 0
Funktion: Heizkreis x wird auf AUS eingestellt.

Wert: 1
Funktion: Heizkreis x wird auf NORMAL (Uhrenprogramm) eingestellt.

Wert: 2
Funktion: Heizkreis x wird auf HEIZEN eingestellt.

Wert: 3
Funktion: Heizkreis x wird auf ABSENKEN eingestellt.

$x=0..8$ (Heizkreis)

Einstellen der Warmwasser-Nachladung

Adresse: $0x0230 + x * 2$

Wert: 1
Funktion: Warmwasser x wird auf NACHLADEN eingestellt.

$x=0..2$ (Warmwasserkreis)

Einstellen der Zusatz-Warmwasser-Nachladung

Adresse: $0x0240 + x * 2$

Wert: 1
Funktion: Zusatz-Warmwasser x wird auf NACHLADEN eingestellt.

$x=0..2$ (Zusatz-Warmwasserkreis)

Fehlerbehandlung

Wenn invalide Nachrichten empfangen werden wird die Verbindung von der Steuerung geschlossen. Dies ist der Fall wenn:

- Modbus-Paket zu groß ist, oder
- mehr Daten empfangen werden als im Header angegeben wurden.

In allen anderen Fällen wird eine Modbus-Ausnahme als Antwort zurückgesendet.

Fehlercodes

Falls bei der Bearbeitung des Modbus Telegramms ein Fehler auftritt, wird ein standardisierter Fehlercode zurückgesendet.

Fehlercode	Bezeichnung	Beschreibung
1	Illegal Function	Diese Modbus-Funktion wird nicht unterstützt.
2	Illegal Data Address	Eine ungültige Register-Adresse wurde zugegriffen.
3	Illegal Data Value	Datenwert ungültig.
4	Slave Device failure	Ein Fehler ist bei der Verarbeitung des Befehls aufgetreten.
5	Slave Device busy	Steuerung kann zur Zeit keine Befehle verarbeiten.

Mapping Übersicht

Die folgende Tabelle gibt den Zweck der Adressbereiche an. Auf nicht angegebene Adressen darf nicht zugegriffen werden.

Adressbereich	Größe	R/W	Funktion	Bezeichnung
0x0100 - 0x017F	128	W	16	Schlüssel für weitere Berechtigungen
0x0200	2	R/W	3,6,16	Reglerprogramm
0x0202	2	R/W	3,6,16	Kesselfreigabe
0x0210 - 0x022F	32	R/W	3,6,16	Heizkreisprogramme (bis zu 16 HK)
0x0230 - 0x023F	16	R/W	3,6,16	Warmwasser-Nachladung (bis zu 8 WW)
0x0240 - 0x024F	16	R/W	3,6,16	Zusatz-Warmwasser-Nachladung (bis zu 8 ZWW)
0x4000 - 0x47FF	2048	R	4	DAQ-Daten
0x4800 - 0x4FFF	2048	R	4	DAQ-Typinformationen
0x5000 - 0x6000	4096	R	4	DAQ-Texte mit je max. 64 Zeichen

Beispiele

Anmeldung mit Schlüssel

Schlüssel = „22E1D2EC5CB44772F3E77C0030657CD47C80“

Anfrage

	Byte Nr.	Größe (Byte)	Wert (Hex)	Beschreibung
Header	1-2	2	xx xx	Transaktionsnummer
	3-4	2	00 00	Protokollkennzeichen (immer 00 00)
	5-6	2	00 2d	Zahl der folgenden Bytes
	7	1	00	Geräteadresse (wird ignoriert)
	8	1	10	Funktionscode
Funktion 16	9-10	2	01 00	Startadresse Schreiben
	11-12	2	00 13	Anzahl der Register (Bytes * 2)
	13	1	26	Anzahl der folgenden Bytes (0x26 = 38d)
	14-15	2	32 32	Register für Adresse 0x0100 => 0x3232 „22“
	16-17	2	45 31	Register für Adresse 0x0102 => 0x4531 „E1“
	...	30		Register für Adresse 0x0104 bis 0x0121
	48-49	2	38 30	Register für Adresse 0x0122 => 0x3830 „80“
50-51	2	00 00	Wert 0x00 => Schlüssel abgeschlossen	

Antwort

	Byte Nr.	Größe (Byte)	Wert (Hex)	Beschreibung
Header	1-2	2	xx xx	Transaktionsnummer
	3-4	2	00 00	Protokollkennzeichen (immer 00 00)
	5-6	2	00 06	Zahl der folgenden Bytes
	7	1	00	Geräteadresse (wird ignoriert)
	8	1	10	Funktionscode
F16	9-10	2	01 00	Startadresse Schreiben
	11-12	2	00 13	Anzahl der Register (Bytes * 2)

DAQ Datenabfrage

Dieses Beispiel zeigt, wie die ersten 4 Kanäle auslesen werden können.

Anfrage

	Byte Nr.	Größe (Byte)	Wert (Hex)	Beschreibung
Header	1-2	2	xx xx	Transaktionsnummer
	3-4	2	00 00	Protokollkennzeichen (immer 00 00)
	5-6	2	00 06	Zahl der folgenden Bytes
	7	1	00	Geräteadresse (wird ignoriert)
	8	1	04	Funktionscode
Fo4	9-10	2	40 00	Startadresse Lesen
	11-12	2	00 08	Anzahl der zu lesenden Register

Antwort

	Byte Nr.	Größe (Byte)	Wert (Hex)	Beschreibung
Header	1-2	2	xx xx	Transaktionsnummer
	3-4	2	00 00	Protokollkennzeichen (immer 00 00)
	5-6	2	00 13	Zahl der folgenden Bytes
	7	1	00	Geräteadresse (wird ignoriert)
	8	1	04	Funktionscode
Fo4	9	1	10	Anzahl der folgenden Datenbytes
	10-13	4	xx xx xx xx	DAQ Kanal 0
	14-17	4	xx xx xx xx	DAQ Kanal 1
	18-21	4	xx xx xx xx	DAQ Kanal 2
	22-25	4	xx xx xx xx	DAQ Kanal 3

Wertdekodierung

- int**
 - 00 01 02 03 → 66051
 - FF FF FF F6 → -10
- float**
 - 00 00 00 00 → 0.0
 - C1 A0 00 00 → -20.0
 - 41 A0 00 00 → 20.0
 - 42 5D 33 33 → 55,3
- bool**
 - 00 00 00 00 → Aus
 - 00 00 00 01 → Ein
- string**
 - 41 55 53 00 → "AUS"

Parameter auslesen

Dieses Beispiel zeigt, wie die Einstellung des Reglerprogramms ausgelesen werden kann.

Anfrage

	Byte Nr.	Größe (Byte)	Wert (Hex)	Beschreibung
Header	1-2	2	xx xx	Transaktionsnummer
	3-4	2	00 00	Protokollkennzeichen (immer 00 00)
	5-6	2	00 06	Zahl der folgenden Bytes
	7	1	00	Geräteadresse (wird ignoriert)
	8	1	03	Funktionscode
Fo3	9-10	2	02 00	Startadresse Lesen
	11-12	2	00 01	Anzahl der zu lesenden Register

Antwort

	Byte Nr.	Größe (Byte)	Wert (Hex)	Beschreibung
Header	1-2	2	xx xx	Transaktionsnummer
	3-4	2	00 00	Protokollkennzeichen (immer 00 00)
	5-6	2	00 05	Zahl der folgenden Bytes
	7	1	00	Geräteadresse (wird ignoriert)
	8	1	03	Funktionscode
Fo3	9	1	02	Anzahl der folgenden Datenbytes
	10-11	2	00 xx	Reglerprogramm

Parameter schreiben

Dieses Beispiel zeigt, wie die Einstellung des Reglerprogramms geändert werden kann.

Anfrage

	Byte Nr.	Größe (Byte)	Wert (Hex)	Beschreibung
Header	1-2	2	xx xx	Transaktionsnummer
	3-4	2	00 00	Protokollkennzeichen (immer 00 00)
	5-6	2	00 06	Zahl der folgenden Bytes
	7	1	00	Geräteadresse (wird ignoriert)
	8	1	06	Funktionscode
Fo6	9-10	2	02 00	Registeradresse
	11-12	2	00 01	Neuer Wert (1..Normal)

Antwort

Die Antwort entspricht der Anfrage.

Fehlerbehandlung

Dieses Beispiel zeigt, wie aufgrund einer fehlerhaften Anfrage ein Fehler zurückgeliefert wird.

Anfrage

	Byte Nr.	Größe (Byte)	Wert (Hex)	Beschreibung
Header	1-2	2	xx xx	Transaktionsnummer
	3-4	2	00 00	Protokollkennzeichen (immer 00 00)
	5-6	2	00 06	Zahl der folgenden Bytes
	7	1	00	Geräteadresse (wird ignoriert)
	8	1	06	Funktionscode
Fo6	9-10	2	02 00	Registeradresse
	11-12	2	00 07	Neuer Wert ist nicht erlaubt

Antwort

	Byte Nr.	Größe (Byte)	Wert (Hex)	Beschreibung
Header	1-2	2	xx xx	Transaktionsnummer
	3-4	2	00 00	Protokollkennzeichen (immer 00 00)
	5-6	2	00 03	Zahl der folgenden Bytes
	7	1	00	Geräteadresse (wird ignoriert)
	8	1	86	Funktionscode + 0x80 → Fehler
	9	1	03	Fehlercode