

H
A
N
D
B
U
C
H



**RESI-KNX-MODBUS
RESI-KNX-ASCII**



Text, Abbildungen und Programme wurden mit größter Sorgfalt erarbeitet. Die Firma RESI Informatik & Automation GmbH, Übersetzer und Autoren können jedoch für eventuell verbliebene fehlerhafte Angaben und deren Folgen weder eine juristische Verantwortung noch irgendeine Haftung übernehmen.

Die vorliegende Publikation ist urheberrechtlich geschützt. Alle Rechte vorbehalten. Kein Teil dieses Buches darf ohne vorherige schriftliche Genehmigung der Firma RESI in irgendeiner Form durch Fotokopie, Mikrofilm oder andere Verfahren reproduziert oder in eine für Maschinen, insbesondere Datenverarbeitungsanlagen, verwendbare Sprache übertragen werden. Auch die Rechte der Wiedergabe durch Vortrag, Funk und Fernsehen sind vorbehalten.

Diese Dokumentation und die dazugehörige Software sind urheberrechtlich von der Firma RESI geschützt.

© Copyright 2009-2015 RESI Informatik & Automation GmbH

RESI Informatik & Automation GmbH	Datum:	22.09.2015	Kunde:		Seiten 42
	Version:	01.00	Titel:	RESI-KNX-MODBUS/ASCII Handbuch	
	Bearbeitet von:	DI HC Sigl			
	Geprüft von:	DI HC Sigl	Projekt:		
	Geprüft von:	-			

1 Historie

Datum	Bearbeiter	Beschreibung
13.01.10	DI HC Sigl	Erstversion
20.05.15	DI HC Sigl	Korrekturen zur Query Group Funktion
22.09.15	DI HC Sigl	Version für RESI-KNX-MODBUS und RESI-KNX-ASCII

Proprietary data, company confidential. All rights reserved.
Confé à titre de secret d'entreprise. Tous droits réservés.
Comunicado como secreto empresarial. Reservados todos os direitos.
Comunicado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich anders bestimmt. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung.

2 Inhalt

1	HISTORIE	2
2	INHALT	3
3	WICHTIGE SICHERHEITSHINWEISE	4
4	ALLGEMEINE INFORMATION	6
5	ANSCHLUSS	8
5.1	AUFBAU	8
5.2	KLEMMEN UND LEDs	9
5.3	DIP SWITCH EINSTELLUNGEN	10
5.4	ANSCHLUSSPLAN.....	11
6	KONFIGURATION MIT DER RESI MODBUSCONFIGURATOR SOFTWARE	12
6.1	HERSTELLEN EINER VERBINDUNG.....	12
6.2	BASISFUNKTIONEN.....	12
6.3	DIE KONFIGURATIONSTABELLE	13
6.4	DAS KONTEXTMENÜ	14
6.5	KONTEXT MENÜ: ADD ENTRY	14
6.6	KONTEXT MENÜ: DELETE SELECTED LINES	15
6.7	KONTEXTMENÜ: INSERT ENTRY	15
6.8	KONTEXTMENÜ: COPY ENTRY.....	16
6.9	KONTEXTMENÜ: CLEAR COMPLETE LIST	16
6.10	KONTEXTMENÜ: RENUMBER MODBUS REGISTERS.....	16
6.11	KONTEXTMENÜ: RENUMBER KNX GROUPS	17
6.12	KONTEXTMENÜ: SORT MODBUS REGISTER	17
6.13	KONTEXTMENÜ: SORT KNX GROUP.....	18
6.14	KONTEXTMENÜ: FIND MODBUS REGISTER	18
6.15	KONTEXTMENÜ: FIND KNX GROUP	19
6.16	KONTEXTMENÜ: FIND COMMENT	19
6.17	KONTEXTMENÜ: EDIT ENTRY	20
6.18	DIE KONFIGURATION TESTEN	25
7	BEISPIELKONFIGURATIONEN	26
7.1	EINLESEN DES STATUS EINES KNX TASTERS.....	26
7.2	EINEN KNX AKTOR BESCHREIBEN	26
7.3	ANALOGUE KNX WERTE EINLESEN.....	27
8	FUNKTIONSBESCHREIBUNG	29
8.1	ASCII PROTOKOLLBESCHREIBUNG.....	29
8.1.1	<i>Übersicht</i>	29
8.1.2	<i>Kommunikationsablauf</i>	30
8.1.3	<i>VERSION abfragen</i>	30
8.1.4	<i>Modul TYPE abfragen</i>	31
8.1.5	<i>Tabelle aller ASCII Befehle</i>	32
8.1.6	<i>Die Konfigurationszeile</i>	35
8.1.7	<i>Die „AddConfigurationLine“</i>	37
8.2	MODBUS – REGISTER BESCHREIBUNG	38
8.2.1	<i>Tabelle der Holding Register</i>	38
8.2.2	<i>MODBUS Datentypen, Speicher und übliche Tücken</i>	38
9	SPEZIFIKATIONEN	41
9.1	ABMESSUNGEN	41
9.2	3D ZEICHNUNG	42

Proprietary data, company confidential. All rights reserved.
 Confidencial à titre de secret d'entreprise. Tous droits réservés.
 Comunicado como segredo empresarial. Reservados todos os direitos.
 Comunicado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich anders benannt. Alle Rechte vorbehalten. Insbesondere zu Schadensersatz. Die Rechte vorbehalten. Insbesondere für den Fall der Patenterteilung oder GW-Eintragung.

3 WICHTIGE SICHERHEITSHINWEISE



Lebensgefahr durch elektrischen Strom!

Nur Elektrofachkräfte und elektrotechnisch unterwiesenes Personal dürfen die im folgenden Kapitel beschriebenen Arbeiten ausführen. Beachten Sie für die Installation des Minimoduls die länderspezifischen Vorschriften und Normen. Führen Sie bei eingeschaltetem Gerät keine elektrischen Arbeiten am Gerät aus!

Beachten Sie folgende Regeln:

1. Freischalten der Anlage
2. Sichern gegen Wiedereinschalten
3. Spannungsfreiheit feststellen
4. Andere spannungsführende Teile abdecken

WICHTIGER HINWEIS: Vor der Installation und Inbetriebnahme ist dieser Sicherheitshinweis, die beigefügte Installationsanleitung und das dazugehörige Handbuch zu lesen und alle darin gemachten Hinweise sind zu beachten!

- Die Installation der Geräte darf nur durch Fachpersonal durchgeführt werden!
- Der Anschluss der Geräte darf nur im spannungslosen Zustand erfolgen!
- Führen Sie bei eingeschaltetem Gerät keine elektrischen Arbeiten am Gerät aus!
- Sichern Sie das Gerät gegen Wiedereinschalten!
- Das Gerät darf nur mit der vorgeschriebenen Spannung versorgt werden!
- Schwankungen und Abweichungen der Netzspannung vom Nennwert dürfen die in den technischen Daten angegebenen Toleranzgrenzen und Vorgaben nicht überschreiten. Bei Nichteinhaltung kann es zu Funktionsbeeinträchtigungen und Funktionsstörungen kommen!
- Es sind die aktuellen EMV Richtlinien in der Verkabelung zu beachten!
- Alle Signal- und Anschlussleitungen sind so zu verlegen, dass induktive und kapazitive Störungen sowie Einstreuungen die Funktionen des Geräts nicht beeinflussen. Falsche Verkabelung kann zu erheblichen Fehlfunktionen des Geräts führen!
- Für Signalleitungen und Sensorleitungen sind geschirmte Kabel zu verwenden, um Schäden durch Spannungsinduktion zu verhindern!
- Es sind die aktuellen Sicherheitsvorschriften der ÖVE, VDE, der Länder, ihrer Überwachungsorgane, des TÜV und des örtlichen EVUs zu beachten!
- Beachten Sie die länderspezifischen Vorschriften und Normen!
- Das Gerät ist nur für den angegebenen Verwendungszweck zu benutzen!
- Für Mängel und Schäden, die durch unsachgemäße Verwendung der Geräte entstehen, werden keinerlei Gewährleistungen und Haftungen übernommen!
- Folgeschäden, welche durch Fehler an diesem Gerät entstehen, sind von der Gewährleistung und Haftung ausgenommen!
- Es gelten ausschließlich die technischen Daten, Anschlussbedingungen und Bedienungsanleitungen, welche den Geräten bei der Lieferung beigefügt sind!
- Alle auf unserer Homepage, oder in unserem Datenblatt, in unseren Handbüchern, in unseren Katalogen oder bei unseren Partnern publizierten technischen Daten müssen im Sinne des technischen Fortschritts nicht immer aktuell sein!
- Bei Veränderungen unserer Geräte durch den Anwender entfallen alle Gewährleistungsansprüche!
- Die beim Gerät spezifizierten technischen Rahmenbedingungen (zb Temperaturen, Spannungsversorgung, etc.) sind unbedingt einzuhalten!

- Der Betrieb von Geräten in der Nähe zu unseren Geräten, welche nicht den EMV-Richtlinien entsprechen, kann zur Beeinflussung der Funktionsweise unseres Gerätes bis zum Ausfall unseres Gerätes führen!
- Unsere Geräte dürfen nicht für Überwachungszwecke, welche ausschließlich dem Schutz von Personen gegen Gefährdung oder Verletzung dienen und nicht als Not-Aus-Schalter in Anlagen und Maschinen oder vergleichbare sicherheitsrelevante Aufgaben verwendet werden!
- Die Gehäuse- und Gehäusezubehörmaße können geringe Toleranzen zu den Angaben in der Installationsanleitung bzw. zu den Angaben im Handbuch aufweisen!
- Veränderungen dieser Unterlagen sind nicht gestattet!
- Reklamationen werden nur in unserer vollständigen Originalverpackung angenommen!

4 Allgemeine Information

Mit unseren RESI-KNX-MODBUS und RESI-KNX-ASCII Gateways kann man KNX bussysteme in fast jedes System mit einer RS232 oder RS485 Schnittstelle und einem MODBUS/RTU Masterprotokoll oder ASCII Textprotokoll integrieren. Das Gateway bietet einen integrierten KNX Bus Koppler mit 2-Drahtanschluss. Die zeitkritische KNX Kommunikation wird direkt im Gateway abgehandelt. Das Mapping zwischen den MODBUS/RTU Holdingregistern und den KNX Gruppenadressen wird mit unserer kostenlosen MODBUSConfigurator Software erstellt und in das Modul geladen. Schreibt der Host nun auf MODBUS Register, so versendet das Gateway automatisch das dazugehörige KNX Telegramm. Wird ein KNX Telegramm empfangen, so werden die ankommenden Daten umgerechnet und in den entsprechenden MODBUS Registern abgelegt, sodass ein Host diese Daten lesen kann. Um die Kopplung zwischen Mediensystemen wie Crestron®, AMX® oder Control4® zu erleichtern, bietet unsere Gateway Serie RESI-KNX-ASCII zusätzlich ein Protokoll basierend auf einfache ASCII Kommandos.

- Einfache Integration eines KNX System in eine beliebige Anlage
- MODBUS/RTU Slave Protokoll
- Nur RESI-KNX-ASCII: Kommandos und Meldungen in reinen Text mit ASCII Zeichen
- KNX und Host Schnittstelle sind galvanisch getrennt
- Unterstützt alle 32768 Gruppenadressen
- Unterstützt alle DPT Datentype
- Integrierter KNX Buskoppler
- Host Schnittstelle: RS232, 9600 bis 57600 bps, 8 Datenbits, Keine oder gerade Parität , 1 Stopbit
- Versorgung mit 24 V Gleichspannung
- Leistungsaufnahme <0.5W
- Montage auf eine DIN EN50022 Schiene

Type	Description	Voltage	Power	Weight
RESI-KNX-MODBUS	KNX auf MODBUS/RTU Slave Gateway mit RS232 und RS485 Schnittstelle für alle 32768 KNX Gruppen und max. 150 Konfigurationseinträgen	24 V=	<0.5W	55 g
RESI-KNX-ASCII	KNX auf MODBUS/RTU Slave Gateway mit zusätzlichem ASCII Textprotokoll und RS232 und RS485 Schnittstelle für alle 32768 KNX Gruppen und max. 150 Konfigurationseinträgen	24 V=	<0.5W	55 g

Technische Daten		
Spannungsversorgung		
Versorgungsspannung	24 V= +/-10%	Lagerungstemperatur -20...85 °C
Spannungs-LED	Ja	Arbeitstemperatur 0...60°C
Leistungsaufnahme	<0.5W	Feuchtigkeit 25...90 % rF nicht kondensierend
		Schutzklasse IP20 (EN 60529)
		Abmessungen LxBxH 17,5mm x 90mm x 58mm
		Gewicht 55g
		Montage Auf DIN EN50022 Schiene
MODBUS/RTU Protokoll		
Protokoll	MODBUS/RTU Slave	
Typ	RS232 oder RS485	
Baudrate	9600 bis 57600/8/N oder E/1	
Kabelanschluss	Über Klemmen	
LED Anzeige	Ja	
Galvanische Trennung zur KNX Schnittstelle	Ja	
ASCII Text Schnittstelle		
Protokoll	ASCII Zeichenketten	
Typ	RS232 oder RS485	
Baudrate	9600 bis 57600/8/N oder E/1	
Kabelanschluss	Über Klemmen	
LED Anzeige	Ja	
Galvanische Trennung zur KNX Schnittstelle	Ja	
KNX-Bus Schnittstelle		
Protokoll	KNX	
Baudrate	9600Bits/s	
Kabelanschluss	Über Klemmen	
Galvanische Trennung zur seriellen Schnittstelle	Ja	
LED Anzeige	Ja	
Klemmen		
Kabelquerschnitt	Max. 1,5 mm ²	
Anzugsmoment	Max. 0.5Nm	
		CE Konformität Ja

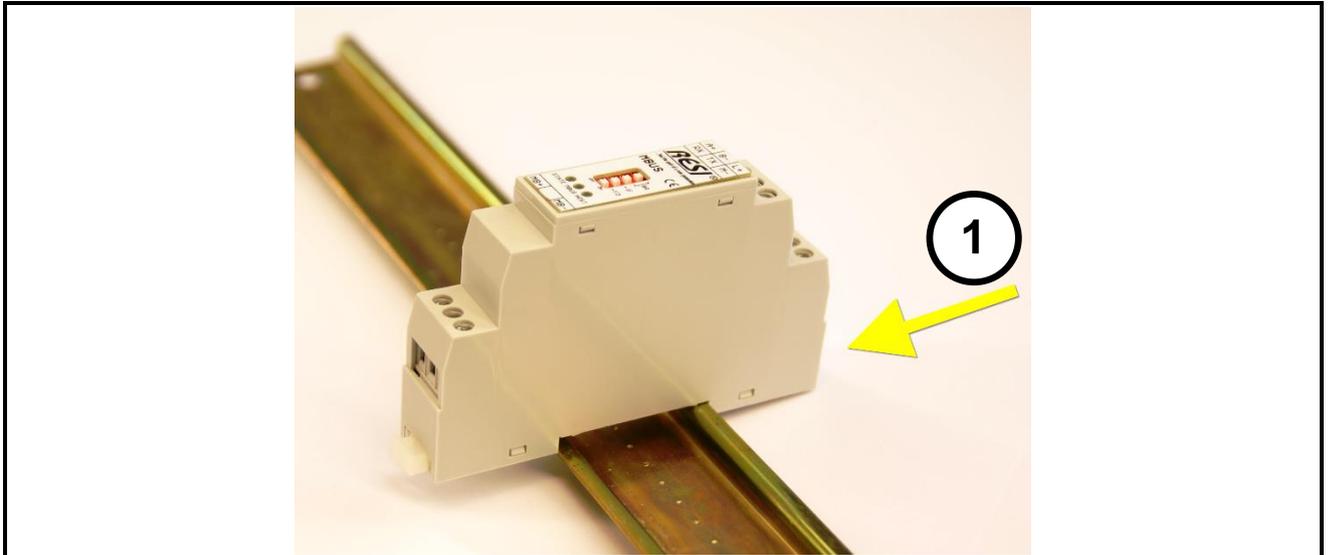
IT Zubehör	
MODBUSConfigurator	Benutzen Sie unsere kostenlose Konfigurationssoftware, um diese Gateways zu konfigurieren und das erstellte Mapping zwischen KNX und MODBUS Registern live zu testen.

5 Anschluss

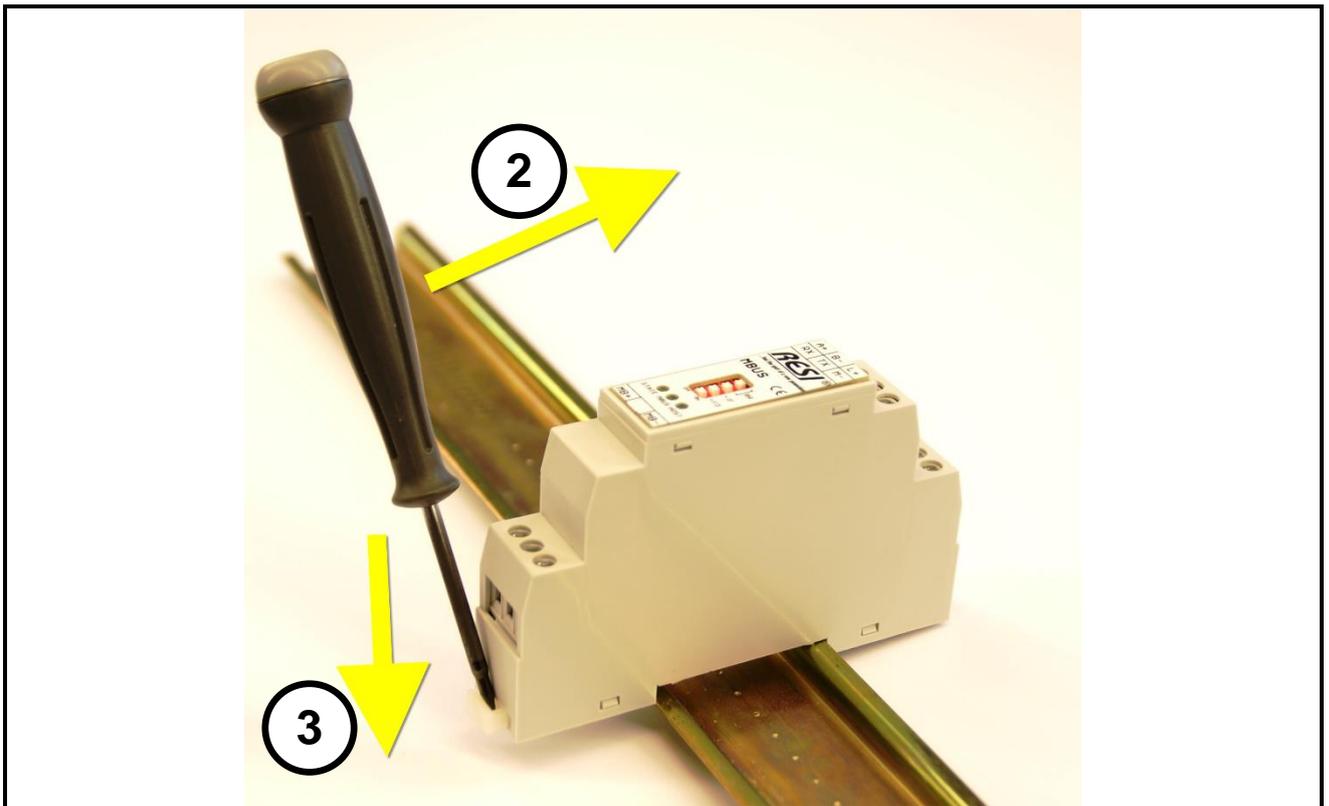
5.1 Aufbau

Unser RESI-KNX-MODBUS/ASCII Gateways sind für die Montage auf eine 35mm DIN-EN50022 Schiene konzipiert. Bitte beachten Sie, dass in der folgenden Montageanleitung nur Symbolphotos verwendet werden.

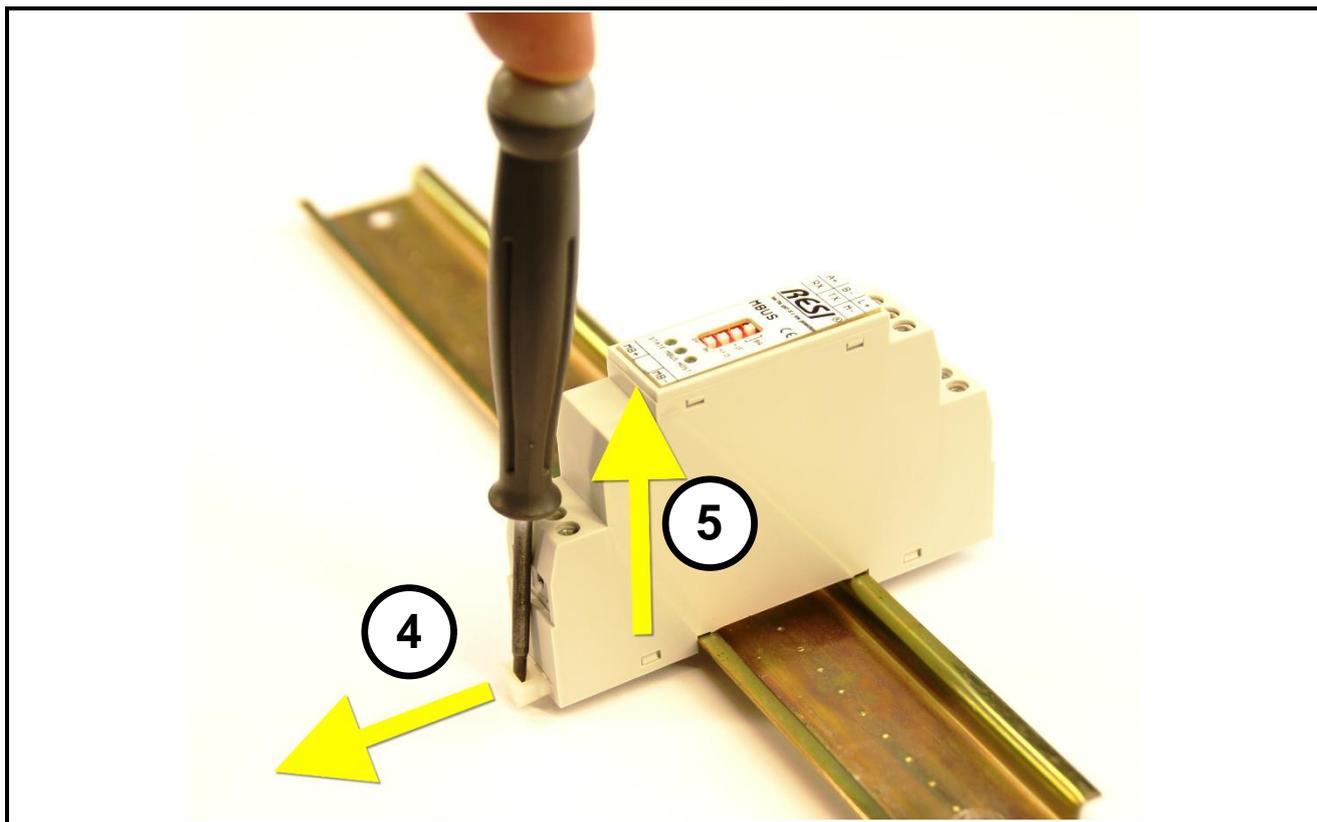
Zuerst stecken Sie die Oberseite des Gateways in die DIN Schiene (1).



Danach öffnen Sie den unteren Haltehebel mit einem Schraubenzieher (2). Pressen Sie nun das Modul mit der Unterseite bei geöffneten Haltehebel auf die DIN Schiene (3). Lassen Sie den Haltehebel los. Dieser rastet nun in die DIN Schiene ein und das Modul ist nun korrekt auf der DIN Schiene fixiert.



Um das Modul wieder von der DIN Schiene zu entfernen, muss man zuerst den Haltehebel mit einem Schraubenzieher öffnen (4). Danach kippt man das Modul bei geöffnetem Haltehebel nach oben. Nun nur mehr das Modul leicht schräg von der DIN Schiene abheben, um auch die Oberseite auszuhacken.



5.2 Klemmen und LEDs

	RESI-KNX-MODBUS/RESI-KNX-ASCII
L+	Spannungsversorgung
M-	L+: 24 V= M-: Masse
A B	RS485 MODBUS/RTU Schnittstelle oder ASCII Kommandos A: DATA+ B: DATA-
RX TX M-	RS232 MODBUS/RTU Schnittstelle oder ASCII Kommandos RX: Serielle Empfangsleitung TX: Serielle Sendeleitung M-: Masseleitung für RS232
K+ K-	Schnittstelle für KNX Anschluss K+: KNX+ Busleitung (rot) K-: KNX- Busleitung (schwarz)
STATE	Status-LED, blinkt langsam, wenn der Konverter funktioniert und der KNX angeschlossen ist, blinkt schnell, wenn der KNX nicht angeschlossen ist
KNX	KNX Aktivitäts-LED, zeigt an, wenn KNX Telegramme versendet oder empfangen werden
HOST	HOST-LED, blinkt, wenn der Host Telegramme sendet/empfängt

Tabelle: Beschreibung der Anschlüsse und Anzeigen der RESI-KNX-MODBUS/RESI-KNX-ASCII Gateways

5.3 DIP Switch Einstellungen

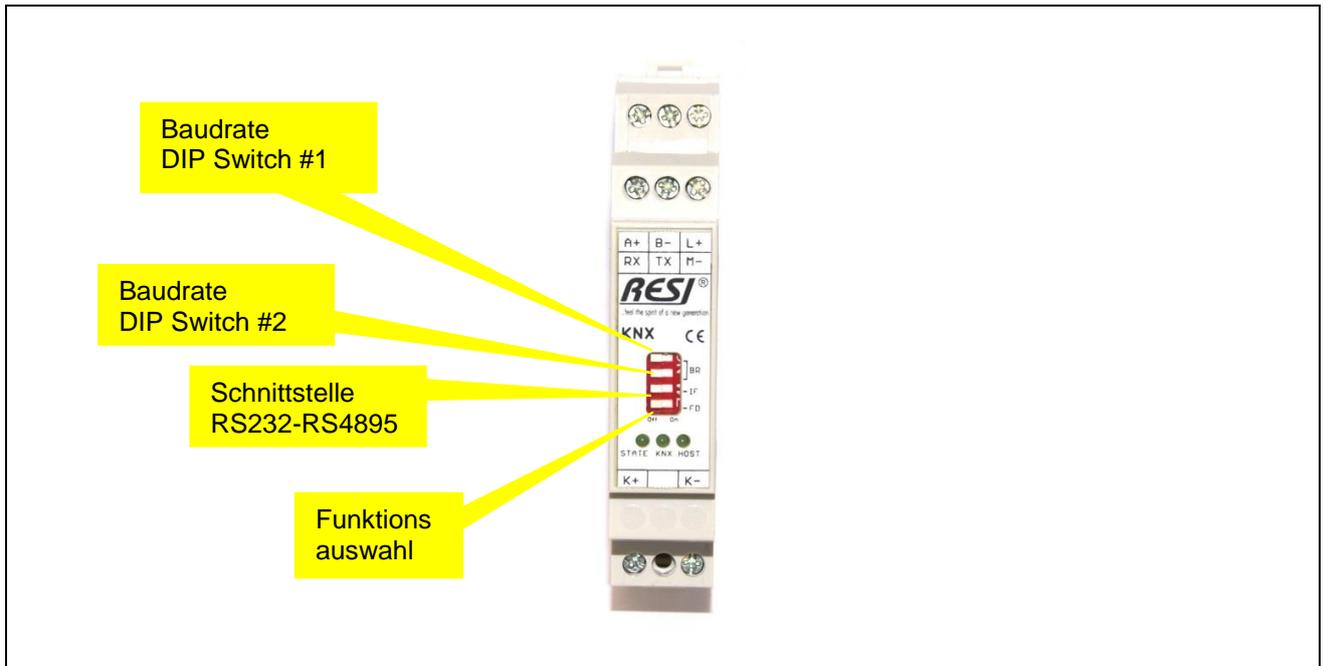


Abbildung: Beschreibung der DIP Switch Einstellungen und LED Anzeigen

DIP Switch	RESI-KNX-MODBUS/RESI-KNX-ASCII
Baudrate BR	Benutzen Sie DIP Switch 1+2, um die Baudrate zu wählen: AUS AUS: 9600Bd EIN AUS: 19200Bd AUS EIN: 38400Bd EIN EIN: 57600Bd HINWEIS: Die korrekte Parität (KEINE, GERADE oder UNGERADE) wird über die PC Software eingestellt, nicht mit DIP Switches.
Schnittstelle IF	Wählt die physikalische Art der seriellen Schnittstelle für den ASCII Host aus: AUS=RS232 EIN=RS485
Funktionswahl FD	Wählt eine Spezialfunktion aus: AUS=Es wird die im FLASH gespeicherte MODBUS/RTU Unit ID verwendet EIN=Es wird immer die MODBUS/RTU Unit ID 255 verwendet

Tabelle: Beschreibung der DIP Switch Funktionen

Proprietary data, company confidential. All rights reserved.
 Confidencial, información de secreto empresarial. Reservados todos los derechos.
 Comunicado como segredo empresarial. Reservados todos os direitos.
 Comunicado como segredo industrial. Nos reservamos todos os direitos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts ist nicht gestattet, soweit nicht ausdrücklich anders bestimmt ist. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung.

5.4 Anschlussplan

In der untenstehenden Abbildung ist die Verdrahtung des Konverters angeführt.

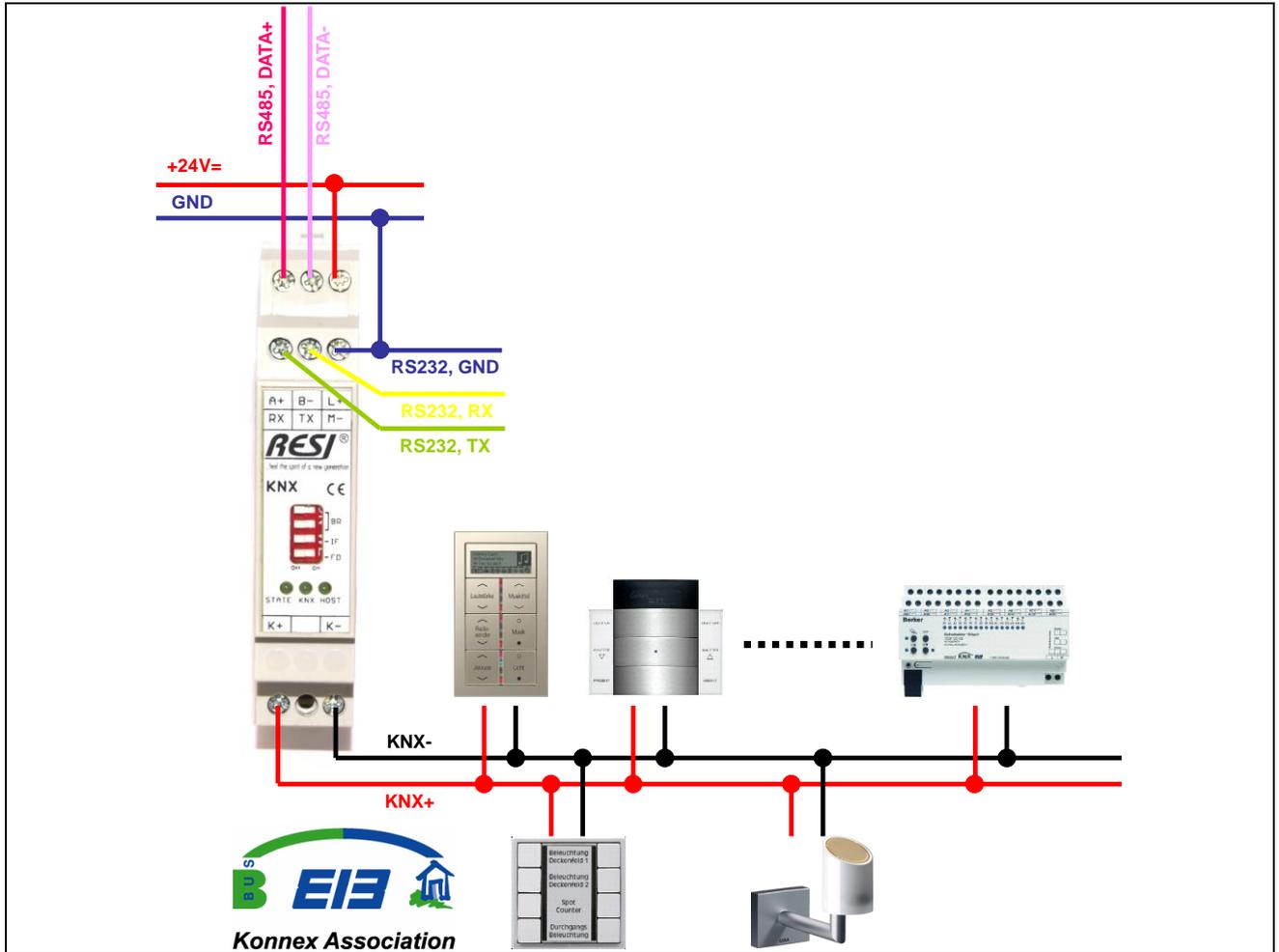


Abbildung: Verdrahtung des Gateways

Proprietary data, company confidential. All rights reserved.
 Confidantia de companie, proprietate intelectuala. Toate drepturile rezervate.
 Comunicado como segredo empresarial. Reservados todos los derechos.
 Comunicado como segredo industrial. Nos reservamos todos los derechos.

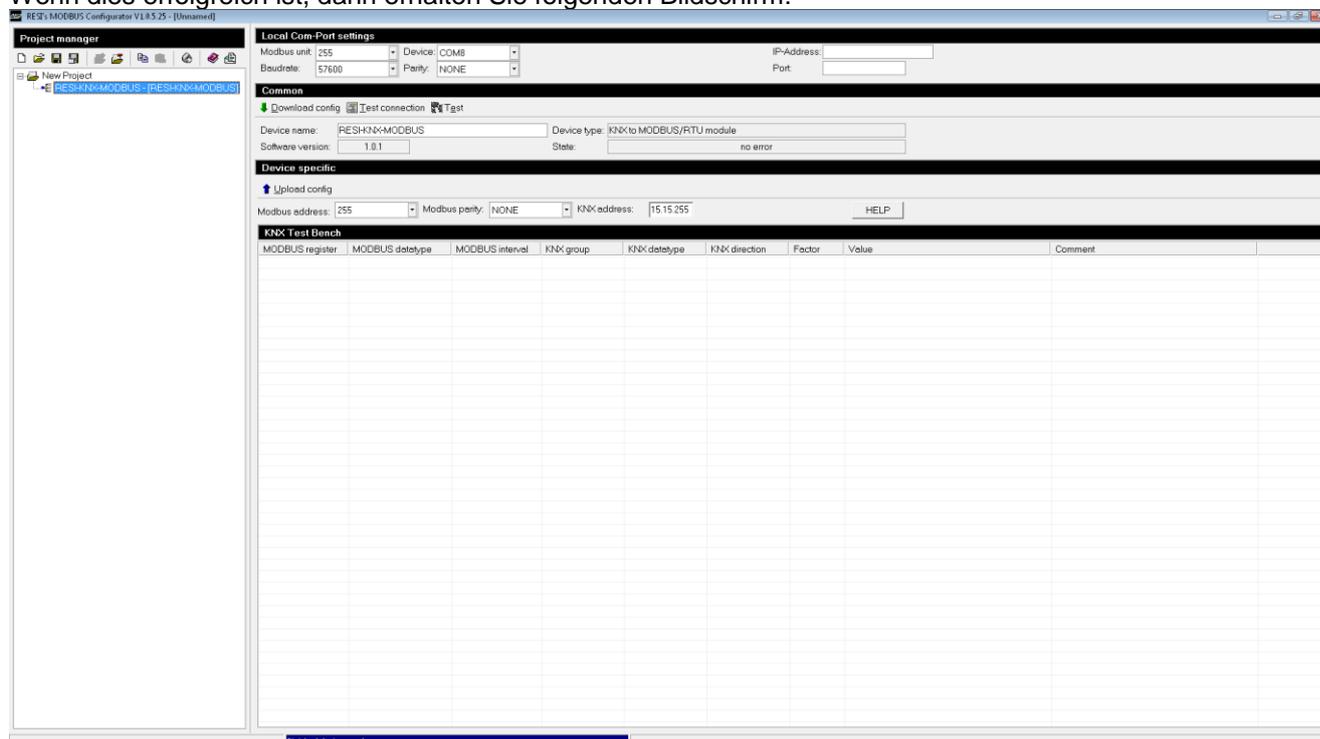
Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich anders bestimmt. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Ertragung.

6 Konfiguration mit der RESI MODBUSConfigurator Software

Laden Sie unsere kostenlose Konfigurationssoftware von unserer Homepage www.RESI.cc herunter und installieren Sie diese Software auf Ihren Computer.

6.1 Herstellen einer Verbindung

Stellen Sie eine Verbindung zwischen dem Gateway und unserer Software RESI MODBUSConfigurator her. Wenn dies erfolgreich ist, dann erhalten Sie folgenden Bildschirm:



6.2 Basisfunktionen

Wie man hier sehen kann, trifft man individuelle Einstellungen im oberen Bereich des Bildschirms:



- Schaltfläche "Download config": Wenn Sie die MODBUS/RTU Slave Adresse, die MODBUS Parität oder die KNX Adresse ändern oder wenn Sie das KNX Mapping ändern, müssen Sie die Änderungen zuerst in das Gateway laden, damit diese Änderungen wirksam werden.
- Schaltfläche "Upload config": Mit dieser Schaltfläche kann man eine gespeicherte Konfiguration aus einem Gateway in die Software hochladen. Aber vergessen Sie nicht, dass der Kommentar nicht in das Gateway gespeichert wird. Somit geht diese Information beim Hochladen verloren!
- Schaltfläche "Test connection": Diese Schaltfläche überprüft, ob mit dem angeschlossenen Gateway kommuniziert werden kann oder nicht.
- Schaltfläche "Test": Diese Schaltfläche aktiviert/deaktiviert eine Testfunktion. Die Software liest ca. alle 5 Sekunden alle konfigurierten MODBUS Register aus, und stellt deren Inhalt in der Tabelle dar. In diesem Testmodus kann auch auf MODBUS/RTU Holdingregister geschrieben werden und somit KNX Telegramme am KNX Bus generiert werden.

6.3 Die Konfigurationstabelle

Im Abschnitt „Device specific“ sehen Sie eine Tabelle mit der aktuellen Konfiguration der MODBUS-KNX Zuordnung:

Ein Zuordnungseintrag besteht aus den folgenden Einträgen:

- **MODBUS register:** Die Nummer des Holding Registers oder der Startindex der Holding Register, wenn mehrere Register benutzt werden, auf welche die empfangenen KNX Daten geschrieben oder von denen die zu sendenden KNX Daten gelesen werden.
- **MODBUS datatype:** Der Datentyp für die MODBUS Register. Dieser Eintrag definiert wie der Konverter die Umrechnung zwischen KNX Daten und MODBUS Daten vornimmt und wie viele MODBUS Register zum Speichern benutzt werden.
- **MODBUS interval:** Dieser Eintrag ist für zukünftige Erweiterungen und definiert ein Zeitintervall in Sekunden, um automatisch KNX Daten vom KNX Bus anzufordern. Derzeit wird dieser Eintrag nicht benutzt.
- **KNX group:** Dieser Eintrag definiert die KNX Gruppenadresse, die zum Versenden oder Empfangen von KNX Telegrammen verwendet wird.
- **KNX datatype:** Dieser Eintrag definiert den KNX Datentyp, der beim Versenden oder Empfangen von KNX Daten mit der eingestellten KNX Gruppenadresse verwendet wird.
- **KNX Richtung:** Dieser Eintrag definiert die Kommunikationsrichtung am KNX Bus: Man kann KNX Daten am KNX Bus versenden, empfangen oder beides.
- **Factor:** Dieser Eintrag definiert einen Faktor, mit dem ankommende KNX Daten multipliziert werden, bevor diese in die MODBUS Register abgelegt werden. Vor dem Versenden von KNX Telegrammen, werden die Daten aus den MODBUS Registern mit diesem Faktor dividiert. Der Wert Null definiert, dass dieser Faktor unbenutzt ist.
- **Comment:** Dieser Eintrag definiert einen benutzerdefinierten Text als Erklärung für diese Zuordnung. Das ist nur für die Dokumentation gedacht und wird nicht im Gateway gespeichert, sondern nur wenn man das PC Projekt speichert. Beim Hochladen einer Konfiguration aus einem Gateway in die Software ist dieser Kommentar nicht dabei und wird durch einen Standardtext ersetzt.

6.6 Kontext Menü: Delete selected lines

Eintrag "Delete selected lines...": Zuerst wählen Sie eine oder mehrere Zeilen aus der Zuordnungstabelle aus. Um mehr als eine Zeile auszuwählen, drücken und halten Sie die STRG oder SHIFT Taste und wählen danach mit der Maus andere Zeilen aus.

MODBUS register	MODBUS datatype	MODBUS interval	KNX group	KNX datatype	KNX direction	Factor	Value	Comment
4x1	UINT16	0	0.0.0	BIT	READ-WRITE	1.00	????	no comment
4x2	UINT16	0	0.0.1	BIT	READ-WRITE	1.00	????	no comment
4x3	UINT16	0	0.0.2	BIT	READ-WRITE	1.00	????	no comment
4x4	UINT16	0	0.0.3	BIT	READ-WRITE	1.00	????	no comment

Dann öffnen Sie das Kontextmenü und wählen Sie die Funktion "Delete selected lines..." aus. Das System wird nun alle ausgewählten Zeilen aus der Konfigurationstabelle löschen und das Ergebnis wird wie folgt aussehen:

MODBUS register	MODBUS datatype	MODBUS interval	KNX group	KNX datatype	KNX direction	Factor	Value	Comment
4x1	UINT16	0	0.0.0	BIT	READ-WRITE	1.00	????	no comment
4x4	UINT16	0	0.0.3	BIT	READ-WRITE	1.00	????	no comment

6.7 Kontextmenü: Insert entry

Eintrag "Insert entry...": Zuerst wählen Sie eine oder mehrere Zeilen aus. Dann wählen Sie diese Funktion aus dem lokalen Kontextmenü aus. Das System wird nun neue Konfigurationszeilen direkt hinter jeder selektierten Zeile einfügen.

MODBUS register	MODBUS datatype	MODBUS interval	KNX group	KNX datatype	KNX direction	Factor	Value	Comment
4x1	UINT16	0	0.0.0	BIT	READ-WRITE	1	????	no comment
4x4	UINT16	0	0.0.3	BIT	READ-WRITE	1	????	no comment

Das Ergebnis wird wie folgt sein:

MODBUS register	MODBUS datatype	MODBUS interval	KNX group	KNX datatype	KNX direction	Factor	Value	Comment
4x1	UINT16	0	0.0.0	BIT	READ-WRITE	1	????	no comment
4x2	UINT16	0	0.0.1	BIT	READ-WRITE	1	????	no comment
4x4	UINT16	0	0.0.3	BIT	READ-WRITE	1	????	no comment

6.8 Kontextmenü: Copy entry

Eintrag "Copy entry...": Zuerst wählen Sie eine oder mehrere Zeilen aus. Dann wählen Sie diese Funktion aus dem lokalen Kontextmenü:

MODBUS register	MODBUS datatype	MODBUS interval	KNX group	KNX datatype	KNX direction	Factor	Value	Comment
4x1	UINT32	0	1.0.0	SIXBITS	WRITE	100	????	A new comment

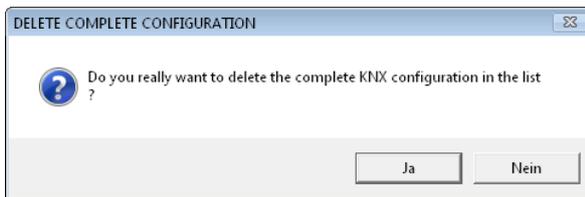
Das System kopiert nun jede selektierte Zeile und fügt eine kopierte Zeile der Konfiguration hinzu. Das Ergebnis sieht wie folgt aus:

MODBUS register	MODBUS datatype	MODBUS interval	KNX group	KNX datatype	KNX direction	Factor	Value	Comment
4x1	UINT32	0	1.0.0	SIXBITS	WRITE	100	????	A new comment
4x3	UINT32	0	1.0.1	SIXBITS	WRITE	100	????	A new comment

Wie man bemerkt, inkrementiert das System automatisch die MODBUS/RTU Register Nummer aufgrund des konfigurierten MODBUS Datentyps. Dasselbe macht das System mit der KNX Gruppenadresse.

6.9 Kontextmenü: Clear complete list

Eintrag "Clear complete list": nach der Auswahl dieser Funktion aus dem lokalen Kontextmenü erscheint die folgende Frage am Bildschirm:



Wenn diese Frage mit JA beantwortet wird, so werden alle Einträge aus Ihrer Konfiguration für immer gelöscht! Die Antwort NEIN bricht diese Funktion ab.

6.10 Kontextmenü: Renumber MODBUS registers

Eintrag "Renumber MODBUS registers": Zuerst wählen Sie die Zeilen aus, welche Sie neu nummerieren wollen, dann wählen Sie diese Funktion aus dem lokalen Kontextmenü aus:

MODBUS register	MODBUS datatype	MODBUS interval	KNX group	KNX datatype	KNX direction	Factor	Value	Comment
4x2	UINT16	0	0.0.0	BIT	READ-WRITE	1	????	no comment
4x1	UINT16	0	0.0.1	BIT	READ-WRITE	1	????	no comment
4x12	UINT16	0	0.0.2	BIT	READ-WRITE	1	????	no comment
4x7	UINT16	0	0.0.3	BIT	READ-WRITE	1	????	no comment

Der Index des MODBUS Registers aus der ersten selektierten Zeile wird für den ersten Eintrag definiert. Danach werden alle weiteren Zeilen aufgrund der konfigurierten MODBUS Datentypen durchnummeriert. Das Ergebnis sieht wie folgt aus:

MODBUS register	MODBUS datatype	MODBUS interval	KNX group	KNX datatype	KNX direction	Factor	Value	Comment
4x2	UINT16	0	0.0.0	BIT	READ-WRITE	1	????	no comment
4x3	UINT16	0	0.0.1	BIT	READ-WRITE	1	????	no comment
4x4	UINT16	0	0.0.2	BIT	READ-WRITE	1	????	no comment
4x5	UINT16	0	0.0.3	BIT	READ-WRITE	1	????	no comment

6.11 Kontextmenü: Renumber KNX groups

Eintrag "Renumber KNX groups": Zuerst wählen Sie die neu zu nummerierenden Zeilen aus. Dann wählen Sie diese Funktion aus dem lokalen Kontextmenü aus.

MODBUS register	MODBUS datatype	MODBUS interval	KNX group	KNX datatype	KNX direction	Factor	Value	Comment
4x2	UINT16	0	1.3.4	BIT	READ-WRITE	1	????	no comment
4x3	UINT16	0	0.0.1	BIT	READ-WRITE	1	????	no comment
4x4	UINT16	0	0.0.2	BIT	READ-WRITE	1	????	no comment
4x5	UINT16	0	0.0.3	BIT	READ-WRITE	1	????	no comment

- Edit entry...
- Add entry...
- Insert entry...
- Copy entry...
- Delete selected entries
- Clear complete list
- Renumber MODBUS registers
- Renumber KNX groups
- Sort MODBUS register
- Sort KNX group
- Find MODBUS register
- Find KNX Group
- Find comment

Die KNX Gruppenadresse des ersten Eintrags wird als Startadresse für den ersten Eintrag verwendet. Die KNX Gruppenadressen aller weiteren Zeilen werden in 1er Schritten aufsteigend durchnummeriert. Das Ergebnis sieht wie folgt aus:

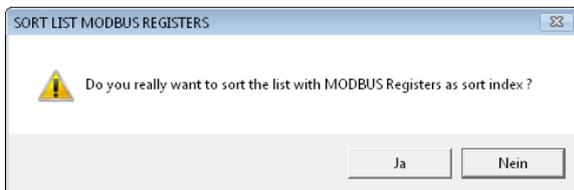
MODBUS register	MODBUS datatype	MODBUS interval	KNX group	KNX datatype	KNX direction	Factor	Value	Comment
4x2	UINT16	0	1.3.4	BIT	READ-WRITE	1	????	no comment
4x3	UINT16	0	1.3.5	BIT	READ-WRITE	1	????	no comment
4x4	UINT16	0	1.3.6	BIT	READ-WRITE	1	????	no comment
4x5	UINT16	0	1.3.7	BIT	READ-WRITE	1	????	no comment

6.12 Kontextmenü: Sort MODBUS register

Eintrag "Sort MODBUS register": Zuerst wählen Sie die neu zu sortierenden Zeilen aus. Dann wählen Sie diese Funktion aus dem lokalen Kontextmenü aus.

MODBUS register	MODBUS datatype	MODBUS interval	KNX group	KNX datatype	KNX direction	Factor	Value	Comment
4x10	UINT16	0	0.0.0	BIT	READ-WRITE	1	????	no comment
4x5	UINT16	0	0.0.1	BIT	READ-WRITE	1	????	no comment
4x7	UINT16	0	0.0.1	BIT	READ-WRITE	1	????	no comment
4x8	UINT16	0	0.0.2	BIT	READ-WRITE	1	????	no comment
4x1	UINT16	0	0.0.2	BIT	READ-WRITE	1	????	no comment
4x4	UINT16	0	0.0.3	BIT	READ-WRITE	1	????	no comment

- Edit entry...
- Add entry...
- Insert entry...
- Copy entry...
- Delete selected entries
- Clear complete list
- Renumber MODBUS registers
- Renumber KNX groups
- Sort MODBUS register
- Sort KNX group
- Find MODBUS register
- Find KNX Group
- Find comment



Wenn Sie diese Frage mit JA beantworten, sortiert das System die selektieren Zeilen aufsteigend. Als Sortierkriterium wird der MODBUS Register Index verwendet. Das Ergebnis sieht wie folgt aus:

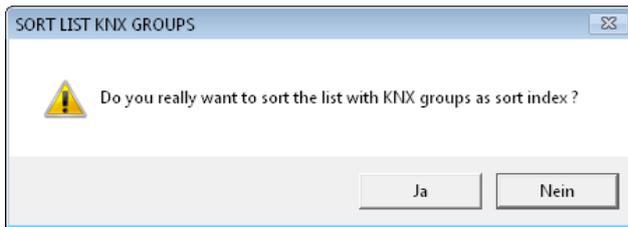
MODBUS register	MODBUS datatype	MODBUS interval	KNX group	KNX datatype	KNX direction	Factor	Value	Comment
4x1	UINT16	0	0.0.2	BIT	READ-WRITE	1	????	no comment
4x4	UINT16	0	0.0.3	BIT	READ-WRITE	1	????	no comment
4x5	UINT16	0	0.0.1	BIT	READ-WRITE	1	????	no comment
4x7	UINT16	0	0.0.1	BIT	READ-WRITE	1	????	no comment
4x8	UINT16	0	0.0.2	BIT	READ-WRITE	1	????	no comment
4x10	UINT16	0	0.0.0	BIT	READ-WRITE	1	????	no comment

6.13 Kontextmenü: Sort KNX group

Eintrag "Sort KNX group": Nachdem Sie eine oder mehrere Zeilen zum Sortieren ausgewählt haben, wählen sie diese Funktion aus dem lokalen Kontextmenü.

MODBUS register	MODBUS datatype	MODBUS interval	KNX group	KNX datatype	KNX direction	Factor	Value	Comment
4x1	UINT16	0	1.5.67	BIT	READ-WRITE	1	????	no comment
4x2	UINT16	0	7.5.255	BIT	READ-WRITE	1	????	no comment
4x3	UINT16	0	1.5.68	BIT	READ-WRITE	1	????	no comment
4x4	UINT16	0	7.5.45	BIT	READ-WRITE	1	????	no comment

- Edit entry...
- Add entry...
- Insert entry...
- Copy entry...
- Delete selected entries
- Clear complete list
- Renumber MODBUS registers
- Renumber KNX groups
- Sort MODBUS register
- Sort KNX group
- Find MODBUS register
- Find KNX Group
- Find comment



Wenn Sie die obige Frage mit JA beantworten, sortiert das System alle selektierten Zeilen aufsteigend nach der KNX Gruppenadresse. Das Ergebnis sieht wie folgt aus:

MODBUS register	MODBUS datatype	MODBUS interval	KNX group	KNX datatype	KNX direction	Factor	Value	Comment
4x1	UINT16	0	1.5.67	BIT	READ-WRITE	1	????	no comment
4x3	UINT16	0	1.5.68	BIT	READ-WRITE	1	????	no comment
4x4	UINT16	0	7.5.45	BIT	READ-WRITE	1	????	no comment
4x2	UINT16	0	7.5.255	BIT	READ-WRITE	1	????	no comment

6.14 Kontextmenü: Find MODBUS register

Eintrag "Find MODBUS register": Nachdem Sie diese Funktion ausgewählt haben, öffnet sich ein Eingabefenster. Geben Sie einen gültigen MODBUS Registerindex an und drücken Sie die OK Schaltfläche.

MODBUS register	MODBUS datatype	MODBUS interval	KNX group	KNX datatype	KNX direction	Factor	Value	Comment
4x1	UINT16	0	1.5.67	BIT	READ-WRITE	1	????	no comment
4x3	UINT16	0	1.5.68	BIT	READ-WRITE	1	????	no comment
4x4	UINT16	0	7.5.45	BIT	READ-WRITE	1	????	no comment
4x2	UINT16	0	7.5.255	BIT	READ-WRITE	1	????	no comment



Das System wird nun alle Zeilen auswählen, in welchen die eingegebene MODBUS Register Nummer vorkommt. Das Ergebnis sieht wie folgt aus:

MODBUS register	MODBUS datatype	MODBUS interval	KNX group	KNX datatype	KNX direction	Factor	Value	Comment
4x1	UINT16	0	1.5.67	BIT	READ-WRITE	1	????	no comment
4x3	UINT16	0	1.5.68	BIT	READ-WRITE	1	????	no comment
4x4	UINT16	0	7.5.45	BIT	READ-WRITE	1	????	no comment
4x2	UINT16	0	7.5.255	BIT	READ-WRITE	1	????	no comment

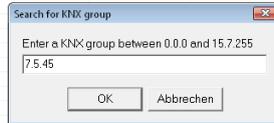
Proprietary data, company confidential. All rights reserved.
 Confidantia de companie, proprietate intelectuală. Toate drepturile rezervate.
 Comunicado como secreto empresarial. Reservados todos los derechos.
 Comunicado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich anders bestimmt. Alle Rechte vorbehalten. Insbesondere für den Fall der Patenterteilung oder GM-Eintragung.

6.15 Kontextmenü: Find KNX group

Eintrag "Find KNX group": Nachdem Sie diese Funktion ausgewählt haben, öffnet sich ein Eingabefenster. Geben Sie eine gültige KONX Gruppenadresse an und drücken Sie die OK Schaltfläche.

MODBUS register	MODBUS datatype	MODBUS interval	KNX group	KNX datatype	KNX direction	Factor	Value	Comment
4x1	UINT16	0	1.5.67	BIT	READ-WRITE	1	????	no comment
4x3	UINT16	0	1.5.68	BIT	READ-WRITE	1	????	no comment
4x4	UINT16	0	7.5.45	BIT	READ-WRITE	1	????	no comment
4x2	UINT16	0	7.5.255	BIT	READ-WRITE	1	????	no comment



Search for KNX group

Enter a KNX group between 0.0.0 and 15.7.255

7.5.45

OK Abbrechen

Das System selektiert nun alle Zeilen, in welchen die KNX Gruppenadresse übereinstimmt. Das Ergebnis sieht wie folgt aus:

MODBUS register	MODBUS datatype	MODBUS interval	KNX group	KNX datatype	KNX direction	Factor	Value	Comment
4x1	UINT16	0	1.5.67	BIT	READ-WRITE	1	????	no comment
4x3	UINT16	0	1.5.68	BIT	READ-WRITE	1	????	no comment
4x4	UINT16	0	7.5.45	BIT	READ-WRITE	1	????	no comment
4x2	UINT16	0	7.5.255	BIT	READ-WRITE	1	????	no comment

6.16 Kontextmenü: Find comment

Eintrag "Find comment": Nachdem Sie diese Funktion aus dem lokalen Kontextmenü gewählt haben erscheint folgendes Eingabefenster. Geben Sie nun einen Teil eines Kommentars oder einen ganzen Kommentar ein und drücken Sie die OK Schaltfläche. Das System markiert nun alle Zeilen in der Konfigurationstabelle, in denen der Kommentar mit diesem Textteil übereinstimmt.

MODBUS register	MODBUS datatype	MODBUS interval	KNX group	KNX datatype	KNX direction	Factor	Value	Comment
4x1	UINT16	0	1.0.1	BIT	READ-WRITE	1	????	A first comment
4x2	UINT16	0	1.0.2	BIT	READ-WRITE	1	????	A second comment
4x3	UINT16	0	1.0.3	BIT	READ-WRITE	1	????	A third comment
4x4	UINT16	0	1.0.4	BIT	READ-WRITE	1	????	A fourth comment



Search for comment

Enter a part of the comment

comm

OK Abbrechen

Das Ergebnis sieht wie folgt aus:

MODBUS register	MODBUS datatype	MODBUS interval	KNX group	KNX datatype	KNX direction	Factor	Value	Comment
4x1	UINT16	0	1.0.1	BIT	READ-WRITE	1	????	A first comment
4x2	UINT16	0	1.0.2	BIT	READ-WRITE	1	????	A second comment
4x3	UINT16	0	1.0.3	BIT	READ-WRITE	1	????	A third comment
4x4	UINT16	0	1.0.4	BIT	READ-WRITE	1	????	A fourth comment

Noch ein Beispiel:

MODBUS register	MODBUS datatype	MODBUS interval	KNX group	KNX datatype	KNX direction	Factor	Value	Comment
4x1	UINT16	0	1.0.1	BIT	READ-WRITE	1	????	A first comment
4x2	UINT16	0	1.0.2	BIT	READ-WRITE	1	????	A second comment
4x3	UINT16	0	1.0.3	BIT	READ-WRITE	1	????	A third comment
4x4	UINT16	0	1.0.4	BIT	READ-WRITE	1	????	A fourth comment
4x5	UINT16	0	1.0.5	BIT	READ-WRITE	1	????	A fifth comment
4x6	UINT16	0	1.0.6	BIT	READ-WRITE	1	????	A sixth comment



Search for comment

Enter a part of the comment

A s

OK Abbrechen

Das Ergebnis sieht wie folgt aus:

MODBUS register	MODBUS datatype	MODBUS interval	KNX group	KNX datatype	KNX direction	Factor	Value	Comment
4x1	UINT16	0	1.0.1	BIT	READ-WRITE	1	????	A first comment
4x2	UINT16	0	1.0.2	BIT	READ-WRITE	1	????	A second comment
4x3	UINT16	0	1.0.3	BIT	READ-WRITE	1	????	A third comment
4x4	UINT16	0	1.0.4	BIT	READ-WRITE	1	????	A fourth comment
4x5	UINT16	0	1.0.5	BIT	READ-WRITE	1	????	A fifth comment
4x6	UINT16	0	1.0.6	BIT	READ-WRITE	1	????	A sixth comment

6.17 Kontextmenü: Edit entry

Eintrag "Edit entry...": Nachdem man eine Zeile ausgewählt und diese Funktion aus dem lokalen Kontextmenü gewählt hat oder aber einen Doppelklick auf eine Konfigurationszeile durchgeführt hat, erscheint folgender untenstehender Bildschirm. Im oberen Bereich der Tabelle wird ein Editierbereich eingeblendet. Angezeigt werden die Daten der ausgewählten Zeile aus der Konfigurationstabelle.

- Schaltfläche "Cancel": Wenn man diese Funktion auswählt, wird der Editiervorgang abgebrochen, alle Änderungen werden verworfen und der eingeblendete Editierbereich verschwindet wieder.
- Schaltfläche "OK": Wenn man diese Funktion wählt, werden alle Änderungen an den Konfigurationsdaten für die aktuelle Zeile in der Zuordnungstabelle gespeichert und der eingeblendete Eingabebereich geschlossen.

HINWEIS: Vergessen Sie nicht, dass diese Änderungen zuerst in das Gateway geladen werden, bevor diese Änderungen auch im Gateway wirksam sind!

KNX Test Bench

MODBUS/RTU

Register: 5 Datatype: SINT16 Interval: 0 Factor: 10 [Cancel] [OK]

KNX

Group: 1.1.9 Datatype: FLOAT16 Direction: READ Comment: F1.09 Aussentemperatur

MODBUS register	MODBUS datatype	MODBUS interval	KNX group	KNX datatype	KNX direction	Factor	Value	Comment
4x1	SINT16	0	1.1.3	FLOAT16	READ	10	????	F1.03 VL-Pelletsessel
4x2	SINT16	0	1.1.4	FLOAT16	READ	10	????	F1.04 RL-Pelletsessel
4x3	SINT16	0	1.1.7	FLOAT16	READ	10	????	F1.07 VL-Pelletsessel
4x4	SINT16	0	1.1.8	FLOAT16	READ	10	????	F1.08 RL-Pelletsessel
4x5	SINT16	0	1.1.9	FLOAT16	READ	10	????	F1.09 Aussentemperatur
4x6	SINT16	0	1.2.1	FLOAT16	READ	10	????	F2.01 Pufferspeicher 1
4x7	SINT16	0	1.2.6	FLOAT16	READ	10	????	F2.06 Pufferspeicher 2
4x8	SINT16	0	1.2.11	FLOAT16	READ	10	????	F2.11 Pufferspeicher 3
4x9	SINT16	0	1.2.17	FLOAT16	READ	10	????	F2.17 Pufferspeicher 4
4x10	SINT16	0	1.2.22	FLOAT16	READ	10	????	F2.18 Pufferspeicher 5
4x11	SINT16	0	1.3.1	FLOAT16	READ	10	????	F3.01 VL-Solarsystem
4x12	SINT16	0	1.3.2	FLOAT16	READ	10	????	F3.02 RL-Solarsystem
4x13	SINT16	0	1.4.1	FLOAT16	READ	10	????	F4.01 VL-Heizsystem
4x14	SINT16	0	1.4.2	FLOAT16	READ	10	????	F4.02 VL-Heizsystem
4x15	SINT16	0	1.4.3	FLOAT16	READ	10	????	F4.03 RL-Heizsystem
4x16	SINT16	0	1.4.4	FLOAT16	READ	10	????	F4.04 RL-Zirkulation
4x17	UINT16	0	10.3.5	BIT	READ	1	????	V3.01 Ventil unterer WT
4x18	UINT16	0	10.3.6	BIT	READ	1	????	V3.02 Ventil unterer WT
4x19	SINT32	0	9.3.4	UINT32	READ	0,001	????	Z3.01.01 WMZ RL-Solar Q
4x20	SINT32	0	9.3.6	UINT32	READ	0,001	????	Z3.01.02 WMZ RL-Solar V
4x21	SINT16	0	1.3.1	FLOAT16	READ	10	????	Z3.01.03 WMZ T-VL
4x24	SINT16	0	1.3.2	FLOAT16	READ	10	????	Z3.01.04 WMZ T-RL
4x25	SINT32	0	9.3.2	UINT32	READ	0,001	????	Z3.01.05 WMZ RL-Solar P
4x27	SINT32	0	9.3.5	UINT32	READ	0,001	????	Z3.01.06 WMZ RL-Solar dV
4x29	SINT32	0	9.4.29	UINT32	READ	0,001	????	Z4.01.01 WMZ RL-Heizsystem Q
4x31	SINT32	0	9.4.30	UINT32	READ	0,001	????	Z4.01.02 WMZ RL-Heizsystem V
4x33	SINT16	0	1.4.2	FLOAT16	READ	10	????	Z4.01.03 WMZ T-VL
4x34	SINT16	0	1.4.3	FLOAT16	READ	10	????	Z4.01.04 WMZ T-RL
4x35	SINT32	0	9.4.21	UINT32	READ	0,001	????	Z4.01.05 WMZ RL-Heizsystem P
4x37	SINT32	0	9.4.31	UINT32	READ	0,001	????	Z4.01.06 WMZ RL-Heizsystem dV
4x39	SINT32	0	9.4.32	UINT32	READ	0,001	????	Z4.02.01 WMZ VL-Zirkulation Q
4x41	SINT32	0	9.4.34	UINT32	READ	0,001	????	Z4.02.02 WMZ VL-Zirkulation V

Hier ist eine Vergrößerung des Editierbereichs:

MODBUS/RTU

Register: 5 Datatype: SINT16 Interval: 0 Factor: 10 [Cancel] [OK]

KNX

Group: 1.1.9 Datatype: FLOAT16 Direction: READ Comment: F1.09 Aussentemperatur

Der Editierbereich wird in zwei Bereiche zerteilt:

Bereich "MODBUS/RTU": Hier finden Sie alle Eingabefelder, welche die Art und Weise der verwendeten MODBUS Holdingregister festlegen.

- Feld "Register": Geben Sie einen gültigen Startindex für die MODBUS Holding Register im Bereich 1 bis 65535 an. Wie viele MODBUS Holding Register durch diesen Konfigurationseintrag verwendet werden, wird durch den MODBUS/RTU Datentyp festgelegt.
- Feld "Datatype": Wählen Sie einen der möglichen Datentypen aus der Auswahlliste. Dieser Datentyp definiert auf der einen Seite wie viele MODBUS Holding Register tatsächlich für diesen Eintrag verwendet werden (z.B.: benötigt der Datentyp UINT16 ein Holdingregister, hingegen benötigt der Datentyp FLOAT32 zwei aufeinanderfolgende MODBUS Holding Register), und auf der anderen Seite definiert dieser Datentyp, wie die KNX Daten in diesen Registern abgelegt werden (z.B.: speichert der Datentyp FLOAT32 das höherwertige 16-Bit Wort in das erste MODBUS Holding Register und das niederwertige 16-Bit Wort in das zweite Holding Register, der Datentyp FLOAT32R dreht diese Speicherung um: Hier steht das niederwertige 16-Bit Wort des 32-Bit Werts im ersten Holdingregister und das höherwertige 16-Bit Wort im zweiten Holdingregister).

MODBUS DATENTYP	GRÖSSE	WORT REIHENFOLGE	BESCHREIBUNG
ERR	Keine	Keine	Definiert einen ungültigen Konfigurationseintrag und wird vom Gateway ignoriert
UINT16	16 Bit 1 Register	Keine	Definiert einen 16 Bit vorzeichenlosen Ganzzahlwert im Bereich von 0 bis 65535 oder 0x0000 bis 0xFFFF
SINT16	16 Bit 1 Register	Keine	Definiert einen 16 Bit vorzeichenbehafteten Ganzzahlwert im Bereich von -32768 bis +32767 oder 0x8000 bis 0x7FFF
UINT32	32 Bit 2 Register	0:High Word 1:Low Word	Definiert einen 32 Bit vorzeichenlosen Ganzzahlwert im Bereich von 0 bis 4.294.967.295 oder 0x00000000 bis 0xFFFFFFFF
SINT32	32 Bit 2 Register	0:High Word 1:Low Word	Definiert einen 32 Bit vorzeichenbehafteten Ganzzahlwert im Bereich von -2.147.483.648 bis +2.147.483.647 oder 0x80000000 bis 0x7FFFFFFF
UINT32R	32 Bit 2 Register	0:Low Word 1:High Word	Definiert einen 32 Bit vorzeichenlosen Ganzzahlwert im Bereich von 0 bis 4.294.967.295 oder 0x00000000 bis 0xFFFFFFFF mit umgekehrter Wortfolge
SINT32R	32 Bit 2 Register	0:Low Word 1:High Word	Definiert einen 32 Bit vorzeichenbehafteten Ganzzahlwert im Bereich von -2.147.483.648 bis +2.147.483.647 oder 0x80000000 bis 0x7FFFFFFF mit umgekehrter Wortfolge
FLOAT32	32 Bit 2 Register	0:High Word 1:Low Word	Definiert eine 32 Bit Fließkommazahl im Bereich von $\pm 1.4 \cdot 10^{-45}$ bis $\pm 3.403 \cdot 10^{38}$. Eine Mantisse mit 23 Bit und ein Exponent mit 8 Bit werden verwendet. Diese Zahl kann 7 bis 8 Nachkommastellen speichern.
FLOAT32R	32 Bit 2 Register	0:Low Word 1:High Word	Definiert eine 32 Bit Fließkommazahl im Bereich von $\pm 1.4 \cdot 10^{-45}$ bis $\pm 3.403 \cdot 10^{38}$. Eine Mantisse mit 23 Bit und ein Exponent mit 8 Bit werden verwendet. Diese Zahl kann 7 bis 8 Nachkommastellen speichern. Die beiden 16 Bit Worte werden in umgekehrter Wortfolge gespeichert.
DOUBLE64	64 Bit 4 Register	0: Highest Word 1: Higher Word 2: Lower Word 3: Lowest Word	Definiert eine 64 Bit Fließkommazahl im Bereich von $\pm 4.24 \cdot 10^{-324}$ bis $\pm 1.798 \cdot 10^{308}$. Eine Mantisse mit 52 Bit und ein Exponent mit 11 Bit werden verwendet. Diese Zahl kann 15 bis 16 Nachkommastellen speichern.
DOUBLE64R	64 Bit 4 Register	0: Lowest Word 1: Lower Word 2: Higher Word 3: Highest Word	Definiert eine 64 Bit Fließkommazahl im Bereich von $\pm 4.24 \cdot 10^{-324}$ bis $\pm 1.798 \cdot 10^{308}$. Eine Mantisse mit 52 Bit und ein Exponent mit 11 Bit werden verwendet. Diese Zahl kann 15 bis 16 Nachkommastellen speichern. Die vier 16 Bit Worte werden in umgekehrter Wortfolge gespeichert.
GENERIC	64 Bit 4 Register	0: ERSTES und ZWEITES Byte 1: DRITTES und VIERTES Byte 2: FÜNFTES und SECHSTES Byte 3: SIEBENTES und ACHTES Byte	Derzeit unbenutzt
STRING	64 Bit 4 Register	0: ERSTES und ZWEITES Byte 1: DRITTES und VIERTES Byte 2: FÜNFTES und SECHSTES Byte 3: SIEBENTES und ACHTES Byte	Derzeit unbenutzt

- Feld "Interval": Dieses Feld hat im Moment keine Funktion. In Zukunft wird mit diesem Eintrag ein Zeitintervall in Sekunden definiert, dass im Gateway zum zyklischen Abfragen des Konfigurationseintrags am KNX Bus verwendet wird.
- Feld "Factor": Dieses Feld definiert einen Ganzzahlwert Faktor, welcher zur Umrechnung der KNX Werte in MODBUS Register beim Senden und beim Empfangen von KNX Telegrammen verwendet wird.

Im Falle des Empfangs von KNX Daten wird folgende Umrechnung vorgenommen:
 MODBUS Wert= KNX Wert multipliziert mit Faktor

Im Falle des Versendens von KNX Daten wird folgende Umrechnung vorgenommen:
 KNX Wert=MODBUS Wert dividiert durch Faktor

Im Falle des KNX Datentyps GENERIC oder STRING definiert dieser Faktor des Starindex, ab wo die Datenbytes aus den KNX Daten gelesen werden. Die Maximalanzahl von Datenbytes in einem KNX Telegramm darf ja 14 Bytes sein.

Ein Faktor 0 wird vom Gateway ignoriert.

Bereich "KNX": Hier sind alle Eingabefelder, die für das Konvertieren des KNX Telegramms notwendig sind.

- Feld "Group": Hiermit wird die KNX Gruppenadresse für diese Konfigurationszeile im Bereich von 0.0.0 bis 15.7.255 festgelegt.
- Feld "Datatype": Hier wird der Datentyp des zu empfangenden oder zu sendenden KNX Telegramms festgelegt.

KNX DATENTYP	GRÖSSE	BESCHREIBUNG
ERR	Keine	Definiert einen ungültigen Konfigurationseintrag, Dieser wird vom Gateway ignoriert.
BIT	1 Bit	Definiert einen Ganzzahlwert im Bereich von 0 bis 1 oder 0x0 bis 0x1. Oftmals mit AUS und EIN interpretiert.
TWOBITS	2 Bits	Definiert eine Ganzzahl bestehend aus zwei Bits mit dem Zahlenbereich 0 bis 3 oder 0x0 bis 0x3
FOURBITS	4 Bits	Definiert eine Ganzzahl bestehend aus vier Bits mit dem Zahlenbereich 0 bis 15 oder 0x0 bis 0xF
SIXBITS	6 Bits	Definiert eine Ganzzahl bestehend aus sechs Bits mit dem Zahlenbereich 0 bis 63 oder 0x00 bis 0x3F
CHARACTER	8 Bits	Definiert ein Textzeichen bestehend aus 8 Bits mit dem Zahlenbereich von 0 bis 255 oder 0x00 bis 0xFF. Bitte konsultieren Sie die KNX Dokumentation wie die Codierung der Textzeichen am KNX durchgeführt wird. Es wird ein ASCII Zeichensatz und ein ISO 8859.1 Zeichensatz am KNX verwendet.
UINT8	8 Bits	Definiert eine 8 Bit vorzeichenlose Ganzzahl im Bereich von 0 bis 255 oder 0x00 bis 0xFF
SINT8	8 Bits	Definiert eine 8 Bit vorzeichenbehaftete Ganzzahl im Bereich von -128 bis +127 oder 0x80 bis 0x7F
UINT16	16 Bits	Definiert eine 16 Bit vorzeichenlose Ganzzahl im Bereich von 0 bis 65535 oder 0x0000 bis 0xFFFF
SINT16	16 Bits	Definiert eine 16 Bit vorzeichenbehaftete Ganzzahl im Bereich von -32768 bis +32767 oder 0x8000 bis 0x7FFF
UINT32	32 Bits	Definiert eine 32 Bit vorzeichenlose Ganzzahl im Bereich von 0 bis 4.294.967.295 oder 0x00000000 bis 0xFFFFFFFF
SINT32	32 Bits	Definiert eine 32 Bit vorzeichenbehaftete Ganzzahl im Bereich von -2.147.483.648 bis +2.147.483.647 oder 0x80000000 bis 0x7FFFFFFF
FLOAT16	16 Bits	Definiert eine 16 Bit Fließkommazahl mit einem 4 Bit Exponenten und einer 12 Bit Mantisse. <div style="text-align: center;"> $2_{MSB} \qquad 1_{LSB}$ </div> <div style="text-align: center;"> </div> $\text{FloatValue} = (0,01 * M) * 2^{(E)}$ <p>E = [0 ... 15] M = [-2 048 ... 2 047], two's complement notation</p> <p>For all Datapoint Types 9.xxx, the encoded value 7FFFh shall always be used to denote invalid data. [-671 088,64 ... 670 760,96]</p>

Proprietary data, company confidential. All rights reserved.
 Confidantia de date, compania confidențială. Toți drepturile rezervate.
 Comunicado como secreto empresarial. Reservados todos los derechos.
 Comunicado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verbreitung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich anders angegeben. Alle Rechte vorbehalten. Inanspruchnahme der Patente durch die RESI AG.
 Further disclosure and reproduction of this document, distribution and disclosure of its content not permitted unless otherwise expressly stated. All rights reserved. Patent claims by RESI AG.

FLOAT32	32 Bits	Definiert eine 32 Bit Fließkommazahl im Bereich von $\pm 1.4 \cdot 10^{-45}$ bis $\pm 3.403 \cdot 10^{38}$. Eine Mantisse von 23 Bits und ein Exponent mit 8 Bits werden verwendet. Es können 7 bis 8 Nachkommastellen dargestellt werden. $4_{\text{MSB}} \quad \quad \quad 3 \quad \quad \quad 2 \quad \quad \quad 1_{\text{LSB}}$ <table border="1" style="width: 100%; text-align: center;"> <tr> <td style="width: 10%;">S</td> <td style="width: 20%;">Exponent</td> <td style="width: 60%;">Fraction</td> </tr> <tr> <td>FFFFFFFFFF</td> <td>FFFFFFFF</td> <td>FFFFFFFFFFFF</td> </tr> </table> <p>The values are encoded in the IEEE floating point format according IEEE 754 single precision format.</p> <p>NOTE 7 This specifies that the exponent is biased. This allows negative exponent values.</p> <p>S (Sign) = {0,1} Exponent = [0 ... 255] Fraction = [0 ... 8 388 607]</p> <p>The resolution is given by the use of the IEEE 754 format and varies with the used exponent.</p>	S	Exponent	Fraction	FFFFFFFFFF	FFFFFFFF	FFFFFFFFFFFF																																															
S	Exponent	Fraction																																																					
FFFFFFFFFF	FFFFFFFF	FFFFFFFFFFFF																																																					
TIME	24 Bits	Definiert einen 24 Bit Wert mit einer Zeit, die wie folgt kodiert wird: $3_{\text{MSB}} \quad \quad \quad 2 \quad \quad \quad 1_{\text{LSB}}$ <table border="1" style="width: 100%; text-align: center;"> <tr> <td>000</td> <td>Day</td> <td>0000</td> <td>Month</td> <td>0</td> <td>Year</td> </tr> <tr> <td>r r r</td> <td>UUUUU</td> <td>r r r r</td> <td>UUUU</td> <td>r</td> <td>UUUUUUUU</td> </tr> </table> <table border="1" style="width: 100%; text-align: center;"> <thead> <tr> <th>Field:</th> <th>Encoding:</th> <th>Range:</th> <th>Unit:</th> <th>Resol.:</th> </tr> </thead> <tbody> <tr> <td>Day</td> <td>1 = Monday ... 7 = Sunday 0 = no day</td> <td>[0...7]</td> <td>none</td> <td>none</td> </tr> <tr> <td>Hour</td> <td>binary encoded</td> <td>[0...23]</td> <td>hours</td> <td>h</td> </tr> <tr> <td>Minutes</td> <td>binary encoded</td> <td>[0...59]</td> <td>minutes</td> <td>min</td> </tr> <tr> <td>Seconds</td> <td>binary encoded</td> <td>[0...59]</td> <td>seconds</td> <td>s</td> </tr> </tbody> </table>	000	Day	0000	Month	0	Year	r r r	UUUUU	r r r r	UUUU	r	UUUUUUUU	Field:	Encoding:	Range:	Unit:	Resol.:	Day	1 = Monday ... 7 = Sunday 0 = no day	[0...7]	none	none	Hour	binary encoded	[0...23]	hours	h	Minutes	binary encoded	[0...59]	minutes	min	Seconds	binary encoded	[0...59]	seconds	s																
000	Day	0000	Month	0	Year																																																		
r r r	UUUUU	r r r r	UUUU	r	UUUUUUUU																																																		
Field:	Encoding:	Range:	Unit:	Resol.:																																																			
Day	1 = Monday ... 7 = Sunday 0 = no day	[0...7]	none	none																																																			
Hour	binary encoded	[0...23]	hours	h																																																			
Minutes	binary encoded	[0...59]	minutes	min																																																			
Seconds	binary encoded	[0...59]	seconds	s																																																			
DATE	24 Bits	Definiert einen 24 Bit Wert mit einem Datum, die wie folgt kodiert wird: $3_{\text{MSB}} \quad \quad \quad 2 \quad \quad \quad 1_{\text{LSB}}$ <table border="1" style="width: 100%; text-align: center;"> <tr> <td>Day</td> <td>Hour</td> <td>00</td> <td>Minutes</td> <td>00</td> <td>Seconds</td> </tr> <tr> <td>NNN</td> <td>UUUUU</td> <td>r r</td> <td>UUUUUU</td> <td>r r</td> <td>UUUUUUUU</td> </tr> </table> <table border="1" style="width: 100%; text-align: center;"> <thead> <tr> <th>Field:</th> <th>Range:</th> <th>Unit:</th> <th>Resol.:</th> </tr> </thead> <tbody> <tr> <td>Day</td> <td>[1...31]</td> <td>Day of month</td> <td>1 day</td> </tr> <tr> <td>Month</td> <td>[1...12]</td> <td>Month</td> <td>1 month</td> </tr> <tr> <td>Year</td> <td>[0...99]</td> <td>Year</td> <td>1 year</td> </tr> </tbody> </table>	Day	Hour	00	Minutes	00	Seconds	NNN	UUUUU	r r	UUUUUU	r r	UUUUUUUU	Field:	Range:	Unit:	Resol.:	Day	[1...31]	Day of month	1 day	Month	[1...12]	Month	1 month	Year	[0...99]	Year	1 year																									
Day	Hour	00	Minutes	00	Seconds																																																		
NNN	UUUUU	r r	UUUUUU	r r	UUUUUUUU																																																		
Field:	Range:	Unit:	Resol.:																																																				
Day	[1...31]	Day of month	1 day																																																				
Month	[1...12]	Month	1 month																																																				
Year	[0...99]	Year	1 year																																																				
DATETIME	64 Bits	Definiert einen 64 Bit Wert mit einem Datum und einer Zeit, die wie folgt kodiert werden: $8_{\text{MSB}} \quad \quad \quad 7 \quad \quad \quad 6 \quad \quad \quad 5$ <table border="1" style="width: 100%; text-align: center;"> <tr> <td style="width: 15%;">Year</td> <td style="width: 15%;">0 0 0 0</td> <td style="width: 15%;">Month</td> <td style="width: 15%;">0 0 0</td> <td style="width: 15%;">DayOf-Week</td> <td style="width: 15%;">HourOfDay</td> </tr> <tr> <td>UUUUUUUU</td> <td>r r r r</td> <td>UUUUU</td> <td>r r r</td> <td>UUUUU</td> <td>UUUUUUUU</td> </tr> </table> $4 \quad \quad \quad 3 \quad \quad \quad 2 \quad \quad \quad 1_{\text{LSB}}$ <table border="1" style="width: 100%; text-align: center;"> <tr> <td style="width: 15%;">0 0</td> <td style="width: 15%;">Minutes</td> <td style="width: 15%;">0 0</td> <td style="width: 15%;">Seconds</td> <td style="width: 15%;">L</td> <td style="width: 15%;">WD</td> <td style="width: 15%;">NWD</td> <td style="width: 15%;">NY</td> <td style="width: 15%;">ND</td> <td style="width: 15%;">NDoW</td> <td style="width: 15%;">NT</td> <td style="width: 15%;">SUT</td> <td style="width: 15%;">CLQ</td> <td style="width: 15%;">0</td> </tr> <tr> <td>r r</td> <td>UUUUUU</td> <td>r r</td> <td>UUUUUU</td> <td>B</td> <td>r</td> <td>r</td> <td>r</td> <td>r</td> <td>r</td> <td>r</td> <td>r</td> </tr> </table>	Year	0 0 0 0	Month	0 0 0	DayOf-Week	HourOfDay	UUUUUUUU	r r r r	UUUUU	r r r	UUUUU	UUUUUUUU	0 0	Minutes	0 0	Seconds	L	WD	NWD	NY	ND	NDoW	NT	SUT	CLQ	0	0	0	0	0	0	0	r r	UUUUUU	r r	UUUUUU	B	B	B	B	B	B	B	B	B	B	r	r	r	r	r	r	r
Year	0 0 0 0	Month	0 0 0	DayOf-Week	HourOfDay																																																		
UUUUUUUU	r r r r	UUUUU	r r r	UUUUU	UUUUUUUU																																																		
0 0	Minutes	0 0	Seconds	L	WD	NWD	NY	ND	NDoW	NT	SUT	CLQ	0	0	0	0	0	0	0																																				
r r	UUUUUU	r r	UUUUUU	B	B	B	B	B	B	B	B	B	B	r	r	r	r	r	r	r																																			

Field	Description	Encoding	Range	Unit	Resol.:
Year	Year	Value binary encoded, offset 1900 0 = 1900 255 = 2155	[0...255]	year	1 year
Month	Month	Value binary encoded 1 = January ... 12 = December	[1...12]	Month	1 month
DayOfMonth	D	Value binary encoded 1 = 1st day 31 = 31st day	[1...31]	none	none
DayOfWeek	Day of week	Value binary encoded 0 = any day 1 = Monday ... 7 = Sunday	[0...7]	none	none
HourOfDay	Hour of day	Value binary encoded.	[0...24]	h	1 h
Minutes	Minutes	Value binary encoded.	[0...59]	min	1 min
Seconds	Seconds	Value binary encoded.	[0...59]	s	1 s
F	Fault	0 = Normal (No fault) 1 = Fault	{0,1}	none	none
WD	Working Day	0 = Bank day (No working day) 1 = Working day	{0,1}	none	none
NWD	No WD	0 = WD field valid 1 = WD field not valid	{0,1}	none	none
NY	No Year	0 = Year field valid 1 = Year field not valid	{0,1}	none	none
ND	No Date	0 = Month and Day of Month fields valid 1 = Month and Day of Month fields not valid	{0,1}	none	none

STRING	Max. 14 Bytes	Definiert bis zu 14 Bytes Textdaten <div style="text-align: center;"> 14_{MSB} ... 1_{LSB} </div> <div style="display: flex; justify-content: space-around; align-items: center; margin-top: 5px;"> <div style="border: 1px solid black; padding: 2px;">Character 1</div> ... <div style="border: 1px solid black; padding: 2px;">Character 14</div> </div> <div style="display: flex; justify-content: space-around; align-items: center; margin-top: 5px;"> <div style="border: 1px solid black; padding: 2px;">AAAAA</div> ... <div style="border: 1px solid black; padding: 2px;">AAAAA</div> </div> <p>These Datapoint Types are used to transmit strings of textual characters. The length is fixed to 14 octets. The contents are filled starting from the most significant octet. Each octet shall be encoded as specified for the chosen character set, as defined in clause 0. If the string to be transmitted is smaller than 14 octets, unused trailing octets in the character string shall be set to NULL (00h).</p> <p><u>Example:</u> 'KNX is OK' is encoded as follows : 4B 4E 58 20 69 73 20 4F 4B 00 00 00 00 00</p>
GENERIC	64 Bits	Definiert einen 64 Bit Wert, welcher bis zu 8 Bytes aus einem generischen KNX Datentelegramm darstellt. Dadurch, dass ein KNX Datentelegramm bis zu 14 Datenbytes besitzen kann, spezifiziert das Feld Factor den Startindex, ab wo die maximal 8 Datenbytes entnommen werden im Bereich von 0 bis 13. Das System speichert das erste Byte in den unteren 8 Bits des ersten MODBUS Registers, das zweite Byte wird in den oberen 8 Bits des selben Registers abgelegt, das dritte Byte wieder in den unteren 8 Bits des nachsten Registers usw.

- Feld "Richtung": Legt dir Richtung der KNX Kommunikation am Bus fur diesen Eintrag fest. Wahlen Sie READ fur nur ankommende KNX Telegramme, WRITE fur nur ausgehende KNX Telegramme und READ_WRITE fur ankommende und ausgehende KNX Telegramme. ERR definiert eine ungultige Konfiguration und wird vom Gateway ignoriert.
- Feld "Comment": Geben Sie einen Kommentar ein, um Ihre KNX MODBUS Zuordnung zu dokumentieren. Beachten Sie, dass der Kommentar nur am PC aber nicht im Gateway gespeichert wird! Wenn Sie eine Konfiguration aus einem Gateway hochladen, geht dieser Kommentar verloren.

6.18 Die Konfiguration testen

Wenn man die aktuelle Konfiguration in das Gateway lädt und den Testmodus über die Schaltfläche „Test“ startet, erhält man die folgende Anzeige. Das System updatet alle MODBUS Register ca. alle 5 Sekunden.

The screenshot shows a software interface with several sections:

- Local Com-Port settings:** Modbus unit: 255, Device: COM8, Baudrate: 57600, Parity: NONE, IP-Address: [empty], Port: [empty].
- Common:** Download config, Test connection, Test. Device name: RESI-KNX-MODBUS, Device type: KNX to MODBUS/RTU module, Software version: 1.0.1, State: no error.
- Device specific:** Upload config. Modbus address: 255, Modbus parity: NONE, KNX address: 15.15.255, HELP.
- KNX Test Bench:** A table with columns: MODBUS register, MODBUS datatype, MODBUS interval, KNX group, KNX datatype, KNX direction, Factor, Value, Comment.

MODBUS register	MODBUS datatype	MODBUS interval	KNX group	KNX datatype	KNX direction	Factor	Value	Comment
4x1	UINT16	0	1.7.10	BIT	WRITE	1	0x0000,0	KNX actuator #1 Output #1
4x2	UINT16	0	1.7.11	BIT	WRITE	1	0x0000,0	KNX actuator #1 Output #2
4x3	UINT16	0	1.7.12	BIT	WRITE	1	0x0000,0	KNX actuator #1 Output #3
4x4	UINT16	0	1.7.13	BIT	WRITE	1	0x0000,0	KNX actuator #1 Output #4

Um einen neuen Wert für eine Konfigurationszeile zu setzen, doppelklicken Sie eine Konfigurationszeile. Ein Eingabefenster erscheint, in dem Sie den neuen Wert eingeben können. Bestätigen Sie dieses Fenster mit JA, so erzeugt die Software die notwendigen MODBUS Schreibbefehle, um alle MODBUS Register des Konfigurationseintrags mit dem neuen Wert zu beschreiben.

The screenshot shows the same MODBUS register table as above. A dialog box titled "SET NEW VALUES" is overlaid on the table. The dialog box contains the text "Enter a new value to set this modbus registers" and a text input field with the value "1". There are "OK" and "Abbrechen" buttons at the bottom of the dialog box.

Wenn die KNX Richtung auch WRITE enthält, versendet das Gateway sofort das entsprechende KNX Telegramm am KNX Bus, wenn alle notwendigen MODBUS Register beschrieben wurden.

Proprietary data, company confidential. All rights reserved. Conflicto de secreto empresarial. Reservados todos los derechos. Comunicado como secreto empresarial. Reservados todos los derechos.

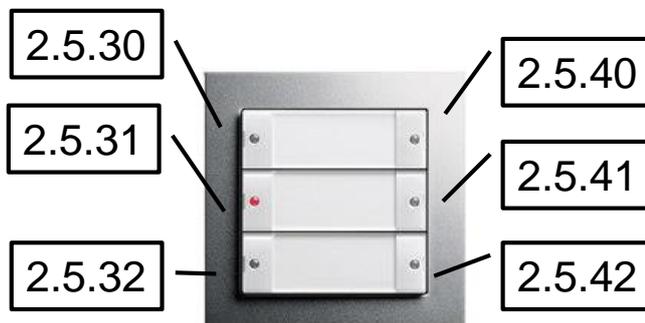
Weitergabe sowie Vervielfältigung dieser Unterlage, Wertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich anders angegeben. Alle Rechte vorbehalten. Inanspruchnahme der Haftung für Schäden aus dem Gebrauch dieser Unterlage ist ausgeschlossen.

7 Beispielkonfigurationen

Hier finden Sie ein paar Beispielkonfigurationen, um zu erklären, wie unser Gateway arbeitet und welches Konfigurationsprinzip angewendet wird.

7.1 Einlesen des Status eines KNX Tasters

Nehmen wir das folgende Setup an: 1 KNX Taster Modul z.B. GIRA mit sechs Tasten, welche mit folgenden KNX Gruppenadressen belegt wurden:



Alle sechs KNX Gruppenadressen senden als Datentyp ein BIT, welches den aktuellen Schalterzustand darstellt (0=AUS, 1=EIN). Wenn man den linken oberen Taster drückt, sendet das KNX Gerät das KNX Telegramm 2.5.30=1 oder 2.5.30=0, abhängig vom internen Schalterzustand im KNX Gerät.

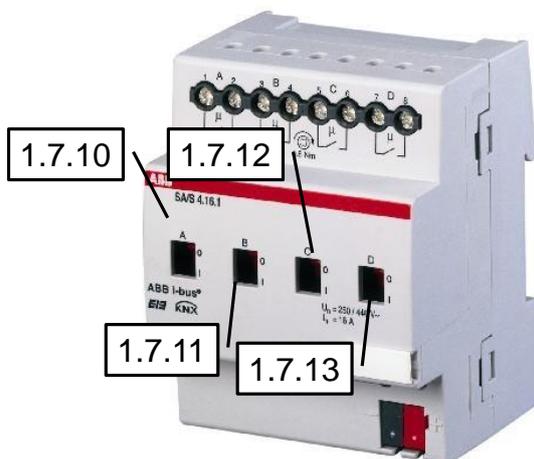
Die Konfiguration wird wie folgt aussehen:

MODBUS register	MODBUS datatype	MODBUS interval	KNX group	KNX datatype	KNX direction	Factor	Value	Comment
4x1	UINT16	0	2.5.30	BIT	READ	1	0x0001,1	KNX Device #1 Switch #1
4x2	UINT16	0	2.5.31	BIT	READ	1	0x0000,0	KNX Device #1 Switch #2
4x3	UINT16	0	2.5.32	BIT	READ	1	0x0000,0	KNX Device #1 Switch #3
4x4	UINT16	0	2.5.40	BIT	READ	1	0x0000,0	KNX Device #1 Switch #4
4x5	UINT16	0	2.5.41	BIT	READ	1	0x0000,0	KNX Device #1 Switch #5
4x6	UINT16	0	2.5.42	BIT	READ	1	0x0001,1	KNX Device #1 Switch #6

Wie man im Testmodus erkennt, sind die Schalter Switch #1 und #6 EIN. Alle übrigen Schalter sind AUS. Drücken Sie alle sechs Taster am KNX Schalter und prüfen Sie, wie sich der Inhalt der MODBUS Register ändert.

7.2 Einen KNX Aktor beschreiben

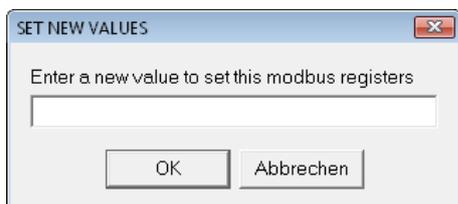
Es ist auch sehr einfach einen KNX Aktor mit vier Ausgängen, z.B. einen ABB KNX Aktor, zu beschreiben. Nehmen wir folgende KNX Gruppenadressen für die vier Ausgänge an. Jeder von diesen Ausgängen erwartet ein KNX Telegramm mit Bit Daten.



Die korrekte Konfiguration schaut wie folgt aus:

MODBUS register	MODBUS datatype	MODBUS interval	KNX group	KNX datatype	KNX direction	Factor	Value	Comment
4x1	UINT16	0	1.7.10	BIT	WRITE	1	0x0000,0	KNX actuator #1 Output #1
4x2	UINT16	0	1.7.11	BIT	WRITE	1	0x0000,0	KNX actuator #1 Output #2
4x3	UINT16	0	1.7.12	BIT	WRITE	1	0x0000,0	KNX actuator #1 Output #3
4x4	UINT16	0	1.7.13	BIT	WRITE	1	0x0000,0	KNX actuator #1 Output #4

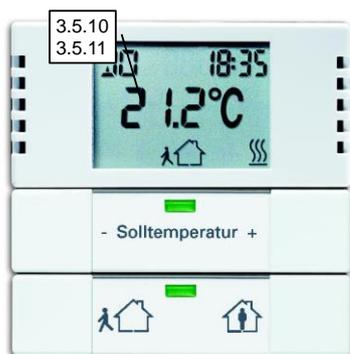
Laden Sie diese Konfiguration in das Gateway und starten Sie den Testmodus. Um den Zustand des ersten Ausgangs zu ändern, doppelklicken Sie auf die erste Konfigurationszeile. Es öffnet sich folgendes Eingabefenster:



Geben Sie nun einen neuen Wert für den Ausgang ein, z.B. 1, und betätigen Sie die OK Schaltfläche. Sofort schaltet der Aktor am KNX Bus in den Zustand EIN. Ein erneuter Doppelklick mit der Eingabe des Wertes 0 ändert den Zustand des Ausgangs auf AUS. Probieren Sie das für die übrigen drei Ausgängen ebenfalls.

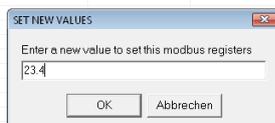
7.3 Analoge KNX Werte einlesen

Etwas komplizierter ist das Zuordnen von Analogwerten vom KNX Bus in MODBUS Register. Wir starten mit einem einfachen KNX Gerät, z.B. ein Raumcontroller. Wir nehmen an, dass dieses Gerät zyklisch die aktuelle Raumtemperatur mit der KNX Gruppenadresse 3.5.10 versendet. Das Gerät versendet/empfängt ein KNX Telegramm mit der KNX Gruppenadresse 3.5.11. Dies stellt den aktuellen Sollwert dar. Beide Werte werden mit dem KNX Datentyp 9.001 temperature (°C) am KNX Bus durch das KNX Gerät dargestellt.



Benutzen sie folgende Konfiguration als Beispiel:

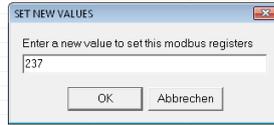
MODBUS register	MODBUS datatype	MODBUS interval	KNX group	KNX datatype	KNX direction	Factor	Value	Comment
4x1	FLOAT32	0	3.5.10	FLOAT16	READ	1	0x41BB851F,23.4400005340576	Actual temperature
4x3	FLOAT32	0	3.5.11	FLOAT16	READ-WRITE	1	0x41BC0000,23.5	New setpoint



Wie Sie erkennen können, ordnen wir den KNX Datentyp FLOAT16 (Das ist die interne KNX Darstellung des KNX Datentyps 9.001) dem MODBUS Datentyp FLOAT32 zu. Ein FLOAT32 Wert belegt zwei aufeinanderfolgende MODBUS Register. Das ist der Grund, warum der erste Wert mit Holding Registerindex 4x1 startet und der zweite Wert mit dem Holding Registerindex 4x3 startet. Somit belegen beide Werte insgesamt vier MODBUS Register 4x1, 4x2, 4x3 und 4x4.

Aber man kann auch eine Fließkommazahl vom KNX Bus auf eine Ganzzahl am MODBUS konvertieren. Wir erweitern die bestehende Konfiguration um folgende Zeilen:

MODBUS register	MODBUS datatype	MODBUS interval	KNX group	KNX datatype	KNX direction	Factor	Value	Comment
4x1	FLOAT32	0	3.5.10	FLOAT16	READ	1	0x41BD47AE.23.6599998474121	Actual temperature
4x3	FLOAT32	0	3.5.11	FLOAT16	READ-WRITE	1	0x41BC0000.23.5	New setpoint
4x5	UINT16	0	3.5.10	FLOAT16	READ	10	0x00EC.236	Actual tempertaur multiplied by 10
4x6	UINT16	0	3.5.11	FLOAT16	READ-WRITE	10	0x00EB.235	New setpoint multiplied by 10



Wie Sie bemerken, verwenden wir nun UINT16 als MODBUS Datentyp und eine Faktor 10 um die erste Kommastelle zu erhalten. Somit enthält das MODBUS Register 4x5 den Wert 236, wenn die aktuelle KNX Temperatur 23.66 vom KNX Bus empfangen wird.

Beachten Sie auch, dass Sie mehr als eine Konfigurationszeile mit derselben KNX Gruppenadresse verwenden können, um unterschiedliche Werte in unterschiedliche MODBUS Register darzustellen.

8 Funktionsbeschreibung

Die RESI-KNX-MODBUS und RESI-KNX-ASCII Gateways kommunizieren mit einem Host System mit dem MODBUS/RTU Slave Protokoll. Die Version RESI-KNX-ASCII hat zusätzlich ein ASCII Protokoll implementiert. Die Kommunikation erfolgt über eine RS485 Schnittstelle (Halb-Duplex) oder über eine RS232 Schnittstelle (Full-Duplex).

Für die Kommunikation via ASCII Texte werden ASCII Nachrichten mit einem speziellen Startzeichen # (0x23, 35dec) und spezielle Endezeichen (0x0d, 13dec oder CARRIAGE RETURN) vom Host an das Modul gesendet. Das Modul sendet seine Antworten ebenfalls mit diesem speziellen Start und Endezeichen. Siehe dazu weiter unten die ASCII Befehlsbeschreibung. Im ASCII Modus kann mit oder ohne Busnummer kommuniziert werden.

Für die Kommunikation über MODBUS/RTU stehen folgenden MODBUS Funktionen zur Verfügung:

- READ HOLDING REGISTER (Funktionscode: 3)
- PRESET SINGLE REGISTER (Funktionscode: 6)
- PRESET MULTIPLE REGISTERS (Funktionscode: 16)

Hinweis:

Die Funktionen READ HOLDING REGISTER und PRESET MULTIPLE REGISTERS sind auf max. 125 Register pro Anfrage beschränkt!

8.1 ASCII Protokollbeschreibung

8.1.1 Übersicht

Das IO Modul kommuniziert mit ganz einfachen ASCII Befehlen. Folgende Sonderzeichen werden in dieser Beschreibung verwendet:

steht für das **Kanalgitter** ASCII Zeichen 35dec oder 0x23

: steht für den **Doppelpunkt** ASCII Zeichen 58dec oder 0x3A

= steht für das **Gleichheitszeichen** mit dem ASCII Code 61ec oder 0x3D

- steht für das **Minuszeichen** mit dem ASCII Code 45dec oder 0x2D

, steht für den **Beistrich** mit dem ASCII Code 44dec oder 0x2C

<CR> steht für das **CARRIAGE RETURN** ASCII Zeichen 13dec oder 0x0D. Im folgende wird das als **CR** dargestellt.

<SP> steht für **SPACE** oder das **LEERZEICHEN**. Das ist das Leerzeichen im ASCII Code 32dec oder 0x20. Im Folgenden wird das Leerzeichen als □ dargestellt.

<ADR> wird im Folgenden für die **Busadresse** verwendet. Diese kann dezimal oder hexadezimal übertragen werden und wird mit einem Beistrich (ASCII Zeichen 44dec oder 0x2C) vom nachfolgenden Befehl getrennt. Hexadezimalzahlen beginnen immer mit 0x. Es dürfen nur die ASCII Zeichen ‚0‘-‚9‘ 48dec bis 57dec, 0x30-0x39 und ‚A‘ bis ‚F‘, 65dec bis 70dec, 0x41-0x46 verwendet werden. Jedes Modul reagiert immer auf die Broadcast Adresse 0 und auf seine eigene Busadresse. Über einen externen DIP Switch kann man schnell zwischen der fixen Busadresse 255 und der programmierten Busadresse wechseln. Siehe dazu die DIP Switch Beschreibung.

8.1.2 Kommunikationsablauf

Prinzipiell sendet das Modul von sich aus keine Zeichen. Die Kommunikation geht immer von Host aus. Wenn nur ein Modul auf einer Buslinie verwendet wird (zb. mit RS232 Schnittstelle), kann im Protokoll auf eine Busadresse verzichtet werden. Im RS485 Modus können aber mehrere Module auf einer RS485 Linie zusammengeschlossen werden. Dann ist eine Busadresse zwingend in der Kommunikation notwendig.

Der Befehlsaufbau sieht wie folgt aus:

Der Host sendet einen Befehl oder einen Befehl mit Parameter ohne eine Busadresse:

#<Befehl><CR> oder
#<Befehl>:<Parameter><CR>

Das Modul antwortet, wenn er sich angesprochen fühlt mit dem Telegramm:

#<Antwort><CR>

Mit Busadresse sendet der Host an das Modul folgendes:

#<ADR>,<Befehl><CR> oder
#<ADR>,<Befehl>:<Parameter><CR>

Das Modul antwortet dann mit:

#<ADR>,<Antwort><CR>

Die Busadresse liegt im Bereich von 1dec bis 255dec oder 0x00 bis 0xFF hexadezimal. Die Einstellung wird über unsere kostenlose Konfigurationssoftware MODBUSConfigurator vorgenommen. Jedes Modul reagiert immer auch auf die Busadresse 0 als Broadcast Adresse.

Für jeden Befehl gibt es zwei Schreibweisen. Eine Langversion und eine Kurzversion, damit man weniger senden muss. Beispielsweise kann man die Softwareversion mit dem Kommando VERSION oder mit dem Kommando VER abfragen.

8.1.3 VERSION abfragen

Dieser Befehl liefert die aktuelle Softwareversion des Moduls.

Host Langversion:

#VERSION<CR> oder
#<ADR>,VERSION<CR>

Host Kurzversion:

#VER<CR> oder
#<ADR>,VER<CR>

Antwort:

#VERSION:<HIGH>.<MED>.<LOW><CR> oder
#<ADR>,VERSION:<HIGH>.<MED>.<LOW><CR>

<HIGH>.<MED>.<LOW> stellt die aktuelle Softwareversion dar. zB: 3.0.0

Beispiele:

→ **#VERSION_{CR}**
← **#VERSION:3.0.0_{CR}**

Mit Broadcast Adresse in Dezimal und Langversion:

→ **#0,VERSION_{CR}**
← **#0,VERSION:3.0.0_{CR}**

Mit Broadcast Adresse in Hexadezimal und Kurzversion:

→ **#0x00,VER_{CR}**
← **#0x00,VERSION:3.0.0_{CR}**

Mit Busadresse 255 in Dezimal

→ **#255,VER_{CR}**
← **#255,VERSION:3.0.0_{CR}**

Mit Busadresse 255 in Hexadezimal

→ #0xFF,VERSION_{CR}

← #0xFF,VERSION:3.0.0_{CR}

Mit Busadresse 43 in Dezimal

→ #43,VER_{CR}

← #43,VERSION:3.0.0_{CR}

Mit Busadresse 43 in Hexadezimal

→ #0x2B,VER_{CR}

← #0x2B,VERSION:3.0.0_{CR}

8.1.4 Modul TYPE abfragen

Dieser Befehl liefert den aktuellen Typ des Moduls.

Host Langversion:

#TYPE<CR> oder

#<ADR>,TYPE<CR>

Host Kurzversion:

#TYP<CR> oder

#<ADR>,TYP<CR>

Antwort:

#TYPE:<TYP><CR> oder

#<ADR>,TYPE:<TYP><CR>

<TYP> stellt den aktuellen Typ des Moduls dar. Derzeit RESI-KNX-ASCII

Beispiele:

→ #TYPE_{CR}

← #TYPE:RESI-KNX-ASCII_{CR}

→ #255,TYP_{CR}

← #255,TYPE:RESI-KNX-ASCII_{CR}

Richtung	ASCII Befehl
Host	#<BusAdr>,GDIP _{CR} #<BusAdr>,GET <input type="checkbox"/> DIP _{CR}
Antwort	#<BusAdr>,GDIP:<DIPSwitchDec>,<DIPSwitchHex> _{CR} Retourniert die aktuelle Stellung des DIP Switches als Dezimalzahl und als Hexadezimalzahl DIPSwitchDec DIPSwitchHex Der aktuelle Wert des DIP Switches: Bit 0: DIP Switch 1 (=0:AUS, =1:EIN) Bit 1: DIP Switch 2 (=0:AUS, =1:EIN) Bit 2: DIP Switch 3 (=0:AUS, =1:EIN) Bit 3: DIP Switch 4 (=0:AUS, =1:EIN)
Host	#<BusAdr>,CC _{CR} #<BusAdr>,CLEAR <input type="checkbox"/> CONFIG _{CR}
Antwort	#<BusAdr>,OK _{CR} Dieses Kommando löscht die gesamte Konfigurationstabelle des Gateways im FLASH.
Host	#<BusAdr>,GCS _{CR} #<BusAdr>,GET <input type="checkbox"/> CONFIG <input type="checkbox"/> SIZE _{CR}
Antwort	#<BusAdr>,GCS:<ConfigSize> _{CR} Retourniert die aktuelle Anzahl von Konfigurationszeilen im internen FLASH des Gateways. ConfigSize Die aktuelle Anzahl von Konfigurationszeilen im Gateway
Host	#<BusAdr>,RC:<Index> _{CR} #<BusAdr>,READ <input type="checkbox"/> CONFIG:<Index> _{CR}
Antwort	#<BusAdr>,KNX:I:<ConfigurationLine> _{CR} Retourniert die angeforderte Konfigurationszeile <Index> Index Die Nummer der angeforderten Konfigurationszeile im Bereich von 1..n, wobei n die letzte Zeile in der Konfiguration darstellt. ConfigurationLine Siehe die Erklärungen unten für mehr Information
Host	#<BusAdr>,GC _{CR} #<BusAdr>,GET <input type="checkbox"/> CONFIG _{CR}
Antwort	Das Gateway sendet für jede Konfigurationszeile folgendes #<BusAdr>,KNX:I:<ConfigurationLine> _{CR} Am Ende der Konfigurationstabelle sendet das Gateway die Zeile #<BusAdr>,OK _{CR} Retourniert die komplette Konfigurationstabelle des Gateways ConfigurationLine Siehe die Erklärungen unten für mehr Information
Host	#<BusAdr>,AC:<AddConfigurationLine> _{CR} #<BusAdr>,ADD <input type="checkbox"/> CONFIG:<AddConfigurationLine> _{CR}
Antwort	Wenn die Konfigurationszeile korrekt war, antwortet das Gateway mit #<BusAdr>,OK:<Index> _{CR} Andernfalls retourniert das Gateway einen Fehler #<BusAdr>,ERR:<ErrorText> _{CR} Fügt eine Zeile zur bestehenden Konfigurationstabelle hinzu. AddConfigurationLine Siehe die Erklärungen unten für mehr Information Index Die Nummer der neu hinzugefügten Konfigurationszeile im Bereich von 1..n. ErrorText Ein Text, der den Fehler näher beschreibt

Richtung	ASCII Befehl				
Host	#<BusAdr>,RV:<Index>CR #<BusAdr>,READVALUE:<Index>CR				
Antwort	#<BusAdr>,RV:<FloatValue>CR				
	Retourniert den aktuellen Wert der angeforderten Konfigurationszeile <Index>				
	<table border="0"> <tr> <td style="padding-right: 20px;">Index</td> <td>Die Nummer der angeforderten Konfigurationszeile im Bereich von 1..n, wobei n für die letzte Zeile in der aktuellen Konfigurationstabelle steht.</td> </tr> <tr> <td>FloatValue</td> <td>Der aktuelle Wert in den MODBUS Registern formatiert als Fließkommazahl.</td> </tr> </table>	Index	Die Nummer der angeforderten Konfigurationszeile im Bereich von 1..n, wobei n für die letzte Zeile in der aktuellen Konfigurationstabelle steht.	FloatValue	Der aktuelle Wert in den MODBUS Registern formatiert als Fließkommazahl.
Index	Die Nummer der angeforderten Konfigurationszeile im Bereich von 1..n, wobei n für die letzte Zeile in der aktuellen Konfigurationstabelle steht.				
FloatValue	Der aktuelle Wert in den MODBUS Registern formatiert als Fließkommazahl.				
Host	#<BusAdr>,WV:<Index>=<FloatValue>CR #<BusAdr>,WRITEVALUE:<Index>=<FloatValue>CR				
Antwort	#<BusAdr>,OKCR				
	Schreibt einen neuen Wert in die definierte Konfigurationszeile <Index>. Wenn die Konfiguration das Versenden eines KNX Telegramms unterstützt, wird dieses unmittelbar an den KNX Bus gesendet.				
	<table border="0"> <tr> <td style="padding-right: 20px;">Index</td> <td>Die Nummer der angeforderten Konfigurationszeile im Bereich von 1..n, wobei n für die letzte Zeile in der aktuellen Konfigurationstabelle steht.</td> </tr> <tr> <td>FloatValue</td> <td>Der neue Wert für die MODBUS Register formatiert als Fließkommazahl.</td> </tr> </table>	Index	Die Nummer der angeforderten Konfigurationszeile im Bereich von 1..n, wobei n für die letzte Zeile in der aktuellen Konfigurationstabelle steht.	FloatValue	Der neue Wert für die MODBUS Register formatiert als Fließkommazahl.
Index	Die Nummer der angeforderten Konfigurationszeile im Bereich von 1..n, wobei n für die letzte Zeile in der aktuellen Konfigurationstabelle steht.				
FloatValue	Der neue Wert für die MODBUS Register formatiert als Fließkommazahl.				

Richtung	ASCII Befehl
Host	#<BusAdr>,SMBADR:<MBUnit>CR #<BusAdr>,SETMODBUSADDRESS:<MBUnit>CR
Antwort	#<BusAdr>,OK CR
	Stellt die Unit Adresse des Moduls um. Diese Umstellung wird auf der MODBUS Seite sofort wirksam. Als Unit Adressen sind die Werte 0dec bis 255dec zulässig.
Host	#<BusAdr>,GMBADR CR #<BusAdr>,GETMODBUSADDRESS CR
Antwort	#<BusAdr>,GMBADR:<MBUnitDec>,<MBUnitHex> CR
	Zeigt die aktuelle MODBUS Unit Adresse des Moduls an. MBUnitDec MBUnitHex Die aktuell eingestellte MODBUS/RTU Unit Adresse für die Kommunikation.
Host	#<BusAdr>,SKNXADR:<KNXAddress>CR #<BusAdr>,SETKNXADDRESS:<KNXAddress>CR
Antwort	#<BusAdr>,OK CR
	Definiert die KNX Adresse für die KNX Kommunikation des Gateway neu. KNXAddress Die neue KNX Adresse für die Kommunikation am KNX Bus im Format <Hi>.<Med>.<Low> e.g. 15.15.255 Adressbereich von 0.0.0 bis 15.15.255
Host	#<BusAdr>,GKNXADR CR #<BusAdr>,GETKNXADDRESS CR
Antwort	#<BusAdr>,GKNXADR:<KNXAdrDec>,<KNXAdrHex>,<KNXAdrKNX> CR
	Retourniert die aktuelle KNX Adresse des Moduls. KNXAdrDec KNXAdrHex Die aktuell konfigurierte KNX Adresse für die Kommunikation am KNX Bus. KNXAdrKNX Die aktuelle KNX Adresse für die Kommunikation am KNX Bus im Format <Hi>.<Med>.<Low> e.g. 15.15.255 Adressbereich von 0.0.0 bis 15.15.255
Host	#<BusAdr>,RST CR #<BusAdr>,RESET CR
Antwort	Keine
	Führt einen Softwarereset (Neustart) des Moduls durch.

8.1.6 Die Konfigurationszeile

Die beiden ASCII Befehle READ CONFIG und GET CONFIG retournieren alle Daten einer Konfigurationszeile. Hier ist eine detaillierte Beschreibung eines jedes Feldes in dieser Antwortzeile.

Die allgemeine Syntax für diese zeile ist wie folgt:

#<BusAdr>,KNX:I:<Index>=<MBRegister>,<MBDataType>,<MBRegisterCount>,<Interval>,<KNXGroup>,<KNXDataType>,<KNXRichtung>,<Factor>

<BusAdr> steht für die aktuelle MODBUS Unit ID des Moduls als dezimale Nummer. z.B. 255

<Index> steht für den Index in die Konfigurationstabelle. Dieser Index startet mit 1 und endet mit n, abhängig von der tatsächlichen Anzahl von Einträgen in der Konfigurationstabelle. Der Index wird als Dezimalzahl übertragen.

<MBRegister> steht für den Startindex der Holding Register. Dieser Wert startet mit 1 für das erste Holding Register 4x00001 und endet mit 65535 für das letzte Holding Register 4x65535. Der Wert wird als Dezimalzahl dargestellt.

<MBDataType> definiert den Datentyp des/der MODBUS Registers. Das ist ein ASCII Text in Großbuchstaben. Folgende Texte sind möglich: UINT16, SINT16, UINT32, SINT32, UINT32R, SINT32R, FLOAT32, FLOAT32R, DOUBLE64, DOUBLE64R, GENERIC, ASCII oder ERR. Siehe die Erklärung der MODBUS Datentypen für mehr Details über diese Texte.

<MBRegisterCount> definiert die Anzahl der verwendeten MODBUS Holding Register als Dezimalzahl. Zum Beispiel ein UINT16 benötigt 1 Register, ein UINT32 oder ein FLOAT32 benötigt 2 Register.

<Interval> steht für ein Zeitintervall in Sekunden als Dezimalzahl. Es wird ein zyklisches Pollen von KNX Daten mit der konfigurierten KNX Gruppenadresse ausgelöst. Dies ist für zukünftige Erweiterungen und wird derzeit nicht benutzt!

<KNXGroup> definiert die KNX Gruppenadresse im Format <Hi>.<Medium>.<Low>. KNX Gruppenadressen sind hier im Bereich von 0.0.0 bis 15.7.255 zulässig.

<KNXDataType> ist ein ASCII Text, der den Datentyp des eingehenden oder ausgehenden KNX Telegramms festlegt. Es werden folgende ASCII Texte in Großbuchstaben verwendet: ERR, BIT, TWOBITS, FOURBITS, SIXBITS, CHARACTER, UINT8, SINT8, UNIT16, SINT16, FLOAT16, TIME, DATE, UINT32, SINT32, FLOAT32, STRING, GENERIC und DATETIME. Siehe die Erklärung der KNX Datentypen für mehr Details.

<KNXRichtung> ist ein ASCII Text, der die Kommunikationsrichtung am KNX Bus festlegt. Es werden folgende ASCII Texte in Großbuchstaben verwendet: ERR, R, W, RW. Siehe die Erklärung der KNX Richtungen für mehr Details.

<Factor> definiert eine Fließkommazahl, welche als Multiplikator der Daten aus ankommenden KNX Telegrammen und als Divisor für Daten der ausgehenden KNX Telegramme verwendet wird. Benutzen Sie das Fließkommaformat 1234.567. Benutzen Sie kein Kommazeichen als Dezimalpunkt!

Hier ist ein realer Auszug einer Konfiguration aus einem Terminalprogramm:

```

1   5   10  15  20  25  30  35  40  45  50  55  60  6.
#VERSION:1.0.1.
TYPE:RESI-KNX-MODBUS.
#KNX:I:1=1,SINT16,1,0,1.1.3,FLOAT16,R,10.000000.
#KNX:I:2=2,SINT16,1,0,1.1.4,FLOAT16,R,10.000000.
#KNX:I:3=3,SINT16,1,0,1.1.7,FLOAT16,R,10.000000.
#KNX:I:4=4,SINT16,1,0,1.1.8,FLOAT16,R,10.000000.
#KNX:I:5=5,SINT16,1,0,1.1.9,FLOAT16,R,10.000000.
#KNX:I:6=6,SINT16,1,0,1.2.1,FLOAT16,R,10.000000.
#KNX:I:7=7,SINT16,1,0,1.2.6,FLOAT16,R,10.000000.
#KNX:I:8=8,SINT16,1,0,1.2.11,FLOAT16,R,10.000000.
#KNX:I:9=9,SINT16,1,0,1.2.17,FLOAT16,R,10.000000.
#KNX:I:10=10,SINT16,1,0,1.2.22,FLOAT16,R,10.000000.
#KNX:I:11=11,SINT16,1,0,1.3.1,FLOAT16,R,10.000000.
#KNX:I:12=12,SINT16,1,0,1.3.2,FLOAT16,R,10.000000.
#KNX:I:13=13,SINT16,1,0,1.4.1,FLOAT16,R,10.000000.
#KNX:I:14=14,SINT16,1,0,1.4.2,FLOAT16,R,10.000000.
#KNX:I:15=15,SINT16,1,0,1.4.3,FLOAT16,R,10.000000.
#KNX:I:16=16,SINT16,1,0,1.4.4,FLOAT16,R,10.000000.
#KNX:I:17=17,UINT16,1,0,10.3.5,BIT,R,1.000000.
#KNX:I:18=18,UINT16,1,0,10.3.6,BIT,R,1.000000.
#KNX:I:19=19,SINT32,2,0,9.3.4,UINT32,R,0.001000.
#KNX:I:20=21,SINT32,2,0,9.3.6,UINT32,R,0.001000.
#KNX:I:21=23,SINT16,1,0,1.3.1,FLOAT16,R,10.000000.
#KNX:I:22=24,SINT16,1,0,1.3.2,FLOAT16,R,10.000000.
#KNX:I:23=25,SINT32,2,0,9.3.2,UINT32,R,0.001000.
#KNX:I:24=27,SINT32,2,0,9.3.5,UINT32,R,0.001000.
#KNX:I:25=29,SINT32,2,0,9.4.29,UINT32,R,0.001000.
#KNX:I:26=31,SINT32,2,0,9.4.30,UINT32,R,0.001000.
#KNX:I:27=33,SINT16,1,0,1.4.2,FLOAT16,R,10.000000.
#KNX:I:28=34,SINT16,1,0,1.4.3,FLOAT16,R,10.000000.
#KNX:I:29=35,SINT32,2,0,9.4.21,UINT32,R,0.001000.
#KNX:I:30=37,SINT32,2,0,9.4.31,UINT32,R,0.001000.
#KNX:I:31=39,SINT32,2,0,9.4.32,UINT32,R,0.001000.
#KNX:I:32=41,SINT32,2,0,9.4.34,UINT32,R,0.001000.
#KNX:I:33=43,SINT16,1,0,1.4.1,FLOAT16,R,10.000000.
#KNX:I:34=44,SINT16,1,0,1.4.4,FLOAT16,R,10.000000.
#KNX:I:35=45,SINT32,2,0,9.4.27,UINT32,R,0.001000.
#KNX:I:36=47,SINT32,2,0,9.4.33,UINT32,R,0.001000.
#KNX:I:37=49,SINT16,1,0,1.3.4,FLOAT16,R,10.000000.
#KNX:I:38=50,SINT16,1,0,1.3.5,FLOAT16,R,10.000000.
#KNX:I:39=51,SINT32,2,0,9.4.35,UINT32,R,0.001000.

```

Selection (-)

8.1.7 Die „AddConfigurationLine“

Der ASCII Befehl ADD CONFIG benutzt eine komplexe Konfigurationszeile, um einen neuen Eintrag zur aktuellen Tabelle hinzuzufügen. Hier ist eine detaillierte Beschreibung eines jeden Feldes dieser Konfigurationszeile.

The common syntax for the line is the following:

#<BusAdr>,ADD CONFIG:<MBRegister>,<MBDataType>,<Interval>,<KNXGroup>,<KNXDataType>,<KNXRichtung>,<Factor>

<BusAdr> steht für die aktuelle MDBUS Unit ID des Moduls als dezimale Nummer. z.B. 255

<Index> steht für den Index in die Konfigurationstabelle. Dieser Index startet mit 1 und endet mit n, abhängig von der tatsächlichen Anzahl von Einträgen in der Konfigurationstabelle. Der Index wird als Dezimalzahl übertragen.

<MBRegister> steht für den Startindex der Holding Register. Dieser Wert startet mit 1 für das erste Holding Register 4x00001 und endet mit 65535 für das letzte Holding Register 4x65535. Der Wert wird als Dezimalzahl dargestellt.

<MBDataType> definiert den Datentyp des/der MODBUS Registers. Das ist ein ASCII Text in Großbuchstaben. Folgende Texte sind möglich: UINT16, SINT16, UINT32, SINT32, UINT32R, SINT32R, FLOAT32, FLOAT32R, DOUBLE64, DOUBLE64R, GENERIC, ASCII oder ERR. Siehe die Erklärung der MODBUS Datentypen für mehr Details über diese Texte.

<Interval> steht für ein Zeitintervall in Sekunden als Dezimalzahl. Es wird ein zyklisches Pollen von KNX Daten mit der konfigurierten KNX Gruppenadresse ausgelöst. Dies ist für zukünftige Erweiterungen und wird derzeit nicht benutzt!

<KNXGroup> definiert die KNX Gruppenadresse im Format <Hi>.<Medium>.<Low>. KNX Gruppenadressen sind hier im Bereich von 0.0.0 bis 15.7.255 zulässig.

<KNXDataType> ist ein ASCII Text, der den Datentyp des eingehenden oder ausgehenden KNX Telegramms festlegt. Es werden folgende ASCII Texte in Großbuchstaben verwendet: ERR, BIT, TWOBITS, FOURBITS, SIXBITS, CHARACTER, UINT8, SINT8, UNIT16, SINT16, FLOAT16, TIME, DATE, UINT32, SINT32, FLOAT32, STRING, GENERIC und DATETIME. Siehe die Erklärung der KNX Datentypen für mehr Details.

<KNXRichtung> ist ein ASCII Text, der die Kommunikationsrichtung am KNX Bus festlegt. Es werden folgende ASCII Texte in Großbuchstaben verwendet: ERR, R, W, RW. Siehe die Erklärung der KNX Richtungen für mehr Details.

<Factor> definiert eine Fließkommazahl, welche als Multiplikator der Daten aus ankommenden KNX Telegrammen und als Divisor für Daten der ausgehenden KNX Telegramme verwendet wird. Benutzen Sie das Fließkommaformat 1234.567. Benutzen Sie kein Kommazeichen als Dezimalpunkt!

Ein einfaches Beispiel für ein gültiges ADD CONFIG Kommando:

```
#AC:1,UINT16,0,1.0.0,BIT,READ,1.0
#255,AC:0,UINT16,0,1.0.1,FLOAT16,RW,1.0
```


Das Ergebnis sieht dann wie folgt aus:

MODBUS Register	Speicherung eines UINT32R Datentyps
4x00010 I:9	Das untere Wort des 32 Bit Werts 0x12345678 wird im ersten 16 Bit breiten MODBUS Register abgelegt. Dies bedeutet, dass der Wert 0x5678 hier abgelegt wird.
4x00011 I:10	Das obere Wort des 32 Bit Werts 0x12345678 wird im zweiten 16 Bit breiten MODBUS Register abgelegt. Dies bedeutet, dass der Wert 0x1234 hier abgelegt wird.

Komplizierter ist das Ablegen eines FLOAT32 Wertes in zwei aufeinanderfolgende Holding Register. Wir benutzen eine Raumtemperatur z.B. 23,45 °C als einen Wert, den wir in zwei Registern ablegen wollen.

Zuerst müssen wir den Wert in eine gültige IEEE754 Fließkommazahl umwandeln. Dazu benutzen wir eine perfekte Internetseite (<http://www.h-schmidt.net/FloatConverter/IEEE754.html>):

The screenshot shows the IEEE754 Float Converter interface. The input value is 23.45. The output shows the binary representation (01000001101110111001100110011010), the hexadecimal representation (0x41bb999a), and the decimal representation (23.45). The 'Encoded as' field shows 0 and 131, and the 'Mantissa' field shows 1.4656250476837158.

Wir geben den Wert 23.45 ein und erhalten eine Repräsentation der Fließkommazahl als 32 Bit Hexadezimalzahl. Es ist die Nummer 0x41BB999A. Nun speichern wir diesen Wert in derselben Weise, wie wir den UINT32 Wert gespeichert haben:

MODBUS Register	Speicherung eines FLOAT32 Datentyps
4x00010 I:9	Das obere Wort des 32 Bit Werts 0x41BB999A wird im ersten 16 Bit breiten MODBUS Register abgelegt. Dies bedeutet, dass der Wert 0x41BB hier abgelegt wird.
4x00011 I:10	Das untere Wort des 32 Bit Werts 0x41BB999A wird im zweiten 16 Bit breiten MODBUS Register abgelegt. Dies bedeutet, dass der Wert 0x999A hier abgelegt wird.

But we can also use the reverse notation:

MODBUS Register	Speicherung eines FLOAT32R Datentyps
4x00010 I:9	Das untere Wort des 32 Bit Werts 0x41BB999A wird im ersten 16 Bit breiten MODBUS Register abgelegt. Dies bedeutet, dass der Wert 0x999A hier abgelegt wird.
4x00011 I:10	Das obere Wort des 32 Bit Werts 0x41BB999A wird im zweiten 16 Bit breiten MODBUS Register abgelegt. Dies bedeutet, dass der Wert 0x41BB hier abgelegt wird.

9 Spezifikationen

9.1 Abmessungen

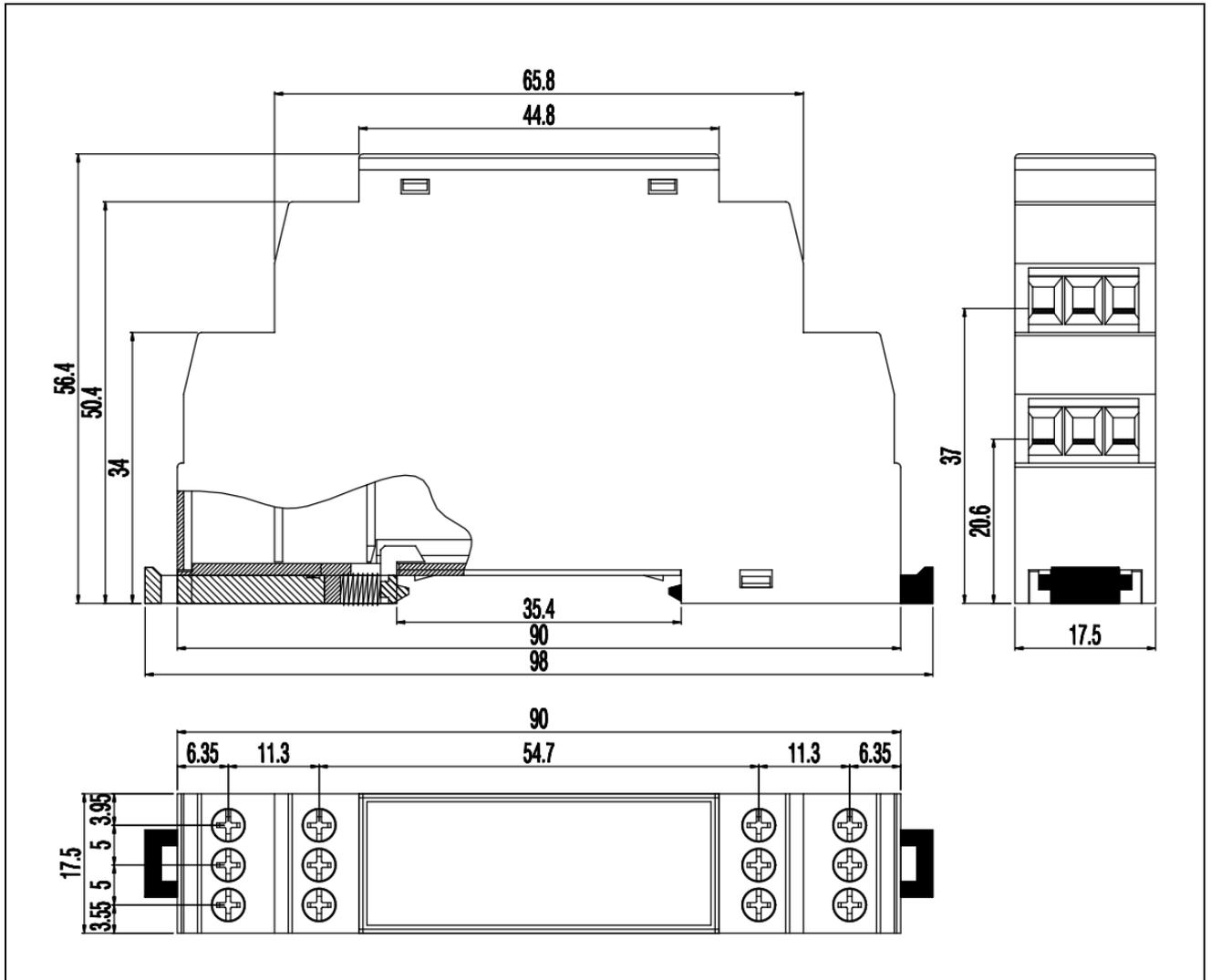


Abbildung: Abmessungen des Gehäuses in mm

Abmessungen	
Gehäuseabmessungen L x B x H (mm)	17,5 x 90 x 58
Gewicht	55 g
Farbe	Grau, RAL7035
Material	PA - UL 94 V0
Schutzklasse	IP20 basierend auf DIN 40050/EN 60529

Tabelle: Daten des Gehäuses

Proprietary data, company confidential. All rights reserved.
 Confidantia de companie, confidențialitate. Toți drepturile sunt rezervate.
 Comunicado como secreto empresarial. Reservados todos los derechos.
 Comunicado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verbreitung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich anders angegeben. Alle Rechte vorbehalten. Insbesondere für den Fall der Patenterteilung oder GM-Ertragung.

9.2 3D Zeichnung

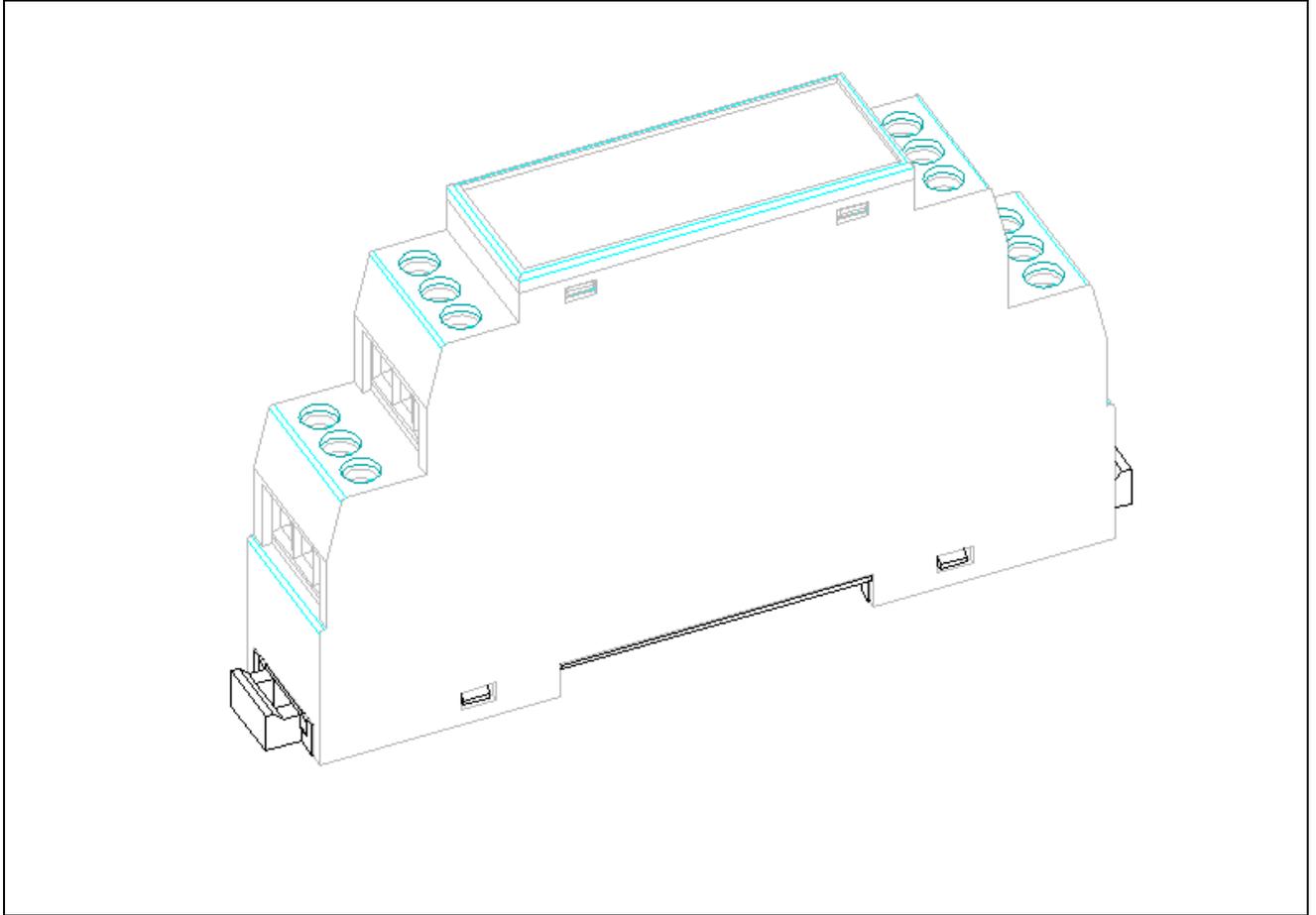


Abbildung: Gehäuseabbildung in 3D

Proprietary data, company confidential. All rights reserved.
Confé à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Comunicado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich anders bestimmt. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung.