Bonfiglioli
Vectron

Agile

Modbus Kommunikationshandbuch
Frequenzumrichter 230V / 400V

Bonfiglioli
power, control and green solutions

1 ALLGEMEINES ZUR DOKUMENTATION 6

1.1 Anleitungen 6

1.2 Verwendete Piktogramme und Signalworte 7

2 ALLGEMEINE SICHERHEITS- UND ANWENDUNGSHINWEISE 8

2.1 Allgemeine Hinweise 8

2.2 Bestimmungsgemäße Verwendung 8

2.3 Transport und Lagerung 8

2.4 Handhabung und Aufstellung 9

2.5 Elektrische Installation 9

2.6 Betriebshinweise 10
2.6.1 Betrieb mit Fremdprodukten 10

2.7 Wartung und Instandhaltung 10

2.8 Entsorgung 10

3 MÖGLICHKEITEN DER KOMMUNIKATION 11

4 MONTAGE EINES OPTIONALEN KOMMUNIKATIONSMODULS 12

4.1 Montage 12

4.2 Demontage 12

5 SCHNITTSTELLEN RS485 UND RS232 13

5.1 Anschluss X21 14

5.2 Kommunikationsmodule 15
5.2.1 Installationshinweise 15
5.2.2 Kontaktbelegung 16
5.2.3 Busabschluss RS485 (Terminierung) 17

5.3 Inbetriebnahme über das Bedienfeld 19
5.3.1 Menü zur Inbetriebnahme der Kommunikation 19
5.3.2 Das Protokoll wählen 20
5.3.3 Die Parameter für die Kommunikation einstellen 20

5.4 Protokoll zuweisen für Anschluss X21 und Kommunikationsmodul 21

6 MODBUS 22

7 PROTOKOLL 24

7.1 Adressdarstellung 24

7.2 Telegrammaufbau 24

7.3 Unterstützte Funktionscodes 25
7.3.1 Funktionscode 3, 16-Bit oder 32-Bit-Parameter lesen 26
7.3.2 Funktionscode 6, 16-Bit-Parameter schreiben 28
7.3.3 Funktionscode 16, 16-Bit-Parameter schreiben 29
7.3.4 Funktionscode 16, 32-Bit-Parameter schreiben 30
7.3.5 Funktionscode 100, 32-Bit-Parameter lesen 31
7.3.6 Funktionscode 101, 32-Bit-Parameter schreiben 33
7.3.7 Funktionscode 8, Diagnose 34
7.3.8 Ausnahmebedingungsantworten 36
7.3.9 Ausnahmebedingungscodes 36
7.3.10 Modbus-Übertragungsarten 37

7.4 Zeitüberwachungsfunktion 42

8 HANDHABUNG DER DATENSÄTZE/ZYKLISCHES SCHREIBEN 43

9 BEISPIELTELEGRAMME 45

9.1 Modbus RTU-Nachrichten Beispiele 45
9.1.1 16 Bit Zugriff 45
9.1.2 32 Bit Zugriff 48
9.1.3 Funktionscode 8, Diagnose 52

9.2 Modbus ASCII-Nachrichten Beispiele 53
9.2.1 16 Bit Zugriff 53
9.2.2 32 Bit Zugriff 56
9.2.3 Funktionscode 8, Diagnose 60

9.3 ASCII-Tabelle (0x00 – 0x7F) 61

10 STEUERUNG/SOLLWERT 62

10.1 Steuerung über Kontakte/Remote-Kontakte 64
10.1.1 Geräte Statemachine 66

10.2 Steuerung über Statemachine 67
10.2.1 Statemachine Diagramm 69

10.3 Verhalten bei Schnellhalt 71

10.4 Verhalten bei Übergang 5 der Statemachine 72

11 ISTWERTE 72

12 PARAMETERLISTE 73

12.1 Istwerte (Menü „Actual") 73

12.2 Parameter (Menü „Para") 73

13 ANHANG 74

13.1 Warnmeldungen 74

13.2 Warnmeldungen Applikation **75**

13.3 Fehlermeldungen **75**

INDEX **76**

1 Allgemeines zur Dokumentation

Diese Dokumentation beschreibt die Kommunikation über das Protokoll Modbus mit Frequenzumrichtern der Gerätereihe *Agile*. Die modulare Hard- und Softwarestruktur ermöglicht die kundengerechte Anpassung der Frequenzumrichter. Anwendungen, die eine hohe Funktionalität und Dynamik verlangen, sind komfortabel realisierbar.

1.1 Anleitungen

Die Anwenderdokumentation ist zur besseren Übersicht entsprechend den kundenspezifischen Anforderungen an den Frequenzumrichter strukturiert.

Kurzanleitung „Quick Start Guide"
Die Kurzanleitung „Quick Start Guide" beschreibt die grundlegenden Schritte zur mechanischen und elektrischen Installation des Frequenzumrichters. Die geführte Inbetriebnahme unterstützt Sie bei der Auswahl notwendiger Parameter und der Softwarekonfiguration.

Betriebsanleitung
Die Betriebsanleitung dokumentiert die vollständige Funktionalität des Frequenzumrichters. Die für spezielle Anwendungen notwendigen Parameter zur Anpassung an die Applikation und die umfangreichen Zusatzfunktionen sind detailliert beschrieben.

Anwendungshandbuch
Das Anwendungshandbuch ergänzt die Dokumentation zur zielgerichteten Installation und Inbetriebnahme des Frequenzumrichters. Informationen zu verschiedenen Themen im Zusammenhang mit dem Einsatz des Frequenzumrichters werden anwendungsspezifisch beschrieben.

Die Dokumentation und zusätzliche Informationen können über die örtliche Vertretung der Firma BONFIGLIOLI angefordert werden.

Für die Gerätereihe *Agile* sind folgende Anleitungen verfügbar:

Betriebsanleitung *Agile*	Funktionalität des Frequenzumrichters.
Quick Start Guide *Agile*	Installation und Inbetriebnahme. Der Lieferung des Geräts beigefügt.
Anwendungshandbücher Kommunikation	Kommunikation über die Schnittstelle RS485 am Anschluss X21 (RJ45): Anleitungen Modbus und VABus. Kommunikation über die Steuerklemmen X12.5 und X12.6: Anleitungen Systembus und CANopen®[1]. Kommunikation über Kommunikationsmodule: CM-232/CM-485: Anleitungen Modbus und VABus. CM-CAN: Anleitungen Systembus und CANopen®. CM-PDPV1: Anleitung Profibus-DP-V1
Anwendungshandbuch SPS	Logische Verknüpfungen von digitalen Signalen. Funktionen für analoge Signale wie Vergleiche und mathematische Funktionen. Grafische Unterstützung für die Programmierung mit Funktionsbausteinen.
Wartungsanleitung	Für Servicepersonal. Wartungsarbeiten, Überwachung von Wartungsintervallen und Austausch von Lüftern.

Die vorliegende Dokumentation wurde mit größter Sorgfalt erstellt und mehrfach ausgiebig geprüft. Aus Gründen der Übersichtlichkeit konnten nicht sämtliche Detailinformationen zu allen Typen des Produkts und auch nicht jeder denkbare Fall der Aufstellung, des Betriebes oder der Instandhaltung berücksichtigt werden. Sollten Sie weitere Informationen wünschen, oder sollten besondere Probleme auftreten, die in der Dokumentation nicht ausführlich genug behandelt werden, können Sie die erforderliche Auskunft über die Landesvertretung der Firma BONFIGLIOLI anfordern.

[1] Die Produkte für die CANopen®-Kommunikation erfüllen die Spezifikationen der Nutzerorganisation CiA® (CAN in Automation).

Außerdem weisen wir darauf hin, dass der Inhalt dieser Dokumentation nicht Teil einer früheren oder bestehenden Vereinbarung, Zusage oder eines Rechtsverhältnisses ist oder dieses abändern soll. Sämtliche Verpflichtungen des Herstellers ergeben sich aus dem jeweiligen Kaufvertrag, der auch die vollständige und allein gültige Gewährleistungsregelung enthält. Diese vertraglichen Gewährleistungsbestimmungen werden durch die Ausführung dieser Dokumentation weder erweitert noch beschränkt.

Der Hersteller behält sich das Recht vor, Inhalt und Produktangaben sowie Auslassungen in der Betriebsanleitung ohne vorherige Bekanntgabe zu korrigieren, bzw. zu ändern und übernimmt keinerlei Haftung für Schäden, Verletzungen bzw. Aufwendungen, die auf vorgenannte Gründe zurückzuführen sind.

1.2 Verwendete Piktogramme und Signalworte

Folgende Piktogramme und Signalworte werden in der Dokumentation verwendet:

Gefahr!

Gefahr bedeutet unmittelbar drohende Gefährdung. Tod, schwerer Personenschaden und erheblicher Sachschaden werden eintreten, wenn die Vorsichtsmaßnahme nicht getroffen wird.

Warnung!

Warnung kennzeichnet eine mögliche Gefährdung. Tod, schwerer Personenschaden und erheblicher Sachschaden können die Folgen sein, wenn der Hinweistext nicht beachtet wird.

Vorsicht!

Vorsicht weist auf eine unmittelbar drohende Gefährdung hin. Personen oder Sachschaden kann die Folge sein.

Achtung!
Achtung weist auf ein mögliches Betriebsverhalten oder einen unerwünschten Zustand hin, der entsprechend dem Hinweistext auftreten kann.

Hinweis
Hinweis kennzeichnet eine Information die Ihnen die Handhabung erleichtert und ergänzt den entsprechenden Teil der Dokumentation.

2 Allgemeine Sicherheits- und Anwendungshinweise

Warnung!

Bei der Installation und Inbetriebnahme die Hinweise in der Dokumentation beachten. Vor Beginn der Tätigkeit die Dokumentation sorgfältig lesen und die Sicherheitshinweise beachten. Für die Zwecke der Anleitung bezeichnet „qualifizierte Person" eine Person, welche mit der Aufstellung, Montage, Inbetriebsetzung und dem Betrieb der Frequenzumrichter vertraut ist und über die ihrer Tätigkeit entsprechende Qualifikation verfügt.

2.1 Allgemeine Hinweise

Warnung!

Frequenzumrichter führen während des Betriebes ihrer Schutzart entsprechend hohe Spannungen, treiben bewegliche Teile an und besitzen heiße Oberflächen.

Bei unzulässigem Entfernen der erforderlichen Abdeckungen, bei unsachgemäßem Einsatz, bei falscher Installation oder Bedienung, besteht die Gefahr von schweren Personen- oder Sachschäden.

Zur Vermeidung dieser Schäden dürfen nur qualifizierte Personen die Arbeiten zum Transport, zur Installation, Inbetriebnahme, Einstellung und Instandhaltung ausführen. Die Normen DIN EN 50178, IEC 60364 (Cenelec HD 384 oder DIN VDE 0100), IEC 60664-1 (Cenelec HD 625 oder VDE 0110-1), BGV A2 (VBG 4) und nationale Vorschriften beachten. Qualifizierte Personen im Sinne dieser grundsätzlichen Sicherheitshinweise sind Personen, die mit Aufstellung, Montage, Inbetriebsetzung und Betrieb von Frequenzumrichtern und den möglichen Gefahrenquellen vertraut sind sowie über die ihrer Tätigkeit entsprechenden Qualifikationen verfügen.

Nicht mit der Bedienung des Frequenzumrichters vertrauten Personen und Kindern darf der Zugang zum Gerät nicht ermöglicht werden.

2.2 Bestimmungsgemäße Verwendung

Warnung!

Die Frequenzumrichter sind elektrische Antriebskomponenten, die zum Einbau in industrielle Anlagen oder Maschinen bestimmt sind. Die Inbetriebnahme und Aufnahme des bestimmungsgemäßen Betriebs ist solange untersagt, bis festgestellt wurde, dass die Maschine den Bestimmungen der EG-Maschinenrichtlinie 2006/42/EG und DIN EN 60204 entspricht. Gemäß der CE-Kennzeichnung erfüllen die Frequenzumrichter die Anforderungen der Niederspannungsrichtlinie 2006/95/EG und entsprechen der Norm DIN EN 61800-5-1. Die Verantwortung für die Einhaltung der EMV-Richtlinie 2004/108/EG liegt beim Anwender. Frequenzumrichter sind eingeschränkt erhältlich und als Komponenten ausschließlich zur professionellen Verwendung im Sinne der Norm DIN EN 61000-3-2 bestimmt.

Jede anderweitige Verwendung stellt eine Zweckentfremdung dar und kann zum Verlust von Gewährleistungsansprüchen führen.

Mit der Erteilung des UL-Prüfzeichens gemäß UL508c sind auch die Anforderungen des CSA Standards C22.2-No. 14-95 erfüllt.

Die technischen Daten und die Angaben zu Anschluss- und Umgebungsbedingungen müssen dem Typenschild und der Dokumentation entnommen und unbedingt eingehalten werden. Die Anleitung muss vor Arbeiten am Gerät aufmerksam gelesen und verstanden worden sein.

2.3 Transport und Lagerung

Den Transport und die Lagerung sachgemäß in der Originalverpackung durchführen.

Nur in trockenen, staub- und nässegeschützten Räumen, mit geringen Temperaturschwankungen lagern. Die Bedingungen nach DIN EN 60721-3-1 für die Lagerung, DIN EN 60721-3-2 für den Transport und die Kennzeichnung auf der Verpackung beachten.

Die Lagerdauer, ohne Anschluss an die zulässige Nennspannung, darf ein Jahr nicht überschreiten.

2.4 Handhabung und Aufstellung

Warnung!

Beschädigte oder zerstörte Komponenten dürfen nicht in Betrieb genommen werden, da sie Ihre Gesundheit gefährden können.

Den Frequenzumrichter nach der Dokumentation, den Vorschriften und Normen verwenden.

Sorgfältig handhaben und mechanische Überlastung vermeiden.

Keine Bauelemente verbiegen oder Isolationsabstände ändern.

Keine elektronischen Bauelemente und Kontakte berühren. Die Geräte enthalten elektrostatisch gefährdete Bauelemente, die durch unsachgemäße Handhabung beschädigt werden können. Bei Betrieb von beschädigten oder zerstörten Bauelementen ist die Einhaltung angewandter Normen nicht gewährleistet.

Das Entfernen von Plomben am Gehäuse kann die Ansprüche auf Gewährleistung beeinträchtigen.

Warnschilder am Gerät nicht entfernen.

2.5 Elektrische Installation

Warnung!

Vor Montage- und Anschlussarbeiten den Frequenzumrichter spannungslos schalten. Die Spannungsfreiheit prüfen.

Spannungsführende Anschlüsse nicht berühren, da die Kondensatoren aufgeladen sein können.

Die Hinweise in der Betriebsanleitung und die Kennzeichnung des Frequenzumrichters beachten.

Die Sicherheitsregeln für das Arbeiten an elektrischen Anlagen einhalten.

Sicherheitsregeln für das Arbeiten an elektrischen Anlagen:

- Freischalten: Das Anlagenteil muss allpolig und allseitig abgeschaltet werden.
- Gegen Wiedereinschalten sichern. Nur die an der Anlage tätigen Personen dürfen das betreffende Anlagenteil wieder in Betrieb nehmen.
- Spannungsfreiheit feststellen: Durch Messung mit Messgerät oder Spannungsprüfer vergewissern, dass keine Spannung gegen Erde am betreffenden Anlagenteil vorhanden ist.
- Erden und Kurzschließen: Von der Erdungsklemme ausgehend alle Leiter untereinander verbinden.[1)]
- Benachbarte, unter Spannung stehende Teile abdecken und abschranken: Durch Abdecken, Abschranken oder Isolieren von spannungsführenden Anlagenteilen soll verhindert werden, dass diese Teile berührt werden können.

1) Unter bestimmten Bedingungen darf davon abgewichen werden.

Bei Tätigkeiten am Frequenzumrichter die Unfallverhütungsvorschriften, die geltenden Normen BGV A2 (VBG 4), VDE 0100, die Normen zu Arbeiten an Anlagen mit gefährlichen Spannungen (z. B. DIN EN 50178) und andere nationale Vorschriften beachten.

Die Hinweise der Dokumentation zur elektrischen Installation und die einschlägigen Vorschriften beachten.

Die Verantwortung für die Einhaltung und Prüfung der Grenzwerte der EMV-Produktnorm DIN EN 61800-3 drehzahlveränderlicher elektrischer Antriebe liegt beim Hersteller der industriellen Anlage oder Maschine. Die Dokumentation enthält Hinweise für die EMV-gerechte Installation.

Die an den Frequenzumrichter angeschlossenen Leitungen dürfen, ohne vorherige schaltungstechnische Maßnahmen, keiner Isolationsprüfung mit hoher Prüfspannung ausgesetzt werden.

Keine kapazitiven Lasten anschließen.

2.6 Betriebshinweise

Warnung!

Der Frequenzumrichter darf alle 60 s an das Netz geschaltet werden. Dies beim Tippbetrieb eines Netzschützes berücksichtigen. Für die Inbetriebnahme oder nach Not-Aus ist einmaliges direktes Wiedereinschalten zulässig.

Nach einem Ausfall und Wiederanliegen der Versorgungsspannung kann es zum plötzlichen Wiederanlaufen des Motors kommen, wenn die Autostartfunktion aktiviert ist.

Ist eine Gefährdung von Personen möglich, muss eine externe Schaltung installiert werden, die ein Wiederanlaufen verhindert.

Vor der Inbetriebnahme und Aufnahme des bestimmungsgemäßen Betriebs alle Abdeckungen anbringen und die Klemmen überprüfen. Zusätzliche Überwachungs- und Schutzeinrichtungen gemäß DIN EN 60204 und den jeweils gültigen Sicherheitsbestimmungen kontrollieren (z. B. Gesetz über technische Arbeitsmittel, Unfallverhütungsvorschriften usw.).

Während des Betriebes dürfen keine Anschlüsse vorgenommen werden.

2.6.1 Betrieb mit Fremdprodukten

Bitte beachten Sie, dass Bonfiglioli Vectron keine Verantwortung für die Kompatibilität zu Fremdprodukten (z.B. Motoren, Kabel, Filter, usw.) übernimmt.

Um die beste Systemkompatibilität zu ermöglichen, bietet Bonfiglioli Vectron Komponenten, die die Inbetriebnahme vereinfachen und die beste Abstimmung untereinander im Betrieb bieten.

Die Verwendung des Gerätes mit Fremdprodukten erfolgt auf eigenes Risiko.

2.7 Wartung und Instandhaltung

Warnung!

Unbefugtes Öffnen und unsachgemäße Eingriffe können zu Körperverletzung bzw. Sachschäden führen. Reparaturen der Frequenzumrichter dürfen nur vom Hersteller bzw. von ihm autorisierten Personen vorgenommen werden.

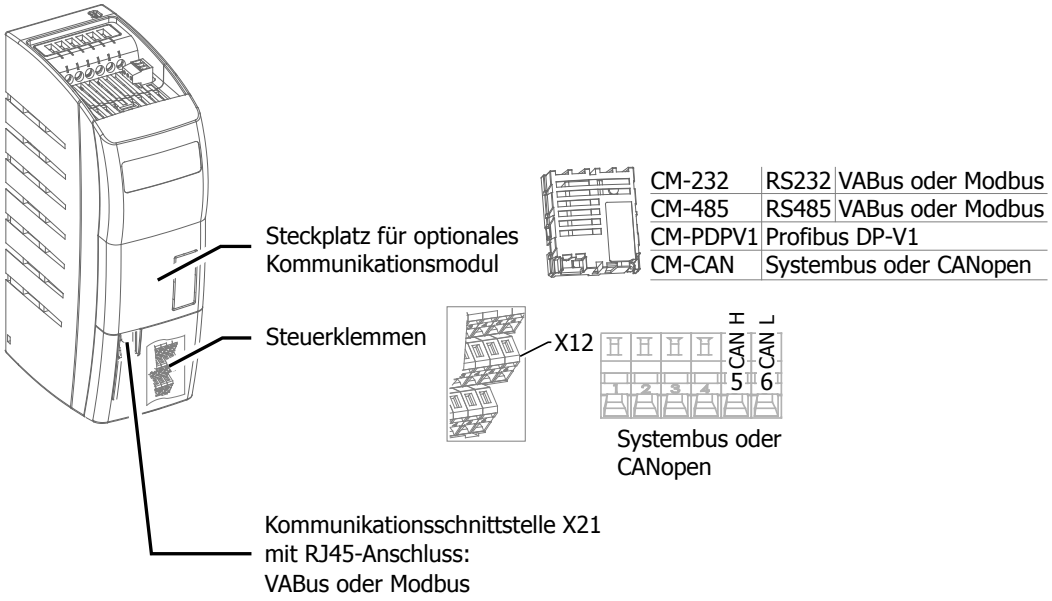
Schutzeinrichtungen regelmäßig überprüfen.

Reparaturen müssen von qualifizierten Elektrofachkräften durchgeführt werden.

2.8 Entsorgung

Die Bauteile des Frequenzumrichters in Übereinstimmung mit den örtlichen und landesspezifischen Gesetzen, Vorschriften und Normen entsorgen.

3 Möglichkeiten der Kommunikation



	Siehe
CM-232	Anleitung zu VABus oder Modbus.
CM-485	Anleitung zu VABus oder Modbus.
CM-PDPV1	Anleitung zu Profibus DP-V1.
CM-CAN	Anleitung zu Systembus oder CANopen.
Steuerklemmen CAN-Anschluss	Anleitung zu Systembus oder CANopen®[1].
Kommunikationsschnittstelle X21[2]	Anleitung zu VABus oder Modbus.

[1] Die Produkte für die CANopen®-Kommunikation erfüllen die Spezifikationen der Nutzerorganisation CiA® (CAN in Automation).

[2] Einen Schnittstellenadapter installieren, um einen PC anzuschließen. Dies ermöglicht die Parametrierung und Überwachung mit der PC-Software VPlus.

4 Montage eines optionalen Kommunikationsmoduls

4.1 Montage

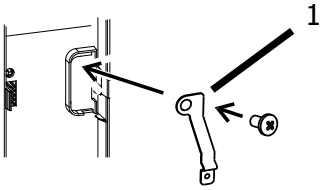
Das Kommunikationsmodul wird für die Montage vormontiert in einem Gehäuse geliefert. Zusätzlich ist für die PE-Anbindung (Schirmung) eine PE-Feder beigelegt.

Vorsicht!
Vor der Montage oder Demontage des Kommunikationsmoduls muss der Frequenzumrichter spannungsfrei geschaltet werden.

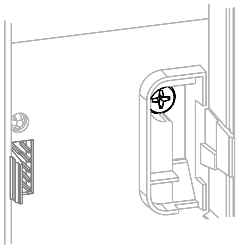
Die Montage unter Spannung ist nicht zulässig und führt zur Zerstörung des Frequenzumrichters und/oder des Kommunikationsmoduls.

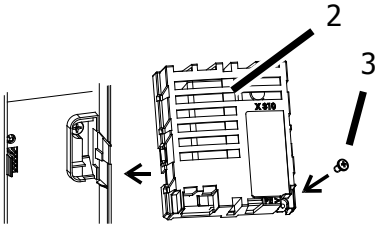
Die auf der Rückseite sichtbare Leiterkarte darf nicht berührt werden, da Bauteile beschädigt werden können.

- Die Abdeckung des Modul-Steckplatzes entfernen.



- Die PE-Feder (1) anschrauben. Die am Frequenzumrichter vorhandene Schraube verwenden.



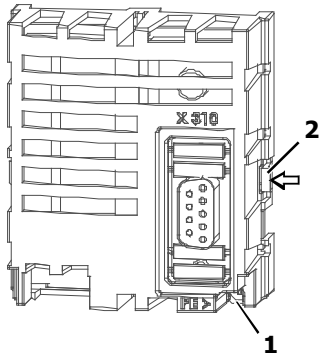


- Das Kommunikationsmodul (2) aufstecken.
- Das Kommunikationsmodul (2) mit der Schraube (3) am Frequenzumrichter anschrauben.

- Den vorgestanzten Durchbruch aus der Abdeckung herausbrechen.
- Die Abdeckung wieder aufsetzen.

4.2 Demontage

- Die Abdeckung des Modul-Steckplatzes entfernen.



- Die Schraube (1) am Kommunikationsmodul lösen.
- Zuerst rechts und dann links die Rasthaken (2) mit einem kleinen Schraubendreher entriegeln.
- Das Kommunikationsmodul vom Steckplatz abziehen.
- Die PE-Feder abschrauben.
- Die Abdeckung am Frequenzumrichter montieren.

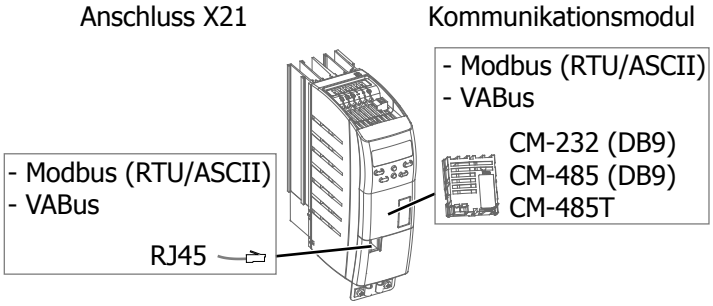
5 Schnittstellen RS485 und RS232

Der Frequenzumrichter kann von einer SPS oder einem anderen Master-Gerät über eine serielle Schnittstelle mit Hilfe des Modbus-Protokolls oder VABus-Protokolls gesteuert werden. Das VABus-Protokoll wird zur Parametrierung über die PC-Bediensoftware VPlus benötigt.

Die Verbindung kann über die RJ45-Steckbuchse des Anschlusses X21 oder ein optionales Kommunikationsmodul hergestellt werden.

Die folgenden Protokolle können gewählt werden:

- Modbus RTU
- Modbus ASCII
- VABus



Die Kommunikation über Modbus-Protokoll erfolgt über:

- die RJ45-Steckbuchse des Anschlusses X21
- das Kommunikationsmodul CM-232 mit RS232-Anschluss über 9-polige D-Sub-Buchse
- das Kommunikationsmodul CM-485 mit RS485-Anschluss über 9-polige D-Sub-Buchse
- das Kommunikationsmodul CM-485T mit RS485-Anschluss über 7-polige Klemmleiste

Die Modbus-Kommunikation ist entweder über die RJ45-Steckbuchse des Anschlusses X21 oder über ein optionales Kommunikationsmodul möglich. Die gleichzeitige Modbus-Kommunikation über den Anschluss X21 und über ein optionales Kommunikationsmodul ist nicht möglich.

Die VABus-Kommunikation ist gleichzeitig über den Anschluss X21 und über ein optionales Kommunikationsmodul möglich.

Kombinationsmöglichkeiten von VABus mit Modbus:

Kommunikationsmodul		**X21 (RJ45)**
Modbus (RTU oder ASCII)	und	VABus
VABus	und	Modbus (RTU oder ASCII)
VABus	und	VABus

Kombinationsmöglichkeiten mit der Scopefunktion:

Kommunikationsmodul		**X21 (RJ45)**
VABus	und	Scopefunktion (VABus)
Scopefunktion (VABus)	und	VABus
Scopefunktion (VABus)	und	Modbus (RTU oder ASCII)
Modbus (RTU oder ASCII)	und	Scopefunktion (VABus)

Die Scopefunktion wird mit Hilfe der Bediensoftware VPlus gestartet. Das gleichzeitige Starten der Scopefunktion über VPlus und über ein optionales Kommunikationsmodul ist nicht möglich.

Für den Anschluss X21 und das Kommunikationsmodul sind die Baudraten separat einstellbar.

Hinweis
Das Dokument ist keine Grundlageninformation zur seriellen Schnittstelle RS232 oder RS485. Grundlegende Kenntnisse über das Modbus-Protokoll und die seriellen Schnittstellen RS232 und RS485 werden vorausgesetzt.

In einigen Abschnitten sind – alternativ zur Bedienung über das Bedienfeld – Einstell- und Anzeigemöglichkeiten mit Hilfe der PC-Bediensoftware VPlus beschrieben. Hierbei kommuniziert VPlus über den Anschluss X21 oder über ein optionales Kommunikationsmodul CM-232 oder CM-485 mit dem Frequenzumrichter.

Ist die serielle Schnittstelle eines optionalen Kommunikationsmoduls CM-232 oder CM-485 z. B. mit einer SPS verbunden, ist ein gleichzeitiger Zugriff der PC-Bediensoftware VPlus auf den Frequenzumrichter über diese Schnittstelle nicht mehr möglich. In diesem Fall kann die Verbindung zum PC über USB mit Hilfe eines optionalen Schnittstellenadapters am Anschluss X21 hergestellt werden.

Warnung!
Über die Modbus-Kommunikation kann eine Steuerung auf sämtliche Parameter des Frequenzumrichters zugreifen.

Eine Veränderung von Parametern, deren Bedeutung dem Anwender unbekannt ist, kann zur Funktionsunfähigkeit des Frequenzumrichters und zu gefährlichen Zuständen in der Anlage führen.

Vorsicht!
Sollen Werte zyklisch mit hoher Wiederholrate geschrieben werden, darf kein Eintrag in das EEPROM erfolgen, da dieses nur eine begrenzte Anzahl zulässiger Schreibzyklen hat (ca. 1 Millionen Zyklen).Wird die Anzahl zulässiger Schreibzyklen überschritten, wird das EEPROM beschädigt. Siehe Kapitel 8 „Handhabung der Datensätze/zyklisches Schreiben".

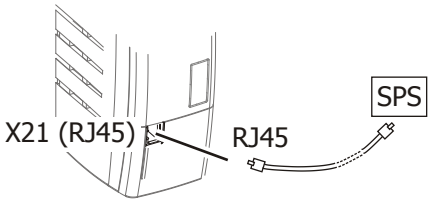
RS485-Verbindung:
Frequenzumrichter können durch Kommunikationsmodule CM-485 zu einem Bussystem verbunden werden. Die Bus-Struktur ist linienförmig und als Zweidrahtleitung ausgeführt. Über einen Bus-Master können über Modbus bis zu 247 Frequenzumrichter adressiert und angesprochen werden.

Die Frequenzumrichter können über das Bussystem parametriert und gesteuert werden. Während des Betriebs können von einem PC oder einer SPS Daten von den Frequenzumrichtern abgefragt und gesetzt werden.

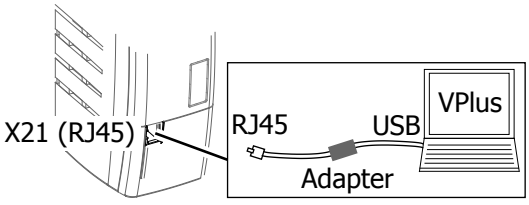
RS232-Verbindung:
Die RS232-Verbindung ermöglicht eine Punkt-zu-Punkt-Verbindung zwischen zwei Teilnehmern.

5.1 Anschluss X21

Der Anschluss X21 mit RJ45-Steckbuchse ermöglicht die Verbindung mit der seriellen Schnittstelle RS485 einer SPS.



Über einen optionalen USB-Adapter kann die USB-Schnittstelle eines PC mit der Kommunikationsschnittstelle X21 verbunden werden. Dies ermöglicht die Parametrierung und Überwachung mit Hilfe der PC-Software VPlus.



5.2 Kommunikationsmodule

Direktverbindung des CM-232 zum PC oder zur SPS

Das Kommunikationsmodul CM-232 ermöglicht die direkte Verbindung zwischen der 9-poligen D-Sub Buchse (X310) des CM-232 und der seriellen Schnittstelle eines PC oder einer SPS.

Die Konfiguration des installierten Kommunikationsmoduls erfolgt über die Bediensoftware VPlus oder über das Bedienfeld.

Beim CM-232 erfolgt die Verbindung zum PC oder zur SPS über eine RS232-Verbindungsleitung (Belegung 1:1).

Direktverbindung des CM-485 zum PC oder zur SPS

Für die direkte Verbindung zwischen der 9-poligen Sub-D Schnittstelle (X310) des CM-485 und der seriellen RS232-Schnittstelle eines PC oder einer SPS einen Schnittstellenumsetzer RS485/RS232 in die Signalleitung installieren.

Die Konfiguration des installierten Kommunikationsmoduls erfolgt über die Bediensoftware VPlus oder das Bedienfeld.

Achtung!
Sender und Empfänger müssen auf die gleiche Übertragungsrate (Baudrate) eingestellt sein.

Die eingestellte Baudrate ist für die Kommunikationsmodule CM-232 und CM-485 wirksam.

Arbeitsschritte:

- Kommunikationsmodul CM-232 bzw. CM-485 an den Frequenzumrichter montieren.
- Bei Kommunikationsmodul CM-232:
  CM-232 und PC über RS232-Kabel verbinden.
- Bei Kommunikationsmodul CM-485:
  CM-485 über RS485-Kabel mit einem Schnittstellenumsetzer RS485/RS232 verbinden.
  RS232-Anschluss des Schnittstellenumsetzers mit PC/SPS verbinden.

Protokolltypeinstellung

Ab Werk sind die Kommunikationsmodule CM-232/CM-485 auf das BONFIGLIOLI VECTRON Standardprotokoll (VABus) eingestellt. Nur mit diesem Protokolltyp ist eine Kommunikation mit der Bediensoftware VPlus möglich.

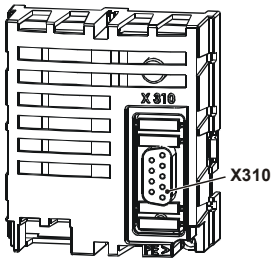
Das Modbus-Protokoll ermöglicht den Betrieb als reines Master/Slave-System. Der Bus-Master ist ein PC, eine SPS oder ein beliebiges Rechnersystem.

5.2.1 Installationshinweise

- Für die RS485 Busleitung eine verdrillte, geschirmte Leitung verwenden.
- Den Schirm als Geflechtschirm ausführen. (kein Folienschirm)
- Den Leitungsschirm an beiden Enden flächig mit PE verbinden.
- Die Steckerbelegungen des RS485-Kabels und des RS232-Kabels sind unterschiedlich. Wird das falsche Kabel verwendet, ist keine Datenübertragung möglich.

5.2.2 Kontaktbelegung

5.2.2.1 RS232 Kommunikationsmodul CM-232 DB9



Der Anschluss der RS232-Schnittstelle an einen PC oder an eine Steuerung erfolgt über die 9-polige D-Sub Buchse X310.

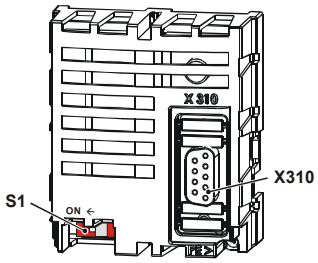
Die Belegung entspricht dem Standard, so dass zum Anschluss lediglich eine RS232-Verbindungsleitung (1:1) notwendig ist.

Busstecker X310 CM-232 (9-polig D-Sub)

Kontakt	Bezeichnung	Funktion
Gehäuse	Schirm	verbunden mit PE
1	–	n. c.
2	RxD	Receive data (input)
3	TxD	Transmit data (output)
4	–	n. c.
5	0 V	Masse
6	–	n. c.
7	–	n. c.
8	–	n. c.
9	–	n. c.

n.c: not connected

5.2.2.2 RS485-Kommunikationsmodul CM-485 DB9



Der Anschluss der RS485-Schnittstelle erfolgt über die 9-polige D-Sub Buchse X310.

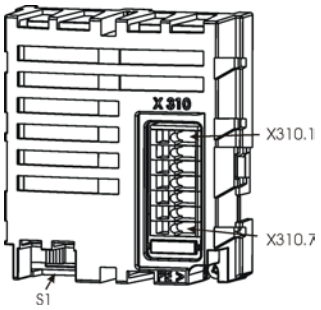
Details zur Belegung der Buchse der folgenden Tabelle entnehmen.

Busstecker X310 CM-485 (9-polig D-Sub)

Kontakt	Bezeichnung	Funktion
Gehäuse	Schirm	verbunden mit PE
1	Datenleitung B	kurzschlussfest und funktionsisoliert; max. Strom 60 mA
2	Datenleitung B’	Brücke von Pin 1 für Kabelschleifen
3	0 V	GND/Masse
4	–	n. c.
5	+5 V	Versorgungsspannung Schnittstellenumsetzer +5 V
6	–	n. c.
7	Datenleitung A	kurzschlussfest und funktionsisoliert; max. Strom 60 mA
8	Datenleitung A’	Brücke von Pin 7 für Kabelschleifen
9	–	n. c.

n.c: not connected

5.2.2.3 RS485-Kommunikationsmodul CM-485T



Der Anschluss der RS485-Schnittstelle erfolgt über die 7-polige Federklemmleiste X310.

Die Belegung der Kontakte der folgenden Tabelle entnehmen.

Busstecker X310 CM-485 T (7-polige Klemmleiste)

Klemme	Name	Funktion
1	Datenleitung A	kurzschlussfest und funktionsisoliert; max. Strom 60
2	Datenleitung A'	Brücke von Pin 1 für Kabelschleifen
3	Datenleitung B	kurzschlussfest und funktionsisoliert; max. Strom 60
4	Datenleitung B'	Brücke von Pin 3 für Kabelschleifen
5	+5 V	Versorgungsspannung Schnittstellenumsetzer +5 V
6	0 V	Masse/GND
7	PE	Schirm

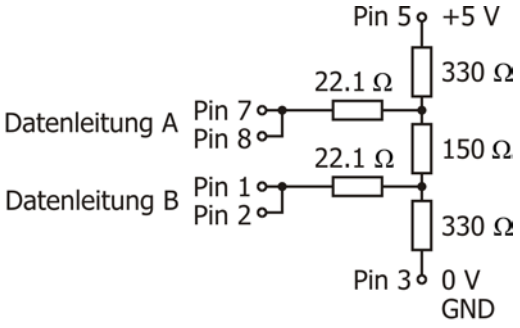
5.2.3 Busabschluss RS485 (Terminierung)

Achtung!
Der physikalisch erste und letzte Teilnehmer muss terminiert, also mit Busabschluss-Widerstand versehen werden. Beim CM-485 und CM-485T kann dazu der DIP-Schalter S1 verwendet werden.

Die Werkseinstellung für den Busabschluss ist OFF (Aus).

Auf eine richtige Terminierung (Busabschluss) achten. Ansonsten ist eine Kommunikationsverbindung über die RS485-Schnittstelle nicht möglich.

Alternativ ist eine aktive Terminierung (Busabschluss) über eine entsprechende externe Schaltung möglich:

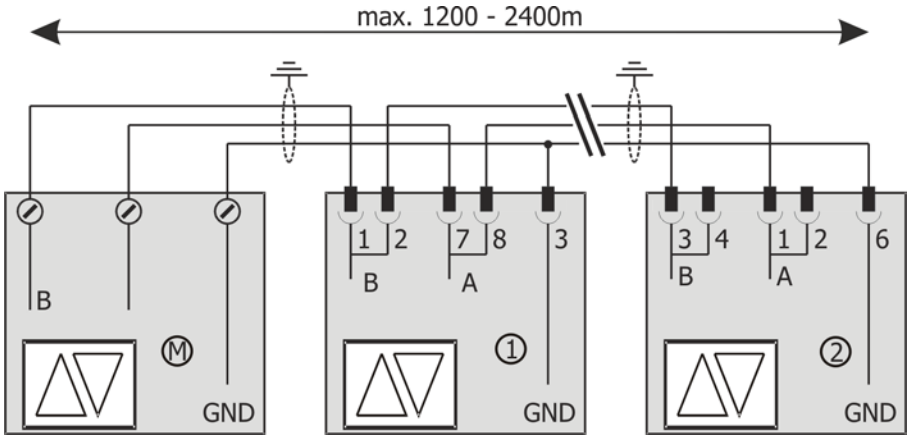


Die aktive Terminierung ist nur einmal je Netzwerk zulässig. Die gleichzeitige Terminierung über eine externe Schaltung und über den DIP-Schalter ist nicht zulässig.

Bei der Verdrahtung auf eine durchgehende GND-Leitung achten. Dies führt in der Praxis zu einem besseren Verhalten gegen Störungen.

Die Klemmen für die Signale A und B sind parallel ausgeführt. Dies erleichtert eine Verdrahtung mehrerer Umrichter.

Beispiel für die Verdrahtung mit verschiedenen CM-485 Baugruppen:



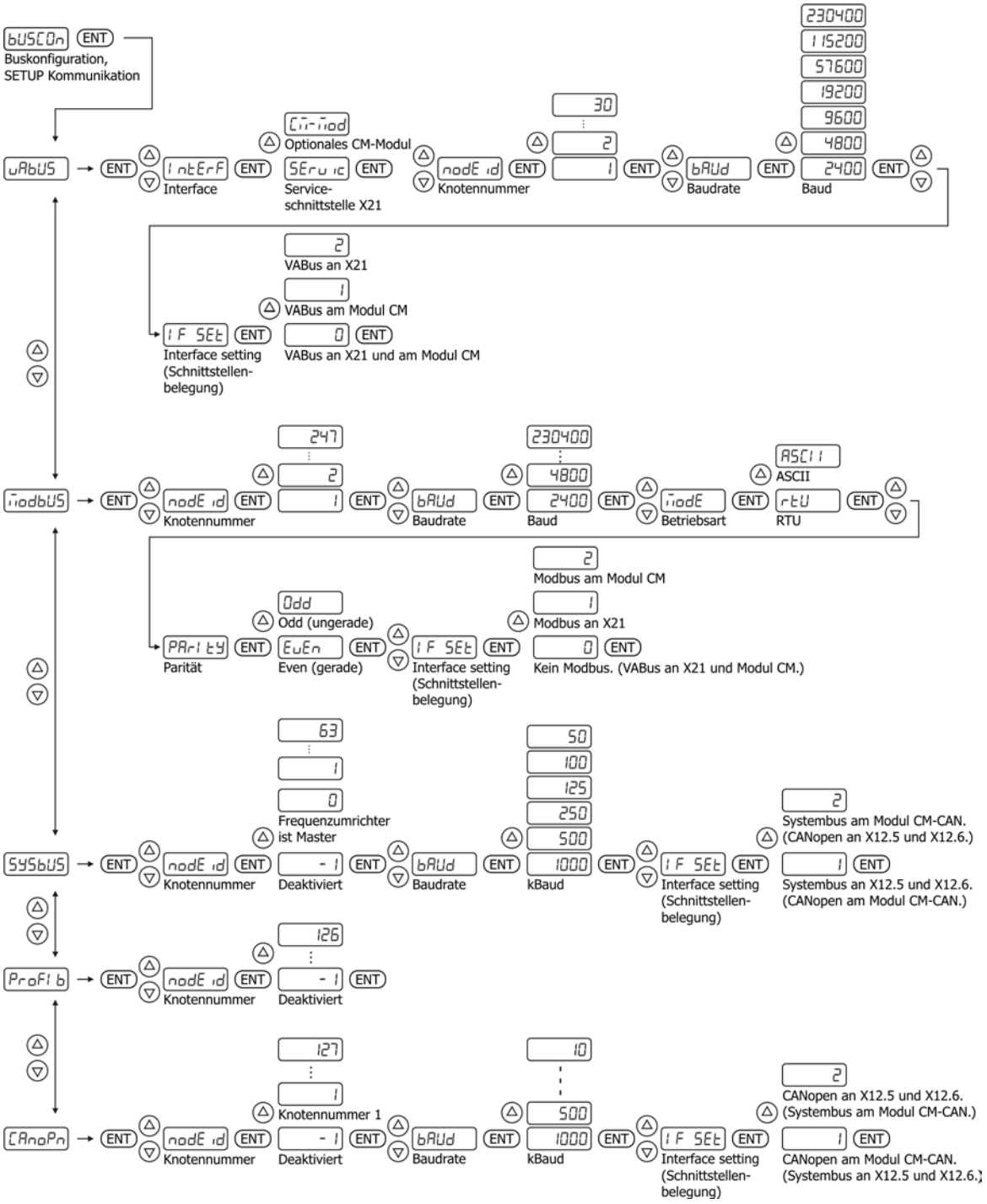
M	Master (i.e. PC)			Terminierung EIN
1	Inverter 1	CM-485 DB9	9-polig D-Sub	Terminierung AUS
2	Inverter 2	CM-485T	7-polige Klemmleiste	Terminierung EIN

5.3 Inbetriebnahme über das Bedienfeld

Eine Kommunikationsschnittstelle kann im Menü „Setup" am Bedienfeld in Betrieb genommen werden. Weitere Parameter zur Kommunikation können im Menü „Para" eingestellt werden.

5.3.1 Menü zur Inbetriebnahme der Kommunikation

Die Kommunikationsschnittstelle kann über das Bedienfeld schnell und einfach in Betrieb genommen werden.



5.3.2 Das Protokoll wählen

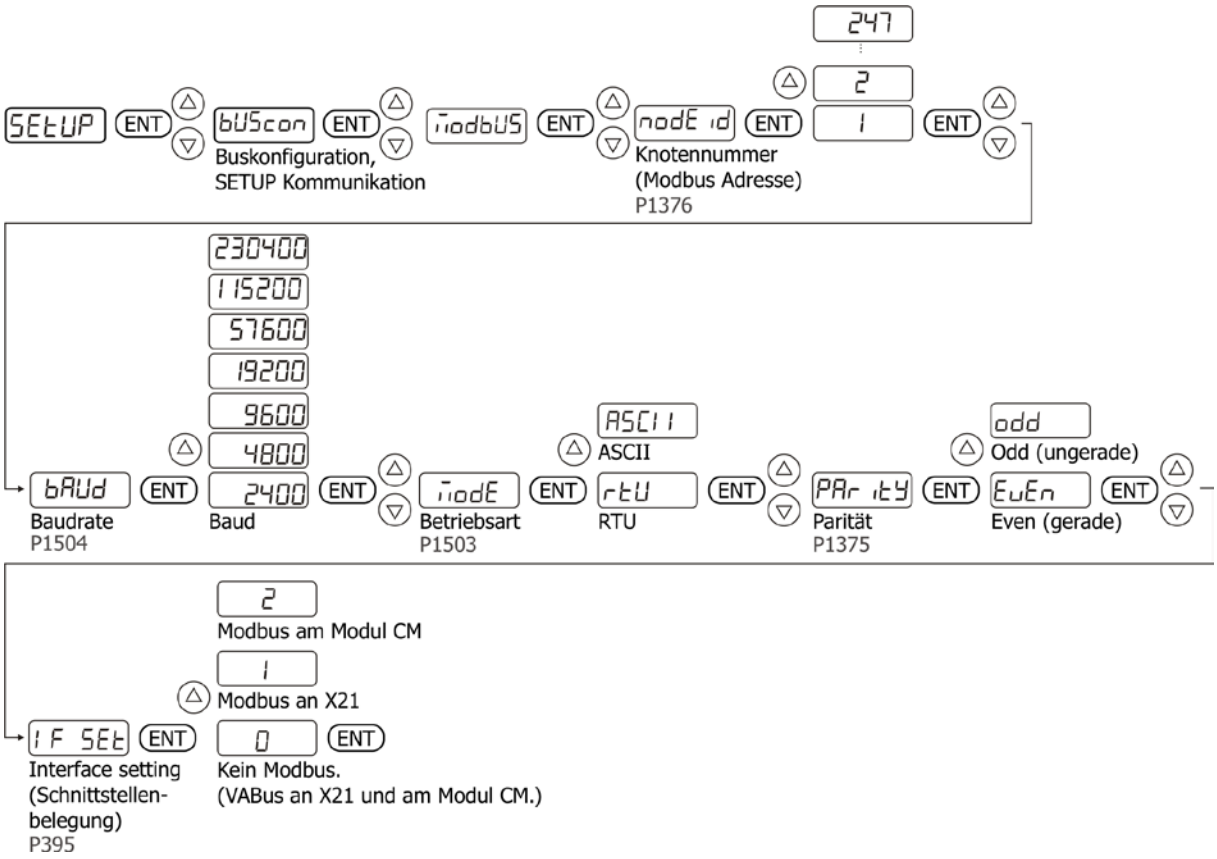
- Modbus wählen.

	Anzeige
Mit den Pfeiltasten das Menü „Setup" wählen.	SEtUP ENT
Mit den Pfeiltasten wählen:	△ ▽
Inbetriebnahme einer Kommunikationsschnittstelle (Buskonfiguration)	bUSCOn ENT
Mit den Pfeiltasten ein Protokoll wählen:	△ ▽
CANopen	CAnoPn
Profibus[1]	Pr0FIb
Systembus	SYSbUS
Modbus	ModbUS
VABus	uAbUS
	ENT

5.3.3 Die Parameter für die Kommunikation einstellen

Parameter		Anzeige
1376	Modbus Adresse (Node-ID)	nodE id
1504	Modbus Baudrate	bAUd
1503	Modbus mode (Betriebsart RTU oder ASCII)	ModE
1375	Modbus Parität	PAr itY
395	Schnittstellenbelegung (Interface setting). Protokoll (CM/X21).	IF SEt
	– Die Service-Schnittstelle X21 auf Modbus einstellen. Oder:	1
	– Ein optionales Kommunikationsmodul CM-232 oder CM-485 auf Modbus einstellen.	2

[1] Die Auswahl ist nur möglich, wenn ein optionales Kommunikationsmodul CM-PDPV1 installiert ist.



5.4 Protokoll zuweisen für Anschluss X21 und Kommunikationsmodul

- **395 Protokoll (CM/X21)**

Über den Parameter *Protokoll (CM/X21)* **395** kann das Kommunikationsprotokoll für ein optionales Kommunikationsmodul (CM) und für den Anschluss X21 gewählt werden.

Protokoll (CM/X21) **395**	**Funktion**
(CM)/X21	
0 - CM: VABus/X21: VABus	Der Steckplatz für ein optionales Kommunikationsmodul und der Anschluss X21 (RJ45) sind auf das Kommunikationsprotokoll VABus eingestellt. Werkseinstellung.
1 - CM: VABus/X21: Modbus	Der Steckplatz für ein optionales Kommunikationsmodul ist auf das Kommunikationsprotokoll VABus eingestellt. Der Anschluss X21 (RJ45) ist auf das Kommunikationsprotokoll Modbus eingestellt.
2 - CM: Modbus/X21: VABus	Der Steckplatz für ein optionales Kommunikationsmodul ist auf das Kommunikationsprotokoll Modbus eingestellt. Der Anschluss X21 (RJ45) ist auf das Kommunikationsprotokoll VABus eingestellt.

6 Modbus

- **1503 Modbus Mode (RTU/ASCII)**

Über den Parameter *Modbus Mode* **1503** kann die Betriebsart der Modbus-Kommunikation gewählt werden. Die Einstellung wird für den Steckplatz eines optionalen Kommunikationsmoduls oder für den Anschluss X21 übernommen (abhängig von der Einstellung des Parameters *Protokoll (CM/X21)* **395**).

Modbus Mode **1503**	Funktion
0- RTU	Modbus über serielle Verbindung mit Betriebsart RTU Übertragung.
1- ASCII	Modbus über serielle Verbindung mit Betriebsart ASCII Übertragung.

Achtung!
Änderungen sind sofort und ohne Neustart des Frequenzumrichters wirksam.
Die Telegramme sind im Kapitel 7.3.9 „Modbus-Übertragungsarten" beschrieben.

- **1504 Modbus Baudrate**

Über den Parameter *Modbus Baudrate* **1504** kann die Übertragungsrate der Modbus-Kommunikation eingestellt werden. Die Einstellung wird für den Steckplatz eines optionalen Kommunikationsmoduls oder für den Anschluss X21 übernommen (abhängig von der Einstellung des Parameters *Protokoll* **395**).

Modbus Baudrate **1504**	Funktion	max. Leitungslänge[1]
1 – 2400 Baud	Übertragungsrate 2400 Baud	30 m
2 – 4800 Baud	Übertragungsrate 4800 Baud	30 m
3 – 9600 Baud	Übertragungsrate 9600 Baud	30 m
4 – 19200 Baud	Übertragungsrate 19200 Baud	30 m
5 – 57600 Baud	Übertragungsrate 57600 Baud	10 m
6 – 115200 Baud	Übertragungsrate 115200 Baud	10 m
7 – 230400 Baud	Übertragungsrate 230400 Baud	10 m

Achtung!
Änderungen sind sofort und ohne Neustart des Frequenzumrichters wirksam.
Alle Busteilnehmer müssen auf die gleiche Baudrate eingestellt sein.

- **1375 Modbus Parität**

Über den Parameter *Modbus Parität* **1375** kann die Parität der Modbus-Kommunikation eingestellt werden. Die Einstellung wird für den Steckplatz eines optionalen Kommunikationsmoduls oder für den Anschluss X21 übernommen (abhängig von der Einstellung des Parameters *Protokoll (CM/X21)* **395**).

Modbus Parität **1375**	Funktion
0 – Gerade	Gerade Parität wird für die Übertragung genutzt.
1 – Ungerade	Ungerade Parität wird für die Übertragung genutzt.
2 – Keine	Keine Parität wird für die Übertragung genutzt.

Achtung!
Änderungen sind sofort und ohne Neustart des Frequenzumrichters wirksam.

[1]Die angegebenen Leitungslängen sind empfohlene Maximalwerte, die u. a. von der Beschaffenheit des Kabels abhängig sind.

- **1376 Modbus Adresse**

Die Knotenadresse kann über den Parameter *Modbus Adresse* **1376** eingestellt werden. Bis zu 247 Frequenzumrichter können am Modbus betrieben werden. Diese erhalten eindeutige Adressen im Bereich von 1 bis 247. Die Einstellung wird für den Steckplatz eines optionalen Kommunikationsmoduls oder für den Anschluss X21 übernommen (abhängig von der Einstellung des Parameters *Protokoll (CM/X21)* **395**).

Parameter		Einstellung		
Nr.	**Beschreibung**	**Min.**	**Max.**	**Werkseinst.**
1376	Modbus Adresse	1	247	1

In den meisten Fällen ist eine Änderung der Werkseinstellung nicht erforderlich (Adresse 1).

Achtung!
Änderungen sind sofort und ohne Neustart des Frequenzumrichters wirksam.

Für jeden Teilnehmer muss eine Adresse vergeben werden.

Bei der Vergabe der Busadressen darf keine Doppelbelegung auftreten.

Details zur Adressierung siehe Kapitel 7.1 „Adressdarstellung".

- **1505 Modbus Watchdog Timer**

Die Kommunikation kann überwacht werden. Fällt die Kommunikation aus, werden keine oder fehlerhafte Daten übertragen. Dieser Zustand wird vom Kommunikations-Watchdog erkannt.

Die Watchdog-Funktion überwacht die Zeit, innerhalb der keine korrekte Kommunikation stattfindet. Diese Zeit ist über den Parameter *Modbus Watchdog Timer* **1505** einstellbar. Der Einstellwert ist die Zeit in Sekunden innerhalb der mindestens ein korrekter Datenaustausch erfolgt sein muss.

Wird die eingestellte Überwachungszeit erreicht, geht der Frequenzumrichter in Störung.

Die Einstellung wird für den Steckplatz eines optionalen Kommunikationsmoduls oder für den Anschluss X21 übernommen (abhängig von der Einstellung des Parameters *Protokoll (CM/X21)* **395**).

Parameter		Einstellung		
Nr.	**Beschreibung**	**Min.**	**Max.**	**Werkseinst.**
1505	Modbus Watchdog Timer	0 s	10000 s	0 s

Wird der Parameter auf Null eingestellt (Werkseinstellung), ist die Überwachungsfunktion ausgeschaltet.

7 Protokoll

Das serielle Übertragungsprotokoll MODBUS ist ein Master/Slave-Protokoll. Mit dem Bus ist nur ein Master (gleichzeitig) verbunden. Ein oder mehrere (maximal 247) Slave-Knoten sind über denselben Bus miteinander verbunden. Eine MODBUS Kommunikation wird immer vom Master initiiert. Die Slave-Knoten kommunizieren nicht miteinander. Der Master initiiert nur eine MODBUS Übertragung gleichzeitig.

Der Master sendet eine MODBUS Anforderung an die Slave-Knoten in zwei verschiedenen Betriebsarten:

- In der **Unicast** Betriebsart adressiert der Master einen einzelnen Slave-Knoten. Nach dem Empfangen und Verarbeiten der Anforderung sendet der Slave-Knoten eine Nachricht (eine Antwort) an den Master. In dieser Betriebsart besteht eine MODBUS Übertragung aus 2 Nachrichten: eine Anforderung vom Master und eine Antwort vom Slave-Knoten. Jeder Slave-Knoten muss eine eindeutige Adresse (von 1 bis 247) haben, so dass die Slave-Knoten unabhängig voneinander Anforderungen erhalten können.
- In der **Broadcast** Betriebsart kann der Master eine Anforderung an alle Slave-Knoten senden. Die Slave-Knoten senden keine Antwort auf die Anforderung. Die Anforderung besteht nur aus Schreibbefehlen. Alle Teilnehmer müssen die Schreibbefehle akzeptieren. Die Adresse 0 ist zur Erkennung einer Broadcast-Übermittlung reserviert.

7.1 Adressdarstellung

Bis zu 247 Frequenzumrichter können am MODBUS betrieben werden. Diese erhalten die Adressen 1 … 247. Über die Adresse 0 können alle angeschlossenen Teilnehmer am Bus gleichzeitig angesprochen werden. Die Adresse 0 wird auch als Broadcast-Adresse bezeichnet.

7.2 Telegrammaufbau

Ein MODBUS Telegramm besteht aus den folgenden Feldern:

Adresse	Funktionscode	Daten	CRC (oder LRC)

Das **Adressfeld** enthält im Modbus Telegramm nur die Slave-Adresse.
Gültige Werte für die für Adressen der Slave-Knoten liegen im Bereich von 0 … 247 dezimal. Die einzelnen Slave-Geräte haben Adressen von 1 … 247. Ein Master adressiert einen Slave, indem die Slave-Adresse im Adressfeld des Telegramms eingetragen wird. Der Slave sendet die Antwort mit der Angabe der eigenen Adresse im Adressfeld des Telegramms, so dass der Master zuordnen kann, welcher Slave geantwortet hat.

Der **Funktionscode** teilt dem Frequenzumrichter mit, welche Aktion ausgeführt werden soll. Dem Funktionscode kann ein Datenfeld folgen, welches Parameter für eine Anforderung oder für die Antwort eines Frequenzumrichters enthält.

Falls beim Empfang einer Anforderung über den Modbus keine Fehler auftreten, enthält das Datenfeld die angeforderten Daten. Falls ein Fehler auftritt, enthält das Feld einen Ausnahmebedingungscode, um dem Master mitzuteilen, dass die Anforderung nicht erfolgreich abgearbeitet wurde. Die Behandlung von Ausnahmebedingungen und die Ausnahmebedingungscodes sind im Kapitel 7.3.8 „Ausnahmebedingungscodes" beschrieben.

Falls eine Anforderung nicht erfolgreich empfangen wurde (Kommunikationsfehler, Prüfsummenfehler), wird keine Antwort gesendet. Der Frequenzumrichter wartet auf die nächste Anforderung.

Das **Fehlerprüffeld** enthält das Ergebnis der Redundanzprüfung (CRC) von Telegrammdaten. Abhängig von der genutzten Betriebsart der Übertragung (RTU oder ASCII) werden verschiedene Berechnungsverfahren genutzt. Siehe Kapitel 7.3.9 „Modbus-Übertragung".

7.3 Unterstützte Funktionscodes

Die Modbus Definitionen für das Schreiben und Lesen von Daten sind nicht direkt mit dem Parameterzugriff eines Frequenzumrichters kompatibel. Modbus ist für das Schreiben und Lesen von Bits ausgelegt und erfasst Daten auf andere Art. Der Datenzugriff ist auf eine Bitbreite von 16 begrenzt.

Um die Anforderungen des Modbus zu erfüllen, ist der Datenzugriff in den Frequenzumrichtern durch die folgenden Funktionscodes festgelegt.

16 Bit-Werte:

- Funktionscode 3, EINE Datenbreite von 16 Bit lesen (Lesen des Halteregisters)
- Funktionscode 6, EINE Datenbreite von 16 Bit schreiben (Schreiben des Einzelregisters)
- Funktionscode 16, EINE Datenbreite von 16 Bit schreiben (Schreiben multipler Register)

32 Bit Werte:

Für den Zugriff auf 32 Bit Daten gibt es zwei weitere an den Frequenzumrichter angepasste Funktionscodes:

- Funktionscode 3, Zwei Datenbreiten von 16 Bit (=32 Bit) lesen (Lesen des Halteregisters)
- Funktionscode 16, Zwei Datenbreiten von 16 Bit (=32 Bit) schreiben (Schreiben multipler Register)
- Funktionscode 100, EINE Bitbreite 32 lesen
- Funktionscode 101, EINE Bitbreite 32 schreiben

Hinweis: Die Modbus Spezifikation beschreibt die Handhabung von 32 Bit Werten nicht. Die implementierten Handhabungen und Funktionscodes sind weitverbreitet und werden häufig verwendet. Diese Funktionen ermöglichen den Datenzugriff auf 32 Bit „Long-" Variablen im Frequenzumrichter.

Für Diagnosezwecke wird der Modbus Funktionscode „8" unterstützt.

Diese Funktionscodes und die entsprechenden Datenfelder sind in den folgenden Kapiteln beschrieben.

Hinweis:

In allen Datenfeldern mit mehr als einem Byte wird zuerst das höchstwertige Byte übertragen.

Im Kapitel 9 „Beispieltelegramme" sind Beispieltelegramme für alle unterstützten Funktionscodes in den Übertragungsarten RTU und ASCII aufgeführt.

7.3.1 Funktionscode 3, 16-Bit oder 32-Bit-Parameter lesen

Dieser Funktionscode wird zum Lesen von 16 Bit oder 32 Bit Werten aus dem Frequenzumrichter verwendet.

Anforderung 16 Bit Parameter lesen:

Start Frame (RTU oder ASCII mode)		
Adresse	1 Byte	1 – 0xF7 (=247)
Funktionscode	1 Byte	0x03
Startadresse (Datensatz / Para-Nr.)	2 Bytes	0x0000 – 0x963F
Registeranzahl	2 Bytes	0x0001
CRC oder LRC & Endframe		

Antwort 16 Bit Parameter lesen:

Start Frame (RTU oder ASCII mode)		
Adresse	1 Byte	1 – 0xF7 (=247)
Funktionscode	1 Byte	0x03
Byteanzahl	1 Byte	0x02
Registerwert (Parameterwert)	2 Bytes	0 – 0xFFFF
CRC oder LRC & Endframe		

Anforderung 32 Bit Parameter lesen:

Start Frame (RTU oder ASCII mode)		
Adresse	1 Byte	1 – 0xF7 (=247)
Funktionscode	1 Byte	0x03
Startadresse (Datensatz / Para-Nr.)	2 Bytes	0x0000 – 0x963F
Registeranzahl	2 Bytes	0x0002
CRC oder LRC & Endframe		

Antwort 32 Bit Parameter lesen:

Start Frame (RTU oder ASCII mode)		
Adresse	1 Byte	1 – 0xF7 (=247)
Funktionscode	1 Byte	0x03
Byteanzahl	1 Byte	0x04
Registerwert (Parameterwert)	4 Bytes	0 – 0xFFFFFFFF
CRC oder LRC & Endframe		

Ausnahmebedingung Antwort:

Start Frame (RTU oder ASCII mode)		
Adresse	1 Byte	1 – 0xF7 (=247)
Fehlercode	1 Byte	0x83
Ausnahmebedingungscode	1 Byte	2, 3 oder 4
CRC oder LRC & Endframe		

Startadresse

Dieses Feld wird genutzt, um die Parameternummer und die Datensatznummer zu speichern. Die Parameternummer liegt im Bereich von 0 – 1599 und wird in den 12 niederwertigen Bits gespeichert. Die Datensatznummer liegt im Bereich von 0 – 9 und wird in den 4 höherwertigen Bits gespeichert.

Zum Beispiel: Parameter 372 (hex. 0x174), Datensatz 2 (hex. 0x2) wird gespeichert als hex. 0x2174.

	Startadresse															
	Datensatz				Parameternummer											
Bits	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Für das obige Beispiel:															
Hex.	0	0	1	0	0	0	0	1	0	1	1	1	0	1	0	0
Bin.	2				1				7				4			

Registeranzahl
Dieses Feld wird genutzt, um die Anzahl der Parameter, die geschrieben werden sollen, zu speichern. Der Wert muss immer 1 sein, da jeweils nur ein Parameter geschrieben werden kann.

Byteanzahl
Dieses Feld wird gesetzt auf

- 2 für 16 Bit Parameter
- 4 für 32 Bit Parameter

Registerwert
Dieses Feld enthält den 16-Bit oder 32-Bit Parameterwert.

Hinweis:
Parameterwerte mit Dezimalstellen werden ohne Dezimalkomma übertragen. Abhängig von der Anzahl der Dezimalstellen werden die Werte mit 10, 100 oder 1000 multipliziert.

Beispiel:
Ein Stromwert von 10,3 A wird übertragen. Der tatsächlich übertragene Zahlenwert ist 103, was einem Hexadezimalwert von 0x67 entspricht.

Ausnahmebedingungscode
Die folgenden Ausnahmebedingungscodes können auftreten:

2	UNGÜLTIGE DATENADRESSE	– Wert des Feldes Registeranzahl ist nicht gleich 1 – Parameter unbekannt
3	UNGÜLTIGER DATENWERT	– Anzahl der Bytes im Datenfeld zu klein oder zu groß
4	FEHLER SLAVE GERÄT	– Fehler beim Parameterlesen

Ausnahmebedingungscodes sind im Kapitel 7.3.8 „Ausnahmebedingungscodes" beschrieben.

Beispiele:

	16 Bit	32 Bit
Modbus RTU	Kapitel 9.1.1.1	Kapitel 9.1.2.1
Modbus ASCII	Kapitel 9.2.1.1	Kapitel 9.2.2.1

7.3.2 Funktionscode 6, 16-Bit-Parameter schreiben

Dieser Funktionscode wird zum Schreiben von Integer oder Unsigned Integer Werten in den Frequenzumrichter verwendet.

Anforderung 16 Bit Parameter schreiben:

Start Frame (RTU oder ASCII mode)		
Adresse	1 Byte	1 – 0xF7 (=247)
Funktionscode	1 Byte	0x06
Startadresse (Datensatz / Para-Nr.)	2 Bytes	0x0000 – 0x963F
Registerwert (Parameterwert)	2 Bytes	0 – 0xFFFF
CRC oder LRC & Endframe		

Antwort:

Start Frame (RTU oder ASCII mode)		
Adresse	1 Byte	1 – 0xF7 (=247)
Funktionscode	1 Byte	0x06
Startadresse (Datensatz / Para-Nr.)	2 Bytes	0x0000 – 0x963F
Registerwert (Parameterwert)	2 Bytes	0 – 0xFFFF
CRC oder LRC & Endframe		

Ausnahmebedingung Antwort:

Start Frame (RTU oder ASCII mode)		
Adresse	1 Byte	1 – 0xF7 (=247)
Fehlercode	1 Byte	0x86
Ausnahmebedingungscode	1 Byte	2, 3 oder 4
CRC oder LRC & Endframe		

Startadresse
Dieses Feld wird genutzt, um die Parameternummer und die Datensatznummer zu speichern. Die Parameternummer liegt im Bereich von 0 – 1599 und wird in den 12 niederwertigen Bits gespeichert. Die Datensatznummer liegt im Bereich von 0 – 9 und wird in den 4 höherwertigen Bits gespeichert.

Zum Beispiel: Parameter 372 (hex. 0x174), Datensatz 2 (hex. 0x2) wird gespeichert als hex. 0x2174.

	Startadresse															
	Datensatz				Parameternummer											
Bits	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Für das obige Beispiel:															
Hex.	0	0	1	0	0	0	0	1	0	1	1	1	0	1	0	0
Bin.	2				1				7				4			

Registerwert
Dieses Feld wird genutzt, um den 16-Bit Parameterwert zu speichern.

Hinweis:
Parameterwerte mit Dezimalstellen werden ohne Dezimalkomma übertragen. Abhängig von der Anzahl der Dezimalstellen werden die Werte mit 10, 100 oder 1000 multipliziert.

Beispiel:
Ein Stromwert von 10,3 A soll übertragen werden. Der tatsächlich übertragene Zahlenwert ist 103, was einem Hexadezimalwert von 0x67 entspricht.

Ausnahmebedingungscode
Die folgenden Ausnahmebedingungscodes können auftreten:

2	UNGÜLTIGE DATENADRESSE	– Parameter unbekannt
3	UNGÜLTIGER DATENWERT	– Anzahl der Bytes im Datenfeld zu klein oder zu groß
4	FEHLER SLAVE GERÄT	– Fehler beim Parameterschreiben

Ausnahmebedingungscodes sind im Kapitel 7.3.8 „Ausnahmebedingungscodes" beschrieben.

Beispiel eines Modbus ASCII-Telegramms siehe Kapitel 9.2.1.2.

Beispiel eines Modbus RTU-Telegramms siehe Kapitel 9.1.1.2.

7.3.3 Funktionscode 16, 16-Bit-Parameter schreiben

Funktionscode 16 kann zum Schreiben von 16 Bit Werten in den Frequenzumrichter verwendet werden.

Anforderung 16 Bit Parameter schreiben:

Start Frame (RTU oder ASCII mode)		
Adresse	1 Byte	1 – 0xF7 (=247)
Funktionscode	1 Byte	0x10
Startadresse (Datensatz / Para-Nr.)	2 Bytes	0x0000 – 0x963F
Registeranzahl	2 Bytes	0x0001
Byteanzahl	1 Byte	0x02
Registerwert (Parameterwert)	2 Bytes	0 – 0xFFFF
CRC oder LRC & Endframe		

Antwort:

Start Frame (RTU oder ASCII mode)		
Adresse	1 Byte	1 – 0xF7 (=247)
Funktionscode	1 Byte	0x10
Startadresse (Datensatz / Para-Nr.)	2 Bytes	0x0000 – 0x963F
Registeranzahl	2 Bytes	0x0001
CRC oder LRC & Endframe		

Ausnahmebedingung Antwort:

Start Frame (RTU oder ASCII mode)		
Adresse	1 Byte	1 – 0xF7 (=247)
Fehlercode	1 Byte	0x90
Ausnahmebedingungscode	1 Byte	2, 3 oder 4
CRC oder LRC & Endframe		

Startadresse
Dieses Feld wird genutzt, um die Parameternummer und die Datensatznummer zu speichern. Die Parameternummer liegt im Bereich von 0 – 1599 und wird in den 12 niederwertigen Bits gespeichert. Die Datensatznummer liegt im Bereich von 0 – 9 und wird in den 4 höherwertigen Bits gespeichert.
Zum Beispiel: Parameter 372 (hex. 0x174), Datensatz 2 (hex. 0x2) wird gespeichert als hex. 0x2174.

	Startadresse															
	Datensatz				Parameternummer											
Bits	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Für das obige Beispiel:															
Hex.	0	0	1	0	0	0	0	1	0	1	1	1	0	1	0	0
Bin.	2				1				7				4			

Registerwert
Dieses Feld wird genutzt, um den 16-Bit Parameterwert zu speichern.

Hinweis:
Parameterwerte mit Dezimalstellen werden ohne Dezimalkomma übertragen. Abhängig von der Anzahl der Dezimalstellen werden die Werte mit 10, 100 oder 1000 multipliziert.

Beispiel:
Ein Stromwert von 10,3 A soll übertragen werden. Der tatsächlich übertragene Zahlenwert ist 103, was einem Hexadezimalwert von 0x67 entspricht.

Ausnahmebedingungscode
Die folgenden Ausnahmebedingungscodes können auftreten:

2	UNGÜLTIGE DATENADRESSE	– Parameter unbekannt
3	UNGÜLTIGER DATENWERT	– Anzahl der Bytes im Datenfeld zu klein oder zu groß
4	FEHLER SLAVE GERÄT	– Fehler beim Parameterschreiben

Ausnahmebedingungscodes sind im Kapitel 7.3.8 „Ausnahmebedingungscodes" beschrieben.

Beispiel eines Modbus ASCII-Telegramms siehe Kapitel 9.2.1.3.

Beispiel eines Modbus RTU-Telegramms siehe Kapitel 9.1.1.3.

7.3.4 Funktionscode 16, 32-Bit-Parameter schreiben

Funktionscode 16 kann zum Schreiben von 32 Bit Werten in den Frequenzumrichter verwendet werden.

Anforderung 32 Bit Parameter schreiben:

Start Frame (RTU oder ASCII mode)		
Adresse	1 Byte	1 – 0xF7 (=247)
Funktionscode	1 Byte	0x10
Startadresse (Datensatz / Para-Nr.)	2 Bytes	0x0000 – 0x963F
Registeranzahl	2 Bytes	0x0002
Byteanzahl	1 Byte	0x04
Registerwert (Parameterwert)	2 Bytes	0 – 0xFFFF FFFF
CRC oder LRC & Endframe		

Antwort:

Start Frame (RTU oder ASCII mode)		
Adresse	1 Byte	1 – 0xF7 (=247)
Funktionscode	1 Byte	0x10
Startadresse (Datensatz / Para-Nr.)	2 Bytes	0x0000 – 0x963F
Registeranzahl	2 Bytes	0x0002
CRC oder LRC & Endframe		

Ausnahmebedingung Antwort:

Start Frame (RTU oder ASCII mode)		
Adresse	1 Byte	1 – 0xF7 (=247)
Fehlercode	1 Byte	0x90
Ausnahmebedingungscode	1 Byte	2, 3 oder 4
CRC oder LRC & Endframe		

Startadresse
Dieses Feld wird genutzt, um die Parameternummer und die Datensatznummer zu speichern. Die Parameternummer liegt im Bereich von 0 – 1599 und wird in den 12 niederwertigen Bits gespeichert. Die Datensatznummer liegt im Bereich von 0 – 9 und wird in den 4 höherwertigen Bits gespeichert.

Zum Beispiel: Parameter 372 (hex. 0x174), Datensatz 2 (hex. 0x2) wird gespeichert als hex. 0x2174.

	Startadresse															
	Datensatz				Parameternummer											
Bits	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Für das obige Beispiel:															
Hex.	0	0	1	0	0	0	0	1	0	1	1	1	0	1	0	0
Bin.	2				1				7				4			

Registerwert
Dieses Feld wird genutzt, um den 32-Bit Parameterwert zu speichern.

Hinweis:
Parameterwerte mit Dezimalstellen werden ohne Dezimalkomma übertragen. Abhängig von der Anzahl der Dezimalstellen werden die Werte mit 10, 100 oder 1000 multipliziert.

Beispiel:
Ein Frequenzwert von 123,45 Hz soll übertragen werden. Der tatsächlich übertragene Zahlenwert ist 12345, was einem Hexadezimalwert von 0x3039 entspricht.

Ausnahmebedingungscode
Die folgenden Ausnahmebedingungscodes können auftreten:

2 UNGÜLTIGE DATENADRESSE – Parameter unbekannt

3 UNGÜLTIGER DATENWERT – Anzahl der Bytes im Datenfeld zu klein oder zu groß

4 FEHLER SLAVE GERÄT – Fehler beim Parameterschreiben

Ausnahmebedingungscodes sind im Kapitel 7.3.8 „Ausnahmebedingungscodes" beschrieben.

Beispiel eines Modbus ASCII-Telegramms siehe Kapitel 9.2.2.2.

Beispiel eines Modbus RTU-Telegramms siehe Kapitel 9.1.2.2.

7.3.5 Funktionscode 100, 32-Bit-Parameter lesen

Anforderung:

Start Frame (RTU oder ASCII mode)		
Adresse	1 Byte	1 – 0xF7 (=247)
Funktionscode	1 Byte	0x64
Startadresse (Datensatz / Para-Nr.)	2 Bytes	0x0000 – 0x963F
CRC oder LRC & Endframe		

Antwort:

Start Frame (RTU oder ASCII mode)		
Adresse	1 Byte	1 – 0xF7 (=247)
Funktionscode	1 Byte	0x64
Registerwert (Parameterwert)	4 Bytes	0 – 0x FFFF FFFF
CRC oder LRC & Endframe		

Ausnahmebedingung Antwort:

Start Frame (RTU oder ASCII mode)		
Adresse	1 Byte	1 – 0xF7 (=247)
Fehlercode	1 Byte	0xE4
Ausnahmebedingungscode	1 Byte	2, 3 oder 4
CRC oder LRC & Endframe		

Startadresse
Dieses Feld wird genutzt, um die Parameternummer und die Datensatznummer zu speichern. Die Parameternummer liegt im Bereich von 0 – 1599 und wird in den 12 niederwertigen Bits gespeichert. Die Datensatznummer liegt im Bereich von 0 – 9 und wird in den 4 höherwertigen Bits gespeichert.

Zum Beispiel: Parameter 372 (hex. 0x174), Datensatz 2 (hex. 0x2) wird gespeichert als hex. 0x2174.

	Startadresse															
	Datensatz				Parameternummer											
Bits	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Für das obige Beispiel:															
Hex.	0	0	1	0	0	0	0	1	0	1	1	1	0	1	0	0
Bin.	2				1				7				4			

Registeranzahl
Dieses Feld wird genutzt, um die 32-Bit Parameterwerte zu speichern.

Hinweis:
Parameterwerte mit Dezimalstellen werden ohne Dezimalkomma übertragen. Abhängig von der Anzahl der Dezimalstellen werden die Werte mit 10, 100 oder 1000 multipliziert.

Beispiel:
Ein Frequenzwert von 100,25 Hz soll übertragen werden. Der tatsächlich übertragene Zahlenwert ist 10025, was einem Hexadezimalwert von 0x2729 entspricht.

Ausnahmebedingungscode
Die folgenden Ausnahmebedingungscodes können auftreten:

- 2 UNGÜLTIGE DATENADRESSE – Parameter unbekannt
- 3 UNGÜLTIGER DATENWERT – Anzahl der Bytes im Datenfeld zu klein oder zu groß
- 4 FEHLER SLAVE GERÄT – Fehler beim Parameterlesen

Ausnahmebedingungscodes sind im Kapitel 7.3.8 „Ausnahmebedingungscodes" beschrieben.

Beispiel eines Modbus ASCII-Telegramms siehe Kapitel 9.2.2.3

Beispiel eines Modbus RTU-Telegramms siehe Kapitel 9.1.2.3

7.3.6 Funktionscode 101, 32-Bit-Parameter schreiben

Anforderung:

Start Frame (RTU oder ASCII mode)		
Adresse	1 Byte	1 – 0xF7 (=247)
Funktionscode	1 Byte	0x65
Startadresse (Datensatz/Para-Nr.)	2 Bytes	0x0000 – 0x963F
Registerwert (Parameterwert)	4 Bytes	0 – 0xFFFF FFFF
CRC oder LRC & Endframe		

Antwort:

Start Frame (RTU oder ASCII mode)		
Adresse	1 Byte	1 – 0xF7 (=247)
Funktionscode	1 Byte	0x65
Startadresse (Datensatz/Para-Nr.)	2 Bytes	0x0000 – 0x963F
Registerwert (Parameterwert)	4 Bytes	0 – 0xFFFF FFFF
CRC oder LRC & Endframe		

Ausnahmebedingung Antwort:

Start Frame (RTU oder ASCII mode)		
Adresse	1 Byte	1 – 0xF7 (=247)
Fehlercode	1 Byte	0xE5
Ausnahmebedingungscode	1 Byte	2, 3 oder 4
CRC oder LRC & Endframe		

Startadresse
Dieses Feld wird genutzt, um die Parameternummer und die Datensatznummer zu speichern. Die Parameternummer liegt im Bereich von 0 – 1599 und wird in den 12 niederwertigen Bits gespeichert. Die Datensatznummer liegt im Bereich von 0 – 9 und wird in den 4 höherwertigen Bits gespeichert.

Zum Beispiel: Parameter 372 (hex. 0x174), Datensatz 2 (hex. 0x2) wird gespeichert als hex. 0x2174.

	Startadresse															
	Datensatz				Parameternummer											
Bits	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Für das obige Beispiel:															
Hex.	0	0	1	0	0	0	0	1	0	1	1	1	0	1	0	0
Bin.	2				1				7				4			

Registerwert
Dieses Feld wird genutzt, um den 32-Bit Parameterwert zu speichern.

Hinweis:
Parameterwerte mit Dezimalstellen werden ohne Dezimalkomma übertragen. Abhängig von der Anzahl der Dezimalstellen werden die Werte mit 10, 100 oder 1000 multipliziert.

Beispiel: Frequenzwert
Ein Frequenzwert von 100,25 Hz soll übertragen werden. Der tatsächlich übertragene Zahlenwert ist 10025, was einem Hexadezimalwert von 0x2729 entspricht.

Ausnahmebedingungscode
Die folgenden Ausnahmebedingungscodes können auftreten:

2	UNGÜLTIGE DATENADRESSE	– Parameter unbekannt
3	UNGÜLTIGER DATENWERT	– Anzahl der Bytes im Datenfeld zu klein oder zu groß
4	FEHLER SLAVE GERÄT	– Fehler beim Parameterlesen

Ausnahmebedingungscodes sind im Kapitel 7.3.8 „Ausnahmebedingungscodes" beschrieben.

Beispiel eines Modbus ASCII-Telegramms siehe Kapitel 9.2.2.4

Beispiel eines Modbus RTU-Telegramms siehe Kapitel 9.1.2.4.

7.3.7 Funktionscode 8, Diagnose

Dieser Funktionscode wird genutzt, um auf den Modbus Diagnosezähler des Frequenzumrichters zuzugreifen. Jeder Zähler kann über einen Unterfunktionscode zusammen mit der Zählernummer erreicht werden. Jeder Zähler kann durch den hexadezimalen Unterfunktionscode 0x0A gelöscht werden.
Die folgenden Unterfunktionscodes werden unterstützt:

Unterfunktion	Name	Beschreibung
0x0A	Alle Zähler löschen	Setzt alle Zähler auf 0
0x0B	Anzahl Busnachrichten zurückgeben	Anzahl der empfangenen gültigen Nachrichten (mit allen Adressen)
0x0C	Anzahl Busübertragungsfehler zurückgeben	Anzahl der Nachrichten mit CRC oder Paritäts-/Blockprüfungs-/Datenverlustfehler
0x0D	Anzahl Bus Ausnahmefehler zurückgeben	Anzahl der gesendeten Ausnahmeantworten
0x0E	Anzahl Slave Nachrichten zurückgeben	Anzahl der empfangenen Nachrichten (mit Slave Adresse)
0x0F	Anzahl „Slave – keine Antwort" Nachrichten zurückgeben	Anzahl der empfangenen Broadcast Nachrichten
0x10	Anzahl Slave NAK (negative Empfangsbestätigung) zurückgeben	Nicht verwendet, Rückgabewert ist immer 0
0x11	Anzahl „Slave beschäftigt" zurückgeben	Nicht verwendet, Rückgabewert ist immer 0
0x12	Anzahl Datenverlustfehler Buszeichen zurückgeben	Anzahl der Nachrichten mit Datenverlustfehlern

Anforderung (Unterfunktion 0x0A, Alle Zähler löschen):

Start Frame (RTU oder ASCII mode)		
Adresse	1 Byte	1 – 0xF7 (=247)
Funktionscode	1 Byte	0x08
Unterfunktion	2 Bytes	0x000A
Daten	2 Bytes	0x0000
CRC oder LRC & Endframe		

Antwort:

Start Frame (RTU oder ASCII mode)		
Adresse	1 Byte	1 – 0xF7 (=247)
Funktionscode	1 Byte	0x08
Unterfunktion	2 Bytes	0x000A
Daten	2 Bytes	0x0000
CRC oder LRC & Endframe		

Ausnahmebedingung Antwort:

Start Frame (RTU oder ASCII mode)		
Adresse	1 Byte	1 – 0xF7 (=247)
Fehlercode	1 Byte	0x88
Ausnahmebedingungscode	1 Byte	1, 3 oder 4
CRC oder LRC & Endframe		

Daten
Dieses Feld ist immer 0x0000.

Ausnahmebedingungscode

1	UNGÜLTIGER FUNKTIONSCODE	– Unterfunktion wird nicht unterstützt
3	UNGÜLTIGER DATENWERT	– Anzahl der Bytes im Datenfeld zu klein oder zu groß – „Datenfeld" nicht gleich 0x0000
4	FEHLER SLAVE GERÄT	– Fehler beim Ausführen der Funktion

Ausnahmebedingungscodes sind im Kapitel 7.3.8 „Ausnahmebedingungscodes" beschrieben.

Anforderung (Unterfunktion 0x0B – 0x12, Zählerwert zurückgeben):

Start Frame (RTU oder ASCII mode)		
Adresse	1 Byte	1 – 0xF7 (=247)
Funktionscode	1 Byte	0x08
Unterfunktion	2 Bytes	0x000B – 0x0012
Daten	2 Bytes	0x0000
CRC oder LRC & Endframe		

Antwort:

Start Frame (RTU oder ASCII mode)		
Adresse	1 Byte	1 – 0xF7 (=247)
Funktionscode	1 Byte	0x08
Unterfunktion	2 Bytes	0x000B – 0x0012
Daten (Zählerwert)	2 Bytes	0 – 0xFFFF
CRC oder LRC & Endframe		

Ausnahmebedingung Antwort:

Start Frame (RTU oder ASCII mode)		
Adresse	1 Byte	1 – 0xF7 (=247)
Fehlercode	1 Byte	0x88
Ausnahmebedingungscode	1 Byte	1, 3 oder 4
CRC oder LRC & Endframe		

Daten
In der Anforderung ist dieses Feld immer auf 0x0000 gesetzt und enthält in der Antwort den aktuellen Zählerwert.

Ausnahmebedingungscode
Die folgenden Ausnahmebedingungscodes können auftreten:

1	UNGÜLTIGER FUNKTIONSCODE	– Unterfunktion wird nicht unterstützt
3	UNGÜLTIGER DATENWERT	– Anzahl der Bytes im Datenfeld zu klein oder zu groß – „Datenfeld" nicht gleich 0x0000
4	FEHLER SLAVE GERÄT	– Fehler beim Lesen des Diagnosezählers

Ausnahmebedingungscodes sind im Kapitel 7.3.8 „Ausnahmebedingungscodes" beschrieben.

7.3.8 Ausnahmebedingungsantworten

Das Mastergerät erwartet eine normale Antwort, wenn es eine Anforderung an den Frequenzumrichter sendet. Auf die Anforderung des Masters kann eine von vier Reaktionen erfolgen:

- Falls der Frequenzumrichter die Anforderung ohne Übertragungsfehler empfängt, kann er diese normal bearbeiten und eine normale Antwort senden.
- Falls der Frequenzumrichter die Anforderung aufgrund eines Übertragungsfehlers nicht empfängt, sendet er keine Antwort. Der Master wird auf die Bedingungen für die Zeitüberwachung der Anforderung prüfen.
- Falls der Frequenzumrichter die Anforderung empfängt und einen Übertragungsfehler feststellt (Parität, LCR, CRC, …), sendet er keine Antwort. Der Master wird auf die Bedingungen für die Zeitüberwachung der Anforderung prüfen.
- Falls der Frequenzumrichter die Anforderung ohne Übertragungsfehler empfängt und diesen nicht bearbeiten kann, zum Beispiel, weil ein unbekannter Parameter gelesen werden soll, sendet er eine Ausnahmeantwort mit einer Information über die Art des Fehlers.

Die Ausnahmebedingungsantwort hat zwei Felder, die sich von der normalen Antwort unterscheiden:

Funktionscodefeld:
In einer normalen Antwort erfolgt eine Rückmeldung des Frequenzumrichters mit dem Funktionscode der ursprünglichen Anforderung. Alle Funktionscodes haben eine 0 als höchstwertiges Bit (most significant bit, MSB); ihre Werte liegen unter dem Hexadezimalwert 0x80. In einer Ausnahmebedingungsantwort setzt der Frequenzumrichter das höchstwertige Bit des Funktionscodes auf den Wert 1. Dies erhöht den Hexadezimalwert des Funktionscodes in einer Ausnahmebedingungsantwort um 0x80 im Vergleich zu dem Wert in einer normalen Antwort. Mit dem Setzen des höchstwertigen Bits im Funktionscode kann der Master die Ausnahmeantwort erkennen und den Ausnahmebedingungscode im Datenfeld untersuchen.

Datenfeld:
In einer normalen Antwort sendet der Frequenzumrichter Daten oder statistische Werte im Datenfeld (jede Information, die angefragt wurde). In einer Ausnahmebedingungsantwort sendet der Frequenzumrichter einen Ausnahmebedingungscode im Datenfeld. Dieser bestimmt die Ursache der Ausnahmebedingung.

Die vom Frequenzumrichter erzeugten Ausnahmebedingungscodes sind im Kapitel 7.3.8 „Ausnahmebedingungscodes" aufgelistet.

7.3.9 Ausnahmebedingungscodes

Der Frequenzumrichter erzeugt die folgenden Ausnahmebedingungscodes:

Code	Modbus Name	Ursachen für die Erzeugung durch den Frequenzumrichter
1	UNGÜLTIGE FUNKTION	– Funktionscode unbekannt – Unterfunktionscode unbekannt (Diagnosefunktion)
2	UNGÜLTIGE DATENADRESSE	– Registeranzahl fehlerhaft (muss immer 0x01 sein) – Unbekannter Parameter oder Datentyp des Parameters ungültig
3	UNGÜLTIGER DATENWERT	– Fehler Blockprüfung – Anzahl der Bytes zu klein oder zu groß – Bestimmte Felder nicht auf typische Werte gesetzt
4	FEHLER SLAVE GERÄT	– Lesen oder Schreiben von Parametern erfolglos Die Ursache des Fehlers kann über Auslesen des Parameters *VA-BusSST Error Register* **11** untersucht werden.

VABusSST Error Register **11**	
Fehlernummer	**Bedeutung**
0	Kein Fehler
1	Unzulässiger Parameterwert
2	Unzulässiger Datensatz
3	Parameter nicht lesbar (nur schreibbar)
4	Parameter nicht schreibbar (nur lesbar)
5	Lesefehler EEPROM
6	Schreibfehler EEPROM
7	Prüfsummenfehler EEPROM
8	Parameter kann nicht geschrieben werden, während der Antrieb läuft
9	Werte der Datensätze unterscheiden sich voneinander
10	Falscher Parametertyp
11	Unbekannter Parameter
12	Prüfsummenfehler im empfangenen Telegramm
13	Syntaxfehler im empfangenen Telegramm
14	Datentyp des Parameters stimmt nicht mit der Anzahl der Bytes im Telegramm überein
15	Unbekannter Fehler

Wenn der Parameter *VABusSST Error Register* **11** ausgelesen wird, wird es automatisch zeitgleich gelöscht.

7.3.10 Modbus-Übertragungsarten

Zwei unterschiedliche Betriebsarten zur seriellen Übertragung sind definiert: Die Betriebsarten **RTU** und **ASCII**. Sie beschreiben den bitweisen Inhalt von Nachrichtenfeldern für die serielle Übertragung. Sie legen fest, wie Informationen in den Nachrichtenfeldern abgelegt und dekodiert werden.

Die Betriebsart (und serielle Schnittstellenparameter) müssen für alle Geräte am Modbus gleich sein.

Über den Parameter *Modbus Mode* **1503** kann die Betriebsart der Modbus-Kommunikation gewählt werden. Siehe Kapitel 6 „Modbus".

7.3.10.1 RTU Übertragung

Wenn Geräte in der Betriebsart RTU (Remote Terminal Unit) über einen Modbus kommunizieren, besteht jedes Byte (8 Bit) in der Nachricht aus zwei 4-Bit Hexadezimalzeichen.

Ein Vorteil dieser Betriebsart ist, dass eine größere Zeichendichte einen höheren Datendurchsatz im Vergleich zur ASCII Betriebsart bei gleicher Baudrate ermöglicht. Beide Nachrichten müssen mit einem kontinuierlichen Zeichenfluss übertragen werden.

7.3.10.1.1 Zeichenformat

Ein Zeichen besteht aus 11 Bits.

- 1 Startbit
- 8 Datenbits, niederwertigstes Bit wird zuerst gesendet
- 1 Paritätsbit
- 1 Stoppbit

Hinweis:
Wird keine Parität genutzt, wird ein zusätzliches Stoppbit hinzugefügt.

Zeichenformat mit Paritätsprüfung:

Start	B1	B2	B3	B4	B5	B6	B7	B8	Parität	Stopp

Zeichenformat ohne Paritätsprüfung:

Start	B1	B2	B3	B4	B5	B6	B7	B8	Stopp	Stopp

7.3.10.1.2 Telegrammbeschreibung

Slave-Adresse	Funktions-code	Daten	CRC (zyklische Blockprüfung)
1 Byte	1 Byte	0 bis 252 Bytes	2 Bytes (CRC-Lo, CRC-Hi)

Die maximale Länge eines Modbus RTU Telegramms ist 256 Bytes.

7.3.10.1.3 Modbus RTU Nachrichtentelegramm

Eine Modbus Nachricht wird von einem sendenden Gerät in ein Telegramm gefügt, das einen festgelegten Anfangs- und Endpunkt hat. Dies ermöglicht empfangenden Geräten den Beginn und das Ende der Nachricht zu erkennen. Teilnachrichten müssen erkannt und als Ergebnis ein Fehler gesetzt werden. In der Betriebsart RTU werden Nachrichtentelegramme durch ein Ruheintervall von mindestens 3,5 Zeichen voneinander getrennt.

MODBUS Nachricht

Start	Adresse	Funktion	Daten	CRC	Ende
>= 3,5 Zeichen	8 Bits	8 Bits	N x 8 Bits	16 Bits	>= 3,5 Zeichen

Das gesamte Nachrichtentelegramm muss als zusammenhängender Zeichenfluss übertragen werden.

Falls ein Ruheintervall von mehr als 1,5 Zeichen zwischen zwei Zeichen auftritt, wird das Nachrichtentelegramm als unvollständig gekennzeichnet und vom Frequenzumrichter verworfen.

7.3.10.1.4 CRC Prüfung

Die Betriebsart RTU enthält ein Fehlerprüffeld, welches auf einer zyklischen Blockprüfung (CRC – Cyclic Redundancy Check) der Nachrichteninhalte basiert.

Das CRC-Feld prüft den Inhalt der gesamten Nachricht. Dies wird ungeachtet einer Paritätsprüfung durchgeführt, welche die einzelnen Zeichen der Nachricht prüft.

Das CRC-Feld enthält einen 16-Bit Wert, ausgeführt als zwei Bytes (je 8 Bit).

Das CRC-Feld wird als letztes Feld an die Nachricht angehängt. Es wird in der Reihenfolge „niederwertiges Byte", „höherwertiges Byte" angefügt. Das CRC höherwertige Byte ist das letzte Byte, das mit der Nachricht gesendet wird.

Der CRC-Wert wird vom sendenden Gerät berechnet und an die Nachricht angehängt. Das empfangende Gerät berechnet während des Empfangs der Nachricht einen CRC-Wert und vergleicht den berechneten Wert mit dem aktuell empfangenen Wert des CRC-Feldes. Sind die beiden Werte nicht gleich, wird ein Fehler ausgelöst.

Die CRC Berechnung beginnt mit dem Anfangsladen eines 16-Bit Registers mit allen Einerkomplementen (0xFFFF). Danach werden nacheinander die Bytes der Nachricht mit dem aktuellen Inhalt des Registers verknüpft.

Während der Erzeugung des CRC wird jedes 8-Bit Zeichen über eine ExOR (Exclusiv-ODER)-Funktion mit dem Registerinhalt verknüpft. Das Ergebnis wird in Richtung des niederwertigsten Bit (LSB – least significant bit) geschoben und eine Null in das höherwertigste Bit (MSB – most significant bit) geschrieben. Das niederwertigste Bit (LSB) wird herausgezogen und geprüft. Falls das LSB eine 1 ist, wird das Register mit dem festen Wert 0xA001 EXOR-verknüpft. Falls das LSB eine 0 ist, erfolgt keine ExOR-Verknüpfung.

Dieser Ablauf wird wiederholt, bis acht Verschiebungen durchgeführt wurden. Nach der letzten (der achten) Verschiebung wird das nächste 8-Bit Zeichen mit dem aktuellen Registerwert EXOR-verknüpft; der Ablauf wird achtmal wiederholt, wie oben beschrieben. Der endgültige Inhalt des Registers nach dem Durchlauf aller Bytes der Nachricht ist der CRC-Wert.

Zuerst wird das niederwertige Byte und dann das höherwertige Byte des CRC-Wertes an die Nachricht angehängt.

Beispiel für die Berechnung von CRC-Werten für das Telegramm: „0x02 0x07"

	MSB			**LSB**	
Start CRC	1111	1111	1111	1111	0xFFFF
1.Zeichen	0000	0000	0000	0010	**0x02**
ExOR					
CRC	1111	1111	1111	1101	
1.Schieben rechts	0111	1111	1111	1110	⇒1 = ExOR
0xA001	1010	0000	0000	0001	
ExOR					
CRC	1101	1111	1111	1111	
2.Schieben rechts	0110	1111	1111	1111	
0xA001	1010	0000	0000	0001	
ExOR					
CRC	1100	1111	1111	1110	
3.Schieben rechts	0110	0111	1111	1111	⇒0 = kein ExOR
4.Schieben rechts	0011	0011	1111	1111	⇒1 = ExOR
0xA001	1010	0000	0000	0001	
ExOR					
CRC	1001	0011	1111	1110	
5.Schieben rechts	0100	1001	1111	1111	⇒0 = kein ExOR
6.Schieben rechts	0010	0100	1111	1111	⇒1 = ExOR
0xA001	1010	0000	0000	0001	
ExOR					
CRC	1000	0100	1111	1110	
7.Schieben rechts	0100	0010	0111	1111	⇒0 = kein ExOR
8.Schieben rechts	0010	0001	0011	1111	⇒1 = ExOR
0xA001	1010	0000	0000	0001	
ExOR					
CRC	1000	0001	0011	1110	
2.Zeichen	0000	0000	0000	0111	**0x07**
ExOR					
CRC	1000	0001	0011	1001	
1.Schieben rechts	0100	0000	1001	1100	⇒1 = ExOR
0xA001	1010	0000	0000	0001	
ExOR					
CRC	1110	0000	1001	1101	
2.Schieben rechts	0111	0000	0100	1110	⇒1 = ExOR
0xA001	1010	0000	0000	0001	
ExOR					
CRC	1101	0000	0100	1111	
3.Schieben rechts	0110	1000	0010	0111	⇒1 = ExOR
0xA001	1010	0000	0000	0001	
ExOR					
CRC	1100	1000	0010	0110	
4.Schieben rechts	0110	0100	0001	0011	⇒0 = kein ExOR
5.Schieben rechts	0011	0010	0000	1001	⇒1 = ExOR
0xA001	1010	0000	0000	0001	
ExOR					
CRC	1001	0010	0000	1000	
6.Schieben rechts	0100	1001	0000	0100	⇒0 = kein ExOR
7.Schieben rechts	0010	0100	1000	0010	⇒0 = kein ExOR
8.Schieben rechts	0001	0010	0100	0001	⇒0 = kein ExOR
Ergebnis	**1**	**2**	**4**	**1**	CRC = 0x41 0x12

Beispiel: Programmfolge (C#)

```
Private int Modbus_CRC(string frame)
{
  int poly     = 0xA001;                              // polynom
  int CRC      = 0xFFFF;                              // start CRC

  for (int i = 0; i < (int)(frame.Length); i++)      // für jedes Zeichen
  {
    CRC ^= Convert.ToInt16(frame[i]);                //  ExOR

    for (int j = 0; j < 8; j++)                      //  acht mal
    {
      if ((CRC& 0x01) == 0x01)                       //  lsb == 1 ?
      {
       CRC>>= 1;                                     //  schieben links
       CRC ^= poly;                                  //  ExOR
      }
      else
      {
        CRC>>= 1;                                    //  schieben links
      }
    }
  }
  Return CRC;
}
```

Aufruf der Funktion für das Telegramm: „0x02 0x07"

```
stringtemp_s = Convert.ToString((char)(0x2))+ Convert.ToString((char)(0x7));

Console.WriteLine(Modbus_CRC(temp_s));
```

Ergebnis: 4673 = 0x1241

CRC niederwertiges Byte (low byte) = 0x41
CRC höherwertiges Byte (high byte) = 0x12

Die Modbus CRC Abfolge ist:

⇒	CRC	Low Byte/High Byte	0x41	0x12

7.3.10.2 ASCII Übertragung

Wenn Geräte für die Datenübertragung über ein Modbus Telegramm in der Betriebsart 7-Bit ASCII (American Standard Code for Information Interchange) eingerichtet werden, wird jedes 8-Bit Zeichen in einer Nachricht als zwei ASCII Zeichen gesendet. Diese Betriebsart wird verwendet, wenn der physikalische Datenübertragungskanal oder die Leistungsfähigkeit des Gerätes nicht den Anforderungen hinsichtlich der zeitlichen Bearbeitung in der Betriebsart RTU entsprechen.

Hinweis:
Diese Betriebsart ist weniger leistungsfähig als RTU, da jedes Byte zwei Zeichen benötigt.

Beispiel: Das Byte 0x5B wird als zwei Zeichen kodiert:
0x35 und 0x42 (0x35 = „5", und 0x42 = „B" in ASCII).

7.3.10.2.1 Zeichenformat

Ein Zeichen besteht aus 10 Bits:

- 1 Startbit
- 7 Datenbits, niederwertigstes Bit wird zuerst gesendet (Wert = 0x00 .. 0x7F)
- 1 Paritätsbit
- 1 Stoppbit

Hinweis:
Wird keine Parität genutzt, wird ein zusätzliches Stoppbit hinzugefügt.

Zeichenformat mit Paritätsprüfung:

Start	B1	B2	B3	B4	B5	B6	B7	Parität	Stopp

Zeichenformat ohne Paritätsprüfung:

Start	B1	B2	B3	B4	B5	B6	B7	Stopp	Stopp

7.3.10.2.2 Modbus ASCII Nachrichtentelegramm

Eine Modbus Nachricht wird von einem sendenden Gerät in ein Telegramm eingefügt, das einen festgelegten Anfangs- und Endpunkt hat. Dies ermöglicht empfangenden Geräten den Beginn und das Ende der Nachricht zu erkennen. Teilnachrichten müssen erkannt und als Ergebnis ein Fehler gesetzt werden.

Das Adressfeld eines Nachrichtentelegramms enthält zwei Zeichen.

In der Betriebsart ASCII ist eine Nachricht durch bestimmte Zeichen für Telegrammbeginn (SoF – Start-of-Frame) und Telegrammende begrenzt. Eine Nachricht muss mit einem einleitenden Doppelpunkt-Zeichen („:" = ASCII 0x3A) beginnen und mit einem Wagenrücklauf/Zeilenvorschub (CRLF)-Zeichenpaar (ASCII 0x0D und 0x0A) enden.

Die zulässigen Zeichen für die zu übertragenden anderen Felder sind hexadezimal 0 - 9, A - F (ASCII kodiert). Die Geräte überwachen den Bus ständig auf das Doppelpunkt-Zeichen. Wird dieses Zeichen empfangen, kodiert jedes Gerät das nächste Zeichen, bis das Telegrammendezeichen (EoF – End-of-Frame) erkannt wird.

Zeitabschnitte von bis zu einer Sekunde dürfen zwischen Zeichen der Nachricht vergehen. Wenn der Anwender keine längere Auszeit einstellt, wird bei einem Zeitabschnitt von mehr als 1 Sekunde ein Fehler ausgelöst.

Ein typisches Nachrichtentelegramm.

SoF	Adresse	Funktion	Daten	LRC	EoF
1 Zeichen :	2 Zeichen	2 Zeichen	0 bis zu 2* 252 Zeichen	2 Zeichen	2 Zeichen CR, LF

7.3.10.2.3 LRC Prüfung

In der Betriebsart ASCII enthalten Nachrichten ein Fehlerprüffeld, das auf einer Längsprüfung (LRC – Longitudinal Redundancy Checking) basiert. Die Berechnung wird mit dem Nachrichteninhalt durchgeführt, außer dem einleitenden Doppelpunkt und dem abschließenden CRLF-Zeichenpaar. Dies wird ungeachtet einer Paritätsprüfung durchgeführt, welche die einzelnen Zeichen der Nachricht prüft.

Das LRC-Feld besteht aus einem Byte (8-Bit Binärwert). Der LRC-Wert wird vom sendenden Gerät berechnet und an die Nachricht angehängt. Das empfangende Gerät berechnet während des Empfangs der Nachricht einen LRC-Wert und vergleicht den berechneten Wert mit dem aktuell empfangenen Wert des LRC-Feldes. Sind die beiden Werte nicht gleich, wird ein Fehler ausgelöst.

Der LRC-Wert wird berechnet, indem nacheinander jedes Byte der Nachricht addiert wird und dann das Zweierkomplement des Ergebnisses gebildet wird. Die Berechnung wird mit dem ASCII Nachrichteninhalt durchgeführt, ausgenommen dem einleitenden Doppelpunkt und dem abschließenden CRLF-Zeichenpaar. In der Betriebsart ASCII wird der erhaltene LRC-Wert in zwei Bytes ASCII kodiert und an das Ende des ASCII Nachrichtentelegramms – vor CRLF – angehängt.

Beispiel:

Feld:	SoF	Adr.		Fkt.Code		Ausn. Code		LRC		EoF	
ASCII	:	0	1	8	3	0	2	7	A	CR	LF
Hex	3A	30	31	38	33	30	32	37	41	0D	0A

LRC relevant

Adr.	0x01 =	1	
Fkt.Code	0x83 =	131	
Ausn.	0x02 =	2	
Sum:		134	LRC
Zweierkompl.		122 = 0x7A	⇒ „7" = 0x37 „A" = 0x41

7.4 Zeitüberwachungsfunktion

Das Modbus Protokoll definiert einen reinen Master/Slave Betrieb. Wird ein Frequenzumrichter vom Bus-Master angesprochen, wird ein weiterer Frequenzumrichter erst dann angesprochen, wenn das Protokoll mit dem ersten Frequenzumrichter vollständig abgewickelt oder die Time-Out-Zeit abgelaufen ist.

Nachdem ein Frequenzumrichter ein Telegramm gesendet hat, muss eine Wartezeit von min. 2 ms eingehalten werden, die der Frequenzumrichter benötigt, um den RS485-Sender auszuschalten. Erst danach darf der Bus-Master ein neues Telegramm senden.

Achtung!
Bei hoher CPU-Auslastung (>90%) kann die Antwortzeit vom Frequenzumrichter größer als 500 ms werden.

Hinweis:
Die angegebenen Zeiten gelten für den RS485 Betrieb und den RS232 Betrieb.

8 Handhabung der Datensätze/zyklisches Schreiben

Der Zugriff auf die Parameterwerte erfolgt anhand der Parameternummer und des gewünschten Datensatzes.

Es existieren Parameter, deren Werte einmal vorhanden sind (Datensatz 0), sowie Parameter, deren Werte viermal vorhanden sind (Datensatz 1 ... 4). Diese werden für die Datensatzumschaltung genutzt.

Werden Parameter, die viermal in den Datensätzen vorhanden sind, mit der Vorgabe Datensatz = 0 beschrieben, werden alle vier Datensätze auf den gleichen übertragenen Wert gesetzt.

Ein Lesezugriff mit Datensatz = 0 auf derartige Parameter gelingt nur dann, wenn alle vier Datensätze auf dem gleichen Wert eingestellt sind. Ist dies nicht der Fall, wird über den Parameter *VABusSST-Error-Register* **11** der Fehler 9 („Werte der Datensätze unterscheiden sich") gemeldet. In diesem Fall muss für den betreffenden Parameter jeder Datensatz separat ausgelesen werden.

Der Eintrag der Werte erfolgt auf dem Controller automatisch in das EEPROM. Sollen Werte zyklisch mit hoher Wiederholrate geschrieben werden, darf kein Eintrag in das EEPROM erfolgen, da dieses nur eine begrenzte Anzahl zulässiger Schreibzyklen besitzt (ca. 1 Millionen Zyklen).

Vorsicht!
Wird die Anzahl zulässiger Schreibzyklen überschritten, wird das EEPROM beschädigt.

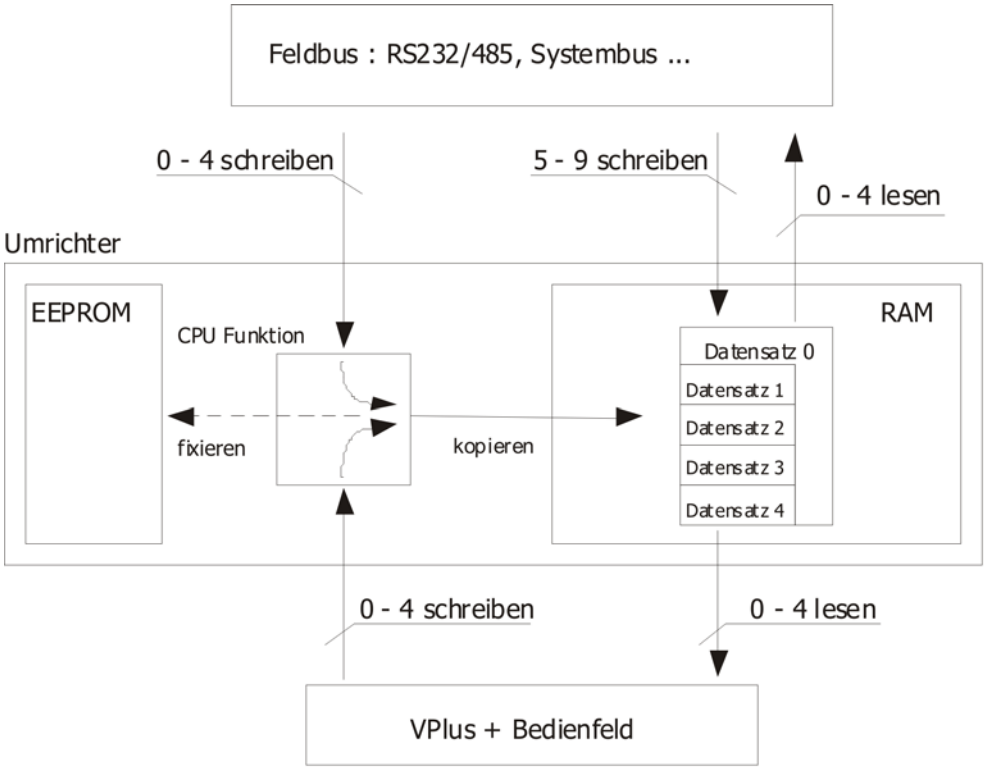
Um die Zerstörung des EEPROM's zu vermeiden, können zyklisch zu schreibende Daten exklusiv in das RAM eingetragen werden, ohne dass ein Schreibzyklus auf das EEPROM erfolgt. In diesem Fall sind die Daten flüchtig gespeichert und gehen nach Abschalten der Versorgungsspannung verloren. Sie müssen nach dem Wiedereinschalten (Netz-Ein) erneut in das RAM geschrieben werden.

Der Schreibvorgang in das RAM wird dadurch aktiviert, dass die Nummer des Zieldatensatzes um fünf erhöht wird.

Zugriff auf die Datensätze des Frequenzumrichters

Parameter	**EEPROM**	**RAM**
Datensatz 0	0	5
Datensatz 1	1	6
Datensatz 2	2	7
Datensatz 3	3	8
Datensatz 4	4	9

Hinweis:
Die Datensätze für die Parameter *Steuerwort* **410**, *Frequenzsollwert RAM* **484** und *Prozentsollwert RAM* **524** sind immer Null. Intern werden diese nicht in das EEPROM geschrieben.


Feldbus : RS232/485, Systembus ...
0 - 4 schreiben
5 - 9 schreiben
0 - 4 lesen
Umrichter
EEPROM
CPU Funktion
RAM
Datensatz 0
Datensatz 1
Datensatz 2
Datensatz 3
Datensatz 4
fixieren
kopieren
0 - 4 schreiben
0 - 4 lesen
VPlus + Bedienfeld

9 Beispieltelegramme

Dieses Kapitel beschreibt einige Beispieltelegramme für Modbus RTU und Modbus ASCII.

9.1 Modbus RTU-Nachrichten Beispiele

9.1.1 16 Bit Zugriff

9.1.1.1 Funktionscode 3, 16-Bit Parameter lesen

Beispiel 1:
Lesen des Parameters *Bemessungsdrehzahl* **372** (0x0174) im Datensatz 2 vom Frequenzumrichter mit der Adresse 1.

Anforderung: Master ➔ Frequenzumrichter

Feld:	Adr.	Funk.	DSatz/Par-Nr.		Registeranzahl		CRC	
Hex	01	03	21	74	00	01	CE	2C

Antwort: Frequenzumrichter ➔ Master

Feld:	Adr.	Funk.	Anz.Bytes	Par-Wert		CRC	
Hex	01	03	02	05	6E	3A	F8

Der gesendete Hexadezimalwert ist 0x056E = Dezimal 1390. Der Parameter *Bemessungsdrehzahl* **372** hat keine Nachkommastelle.
Somit ist die Bemessungsdrehzahl 1390 min^{-1}.

Beispiel 2:
Lesen des Parameters *Bemessungsdrehzahl* **372** (0x0174) im Datensatz 0 vom Frequenzumrichter mit der Adresse 1 und Registeranzahl auf 2 (unzulässiger Wert) gesetzt.

Anforderung: Master ➔ Frequenzumrichter

Feld:	Adr.	Funk.	DSatz/Par-Nr.		Registeranzahl		CRC	
Hex	01	03	01	74	00	02	85	ED

Fehlerantwort: Frequenzumrichter ➔ Master

Feld:	Adr.	Funk.	Ausn.	CRC	
Hex	01	83	04	40	F3

Der gesendete Ausnahmebedingungscode ist der hexadezimale Wert 0x04 = FEHLER SLAVE GERÄT.

9.1.1.2 Funktionscode 6, 16-Bit Parameter schreiben

Beispiel 1:
Schreiben des Parameters *Mech. Bemessungsleistung* **376** (0x0178) in Datensatz 4 des Frequenzumrichters mit der Adresse 3.

Die mechanische Bemessungsleistung soll auf 1,5 kW gesetzt werden. Parameter *Mech. Bemessungsleistung* **376** hat eine Dezimalstelle.

Somit ist der zu sendende Wert 15 = 0x000F.

Anforderung: Master ➔ Frequenzumrichter

Feld:	Adr.	Funk.	DSatz/Par-Nr.		Par-Wert		CRC	
Hex	03	06	41	78	00	0F	5C	09

Antwort: Frequenzumrichter ➔ Master

Feld:	Adr.	Funk.	DSatz/Par-Nr.		Par-Wert		CRC	
Hex	03	06	41	78	00	0F	5C	09

Die Antwort ist das reflektierte Signal der Anforderungsnachricht.

Beispiel 2:
Schreiben des unzulässigen Wertes 0 in den Parameter *Mech. Bemessungsleistung* **376** (0x0178) im Datensatz 2 des Frequenzumrichters mit der Adresse 3.

Anforderung: Master ➔ Frequenzumrichter

Feld:	Adr.	Funk.	DSatz/Par-Nr.		Par-Wert		CRC	
Hex	03	06	21	78	00	00	02	0D

Fehlerantwort: Frequenzumrichter ➔ Master

Feld:	Adr.	Funk.	Ausn.	CRC	
Hex	03	86	04	E2	63

Der gesendete Ausnahmebedingungscode ist der hexadezimale Wert 0x04 = FEHLER SLAVE GERÄT.

9.1.1.3 Funktionscode 16, 16-Bit Parameter schreiben

Beispiel 1:
Schreiben des Parameters *Mech. Bemessungsleistung* **376** (0x0178) in Datensatz 4 des Frequenzumrichters mit der Adresse 1.

Die mechanische Bemessungsleistung soll auf 1,5 kW gesetzt werden. Parameter *Mech. Bemessungsleistung* **376** hat eine Dezimalstelle.

Somit ist der zu sendende Wert 15 = 0x000F.

Anforderung: Master ➔ Frequenzumrichter

Feld:	Adr.	Funk.	DSatz/Par-Nr.		Anzahl Register		Anz. Byte	Par-Wert		CRC	
Hex	01	10	41	78	00	01	02	00	0F	BD	28

Antwort: Frequenzumrichter ➔ Master

Feld:	Adr.	Funk.	DSatz/Par-Nr.		Anzahl Register		CRC	
Hex	01	10	41	78	00	01	95	EC

Die Antwort enthält die Anzahl der geschriebenen Register.

Beispiel 2:
Schreiben des unzulässigen Wertes 0 in den Parameter *Mech. Bemessungsleistung* **376** (0x0178) im Datensatz 2 des Frequenzumrichters mit der Adresse 3.

Anforderung: Master ➔ Frequenzumrichter

Feld:	Adr.	Funk.	DSatz/Par-Nr.		Anzahl Register		Anz. Byte	Par-Wert		CRC	
Hex	03	10	41	78	00	01	02	00	00	E4	4C

Fehlerantwort: Frequenzumrichter ➔ Master

Feld:	Adr.	Funk.	Ausn.	CRC	
Hex	03	90	04	EC	03

Der gesendete Ausnahmebedingungscode ist der hexadezimale Wert 0x04 = FEHLER SLAVE GERÄT.

9.1.2 32 Bit Zugriff

9.1.2.1 Funktionscode 3, 32-Bit Parameter lesen

Beispiel 1:
Lesen des Parameters *Festfrequenz 2* **481** (0x01E1) im Datensatz 1 vom Frequenzumrichter mit der Adresse 1.

Anforderung: Master ➔ Frequenzumrichter

Feld:	Adr.	Funk.	DSatz/Par-Nr.		Registeranzahl		CRC	
Hex	01	03	11	E1	00	02	91	01

Antwort: Frequenzumrichter ➔ Master

Feld:	Adr.	Funk.	Anz.Bytes	Par-Wert				CRC	
Hex	01	03	04	00	00	03	E8	FA	8D

Der gesendete Hexadezimalwert ist 0x03E8 = Dezimal 1000. Der Parameter *Festfrequenz 2* **481** hat 2 Nachkommastellen.
Somit ist die Frequenz 10,00 Hz.

Beispiel 2:
Lesen des Parameters *Festfrequenz 2* **481** (0x01E1) im Datensatz 0 vom Frequenzumrichter mit der Adresse 1 und Registeranzahl auf 1 (unzulässiger Wert) gesetzt.

Anforderung: Master ➔ Frequenzumrichter

Feld:	Adr.	Funk.	DSatz/Par-Nr.		Registeranzahl		CRC	
Hex	01	03	01	E0	00	01	D5	C0

Fehlerantwort: Frequenzumrichter ➔ Master

Feld:	Adr.	Funk.	Ausn.	CRC	
Hex	01	83	04	40	F3

Der gesendete Ausnahmebedingungscode ist der hexadezimale Wert 0x04 = FEHLER SLAVE GERÄT.

9.1.2.2 Funktionscode 16, 32-Bit Parameter schreiben

Beispiel 1:
Schreiben des Parameters *Festfrequenz 3* **482** (0x01E2) in Datensatz 9 (= RAM für Datensatz 4) des Frequenzumrichters mit der Adresse 1.

Die Frequenz soll auf 44,50 Hz gesetzt werden. Parameter *Festfrequenz 3* **482** hat zwei Dezimalstellen.

Somit ist der zu sendende Wert 4450 = 0x00001162.

Anforderung: Master ➔ Frequenzumrichter

Feld:	Adr.	Funk.	DSatz/Par-Nr.		Anzahl Register		Anz. Byte	Par-Wert				CRC	
Hex	01	10	91	E2	00	02	04	00	00	11	62	50	11

Antwort: Frequenzumrichter ➔ Master

Feld:	Adr.	Funk.	DSatz/Par-Nr.		Anzahl Register		CRC	
Hex	01	10	91	E2	00	02	CD	02

Die Antwort enthält die Anzahl der geschriebenen Register.

Beispiel 2:
Schreiben des Parameters *Festfrequenz 3* **482** (0x01E2) in Datensatz 9 (= RAM für Datensatz 4) des Frequenzumrichters mit der Adresse 1.

Die Frequenz soll auf den unzulässigen Wert 2000,00 Hz gesetzt werden. Parameter *Festfrequenz 3* **482** hat zwei Dezimalstellen.

Somit ist der zu sendende Wert 20000 = 0x00030D40.

Anforderung: Master ➔ Frequenzumrichter

Feld:	Adr.	Funk.	DSatz/Par-Nr.		Anzahl Register		Anz. Byte	Par-Wert				CRC	
Hex	01	10	91	E2	00	02	04	00	03	0D	40	28	C8

Fehlerantwort: Frequenzumrichter ➔ Master

Feld:	Adr.	Funk.	Ausn.	CRC	
Hex	03	90	04	4D	C3

Der gesendete Ausnahmebedingungscode ist der hexadezimale Wert 0x04 = FEHLER SLAVE GERÄT.

9.1.2.3 Funktionscode 100, 32-Bit Parameter lesen

Beispiel 1:
Lesen des Parameters *Festfrequenz 2* **481** im Datensatz 0 vom Frequenzumrichter mit der Adresse 1.

Anforderung: Master ➔ Frequenzumrichter

Feld:	Adr.	Funk.	DSatz/Par-Nr.		CRC	
Hex	01	64	01	E1	81	DF

Antwort: Frequenzumrichter ➔ Master

Feld:	Adr.	Funk.	Par-Wert				CRC	
Hex	01	64	00	00	03	E8	70	BC

Der gesendete Hexadezimalwert ist 0x000003E8 = 1000. Der Parameter *Festfrequenz 2* **481** hat zwei Dezimalstellen.
Somit ist die Festfrequenz 2 = 10,00 Hz.

Beispiel 2:
Lesen des unbekannten Parameters **1600** (0x0640) im Datensatz 2 vom Frequenzumrichter mit der Adresse 1.

Anforderung: Master ➔ Frequenzumrichter

Feld:	Adr.	Funk.	DSatz/Par-Nr.		CRC	
Hex	01	64	26	40	5B	97

Fehlerantwort: Frequenzumrichter ➔ Master

Feld:	Adr.	Funk.	Ausn.	CRC	
Hex	01	E4	04	6A	C3

Der Ausnahmebedingungscode ist der Hexadezimalwert 0x04 = FEHLER SLAVE GERÄT.

9.1.2.4 Funktionscode 101, 32-Bit Parameter schreiben

Beispiel 1:
Schreiben des Parameters *Bemessungsfrequenz* **375** (0x0177) in Datensatz 2 des Frequenzumrichters mit der Adresse 1.

Die Bemessungsfrequenz soll auf 10,00 Hz gesetzt werden. Der Parameter *Bemessungsfrequenz* **375** hat zwei Dezimalstellen.
Somit ist der zu sendende Wert 1000 = 0x03E8.

Anforderung: Master ➔ Frequenzumrichter

Feld:	Adr.	Funk.	DSatz/Par-Nr.		Par-Wert				CRC	
Hex	01	65	21	77	00	00	03	E8	46	C5

Antwort: Frequenzumrichter ➔ Master

Feld:	Adr.	Funk.	DSatz/Par-Nr.		Par-Wert				CRC	
Hex	01	65	21	77	00	00	03	E8	46	C5

Die Antwort ist das reflektierte Signal der Anforderungsnachricht.

Beispiel 2:
Schreiben des unzulässigen Wertes 9,00 Hz in den Parameter *Bemessungsfrequenz* **375** im Datensatz 2 des Frequenzumrichters mit der Adresse 1.
Der Parameter *Bemessungsfrequenz* **375** hat 2 Dezimalstellen. Der zu sendende Wert ist 900 = 0x0384.

Anforderung: Master ➔ Frequenzumrichter

Feld:	Adr.	Funk.	DSatz/Par-Nr.		Par-Wert				CRC	
Hex	01	65	21	77	00	00	03	84	46	E8

Fehlerantwort: Frequenzumrichter ➔ Master

Feld:	Adr.	Funk.	Ausn.	CRC	
Hex	01	E5	04	6B	53

Der gesendete Ausnahmebedingungscode ist der hexadezimale Wert 0x04 = FEHLER SLAVE GERÄT.

9.1.3 Funktionscode 8, Diagnose

Beispiel 1a:
Löschen aller Diagnosezähler (Unterfunktion 0x0A) im Frequenzumrichter mit der Adresse 1.

Anforderung: Master ➔ Frequenzumrichter

Feld:	Adr.	Funk.	Unterfunktion		Daten		CRC	
Hex	01	08	00	0A	00	00	C0	09

Antwort: Frequenzumrichter ➔ Master

Feld:	Adr.	Funk.	Unterfunktion		Daten		CRC	
Hex	01	08	00	0A	00	00	C0	09

Die Antwort ist das reflektierte Signal der Anforderungsnachricht. Alle Zähler sind auf Null gesetzt.

Beispiel 1b:
Mit allen Zählern auf Null gesetzt, lesen des Diagnosezählers 4 „Slave Nachrichten Zähler" (Unterfunktion 0x0E) vom Frequenzumrichter mit der Adresse 1.

Anforderung: Master ➔ Frequenzumrichter

Feld:	Adr.	Funk.	Unterfunktion		Daten		CRC	
Hex	01	08	00	0E	00	00	81	C8

Antwort: Frequenzumrichter ➔ Master

Feld:	Adr.	Funk.	Unterfunktion		Daten		CRC	
Hex	01	08	00	0E	00	01	40	08

Der Zählerwert ist 1, da dies die erste empfangene Nachricht nach dem Setzen aller Zähler auf Null ist.

Beispiel 2:
Lesen des unbekannten Diagnosezählers 8 (Unterfunktion 0x13) vom Frequenzumrichter mit der Adresse 1.

Anforderung: Master ➔ Frequenzumrichter

Feld:	Adr.	Funk.	Unterfunktion		Daten		CRC	
Hex	01	08	00	13	00	00	11	CE

Fehlerantwort: Frequenzumrichter ➔ Master

Feld:	Adr.	Funk.	Ausn.	CRC	
Hex	01	88	01	87	C0

Der gesendete Ausnahmebedingungscode ist der Hexadezimalwert 0x01 = UNGÜLTIGER FUNKTIONSCODE.

9.2 Modbus ASCII-Nachrichten Beispiele

9.2.1 16 Bit Zugriff

9.2.1.1 Funktionscode 3, 16-Bit Parameter lesen

Beispiel 1:
Lesen des Parameters *Bemessungsdrehzahl* **372** (0x0174) im Datensatz 2 vom Frequenzumrichter mit der Adresse 1.

Anforderung: Master ➔ Frequenzumrichter

Feld	SoF	Adr.		Fkt.Code		DSatz/Par-Nr.				Registeranzahl				LRC		EoF	
ASC	:	0	1	0	3	2	1	7	4	0	0	0	1	6	6	CR	LF
Hex	3A	30	31	30	33	32	31	37	34	30	30	30	31	36	36	0D	0A

Antwort: Frequenzumrichter ➔ Master

Feld:	SoF	Adr.		Fkt.Code		Anz.Bytes		Par-Wert				LRC		EoF	
ASC	:	0	1	0	3	0	2	0	5	6	E	8	7	CR	LF
Hex	3A	30	31	30	33	30	32	30	35	36	45	38	37	0D	0A

Der gesendete Hexadezimalwert ist 0x056E = 1390. Parameter *Bemessungsdrehzahl* **372** hat keine Nachkommastelle.
Somit ist die Bemessungsdrehzahl 1390 min^{-1}.

Beispiel 2:
Lesen des Parameters *Bemessungsdrehzahl* **372** (0x0174) im Datensatz 0 vom Frequenzumrichter mit der Adresse 1 und Registeranzahl auf 2 (unzulässiger Wert) gesetzt.

Feld:	SoF	Adr.		Fkt.Code		DSatz/Par-Nr.				Registeranzahl				LRC		EoF	
ASC	:	0	1	0	3	0	1	7	4	0	0	0	2	8	5	CR	LF
Hex	3A	30	31	30	33	30	31	37	34	30	30	30	32	38	35	0D	0A

Fehlerantwort: Frequenzumrichter ➔ Master

Feld:	SoF	Adr.		Fkt.Code		Ausn. Code		LRC		EoF	
ASC	:	0	1	8	3	0	4	7	8	CR	LF
Hex	3A	30	31	38	33	30	34	37	38	0D	0A

Der gesendete Ausnahmebedingungscode ist der hexadezimale Wert 0x02 = 0x04 = FEHLER SLAVE GERÄT.

9.2.1.2 Funktionscode 6, 16-Bit Parameter schreiben

Beispiel 1:
Schreiben des Parameters *Mech. Bemessungsleistung* **376** (0x0178) in Datensatz 4 des Frequenzumrichters mit der Adresse 3.

Die mechanische Bemessungsleistung soll auf 1,5 kW gesetzt werden. Parameter *Mech. Bemessungsleistung* **376** hat eine Dezimalstelle.

Somit ist der zu sendende Wert 15 = 0x000F.

Anforderung: Master ➔ Frequenzumrichter

Feld:	SoF	Adr.		Fkt.Code		DSatz/Par-Nr.				Para-Wert				LRC		EoF	
ASC	:	0	3	0	6	4	1	7	8	0	0	0	F	2	F	CR	LF
Hex	3A	30	33	30	36	34	31	37	38	30	30	30	46	32	46	0D	0A

Antwort: Frequenzumrichter ➔ Master

Feld:	SoF	Adr.		Fkt.Code		DSatz/Par-Nr.				Para-Wert				LRC		EoF	
ASC	:	0	3	0	6	4	1	7	8	0	0	0	F	2	F	CR	LF
Hex	3A	30	33	30	36	34	31	37	38	30	30	30	46	32	46	0D	0A

Die Antwort ist das reflektierte Signal der Anforderungsnachricht.

Beispiel 2:
Schreiben des unzulässigen Wertes 0 in den Parameter *Mech. Bemessungsleistung* **376** (0x0178) im Datensatz 2 des Frequenzumrichters mit der Adresse 3.

Anforderung: Master ➔ Frequenzumrichter

Feld:	SoF	Adr.		Fkt.Code		DSatz/Par-Nr.				Para-Wert				LRC		EoF	
ASC	:	0	3	0	6	2	1	7	8	0	0	0	0	5	E	CR	LF
Hex	3A	30	33	30	36	32	31	37	38	30	30	30	30	35	45	0D	0A

Fehlerantwort: Frequenzumrichter ➔ Master

Feld:	SoF	Adr.		Fkt.Code		Ausn. Code		LRC		EoF	
ASC	:	0	3	8	6	0	4	7	3	CR	LF
Hex	3A	30	33	38	36	30	34	37	33	0D	0A

Der gesendete Ausnahmebedingungscode ist der hexadezimale Wert 0x04 = FEHLER SLAVE GERÄT.

9.2.1.3 Funktionscode 16, 16-Bit Parameter schreiben

Beispiel 1:
Schreiben des Parameters *Mech. Bemessungsleistung* **376** (0x0178) in Datensatz 4 des Frequenzumrichters mit der Adresse 1.

Die mechanische Bemessungsleistung soll auf 1,5 kW gesetzt werden. Parameter *Mech. Bemessungsleistung* **376** hat eine Dezimalstelle.

Somit ist der zu sendende Wert 15 = 0x000F.

Anforderung: Master ➔ Frequenzumrichter

Feld:	SoF	Adr.		Fkt.Code		DSatz/Par-Nr.				Anzahl Register				Anz. Byte		Para-Wert				LRC		EoF	
ASC	:	0	1	1	0	4	1	7	8	0	0	0	1	0	2	0	0	0	F	2	4	CR	LF
Hex	3A	30	31	31	30	34	31	37	38	30	30	30	31	30	32	30	30	30	46	32	34	0D	0A

Antwort: Frequenzumrichter ➔ Master

Feld:	SoF	Adr.		Fkt.Code		DSatz/Par-Nr.				Anzahl Register				LRC		EoF	
ASC	:	0	1	1	0	4	1	7	8	0	0	0	1	3	5	CR	LF
Hex	3A	30	31	31	30	34	31	37	38	30	30	30	31	33	35	0D	0A

Beispiel 2:
Schreiben des unzulässigen Wertes 0 in den Parameter *Mech. Bemessungsleistung* **376** (0x0178) im Datensatz 2 des Frequenzumrichters mit der Adresse 3.

Anforderung: Master ➔ Frequenzumrichter

Feld:	SoF	Adr.		Fkt.Code		DSatz/Par-Nr.				Anzahl Register				Anz. Byte		Para-Wert				LRC		EoF	
ASC	:	0	1	1	0	2	1	7	8	0	0	0	1	0	2	0	0	0	0	5	3	CR	LF
Hex	3A	30	33	30	36	32	31	37	38	30	30	30	31	30	32	30	30	30	30	35	33	0D	0A

Fehlerantwort: Frequenzumrichter ➔ Master

Feld:	SoF	Adr.		Funk.		Ausn.		LRC		EoF	
ASC	:	0	1	9	0	0	4	6	B	CR	LF
Hex	3A	30	31	39	30	30	34	36	42	0D	0A

Der gesendete Ausnahmebedingungscode ist der hexadezimale Wert 0x04 = FEHLER SLAVE GERÄT.

9.2.2 32 Bit Zugriff

9.2.2.1 Funktionscode 3, 32-Bit Parameter lesen

Beispiel 1:
Lesen des Parameters *Festfrequenz 2* **481** (0x01E1) im Datensatz 1 vom Frequenzumrichter mit der Adresse 1.

Anforderung: Master ➔ Frequenzumrichter

Feld:	SoF	Adr.		Fkt.Code		DSatz/Par-Nr.				Anzahl Register				LRC		EoF	
ASC	:	0	1	0	3	1	1	E	1	0	0	0	2	0	8	CR	LF
Hex	3A	30	31	30	33	31	31	45	31	30	30	30	32	30	38	0D	0A

Antwort: Frequenzumrichter ➔ Master

Feld:	SoF	Adr.		Fkt.Code		Anz. Bytes		Para-Wert								LRC		EoF	
ASC	:	0	1	0	3	0	4	0	0	0	0	0	3	E	8	0	3	CR	LF
Hex	3A	30	31	31	30	30	34	30	30	30	30	30	33	45	38	30	33	0D	0A

Der gesendete Hexadezimalwert ist 0x03E8 = Dezimal 1000. Der Parameter *Festfrequenz 2* **481** hat 2 Nachkommastellen.
Somit ist die Frequenz 10,00 Hz.

Beispiel 2:
Lesen des Parameters *Festfrequenz 2* **481** (0x01E1) im Datensatz 0 vom Frequenzumrichter mit der Adresse 1 und Registeranzahl auf 1 (unzulässiger Wert) gesetzt.

Anforderung: Master ➔ Frequenzumrichter

Feld:	SoF	Adr.		Fkt.Code		DSatz/Par-Nr.				Anzahl Register				LRC		EoF	
ASC	:	0	1	0	3	0	1	E	0	0	0	0	1	1	A	CR	LF
Hex	3A	30	31	30	33	30	31	45	30	30	30	30	31	31	41	0D	0A

Fehlerantwort: Frequenzumrichter ➔ Master

Field:	SoF	Adr.		Funk.		Ausn.		LRC		EoF	
ASC	:	0	1	8	3	0	4	7	8	CR	LF
Hex	3A	30	31	38	33	30	34	37	38	0D	0A

Der gesendete Ausnahmebedingungscode ist der hexadezimale Wert 0x04 = FEHLER SLAVE GERÄT.

9.2.2.2 Funktionscode 16, 32-Bit Parameter schreiben

Beispiel 1:
Schreiben des Parameters *Festfrequenz 3* **482** (0x01E2) in Datensatz 4 des Frequenzumrichters mit der Adresse 1.

Die Frequenz soll auf 44,50 Hz gesetzt werden. Parameter *Festfrequenz 3* **482** hat zwei Dezimalstellen.

Somit ist der zu sendende Wert 4450 = 0x00001162.

Anforderung: Master ➔ Frequenzumrichter

Feld:	SoF	Adr.		Fkt.Code		DSatz/Par-Nr.				Anzahl Register				Anz. Byte		Para-Wert								LRC		EoF	
ASC	:	0	1	1	0	9	1	E	2	0	0	0	2	0	4	0	0	0	0	1	1	6	2	0	3	CR	LF
Hex	3A	30	31	31	30	39	31	45	32	30	30	30	32	30	34	30	30	30	30	31	31	36	32	30	33	0D	0A

Antwort: Frequenzumrichter ➔ Master

Feld:	SoF	Adr.		Fkt.Code		DSatz/Par-Nr.				Anzahl Register				LRC		EoF	
ASC	:	0	1	1	0	9	1	E	2	0	0	0	2	7	A	CR	LF
Hex	3A	30	31	31	30	39	31	45	32	30	30	30	32	37	41	0D	0A

Beispiel 2:
Schreiben des Parameters *Festfrequenz 3* **482** (0x01E2) in Datensatz 4 des Frequenzumrichters mit der Adresse 1.

Die Frequenz soll auf den unzulässigen Wert 2000,00 Hz gesetzt werden. Parameter *Festfrequenz 3* **482** hat zwei Dezimalstellen.

Somit ist der zu sendende Wert 20000 = 0x00030D40.

Anforderung: Master ➔ Frequenzumrichter

Feld:	SoF	Adr.		Fkt.Code		DSatz/Par-Nr.				Anzahl Register				Anz. Byte		Para-Wert								LRC		EoF	
ASC	:	0	1	1	0	9	1	E	2	0	0	0	2	0	4	0	0	0	3	0	D	4	0	0	3	CR	LF
Hex	3A	30	31	31	30	39	31	45	32	30	30	30	32	30	34	30	30	30	30	31	31	36	32	30	33	0D	0A

Fehlerantwort: Frequenzumrichter ➔ Master

Field:	SoF	Adr.		Funk.		Ausn.		LRC		EoF	
ASC	:	0	1	9	0	0	4	6	B	CR	LF
Hex	3A	30	31	39	30	30	34	36	42	0D	0A

Der gesendete Ausnahmebedingungscode ist der hexadezimale Wert 0x04 = FEHLER SLAVE GERÄT.

9.2.2.3 Funktionscode 100, 32-Bit Parameter lesen

Beispiel 1:
Lesen des Parameters *Festfrequenz 2* **481** im Datensatz 0 vom Frequenzumrichter mit der Adresse 1.

Anforderung: Master ➔ Frequenzumrichter

Feld:	SoF	Adr.		Fkt.Code		DSatz/Par-Nr.				LRC		EoF	
ASC	:	0	1	6	4	0	1	E	1	B	9	CR	LF
Hex	3A	30	31	36	34	30	31	45	31	42	39	0D	0A

Antwort: Frequenzumrichter ➔ Master

Feld:	SoF	Adr.		Fkt.Code		Para-Wert								LRC		EoF	
ASC	:	0	1	6	4	0	0	0	0	0	3	E	8	B	0	CR	LF
Hex	3A	30	31	36	34	30	30	30	30	30	33	45	38	42	30	0D	0A

Der gesendete Hexadezimalwert ist 0x000003E8 = 1000. Der Parameter *Festfrequenz 2* **481** hat zwei Dezimalstellen.
Somit ist die Festfrequenz 2 = 10,00 Hz.

Beispiel 2:
Lesen des unbekannten Parameters **1600** (0x0640) im Datensatz 2 vom Frequenzumrichter mit der Adresse 1.

Anforderung: Master ➔ Frequenzumrichter

Feld:	SoF	Adr.		Fkt.Code		DSatz/Par-Nr.				LRC		EoF	
ASC	:	0	1	6	4	2	6	4	0	3	5	CR	LF
Hex	3A	30	31	36	34	32	36	34	30	33	35	0D	0A

Fehlerantwort: Frequenzumrichter ➔ Master

Feld:	SoF	Adr.		Fkt.Code		Ausn. Code		LRC		EoF	
ASC	:	0	1	E	4	0	2	1	9	CR	LF
Hex	3A	30	31	45	34	30	34	31	37	0D	0A

Der Ausnahmebedingungscode ist der Hexadezimalwert 0x04 = FEHLER SLAVE GERÄT.

9.2.2.4 Funktionscode 101, 32-Bit Parameter schreiben

Beispiel 1:
Schreiben des Parameters *Bemessungsfrequenz* **375** (0x0177) in Datensatz 2 des Frequenzumrichters mit der Adresse 1.

Die Bemessungsfrequenz soll auf 10,00 Hz gesetzt werden. Der Parameter *Bemessungsfrequenz* **375** hat zwei Dezimalstellen.
Somit ist der zu sendende Wert 1000 = 0x03E8.

Anforderung: Master ➔ Frequenzumrichter

Feld	SoF	Adr.		Fkt.Kd		DSatz/Par-Nr.				Para-Wert								LRC		EoF	
ASC	:	0	1	6	5	2	1	7	7	0	0	0	0	0	3	E	8	1	7	CR	LF
Hex	3A	30	31	36	35	32	31	37	37	30	30	30	30	30	33	45	38	31	37	0D	0A

Antwort: Frequenzumrichter ➔ Master

Feld	SoF	Adr.		Fkt.Kd		DSatz/Par-Nr.				Para-Wert								LRC		EoF	
ASC	:	0	1	6	5	2	1	7	7	0	0	0	0	0	3	E	8	1	7	CR	LF
Hex	3A	30	31	36	35	32	31	37	37	30	30	30	30	30	33	45	38	31	37	0D	0A

Die Antwort ist das reflektierte Signal der Anforderungsnachricht.

Beispiel 2:
Schreiben des unzulässigen Wertes 9,00 Hz in den Parameter *Bemessungsfrequenz* **375** im Datensatz 2 des Frequenzumrichters mit der Adresse 1.

Der Parameter *Bemessungsfrequenz* **375** hat 2 Dezimalstellen. Der zu sendende Wert ist 900 = 0x0384

Anforderung: Master ➔ Frequenzumrichter

Feld	SoF	Adr.		Fkt.Kd		DSatz/Par-Nr.				Para-Wert								LRC		EoF	
ASC	:	0	1	6	5	2	1	7	7	0	0	0	0	0	3	8	4	7	B	CR	LF
Hex	3A	30	31	36	35	32	31	37	37	30	30	30	30	30	33	38	34	37	42	0D	0A

Fehlerantwort: Frequenzumrichter ➔ Master

Feld	SoF	Adr.		Fkt.Code		Ausn. Code		LRC		EoF	
ASC	:	0	1	E	5	0	4	1	6	CR	LF
Hex	3A	30	31	45	35	30	34	31	36	0D	0A

Der gesendete Ausnahmebedingungscode ist der hexadezimale Wert 0x04 = FEHLER SLAVE GERÄT.

9.2.3 Funktionscode 8, Diagnose

Beispiel 1a:
Löschen aller Diagnosezähler (Unterfunktion 0x0A) im Frequenzumrichter mit der Adresse 1.

Anforderung: Master ➔ Frequenzumrichter

Feld:	SoF	Adr.		Fkt.Code		Unterfunktion				Daten				LRC		EoF	
ASC	:	0	1	0	8	0	0	0	A	0	0	0	0	E	D	CR	LF
Hex	3A	30	31	30	38	30	30	30	41	30	30	30	30	45	44	0D	0A

Antwort: Frequenzumrichter ➔ Master

Feld:	SoF	Adr.		Fkt.Code		DSatz/ Para-Nr.				Registeranzahl				LRC		EoF	
ASC	:	0	1	0	8	0	0	0	A	0	0	0	0	E	D	CR	LF
Hex	3A	30	31	30	38	30	30	30	41	30	30	30	30	45	44	0D	0A

Die Antwort ist das reflektierte Signal der Anforderungsnachricht. Alle Zähler sind auf Null gesetzt.

Beispiel 1b:
Mit allen Zählern auf Null gesetzt, lesen des Diagnosezählers 4 „Slave Nachrichten Zähler" (Unterfunktion 0x0E) vom Frequenzumrichter mit der Adresse 1.

Feld:	SoF	Adr.		Fkt.Code		Unterfunktion				Daten				LRC		EoF	
ASC	:	0	1	0	8	0	0	0	E	0	0	0	0	E	9	CR	LF
Hex	3A	30	31	30	38	30	30	30	45	30	30	30	30	45	39	0D	0A

Antwort: Frequenzumrichter ➔ Master

Feld:	SoF	Adr.		Fkt.Code		Unterfunktion				Daten				LRC		EoF	
ASC	:	0	1	0	8	0	0	0	E	0	0	0	1	E	8	CR	LF
Hex	3A	30	31	30	38	30	30	30	45	30	30	30	31	45	38	0D	0A

Beispiel 2:
Lesen des unbekannten Diagnosezählers 8 (Unterfunktion 0x13) vom Frequenzumrichter mit der Adresse 1.

Anforderung: Master ➔ Frequenzumrichter

Feld:	SoF	Adr.		Fkt.Code		Unterfunktion				Daten				LRC		EoF	
ASC	:	0	1	0	8	0	0	1	3	0	0	0	0	E	4	CR	LF
Hex	3A	30	31	30	38	30	30	31	33	30	30	30	30	45	34	0D	0A

Fehlerantwort: Frequenzumrichter ➔ Master

Feld:	SoF	Adr.		Fkt.Code		Ausn. Code		LRC		EoF	
ASC	:	0	1	8	8	0	1	7	6	CR	LF
Hex	3A	30	31	38	38	30	31	37	36	0D	0A

Der gesendete Ausnahmebedingungscode ist der Hexadezimalwert 0x01 = UNGÜLTIGER FUNKTIONSCODE.

9.3 ASCII-Tabelle (0x00 – 0x7F)

Dez.	Hex.	Char.	Dez.	Hex.	Char.	Dez.	Hex.	Char.
0	00	NUL	43	2B	+	86	56	V
1	01		44	2C	,	87	57	W
2	02	STX	45	2D	-	88	58	X
3	03	ETX	46	2E	.	89	59	Y
4	04	EOT	47	2F	/	90	5A	Z
5	05	ENQ	48	30	0	91	5B	[
6	06	ACK	49	31	1	92	5C	\
7	07	BEL	50	32	2	93	5D	]
8	08	BS	51	33	3	94	5E	^
9	09	TAB	52	34	4	95	5F	_
10	0A	LF	53	35	5	96	60	`
11	0B	VT	54	36	6	97	61	a
12	0C	FF	55	37	7	98	62	b
13	0D	CR	56	38	8	99	63	c
14	0E		57	39	9	100	64	d
15	0F		58	3A	:	101	65	e
16	10		59	3B	;	102	66	f
17	11		60	3C	<	103	67	g
18	12		61	3D	=	104	68	h
19	13		62	3E	>	105	69	i
20	14		63	3F	?	106	6A	j
21	15	NAK	64	40	@	107	6B	k
22	16		65	41	A	108	6C	l
23	17		66	42	B	109	6D	m
24	18		67	43	C	110	6E	n
25	19		68	44	D	111	6F	o
26	1A		69	45	E	112	70	p
27	1B	ESC	70	46	F	113	71	q
28	1C		71	47	G	114	72	r
29	1D		72	48	H	115	73	s
20	1E		73	49	I	116	74	t
31	1F		74	4A	J	117	75	u
32	20	SPACE	75	4B	K	118	76	v
33	21	!	76	4C	L	119	77	w
34	22	„	77	4D	M	120	78	x
35	23	#	78	4E	N	121	79	y
36	24	$	79	4F	O	122	7A	z
37	25	%	80	50	P	123	7B	{
38	26	&	81	51	Q	124	7C	|
39	27	'	82	52	R	125	7D	}
40	28	(	83	53	S	126	7E	~
41	29	)	84	54	T	127	7F	DEL
42	2A	*	85	55	U			

Häufig genutzte Werte sind grau markiert.

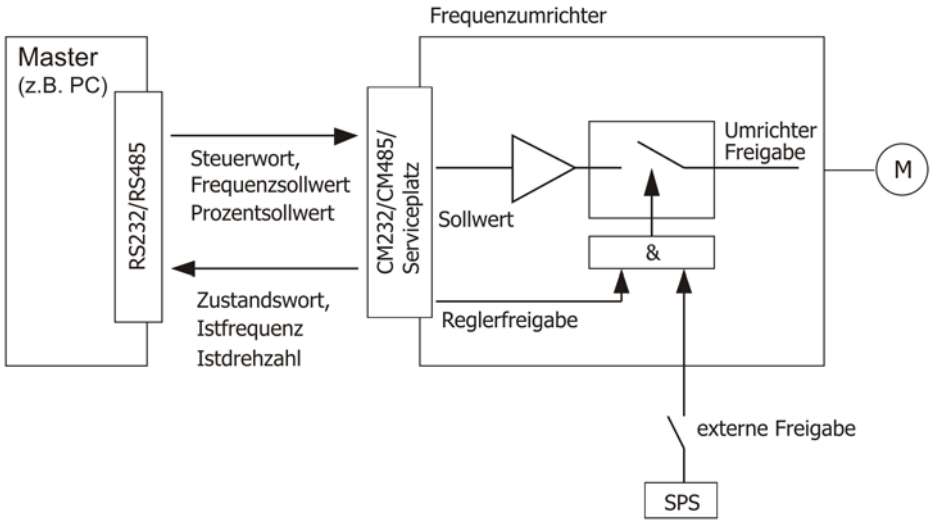
10 Steuerung/Sollwert

- **410 Steuerwort**
- **411 Zustandswort**
- **484 Frequenzsollwert RAM [Hz]**
- **524 Prozentsollwert RAM [%]**

Der Frequenzumrichter kann vollständig über die serielle Schnittstelle gesteuert werden. Dazu können folgende Parameter und Istwerte verwendet werden:

Parameter		Einstellung			
Nr.	**Name/Bedeutung**	**Min.**	**Max.**	**Werkseinst.**	**Typ**
410	Steuerwort	0x0000	0xFFFF	-	uInt
411	Zustandswort	0x0000	0xFFFF	-	uInt
484	Frequenzsollwert RAM [Hz]	-999,99	999,99	0,00	Long
524	Prozentsollwert RAM [%]	-300,00	300,00	0,00	Long

Mit dem *Steuerwort* **410** (Datentyp uInt) werden Steuerkommandos an den Frequenzumrichter gesendet und mit dem *Frequenzsollwert RAM* **484** (Datentyp Long [Hz]), bzw. *Prozentsollwert RAM* **524** (Datentyp Long [%]) der Liniensollwert. Über das *Zustandswort* **411** (Datentyp uInt) wird der Zustand des Frequenzumrichters ausgelesen.



Hinweis:
Steuerwort **410**, *Frequenzsollwert RAM* **484** und *Prozentsollwert RAM* **524** werden im RAM des Frequenzumrichters gespeichert. Diese werden über den Datensatz 0 angesprochen.

- **412 Local/Remote**

Der Frequenzumrichter kann über verschiedene Betriebsarten gesteuert werden. Diese Betriebsarten werden über den Parameter *Local/Remote* **412** eingestellt.

Local/Remote 412	Funktion
0 - Steuerung über Kontakte	Die Befehle Start und Stopp, sowie die Vorgabe der Drehrichtung erfolgen über Digitaleingänge.
1 - Steuerung über Steuerwort der Statemachine	Die Befehle Start und Stopp, sowie die Vorgabe der Drehrichtung erfolgen über die Remote Statemachine[1] der Kommunikationsschnittstelle.
2 - Steuerung über Remote-Kontakte	Die Befehle Start und Stopp, sowie die Vorgabe der Drehrichtung erfolgen mit Hilfe von virtuellen Digitalsignalen durch das Kommunikationsprotokoll.

Für den Betrieb über die serielle Schnittstelle sind die Einstellwerte 0, 1 und 2 relevant. Weitere mögliche Betriebsarten von *Local/Remote* **412** zur Steuerung über das Bedienfeld und über Digitaleingänge sind in der Betriebsanleitung beschrieben.

Der Parameter *Local/Remote* **412** ist datensatzumschaltbar, d. h. per Datensatzanwahl kann zwischen den unterschiedlichen Betriebsarten umgeschaltet werden.

- **414 Datensatzanwahl**

Die Datensatzumschaltung kann über Steuerkontakte an den Digitaleingängen des Frequenzumrichters erfolgen oder über den Bus. Für die Datensatzumschaltung über den Bus wird der Parameter *Datensatzanwahl* **414** genutzt.

Parameter		Einstellung		
Nr.	**Bezeichnung**	**Min.**	**Max.**	**Werkseinst.**
414	Datensatzanwahl	0	5	0

Mit der werkseitigen Einstellung *Datensatzanwahl* **414** = 0 erfolgt die Datensatzumschaltung über die Digitaleingänge.

Ist *Datensatzanwahl* **414** auf 1, 2, 3 oder 4 gesetzt, dann wird über den Bus der damit angewählte Datensatz aktiviert. Gleichzeitig ist die Datensatzumschaltung über die Digitaleingänge deaktiviert.

Ist *Datensatzanwahl* **414** auf 5 gesetzt, erfolgt die Datensatzumschaltung nur dann, wenn der Frequenzumrichter nicht freigegeben ist.

Über den Parameter *aktiver Datensatz* **249** kann der aktuell angewählte Datensatz ausgelesen werden. Dies ist unabhängig davon, ob die Datensatzumschaltung über die Digitaleingänge oder per *Datensatzanwahl* **414** erfolgte.

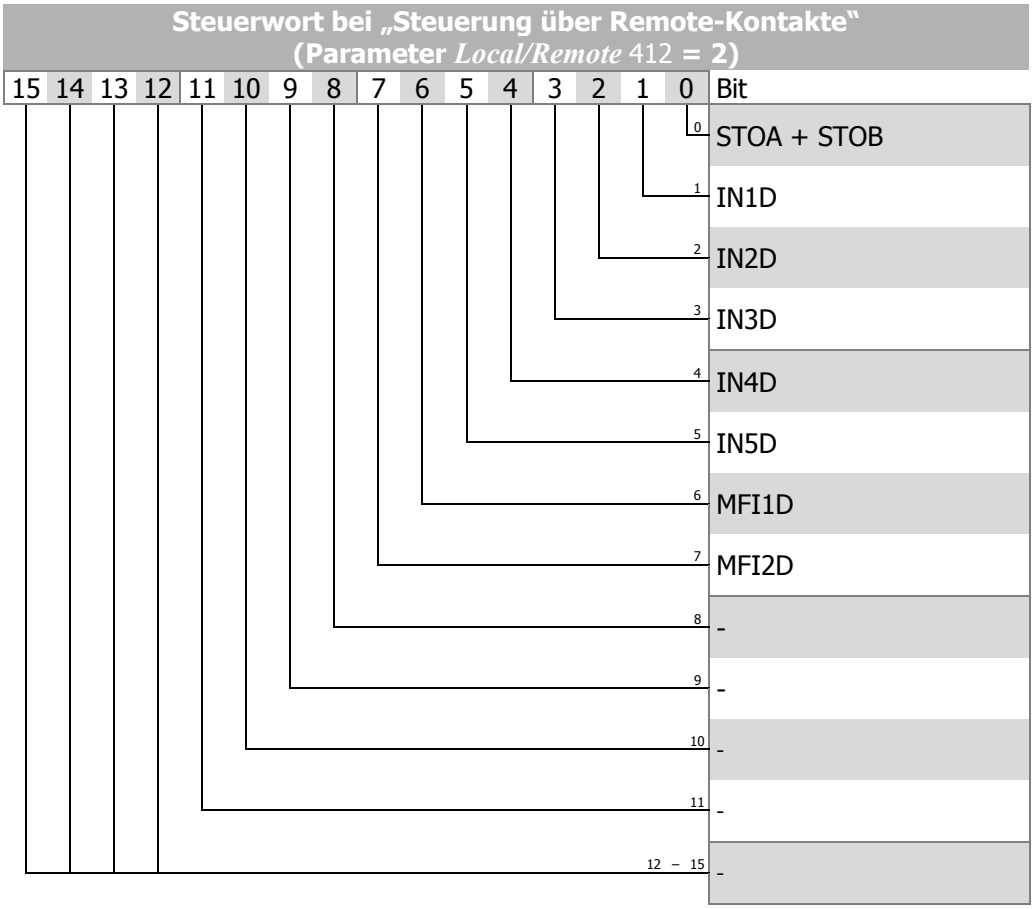
[1] Statemachine ist ein genormtes Softwaremodul innerhalb der Steuerung des Frequenzumrichters. Die Statemachine bildet vorgegebene Betriebszustände und die Steuerung innerhalb des Frequenzumrichters ab.

10.1 Steuerung über Kontakte/Remote-Kontakte

In der Betriebsart „Steuerung über Kontakte" (*Local/Remote* **412** = 0) wird der Frequenzumrichter über die Digitaleingänge oder die als Digitaleingänge eingestellten Multifunktionseingänge gesteuert.

In der Betriebsart „Steuerung über Remote-Kontakte" (Parameter *Local/Remote* **412** = 2) wird der Frequenzumrichter über die einzelnen Bits der virtuellen Digitalsignale im Steuerwort gesteuert.

Wird der Frequenzumrichter über die Digitaleingänge angesteuert, entfällt in dieser Betriebsart die Ansteuerung über das *Steuerwort* **410**.



Wird die Betriebsart „Steuerung über Remote-Kontakte" genutzt, muss die Reglerfreigabe „STOA+STOB" eingeschaltet sein und das Bit 0 des Steuerwortes gesetzt werden, um den Antrieb zu starten.

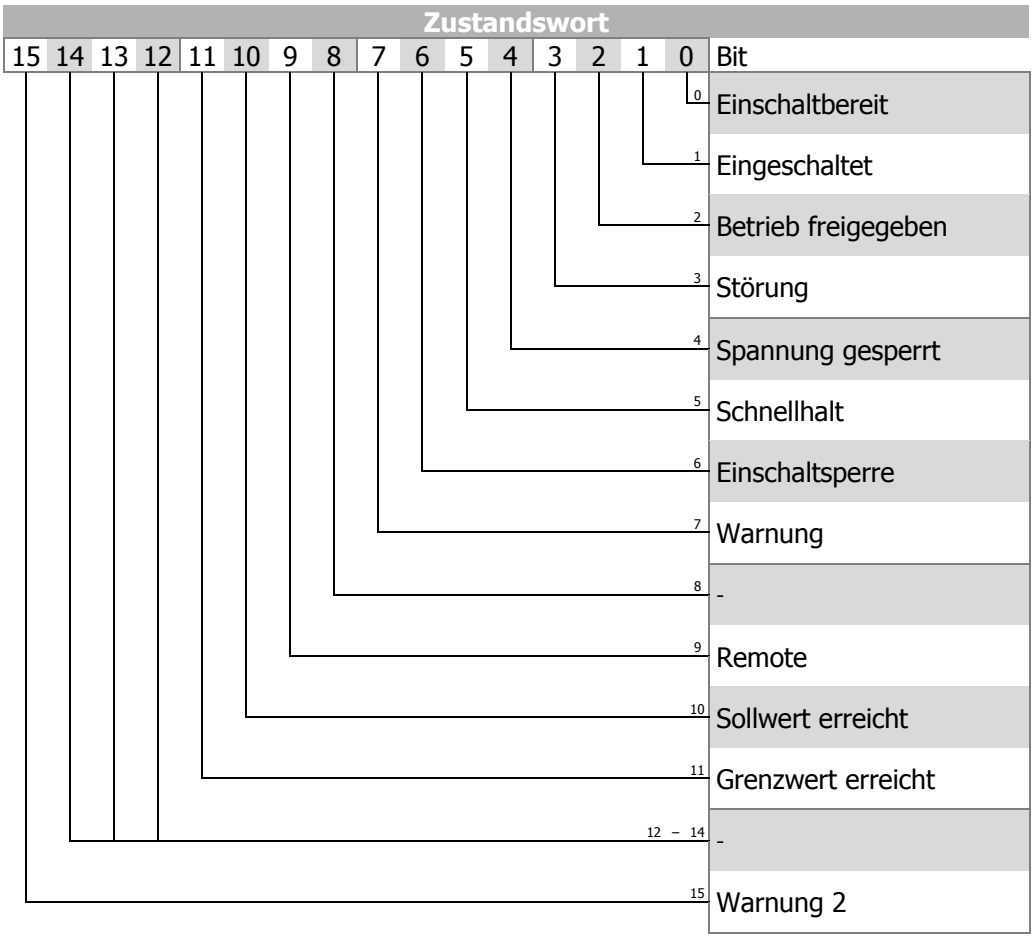
Bei der Verwendung von Remote-Kontakten werden die Signalquellen (Digitaleingänge oder als Digitaleingänge eingestellte Multifunktionseingänge) virtuell vom *Steuerwort* **410** übernommen. Signale an den Hardware-Klemmen werden über die Standard-Betriebsarten (z. B. 72 - IN2D) nicht ausgewertet.

Um Signale an den Hardware-Klemmen auswerten zu können, stehen Betriebsarten zur Verfügung, die mit dem Zusatz „(Hardware)" gekennzeichnet sind.

Ausnahme: Die Freigabe muss immer über die Hardware-Eingänge STOA (Klemme X11.3) und STOB (Klemme X13.3) und das Bit 0 „STOA+STOB" des Steuerwortes erfolgen.

Eine Reglerfreigabe allein per Software ist nicht möglich.

Der Parameter *Zustandswort* **411** hat eine Länge von 16 Bit. Die gesetzten Bits haben folgende Bedeutung:



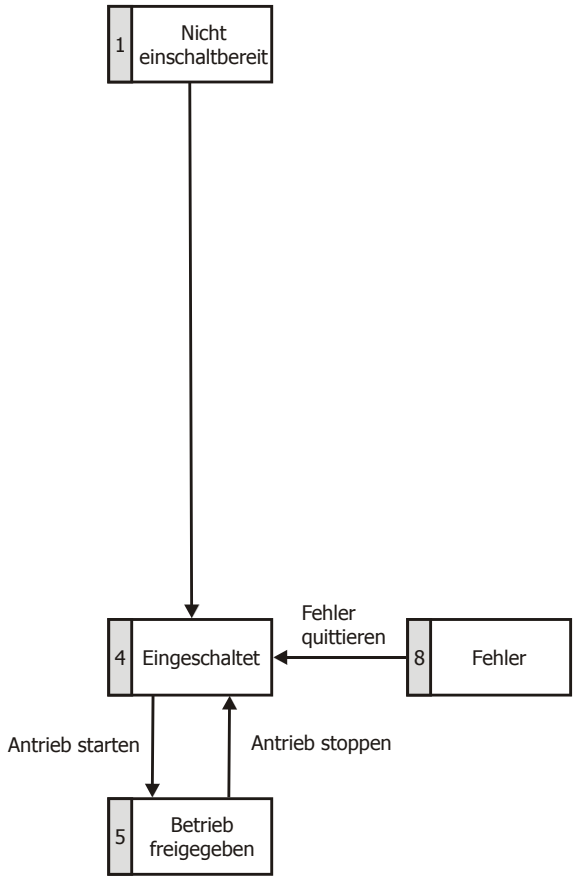
Hinweis:
Der Frequenzumrichter unterstützt eine externe 24 V Spannungsversorgung für die Steuerelektronik des Frequenzumrichters. Auch bei ausgeschalteter Netzspannung ist die Kommunikation zwischen der Steuerung (SPS) und dem Frequenzumrichter möglich.

Das Bit 4 „Spannung freigegeben" des Statuswortes zeigt den aktuellen Status der Netzversorgung.

Bit 4 „Spannung freigegeben" = 0 signalisiert „Keine Netzspannung" und das Starten des Antriebs ist nicht möglich.

Bit 4 „Spannung freigegeben" = 1 signalisiert „Netzspannung eingeschaltet" und der Antrieb ist startbereit.

10.1.1 Geräte Statemachine



Statuswort	Bit 5	Bit 3	Bit 2	Bit 1	Bit 0
Eingeschaltet	1	0	0	1	1
Betrieb freigegeben	1	0	1	1	1
Fehler	x	1	x	x	x

„x" bedeutet beliebiger Wert.

Ist eine Störung aufgetreten, kann die Störungsursache über den Parameter *Aktueller Fehler* **260** ausgelesen werden.

Zustandswort Bit 7 bis Bit 15:

Das Bit 7 „**Warnbit**" kann zu beliebigen Zeitpunkten eine geräteinterne Warnmeldung anzeigen und führt, abhängig von der Ursache, zur Abschaltung des Frequenzumrichters. Die Auswertung, welche Warnung anliegt, erfolgt durch das Auslesen des Warnstatus mit dem Parameter *Warnungen* **270**.

Das Bit 9, „**Remote**" ist bei der Steuerung über Kontakte stets auf den Wert „0" gesetzt.

Das Bit 10 „**Sollwert erreicht**" wird gesetzt, wenn der vorgegebene Sollwert erreicht wurde. Im Sonderfall Netzausfallstützung wird das Bit auch dann gesetzt, wenn die Netzausfallstützung die Frequenz 0 Hz erreicht hat. Für „Sollwert erreicht" gilt eine Hysterese (Toleranzbereich), die über den Parameter *Sollwert erreicht: Schalthysterese* **549** eingestellt werden kann.

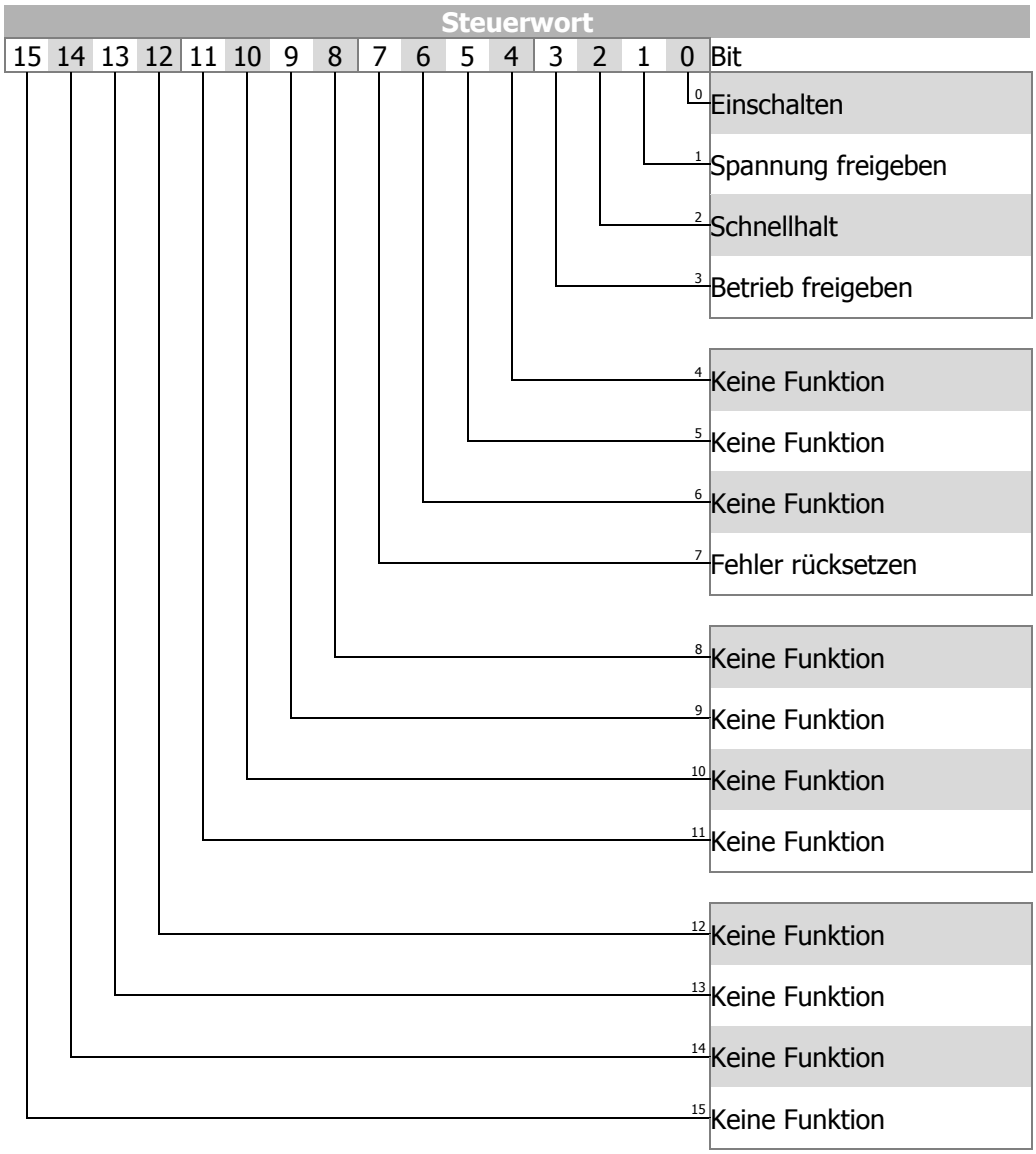
Das Bit 11 „**Grenzwert erreicht**" zeigt an, dass eine interne Begrenzung aktiv ist. Dies kann beispielsweise die Strombegrenzung, die Drehmomentbegrenzung oder die Überspannungsregelung sein. Alle Funktionen führen dazu, dass der Sollwert verlassen oder nicht erreicht wird.

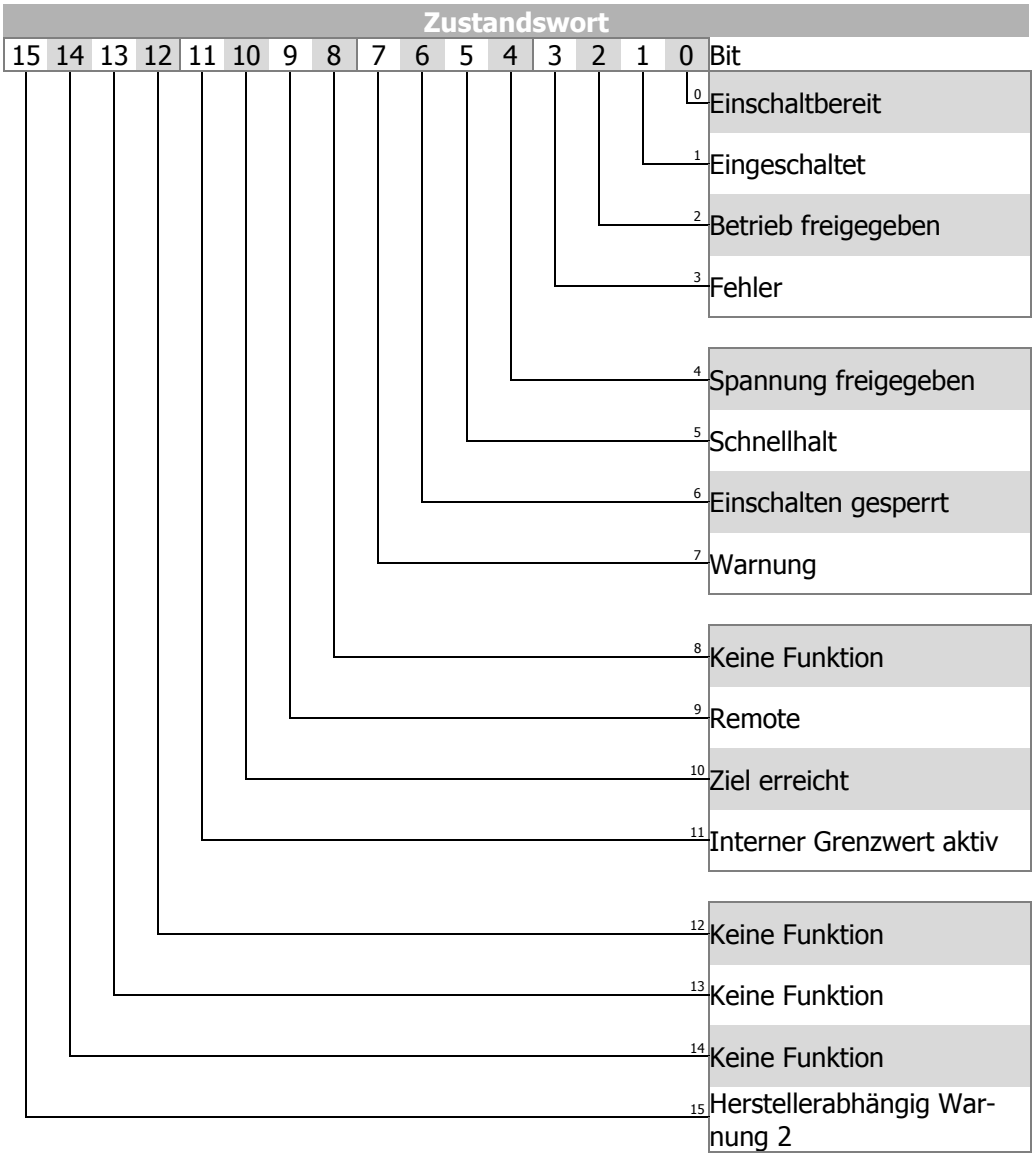
Das Bit 15 „**Warnung 2**" meldet einen kritischen Betriebszustand, der innerhalb kurzer Zeit zu einer Störungsabschaltung des Frequenzumrichters führt. Dieses Bit wird gesetzt, wenn eine zeitverzögerte Warnung für Motor-Temperatur, Kühlkörper-/Innenraum-Temperatur, Ixt-Überwachung oder Netzphasenausfall anliegt.

10.2 Steuerung über Statemachine

In der Betriebsart „Steuerung über Statemachine" (*Local/Remote* **412** = 1) wird der Frequenzumrichter über das Steuerwort der Statemachine angesteuert.

Der Übergang 4 zum Zustand „Betrieb freigegeben" ist nur möglich, wenn die Reglerfreigabe über STOA und STOB und einer der Digitaleingänge für Start Rechtslauf oder Start Linkslauf gesetzt ist.





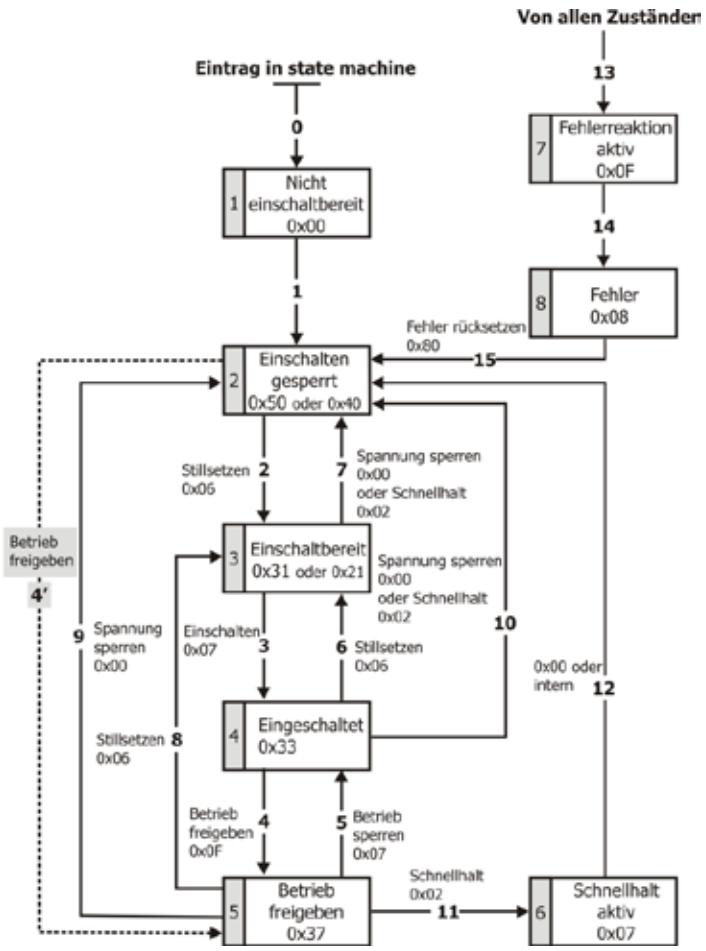
Das Bit 4 „Spannung freigegeben" des Zustandswortes zeigt den aktuellen Status der Netzversorgung.

Bit 4 „Spannung freigegeben" = 0 signalisiert „Keine Netzspannung" und das Starten des Antriebs ist nicht möglich.

Bit 4 „Spannung freigegeben" = 1 signalisiert „Netzspannung eingeschaltet" und der Antrieb ist startbereit.

Hinweis:
Der Frequenzumrichter unterstützt eine externe 24 V Spannungsversorgung für die Steuerelektronik des Frequenzumrichters. Auch bei ausgeschalteter Netzspannung ist die Kommunikation zwischen der Steuerung (SPS) und dem Frequenzumrichter möglich.

10.2.1 Statemachine Diagramm



Die Befehle zur Gerätesteuerung werden durch die folgenden Bitmuster im Steuerwort ausgelöst.

Steuerwort						
	Bit 7	Bit 3	Bit 2	Bit 1	Bit 0	
Befehl	Fehler rück-setzen	Betrieb frei-geben	Schnellhalt	Spannung freigeben	Einschalten	Übergänge
Stillsetzen	X	X	**1**	**1**	**0**	2, 6, 8
Einschalten	X	**0**	**1**	**1**	**1**	3
Einschalten	X	**1**	**1**	**1**	**1**	3
Spannung sperren	X	X	X	0	X	7, 9, 10, 12
Schnellhalt	X	X	**0**	**1**	X	7, 10, 11
Betrieb sperren	X	**0**	**1**	**1**	**1**	5
Betrieb freigeben	X	**1**	**1**	**1**	**1**	4
Fehler rücksetzen	**0 ⇨ 1**	x	x	x	x	15

„X" bedeutet beliebiger Wert.

Der Übergang 3 (Befehl „Einschalten") wird nur verarbeitet, wenn das Bit 4 „Spannung freigegeben" des Zustandsworts gesetzt ist.

Das Zustandswort zeigt den Betriebszustand.

Zustandswort						
	Bit 6	Bit 5	Bit 3	Bit 2	Bit 1	Bit 0
Zustand	Einschalten gesperrt	Schnellhalt	Fehler	Spannung freigegeben	Eingeschaltet	Einschaltbereit
Einschalten gesperrt	**1**	X	**0**	**0**	**0**	**0**
Einschaltbereit	**0**	**1**	**0**	**0**	**0**	**1**
Eingeschaltet	**0**	**1**	**0**	**0**	**1**	**1**
Spannung freigegeben	**0**	**1**	**0**	**1**	**1**	**1**
Schnellhalt aktiv	**0**	**0**	**0**	**1**	**1**	**1**
Fehlerreaktion aktiv	**0**	X	**1**	**1**	**1**	**1**
Fehler	**0**	X	**1**	**0**	**0**	**0**

„X" bedeutet beliebiger Wert.

Das Bit 7 „**Warnung**" kann zu beliebigen Zeitpunkten gesetzt werden. Es zeigt eine geräteinterne Warnmeldung an. Die anliegende Warnung kann im Warnstatus mit dem Parameter *Warnungen* **270** ausgelesen werden.

Das Bit 9 „**Remote**" wird gesetzt, wenn die Betriebsart auf Steuerung über Statemachine (*Local/Remote* **412** = 1) gesetzt ist und die Reglerfreigabe eingeschaltet ist.

Logische Verknüpfung der digitalen Steuersignale:
STO (= STOA und STOB) UND (Start Rechtslauf ODER Start Linkslauf).

Der Frequenzumrichter kann nur gesteuert werden, wenn die logische Verknüpfung wahr ist. Die logischen Eingänge für Start Rechtslauf und Start Linkslauf können direkt mit „Ein" oder „Aus" verbunden werden (Parameter *Start-rechts* **68** und *Start-links* **69**).

Das Bit 10 „**Sollwert erreicht**" wird gesetzt, wenn der eingestellte Sollwert erreicht wird. Im Sonderfall Netzausfallstützung wird das Bit auch dann gesetzt, wenn die Netzausfallstützung die Frequenz 0 Hz erreicht hat (siehe Betriebsanleitung zum Frequenzumrichter). Für „Sollwert erreicht" gilt eine Hysterese (Toleranzbereich), die über den Parameter *Sollwert erreicht: Schalthysterese* **549** eingestellt werden kann.

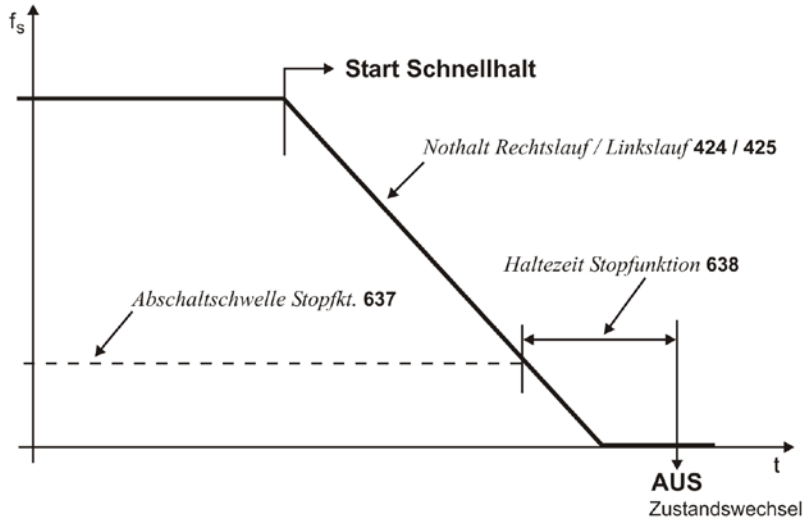
Das Bit 11 „**Grenzwert erreicht**" zeigt an, dass eine interne Begrenzung aktiv ist. Dies kann beispielsweise die Strombegrenzung, die Drehmomentbegrenzung oder die Überspannungsregelung sein. Alle Funktionen führen dazu, dass der Sollwert verlassen oder nicht erreicht wird.

Das Bit 15 „**Warnung 2**" meldet einen kritischen Betriebszustand, der innerhalb kurzer Zeit zu einer Störungsabschaltung des Frequenzumrichters führt. Dieses Bit wird gesetzt, wenn eine zeitverzögerte Warnung für Motor-Temperatur, Kühlkörper-/Innenraum-Temperatur, Ixt-Überwachung oder Netzphasenausfall anliegt.

10.3 Verhalten bei Schnellhalt

Für den Schnellhalt sind die Parameter *Abschaltschwelle Stopfkt.* **637** (Prozentwert von Parameter *maximale Frequenz* **419**) und *Haltezeit Stopfunktion* **638** (Haltezeit nach Unterschreiten der Abschaltschwelle) relevant.

Beim Schnellhalt wird der Antrieb über die Notstopp-Rampen (*Nothalt Rechtslauf* **424** oder *Nothalt Linkslauf* **425**) stillgesetzt.



Ist während der Abschaltzeit die Frequenz/Drehzahl Null erreicht, wird der Antrieb weiterhin bestromt, bis die Abschaltzeit abgelaufen ist. Mit dieser Maßnahme wird sichergestellt, dass beim Zustandswechsel der Antrieb steht.

10.4 Verhalten bei Übergang 5 der Statemachine

- **392 Übergang 5 der Statemachine**

Das Verhalten im Übergang „5" von „Betrieb freigegeben" nach „Eingeschaltet" ist über den Parameter *Übergang 5 der Statemachine* **392** parametrierbar.

Übergang 5 der Statemachine **392**	**Funktion**
0 - freier Auslauf	Sofortiger Übergang von „Betrieb freigegeben" nach „Eingeschaltet", freier Auslauf des Antriebs.
1 - Gleichstrombremse	Aktivierung Gleichstrombremse, mit dem Ende der Gleichstrombremsung erfolgt der Wechsel von „Betrieb freigegeben" nach „Eingeschaltet".
2 - Rampe	Übergang mit normaler Rampe, nach Erreichen des Stillstands erfolgt der Wechsel von „Betrieb freigegeben" nach „Eingeschaltet". Werkseinstellung.

Die Einstellung 1 „Gleichstrombremse" ist nur bei Anwendungen mit U/f-Kennliniensteuerung (Konfiguration 110) möglich. Die anderen Konfigurationen unterstützen diese Betriebsart nicht.

Wird der Frequenzumrichter mit einer Konfiguration betrieben, welche die Betriebsart „Gleichstrombremse" nicht unterstützt, kann der Wert „1" nicht eingestellt werden. Die Betriebsart wird in diesem Fall auch nicht in den Auswahlmenüs des Bedienfelds oder der Bediensoftware VPlus angeboten.

Die Werkseinstellung für *Übergang 5 der Statemachine* **392** ist die Betriebsart 2 (Rampe). Für Konfigurationen mit Drehmomentregelung (Konfigurationen 410 und 610) ist die Werkseinstellung gleich 0 (freier Auslauf).

Bei einem Umschalten der Konfiguration wird gegebenenfalls der Einstellwert für *Übergang 5 der Statemachine* **392** geändert.

Ist *Übergang 5 der Statemachine* **392** mit dem Wert 1 „Gleichstrombremse" ausgelöst worden, wird erst nach dem Abschluss des Übergangsvorgangs ein neues Steuerwort akzeptiert. Der Zustandswechsel von „Betrieb freigegeben" nach „Eingeschaltet" erfolgt nach Ablauf der für die Gleichstrombremse parametrierten *Bremszeit* **632**.

Ist der Parameter *Übergang 5 der Statemachine* **392** = 2 „Rampe" eingestellt, kann während des Herunterfahrens des Antriebs das Steuerwort wieder auf 0x0F gesetzt werden. Damit läuft der Antrieb wieder auf seinen eingestellten Sollwert hoch und verbleibt im Zustand „Betrieb freigegeben".

Der Zustandswechsel von „Betrieb freigegeben" nach „Eingeschaltet" erfolgt nach Unterschreiten der eingestellten Abschaltschwelle und nach Ablauf der eingestellten Haltezeit (äquivalent zum Verhalten bei Schnellhalt). Hierbei sind die Parameter *Abschaltschwelle Stopfkt.* **637** (Prozentwert von Parameter *Maximale Frequenz* **419**) und *Haltezeit Stopfunktion* **638** (Haltezeit nach Unterschreiten der Abschaltschwelle) relevant.

11 Istwerte

Istwerte		
Nr.	**Beschreibung**	**Funktion**
11	VABus SST-Error-Register	Modbus oder VABus Fehlerregister. Siehe Kapitel 7.3.8 „Ausnahmebedingungscodes".
282	Sollfrequenz Bus	Sollwert von serieller Schnittstelle.
283	Sollfrequenz Rampe	Sollwert vom Frequenzsollwertkanal.
411	Zustandswort	Zustandswort. Siehe Kapitel 10 „Steuerung/Sollwert".

12 Parameterliste

Der Parameter ist in den vier Datensätzen verfügbar.

12.1 Istwerte (Menü „Actual")

Istwertparameter				
Nr.	**Beschreibung**	**Einh.**	**Anzeigebereich**	**Kapitel**
RS485/RS232				
11	VABusSST-Error-Register	-	0 ... 15	7.3.8
Istwerte des Frequenzumrichters				
249	Aktiver Datensatz	-	0 ... 4	10
260	Aktueller Fehler	-	0 ... 0xFFFF	13.3
270	Warnungen	-	0 ... 0xFFFF	13.1
274	Warnungen Applikation	-	0 ... 0xFFFF	13.2
282	Sollfrequenz Bus	Hz	-999,99 ... 999,99	11
283	Sollfrequenz Rampe	Hz	-999,99 ... 999,99	11
Bussteuerung				
411	Zustandswort	-	0 ... 0xFFFF	10

12.2 Parameter (Menü „Para")

Parameter				
Nr.	**Beschreibung**	**Einh.**	**Einstellbereich**	**Kapitel**
Bussteuerung				
392	Übergang 5 der Statemachine	-	Auswahl	10.4
RS485/RS232				
395	Protokoll (CM/X21)	-	Auswahl	5.4
Bussteuerung				
410	Steuerwort	-	0 ... 0xFFFF	10
412	Local/Remote	-	Auswahl	10
414	Datensatzanwahl	-	0 ... 4	10
Festfrequenzwerte				
484	Frequenzsollwert RAM	Hz	-999,99 ... 999,99	10
Festprozentwerte				
524	Prozentsollwert RAM	%	-300,00 ... 300,00	10
Modbus (RTU/ASCII)				
1375	Modbus Parität	-	Auswahl	6
1376	Modbus Adresse	-	1 ... 247	6
1503	Modbus Mode	-	Auswahl	6
1504	Modbus Baudrate	Baud	Auswahl	6
1505	Modbus Watchdog Timer	s	0 ... 1000	6

13 Anhang

13.1 Warnmeldungen

Die Warnmeldungen erfolgen bitkodiert gemäß folgendem Schema über den Parameter *Warnungen* **270**.
Parameter *Warnungen* **269** zeigt die Warnungen als Klartext im Bedienfeld und der PC Bediensoftware VPLus.
Verwenden Sie Parameter *Warnungen* **270** um die Warnmeldungen über Modbus auszulesen.

Warnmeldungen			
Bit-Nr.	**Warncode**	**Beschreibung**	
0	0x0001	Warnung Ixt	1) 2)
1	0x0002	Warnung Kurzzeit Ixt	1)
2	0x0004	Warnung Langzeit Ixt	2)
3	0x0008	Warnung Kühlkörpertemperatur Tk	
4	0x0010	Warnung Innenraumtemperatur Ti	
5	0x0020	Warnung I-Limit	
6	0x0040	Warnung Init	
7	0x0080	Warnung Motortemperatur	
8	0x0100	Warnung Netzphasenausfall	
9	0x0200	Warnung Motorschutzschalter	
10	0x0400	Warnung Fmax	
11	0x0800	Warnung Analogeingang MFI1A	
12	0x1000	Warnung Analogeingang MFI2A	
13	0x2000	Warnung Systembus-Slave in Störung	
14	0x4000	Warnung Udc	
15	0x8000	Warnung Applikation	

1) 2): Bit 0 „Warnung Ixt" wird gesetzt,

- wenn Bit 1 „Warnung Kurzzeit Ixt" oder
- wenn Bit 2 „Warnung Langzeit Ixt" gesetzt ist.

Im Parameter *Warnungen* **270** können mehrere Warnungen gleichzeitig angezeigt werden.
Beispiel:

Meldung	Warncode	Bemerkung
Warnung Ixt	0x0001	Wird gesetzt bei Kurzzeit oder Langzeit Ixt.
Kurzzeit Ixt	0x0002	
Warngrenze Kühlkörpertemperatur	0x0008	
Warngrenze Motortemperatur	0x0080	
Summe	**0x008B**	

13.2 Warnmeldungen Applikation

Ist das höchste Bit der Warnmeldung gesetzt, liegt eine „Warnmeldung Applikation" an. Die Applikationswarnmeldungen erfolgen bitkodiert gemäß folgendem Schema über den Parameter *Warnungen Applikation* **274.** Parameter *Warnungen Applikation* **273** zeigt die Warnungen als Klartext im Bedienfeld und der PC Bediensoftware VPlus.
Verwenden Sie Parameter *Warnungen Applikation* **274** um die Warnmeldungen über Modbus auszulesen.

Warnmeldungen		
Bit-Nr.	**Warncode**	**Beschreibung**
0	0x0001	BELT - Keilriemen
1	0x0002	(reserviert)
2	0x0004	(reserviert)
3	0x0008	(reserviert)
4	0x0010	(reserviert)
5	0x0020	(reserviert)
6	0x0040	SERVICE
7	0x0080	User 1
8	0x0100	User 2
9	0x0200	(reserviert)
10	0x0400	(reserviert)
11	0x0800	(reserviert)
12	0x1000	(reserviert)
13	0x2000	(reserviert)
14	0x4000	(reserviert)
15	0x8000	(reserviert)

Hinweis: Die Warnungen sind in der Betriebsanleitung detailliert beschrieben.

13.3 Fehlermeldungen

Modbus		
F20	10	Watchdog für Anschluss X21. Modbus Kommunikationsfehler entsprechend Parameter *X21: Modbus Watchdog Timer* **1505**.
	11	Watchdog für Kommunikationsmodul. Modbus Kommunikationsfehler entsprechend Parameter *X21: Modbus Watchdog Timer* **1505**.

Der aktuelle Fehler kann über Parameter *Aktueller Fehler* **260** ausgelesen werden.
Parameter *Aktueller Fehler* **259** zeigt den aktuellen Fehler als Klartext im Bedienfeld und der PC Bediensoftware VPlus.

Index

A
Adresse 23
Aktiver Datensatz 54
Applikations-Warnungen 66
ASCII 33, 37
  Beispieltelegramm 47
  Tabelle 52
Aufstellung 9
Ausnahmebedingungscode 33
B
Baudrate 22
Beispieltelegramme 42
Bestimmungsgemäße Verwendung 8
Betriebshinweise 10
Busabschluss 17
C
CRC 34
Cyclic Redundancy Check 34
D
Datensatz 54
Datensatzanwahl 54
E
EEPROM Zugriff 40
Elektrische Installation
  Sicherheit 9
F
Fehlermeldungen 66
Frequenzsollwert RAM 53
Funktionscode 25
I
Istwerte 63
L
Lagerung 8
Local/Remote 54
Longitudinal RedundancyChecking 38
LRC 38
M
Master/Slave 24
Modbus
  Adresse 23
  Baudrate 22
  Mode 22
  Parität 22
  Watchdog Timer 23
Modbus Adresse 23
Modbus Baudrate 22
Modbus Mode 22
Modbus Parität 22
Modbus Watchdog Timer 23
Montage
  Kommunikationsmodul 12
P
Parameterliste 64
Parität 22
Protokoll 21, 24
Protokoll (CM/X21) 21
Prozentsollwert RAM 53
R
RAM Zugriff 40
RTU 33, 34
  Beispieltelegramm 42
S
Sicherheit
  Allgemein 8
Sollfrequenz Bus 63
Sollfrequenz Rampe 63
Sollwert 53
Steuerung 53
Steuerwort 53
T
Telegramm
  Aufbau 24
Transport 8
U
Übergang 5 der Statemachine 63
V
VABusSST Error Register 33
W
Warnmeldungen 65
Warnmeldungen Applikation 66
Warnungen 65
Wartung 10
Watchdog Timer 23
Z
Zustandswort 53
ZyklischesSchreiben 40

Bonfiglioli
power, control and green solutions

Seit 1956 plant und realisiert Bonfiglioli innovative und zuverlässige Lösungen für die Leistungsüberwachung und -übertragung in industrieller Umgebung und für selbstfahrende Maschinen sowie Anlagen im Rahmen der erneuerbaren Energien.

www.bonfiglioli.com

Bonfiglioli Riduttori S.p.A.
Via Giovanni XXIII, 7/A
40012 Lippo di Calderara di Reno
Bologna, Italy

tel: +39 051 647 3111
fax: +39 051 647 3126
bonfiglioli@bonfiglioli.com
www.bonfiglioli.com

VEC 692 R1