

# Protokollspezifikation SI Modbus

## SolarInvert Wechselrichter



## Vorwort

Sehr geehrter Kunde,

vielen Dank für Ihr Vertrauen in unsere Produkte. Die SolarInvert GmbH hat über 10 Jahre Erfahrung in der Entwicklung und Produktion von leistungsstarken elektronischen Bauteilen und Systemen für die Photovoltaik. Wir haben uns zum Leitbild gemacht, unsere Kunden immer nach höchsten Anforderungen zu bedienen.

Technik und Ausstattung unserer Produkte entsprechen funktionell und sicherheitstechnisch dem neuesten Stand nationaler und internationaler Anforderungen. Weiterentwicklungen und Verbesserungen werden laufend berücksichtigt, daher können sich Abbildungen, Maße und technische Daten sowie allgemeine Inhalte, die in diesem Handbuch aufgeführt sind, durch Anpassung an neue Erkenntnisse verändern. Dabei haben wir uns entschieden, nicht jede Neuentwicklung oder Tendenz des Marktes auf Kosten unserer Kunden zu testen. So setzen wir neue Techniken und Konzepte erst nach eingehenden eigenen Tests und Studien ein.

Copyright© 2017 Solarinvert GmbH.

Dieses Handbuch sowie die beschriebenen Geräte sind urheberrechtlich geschützt. Jede Vervielfältigung dieses Dokuments ist nicht erlaubt, sofern nicht ausdrücklich genehmigt. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten.

Die Nennung von Produkten anderer Hersteller in diesem Handbuch dient ausschließlich Informationszwecken, und stellt keinen Warenzeichenmissbrauch dar.

### **Haftungsausschluss**

Bei der Zusammenstellung der Texte und Darstellungen wurde mit größter Sorgfalt vorgegangen. Trotz aller Bemühungen können Fehler nicht vollständig vermieden werden. Für die Richtigkeit des Inhalts kann daher keine Garantie übernommen werden. Für fehlerhafte Angaben und deren Folgen können wir weder eine juristische Verantwortung und keinerlei Haftung übernehmen.

Für Verbesserungsvorschläge und Hinweise auf Fehler sind wir jederzeit dankbar. Wir sind bemüht, diese Anleitung immer der aktuellsten Geräteversion anzupassen.

Es ist jedoch möglich, dass Unterschiede zwischen Geräten und Anleitung vorhanden sein können. Technische Änderungen behalten wir uns vor.

# Inhaltsverzeichnis

Vorwort .....	2
Inhaltsverzeichnis .....	3
Revision History .....	4
<b>1 Kommunikation über SI-Modbus .....</b>	<b>5</b>
1.1 Pin-Belegung für RJ45 Stecker .....	5
1.2 Bus Layout.....	5
1.3 Terminierung .....	5
1.4 Galvanische Trennung.....	6
1.5 Netzwerkmanagement und Adressvergabe.....	6
<b>2 Datenübermittlung.....</b>	<b>7</b>
2.1 Codebeispiele für die Checksummenberechnung .....	7
2.1.1 Musterimplementierung zur Berechnung der Prüfsumme: .....	7
2.1.2 Musterimplementierung zur Prüfung der Prüfsumme:.....	8
2.1 Anfragetelegramme.....	8
2.2 Antworttelegramme .....	9
<b>3 Standard Funktionscodes .....</b>	<b>10</b>
<b>4 Proprietäre Funktionscodes.....</b>	<b>11</b>
4.1 Fabrikationsdaten auslesen (cmd 31h).....	11
4.2 Betriebsdaten auslesen (cmd 33h/3Eh).....	12
4.3 Wechselrichter ausschalten (cmd 34h/35h).....	15
4.4 Leerlauf erzwingen (cmd 36h/37h) .....	16
4.5 In den Konstantspannungsmodus wechseln bzw. Spannung steuern (cmd 38h/39h) .....	17
4.6 Leistungsfaktor $\cos \varphi$ festlegen (cmd 3Bh).....	18
4.7 Wechselrichterleistung festlegen (cmd 3Fh/44h) .....	18
4.7.1 Steuerung/Begrenzung der Leistung beim BatteryInvert in Modus 0 und 3 .....	18
4.7.2 Leistungsbegrenzung Solarwechselrichter .....	19
4.8 Fehlerlogbuch auslesen (cmd 40h).....	20
<b>5 Kundendienst und Garantie.....</b>	<b>22</b>

## Revision History

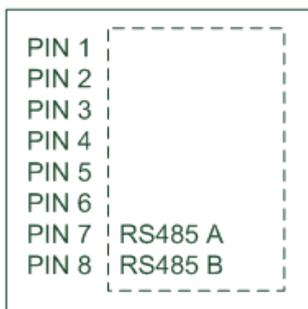
index	date	modification
005	01.06.2014	Überführung in neue Formatvorlage
009	30.09.2014	Funktionscode 3Bh überarbeitet
010	08.10.2014	Wertebereiche und Auflösung für alle FCs überarbeitet
011	02.01.2015	Funktionscode 3Bh überarbeitet
012	10.01.2015	Einführung weiterer Funktionscodes
013	21.2.2015	<ul style="list-style-type: none"> <li>- Entfernung des Offsets für Antwortadresse</li> <li>- Einführung von CRC16</li> <li>- Entfernung der automatischen Adresszuweisung durch den Master</li> <li>- Udc-Schrittweite auf 0,1V</li> <li>- Funktionscode 44h hinzugefügt</li> <li>- Funktionscode 33h überarbeitet</li> </ul>
014	25.2.2015	<ul style="list-style-type: none"> <li>- negative Antwort überarbeitet</li> <li>- Änderung auf No-Parity</li> <li>- Funktionscode 38h überarbeitet</li> <li>- Beispiel für Antwort 33h überarbeitet</li> </ul>
015	20.3.2015	Antwort 3Eh überarbeitet
016	26.03.2015	Antwort 33h überarbeitet
017	21.04.2015	Ergänzung von 1.4 um Hinweis auf Sendepause beim Bootvorgang Korrektur in der Bezeichnerliste von Antwort 31h Korrektur in der Bezeichnerliste von Antwort 33h Korrektur in der Bezeichnerliste von Antwort 3Eh Korrektur in der Bezeichnerliste von Antwort 3Bh Korrektur in der Bezeichnerliste von Antwort 3Fh Korrektur in der Bezeichnerliste von Antwort 44h
018	05.05.2015	Pin-Belegung der RJ45 Buchsen hinzugefügt Antwort 3Fh überarbeitet Antwort 38h überarbeitet
019	12.05.2015	Korrektur: Keine Terminierung bei Windinvertern Jumperbelegung für Netzwerkadresse hinzugefügt
020	26.05.2015	<ul style="list-style-type: none"> <li>- Antwort 3Eh überarbeitet</li> <li>- Zuordnungsmatrix für Spannungstyp und Wechselrichtermodell hinzugefügt</li> </ul>
021	04.12.2015	Korrektur des Skalenfaktors für Udc (cmd 33h) betroffen sind ausschließlich WR mit Spannungstyp 4 (SOL-50, WIN-36, BAT-30)
022	23.05.2016	Korrektur in Antwort 31h
030	5.7.2016	Kap. 4.7 und Korrektur aller Checksummen
031	03.05.2017	Grundlegende Überarbeitung mit umfassenden Fehlerkorrekturen und Angleichung an Geräte-Software

## 1 Kommunikation über SI-Modbus

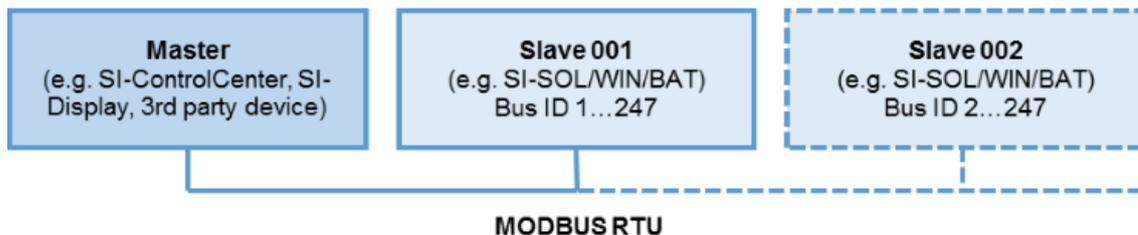
SolarInvert Wechselrichter bieten eine serielle Schnittstelle (RS485) und nutzen proprietäre Funktionscodes des Modbus RTU Kommunikationsprotokolls. Detailliertere Informationen zu Modbus RTU über diese Anleitung hinaus sind unter [www.modbus.org](http://www.modbus.org) verfügbar.

Modbus ist ein Master-Slave-Protokoll, das bis zu 247 Slaves aber nur 1 Master zur selben Zeit unterstützt. Der Master kann beispielsweise ein PC sein, auf welchem SolarInvert Control Center (SICC) installiert ist, oder ein anderes kompatibles Gerät wie z.B. ein PowerDog.

### 1.1 Pin-Belegung für RJ45 Stecker



### 1.2 Bus Layout



### 1.3 Terminierung

Um Signalstörungen zu vermeiden, müssen die beiden Endpunkte des Busses mit einem Endwiderstand terminiert werden. Verwenden Sie zur Verbindung der Geräte ein Twisted-Pair-Kabel und falls erforderlich einen 120 Ohm Widerstand zur Terminierung.

Geräte, die über nur eine RJ45 Buchse verfügen wie Windwechselrichter und BatteryInvert werden standardmäßig mit einem internen Endwiderstand ausgeliefert. Bei allen anderen Geräten kann ein Endwiderstand über die freie RJ45 Buchse angeschlossen werden.

## 1.4 Galvanische Trennung

Sollen Geräte von Fremdherstellern in den Bus integriert werden, muss zwingend darauf geachtet werden, dass der RS485-Bus galvanisch vom Stromkreis der Geräte getrennt ist.

## 1.5 Netzwerkmanagement und Adressvergabe

Nach dem Einschalten führt der Master einen Netzwerkscan durch, indem er alle verfügbaren Adressen im Netzwerk abfragt (Polling). Damit alle Geräte korrekt erkannt werden, muss sichergestellt sein, dass diese eingeschaltet und empfangsbereit sind.

Während sich ein Wechselrichter im Bootvorgang befindet, darf der Wechselrichter vom Master nicht angesprochen werden. Dies würde vom Bootloader als Versuch interpretiert, die Firmware zu flashen. Der Master muss dem Wechselrichter also einige Sekunden Zeit geben, da der Wechselrichter den Bootvorgang sonst nicht abschließen kann.



### HINWEIS

SolarInvert Wechselrichter schalten sich ohne DC-Quelle ab und sind dann nicht mehr über RS485 ansprechbar! Polling muss deshalb bei ausreichender Einstrahlung, Windstärke oder geschlossenem Batterietrennschalter erfolgen.

### Für die Adressvergabe der Slaves gelten folgende Bedingungen:

- Bei Auslieferung antwortet der Slave unter seiner Default Geräte-ID, welche basierend auf der Produktionsnummer zwischen 11 und 20 festgelegt wird.
- Im Fall eines Adresskonflikts kann das SolarInvert Control Center (SICC) benutzt werden, um die Geräte-ID zu ändern. Dabei darf alleine das Gerät mit dem PC verbunden sein, dessen Geräte-ID geändert werden soll.

## 2 Datenübermittlung

- Baud Rate 57600, 1 Startbit, 1 Stopbit, 8 Databits, No Parity
- High-Byte wird vor Low-Byte übertragen
- Bei allen Werten, die länger als 1 Byte sind, wird das MSB zuerst übertragen
- Die CRC Prüfsumme wird über die ganze Nachricht errechnet

### 2.1 Codebeispiele für die Checksummenberechnung

#### 2.1.1 Musterimplementierung zur Berechnung der Prüfsumme:

```
void CALC_ CRC16(unsigned char len)
{
    unsigned char    I;
    unsigned char    I1;
    unsigned short   CRC;
    unsigned short   LSB;
    CRC = 0xFFFF;
    for (I=0; I<len; I++)
    {
        CRC = ((CRC^TXD_TAB[I] | 0xFF00) &(CRC | 0x00FF));
        //TXD_TAB[I] = OUTGOING DATASTREAM
        for (I1=0; I1<8; I1++)
        {
            LSB=(CRC & 0x0001);
            CRC= CRC/2;
            if (LSB)
                CRC=CRC ^0xA001;
        }
    }
    TXD_TAB[I] = (CRC & 0x00FF);           // HCRC
    TXD_TAB[I+1] = (CRC & 0xFF00)/256;   // LCRC
}
```

### 2.1.2 Musterimplementierung zur Prüfung der Prüfsumme:

```

unsigned char CHECK_CRC16(unsigned char len)
{
    unsigned char    I;
    unsigned char    I1;
    unsigned short   CRC;
    unsigned short   LSB;
    unsigned char    RESULT;
    CRC = 0xFFFF;
    for (I=0; I<len; I++)
    {
        CRC = ((CRC^RXD_TAB[I]) | 0xFF00) &(CRC | 0x00FF);
        //RXD_TAB[I] = INCOMING DATASTREAM
        for (I1=0; I1<8; I1++)
        {
            LSB=(CRC & 0x0001);
            CRC= CRC/2;
            if (LSB)
                CRC=CRC ^0xA001;
        }
    }

    if ((RXD_TAB[I] == (CRC & 0x00FF)) && ((RXD_TAB[I+1] == (CRC &
0xFF00)/256)))
        return RESULT = true;
    else
        return RESULT = false;
}

```

## 2.1 Anfragetelegramme

Die Kommunikation auf dem Bus wird immer durch eine Anfrage des Masters initiiert. Dieser nutzt Funktionscodes, um Zugriff auf Informationen und Steuerungsoptionen der Slaves zu gewähren. Je nach Definition des Funktionscodes wird eine variable Menge Daten gesendet, die vom Slave entsprechend verarbeitet werden.

**Eine Anfrage hat folgende Telegrammstruktur:**

Slave-Adresse	Funktionscode	Daten	Prüfsumme
Slave Adresse	Geräte-ID des adressierten Slaves (1 Byte)		
Funktionscode	Bestimmt die Aktion, die der adressierte Slave ausführt (1 Byte)		
Daten	Abhängig vom verwendeten Funktionscode (variable Länge)		
Prüfsumme	CRC16 (2 Bytes)		

## 2.2 Antworttelegramme

Jede gesendete Anfrage (mit Ausnahme von Broadcasts) wird durch eine Antwort des adressierten Slaves quittiert. Die Antwort ist abhängig vom jeweils verwendeten Funktionscode. Falls der Funktionscode keine bestimmte Antwort definiert, wird der Slave eine Standardantwort senden.

**Eine Standardantwort hat folgende Telegrammstruktur:**

Slave-Adresse	Funktionscode	Länge (0x 01)	Daten (0x 01)	Prüfsumme
---------------	---------------	---------------	---------------	-----------

Wenn die empfangene Anfrage nicht bearbeitet werden konnte, wird stattdessen eine Ausnahmeantwort wird gesendet. Sie enthält einen Ausnahmecode, der Rückschlüsse auf die Ursache des Fehlers zulässt.

**Eine Ausnahmeantwort hat folgende Telegrammstruktur:**

Slave-Adresse	Funktionscode (+80h)	Ausnahmecode (s.u.)	Prüfsumme
---------------	----------------------	---------------------	-----------

Um die Ausnahmeantwort als solche erkennbar zu machen, wird der Funktionscode der Anfrage um 80h inkrementiert.

**Ausnahmecodes:**

Code	Bezeichnung	Beschreibung
0x 01	Ungültiger Funktionscode	Der verwendete Funktionscode wird nicht unterstützt
0x 02	Ungültige Registeradresse	Register nicht innerhalb des zulässigen Speicherbereichs
0x 03	Ungültiger Datenwert	Falsche Anzahl, Typen oder Wertebereiche der Argumente
0x 04	Gerätefehler	Es kam zu einem internen Gerätefehler

### 3 Standard Funktionscodes

Die Modbus Standardimplementierung verwendet Funktionscodes zum Lesen und Schreiben von 16-Bit-Registern, die im Arbeitsspeicher des Slaves liegen. Dies setzt jedoch genaue Informationen über die Adressen und Datentypen aller verfügbaren Register, sowie deren Bedeutung voraus und ist nur für Experten geeignet.

SolarInvert Geräte unterstützen grundsätzlich folgende Standard-Funktionscodes:

- 0x 03 (Auslesen von Holdingregistern) - **ab Firmware v.60**
- 0x 16 (Schreiben von Holdingregistern) - **ab Firmware v.60**
- 0x 08 („Echo“ Verbindungstest, nur Subfunktion 00) - **ab Firmware v.60**

Ab Firmware v.60 (**noch nicht veröffentlicht**) werden die Register gemäß dem SUNSPEC Standard organisiert sein. Mehr Informationen unter <http://sunspec.org/>

## 4 Proprietäre Funktionscodes

Als Alternative unterstützen SolarInvert Wechselrichter für die häufigsten Anwendungsfälle spezielle Funktionscodes, die ohne genaue Kenntnis der Registeradressen benutzt werden können und schnellen Zugriff auf die Grundfunktionen der Geräte ermöglichen.

Bei der Verwendung der proprietären Funktionscodes muss lediglich auf die korrekte Reihenfolge, die Anzahl und den Wertebereich der Argumente geachtet werden, welche vom Funktionscode vorgegeben werden. Adressierungsfehler sind somit ausgeschlossen.



### HINWEIS

Alle Datenwerte außer String sind als vorzeichenlose Ganzzahlen repräsentiert! Für negative Datenwerte wird das Zweierkomplementformat verwendet. Für Dezimalzahlen wird der Bitwert masterseitig mit einem Skalenfaktor multipliziert. Diese und weitere nützliche Informationen finden Sie in der Tabelle „verwendete Argumente“ des jeweiligen Funktionscodes.

### 4.1 Fabrikationsdaten auslesen (cmd 31h)

Mit dieser Anfrage erhalten Sie Informationen über Seriennummer, Produktionsdatum und Softwareversionen Ihres Wechselrichters. Die Modellbezeichnung kann nicht einzeln ausgelesen werden (siehe Abschnitt 4.2, cmd 3Eh).

#### Anfrage 31h:

Adresse	Funktionscode	Länge	Data	CRC
Adresse	0x 01	Geräte-ID des adressierten Slaves (1 Byte)		
Funktionscode	0x 31	Ruft die Funktion 31h auf (1 Byte)		
Länge	0x 01	Anzahl der Bytes im Data-Block (1 Byte)		
Data	0x 01	Keine Daten (1 Bytes)		
CRC	0x 90 47	Prüfsumme (2 Bytes)		

#### 4 Proprietäre Funktionscodes

##### Antwort 31h:

Adresse	Funktionscode	Länge	Data	CRC
Adresse	0x 01		Geräte-ID des adressierten Slaves (1 Byte)	
Funktionscode	0x 31		ruft die Funktion 31h auf (1 Byte)	
Länge	0x 0D		Anzahl der Bytes im Data-Block (1 Byte)	
Data	Bezeichner	Wert	Klartext	
	SNR	0x 00 01	1	Seriennummer (2 Bytes)
	PDATE	0x 1E 04 0D	30.04.13	Produktionsdatum TT-MM-JJ (3 Byte)
	BOM	0x 22	v.34	Hardware Versionsnummer (1 Byte)
	FW	0x 32	v.50	Firmware Version (1 Byte)
	BB-ID	0x 28	v.40	Bootstrapper Versionsnummer (1 Byte)
	RES	0x 00 00 00 00	0	reserviert, wird nicht ausgewertet (4 Bytes)
CRC	...		Prüfsumme (2 Bytes)	

##### Verwendete Argumente 31h:

Bezeichner	Typ	Format	Wertebereich	Skalenfaktor	Sonstiges
SNR	UInt16	0	0... 65535	1	
PDATE	3x UInt8	DD.MM.JJ	1...31 1...12 0...99	1	
BOM	UInt8	0	0...255	1	
FW	UInt8	0	0...255	1	
BB-ID	UInt8	0	0...255	1	

## 4.2 Betriebsdaten auslesen (cmd 33h/3Eh)

Zum Auslesen der Betriebsdaten des Wechselrichters stehen zwei Funktionen (33h und 3Eh) zur Verfügung. Wählen Sie die passende Funktion anhand der Daten, die Sie interessieren.

##### Anfrage 33h:

Adresse	Funktionscode	Länge	Data	CRC
Adresse	0x 01		Geräte-ID des adressierten Slaves (1 Byte)	
Funktionscode	0x 33		Ruft die Funktion 33h auf (1 Byte)	
Länge	0x 01		Anzahl der Bytes im Data-Block (1 Byte)	
Data	0x 01		Keine Daten (1 Bytes)	
CRC	0x 31 87		Prüfsumme (2 Bytes)	

#### 4 Proprietäre Funktionscodes

#### Antwort 33h:

Adresse	Funktionscode	Länge	Data	CRC
Adresse	0x 01		Geräte-ID des adressierten Slaves (1 Byte)	
Funktionscode	0x 33		ruft die Funktion 33h auf (1 Byte)	
Länge	0x 15		Anzahl der Bytes im Data-Block (1 Byte)	
Data	Bezeichner	Wert	Klartext	
	UDC	0x 01 90	74,4 V	DC-Spannung (2 Bytes) Skalenfaktor 0,186 s.u.
	RES	0x B4	reserviert	Reserviert für interne Zwecke (1 Byte)
	UAC	0x B4	230V	AC-Spannung nach Softwarekorrektur (1 Byte) +32h
	RES	0x FA	reserviert	Reserviert für interne Zwecke (1 Byte)
	FREQ	0x 13 88	50,00 Hz	Netzfrequenz in Hz (2 Bytes) Skalenfaktor 0,01
	PAC	0x 03 84	+90 W	Istwert der AC-Leistung (2 Bytes) Skalenfaktor 0,1
	RES	0x FA	reserviert	Reserviert für interne Zwecke (1 Byte)
	STATE	0x 03	MPPT aktiv	Wechselrichterstatus (1 Byte)
RES	0x 00 00 00 ...	Intern	Reserviert, wird nicht ausgewertet (10 Bytes)	
CRC	...		Prüfsumme (2 Bytes)	

#### Verwendete Argumente 33h:

Bezeichner	Typ	Format	Wertebereich	Skalenfaktor	Sonstiges
UDC	Uint16	0 V	0...65535	Volt=01: 0,028 V Volt=02: 0,054 V Volt=03: 0,054 V Volt=04: 0,079 V Volt=05: 0,108 V Volt=06: 0,142 V Volt=07: 0,186 V Volt=08: 0,244 V Volt=09: 0,357 V Volt=10: 0,479 V	Skalenfaktor ist abhängig vom Spannungstyp des Wechselrichters, siehe Rückgabewert VOLT (cmd 3Eh)
UAC	Uint8	0 V	50...305	1	Offset +32h 0x 00 = 50V
FREQ	Uint16	0 Hz	0...65535	0,01	
PAC	Uint16	0 W	0...32767/ -32768...-1	0,1	Binär-komplement
State	Uint8		00: Init level 1 01: Init level 2 02: Init level 3 03: MPPT 04: Reset 05: Konstantspng. 06: Low Power Mode 07: Wind 08: Factory Mode 09: Battery 10: Battery Safety	1	



#### HINWEIS

Der korrekte Skalenfaktor für UDC ist anhand des Rückgabewerts VOLT zu wählen. VOLT kann mit Funktionscode 3Eh abgefragt werden (siehe unten).

#### 4 Proprietäre Funktionscodes

##### Anfrage 3Eh:

Adresse	Funktionscode	Länge	Data	CRC
Adresse	0x 01	Geräte-ID des adressierten Slaves (1 Byte)		
Funktionscode	0x 3E	Ruft die Funktion 3Eh auf (1 Byte)		
Länge	0x 01	Anzahl der Bytes im Data-Block (1 Byte)		
Data	0x 01	Keine Daten (1 Bytes)		
CRC	0x A0 44	Prüfsumme (2 Bytes)		

##### Antwort 3Eh:

Adresse	Funktionscode	Länge	Data	CRC
Adresse	0x 01	Geräte-ID des adressierten Slaves (1 Byte)		
Funktionscode	0x 3E	ruft die Funktion 3Eh auf (1 Byte)		
Länge	0x 18	Anzahl der Bytes im Data-Block (1 Byte)		
Data	Bezeichner	Wert	Klartext	
	SNR	0x 30 39	12345	Seriennummer (2 Bytes)
	TYP	0x 01	solar	Wechselrichtertyp (1 Byte)
	VOLT	0x 07	120 V	Spannungstyp (1 Byte)
	PNOM	0x 06	600 W	Leistungsklasse (1 Byte)
	COSPHI	0x 78	0,90 (k)	Power Faktor $\cos \varphi$ (1 Byte) Skalenfaktor 0,005
	E	0x 0D 4D C3 00	6200 Wh	Tagessumme der Netzeinspeisung (4 Byte)
	PMAX	0x 07 6C	1900 W	Aktuell gültige Leistungsbegrenzung in W (2 Byte)
	OPH	0x 00 7E 57 C0	230 h	Summe Betriebsstunden (4 Byte)
	PHASE	0x 02	2	Anschlussphase (1 Byte)
	JMPR	0x 04	4	Modbus Geräte ID per Jumper gesetzt (1 Byte)
RES	0x 00 00 00 00	00	reserviert, wird nicht ausgewertet (4 Bytes)	
CRC	...	Prüfsumme (2 Bytes)		

#### 4 Proprietäre Funktionscodes

##### Verwendete Argumente 3Eh:

Bezeichner	Typ	Format	Wertebereich	Skalenfaktor	Sonstiges			
SNR	Uint16	0	0...65535	1				
TYP	Uint8	0	01 = SOL 02 = WIN 03 = BAT	1				
VOLT	Uint8	0	VOLT	1	Spannungstyp des Wechselrichters in Abhängigkeit vom Einsatzbereich			
			01			12 V	12 V	8 V
			02			17 V	18 V	12 V
			03			35 V	24 V	24 V
			04			50 V	36 V	30 V
			05			70 V	48 V	36 V
			06			90 V	72 V	48 V
			07			120 V	96 V	96 V
			08			160 V	120 V	120 V
			09			240 V	160 V	160 V
			10			320 V	240 V	240 V
PNOM	Uint8	0 W	01 ... 32	100				
COSPHI	Uint16	0,00 (i/k)	0,9 (i)...1...0,9 (k)	1±(0x64-COSPHI)*0,005	0x 64 = 1 0x 50 = 0,9 (i) 0x 78 = 0,9 (k)			
E	Uint32	0,0 Wh	0...4294967295	1/36000				
PMAX	Uint16	0 W	0...32767/-32768...-1	0,1	Binär-komplement			
OPH	Uint32	0 h	0...4294967295	1/36000				
PHASE	Uint8	0	0...3	1				
JMPR	Uint8	0	0...15	1				



#### HINWEIS

Die genaue Modellbezeichnung des Wechselrichters setzt sich aus den Rückgabewerten „TYP“, „PNOM“ und „VOLT“ zusammen. Beispiel: SOL 1900-70

TYP [01] = SOL  
PNOM [19] = 1900  
VOLT [06] = 70

### 4.3 Wechselrichter ausschalten (cmd 34h/35h)

Sie können den Wechselrichter ausschalten, indem Sie die H-Brücke abschalten und den Wechselrichter über die integrierten Relais vom Hausnetz trennen.

#### Anfrage 34h:

Adresse	Funktionscode	Länge	Data	CRC
Adresse	0x 01	Geräte-ID des adressierten Slaves (1 Byte)		
Funktionscode	0x 34	Ruft die Funktion 34h auf (1 Byte)		
Länge	0x 01	Anzahl der Bytes im Data-Block (1 Byte)		
Data	0x 01	Keine Daten (1 Bytes)		
CRC	0x 80 46	Prüfsumme (2 Bytes)		

#### 4 Proprietäre Funktionscodes

Der Wechselrichter befindet sich nun im Betriebszustand INIT (00). Um den Betrieb wieder aufzunehmen, müssen Sie Relais und H-Brücke mit Funktionscode 35h wieder aktivieren oder den Wechselrichter von Hand zurücksetzen (DC-Spannung trennen). Nach Trennen der DC-Spannung wechselt das Gerät dann falls möglich automatisch in die Standard-Betriebsart je nach Gerätetyp.

##### Anfrage 35h:

Adresse	Funktionscode	Länge	Data	CRC
<b>Adresse</b>	0x 01	Geräte-ID des adressierten Slaves (1 Byte)		
<b>Funktionscode</b>	0x 35	Ruft die Funktion 35h auf (1 Byte)		
<b>Länge</b>	0x 01	Anzahl der Bytes im Data-Block (1 Byte)		
<b>Data</b>	0x 01	Keine Daten (1 Bytes)		
<b>CRC</b>	0x D1 86	Prüfsumme (2 Bytes)		

Nach Wiedereinschalten mit Funktionscode 35h nimmt das Gerät wieder den Betriebszustand ein, der vor dem Abschalten mit Funktionscode 34h vorlag.

#### 4.4 Leerlauf erzwingen (cmd 36h/37h)

Sie können mit dem Funktionscode 36h die PWM der H-Brücke auf 0%-Last stellen und den Wechselrichter so in den Leerlauf bringen, ohne den Betriebszustand oder den Relais-Status zu beeinflussen.

##### Anfrage 36h:

Adresse	Funktionscode	Länge	Data	CRC
<b>Adresse</b>	0x 01	Geräte-ID des adressierten Slaves (1 Byte)		
<b>Funktionscode</b>	0x 36	Ruft die Funktion 36h auf (1 Byte)		
<b>Länge</b>	0x 01	Anzahl der Bytes im Data-Block (1 Byte)		
<b>Data</b>	0x 01	Keine Daten (1 Bytes)		
<b>CRC</b>	0x 21 86	Prüfsumme (2 Bytes)		

Stellen Sie anschließend sicher, dass Sie den Leerlauf mit 37h wieder deaktivieren:

##### Anfrage 37h:

Adresse	Funktionscode	Länge	Data	CRC
<b>Adresse</b>	0x 01	Geräte-ID des adressierten Slaves (1 Byte)		
<b>Funktionscode</b>	0x 37	Ruft die Funktion 37h auf (1 Byte)		
<b>Länge</b>	0x 01	Anzahl der Bytes im Data-Block (1 Byte)		
<b>Data</b>	0x 01	Keine Daten (1 Bytes)		
<b>CRC</b>	0x 70 46	Prüfsumme (2 Bytes)		

## 4.5 In den Konstantspannungsmodus wechseln bzw. Spannung steuern (cmd 38h/39h)

SOL- und WIN-Geräte können mit dem Funktionscode 38h in den Konstantspannungsbetrieb versetzt werden. Dabei kann die Eingangsspannung vorgeben oder auf 0 gesetzt werden, um die interne Voreinstellung zu verwenden. Der Modus bleibt aktiv, bis er mit dem Funktionscode 39h beendet oder der Wechselrichter durch trennen der DC-Spannung zurückgesetzt wird. Wird der Funktionscode 38h mit einer Vorgabe von 0V gesendet, wechselt das Gerät in den Konstantspannungsmodus und verwendet den Parameter U\_SOLCV als Spannungsvorgabe.

Beim BatteryInvert wird der Funktionscode 38h verwendet, um in den Modi 0 und 3 die Spannungsvorgabe an den Wechselrichter zu senden. Wird an den BatteryInvert eine Spannungsvorgabe von 0V gesendet, so regelt der Wechselrichter alleine auf die mit Funktionscode 3Fh gesendete Leistung. Ist diese ebenfalls 0, so regelt der BatteryInvert in Modus 0 auf den Parameter U\_SOLBAT, in Modus 3 wechselt er in den Sicherheitsmodus. Ist auch U\_SOLBAT=0V, so wechselt das Gerät auch in Modus 0 in den Sicherheitsmodus. In Modus 3 wird beim Empfangen von 38h außerdem das Timeout zurückgesetzt.

### Anfrage 38h:

Adresse	Funktionscode	Länge	Data	CRC
<b>Adresse</b>	0x 01	Geräte-ID des adressierten Slaves (1 Byte)		
<b>Funktionscode</b>	0x 38	ruft die Funktion 38h auf (1 Byte)		
<b>Länge</b>	0x 02	Anzahl der Bytes im Data-Block (1 Byte)		
<b>Data</b>	Bezeichner	Wert	Klartext	
	USET	0x 02 E8	74,4 V	DC-Spannung (2 Bytes) Skalenfaktor 10
<b>CRC</b>	0x B4 4E	Prüfsumme (2 Bytes)		

### Verwendete Argumente 38h:

Bezeichner	Typ	Format	Wertebereich	Skalenfaktor	Sonstiges
USET	Uint16	0 V	0...65535	10	

Um den Konstantspannungsmodus bei SOL und WIN zu beenden, nutzen Sie die Funktion 39h. Beim BatteryInvert ist der Funktionscode 39h ohne Funktion. Soll hier eine Spannungsvorgabe aufgehoben werden, ohne sie durch einen neuen Wert zu ersetzen, so ist ein Vorgabewert von 0V zu senden.

### Anfrage 39h:

Adresse	Funktionscode	Länge	Data	CRC
<b>Adresse</b>	0x 01	Geräte-ID des adressierten Slaves (1 Byte)		
<b>Funktionscode</b>	0x 39	Ruft die Funktion 39h auf (1 Byte)		
<b>Länge</b>	0x 01	Anzahl der Bytes im Data-Block (1 Byte)		
<b>Data</b>	0x 01	Keine Daten (1 Bytes)		
<b>CRC</b>	0x 11 85	Prüfsumme (2 Bytes)		

## 4.6 Leistungsfaktor $\cos \varphi$ festlegen (cmd 3Bh)

Sie können dem Wechselrichter einen Sollwert für den Leistungsfaktor  $\cos \varphi$  vorgeben.

### Anfrage 3Bh:

Adresse	Funktionscode	Länge	Data	CRC
<b>Adresse</b>	0x 01	Geräte-ID des adressierten Slaves (1 Byte)		
<b>Funktionscode</b>	0x 3B	ruft die Funktion 3Bh auf (1 Byte)		
<b>Länge</b>	0x 01	Anzahl der Bytes im Data-Block (1 Byte)		
<b>Data</b>	Bezeichner	Wert	Klartext	
	COSPHI	0x 78	0,90 (k)	Leistungsfaktor $\cos \varphi$ (1 Byte) Skalenfaktor 0,005
<b>CRC</b>	0x 71 A7	Prüfsumme (2 Bytes)		

### Verwendete Argumente 3Bh:

Bezeichner	Typ	Format	Wertebereich	Skalenfaktor	Sonstiges
COSPHI	Uint8	0,00 (i/k)	0,9 (i)...1...0,9 (k)	$1 \pm (0x64 - \text{COSPHI}) * 0,005$	0x 64 = 1 0x 50 = 0,9 (i) 0x 78 = 0,9 (k)

Diese Vorgabe für den Leistungsfaktor wird nicht gespeichert und ist somit nur solange gültig, bis der Wechselrichter neu gestartet wird.

## 4.7 Wechselrichterleistung festlegen (cmd 3Fh/44h)

Sie haben je nach Gerätetyp zwei unterschiedliche Möglichkeiten, die Leistung des Wechselrichters zu steuern bzw. zu begrenzen.



### HINWEIS

Der Wechselrichter fährt diesen Punkt sofort an (falls möglich). Für Batterie-wechselrichter bedeutet ein positiver Wert Entladung der Batterie (Einspeisung), während ein negativer eine Beladung (Netzbezug) bedeutet.

### 4.7.1 Steuerung/Begrenzung der Leistung beim BatteryInvert in Modus 0 und 3

Dem BatteryInvert kann im Modus 0 oder 3 mit dem Funktionscode 3Fh die gewünschte Leistung vorgegeben werden. Außerdem wird mit dem Kommando Umin und Umax gesendet, mit denen das bei Auslieferung eingestellte Spannungsfenster (U\_MIN...U\_MAX) weiter eingeschränkt werden kann.

## 4 Proprietäre Funktionscodes

Der Wert für das Timeout ist in Modus 3 notwendig, um die Timeout-Funktion zu initialisieren. D.h. auch wenn der BatteryInvert nur über die Spannung (mit 38h) gesteuert werden soll, ist beim Einschalten einmalig 3Fh zu senden. Wenn hierbei die Leistung nicht begrenzt werden soll, so ist PMAX mit 0 oder einem Wert über der Maximalleistung des Wechselrichters zu senden. Das Timeout wird jedes Mal zurückgesetzt, wenn eine Nachricht mit 38h oder 3Fh empfangen wird. D.h. im Modus 3 muss regelmäßig innerhalb des mit 3Fh gesendeten Intervalls eine Spannungs- oder Leistungsvorgabe gesendet werden.

### Anfrage 3Fh:

Adresse	Funktionscode	Länge	Data	CRC
Adresse	0x 01	Geräte-ID des adressierten Slaves (1 Byte)		
Funktionscode	0x 3F	ruft die Funktion 3Fh auf (1 Byte)		
Länge	0x 0D	Anzahl der Bytes im Data-Block (1 Byte)		
Data	Bezeichner	Wert	Klartext	
	PMAX	0x 012c	300W	entladen 300W
	Umin	0x 016a	36,2V	
	Umax	0x 03e8	100,0V	
	Timeout	0x 0096	30s	30s
	reserviert	00 00 00 00 00		
CRC	0x BE D9	Prüfsumme (2 Bytes)		

### Verwendete Argumente 3Fh:

Bezeichner	Typ	Format	Wertebereich	Skalenfaktor	Sonstiges
PMAX	Uint16	0 W	0...3200	1,0	MSB=1: laden, MSB=0: entladen
Umin	Uint16	0 V	U_MIN...Umax	10	
Umax	Uint16	0 V	Umin...U_MAX	10	
Timeout	Uint16	0 s	15s...12000s	5	

Diese Werte werden alle nicht gespeichert, d.h. sie sind nur bis zum nächsten Neustart gültig.

## 4.7.2 Leistungsbegrenzung Solarwechselrichter

Die gewünschte Leistung wird relativ zur am Wechselrichter angeschlossenen PV-Generatorleistung (in %) angegeben:

### Anfrage 44h:

Adresse	Funktionscode	Länge	Data	CRC
Adresse	0x 01	Geräte-ID des adressierten Slaves (1 Byte)		
Funktionscode	0x 44	ruft die Funktion 44h auf (1 Byte)		
Länge	0x 01	Anzahl der Bytes im Data-Block (1 Byte)		
Data	Bezeichner	Wert	Klartext	
	Pmax_rel	0x 46	70%	Max. Einspeiseleistung in % (1 Bytes)
CRC	0x C1 AF	Prüfsumme (2 Bytes)		

#### 4 Proprietäre Funktionscodes

##### Verwendete Argumente 44h:

Bezeichner	Typ	Format	Wertebereich	Skalenfaktor	Sonstiges
Pmax_rel	Uint8	0 %	0...100	1	0

Pmax\_rel wird im Wechselrichter gespeichert, d.h. die Begrenzung ist so lange gültig, bis sie mit einem neuen Wert überschrieben wird und greift auch noch nach einem Neustart. Um die Begrenzung zu deaktivieren wird sie einfach wieder auf 100% gesetzt.

## 4.8 Fehlerlogbuch auslesen (cmd 40h)

SolarInvert Wechselrichter verfügen über Zähler, welche Auskunft über die Anzahl bestimmter Fehler-typen während der Lebensdauer geben. Diese Zähler können ausgelesen aber nicht manipuliert werden.

##### Anfrage 40h:

Adresse	Funktionscode	Länge	Data	CRC
Adresse	0x 01	Geräte-ID des adressierten Slaves (1 Byte)		
Funktionscode	0x 40	Ruft die Funktion 40h auf (1 Byte)		
Länge	0x 01	Anzahl der Bytes im Data-Block (1 Byte)		
Data	0x 01	Keine Daten (1 Bytes)		
CRC	0x C0 5C	Prüfsumme (2 Bytes)		

##### Antwort 40h:

Adresse	Funktionscode	Länge	Data	CRC
Adresse	0x 01	Geräte-ID des adressierten Slaves (1 Byte)		
Funktionscode	0x 40	ruft die Funktion 40h auf (1 Byte)		
Länge	0x 28	Anzahl der Bytes im Data-Block (1 Byte)		
Data	Bezeichner	Wert	Klartext	
	ACMX	0x 00 03	3	Überspannung AC (2 Bytes)
	ACMN	0x 00 03	3	Unterspannung AC (2 Bytes)
	DCMX	0x 00 10	16	Überspannung DC (2 Bytes)
	DCMN	0x 01 00	256	Unterspannung DC (2 Bytes)
	FMX	0x 02 00	512	Überfrequenz (2 Bytes)
	FMN	0x 01 00	256	Unterfrequenz (2 Bytes)
	OT1	0x 00 F0	15	Temperaturabh. Leistungsred. 87,5% (2 Bytes)
	OT2	0x 00 F0	15	Temperaturabh. Leistungsred. 75% (2 Bytes)
	OT3	0x 00 F0	15	Temperaturabh. Leistungsred. 62,5% (2 Bytes)
	OT4	0x 00 F0	15	Temperaturabh. Leistungsred. 50% (2 Bytes)
	OT5	0x 00 F0	15	Temperaturabh. Leistungsred. 37,5% (2 Bytes)
	COM	0x 00 00	0	nicht implementiert (2 Bytes)
	ISL	0x 00 F0	15	Netzausfall (2 Bytes)
	RES	0x 00 00	0	reserviert, für interne Zwecke (2 Bytes)
	RES	0x 00 00	0	reserviert, für interne Zwecke (2 Bytes)
	OVRCURR	0x 00 00	0	Over Current Interrupt ausgelöst (2 Bytes)
RES	0x 00 00	0	reserviert, für interne Zwecke (2 Bytes)	
OVRVOLT	0x 00 00	0	Überspannungs-Interrupt ausgelöst (2 Bytes)	
AVRG	0x 00 00	0	10min-Durchschnitt Netzspng. zu hoch (2 Bytes)	

#### 4 Proprietäre Funktionscodes

	OVRHT	0x 00 00	0	Übertemperaturabschaltung (2 Bytes)
CRC		...		Prüfsumme (2 Bytes)

#### Verwendete Argumente 40h:

Bezeichner	Typ	Format	Wertebereich	Skalenfaktor	Sonstiges
ACMX	Uint16	0	0...65535	1	0
ACMN	Uint16	0	0...65535	1	0
DCMX	Uint16	0	0...65535	1	0
DCMN	Uint16	0	0...65535	1	0
FMX	Uint16	0	0...65535	1	0
FMN	Uint16	0	0...65535	1	0
OT1	Uint16	0	0...65535	1	0
OT2	Uint16	0	0...65535	1	0
OT3	Uint16	0	0...65535	1	0
OT4	Uint16	0	0...65535	1	0
OT5	Uint16	0	0...65535	1	0
COM	Uint16	0	0...65535	1	0
ISL	Uint16	0	0...65535	1	0
RES	Uint16	0	0...65535	1	0
RES	Uint16	0	0...65535	1	0
OVRCURR	Uint16	0	0...65535	1	0
RES	Uint16	0	0...65535	1	0
OVRVOLT	Uint16	0	0...65535	1	0
AVRG	Uint16	0	0...65535	1	0

## 5 Kundendienst und Garantie

### Erreichbarkeit

Sollte einmal eine Störung an einem unserer Produkte auftreten, stehen wir Ihnen natürlich mit Rat und Tat zur Seite.

**SolarInvert GmbH | Kundendienst | Monreposstr. 49 | D - 71634 Ludwigsburg**  
Tel. 07141/299 21 13 | Fax 07141/299 21 21 | [info@solarinvert.de](mailto:info@solarinvert.de)

Unseren Kundendienst erreichen Sie telefonisch zu unseren Geschäftszeiten.



### HINWEIS

Die Garantiebedingungen und technische Daten finden Sie unter:  
[www.solarinvert.de](http://www.solarinvert.de)