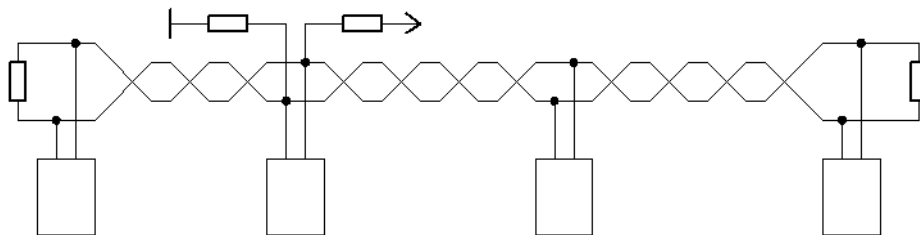


Technische Dokumentation



KSR MODBUS



Änderungsliste

Datum	Name	Revision	Kommentar
21.11.05	ATh	01	neues Dokument Funktionen Softwarestand (1.08 MB)
20.09.07	Le	02	Anpassung an neuen 3poligen Steckverbinder auf der Modbus-Platine
16.12.12	Le	03	Einfügen Adressfelder Datenspeicher
28.08.18	SMi	04	Inhaltliche Verbesserungen



Inhalt

1	ÜBERSICHT	4
2	MODBUS / RS485	5
2.1	RS485 (definiert in EIA485/ISO8482).....	5
2.1.1	Anschluss.....	6
2.1.2	Leitungsabschluss.....	7
2.1.3	Vorspannung	7
2.1.4	Kommunikationsanzeige	8
2.2	Das MODBUS Protokoll.....	8
2.2.1	MODBUS - Beschreibung.....	8
2.2.2	Serielle Datenformat und Datenrahmen	8
2.2.3	Serielle Übertragungsarten	9
2.2.4	Function Codes.....	10
2.2.5	Exception Codes	10
2.2.6	Master-Slave Protokoll	11
2.2.7	KSR - MODBUS Setup	11
2.2.8	Adressraum	11
2.2.9	Messwerte.....	13
2.2.10	Harmonische	14
2.2.11	Systemeinstellungen (Parameter).....	15
2.2.12	Relais Einstellungen.....	15
2.2.13	Alarm Einstellungen	16
2.2.14	Einstellungen speichern	20
2.2.15	Fehlerspeicher auslesen.....	20
3	PROBLEMBEHEBUNG.....	23

Wichtige Information!



Dieses Symbol steht neben dem Text an Stellen, denen besondere Aufmerksamkeit geschenkt werden sollte, da sie wichtige Informationen zum Gebrauch des Gerätes enthalten.

Dies können Sicherheitshinweise oder Vorschriften zum korrekten Gebrauch des Gerätes sein.

Werden Die Hinweise nicht befolgt, so können irreparable Schäden am Gerät entstehen.

Weitere Information zum MODBUS Protokoll können unter www.modbus.org gefunden werden. Dort sind auch die Bus-Spezifikationen erhältlich.

1 Übersicht

Die MODBUS - Erweiterung des KSR bietet die Möglichkeit, Messwerte über eine Datenverbindung von entfernten Rechnersystemen aus auszulesen und sie dort weiterzuverarbeiten.

Dieses Dokument beschreibt die Datenübertragung unter Verwendung des MODBUS Protokolls. Dieses Protokoll definiert Methoden des Datentransportes und der Adressierung, aber es legt sich nicht generell auf ein spezielles Übertragungsmedium (physikalische Übertragungsebene) fest.

Beim KSR wird hierzu RS485 benutzt, was durch die Konzipierung als Bus-System außerdem die Möglichkeit bietet mehrere Geräte am gleichen Kabelstrang zu betreiben.

Viele kommerzielle Geräte und SPS unterstützen das MODBUS Protokoll. Damit ist es oft mit geringem Aufwand möglich, eine Buslösung zu realisieren.

2 MODBUS / RS485

Der MODBUS besteht im Grunde aus zwei Teilen:

- RS485 wird als zugrundeliegendes Datenübertragungsmedium genutzt. Es regelt die physikalische Übertragung der Datenbytes zwischen den Busteilnehmern. Dieser „Übertragungsdienst“ wird von höherwertigen Schichten, dem MODBUS Applikationsprotokoll wahrgenommen.
- Das MODBUS Applikationsprotokoll benutzt das zugrundeliegende RS485-Protokoll zur Datenübertragung. Es definiert Kommandos (genannt „Function Codes“, FC), Adress- und Datenstrukturen.

2.1 RS485 (definiert in EIA485/ISO8482)

RS485 benutzt 2 Leitungen zum Datentransport D(+) und D(-). Die beiden Datenleitungen besitzen immer gegensätzliche Zustände: Eine Leitung 0V (=GND), die andere +5V. Damit ergeben sich insgesamt zwei Möglichkeiten, welche die beiden logischen Pegel „low“ und „high“ repräsentieren. Durch diese differentielle Übertragung besitzt RS485 eine große Immunität gegenüber Störungen durch magnetische oder elektrische Felder. Auf diese Weise werden Reichweiten von über 1000 Metern erreicht.

Die Übertragungsraten können beim KSR aus folgenden Werten gewählt werden: 1200, 2400, 4800, 19200 und 38400 Baud. Alle drei Paritäten (gerade, ungerade und keine Parität) werden unterstützt.

RS485 gibt es in zwei Varianten:

- 2-Draht RS485: Dieser Typ benutzt 2 Leitungen für die Kommunikation. Dies bedeutet, dass für beide Datenrichtungen das selbe Leitungspaar benutzt werden muss. Somit ist also ein Umschalten zwischen Senden und Empfangen bei jedem Gerät nötig (Halbduplex-Betrieb).
- 4-Draht RS485: Hier sind jeweils ein Paar Leitungen für die beiden Datenrichtungen vorhanden. Ein Umschalten ist dabei nur für die Sender der Slave-Geräte nötig. Aufgrund des MODBUS Protokolls kann aber auch hier nur der Halbduplex-Betrieb genutzt werden, somit ist keine Steigerung der Übertragungsleistung möglich.

Das KSR unterstützt daher nur den 2-Draht Betrieb!

Beide Typen benötigen jeweils eine weitere Leitung, welche zwischen allen Busgeräten verbunden sein muss: Die gemeinsame Signalmasse GND. Hierfür darf **nicht** die Abschirmung des Kabels verwendet werden. Diese sollte mit der Erdung verbunden werden, um Störungseinflüsse von außen weiter zu vermindern.

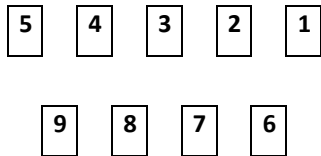
Der RS485-Bus unterstützt mehrere Geräte an einem Bus-Kabel (typisch bis zu 32). Hierzu müssen alle Signale der Geräte parallel miteinander verbunden werden. Typischerweise sind dies die Datenleitungen D(+) und D(-) sowie die Signalmasse (GND oder GROUND).

Ein Buskabel mit allen daran angeschlossenen Teilnehmern wird als Bus-Segment bezeichnet. Mit so genannten „Repeatern“ können die Daten zwischen mehreren Segmenten ausgetauscht werden.

2.1.1 Anschluss

Den MODBUS-Anschluss gibt es in zwei Varianten:

a) 9poliger D-SUB



PIN1 +5V zur Vorspannungserzeugung. **Diese Spannung niemals zur Versorgung externer Geräte benutzen!**

PIN2 GND zur Vorspannungserzeugung

PIN5 D (B) - Datensignal B

PIN9 D (A) - Datensignal A

b) 3poliger Steckanschluss

Zum Anschluss ist ein 3poliger Steckverbinder vorgesehen. Die Anschlussbelegung kann der Beschriftung auf dem Bild entnommen werden. Zur Inbetriebnahme sind die Datenleitungen + und -, sowie der Masseanschluss (mittlerer Anschluss) mit den entsprechenden Busleitungen zu verbinden.



2.1.2 Leitungsabschluss

Ein weiterer wichtiger Punkt ist der Abschluss der Busleitung. Dieser ist für ein korrekt funktionierendes Bussystem unbedingt nötig, um Störungen durch Rückwirkungen der Leitungsenden zu vermeiden. Um eine Busleitung abzuschließen, muss deren Ende mit einem Widerstand versehen werden. Der Wert des Widerstandes muss zur Kabelimpedanz passen. In den meisten Fällen ist ein 120 Ω Widerstand geeignet. Verbinden Sie den Abschlusswiderstand mit den beiden Datenleitungen an jedem Ende eines Bussegmentes.

Einige Geräte, speziell Bus-Konverter, haben integrierte Abschlusswiderstände. Kontrollieren Sie deshalb die Bedienungsanleitungen aller verwendeten Busgeräte. Wenn diese integrierten Widerstände nicht abgeschaltet werden können hat das zur Folge, dass diese Geräte am Ende des Busses platziert werden müssen! Da der Bus nur zwei Leitungsenden hat, können auch nur zwei Geräte mit festen Abschlusswiderständen verwendet werden!

2.1.3 Vorspannung

Ein anderer wichtiger Punkt ist die Vorspannung der Busleitungen. Wenn auf dem Bus keine Daten übertragen werden, hätten die Datenleitungen ohne Vorspannung keinen definierten Zustand. Wegen des Abschlusswiderstandes würden beide Datenleitungen annähernd die gleiche Spannung haben. Durch äußere Einflüsse kann es dann zu Störungen im Datensignal kommen. Aus diesem Grund ist eine Vorspannung der Busleitungen nötig, um für diesen Fall den Datenleitungen einen definierten Zustand zu geben.

Dazu müssen zwei Widerstände mit etwa 500-600 Ω verwendet werden. Einer von der Datenleitung D(+) nach +5V, der andere von D(-) nach GND. Diese Widerstände werden nur einmal pro Bus benötigt, die Position kann frei gewählt werden. Falls möglich wird eine Position in der Mitte empfohlen. Bitte überprüfen Sie anhand der Bedienungsanleitungen der verwendeten Busgeräte, ob bereits integrierte Vorspannungswiderstände vorhanden sind!

Bei der Anschlußvariante a) (9-pol D-SUB) sind die beiden Spannungen +5V und GND verfügbar, daher können diese beiden Vorspannungswiderstände auch im Gehäuse des Steckers untergebracht werden.

Bei der Anschlussvariante b) (3pol. Stecker) ist dies leider nicht möglich. **Achtung bei verschiedenen Herstellern stimmt die Anschlussbezeichnung A = + und B = - nicht überein. Dies muss von Fall zu Fall überprüft werden.**

2.1.4 Kommunikationsanzeige



Die gelbe Leuchtdiode auf der Rückseite des Gerätes zeigt die Aktivität der Datenübertragung an. Sie leuchtet nur, wenn das Gerät mit dem Busmaster kommuniziert.



Die Kommunikationsanzeige ist bei beiden Anschlussvarianten vorhanden.

2.2 Das MODBUS Protokoll

2.2.1 MODBUS - Beschreibung

Das MODBUS Protokoll benutzt die RS485 Schnittstelle als untergeordnete physikalische Ebene und implementiert den Kontrollmechanismus für die Datenübertragung. Es ist daher auf der Ebene 2 ("Link Layer") des OSI Schichtenmodells für den Datenaustausch einzuordnen.

2.2.2 Serielles Datenformat und Datenrahmen

Die Daten werden in einem festen Datenrahmen übertragen. Die einzelnen Datensätze werden voneinander separiert, indem der Bus für 3,5 Zeichen inaktiv bleibt. Die gesamten Daten sind in "Protocol Data Units" (PDUs) organisiert, welche von der untergeordneten physikalischen Datenebene über das serielle Bussystem übertragen werden.

Die PDU definiert eine einzelne Dateneinheit, die ein bestimmtes Busgerät erreichen soll um dort eine bestimmte Funktion auszuführen. Die Übertragung unterscheidet sich je nach der verwendeten physikalischen Ebene.

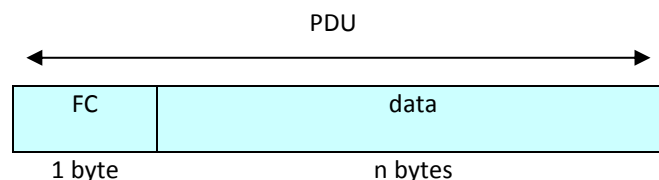


Abbildung 1 : "Protocol Data Unit" - PDU

Die PDU besteht aus zwei Teilen:

- Der "Function Code" (FC) gibt einen Befehl an. Dieser definiert, was der Slave zu tun hat.
- Der Datenblock besteht aus den entsprechenden Daten für einen FC. Die Verwendung hängt vom FC ab, er kann sowohl reine Daten aber auch Registeradressen für den Datenzugriff des Slave beinhalten.

Um die Übertragung kontrollieren zu können ist die PDU mit weiteren Datenblöcken versehen. Für RS485 ist diese Erweiterung die "Application Data Unit" (ADU).

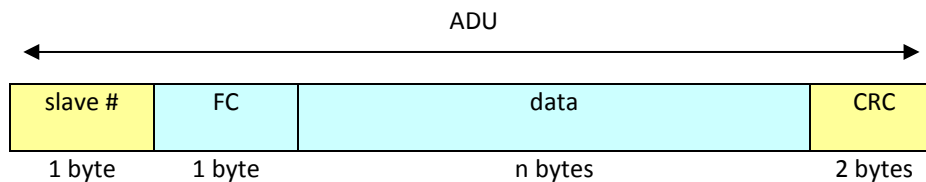


Abbildung 2 : "Application Data Unit" - ADU

Die ADU, so wie sie bei einer Übertragung für RS485 verwendet wird, beinhaltet zwei weitere Datenblöcke:

- Das erste Feld spezifiziert das Ziel für den Datensatz, die "Slave Number" ("Slave Address").
- Zusätzlich wird die Übertragung durch den CRC16 Fehlerkorrekturcode abgesichert.

2.2.3 Serielle Übertragungsarten

Das Protokoll definiert zwei verschiedene Kodierungen für den Inhalt des Datensatzes. **Das KSR benutzt immer den RTU-Modus! Der ASCII Modus ist nicht implementiert und wird hier nur zum Zwecke der Vollständigkeit genannt.**

"Remote Terminal Unit" (RTU)

Bei diesem Übertragungsmodus beinhaltet jedes 8-Bit Datenwort zwei 4-Bit Hexadezimalzahlen. Diese werden als ein komplettes Byte übertragen; dadurch wird eine maximale Übertragungsdichte erreicht.

Mit jedem Datenwort werden die folgenden Informationen übertragen:

- 1 Startbit
- 8 Datenbits, "Least Significant Bit" zuerst
- 1 Paritätsbit (falls gesetzt)
- 1 Stopbit für die Parität "even" oder "odd" / 2 Stopbits wenn die Parität "none" ist, um ein fehlendes Paritätsbit zu auszugleichen

"American Standard Code for Information Interchange" (ASCII)

Im ASCII-Modus werden die zwei 4-Byte Nibbles eines 8-Bit Datenwortes separat übertragen. Ein Datenbyte, welches den Inhalt 5B_{hex} hat, wird in zwei Teile aufgeteilt und dann einzeln übertragen. Das Ergebnis ist, dass zwei Datenbyte mit den Inhalten 35_{hex} (=ASCII-Code "5") und 42_{hex} (=ASCII-Code "B") übertragen werden. Dieser Modus wurde aus Kompatibilitätsgründen geschaffen und vereinfacht die Fehlersuche auf der Übertragungsleitung. Jedoch verlangsamt dieser Modus auch deutlich die Übertragungsgeschwindigkeit.

2.2.4 Function Codes

Wie bereits erwähnt beinhaltet das Datenpaket "Function Codes", die einen Befehl vom Bus-Master zum Bus-Slave spezifizieren. Der Slave führt den Befehl aus (falls möglich) und antwortet dann mit dem gleichen Function Code als Empfangsbestätigung. Der gültige Bereich für Function Codes ist zwischen 1 und 127 festgelegt, jedoch wird nur ein Teil davon verwendet. Für genauere Informationen wird auf die MODBUS Spezifikationen verwiesen. Wenn es für den Slave nicht möglich ist einen Befehl auszuführen, wird ein Fehlercode gesendet (Exception Code). Der Function Code eines Fehlercodes beinhaltet den Function Code des empfangenen Befehls, der den Fehler verursachte. Dieser wurde jedoch abgeändert: vom Slave wurde das MSB (=höchstwertiges Bit) gesetzt, um dem Master einen Fehler anzuzeigen. Der Inhalt des Datensatzes beschreibt den Fehler genauer.

Das KSR unterstützt die Function Codes 03_{hex} (Read Holding Register), 04_{hex} (Read Input Register) und 06_{hex} (Write Single Register).

2.2.5 Exception Codes

Wenn ein Slave einen Befehl des Masters nicht ausführen kann, so antwortet dieser mit einem "Exception Code". In der MODBUS Spezifikation kann die vollständige Liste eingesehen werden. Diese Liste ist hier nicht aufgeführt, da die Master-Software die meisten Ausnahmefehler automatisch behandelt. Wenn man den MODBUS Master Stack selbst programmieren muss, dann benötigt man die vollständigen Spezifikationen. Darin ist eine vollständige Liste an Fehlercodes enthalten.

2.2.6 Master-Slave Protokoll

Für die Kommunikation wird das Master-Slave Protokoll verwendet. Nur der Bus-Master kann einen Transfer initialisieren. Indem der Master einen Datensatz mit dem entsprechenden Function Code zum Slave sendet beginnt der Datenaustausch. Der Slave führt dann diesen Befehl aus.

- Im Modbus-System wird normalerweise der Unicast-Modus für die Kommunikation verwendet. Hierbei wird ein einzelner Slave im Datenpaket des Masters adressiert. Der gültige Adressbereich liegt zwischen 1 und 247. Der Slave führt dann den Befehl aus und antwortet indem er ein Datenpaket als Bestätigung zurück an den Master sendet.
- Nicht in jedem Fall erhält der Master eine Antwort auf seine Anfrage: im Multicast-Modus werden alle Slaves im Bus-System parallel adressiert. Alle Slaves führen den gleichen Befehl aus, aber keiner antwortet. Der Master initialisiert diesen Multicast-Transfer indem eine "0" als Slave-Adresse verwendet wird.

2.2.7 KSR - MODBUS Setup

Wenn das verwendete Gerät Modbus unterstützt ist ein zusätzlicher Eintrag im "Setup"-Menü verfügbar. Nachdem dieser Menüpunkt angewählt wurde sind folgende Einstellungen möglich:

- ADDRESS: Auswahl der Slave-Adresse (Slave ID). Der gültige Bereich liegt zwischen 1 und 247.
- BAUD RATE: Auswahl der Baudrate. Der gültige Bereich liegt zwischen 1200 und 38400.
- PARITY: Auswahl der Parität zwischen EVEN, ODD oder NONE.

Die Einstellungen für Baudrate und Parität müssen für alle Bus-Geräte identisch sein. Die Slave-Adresse eines Gerätes hingegen darf im Bus-System jeweils nur einmal vorhanden sein.

2.2.8 Adressraum

Die Daten im KSR werden mit Hilfe von Adressen organisiert und zugänglich gemacht. Jede Adresse bietet Zugriff auf ein Datenwort. Die Länge eines Datenwortes beträgt immer 16 Bit.

Das KSR unterscheidet nicht die Adressen zwischen den Function Codes. Es ist ein einziger großer Adressraum verfügbar und um an die Daten einer bestimmten Adresse zu gelangen kann jeder gültige Function Code verwendet werden. Dennoch machen die Daten nur Sinn, wenn sie auf die richtige Art und Weise interpretiert werden!

Die Daten können von den folgenden Typen sein:

- FLOAT: 32 Bit Floating-Point Wert, wie im IEEE Standard 754 definiert.
- UINT16: 16 Bit Integer Wert ohne Vorzeichen.
- SINT16: 16 Bit Integer Wert mit Vorzeichen.
- LONG: 32 Bit Integer Wert.
- Struct: Vordefinierte Daten-Feld

Da die Daten in 16 Bit breiten Worten organisiert sind, müssen für längere Datenfelder mehrere aufeinanderfolgende Adressen gelesen werden. Für diese Fälle ist in den Tabellen die Basisadresse angegeben. Um beispielsweise Daten im Format FLOAT mit der Basisadresse 12 zu lesen, müssen zwei 16 Bit Worte von den Adressen 12 und 13 gelesen werden. Diese beiden Werte müssen geeignet miteinander verknüpft werden, um das gewünschte Ergebnis mit 32 Bits zu erhalten. Die meisten SCADA- und SPS- Software-Pakete erledigen diesen Vorgang selbstständig.



Es gibt verschieden Adresstypen:

- Die MODBUS Adresse beginnt immer mit 0 und kann bis zu 65535 gehen. Diese Variante kann mit jedem Function Code verwendet werden.
- Manche SPS-Steuerungen können mit der 0 nicht arbeiten und addieren daher zur Adresse eine 1. Deshalb beginnen die Adressen (MODBUS Adresse +1) immer mit 1.
- Einige SCADA Systeme addieren zum Festlegen des Function Codes einen Offset, der dazu verwendet werden soll, das Gerät unter der angegebenen Adresse zu erreichen. Hierbei wird oftmals auch eine 1 zur Modbus Adresse addiert. Beispielsweise würde Adresse 40001 bedeuten "lese MODBUS Adresse 0 mit Function Code 03_{hex}", 30012 würde bedeuten "lese MODBUS Adresse 11 mit Function Code 04_{hex}". Um die korrekte Adresse herauszufinden verwenden Sie bitte die Bedienungsanleitung der verwendeten Software.

Die folgenden Tabellen geben die MODBUS Adressen immer auf die erste oben genannte Art an.

2.2.9 Messwerte

Die zur Verfügung stehenden Messwerte beginnen bei der Adresse 0 mit Intervallen von 6 Datenworten. Für jeden dieser Messwerte sind auch Minimum- und Maximumwerte verfügbar. Um das Maximum auszulesen, muss man 2 zur Adresse des jeweiligen Wertes addieren, für das Minimum muss man 4 addieren. (Beispiel: um das Minimum der Spannung L1-N zu erhalten, muss man den Inhalt der Adresse 00034 = 00030+4 auslesen). All diese Werte können mit den Function Codes 03_{hex} und 04_{hex} angesprochen werden.

Adresse	Wert	AnzahlW orte	Daten- typ	Einheit
00000	Strom I-1	2	FLOAT	A
00006	Strom I-2	2	FLOAT	A
00012	Strom I-3	2	FLOAT	A
00018	Strom I-4	2	FLOAT	A
00024	Spannung L1-N	2	FLOAT	V
00030	Spannung L2-N	2	FLOAT	V
00036	Spannung L3-N	2	FLOAT	V
00042	Spannung L1-L2	2	FLOAT	V
00048	Spannung L2-L3	2	FLOAT	V
00054	Spannung L3-L1	2	FLOAT	V
00060	Strom Grundwelle If-1	2	FLOAT	A
00066	Strom Grundwelle If-2	2	FLOAT	A
00072	Strom Grundwelle If-3	2	FLOAT	A
00078	Strom Grundwelle If-4	2	FLOAT	A
00084	Total harmonic distortion THD-I1	2	FLOAT	%
00090	Total harmonic distortion THD-I2	2	FLOAT	%
00096	Total harmonic distortion THD-I3	2	FLOAT	%
00102	Total harmonic distortion THD-I4	2	FLOAT	%
00108	Total harmonic distortion THD-U1	2	FLOAT	%
00114	Total harmonic distortion THD-U2	2	FLOAT	%
00120	Total harmonic distortion THD-U3	2	FLOAT	%
00126	Umgebungstemperatur TEMP	2	FLOAT	°C
00132	Gedämpfter Strom Ith-1	2	FLOAT	A
00138	Gedämpfter Strom Ith-2	2	FLOAT	A
00144	Gedämpfter Strom Ith-3	2	FLOAT	A
00150	Gedämpfter Strom Ith-4	2	FLOAT	A

Sind nur die aktuellen Messwerte von Interesse, können diese auch ohne den dazugehörigen min. und max. Werten ausgelesen werden. Der entsprechende Adressbereich startet bei Adresse 2816 und ist von der Reihenfolge wie zuvor aufgebaut.

Adresse	Wert	AnzahlW orte	Daten- typ	Einheit
02816	Strom I-L1	2	FLOAT	A
02818	Strom I-L2	2	FLOAT	A
02820	Strom I-L3	2	FLOAT	A
...
...

2.2.10 Harmonische

Harmonische werden in separaten Feldern für jeden Strom / Spannung vom Typ FLOAT gespeichert. Die untenstehende Tabelle zeigt die jeweilige Basisadresse. Jedes Datenfeld beinhaltet 64 Werte, in dem jeder eine Länge von 2 Worten hat. Der erste Tabelleneintrag zeigt den Wert der Grundwelle. Da alle Werte auf 100% = Grundwelle normiert werden, ist dieser erste Wert immer "100.0". Nach der Grundwelle (=Harmonische der Ordnung 1) folgen die anderen Harmonischen in aufsteigender Reihenfolge bis zur 63. Oberwelle. Es ist zu beachten, dass jeder Wert vom Format FLOAT 2 Worte belegt. Daher muss die Adresse für den nächsten Wert jeweils um 2 erhöht werden.

Wenn der Strom oder die Spannung zur Berechnung der Harmonischen zu gering ist, erscheint an der Basisadresse (=Grundwelle) 0.0% anstelle von 100.0%. Dies weist ebenfalls darauf hin, dass die Harmonischen höherer Ordnung nicht zur Verfügung stehen!

All diese Werte können mit den Function Codes 03_{hex} und 04_{hex} angesprochen werden.

Adresse	Wert	AnzahlW orte	Daten- typ	Einheit
1538	Basisadresse für Harmonische I - 1	64*2	FLOAT	%
1668	Basisadresse für Harmonische I - 2	64*2	FLOAT	%
1798	Basisadresse für Harmonische I - 3	64*2	FLOAT	%
1928	Basisadresse für Harmonische I - 4	64*2	FLOAT	%
2058	Basisadresse für Harmonische U - L1-N	64*2	FLOAT	%
2188	Basisadresse für Harmonische U - L2-N	64*2	FLOAT	%
2318	Basisadresse für Harmonische U - L3-N	64*2	FLOAT	%

2.2.11 Systemeinstellungen (Parameter)

Die Parameter des KSR können ebenfalls über MODBUS eingestellt werden. Diese werden ab der Adresse 512 im UINT16 Format gespeichert. Die Verfügbaren Adressen können der unteren Tabelle entnommen werden.

Alle Adressen können mit den Function Codes 03_{hex}, 04_{hex} und 06_{hex} angesprochen werden.

Adresse	Wert	AnzahlW orte	Daten- typ	Einheit
512	Stromwandlerfaktor ct-I123	1	UINT16	-
513	Spannungswandlerfaktor vt	1	UINT16	-
514	Stromwandlerfaktor ct-IUB	1	UINT16	-
515	Thermische Zeitkonstante TAU	1	UINT16	-

2.2.12 Relais Einstellungen

Das Verhalten der Relaisausgänge kann ebenfalls über MODBUS beeinflusst werden. Deren Funktionsweise wird über die Bitmaske in der Adresse 769 bestimmt (UINT16). Hierbei bezeichnet das Least Significant Bit das Relais Nummer 1. Die höherwertigen Bits bezeichnen der Reihe nach die einzelnen Relais.

Eine „0“ in der Bitmaske bedeutet, dass das jeweilige Relais die Funktion eines Schließers besitzt. Im Umkehrschluss haben Relais mit der Maskierung „1“ die Funktion eines Öffners.

Innerhalb der Reset-Bitmaske bezeichnet eine „0“ automatischer Reset und „1“ verlangt ein manuelles Zurücksetzen.

Alle Adressen können mit den Function Codes 03_{hex}, 04_{hex} und 06_{hex} angesprochen werden.

Adresse	Wert	AnzahlW orte	Daten- typ	Einheit
769	Relaislogik: Schließer/Öffner	1	UINT16	-
770	Relais Reset: automatisch/manuell	1	UINT16	-

2.2.13 Alarm Einstellungen

Alarmer werden in einzelnen Daten-Feldern konfiguriert. Die folgende Tabelle zeigt die Basisadressen für alle konfigurierbaren Alarmer.

Die folgenden Adressen können mit den Function Codes 03_{hex}, 04_{hex} und 06_{hex} angesprochen werden.

Adresse	Wert	AnzahlW orte	Daten- typ	Einheit
1024	Basisadresse für Alarm 1	6	Struct	-
1030	Basisadresse für Alarm 2	6	Struct	-
1036	Basisadresse für Alarm 3	6	Struct	-
1042	Basisadresse für Alarm 4	6	Struct	-
1048	Basisadresse für Alarm 5	6	Struct	-
1054	Basisadresse für Alarm 6	6	Struct	-
1060	Basisadresse für Alarm 7	6	Struct	-
1066	Basisadresse für Alarm 8	6	Struct	-
1072	Basisadresse für Alarm 9	6	Struct	-
1078	Basisadresse für Alarm 10	6	Struct	-
1084	Basisadresse für Alarm 11	6	Struct	-
1090	Basisadresse für Alarm 12	6	Struct	-
1096	Basisadresse für Alarm 13	6	Struct	-
1102	Basisadresse für Alarm 14	6	Struct	-
1108	Basisadresse für Alarm 15	6	Struct	-
1114	Basisadresse für Alarm 16	6	Struct	-
1120	Basisadresse für Alarm 17	6	Struct	-
1126	Basisadresse für Alarm 18	6	Struct	-
1132	Basisadresse für Alarm 19	6	Struct	-
1138	Basisadresse für Alarm 20	6	Struct	-
1144	Basisadresse für Alarm 21	6	Struct	-
1150	Basisadresse für Alarm 22	6	Struct	-
1156	Basisadresse für Alarm 23	6	Struct	-
1162	Basisadresse für Alarm 24	6	Struct	-
1168	Basisadresse für Alarm 25	6	Struct	-

1174	Basisadresse für Alarm 26	6	Struct	-
1180	Basisadresse für Alarm 27	6	Struct	-
1186	Basisadresse für Alarm 28	6	Struct	-
1192	Basisadresse für Alarm 29	6	Struct	-
1198	Basisadresse für Alarm 30	6	Struct	-
1204	Basisadresse für Alarm 31	6	Struct	-
1210	Basisadresse für Alarm 32	6	Struct	-

Ein Alarm-Datenfeld besteht aus einer Struct die 5 Wert beinhaltet. Hierbei ist für jeden Alarm das Datenfeld gleich aufgebaut. Deren Zusammensetzung ist in der unteren Tabelle dargestellt.

Adresse	Wert	AnzahlW orte	Daten- typ	Einheit
Base address + 0	Quelle (siehe unten)	1	SINT16	-
Base address + 1	Limit	2	SINT32	-
Base address + 3	T-On x 100	1	UINT16	s
Base address + 4	T-Off x 100	1	UINT16	s
Base address + 5	Ausgabe (siehe unten)	1	UINT16	-

Ein Limit kann über einen Wert im SINT32 Format eingegeben werden. Die Werte für T-On und T-Off können ebenfalls direkt gesetzt werden (UINT16), jedoch muss hier der Faktor 100 berücksichtigt werden. Die folgende Tabelle zeigt mögliche Alarmquellen.

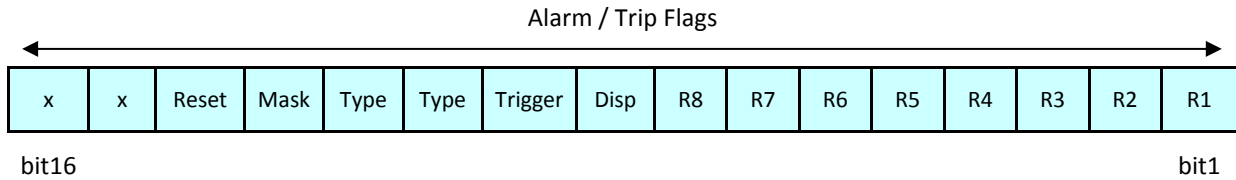
Adresse	Quelle	AnzahlW orte	Faktor
-1	Inaktiv	SINT16	-
0	Strom I-L1	SINT16	100
1	Strom I-L2	SINT16	100
2	Strom I-L3	SINT16	100



3	Strom I-UB	SINT16	100
4	Spannung U-L1N	SINT16	1
5	Spannung U-L2N	SINT16	1
6	Spannung U-L3N	SINT16	1
7	Spannung U-L12	SINT16	1
8	Spannung U-L23	SINT16	1
9	Spannung U-L31	SINT16	1
10	Strom Grundwelle If-L1	SINT16	100
11	Strom Grundwelle If-L2	SINT16	100
12	Strom Grundwelle If-L3	SINT16	100
13	Strom Grundwelle If-UB	SINT16	100
14	Total harmonic distortion THD-I1	SINT16	100
15	Total harmonic distortion THD-I2	SINT16	100
16	Total harmonic distortion THD-I3	SINT16	100
17	Total harmonic distortion THD-IUB	SINT16	100
18	Total harmonic distortion THD-U1N	SINT16	100
19	Total harmonic distortion THD-U2N	SINT16	100
20	Total harmonic distortion THD-U3N	SINT16	100
21	Umgebungstemperatur T	SINT16	1
22	Gedämpfter Strom Ith-L1	SINT16	100
23	Gedämpfter Strom Ith-L2	SINT16	100
24	Gedämpfter Strom Ith-L3	SINT16	100
25	Gedämpfter Strom Ith-UB	SINT16	100



Die Bitmaske der „Ausgabe“ – Adresse enthält die folgenden Informationen.



- R1 – R8: Ausgabe Relais 1 – 8
 “0” Relais wird nicht betätigt, wenn ein Trip oder Alarm auftritt
 “1” bedeutet Relais wird betätigt.
- Disp: Ausgabe Display
 “0” Display Meldung ist deaktiviert, wenn ein Trip oder Alarm auftritt
 “1” Display Meldung ist aktiviert
- Trigger: “0” Wert > Limit
 “1” Wert < Limit
- Type: Code 00: Überlast
 Code 01: Unsymmetrie
 Code 10: Überspannung
 Code 11: Unterspannung
- Mask: “0” Trip
 “1” Alarm
- Reset: “0” Automatischer Reset der Meldung
 “1” Manueller Reset der Meldung
- X Reserviert



Alle Einstellungen über MODBUS werden sofort übernommen, jedoch nur im RAM gespeichert. Dies bedeutet, dass die vorgenommenen Änderungen bei einem Spannungsverlust verlorengehen. Um Änderungen dauerhaft zu speichern, müssen diese im Flash gespeichert werden.

2.2.14 Einstellungen speichern

Um Einstellungen in den Flash zu speichern (EPROM), müssen die folgenden Register beschrieben werden.

Beide Adressen können mit den Function Codes 03_{hex}, 04_{hex} und 06_{hex} angesprochen werden.

Adresse	Wert	AnzahlW orte	Daten- typ	Einheit
2560	Systemeinstellungen im EPROM speichern	1	UINT16	-
2561	Alarm und Relais Daten im EPROM speichern	1	UINT16	-

Für den Schreibprozess, muss in das jeweilige Register der Schlüssel "24276" geschrieben werden. Ein erfolgreicher Speichern wird anschließend durch eine "1" in der beschriebenen Adresse signalisiert.

2.2.15 Fehlerspeicher auslesen

Die folgende Tabelle stellt die Basisadressen für alle gespeicherten Werte dar. Das X repräsentiert die Speichernummer.

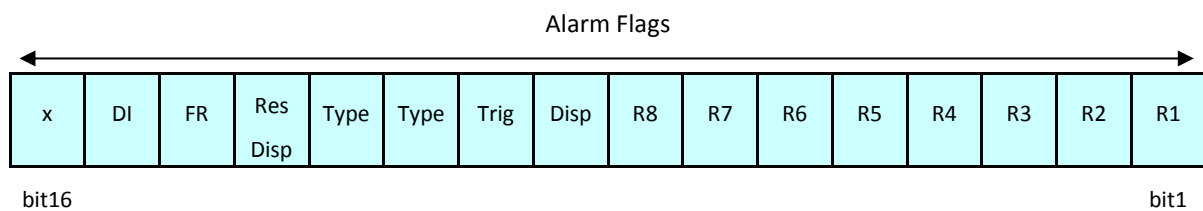
Adresse	Wert	AnzahlW orte	Daten- typ	Einheit
10000+28*X	Alarm-Nummer	1	UINT16	-
10001+28*X	Ereignis anstehend / vorüber	1	UINT16	-
10002+28*X	Monat_Jahr des Ereignisses	1	HEX	-
10003+28*X	Tag_Stunde des Ereignisses	1	HEX	-
10004+28*X	Minute_Sekunde des Ereignisses	1	HEX	-
10005+28*X	Alarm Quelle (siehe unten)	1	SINT16	-
10006+28*X	Limit	2	FLOAT	
10008+28*X	max. Wert	2	FLOAT	
10010+28*X	Auslösezeit	2	FLOAT	s
10012+28*X	Alarm Flags	1	SINT16	-
10013+28*X	Spannung L12	2	FLOAT	V
10015+28*X	Spannung L23	2	FLOAT	V
10017+28*X	Spannung L31	2	FLOAT	V
10019+28*X	Strom I1	2	FLOAT	A



10021+28*X	Strom I2	2	FLOAT	A
10023+28*X	Strom I3	2	FLOAT	A
10025+28*X	Strom IN	2	FLOAT	A

Mögliche Alarm Quellen:

Quelle	Bedeutung	Quelle	Bedeutung
0	I-L1	14	THD1
1	I-L2	15	THD2
2	I-L3	16	THD3
3	I-UB	17	THD4
4	U-L1N	18	THDU1N
5	U-L2N	19	THDU2N
6	U-L3N	20	THDU3N
7	U-L12	21	TEMP
8	U-L23	22	Ith-1
9	U-L31	23	Ith-2
10	If-1	24	Ith-3
11	If-2	25	Ith-UB
12	If-3	26	Iuc-4
13	If-Ub		



R1 – R8: Ausgabe Relais 1 – 8

“0” Relais wird nicht betätigt, wenn ein Trip oder Alarm auftritt

“1” Relais wird betätigt

Disp: Ausgabe Display:

“0” Display Meldung ist deaktiviert, wenn ein Trip oder Alarm auftritt

“1” Display Meldung ist aktiviert

Trig: “0” Wert > limit

“1” Wert < limit

Type: Code 00: Überlast

Code 01: Unsymmetrie

Code 10: Überspannung

Code 11: Unterspannung

Res Disp: “0” Automatischer Reset der Display Meldung

“1” Manueller Reset der Display Meldung

FR: “0” Alarm / Trip Ereignis wird im Fehlerspeicher nicht gespeichert

“1” Alarm / Trip Ereignis wird gespeichert

DI: “0” Alarm / Trip kann durch den Digitaleingang nicht blockiert werden

“1” Alarm / Trip kann durch den Digitaleingang blockiert werden

X Reserviert



Im Gerätespeicher sind noch weitere Werte verfügbar, die oben nicht aufgelistet wurden. Diese können jedoch wichtige Einstellungsdaten für das Gerät beinhalten. Schreiben Sie daher niemals auf eine beliebige Adresse, deren Inhalt nicht bekannt ist!



3 Problembhebung

Sollte die Busanbindung nicht funktionieren, so überprüfen Sie bitte die folgenden Punkte:

1. Kommt überhaupt keine Kommunikation zustande, dann muss der Fehler zwischen KSR und PC gesucht werden!

Mögliche Ursachen können sein:

- Einstellungen Baudrate, Parität und Adresse am KSR überprüfen, eventuell Konfigurationsänderungen vornehmen
 - Eventuell ist bei der Busleitung A und B vertauscht, ggf. richtig stellen
 - Einstellungen des Konverters RS485/RS232 überprüfen, eventuell Datenblatt des Konverters verwenden
 - Mehrfachbelegung der Schnittstelle am PC vorhanden, ggf. diese Mehrfachbelegung abstellen
 - Abschlusswiderstände bzw. Vorspannungswiderstände überprüfen, ggf. richtig stellen
2. Weist das Kabel der Busverbindung eine Beschädigung auf? Sind alle Steckverbindungen in Ordnung? Gegebenenfalls ersetzen.
 3. Ist die Pinbelegung des RS485-Anschlusses richtig? Gegebenenfalls ändern.
 4. Die Abschirmung der Busleitung darf nicht geerdet sein. Falls dies der Fall ist, Abschirmung von der Erdung trennen.
 5. Funktioniert zwar die Kommunikation, aber macht die Kundensoftware Probleme, dann beachten Sie bitte die folgenden Punkte:
 - Einstellungen Busadresse, Parität und Baudrate in der Software überprüfen
 - Datenformat überprüfen