



*Betriebsanleitung*

# **SM- Applications**

Solutions-Modul für  
Unidrive SP

Artikelnummer: 0471-0037-04  
Ausgabe: 4

## **Allgemeine Informationen**

Der Hersteller übernimmt keinerlei Haftung für Schäden, die durch fehlerhafte, falsche oder unpassende Installation oder falsche Einstellung der optionalen Parameter des Produktes oder durch eine unpassende Kombination eines Motors mit diesem Produkt (Umrichter) entstehen.

Der Inhalt der vorliegenden Betriebsanleitung gilt zum Zeitpunkt der Drucklegung als richtig. Zur Aufrechterhaltung kontinuierlicher Entwicklungs- und Verbesserungsanstrengungen behält sich der Hersteller das Recht vor, die Spezifikationen des Produkts und seine Leistungsdaten sowie den Inhalt der Betriebsanleitung ohne vorherige Ankündigung zu ändern.

Alle Rechte vorbehalten. Ohne schriftliche Genehmigung des Herstellers darf kein Teil dieser Betriebsanleitung reproduziert oder in irgendeiner Form elektronisch oder mechanisch versendet oder in ein Speichersystem kopiert oder aufgezeichnet werden.

## **Version der Umrichtersoftware**

Dieses Produkt wird mit der neuesten Version der Benutzeroberfläche und Antriebssteuersoftware ausgeliefert. Falls dieses Produkt mit anderen Umrichtern in einem neuen oder bestehenden System eingesetzt werden soll, können zwischen der Softwareversion dieser Umrichter und der Softwareversion dieses Produktes Unterschiede auftreten. Diese Unterschiede können zu einem abweichenden Funktionsverhalten führen. Dies gilt möglicherweise auch für Umrichter, die aus einem Servicezentrum von Control Techniques wieder zurückgeliefert werden.

Sollten diesbezüglich irgendwelche Zweifel bestehen, muss ein Servicezentrum von Control Techniques kontaktiert werden.

Copyright      © 25 June 2004 Control Techniques Drives Ltd  
Ausgabe:        4  
Hardware:      SM-Applications Ausgabe 03.00  
Firmware:      V01.03.02

---

# Inhaltsverzeichnis

---

<b>1</b>	<b>Sicherheitsinformationen</b>	<b>6</b>
1.1	Warnungen, Vorsichtsmaßnahmen und Hinweise	6
1.2	Elektrische Sicherheit - allgemeine Warnung	6
1.3	Systemauslegung und Sicherheit des Personals	6
1.4	Umgebungsgrenzwerte	7
1.5	Einhaltung von Vorschriften	7
1.6	Motor	7
1.7	Einstellen der Parameter	7
<b>2</b>	<b>Einführung</b>	<b>8</b>
2.1	Einführung	8
2.2	SM-Applications-Modul für Unidrive SP	8
2.3	Spezifikationen	9
2.4	PC-Entwicklungssoftware	9
2.5	Benutzerkenntnisse	9
<b>3</b>	<b>Installation</b>	<b>10</b>
3.1	Mechanische Installation	10
3.2	Elektrische Verbindungen	10
3.3	CTNet-Anschlüsse	11
3.4	CTNet-Kabel	11
3.5	CTNet-Netzwerkabschluss	12
3.6	EIA-RS485-Anschlüsse	13
3.7	Digital-E/A-Anschlüsse	14
3.8	Anschlussisolierung	15
3.9	SmartCard-Installation	15
<b>4</b>	<b>Erste Schritte</b>	<b>17</b>
4.1	Verwendung von SYPT Workbench	18
4.2	Verbinden des PC mit dem SM-Applications-Modul	18
4.3	SM Applications-Steckplatzauswahl	18
4.4	Konfigurieren der Kommunikation in SYPT	19
4.5	Erstellen eines Unidrive SP-Knotens in SYPT	19
4.6	Portieren von UD70-Programmen in das SM-Applications-Modul	19
4.7	Grundlagen der DPL-Programmierung	19
4.8	Programmbeispiel	20
4.9	Herunterladen von Programmen	22
4.10	Vorbehalte zur SYPT Workbench	22

<b>5</b>	<b>Parameter</b>	<b>23</b>
5.1	Überblick	23
5.2	Speichern der Parameter	23
5.3	Konfigurationsparameter	24
5.4	Menüs 70 bis 75 - SPS-Register	33
5.5	Menü 85 - Zeitgeberfunktionsparameter	34
5.6	Menü 86 - Digital-E/A-Parameter	36
5.7	Menü 88 - Statusparameter	37
5.8	Menü 90 - allgemeine Parameter	38
5.9	Menü 91 - Schnellzugangsparameter	45
5.10	Menüs 18 und 19 - Anwendungsparameter	49
5.11	Menü 20 - Anwendungsmenü	50
<b>6</b>	<b>Kommunikation</b>	<b>51</b>
6.1	Serieller RS485-Kommunikationsanschluss	51
6.2	CTNet	53
6.3	SM-Applications-Zuordnungsparameter	54
<b>7</b>	<b>DPL-Programmierung</b>	<b>56</b>
7.1	Programmkopfzeile	56
7.2	Tasks	56
7.3	Variablen	59
7.4	Parameter	61
7.5	Operatoren	62
7.6	Grundlegende DPL-Befehle	63
7.7	Benutzerdefinierte Funktionsblöcke	68
<b>8</b>	<b>SM-Applications-Freeze und -Nullimpuls</b>	<b>70</b>
8.1	Freeze-Eingang	70
8.2	Nullimpuls	71
<b>9</b>	<b>CTSync</b>	<b>72</b>
9.1	Überblick	72
9.2	Anschlüsse	72
9.3	Beschränkungen	72
9.4	CTSync-Funktionsblöcke	72
9.5	Motion Engine	74
9.6	Beispiel für einen Virtual Master	76
<b>10</b>	<b>Synchronisation zwischen den Optionsmodulen</b>	<b>78</b>
10.1	Überblick	78
10.2	Beispiel 1 zur Synchronisation zwischen Optionen	78
10.3	Beispiel 2 zur Synchronisation zwischen Optionen	81
10.4	Lageregelungs-Tasks	84

<b>11</b>	<b>Fehlersuche</b>	<b>85</b>
11.1	Laufzeitfehler	85
11.2	Unidrive SP-Displaycodes bei der Fehlerabschaltung	85
11.3	SM-Applications-Laufzeitfehlercodes	86
11.4	Bearbeiten von Laufzeitfehlern mit dem ERROR-Task	89
11.5	Support	90
<b>12</b>	<b>Migrationsanleitung</b>	<b>91</b>
12.1	Unterschiede bei den Umrichterparametern	91
12.2	UD70-Parameter	91
12.3	Allgemeine Funktionen	92
12.4	SM-Applications-Portierhilfe	94
<b>13</b>	<b>Kurzreferenz</b>	<b>96</b>

---

# 1 Sicherheitsinformationen

---

## 1.1 Warnungen, Vorsichtsmaßnahmen und Hinweise



Eine **Warnung** enthält Informationen, die für die Vermeidung eines Sicherheitsrisikos von entscheidender Bedeutung sind.



Ein mit **Vorsicht** gekennzeichnete Absatz enthält Informationen, die zur Vermeidung von Schäden am Umrichter oder an Zubehör notwendig sind.



Ein **Hinweis** enthält Informationen, die für den ordnungsgemäßen Betrieb des Produkts hilfreich sind.

## 1.2 Elektrische Sicherheit - allgemeine Warnung

Umrichterspannungen können schwere bis tödliche Elektroschocks bzw. Verbrennungen verursachen. Beim Umgang mit dem Umrichter oder der Arbeit in dessen Nähe ist besondere Vorsicht geboten. Spezifische Warnungen sind an den entsprechenden Stellen in dieser Betriebsanleitung enthalten.

## 1.3 Systemauslegung und Sicherheit des Personals

Der Umrichter ist für den professionellen Einbau in Komplettanlagen bzw. -systeme bestimmt. Bei nicht fachgerechtem Einbau kann der Umrichter ein Sicherheitsrisiko darstellen.

Der Umrichter arbeitet mit hohen Spannungen und Strömen sowie mit hohen elektrischen Ladungen. Er dient der Steuerung von Geräten, die ebenfalls gefährlich sein können.

Die Elektroinstallation und die Systementwicklung erfordern besondere Aufmerksamkeit, damit Gefahren sowohl beim normalen Betrieb als auch im Falle einer Funktionsstörung vermieden werden können. Systementwicklung, Installation, Inbetriebnahme und Wartung müssen von Mitarbeitern durchgeführt werden, die entsprechend geschult und erfahren sind. Zuvor müssen diese Sicherheitsinformationen und diese Betriebsanleitung sorgfältig durchgelesen werden.

Durch die Funktionen STOP und SICHERER HALT des Umrichters werden gefährliche Spannungen nicht vom Umrichterausgang oder anderen externen Modulen ferngehalten. Das Netz muss durch eine zugelassene Trennungseinrichtung vom Umrichter getrennt werden, bevor an den elektrischen Verbindungen gearbeitet werden kann.

**Mit Ausnahme der Funktion SICHERER HALT darf keine der Umrichterfunktionen zum Schutz des Personals genutzt werden, d. h. diese Funktionen dürfen nicht zu Sicherheitszwecken eingesetzt werden.**

Besondere Vorsicht ist bei den Funktionen des Umrichters geboten, die entweder durch ihre vorgesehene Wirkung oder durch auftretende Fehlfunktionen gefährlich werden können. Bei allen Anwendungen, bei denen eine Funktionsstörung des Umrichters bzw. seines Steuersystems Beschädigungen, Verluste oder Verletzungen herbeiführen kann, muss eine Gefahrenanalyse vorgenommen werden; falls erforderlich, sind weitere Maßnahmen zur Verringerung solcher Risiken zu treffen. Bei Ausfall der Drehzahlregelung kann dies z. B. ein Überdrehzahlschutz oder bei Versagen der Motorbremse eine ausfallsichere mechanische Bremse sein.

Die Funktion SICHERER HALT erfüllt die Anforderungen der Norm EN954-1, Kategorie 3 (Verhindern des unbeabsichtigten Motoranlaufes) und wurde dafür zugelassen<sup>1</sup>. Sie kann in Anwendungen mit Sicherheitsfunktionen genutzt werden. **Der Systementwickler ist dafür verantwortlich, dass das gesamte System sicher ist und gemäß den geltenden Sicherheitsbestimmungen ausgelegt wird.**

<sup>1</sup>Eine unabhängige BIA-Genehmigung wurde für die Baugrößen 1 bis 3 erteilt.

## 1.4 Umgebungsgrenzwerte

Die in der *Unidrive SP-Betriebsanleitung* enthaltenen Anweisungen bezüglich Transport, Lagerung, Installation und Einsatz des Umrichters müssen eingehalten werden, einschließlich der angegebenen Umgebungsgrenzwerte. Umrichter dürfen keinen übermäßigen mechanischen Kräfteinwirkungen ausgesetzt werden.

## 1.5 Einhaltung von Vorschriften

Der Installierer ist für das Befolgen aller entsprechenden Vorschriften verantwortlich. Dazu zählen nationale Bestimmungen zur Auslegung von Stromleitungen, Unfallverhütungsvorschriften und Vorschriften zur elektromagnetischen Verträglichkeit (EMV). Besondere Aufmerksamkeit muss dabei den Querschnittsflächen von Leitern, der Auswahl von Sicherungen oder anderen Schutzvorrichtungen und den Schutzerdungsanschlüssen gewidmet werden.

Die *Unidrive SP-Betriebsanleitung* enthält Anweisungen zur Einhaltung bestimmter EMV-Normen.

Innerhalb der Europäischen Union müssen alle Maschinen, in denen dieses Produkt eingesetzt wird, den folgenden Richtlinien entsprechen:

98/37/EG: Maschinensicherheit.

89/336/EWG: Elektromagnetische Verträglichkeit.

## 1.6 Motor

Stellen Sie sicher, dass der Motor gemäß den Empfehlungen des Herstellers installiert wird. Achten Sie darauf, dass die Antriebswelle des Motors nicht offen liegt.

Standard-Asynchronmotoren mit Käfigläufen sind für den Betrieb mit konstanter Drehzahl ausgelegt. Wenn die Fähigkeit des Umrichters, einen Motor mit Drehzahlen oberhalb des Höchstwerts der Auslegung zu betreiben, genutzt werden soll, ist dringend zu empfehlen, dies zunächst mit dem Hersteller abzusprechen.

Niedrige Drehzahlen können zu einer Überhitzung des Motors führen, da der Kühllüfter an Effektivität verliert. Der Motor sollte mit einem Schutzthermistor ausgestattet werden. Gegebenenfalls sollte ein elektrischer Fremdlüfter verwendet werden.

Die Werte der im Umrichter eingestellten Motorparameter beeinflussen die Schutzfunktionen für den Motor. Die für den Umrichter eingestellten Standardwerte dürfen für den Schutz des Motors nicht als ausreichend betrachtet werden.

Es ist wichtig, dass in Parameter **0.46** (Motornennstrom) der richtige Wert eingegeben wird. Das wirkt sich auf den thermischen Schutz des Motors aus.

## 1.7 Einstellen der Parameter

Einige Parameter können den Betrieb des Umrichters stark beeinflussen. Vor einer Änderung dieser Parameter sind die entsprechenden Auswirkungen auf das Steuersystem sorgfältig abzuwägen. Es müssen Maßnahmen getroffen werden, um unerwünschte Reaktionen durch Fehlbedienung oder unsachgemäßen Eingriff zu vermeiden.

---

## 2 Einführung

---

**HINWEIS** Unidrive SP- und SM-Applications-Parameter sind in diesem Handbuch als „#MM.PP“ angegeben, wobei MM für die Menünummer steht und PP für die Parameternummer in dem jeweiligen Menü. Vollständige Parameterdefinitionen finden Sie im Unidrive SP-Handbuch. Außerdem finden Sie in Abschnitt 5.2 *Speichern der Parameter* weitere Informationen zur Speicherung der Parameter.

### 2.1 Einführung

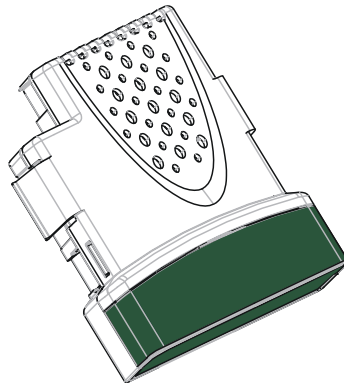
Moderne Frequenzumrichter wie der Unidrive SP bieten eine Vielzahl von eingebauten Funktionen, zum Beispiel Rampensteuerung, PID-Schleifen, einfache Positionierung usw. Diese Funktionalität ist jedoch begrenzt. Der Umrichter verfügt nur über gewisse Funktionen, und wenn es um komplexere Anwendungen geht, müssen Benutzer aus Systemsicht häufig auf externe Geräte wie ein SPS zurückgreifen, um den Umrichter zu steuern.

Die Flexibilität des Unidrive SP kann jedoch mit Hilfe eines SM-Applications-Moduls erheblich gesteigert werden. Dieses Modul ist ein zusätzlicher Prozessor für den Umrichter und ermöglicht es dem Benutzer, bestehende anwendungsspezifische Software zu nutzen oder eine eigene zu schreiben. Es bietet außerdem leistungsfähige Netzwerkfunktionen, so dass viele Umrichter (und andere Geräte) miteinander verbunden werden können, um Informationen im gesamten Prozess weiterzugeben und so eine komplette Anwendungslösung zu erreichen.

### 2.2 SM-Applications-Modul für Unidrive SP

Das SM-Applications-Modul für den Unidrive SP ist ein Optionsmodul, das an Erweiterungssteckplätzen des Unidrive SP angebracht werden kann.

**Figure 2-1 Unidrive SP-Optionsmodul**



Das SM-Applications-Modul wird über das interne Netzteil des Unidrive SP versorgt.

**HINWEIS** Wenn Sie die SYPT Workbench zum Programmieren des SM-Applications-Moduls verwenden, müssen Sie in SYPT angeben, an welchem Optionssteckplatz das Modul angebracht ist. Standardmäßig wird SYPT für die Verwendung von Steckplatz 1 installiert. Dies kann jedoch mit dem Hilfsprogramm zur *Steckplatzauswahl* geändert werden. Darauf kann folgendermaßen zugegriffen werden: Menü „Start“ -> „Programme“ -> „SYPT Workbench“ -> „Slot Selector“.



## 2.3 Spezifikationen

- Frei programmierbarer Hochgeschwindigkeits-Mikroprozessor
- 384 KB Flash-Speicher für das Benutzerprogramm
- 80 KB Speicher für das Benutzerprogramm
- EIA-RS485-Anschluss mit den Protokollen ANSI, Modbus-RTU (Slave- und Master-Modus) sowie Modbus-ASCII (Slave- und Master-Modus)
- Hochgeschwindigkeitsverbindung über ein CTNet-Netzwerk mit einer Datenrate bis zu 5 MBit/s
- Zwei Digitaleingänge (24 V)
- Zwei Digitalausgänge (24 V)
- Dual Port RAM Schnittstelle mit zwei Anschlüssen für die Kommunikation mit dem Unidrive SP und anderen Optionsmodulen
- Task-basiertes Programmiersystem für Echtzeit-Steuerung von Umrichter und Prozess

## 2.4 PC-Entwicklungssoftware

Anwendungsprogramme für das SM-Applications-Modul können vom Benutzer mit Hilfe des Programmierpakets SYPT Workbench entwickelt werden. Benutzer, die das UD70-Produkt für den Unidrive 1 kennen, sind möglicherweise bereits mit dieser Software vertraut.

Die SYPT Workbench bietet verschiedene Tools als Hilfsmittel zum Entwickeln von Lösungen:

- Konfigurationseditor zum Konfigurieren von Umrichtern und Verbindungen in einem CTNet-Netzwerk
- IEC61131-3-basierte statt "Applications Lite"-> "Kontaktplan-" und Funktionsblock-Programmierung
- Systemeigene Programmierung in der DPL-Sprache
- Beobachtungsfenster zur Überwachung von Umrichter- oder Optionsparametern sowie Benutzerprogrammvariablen
- Debugging-Funktionen (Einzelschritt und Stützpunkt)

Mit SYPT Workbench können Sie eine der folgenden Verbindungen zum SM-Applications-Modul herstellen:

- Direkte Verbindung zum RS485-Anschluss an der Vorderseite des Unidrive SP
- Verbindung zu einer oder mehrerer Optionen in einem CTNet-Netzwerk (eine CTNet-Schnittstellenkarte für den PC ist erforderlich)

SYPT Workbench kann unter Microsoft Windows™ 98, ME, NT4, 2000 oder XP ausgeführt werden.

## 2.5 Benutzerkenntnisse

Beim Entwickeln von benutzerdefinierter Anwendungssoftware ist es vorteilhaft, über gewisse Kenntnisse in Programmierung auf der Grundlage von Echtzeit-Tasks und Ereignissen zu verfügen. Grundkenntnisse in der Programmiersprache BASIC sind ebenfalls vorteilhaft, jedoch nicht von entscheidender Bedeutung. Die Applications Lite- und Funktionsblock-Funktionen der SYPT Workbench (LD und FBD) machen die Umstellung für Benutzer, die bereits mit SPS-Programmierung vertraut sind, erheblich einfacher.

In dieser Betriebsanleitung wird davon ausgegangen, dass der Benutzer zumindest über oberflächliche Kenntnisse in Microsoft Windows™ verfügt.

# 3 Installation

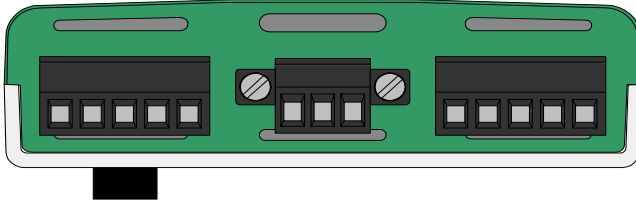
## 3.1 Mechanische Installation

Details zur Installation des SM-Applications-Moduls im Unidrive SP finden Sie in der mit dem Modul ausgelieferten Installationsanleitung.

## 3.2 Elektrische Verbindungen

Das SM-Applications-Modul verfügt über 2 5-fach-Schraubanschlüsse und 1 3-fach-Schraubanschluss.

Abbildung 3-1 SM-Applications-Modul - Vorderansicht



Die Anschluss-klemme sind von Klemme 1 an der linken Seite bis Klemme 13 an der rechten Seite durchnummeriert. Die Funktionen der Anschlussklemmen sind in der folgenden Tabelle angegeben:

Tabelle 3.1 Anschlüsse des SM-Applications-Moduls

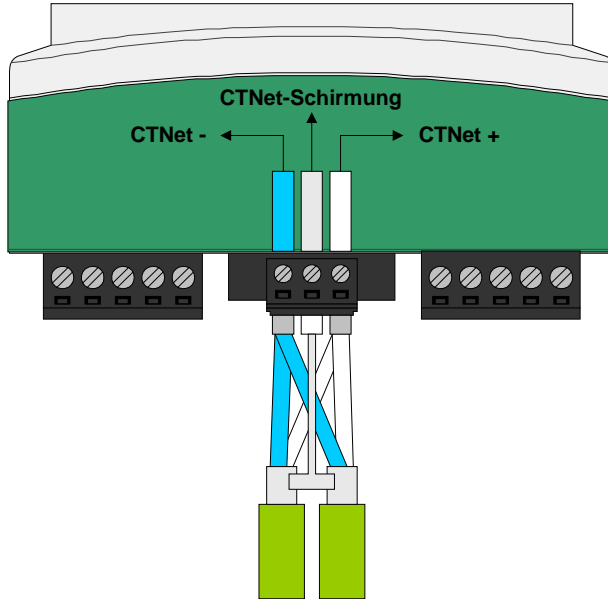
Anschluss-klemme	Funktion	Beschreibung
1	0V SC	0 V-Verbindung für RS485-Anschluss
2	/RX	EIA-RS485-Empfangsleitung (negativ). Eingehend.
3	RX	EIA-RS485-Empfangsleitung (positiv). Eingehend.
4	/TX	EIA-RS485-Sendeleitung (negativ). Abgehend.
5	TX	EIA-RS485-Sendeleitung (positiv). Abgehend.
6	CTNet A	CTNet-Datenleitung
7	CTNet Shield	Schirmverbindung für CTNet
8	CTNet B	CTNet-Datenleitung
9	0V	0 V-Verbindung für Digital-E/A
10	DIO	Digitaleingang 0
11	DI1	Digitaleingang 1
12	DO0	Digitalausgang 0
13	DO1	Digitalausgang 1

### 3.3 CTNet-Anschlüsse

**HINWEIS** Diese Betriebsanleitung enthält lediglich die Grundlagen zu den Anschlüssen in einem CTNet-Netzwerk. Vollständige Informationen finden Sie in der CTNet-Betriebsanleitung.

Wenn Sie das SM-Applications-Modul an das CTNet-Netzwerk anschließen möchten, stellen Sie die in der folgenden Abbildung dargestellten Verbindungen her.

**Abbildung 3-2 CTNet-Netzwerkanschlüsse**



Die Schirmungen der beiden Kabel sollten miteinander verdreht und an die mittlere Anschlussklemme des 3-fach-Schraubanschlusses angeschlossen werden. Diese „Anschlusslitze“ sollte möglichst kurz gehalten werden. Durch diese Konstruktion wird eine durchgängige Schirmung gewährleistet.

### 3.4 CTNet-Kabel

CTNet-Kabel verfügt über ein einzelnes verdrehtes Leitungspaar sowie eine Gesamtschirmung. Ein Satz Datenanschlussklemmen wird mitgeliefert. Dies hat den Vorteil, dass für den Fall einer Trennung der Anschlussklemme die Kontinuität des CTNet-Netzwerks nicht unterbrochen wird.

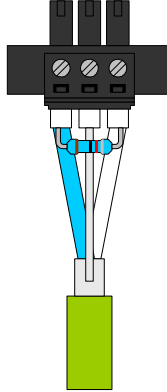
CTNet-Netzwerke werden mit hohen Datenraten betrieben und erfordern Kabel, die speziell für Hochfrequenzsignale ausgelegt sind. In Kabeln von geringer Qualität werden die Signale gedämpft, was dazu führen kann, dass das Signal für die anderen Knoten in dem Netzwerk nicht mehr lesbar ist. Das einzige empfohlene CTNet-Kabel ist das von Control Techniques gelieferte.

## 3.5 CTNet-Netzwerkabschluss

In Hochgeschwindigkeits-Kommunikationsnetzwerken ist es sehr wichtig, dass das Netzkommunikationskabel mit dem spezifizierten Abschlusswiderstandsnetzwerk an jedem Ende des Kabels versehen wird. Dadurch wird verhindert, dass Signale in das Kabel reflektiert werden und dort Störungen verursachen.

Der Abschlusswiderstand sollte möglichst genau der Impedanz des Kabels entsprechen. Für das empfohlene grüne CTNet-Kabel von Control Techniques sollten Abschlusswiderstände mit 82 Ohm und 0,25 W über die Datenleitungen CTNet+ und CTNet- an BEIDEN Enden des Kabels angebracht werden.

**Abbildung 3-3 CTNet-Netzwerkabschluss**



Wenn ein Netzwerk nicht korrekt abgeschlossen wird, kann der Betrieb des Netzwerks erheblich beeinträchtigt werden. Ohne die richtigen Abschlusswiderstände ist die Störungsunempfindlichkeit des Netzwerks erheblich reduziert.

Wenn zu viele Abschlusswiderstände in einem CTNet-Netzwerk angebracht sind, wird das Netzwerk überlastet, was zu niedrigeren Signalpegeln führt. Aufgrund dessen werden in Knoten möglicherweise nicht alle gesendeten Bits empfangen, was dazu führt, dass Sendefehler gemeldet werden. Wenn die Netzwerküberlastung zu stark ist, können die Signalpegel so niedrig werden, dass an Knoten überhaupt keine Netzwerkaktivität erkannt wird.

### 3.5.1 CTNet-Kabelschirmanschlüsse

Die CTNet-Verbindung sollte an jedem Unidrive SP verdrahtet werden, wobei die Kabelschirmungen gegenüber der Erde isoliert sein müssen. Die Kabelschirmungen sollten an der Stelle miteinander verbunden werden, an der sie aus dem Kabel austreten. Aus ihnen sollte eine kurze Anschlusslitze für den Anschluss an die CTNet-Schirmverbindung an der Anschlussklemme geformt werden, wie bereits dargestellt. Aus Sicherheitsgründen muss die CTNet-Schirmung an einem Punkt mit der Erde verbunden werden. Mit dieser Erdverbindung soll verhindert werden, dass die Kabelschirmung im Falle eines katastrophalen Fehlers an einem anderen Gerät in dem CTNet-Netzwerk oder entlang des Kabels Strom führt. Aus EMV-Gründen sollte die Erdverbindung nur an einem Punkt hergestellt werden.

### 3.5.2 Maximale Netzwerklänge und Anzahl der Knoten

Die maximale Anzahl an Knoten, die mit einem einzelnen CTNet-Netzwerk verbunden werden können, beträgt 255. Möglicherweise muss ein Netzwerk jedoch in Segmente unterteilt werden, die durch Verstärker getrennt sind. Die maximale Länge des Netzkabels für ein CTNet-Netzwerk ist von der Baudrate abhängig. Verstärker sind bei Control Techniques erhältlich. In der folgenden Tabelle sind die Artikelnummern für die verschiedenen erhältlichen Verstärker aufgeführt.

Artikelnummer	Beschreibung
4500-0033	A13-485X - Version A
4500-0083	A13-485X-CT - Version D (Anschluss 1), Version A (Anschlüsse 2 und 3)
4500-0082	A13-CT - Version D
4500-0032	A12-485X/FOG-ST - Version A (Glasfaser)
4500-0081	A12-CT/FOG-ST - Version D (Glasfaser)

Weitere Informationen zu diesen Aspekten finden Sie in der CTNet-Betriebsanleitung.

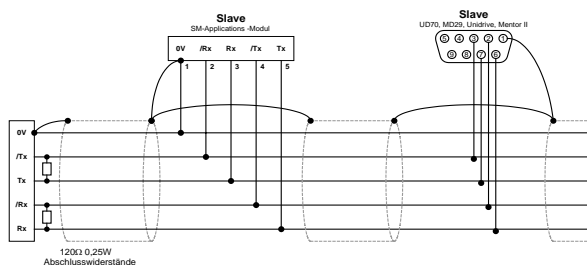
## 3.6 EIA-RS485-Anschlüsse

Der EIA-RS485-Anschluss ist für die Kommunikation mit geringeren Geschwindigkeiten (bis zu 115 200 Bit/s) verfügbar. Standardmäßig werden von dem Anschluss die Protokolle CT-ANSI (Slave-Modus), Modbus-RTU (Master- und Slave-Modus) sowie Modbus-ASCII (Master- und Slave-Modus) unterstützt. Sowohl Zweidraht- als auch Vierdrahtverbindungen sind möglich.

Weitere Informationen zur Verwendung des RS485-Anschlusses finden Sie in Kapitel 6 *Kommunikation*.

### 3.6.1 Vierdraht-RS485-Netzwerk

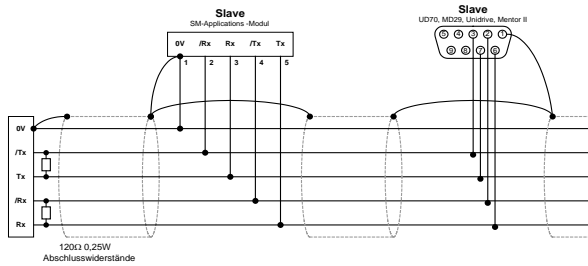
In dem Diagramm unten sind die für ein Vierdraht-RS485-Netzwerk erforderlichen Verbindungen dargestellt, in denen ein Master-Regler mit RS485-Anschluss verwendet wird. SM-Applications-Module können so konfiguriert werden, dass sie als Master-Regler fungieren, jedoch erfordert dies eine DPL-Programmierung, um das Netzwerk zu steuern.



Ein Umsetzer von RS232 nach RS485 wird benötigt, damit eine Kommunikation zwischen einem standardmäßigen seriellen PC-Anschluss und einem Vierdraht-RS485-Netzwerk möglich ist.

### 3.6.2 Zweidraht-RS485-Netzwerk

In dem Diagramm unten sind die für ein Zweidraht-RS485-Netzwerk erforderlichen Verbindungen dargestellt, in dem ein Master-Regler mit RS485-Anschluss verwendet wird. SM-Applications-Module können so konfiguriert werden, dass sie als Master-Regler fungieren, jedoch erfordert dies eine DPL-Programmierung, um das Netzwerk zu steuern.



Ein Umsetzer von RS232 nach RS485 mit „intelligenter Transceiver-Umschaltung“ (auch als „magischer“ RS485-Umsetzer bezeichnet) wird benötigt, damit eine Kommunikation zwischen einem standardmäßigen seriellen PC-Anschluss und einem Zweidraht-RS485-Netzwerk möglich ist. Ein Beispiel für einen „magischen“ Umsetzer ist der Umsetzer MA485F von Amplicon.

**HINWEIS** Ein „magischer“ Umsetzer ist nicht erforderlich, wenn der Master-Regler über einen RTS-Steuerungsausgang verfügt. Dieser Ausgang wird freigegeben, wenn Daten vom Master gesendet werden, und deaktiviert, wenn vom Master keine Daten gesendet werden. Mit Software-Paketen von Control Techniques (UniSoft, MentorSoft und SystemWise) wird die RTS-Leitung NICHT umgeschaltet.

### 3.6.3 Erdung

Aus Sicherheitsgründen wird empfohlen, die Schirmung des Kommunikationskabels über einen Weg mit geringer Induktivität mit einem „sauberen“ Erdungspunkt zu verbinden. Dies darf nur an einem Punkt geschehen.

### 3.6.4 Kabelführung

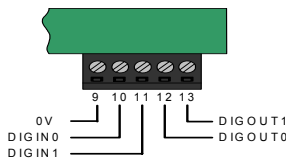
Datenkommunikationskabel, insbesondere solche zwischen Umrichter und Motor, sollten nicht parallel zu Netzkabeln verlaufen. Wenn ein paralleler Verlauf der Kabel unvermeidbar ist, muss ein Mindestabstand von 300 mm zwischen dem Kommunikationskabel und dem Netzkabel eingehalten werden.

Sich rechtwinklig überschneidende Kabel verursachen gewöhnlich keine Störungen. Die maximale Kabellänge einer RS-485-Verbindung beträgt 1200 Meter. Dies gilt nur bei niedrigen Baudraten. Je höher die Baudrate, desto geringer die maximale Kabellänge.

### 3.6.5 Abschluss

Wenn über lange Strecken mit einem RS485-Mehrpunktsystem gearbeitet wird, sollten die Sende- und Empfangspaare mit einem 120 Ω -Abschlusswiderstand versehen werden, damit weniger Signale reflektiert werden. Dies ist jedoch bei den niedrigeren Datenraten nicht so kritisch.

## 3.7 Digital-E/A-Anschlüsse



Das SM-Applications-Modul ist mit 2 Digitaleingängen (Bezeichnungen DIGIN0 und DIGIN1) und 2 Digitalausgängen (Bezeichnungen DIGOUT0 und DIGOUT1) ausgestattet. Diese Ein- und Ausgänge können über das in das SM-Applications-Modul geladene Benutzerprogramm abgefragt bzw. gesteuert werden.

Die Digitalausgänge haben positive Logik, die im aktiven Zustand +24 V treiben und bis zu 20 mA liefern. Im inaktiven Zustand fungieren sie effektiv als Puffer. Die Digitalausgänge sind gegen Kurzschlüsse und Überlastung geschützt. Die Schwelle für eine Fehlerabschaltung liegt bei 20 mA, und im Falle einer Fehlerabschaltung werden beide Ausgänge deaktiviert.

Die Digital-E/A werden mit Hilfe von Menü 86 gesteuert (siehe Abschnitt 5.6 *Menü 86 - Digital-E/A-Parameter* ).

### 3.8 Anschlussisolierung

Die Digital-E/A-Anschlüsse sind mit den Hauptsteuerkreisen des Umrichters verbunden.



Die E/A-Schaltkreise sind nur durch eine Grundisolierung (einfache Isolierung) von den Netzschaltkreisen getrennt. Der Installierer muss sicherstellen, dass externe Steuerkreise durch mindestens eine Isolierungsschicht, die für die angegebene Netzspannung ausgelegt ist, gegen Berührung isoliert sind.

Die CTNet- und EIA/RS485-Anschlüsse sind zusätzlich gegenüber den E/A-Anschlüssen isoliert, so dass sich insgesamt eine doppelte Isolierung gegenüber dem Netzschaltkreis ergibt.

Sie sind einfach voneinander getrennt (funktionale Isolierung).



So erhalten Sie die doppelte Isolierung der Datenanschlüsse:

- Alle Schaltkreise, mit denen einer der Anschlüsse verbunden ist, müssen über eine Schutzabtrennung verfügen (d. h. doppelte Isolierung oder einfache Isolierung mit Erdung)
- Alle Schaltkreise, mit denen die Steuerkreise des Umrichters verbunden sind, müssen mindestens eine grundlegende Isolierung gegenüber den stromführenden Teilen aufweisen.

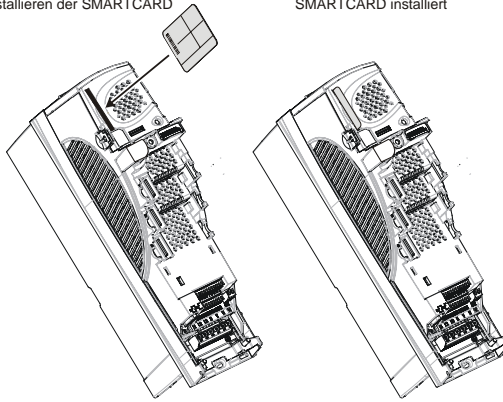
### 3.9 SmartCard-Installation

Der Unidrive SP kann mit einer *a* ausgestattet werden (siehe Abbildung 3-4 *Installieren der SmartCard* unten), mit deren Hilfe der Benutzer Daten abfragen und schreiben kann. Dies ist zum Speichern von Parametersätzen und anderen Datenwerten nützlich. Weitere Informationen zur Verwendung der *a* über die Bedieneinheit finden Sie in der Unidrive SP-Betriebsanleitung. In Abschnitt 7.6.1 *Neue Befehle für das SM-Applications-Modul* finden Sie Informationen zur Verwendung von DPL-Befehlen in Verbindung mit der *a*.

### Abbildung 3-4 Installieren der SmartCard

Installieren der SMARTCARD

SMARTCARD installiert



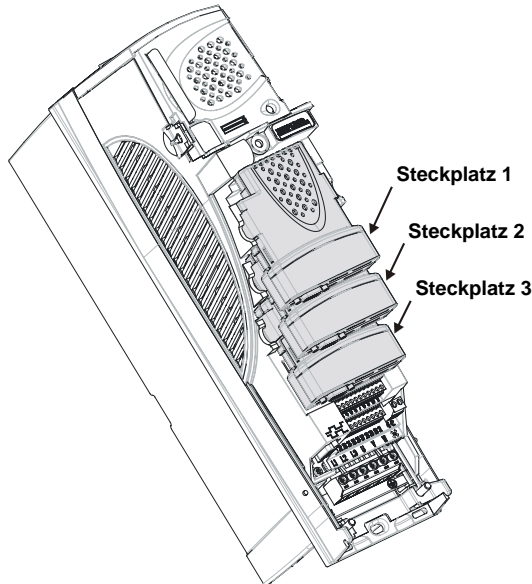


## 4 Erste Schritte

In diesem Kapitel werden die Grundlagen eines Benutzerprogramms beschrieben, bei dem das SM-Applications-Modul verwendet wird, sowie einige Aspekte der Verwendung der SYPT Workbench.

Der Unidrive SP verfügt über 3 Steckplätze für Optionsmodule, und jeder Steckplatz besitzt ein fest programmiertes Menü von Konfigurationsparametern.

**Abbildung 4-1 Unidrive SP-Steckplatzanordnung**



**Tabelle 4.1 Steckplatzkonfigurationsmenü**

Steckplatz	Menü
1	#15.PP
2	#16.PP
3	#17.PP

Innerhalb des SM-Applications-Moduls wird Menü 81 als Alias des aktuellen Steckplatzmenüs verwendet. Daher ist es bei einer Verbindung zum Modul über eine Kommunikationssteckbrücke oder vom Benutzerprogramm aus am einfachsten, wenn die Konfigurationsparameter als Menü 81 referenziert werden.

Im Folgenden wird in dieser Betriebsanleitung bei Verweisen auf einen bestimmten Parameter für einen beliebigen Steckplatz das Format #81.PP verwendet. Beispiel: Der Parameter *automatischer Programmstart nach Netz Ein* wird als #81.13 bezeichnet. Dadurch wird außerdem die Portierbarkeit unterstützt, da ein SM-Applications-Modul, in dessen Code Menü 81 verwendet wird, an jedem Steckplatz angebracht werden kann, und der Code sollte wie gewohnt ausgeführt werden.

Wenn das SM-Applications-Modul angebracht wurde, wird im Modulkennungsparameter #81.01 der Wert 301 angezeigt. Durch die Parameterkombination #81.02 und #81.51 wird die Firmware-Version des Moduls angegeben. Diese Betriebsanleitung wurde für die Firmware-Version V01.03.02 geschrieben.

## 4.1 Verwendung von SYPT Workbench

Mit dem SYPT Workbench-Programm, wie es vom UD70-Produkt für Unidrive 1 verwendet wurde, wird eine der Entwicklungsplattformen für das SM-Applications-Modul bereitgestellt.

Es gibt bestimmte Vorbehalte bezüglich der Verwendung von SYPT Workbench zusammen mit dem SM-Applications-Modul, die später behandelt werden.

## 4.2 Verbinden des PC mit dem SM-Applications-Modul

Es gibt zwei Methoden zum Verbinden des PC mit dem SM-Applications-Modul, und diese werden im Folgenden skizziert:

### 4.2.1 CNet

Bei einer CNet-Verbindung können Sie den PC an ein Netzwerk von Umrichtern anschließen und dadurch alle Umrichter direkt vom PC aus programmieren und steuern. Sie benötigen jedoch eine CNet-Schnittstellenkarte in Ihrem PC. PCI- und PCMCIA-Karten sind von Control Techniques für Desktop- und Laptop-Computer erhältlich.

CNet ermöglicht ein wesentlich schnelleres Laden und schnellere Aktualisierungsraten für die Online-Überwachung.

In Abschnitt 3.3 *CNet-Anschlüsse* finden Sie Details zu den CNet-Anschlüssen am SM-Applications-Modul.

### 4.2.2 Serieller RS485-Anschluss

Mit Hilfe eines speziellen Umsetzers von RS232 nach RS485 können Sie den PC mit dem seriellen RJ45-Anschluss an der Vorderseite des Umrichters verbinden. Zu diesem Zweck ist bei Control Techniques ein spezielles vorgefertigtes Kabel erhältlich.

Dasselbe Kabel wird mit anderen Control Techniques-Produkten verwendet, bei denen ein RJ45 RS485-Anschluss verwendet wird, z. B. mit dem Commander SE.

Der RJ45-Anschluss befindet sich unter einer kleinen Klappe an der Vorderseite des Unidrive SP, direkt unterhalb der Bedieneinheit. Die Pin-outs dieses Anschlusses werden in der Unidrive SP-Betriebsanleitung beschrieben.

Mit dieser Art von Verbindung können Sie nur den Umrichter und das SM-Applications-Modul steuern, mit dem der PC verbunden ist.

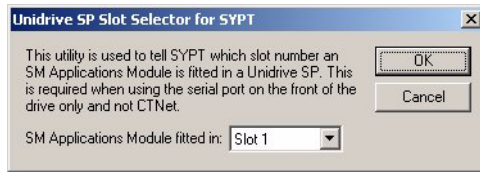
**Abbildung 4-2 Kommunikationskabel**



## 4.3 SM Applications-Steckplatzauswahl

Bei der Verwendung von SYPT Workbench zusammen mit dem RS232/485-Anschluss müssen Sie SYPT darüber informieren, an welchem Steckplatz das SM-Applications-Modul angebracht ist. Dazu wird das Programm *Steckplatzselektor* verwendet (verfügbar über das Menü „Start“ > „Programme“ > „SYPT Workbench“).

### Abbildung 4-3 Steckplatzauswahlprogramm



## 4.4 Konfigurieren der Kommunikation in SYPT

Bevor Sie versuchen, *online* zum SM-Applications-Modul zu gehen, müssen Sie SYPT so konfigurieren, dass das richtige Kommunikationsprotokoll verwendet wird:

1. Wählen Sie im SYPT-Konfigurationseditor aus dem Optionenmenü das Element **Kommunikation** aus.
2. Wenn Sie die Verbindung über CTNet herstellen, wählen Sie CTNet als Protokoll aus, und vergewissern Sie sich, dass die Baudrate korrekt ist. (In der Hilfe finden Sie Details zu den anderen Einstellungen.)
3. Wenn Sie die Verbindung über den RS232- bzw. RS485-Anschluss an der Vorderseite des Umrichters herstellen, wählen Sie **CT-RTU** als Protokoll und anschließend den entsprechenden RS232-COM-Anschluss aus. Vergewissern Sie sich außerdem, dass die Umrichterparameter #11.24 und #11.25 auf **rtU** bzw. **19200** eingestellt sind (dies sind die Standardwerte). Zurzeit ist in SYPT bei diesem Protokoll nur eine Kommunikation mit 19200 Baud möglich.
4. Bestätigen Sie mit OK.

## 4.5 Erstellen eines Unidrive SP-Knotens in SYPT

Ein Unidrive SP-Knoten wird auf dieselbe Art und Weise erstellt wie andere Knotentypen (Umrichtertypen):

1. Fügen Sie einen neuen Knoten ein, indem Sie im Menü „Einfügen“ die Option **Knoten** auswählen.
2. Doppelklicken Sie auf den Namen des neuen Knotens (oder wählen Sie „Bearbeiten“ -> „Knoteneigenschaften“ aus).
3. Wählen Sie aus der Dropdown-Liste der Zieltypen die Option **SMAPPL** aus (der Umrichtertyp wird automatisch in Unidrive SP geändert).
4. Bestätigen Sie mit OK.



1: Node\_1

Der neue Knoten wird jetzt ein Unidrive SP.

## 4.6 Portieren von UD70-Programmen in das SM-Applications-Modul

Wenn Sie beabsichtigen, Programme aus der UD70-Plattform für Unidrive 1 auf den Unidrive SP zu konvertieren, finden Sie in Kapitel 12 *Migrationsanleitung* wichtige Informationen.

## 4.7 Grundlagen der DPL-Programmierung

Das SM-Applications-Modul kann mit Hilfe einer Mischung aus Applications Lite-Diagrammen (LD), Funktionsblockdiagrammen (FBD) und DPL-Code (Drive Programming Language, Umrichterprogrammiersprache) programmiert werden. Zusammen werden sie als *DPL-Programm* bezeichnet.

Sehr grob besteht ein Programm aus Folgendem:

- Programmkopfzeile - Angabe von Programmittel, Autor, Version usw. Diese wird mit Hilfe des Dialogfelds „Knoteneigenschaften“ in SYPT konfiguriert.

- Programmhauptteil - bestehend aus *Task*-Abschnitten mit LD-, FBD- und DPL-Abschnitten. Dieser wird innerhalb von SYPT im DPL-Editor erstellt.

In *Task*-Abschnitten sind Anweisungsblöcke gekapselt, die vom Mikroprozessor zu einer bestimmten Zeit ausgeführt werden, zum Beispiel alle 8 ms oder beim ersten Einschalten des Moduls. Jeder *Task* besitzt einen bestimmten Namen, einen Zweck und eine Priorität. Weitere Informationen finden Sie in Abschnitt 7.2 *Tasks* .

#### 4.7.1 Funktionsblock-Bibliothek

Die SYPT Workbench verfügt über eine umfassende Bibliothek von vorkonfigurierten Funktionsblöcken. Mit diesen werden einfache Aufgaben wie Zähler, aber auch komplexere Vorgänge wie PID-Schleifen oder S-Rampen-Profilgeneratoren ausgeführt. Diese vorgefertigten Blöcke werden zusammen als Funktionsblock-Bibliothek (FBL) bezeichnet.

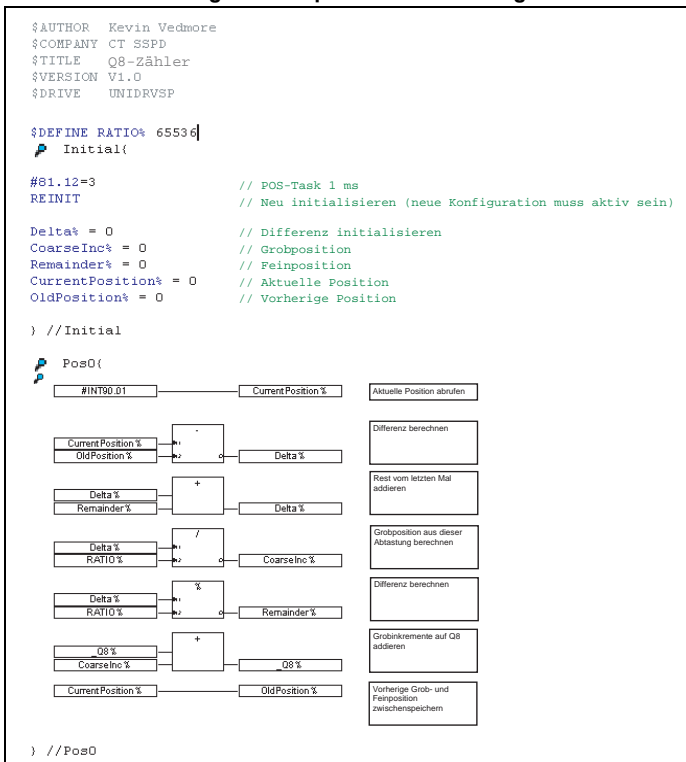
Die Funktionen in der FBL sind in der Online-Hilfe dokumentiert.

Sie können in Ihrem Programm auch eigene Funktionsblöcke erstellen. Wenn Sie also einen tollen neuen Profilergenerator erstellt haben, können Sie ihn in einem benutzerdefinierten Funktionsblock (UDFB) kapseln und in Ihr Haupt-DPL-Programm integrieren. Informationen finden Sie in Abschnitt 7.7 *Benutzerdefinierte Funktionsblöcke* und in der Online-Hilfe.

## 4.8 Programmbeispiel

Im Folgenden ist ein Beispiel für ein DPL-Programm abgebildet:

**Abbildung 4-4 Beispiel für ein DPL-Programm**



Bei diesem Programm werden die Positions-Istwertdaten aus dem Umrichter übernommen (der auf  $2^{32}/U$  skaliert ist), die Differenz wird ermittelt (proportional zur Drehzahl) und in Encoder-Zählerwerte umgerechnet (basierend auf einem Encoder mit Standardquadratur), und diese werden zu einem Akkumulator addiert.

In diesem Beispiel werden die Grundbegriffe des Zugangs zu Parametern und der Verwendung von mathematischen Funktionen dargestellt. Es ist möglicherweise nützlich für Benutzer, die von der UD70-Plattform für Unidrive 1 wechseln, da gezeigt wird, wie der akkumulierende `_Q8%`-Encoderpositionswert wieder erstellt wird, wie er bei diesem Produkt verfügbar war.

In dem Programm werden vier verschiedene Abschnitte durchlaufen:

- Kopfabschnitt
- Ein **Initial**-Task
- Ein **Pos0**-Task
- Ein Funktionsblockdiagramm

#### 4.8.1 Kopfabschnitt

Dieser Abschnitt wird von SYPT aus den Details im Dialogfeld „Knoteneigenschaften“ automatisch generiert. Er enthält Informationen wie Titel, Autor und Version des Programms.

#### 4.8.2 Initial-Task

Wie später in Abschnitt 7.2 *Tasks* erläutert wird, ist dies ein *Task*, der ausgeführt wird, wenn das SM-Applications-Modul erstmals eingeschaltet wird oder wenn es zurückgesetzt wird (siehe den Abschnitt zum Speichern der Parameter). In diesem Task sind einige DPL-Anweisungen enthalten, durch die einige ganzzahlige Variablen (gekennzeichnet durch ein nachgestelltes `%`-Symbol) auf Null initialisiert werden.

#### 4.8.3 Pos0-Task und Funktionsblockdiagramm

Da bei diesem Programm mit Positionswertdaten gearbeitet werden soll, wird der Hauptteil der Arbeit im POS0-Task durchgeführt. Alle Vorgänge im Zusammenhang mit Drehzahl-, Positions- oder Drehmomentregelung werden normalerweise in den Tasks POS0 und POS1 oder im CLOCK-Task durchgeführt, der jetzt mit dem Umrichter synchronisiert ist. In diesem Fall ist ein einzelnes Funktionsblockdiagramm vorhanden, mit dem alle Berechnungen durchgeführt werden, die benötigt werden, um die inkrementelle Encoder-Position zu ermitteln.

Die folgenden Grundschrirte werden ausgeführt:

1. Fragen Sie den aktuellen Encoder-Istwert ab.
2. Subtrahieren Sie den zuvor abgefragten Encoder-Istwert, um die Differenz zu erhalten.
3. Skalieren Sie den Wert auf tatsächliche Encoder-Zählerwerte, wobei ein standardmäßiger inkrementeller Encoder angenommen wird (statt des SinCos-Typs).
4. Addieren Sie diese Differenz zu einem Akkumulator.
5. Merken Sie sich die aktuelle Encoder-Position für das nächste Mal.

In diesem Beispielprogramm wird die Variable `_Q8%` verwendet. Dies ist ein 32 Bit-Wert wie jede andere Variable, er ist jedoch Teil eines bestimmten *Register*-Satzes, der als SPS-Registersatz bezeichnet wird. Diese SPS-Register haben den Vorteil, dass sie im nicht flüchtigen Speicher gespeichert werden können und auch über Parameter in den Menüs 70 bis 76 verfügbar sind. Weitere Informationen dazu finden Sie in Abschnitt 5.4 *Menüs 70 bis 75 - SPS-Register*.

**HINWEIS**

Wenn Sie dieses Programm selbst erstellen und testen möchten und die SYPT Workbench-Software zuvor noch nicht verwendet haben, wird empfohlen, zuerst den Rest dieses Kapitels zu lesen und anschließend das Kapitel „Erste Schritte“ in der SYPT Workbench-Betriebsanleitung, in dem die Erstellung eines solchen Programms erklärt wird.

Um sicherzustellen, dass der POS0-Task ausgeführt wird, muss Parameter #81.12 im Initial-Task auf einen Wert gesetzt werden, der ungleich Null ist. Nachdem dies eingestellt wurde, muss ein REINIT-Befehl ausgegeben werden (siehe unten).

```
#81.12 = 3 //Koordinierungszeitraum für den Pos-Task 1 ms  
REINIT //Neuinitialisierung
```

## 4.9 Herunterladen von Programmen

Standardmäßig können Programme nur dann in das SM-Applications-Modul heruntergeladen werden, wenn das Signal für die Freigabe des Umrichters nicht aktiv ist (#6.15=0). Dieses Verhalten kann deaktiviert werden, indem #81.37 auf 0 gesetzt wird.

## 4.10 Vorbehalte zur SYPT Workbench

Bei der Verwendung der SYPT Workbench mit dem SM-Applications-Modul können die folgenden bekannten Probleme auftreten:

- **Steckplatzauswahl**

Wenn Sie die Verbindung zum Unidrive SP über den seriellen RJ45-Anschluss an der Vorderseite herstellen, wird in SYPT standardmäßig erwartet, dass das SM-Applications-Modul in Steckplatz 1 des Umrichters eingesetzt wird. Sie können diesen jedoch mit Hilfe des mitgelieferten Hilfsprogramms zur Steckplatzauswahl in einen anderen beliebigen Steckplatz ändern. (Zu diesem gelangen Sie über das Menü „Start“ -> „Programme“ -> „SYPT Workbench“ -> „Slot Selector“.)

- **Stützpunkte in den Tasks POS0 und POS1**

Wenn Sie einen Stützpunkt in einem dieser Tasks einstellen, ist der Stützpunkt vorhanden, jedoch wird das Flag nicht angezeigt, und der Stützpunkt kann nur zurückgesetzt werden, indem für das Optionsmodul ein Reset durchgeführt wird.

- **EVENTx-Tasks**

In den EVENT-Tasks können keine Haltepunkte platziert oder Einzelschrittanweisungen ausgeführt werden.

- **Benutzerdefinierte Funktionsblöcke (UDFB)**

Das SM Applications-Modul ist im Hinblick auf die Eingangs- und Ausgangsargumente von UDFBs wesentlich flexibler. (Es sind mehr Eingänge und Ausgänge verfügbar, und es bestehen keine Ausrichtungsbeschränkungen.) In SYPT gelten jedoch noch immer die UD70-Beschränkungen, so dass die neue Flexibilität momentan noch nicht genutzt werden kann.

---

# 5 Parameter

---

## 5.1 Überblick

Das SM-Applications-Modul enthält zwei Parameterdatenbanken:

- Die Unidrive SP-Datenbank  
Diese Datenbank enthält den gesamten Umrichterparametersatz. Sie wird vom SM-Applications-Modul in dem dazugehörigen nicht flüchtigen *Flash*-Speicher zwischengespeichert. Beim Einschalten wird vom Modul überprüft, ob dieser Cache mit dem des Umrichters übereinstimmt. Wenn dies nicht der Fall ist, wird die Datenbank aus dem Umrichter geladen. Während dieser Zeit wird für ein paar Sekunden das Wort „Loading“ auf dem Display des Umrichters angezeigt. Dieser Vorgang wird nur dann wiederholt, wenn das SM-Applications-Modul an einem anderen Umrichter mit anderer Firmware angebracht oder die Firmware des Umrichters aktualisiert wird.
- Die Datenbank des SM-Applications-Moduls  
Diese Datenbank enthält alle lokal in dem Modul gespeicherten Parameter, zum Beispiel SPS-Register, sowie alle anderen Verknüpfungparameter (Menüs 90, 91 usw.)

## 5.2 Speichern der Parameter

Es gibt verschiedene Möglichkeiten, Parameter bei Unterspannung, abhängig von dem Typ der Parameter, die gespeichert werden müssen. Diese werden in den folgenden Abschnitten detailliert erläutert:

### 5.2.1 Speichern der SM-Applications-Parameter

Die folgenden Parameter werden während des Ausführens der unten dargestellten Maßnahmen im SM-Applications-Modul gespeichert:

- Menüs 70, 71, 74 und 75 (äquivalent zu den Registersätzen P, Q, T und U)
- Menü 20

So speichern Sie die Parameter bei Bedarf:

1. Setzen Sie #81.19 auf 1
2. Drücken Sie die Reset-Taste  
Parameter #81.19 wird automatisch auf Null zurückgesetzt, und Modul und Umrichter werden zurückgesetzt.

So speichern Sie die Parameter bei Unterspannung (UU):

- Setzen Sie #81.20 auf 1 (denken Sie daran, dass eine Änderung an diesem Parameter erst nach einem Modul-Reset wirksam wird)

Beachten Sie, dass Menü 20 nicht einfach durch das Ausführen der oben genannten Vorgänge gespeichert werden kann. Um Menü 20 speichern zu können, müssen Sie die oben genannten Vorgänge ausführen, jedoch darauf achten, dass Parameter #81.21 zuvor auf 1 gesetzt wird. Bei diesem Parameter ist kein Modul-Reset erforderlich, damit die Änderungen aktiviert werden.

### 5.2.2 Wiederherstellen der Parameter in Menü 20

Um die Parameter in Menü 20 beim Einschalten wiederherzustellen, muss Parameter #81.21 zu diesem Zeitpunkt auf 1 gesetzt sein. Daher müssen die Umrichterparameter gespeichert werden. Siehe Abschnitt 5.2.3 *Speichern der Umrichterparameter*.

### 5.2.3 Speichern der Umrichterparameter

Die folgenden Parameter werden während des Ausführens der unten dargestellten Maßnahmen im Umrichter gespeichert:

- Menüs 1 bis 14, 18, 19 und 21

- Menüs 15, 16 und 17, wenn in dem entsprechenden Steckplatz ein Modul vorhanden ist

So speichern Sie die Umrichterparameter:

1. Setzen Sie #0.00=1000 (Parameter Null in jedem Menü bei Einsatz der Umrichter-Bedieneinheit).
2. Setzen Sie #10.38=100 (das Drücken der Reset-Taste auf der Bedieneinheit des Umrichters wird simuliert).

### 5.3 Konfigurationsparameter

Die grundlegenden Konfigurationsparameter werden in dem entsprechenden Menü für den Steckplatz gespeichert, an dem das SM-Applications-Modul angebracht ist.

Steckplatz	Menü
1	15
2	16
3	17

Zusätzlich zu diesen Menüs ist ein Alias des entsprechenden Menüs als lokales Menü 81 innerhalb des SM-Applications-Moduls verfügbar. Der Zugang zu diesem Menü ist über das benutzerdefinierte DPL-Programm oder über die Kommunikation (CTNet/CT-RTU/RS485) möglich. Es stellt eine bequeme Methode dar, um die Konfigurationsparameter abzufragen oder zu ändern, ohne wissen zu müssen, an welchem Steckplatz das SM-Applications-Modul angebracht ist.



**WARNUNG**

Soweit nicht anders angegeben, werden diese Parameter nur beim ersten Einschalten des SM-Applications-Moduls, bei einem Reset oder bei dem DPL-Befehl *REINIT* abgefragt. Das Ändern eines Parameters während des Betriebs hat keine unmittelbaren Auswirkungen.

Um das SM-Applications-Modul über die Anzeige des Umrichters zurückzusetzen, geben Sie in Parameter Null jedes Menüs den Wert 1070 ein, und drücken Sie die Reset-Taste.

**HINWEIS**

In dieser Betriebsanleitung werden die Konfigurationsparameter als „#81.xx“ bezeichnet. Verwenden Sie beim Einstellen von Parametern direkt an der Bedieneinheit des Umrichters stattdessen das entsprechende Menü (15, 16 oder 17).

**HINWEIS**

Die für einen Parameter angegebene Aktualisierungsrate bezieht sich beim Lesen auf die Rate, mit der der Parameter aktualisiert wird, oder beim Schreiben auf den Zeitpunkt, zu dem der neue Wert wirksam wird. Die Angabe „Initialisierung“ bedeutet, dass der Parameter nur bei einem Modul-Reset oder bei dem DPL-Befehl *REINIT* abgefragt wird.



**TIPP**

Durch eine Änderung des Umrichtermodus werden alle Konfigurations- und Anwendungsparameter sowie Umrichterparameter auf ihre Standardwerte zurückgesetzt. Dies kann vermieden werden, indem in Parameter Null der Code **1255** verwendet wird statt des normalen Codes **1253**. Nur Umrichterparameter werden auf die Standardwerte zurückgesetzt, und die Menüs 15 bis 20 bleiben unverändert.



### 5.3.1 Beschreibung der Parameter

#81.01	Modulcode		
Zugang	NL	Bereich	0 bis 499
Standard	n.b.	Aktualisierungsrate	n.b.

Mit dem Modulcode wird der Typ des an dem entsprechenden Steckplatz angebrachten Moduls angezeigt. Der Code für ein SM-Applications-Modul lautet **301**.

#81.02	Firmware - Hauptversion		
Zugang	NL	Bereich	00,00 bis 99,99
Standard	n.b.	Aktualisierungsrate	n.b.

Die Hauptversionsnummer des Betriebssystems für das SM-Applications-Modul wird angegeben. Zusammen mit #81.51 ergibt dieser Parameter die vollständige Versionsnummer.

#81.03	DPL-Programmstatus		
Zugang	NL	Bereich	0 bis 3
Standard	0	Aktualisierungsrate	1 ms Änderung

Der Ausführungsstatus des benutzerdefinierten DPL-Programms im SM-Applications-Modul wird angegeben. Die folgenden Werte sind definiert:

Anzeige	Wert	Beschreibung
nonE	0	Kein DPL-Programm vorhanden
StoP	1	DPL-Programm angehalten
run	2	DPL-Programm läuft
triP	3	Laufzeitfehler. ERROR-Task läuft oder DPL-Programm angehalten

#81.04	Verfügbare Systemressource		
Zugang	NL	Bereich	0 bis 100
Standard	n.b.	Aktualisierungsrate	200 ms

Die freien CPU-Ressourcen werden als Prozentsatz der aktuellen Hintergrund-Ausführungszeit angezeigt, berechnet über 200 ms.

#81.05	RS485-Adresse		
Zugang	LS	Bereich	0 bis 255
Standard	11	Aktualisierungsrate	Initialisierung

Die Adresse dieses Knotens für die Kommunikationsprotokolle ANSI und Modbus wird definiert.

Beim ANSI-Protokoll umfasst der Adressenbereich die Werte 11 bis 99, wobei die erste Ziffer die Gruppenadresse und die zweite Ziffer die Nummer der Einheit ist. Beide Ziffern müssen zwischen 1 und 9 liegen. Die Null ist nicht zulässig, da sie vom Master zum Adressieren von Knotengruppen verwendet wird.

Dieser Parameter hat in den RS485-Betriebsarten 25 (CTSync) und 26 (CTSync) keine Auswirkungen.

#81.06	RS-485-Betrieb		
Zugang	LS	Bereich	0 bis 255
Standard	1	Aktualisierungsrate	Initialisierung

Die Betriebsart (das Protokoll) für den eingebauten RS485-Anschluss wird definiert. Details zu diesen Betriebsarten finden Sie in Tabelle 6.1 *Serielle Modi - Parameter #81.06*.

#81.07	RS-485-Baudrate		
<b>Zugang</b>	LS	<b>Bereich</b>	0 bis 9 (300 bis 115200 Bit/s)
<b>Standard</b>	4 (4800)	<b>Aktualisierungsrate</b>	Initialisierung

Die Baudrate (Bit pro Sekunde) für den eingebauten RS485-Anschluss wird definiert. Die folgenden Optionen werden unterstützt:

Anzeige	Wert	Beschreibung
300	0	300 Bit/s
600	1	600 Bit/s
1200	2	1200 Bit/s
2400	3	2400 Bit/s
4800	4	4800 Bit/s
9600	5	9600 Bit/s
19200	6	19200 Bit/s
38400	7	38400 Bit/s
57600	8	57600 Bit/s
115200	9	115200 Bit/s

Beachten Sie, dass dieser Parameter nicht relevant ist, wenn die Betriebsart des RS485-Anschlusses auf 25 (CTSync-Master) oder 26 (CTSync-Slave) eingestellt ist.

#81.08	RS485-Antwortverzögerung		
<b>Zugang</b>	LS	<b>Bereich</b>	0 bis 255 ms
<b>Standard</b>	2 ms	<b>Aktualisierungsrate</b>	Initialisierung

Eine feste Verzögerung zwischen dem Empfangen einer Meldung am RS485-Anschluss und dem Senden der Antwort wird definiert. Dies kann in 2-Drahtkonfigurationen nützlich sein, in denen eine endliche Zeit benötigt wird, um den Master (Host) vom Sendemodus in den Empfangsmodus umzuschalten. Die Verzögerung dauert immer mindestens 1 ms und kann mit Hilfe dieses Parameters verlängert werden.

#81.09	RS485: Verzögerung für TX Enable		
<b>Zugang</b>	LS	<b>Bereich</b>	0 bis 1 ms
<b>Standard</b>	0 ms	<b>Aktualisierungsrate</b>	Initialisierung

Mit Hilfe dieses Parameters kann eine Verzögerung von 1 ms zwischen der Freigabe des RS485-Senders durch das SM-Applications-Modul und dem eigentlichen Sendebeginn eingeführt werden. Dies sollte nur dann erforderlich sein, wenn festgestellt wird, dass der Empfänger der Sendung einen beschädigten Meldungsbeginn erhält.

#81.10	DPL-Druckerpfad		
<b>Zugang</b>	LS	<b>Bereich</b>	0/1
<b>Standard</b>	0	<b>Aktualisierungsrate</b>	Initialisierung

Mit diesem Parameter wird gesteuert, wohin die Ausgabe des DPL-Befehls PRINT gesendet wird. Bei der Einstellung 0 (Aus) wird die Ausgabe an den Programmier-Client (SYPT Workbench) gesendet, bei der Einstellung 1 (Ein) an den RS485-Anschluss.

#81.11	Clock Task Zykluszeit		
<b>Zugang</b>	LS	<b>Bereich</b>	0 bis 200 ms
<b>Standard</b>	10 ms	<b>Aktualisierungsrate</b>	Initialisierung

Der Koordinierungszeitraum (Taktzeitbasis) für den DPL-Task CLOCK wird in Millisekunden definiert. Durch den Wert Null wird der CLOCK-Task deaktiviert.

**HINWEIS** Bis zur Unidrive SP-Version 01.05.00 betrug die Standardeinstellung für diesen Parameter 10 ms.

#81.12	POS Task Zykluszeit		
Zugang	LS	Bereich	0 bis 6
Standard	0	Aktualisierungsrate	Initialisierung

Die Zeitkoordinierungsrate für die POS-Tasks wird gemäß der zum Ausführen des benutzerdefinierten DPL-Programms benötigten Applikationsleistung und den erforderlichen Ressourcen definiert. Die folgenden Werte sind definiert:

Anzeige	Wert	Beschreibung
diSAbled	0	Deaktiviert
0.25	1	250 $\mu$ s
0.5	2	500 $\mu$ s
1	3	1 ms
2	4	2 ms
4	5	4 ms
8	6	8 ms

Stellen Sie diesen Parameter ein, wenn das benutzerdefinierte DPL-Programm beim Einschalten bzw. Reset automatisch ausgeführt wird. Wenn dies geändert wird und beim Einschalten die neue Einstellung verwendet werden muss, achten Sie darauf, dass die **Umrichterparameter** gespeichert werden.

**HINWEIS** Bis zur Unidrive SP-Version 01.03.00 wird auf dem Display nicht die eigentliche Rate, sondern die dazugehörige Aliaszahl angezeigt. Zum Beispiel wird eine Rate von 250  $\mu$ s auf dem Display als „1“ ausgelöst.

#81.13	Automatisches Ausführen freigeben		
Zugang	LS	Bereich	0/1
Standard	1	Aktualisierungsrate	Initialisierung

#81.14	Globale Laufzeit-Fehlerabschaltung freigeben		
Zugang	LS	Bereich	0/1
Standard	0	Aktualisierungsrate	Initialisierung

Die Einstellung 1 für diesen Parameter führt dazu, dass eine Fehlerabschaltung des Unidrive SP ausgelöst wird, wenn innerhalb des SM-Applications-Moduls bzw. des benutzerdefinierten DPL-Programms bestimmte Laufzeitfehler auftreten.

Weitere Informationen finden Sie in Abschnitt 11.1 *Laufzeitfehler*.

#81.15	Reset nach Zurücksetzen der Fehlerabschaltung deaktivieren		
Zugang	LS	Bereich	0/1
Standard	0	Aktualisierungsrate	Initialisierung

Wenn dieser Parameter gleich 0 ist, wird das Modul nach dem Zurücksetzen einer Umrichter-Fehlerabschaltung ebenfalls zurückgesetzt. Bei der Einstellung 1 hat das Zurücksetzen einer Umrichter-Fehlerabschaltung keine Auswirkungen auf das Modul (d. h. es läuft weiter).

#81.16	Aktualisierungsrate der Encoder-Daten		
Zugang	LS	Bereich	0-3
Standard	0	Aktualisierungsrate	Initialisierung

Wenn dieser Parameter gleich 0 ist, werden die APR-Daten und die Encoder-Parameter in Menü 90 alle 250µs aktualisiert.

Wenn dieser Parameter gleich 1 ist, werden die APR-Daten und die Encoder-Parameter in Menü 90 unmittelbar vor jedem POS-Task aktualisiert.

Wenn dieser Parameter gleich 2 ist, werden die APR-Daten und die Encoder-Parameter in Menü 90 unmittelbar vor jedem CLOCK-Task aktualisiert.

Bei dem Wert 3 werden die APR-Daten und die Encoder-Parameter in Menü 90 nie aktualisiert. Wenn diese Daten nie aktualisiert werden, werden zusätzliche Prozessorressourcen freigesetzt.

#81.17	Fehlerabschaltung beim Überschreiten von Parameterbereichsgrenzen freigeben		
Zugang	LS	Bereich	0/1
Standard	0	Aktualisierungsrate	Initialisierung

Hier wird definiert, welche Maßnahme ergriffen wird, wenn durch ein benutzerdefiniertes DPL-Programm ein außerhalb des zulässigen Bereichs liegender Wert in einen Parameter geschrieben werden soll. Bei der Einstellung 1 wird eine Laufzeit-Fehlerabschaltung ausgelöst (Nummer 44), bei der Einstellung 0 wird der Wert automatisch auf den Höchst- bzw. Mindestwert dieses Parameters begrenzt.

#81.18	Watchdog freigeben		
Zugang	LS	Bereich	0/1
Standard	0	Aktualisierungsrate	Initialisierung

Wenn dieser Parameter eingestellt ist, wird die Watchdog-Funktion für das DPL-Programm freigegeben. Der DPL-Befehl WDOG muss anschließend alle 200 ms ausgeführt werden. Dadurch kann das Programm gegen Funktionsstörungen geschützt werden. Falls der Befehl nicht innerhalb von 200 ms ausgeführt wurde, wird eine Umrücker-Fehlerabschaltung an **SLx.tO** ausgelöst. Bitte beachten Sie, dass der Befehl WDOG außerdem einmal ausgeführt werden muss, um die Watchdog-Funktion freizugeben. Dies geschieht normalerweise am Ende des Initial-Tasks.

#81.19	Speicheranforderung		
Zugang	LS	Bereich	0/1
Standard	0	Aktualisierungsrate	100 ms

Wenn Sie diesen Parameter auf 1 setzen, löst dies eine sofortige Speicherung aller nicht flüchtigen Daten für das SM-Applications-Modul aus. Dies besteht aus den SPS-Registersätzen P, Q, T und U sowie optional aus Menü 20 (abhängig von der Einstellung von #81.21).

**HINWEIS** Beachten Sie, dass dies auch zu einem Reset des Moduls führt, und dieser Parameter wird automatisch auf Null zurückgesetzt. Außerdem wird er bei einer Fehlerabschaltung des Unidrive SP zurückgesetzt. Menü 81 wird nicht gespeichert.

#81.20	Speichern bei „UU“-Fehlerabschaltung freigeben		
Zugang	LS	Bereich	0/1
Standard	0	Aktualisierungsrate	Sofort

Mit der Einstellung 1 für diesen Parameter wird signalisiert, dass alle nicht flüchtigen Daten für das SM-Applications-Modul beim Ausschalten des Umrichters automatisch gespeichert werden.

**HINWEIS** Beachten Sie, dass ein „UU“-Speichervorgang mit einem Reset des SM-Applications-Moduls verbunden ist.

<b>#81.21</b>	<b>Speichern und Wiederherstellen (Menü 20) freigeben</b>		
<b>Zugang</b>	LS	<b>Bereich</b>	0/1
<b>Standard</b>	0	<b>Aktualisierungsrate</b>	Sofort

Bei der Einstellung 1 wird Menü 20 zusammen mit anderen nicht flüchtigen Parametern nach einer Speicheranforderung (#x.19=1) oder einer Speicherung beim Ausschalten (#x.20=1) gespeichert bzw. wiederhergestellt. Wenn Menü 20 beim Einschalten wiederhergestellt werden soll, muss der Benutzer darauf achten, dass dieser Parameter vor dem Ausschalten im Umrichter gespeichert wird.

Da Menü 20 ein globales Umrichter-Menü ist, sollte nur eine am Unidrive SP angebrachte Option verwendet werden, um es zu speichern und wiederherzustellen. Wenn mehrere SM-Applications-Module an dem Umrichter angebracht sind, sollte **nur für eines** davon dieser Parameter eingestellt werden. Andernfalls wird Menü 20 beim Einschalten nicht richtig wiederhergestellt.

**HINWEIS** Im Gegensatz zu anderen Konfigurationsparametern werden die Parameter #81.20 und #82.21 **nicht zwischengespeichert**, d. h. eine Änderung an dem Parameter wird sofort wirksam.

<b>#81.22</b>	<b>CTNet-Token Ring-Kennung</b>		
<b>Zugang</b>	LS	<b>Bereich</b>	0 bis 255
<b>Standard</b>	0	<b>Aktualisierungsrate</b>	Initialisierung

Mit Hilfe dieses Parameters kann der Benutzer die Kennung für den CTNet-Token Ring angeben, mit dem ein SM-Applications-Modul verbunden ist. In einem System mit einem einzelnen Token Ring kann für diesen Parameter der Standardwert übernommen werden. In einem System mit mehreren Token Rings sollten separate Kennungen für jeden Ring eingestellt werden. Die Kombination aus CTNet-Token Ring-Kennung und CTNet-Knotenadresse muss eindeutig sein.

<b>#81.23</b>	<b>CTNet-Knotenadresse</b>		
<b>Zugang</b>	LS	<b>Bereich</b>	0 bis 255
<b>Standard</b>	0	<b>Aktualisierungsrate</b>	Initialisierung

Die CTNet-Knotenadresse wird definiert. Jeder Knoten in einem CTNet-Netzwerk muss eine eindeutige Adresse besitzen. Mit der Einstellung Null wird CTNet für den entsprechenden Knoten deaktiviert.

<b>#81.24</b>	<b>CTNet-Baudrate</b>		
<b>Zugang</b>	LS	<b>Bereich</b>	0 bis 3
<b>Standard</b>	1(2.5)	<b>Aktualisierungsrate</b>	Initialisierung

Die CTNet-Datenrate wird angegeben. Alle Knoten im Netzwerk müssen auf dieselbe Datenrate eingestellt sein. Die Raten sind folgendermaßen definiert:

Anzeige	Wert	Beschreibung
5.000	0	5 MBit/s
2.500	1	2,5 MBit/s
1.250	2	1,25 MBit/s
0.625	3	625 kBit/s

<b>#81.25</b>	<b>CTNet-Synchronisationseinrichtung</b>		
<b>Zugang</b>	LS	<b>Bereich</b>	0 bis 9999
<b>Standard</b>	0	<b>Aktualisierungsrate</b>	Initialisierung

Die Generierungsrate für CTNet-Synchronisationsmeldungen wird angegeben. Durch diese Meldung wird allen Knoten mitgeteilt, wann zyklische Daten gesendet werden müssen. Dieser Parameter sollte nur für einen Knoten im CTNet-Netzwerk eingestellt werden.

Das Format des Aktualisierungsparameters lautet LLSS, wobei SS für die Aktualisierungsrate des schnellen zyklischen Datenkanals steht und LL für die langsame zyklische Datenrate in Vielfachen von SS. Wenn also der Parameterwert 1510 eingestellt wird, erfolgt die schnelle zyklische Datenaktualisierung alle 10 ms und die langsame alle 150 ms. Beim Arbeiten mit Easy Mode (siehe unten) muss nur die schnelle zyklische Rate eingestellt werden.

<b>#81.26, #81.28, #81.30</b>	<b>CTNet Easy Mode-Konfigurationsparameter</b>		
<b>Zugang</b>	LS	<b>Bereich</b>	0 bis 25503
<b>Standard</b>	0	<b>Aktualisierungsrate</b>	Initialisierung

<b>#81.27, #81.29, #81.31- #81.34</b>	<b>CTNet Easy Mode-Konfigurationsparameter</b>		
<b>Zugang</b>	LS	<b>Bereich</b>	0 bis 9999
<b>Standard</b>	0	<b>Aktualisierungsrate</b>	Initialisierung

Mit diesen Parametern werden Quelle und Ziele für zyklische CTNet Easy Mode-Daten definiert.

Parameter	Format	Kanal	Beschreibung
#81.26	KKKSS	1	Zielknotennummer und -steckplatz werden definiert KKK = Knotennummer (1 bis 255) SS = Steckplatznummer (1 bis 3) Beispiel: Ein Wert von 201 bedeutet Knotenkennung 2 und Steckplatz 1.
#81.27	MMPP	1	Der zu sendende Quell-Umrichterparameter wird definiert MM = Menünummer PP = Parameternummer Beispiel: Ein Wert von 302 bedeutet #3.02 (Drehzahl)
#81.28	KKKSS	2	Zielknotennummer und -steckplatz für Kanal 2
#81.29	MMPP	2	Quell-Umrichterparameter für Kanal 2
#81.30	KKKSS	3	Zielknotennummer und -steckplatz für Kanal 3
#81.31	MMPP	3	Quell-Umrichterparameter für Kanal 3
#81.32	MMPP	1	Zielparameter für eingehende Daten an Steckplatz 1
#81.33	MMPP	2	Zielparameter für eingehende Daten an Steckplatz 2
#81.34	MMPP	3	Zielparameter für eingehende Daten an Steckplatz 3

Weitere Informationen zu CTNet Easy Mode-Daten finden Sie in der CTNet-Betriebsanleitung.

<b>#81.35</b>	<b>CTNet-Synchronisation: Event-Task-Kennung</b>		
<b>Zugang</b>	LS	<b>Bereich</b>	0 bis 4
<b>Standard</b>	0	<b>Aktualisierungsrate</b>	Initialisierung

Mit diesem Parameter wird angegeben, welcher der EVENT-Tasks geplant werden soll, wenn eine CTNet-Synchronisationsmeldung empfangen oder generiert wird. Diese Synchronisation wird von einem *Master*-Knoten im CTNet-Netzwerk (was dieser Knoten sein kann) auf einer festen Zeitbasis generiert. Die folgenden Werte sind definiert:

Anzeige	Wert	Beschreibung
Disabled	0	Keine Zeitkoordinierung für Event-Tasks
Event	1	Zeitkoordinierung EVENT-Task
Event1	2	Zeitkoordinierung EVENT1-Task
Event2	3	Zeitkoordinierung EVENT2-Task
Event3	4	Zeitkoordinierung EVENT3-Task

#81.36	CTNet-Fehlersuche		
Zugang	NL	Bereich	-3 bis 32767
Standard	n.b.	Aktualisierungsrate	1 Sekunde

Der Status des CTNet-Netzwerks wird im Parameter „CTNet-Fehlersuche“ angezeigt. Wenn die Kommunikation des SM-Applications-Moduls im CTNet-Netzwerk erfolgreich ist, wird die Anzahl der Meldungen pro Sekunde angezeigt

**Tabelle 5.1 CTNet-Fehlersuche**

#MM.36	Status	Beschreibung
>0	Netzwerk betriebsbereit	Die Anzahl der pro Sekunde zu verarbeitenden Meldungen wird angezeigt.
0	Netzwerk betriebsbereit, keine Datenübertragung	Das Niederpegel-Token Ring-Netzwerk wurde eingerichtet und ist aktiv, jedoch werden vom Knoten keine CTNet-Datenmeldungen empfangen.
-1	RECON	Eine Neukonfiguration des Netzwerks wurde erkannt.
-2	Initialisierungsfehler	Die CTNet-Schnittstelle konnte vom SM-Applications-Modul nicht konfiguriert werden. Überprüfen Sie, ob Knotenadresse und Datenrate richtig eingestellt sind.
-3	MYRECON	Eine Neukonfiguration des CT-Netzwerks wurde vom SM-Applications-Modul erzwungen.

#81.37	Laden bei freigegebenem Umrichter verweigern		
Zugang	LS	Bereich	0/1
Standard	0	Aktualisierungsrate	Initialisierung

Wenn dieser Parameter eingestellt ist, wird bei einem Versuch des Benutzers, ein neues benutzerdefiniertes DPL-Programm oder ein neues Betriebssystem in dieses Modul zu laden, und bei freigegebenem Umrichter das Laden verweigert, und eine Laufzeit-Fehlerabschaltung (Nummer 70) wird ausgelöst, vorausgesetzt, der globale Laufzeit-Fehlerabschaltungsparameter (#81.14) ist eingestellt.

Da durch einen Ladevorgang der normale Betrieb des Moduls angehalten wird, kann es als unsicher angesehen werden, dies bei laufendem Umrichtersystem zu tun. Daher kann durch Einstellen dieses Parameters ein Laden unter dieser Bedingung verhindert werden.

#81.38	APR-Laufzeit-Fehlerabschaltung		
Zugang	LS	Bereich	0/1
Standard	0	Aktualisierungsrate	Initialisierung

Wenn dieser Parameter gleich 0 ist, wird im Falle eines irreparablen APR-Fehlers, zum Beispiel Verwendung einer nicht initialisierten CAM-Funktion, eine Umrichter-Fehlerabschaltung mit Laufzeitfehler 81 ausgelöst. Wenn dieser Parameter gleich 1 ist, wird im Falle eines irreparablen APR-Fehlers keine Umrichter-Fehlerabschaltung ausgelöst.

#81.39	Synchronisationsstatus zwischen SM-Applications-Umrichtern		
Zugang	LS	Bereich	0/1
Standard	0	Aktualisierungsrate	n.b.

In diesem Parameter wird der Synchronisationsstatus des aktuellen Moduls angezeigt. Es wird angezeigt, ob das Modul als Synchronisationsmaster fungiert:

Synchronisationsstatus	Status
0	Die Synchronisationsmaster-Abfrage ist Null, oder ein anderes Optionsmodul ist Synchronisationsmaster.
1	Das Optionsmodul ist Synchronisationsmaster.
3	Das Optionsmodul ist Synchronisationsmaster, jedoch liegt die Synchronisationsfrequenz außerhalb der Spezifikation oder ist nicht vorhanden.

#81.42	Freeze: Position Istwert-Encoder		
Zugang	LS	Bereich	0/1
Standard	0	Aktualisierungsrate	Initialisierung

Bei einem Freeze im SM-Applications-Modul können Position und Umdrehungszähler des Istwert-Encoders gespeichert werden, wenn dieser Parameter auf 1 gesetzt ist. Diese Werte werden in den Parametern #90.19 und #90.20 gespeichert (siehe Abschnitt 5.8 *Menü 90 - allgemeine Parameter*). Weitere Informationen zum Freeze-Eingang finden Sie in Kapitel 8 *SM-Applications-Freeze und -Nullimpuls*.

#81.43	Freeze: Invertierung		
Zugang	LS	Bereich	0/1
Standard	0	Aktualisierungsrate	Initialisierung

Wenn dieser Parameter auf Null gesetzt ist, erfolgt ein Freeze an der aufsteigenden Flanke des SM-Applications-Signals DIGIN0. Bei der Einstellung 1 erfolgt ein Freeze an der abfallenden Flanke des SM-Applications-Signals DIGIN0. Weitere Informationen zum Freeze-Eingang finden Sie in Kapitel 8 *SM-Applications-Freeze und -Nullimpuls*.

#81.44	Task-Prioritätsebene		
Zugang	LS	Bereich	0 bis 255
Standard	0	Aktualisierungsrate	Initialisierung

Die Prioritätsebenen für verschiedene Tasks können mit diesem Parameter geändert werden. Der Zugang zu diesem Parameter erfolgt in den einzelnen Bits:

Bit	Bedeutung
0	Der Wert 0 bedeutet dieselben Task-Prioritätsebenen wie beim UD70. Durch Einstellen des Werts auf 1 wird die Priorität des CTNet-Tasks niedriger eingestuft als die von POS0 und POS1. Dadurch wird die Schwankung der POS-Tasks verringert, jedoch kann möglicherweise der CTNet-Task nicht mehr versorgt werden.
1	Der Wert 0 bedeutet, dass die Priorität der Tasks für die Kommunikation zwischen Optionen höher ist als die Priorität der POS-Tasks. Durch Einstellen des Werts 1 wird die Priorität der Tasks für die Kommunikation zwischen Optionen niedriger als die Priorität der POS-Tasks.



<b>#81.48</b>	<b>Zeilennummer Fehler</b>		
<b>Zugang</b>	NL	<b>Bereich</b>	32 Bit
<b>Standard</b>	0	<b>Aktualisierungsrate</b>	Bei Fehler

Die DPL-Programmzeilennummer, die einen Laufzeitfehler verursacht hat, wird angegeben. Dies gilt nur in den folgenden Fällen:

- Das Benutzerprogramm wurde mit eingestellter *Debug*-Option kompiliert
- Der Fehler kann durch Benutzercode generiert werden, zum Beispiel Division durch Null (50) oder nicht vorhandener Parameter (41)

Wenn diese Bedingungen beide nicht erfüllt sind, wird für den Zeilennummernparameter der Wert Null (0) angezeigt.

<b>#81.49</b>	<b>Kennung für Benutzerprogramme</b>		
<b>Zugang</b>	NL/LS	<b>Bereich</b>	16 Bit mit Vorzeichen
<b>Standard</b>	0	<b>Aktualisierungsrate</b>	Siehe Hinweis

Dieser Parameter ist verfügbar, damit der Benutzer einen Kennungscode für sein Programm eingeben kann. Dies kann zum Beispiel die Software-Versionsnummer sein. Verwenden Sie zum Schreiben in diesen Parameter den Funktionsblock SETUSERID().

<b>#81.50</b>	<b>Laufzeitfehlercode</b>		
<b>Zugang</b>	NL	<b>Bereich</b>	0 bis 255
<b>Standard</b>	0	<b>Aktualisierungsrate</b>	Bei Fehler

Wenn ein Laufzeitfehler auftritt, wird die Fehlernummer in diesem Parameter platziert. Weitere Informationen finden Sie in Kapitel 11.1 *Laufzeitfehler*.

<b>#81.51</b>	<b>Firmware - Nebenversion</b>		
<b>Zugang</b>	NL	<b>Bereich</b>	0 bis 99
<b>Standard</b>	n.b.	<b>Aktualisierungsrate</b>	n.b.

Die Nebenversionsnummer des Betriebssystems für das SM-Applications-Modul wird angegeben. Zusammen mit #81.02 ergibt dieser Parameter die vollständige Versionsnummer.

## 5.4 Menüs 70 bis 75 - SPS-Register

Diese Menüs ermöglichen den Zugang zu den SPS-Registern. Die SPS-Register sind ganze 32 Bit-Zahlen mit Vorzeichen, die für Benutzerprogramme und CTNet-Kommunikation zur Verfügung stehen.

Die SPS-Register sind in 6 Sätze zu je 100 Parametern mit den Nummern 00 bis 99 aufgeteilt. Der Zugang zu den Registern ist auch aus einem benutzerdefinierten DPL-Programm heraus durch einen speziellen Variablennamen oder Array-Namen möglich.

Menünummer	Zugang	DPL-Variablen (x = Registernummer)	Anzahl der Register	Beschreibung
70	LS	_Px%, _P%[x]	100	Universal. Speicherbar.
71	LS	_Qx%, _Q%[x]	100	Universal. Speicherbar.
72	LS	_Rx%, _R%[x]	100	Für abgehende zyklische CTNet-Datenverbindungen verwendet. Nicht speicherbar.
73	LS	_Sx%, _S%[x]	100	Für eingehende zyklische CTNet-Datenverbindungen verwendet. Nicht speicherbar.
74	LS	_Tx%, _T%[x]	100	Universal. Speicherbar.
75	LS	_Ux%, _U%[x]	100	Universal. Speicherbar.

In der Tabelle oben sehen Sie, dass jeder Parameter in den Menüs 70 bis 75 eine äquivalente DPL-Variable besitzt. Dies bedeutet, dass Sie beide Formate verwenden können, um auf einen Parameter in diesen Menüs zuzugreifen.

Beispiel: Die Einstellung **#72.01=1** bewirkt dasselbe wie die Einstellung **\_R01%=1**, die Einstellung **#75.65=66** dasselbe wie die Einstellung **\_U65%=66** usw.

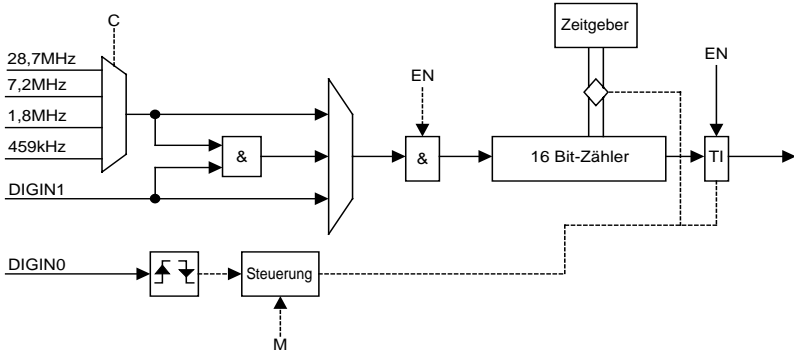
Die Menüs 70, 71, 74 und 75 können alle im nicht flüchtigen Flash-Speicher gespeichert werden, entweder auf Anforderung oder automatisch beim Umstellen des Umrichters auf Unterspannung. (Weitere Informationen finden Sie in Abschnitt 5.2 *Speichern der Parameter.*)

Die Menüs 72 und 73 werden für die zyklische CTNet-Datenübertragung verwendet. Wenn diese Funktion jedoch nicht eingesetzt wird, können die Register für einen beliebigen anderen Zweck verwendet werden. Dies sollte jedoch möglichst vermieden werden, für den Fall, dass zyklische Daten in Zukunft verwendet werden.

## 5.5 Menü 85 - Zeitgeberfunktionsparameter

Eine Hardware/Zähler mit den folgenden Merkmalen ist in das SM-Applications-Modul eingebaut:

- Zähler in Schritten von 16 Bit.
- Aus dem internen Takt auswählbare Zählerrate. Der Divisor für die Taktrate kann unter den Werten Rate/1, Rate/4, Rate/16 und Rate/64 ausgewählt werden.
- Aus einem externen Takt über den Digitaleingang DIGIN1 kann eine Taktrate ausgewählt werden. Die maximale Taktrate beträgt 600 kHz.
- Der Zeitgeber kann für die Zeitkoordinierung einer der 4 DPL-Event-Tasks bei Umlauf oder bei einer Eingangserfassung auf DIGIN1 verwendet werden.
- Der Zählerüberlauf kann vom Benutzer bis zum maximalen Bereich von 16 Bit ausgewählt werden, der für den Zähler verfügbar ist.
- Der Zeitgeber kann so eingestellt werden, dass der Zählerwert am Übergang zur aufsteigenden bzw. abfallenden Flanke von DIGIN0 zwischengespeichert wird.



#85.01	Steuerwort		
Zugang	LS	Bereich	13 Bit
Standard	n.b.	Aktualisierungsrate	Sofort

**Tabelle 5.2 Steuerwort - Parameter #85.01**

Bit	Symbol	Funktion
b0-b2	TE	Zeitgeber EVENT-Task bei gesetztem TI-Flag: 0 = Keine Zeitkoordinierung für Event-Tasks 1 = Zeitkoordinierung Event-Task 2 = Zeitkoordinierung Event1-Task 3 = Zeitkoordinierung Event2-Task 4 = Zeitkoordinierung Event3-Task
b3	EN	Zeitgeber freigeben: 0 = Zeitgeber is deaktiviert 1 = Zeitgeber ist freigegeben
b4	CS	Taktquelle: 0 = interner Takt 1 = externer Takt über DIGIN1
b5-b6	C	Auswahl der Vorkalierung für den internen Takt (ignorieren, falls externer Takt ausgewählt): 0 = Rate/1 (28,7 MHz) 1 = Rate/4 2 = Rate/16 3 = Rate/64
b7-b8	M	Zeitgebermodus: <u>Modus 0 (00): Freilaufmodus</u> Der Zähler wird von dem ausgewählten Takt angetrieben. Das TI-Flag wird beim Umlauf gesetzt. <u>Modus 1 (01): Erfassungsmodus 1</u> Der Zähler wird von dem ausgewählten Takt angetrieben. Ein Übergang zur aufsteigenden Flanke von DIGIN0 führt dazu, dass der aktuelle Zählerwert mit dem Parameter „Cache Zeitgebererfassung“ verknüpft und das TI-Flag gesetzt wird. Anschließend wird der Zähler weiterhin schrittweise erhöht. (Das TI-Flag wird beim Umlauf nicht gesetzt.) <u>Modus 2 (10): Erfassungsmodus 2</u> Der Zähler wird von dem ausgewählten Takt angetrieben. Ein Übergang zur abfallenden Flanke von DIGIN0 führt dazu, dass der aktuelle Zählerwert mit dem Parameter „Cache Zeitgebererfassung“ verknüpft und das TI-Flag gesetzt wird. Anschließend wird der Zähler weiterhin schrittweise erhöht. (Das TI-Flag wird beim Umlauf nicht gesetzt.)

#85.02	Status-Datenwort		
Zugang	NL	Bereich	0 bis 3
Standard	n.b.	Aktualisierungsrate	Sofort

**Tabelle 5.3 Status-Datenwort - Parameter #85.02**

Bit	Symbol	Funktion
b0	TI	Zeitgeber-Ereignisflag: 0 = Kein Ereignis eingetreten 1 = Ereignis eingetreten (siehe Beschreibung für Parameter #85.01)  <b>Hinweis:</b> Das TI-Bit wird vom Betriebssystem automatisch zurückgesetzt, wenn das TE-Flag in #85.01 nicht auf Null gesetzt ist. Andernfalls wird es beim Abfragen des Status-Datenworts zurückgesetzt.
b1	OV	Umlauf-Flag 0 = Umlauf ist nicht eingetreten 1 = Zählerumlauf ist eingetreten  <b>Hinweis:</b> Dieses Flag ist für ALLE Zeitgebermodi gültig und wird beim Abfragen des Statusregisters automatisch zurückgesetzt.

<b>#85.03</b>	<b>16 Bit-Zeitgeberzähler</b>		
<b>Zugang</b>	LS	<b>Bereich</b>	16 Bit
<b>Standard</b>	n.b.	<b>Aktualisierungsrate</b>	Sofort

Mit Hilfe dieses Parameters kann der aktuelle Zeitgeberwert jederzeit abgefragt und geschrieben werden.

<b>#85.04</b>	<b>Umlaufgrenze</b>		
<b>Zugang</b>	LS	<b>Bereich</b>	16 Bit
<b>Standard</b>	n.b.	<b>Aktualisierungsrate</b>	Sofort

Mit diesem Parameter wird der Wert angegeben, bei dem ein Umlauf von Parameter #85.03 erfolgt.



**WARNUNG**

Wenn Sie bei schnellen Zeitgeberraten eine kleine Umlaufzahl einstellen, könnte dies dazu führen, dass das SM-Applications-Modul scheinbar blockiert. Dies liegt daran, dass die Umlaufunterbrechung ständig bedient wird und den anderen Produktfunktionen Prozessorressourcen entzogen werden. In diesem Fall kann der Benutzer das SM-Applications-Modul zurücksetzen, indem er ein 1070-Reset des Umrichters durchführt. Wenn die Zahlen für den Zeitgeber durch ein DPL-Programm eingestellt werden, sollte vor dem Reset die automatische Ausführung des Programms deaktiviert werden (#17.13=0). Nach dem Reset kann der Benutzer den Zeitgeber mit einer geeigneteren Umlaufzahl neu initialisieren.

<b>#85.05</b>	<b>Cache Zeitgebererfassung</b>		
<b>Zugang</b>	LS	<b>Bereich</b>	16 Bit
<b>Standard</b>	n.b.	<b>Aktualisierungsrate</b>	Sofort

Mit Hilfe dieses Parameters wird der zwischengespeicherte Wert gespeichert, wenn der Zeitgeber in Modus 1 oder 2 (den Erfassungsmodi) verwendet wird.

## 5.6 Menü 86 - Digital-E/A-Parameter

In das SM-Applications-Modul sind zwei Digitalausgänge und zwei Digitaleingänge eingebaut. Diese Aus- und Eingänge werden durch dieses Menü gesteuert.

<b>#86.01</b>	<b>Digitaleingang 0</b>		
<b>Zugang</b>	NL	<b>Bereich</b>	0/1
<b>Standard</b>	n.b.	<b>Aktualisierungsrate</b>	Sofort

<b>#86.02</b>	<b>Digitaleingang 1</b>		
<b>Zugang</b>	NL	<b>Bereich</b>	0/1
<b>Standard</b>	n.b.	<b>Aktualisierungsrate</b>	Sofort

Mit diesen zwei Parametern wird der Zustand der Digitaleingänge 0 und 1 abgefragt. Ein inaktiver Eingang (niedrig 0 V) ergibt den Wert 0 und ein aktiver Eingang (hoch +24 V) den Wert 1.

<b>#86.03</b>	<b>Digitalausgang 0</b>		
<b>Zugang</b>	LS	<b>Bereich</b>	0/1
<b>Standard</b>	n.b.	<b>Aktualisierungsrate</b>	Sofort

<b>#86.04</b>	<b>Digitalausgang 1</b>		
<b>Zugang</b>	LS	<b>Bereich</b>	0/1
<b>Standard</b>	n.b.	<b>Aktualisierungsrate</b>	Sofort

Mit diesen zwei Parametern werden die Digitalausgänge 0 und 1 gesteuert. Mit der Einstellung 0 wird der Ausgang niedrig (0 V) und mit der Einstellung 1 der Eingang hoch (+24 V) eingestellt.

#86.05	Digitalausgänge 0 und 1		
<b>Zugang</b>	LS	<b>Bereich</b>	0 bis 3
<b>Standard</b>	n.b.	<b>Aktualisierungsrate</b>	Sofort

Dieser Parameter ermöglicht eine Steuerung beider Digitalausgänge und stellt eine Alternative zu der Methode dar, mit Hilfe der Parameter #86.03 und #86.04 jeden Ausgang einzeln zu steuern. Mit Bit0 dieses Parameters wird Digitalausgang 0 (#86.03) gesteuert und mit Bit1 Digitalausgang 1 (#86.04).

## 5.7 Menü 88 - Statusparameter

#88.01	Fehlercode/Reset		
<b>Zugang</b>	LS	<b>Bereich</b>	0 bis 9999
<b>Standard</b>	n.b.	<b>Aktualisierungsrate</b>	Bei Fehler

Dieser Parameter dient zu zwei Zwecken - wenn er abgefragt wird, wird derselbe Laufzeitfehlercode wie bei #81.50 zurückgegeben. (Beachten Sie, dass keine Umrichter-Fehlerabschaltungs-codes zurückgegeben werden.) Der Parameter wird auf Null zurückgesetzt, wenn ein Reset durchgeführt wird und wenn die Ausführung des Benutzerprogramms gestartet wird.

Wenn in den Parameter ein Wert von 1070 geschrieben wurde, wird vom SM-Applications-Modul ein Warm-Neustart des Umrichters und aller anderen Optionen ausgelöst. Damit können das Benutzerprogramm neugestartet (Bereitstellung des automatischen Programmstarts #81.13=1) und jede Umrichter-Fehlerabschaltung zurückgesetzt werden. Diese Reset-Maßnahme kann jederzeit durchgeführt werden, nicht nur nach einem Laufzeitfehler oder in einem ERROR-Task.



Wenn der Wert 1070 in Parameter #88.01 geschrieben wird, führt dies dazu, dass eine beliebige Umrichter-Fehlerabschaltung automatisch zurückgesetzt wird und alle Optionen im Unidrive SP zurückgesetzt werden. Dieses Verhalten ist anders als bei dem UD70-Produkt am Unidrive 1, bei dem der Umrichter nicht zurückgesetzt wurde.

#88.02	Task mit Fehler		
<b>Zugang</b>	NL	<b>Bereich</b>	0 bis 50
<b>Standard</b>	n.b.	<b>Aktualisierungsrate</b>	Bei Fehler

Mit Hilfe des Parameters „Task mit Fehler“ kann der Task ermittelt werden, in dem der Fehler generiert wurde. Dieser Parameter ist nur dann gültig, wenn er nach einer Laufzeit-Fehlerabschaltung aus dem ERROR-Task abgefragt wird. Die Werte haben die folgenden Bedeutungen:

Wert	Task
50	System
1	Initial
2	Background
3	Clock
4	Error
5	Pos0
6	Pos1
7	Event

Wert	Task
8	Event1
9	Event2
10	Event3

Ein Wert von Null wird zurückgegeben, wenn kein Fehlerzustand vorliegt.

Weitere Informationen zu diesen Parametern finden Sie in Kapitel 11 *Fehlersuche* auf Seite 85.

## 5.8 Menü 90 - allgemeine Parameter

Dieses Menü enthält die Soll- und Istwerte vom Umrichter sowie weitere Statusinformationen.

**HINWEIS** Beim Portieren von UD70-Programmen in SM-Applications muss besonders sorgfältig vorgegangen werden, da diese Parameter vom UD70 abweichen.

#90.01	Istwert-Encoderposition ( $2^{32}/U$ )		
Zugang	NL	Bereich	32 Bit mit Vorzeichen
Standard	n.b.	Aktualisierungsrate	Siehe #81.16

Enthält die Position des Istwert-Encoders.

Die oberen 16 Bits sind auf 65536 Zählerwerte pro Umdrehung skaliert, unabhängig vom Typ des Istwert-Geräts und von der im Umrichter konfigurierten Skalierung. Mit den unteren 16 Bits wird die Feinposition angegeben, wie sie von dem auf 65536 skalierten Istwert-Gerät geliefert wird. Bei Standard-Encodern ist dies normalerweise Null, jedoch ist für Geräte mit höherer Genauigkeit, zum Beispiel SinCos-Encoder, diese zusätzliche Präzision verfügbar.

Nullimpulse usw. haben keinen Einfluss auf diesen Parameter.



Weitere Informationen zur Verwendung dieser Istwertparameter finden Sie in der Online-Hilfe der SYPT Workbench.

#90.02	Umdrehungszähler Istwert-Encoder		
Zugang	NL	Bereich	16 Bit ohne Vorzeichen
Standard	n.b.	Aktualisierungsrate	Siehe #81.16

Enthält den Umdrehungszähler des Istwert-Encoders.

#90.03	Sollwert-Encoderposition ( $2^{32}/U$ )		
Zugang	NL	Bereich	32 Bit mit Vorzeichen
Standard	n.b.	Aktualisierungsrate	Siehe #81.16

Enthält die Position des Sollwert-Encoders.

Die oberen 16 Bits sind auf 65536 Zählerwerte pro Umdrehung skaliert, unabhängig vom Typ des Istwert-Geräts und von der im Umrichter konfigurierten Skalierung. Mit den unteren 16 Bits wird die Feinposition angegeben, wie sie von dem auf 65536 skalierten Istwert-Gerät geliefert wird. Bei Standard-Encodern ist dies normalerweise Null, jedoch ist für Geräte mit höherer Genauigkeit, zum Beispiel SinCos-Encoder, diese zusätzliche Präzision verfügbar.

Nullimpulse usw. haben keinen Einfluss auf diesen Parameter.

#90.04	Umdrehungszähler Sollwert-Encoder		
<b>Zugang</b>	NL	<b>Bereich</b>	16 Bit ohne Vorzeichen
<b>Standard</b>	n.b.	<b>Aktualisierungsrate</b>	Siehe #81.16

Enthält den Umdrehungszähler des Sollwert-Encoders.

#90.10	Umrichtermodus		
<b>Zugang</b>	NL	<b>Bereich</b>	16 Bit mit Vorzeichen
<b>Standard</b>	n.b.	<b>Aktualisierungsrate</b>	Sofort

Stellt eine definitive Methode zur Bestimmung des Modus dar, in dem sich der Unidrive SP befindet. Es wird empfohlen, diesen Parameter an Stelle von #11.31 oder #0.48 zu verwenden, da mit diesen Parametern der angeforderte Modus angegeben wird, nicht der tatsächliche.

Die Werte sind folgendermaßen definiert:

Wert	Modus
26	Open Loop-Modus
27	Closed Loop-Vektormodus
28	Servomodus
29	Betrieb als Netzwechselrichter
20	Commander SE-Modus

Wenn Sie den Umrichtermodus über das Programm ändern möchten, verwenden Sie den Funktionsblock MODEXFER oder CMODEXFER.

#90.11	Umrichterstatus und Steuerwort		
<b>Zugang</b>	LS	<b>Bereich</b>	16 Bit mit Vorzeichen
<b>Standard</b>	n.b.	<b>Aktualisierungsrate</b>	Sofort

Durch Schreiben in diesem Parameter wird das Steuerwort aktualisiert. Beim Abfragen dieses Parameters wird das Status-Datenwort gelesen (wie bei Parameter #10.40).

**Tabelle 5.4 Steuerwort**

Bit	Beschreibung
b15	Falls eingestellt, wird der Wert von #1.46 über b6 eingestellt
b14	Falls eingestellt, wird der Wert von #1.45 über b5 eingestellt
b13	Der Wert von #18.33 (Anwendungsmenü 1, Bit 3) wird eingestellt
b12	Falls eingestellt, wird der Wert von #6.32 über b3 eingestellt
b11	Falls eingestellt, wird der Wert von #6.31 über b2 eingestellt
b10	Falls eingestellt, wird der Wert von #6.30 über b1 eingestellt
b9	Falls eingestellt, wird der Wert von #6.15 über b0 eingestellt
b8	Der Wert von #18.32 (Anwendungsmenü 1, Bit 2) wird eingestellt
b7	Der Wert von #18.31 (Anwendungsmenü 1, Bit 1) wird eingestellt
b6	Der Wert von #1.46 (Voreinstellungs-Auswahlbit 1) wird eingestellt
b5	Der Wert von #1.45 (Voreinstellungs-Auswahlbit 0) wird eingestellt

**Tabelle 5.4 Steuerwort**

Bit	Beschreibung
b4	Benutzer-Fehlerabschaltung. Falls eingestellt, wird sofort eine Fehlerabschaltung des Umrichters ausgelöst
b3	Der Wert von #6.32 wird eingestellt (Ansteuerbit 2: Linkslauf)
b2	Der Wert von #6.31 wird eingestellt (Ansteuerbit 1: Tippen)
b1	Der Wert von #6.30 wird eingestellt (Ansteuerbit 0: Rechtslauf)
b0	Der Wert von #6.15 (Umrichter freigeben) wird eingestellt

**Tabelle 5.5 Status-Datenwort**

Bit	Beschreibung
b15	Nicht verwendet
b14	#10.15 (Netzausfall)
b13	#10.14 (Ist-Drehrichtung)
b12	#10.13 (Soll-Drehrichtung)
b11	#10.12 (Alarm Bremswiderstand)
b10	#10.11 (Bremschopper aktiv)
b9	#10.10 (Generatorischer Betrieb)
b8	#10.09 (Umrichterleistung an der Stromgrenze)
b7	#10.08 (Nennlaststrom erreicht)
b6	#10.07 (Oberhalb der Solldrehzahl)
b5	#10.06 (Drehzahl erreicht)
b4	#10.05 (Unterhalb der Solldrehzahl)
b3	#10.04 (Auf oder unterhalb der Minimaldrehzahl)
b2	#10.03 (Nulldrehzahl)
b1	#10.02 (Umrichter freigegeben)
b0	#10.01 (Umrichter betriebsbereit)

#90.12	Grund Event-Task-Zeitkoordinierung		
<b>Zugang</b>	NL	<b>Bereich</b>	16 Bit ohne Vorzeichen
<b>Standard</b>	n.b.	<b>Aktualisierungsrate</b>	Bei Event

#90.13	Grund Event1-Task-Zeitkoordinierung		
<b>Zugang</b>	NL	<b>Bereich</b>	16 Bit ohne Vorzeichen
<b>Standard</b>	n.b.	<b>Aktualisierungsrate</b>	Bei Event1

#90.14	Grund Event2-Task-Zeitkoordinierung		
<b>Zugang</b>	NL	<b>Bereich</b>	16 Bit ohne Vorzeichen
<b>Standard</b>	n.b.	<b>Aktualisierungsrate</b>	Bei Event2

#90.15	Grund Event3-Task-Zeitkoordinierung		
<b>Zugang</b>	NL	<b>Bereich</b>	16 Bit ohne Vorzeichen
<b>Standard</b>	n.b.	<b>Aktualisierungsrate</b>	Bei Event3

Diese vier Parameter ergeben den Grund, aus dem der bestimmte EVENT-Task geplant wurde. Der Wert hat nur dann eine Bedeutung, wenn der bestimmte EVENT-Task ausgeführt wird.



Für den Wert gilt die folgende Bit-Zuordnung und Definition:

Bits	Beschreibung
0-1	Auslöse-Task Steckplatz 0 = Lokaler Steckplatz 1 = Steckplatz 1 2 = Steckplatz 2 3 = Steckplatz 3
2-7	Auslösergrund 0-31 = Anderes Optionsmodul ausgelöst 32 = CTNet-Synchronisation 33 = Zeitgebereinheit 34-63 = Benutzerdefinierter Grund über den DPL-Befehl SCHEDULEEVENT.

#90.18	Flag: Freeze Istwert-Encoder		
Zugang	LS	Bereich	0/1
Standard	n.b.	Aktualisierungsrate	250 µs

Dieser Parameter muss auf Null gesetzt werden, damit die Freeze-Position erfasst werden kann. Sobald der Freeze erfolgt ist, wird dieser Parameter auf 1 gesetzt. Um ihn wieder zu aktivieren, müssen Sie ihn einfach auf Null setzen.

#90.19	Freeze-Position Istwert-Encoder		
Zugang	NL	Bereich	32 Bit mit Vorzeichen
Standard	n.b.	Aktualisierungsrate	250 µs

#90.20	Drehen: Freeze Istwert-Encoder		
Zugang	NL	Bereich	16 Bit ohne Vorzeichen
Standard	n.b.	Aktualisierungsrate	250 µs

In diesen 2 Parametern werden Position und Umdrehungen des Istwert-Encoders zum Zeitpunkt der Aktivierung des Freeze-Eingangs gespeichert.

#90.21	Umrichter-Encoder: Positionsprüfung deaktivieren		
Zugang	LS	Bereich	0/1
Standard	0	Aktualisierungsrate	Sofort

Die abgeleitete Position wird vom Umrichter mit Hilfe des Sinus- und Cosinus-Signalverlaufs aus einem SINCOS-Encoder über serielle Kommunikation regelmäßig geprüft. Setzen Sie diesen Parameter auf 1, um dies zu deaktivieren.

#90.22	Umrichter-Encoder: Übertragungsregister Kommunikation		
Zugang	LS	Bereich	16 Bit ohne Vorzeichen
Standard	n.b.	Aktualisierungsrate	Sofort

Wenn der Parameter „Umrichter-Encoder: Positionsprüfung“ deaktiviert ist (#90.21=1), kann dieser Parameter für die Kommunikation mit dem Encoder verwendet werden, der über eine serielle Schnittstelle mit dem Umrichter verbunden ist.

#90.23	Umrichter-Encoder: Empfangsregister Kommunikation		
Zugang	LS	Bereich	16 Bit ohne Vorzeichen
Standard	n.b.	Aktualisierungsrate	Sofort

Wenn der Parameter „Umrichter-Encoder: Positionsprüfung“ deaktiviert ist (#90.21=1), kann dieser Parameter für die Kommunikation mit dem Encoder verwendet werden, der über eine serielle Schnittstelle mit dem Umrichter verbunden ist.

#90.24	SM-Applications-Steckplatznummer		
<b>Zugang</b>	NL	<b>Bereich</b>	8 Bit ohne Vorzeichen
<b>Standard</b>	n.b.	<b>Aktualisierungsrate</b>	Initialisierung

Mit diesem Parameter wird die Nummer des Steckplatzes gemeldet, an dem das SM-Applications-Modul angebracht ist.

#90.25	Nullimpulsposition Istwert-Encoder ( $2^{32}/U$ )		
<b>Zugang</b>	NL	<b>Bereich</b>	32 Bit mit Vorzeichen
<b>Standard</b>	n.b.	<b>Aktualisierungsrate</b>	Siehe #81.16

Die oberen 16 Bits sind auf 65536 Zählerwerte pro Umdrehung skaliert, unabhängig vom Typ des Istwert-Geräts und von der im Umrichter konfigurierten Skalierung.

#90.26	Drehen: Nullimpuls Istwert-Encoder ( $2^{16}/U$ )		
<b>Zugang</b>	NL	<b>Bereich</b>	16 Bit ohne Vorzeichen
<b>Standard</b>	n.b.	<b>Aktualisierungsrate</b>	Siehe #81.16

Mit diesem Parameter wird der Umdrehungszähler für den Nullimpuls des Istwert-Encoders angegeben.

#90.27	Datenbank-Versionsnummer für SM-Applications		
<b>Zugang</b>	NL	<b>Bereich</b>	16 Bit ohne Vorzeichen
<b>Standard</b>	n.b.	<b>Aktualisierungsrate</b>	Initialisierung

Die Datenbank-Versionsnummer wird nach dem Einschalten aus der Datenbank abgefragt.

#90.28	Flag: Freeze Sollwert-Encoder		
<b>Zugang</b>	LS	<b>Bereich</b>	0/1
<b>Standard</b>	n.b.	<b>Aktualisierungsrate</b>	250 $\mu$ s

Dieser Parameter muss auf Null gesetzt werden, damit die Freeze-Position erfasst werden kann. Sobald der Freeze erfolgt ist, wird dieser Parameter auf 1 gesetzt. Um ihn wieder zu aktivieren, müssen Sie ihn einfach auf Null setzen.

#90.29	Freeze-Position Sollwert-Encoder		
<b>Zugang</b>	NL	<b>Bereich</b>	32 Bit mit Vorzeichen
<b>Standard</b>	n.b.	<b>Aktualisierungsrate</b>	250 $\mu$ s

#90.30	Drehen: Freeze Sollwert-Encoder		
<b>Zugang</b>	NL	<b>Bereich</b>	16 Bit ohne Vorzeichen
<b>Standard</b>	n.b.	<b>Aktualisierungsrate</b>	250 $\mu$ s

In diesen 2 Parametern werden Position und Umdrehungen des Sollwert-Encoders zum Zeitpunkt der Aktivierung des Freeze-Eingangs gespeichert.

#90.31	Umdrehungen und Grobposition Istwert-Encoder		
Zugang	NL	Bereich	32 Bit mit Vorzeichen
Standard	n.b.	Aktualisierungsrate	Siehe #81.16

#90.32	Umdrehungen und Grobposition Sollwert-Encoder		
Zugang	NL	Bereich	32 Bit mit Vorzeichen
Standard	n.b.	Aktualisierungsrate	Siehe #81.16

In diesen 2 Parametern werden die Umdrehungen mit 16 Bit im oberen Wort und die Position mit 16 Bit im unteren Wort gespeichert, und zwar für den Istwert-Encoder (#90.31) und für den Sollwert-Encoder (#90.32).

#90.33	Freeze: Umdrehungen und Grobposition Istwert-Encoder		
Zugang	NL	Bereich	32 Bit mit Vorzeichen
Standard	n.b.	Aktualisierungsrate	250 $\mu$ s

#90.34	Freeze: Umdrehungen und Grobposition Sollwert-Encoder		
Zugang	NL	Bereich	32 Bit mit Vorzeichen
Standard	n.b.	Aktualisierungsrate	250 $\mu$ s

In diesen 2 Parametern werden zum Zeitpunkt der Aktivierung des Freeze-Eingangs die Umdrehungen mit 16 Bit im oberen Wort und die Position mit 16 Bit im unteren Wort gespeichert.

#90.35	Nullimpulsposition Sollwert-Encoder ( $2^{32}/U$ )		
Zugang	NL	Bereich	32 Bit mit Vorzeichen
Standard	n.b.	Aktualisierungsrate	Siehe #81.16

In diesem Parameter wird die Position des Sollwert-Encoders zum Zeitpunkt der Aktivierung des Nullimpulses gespeichert.

#90.36	Drehen: Nullimpuls Sollwert-Encoder ( $2^{16}/U$ )		
Zugang	NL	Bereich	16 Bit ohne Vorzeichen
Standard	n.b.	Aktualisierungsrate	Siehe #81.16

In diesem Parameter wird der Umdrehungszähler des Sollwert-Encoders zum Zeitpunkt der Aktivierung des Nullimpulses gespeichert.

#90.37	Nullimpuls: Umdrehungen und Grobposition Istwert-Encoder		
Zugang	NL	Bereich	32 Bit mit Vorzeichen
Standard	n.b.	Aktualisierungsrate	Siehe #81.16

#90.38	Nullimpuls: Umdrehungen und Grobposition Sollwert-Encoder		
Zugang	NL	Bereich	32 Bit mit Vorzeichen
Standard	n.b.	Aktualisierungsrate	Siehe #81.16

In diesen 2 Parametern werden zum Zeitpunkt der Aktivierung des Nullimpulses die Umdrehungen mit 16 Bit im oberen Wort und die Position mit 16 Bit im unteren Wort gespeichert, und zwar für den Istwert-Encoder (#90.37) und für den Sollwert-Encoder (#90.38).

#90.39	Tastenstatus Umrichter-Bedieneinheit		
<b>Zugang</b>	NL	<b>Bereich</b>	16 Bit mit Vorzeichen
<b>Standard</b>	n.b.	<b>Aktualisierungsrate</b>	> 40 ms

#90.40	Event-Taskauslöser		
<b>Zugang</b>	NS	<b>Bereich</b>	16 Bit ohne Vorzeichen
<b>Standard</b>	0	<b>Aktualisierungsrate</b>	Sofort

Nach dem Einstellen eines Werts für diesen Parameter wird einer der Event-Tasks für das SM-Applications-Modul ausgeführt.

Wert	Vorgang
0	Event-Task nicht auslösen
1	Event-Task auslösen
2	Event1-Task auslösen
3	Event2-Task auslösen
4	Event3-Task auslösen

#90.41	Flag: Nullimpuls Sollwert-Encoder		
<b>Zugang</b>	LS	<b>Bereich</b>	0/1
<b>Standard</b>	n.b.	<b>Aktualisierungsrate</b>	Siehe #81.16

#90.42	Flag: Nullimpuls Istwert-Encoder		
<b>Zugang</b>	LS	<b>Bereich</b>	0/1
<b>Standard</b>	n.b.	<b>Aktualisierungsrate</b>	Siehe #81.16

Diese 2 Parameter werden auf 1 gesetzt, wenn ein relevanter Encoder-Nullimpuls aktiviert ist, jedoch nur dann, wenn die Parameter für die Freigabe des Nullimpuls-Flags eingestellt wurden (Parameter #90.45 und #90.46). Damit der Nullimpuls wieder aktiviert wird, müssen diese Parameter vom Benutzer auf Null gesetzt werden. Sie können nicht vom Benutzer auf 1 gesetzt werden.

#90.43	Quelle Sollwert-Encoder		
<b>Zugang</b>	LS	<b>Bereich</b>	8 Bit ohne Vorzeichen
<b>Standard</b>	n.b.	<b>Aktualisierungsrate</b>	Sofort

#90.44	Quelle Istwert-Encoder		
<b>Zugang</b>	LS	<b>Bereich</b>	8 Bit ohne Vorzeichen
<b>Standard</b>	n.b.	<b>Aktualisierungsrate</b>	Sofort

Mit diesen 2 Parametern wird die Quelle für Soll- und Istwertdaten definiert. Die gültigen Quellen sind in der folgenden Tabelle aufgeführt.

Wert	Beschreibung
0	Umrichter-Encoder
1	Steckplatz 1
2	Steckplatz 2
3	Steckplatz 3
4	Benutzerprogramm
5	Unkonfiguriert

#90.45	Flag für den Sollwert des Nullimpulses freigeben		
Zugang	LS	Bereich	0/1
Standard	n.b.	Aktualisierungsrate	Sofort

#90.46	Flag für den Istwert des Nullimpulses freigeben		
Zugang	LS	Bereich	0/1
Standard	n.b.	Aktualisierungsrate	Sofort

Diese 2 Parameter müssen auf 1 gesetzt werden, damit die Nullimpuls-Flags (Parameter #90.41 und #90.42) bei aktiviertem Nullimpuls eingestellt werden können.

#90.47	Sollwert: Freeze freigeben		
Zugang	LS	Bereich	0/1
Standard	n.b.	Aktualisierungsrate	Sofort

#90.48	Istwert: Freeze freigeben		
Zugang	LS	Bereich	0/1
Standard	n.b.	Aktualisierungsrate	Sofort

Diese 2 Parameter müssen auf 1 gesetzt werden, damit die Freeze-Flags (Parameter #90.18 und #90.28) bei aktiviertem Freeze-Eingang eingestellt werden können.

## 5.9 Menü 91 - Schnellzugangsparameter

Die Parameter in diesem Menü sind virtuelle SM-Applications-Parameter, die im Vergleich zu Umrichterparametern eine schnellere Aktualisierungsrate oder eine verbesserte Auflösung ermöglichen.

#91.01	Verknüpfung freigeben		
Zugang	LS	Bereich	8 Bit ohne Vorzeichen
Standard	0	Aktualisierungsrate	Sofort

Mit diesem Parameter werden die im Folgenden beschriebenen Verknüpfungparameter freigegeben. Sie müssen das entsprechende Bit in diesem Parameter einstellen. Siehe folgende Tabelle.

Bit	Funktion	Dazugehöriger Parameter
0	Direktzugriff Drehzahlsollwert freigeben	#91.02
1	Direktzugriff interner Drehzahlsollwert freigeben	#91.03
2	Direktzugriff Drehmomentsollwert freigeben	#91.04
3	Analogausgang 1	-
4	Analogausgang 2	-

Bit	Funktion	Dazugehöriger Parameter
5	Reserviert	-
6	Reserviert	-
7	Reserviert	-

#91.02	Drehzahlsollwert (#1.21)		
Zugang	LS	Bereich	32 Bit mit Vorzeichen
Standard	n.b.	Aktualisierungsrate	250 $\mu$ s

Der Drehzahlsollwert wird in Einheiten von  $0,001 \text{ min}^{-1}$  eingestellt. Dieser Wert wird in Umrichterparameter #1.21 (Drehzahlvoreinstellung 1) widergespiegelt. Um also die Drehzahl für den Umrichter steuern zu können, müssen Sie darauf achten, dass am Umrichter die Drehzahlvoreinstellung 1 ausgewählt ist (#1.14=3, #1.15=1).

Vergewissern Sie sich bei der Verwendung dieses Parameters, dass Bit 0 von #91.01 eingestellt ist und dass die maximale Drehzahl in #91.05 eingestellt ist. Beachten Sie, dass dieser Parameter nur im Closed Loop-Vektormodus und im Servomodus gültig ist.

#91.03	Interner Drehzahlsollwert (#3.22)		
Zugang	LS	Bereich	32 Bit mit Vorzeichen
Standard	n.b.	Aktualisierungsrate	250 $\mu$ s

Der interne Drehzahlsollwert des Umrichters wird in Einheiten von  $0,001 \text{ min}^{-1}$  gesteuert.

Vergewissern Sie sich bei der Verwendung dieses Parameters, dass Bit 1 von #91.01 eingestellt ist und dass die maximale Drehzahl in #91.05 eingestellt ist. Beachten Sie, dass dieser Parameter nur im Closed Loop-Vektormodus und im Servomodus gültig ist.

#91.04	Drehmomentsollwert (#4.08)		
Zugang	LS	Bereich	32 Bit mit Vorzeichen
Standard	n.b.	Aktualisierungsrate	250 $\mu$ s

Der Drehmomentsollwert (Umrichterparameter #4.08) wird in Einheiten von 0,01 % angegeben.

Wenn Sie diesen Parameter verwenden möchten, vergewissern Sie sich, dass Bit 2 von #91.01 eingestellt ist.

#91.05	Maximale Drehzahl ( $\text{min}^{-1}$ )		
Zugang	LS	Bereich	32 Bit mit Vorzeichen
Standard	1500	Aktualisierungsrate	n.b.

Stellen Sie dies auf die maximale (absolute) Drehzahl ein, die jemals in #91.02 oder #91.03 geschrieben werden soll. Die Einheit lautet  $1 \text{ min}^{-1}$ .

Hiermit wird die Auflösung für die an den Umrichter gesendeten Drehzahlwerte bestimmt. Der Versuch, in #91.02 oder #91.03 Drehzahlwerte zu schreiben, die den in #91.05 angegebenen Drehzahlwert überschreiten, führt dazu, dass der Wert begrenzt wird, oder verursacht einen Laufzeitfehler durch Überschreitung des Wertebereichs.

#91.06	Drehzahlwert		
Zugang	NL	Bereich	32 Bit mit Vorzeichen
Standard	n.b.	Aktualisierungsrate	250 $\mu$ s

In Closed Loop-Modi wird durch diesen Parameter der Unidrive SP-Drehzahlwert in Einheiten von  $0,01 \text{ min}^{-1}$  zurückgegeben. Wenn jedoch ein Encoder mit niedriger Auflösung verwendet wird, kann bei niedrigen Drehzahlen eine gewisse Schwankung auftreten. Zum Beispiel kann dieser Parameter bei  $10 \text{ min}^{-1}$  mit einem 1024 I/U-Encoder zwischen 0 und  $14,65 \text{ min}^{-1}$  springen. Dies ähnelt Unidrive SP-Parameter #3.02.

#91.07	Stromistwert (#4.02)		
Zugang	NL	Bereich	16 Bit mit Vorzeichen
Standard	n.b.	Aktualisierungsrate	250 $\mu\text{s}$

Dieser Wert wird aus Parameter #4.02 übernommen und in Einheiten von 0,01 A angegeben (d. h.  $150=1,5 \text{ A}$ ).

#91.08	Wert Umrichter-Analogeingang 1		
Zugang	NL	Bereich	$\pm 4000$
Standard	n.b.	Aktualisierungsrate	250 $\mu\text{s}$

Dieser Wert wird vom Analogeingang 1 des Umrichters übernommen und auf  $\pm 4000$  skaliert, um das maximale positive/negative Signal an dem Eingang darzustellen. Informationen zur Aktualisierungsrate der Analogeingänge finden Sie in der Betriebsanleitung des Umrichters.

#91.09	Wert Umrichter-Analogeingang 2		
Zugang	NL	Bereich	$\pm 1000$
Standard	n.b.	Aktualisierungsrate	250 $\mu\text{s}$

Dieser Wert wird vom Analogeingang 2 des Umrichters übernommen und auf  $\pm 1000$  skaliert, um das maximale positive/negative Signal an dem Eingang darzustellen. Informationen zur Aktualisierungsrate der Analogeingänge finden Sie in der Betriebsanleitung des Umrichters.

#91.10	Wert Umrichter-Analogeingang 3		
Zugang	NL	Bereich	$\pm 1000$
Standard	n.b.	Aktualisierungsrate	250 $\mu\text{s}$

Dieser Wert wird vom Analogeingang 3 des Umrichters übernommen und auf  $\pm 1000$  skaliert, um das maximale positive/negative Signal an dem Eingang darzustellen. Informationen zur Aktualisierungsrate der Analogeingänge finden Sie in der Betriebsanleitung des Umrichters.

#91.16	Umrichter-Digitaleingänge		
Zugang	NL	Bereich	8 Bit ohne Vorzeichen
Standard	n.b.	Aktualisierungsrate	250 $\mu\text{s}$

Dieser Parameter ähnelt insofern Umrichterparameter #8.20, als der Status von 7 Digitaleingängen in einem einzigen Parameter angegeben wird. Logikpolarität und Umkehrungen werden berücksichtigt.

Die Bits sind folgendermaßen zugewiesen:

Bit	Digitaleingang
0	F1
1	F2
2	F3
3	F4
4	F5

Bit	Digitaleingang
5	F6
6	FREIGABE
7	Reserviert - Als Null gelesen

#91.17	Anzahl der erhaltenen gültigen CTSync-Meldungen		
Zugang	LS	Bereich	32 Bit mit Vorzeichen
Standard	n.b.	Aktualisierungsrate	

Dieser Parameter wird jedes Mal um 1 erhöht, wenn eine fehlerfreie CTSync-Meldung mit gültiger Prüfsumme empfangen wird.

#91.18	Anzahl der erhaltenen fehlerhaften CTSync-Meldungen		
Zugang	LS	Bereich	32 Bit mit Vorzeichen
Standard	n.b.	Aktualisierungsrate	

Dieser Parameter wird jedes Mal um 1 erhöht, wenn eine CTSync-Meldung mit fehlerhafter Prüfsumme empfangen wird.

#91.19	Anzahl der fehlenden CTSync-Meldungen		
Zugang	LS	Bereich	32 Bit mit Vorzeichen
Standard	n.b.	Aktualisierungsrate	

Dieser Parameter wird jedes Mal um 1 erhöht, wenn eine vom Modul erwartete Meldung nicht empfangen wird.

#91.20	CTSync-Synchronisationssignal: Breite zu kurz		
Zugang	LS	Bereich	32 Bit mit Vorzeichen
Standard	n.b.	Aktualisierungsrate	Synchronisation

Dieser Parameter wird jedes Mal um 1 erhöht, wenn das Synchronisationssignal die falsche Breite aufweist. Während der Synchronisation wird dieser Parameter wahrscheinlich erhöht, er sollte sich jedoch nach Abschluss der Synchronisation stabilisieren. Wenn dieser Parameter nach der Synchronisation weiter erhöht wird, ist wahrscheinlich das Rauschen im RS485-Netzwerk zu stark. Überprüfen Sie die Verbindungen.

#91.21	Synchronisationssteuerung zwischen Optionen		
Zugang	LS	Bereich	0 bis 2
Standard	0	Aktualisierungsrate	Sofort

Mit Hilfe dieses Parameters kann der Benutzer das SM-Applications-Modul im Synchronisationsprogramm zwischen den Optionsmodulen konfigurieren. Weitere Informationen finden Sie in Abschnitt 10 *Synchronisation zwischen den Optionsmodulen*.

Wert	Beschreibung
0	Das Modul muss nicht am Synchronisationsprogramm zwischen den Optionsmodulen teilnehmen.
1	Das SM-Applications-Modul soll als Produzent am Synchronisationsprogramm zwischen den Optionsmodulen teilnehmen. Details zum Begriff „Produzent“ finden Sie in Abschnitt 10.1 <i>Überblick</i> .
2	Das SM-Applications-Modul soll als Konsument am Synchronisationsprogramm zwischen den Optionsmodulen teilnehmen. Details zum Begriff „Konsument“ finden Sie in Abschnitt 10.1 <i>Überblick</i> .

#91.22	Synchronisationsstatus zwischen Optionen		
Zugang	NL	Bereich	8 Bit ohne Vorzeichen
Standard	n.b.	Aktualisierungsrate	Sofort



Mit diesem Parameter wird der Status des SM-Applications-Moduls im Synchronisationsprogramm zwischen den Optionsmodulen angezeigt. Weitere Informationen finden Sie in Kapitel 10 *Synchronisation zwischen den Optionsmodulen* .

Bit	Bedeutung	Beschreibung
1 bis 0	Angeforderte Rolle für die Synchronisation zwischen den Modulen	Dies ist identisch mit den Bits 1:0 des oben beschriebenen Parameters für die Synchronisationssteuerung zwischen den Optionsmodulen.
2	Rolle für die Synchronisation zwischen den Modulen erreicht	Mit diesem Bit wird angezeigt, dass für ein Modul, das in der Synchronisation zwischen den Modulen Produzent werden soll, diese Rolle erreicht wurde. Falls mehrere Module an einem gegebenen Umrichter Produzenten für die Synchronisation werden sollen, wird dieses Bit für mindestens eines dieser Module zurückgesetzt. Bei einem Modul, das in der Synchronisation zwischen den Modulen Konsument werden soll, wird mit diesem Bit angezeigt, dass keine Produzentenrolle angefordert wurde und dass an einem anderen Steckplatz ein Produzent gefunden wurde, der nun als Quelle für Synchronisationsdaten verwendet wird. (Falls an einem anderen Steckplatz ein Produzent gefunden wurde, jedoch die Rate der Synchronisationsdaten mit dem Konsumenten nicht kompatibel ist, wird dieses Bit zurückgesetzt, da der Produzent - obwohl er gefunden wurde - nun nicht als Quelle für Synchronisationsdaten verwendet wird.)
3	Leistung des Synchronisationsproduzenten innerhalb der Spezifikation	Dieses Bit ist nur für Module relevant, die als Produzent für die Synchronisation zwischen den Optionsmodulen und als CTSync-Slave gekennzeichnet wurden. Mit diesem Bit wird angezeigt, dass das von dem Produzenten für die Synchronisation zwischen den Optionsmodulen gelieferte Signal innerhalb der spezifizierten Toleranz des Umrichters liegt und dass der Umrichter jetzt fest auf die Frequenz des CTSync-Slave eingestellt ist. Beachten Sie, dass dieses Bit nur dann eingestellt werden kann, wenn die Bits 2 bis 0 dieses Parameters 1, 0 und 1 lauten, was anzeigt, dass das Modul Produzent ist (0, 1) und dass die Produzentenrolle erreicht wurde (1).
7 bis 4	Reserviert	Als Null gelesen

## 5.10 Menüs 18 und 19 - Anwendungsparameter

Diese zwei Menüs werden als Anwendungsparameter bezeichnet, da sie alle frei für jeden vom Benutzer gewünschten Zweck verwendet werden können.

Beide Menüs sind identisch aufgebaut. Alle Parameter sind mit einer Lese- und Schreibberechtigung für das SM-Applications-Modul (und über Kommunikation) verbunden, können jedoch auf der Bedieneinheit des Umrichters schreibgeschützt sein.

#1x.01	Ganze Zahl, Lesen und Schreiben, Speicherung beim Ausschalten		
Zugang	LS	Bereich	16 Bit mit Vorzeichen
Standard	0	Aktualisierungsrate	n.b.

Dieser Parameter wird vom Umrichter beim Ausschalten automatisch gespeichert.

<b>#1x.02 bis #1x.10</b>	<b>Ganze Zahl, nur Lesen</b>		
<b>Zugang</b>	LS (NL Umrichter)	<b>Bereich</b>	16 Bit mit Vorzeichen
<b>Standard</b>	0	<b>Aktualisierungsrate</b>	n.b.

<b>#1x.11 bis #1x.30</b>	<b>Ganze Zahl, Lesen und Schreiben</b>		
<b>Zugang</b>	LS	<b>Bereich</b>	16 Bit mit Vorzeichen
<b>Standard</b>	0	<b>Aktualisierungsrate</b>	n.b.

<b>#1x.31 bis #1x.50</b>	<b>Bit, Lesen und Schreiben</b>		
<b>Zugang</b>	NL	<b>Bereich</b>	0/1
<b>Standard</b>	0	<b>Aktualisierungsrate</b>	n.b.

Die Parameter #1x.11 bis #1x.50 können im nicht flüchtigen Speicher des Umrichters gespeichert werden.

## 5.11 Menü 20 - Anwendungsmenü

Dieses Menü enthält, ebenso wie die Menüs 18 und 19, Parameter, die keine Auswirkungen auf den Betrieb des Umrichters haben und daher universal verwendet werden können.

**HINWEIS** Dieses Menü wird NICHT im nicht flüchtigen Speicher des Umrichters gespeichert. Stattdessen kann es auf Anforderung im Flash-Speicher des SM-Applications-Moduls gespeichert werden. Wenn mehrere SM-Applications-Module angebracht sind, sollte aus nahe liegenden Gründen nur eines davon für Speicherung und Wiederherstellung dieses Menüs konfiguriert werden.

Wenn Parameter #81.21 eingestellt ist, wird dieses Menü vom SM-Applications-Modul gespeichert und wiederhergestellt.

<b>#20.01- #20.20</b>	<b>Ganze Zahl, Lesen und Schreiben</b>		
<b>Zugang</b>	LS	<b>Bereich</b>	16 Bit mit Vorzeichen
<b>Standard</b>	0	<b>Aktualisierungsrate</b>	n.b.

<b>#20.21- #20.40</b>	<b>Lange ganze Zahl, Lesen und Schreiben</b>		
<b>Zugang</b>	LS	<b>Bereich</b>	32 Bit mit Vorzeichen
<b>Standard</b>	0	<b>Aktualisierungsrate</b>	n.b.

Diese Parameter sind standardmäßige ganze Zahlen, so dass ein Wertebereich von -32768 bis 32767 möglich ist.

Diese Parameter sind lange ganze Zahlen, so dass ein Wertebereich von  $-2^{31}$  bis  $2^{31}-1$  möglich ist. Wenn der maximale anzeigbare Wert (9 999 999) überschritten wurde, wird auf dem Display bzw. der Bedieneinheit des Umrichters „-----“ angezeigt. Es ist nicht möglich, auf der Bedieneinheit größere Werte einzugeben als den maximalen Anzeigewert.

# 6 Kommunikation

## 6.1 Serieller RS485-Kommunikationsanschluss

Das SM-Applications-Modul verfügt über einen eingebauten seriellen RS485-Kommunikationsanschluss. Informationen zu Hardware-Anschluss und Verdrahtung finden Sie in Abschnitt 3 *Installation* .

Eine Reihe von eingebauten Protokollen werden von diesem Anschluss unterstützt: CT-ANSI (Slave), Modbus RTU (Master- und Slave-Modus), Modbus ASCII (Master- und Slave-Modus) sowie 3 Benutzermodi. Sowohl Zweidraht- als auch Vierdrahtkonfigurationen sind möglich.

**Tabelle 6.1 Serielle Modi - Parameter #81.06**

Modus	Beschreibung
1	Vierdraht-CT-ANSI Der Anschluss wird auf 1 Startbit, 7 Datenbits, gerade Parität und 1 Stoppbit eingestellt.
2	Reserviert
3	Reserviert
4	Reserviert
5	Zweidraht-CT-ANSI Der Anschluss wird auf 1 Startbit, 7 Datenbits, gerade Parität und 1 Stoppbit eingestellt.
6	Benutzermodus. 1 Startbit, 7 Datenbits, gerade Parität und 1 Stoppbit (insgesamt 10 Bits).
7	Benutzermodus. 1 Startbit, 8 Datenbits, gerade Parität und 1 Stoppbit (insgesamt 11 Bits).
8	Benutzermodus. 1 Startbit, 8 Datenbits, keine Parität und 1 Stoppbit (insgesamt 10 Bits).
9	Reserviert
10	Reserviert
11	Reserviert
12	Reserviert
13, 43, 73	Vierdraht-Modbus-RTU: Slave Der RS485-Anschluss wird folgendermaßen eingestellt: Modus 13: 1 Startbit, 8 Datenbits, keine Parität, 2 Stoppbits. Modus 43: 1 Startbit, 8 Datenbits, gerade Parität, 1 Stoppbit. Modus 73: 1 Startbit, 8 Datenbits, ungerade Parität, 1 Stoppbit.
14, 44, 74	Vierdraht-Modbus-ASCII: Slave Der RS485-Anschluss wird folgendermaßen eingestellt: Modus 14: 1 Startbit, 7 Datenbits, keine Parität, 2 Stoppbits. Modus 44: 1 Startbit, 7 Datenbits, gerade Parität, 1 Stoppbit. Modus 74: 1 Startbit, 7 Datenbits, ungerade Parität, 1 Stoppbit.
15, 45, 75	Zweidraht-Modbus-RTU: Slave Der RS485-Anschluss wird folgendermaßen eingestellt: Modus 15: 1 Startbit, 8 Datenbits, keine Parität, 2 Stoppbits. Modus 45: 1 Startbit, 8 Datenbits, gerade Parität, 1 Stoppbit. Modus 75: 1 Startbit, 8 Datenbits, ungerade Parität, 1 Stoppbit.
16, 46, 76	Zweidraht-Modbus-ASCII: Slave Der RS485-Anschluss wird folgendermaßen eingestellt: Modus 16: 1 Startbit, 7 Datenbits, keine Parität, 2 Stoppbits. Modus 46: 1 Startbit, 7 Datenbits, gerade Parität, 1 Stoppbit. Modus 76: 1 Startbit, 7 Datenbits, ungerade Parität, 1 Stoppbit.
17, 47, 77	Vierdraht-Modbus-RTU: Master Der RS485-Anschluss wird folgendermaßen eingestellt: Modus 17: 1 Startbit, 8 Datenbits, keine Parität, 2 Stoppbits. Modus 47: 1 Startbit, 8 Datenbits, gerade Parität, 1 Stoppbit. Modus 77: 1 Startbit, 8 Datenbits, ungerade Parität, 1 Stoppbit.
18, 48, 78	Vierdraht-Modbus-ASCII: Master Der RS485-Anschluss wird folgendermaßen eingestellt: Modus 18: 1 Startbit, 7 Datenbits, keine Parität, 2 Stoppbits. Modus 48: 1 Startbit, 7 Datenbits, gerade Parität, 1 Stoppbit. Modus 78: 1 Startbit, 7 Datenbits, ungerade Parität, 1 Stoppbit.

**Tabelle 6.1 Serielle Modi - Parameter #81.06**

Modus	Beschreibung
19, 49, 79	Zweidraht-Modbus-RTU: Master Der RS485-Anschluss wird folgendermaßen eingestellt: Modus 19: 1 Startbit, 8 Datenbits, keine Parität, 2 Stoppbits. Modus 49: 1 Startbit, 8 Datenbits, gerade Parität, 1 Stoppbit. Modus 79: 1 Startbit, 8 Datenbits, ungerade Parität, 1 Stoppbit.
20, 50, 80	Zweidraht-Modbus-ASCII: Master Der RS485-Anschluss wird folgendermaßen eingestellt: Modus 18: 1 Startbit, 7 Datenbits, keine Parität, 2 Stoppbits. Modus 48: 1 Startbit, 7 Datenbits, gerade Parität, 1 Stoppbit. Modus 78: 1 Startbit, 7 Datenbits, ungerade Parität, 1 Stoppbit.
25	CT-Sync-Master für SM-Applications Die Baudrate ist auf 896875 Bit/s festgelegt.
26	CT-Sync-Slave für SM-Applications Die Baudrate ist auf 896875 Bit/s festgelegt.

Wenn ein ungültiger oder nicht unterstützter Modus ausgewählt wurde, wird der Modus auf den Standardwert 1 (Vierdraht-CT-ANSI) zurückgesetzt, und möglicherweise tritt Laufzeitfehler 49 auf.

Die Baudrate wird in Parameter #81.07 angegeben.

Die Adresse dieser Einheit wird in Parameter #81.05 angegeben.

### 6.1.1 CT-ANSI

Das CT-ANSI-Protokoll ist mit dem im Umrichter implementierten Protokoll identisch. Weitere Informationen finden Sie in der Dokumentation zum Umrichter oder in der Online-Hilfe.

**HINWEIS**

Das Abfragen und Schreiben von vollständigen 32 Bit-Werten wird vom CT-ANSI-Protokoll unterstützt. Solche großen Werte werden vom UD70-Produkt für Unidrive 1 nicht unterstützt.

Über den RS485-Anschluss kann auf alle Umrichterparameter sowie auf die Parameter des SM-Applications-Moduls zugegriffen werden.

### 6.1.2 Modbus-RTU

Sowohl der Slave-Modus als auch der Master-Modus des Modbus-RTU-Protokolls von Modicon werden unterstützt.

Im RTU-Slave-Modus werden die folgenden Funktionscodes unterstützt:

Funktion	Beschreibung
FC3	Voreinstellung für ein einzelnes Register
FC16	Voreinstellung für mehrere Register
FC23	Abfragen bzw. Schreiben von mehreren Registern

Die maximale Anzahl der Register, die in einem Schritt abgefragt bzw. geschrieben werden können, beträgt 16.

Umrichterparameter werden Modbus-Registern als **40000 + Menü × 100 + Parameter** zugeordnet. Zum Beispiel entspricht Parameter #01.21 Registernummer 40121.

Auf Parameterdaten kann entweder im 16 Bit-Modus oder im 32 Bit-Modus zugegriffen werden. Der Modus wird mit Hilfe der oberen 2 Bits der Registeradresse folgendermaßen ausgewählt:



Typfeld (Bits 15 bis 14)	Zugang
00	16 Bit. Rückwärtskompatibel.
01	32 Bit.
10	Reserviert
11	Reserviert

Daher ist die Registernummer für den Zugriff auf Parameter #70.01 im 32 Bit-Modus gleich  $40000 + (0 \times 4000 + 70 \times 100 + 01) = 63385$ .

Wenn ein 32 Bit-Parameter im 16 Bit-Zugriffsmodus abgefragt wird, werden die 16 am wenigsten signifikanten Bits zurückgegeben.

Beachten Sie, dass die tatsächlich auf der Protokollebene gesendete Registernummer um 1 kleiner ist als die angeforderte und den Versatz um 40000 nicht enthält. Diese Differenz von -1 wird von den meisten Modbus-Mastern automatisch verarbeitet, von einigen jedoch nicht.

Für den Master-Modus werden die folgenden Befehle im benutzerdefinierten DPL-Programm verwendet:

- RtuReadHoldingRegs
- RtuReadHoldingParas
- RtuReadInputRegs
- RtuPresetMultipleRegs
- RtuPresetMultipleParas
- RtuMasterReply
- RtuMasterStatus

### 6.1.3 Modbus ASCII

Sowohl der Slave-Modus als auch der Master-Modus des Modbus ASCII-Protokolls von Modicon werden unterstützt.

Im Modbus ASCII-Slave-Modus werden die folgenden Funktionscodes unterstützt:

Funktion	Beschreibung
FC3	Voreinstellung für ein einzelnes Register
FC16	Voreinstellung für mehrere Register
FC23	Abfragen bzw. Schreiben von mehreren Registern



Weitere Informationen finden Sie in der Online-Hilfe.

Vollständige Details zum Modbus-Protokoll finden Sie auf der Modicon-Website unter [www.modicon.com](http://www.modicon.com). Beachten Sie, dass der 32 Bit-Zugangsmodus Control Techniques-spezifisch ist.

### 6.1.4 Benutzermodi

Mit diesen Modi werden alle internen Protokolle abgeschaltet, und der Benutzer kann direkt vom DPL-Programm aus auf den RS485-Anschluss zugreifen. Sie können in Verbindung mit den DPL-ANSI-Befehlen (ANSIREAD, ANSIWRITE usw.) verwendet werden. Benutzerdefinierte Protokolle können auch mit Hilfe der DPL-Befehle PUTCHAR und GETCHAR implementiert werden.

## 6.2 CTNet

Die vollständigen Details zu CTNet gehen über den Umfang dieser Betriebsanleitung hinaus und sind in der separaten CTNet-Betriebsanleitung zu finden.

## 6.3 SM-Applications-Zuordnungsparameter

Das SM-Applications-Modul verfügt über interne Parameter, die über Feldbusoptionen beschrieben oder abgefragt werden können, die ebenfalls am Unidrive SP angebracht sind. Dies kann eine bequeme Methode für die Kommunikation zwischen 2 Feldbussen sein. Diese Parameter sind in der folgenden Tabelle aufgeführt.

**Tabelle 6.2 SM-Applications-modulinterne Parameter**

SM-Applications-Parameter	Parameterreferenz	Direkt an Steckplatz 1	Direkt an Steckplatz 2	Direkt an Steckplatz 3
_Pxx% (SPS-Register)	#70.xx	#100.xx	#130.xx	#160.xx
_Qxx% (SPS-Register)	#71.xx	#101.xx	#131.xx	#161.xx
_Rxx% (SPS-Register)	#72.xx	#102.xx	#132.xx	#162.xx
_Sxx% (SPS-Register)	#73.xx	#103.xx	#133.xx	#163.xx
_Txx% (SPS-Register)	#74.xx	#104.xx	#134.xx	#164.xx
_Uxx% (SPS-Register)	#75.xx	#105.xx	#135.xx	#165.xx
Lokale Konfigurationsparameter	#81.xx	#111.xx	#141.xx	#171.xx
Zeitgeberfunktionsparameter	#85.xx	#115.xx	#145.xx	#175.xx
Digital-E/A-Parameter	#86.xx	#116.xx	#146.xx	#176.xx
Statusparameter	#88.xx	#118.xx	#148.xx	#178.xx
Allgemeine Parameter	#90.xx	#120.xx	#150.xx	#180.xx
Schnellzugangsparameter	#91.xx	#121.xx	#151.xx	#181.xx

Mit dem Feldbuschnittstellenmodul werden Daten direkt in und aus den internen Registern des SM-Applications-Moduls abgefragt und geschrieben. Mit dem Feldbuschnittstellenmodul können Daten in einem SM-Applications-Modul abgefragt und geschrieben werden, das an einem beliebigen Steckplatz im Unidrive SP angebracht ist, einfach durch Angabe des Zielparameters, wie in Tabelle 6.2 dargestellt.

**HINWEIS** Wenn ein einzelnes SM-Applications-Modul am Unidrive SP angebracht ist, können normale SM-Applications-Parameterreferenzen problemlos verwendet werden, da sie vom SM-PROFIBUS-DP automatisch zum SM-Applications-Modul umgeleitet werden.

### 6.3.1 Beispielkonfiguration 1

Betrachten Sie einen Unidrive SP mit der folgenden Konfiguration:

- Steckplatz 1 - Frei
- Steckplatz 2 - SM-Applications-Modul
- Steckplatz 3 - SM-PROFIBUS-DP-Modul

Wenn über das PROFIBUS-DP-Netzwerk eine Anforderung zum Abfragen von Parameter #71.08 ankommt, wird diese an das SM-Applications-Modul in dem Steckplatz mit der niedrigsten Nummer weitergeleitet, d. h. Steckplatz 2. Der Wert in \_Q08% von Steckplatz 2 wird zurückgegeben.

Wenn über das PROFIBUS-DP-Netzwerk eine Anforderung zum Abfragen von Parameter #131.08 ankommt, wird diese direkt an das SM-Applications-Modul in Steckplatz 2 gesendet. Der Wert in \_Q08% von Steckplatz 2 wird zurückgegeben.

Wenn über das PROFIBUS-DP-Netzwerk eine Anforderung zum Abfragen von Parameter #101.08 ankommt, wird diese direkt an das SM-Applications-Modul in Steckplatz 1 gesendet. Da an Steckplatz 1 kein SM-Applications-Modul angebracht ist, wird eine Fehlermeldung zurückgegeben, dass der Parameter nicht vorhanden ist.

Wenn beispielsweise ein SM-DeviceNet-Modul an Steckplatz 1 angebracht ist, könnten Sie die direkte Parameterreferenz des Steckplatzes verwenden, um Daten abzufragen oder zu schreiben, wodurch ein einfacher Kommunikations-Gateway zwischen DeviceNet und Profibus-DP entsteht.

### 6.3.2 Beispielkonfiguration 2

Betrachten Sie einen Unidrive SP mit der folgenden Konfiguration:

- Steckplatz 1 - SM-Applications-Modul
- Steckplatz 2 - SM-Applications-Modul
- Steckplatz 3 - SM-PROFIBUS-DP-Modul

Wenn über das PROFIBUS-DP-Netzwerk eine Anforderung zum Abfragen von Parameter #71.08 ankommt, wird diese an das SM-Applications-Modul in dem Steckplatz mit der niedrigsten Nummer weitergeleitet, d. h. Steckplatz 1. Der Wert in `_Q08%` von Steckplatz 1 wird zurückgegeben.

Wenn über das PROFIBUS-DP-Netzwerk eine Anforderung zum Abfragen von Parameter #131.08 ankommt, wird diese direkt an das SM-Applications-Modul in Steckplatz 2 gesendet. Der Wert in `_Q08%` von Steckplatz 2 wird zurückgegeben.

Wenn über das PROFIBUS-DP-Netzwerk eine Anforderung zum Abfragen von Parameter #101.08 ankommt, wird diese direkt an das SM-Applications-Modul in Steckplatz 1 gesendet. Der Wert in `_Q08%` von Steckplatz 1 wird zurückgegeben.

**HINWEIS** Wenn zwei SM-Applications-Module am Unidrive SP angebracht sind, ist es am besten, mit Hilfe der direkten Parameterreferenzen für die Steckplätze auf die Parameter des SM-Applications-Moduls zuzugreifen. Wenn normale SM-Applications-Modulparameter verwendet werden und das SM-Applications-Modul aus Steckplatz 1 entfernt wird, werden diese Parameterreferenzen stattdessen zu Steckplatz 2 umgeleitet.

### 6.3.3 Beispielkonfiguration 3

Betrachten Sie einen Unidrive SP mit der folgenden Konfiguration:

- Steckplatz 1 - SM-Applications-Modul
- Steckplatz 2 - SM-Applications-Modul
- Steckplatz 3 - SM-Applications-Modul

Mit dem SM-Applications-Modul ist das Abfragen oder Schreiben in direkte Parameterreferenzen für Steckplätze nicht möglich, so dass die Datenübertragung auf andere Art und Weise erfolgen muss.

Wenn mit dem SM-Applications-Modul in Steckplatz 1 Register #71.08 aus dem SM-Applications-Modul in Steckplatz 3 abgefragt werden sollte, müsste dies durch einen der 32 Bit-Anwendungsparameter erfolgen (#20.21 bis #20.40). Mit dem SM-Applications-Modul in Steckplatz 1 müssen die Registerdaten in den Anwendungsparameter geschrieben werden. Mit dem SM-Applications-Modul in Steckplatz 3 kann anschließend dieser Parameter abgefragt werden.

Code in Steckplatz 1:

```
#20.21 = #71.08
```

Code in Steckplatz 3:

```
#71.08 = #20.21
```

Der Zugang zu diesen Daten ist auch mit dem SM-Applications-Modul in Steckplatz 2 möglich.

**HINWEIS** Mit dem SM-Applications-Modul ist kein Zugang zu den internen Menüs (#1xx.xx) möglich. Der Zugang muss direkt über die Parameterreferenznummer erfolgen. Beispiel: Wenn Sie Parameter #104.35 abfragen möchten, müssen Sie #74.35 verwenden. Durch den oben angegebenen Beispielcode wird die Aktualisierung von Parameter #71.08 in den 2 Modulen nicht synchronisiert. Weitere Informationen finden Sie in Abschnitt 10 *Synchronisation zwischen den Optionsmodulen*. Wenn die SM-Applications-Module an dasselbe CTNet-Netzwerk angeschlossen sind, kann dies mit Hilfe von zyklischen Daten geschehen. Weitere Informationen zu CTNet finden Sie in der CTNet-Betriebsanleitung.

---

# 7 DPL-Programmierung

---

Dieses Kapitel umfasst Folgendes:

- Grundlegende Struktur und Syntax eines DPL-Programms
- Grundlegende DPL-Befehle
- Neue Funktionen im SM-Applications-Modul



Dies ist keineswegs ein vollständiges Handbuch der DPL-Sprache. Die vollständige Anleitung zu allen DPL-Befehlen und Funktionsblöcken ist in der Online-Hilfe zu finden.

## 7.1 Programmkopfzeile

Jedes DPL-Programm beginnt mit einem Kopfabschnitt. Dieser Abschnitt wird von der SYPT Workbench für den Benutzer erstellt. Er besteht aus den folgenden Grundelementen:

- Programmtitel
- Autor des Programms
- Versionsnummer des Programms

### 7.1.1 Aliasnamen

Unmittelbar unterhalb der Kopfzeile kann der Benutzer einen Abschnitt mit *Aliasnamen* eingeben. Aliasnamen werden verwendet, um verschiedene Ausdrücke oder Konstanten zu „ersetzen“:

- Eine numerische Konstante
- Die Adresse eines Registers oder Parameters
- Einen DPL-Ausdruck oder eine DPL-Anweisung

Aliasnamen werden mit der \$DEFINE-Anweisung erstellt.

```
$DEFINE Namenwert
```

Zum Beispiel ist es empfehlenswert, mit Hilfe von Aliasnamen allen in einem Programm verwendeten Umrichterparametern Namen zu geben.

```
$DEFINE PRESET_REF_1 #1.21  
$DEFINE PRESET_REF_2 #1.22  
$DEFINE SPEED_FB #3.02
```

Außerdem wird empfohlen, den Aliasnamen in GROSSbuchstaben einzugeben, um ihn von normalen Variablen unterscheiden zu können.

#### HINWEIS

Es wird empfohlen, an Aliasnamen, die für ganzzahlige Werte stehen, ein „%-Symbol anzuhängen. In graphischen Programmier-Tools (QLD/FBD) werden alle Aliasnamen ohne %-Symbol von SYPT als Gleitkomma-Werte behandelt. Daher werden sie an LD-Eingängen oder auf ganze Zahlen beschränkten Eingängen abgelehnt.

Mit der \$DEFINE-Anweisung wird KEIN Code erzeugt, und die Ausführung Ihres Programms wird nicht beschleunigt - Sie haben lediglich die Möglichkeit, andere Namen zu vergeben.

## 7.2 Tasks

Ein DPL-Programm wird in separate Abschnitte, so genannte Tasks, unterteilt. Innerhalb der Tasks schreibt ein Benutzer die Programmanweisungen, die vom Mikroprozessor unter bestimmten Bedingungen oder auf einer bestimmten Zeitbasis



ausgeführt werden. Jeder Task besitzt einen bestimmten Namen, einen Zweck und eine Priorität, und jeder Task kann in dem DPL-Programm nur einmal vorkommen. Die gebräuchlichen Tasks werden im Folgenden skizziert:

**Tabelle 7.1 Gebräuchliche Tasks**

Task-Name	Priorität	Zweck
INITIAL	3	Der allererste Task, der nach dem Einschalten oder einem Reset ausgeführt wird. Dieser Task wird häufig verwendet, um Umrichterparameter und Programmvariablen zu initialisieren. Vor Beendigung dieses Tasks können keine anderen Tasks ausgeführt werden.
BACKGROUND	1	Task mit niedriger Priorität, der für nicht zeitkritische Funktionen verwendet wird. Dieser Task funktioniert sehr ähnlich wie der Fangregelkreis einer SPS. Normalerweise wird dieser Task-Abschnitt als eine große Schleife erstellt, wobei am Ende des Tasks ein Befehl zur Rückkehr an den Anfang steht. Wenn der Task abgeschlossen wurde, wird er nicht erneut ausgeführt.
CLOCK	2	Task, der auf einer festen Zeitbasis ausgeführt wird (zwischen 1 und 200 ms) und zur Ausführung einiger zeitbezogener Vorgänge dient, zum Beispiel zum Generieren eines Rampenprofils. Dieser Task ist jetzt mit dem Regelkreis der Ebene 2 des Umrichters synchronisiert und kann an Stelle des alten Encoder-Tasks verwendet werden.
POS0 POS1	4	Zwei Echtzeit-Tasks, die mit einem Vielfachen der Umrichter-Regelkreise synchron laufen (Wertebereich: 250 $\mu$ s bis 8 ms). Diese Tasks werden häufig verwendet, um den Umrichter-Regelkreis für Drehzahl und/oder Strom in Anwendungen wie der Positionierung zu steuern. Zuerst wird der POS0-Task ausgeführt, unmittelbar gefolgt vom POS1-Task.
EVENT	5	Event-Tasks werden nur ausgeführt, wenn ein bestimmtes Ereignis eintritt. Ereignisse können aus verschiedenen Quellen stammen, zum Beispiel aus CTNet, aus anderen Optionsmodulen im Unidrive SP oder aus dem Benutzerprogramm. EVENT-Tasks besitzen höchste Priorität und bestehen daher normalerweise nur aus sehr wenigen Anweisungen. Sie sind mit Dienstunterbrechungsroutrinen vergleichbar.
EVENT1	5	Siehe Beschreibung oben.
EVENT2	5	Siehe Beschreibung oben.
EVENT3	5	Siehe Beschreibung oben.
ERROR	1	Ein Task, der nur dann ausgeführt wird, wenn innerhalb des benutzerdefinierten DPL-Programms ein Laufzeitfehler auftritt (zum Beispiel eine Division durch Null). Auf diese Weise kann ein anomales Programmverhalten sicher korrigiert werden. Vor dem Ausführen des ERROR-Tasks werden alle anderen Tasks angehalten.

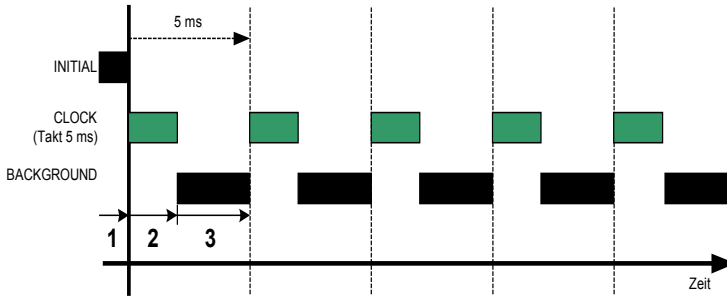
**HINWEIS** Beim Arbeiten mit den Tasks CLOCK, POS0 und POS1 ist es ratsam, Code wie die FOR- und DO WHILE-Schleifen zu vermeiden. Dies kann einen DPL-Überlauffehler (tr54) verursachen.

**HINWEIS** Die UD70-Tasks ENCODER und SPEED können weiterhin verwendet werden. Diese sind jetzt Aliasnamen für den Task POS0 bzw. POS1 (d. h. wenn das Programm über einen ENCODER-Task verfügt, ist dies dasselbe wie ein POS0-Task). Die Zeitbasis für beide Tasks ist nicht wie beim UD70 fest, sondern wird vom Benutzer angegeben. Der CLOCK-Task im SM-Applications-Modul kann an Stelle des ENCODER-Tasks im UD70 verwendet werden. So ergibt sich eine Zeitbasis, die näher an diejenige des UD70-Tasks ENCODER heranreicht, als dies mit den Tasks POS0 und POS1 möglich ist.

Alle Programmanweisungen **müssen** in einen Task eingebettet sein. Für zeitbasierte Tasks wie POS0, POS1 und CLOCK müssen die Anweisungen innerhalb des Tasks in einer endlichen Zeit abgeschlossen werden. Daher sollten nur zeitkritische Funktionen in diesem Rahmen ausgeführt werden.

Die Positions-Tasks bestehen aus POS0, APR und POS1 und werden gegebenenfalls in der folgenden Reihenfolge ausgeführt (d. h. falls sie entsprechend eingestellt sind).

Tasks besitzen unterschiedliche Prioritätsebenen. Daher kann ein Task durch einen anderen Task unterbrochen werden. In der Tabelle oben gilt: Je höher die Prioritätszahl, desto höher die Priorität. Daher kann ein BACKGROUND-Task durch einen CLOCK-Task unterbrochen werden, der wiederum durch einen POS0-Task unterbrochen werden kann. In dem folgenden einfachen Diagramm wird das Konzept des gegenseitigen Unterbrechens von Tasks veranschaulicht:

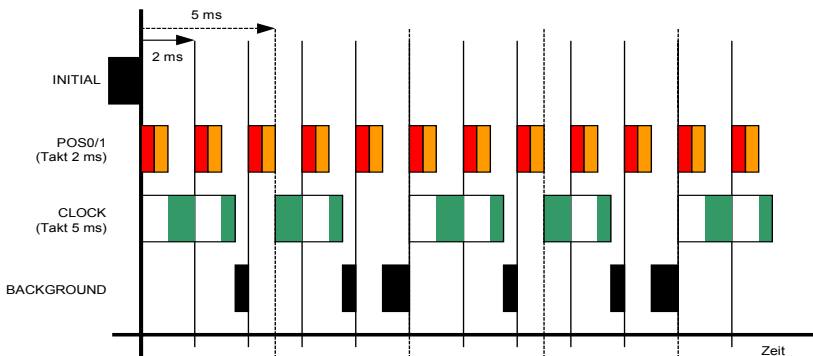


Schlüssel:

1. Ausschließliche Steuerung durch den INITIAL-Task. Keine anderen Tasks möglich.
2. CLOCK-Task mit höherer Priorität als die Durchläufe des BACKGROUND-Tasks.
3. CLOCK-Task abgeschlossen, jetzt kann der BACKGROUND-Task ausgeführt werden - bis zum nächsten Takt.

Beachten Sie besonders, dass der CLOCK-Task auf einer festen Zeitbasis ausgeführt wird. (In dem Diagramm oben sind dies 5 ms.) Dies bedeutet, dass die Anweisungen innerhalb des CLOCK-Tasks innerhalb von weniger als 5 ms abgeschlossen werden MÜSSEN. Andernfalls kann der BACKGROUND-Task keinen Einblick erhalten, oder eine Fehlerabschaltung wegen Prozessorüberlastung wird ausgelöst.

Das folgende Diagramm zeigt, was geschieht, wenn auch die POS-Tasks verwendet werden:



Dies zeigt, dass der BACKGROUND-Task vom CLOCK-Task unterbrochen wird, der wiederum von den Tasks POS0 und POS1 unterbrochen wird. Es ist zu erkennen, dass dieses Programm recht stark belastet ist, da der Background-Task nur gelegentlich ausgeführt wird. Mit Hilfe des Parameters für freie Prozessorressourcen (#81.04) kann bestimmt werden, wie stark das SM-Applications-Modul belastet ist.

## 7.2.1 EVENT-Tasks

Vier Event-Tasks werden bereitgestellt. Die Event-Tasks können durch Folgendes ausgelöst werden:

- Empfangen eines CTNet SYNC-Rahmens (konfiguriert über #81.35).
- Starten des Benutzerprogramms.  
Neuer DPL-Befehl SCHEDULEEVENT. Informationen finden Sie in der Online-Hilfe.

## 7.3 Variablen

### 7.3.1 Typen

Es gibt drei grundlegende Variablentypen:

1. Ganzzahlige Variablen
2. Gleitkomma-Variablen mit doppelter Präzision
3. Gleitkomma-Variablen mit einfacher Präzision

Eine ganzzahlige Variable ist durch ein %-Symbol hinter dem Variablennamen gekennzeichnet. Eine Gleitkomma-Variable ist durch das Fehlen des %-Symbols gekennzeichnet.

**Tabelle 7.2 Variablentypen**

Typ	Darstellung	Bereich
Ganze Zahl	32 Bit mit Vorzeichen	-2147483648 bis 2147483647
Gleitkomma einfach	32 Bit, 1 Vorzeichenbit, 8 Bit Exponent und 23 Bit Mantisse.	$\pm 3,40282e+038$
Gleitkomma doppelt	64 Bits: 1 Vorzeichenbit, 52 Bit Mantisse, 11 Bit Exponent	$\pm 1,79769e+308$

Beispiel für Variablen:

```
speed% = 1234 // Ganzzahlige Variable  
value = 55.6 // Gleitkomma-Variablen
```

Eine spezielle Anweisung wird am Anfang des Programms platziert, um festzulegen, welcher Typ von Gleitkomma-Variablen in dem ganzen Programm verwendet wird - entweder einfache oder doppelte Präzision. Standardmäßig werden Variablen mit doppelter Präzision verwendet. Durch Einfügen der folgenden Zeile direkt unterhalb des Programmkopfbereichs (mit \$TITLE usw.) wird der Gleitkomma-Typ auf einfache Präzision festgelegt:

```
$flt single
```

### 7.3.2 Variablennamen

Das erste Zeichen eines Variablennamens muss ein Buchstabe sein. Für die nachfolgenden Zeichen sind Buchstaben, Zahlen und der Unterstrich (\_) zulässig.

- HINWEIS**
- Bei Variablennamen wird die Groß- und Kleinschreibung beachtet. (Beispiel: Die Variablennamen speed%, SPEED% und Speed% stehen für unterschiedliche Variablen.)
  - Im SYPT QuickLD-Editor und im SYPT FBD-Editor dürfen nur Variablen verwendet werden, die nicht länger sind als 16 Zeichen, einschließlich eines %-Zeichens.

### 7.3.3 Initialisierung von Variablen

Allen Variablen muss ein Anfangswert gegeben werden, bevor sie verwendet werden können. Dies geschieht normalerweise mit Hilfe des INITIAL-Tasks. Beispiel:

```
Initial {  
  speed_sp% = 0  
  ramp% = 0  
}
```

### 7.3.4 Umfang und Lebensdauer von Variablen

Variablen können entweder global oder lokal sein. Alle in DPL-Programmen deklarierten Variablen sind global, d. h. Zugang und Änderung sind durch jeden Task möglich, mit Ausnahme von Variablen innerhalb eines benutzerdefinierten Funktionsblocks, die lokal sind (d. h. ein Zugang von außerhalb des benutzerdefinierten Funktionsblocks ist nicht möglich). Keine DPL-Variablen bleiben nach einem Reset des SM-Applications-Moduls erhalten. Denken Sie daran, dass ein Zurücksetzen des Umrichters aus einem Fehlerabschaltungszustand ebenfalls zu einem Reset führt.

### 7.3.5 Variablen-Arrays mit fester Größe

Ein DPL-Programm kann Arrays entweder mit ganzzahligen Variablen oder mit Gleitkomma-Variablen enthalten. Nur Arrays mit fester Größe (eindimensionale Arrays) sind zulässig. Ein Array muss zunächst mit Hilfe der DIM-Anweisung deklariert werden (normalerweise im Initial-Task), und die Anzahl der Elemente ist in eckigen Klammern hinter dem Variablennamen anzugeben. Beispiel:

```
DIM myarray%[20]    // Array aus ganzen Zahlen mit 20 Elementen
DIM array2[30]     // Gleitkomma-Array mit 30 Elementen
```

Die Elemente in einem Array werden von 0 bis Anzahl\_der\_Elemente - 1 durchnummeriert. In dem oben genannten Beispiel heißt also das erste Element von myarray%[]:

```
myarray%[0]
```

Und das letzte Element heißt:

```
myarray%[19]
```

Es werden zwei Funktionen bereitgestellt, die zur Laufzeit verwendet werden können, um die obere und untere Grenze eines Arrays zu bestimmen. Diese heißen UPPER und LOWER. Bei myarray%[] ergibt sich für UPPER der Wert 19 und für LOWER der Wert 0.

### 7.3.6 Konstante Arrays

Konstante Arrays enthalten, wie der Name schon sagt, feste, vordefinierte Werte. Die Werte des konstanten Arrays werden innerhalb des DPL-Programms mit Hilfe eines speziellen Abschnitts definiert (siehe das Thema CONST in der Online-Hilfe). Es können nur ganzzahlige Werte definiert werden.

Der Vorteil von konstanten Arrays besteht darin, dass die Größe des Arrays nur durch den verfügbaren Programmspeicherplatz begrenzt ist - und nicht durch den variablen RAM. Der Programmspeicher ist 384 KB groß - in ihm werden die kompilierte DPL-Datei, die Daten für das konstante Array und optional die DPL-Datei selbst gespeichert.

### 7.3.7 Speicherplatz - Anzahl der Variablen

Alle Variablen, dynamischen Arrays und SPS-Register werden in einem 80 Kilobyte großen Speicher gespeichert. Jede ganzzahlige Variable und jede Gleitkomma-Variable mit einfacher Präzision belegt 4 Byte (32 Bit), und Gleitkomma-Variablen mit doppelter Präzision belegen 8 Byte (64 Bit). Andere Elemente nehmen ebenfalls Speicherplatz in Anspruch, zum Beispiel Parameterzugänge.

Sie werden vom DPL-Kompilierer benachrichtigt, wenn Sie die Grenze des verfügbaren Speicherplatzes erreichen.

### 7.3.8 Bit-Adressierung von Variablen

Alle ganzzahligen Variablen und Arrays können mit Bit-Adressen versehen werden. Dies bedeutet, dass jedes einzelne Bit innerhalb der Variablen separat abgefragt oder geschrieben werden kann. Setzen Sie hinter den Variablennamen einen Punkt (.), gefolgt von der Bit-Nummer zwischen 0 und 31.

Beispiel 1 (einfache Variable):

```
flags% = 0 // Alle 32 Bits auf 0 initialisieren
flags%.0 = 1 // Bit 0 auf 1 setzen

// Jetzt testen, ob Bit 0 UND Bit 1 auf 1 gesetzt sind.
IF flags%.0 & flags%.1 = 1 THEN
    PRINT "Test erfüllt."
ENDIF
```

Beispiel 2 (Array):

```
DIM myarray%[10]
...
IF myarray%.1[4] = 1 THEN;test bit 1 of element #4.
    PRINT "Test erfüllt."
ENDIF
```

Hinweis: Die Bit-Nummer muss eine konstante Zahl sein - Variablen sind nicht zulässig.

### 7.3.9 SPS-Register

Der Bereich „SPS“ ist ein spezieller Bereich von vordefinierten 32 Bit-Registern. Die SPS-Register sind in 6 Sätze zu je 100 Parametern mit den Nummern 00 bis 99 aufgeteilt. Der Zugang zu den Registern ist auch aus einem benutzerdefinierten DPL-Programm heraus durch einen speziellen Variablennamen oder Array-Namen möglich. Vier der Registersätze können außerdem im SM-Applications-Modul im *Flash*-Speicher gespeichert werden.

Weitere Informationen zu SPS-Registern finden Sie in Abschnitt 5.4 *Menüs 70 bis 75 - SPS-Register*.

### 7.3.10 RAM-Dateien

Mit Hilfe von RAM-Dateien kann der Benutzer „Dateien“ im Benutzer-RAM des SM-Applications-Moduls speichern. Diese können mit Hilfe von DPL-Befehlen hoch- und heruntergeladen werden. Sie haben den Vorteil, dass Sie ein Array von Zahlen in einem Schritt abrufen oder schreiben können, statt jedes Element des Arrays einzeln zu bearbeiten.



Weitere Informationen zu RAM-Dateien, einschließlich Beispielprogrammen, finden Sie in der Online-Hilfe.

## 7.4 Parameter

Parameter sind in zwei Kategorien unterteilt:

- Umrichterparameter
- SM-Applications-Parameter

Umrichterparameter sind solche, die im Host-Umrichter gespeichert sind. Die Mehrzahl davon hat Auswirkungen auf den Betrieb des Umrichters, jedoch werden einige als „Anwendungsparameter“ bereitgestellt. Dies sind die Menüs 18, 19 und 20.

Die SM-Applications-Parameter sind lokal, und der Zugang ist nur über das SM-Applications-Modul möglich. Diese Parameter ermöglichen den Zugang zu zusätzlichen Funktionen des SM-Applications-Moduls sowie einen schnelleren Zugang zu einigen Umrichterparametern.

**HINWEIS**

Im SM-Applications-Modul ist immer gewährleistet, dass die verwendete Umrichterparameter-Datenbank mit der des als Host fungierenden Unidrive SP übereinstimmt. Wenn ein SM-Applications-Modul zum ersten Mal an einem Unidrive SP angebracht und eingeschaltet wird, kann für ein paar Sekunden das Wort „Loading“ auf dem Display des Umrichters angezeigt werden. Damit wird angezeigt, dass die Datenbanken vom SM-Applications-Modul synchronisiert werden. Dies geschieht nur, wenn das Modul zum ersten Mal an dem Umrichter angebracht wird. Bei nachfolgenden Einschaltvorgängen wird das Wort „Loading“ nur für sehr kurze Zeit angezeigt.

### 7.4.1 Abfragen und Schreiben von Parametern

Das Abfragen und Schreiben von Parametern erfolgt mit Hilfe des #-Befehls. Der Zugang zu Parametern erfolgt wie bei der Bedieneinheit des Umrichters im Format „#MM.PP“. Verwenden Sie zum Beispiel den folgenden Befehl, den Drehzahlwert-Parameter (Parameter 03.02) abzufragen:

```
speed% = #3.02
```

Verwenden Sie zum Schreiben in einen Drehzahlsollwert-Parameter (z. B. 01.22) den folgenden Befehl:

```
#01.22 = 1500
```

Beachten Sie, dass die führende Null im Menü- bzw. Parameterfeld optional ist. Zum Beispiel erhalten Sie mit den Befehlen #3.02, #03.02, #03.2 und #3.2 Zugang zu genau demselben Parameter.

### 7.4.2 Festkomma-Parameter

Das Arbeiten mit Festkomma-Parametern kann wesentlich zeitaufwendiger sein als bei ganzzahligen Parametern. Um dies zu beschleunigen, kann beim Abfragen und Schreiben von Parametern der spezielle Befehl #INT verwendet werden. Wenn dieser Befehl auf Festkomma-Zahlen angewendet wird, werden die Dezimalstellen automatisch entfernt.

Beispiel: Der Parameter #1.19 besitzt einen Wertebereich von 0,000 bis 0,099. Beim Abfragen des Parameters mit dem Befehl:

```
speed_fine% = #INT1.19
```

werden ganzzahlige Werte zwischen 0 und 99 zurückgegeben. Beim Schreiben wird mit dem Befehl:

```
#INT1.19 = 45
```

der Parameter auf 0,045 gesetzt (wie mit #1.19=0.045). Der Vorteil besteht darin, dass in dem DPL-Programm ganzzahlige Variablen (%) an Stelle von Gleitkomma-Variablen verwendet werden können, was eine höhere Geschwindigkeit bedeutet.

## 7.5 Operatoren

DPL bietet alle Standardoperatoren in folgender Form:

**Tabelle 7.3 Standardoperatoren in der Reihenfolge der Priorität**

Operator	Bedeutung
-	Arithmetische Negation
!	Boolesche Negation (unär)
!(..., nbit)	Negation von <i>nbit</i> Bits

**Tabelle 7.3 Standardoperatoren in der Reihenfolge der Priorität**

Operator	Bedeutung
*	Multiplikation
/	Division
%	Modulo (Rest)
+	Addition
-	Subtraktion
&	Bit-weises UND
	Bit-weises ODER
^	Bit-weises exklusives ODER (XOR)

**Tabelle 7.4 Konditionale Operatoren in der Reihenfolge der Priorität**

Operator	Bedeutung
=	Gleichheit
<	Kleiner als
>	Größer als
<=	Kleiner oder gleich
>=	Größer oder gleich
<>	Ungleichheit
AND	Boolesches UND
OR	Boolesches ODER
NOT	Boolesches NICHT

## 7.6 Grundlegende DPL-Befehle

Die für das SM-Applications-Modul implementierte DPL-Sprache ist rückwärtskompatibel mit dem UD70-Produkt für Unidrive 1, besitzt jedoch ein paar neue Erweiterungen.



Die vollständige Anleitung zu DPL-Sprache und Funktionsblock-Bibliothek finden Sie in der Online-Hilfe.

### 7.6.1 Neue Befehle für das SM-Applications-Modul

Dies sind neue Befehle, die für das SM-Applications-Modul in die DPL-Sprache eingeführt wurden und im UD70 für Unidrive 1 nicht vorhanden waren.

#### FOR-Schleife

Dies ist eine Neuerung im SM-Applications-Modul.

```
FOR variable = integer_expression to integer_expression [STEP constant]
    Anweisungen
LOOP
```

#### CASE

Dies bietet eine Alternative zu der Konstruktion IF-ELSEIF-ENDIF.

```
SELECT integer_expression
    CASE integer_constant
        Anweisungen
    [CASE integer_constant, integer_constant ...
        [Anweisungen]]
    [ELSE
        [Anweisungen]]
ENDSELECT
```

Diese Konstruktion stellt eine bequeme Methode dar, um Tests auf mehrere konstante Werte durchzuführen. Eine beliebige Anzahl von CASE-Anweisungen kann integriert werden.

**HINWEIS** In den zwei oben dargestellten Beispielen sind einige Abschnitte in eckige Klammern ([ und ]) eingeschlossen. Dieser Codeabschnitt innerhalb der eckigen Klammern ist optional.

**HINWEIS** Die CASE-Anweisungen funktionieren insofern genauso wie Programme wie Visual Basic, als der Programmfluss NICHT wie in der Programmiersprache C zum nächsten CASE springt.

### MAX\_INT, MIN\_INT, MIN\_FLOAT, MAX\_FLOAT

Dies sind spezielle vordefinierte Schlüsselwörter, die vom DPL-Kompilierer erkannt und durch den entsprechenden numerischen Wert ersetzt werden.

**Tabelle 7.5 Min/Max**

Schlüsselwort	Wert
MIN_INT	-2147483648
MAX_INT	2147483647
MIN_FLOAT	-3,40282e+038 (Modell mit einfacher Präzision) -1,79769e+308 (Modell mit doppelter Präzision)
MAX_FLOAT	3,40282e+038 (Modell mit einfacher Präzision) 1,79769e+308 (Modell mit doppelter Präzision)

### UPPER/LOWER

Bei diesen Funktionen wird ein Array als Parameter verwendet, und der obere bzw. untere Array-Indexwert wird zurückgegeben. Beispiel:

```
// Array mit 1000 Elementen erstellen
DIM array%(1000)

// Jetzt:

l% = LOWER(array%) // Der Wert 0 wird zurückgegeben
u% = UPPER(array%) // Der Wert 999 wird zurückgegeben.

// Die Summe aller werte in array% abrufen
total%=0
FOR i% = LOWER(array%) TO UPPER(array%)
total% = total% + array%(i%) //Den wert des Array-Elements zur Summe
addieren
LOOP
```

### TRUNC

Dies wird verwendet, um einen Gleitkomma-Wert in eine ganze Zahl zu konvertieren, wobei der Wert abgeschnitten und nicht gerundet wird.

Beispiel:

```
// Gleitkomma-Variablen initialisieren
floatval = 1.56

int1% = floatval // Automatische Saldierung bedeutet Rundung auf 2.
int2% = INT(floatval) // Explizite Saldierung mit INT bedeutet Rundung
auf 2
int3% = TRUNC(floatval) // Explizite Saldierung mit TRUNC ergibt 1
```



## SCHEDULEEVENT

Dieser Funktionsblock wird für die Zeitkoordination eines EVENT-Tasks verwendet. Die Argumente sind:

- Steckplatznummer  
*Der Steckplatz, in dem der Event-Task geplant werden soll, wird angegeben. Zurzeit ist hier nur der Wert 0 zulässig, der für den lokalen Steckplatz steht.*
- Task-Kennung  
*Bereich 0 bis 3 für die Angabe, welcher EVENT-Task ausgelöst werden soll.*
- Grund  
*Ein benutzerdefinierter Grund. Der Wert muss mindestens gleich 34 sein. Dieser Wert ist über den EVENT-Task (Parameter #90.12 bis #90.15) zugänglich.*

```
BACKGROUND {
... etwas Code
// Zeitkoordination für lokalen event1-Task mit Grundcode 45.
a% = SCHEDULEEVENT(0, 1, 45)
... weiterer Code
}

EVENT1 {
IF #90.13 = 45 THEN
// Task mit Zeitkoordination aus DPL
ENDIF
}
```

## CTNETDIAGNOSTICS

Mit diesem Befehl werden Daten zur CNet-Fehlersuche zurückgegeben. Weitere Informationen finden Sie in der Online-Hilfe. Dieser Befehl tritt an die Stelle der speziellen Variablen wie NOFMESSAGES, die beim UD70-Produkt verwendet wurden. Für diesen Befehl wird kein Eingang angenommen, und 10 Ausgänge werden zurückgegeben. In der folgenden Tabelle sind die Ausgänge und die jeweils äquivalente UD70-Variable detailliert aufgeführt.

**Tabelle 7.6 CNetDiagnostics-Ausgangswerte**

Ausgang	Wert	UD70-Variablen
1	Gesamtzahl der von diesem Knoten bearbeiteten Meldungen	NOFMESSAGES
2	Anzahl der zyklischen Datenüberläufe	NOFOVERRUNS
3	Verloren gegangene RX-Meldungen	NOFLOSTMESSAGES
4	Anzahl der Wiederholungsversuche	NOFRETRIES
5	Anzahl der Neuverbindungen	NOFRECONS
6	Anzahl der überzähligen NAKs	NOFEXENAKS
7	Doppelte Synchronisation	NOFDUPSYNCS
8	Anzahl der lokal generierten Neuverbindungen	NOFMYRECONS
9	Anzahl der nicht zyklischen Meldungen	NOFNONCYCLICMESSAGES
10	Anzahl der verloren gegangenen Weiterleitungsmeldungen	n.b.

## GETPARATTR

Dies wird verwendet, um Parameterattribute abzurufen, zum Beispiel Höchst- und Mindestwerte, Schreibschutz-Flag usw.

```
(max%, min%, flags%) = GETPARATTR(menu%, par%)
```

## **CModeXfer**

Dies ermöglicht es dem Benutzer, den Umrichtermodus zu ändern, ohne das angebrachte Optionsmodul einem harten Reset zu unterziehen. Dadurch kann der Umrichtermodus reibungsloser geändert werden. Während der Änderung des Umrichtermodus ist ein Schreiben von Feldbussen in Parameter nicht möglich, und dies wird auf Systemebene bearbeitet. Während dieser Zeit wird KEINE Meldung ausgegeben, dass das Schreiben in den Parameter fehlgeschlagen ist.

**HINWEIS** Die Menüs 15 bis 20 werden durch diesen Befehl nicht auf die Standardwerte zurückgesetzt.

### **RtuReadRegs**

### **RtuReadParas**

### **RtuReadInputRegs**

### **RtuPresetRegs**

### **RtuPresetParas**

### **RtuMasterReply**

### **RtuMasterStatus**

Diese Befehle sind implementiert, damit der Benutzer die Modbus-RTU-Masterfunktionalität des SM-Applications-Moduls nutzen kann. Weitere Details finden Sie in der Online-Hilfe.

### **PFIXREAD6/PFIXWRITE6**

Diese Blöcke ermöglichen das Abfragen und Schreiben von Umrichterparametern mit einer festen Genauigkeit von 6 Dezimalstellen.

### **SETUSERID**

Mit Hilfe dieses Befehls wird der Benutzerkennungsparameter #81.49 eingestellt.

```
SETUSERID(101) // #81.49 auf 101 setzen.
```

### **ANSIREPLY**

Diesen Befehl gab es im UD70, jedoch wurde die Syntax für das SM-Applications-Modul geändert.

```
(status%, reply%) = ANSIREPLY()
```

Es handelt sich um eine RS485-Anschlussfunktion, die in Verbindung mit den Funktionen ANSIREAD und ANSIWRITE verwendet wird.

Beispiele für den Befehl ANSIREPLY:

```
result% = ANSIREADN(12, 1811) //ANSI-Abfrage ausführen
```

```
//wurde die Meldung erfolgreich gesendet?
```

```
IF result% = 0 THEN
```

```
    //Meldung nicht erfolgreich gesendet
```

```
    goto top:
```

```
ENDIF
```

```
timeout% = 0 //Meldung erfolgreich gesendet, also
```

```
Zeitbereichsüberschreitung initialisieren
```

```
DO
```

```
    (status%, reply%) = ANSIREPLY()// Status und wert der Abfrage abrufen
```

```
    DELAY(1) // Verzögerung 100 ms
```

```
LOOP WHILE status% = -65536 AND timeout% < 50 //
```

```
Zeitbereichsüberschreitung = 50 x clock-Zeitbasis
```

```

Background{
top:

value% = #18.11
result% = ANSIWRITEN(12, 1811, value%, 1);write value to remote drive
IF result% = 0 THEN
    //Meldung nicht erfolgreich gesendet
    goto top:
ENDIF

CALL get_reply: //Antwort abrufen

GOTO top:
} //Background

get_reply:{
timeout% = 0
DO
    (status%, reply%) = ANSIREPLY()
LOOP WHILE status% = -65536 AND timeout% < 50
} //get_reply:

```

Mit dem ersten Ausgangsargument wird der Status des Befehls ANSIREPLY zurückgegeben. Es kann einen der folgenden Werte aufweisen:

- 65536 = Noch keine Antwort empfangen
- 65537 = Antwort empfangen, jedoch mit fehlerhafter Prüfsumme
- 65538 = EOT empfangen (d. h. Parameter existiert nicht)
- 65539 = NAK empfangen
- 65540 = ACK empfangen

#### **AssRAM**

#### **UnassRAM**

#### **RamLength**

#### **SetRamLength**

Mit Hilfe dieser Befehle kann der Programmierer die RAM-Dateien innerhalb des SM-Applications-Moduls verwenden. RAM-Dateien stellen ein Mittel für den Zugang zu Benutzerprogramm-Arrays über die CMP-Dateidienste dar. Weitere Informationen zu diesen Befehlen und zu RAM-Dateien finden Sie in der Online-Hilfe.

#### **OpenReadSmartCard**

#### **OpenWriteSmartCard**

#### **CloseSmartCard**

#### **ReadSmartCardByte**

#### **WriteSmartCardByte**

#### **GetNextSmartCardFile**

#### **ReadReadOnlyBit**

#### **WriteReadOnlyBit**

Mit Hilfe dieser Befehle kann der Benutzer die SMARTCARD des Unidrive SP verwenden. Weitere Informationen erhalten Sie bei Ihrem Umrichterzentrum.

### **7.6.2 DPL-Befehle und Funktionsblöcke**

Es gibt eine umfangreiche Liste von Befehlen und Funktionen, die in einem DPL-Programm verwendet werden können. Diese Liste ist zu umfangreich für diese Betriebsanleitung, Sie finden sie jedoch in der Online-Hilfe.

## 7.7 Benutzerdefinierte Funktionsblöcke

### 7.7.1 Überblick

SYPT verfügt standardmäßig über eine vordefinierte Bibliothek von Funktionsblöcken, die in den graphischen Programmier-Tools (LD und FBD) sowie in DPL-Rohdateien verwendet werden können.

Das System der benutzerdefinierten Funktionsblöcke ermöglicht es dem Benutzer, eigene Funktionsblöcke zu erstellen, die automatisch in den graphischen Programmier-Tools (Funktionsblockdiagramme und QuickLD-Diagramme) zusätzlich zu den Funktionen der Standardbibliothek verfügbar werden.

Ein UDFB selbst ist wie ein eigenständiger, in sich abgeschlossener DPL-Programmabschnitt und kann daher aus einer Mischung von DPL-Rohbefehlen, FBDs und QLD-Diagrammen sowie anderen UDFBs bestehen. Beachten Sie jedoch, dass Sie mit einem UDFB keine standardmäßigen Task-Abschnitte (zum Beispiel POS0) erstellen können.

### 7.7.2 Umfang eines UDFB

Jeder UDFB ist lokal für das DPL-Programm des Knotens, in dem er definiert wurde.

Um einen UDFB in anderen Knotenprogrammen verfügbar zu machen, müssen Sie lediglich den UDFB-Abschnitt kopieren und in das andere Knotenprogramm einfügen.

Ein UDFB wird im DPL-Editor von SYPT ähnlich angezeigt wie ein Task - d. h. in einem ausblendbaren Abschnitt -, und es wird empfohlen, alle UDFBs am Anfang eines Programms zu platzieren, da ein UDFB definiert werden muss, bevor er verwendet werden kann.

### 7.7.3 Kapselung und Datenspeicherung

Im Gegensatz zu einem Task eines DPL-Programms sind UDFBs in sich abgeschlossene Einheiten (d. h. gekapselt). Dies bedeutet, dass jeder UDFB einen eigenen eindeutigen Variablensatz (lokale Variablen) besitzt.

Die Schnittstelle zwischen einem UDFB und dem DPL-Programm des Knotens wird nur durch die Eingangs- und Ausgangsargumente gebildet. Ein Zugang von einem UDFB zu den globalen DPL-Variablen im DPL-Programm oder zu Variablen in anderen UDFBs ist nicht möglich.

Ein UDFB besitzt natürlich Zugangsmöglichkeiten zu Umrichterparametern und SM-Applications-Registern, die als global betrachtet werden, jedoch ist von deren Nutzung abzuraten, besonders bei Blöcken, die in anderen Programmen oder Applikationen wiederverwendet werden könnten. Die einzigen Fälle, in denen ein Block möglicherweise einen direkten Zugang zu Parametern oder Registern benötigt, wären applikations- bzw. produktspezifisch.

Bei jeder Verwendung eines UDFB in einem DPL-Programm wird eine separate Instanz erstellt. Diese ist eine Kopie des UDFB mit eindeutigen lokalen Variablen.

Hinweis: Die lokalen Variablen eines UDFB können nicht im SYPT-Beobachtungsfenster überwacht werden.

### 7.7.4 Benennung von UDFBs

Um die UDFBs eindeutig zu halten sowie Benennungskollisionen zwischen UDFBs und den Funktionsblöcken der Standardbibliothek zu vermeiden, muss ein UDFB-Name mit dem Unterstrich ( \_ ) beginnen. Außerdem ist die Länge des Namens auf 16 Zeichen begrenzt. Es wird jedoch empfohlen, den Namen kurz zu halten, damit er im SYPT QuickLD-Editor und im SYPT FBD-Editor sauber angezeigt werden kann, z. B.

`_MyFunc`, `_PID1` und `_My_Func`.

Die folgenden Beispielnamen sind nicht zulässig:

`MyFunc`, `UDFB1`

### 7.7.5 Eingangs- und Ausgangsargumente

In einem UDFB können die folgenden Datentypen empfangen und gesendet werden:

- Ganzzahlige Variablen
- Gleitkomma-Variablen
- Arrays mit ganzen Zahlen
- Gleitkomma-Arrays

Die Eingangs- und Ausgangsargumente sind standardmäßige DPL-Variablen - d. h. die Groß- und Kleinschreibung wird beachtet, und der Name muss mit einem Buchstaben beginnen, nicht mit einer Zahl. Die Länge von Eingangsargumentnamen ist nicht begrenzt, jedoch werden im FBD-Editor und im QuickLD-Editor innerhalb von SYPT nur die ersten 5 Zeichen des Arguments angezeigt.

Die Menge der Ein- und Ausgänge ist nur durch den verfügbaren Speicherplatz begrenzt, im Gegensatz zum UD70-Produkt, das auf 10 ganzzahlige Eingänge und 10 ganzzahlige Ausgänge begrenzt war.

**HINWEIS** Aufgrund der Tatsache, dass die SYPT Workbench für den UD70 entwickelt wurde, werden in den zum Erstellen von UDFBs verwendeten Tools weiterhin die Beschränkung der Eingangs- und Ausgangsargumente sowie die Reihenfolge der Gleitkomma-Variablen und ganzzahligen Variablen erzwungen. Daher kann die volle Flexibilität der Implementierung des SM-Applications-Moduls mit SYPT Workbench V1 nicht realisiert werden. Weitere Informationen zu den Beschränkungen finden Sie in der Online-Hilfe.

### 7.7.6 UDFB-Codeabschnitte

Der Code innerhalb eines UDFB ist in zwei Abschnitte aufgeteilt:

- Anfangsabschnitt
- Hauptabschnitt

Der Anfangsabschnitt wird verwendet, um lokale Variablen, die von dem UDFB verwendet werden sollen, zu deklarieren und zu initialisieren. Der Anfangsabschnitt wird beim Starten oder bei einem Reset für jede Instanz eines UDFB ausgeführt (vor dem DPL-Task „Initial“).

**HINWEIS** Die Eingangs- und Ausgangsargumente eines UDFB sind im Anfangsabschnitt nicht verfügbar.

Im Hauptabschnitt steht der eigentliche Code des Funktionsblocks, der für die Ausführung der Funktion benötigt wird. Eingangs- und Ausgangsargumente besitzen nur innerhalb des Hauptabschnitts einen Kontext.

Die zwei Abschnitte werden durch das Schlüsselwort `FBbody` getrennt. Der Anfangscode steht vor diesem Schlüsselwort, der Hauptcode dahinter.

Denken Sie daran, dass der eigentliche Code aus einer Mischung aus DPL, FBD-Diagrammen und QuickLD-Diagrammen bestehen kann.

Es folgt ein Beispiel für einen einfachen UDFB, mit dem zwei Zahlen addiert und auf einen vordefinierten Wert (0,5) skaliert werden:

```
(output%) = _simplefb(input1%, input2%) {
// Initialisierungscode:

scale% = 500 // Lokale variable initialisieren

FBbody
// Hauptteilcode:

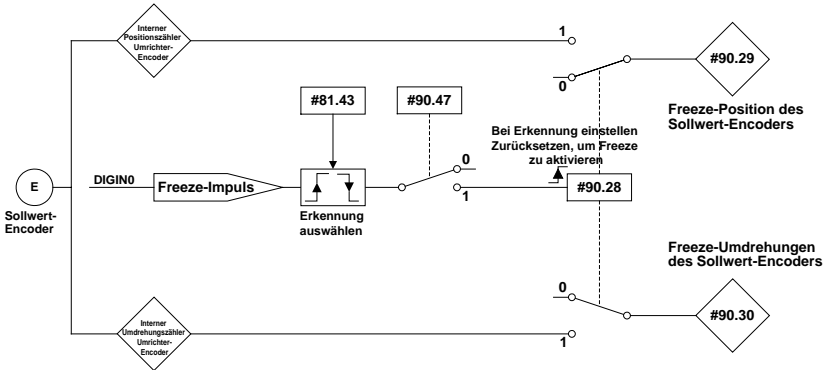
output% = input1% + input2% * scale% / 1000
}
```

# 8 SM-Applications-Freeze und -Nullimpuls

## 8.1 Freeze-Eingang

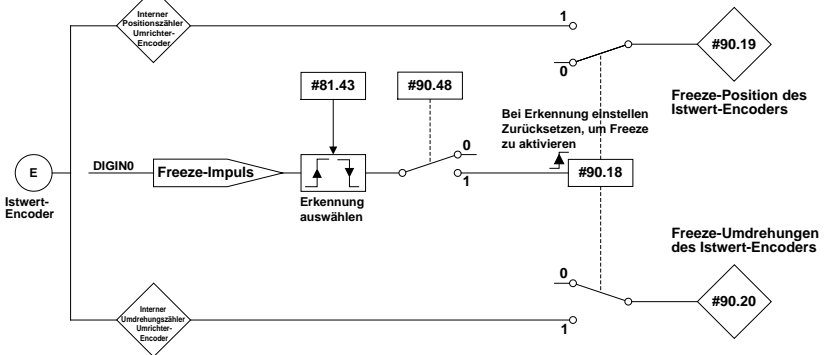
Der SM-Applications-Digitaleingang 0 (DIGIN0) kann zum „Einfrieren“ der Zähler von Sollwert- und Istwert-Encoder verwendet werden.

Abbildung 8-1 SM-Applications-Sollwert für den Freeze-Eingang



Der Encoder-Umdrehungszähler wird in Parameter #90.29 zwischengespeichert und die Encoder-Position in Parameter #90.30.

Abbildung 8-2 SM-Applications-Istwert für den Freeze-Eingang



Der Encoder-Umdrehungszähler wird in Parameter #90.19 zwischengespeichert und die Encoder-Position in Parameter #90.20.

Die Freeze-Position von Sollwert- und Istwert-Encoder kann an der aufsteigenden oder an der abfallenden Flanke des Freeze-Impulses erfasst werden. Dies wird bestimmt, indem Parameter #81.43 entweder auf 0 (aufsteigende Flanke) oder auf 1 (abfallende Flanke) gesetzt wird. Parameter #81.42 ermöglicht, dass die Position in die Parameter #90.19 und #90.29 und die Umdrehungszähler in die Parameter #90.20 und #90.30 geschrieben werden.

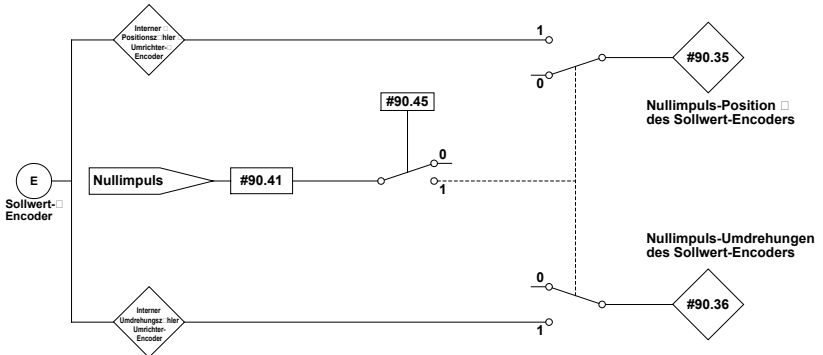
Wenn ein Freeze-Eingang erkannt wird, werden die Parameter #90.18 und #90.28 automatisch auf 1 gesetzt, so dass die Position in #90.19 und #90.29 geschrieben

werden kann und die Umdrehungszähler in #90.20 und #90.30 geschrieben werden können. Die Parameter #90.18 und #90.28 müssen auf Null zurückgesetzt werden, wenn der Benutzer die Daten beim nächsten Freeze-Impuls erneut aktualisieren möchte.

## 8.2 Nullimpuls

Im SM-Applications-Modul können Position und Umdrehungszähler an dem Punkt zwischengespeichert werden, an dem ein Nullimpuls am Sollwert- oder Istwert-Encoder erkannt wird.

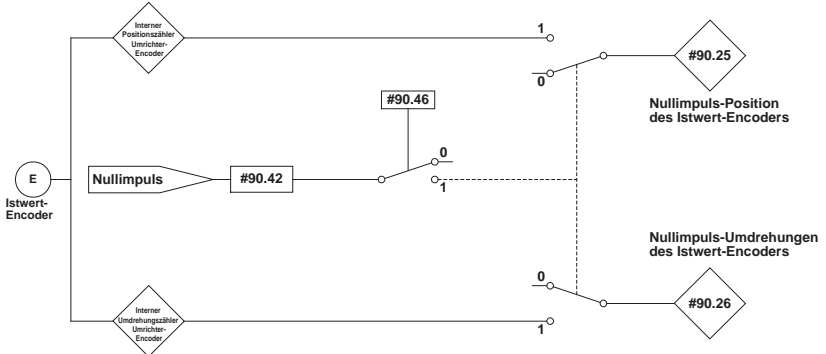
**Abbildung 8-3 SM-Applications-Sollwert-Nullimpuls**



Die Nullimpuls-Position wird in Parameter #90.35 zwischengespeichert und der Nullimpuls-Umdrehungszähler in Parameter #90.36.

Der Benutzer setzt #90.41 auf 0, und vom Umrücker wird #90.41 auf 1 gesetzt, und zwar JEDES Mal, wenn ein Nullimpuls ermittelt wird. Der Benutzer muss sicherstellen, dass die Nullimpulsdaten vor dem nächsten Nullimpuls verwendet werden.

**Abbildung 8-4 SM-Applications-Istwert-Nullimpuls**



Die Nullimpuls-Position wird in Parameter #90.25 zwischengespeichert und der Nullimpuls-Umdrehungszähler in Parameter #90.26.

Der Benutzer setzt #90.42 auf 0, und vom Umrücker wird #90.42 auf 1 gesetzt, und zwar JEDES Mal, wenn ein Nullimpuls erkannt wird. Der Benutzer muss sicherstellen, dass die Nullimpulsdaten vor dem nächsten Nullimpuls verwendet werden.

---

## 9 CTSync

---

### 9.1 Überblick

Das SM-Applications-Modul kann verwendet werden, um zwei oder mehr Unidrive SP-Umrichter zu synchronisieren. Dadurch wird sichergestellt, dass die internen Funktionen der Umrichter genau mit derselben Frequenz und Dauer ausgeführt werden, d. h. alle Maßnahmen werden gleichzeitig durchgeführt.

Außerdem können 3 Datenwerte von einem Modul (dem Master) an andere Module (Slaves) in demselben CTSync-Netzwerk weitergegeben werden. Dies umfasst 2 ganze Zahlen mit 32 Bit und Vorzeichen sowie 1 ganze Zahl mit 8 Bit ohne Vorzeichen.

Ein (und nur ein) SM-Applications-Modul muss als Master konfiguriert werden und alle anderen als Slaves, wenn sie an dem CTSync-Programm teilnehmen müssen. Im Master werden Sollwertdaten generiert, die an alle Slaves in dem Netzwerk gesendet werden. Der Master kann auf Slave-Betrieb eingestellt werden, wenn zum Beispiel zwei Umrichter synchronisiert werden müssen. In diesem Fall werden die Sollwertdaten vom Master generiert, und gleichzeitig richtet sich der Master nach diesen Sollwertdaten. Der Slave richtet sich ebenfalls nach diesen Sollwertdaten.

### 9.2 Anschlüsse

Das CTSync-System für SM-Applications funktioniert über eine Verbindung zwischen den RS485-Anschlüssen der SM-Applications-Module im Netzwerk (entweder 2-Draht oder 4-Draht). In Abschnitt 3.6 *EIA-RS485-Anschlüsse* finden Sie Informationen zum Anschließen der RS-485-Anschlüsse des SM-Applications-Moduls.

Um die Verdrahtung des vom Slave gesendeten und des vom Master empfangenen Signals zu vereinfachen, können im 4-Drahtmodus Leitungsverbindungen ausgelassen werden. Dies liegt daran, dass der Master keine Antwort vom Slave erhält.

### 9.3 Beschränkungen

- Im Netzwerk ist nur ein CTSync-Master zugelassen
- Alle POS-Tasks für CTSync-Master und -Slaves müssen auf dieselbe Aktualisierungszeit eingestellt sein (Parameter #81.12)
- Nach maximal 8 Knoten (2-Drahtmodus) bzw. maximal 16 Knoten (4-Drahtmodus) werden Leitungsverstärker benötigt
- Die maximale Kabellänge für ein RS485-Netzwerk beträgt 200 Meter

### 9.4 CTSync-Funktionsblöcke

Sechs Funktionsblöcke sind für die CTSync-Funktionalität des SM-Applications-Moduls verfügbar. Diese sind im Folgenden aufgeführt.



## 9.4.1 CTSYNCSaveMasterReferences

```
CTSYNCSaveMasterReferences(Reference1%, Reference2%, AuxiliaryReference%)
```

Eingangsargumente	
Argumentname	Bereich
Reference1	32 Bit mit Vorzeichen
Reference2	32 Bit mit Vorzeichen
AuxiliaryReference	8 Bit ohne Vorzeichen

Mit Hilfe dieses Funktionsblocks können Sollwertdaten vom CTSync-Master an alle CTSync-Slaves im Netzwerk gesendet werden. Dieser Befehl hat keinerlei Wirkung, wenn er auf einen CTSync-Slave angewendet wird.

## 9.4.2 CTSYNCGetSlaveReferences

```
(Reference1%, Reference2%, AuxiliaryReference%, Status%) =  
CTSYNCGetSlaveReferences()
```

Ausgangsargumente	
Argumentname	Bereich
Reference1%	32 Bit mit Vorzeichen
Reference2%	32 Bit mit Vorzeichen
AuxiliaryReference%	8 Bit ohne Vorzeichen

Mit Hilfe dieses Funktionsblocks können die vom CTSync-Master erstellten Sollwertdaten vom CTSync-System abgefragt werden. Er kann sowohl auf den Master als auch auf Slaves angewendet werden.

Aus dem Block werden die letzten empfangenen Master-Sollwerte zurückgegeben.

### Ausgangsargumente:

- Status%: 1: OK  
0: Null oder zu wenig Daten empfangen.  
-1: Mehr Bytes empfangen als erwartet.  
-2: Prüfsummenfehler in empfangenen Daten.

Wenn der Status nicht „OK“ lautet, werden die Werte für Reference1%, Reference2% und AuxiliaryReference% nicht geändert.

## 9.4.3 CTSYNCSaveOutputChannel

```
Result% = CTSYNCSaveOutputChannel(Channel%, Menu%, Parameter%)
```

Mit diesem Funktionsblock wird einer der 3 Kanäle so konfiguriert, dass dorthin gesendete Daten in einen angegebenen Umrichterparameter weitergeleitet werden.

### Eingangsargumente:

- Channel%: 1, 2 oder 3 für die 3 verfügbaren „Kanäle“.  
Menu%: Nummer des zu schreibenden Umrichtermenus.  
Parameter%: Nummer des zu schreibenden Umrichterparameters.

### Ausgangsargumente:

- Result%: 1: Vorgang erfolgreich abgeschlossen.  
-1: Ungültiger Kanal angegeben.  
-3: Kanalconfiguration durch einen anderen Task.  
-4: Parameter existiert nicht oder ist schreibgeschützt.

## 9.4.4 CTSYNCEnableOutputChannel

```
Result% = CTSYNCEnableOutputChannel(Channel%)
```

Mit Hilfe dieses Funktionsblocks können die Daten aus dem angegebenen Kanal vor Beginn jedes Motion Engine-Abtastzeitraums in den Unidrive SP geschrieben werden.

**Eingangsargumente:**

Channel%: 1, 2 oder 3 für die 3 verfügbaren „Kanäle“.

**Ausgangsargumente:**

Result%: 1: Vorgang erfolgreich abgeschlossen.  
0: Kanal nicht richtig konfiguriert.  
-1: Ungültiger Kanal angegeben.  
-3: Kanalkonfiguration durch einen anderen Task.

## 9.4.5 CTSYNCDisableOutputChannel

```
Result% = CTSYNCDisableOutputChannel(Channel%)
```

Mit Hilfe dieses Funktionsblocks wird das Schreiben der Daten aus dem angegebenen Kanal in den Unidrive SP beendet.

**Eingangsargumente:**

Channel%: 1, 2 oder 3 für die 3 verfügbaren „Kanäle“.

**Ausgangsargumente:**

Result%: 1: Vorgang erfolgreich abgeschlossen.  
-1: Ungültiger Kanal angegeben.  
-3: Kanalkonfiguration durch einen anderen Task.

## 9.4.6 CTSYNCSyncWriteOutputChannel

```
Result% = CTSYNCSyncWriteOutputChannel(Channel%, value%)
```

Mit Hilfe dieses Funktionsblocks wird ein Wert in den angegebenen Kanal geschrieben.

**Eingangsargumente:**

Channel%: 1, 2 oder 3 für die 3 verfügbaren „Kanäle“.  
Value%: Zu schreibender Wert.

**Ausgangsargumente:**

Result%: 1: Vorgang erfolgreich abgeschlossen.  
0: Kanal nicht richtig konfiguriert.  
-1: Ungültiger Kanal angegeben.  
-3: Kanalkonfiguration durch einen anderen Task.

Wenn der zu schreibende Wert den Bereich für den Parameter überschreitet, wird eine Fehlerabschaltung des Umrichters (tr44) ausgelöst, falls #81.14=1 und #81.17=1 ist, oder der Wert wird begrenzt, falls einer dieser Parameter auf Null gesetzt ist.

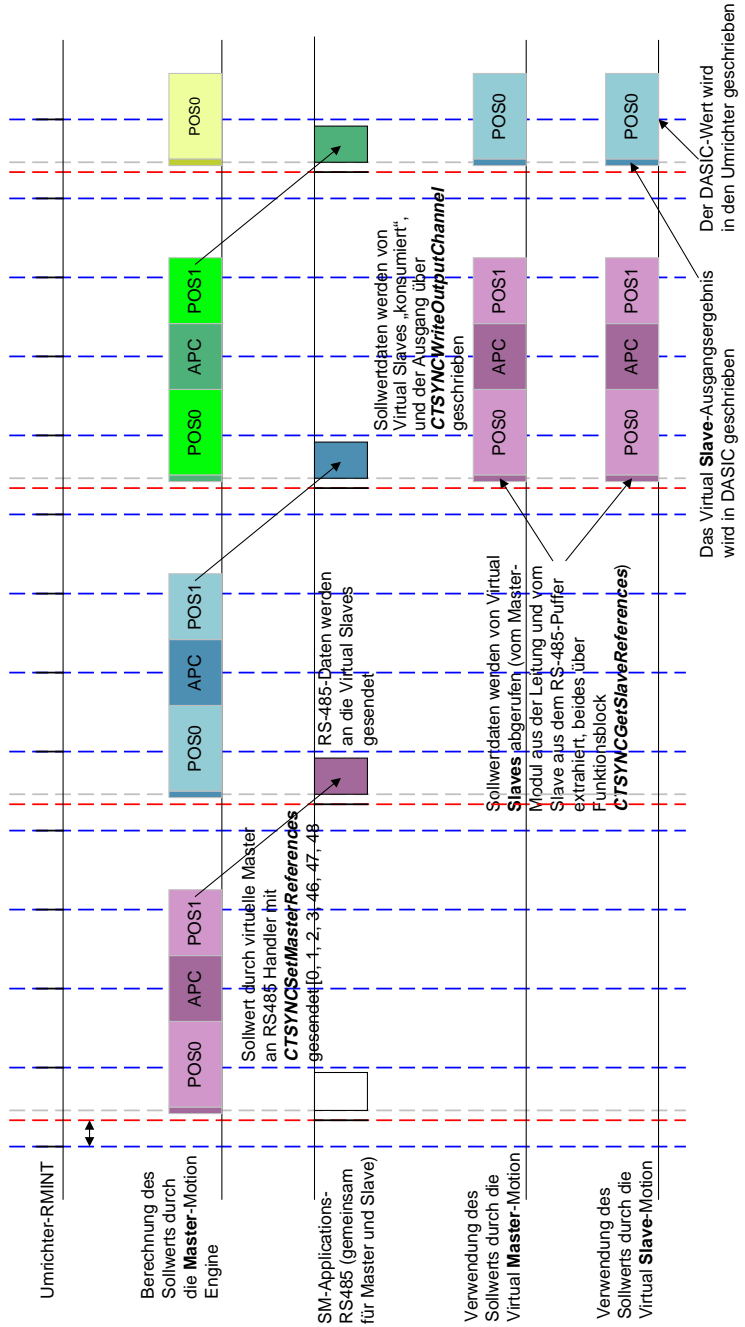
## 9.5 Motion Engine

Die Motion Engine-Tasks für jede Motion Engine-Abtastung sind im Folgenden dargestellt:

1. Der Sollwert wird von der Master-Motion Engine berechnet.
2. Die Sollwertdaten werden vom Master mit Hilfe des Funktionsblocks **CTSYNCSetMasterReferences** an das RS-485-Steuerungsprogramm weitergeleitet.
3. Die RS-485-Daten werden an die Slaves gesendet.
4. Die Sollwertdaten werden von den Slaves mit Hilfe des Funktionsblocks **CTSYNCGetSlaveReferences** empfangen.
5. Die Sollwertdaten werden (falls erforderlich) mit Hilfe des Funktionsblocks **CTSYNCWriteOutputChannel** an die angegebenen Kanäle ausgegeben.
6. Die Slave-Ausgangswerte werden über den ASIC des Umrichters in die Umrichterparameter geschrieben.

Weitere Informationen zu den zeitlichen Abläufen finden Sie in Abbildung 9-1 *Motion Engine-Zeitsteuerung*.

# Abbildung 9-1 Motion Engine-Zeitsteuerung



## 9.6 Beispiel für einen Virtual Master

### 9.6.1 Beispiel-Mastercode

Der folgende Code zeigt, wie der Master konfiguriert werden müsste, damit Sollwertdaten generiert und an die Slaves gesendet werden. Außerdem zeigt er, dass der Master als Slave fungiert, indem er sich nach den von ihm selbst generierten Sollwertdaten richtet. Dies geschieht mit Hilfe des Funktionsblocks **CTSYNCGetSlaveReferences**.

Im Beispiel werden Virtual Master-Daten als Dreiecksrampe generiert, und außerdem wird ein Virtual Slave implementiert, über den die Daten an den Umrichter ausgegeben werden.

```
Initial
{
    #81.06=25 //CTSync-Master
    #81.12=2 //Pos-Task bei 500 us
    REINIT

    //Den Rampensollwert für den virtual Master konfigurieren.
    Ramp% = 0

    //Den Slave-Ausgangskanal 1 auf Menü 18, Parameter 11 konfigurieren.
    CTSYNCSetupOutputChannel(1,18,11)

    //Den konfigurierten Ausgangskanal freigeben.
    CTSYNCEnableOutputChannel(1)
}

POS0
{
    //Zuerst die Slave-Funktion ausführen, damit der zeitliche Ablauf bei Master und
    Slave identisch ist:
    (Slaveref1%,Slaveref2%,Auxref%, Status%) = CTSYNCGetSlaveReferences()

    //Status% sollte hier kontrolliert werden, um sicherzustellen, dass die Daten
    korrekt empfangen wurden.
    if Status% = USR_VALUE_OK then

        //Ausgang zum Umrichter bei Beginn des nächsten Bewegungszeitraums.
        CTSYNCWriteOutputChannel(1,Slaveref1%)

    endif

    //Die Master-Funktion ausführen, in diesem Fall eine Rampe:
    Ramp% = Ramp% + 1

    //Gegebenenfalls das Rampenprofil zurücksetzen
    if Ramp% > 999 then

        //Auf Null zurücksetzen
        Ramp% = 0
    endif

    //Die Sollwertdaten an die Slaves weitergeben. Beachten Sie, dass nur der wert
    reference1 verwendet wird.
    CTSYNCSetMasterReferences(Ramp%,0,0)
}
```

## 9.6.2 Beispiel-Slavecode

Der folgende Code zeigt, wie die vom Virtual Master generierten Sollwertdaten von den Slave(s) abgefragt würden.

```
Initial
{
    #81.06=26 //CTSync-Slave
    #81.12=2 //Pos-Task bei 500 us
    REINIT

    //Den Rampensollwert für den Virtual Master konfigurieren.
    Ramp% = 0

    //Den Slave-Ausgangskanal 1 auf Menü 18, Parameter 11 konfigurieren.
    CTSYNCSetupOutputChannel(1,18,11)

    //Den konfigurierten Ausgangskanal freigeben.
    CTSYNCEnableOutputChannel(1)
}

POS0
{
    //zuerst die Slave-Funktion ausführen, damit der zeitliche Ablauf bei Master und
    Slave identisch ist:
    (Slaveref1%,Slaveref2%,Auxref%, Status%) = CTSYNCGetSlaveReferences(

    //Kontrollieren, ob die Daten ordnungsgemäß empfangen wurden.
    if Status% = USR_VALUE_OK then

        //Ausgang zum Umrichter bei Beginn des nächsten Bewegungszeitraums.
        CTSYNCwriteOutputChannel(1,Slaveref1%)

    endif
}
```

# 10 Synchronisation zwischen den Optionsmodulen

## 10.1 Überblick

Das Synchronisationsprogramm zwischen den Optionsmodulen stellt einen Mechanismus dar, mit dessen Hilfe Lageregelungs-Tasks an einem oder mehreren Modulen mit Lageregelungs-Tasks an einem anderen Modul an demselben Unidrive SP synchronisiert werden können, ohne dass eine externe Verdrahtung benötigt wird.

Die Synchronisation wird dadurch erreicht, dass ein Auslösersignal von einem Modul „produziert“ und von einem oder mehreren Modulen „konsumiert“ wird. Diese Module werden als „Produzenten“ und „Konsumenten“ bezeichnet. Das Auslösersignal wird verwendet, um die Zeitkoordinierung der POS-Engine zu ändern.

Wenn das Modul Produzent ist, kann der Auslöser mit einer Rate von 250  $\mu$ s, 500  $\mu$ s, 1 ms, 2 ms, 4 ms oder 8 ms geliefert werden, d. h. der Auslöser wird von Produzentenmodulen in jedem POS-Zeitraum generiert, wie in Parameter #81.12 angegeben.

Dieses Programm besteht zusätzlich zum CTSync-Programm (weitere Informationen finden Sie in Abschnitt 9 *CTSync*), obwohl es in Verbindung mit CTSync verwendet werden kann, um Module in mehreren Unidrive SP zu synchronisieren (siehe Abschnitt 10.3 *Beispiel 2 zur Synchronisation zwischen Optionen*).

Wenn das SM-Applications-Modul Konsument ist, können die dazugehörigen Lageregelungs-Tasks mit derselben Rate oder schneller als das Synchronisationssignal (der Auslöser) vom Produzenten ausgeführt werden.

Wenn ein SM-Applications-Modul Konsument ist und der dazugehörige Lageregelungs-Task so eingestellt ist, dass er langsamer ausgeführt wird als mit der Rate des Synchronisationssignals (Auslösers) vom Produzenten, kann die Phase des Produzenten unmöglich erkannt werden, so dass eine Synchronisation mit dem Signal des Produzenten nicht versucht wird (siehe Abschnitt 10.3 *Beispiel 2 zur Synchronisation zwischen Optionen*).

## 10.2 Beispiel 1 zur Synchronisation zwischen Optionen

Dieses Beispiel zeigt ein Szenario, in dem die POS0-Tasks der Module in einem Unidrive SP synchronisiert werden müssen.

Die folgende Tabelle enthält die Einstellungen für jedes der 3 Module:

**Tabelle 10.1 Parametereinstellungen**

	Parameter	Wert	Beschreibung
Steckplatz 1	#81.06	1 (Standard)	Nicht im CTSync-Programm
	#81.12	2	500 $\mu$ s Zeit für Lageregelungs-Task
	#91.21	2	Synchronisation zwischen den Optionen: Konsument
	#91.22	6	<i>Synchronisation zwischen den Optionen: Konsumentenstatus erreicht</i>
Steckplatz 2	#81.06	1 (Standard)	Nicht im CTSync-Programm
	#81.12	1	250 $\mu$ s Zeit für Lageregelungs-Task
	#91.21	2	Synchronisation zwischen den Optionen: Konsument
	#91.22	6	<i>Synchronisation zwischen den Optionen: Konsumentenstatus erreicht</i>
Steckplatz 3	#81.06	1 (Standard)	Nicht im CTSync-Programm
	#81.12	3	1 ms Zeit für Lageregelungs-Task
	#91.21	1	Synchronisation zwischen den Optionen: Produzent
	#91.22	5	<i>Synchronisation zwischen den Optionen: Produzentenstatus erreicht</i>

Der kursiv gesetzte Parameter #91.22 ist ein Statusparameter und muss nicht eingestellt werden. Er zeigt, ob das Modul in der Synchronisation zwischen den Optionen die in Parameter #91.21 angegebene Funktionalität erreicht hat. Weitere Informationen finden Sie in Parameter #91.22 .

Vor dem Einstellen der oben genannten Parameter werden die POS-Tasks möglicherweise nicht synchron ausgeführt, wie in Abbildung 10.1 *Überblick* dargestellt. Nach dem Einstellen der Parameter auf die oben genannten Werte werden die POS-Tasks synchron ausgeführt, wie in Abbildung 10.2 *Beispiel 1 zur Synchronisation zwischen Optionen* dargestellt.

**Abbildung 10-1 Ausführung der POS-Tasks VOR der Synchronisation zwischen den Optionen**

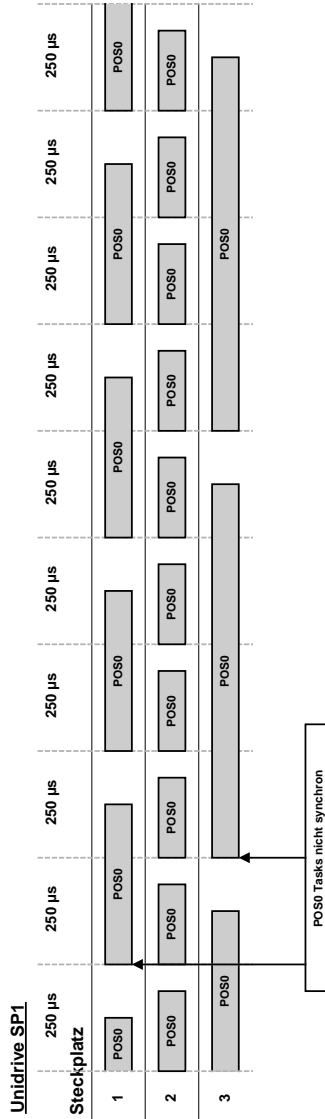
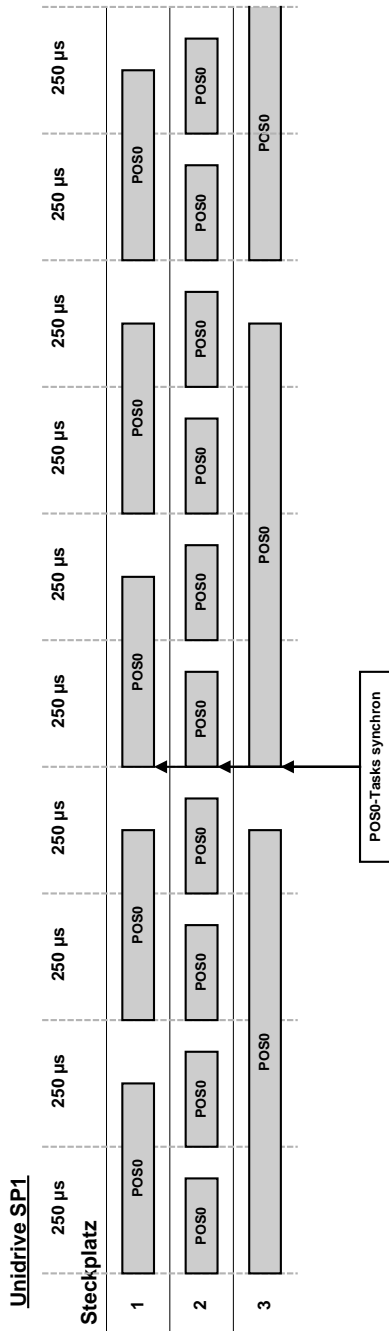


Abbildung 10-2 Ausführung der POS-Tasks NACH der Synchronisation zwischen den Optionen





## 10.3 Beispiel 2 zur Synchronisation zwischen Optionen

Dieses Beispiel zeigt ein Szenario, in dem die POS0-Tasks der Module in 2 Unidrive SP synchronisiert werden müssen.

Damit dies funktioniert, muss ein Modul an jedem Umrichter für die Teilnahme am CTSync-Programm konfiguriert werden. (Weitere Informationen finden Sie in Abschnitt 9 *CTSync* ). Ein Modul muss als CTSync-Master konfiguriert werden und ein Modul als CTSync-Slave. Die folgenden Tabellen enthalten die Einstellungen für jedes der 6 Module:

**Tabelle 10.2 Parametereinstellungen für Unidrive SP 1**

	Parameter	Wert	Beschreibung
Steckplatz 1	#81.06	1 (Standard)	Nicht im CTSync-Programm
	#81.12	2	500 $\mu$ s Zeit für Lageregelungs-Task
	#91.21	2	Synchronisation zwischen den Optionen: Konsument
	#91.22	6	<i>Synchronisation zwischen den Optionen: Konsumentenstatus erreicht</i>
Steckplatz 2	#81.06	1 (Standard)	Nicht im CTSync-Programm
	#81.12	1	250 $\mu$ s Zeit für Lageregelungs-Task
	#91.21	2	Synchronisation zwischen den Optionen: Konsument
	#91.22	6	<i>Synchronisation zwischen den Optionen: Konsumentenstatus erreicht</i>
Steckplatz 3	#81.06	25 (Standard)	CTSync-Master
	#81.12	3	1 ms Zeit für Lageregelungs-Task
	#91.21	1	Synchronisation zwischen den Optionen: Produzent
	#91.22	5	<i>Synchronisation zwischen den Optionen: Produzentenstatus erreicht</i>

**Tabelle 10.3 Parametereinstellungen für Unidrive SP 2**

	Parameter	Wert	Beschreibung
Steckplatz 1	#81.06	1 (Standard)	Nicht im CTSync-Programm
	#81.12	2	500 $\mu$ s Zeit für Lageregelungs-Task
	#91.21	2	Synchronisation zwischen den Optionen: Konsument
	#91.22	6	<i>Synchronisation zwischen den Optionen: Konsumentenstatus erreicht</i>
Steckplatz 2	#81.06	26	CTSync-Slave
	#81.12	3	1 ms Zeit für Lageregelungs-Task
	#91.21	1	Synchronisation zwischen den Optionen: Produzent
	#91.22	13	<i>Synchronisation zwischen den Optionen: Produzentenstatus mit DINT-Signal erreicht</i>
Steckplatz 3	#81.06	1 (Standard)	Nicht im CTSync-Programm
	#81.12	4	2 ms Zeit für Lageregelungs-Task
	#91.21	2	Synchronisation zwischen den Optionen: Konsument
	#91.22	2	<i>Synchronisation zwischen den Optionen: Konsumentenstatus nicht erreicht (POS-Task zu langsam)</i>

Der kursiv gesetzte Parameter #91.22 ist ein Statusparameter und muss nicht eingestellt werden. Er zeigt, ob das Modul in der Synchronisation zwischen den Optionen die in Parameter #91.21 angegebene Funktionalität erreicht hat. Weitere Informationen finden Sie in Parameter #91.22 .

**HINWEIS** Wenn ein Modul CTSync-Slave ist, muss es der Produzent für die Synchronisation zwischen den Optionen an dem Umrichter sein, an dem es angebracht ist. Daher wird Parameter #91.21 automatisch auf den Wert 1 eingestellt.

Vor dem Einstellen der oben genannten Parameter werden die POS-Tasks möglicherweise nicht synchron ausgeführt, wie in Abbildung 10.3 *Beispiel 2 zur Synchronisation zwischen Optionen* dargestellt. Nach dem Einstellen der Parameter auf die oben genannten Werte sollten die POS-Tasks synchron ausgeführt werden, wie in Abbildung 10-4 *Ausführung der POS-Tasks NACH der Synchronisation zwischen den Optionen mit CTSync* dargestellt.

**Abbildung 10-3 Ausführung der POS-Tasks VOR der Synchronisation zwischen den Optionen mit CTSync**

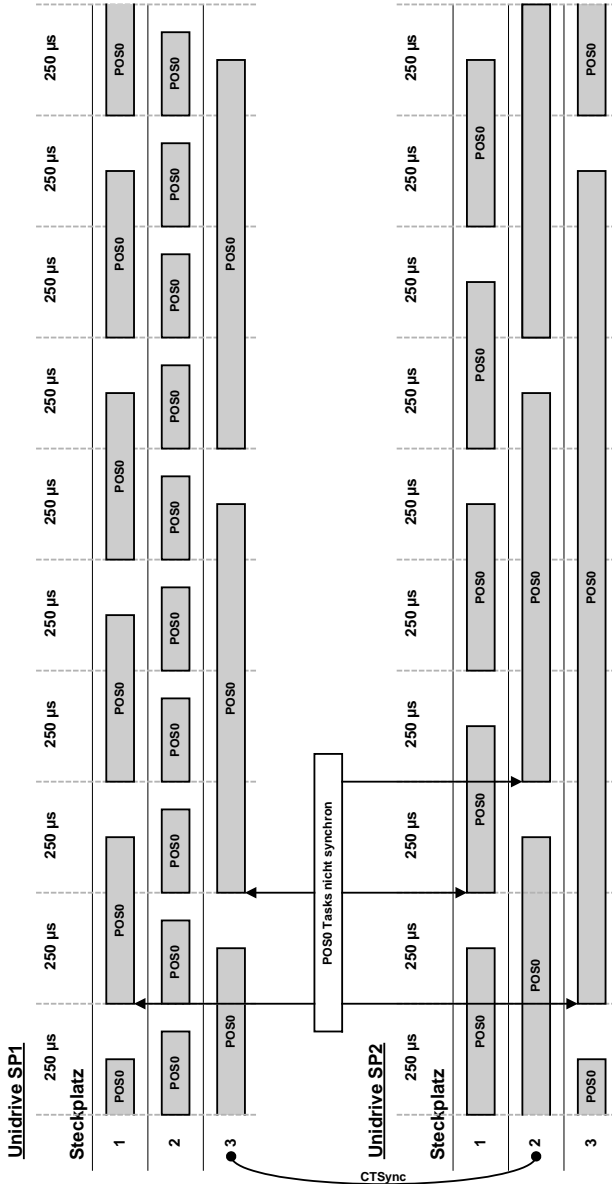
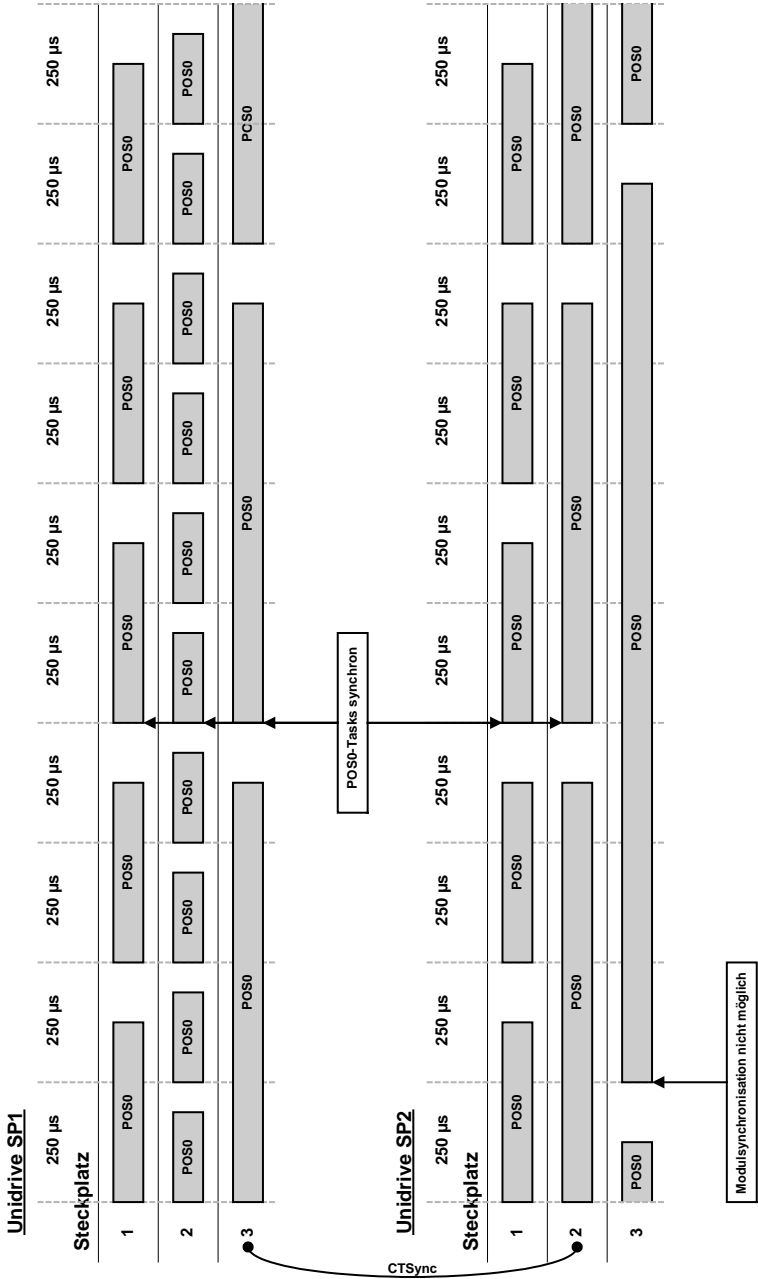


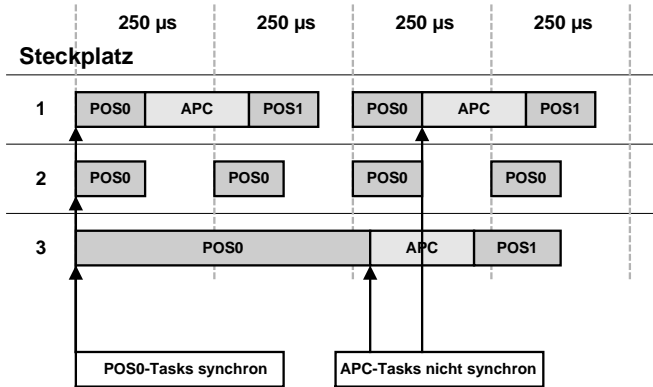
Abbildung 10-4 Ausführung der POS-Tasks NACH der Synchronisation zwischen den Optionen mit CTSync



## 10.4 Lageregelungs-Tasks

Die Lageregelungs-Tasks bestehen aus POS0, APC und POS1. Die Tasks werden in dieser Reihenfolge ausgeführt (zuerst POS0, gefolgt von APC und anschließend POS1). Da die Ausführungsdauer der Tasks POS0 und APC auf verschiedenen Modulen unterschiedlich sein kann, können die Tasks APC und POS1 nicht synchronisiert werden. Daher muss dem Benutzer beim Arbeiten mit der Synchronisation zwischen Optionen bewusst sein, dass die Lageregelungs-Tasks insgesamt für jedes Modul synchron gestartet werden, nicht die einzelnen Tasks. Abbildung 10-5 *Lageregelungs-Tasks* zeigt ein Beispiel dafür.

Abbildung 10-5 Lageregelungs-Tasks



# 11 Fehlersuche

In diesem Kapitel wird Folgendes genauer beschrieben:

- Laufzeitfehler und Fehlerabschaltungs-codes
- Bearbeiten von Laufzeitfehlern
- CTNet-Netzwerkstatus
- Support

## 11.1 Laufzeitfehler

Ein Laufzeitfehler ist ein Fehler, der in einem bestimmten Vorgang des SM-Applications-Moduls auftritt. Er kann infolge eines Fehlers bei der Ausführung des benutzerdefinierten DPL-Programms (z. B. Versuch, in einen nicht existierenden Parameter zu schreiben oder einen Wert durch Null zu dividieren), einer Fehlkonfiguration (zum Beispiel falsche CTNet-Konfiguration) oder eines Systemfehlers auftreten, zum Beispiel Prozessorüberlastung oder *Watchdog*-Zeitbereichsüberschreitung.

Eine oder mehrere der folgenden Maßnahmen können ergriffen werden, wenn ein Fehler auftritt:

- Das Benutzerprogramm kann angehalten oder gar nicht erst gestartet werden
- Für den Umrichter kann eine Fehlerabschaltung auf **SLx.Er** ausgelöst werden (x steht für die Steckplatznummer), wobei der Laufzeitfehlercode in #MM.50 eingefügt wird
- Für den Umrichter kann eine Fehlerabschaltung auf einem anderen **SLx.\*\*\***-Code ausgelöst werden
- Der DPL-Task **ERROR** kann ausgeführt werden (falls vorhanden)

Welche Maßnahme ergriffen wird, ist abhängig vom Fehlertyp und von der Einstellung von Parameter #MM.14 (Globale Laufzeit-Fehlerabschaltung freigeben). Dies wird in Abschnitt 11.3 unten genauer beschrieben.

## 11.2 Unidrive SP-Displaycodes bei der Fehlerabschaltung

In der Tabelle unten sind die möglichen Fehlerabschaltungs-codes aufgeführt, die auf dem Unidrive SP angezeigt werden, wenn ein Fehler im SM-Applications-Modul erkannt wurde, durch den eine Umrichter-Fehlerabschaltung ausgelöst wird. Denken Sie daran, dass nicht durch alle Laufzeitfehler eine Umrichter-Fehlerabschaltung ausgelöst wird.

**Tabelle 11.1 Unidrive SP-Displaycodes bei der Fehlerabschaltung**

Steckplatz, an dem die Fehlerabschaltung ausgelöst wurde			Fehler	Beschreibung
Steckplatz 1	Steckplatz 2	Steckplatz 3		
SL1.HF	SL2.HF	SL3.HF	Hardware-Fehler	Ein Optionsmodul wurde vom Unidrive SP erkannt, jedoch ist mit diesem keine Kommunikation möglich.
SL1.tO	SL2.tO	SL3.tO	Watchdog-Zeitbereichsüberschreitung	Bedeutet, dass in einem Benutzerprogramm, in dem die <i>Watchdog</i> -Funktion verwendet wurde, der WDOG-Befehl nicht innerhalb von 200 ms ausgegeben wurde.

**Tabelle 11.1 Unidrive SP-Displaycodes bei der Fehlerabschaltung**

Steckplatz, an dem die Fehlerabschaltung ausgelöst wurde			Fehler	Beschreibung
Steckplatz 1	Steckplatz 2	Steckplatz 3		
SL1.Er	SL2.Er	SL3.Er	Fehler	Laufzeit-Fehlerabschaltung, die vom SM-Applications-Modul entweder aufgrund eines DPL-Programmfehlers oder eines anderen Ereignisses generiert wird. Der tatsächliche Fehlercode wird in #MM.50 platziert.
SL1.nF	SL2.nF	SL3.nF	Nicht angebracht	Das SM-Applications-Modul wurde während des Betriebs getrennt, oder das SM-Applications-Modul ist abgestürzt. Diese Fehlerabschaltung tritt auch dann auf, wenn ein Unidrive SP-Steckplatz für ein SM-Applications-Modul konfiguriert ist, das Modul jedoch nicht im Steckplatz angebracht ist.
SL1.dF	SL2.dF	SL3.dF	Anderes angebracht	Diese Fehlerabschaltung tritt dann auf, wenn ein SM-Applications-Modul an einem Steckplatz angebracht ist, der zuvor durch ein anderes Solutions-Modul belegt war, oder an einem zuvor nicht verwendeten Steckplatz.

### 11.3 SM-Applications-Laufzeitfehlercodes

Wenn vom SM-Applications-Modul während des Betriebs ein Fehler erkannt wird, wird der Fehlercode im folgenden Parameter platziert:

#81.50	SM-Applications-Fehlercode		
Zugang	NL	Bereich	0 bis 255
Standard	n.b.	Aktualisierungsrate	Bei Fehler

Bei bestimmten Fehlern kann der Benutzer auswählen, ob für den Unidrive SP ebenfalls eine Fehlerabschaltung ausgelöst werden sollte. Dies wird mit dem Parameter „Globale Laufzeit-Fehlerabschaltung freigeben“ konfiguriert:

#81.14	Globale Laufzeit-Fehlerabschaltung freigeben		
Zugang	LS	Bereich	0/1
Standard	0	Aktualisierungsrate	n.b.

Wenn er auf 1 (Ein) eingestellt ist, wird bei allen Laufzeitfehlern eine Fehlerabschaltung des Umrichters ausgelöst.

In der Tabelle unten sind die Fehlercodes und ihre Bedeutung aufgeführt. Außerdem ist angegeben, ob eine Fehlerabschaltung des Umrichter ausgelöst oder das Benutzerprogramm unterbrochen wird und ob der DPL-Task ERROR ausgeführt wird.

### Hinweise:

- Die Angabe „**Vielleicht**“ unter „Fehlerabschaltung des Umrichters?“ bedeutet, dass nur dann eine Fehlerabschaltung des Umrichters ausgelöst wird, wenn der Parameter „Globale Laufzeit-Fehlerabschaltung freigeben“ eingestellt wurde.
- Die Angabe „**Läuft nicht**“ unter „Programm angehalten?“ bedeutet, dass der Fehler bei der Initialisierung auftritt und das Programm nicht gestartet wird.

**Tabelle 11.2 SM-Applications-Fehlercodes**

Fehlercode	Grund	Fehlerabschaltung des Umrichters?	ERROR-Task?	Programm angehalten?
39	Stack-Überlauf im Benutzerprogramm	Ja	Nein	Ja
40	Unbekannter Fehler - bitte Lieferanten kontaktieren	Ja	Nein	Ja
41	Parameter existiert nicht. Der Benutzer hat versucht, im DPL-Programm einen nicht existierenden Parameter abzufragen oder zu schreiben.	Vielleicht	Ja	Ja
42	Versuch, in einem Nur-Lese-Parameter zu schreiben.	Vielleicht	Ja	Ja
43	Versuch, einen Nur-Schreib-Parameter abzufragen.	Vielleicht	Ja	Ja
44	Parameterwert außerhalb des gültigen Bereiches. (Der Benutzer hat einen unzulässigen Wert in einen Parameter innerhalb eines DPL-Programms geschrieben.) Wenn Parameter #MM.17=0 ist, wird der geschriebene Wert automatisch begrenzt, und es tritt kein Fehler auf.	Vielleicht	Ja	Ja
45	Ungültige Synchronisationsmodi	Ja	Nein	Läuft nicht
46	Nicht verwendet	n.b.	n.b.	n.b.
47	Synchronisation mit CTSync-Master verloren	Ja	Ja	Ja
48	RS485 nicht in Benutzermodus. Tritt auf, wenn der Benutzer versucht, einen DPL-Befehl für den Benutzermodus von RS485 zu verwenden, der RS485-Anschluss sich jedoch nicht im Benutzermodus befindet.	Ja	Ja	Ja
49	Ungültige RS485-Konfiguration. Beispiel: ungültiger Modus.	Ja	Ja	Ja
50	Mathematischer Fehler - Division durch Null oder Überlauf.	Vielleicht	Ja	Ja
51	Array-Index außerhalb des gültigen Bereiches. Beispiel: arr%(20), wenn arr% nur für 19 Elemente dimensioniert wurde.	Vielleicht	Ja	Ja
52	Benutzer-Fehlerabschaltung durch Steuerwort. Ausgelöst durch Einstellen des Fehlerabschaltungsbits in Steuerwort #90.11.	Ja	Nein	Nein
53	DPL-Programm nicht kompatibel mit Ziel. Beispiel: Herunterladen eines für UD70 kompilierten Programms.	Ja	n.b.	n.b.
54	DPL-Task-Überlauf. Tritt auf, wenn der DPL-Code innerhalb eines Echtzeit-Tasks (z. B. POS0) nicht rechtzeitig abgeschlossen werden kann. Mit Hilfe von Parameter #88.02 kann der Task ermittelt werden, in dem dieser Fehler auftritt. Überprüfen Sie, ob die Zeitkoordinierungsrate korrekt ist und im Task keine Schleifen vorkommen. Dieser Fehler kann auch eine Folge von externen Einflüssen sein, zum Beispiel eines großen Datenstoßes, der über CTNet eingeht. Dieses Problem kann behoben werden, indem die Priorität des CTNet-Tasks niedriger eingestellt wird als die der POS-Tasks. Dies kann jedoch dazu führen, dass der CTNet-Task nicht mehr versorgt wird. Weitere Informationen finden Sie unter Parameter <b>#81.44</b> auf Seite 32.	Vielleicht	Ja	Ja
55	Nicht verwendet	n.b.	n.b.	n.b.
56	Ungültige Konfiguration der Zeitgebereinheit	Ja	Ja	Ja
57	Funktionsblock existiert nicht.	Ja	Ja	Läuft nicht

**Tabelle 11.2 SM-Applications-Fehlercodes (Fortsetzung)**

Fehlercode	Grund	Fehlerabschaltung des Umrichters?	ERROR-Task?	Programm angehalten?
58	Flash-SPS-Speicherung fehlerhaft. Tritt beim Starten auf und bedeutet, dass SPS-Registersatz (P/Q/T/U) und Menü 20 nicht wiederhergestellt wurden. Wenn dieses Problem weiterhin besteht, könnte dies auf einen Hardware-Fehler hinweisen. Kontaktieren Sie daher Ihren Lieferanten.	Ja	Ja	Läuft nicht
59	Applikationsmodul von Umrichter als Synchronisations-Master abgelehnt.	Ja	Ja	Ja
60	CTNet-Hardware-Fehler. Bitte kontaktieren Sie Ihren Lieferanten.	Vielleicht	Nein	Nein
61	Ungültige CTNet-Konfiguration. Überprüfen Sie alle Konfigurationsparameter.	Vielleicht	Nein	Nein
62	Ungültige CTNet-Baudrate. Überprüfen Sie Parameter #MM.24 und die Netzwerkverbindungen.	Vielleicht	Nein	Nein
63	Ungültige CTNet-Knotenkenung. Überprüfen Sie Parameter #MM.23.	Vielleicht	Nein	Nein
64	Überlastung der Digitalausgänge. Beide Digitalausgänge werden inaktiv, wenn dies auftritt, und bleiben inaktiv, bis der Fehlerzustand zurückgesetzt wurde. Die Schwelle für eine Fehlerabschaltung liegt bei 20 mA.	Ja	Ja	Ja
65	Ungültige(r) Parameter in Funktionsblock. Sie haben innerhalb eines DPL-Programms einen FB aufgerufen, jedoch ist mindestens eine Eingabe ungültig.	Ja	Ja	Ja
66	Heap-Speicher des Benutzers zu groß. Das Programm wurde für ein Ziel kompiliert, das über mehr RAM-Speicher verfügt als dieses. Tritt beim Starten auf.	Ja	Nein	Läuft nicht
67	RAM-Datei existiert nicht, oder es wurde eine Datei angegeben, die keine RAM-Datei ist.	Ja	Ja	Ja
68	Die angegebene RAM-Datei ist nicht mit einem Array verknüpft.	Ja	Ja	Ja
69	Datenbank-Cache für Umrichterparameter konnte im Flash-Speicher nicht aktualisiert werden.	Ja	Nein	Läuft nicht
70	Benutzerprogramm bei freigegebenem Umrichter heruntergeladen. Tritt auf, wenn #MM.37 = 1 ist und ein Programm heruntergeladen wird.	Vielleicht	Nein	Ja
71	Umrichtermodus konnte nicht geändert werden	Ja	Nein	Ja
72	Ungültige CTNet-Pufferoperation	Ja	Ja	Ja
73	Fehler bei der Schnellinitialisierung der Parameter	Ja	Nein	Nein
74	Übertemperatur	Ja	Ja	Ja
75	Hardware nicht verfügbar. Im Benutzerprogramm wurde versucht, auf nicht verfügbare Hardware zuzugreifen. Beispiel: Zugriff auf Digital-E/A, RS485-Anschluss oder CTNet im SM-Applications Lite-Modul.	Ja	Ja	Ja
76	Modultyp kann nicht aufgelöst werden. Das Modul wird nicht erkannt.	Ja	Nein	Läuft nicht
77	Fehler bei der Kommunikation mit dem Optionsmodul in Steckplatz 1.	Ja	Ja	Ja
78	Fehler bei der Kommunikation mit dem Optionsmodul in Steckplatz 2.	Ja	Ja	Ja
79	Fehler bei der Kommunikation mit dem Optionsmodul in Steckplatz 3.	Ja	Ja	Ja



**Tabelle 11.2 SM-Applications-Fehlercodes (Fortsetzung)**

Fehlercode	Grund	Fehlerabschaltung des Umrichters?	ERROR-Task?	Programm angehalten?
80	Fehler bei der Kommunikation mit einem Optionsmodul (unbekannter Steckplatz).	Ja	Ja	Ja
81	APR-interner Fehler. Siehe Parameter #81.38 (Beispiel: Verwendung einer nicht initialisierten CAM-Funktion).	Vielleicht	Ja	Ja
82	Kommunikation zum Umrichter fehlerhaft.	Vielleicht	Ja	Ja

## 11.4 Bearbeiten von Laufzeitfehlern mit dem ERROR-Task

Bestimmte Laufzeitfehler führen dazu, dass der DPL ERROR-Task aufgerufen wird, wenn er vorhanden ist. Dies stellt eine bequeme Methode dar, um den Fehlerzustand sicher zu handhaben und jede notwendige Maßnahme zu ergreifen, z. B. ein kontrolliertes Anhalten des Systems oder die Signalisierung eines Alarms.

Wenn ein ERROR-Task ausgeführt wird, wurden alle anderen DPL-Tasks angehalten. Daher bestehen für den ERROR-Task ausschließliche Ausführungsrechte. Nach Abschluss des ERROR-Tasks wird das DPL-Programm beendet, und es werden keine anderen DPL-Tasks ausgeführt. (Es ist jedoch möglich, das Programm zurückzusetzen und neuzustarten - weitere Details dazu sind im Folgenden aufgeführt.)

**HINWEIS** Durch Umrichter-Fehlerabschaltungen wird keine Ausführung des ERROR-Tasks ausgelöst. Dies geschieht nur bei bestimmten DPL-Programmfehlern.

Innerhalb des ERROR-Tasks können alle standardmäßigen DPL-Befehle sowie die meisten Funktionsblöcke verwendet werden. Auf alle Umrichter- und SM-Applications-Parameter kann zugegriffen werden.

Der Laufzeitfehlercode kann mit Hilfe des folgenden Parameters bestimmt werden:

#88.01	Fehlerstatus/Reset		
<b>Zugang</b>	LS	<b>Bereich</b>	0 bis 9999
<b>Standard</b>	n.b.	<b>Aktualisierungsrate</b>	Bei Fehler

Dieser Parameter dient zu zwei Zwecken - wenn er abgefragt wird, wird derselbe Laufzeitfehlercode wie bei #81.50 zurückgegeben. (Beachten Sie, dass keine Umrichter-Fehlerabschaltungscodes zurückgegeben werden.) Der Parameter wird auf Null zurückgesetzt, wenn ein Reset durchgeführt wird und wenn die Ausführung des Benutzerprogramms gestartet wird.

Wenn in den Parameter ein Wert von 1070 geschrieben wurde, wird vom SM-Applications-Modul ein Warm-Neustart des Umrichters und aller anderen Optionen ausgelöst. Damit können das Benutzerprogramm neugestartet (Bereitstellung des automatischen Programmstarts #81.13=1) und jede Umrichter-Fehlerabschaltung zurückgesetzt werden. Diese Reset-Maßnahme kann jederzeit durchgeführt werden, nicht nur nach einem Laufzeitfehler oder bei einem ERROR-Task.



Wenn der Wert 1070 in Parameter #88.01 geschrieben wird, führt dies dazu, dass eine beliebige Umrichter-Fehlerabschaltung automatisch zurückgesetzt wird und alle Optionen im Unidrive SP zurückgesetzt werden. Dieses Verhalten ist anders als bei dem UD70-Produkt am Unidrive 1, bei dem der Umrichter nicht zurückgesetzt wurde.

Der Task, durch den ein Laufzeitfehler verursacht wurde, kann durch das Abfragen von Parameter #88.02 bestimmt werden, wie zuvor beschrieben.

Wenn der Benutzer eine Fehlerabschaltung des Umrichters auslösen möchte (falls diese noch nicht erfolgt ist), muss der entsprechende Fehlerabschaltungscode in Parameter #10.38 geschrieben werden.

## 11.5 Support

Die Informationen aus den unten beschriebenen Parametern sollten immer notiert werden, bevor der Lieferant zum technischen Support kontaktiert wird.

### 11.5.1 Modul-Firmware

#81.02	Firmware - Hauptversion		
Zugang	NL	Bereich	00,00 bis 99,99
Standard	n.b.	Aktualisierungsrate	n.b.

#81.51	Firmware - Nebenversion		
Zugang	NL	Bereich	0 bis 99
Standard	n.b.	Aktualisierungsrate	n.b.

Die vollständige Firmware-Version des SM-Applications-Moduls kann für den entsprechenden Steckplatz abgefragt werden. Dieses Handbuch wurde für SM-Applications-Module geschrieben, die mit der Firmware V01.03.02 ausgestattet sind. In der Tabelle unten ist dargestellt, wie aus diesen Werten die vollständige Firmware-Version erstellt wird.

**Tabelle 11.3 Firmware-Version**

Haupt-version	Neben-version	Firmware-Version
1.01	5	V01.01.05

# 12 Migrationsanleitung

In diesem Kapitel werden einige der wichtigsten Unterschiede zwischen dem UD70-Produkt für Unidrive 1 und dem SM-Applications-Modul für Unidrive SP skizziert, die beim Konvertieren (Portieren) von benutzerdefinierten DPL-Programmen hilfreich sein können.

## 12.1 Unterschiede bei den Umrichterparametern

Der Unidrive SP-Parametersatz weist einige Unterschiede zum Unidrive 1 auf. Idealerweise sollten alle Parameter, auf die in einem DPL-Programm zugegriffen wird, am Anfang des Programms mit dem Befehl #DEFINE konfiguriert werden, was Konvertierungen erleichtert. Daher müssen Sie beim Portieren das Programm von Hand durchgehen und alle Verweise auf Umrichterparameter suchen. Der erste Schritt besteht darin, nach dem #-Symbol zu suchen und anschließend das Programm auf Funktionsblöcke zu durchsuchen, in die Parameterzeiger eingegeben werden können (z. B. PFixRead oder WRNET).

Wenn Sie Parameter gefunden haben, vergewissern Sie sich, dass sie im Unidrive SP identisch sind und dass dieselben Einheiten verwendet werden. Korrigieren Sie gegebenenfalls die Parameternummer oder die Skalierung.

## 12.2 UD70-Parameter

### 12.2.1 Konfigurationsparameter

Die Konfigurationsparameter befinden sich jetzt in Menü 15,16 oder 17, während sie sich bei UD70 immer in Menü 17 befanden. Es wird dringend empfohlen, neue benutzerdefinierte DPL-Programme, durch die Konfigurationsparameter geändert werden, so anzupassen, dass Menü 81 verwendet wird. So brauchen Sie nicht zu bestimmen, ob Menü 15, 16 oder 17 verwendet wird.

In der folgenden Tabelle werden einige der wichtigsten Unterschiede zwischen den beiden Konfigurationsmenüs skizziert:

**Tabelle 12.1 Änderungen an Konfigurationsparametern**

Parameter	Beschreibung	Änderungen
#17.03	Zeilennummer Fehler	Dies ist jetzt #81.48.
#17.06	RS-485-Betrieb	Die Modi 2, 3, 4, 9, 10, 11 und 12 werden nicht mehr unterstützt. Die Benutzermodi 6, 7 und 8 werden in der aktuellen Version der Software nicht unterstützt.
#17.08	RS485-Zeiger	Nicht unterstützt. Der Parameter ist jetzt die Antwortverzögerung.
#17.09	RS485-Zeiger 2	Nicht unterstützt. Der Parameter ist jetzt die Verzögerung für TX Enable.
#17.10	RS485-Skalierungsfaktor	Nicht unterstützt. Der Parameter ist jetzt der DPL-Druckerpfad.
#17.11	Zeitbasis für CLOCK-Task	Bei Umrichterversion < 01.05.00 ist der Standardwert gleich Null. Bei Umrichterversionen >=01.05.00 ist der Standardwert gleich 10 ms.
#17.12	Lageregler	Mit diesem Parameter wird jetzt die Zeitkoordinierungsrate für die neuen Tasks POS0 und POS1 geregelt, die sich von derjenigen für die UD70-Tasks für Drehzahl und Encoder unterscheidet.
#17.15	RS485-Fehlerabschaltungsmodus	Nicht unterstützt. Parameter wird nicht mehr verwendet.
#17.16	E/A-Steckbrückensynchronisation: Quelle	E/A-Steckbrücke wird nicht unterstützt. Der Parameter wird jetzt dazu verwendet, die Encoder-Aktualisierungsrate auszuwählen.
#17.20	Flash-Speicher beim Ausschalten	Dieser Parameter wird jetzt sofort wirksam.
#17.21	MD29MON deaktivieren	Mit diesem Parameter wird jetzt das Speichern von Menü 20 gesteuert.
#17.22	RS232 (Umrichter-Umrichter)	Nicht unterstützt. Mit dem Parameter wird jetzt die CTNet-Token Ring-Kennung definiert.
#17.23	EVENT-Auslösersteuerung	Siehe #81.35. Dieser Parameter ist jetzt die CTNet-Kennung.
#17.24	ANSI-Zweidraht-Antwort	Siehe Parameter #81.08. Der Parameter wird jetzt von CTNet verwendet.
#17.25+	Nicht verwendet	Es gibt jetzt 51 Konfigurationsparameter.

## 12.2.2 Menüs 18 und 19

Diese Menüs bleiben unverändert.

## 12.2.3 Menü 20

- Menü 20 besteht jetzt aus 40 Parametern statt aus 50. Die Parameter #20.21 bis #20.40 sind jetzt vollständige 32 Bit-Parameter mit Vorzeichen.
- Bisher galten die ersten 20 Parameter von Menü 20 als für Kommunikationsoptionen reserviert. Dies ist jetzt nicht mehr der Fall, da Kommunikationsparameter in den Menüs 15, 16 und 17 konfiguriert werden. Daher kann das ganze Menü 20 frei verwendet werden.
- Die Speicherung von Menü 20 hat sich ebenfalls geändert. Um die Parameter speichern zu können, müssen Sie den Parameter #81.21 auf 1 setzen und anschließend #81.19 oder #81.20 einstellen. Der Grund für diese Änderung liegt darin, dass Sie jetzt die Möglichkeit haben, zwei oder drei SM-Applications-Module in dem Umrichter anzubringen. Da Menü 20 im Flash-Speicher des Moduls gespeichert und aus diesem wiederhergestellt wird, kann nur ein Modul für die Bearbeitung von Menü 20 konfiguriert werden. Andernfalls wird es falsch wiederhergestellt.

## 12.2.4 SPS-Register

- Gute Nachrichten: Jetzt sind 200 zusätzliche **speicherbare** Register verfügbar. Diese befinden sich in den Registerbänken T und U (bzw. in den Menüs 74 und 75). Die Registersätze P, Q, R und S bleiben unverändert. Da jedoch kein eingebauter Lageregler vorhanden ist, kann Registersatz Q vollkommen frei verwendet werden.
- Auf dem UD70 mit CTNet waren einige Register in den Sätzen R und S reserviert. Dies ist nicht mehr der Fall: Alle 100 Register beider Sätze können für zyklische CTNet-Datenverbindungen verwendet werden. (Beachten Sie, dass 3 Register in R und S nach wie vor für die Verwendung mit zyklischen Easy Mode-Datenverbindungen reserviert sind.)

## 12.2.5 Parameter in Menü 90

An Teilen dieses Menüs wurden erhebliche Änderungen vorgenommen, und Programme MÜSSEN geändert werden, um diese zu berücksichtigen. Mit den Parametern für die Encoder-Position (#90.01/#90.03) werden jetzt die vollständigen Positionsdaten (einschließlich der Feinposition) bereitgestellt. Sie sind so skaliert, dass der volle 32 Bit-Wert eine Umdrehung darstellt. Separate Umdrehungszähler sind in #90.02 und #90.04 verfügbar.

# 12.3 Allgemeine Funktionen

## 12.3.1 Hardware

- CTNet ist standardmäßig eingebaut.
- Der RS232-Programmierungsanschluss ist nicht mehr verfügbar. Die Programmierung erfolgt jetzt über das CT-RTU-Protokoll durch den seriellen RJ45-Anschluss an der Vorderseite des Umrichters oder über CTNet. (Die alte DPL Toolkit-Software wird nicht unterstützt.)
- Der Digital-E/A ist jetzt 24 V-Positivlogik statt TTL. Außerdem gibt es jetzt zwei Digitalausgänge statt nur den einen. Darüber hinaus ist der Digital-E/A jetzt über Schraubanschlüsse wesentlich leichter zugänglich.

## 12.3.2 DPL-Sprache

DPL bleibt abwärtskompatibel. Die folgenden Erweiterungen wurden jedoch vorgenommen:

- Zusätzliche neue Konstruktionen wie FOR...LOOP und SELECT...CASE.
- Schachtelung: Die Stack-Größe wird jetzt den einzelnen Tasks zugewiesen und ist größer als beim UD70, so dass eine stärkere Schachtelung zulässig ist.

- Ein neuer Datentyp von Gleitkomma-Zahlen mit einfacher Präzision wurde hinzugefügt. Er bietet in der Ausführung einen geringfügigen Zeitvorteil gegenüber der doppelten Präzision (die weiterhin der Standard ist). Einfache oder doppelte Präzision wird mit Hilfe der neuen \$FLT SINGLE-Anweisung in der Programmkopfeile als globale Programmoption festgelegt.

**HINWEIS** Der Datentyp mit einfacher Präzision ist kein direkter Ersatz für die doppelte Präzision, und Benutzer sollten sich vergewissern, dass er ausreichende Genauigkeit für die jeweilige Anwendung liefert.

### 12.3.3 Mathematik

- Die Ausdruckskomplexität ist jetzt insofern erheblich verbessert, als Sie auf dem UD70 nur begrenzte Möglichkeiten innerhalb eines Ausdrucks hatten. Jetzt können Sie Ausdrücke mit wesentlich größerer Komplexität erstellen.
- Ein neuer Saldierungsoperator TRUNC wurde hinzugefügt. Dadurch können Gleitkomma-Zahlen so in ganze Zahlen konvertiert werden, dass das Ergebnis abgeschnitten statt gerundet wird.
- Gleitkomma-Vergleich: Beim UD70 wurde ein relativ ungenauer und unvorhersehbarer Fuzzy-Vergleich für Gleitkomma-Werte verwendet. Jetzt wird im SM-Applications-Modul mit einer Standard-Vergleichsmethode gearbeitet, wie sie in anderen Programmiersprachen verwendet wird, die keine Fuzzy-Methode ist. Dies kann jedoch zu einer unerwarteten Nebenwirkung führen. Beispiel:

```
f = 1.2 * 3
IF f > 3.59 or f < 3.61 THEN
    // Dies würde nicht „wahr“ ergeben.
ENDIF
```

Verwenden Sie für Prüfungen auf Gleichheit oder Ungleichheit einen Bereich. Beispiel:

```
f = 1.2 * 3
IF f > 3.59 or f < 3.61 THEN
    // Dies würde „wahr“ ergeben.
ENDIF
```

### 12.3.4 Tasks

Die Tasks wurden geändert.

- Die Tasks ENCODER und SPEED werden nicht mehr verwendet. Stattdessen wurden die Tasks POS0 und POS1 hinzugefügt. Außerdem kann der CLOCK-Task als Ersatz für den ENCODER-Task verwendet werden. Die Tasks POS0, POS1 und CLOCK laufen mit dem Umrichter synchron (wie bisher der SPEED-Task), jedoch können Sie jetzt ein Vielfaches dieser Zeit angeben: von 250  $\mu$ s bis 8 ms für die Tasks POS0 und POS1 und von 5 ms bis 200 ms für den CLOCK-Task. Benutzer können weiterhin die Namen SPEED und ENCODER verwenden, da sie Aliasnamen für POS0 und POS1 sind. (Dies kann beim Arbeiten mit SYPT Workbench V1 vorteilhaft sein, da es bei POS0 und POS1 ein Problem mit Stützpunkten gibt.) Beachten Sie jedoch, dass der zeitliche Ablauf anders **ist** und möglicherweise Code entsprechend geändert werden muss. Beachten Sie außerdem, dass sich der zeitliche Ablauf im Gegensatz zum UD70 nicht in Abhängigkeit von der Taktfrequenz des Umrichters ändert.
- Drei neue EVENT-Tasks wurden hinzugefügt, jedoch sollte dies bei der Programm-Migration keine Probleme verursachen.
- Die Zeitkoordinierung der EVENT-Tasks kann jetzt über DPL mit Hilfe des neuen Funktionsblocks SCHEDULEEVENT durchgeführt werden.

### 12.3.5 Benutzerdefinierte Funktionsblöcke

- Der UD70 war auf 10 ganzzahlige Eingänge und 10 ganzzahlige Ausgänge begrenzt. Außerdem gab es Ausrichtungsbeschränkungen. Das SM-Applications-Modul unterliegt keiner dieser Beschränkungen. Die Anzahl der Ein- und Ausgänge ist nur

durch den Speicherplatz begrenzt. Migrationsprobleme sollten nicht auftreten. (Hinweis: In der SYPT Workbench gelten zurzeit bezüglich der für das SM-Applications-Modul entwickelten Programme noch immer die alten Beschränkungen.)

## 12.3.6 Kommunikation

### CTNet

- CTNet-Kommunikation ist jetzt Standard und keine Option mehr. Alle Konfigurationsparameter haben sich geändert. Die Synchronisation wird insofern anders generiert, als sie jetzt in echten Millisekunden angegeben wird statt wie beim UD70 als nächstes Vielfaches von 1,38 ms. Außerdem wird der EVENT-Task jetzt nicht nur auf den Slave-Knoten ausgelöst, sondern auch auf dem Master-Knoten - beim UD70 war dies nicht der Fall.
- Alle Konfigurationsparameter wurden von Menü 20 in das Optionsmenü (15/16/17/81) verschoben.
- Die speziellen Variablen für die Fehlersuche, zum Beispiel NOFMESSAGES, werden nicht mehr unterstützt. Stattdessen werden die Daten zur Fehlersuche von einem neuen Funktionsblock mit dem Namen CTNETDIAGS zurückgegeben. Alte Programme mit diesen Variablen für die Fehlersuche werden nicht kompiliert und müssen so geändert werden, dass der neue Funktionsblock verwendet wird.
- Der EVENT-Task zum Auslösen der SYNC-Meldung wird jetzt nicht nur auf Slave-Knoten ausgeführt, sondern auch auf dem Knoten, auf dem die SYNC-Meldung tatsächlich generiert wird.

### RS485-Anschluss

- Das CT-ANSI-Protokoll wurde so erweitert, dass jetzt volle 32 Bit abgefragt und geschrieben werden können.
- E/A-Feldmodus 10 wird nicht mehr unterstützt.
- Die Umrichter-zu-Umrichter- und Kaskadenmodi 2, 3 und 4 werden nicht unterstützt.
- Der RS232-Modus (Umrichter-Umrichter) wird nicht unterstützt.
- Modus 9 wird aufgrund von Hardware-Beschränkungen nicht unterstützt.
- Die Modi 11 und 12 werden nicht unterstützt. Modus 11, durch den die Software-Puffer des UD70 umgangen wurden, wird nicht mehr benötigt, da für Benutzermodi keine Software-Puffer verwendet werden.
- Neue Master-Modi für Modbus-RTU und damit zusammenhängende Befehle sind jetzt verfügbar.

## 12.4 SM-Applications-Portierhilfe

### 12.4.1 Überblick

Die SYPT Workbench v1.6.0 und höher umfasst jetzt eine Portierhilfe, durch die Benutzer auf alle Funktionen hingewiesen werden, die sich auf dem SM-Applications-Modul von den UD7X-Modulen unterscheiden. Die Warnungen werden von SYPT Workbench in einem Fehlerfenster angezeigt. Dieses Fehlerfenster enthält die Zeilennummer, auf die sich die Fehler- bzw. Portierinformationen beziehen, so dass durch Doppelklicken auf die Zeile in dem Fehlerfenster die DPL-Quellezeile mit dem geänderten Element von SYPT Workbench angezeigt wird. Diese Portierhilfe kann mit Hilfe der neuen \$PORTING-Anweisung aktiviert werden, die hinter den Haupt-Kopfzeilendaten eingefügt wird:

```
$AUTHOR Kevin Vedmore
$COMPANY Control Techniques
$title Testprogramm
$VERSION v1.0
$DRIVE UNIDRVSP
$PORTING
```

## 12.4.2 Gemeldete Unterschiede

Informationen der folgenden Arten werden von SYPT Workbench gemeldet, wenn die \$SPORTING-Anweisung verwendet wird:

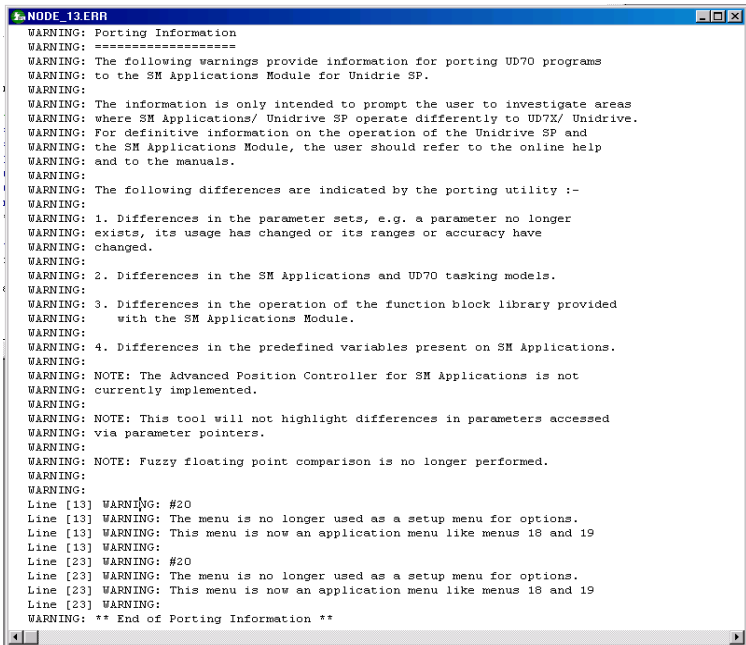
1. Die Tasks POS0 und POS1 werden an Stelle der älteren Tasks für Drehzahl, Drehmoment und Encoder empfohlen. Der Clock-Task wird als Alternative zur Verwendung eines Pos-Tasks an Stelle eines Encoder-Tasks empfohlen.
2. Unterschiede in den Unidrive SP- oder SM-Applications-Parametersätzen, z. B. geänderte Anzahl der Dezimalstellen, geänderte Namen und Verwendung oder Parameter, die nicht mehr existieren oder verschoben wurden.

Hinweis: Spezifische Warnungen bezüglich einer Änderung der Anzahl der Dezimalstellen werden nur dann angezeigt, wenn der Zugang zu einem Parameter mit Hilfe der #INT-Notation erfolgt. Für nicht über #INT erfolgte Parameterzugänge wird eine allgemeine Warnung bezüglich Änderungen der Anzahl der Dezimalstellen angezeigt.

3. Unterschiedliche Funktionsweise von RTL- und OS-Funktionsblöcken.

Ein Beispiel für den von der Portierhilfe angezeigten Meldungstyp ist unten abgebildet.

**Abbildung 12-1 Informationen im Portierfenster**



```

NODE_13.ERR
WARNING: Porting Information
WARNING: =====
WARNING: The following warnings provide information for porting UD70 programs
WARNING: to the SM Applications Module for Unidrive SP.
WARNING:
WARNING: The information is only intended to prompt the user to investigate areas
WARNING: where SM Applications/ Unidrive SP operate differently to UD7X/ Unidrive.
WARNING: For definitive information on the operation of the Unidrive SP and
WARNING: the SM Applications Module, the user should refer to the online help
WARNING: and to the manuals.
WARNING:
WARNING: The following differences are indicated by the porting utility :-
WARNING:
WARNING: 1. Differences in the parameter sets, e.g. a parameter no longer
WARNING: exists, its usage has changed or its ranges or accuracy have
WARNING: changed.
WARNING:
WARNING: 2. Differences in the SM Applications and UD70 tasking models.
WARNING:
WARNING: 3. Differences in the operation of the function block library provided
WARNING: with the SM Applications Module.
WARNING:
WARNING: 4. Differences in the predefined variables present on SM Applications.
WARNING:
WARNING: NOTE: The Advanced Position Controller for SM Applications is not
WARNING: currently implemented.
WARNING:
WARNING: NOTE: This tool will not highlight differences in parameters accessed
WARNING: via parameter pointers.
WARNING:
WARNING: NOTE: Fuzzy floating point comparison is no longer performed.
WARNING:
WARNING:
Line [13] WARNING: #20
Line [13] WARNING: The menu is no longer used as a setup menu for options.
Line [13] WARNING: This menu is now an application menu like menus 18 and 19
Line [13] WARNING:
Line [23] WARNING: #20
Line [23] WARNING: The menu is no longer used as a setup menu for options.
Line [23] WARNING: This menu is now an application menu like menus 18 and 19
Line [23] WARNING:
WARNING: ** End of Porting Information **
```

# 13 Kurzreferenz

Die vollständigen Details dieser Parameter finden Sie in Abschnitt 5 *Parameter*.

**Tabelle 13.1 Parameter**

Parameter	Beschreibung	Bereich	Standard
#81.01	Moduloptionscode	0-499	n.b.
#81.02	Firmware-Version des Moduls	0-99.99	n.b.
#81.03	DPL-Programmstatus	0-3	n.b.
#81.04	Verfügbare Systemressource %	0 bis 100 %	n.b.
#81.05	RS485-Adresse	0-255	11
#81.06	RS-485-Betrieb	0-255	1
#81.07	RS-485-Baudrate	0-9	4
#81.08	RS485-Antwortverzögerung	0 bis 255 ms	2 ms
#81.09	RS485: Verzögerung für TX Enable	0 bis 1 ms	0 ms
#81.10	DPL-Druckerpfad	0-1	0
#81.11	Zeitbasis für CLOCK-Task	0 bis 200 ms	0
#81.12	POS-Zeitkoordinierungsrate	0-6	0
#81.13	Automatisches Ausführen freigeben	0-1	1
#81.14	Globale Laufzeit-Fehlerabschaltung freigeben	0-1	0
#81.15	Reserviert	0	0
#81.16	Aktualisierungsrate der Encoder-Daten	0-3	0
#81.17	Fehlerabschaltung beim Überschreiten von Parameterbereichsgrenzen freigeben	0-1	0
#81.18	Watchdog freigeben	0-1	0
#81.19	Speicheranforderung	0-1	0
#81.20	Bei „UU“-Fehlerabschaltung speichern	0-1	0
#81.21	Menü 20 mit speichern/wiederherstellen	0-1	0
#81.22	CTNet-Token Ring-Kennung	0-255	0
#81.23	CTNet-Knotenadresse	0-255	0
#81.24	CTNet-Baudrate	0-3	1
#81.25	CTNet-Synchronisationseinrichtung (LLSS)	0-9999	0
#81.26	CTNet Easy Mode 1-Zielknoten (KKKSS)	0-25503	0
#81.27	CTNet Easy Mode 1-Quellparameter (MMPP)	0-9999	0
#81.28	CTNet Easy Mode 2-Zielknoten (KKKSS)	0-25503	0
#81.29	CTNet Easy Mode 2-Quellparameter (MMPP)	0-9999	0
#81.30	CTNet Easy Mode 3-Zielknoten (KKKSS)	0-25503	0
#81.31	CTNet Easy Mode 3-Quellparameter (MMPP)	0-9999	0
#81.32	CTNet Easy Mode-Steckplatz 1: Ziel (MMPP)	0-9999	0
#81.33	CTNet Easy Mode-Steckplatz 2: Ziel (MMPP)	0-9999	0
#81.34	CTNet Easy Mode-Steckplatz 3: Ziel (MMPP)	0-9999	0
#81.35	CTNet-Synchronisation: Kennung EVENT-Task	0-4	0
#81.36	CTNet-Fehlersucheparameter	-3-32767	n.b.
#81.37	Laden bei freigegebenem Umrichter verweigern	0-1	0
#81.38	APR-Laufzeit-Fehlerabschaltung	0-1	0
#81.39	Status des Umrichters in der Synchronisation zwischen den Optionsmodulen	0-3	0
#81.42	Freeze: Position Istwert-Encoder	0-1	0
#81.43	Freeze: Invertierung	0-1	0
#81.44	Task-Prioritätsebene	0-255	0
#81.48	DPL-Zeilenummer des Fehlers	32 Bit	0
#81.49	Kennung für Benutzerprogramme	16 Bit	0
#81.50	Laufzeitfehlercode	0-255	0
#81.51	Software-Nebenversion	0-99	n.b.



**Tabelle 13.2 SM-Applications-Fehlercodes**

Fehlercode	Grund	Fehlerabschaltung des Umrichters?	ERROR-Task?	Programm angehalten?
39	Stack-Überlauf im Benutzerprogramm	Ja	Nein	Ja
40	Unbekannter Fehler - bitte Lieferanten kontaktieren	Ja	Nein	Ja
41	Parameter existiert nicht. Der Benutzer hat versucht, im DPL-Programm einen nicht existierenden Parameter abzufragen oder zu schreiben.	Vielleicht	Ja	Ja
42	Versuch, in einem Nur-Lese-Parameter zu schreiben.	Vielleicht	Ja	Ja
43	Versuch, einen Nur-Schreib-Parameter abzufragen.	Vielleicht	Ja	Ja
44	Parameterwert außerhalb des gültigen Bereiches. (Der Benutzer hat einen unzulässigen Wert in einen Parameter innerhalb eines DPL-Programms geschrieben.) Wenn Parameter #MM.17=0 ist, wird der geschriebene Wert automatisch begrenzt, und es tritt kein Fehler auf.	Vielleicht	Ja	Ja
45	Ungültige Synchronisationsmodi	Ja	Nein	Läuft nicht
46	Nicht verwendet	n.b.	n.b.	n.b.
47	Synchronisation mit CTSync-Master verloren	Ja	Ja	Ja
48	RS485 nicht in Benutzermodus. Tritt auf, wenn der Benutzer versucht, einen DPL-Befehl für den Benutzermodus von RS485 zu verwenden, der RS485-Anschluss sich jedoch nicht im Benutzermodus befindet.	Ja	Ja	Ja
49	Ungültige RS485-Konfiguration. Beispiel: ungültiger Modus.	Ja	Ja	Ja
50	Mathematischer Fehler - Division durch Null oder Überlauf.	Vielleicht	Ja	Ja
51	Array-Index außerhalb des gültigen Bereiches. Beispiel: arr%{20}, wenn arr% nur für 19 Elemente dimensioniert wurde.	Vielleicht	Ja	Ja
52	Benutzer-Fehlerabschaltung durch Steuerwort. Ausgelöst durch Einstellen des Fehlerabschaltungsbits in Steuerwort #90.11.	Ja	Nein	Nein
53	DPL-Programm nicht kompatibel mit Ziel. Beispiel: Herunterladen eines für UD70 kompilierten Programms.	Ja	n.b.	n.b.
54	DPL-Task-Überlauf. Tritt auf, wenn der DPL-Code innerhalb eines Echtzeit-Tasks (z. B. POS0) nicht rechtzeitig abgeschlossen werden kann. Mit Hilfe von Parameter #88.02 kann der Task ermittelt werden, in dem dieser Fehler auftritt. Überprüfen Sie, ob die Zeitkoordinierungsrate korrekt ist und im Task keine Schleifen vorkommen. Dieser Fehler kann auch eine Folge von externen Einflüssen sein, zum Beispiel eines großen Datenstoßes, der über CTNet eingeht. Dieses Problem kann behoben werden, indem die Priorität der CTNet-Task niedriger eingestellt wird als die der POS-Tasks. Dies kann jedoch dazu führen, dass der CTNet-Task nicht mehr versorgt wird. Weitere Informationen finden Sie unter Parameter <b>#81.44</b> auf Seite 32.	Vielleicht	Ja	Ja
55	Nicht verwendet	n.b.	n.b.	n.b.
56	Ungültige Konfiguration der Zeitgebereinheit	Ja	Ja	Ja
57	Funktionsblock existiert nicht.	Ja	Ja	Läuft nicht
58	Flash-SPS-Speicherung fehlerhaft. Tritt beim Starten auf und bedeutet, dass SPS-Registersatz (P/Q/T/U) und Menü 20 nicht wiederhergestellt wurden. Wenn dieses Problem weiterhin besteht, könnte dies auf einen Hardware-Fehler hinweisen. Kontaktieren Sie daher Ihren Lieferanten.	Ja	Ja	Läuft nicht
59	Applikationsmodul von Umrichter als Synchronisations-Master abgelehnt.	Ja	Ja	Ja
60	CTNet-Hardware-Fehler. Bitte kontaktieren Sie Ihren Lieferanten.	Vielleicht	Nein	Nein
61	Ungültige CTNet-Konfiguration. Überprüfen Sie alle Konfigurationsparameter.	Vielleicht	Nein	Nein
62	Ungültige CTNet-Baudrate. Überprüfen Sie Parameter #MM.24 und die Netzwerkverbindungen.	Vielleicht	Nein	Nein
63	Ungültige CTNet-Knotenennung. Überprüfen Sie Parameter #MM.23.	Vielleicht	Nein	Nein
64	Überlastung der Digitalausgänge. Beide Digitalausgänge werden inaktiv, wenn dies auftritt, und bleiben inaktiv, bis der Fehlerzustand zurückgesetzt wurde. Die Schwelle für eine Fehlerabschaltung liegt bei 20 mA.	Ja	Ja	Ja
65	Ungültige(r) Parameter in Funktionsblock. Sie haben innerhalb eines DPL-Programms einen FB aufgerufen, jedoch ist mindestens eine Eingabe ungültig.	Ja	Ja	Ja

**Tabelle 13.2 SM-Applications-Fehlercodes (Fortsetzung)**

Fehlercode	Grund	Fehlerabschaltung des Umrichters?	ERROR-Task?	Programm angehalten?
66	Heap-Speicher des Benutzers zu groß. Das Programm wurde für ein Ziel kompiliert, das über mehr RAM-Speicher verfügt als dieses. Tritt beim Starten auf.	Ja	Nein	Läuft nicht
67	RAM-Datei existiert nicht, oder es wurde eine Datei angegeben, die keine RAM-Datei ist.	Ja	Ja	Ja
68	Die angegebene RAM-Datei ist nicht mit einem Array verknüpft.	Ja	Ja	Ja
69	Datenbank-Cache für Umrichterparameter konnte im Flash-Speicher nicht aktualisiert werden.	Ja	Nein	Läuft nicht
70	Benutzerprogramm bei freigegebenem Umrichter heruntergeladen. Tritt auf, wenn #MM.37 = 1 ist und ein Programm heruntergeladen wird.	Vielleicht	Nein	Ja
71	Umrichtermodus konnte nicht geändert werden	Ja	Nein	Ja
72	Ungültige CTNet-Pufferoperation	Ja	Ja	Ja
73	Fehler bei der Schnellinitialisierung der Parameter	Ja	Nein	Nein
74	Übertemperatur	Ja	Ja	Ja
75	Hardware nicht verfügbar. Im Benutzerprogramm wurde versucht, auf nicht verfügbare Hardware zuzugreifen. Beispiel: Zugriff auf Digital-E/A, RS485-Anschluss oder CTNet im SM-Applications Lite-Modul.	Ja	Ja	Ja
76	Modultyp kann nicht aufgelöst werden. Das Modul wird nicht erkannt.	Ja	Nein	Läuft nicht
77	Fehler bei der Kommunikation mit dem Optionsmodul in Steckplatz 1.	Ja	Ja	Ja
78	Fehler bei der Kommunikation mit dem Optionsmodul in Steckplatz 2.	Ja	Ja	Ja
79	Fehler bei der Kommunikation mit dem Optionsmodul in Steckplatz 3.	Ja	Ja	Ja
80	Fehler bei der Kommunikation mit einem Optionsmodul (unbekannter Steckplatz).	Ja	Ja	Ja
81	APR-interner Fehler. Siehe Parameter #81.38 (Beispiel: Verwendung einer nicht initialisierten CAM-Funktion).	Vielleicht	Ja	Ja
82	Kommunikation zum Umrichter fehlerhaft.	Vielleicht	Ja	Ja

**Tabelle 13.3 Anschlussklemmen**

Anschlussklemme	Funktion	Beschreibung
1	0V SC	0 V-Verbindung für RS485-Anschluss
2	/RX	EIA-RS485-Empfangsleitung (negativ). Eingehend.
3	RX	EIA-RS485-Empfangsleitung (positiv). Eingehend.
4	/TX	EIA-RS485-Sendeleitung (negativ). Abgehend.
5	TX	EIA-RS485-Sendeleitung (positiv). Abgehend.
6	CTNet A	CTNet-Datenleitung
7	CTNet Shield	Schirmverbindung für CTNet
8	CTNet B	CTNet-Datenleitung
9	0V	0 V-Verbindung für Digital-E/A
10	DIGIN0	Digitaleingang 0
11	DIGIN1	Digitaleingang 1
12	DIGOUT0	Digitalausgang 0
13	DIGOUT1	Digitalausgang 1