

SAS 341 – Modbus Treiber

Serielle Anschaltungsbaugruppe für S7-300
mit Modbus RTU Master/Slave Treiber

800-341-MOD01

Handbuch

Ausgabe 1 - 02.08.2011 ab HW1 & FW1.16



Handbuch Bestellnummer: 900-341-MOD01/de

Alle Rechte, auch die der Übersetzung, des Nachdruckes und der Vervielfältigung dieses Handbuches, oder Teilen daraus, vorbehalten. Kein Teil des Handbuches darf ohne schriftliche Genehmigung der Systeme Helmholtz GmbH in irgendeiner Form (Fotokopie, Mikrofilm oder ein anderes Verfahren), auch nicht für Zwecke der Unterrichtsgestaltung, oder unter Verwendung elektronischer Systeme reproduziert, verarbeitet, vervielfältigt oder verbreitet werden. Alle Rechte für den Fall der Patenterteilung oder Gebrauchsmustereintragung vorbehalten.

Copyright © 2011 by

Systeme Helmholtz GmbH

Hannberger Weg 2, 91091 Großenseebach

Hinweis:

Der Inhalt dieses Handbuches ist von uns auf die Übereinstimmung mit der beschriebenen Hard- und Software überprüft worden. Da dennoch Abweichungen nicht ausgeschlossen sind, können wir für die vollständige Übereinstimmung keine Gewährleistung übernehmen. Die Angaben in diesem Handbuch werden jedoch regelmäßig aktualisiert. Bitte beachten sie beim Einsatz der erworbenen Produkte jeweils die aktuellste Version des Handbuchs, die im Internet unter www.helmholtz.de einsehbar ist und auch heruntergeladen werden kann.

Unsere Kunden sind uns wichtig. Wir freuen uns über Verbesserungsvorschläge und Anregungen.

Änderungen in diesem Dokument:

Stand	Datum	Änderung
1	2.8.2011	1. Version

Inhaltsverzeichnis

1	Sicherheitshinweise	7
1.1	Allgemein	7
1.2	Zugangsbeschränkung	8
1.3	Benutzerhinweise	8
1.4	Bestimmungsgemäßer Gebrauch	8
1.5	Bestimmungswidrigen Gebrauch vermeiden!	8
2	Installation und Montage	9
2.1	Vertikaler und horizontaler Aufbau	9
2.2	Mindestabstand	10
2.3	Montage der Baugruppe auf die Profilschiene	10
3	Systemübersicht	12
3.1	Einsatzmöglichkeiten	12
3.2	Modbus-Treiber	12
3.3	Einsetzen der Treiber MMC	12
3.4	Schnittstellen	13
3.4.1	SubD-Stecker RS232 (700-341-1AH02 / -2AH02)	13
3.4.2	SubD-Buchse RS422/RS485 (700-341-1CH02 / -2CH02)	14
3.5	LED-Anzeigen	15
4	Modbus Protokoll	16
4.1	Einführung	16
4.2	Telegrammaufbau Anforderung	16
4.3	Telegrammaufbau Antwort des Slaves	17
4.4	Erläuterung der Funktionscodes	18
4.4.1	Funktionscode 01 – Read coil status	18
4.4.2	Funktionscode 02 – Read input status	19
4.4.3	Funktionscode 03 – Read output registers	20
4.4.4	Funktionscode 04 – Read input registers	20
4.4.5	Funktionscode 05 – Force single coil	21
4.4.6	Funktionscode 06 – Preset single register	21
4.4.7	Funktionscode 07 – Read exception status	22
4.4.8	Funktionscode 08 – Loop back diagnostic test	22

4.4.9	Funktionscode 11 – Fetch communications event counter	23
4.4.10	Funktionscode 12 – Fetch communications event log	24
4.4.11	Funktionscode 15 – Force Multiple Coils	25
4.4.12	Funktionscode 16 – Preset multiple registers	26
5	Konfigurieren der Baugruppe	27
6	Parametrierung der Baugruppe	29
6.1	Parametriersoftware installieren	29
6.2	Modbus Master	31
6.3	Modbus Slave	33
7	Programmierung in der SPS	37
7.1	Übersicht	37
7.2	Peripheriedaten in der SPS	38
7.2.1	Byte 0: Baugruppenstatus	38
7.2.2	Byte 1: Statussignale Kanal 1	39
7.2.3	Byte 2: FIFO-Status Bits Kanal 1	39
7.2.4	Byte 3: Fehler-Bits Kanal 1	39
7.2.5	Byte 4: Aktives Protokoll Kanal 1	39
7.3	Hantierungsbausteine Modbus Master	40
7.3.1	FB 4 MODB_MAST_SND	40
7.3.2	FB 5 MODB_MAST_RCV	42
7.3.3	Fehlernummern Modbus Master Treiber	43
7.4	Hantierungsbaustein Modbus Slave	44
7.4.1	FB 80 MODBUS_SLAVE	44
7.4.2	Fehlernummern Modbus Slave	45
8	Anhang	47
8.1	Technische Daten	47
8.2	Steckerbelegung	48
8.2.1	SubD-Stecker RS232 (700-341-1AH02)	48
8.2.2	SubD-Buchse RS422/RS485 (700-341-1CH02)	48

1 Sicherheitshinweise

Beachten Sie die aufgeführten Sicherheitshinweise zu Ihrer eigenen Sicherheit und der Sicherheit Anderer. Die Sicherheitshinweise zeigen Ihnen mögliche Gefahren auf und geben Ihnen Hinweise, wie Sie Gefahrensituationen vermeiden können.

Im vorliegenden Handbuch werden folgende Piktogramme verwendet:



Achtung, macht auf Gefahren und Fehlerquellen aufmerksam



gibt einen Hinweis



Gefahr allgemein oder spezifisch



*Gefahr eines **Stromschlages***

1.1 Allgemein

Die SAS 341 Baugruppe wird nur als Bestandteil eines Gesamtsystems eingesetzt.



Der Betreiber einer Maschinenanlage ist für die Einhaltung der für den speziellen Einsatzfall geltenden Sicherheits- und Unfallverhütungsvorschriften verantwortlich.



Bei der Projektierung sind die einsatzspezifischen Sicherheits- und Unfallverhütungsvorschriften zu beachten.



Not-Aus-Einrichtungen gemäß EN 60204 / IEC 204 müssen in allen Betriebsarten der Maschinenanlage wirksam bleiben. Es darf zu keinem undefinierten Wideranlauf der Anlage kommen.



In der Maschinenanlage auftretende Fehler, die Material- oder Personenschäden verursachen können, müssen durch zusätzliche externe Einrichtungen abgefangen werden. Diese Einrichtungen müssen auch im Fehlerfall einen sicheren Betriebszustand gewährleisten. Solche Einrichtungen sind z.B. elektromechanische Sicherheitsschalter, mechanische Verriegelungen usw. (siehe EN 954-1, Risikoabschätzung).



Sicherheitsrelevante Funktionen niemals über das Bedien-terminal ausführen oder einleiten.



*Zutritt zu den
Baugruppen nur für
berechtigte Personen!*

1.2 Zugangsbeschränkung

Die Baugruppen sind offene Betriebsmittel und dürfen nur in elektrischen Betriebsräumen, Schränken oder Gehäusen installiert werden. Der Zugang zu den elektrischen Betriebsräumen, Schränken oder Gehäusen darf nur über Werkzeug oder Schlüssel möglich sein und nur unterwiesenem oder zugelassenem Personal gestattet werden. Siehe auch Kapitel 2.

1.3 Benutzerhinweise

Dieses Handbuch richtet sich an Projektoren und Monteure der SAS 341 Baugruppe.

Es soll dem Projektoren als Programmierhandbuch und Nachschlagewerk dienen. Dem Monteur sollen alle zur Montage notwendigen Daten bereitgestellt werden.

Die SAS 341 Baugruppe ist ausschließlich zum Gebrauch in einem S7-300 Automatisierungsgerät der Firma Siemens. Aus diesem Grund sind von Projektoren, Anwender und Monteur die für den jeweiligen Einsatzfall geltenden Normen, Sicherheits- und Unfallverhütungsvorschriften unbedingt zu beachten. Der Betreiber des Automatisierungssystems ist für die Einhaltung dieser Vorschriften verantwortlich.

1.4 Bestimmungsgemäßer Gebrauch

Die SAS 341 Baugruppe darf nur, wie im Handbuch beschrieben, als Kommunikationssystem verwendet werden.

1.5 Bestimmungswidrigen Gebrauch vermeiden!

Sicherheitsrelevante Funktionen dürfen nicht mit über die SAS 341 Baugruppe allein gesteuert werden.

2 Installation und Montage

Die Installation der SAS 341 Baugruppe muss nach VDE 0100 IEC 364 erfolgen. Da es sich um „OPEN Type“ Baugruppen handelt, müssen sie in einen (Schalt-) Schrank eingebaut werden.
Umgebungstemperatur: 0 °C – 60 °C.



Bevor Installationsarbeiten durchgeführt werden, alle Systemkomponenten spannungsfrei schalten.



Gefahr eines Stromschlages!



Bei der Montage sind die einsatzspezifischen Sicherheits- und Unfallverhütungsvorschriften zu beachten.

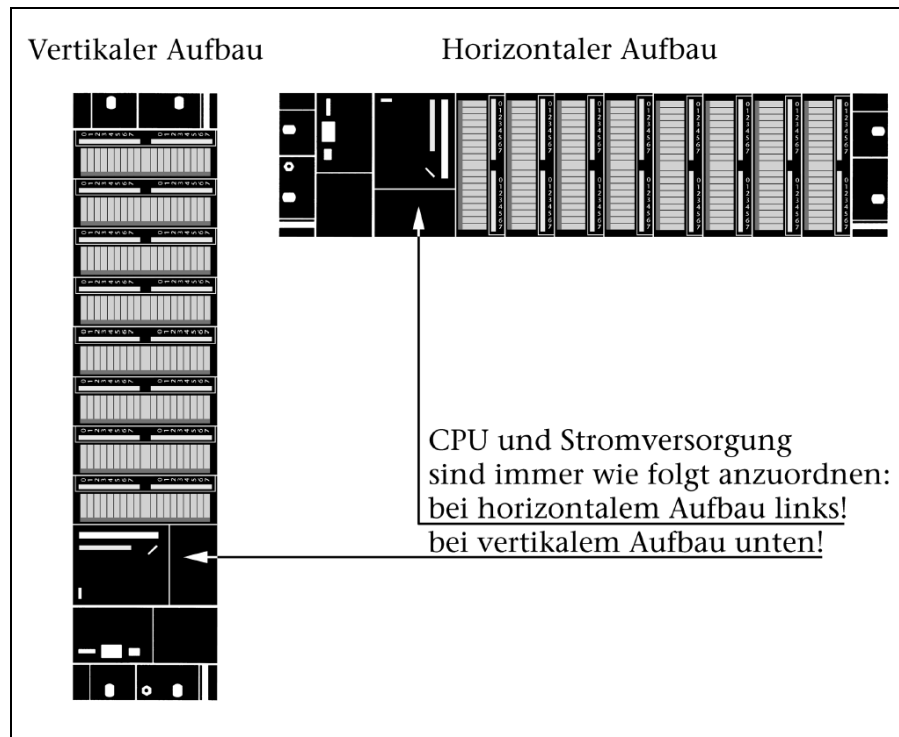
2.1 Vertikaler und horizontaler Aufbau

Die Baugruppen können sowohl vertikal als auch horizontal aufgebaut werden.

Zulässige Umgebungstemperatur:

bei vertikalem Aufbau: von 0 bis 40 °C

bei horizontalem Aufbau: von 0 bis 60 °C



2.2 Mindestabstand

Durch die Einhaltung von Mindestabständen

ist eine Abkühlung der SAS 341 Baugruppen gewährleistet

ist genügend Raum zum Ein- und Aushängen der Baugruppen vorhanden

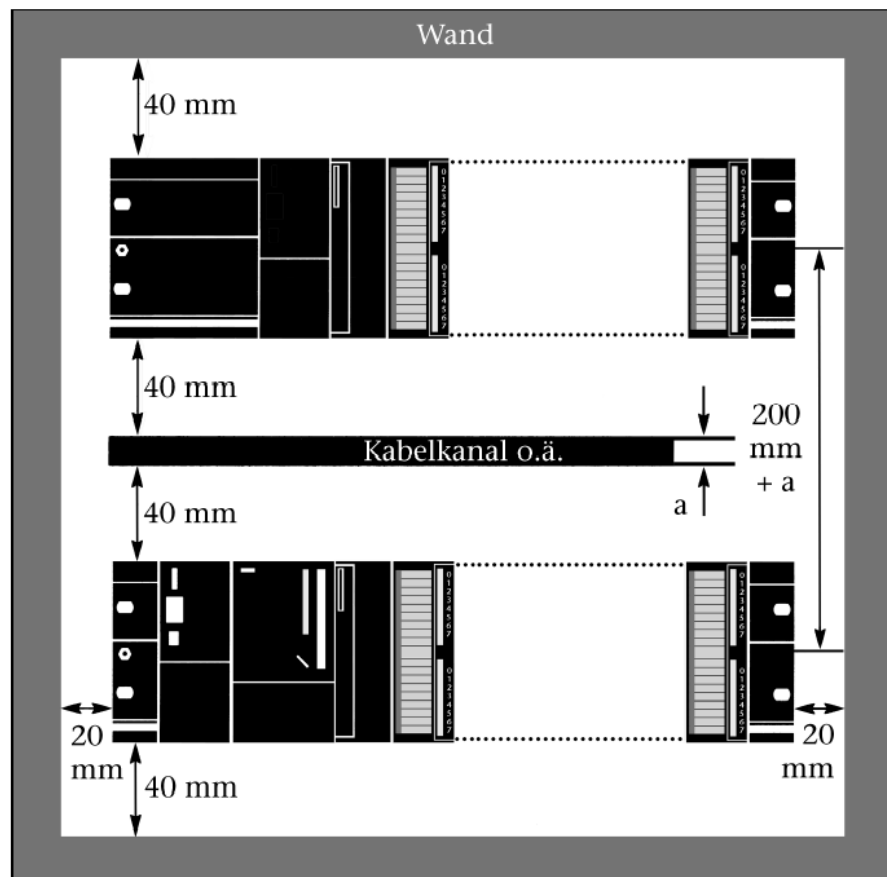
ist genügend Raum zum Verlegen von Leitungen vorhanden

erhöht sich die Einbauhöhe des Baugruppenträgers auf 185 mm, wobei trotzdem das Abstandsmaß von 40 mm eingehalten werden muss.

Im folgenden Bild sind für S7-300 Aufbauten auf mehreren Baugruppenträgern die Mindestabstandsmaße zwischen den jeweiligen Baugruppenträgern, sowie zu benachbarten Schrankwänden, Betriebsmitteln, Kabelkanälen etc. angegeben.



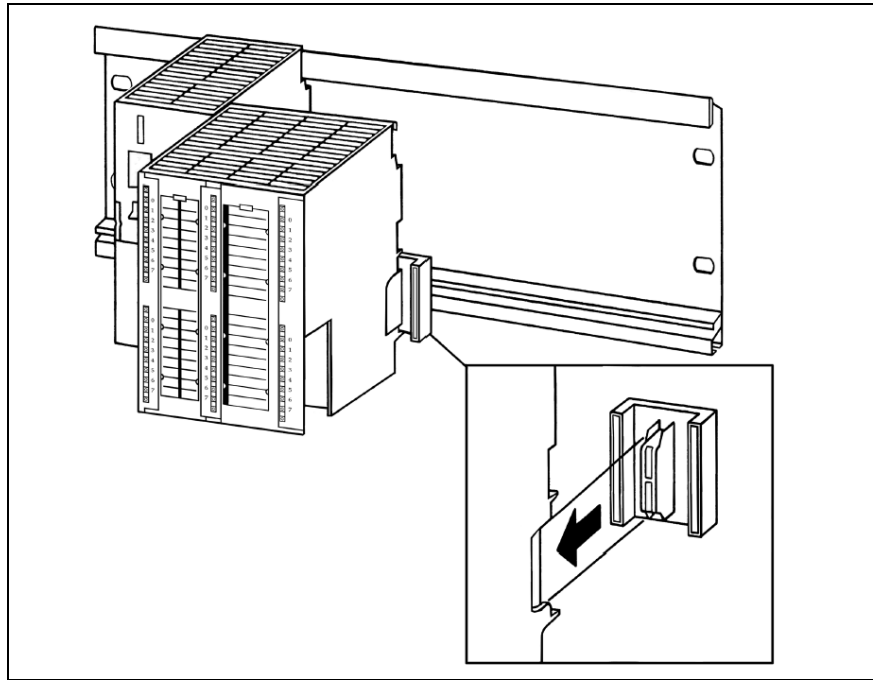
Nichteinhaltung der Mindestabstände kann die Baugruppe bei hohen Umgebungstemperaturen zerstören!



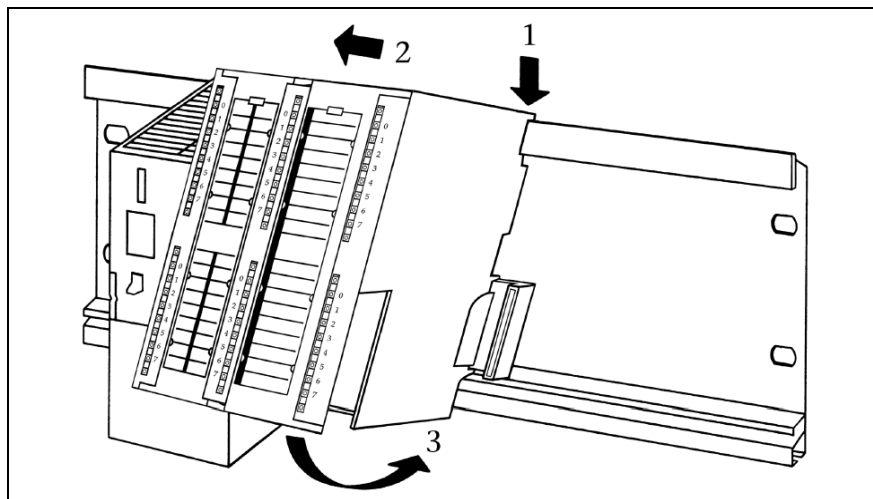
2.3 Montage der Baugruppe auf die Profilschiene

Ein Busverbinder liegt jeder Signalbaugruppe bei, nicht aber der CPU. Beim Aufstecken der Busverbinder immer bei der CPU beginnen.

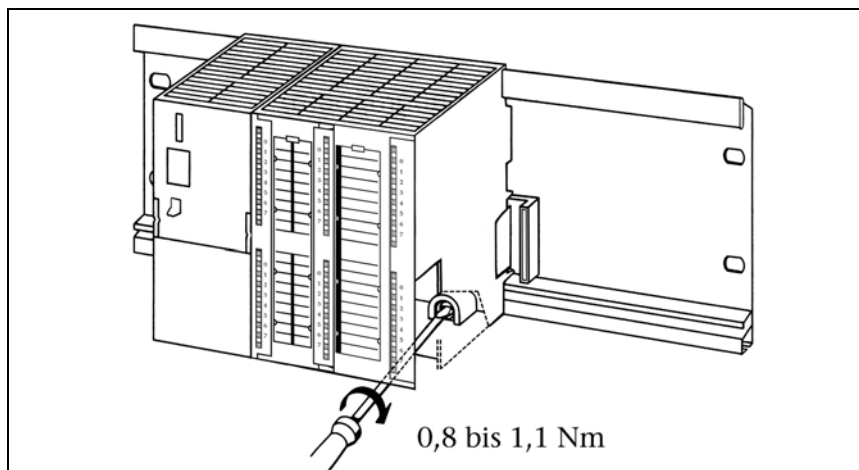
Den Busverbinder von der letzten Baugruppe nehmen und in die CPU stecken. Auf die letzte Baugruppe der Zeile keinen Busverbinder stecken.



Die Baugruppen einhängen (1), bis an die linke Baugruppe heranschieben (2) und nach unten schwenken (3).



Die Baugruppen mit einem Drehmoment von 0,8 bis 1,1 Nm fest schrauben.



3 Systemübersicht

3.1 Einsatzmöglichkeiten

Die SAS 341 ist eine serielle Kommunikationsbaugruppe zum Einsatz in Simatic S7-300 Systemen. Die SAS 341 ermöglicht die Anbindung von seriellen Geräten wie z.B. Barcodescanner, Bedienterminals, serielle Drucker, PCs, oder SPS anderer Hersteller, an die SPS und unterstützt die Protokolle ASCII, 3964R und RK512.

Die Anbindung der seriellen Geräte kann über RS232 (V.24), TTY (20 mA) oder RS422/RS485 erfolgen. Die 9-polige Sub-D Anschluss (15pol. bei RS422/RS485) mit Standardbelegung ist für den Anschluss der Partnergeräte vorgesehen.

Die Parametrierung der Baugruppe wird im Hardwarekonfigurator der SPS vorgenommen. Mit den mitgelieferten Hantierungsbausteinen ist eine einfache und flexible Integration in die SPS möglich.

3.2 Modbus-Treiber

Das Treiber-Addon „Modbus Master/Slave“ ermöglicht die Kommunikation mit Modbus RTU fähigen Geräten. Die SAS 341 kann mit diesem Treiber entweder als Modbus RTU Master oder als Modbus RTU Slave arbeiten.

Der Treiber ist für die SAS 341-1 mit RS232-Schnittstelle (700-341-1AH02) oder für die SAS 341-1 mit RS485-Schnittstelle (700-341-1CH02) verwendbar. Mit der RS232-Schnittstelle ist ein Punkt-zu-Punkt Betrieb möglich, mit der RS485-Schnittstelle können bis zu 32 Teilnehmer im 2-Draht Halbduplexverfahren angesprochen werden.

Der Treiber arbeitet nicht in der SAS 340 oder in der SAS 341 mit 2 Schnittstellen. Für die SAS 341 wird mindestens die Firmware-Version 1.16 benötigt.

Bei der Kommunikation mit Fremdsystemen werden die Modbus RTU Funktionscode 01, 02, 03, 04, 05, 06, 07, 08, 11, 12, 15 und 16 unterstützt.

Der Datenaustausch mit der S7-CPU erfolgt blockweise über die mitgelieferten Funktionsbausteine.

3.3 Einsetzen der Treiber MMC

Die Treiber MMC (800-341-MOD01) sollte im spannungslosen Zustand in den MMC-Schacht der SAS 341 eingesteckt werden. Nach dem Netz-Ein ist der Modbus-Treiber sofort betriebsbereit.

Die Treiber MMC muss dauerhaft in der SAS 341 Baugruppe verbleiben.

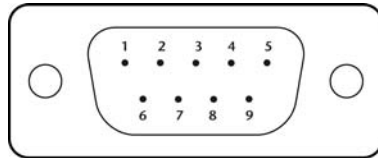
3.4 Schnittstellen

Hinter der Frontklappe der SAS 341 Baugruppe befindet sich der Sub-D Anschluss, welcher als 9-poliger Stecker bei RS232 und 15-polige Buchse bei RS485 ausgelegt ist. Die Pinbelegungen sind kompatibel zu den Siemens CP 341 Baugruppen.

Bei der SAS 341 mit RS232-Schnittstelle ist zusätzlich noch ein USB-Anschluss vorhanden, die bei der Verwendung der Karte als Modbus Master/Slave keine Funktion hat.

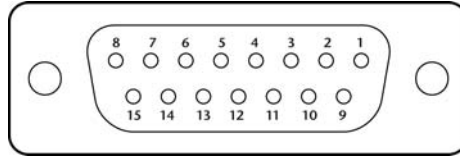
Ein Mini-USB-Anschluss ist für Servicedienste vorgesehen (Update oder Diagnose).

3.4.1 SubD-Stecker RS232 (700-341-1AH02 / -2AH02)



Pin	Bezeichnung		Richtung	Beschreibung
1	DCD	Data Carrier Detect	Eingang	Trägersignal (Modem)
2	RxD	Receive Data	Eingang	Empfangsleitung
3	TxD	Transmit Data	Ausgang	Sendeleitung
4	DTR	Data Terminal Ready	Ausgang	ON = SAS ist betriebsbereit
5	GND	Signal Ground	-	Nullbezugspunkt
6	DSR	Data Set Ready	Eingang	Kommunikationspartner betriebsbereit?
7	RTS	Request to send	Ausgang	ON = SAS sendebereit, OFF = nichts zu senden
8	CTS	Clear to send	Eingang	Kommunikationspartner empfangsbereit?
9	RI	Ring indicator	Eingang	Klingelzeichen (Modem)

3.4.2 SubD-Buchse RS422/RS485 (700-341-1CH02 / -2CH02)



Pin	Bezeichnung	Richtung	Beschreibung
1	-	-	
2	T (A)	Ausgang	Sendedaten (Vierdraht-Betrieb)
3	-	-	
4	R (A) / T (A)	Eingang / Ein-/Ausgang	Empfangsdaten (Vierdraht-Betrieb) Empfangs-/Sendedaten (Zweidraht-Betrieb)
5	-	-	
6	-	-	
7	-	-	
8	GND	-	
9	T (B)	Ausgang	Sendedaten (Vierdraht-Betrieb)
10	-	-	
11	R (B) / T (B)	Eingang / Ein-/Ausgang	Empfangsdaten (Vierdraht-Betrieb) Empfangs-/Sendedaten (Zweidraht-Betrieb)
12	-	-	
13	-	-	
14	-	-	
15	-	-	

3.5 LED-Anzeigen

Die LEDs an der Vorderseite der Baugruppe informieren über den Betriebszustand der SAS 341.

LED „SF“ (Orange):

Systemfehler: zeigt eine fehlerhafte Parametrierung an. Die Anzeige kann auch durch eine nicht korrekt gesteckte Treiber MMC ausgelöst werden.

LED „BF“ (Rot):

Diese LED zeigt einen Fehler an der seriellen Schnittstelle an (z.B. Parity, Framing, Overflow).

LED „RX“ (Grün):

Empfang Aktiv: Zeigt den Empfang eines Zeichens an der serielle Schnittstelle an.

LED „TX“ (Orange):

Senden Aktiv: Zeigt das Senden eines Zeichens an der seriellen Schnittstelle an.

LED „CPU“ (Orange):

Datenübertragung zur SPS aktiv: Zeigt die Übertragung von Daten oder Kommandos am Rückwandbus (zwischen S7-CPU und Baugruppe) an.

LED „ON“ (Grün):

Zeigt an, dass die Baugruppe korrekt mit Spannung versorgt ist und das Betriebssystem läuft.

Ist die SPS in Stop, so blinkt die LED.



4 Modbus Protokoll

4.1 Einführung

Das Modbus-Protokoll ist ein Master/Slave-Protokoll, alleinig der Master hat die Kommunikations-Initiative. Die Slaves haben eindeutige Adressen (1-255). Der Master kann sowohl Daten mit einem bestimmten Slave austauschen, als auch einen Broadcast an alle Slaves senden.

Mit dem Modbus-Protokoll ist sowohl ein Bit-orientierter als auch ein Register-orientierter (16 Bit) Datenaustausch möglich. Der Datenaustausch wird über Funktionscodes gesteuert.

4.2 Telegrammaufbau Anforderung

Adresse	Functioncode	Data	CRC-Check
Byte	Byte	n Bytes	2 Byte

Slaveadresse: 1-255 spricht einen bestimmten Slave am Bus an. Mit der Adresse 0 wird ein Broadcast gesendet. Broadcasts sind nur für die Funktionscodes 05, 06, 15 und 16 erlaubt. Auf Broadcasts senden die Slaves kein Antwort-Telegramm.

Funktionscode: Mit dem Funktionscode wird die Bedeutung des Telegramms definiert, damit ist auch der weitere Aufbau des Telegramms festgelegt.

Funktionscode	Funktion gemäß Modbus	Funktion in der SPS (beim Slave)
01	Read coil status	1...2040 Bits lesen M, A, T, Z
02	Read input status	1...2040 Bits lesen M, E
03	Read output registers	1...127 Worte lesen aus DB
04	Read input registers	1...127 Worte lesen aus DB
05	Force single coil	1 Bit setzen M, A
06	Preset Single Register	1 Wort schreiben in DB
07	Read Exception Status	Lesen der Ereignis-Bits
08	Loopback Test	Empfang spiegeln
11	Fetch communication event counter	<i>Wird im S7-Slave nicht unterstützt</i>
12	Fetch communication event log	<i>Wird im S7-Slave nicht unterstützt</i>
15	Force multiple coils	1...2040 Bits schreiben M,A
16	Preset multiple registers	1...127 Worte schreiben in DB

Data: Abhängig vom Funktionscode liegen ab dem Byte 2 die spezifischen Informationen und Nutzerdaten.

CRC-Check: Jedes Telegramm wird zur Absicherung auf Übertragungsfehlern mit einer 2 Byte CRC-16-Prüfsumme abgeschlossen.

Telegrammendeerkennung: Das Telegramm ist beendet, wenn die 3,5 fache Zeichenverzugszeit abgelaufen ist.

4.3 Telegrammaufbau Antwort des Slaves

Adresse	Functioncode	Data	CRC-Check
Byte	Byte	n Bytes	2 Byte

Die Adresse und der Funktionscode werden gespiegelt zurück-gesendet, der Inhalt der Daten hängt vom Funktionscode ab.

Stellt der Slave einen Fehler im Anforderungstelegramm fest, so antwortet er mit einem Exception-Response:

Adresse	Functioncode	Exception code	CRC-Check
Byte	Byte + 80 _{hex}	1 Byte	2 Byte

Exception code	Bedeutung gemäß Modbus	Ursache
1	Illegal function	Funktionscode wird nicht unterstützt
2	Illegal data address	Zieladresse nicht zugelassen
3	Illegal data value	Datenwert nicht erlaubt
4	Failure in associated device	Fehler im Slave, Slave noch nicht bereit
5	Acknowledge	Funktion wird ausgeführt
6	Busy, rejected message	Slave ist nicht empfangsbereit
7	Negative acknowledgement	Funktion nicht ausführbar

Antwortet ein Slave mit einem Exception Telegramm, so wird dieses am FB 5 – MODB_MAST_RCV (siehe Kap. 7.3.2) als Fehler-code 0xE6x (x= Exceptioncode) angezeigt.

4.4 Erläuterung der Funktionscodes

4.4.1 Funktionscode 01 – Read coil status

Diese Funktion ermöglicht das Lesen einzelner Bits vom Slave. Ist der Slave eine S7, so können abhängig von der Bit-Startadresse Merker, Ausgänge, Zeiten oder Zähler gelesen werden.

Anforderungstelegramm Function 01:

Byte	Name	Typ	Beispiel	Bemerkung
0	Adresse	Byte	B#16#09	
1	Funktionscode	Byte	B#16#01	
2+3	Bit-Startadresse	Word	W#16#0020	wird beim Senden nicht überprüft
4+5	Bit-Count	Int	24	1-2040

Antworttelegramm vom Slave:

Byte	Name	Typ	Beispiel	Bedeutung
0	Adresse	Byte	B#16#09	
1	Funktionscode	Byte	B#16#01	
2	Bytezähler	Byte	B#16#03	
3	Datenbyte 1	Byte	B#16#12	
4	Datenbyte 2	Byte	B#16#34	
5	Datenbyte 3	Byte	B#16#56	

Daten im RCV-Zielbereich der SPS:

Byte	Name	Typ	Beispiel	Bedeutung
+0	Datenbyte 1	Byte	B#16#34	
+1	Datenbyte 2	Byte	B#16#12	
+2	Datenbyte 3	Byte	B#16#00	Füllbyte
+3	Datenbyte 4	Byte	B#16#56	



Die Daten werden wortweise in den RCV-Bereich eingetragen und ein Byteswap durchgeführt. Bei ungeraden Bytezahlen wird eine 00h eingefügt.

4.4.2 Funktionscode 02 – Read input status

Diese Funktion ermöglicht das Lesen einzelner Bits vom Slave. Ist der Slave eine S7, so können abhängig von der Bit-Startadresse Merker oder Eingänge gelesen werden.

Anforderungstelegramm Function 02:

Byte	Name	Typ	Beispiel	Bemerkung
0	Adresse	Byte	B#16#09	
1	Funktionscode	Byte	B#16#02	
2+3	Bit-Startadresse	Word	W#16#0120	wird beim Senden nicht überprüft
4+5	Bit-Count	Int	16	1-2040

Antworttelegramm vom Slave:

Byte	Name	Typ	Beispiel	Bedeutung
0	Adresse	Byte	B#16#09	
1	Funktionscode	Byte	B#16#02	
2	Bytezähler	Byte	B#16#03	
3	Datenbyte 1	Byte	B#16#12	
4	Datenbyte 2	Byte	B#16#34	
5	Datenbyte 3	Byte	B#16#56	



Die Daten werden wortweise in den RCV-Bereich eingetragen und ein Byteswap durchgeführt. Bei ungeraden Bytezahlen wird eine 00h eingefügt.

Daten im RCV-Zielbereich der SPS:

Byte	Name	Typ	Beispiel	Bedeutung
+0	Datenbyte 1	Byte	B#16#34	
+1	Datenbyte 2	Byte	B#16#12	
+2	Datenbyte 3	Byte	B#16#00	Füllbyte
+3	Datenbyte 4	Byte	B#16#56	

4.4.3 Funktionscode 03 – Read output registers

Diese Funktion ermöglicht das Lesen einzelner Register vom Slave. Ist der Slave eine S7, so können Worte aus einem DB gelesen werden.

Anforderungstelegramm Function 03:

Byte	Name	Typ	Beispiel	Bemerkung
0	Adresse	Byte	B#16#09	
1	Funktionscode	Byte	B#16#03	
2+3	Register-Startadresse	Word	W#16#0020	wird beim Senden nicht überprüft
4+5	Register-Anzahl	Int	2	1-127

Antworttelegramm vom Slave:

Byte	Name	Typ	Beispiel	Bedeutung
0	Adresse	Byte	B#16#09	
1	Funktionscode	Byte	B#16#03	
2	Bytezähler	Byte	B#16#04	
3	Register 1 High	Byte	B#16#12	
4	Register 1 Low	Byte	B#16#34	
5	Register 2 High	Byte	B#16#56	
6	Register 2 Low	Byte	B#16#78	

Daten im RCV-Zielbereich der SPS:

Byte	Name	Typ	Beispiel	Bedeutung
+0	Register 1	Word	W#16#1234	
+2	Register 2	Word	W#16#5678	

4.4.4 Funktionscode 04 – Read input registers

Diese Funktion ermöglicht das Lesen einzelner Register vom Slave. Ist der Slave eine S7, so können Worte aus einem DB gelesen werden.

Anforderungstelegramm Function 04:

Byte	Name	Typ	Beispiel	Bemerkung
0	Adresse	Byte	B#16#09	
1	Funktionscode	Byte	B#16#04	
2+3	Register-Startadresse	Word	W#16#0020	wird beim Senden nicht überprüft
4+5	Register-Anzahl	Int	2	1-127

Antworttelegramm vom Slave:

Byte	Name	Typ	Beispiel	Bedeutung
0	Adresse	Byte	B#16#09	
1	Funktionscode	Byte	B#16#04	
2	Bytezähler	Byte	B#16#04	
3	Register 1 High	Byte	B#16#AB	
4	Register 1 Low	Byte	B#16#CD	
5	Register 2 High	Byte	B#16#01	
6	Register 2 Low	Byte	B#16#23	

Daten im RCV-Zielbereich der SPS:

Byte	Name	Typ	Beispiel	Bedeutung
+0	Register 1	Word	W#16#ABCD	
+2	Register 2	Word	W#16#0123	

4.4.5 Funktionscode 05 – Force single coil

Diese Funktion ermöglicht das gezielte Setzen oder Löschen eines Bits. Ist der Slave eine S7, so kann ein Bit in Merkern, Eingängen, Zeitne und Zählern gesetzt oder gelöscht werden.

Anforderungstelegramm Function 05:

Byte	Name	Typ	Beispiel	Bemerkung
0	Adresse	Byte	B#16#09	
1	Funktionscode	Byte	B#16#05	
2+3	Bit-Adresse	Word	W#16#0017	wird beim Senden nicht überprüft
4+5	Bit-Status	WORD	W#16#FF00	FF00h = Bit setzen; 0000h = Bit löschen

Antworttelegramm vom Slave:

Byte	Name	Typ	Beispiel	Bedeutung
0	Adresse	Byte	B#16#09	
1	Funktionscode	Byte	B#16#04	
2+3	Bit-Adresse	Word	W#16#0017	Echo
4+5	Bit-Status	WORD	W#16#FF00	FF00h = Bit gesetzt; 0000h = Bit gelöscht

Im RCV-Bereich in der SPS werden keine Daten abgelegt.

4.4.6 Funktionscode 06 – Preset single register

Diese Funktion ermöglicht das gezielte Schreiben eines Registers. Ist der Slave eine S7, so kann ein Wort in einem DB geschrieben werden.

Anforderungstelegramm Function 06:

Byte	Name	Typ	Beispiel	Bemerkung
0	Adresse	Byte	B#16#09	
1	Funktionscode	Byte	B#16#06	
2+3	Register-Adresse	Word	W#16#0140	wird beim Senden nicht überprüft
4+5	Register-Value	WORD	W#16#1234	

Antworttelegramm vom Slave:

Byte	Name	Typ	Beispiel	Bedeutung
0	Adresse	Byte	B#16#09	
1	Funktionscode	Byte	B#16#06	
2+3	Register-Adresse	Word	W#16#0140	Echo
4+5	Register-Value	WORD	W#16#1234	Gesetzter Wert

Im RCV-Bereich in der SPS werden keine Daten abgelegt.

4.4.7 Funktionscode 07 – Read exception status

Diese Funktion ermöglicht das Lesen der 8 Ereignis-Bits.

Anforderungstelegramm Function 07:

Byte	Name	Typ	Beispiel	Bemerkung
0	Adresse	Byte	B#16#09	
1	Funktionscode	Byte	B#16#07	

Antworttelegramm vom Slave:

Byte	Name	Typ	Beispiel	Bedeutung
0	Adresse	Byte	B#16#09	
1	Funktionscode	Byte	B#16#07	
2	Daten	Byte	B#16#3E	

Daten im RCV-Zielbereich der SPS:

Byte	Name	Typ	Beispiel	Bedeutung
+0	Exception Status	Byte	B#16#3E	
+1		Byte	B#16#xx	wird nicht verändert

4.4.8 Funktionscode 08 – Loop back diagnostic test

Diese Funktion dient der Überprüfung der Kommunikationsverbindung.

Anforderungstelegramm Function 08:

Byte	Name	Typ	Beispiel	Bemerkung
0	Adresse	Byte	B#16#09	
1	Funktionscode	Byte	B#16#08	
2+3	Diag-Code	Word	W#16#0000	Nur Diag-Code 0000h erlaubt
4+5	Testvalue	Word	W#16#A55A	

Antworttelegramm vom Slave:

Byte	Name	Typ	Beispiel	Bedeutung
0	Adresse	Byte	B#16#09	
1	Funktionscode	Byte	B#16#08	
2+3	Diag-Code	Word	W#16#0000	
4+5	Testvalue	Word	W#16#A55A	

Im RCV-Bereich in der SPS werden keine Daten abgelegt.

4.4.9 Funktionscode 11 – Fetch communications event counter

Diese Funktion ermöglicht das Lesen des Status-Words und des Event-Counters.

Anforderungstelegramm Function 11:

Byte	Name	Typ	Beispiel	Bemerkung
0	Adresse	Byte	B#16#09	
1	Funktionscode	Byte	B#16#0B	

Antworttelegramm vom Slave:

Byte	Name	Typ	Beispiel	Bedeutung
0	Adresse	Byte	B#16#09	
1	Funktionscode	Byte	B#16#0B	
2+3	Status-Word	Word	W#16#ABCD	
4+5	Event-Counter	Word	W#16#0123	

Daten im RCV-Zielbereich der SPS:

Byte	Name	Typ	Beispiel	Bedeutung
+0	Status-Word High	Byte	B#16#AB	
+1	Status-Word Low	Byte	B#16#CD	
+2	Event-Counter High	Byte	B#16#01	
+3	Event-Counter Low	Byte	B#16#23	

4.4.10 Funktionscode 12 – Fetch communications event log

Diese Funktion ermöglicht das Lesen des Status-Words, des Event-Counters, des Message-Counters und bis zu 64 Event-Bytes.

Anforderungstelegramm Function 12:

Byte	Name	Typ	Beispiel	Bemerkung
0	Adresse	Byte	B#16#09	
1	Funktionscode	Byte	B#16#0C	

Antworttelegramm vom Slave:

Byte	Name	Typ	Beispiel	Bedeutung
0	Adresse	Byte	B#16#09	
1	Funktionscode	Byte	B#16#0B	
2	Bytezähler	Byte	B#16#04	
3+4	Status-Word	Word	W#16#ABCD	
5+6	Event-Counter	Word	W#16#0123	
7+8	Message-Counter	Word	W#16#0102	
9	Event-Byte 1	Byte		
...				
73	Event-Byte 64	Byte		

Daten im RCV-Zielbereich der SPS:

Byte	Name	Typ	Beispiel	Bedeutung
+0	Status-Word High	Byte	B#16#AB	
+1	Status-Word Low	Byte	B#16#CD	
+2	Event-Counter High	Byte	B#16#01	
+3	Event-Counter Low	Byte	B#16#23	
+4	Message-Counter High	Byte	B#16#01	
+5	Message-Counter Low	Byte	B#16#23	
+6	Event-Byte 1	Byte		
...	...			
+69	Event-Byte 64	Byte		

4.4.11 Funktionscode 15 – Force Multiple Coils

Diese Funktion ermöglicht das Setzen von Bits im Slave. Ist der Slave eine S7, so können abhängig von der Bit-Startadresse Merker, Ausgänge, Zeiten oder Zähler geschrieben werden.



Die Bytes der Sendedaten werden vor dem Senden byteweise gedreht. Der Bytezähler wird automatisch generiert und darf nicht in den Sendedaten enthalten sein.

Anforderungstelegramm Function 15:

Byte	Name	Typ	Beispiel	Bemerkung
0	Adresse	Byte	B#16#09	
1	Funktionscode	Byte	B#16#0F	
2+3	Bit-Startadresse	Word	W#16#0020	wird beim Senden nicht überprüft
4+5	Bit-Count	Int	12	1-2040
	Bytezähler	Byte	2	<i>Wird automatisch errechnet! Der Bytezähler ist nicht in den Sendedaten der S7 enthalten!</i>
	Coil-Status	Word	W#16#1230	

Der Bytezähler wird vom Modbus-Treiber automatisch generiert und in die Sendedaten eingefügt. Er darf nicht in den Auftragsdaten enthalten sein.

Antworttelegramm vom Slave:

Byte	Name	Typ	Beispiel	Bedeutung
0	Adresse	Byte	B#16#09	
1	Funktionscode	Byte	B#16#0F	
2+3	Bit-Startadresse	Word	W#16#0020	
4+5	Bit-Count	Int	12	

Im RCV-Bereich in der SPS werden keine Daten abgelegt.

4.4.12 Funktionscode 16 – Preset multiple registers

Diese Funktion ermöglicht das Schreiben von Worten im Slave. Ist der Slave eine S7, so können Worte in einem DB geschrieben werden.



Der Bytezähler wird automatisch generiert und darf nicht in den Sendedaten enthalten sein.

Anforderungstelegramm Function 16:

Byte	Name	Typ	Beispiel	Bemerkung
0	Adresse	Byte	B#16#09	
1	Funktionscode	Byte	B#16#10	
2+3	Register-Start	Word	W#16#0002	wird beim Senden nicht überprüft
4+5	Register-Anzahl	Int	2	1-127
	Bytezähler	Byte	4	<i>Wird automatisch errechnet! Der Bytezähler ist nicht in den Sendedaten der S7 enthalten!</i>
	Register 1	Word	W#16#1230	
	Register 2	Word	W#16#1230	

Der Bytezähler wird vom Modbustreiber automatisch generiert und in die Sendedaten eingefügt. Er darf nicht in den Auftragsdaten enthalten sein.

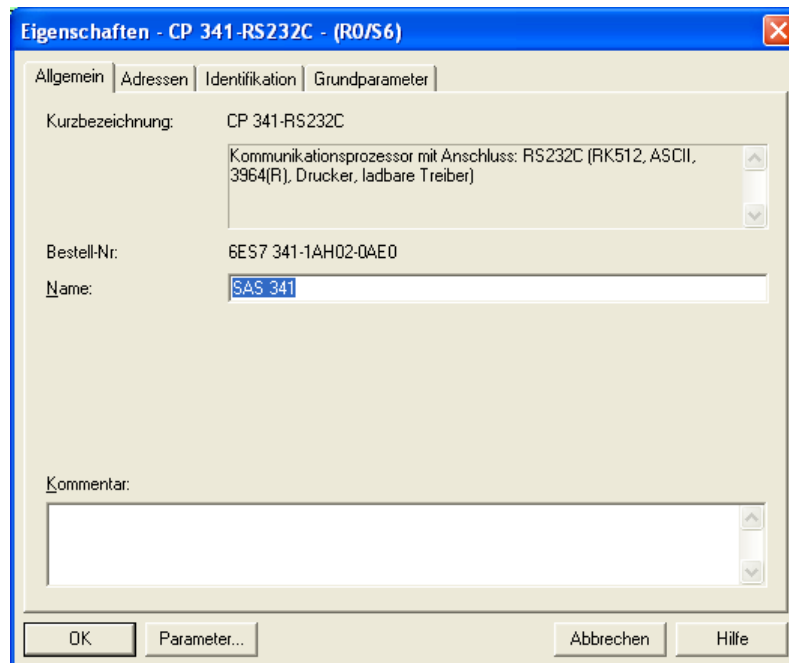
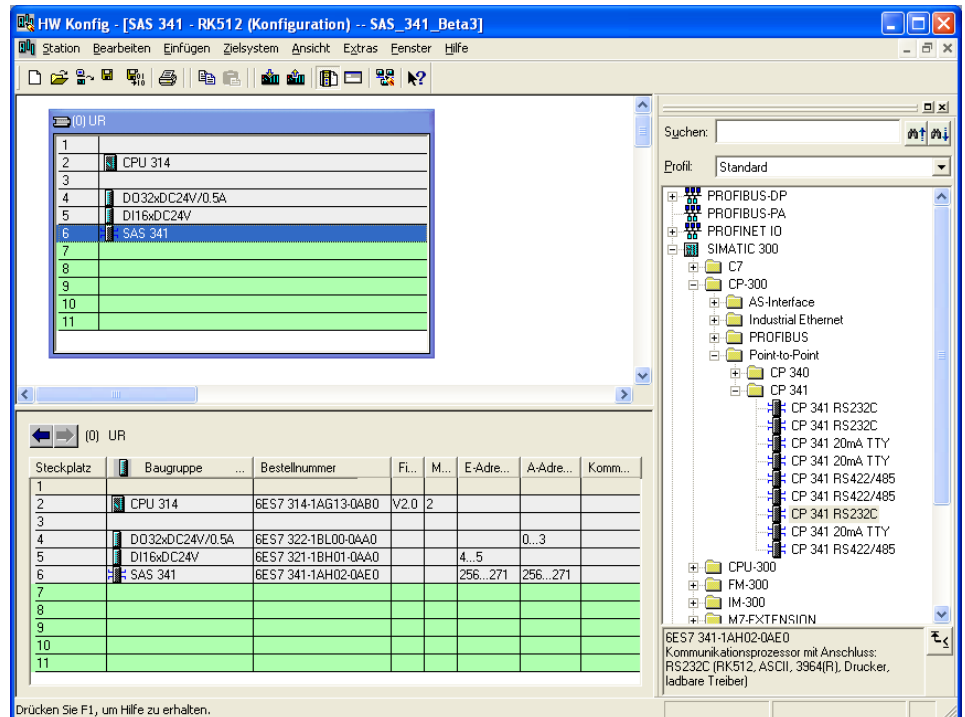
Antworttelegramm vom Slave:

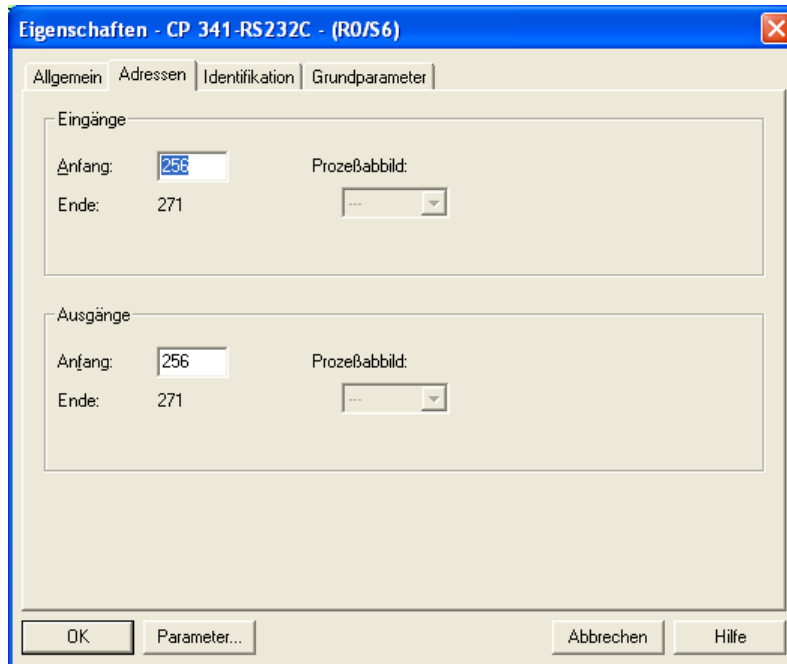
Byte	Name	Typ	Beispiel	Bedeutung
0	Adresse	Byte	B#16#09	
1	Funktionscode	Byte	B#16#10	
2+3	Register-Start	Word	W#16#0002	
4+5	Register-Anzahl	Int	2	

Im RCV-Bereich in der SPS werden keine Daten abgelegt.

5 Konfigurieren der Baugruppe

Die SAS 341 Baugruppe wird in der Programmiersoftware der SPS als CP 341 Kommunikationsbaugruppe projektiert (6ES7-341-1AH02, -1CH02).



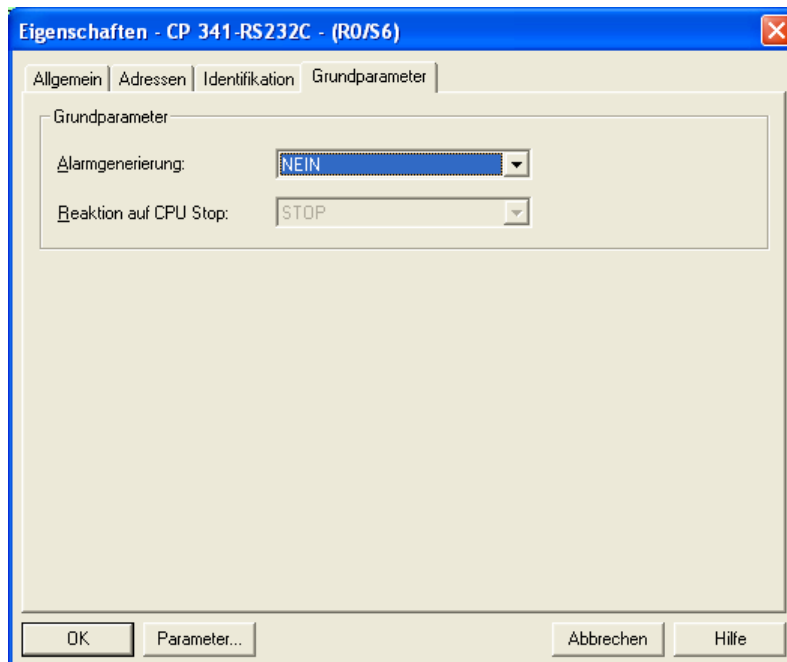


Es wird in den Hantierungsbausteinen nur die Eingangsadressen verwendet, die Ausgangsadressen haben keine funktionale Bedeutung.

Zugriffe auf die Eingangsperipherie können nur mit den Peripherie-direktzugriffsbefehlen durchgeführt werden: L PEB, L PEW.

Bei der CPU 318 müssen die Peripherieadressen außerhalb des zyklischen Prozessabbildes liegen.

Die Einstellung auf den Seiten „Identifikation“ und „Grundparameter“ haben keine Bedeutung, da diese Funktionen nicht unterstützt werden.



6 Parametrierung der Baugruppe

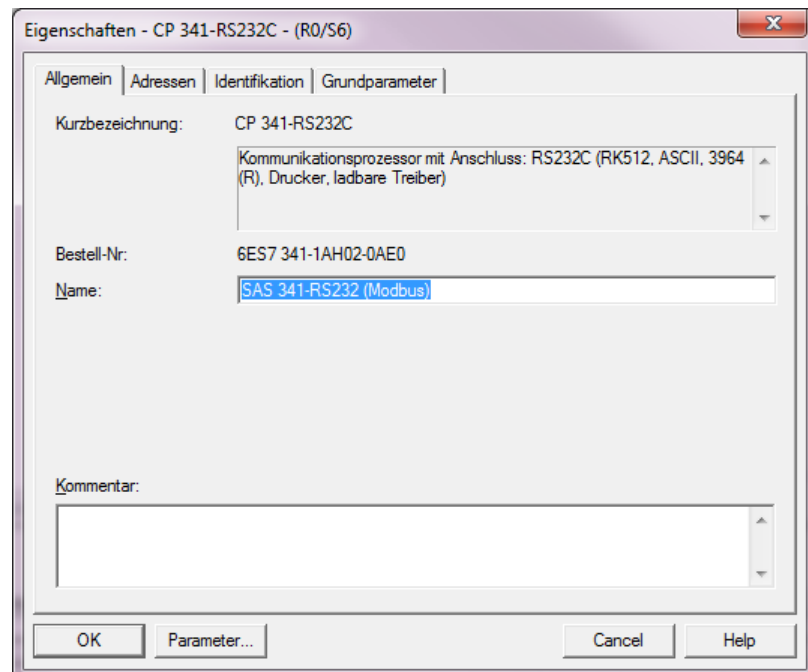
6.1 Parametriersoftware installieren

Zur weiteren Einstellung der Parameter der Prozeduren wird die Parametrieroberfläche „CP341: Punkt-zu-Punkt-Kopplung parametrieren“ (PtP Param V5.1 ab SP10) in Step 7 (ab V5.3) benötigt, welche nachträglich installiert werden muss.

Die Software kann bei Siemens hier heruntergeladen werden:

<http://support.automation.siemens.com/WW/view/de/27013524>

Ist die Software installiert, ist im Hardwaremanager im Dialog „Eigenschaften“ der Button „Parameter...“ aktiv, der zur Parametrierung der Baugruppe führt.



Für die Modbus Parametrierung ist jeweils noch eine weitere Parametriersoftware notwendig.

Modbus Master (ab Version 3.1 + SP6):

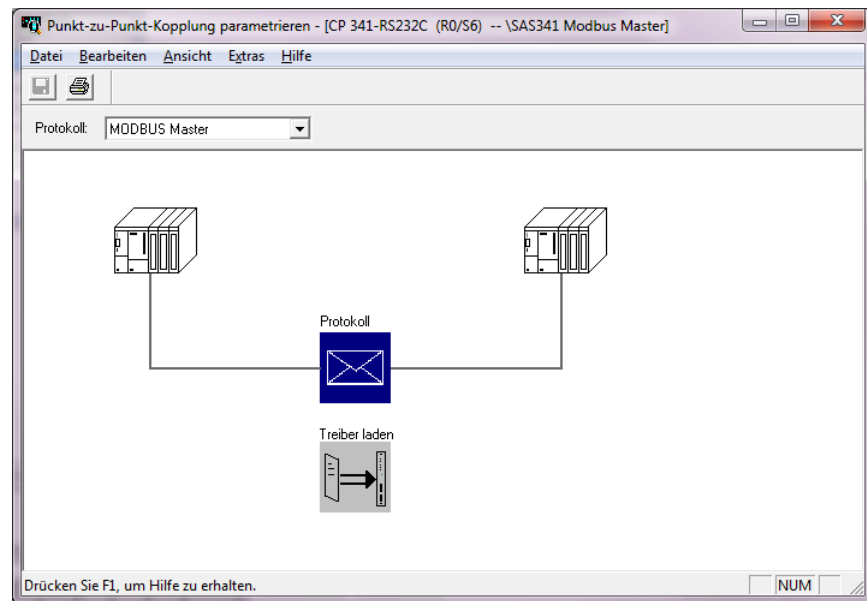
<http://support.automation.siemens.com/WW/view/de/27774018>

Modbus-Slave (ab Version 3.1 + SP7):

<http://support.automation.siemens.com/WW/view/de/27774276>

Die entsprechende Software muss heruntergeladen und installiert sein. Es können auch beide (Master/Slave) installiert sein.

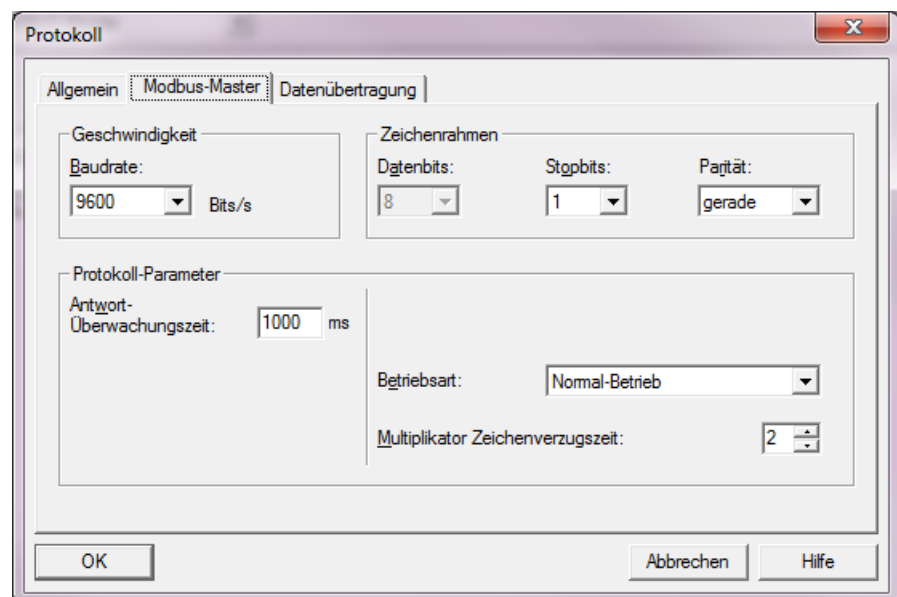
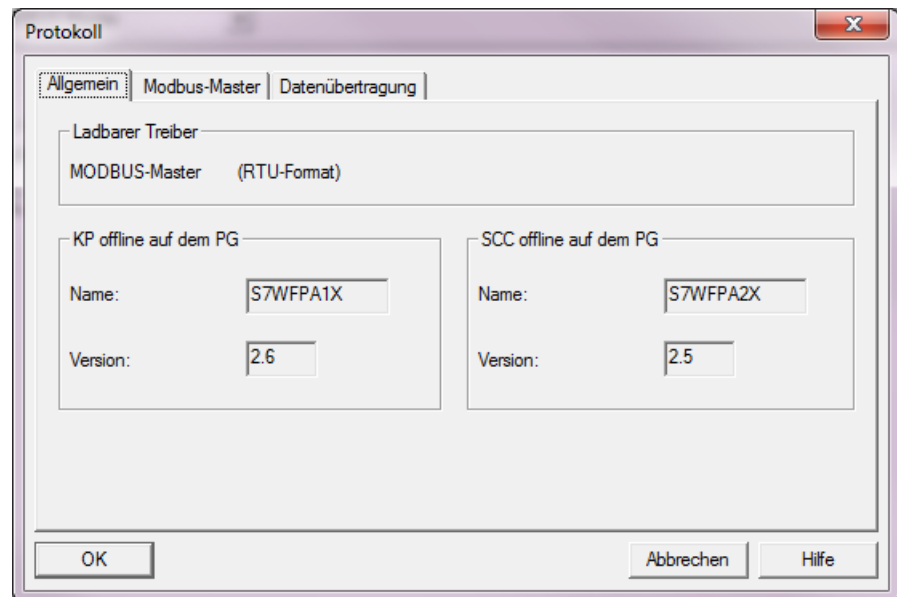
Nach Betätigung des Buttons „Parameter...“ wird die eigenständige Parametrieroberfläche aufgerufen.



Die Funktion „Treiber laden“ ist für die SAS 341 nicht relevant, beide Modbus-Treiber (Master & Slave) werden durch die SAS-Modbus Treiber MMC automatisch freigeschaltet.

6.2 Modbus Master

Der erste Dialog-Reiter „Allgemein“ dient nur der Information. Die Einträge „KP offline auf dem PG“ und „SCC offline auf dem PG“ haben für Anwendung mit der SAS 341 keine Relevanz.



Geschwindigkeit/ Zeichenrahmen: Festlegen der Übertragungsgeschwindigkeit (300 Baud bis 115KBaud) und des Zeichenrahmens (Stopbits/Parity).

Antwort-Überwachungszeit (5 bis 65500 ms): Zeit, die nach der Ausgabe eines Anforderungstelegramms auf ein Antworttelegramm vom Modbus-Slave gewartet wird

Betriebsart:

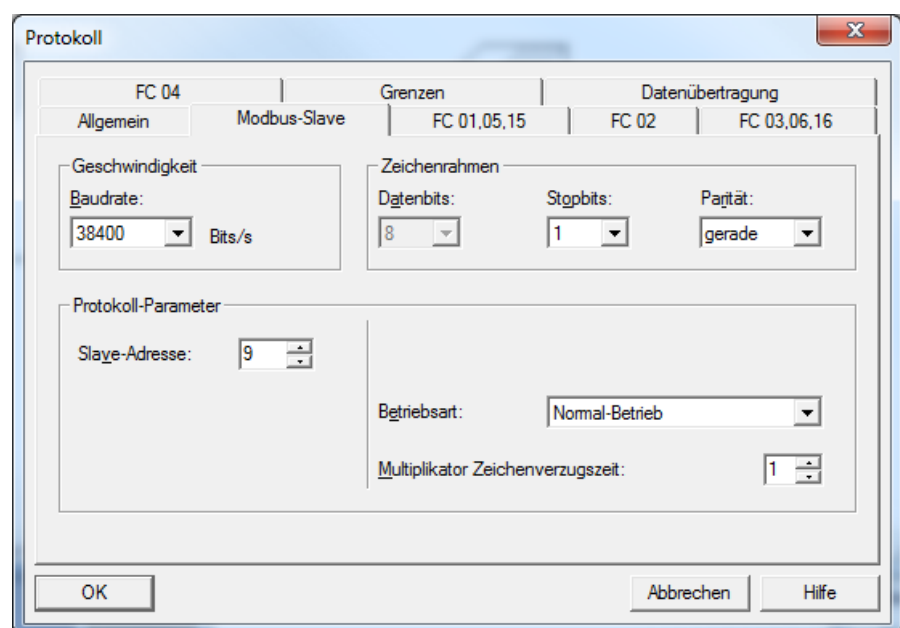
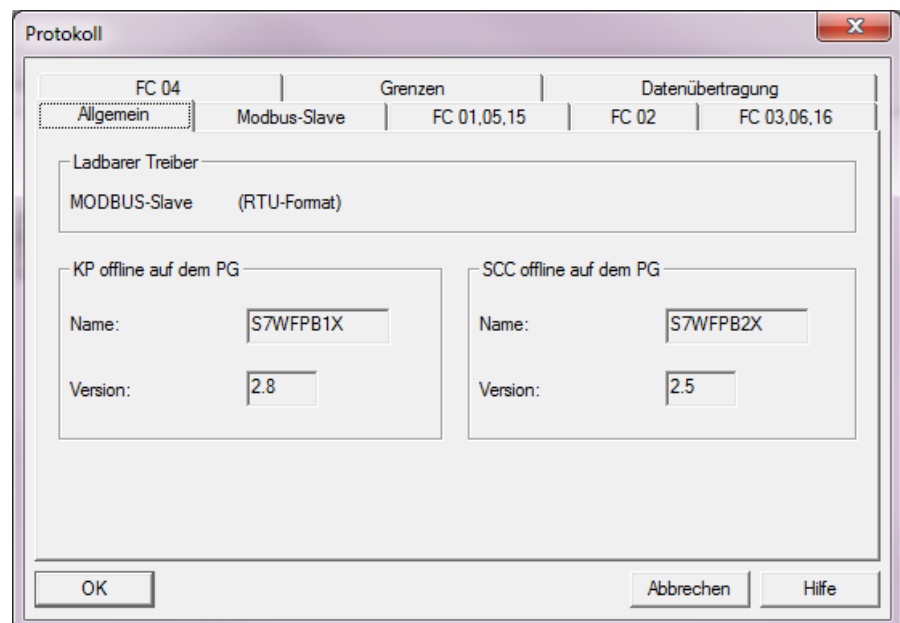
Normal-Betrieb = Übertragungsfehler führen zu einer Fehlermeldung.

Störungsunterdrückung = Übertragungsfehler werden ignoriert.

Multiplikator Zeichenverzugszeit (1 bis 10): Wenn ein Slave die zeitlichen Anforderungen der MODBUS-Spezifikation nicht einhalten kann, kann die Zeichenverzugszeit durch den Multiplikationsfaktor vervielfacht werden.

6.3 Modbus Slave

Der erste Dialog-Reiter „Allgemein“ dient nur der Information. Die Einträge „KP offline auf dem PG“ und „SCC offline auf dem PG“ haben für Anwendung mit der SAS 341 keine Relevanz.



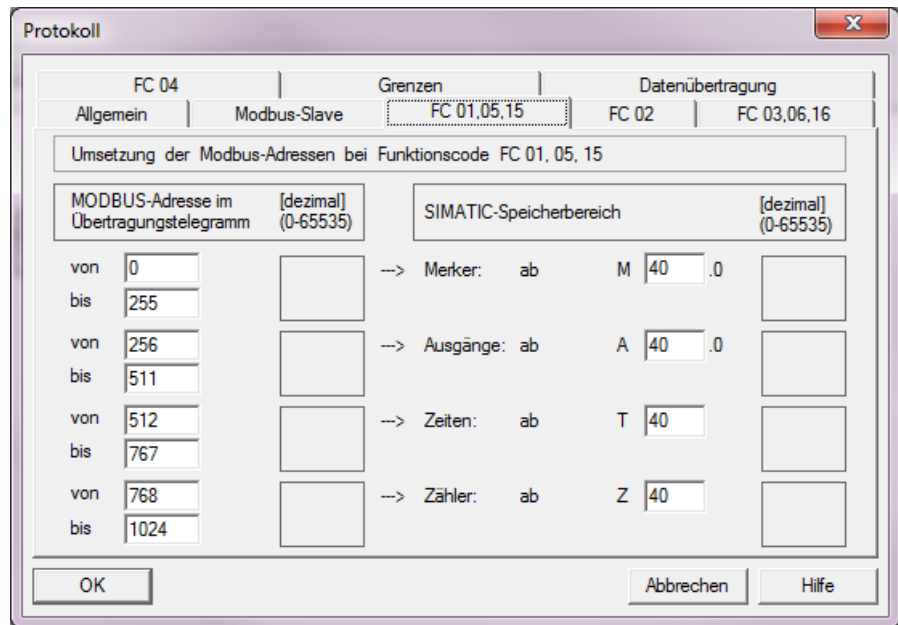
Geschwindigkeit/ Zeichenrahmen: Festlegen der Übertragungsgeschwindigkeit (300 Baud bis 115KBaud) und des Zeichenrahmens (Stopbits/Parity).

Slave-Adresse (1 bis 255): Adresse des Modbus-Teilnehmers

Betriebsart: Normal-Betrieb = Übertragungsfehler führen zu einer Fehlermeldung.

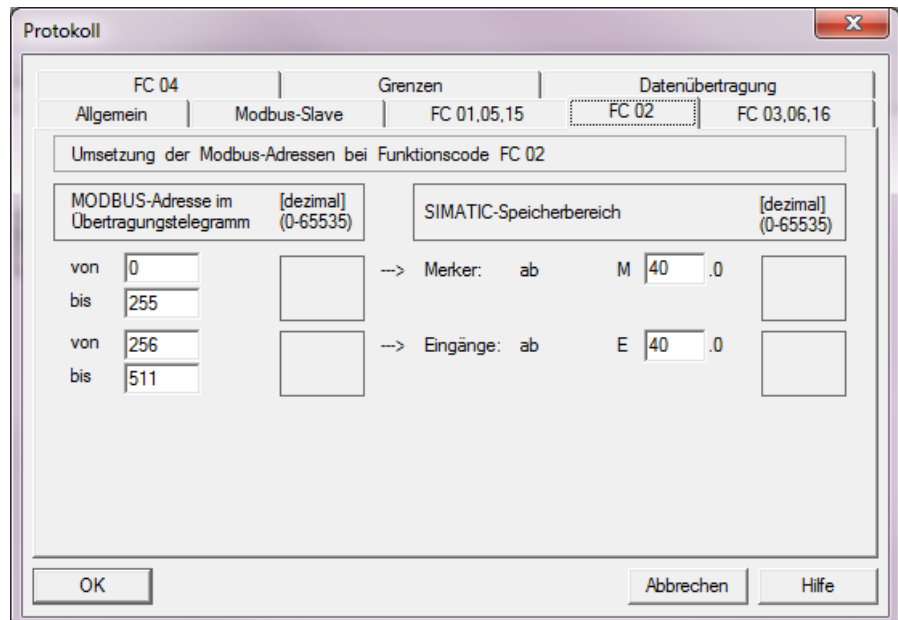
Störungsunterdrückung = Übertragungsfehler werden ignoriert.

Multiplikator Zeichenverzugszeit (1 bis 10): Wenn ein Slave die zeitlichen Anforderungen der MODBUS-Spezifikation nicht einhalten kann, kann die Zeichenverzugszeit durch den Multiplikationsfaktor vervielfacht werden.



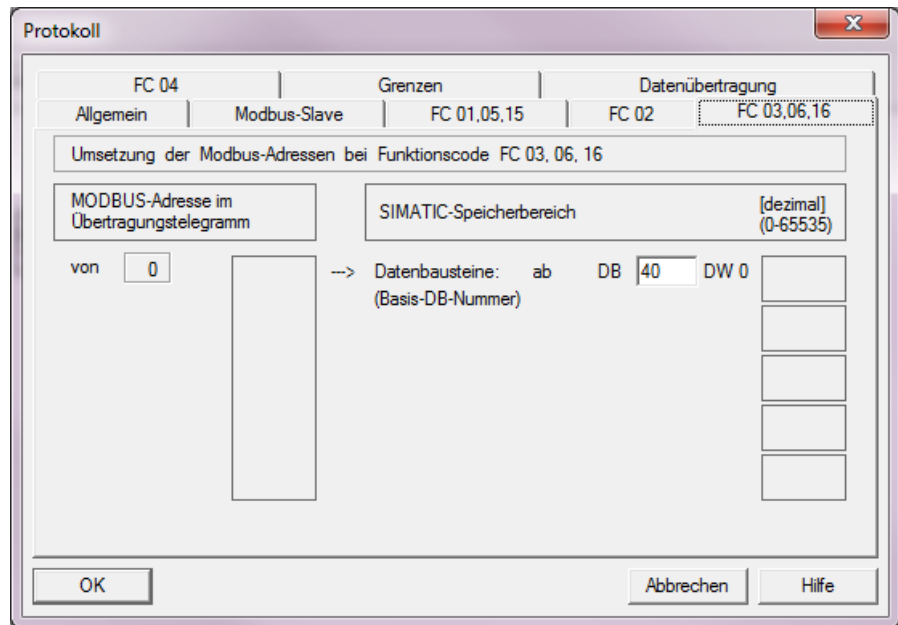
Für die Funktionscodes 1, 5, 15 (Read Coil Status, Force Single Coil, Force Multiple Coils) können für entsprechenden Coiladressen (Bitadresse) die Zuordnungen zu Merkern, Ausgängen, Zeiten und Zählern definiert werden.

Die einzelnen Bereiche dürfen sich nicht überlappen, Lücken sind aber erlaubt. Wenn bei „von“ und bei „bis“ 0 eingetragen wird, wird keine Zuordnung hergestellt.

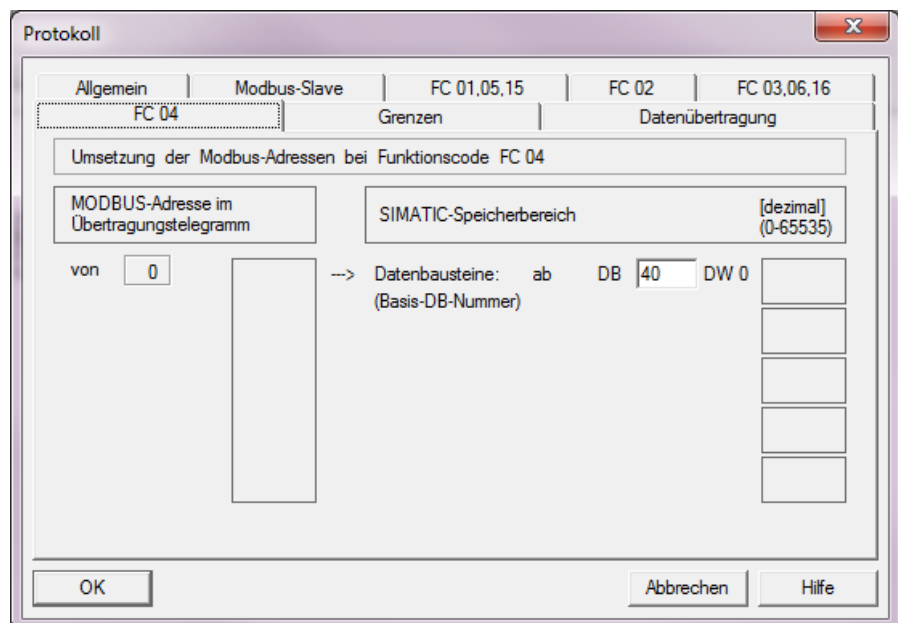


Für den Funktionscodes 2 (Read Input Status) kann für Inputadressen (Bitadressen) die Zuordnungen zu Merkern oder Eingängen hergestellt werden.

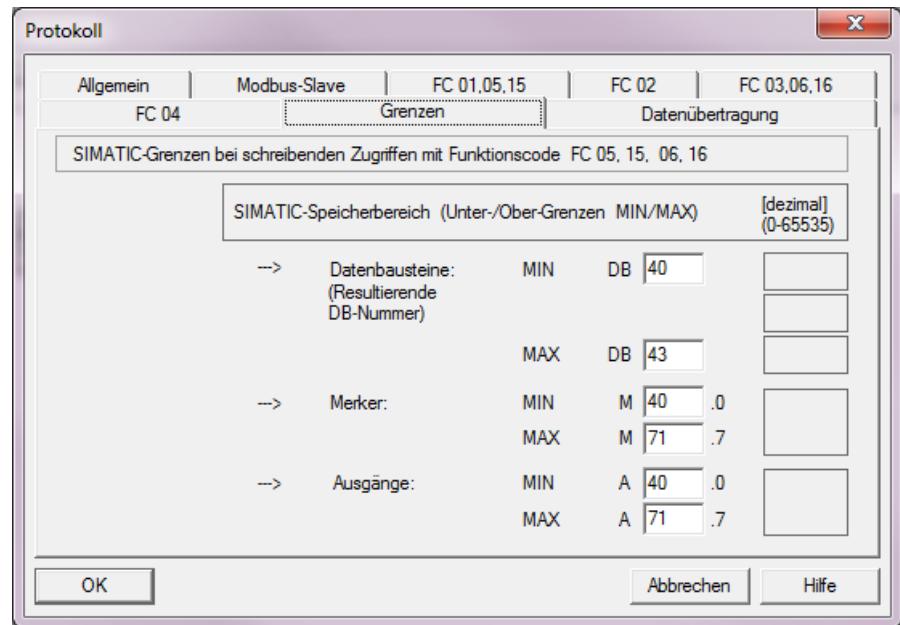
Die einzelnen Bereiche dürfen sich nicht überlappen, Lücken sind aber erlaubt. Wenn bei „von“ und bei „bis“ 0 eingetragen wird, wird keine Zuordnung hergestellt.



Für die Funktionscodes 3, 6, 16 (Read Holding Registers, Preset Single Register, Preset Multiple Registers) kann für den Registerspeicher eine Zuordnungen zu Datenbausteinen definiert werden. Das erste Register 0 greift auf den definierten DB in das DBW 0. Es wird wortweise zugegriffen. Ab Register Adresse 512, 1024, etc. wird auf den nachfolgenden DB zugegriffen.



Für den Funktionscodes 4 (Read Input Registers) kann für den Registerspeicher eine Zuordnungen zu Datenbausteinen definiert werden. Das erste Register 0 greift auf den definierten DB in das DBW 0. Es wird wortweise zugegriffen. Ab Register Adresse 512, 1024, etc. wird auf den nachfolgenden DB zugegriffen.



Bei schreibenden Funktionscodes (5, 15, 6, 16) können Grenzen festgelegt werden, um ein versehentliches Schreiben ausserhalb der gewünschten Bereiche zu verhindern

7 Programmierung in der SPS

7.1 Übersicht

Die Programmierung der SAS 341 Baugruppe erfolgt in der SPS über die im Softwarepaket enthaltenen Hantierungsbausteine.

Die Hantierungsbausteine sind in Ihrer Funktion und in den Aufrufparametern an die Siemens CP341 Hantierungsbausteine angelehnt. Die Hantierungsbausteine liegen getrennt für Modbus Master und Modbus Slave vor.

Folgende Bausteine stehen für die Kommunikation als Modbus Master zur Verfügung:

FB 4 MODB_MAST_SND Modbus-Aufträge versenden

FB 5 MODB_MAST_RCV Modbus-Aufträge empfangen

Folgender Baustein stehen für die Kommunikation als Modbus Slave zur Verfügung:

FB 80 MODBUS_SLAVE Modbus-Aufträge als Slave verarbeiten

7.2 Peripheriedaten in der SPS

Die SAS 341 Baugruppe belegt 16 Bytes im Eingangs- und im Ausgangs-Peripheriebereich der SPS. Der Inhalt des Ausgangsbereiches wird nicht verwendet.

Der Inhalt des Eingangsbereiches kann vom Anwender in der Applikation zu Informationszwecken verwendet werden:

Byte	Bedeutung
0	Baugruppenstatus allgemein, Sammelfehler Anzeige
1	Statussignale Kanal 1
2	FIFO-Status Kanal 1
3	Fehlerbits Kanal 1
4	aktives Protokoll Kanal 1
5	Statussignale Kanal 2
6	FIFO-Status Kanal 2
7	Fehlerbits Kanal 2
8	aktives Protokoll Kanal 2
9	<i>reserviert</i>
10	<i>reserviert</i>
11...15	<i>intern verwendet</i>

Zugriffe auf das Eingangsabbild können nur mit den Peripherie-direktzugriffsbefehlen durchgeführt werden: L PEB, L PEW.

Die Bytes 5, 6 und 7 werden nur bei SAS-Baugruppen mit 2 Schnittstellen (SAS 340-2/341-2) mit auswertbaren Daten belegt und haben bei Verwendung des Modbus-Treibers keine Bedeutung.

7.2.1 Byte 0: Baugruppenstatus

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Immer 1 = Baugruppe ist da	0 = SAS 340 1 = SAS 341	0 = 1 Port 1 = 2 Ports	0	0	Sammelfehler Kanal 2	Sammelfehler Kanal 1	Baugruppe parametrierung und läuft

Bit 0: Die SAS 341 Baugruppe ist betriebsbereit.

Bit 1: Sammelfehler der Kanal 1

Bit 5: Kanalanzahl (0 = 1 Kanal)

Bit 6: Typkennung der Baugruppe (1 = SAS 341)

Bit 7: Bit ist immer gesetzt zur Erkennung der Baugruppe, falls die SAS 341 korrekt angelaufen ist.

7.2.2 Byte 1: Statussignale Kanal 1

Bit 7	Bit 6	Bit 5	Bit 4
0	BREAK erkannt (In)	RI Ring Indicator (In)	DCD Data Carrier Detect (In)

Bit 3	Bit 2	Bit 1	Bit 0
DTR (Out)	DSR (Out)	CTS (In)	RTS (Out)

7.2.3 Byte 2: FIFO-Status Bits Kanal 1

Bit 7	Bit 6	Bit 5	Bit 4
0	0	Send-FIFO halb voll	Send-FIFOs ganz leer

Bit 3	Bit 2	Bit 1	Bit 0
0	0	Receive-FIFO halb voll	Receive-FIFOs ganz leer

7.2.4 Byte 3: Fehler-Bits Kanal 1

Bit 7	Bit 6	Bit 5	Bit 4
Protokoll CRC-Fehler	0	Send-FIFO Overflow	Receive-FIFOs Overflow

Bit 3	Bit 2	Bit 1	Bit 0
0	Schnittstelle Overflow	Parity-Fehler	Framing-Fehler

7.2.5 Byte 4: Aktives Protokoll Kanal 1

0x30 = ASCII

0x31 = 3964R

0x20 = RK512

0x40 = Modbus RTU

7.3 Hantierungsbausteine Modbus Master

7.3.1 FB 4 MODB_MAST_SND

Mit dem Funktionsbaustein FB 4 – MODB_MAST_SND werden Modbusaufträge versendet. Das Telegramm muss vorher in einem Datenbereich (z.B. Datenbaustein) vollständig vorliegen. Der Funktionsbaustein sollte zyklisch aufgerufen werden.

Parameter	Richtung	Typ	Funktion
Req	IN	BOOL	Auftrag auslösen
R	IN	BOOL	Rücksetzen der Auftragsbearbeitung
LADDR	IN	INT	Basisadresse der SAS 341
Src	IN	ANY	Pointer auf Datenbereich für das Sendetelegramm
Len	IN	INT	Länge des Telegramms (in Bytes)
Busy	OUT	BOOL	Auftragsbearbeitung läuft
Done	OUT	BOOL	Auftrag fertig ohne Fehler
Error	OUT	BOOL	Auftrag fertig mit Fehler
Status	OUT	INT	Fehlernummer bei <i>Error</i> (siehe Kap. 7.3.3)

Im Sendebereich müssen die Daten des Modbus Telegramm vorher aufgebaut werden, siehe Kapitel 4.4.

Ausnahmen: Die CRC und der Bytezähler werden vom Modbus-treiber selber generiert.

Der Baustein FB 4 überträgt einen Modbus-Auftrag an die SAS 341 Baugruppe. Wenn der Auftrag korrekt übergeben werden konnte, dann wird **Done** für einen Zyklus auf 1 gesetzt.

Die SAS 341 versendet den Auftrag und wartet auf die Antwort vom Modbus-Slave. Alle Antworten mit und ohne Daten und auch fehlerhafte Verarbeitung (z.B. Slave antwortet nicht) werden durch den FB 5 – MODB_MAST_RCV verarbeitet und angezeigt.

Ein neuer Auftrag sollte erst abgesendet werden, wenn am FB 5 – MODB_MAST_RCV die Antwort (positiv oder Fehler) angekommen ist.

Ausnahme: Beim Versenden von Broadcasts (Adresse 0) werden keine Antworten generiert. Nach dem Versenden des Broadcasts (Bit **Done** gesetzt) kann der nächste Auftrag gestartet werden.



Wird ein Braodcast versendet, so wird kein Antworttelegramm am Bus übertragen.

Aufrufbeispiel für FB 4:

```
CALL FB      4 , DB4
Req   :=M10.0
R     :=M10.1
LADDR :=256
Src   :=P#DB20.DBX0.0 BYTE 262
Len   :=MW12
Busy  :=M10.5
Done  :=M10.6
Error :=M10.7
Status:=MW14

UN    M    10.7           // Error ?
SPB   SX
L     MW   14
T     MW   16           // Status merken
SPA   Next

SX:   UN    M    10.6           // Auftrag Done?
      SPB   Next
      R     M    10.0
      L     0
      T     MW   16           // Status löschen

Next: ...
```

7.3.2 FB 5 MODB_MAST_RCV

Der Funktionsbaustein FB 5 – MODB_MAST_RCV empfängt die Antworttelegramme der Slaves und stellt die Daten des Telegrammes im Receive-Bereich (Pointer **Dest**) zur Verfügung, wenn der Auftrag eine Datenanforderung enthält. Der Funktionsbaustein sollte zyklisch aufgerufen werden.

Parameter	Richtung	Typ	Funktion
EN_R	IN	BOOL	Freigabe für Daten lesen
R	IN	BOOL	Rücksetzen der Empfangsfunktion
LADDR	IN	INT	Basisadresse der SAS 341
Dest	IN	ANY	Pointer auf Datenbereich für empfangene Daten
Done	OUT	BOOL	Auftrag fertig ohne Fehler
NDR	OUT	BOOL	Neue Daten liegen vor
Error	OUT	BOOL	Auftrag fertig mit Fehler
Len	OUT	INT	Länge der Daten (in Bytes)
Status	OUT	INT	Fehlernummer bei <i>Error</i> (siehe Kap. 7.3.3)



Wird ein Braodcast versendet, so wird kein Antworttelegramm am Bus übertragen. Der FB 5 zeigt nichts an!

Alle Antworten auf Aufträge, die vom FB 4 MODB_MAST_SND versendet worden sind, werden hier verarbeitet. Es können Antworten ohne Daten sein (**Done** wird für einen Zyklus gesetzt) und Antworten mit Daten (**Done**, **NDR** und **Len** werden für einen Zyklus gesetzt). Ebenso werden Fehlerantworten verarbeitet (**Error** und **Status** werden für einen Zyklus gesetzt).

Bei welchen Funktionscodes Daten im Receive-Bereich (Pointer **Dest**) zur Verfügung gestellt werden, ist der Protokollbeschreibung im Kapitel 4.4 zu entnehmen (→ „Daten im RCV-Zielbereich der SPS“).

Aufrufbeispiel für FB 5:

```

RCV: CALL FB      5 , DB5
      EN_R  :=M11.0
      R     :=M11.1
      LADDR :=256
      Dest  :=P#DB21.DBX0.0 BYTE 262
      Done  :=M11.5
      NDR   :=M11.6
      Error :=M11.7
      Len   :=MW20
      Status:=MW24

      UN    M    11.7           // Fehler aufgetreten?
      SPB   RDon
      L     MW    24
      T     MW    26           // Fehler merken
      SPA   End

RDon: UN    M    11.5           // Auftrag fertig?
      SPB   End

      UN    M    11.6           // Daten empfangen
      SPB   End
      L     MW    20           // Länge kopieren
      T     MW    22
      ...                       // Daten verarbeiten

End:  ...

```

7.3.3 Fehlernummern Modbus Master Treiber

Fehler bei der Bearbeitung der Funktionsbausteine werden in der Regel durch ein gesetztes **Error**-Bit angezeigt. Die Fehlerursache wird im Ausgabeoperanden **Status** dargestellt.

- 0502 Baugruppe nicht vorhanden, oder nicht bereit
- 0708 Ablauf des Handshake-Timeouts (CTS oder XON)
- 07F0 SRC-Buffer ist kleiner als LEN
- 07F4 Sendefach in der SAS 341 noch belegt
- 0806 Ablauf der Zeichenverzugszeit
- 080A Pufferüberlauf
- 0830 Ablauf der Antwortüberwachungszeit
- 08F0 Falsche Kennung von SAS erhalten (interner Fehler)
- 08F2 Empfangspuffer DEST nicht gross genug

0E4x Fehler beim SEND-Auftrag

- 0E40 Parameter LEN < 2 Bytes
- 0E41 Parameter LEN < 6 Bytes (abhängig von Funktionscode)
- 0E42 Funktionscode nicht zulässig/ungültig
- 0E47 LEN passt nicht zu Bitcount oder Registercount (bei Funktionscode 15 & 16)

0E5x Antworttelegramm fehlerhaft

- 0E50 Falsche Slave-Adresse im Antworttelegramm
- 0E51 Falscher Funktionscode im Antworttelegramm
- 0E57 CRC falsch (oder Telegrammlänge < 4)

0E6x Fehlerantwort vom Slave / x = Exceptioncode

- 0E61 Exceptioncode 1: Illegal function
- 0E62 Exceptioncode 2: Illegal data address
- 0E63 Exceptioncode 3: Illegal data value
- 0E64 Exceptioncode 4: Failure in associated device
- 0E65 Exceptioncode 5: Acknowledge
- 0E66 Exceptioncode 6: Busy, rejected message
- 0E67 Exceptioncode 7: Negative acknowledgement

1Exx Systemfehler

- 1E0E Fehler ist RETVAL von SFC58, SFC 59 oder SFC 20.
Der Wert von RETVAL liegt im Instanzdatenbaustein des aufgerufenen FBs.

7.4 Hantierungsbaustein Modbus Slave

7.4.1 FB 80 MODBUS_SLAVE

Der Funktionsbaustein FB 80 – MODBUS_SLAVE führt die Protokollverarbeitung der von der SAS 341 empfangenen Modbus-Telegramme durch. Der FB 80 verwendet die in der Hardwarekonfiguration eingestellten Speicherbereiche (siehe Kapitel 6.3) um je nach Funktionscode Daten zu lesen oder zu schreiben.

Parameter	Richtung	Typ	Funktion
LADDR	IN	INT	Basisadresse der SAS 341
START_TIMER	IN	T	Timer für Anlaufüberwachung
START_TIME	IN	S5T	Zeit für Anlaufüberwachung
CP_START_OK	OUT	BOOL	TRUE: Anlauf ohne Fehler durchgeführt
CP_START_ERROR	OUT	BOOL	Anlauf mit Fehler (Überwachungszeit)
CP_NDR	OUT	BOOL	Neue Daten wurden empfangen
RCV_ERROR	OUT	BOOL	Fehler beim Telegramm holen von der SAS
SND_ERROR	OUT	BOOL	Fehler beim Telegramm senden zur SAS
RCV_STATUS	OUT	WORD	Fehlernummer beim Telegramm holen
SND_STATUS	OUT	WORD	Fehlernummer beim Telegramm senden
CP_START	IN_OUT	BOOL	FB Initialisierung starten
MB_ERROR_NR	IN_OUT	WORD	Fehlernummer (siehe 7.4.2)
MB_ERROR_INFO	IN_OUT	WORD	Zusatzinfo zum Fehler

Der FB 80 sollte im SPS-Programm zyklisch aufgerufen werden.

Aufrufbeispiel für FB 80:

```
CALL FB 80 , DB80
LADDR      :=256
START_TIMER :=T10
START_TIME  :=S5T#3S
CP_START_OK :=M10.0
CP_START_ERROR:=M10.1
CP_NDR      :=M10.5
RCV_ERROR   :=M10.2
SND_ERROR   :=M10.3
RCV_STATUS  :=MW12
SND_STATUS  :=MW14
CP_START    :=M10.4
MB_ERROR_NR :=MW16
MB_ERROR_INFO :=MW18
```

7.4.2 Fehlernummern Modbus Slave

MB_ERROR_NR & MB_ERROR_INFO:

- 0 = Fehlerfrei
- 1 = SFC51-Fehler; RETVAL in MB_ERROR_INFO
- 2 = Timeout Initialisierung
- 3 = SFC102-Fehler; RETVAL in MB_ERROR_INFO
- 12 = Register-Anzahl = 0
- 13 = max. Register-Anzahl überschritten
- 14 = Speicherbereich Merker überschritten
- 15 = Speicherbereich Ausgang überschritten
- 16 = Speicherbereich Zeiten überschritten
- 17 = Speicherbereich Zähler überschritten
- 18 = Speicherbereich Eingang überschritten
- 19 = Speicherbereich DB überschritten
- 20 = Speicherbereich DB nicht vorhanden
- 40 = ungültiger Modbus Funktionscode empfangen
- 41 = ungültiger Diagnostic-Code bei Funktionscode 8 empfangen
- 50 = Registerzugriff ausserhalb des parametrierten Bereichs
- 51 = Zeiten u. Zähler: Registeranzahl nicht durch 16 teilbar
- 52 = Zeiten u. Zähler: Startregister fehlerhaft
- 60 = Merker: Registerzugriff überschreitet den parametrierten Bereich
- 61 = Ausgang: Registerzugriff überschreitet den parametrierten Bereich
- 62 = Eingang: Registerzugriff überschreitet den parametrierten Bereich
- 63 = Zeiten: Registerzugriff überschreitet den parametrierten Bereich
- 64 = Zähler: Registerzugriff überschreitet den parametrierten Bereich
- 70 = Zugriff auf Merker ausserhalb Grenzen
- 71 = Zugriff auf Ausgang ausserhalb Grenzen
- 72 = Zugriff auf DB ausserhalb Grenzen
- 73 = schreibender Zugriff auf Zeiten und Zähler nicht zulässig

RCV_STATUS & SND_STATUS:

- 0502 Baugruppe nicht vorhanden, oder nicht bereit
- 07F0 SRC-Buffer ist kleiner als LEN
- 07F4 Sendefach in der SAS 341 noch belegt
- 08F0 Falsche Kennung von SAS erhalten (interner Fehler)
- 08F2 Empfangspuffer nicht gross genug
- 1E0E Fehler ist RETVAL von SFC58, SFC 59. Der Wert von RETVAL liegt im Instanzdatenbaustein des aufgerufen FBs.

8 Anhang

8.1 Technische Daten

Bestellnummer SAS 341-1, 1xRS232+USB 700-341-1AH02
SAS 341-1, 1xRS422/485 700-341-1CH02

Abmessungen 116 x 40 x 125 mm (LxBxH)

Gewicht ca. 200g

Kommunikations-Schnittstelle

Typ: RS 232 (V.24), SubD 9pol. male,
RS422/485 (X27), SubD 15pol. male
USB 1.1 (*nur bei 700-341-1AH02*)

Übertragungsrate: 300 Baud bis 115Kbaud

Protokolle: ASCII
3964R
RK512
Modbus RTU Master/Slave (*mit Treiberkarte*)

Service-Schnittstelle (Update/Diagnose)

Typ: USB 1.1
Übertragungsrate: Fullspeed 12MBit
Anschluss: Mini-USB

Versorgung

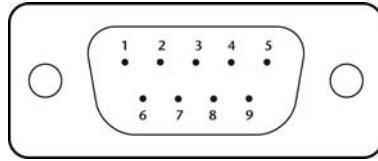
Spannung: DC +5V über Rückwandbus
Stromaufnahme: 160mA (typ.) / 190mA (max.)

Besonderheiten

Qualitätssicherung: nach ISO 9001:2008
Wartung: Wartungsfrei (keine Batterie oder Akku)

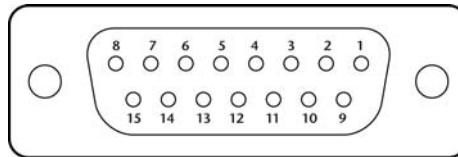
8.2 Steckerbelegung

8.2.1 SubD-Stecker RS232 (700-341-1AH02)



Pin	Bezeichnung		Richtung	Beschreibung
1	DCD	Data Carrier Detect	Eingang	Trägersignal (Modem)
2	RxD	Receive Data	Eingang	Empfangsleitung
3	TxD	Transmit Data	Ausgang	Sendeleitung
4	DTR	Data Terminal Ready	Ausgang	ON = SAS ist betriebsbereit
5	GND	Signal Ground	-	Nullbezugspunkt
6	DSR	Data Set Ready	Eingang	Kommunikationspartner betriebsbereit?
7	RTS	Request to send	Ausgang	ON = SAS sendebereit, OFF = nichts zu senden
8	CTS	Clear to send	Eingang	Kommunikationspartner empfangsbereit?
9	RI	Ring indicator	Eingang	Klingelzeichen (Modem)

8.2.2 SubD-Buchse RS422/RS485 (700-341-1CH02)



Pin	Bezeichnung	Richtung	Beschreibung
1	-	-	
2	T (A)	Ausgang	Sendedaten (Vierdraht-Betrieb)
3	-	-	
4	R (A) / T (A)	Eingang / Ein-/Ausgang	Empfangsdaten (Vierdraht-Betrieb) Empfangs-/Sendedaten (Zweidraht-Betrieb)
5	-	-	
6	-	-	
7	-	-	
8	GND	-	
9	T (B)	Ausgang	Sendedaten (Vierdraht-Betrieb)
10	-	-	
11	R (B) / T (B)	Eingang / Ein-/Ausgang	Empfangsdaten (Vierdraht-Betrieb) Empfangs-/Sendedaten (Zweidraht-Betrieb)
12	-	-	
13	-	-	
14	-	-	
15	-	-	

Notizen