

# Thema: **Modbus RTU**

Bezug: Steuerung SK 5xxE über Modbus RTU per PC

Autor: Tim Köller


Datum: 17.08.2020

Einstellungen am Frequenzumrichter:

Damit der Frequenzumrichter über Modbus RTU gesteuert werden kann, müssen folgende Voraussetzungen geschaffen werden:


- Parameter P509 (Quelle Steuerwort) auf Einstellung ‚[2] USS /Modbus‘ stellen
- Parameter P511 (USS Baudrate) muss auf die des Masters angepasst werden (Werkseinstellung: 38400 baud)
- Parameter P512 (USS-Adresse) darf nicht auf 0 oder 2 stehen, da diese reserviert sind (Werkseinstellung: 0)
- Der Frequenzumrichter muss über die Klemme X14 (RJ12-Buchse) zum Modbus-Master verbunden werden.

Belegung der X14:

Schnittstelle Kommunikation X14		Anschluss des Gerätes an verschiedene Kommunikationstools		
		24 V DC ± 20 %	RS485 (Zum Anschluss einer Parametrierbox) 9600 ... 115000 Baud Abschlusswiderstand (1 kΩ) fest RS232 (Zum Anschluss an einen PC, NORDCON, NORDCON APP) 9600 ... 115000 Baud	
1	RS485 A+	Datenleitung RS485	P502...	 1 - 2 - 3 - 4 - 5 - 6
2	RS485 B-	Datenleitung RS485	P513 [-02]	
3	GND	Bezugspotential Bussignale		
4	RS232 TXD	Datenleitung RS232		
5	RS232 RXD	Datenleitung RS232		
6	+24 V	Spannung Ausgang		

Für die Kommunikation über Modbus sind die Datenleitungen der RS485-Schnittstelle zu verwenden.

- Je nach Topologie ist der Abschlusswiderstand via DIP-Schalter ‚S1‘ am Frequenzumrichter zu aktivieren.

DIP-Schalter USS/CAN S1/S2		
USS	Abschlusswiderstand für RS485-Schnittstelle (RJ12); ON = zugeschaltet [Default = „OFF“] Bei RS232-Kommunikation auf „OFF“	
CAN	Abschlusswiderstand für CAN/CANopen-Schnittstelle, ON = zugeschaltet [Default = „OFF“]	

Für die Ansteuerung des Frequenzumrichters wird in diesem Beispiel das Programm *ModLink Demo* verwendet.

# Thema: Modbus RTU

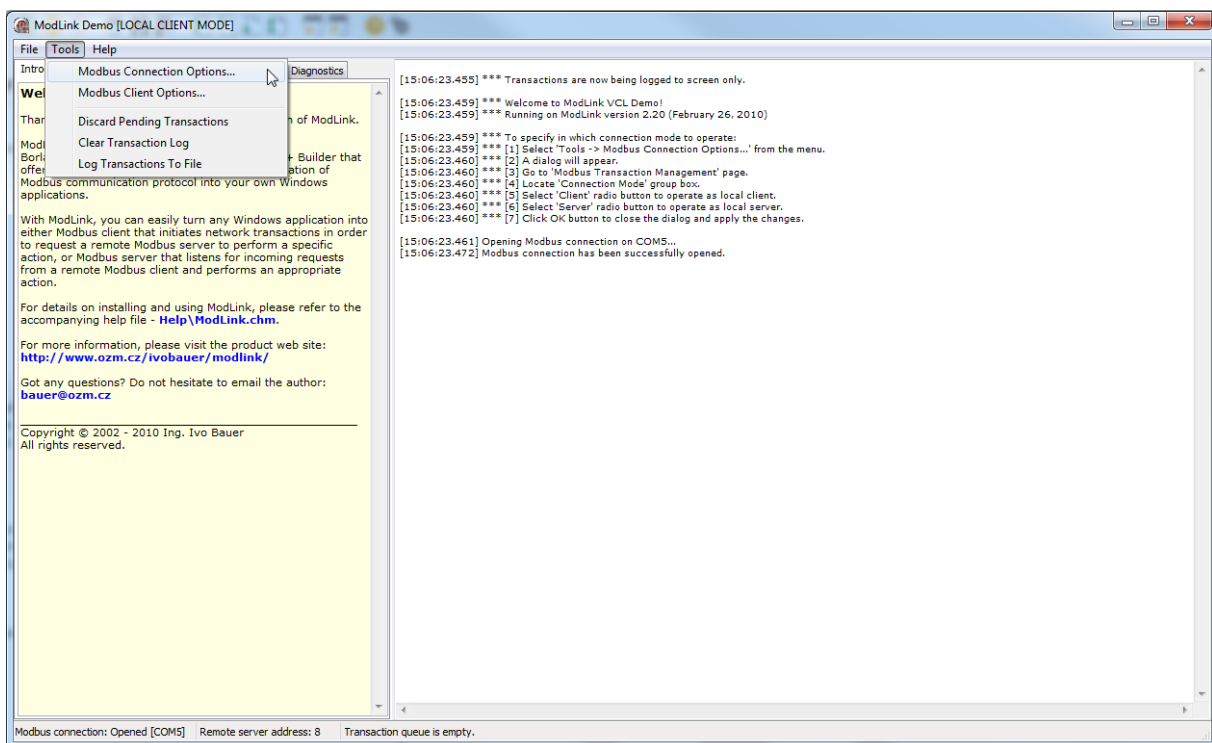
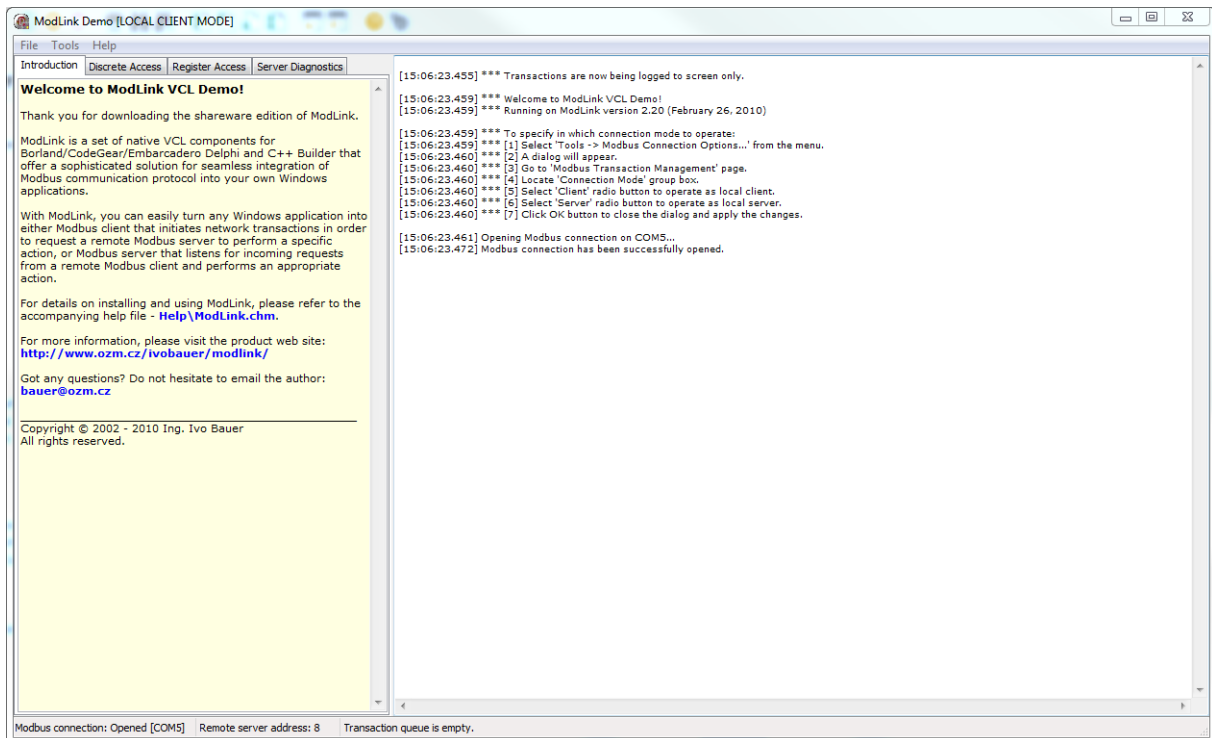
Bezug: Steuerung SK 5xxE über Modbus RTU per PC

Autor: Tim Köller

Datum: 17.08.2020



## Startfenster:

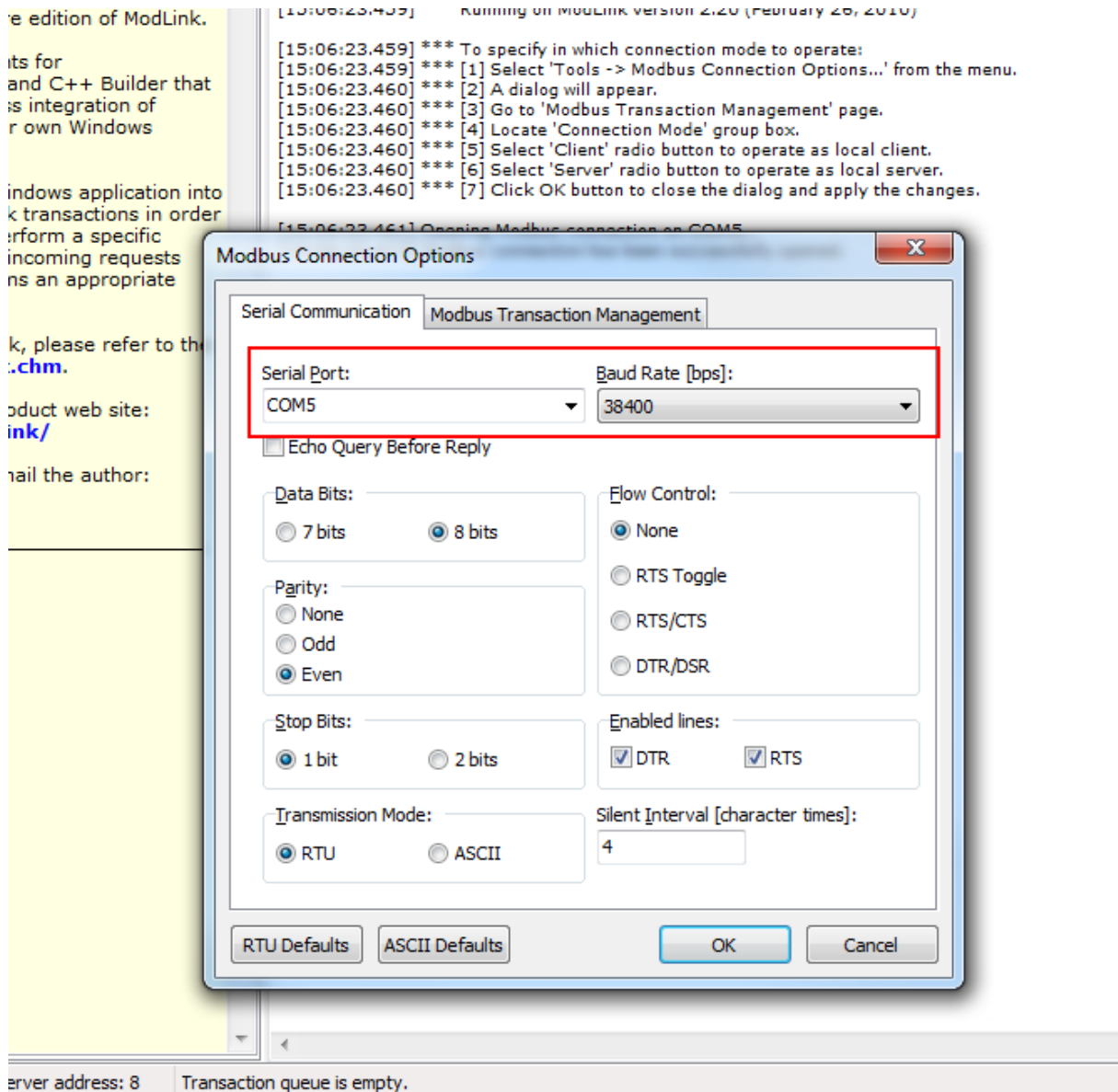


## Thema: Modbus RTU

Bezug: Steuerung SK 5xxE über Modbus RTU per PC

Autor: Tim Köller

Datum: 17.08.2020



Den entsprechenden COM-Port des USB-Umsetzers auswählen, an dem der FU angeschlossen ist. Die Baudrate ist an die des Umrichters aus Parameter 511 anzupassen (Standard 38.400).

Weiterhin sind folgende Einstellungen zu berücksichtigen:

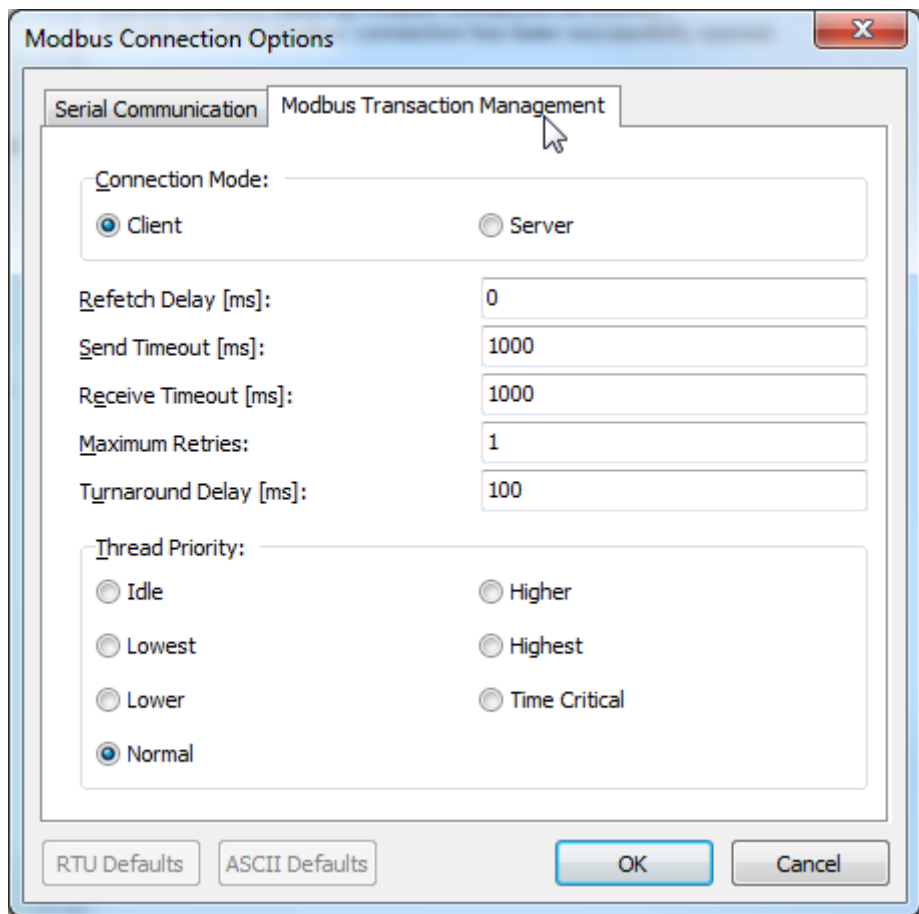
- Anzahl der Datenbits: 8
- Parität: Gerade
- Anzahl der Stop Bits: 1

## Thema: **Modbus RTU**

Bezug: Steuerung SK 5xxE über Modbus RTU per PC

Autor: Tim Köller

Datum: 17.08.2020



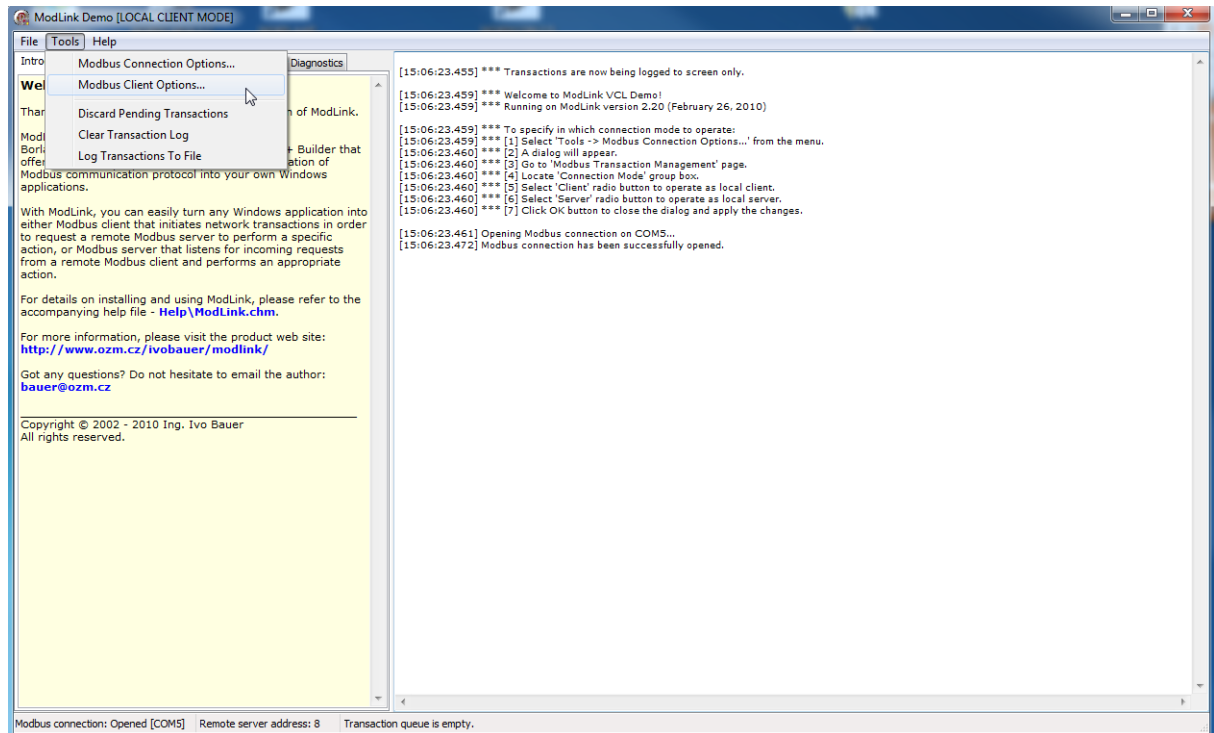
Unter dem Reiter „Modbus Transaction Management“ ist als *Connection Mode* Client einzustellen.

# Thema: Modbus RTU

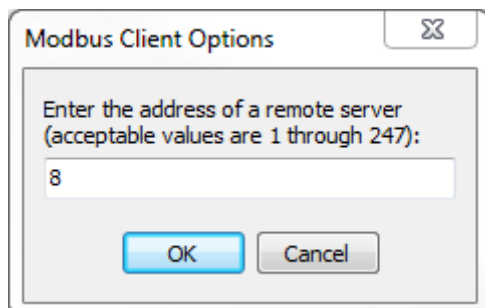
Bezug: Steuerung SK 5xxE über Modbus RTU per PC

Autor: Tim Köller

Datum: 17.08.2020



Unter „Modbus Client Options...“ wird die parametrisierte USS-Adresse des Umrichters aus Parameter 512 eingetragen.



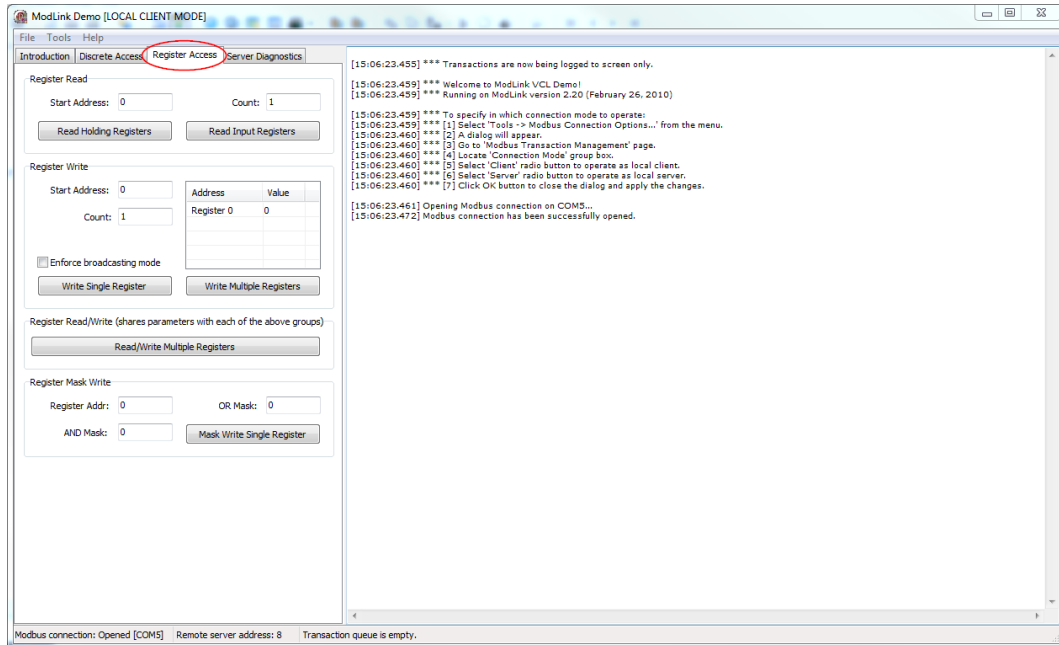
# Thema: Modbus RTU

Bezug: Steuerung SK 5xxE über Modbus RTU per PC

Autor: Tim Köller

Datum: 17.08.2020

Das Lesen / Schreiben von einzelnen oder mehreren Wörtern kann unter dem Reiter „Register Access“ durchgeführt werden.



Um Prozessdaten (Steuerwort, Sollwerte 1...5 bzw. Zustandswort, Istwerte 1...5) zu lesen oder zu schreiben, müssen die Adressen der Parameter bekannt sein. Diese sind in der folgenden Tabelle aufgeführt:

**Tabelle 1: Adressen für Prozessdatenkommunikation**

Adresse	Prozessdaten	Ankommende Daten
3200	Steuerwort	P740[1]
3201	Sollwert 1 (P546[1])	P740[2]
3202	Sollwert 2 (P546[2])	P740[3]
3203	Sollwert 3 (P546[3])	P740[4]
3204	Sollwert 4 (P546[4])	P740[5]
3205	Sollwert 5 (P546[5])	P740[6]
3264	Zustandswort	P741[1]
3265	Istwert 1 (P543[1])	P741[2]
3266	Istwert 2 (P543[2])	P741[3]
3267	Istwert 3 (P543[3])	P741[4]
3268	Istwert 4 (P543[4])	P741[5]
3269	Istwert 5 (P543[5])	P741[6]

# Thema: Modbus RTU

Bezug: Steuerung SK 5xxE über Modbus RTU per PC

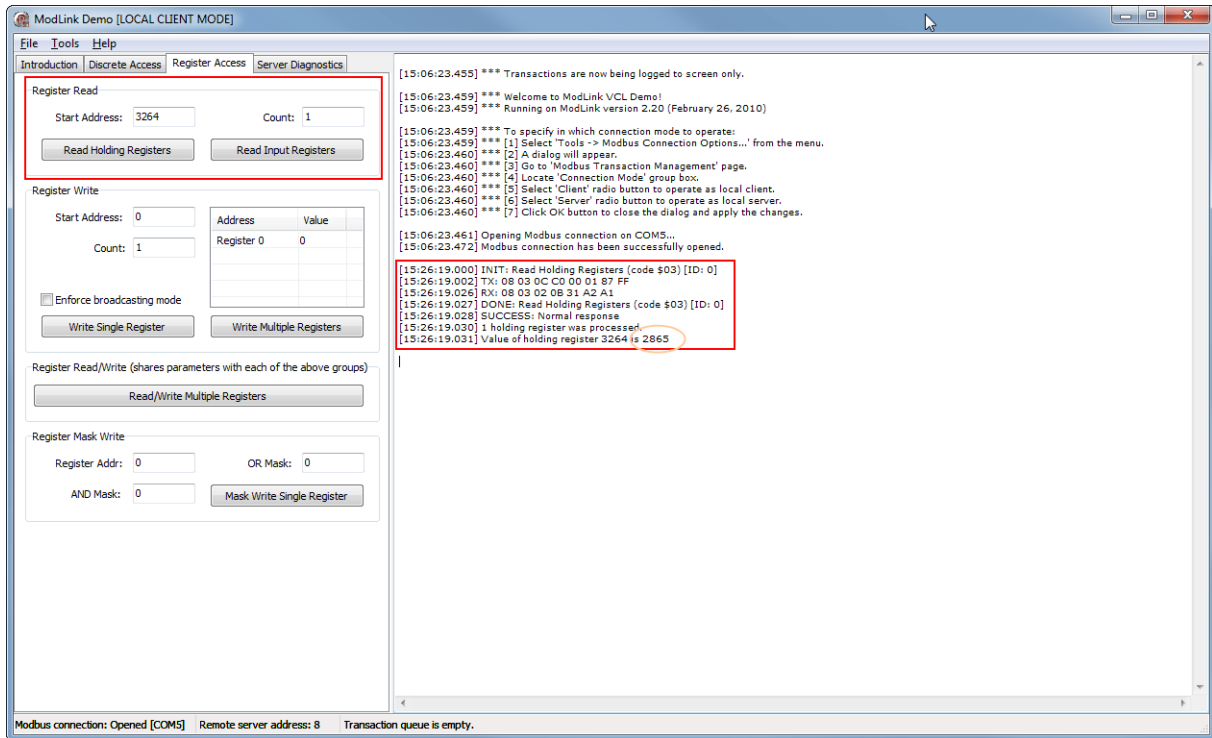
Autor: Tim Köller

Datum: 17.08.2020

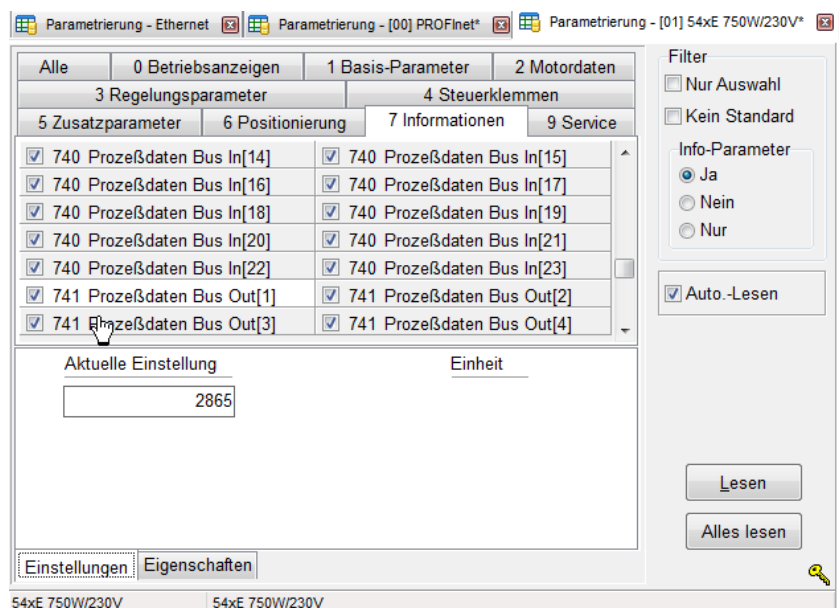


Zum Lesen von Wörtern wird der obere Bereich benutzt. Um z.B. das Zustandswort des Umrichters auszulesen, muss als Adresse 3264 eingetragen werden.

Count gibt die Anzahl der Wörter an, die gelesen werden sollen. Um nur das Zustandswort auszulesen, kann Count auf 1 gesetzt werden. Um neben dem Zustandswort auch die Istwerte 1...3 auslesen zu können, kann der Wert von Count auf 4 erhöht werden.



In dem Terminal wird die Anfrage des Masters sowie die Antwort des Slaves (Umrichter) angezeigt. Der ausgelesene Wert beträgt 2865. Dies stimmt mit dem Wert des Umrichters aus dem Parameter 741[1] überein:



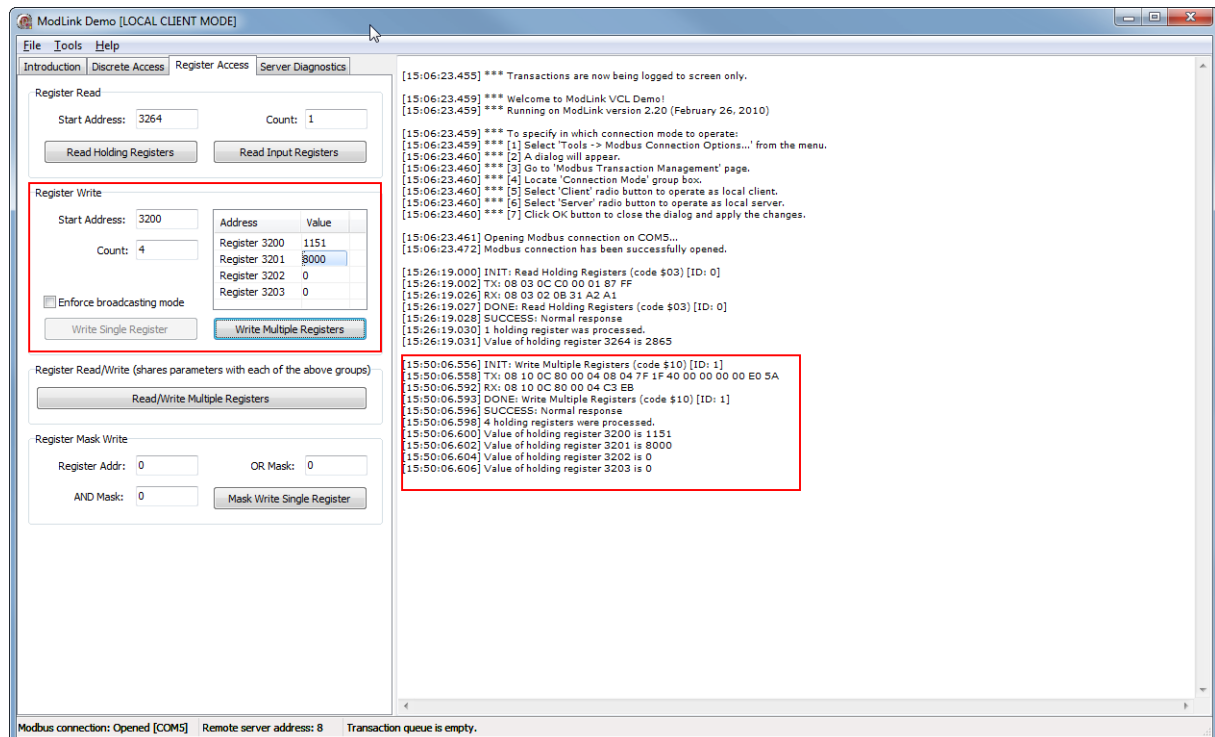
# Thema: Modbus RTU

Bezug: Steuerung SK 5xxE über Modbus RTU per PC

Autor: Tim Köller

Datum: 17.08.2020

Das vorgeben von Prozessdaten (Steuerwort, Sollwerte) erfolgt unter der Rubrik „Register Write“. Hier muss die Startadresse eingegeben werden (3200 für Steuerwort) und mit Count wird erneut eingegeben, wie viele Wörter von der Startadresse aus geschrieben werden.



Mit dem Steuerwort „1151“ wird der Umrichter vom Zustand „Einschaltbereit“ in den Zustand „Umrichter läuft“ versetzt.

Im Register 3201 wird der erste Sollwert (im Standard: Sollfrequenz) übergeben. Der Wert von 8000 entspricht dabei etwa 50% der eingestellten Maximalfrequenz des Umrichters im Parameter 103. Mit der Datenübertragung wird der Umrichter also freigegeben und erhält einen Frequenzsollwert von ca. 25Hz:

Der grundsätzliche Aufbau des Modbus-Telegramms ist folgendermaßen definiert:

Adresse	Funktion codes	Daten	CRC
8 Bit	8 Bit	Variable = N x 8 Bits	16 Bit

Abbildung 1: Modbus - Frameaufbau

Dabei entspricht die Adresse der USS-Adresse des Frequenzumrichters (P512).

Der Funktions-Code gibt an, ob einzelne Bits oder einzelne/mehrere Wörter geschrieben oder gelesen werden sollen:



# Thema: **Modbus RTU**

Bezug: Steuerung SK 5xxE über Modbus RTU per PC

Autor: Tim Köller

Datum: 17.08.2020

Funktion Code	Funktion Name	Beschreibung
01h	Read Coil	Lesezugriff auf alle Bus IN & OUT Bits
05h	Write Single Coils	Schreibzugriff auf einzelne Bus IN Bits
0Fh	Write Multiple Coils	Zeitgleicher Schreibzugriff auf alle Bus IN Bits
03h	Read Holding Register	Parameter Lesezugriff
06h	Write Single Register	Schreibzugriff auf eine einzelne Parameter (max. 16Bit)
10h	Write Multiple Register	Parameter Schreibzugriff auf 32Bit Parameter oder mehrere PZD Daten

**Abbildung 2: Übersicht der Funktion-Codes**

Diese Funktionscodes werden von dem Programm „ModLink Demo“ selbstständig berücksichtigt – je nachdem in welcher Rubrik man sich befindet und ob man den Button für das Lesen oder das Schreiben von Daten betätigt hat.

Die übertragenen Daten können mit den Beispielergrammen aus der BU 0050 verglichen werden:

## Beispiel2:

Es werden folgende 4 Prozessdaten ausgelesen, Zustandswort und Istwert 1 bis 3 (P051[-00] bis P051[-03]  
Zustandswort = 0x2B37 // IW1 = 0x09C4 // IW2 = 0x0203 // IW3 = 0x09C4.

Anfrage (Master → Slave)		Antwort (Slave → Master)	
Adresse	0x08	Adresse	0x08
Funktion Code	0x03	Funktion Code	0x03
Start Adresse High	0x0C	Anzahl der Bytes	0x08
Start Adresse Low	0xC0	Parameter 1 Wert High	0x2B
Anzahl der Parameter High	0x00	Parameter 1 Wert Low	0x37
Anzahl der Parameter Low	0x04	Parameter 2 Wert High	0x09
CRC High	0x47	Parameter 2 Wert Low	0xC4
CRC Low	0xFC	Parameter 3 Wert High	0x02
		Parameter 3 Wert Low	0x03
		Parameter 4 Wert High	0x09
		Parameter 4 Wert Low	0xC4
		CRC High	0x65
		CRC Low	0xD3

**Abbildung 3: Auslesen von Zustandswort und der Istwerte 1...3**

**Beispiel2:**

Es wird der Parameter P050[1] bis P050[3], also Sollwert 1 bis 3 beschrieben.

SW1 = 1000 // SW2 = 2000 // SW3 = 3000.

Anfrage (Master → Slave)		Antwort (Slave → Master)	
Adresse	0x08	Adresse	0x08
Funktion Code	0x10	Funktion Code	0x10
Start Adresse High	0x0C	Start Adresse High	0x0C
Start Adresse Low	0x81	Start Adresse Low	0x81
Anzahl der Parameter High	0x00	Anzahl der Parameter High	0x00
Anzahl der Parameter Low	0x03	Anzahl der Parameter Low	0x03
Anzahl der Bytes	0x06	CRC High	0xD3
Parameter 1 Wert High	0x03	CRC Low	0xE9
Parameter 1 Wert Low	0xE8		
Parameter 2 Wert High	0x07		
Parameter 2 Wert Low	0xD0		
Parameter 3 Wert High	0x0B		
Parameter 3 Wert Low	0xB8		
CRC High	0xF5		
CRC Low	0xDF		

**Abbildung 4: Schreiben der Sollwerte 1...3**

Um beliebige Parameter zu lesen bzw. zu beschreiben, muss auf Basis der Parameternummer und des Parametersatzes bzw. Parameterindex die Parameteradresse ermittelt werden:

Startadresse	
Bit 15 – Bit 6	Bit 5 – Bit 0
Parameternummer	Array Index

**Abbildung 5: Aufbau der Parameteradresse**

Fall 1) – Parameter ohne Parametersatzabhängigkeit und ohne Index (z.B. P 601)

$$Array - Index = 0$$

Fall 2) – Parameter mit Parametersatzabhängigkeit und ohne Index (z.B. P 102)

$$ArrayIndex = PSatz - 1$$

Fall 3) – Parameter ohne Parametersatzabhängigkeit mit Index (z.B. P 607)

$$ArrayIndex = PIndex - 1$$

## Thema: **Modbus RTU**

Bezug: Steuerung SK 5xxE über Modbus RTU per PC

Autor: Tim Köller

Datum: 17.08.2020



### Fall 4) – Parameter mit Parametersatzabhängigkeit

Für Parameter, die parametersatzabhängig sind und gleichzeitig auch Arrayparameter sind, gilt für den Array-Index:

$$ArrayIndex = [(PIndex - 1) * 4 + PSatz] - 1$$

	P-Satz 1	P-Satz 2	P-Satz 3	P-Satz 4
Array [-01]	0x00	0x01	0x02	0x03
Array [-02]	0x04	0x05	0x06	0x07
Array [-03]	0x08	0x09	0x0A	0x0B
...	...	...	...	...

#### Beispiel 1 Adresse von ‚P102‘ in Parametersatz 1:

- Parameternummer  $102_{\text{dez}} = 0x66 = 01100110_{\text{bin}}$
- Bitshift um 6 Bits nach links:  $01100110\_000000_{\text{bin}} = 0x1980$
- Parametersatz 1 => Array Index = 0
- Resultierende Gesamtadresse:  $0x1980 = 6528_{\text{dez}}$

#### Beispiel 2: Adresse von ‚P102‘ in Parametersatz 4:

- Parameternummer  $102_{\text{dez}} = 0x66 = 01100110_{\text{bin}}$
- Bitshift um 6 Bits nach links:  $01100110\_000000_{\text{bin}} = 0x1980$
- Parametersatz 4 => Array Index = 3
- Resultierende Gesamtadresse:  $0x1983 = 6531_{\text{dez}}$

#### Beispiel 3: Adresse von ‚P543‘ Index [1] in Parametersatz 1:

- Parameternummer  $543_{\text{dez}} = 0x21F = 1000011111_{\text{bin}}$
- Bitshift um 6 Bits nach links:  $1000011111\_000000_{\text{bin}} = 0x87C0$
- Parametersatz 1 => Array-Index = 0
- Resultierende Gesamtadresse:  $1000011111\_000000_{\text{bin}} = 0x87C0 = 34752_{\text{dez}}$

#### Beispiel 4: Adresse von ‚P543‘ Index [1] in Parametersatz 2:

- Parameternummer  $543_{\text{dez}} = 0x21F = 1000011111_{\text{bin}}$
- Bitshift um 6 Bits nach links:  $1000011111\_000000_{\text{bin}} = 0x87C0$
- Parametersatz 2 => Array-Index = 1
- Resultierende Gesamtadresse:  $1000011111\_00001_{\text{bin}} = 0x87C1 = 34753_{\text{dez}}$

#### Beispiel 5: Adresse von ‚P543‘ Index [1] in Parametersatz 4:

- Parameternummer  $543_{\text{dez}} = 0x21F = 1000011111_{\text{bin}}$
- Bitshift um 6 Bits nach links:  $1000011111\_000000_{\text{bin}} = 0x87C0$
- Parametersatz 4 => Array-Index = 3
- Resultierende Gesamtadresse:  $1000011111\_00011_{\text{bin}} = 0x87C3 = 34755_{\text{dez}}$

Beispiel 6: Adresse von ‚P543‘ Index [2] in Parametersatz 1:

- Parameternummer  $543_{\text{dez}} = 0x21F = 1000011111_{\text{bin}}$
- Bitshift um 6 Bits nach links:  $1000011111\_000000_{\text{bin}} = 0x87C0$
- Parametersatz 2 => Array-Index = 4
- Resultierende Gesamtadresse:  $1000011111\_000100_{\text{bin}} = 0x87C4 = 34756_{\text{dez}}$

Beispiel 7: Adresse von ‚P607‘ Index [1]:

- Parameternummer  $607_{\text{dez}} = 0x25F = 1001011111_{\text{bin}}$
- Bitshift um 6 Bits nach links:  $1001011111\_000000_{\text{bin}} = 0x97C0$
- Index 1 => Array-Index = 0
- Resultierende Gesamtadresse:  $1001011111\_000000_{\text{bin}} = 0x97C0 = 38848_{\text{dez}}$

Beispiel 8: Adresse von ‚P607‘ Index [2]:

- Parameternummer  $607_{\text{dez}} = 0x25F = 1001011111_{\text{bin}}$
- Bitshift um 6 Bits nach links:  $1001011111\_000000_{\text{bin}} = 0x97C0$
- Index 2 => Array-Index = 1
- Resultierende Gesamtadresse:  $1001011111\_000001_{\text{bin}} = 0x97C1 = 38850_{\text{dez}}$

Beispiel 9: Ansteuerung des Umrichters über das Steuerwort ‚P050‘ Index 1

- Parameternummer  $50_{\text{dez}} = 0x32 = 00110010_{\text{bin}}$
- Bitshift um 6 Bits nach links:  $00110010\_000000_{\text{bin}} = 0xC80$
- Index 1 => Array-Index = 0
- Resultierende Gesamtadresse:  $00110010\_000000_{\text{bin}} = 0xC80 = 3200_{\text{dez}}$
- Nutzdaten:
  - o Versetzen des Umrichters in ‚Einschaltbereit‘: Steuerwort = 0x047E
  - o Versetzen des Umrichters in ‚Eingeschaltet‘: Steuerwort = 0x047F
  - o Aufbau des Steuerworts siehe nächste Seite

Weitere Beispiele können dem Handbuch BU0050 ab Seite 60 entnommen werden:

[https://www.nord.com/cms/de/documentation/manuals/details\\_1139/detail\\_7436.jsp](https://www.nord.com/cms/de/documentation/manuals/details_1139/detail_7436.jsp)

# Thema: Modbus RTU

Bezug: Steuerung SK 5xxE über Modbus RTU per PC

Autor: Tim Köller

Datum: 17.08.2020

Der Aufbau des Steuerwortes ist folgendermaßen:

Im Auftragstelegramm wird im Bereich der Prozessdaten das Steuerwort (STW) als erstes Wort dem Frequenzumrichter übertragen. Ein Steuerwort „Einschaltbereit“ entspricht beispielsweise 047E<sub>(hex)</sub>. Als erster Befehl sollte generell ein „Einschaltbereit“ an den Umrichter übermittelt werden.

Bit	Wert	Bedeutung	Bemerkung
0	0	Nicht betriebsbereit	Rücklauf mit der Bremsrampe, bei f=0Hz Spannungsfreischaltung (AUS1)
	1	Betriebsbereit	Frequenzumrichter Betriebsbereit
1	0	Spannung sperren	Spannung sperren; Die Umrichter- Ausgangsspannung wird abgeschaltet ; der FU geht in Zustand Einschaltsperrung. (AUS2)
	1	Nicht Spannung sperren	Betriebsbedingung - AUS 2 ist aufgehoben
2	0	Schnellhalt aktiv	Schnellhalt mit programmierter Schnellhaltzeit; bei f=0Hz Spannungsfreischaltung; Der FU geht in Zustand Einschaltsperrung (AUS3)
	1	Schnellhalt nicht aktiv	Betriebsbedingung - AUS 3 ist aufgehoben
3	0	Betrieb sperren	Spannung sperren; Die Umrichter- Ausgangsspannung wird abgeschaltet; Der FU geht in Zustand Einschaltbereit
	1	Betrieb freigeben	Freigabe der Ausgangsspannung; Hochlauf auf anliegenden Sollwert
4	0	Impulse nicht freigeben	Hochlaufgeber wird auf Null gesetzt; bei f=0Hz keine Spannungsfreischaltung; FU bleibt in Zustand Betrieb freigeben
	1	Impulse freigeben	Hochlaufgeber ist freigeben
5	0	Rampe nicht freigeben	Einfrieren des aktuellen vom Hochlaufgeber vorgegebenen Sollwertes (Frequenz halten).
	1	Rampe freigeben	Sollwert am Hochlaufgeber freigeben.
6	0	Sollwert nicht freigeben	Angewählter Sollwert wird am Hochlaufgeber auf Null gesetzt.
	1	Sollwert freigeben	Angewählter Sollwert wird am Hochlaufgeber aktiviert.
7	0	Keine Quittierung	Mit Wechsel von 0 auf 1 werden nicht mehr aktive Störungen quittiert.
	1	Fehler quittieren (0→1)	Hinweis: Wenn ein Digitaleingang auf die Funktion „Stoer.Quit.“ programmiert ist, darf dieses Bit über den Bus nicht dauerhaft auf 1 gesetzt sein (Flankenwertung wird sonst verhindert).
8	0		
	1	Funktion 480, 11 starten	Bus Bit 8 vom Steuerwort ist gesetzt. Nur bei SK 2xxE und SK 5xxE. Näheres zur Funktion siehe unter Parameter P480.
9	0		
	1	Funktion 480, 12 starten	Bus Bit 9 vom Steuerwort ist gesetzt. Nur bei SK 2xxE und SK 5xxE. Näheres zur Funktion siehe unter Parameter P480.
10	0	Steuerdaten ungültig	Die gesendeten Prozessdaten sind ungültig.
	1	Steuerdaten gültig	Vom Master werden gültige Prozessdaten übertragen. Hinweis: Auch wenn nur Sollwerte über den Bus übertragen werden (Einstellung: Schnittstelle), dann muss dieses Bit gesetzt sein, damit der übertragene Sollwert gültig wird.
11	0		
	1	Drehrichtung rechts ein	Drehrichtung rechts (vorrangig) ein.*
12	0		
	1	Drehrichtung links ein	Drehrichtung links ein.*
13	0/1	reserviert	Reserviert
14	0/1	Parametersatz Bit 0 ein	00 = Parametersatz 1
15	0/1	Parametersatz Bit 1 ein	10 = Parametersatz 3
			11 = Parametersatz 4

\* wenn Bit 12=0, dann gilt „Drehrichtung rechts ein“

Abbildung 6: Aufbau Steuerwort

# Thema: **Modbus RTU**

Bezug: Steuerung SK 5xxE über Modbus RTU per PC

Autor: Tim Köller

Datum: 17.08.2020



Das Zustandswort des Umrichters ist folgendermaßen aufgebaut:

Im Umrichter- Antwort- Telegramm wird im Bereich der Prozessdaten das Zustandswort (ZSW) als erstes Wort dem Frequenzumrichter übertragen. Die Bedeutungen der einzelnen Bits weichen bei einigen Gerätetypen ab.

Bit	Wert	Bedeutung	Bemerkung
0	0	Nicht Einschaltbereit	
	1	Einschaltbereit	Initialisierung beendet, Laderelais ein, Ausgangsspannung gesperrt
1	0	Nicht betriebsbereit	Ursachen: Ein- Befehl liegt nicht an, Störung liegt an , AUS2 oder AUS 3 liegen an, Zustand Einschaltsperrung liegt an
	1	Betriebsbereit	EIN- Befehl liegt an, es liegt keine Störung an. Der Umrichter kann mit dem Befehl BETRIEB FREIGEBEN starten
2	0	Betrieb nicht freigegeben	
	1	Betrieb freigegeben	Freigabe der Ausgangsspannung; Hochlauf auf anliegenden Sollwert
3	0	Keine Störung	
	1	Störung	Antrieb gestört und dadurch außer Betrieb; geht nach erfolgreicher Quittierung in Zustand Einschaltsperrung
4	0	Spannung nicht freigegeben	AUS 2-Befehl Spannung sperren liegt an
	1	Spannung freigegeben	
5	0	Schnellhalt	AUS3-Befehl Schnellhalt liegt an
	1	kein Schnellhalt	
6	0	Keine Einschaltsperrung	
	1	Einschaltsperrung	Geht durch AUS1-Befehl Freigabe in Zustand Einschaltbereit
7	0	Keine Warnung	
	1	Warnung aktiv	Antrieb weiter in Betrieb, keine Quittierung nötig
8	0	Sollwert nicht erreicht	Istwert entspricht nicht dem Sollwert (bei posicon: Sollposition nicht erreicht)
	1	Sollwert erreicht	Istwert entspricht dem gewünschten Sollwert (Sollwert erreicht) (bei posicon: Sollposition erreicht)
9	0	Bus-Steuerung nicht aktiv	Führung lokal am Gerät aktiv
	1	Bus-Steuerung aktiv	Der Master wird aufgefordert, die Führung zu übernehmen.
10	0		
	1	SK 5xxE: Funktion 481.9 starten SK 300E / 7x0E: Vergleichswert MFR 1 erreicht	Bus Bit 10 vom Statuswort ist gesetzt. Näheres zur Funktion siehe unter Parameter P481. Programmierte Funktion des MFR 1 erfüllt bzw. Istwert $\geq$ programmierter Vergleichswert
11	0		
	1	Drehrichtung rechts ein	Umrichter- Ausgangsspannung hat rechtes Drehfeld
12	0		
	1	Drehrichtung links ein	Umrichter- Ausgangsspannung hat linkes Drehfeld
13	0		
	1	SK 5xxE: Funktion 481.10 starten SK 7x5E: Vergleichswert MFR 4 erreicht	Bus Bit 13 vom Statuswort ist gesetzt. Näheres zur Funktion siehe unter Parameter P481. Nur mit posicon Erweiterung: Zustand MFR 4 = 1
14	0/1	Parametersatz Bit 0 an	00 = Parametersatz 1 01 = Parametersatz 2 10 = Parametersatz 3 11 = Parametersatz 4
15	0/1	Parametersatz Bit 1 an	

Abbildung 7: Aufbau Zustandswort