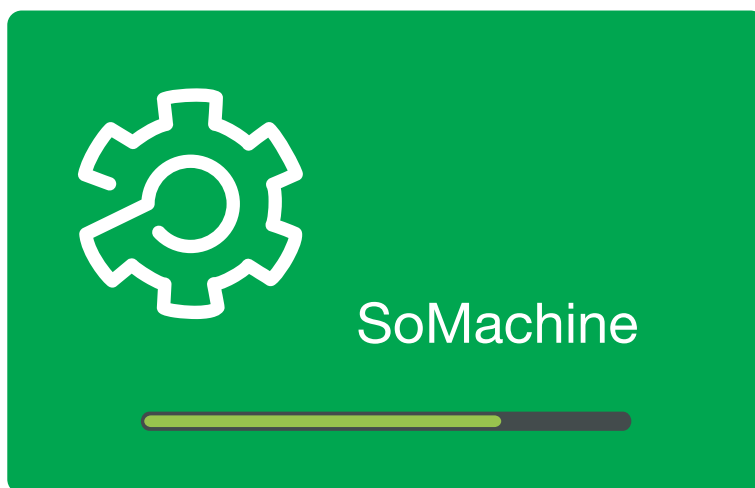


Altivar Library

Funktionsbausteine

Softwarehandbuch

V2.08, 04.2011



Wichtige Hinweise

Dieses Handbuch ist Teil des Produkts.

Lesen und befolgen Sie dieses Handbuch.

Bewahren Sie dieses Handbuch auf.

Geben Sie dieses Handbuch und alle zum Produkt gehörenden Unterlagen an alle Benutzer des Produktes weiter.

Lesen und beachten Sie besonders alle Sicherheitshinweise und das Kapitel "Bevor Sie beginnen - Sicherheitsinformationen".

Nicht alle Produkte sind in allen Ländern erhältlich.
Die Verfügbarkeit der Produkte entnehmen Sie bitte dem aktuellen Katalog.

Wir behalten uns das Recht vor ohne Ankündigung technische Änderungen vorzunehmen.

Alle Angaben sind technische Daten und keine zugesicherten Eigenschaften.

Die meisten Produktbezeichnungen sind auch ohne besondere Kennzeichnung als Warenzeichen der jeweiligen Inhaber zu betrachten.

Inhaltsverzeichnis

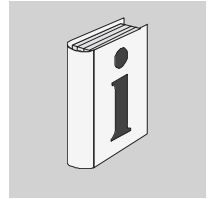


Wichtige Hinweise	2
Inhaltsverzeichnis	3
Über dieses Handbuch	7
1 Bevor Sie beginnen - Sicherheitsinformationen	9
1.1 Qualifikation des Personals.....	9
1.2 Bestimmungsgemäße Verwendung	9
1.3 Gefahrenklassen	10
1.4 Grundlegende Informationen.....	11
1.5 Normen und Begrifflichkeiten.....	12
2 Anleitung Altivar Library	13
2.1 Liste der Funktionsbausteine	15
2.2 Grundlegende Eingänge und Ausgänge.....	17
2.2.1 Signalverhalten bei Funktionsbausteinen mit dem Eingang <i>Enable</i>	19
2.2.2 Signalverhalten bei Funktionsbausteinen mit dem Eingang <i>Execute</i>	21
2.3 Single axis.....	23
2.3.1 Initialisierung.....	24
2.3.1.1 <i>MC_Power_ATV</i>	24
2.3.2 Betriebsart Jog.....	25
2.3.2.1 <i>MC_Jog_ATV</i>	25
2.3.3 Betriebsart Speed Control.....	27
2.3.3.1 <i>VelocityControlAnalogInput_ATV</i>	27
2.3.3.2 <i>VelocityControlSelectAI_ATV</i>	28
2.3.4 Betriebsart Profile Velocity	29
2.3.4.1 <i>MC_MoveVelocity_ATV</i>	29
2.3.5 Stoppen.....	31
2.3.5.1 <i>MC_Stop_ATV</i>	31
2.4 Administrative	32
2.4.1 Parameter lesen	32
2.4.1.1 <i>MC_ReadActualVelocity_ATV</i>	32
2.4.1.2 <i>MC_ReadActualTorque_ATV</i>	33
2.4.1.3 <i>MC_ReadStatus_ATV</i>	34
2.4.1.4 <i>MC_ReadParameter_ATV</i>	36
2.4.1.5 <i>GetSupplierVersion</i>	37
2.4.2 Parameter schreiben.....	38
2.4.2.1 <i>MC_WriteParameter_ATV</i>	38
2.4.2.2 <i>SetDriveRamp_ATV</i>	40
2.4.2.3 <i>SetFrequencyRange_ATV</i>	41
2.4.2.4 <i>ResetParameters_ATV</i>	42

2.4.2.5	StoreParameters_ATV.....	43
2.4.3	Gerätekonfiguration speichern und wiederherstellen	44
2.4.3.1	UploadDriveParameter_ATV.....	44
2.4.3.2	DownloadDriveParameter_ATV.....	45
2.4.4	Eingänge und Ausgänge	46
2.4.4.1	ReadAnalogInput_ATV.....	46
2.4.4.2	MC_ReadDigitalInput_ATV.....	47
2.4.4.3	MC_ReadDigitalOutput_ATV.....	49
2.4.4.4	MC_WriteDigitalOutput_ATV.....	51
2.4.5	Fehlerbehandlung.....	52
2.4.5.1	MC_ReadAxisError_ATV.....	52
2.4.5.2	MC_Reset_ATV.....	56
2.5	Device Function.....	57
2.5.1	Startup.....	57
2.5.1.1	Altivar_Startup.....	57
2.5.2	Control.....	66
2.5.2.1	Altivar31_Control.....	66
2.5.2.2	Altivar71_Control.....	77
2.5.2.3	Altivar32_Control.....	88
3	Glossar.....	99
3.1	Einheiten und Umrechnungstabellen.....	99
3.1.1	Länge.....	99
3.1.2	Masse.....	99
3.1.3	Kraft.....	99
3.1.4	Leistung.....	99
3.1.5	Rotation.....	100
3.1.6	Drehmoment.....	100
3.1.7	Trägheitsmoment.....	100
3.1.8	Temperatur.....	100
3.1.9	Leiterquerschnitt.....	100
3.2	Begriffe und Abkürzungen.....	101
4	Stichwortverzeichnis.....	103

019844113879, V2.08, 04.2011

Über dieses Handbuch



Dieses Handbuch ist ein Auszug aus der Online-Hilfe von SoMachine. Lesen Sie aufmerksam und vollständig alle Handbücher der Online-Hilfe von SoMachine sowie der verwendeten Produkte.

Ziel dieses Dokuments

Dieses Dokument beschreibt die Funktionen der Altivar Library.

Software-Umgebung	Geräte	Feldbus
SoMachine Unterstützt werden Device Descriptions ab Version 4.0.0.0	ATV31/ATV312 ATV71/ATV32	CANopen

Gültigkeitsbereich

Dieses Dokument ist gültig für SoMachine ab Version 2.0.

Bezugsquelle Handbücher

Die aktuellen Handbücher stehen im Internet unter folgender Adresse zum Download bereit:

<http://www.schneider-electric.com>

Korrekturen und Anregungen

Auch wir sind ständig bemüht uns zu verbessern. Deswegen freuen wir uns über Ihre Anregungen und Korrekturen zu diesem Handbuch.

Sie erreichen uns per eMail unter:
techcomm@schneider-electric.com.

Arbeitsschritte

Wenn Arbeitsschritte nacheinander durchgeführt werden müssen, finden Sie folgende Darstellung:

- Besondere Voraussetzungen für die nachfolgenden Arbeitsschritte
 - ▶ Arbeitsschritt 1
 - ◁ Besondere Reaktion auf diesen Arbeitsschritt
 - ▶ Arbeitsschritt 2

Wenn zu einem Arbeitsschritt eine Reaktion angegeben ist, können Sie daran die korrekte Ausführung des Arbeitsschritts kontrollieren.

Wenn nicht anders angegeben, sind die einzelnen Handlungsschritte in der angegebenen Reihenfolge auszuführen.

SI-Einheiten

SI-Einheiten sind die Originalwerte. Umgerechnete Einheiten stehen in Klammern hinter dem Originalwert und können gerundet sein.

Beispiel:
Minimaler Leiterquerschnitt: 1,5 mm² (AWG 14)

Glossar

Erklärung von Fachbegriffen und Abkürzungen.

Stichwortverzeichnis

Liste von Suchbegriffen, die zum entsprechenden Inhalt verweisen.

Rechtliche Hinweise

Die Informationen in der vorliegenden Dokumentation enthalten allgemeine Beschreibungen und/oder technische Leistungsmerkmale der hier erwähnten Produkte. Diese Dokumentation dient nicht als Ersatz für das Ermitteln der Eignung oder Verlässlichkeit dieser Produkte für bestimmte Verwendungsbereiche des Benutzers und darf nicht zu diesem Zweck verwendet werden. Jeder Benutzer oder Integrator ist verpflichtet, angemessene und vollständige Risikoanalysen, Bewertungen und Tests der Produkte im Hinblick auf deren jeweils spezifischen Verwendungszweck vorzunehmen. Weder Schneider Electric noch deren Tochtergesellschaften oder verbundenen Unternehmen sind für einen Missbrauch der Informationen in der vorliegenden Dokumentation verantwortlich oder können diesbezüglich haftbar gemacht werden. Verbesserungs- und Änderungsvorschläge sowie Hinweise auf angetroffene Fehler werden jederzeit gern entgegengenommen.

Dieses Dokument darf ohne entsprechende vorhergehende, ausdrückliche und schriftliche Genehmigung durch Schneider Electric weder in Teilen noch als Ganzes in keiner Form und auf keine Weise, weder anhand elektronischer noch mechanischer Hilfsmittel, reproduziert oder fotokopiert werden.

Bei der Montage und Verwendung dieses Produkts sind alle zutreffenden staatlichen, landesspezifischen, regionalen und lokalen Sicherheitsbestimmungen zu beachten. Aus Sicherheitsgründen und um die Übereinstimmung mit dokumentierten Systemdaten besser zu gewährleisten, sollten Reparaturen an Komponenten nur vom Hersteller vorgenommen werden.

Beim Einsatz von Geräten für Anwendungen mit technischen Sicherheitsanforderungen sind die relevanten Anweisungen zu beachten.

Die Verwendung anderer Software als der Schneider Electric-eigenen bzw. einer von Schneider Electric genehmigten Software in Verbindung mit den Hardwareprodukten von Schneider Electric kann Körperverletzung, Schäden oder einen fehlerhaften Betrieb zur Folge haben.

Die Nichtbeachtung dieser Informationen kann Verletzungen oder Materialschaden zur Folge haben.

1 Bevor Sie beginnen - Sicherheitsinformationen

1

1.1 Qualifikation des Personals

Arbeiten an und mit diesem Produkt dürfen nur von Fachkräften vorgenommen werden, die den Inhalt dieses Handbuchs und alle zum Produkt gehörenden Unterlagen kennen und verstehen. Weiterhin müssen diese Fachkräfte eine Sicherheitsunterweisung erhalten haben, um die entsprechenden Gefahren zu erkennen und zu vermeiden. Die Fachkräfte müssen aufgrund ihrer fachlichen Ausbildung sowie ihrer Kenntnisse und Erfahrungen in der Lage sein, mögliche Gefahren vorherzusehen und zu erkennen, die durch Einsatz des Produktes, durch Änderung der Einstellungen sowie durch mechanische, elektrische und elektronische Ausrüstung der Gesamtanlage entstehen können.

Den Fachkräften müssen alle geltenden Normen, Bestimmungen und Unfallverhütungsvorschriften, die bei Arbeiten am und mit dem Produkt beachtet werden müssen, bekannt sein.

1.2 Bestimmungsgemäße Verwendung

Dieses Produkt ist eine Bibliothek zur Benutzung mit den dafür vorgesehenen Steuerungen und Antriebsverstärkern im Industriebereich.

Die gültigen Sicherheitsvorschriften, die spezifizierten Bedingungen und technischen Daten sind jederzeit einzuhalten.

Vor dem Einsatz des Produktes ist eine Risikobeurteilung in Bezug auf die konkrete Anwendung durchzuführen. Entsprechend dem Ergebnis sind die Sicherheitsmaßnahmen zu ergreifen.

Da das Produkt als Teil eines Gesamtsystems verwendet wird, müssen Sie die Personensicherheit durch das Konzept dieses Gesamtsystems (zum Beispiel Maschinenkonzept) gewährleisten.

Andere Verwendungen sind nicht bestimmungsgemäß und können Gefahren verursachen.

Elektrische Geräte und Einrichtungen dürfen nur von qualifiziertem Personal installiert, betrieben, gewartet und instand gesetzt werden.

1.3 Gefahrenklassen

Sicherheitshinweise sind im Handbuch mit Warnsymbolen gekennzeichnet. Zusätzlich finden Sie Symbole und Hinweise am Produkt, die Sie vor möglichen Gefahren warnen.

Abhängig von der Schwere einer Gefahrensituation werden Sicherheitshinweise in 4 Gefahrenklassen unterteilt.

⚠ GEFAHR
GEFAHR macht auf eine unmittelbar gefährliche Situation aufmerksam, die bei Nichtbeachtung unweigerlich einen schweren oder tödlichen Unfall zur Folge hat.

⚠ WARNUNG
WARNUNG macht auf eine möglicherweise gefährliche Situation aufmerksam, die bei Nichtbeachtung unter Umständen einen schweren oder tödlichen Unfall oder Beschädigung an Geräten zur Folge hat.

⚠ VORSICHT
VORSICHT macht auf eine möglicherweise gefährliche Situation aufmerksam, die bei Nichtbeachtung unter Umständen einen Unfall oder Beschädigung an Geräten zur Folge hat.

VORSICHT
VORSICHT ohne das Warnsymbol macht auf eine möglicherweise gefährliche Situation aufmerksam, die bei Nichtbeachtung unter Umständen eine Beschädigung an Geräten zur Folge hat.

1.4 Grundlegende Informationen

⚠ WARNUNG

VERLUST DER STEUERUNGSKONTROLLE

- Bei der Entwicklung des Steuerungskonzeptes muss der Anlagenhersteller die potentiellen Ausfallmöglichkeiten der Steuerungspfade berücksichtigen und für bestimmte kritische Funktionen Mittel bereitstellen, mit denen während und nach dem Ausfall eines Steuerungspfades sichere Zustände erreicht werden. Beispiele für kritische Steuerungsfunktionen sind: NOT-HALT, Endlagen-Begrenzung, Spannungsausfall und Wiederanlauf.
- Für kritische Funktionen müssen separate oder redundante Steuerungspfade vorhanden sein.
- Die Anlagensteuerung kann Kommunikationsverbindungen umfassen. Der Anlagenhersteller muss die Folgen unerwarteter Zeitverzögerungen oder Ausfälle der Kommunikationsverbindung berücksichtigen.
- Beachten Sie alle Unfallverhütungsvorschriften sowie alle geltenden Sicherheitsbestimmungen.¹⁾
- Jede Anlage, in der das in diesem Handbuch beschriebene Produkt verwendet wird, muss vor dem Betrieb einzeln und gründlich auf korrekte Funktion überprüft werden.

Nichtbeachtung dieser Vorkehrungen kann zu Tod oder schwerwiegenden Verletzungen führen.

1) Für USA: siehe NEMA ICS 1.1 (neueste Ausgabe), "Safety Guidelines for the Application, Installation, and Maintenance of Solid State Control" sowie NEMA ICS 7.1 (neueste Ausgabe), "Safety Standards for Construction and Guide for Selection, Installation and Operation of Adjustable-Speed Drive Systems".

⚠ WARNUNG

UNBEABSICHTIGTES VERHALTEN DURCH UNGEEIGNETE FEHLERBEHANDLUNG

Ungeeignete Fehlerbehandlung kann Bewegungen oder Signale verändern sowie Überwachungsfunktionen deaktivieren.

- Programmieren Sie die Fehlerbehandlung sorgfältig.
- Überprüfen Sie die Wirksamkeit der Fehlerbehandlung.

Nichtbeachtung dieser Vorkehrungen kann zu Tod, schwerwiegenden Verletzungen oder Materialschäden führen.

⚠ WARNUNG

UNBEABSICHTIGTES VERHALTEN DURCH VERÄNDERUNGEN AN DER BIBLIOTHEK

- Führen Sie keine Veränderungen oder Manipulationen an der Bibliothek durch.

Nichtbeachtung dieser Vorkehrungen kann zu Tod, schwerwiegenden Verletzungen oder Materialschäden führen.

1.5 Normen und Begrifflichkeiten

In diesem Handbuch verwendete Fachbegriffe, Terminologie und die entsprechenden Beschreibungen sollen die Begriffe und Definitionen der einschlägigen Normen wiedergeben.

Im Bereich der Antriebstechnik handelt es sich dabei unter anderem um die Begriffe "Sicherheitsfunktion", "sicherer Zustand", "Fault", "Fault Reset", "Ausfall", "Fehler", "Fehlermeldung", "Warnung", "Warnmeldung" usw.

Zu den einschlägigen Normen gehören unter anderem:

- IEC 61800 Reihe: "Elektrische Leistungsantriebssysteme mit einstellbarer Drehzahl"
- IEC 61158 Reihe: "Digitale Datenkommunikation in der Leittechnik - Feldbus für industrielle Leitsysteme"
- IEC 61784 Reihe: "Industrielle Kommunikationsnetze - Profile"
- IEC 61508 Reihe: "Funktionale Sicherheit sicherheitsbezogener elektrischer/elektronischer/programmierbarer elektronischer Systeme"

Siehe hierzu auch das Glossar am Ende dieses Handbuchs.

2 Anleitung Altivar Library



Name der Bibliothek Altivar Library (ATV)

Software-Umgebung	Geräte	Feldbus
SoMachine Unterstützt werden Device Descriptions ab Version 4.0.0.0	ATV31/ATV312 ATV71/ATV32	CANopen

Die hier beschriebenen Funktionsbausteine dienen zur Steuerung der Antriebsverstärker ATV in CANopen-Feldbussen mit der Software-Umgebung SoMachine. Die Funktionsbausteine entsprechen IEC 61131-3.

Namenskonventionen

- Die Funktionsbausteine mit dem Präfix MC_ ("Motion Control") sind konform zur PLCopen-Spezifikation. Sie entsprechen damit einem weltweit gültigen Standard zur Programmierung von Anwendungen für Motion Control.
- Die Funktionsbausteine ohne Präfix sind herstellerspezifisch (Schneider Electric), folgen aber den allgemeinen Regeln von PLCopen.

Einfache Anwendung

- Die Funktionsbausteine werden auf dieselbe Art und Weise verwendet.
- Die Funktionsbausteine entsprechen dem PLCopen-Zustandsdiagramm.
- Die Funktionsbausteine verfügen über eine Visualisierung, die sich einfach in eine Anwendung integrieren lässt.

Kategorisierung der Funktionsbausteine

- Single axis: Diese Funktionsbausteine werden für Bewegungen oder Funktionen einer unabhängigen Achse verwendet.
- Administrative: Diese Funktionsbausteine werden für Konfigurationsaufgaben verwendet (zum Beispiel Lesen und Schreiben von Parametern, Wiederherstellen einer Gerätekongfiguration).
- Device Function: Diese Funktionsbausteine unterstützen bei der ersten Inbetriebnahme eines Antriebsverstärkers an einer Steuerung. Voraussetzung dafür ist, dass die Kommunikationsparameter Baudrate und Knotenadresse richtig eingestellt sind.

Antriebsverstärker vorbereiten

Bevor Sie über CANopen oder CANmotion auf den Antriebsverstärker zugreifen können, sind einige Einstellungen im Antriebsverstärker notwendig. Hierzu gehören unter anderem:

- Adresse und Baudrate
- Profile (CHCF) = Separate
- Reference 1 (Fr1) = CAN
- Control channel (Cd1) = CAN
- Control channel switching (CSS) = Cd1
- Reference switching (rFC) = C214

Beachten Sie auch die Hinweise im Produkthandbuch.

Wenn Ihnen die vorhandene Konfiguration unbekannt ist, kann es sinnvoll sein die Werkseinstellungen zurückzusetzen. Siehe "2.4.2.4 ResetParameters_ATV".

2.1 Liste der Funktionsbausteine

Kategorie Single axis

Kategorie	Unterkategorie	Funktionsbaustein	Typ	ATV31/ ATV312	ATV71	ATV32
Single axis						
	Initialisierung	"2.3.1.1 MC_Power_ATV"	PLCopen	X	X	X
	Betriebsart Jog	"2.3.2.1 MC_Jog_ATV"	PLCopen	X	X	X
	Betriebsart Speed Control	"2.3.3.1 Velocity-ControlAnalogInput_ATV"	Herstellerspezifisch	X	X	X
		"2.3.3.2 Velocity-ControlSelectAI_ATV"	Herstellerspezifisch	X	X	X
	Betriebsart Profile Velocity	"2.3.4.1 MC_MoveVelocity_ATV"	PLCopen	X	X	X
	Stoppen	"2.3.5.1 MC_Stop_ATV"	PLCopen	X	X	X

Kategorie Administrative

Unterkategorie	Funktionsbaustein	Typ	ATV31/ ATV312	ATV71	ATV32
Parameter lesen	"2.4.1.1 MC_ReadActualVelocity_ATV"	PLCopen	X	X	X
	"2.4.1.2 MC_ReadActualTorque_ATV"	PLCopen	X	X	X
	"2.4.1.3 MC_ReadStatus_ATV"	PLCopen	X	X	X
	"2.4.1.4 MC_ReadParameter_ATV"	PLCopen	X	X	X
	"2.4.1.5 GetSupplierVersion"	Herstellerspezifisch	X	X	X
Parameter schreiben	"2.4.2.1 MC_WriteParameter_ATV"	PLCopen	X	X	X
	"2.4.2.2 SetDriveRamp_ATV"	Herstellerspezifisch	X	X	X
	"2.4.2.3 SetFrequencyRange_ATV"	Herstellerspezifisch	X	X	X
	"2.4.2.4 ResetParameters_ATV"	Herstellerspezifisch	X	X	X
	"2.4.2.5 StoreParameters_ATV"	Herstellerspezifisch	X	X	X
Gerätekonfiguration speichern und wiederherstellen	"2.4.3.1 UploadDriveParameter_ATV"	Herstellerspezifisch	X	X	X
	"2.4.3.2 DownloadDriveParameter_ATV"	Herstellerspezifisch	X	X	X
Eingänge und Ausgänge	"2.4.4.1 ReadAnalogInput_ATV"	Herstellerspezifisch	X	X	X
	"2.4.4.2 MC_ReadDigitalInput_ATV"	PLCopen	X	X	X
	"2.4.4.3 MC_ReadDigitalOutput_ATV"	PLCopen	X	X	X
	"2.4.4.4 MC_WriteDigitalOutput_ATV"	PLCopen	X	X	X
Fehlerbehandlung	"2.4.5.1 MC_ReadAxisError_ATV"	PLCopen	X	X	X
	"2.4.5.2 MC_Reset_ATV"	PLCopen	X	X	X

Kategorie Device Function

Unterkategorie	Funktionsbaustein	Typ	ATV31/ ATV312	ATV71	ATV32
Startup	"2.5.1.1 Altivar_Startup"	Herstellerspezifisch	X	X	X
Control	"2.5.2.1 Altivar31_Control"	Herstellerspezifisch	X	-	-
	"2.5.2.2 Altivar71_Control"	Herstellerspezifisch	-	X	-
	"2.5.2.3 Altivar32_Control"	Herstellerspezifisch	-	-	X

2.2 Grundlegende Eingänge und Ausgänge

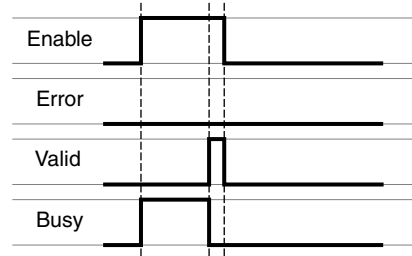
Eingang/Ausgang	Datentyp	Beschreibung
Axis	Axis_Ref_ATV	Name der Achse (Instanz), für die der Funktionsbaustein ausgeführt werden soll. Der Name muss in der SPS-Konfiguration deklariert sein. Den Namen der Achse finden Sie links in der Baumstruktur Ihrer Software.
Input	Input_Ref_ATV	Input ist ein spezieller Datentyp für digitale und analoge Eingänge. Der Datentyp entspricht dem Namen der Achse (Instanz), zu der die Eingänge gehören (ähnlich Axis). Bei Funktionsbausteinen, die speziell dem Auslesen analoger und digitaler Eingänge dienen, ersetzt Input den Eingang Axis.
Output	Output_Ref_ATV	Output ist ein spezieller Datentyp für digitale Ausgänge. Der Datentyp entspricht dem Namen der Achse (Instanz), zu der die Ausgänge gehören (ähnlich Axis). Bei Funktionsbausteinen, die speziell dem Schreiben und Auslesen digitaler Eingänge dienen, ersetzt Output den Eingang Axis.

Eingang	Datentyp	Beschreibung
Enable	BOOL	Wertebereich: TRUE, FALSE Initialwert: FALSE Der Eingang Enable führt einen Funktionsbaustein aus oder beendet diesen. (Ausnahme "2.3.1.1 MC_Power_ATV") FALSE: Der Funktionsbaustein wird beendet. Die Ausgänge Valid, Busy, CommandAborted und Error werden auf FALSE gesetzt. TRUE: Der Funktionsbaustein wird wiederholend ausgeführt.
Execute	BOOL	Wertebereich: TRUE, FALSE Initialwert: FALSE Der Eingang Execute führt einen Funktionsbaustein bei einer steigenden Flanke (FALSE->TRUE) aus. Wird ein Funktionsbaustein während seiner Ausführung ein weiteres mal durch eine steigende Flanke ausgeführt, so wird die aktuelle Ausführung abgebrochen und eine neue Ausführung gestartet. Die Ausführung wird beendet, sobald der Ausgang Busy = FALSE wird. FALSE mit gleichzeitig Busy = FALSE: Entweder Done, Error oder CommandAborted wird für einen Aufruf = TRUE gesetzt. TRUE mit gleichzeitig Busy = FALSE: Entweder Done, Error oder CommandAborted wird TRUE gesetzt und bleiben TRUE, bis Execute = FALSE gesetzt wird.

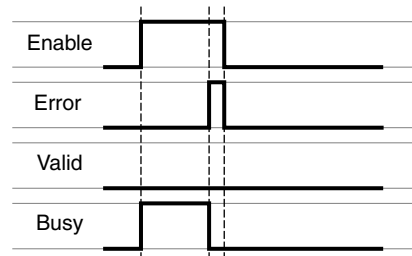
Ausgang	Datentyp	Beschreibung
Done	BOOL	Wertebereich: TRUE, FALSE Initialwert: FALSE FALSE: Die Ausführung wurde (bis jetzt) nicht ohne Fehler beendet. TRUE: Die Ausführung wurde ohne Fehler abgeschlossen.
Valid	BOOL	Wertebereich: TRUE, FALSE Initialwert: FALSE FALSE: Die Ausführung wurde (bis jetzt) nicht ohne Fehler beendet. Die Werte an den Ausgängen sind (bis jetzt) nicht gültig. TRUE: Die Ausführung wurde ohne Fehler abgeschlossen. Die Werte an den Ausgängen sind gültig und können weiterverarbeitet werden.
Busy	BOOL	Wertebereich: TRUE, FALSE Initialwert: FALSE FALSE: Die Ausführung des Funktionsbausteins wurde beendet. TRUE: Der Funktionsbaustein wird ausgeführt. HINWEIS: In der Betriebsart Profile Velocity bleibt der Ausgang TRUE, auch wenn die Zielgeschwindigkeit erreicht oder <code>Execute = FALSE</code> wird. Der Ausgang <code>Busy</code> wird FALSE, sobald ein anderer Funktionsbaustein, wie zum Beispiel <code>MC_Stop</code> , ausgeführt wird.
CommandAborted	BOOL	Wertebereich: TRUE, FALSE Initialwert: FALSE FALSE: Die Ausführung wurde (bis jetzt) nicht abgebrochen. TRUE: Die Ausführung wurde durch einen anderen Funktionsbaustein abgebrochen.
Error	BOOL	Wertebereich: TRUE, FALSE Initialwert: FALSE FALSE: Der Funktionsbaustein wurde (bis jetzt) ohne Fehler ausgeführt. TRUE: Bei der Ausführung des Funktionsbausteins trat ein Fehler auf.

2.2.1 Signalverhalten bei Funktionsbausteinen mit dem Eingang Enable

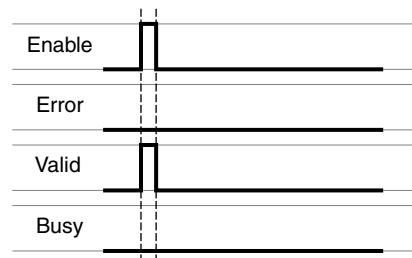
Beispiel 1 Einmalige Ausführung ohne Fehler (Ausführung erfordert mehr als einen Aufruf).



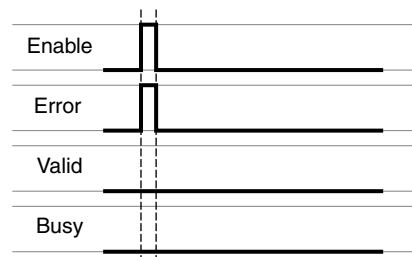
Beispiel 2 Einmalige Ausführung mit Fehler (Ausführung erfordert mehr als einen Aufruf).



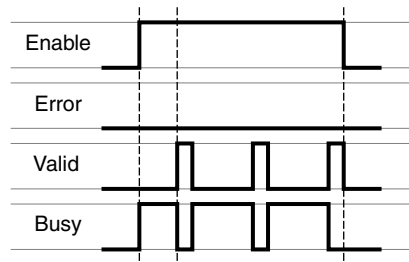
Beispiel 3 Einmalige Ausführung ohne Fehler (Ausführung erfordert nur einen Aufruf).



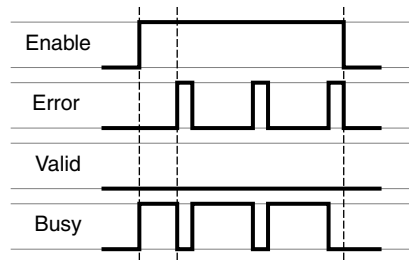
Beispiel 4 Einmalige Ausführung mit Fehler (Ausführung erfordert nur einen Aufruf).



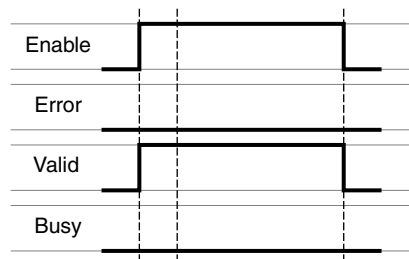
Beispiel 5 Wiederholte Ausführung ohne Fehler (Ausführung erfordert mehr als einen Aufruf).



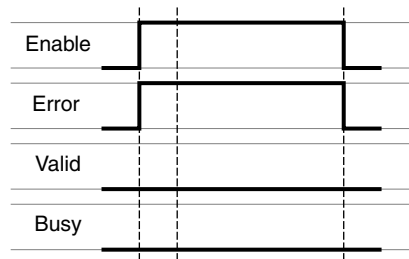
Beispiel 6 Wiederholte Ausführung mit Fehler (Ausführung erfordert mehr als einen Aufruf).



Beispiel 7 Wiederholte Ausführung ohne Fehler (Ausführung erfordert nur einen Aufruf).

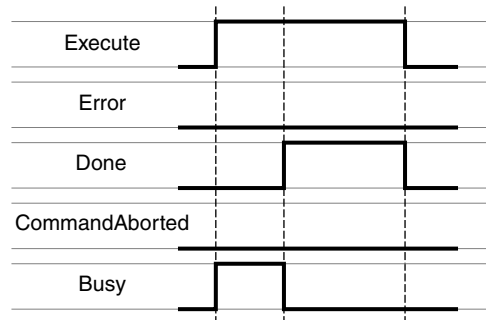


Beispiel 8 Wiederholte Ausführung mit Fehler (Ausführung erfordert nur einen Aufruf).

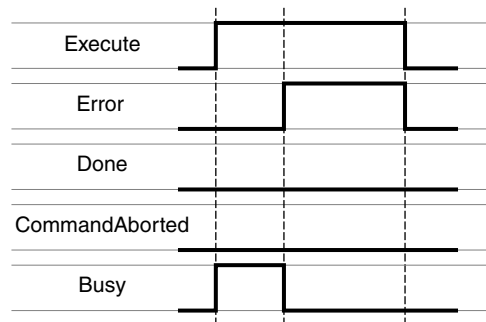


2.2.2 Signalverhalten bei Funktionsbausteinen mit dem Eingang **Execute**

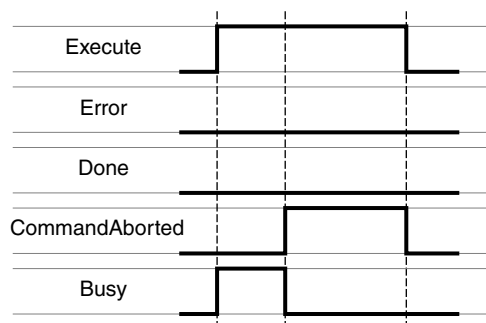
Beispiel 1 Ausführung ohne Fehler beendet.



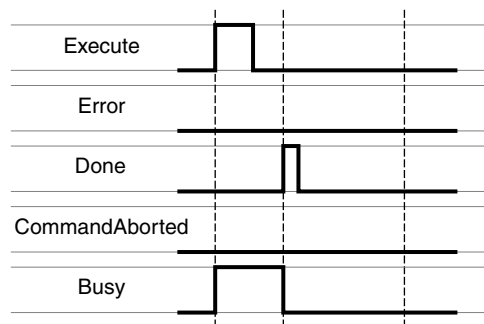
Beispiel 2 Ausführung mit Fehler beendet.



Beispiel 3 Abbruch der Ausführung durch Übernahme der Steuerung durch einen anderen Funktionsbaustein.



Beispiel 4 Ausführung ohne Fehler beendet, nachdem Execute während der Ausführung bereits auf FALSE gesetzt wurde.



2.3 Single axis

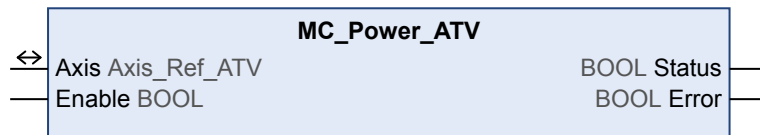
2.3.1 Initialisierung

Mit dem Funktionsbaustein zur Initialisierung wird die Endstufe des Antriebverstärkers aktiviert oder deaktiviert. Nur bei aktivierter Endstufe können weitere Funktionsbausteine benutzt werden.

2.3.1.1 MC_Power_ATV

Funktionsbeschreibung Der Funktionsbaustein aktiviert oder deaktiviert die Endstufe. Mit TRUE am Eingang `Enable` wird die Endstufe aktiviert. Sobald die Endstufe aktiviert ist, wird der Ausgang `Status` gesetzt. Mit FALSE am Eingang `Enable` wird die Endstufe deaktiviert. Sobald die Endstufe deaktiviert ist, wird der Ausgang `Status` zurückgesetzt. Wenn bei der Ausführung Fehler erkannt werden, wird der Ausgang `Error` gesetzt .

Grafische Darstellung



Kompatible Geräte ATV31/ATV312 und ATV71/ATV32

Eingänge/Ausgänge Folgende Tabelle zeigt die Ausgänge.

Ausgang	Datentyp	Beschreibung
Status	BOOL	Wertebereich: FALSE, TRUE Initialwert: FALSE FALSE: Endstufe ist deaktiviert. TRUE: Endstufe ist aktiviert.

"2.2 Grundlegende Eingänge und Ausgänge"

Hinweise Bei einem Node-Guarding-Fehler muss zuerst die Fehlermeldung im Fehlerspeicher durch den Funktionsbaustein "2.4.5.2 MC_Reset_ATV" zurückgesetzt werden, bevor die Endstufe wieder aktiviert werden kann.

- Ein Asynchronmotor hat bei Stillstand kein Drehmoment. Ein Aktivieren der Endstufe führt nicht zwangsweise zum Aufbau eines Drehmoments.
- Wenn der Eingang `Enable = TRUE` ist, wird bei Verlust der Spannungsversorgung einer der folgenden Fehler gemeldet:
 - 3120_h (Unterspannung)
 - ATV71/ATV32: FF34_h (PowerOnTimeout_ATV)
 - Bei Verlust der 24V-Spannungsversorgung: 8100_h (NodeguardError_ATV)
 - Der Ausgang `Status` wird FALSE und der Ausgang `Error` wird TRUE. Nach dem Wiederherstellen der Spannungsversorgung wird der Ausgang `Status` wieder TRUE.

2.3.2 Betriebsart Jog

In der Betriebsart Jog (Manuellfahrt) wird eine Bewegung von der Istposition in eine gewünschte Richtung ausgeführt. Die Geschwindigkeit kann eingestellt werden. Solange das Signal für die Richtung anliegt, wird eine Bewegung in die gewünschte Richtung ausgeführt.

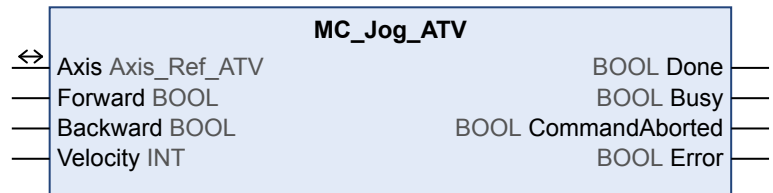
Wenn eine Bewegung in positive und negative Richtung gleichzeitig angefordert wird, erfolgt keine Bewegung.

2.3.2.1 MC_Jog_ATV

Funktionsbeschreibung

Der Funktionsbaustein startet die Betriebsart Jog (Manuellfahrt). Mit TRUE am Eingang `Forward` oder `Backward` wird die Manuellfahrt gestartet. Sind die Eingänge `Forward` und `Backward` = FALSE, wird die Betriebsart beendet und der Ausgang `Done` gesetzt. Sind die Eingänge `Forward` und `Backward` = TRUE, bleibt die Betriebsart aktiv, die Manuellfahrt wird gestoppt und der Ausgang `Busy` bleibt gesetzt.

Grafische Darstellung



Kompatible Geräte ATV31/ATV312 und ATV71/ATV32

Eingänge/Ausgänge Folgende Tabelle zeigt die Eingänge.

Eingang	Datentyp	Beschreibung
Forward	BOOL	<p>Wertebereich: FALSE, TRUE Initialwert: FALSE</p> <p>Forward = FALSE und Backward = FALSE: Bewegung wird beendet.</p> <p>Forward = TRUE und Backward = FALSE: Bewegung in positive Richtung wird gestartet, wenn Velocity >0. Bewegung in negative Richtung wird gestartet, wenn Velocity <0.</p> <p>Forward = FALSE und Backward = TRUE: Bewegung in negative Richtung wird gestartet, wenn Velocity >0. Bewegung wird gestoppt, wenn Velocity <0.</p> <p>Forward = TRUE und Backward = TRUE: Bewegung in aktuelle Richtung wird weiter ausgeführt. Wenn der Eingang Forward oder Backward FALSE wird, wird die Bewegung mit der dann gültigen Richtung und Geschwindigkeit fortgesetzt.</p>
Backward	BOOL	<p>Wertebereich: FALSE, TRUE Initialwert: FALSE</p> <p>Forward = FALSE und Backward = FALSE: Bewegung wird beendet.</p> <p>Forward = TRUE und Backward = FALSE: Bewegung in positive Richtung wird gestartet, wenn Velocity >0. Bewegung in negative Richtung wird gestartet, wenn Velocity <0.</p> <p>Forward = FALSE und Backward = TRUE: Bewegung in negative Richtung wird gestartet, wenn Velocity >0. Bewegung wird gestoppt, wenn Velocity <0.</p> <p>Forward = TRUE und Backward = TRUE: Bewegung in aktuelle Richtung wird weiter ausgeführt. Wenn der Eingang Forward oder Backward FALSE wird, wird die Bewegung mit der dann gültigen Richtung und Geschwindigkeit fortgesetzt.</p>
Velocity	INT	<p>Wertebereich: -5000 ... +5000 Initialwert: 0</p> <p>Gewünschte Zielgeschwindigkeit für Betriebsart. Einstellbar in 0,1 Hz Schritten.</p> <p>HINWEIS: Im Funktionsbaustein SetFrequencyRange_ATV wird der Wert für LowFrequency und HighFrequency eingestellt. Wenn der Wert für die Zielgeschwindigkeit velocity kleiner ist als der Wert für LowFrequency, wird die Bewegung mit dem Geschwindigkeitswert für LowFrequency ausgeführt. Es wird kein Fehler angezeigt. Wenn der Wert für die Zielgeschwindigkeit velocity größer ist als der Wert für HighFrequency, wird die Bewegung mit dem Geschwindigkeitswert für HighFrequency ausgeführt. Es wird kein Fehler angezeigt.</p>

"2.2 Grundlegende Eingänge und Ausgänge"

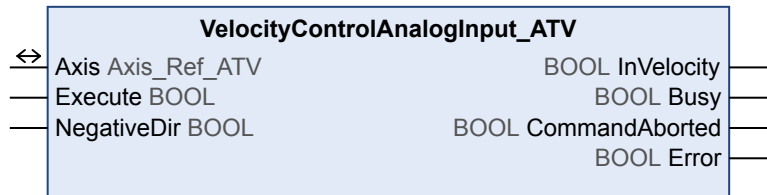
2.3.3 Betriebsart Speed Control

In der Betriebsart Speed Control können Sie eine gewünschte Sollgeschwindigkeit über einen Analogeingang vorgeben.

2.3.3.1 VelocityControlAnalogInput_ATV

Funktionsbeschreibung Der Funktionsbaustein übernimmt die Sollwerte des Analogeinganges, der mithilfe des Funktionsbausteins "2.3.3.2 VelocityControlSelectAI_ATV" ausgewählt wurde.

Grafische Darstellung



Kompatible Geräte ATV31/ATV312 und ATV71/ATV32

Eingänge/Ausgänge Folgende Tabelle zeigt die Eingänge.

Eingang	Datentyp	Beschreibung
NegativeDir	BOOL	Wertebereich: FALSE, TRUE Initialwert: FALSE FALSE: Drehrichtung im Uhrzeigersinn. TRUE: Drehrichtung gegen den Uhrzeigersinn.

Folgende Tabelle zeigt die Ausgänge.

Ausgang	Datentyp	Beschreibung
InVelocity	BOOL	Wertebereich: FALSE, TRUE Initialwert: FALSE FALSE: Die Geschwindigkeit entspricht nicht dem Sollwert. TRUE: Die Geschwindigkeit entspricht dem Sollwert.

"2.2 Grundlegende Eingänge und Ausgänge"

Hinweise Siehe auch "2.4.2.3 SetFrequencyRange_ATV" und "2.3.3.2 VelocityControlSelectAI_ATV".

Wenn Spannungspegel von -10V ... 10V verwendet werden, schaltet mit der Veränderung des Vorzeichens die Bewegungsrichtung (Drehrichtung) um. Liegt 0V an, kann es zu Sprüngen in der Bewegungsrichtung, dem Stillstand und der minimalen Frequenz kommen.

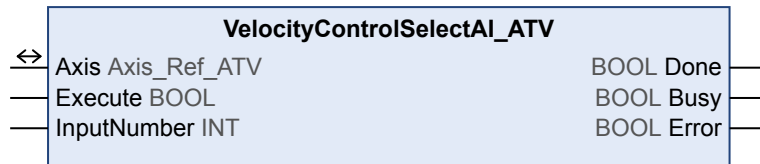
Hinweis zu ATV31: Wenn der analoge Stromeingang ausgewählt wurde (0mA ... 20mA), werden folgende Frequenzstufen verwendet:

- Unter 4 mA wird die niedrigste Frequenz verwendet.
- Bei 12 mA wird die mittlere Frequenz verwendet.
- Bei 20 mA wird die maximale Frequenz verwendet.

2.3.3.2 VelocityControlSelectAI_ATV

Funktionsbeschreibung Der Funktionsbaustein wählt den Analogeingang aus, über den ein Sollwert vorgegeben werden soll. Siehe auch "2.3.3.1 VelocityControlAnalogInput_ATV".

Grafische Darstellung



Kompatible Geräte ATV31/ATV312 und ATV71/ATV32

Eingänge/Ausgänge Folgende Tabelle zeigt die Eingänge.

Eingang	Datentyp	Beschreibung
InputNumber	INT	Wertebereich: 1 ... 16 Initialwert: 1 1: AI1 2: AI2 3: AI3 (bei ATV71 nur mit Erweiterungskarte) 4: AI4 (nur bei ATV71 mit Erweiterungskarte) 16: AIP (internes Potentiometer, nur bei ATV31/312)

"2.2 Grundlegende Eingänge und Ausgänge"

Hinweise Der Funktionsbaustein kann nur im Betriebszustand 3 Switch On Disabled (des Antriebsverstärkers) ausgeführt werden. Deaktivieren Sie hierzu die Endstufe mit dem Funktionsbaustein "2.3.1.1 MC_Power_ATV".

2.3.4 Betriebsart Profile Velocity

In der Betriebsart Profile Velocity können Sie eine gewünschte Zielgeschwindigkeit einstellen. Die Bewegung wird in der Betriebsart Profile Velocity mit dieser Zielgeschwindigkeit ausgeführt. Die Bewegung erfolgt solange, bis eine andere Zielgeschwindigkeit übergeben wird oder bis die Betriebsart abgebrochen wird.

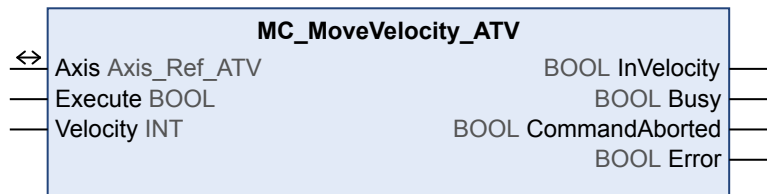
Der Übergänge zwischen zwei Zielgeschwindigkeiten erfolgt durch ein Bewegungsprofil. Das Bewegungsprofil wird vom Profilvergenerator im Antriebsverstärker anhand der Istgeschwindigkeit, der Zielgeschwindigkeit und von Beschleunigungs- und Verzögerungsrampen berechnet.

2.3.4.1 MC_MoveVelocity_ATV

Funktionsbeschreibung

Der Funktionsbaustein startet die Betriebsart Profile Velocity (Geschwindigkeitsprofil) mit der Geschwindigkeit `Velocity`. Wenn die Zielgeschwindigkeit erreicht ist, wird `InVelocity` gesetzt.

Grafische Darstellung



Kompatible Geräte ATV31/ATV312 und ATV71/ATV32

Eingänge/Ausgänge Folgende Tabelle zeigt die Eingänge.

Eingang	Datentyp	Beschreibung
Velocity	INT	Wertebereich: -5000 ... +5000 Initialwert: 0 Zielgeschwindigkeit in [0,1Hz] HINWEIS: Im Funktionsbaustein SetFrequencyRange_ATV wird der Wert für LowFrequency und HighFrequency eingestellt. Wenn der Wert für die Zielgeschwindigkeit <code>velocity</code> kleiner ist als der Wert für LowFrequency, wird die Geschwindigkeit mit dem Wert für LowFrequency ausgeführt. Es wird kein Fehler angezeigt. Wenn der Wert für die Zielgeschwindigkeit <code>velocity</code> größer ist als der Wert für HighFrequency, wird die Geschwindigkeit mit dem Wert für HighFrequency ausgeführt. Es wird kein Fehler angezeigt.

Folgende Tabelle zeigt die Ausgänge.

Ausgang	Datentyp	Beschreibung
InVelocity	BOOL	Wertebereich: FALSE, TRUE Initialwert: FALSE FALSE: Zielgeschwindigkeit wurde noch nicht erreicht. TRUE: Zielgeschwindigkeit wurde erreicht.

"2.2 Grundlegende Eingänge und Ausgänge"

Hinweise In der Betriebsart Profile Velocity (Geschwindigkeitsprofil) löst ein Positionsüberlauf keinen Fehler aus. Durch einen Positionsüberlauf geht der Nullpunkt verloren.

2.3.5 Stoppen

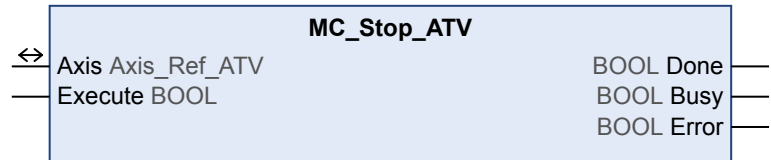
Jede Betriebsart kann durch Stoppen abgebrochen werden. Das Stoppen der Betriebsart wird nicht als Fehler gewertet.

2.3.5.1 MC_Stop_ATV

Funktionsbeschreibung

Der Funktionsbaustein dient zum Stoppen der aktuellen Bewegung. Die Betriebsart wird durch den Funktionsbaustein angehalten.

Grafische Darstellung



Kompatible Geräte

ATV31/ATV312 und ATV71/ATV32

Eingänge/Ausgänge

"2.2 Grundlegende Eingänge und Ausgänge"

Hinweise

Die Art der Verzögerung (Verzögerungsrampe, ungebremstes Auslaufen...) kann über den Parameter `Stt` eingestellt werden. Beachten Sie die Hinweise im Produkthandbuch.

- Die Verzögerungsrampe wird über den Funktionsbaustein "2.4.2.2 SetDriveRamp_ATV" eingestellt.
- Der Funktionsbaustein kann nur durch Deaktivieren der Endstufe über den Funktionsbaustein "2.3.1.1 MC_Power_ATV" unterbrochen werden.
- Solange der Eingang `Execute` = TRUE ist, kann kein anderer Funktionsbaustein gestartet werden, mit Ausnahme "2.3.1.1 MC_Power_ATV".

2.4 Administrative

2.4.1 Parameter lesen

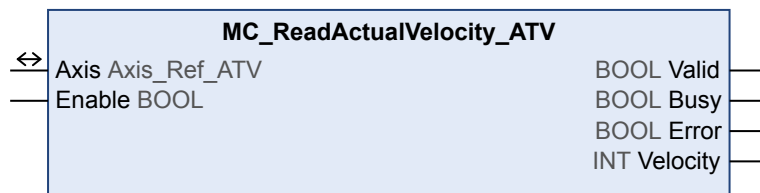
Mit den folgenden Funktionsbausteinen können Parameter des Antriebsverstärkers gelesen werden, zum Beispiel die Istposition oder die Istgeschwindigkeit.

Ein weiterer Funktionsbaustein ermöglicht einen Lesezugriff auf einen einzelnen Parameter des Antriebsverstärkers. Die Beschreibung der Parameter des Antriebsverstärkers finden Sie im Produkthandbuch.

2.4.1.1 MC_ReadActualVelocity_ATV

Funktionsbeschreibung Der Funktionsbaustein dient zum Auslesen der Istgeschwindigkeit des Motors.

Grafische Darstellung



Kompatible Geräte ATV31/ATV312 und ATV71/ATV32

Eingänge/Ausgänge Folgende Tabelle zeigt die Ausgänge.

Ausgang	Datentyp	Beschreibung
Velocity	INT	Wertebereich: -5000 ... +5000 Initialwert: Istgeschwindigkeit in min ⁻¹ .

"2.2 Grundlegende Eingänge und Ausgänge"

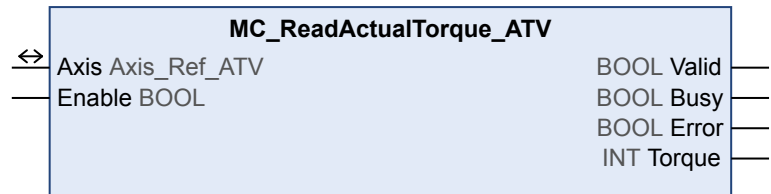
Hinweise Der Funktionsbaustein verwendet Service Data Objects (SDO) um den Parameter aus dem Gerät zu lesen. Aus diesem Grund wird dringend empfohlen den Eingang `Enable` nicht dauerhaft auf TRUE zu setzen. Dies kann zu einer Überlastung des Feldbusses führen. Es wird empfohlen den Funktionsbaustein zu deaktivieren sobald der Ausgang `Busy` = FALSE wird.

Der Wert wird in min⁻¹ angegeben. Beispiel: Bei einer Frequenz von 3 Hz und 2 Polpaaren ergibt sich eine Geschwindigkeit von 90 min⁻¹. (3 Hz * 60 s / 2 Polpaare = 90 min⁻¹). Beachten Sie, dass der Sollwert für die Geschwindigkeit ("2.3.4.1 MC_MoveVelocity_ATV") in 0,1 Hz Schritten angegeben wird (3 Hz -> Velocity = 30).

2.4.1.2 MC_ReadActualTorque_ATV

Funktionsbeschreibung Der Funktionsbaustein dient zum Auslesen des Istmoments des Motors.

Grafische Darstellung



Kompatible Geräte ATV31/ATV312 und ATV71/ATV32

Eingänge/Ausgänge Folgende Tabelle zeigt die Ausgänge.

Ausgang	Datentyp	Beschreibung
Torque	INT	Wertebereich: -3276,7 ... 3276,7 Initialwert: Istmoment des Motors in 0,1% Schritten.

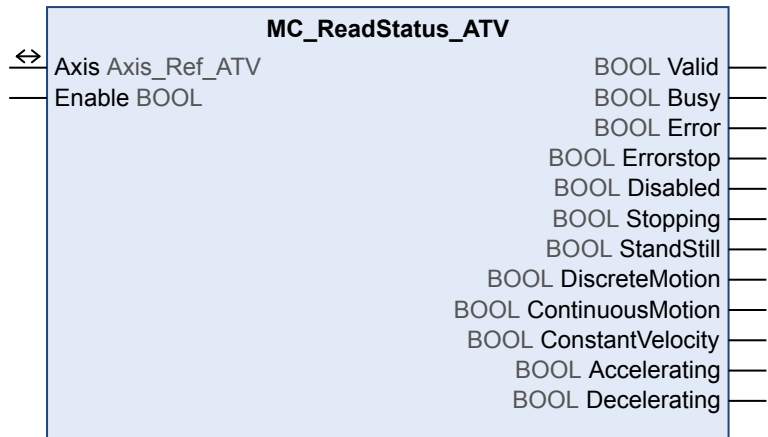
"2.2 Grundlegende Eingänge und Ausgänge"

Hinweise Der Funktionsbaustein verwendet Service Data Objects (SDO) um den Parameter aus dem Gerät zu lesen. Aus diesem Grund wird dringend empfohlen den Eingang `Enable` nicht dauerhaft auf TRUE zu setzen. Dies kann zu einer Überlastung des Feldbusses führen. Es wird empfohlen den Funktionsbaustein zu deaktivieren sobald der Ausgang `Busy = FALSE` wird.

2.4.1.3 MC_ReadStatus_ATV

Funktionsbeschreibung Der Funktionsbaustein dient zur Ausgabe des aktuellen Status des Gerätes.

Grafische Darstellung



Kompatible Geräte ATV31/ATV312 und ATV71/ATV32

Eingänge/Ausgänge Folgende Tabelle zeigt die Ausgänge.

Ausgang	Datentyp	Beschreibung
ErrorStop	BOOL	Wertebereich: FALSE, TRUE Initialwert: FALSE TRUE: Die Bewegung wurde durch einen Fehler unterbrochen.
Disabled	BOOL	Wertebereich: FALSE, TRUE Initialwert: FALSE FALSE: Die Endstufe ist aktiviert TRUE: Die Endstufe ist deaktiviert
Stopping	BOOL	Wertebereich: FALSE, TRUE Initialwert: FALSE TRUE: Der Funktionsbaustein "2.3.5.1 MC_Stop_ATV" wird ausgeführt oder die Bewegung wird gerade angehalten.
StandStill	BOOL	Wertebereich: FALSE, TRUE Initialwert: FALSE TRUE: Die Bewegung wurde angehalten.
DiscreteMotion	BOOL	Wertebereich: FALSE, TRUE Initialwert: FALSE TRUE: Die Betriebsart Profile Position wurde gestartet.
ContinuousMotion	BOOL	Wertebereich: FALSE, TRUE Initialwert: FALSE TRUE: Die Betriebsart Profile Velocity wurde gestartet.
ConstantVelocity	BOOL	Wertebereich: FALSE, TRUE Initialwert: FALSE TRUE: Eine Bewegung wird mit konstanter Geschwindigkeit ausgeführt.
Accelerating	BOOL	Wertebereich: FALSE, TRUE Initialwert: FALSE TRUE: Der Motor wird beschleunigt.
Decelerating	BOOL	Wertebereich: FALSE, TRUE Initialwert: FALSE TRUE: Der Motor wird verzögert.

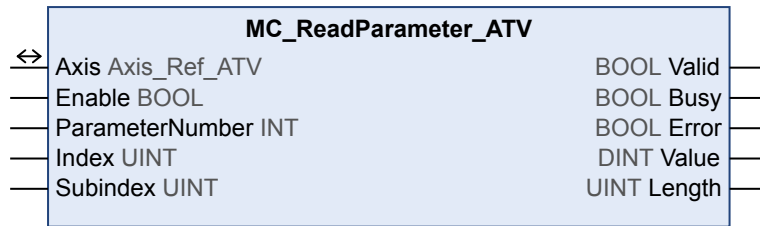
"2.2 Grundlegende Eingänge und Ausgänge"

Hinweise Der Antriebsverstärker ist zu jedem Zeitpunkt in einem der folgenden Zustände: StandStill, DiscreteMotion, ContinuousMotion, Stopping, Disabled oder ErrorStop. Der entsprechende Ausgang ist dann TRUE.

2.4.1.4 MC_ReadParameter_ATV

Funktionsbeschreibung Lesen eines Objekts aus der Geräteparameterliste.

Grafische Darstellung



Kompatible Geräte ATV31/ATV312 und ATV71/ATV32

Eingänge/Ausgänge Folgende Tabelle zeigt die Eingänge.

Eingang	Datentyp	Beschreibung
ParameterNumber	INT	Wertebereich: 0 ... 65535 Initialwert: 1000 Nummer des Parameters: 10: Istgeschwindigkeit. 11: Zielgeschwindigkeit. 1000: Auswahl über Index und Subindex.
Index	UINT	Wertebereich: 0 ... 65535 Initialwert: Index des zu lesenden Parameters. Nur gültig, wenn ParameterNumber = 1000. Eine Übersicht der Parameter finden Sie im Produkthandbuch.
Subindex	UINT	Wertebereich: 0 ... 255 Initialwert: Subindex des zu lesenden Parameters. Nur gültig, wenn ParameterNumber = 1000. Eine Übersicht der Parameter finden Sie im Produkthandbuch.

Folgende Tabelle zeigt die Ausgänge.

Ausgang	Datentyp	Beschreibung
Value	DINT	Wertebereich: -2147483648 ... +2147483647 Initialwert: 0 Wert des Parameters.
Length	UINT	Wertebereich: 1 ... 4 Initialwert: 4 Länge des Parameters in Byte.

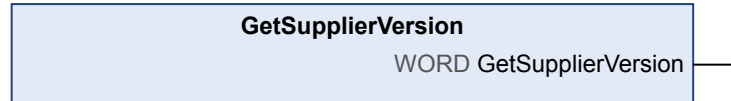
"2.2 Grundlegende Eingänge und Ausgänge"

Hinweise Der Funktionsbaustein verwendet Service Data Objects (SDO) um den Parameter aus dem Gerät zu lesen. Aus diesem Grund wird dringend empfohlen den Eingang `Enable` nicht dauerhaft auf TRUE zu setzen. Dies kann zu einer Überlastung des Feldbusses führen. Es wird empfohlen den Funktionsbaustein zu deaktivieren sobald der Eingang `Busy` = FALSE wird.

2.4.1.5 GetSupplierVersion

Funktionsbeschreibung Der Funktionsbaustein liefert die Version der verwendeten Bibliothek des Gerätes.

Grafische Darstellung



Kompatible Geräte ATV31/ATV312 und ATV71/ATV32

Eingänge/Ausgänge Folgende Tabelle zeigt die Ausgänge.

Ausgang	Datentyp	Beschreibung
GetSupplierVersion	WORD	Der Ausgang liefert die Versionsnummer der Bibliothek. Rechnen Sie den dezimalen Wert in Hex um. Beispiel: GetSupplierVersion = 12368 = 3050 _h = Version 3.0.5.0

2.4.2 Parameter schreiben

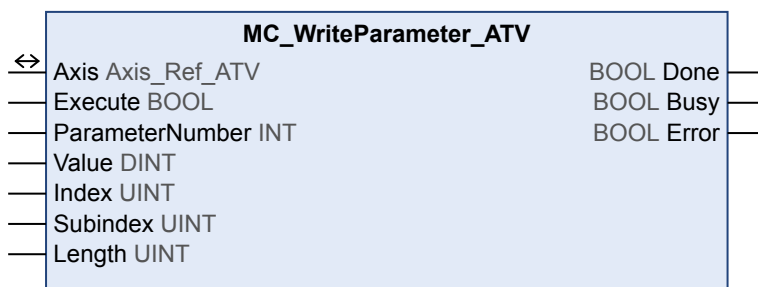
Mit den folgenden Funktionsbausteinen können Parameter des Antriebsverstärkers geschrieben werden, zum Beispiel Werte für die Beschleunigungsrampe und Verzögerungsrampe.

Ein weiterer Funktionsbaustein ermöglicht einen Schreibzugriff auf einen einzelnen Parameter des Antriebsverstärkers. Die Beschreibung der Parameter des Antriebsverstärkers finden Sie im Produkthandbuch.

2.4.2.1 MC_WriteParameter_ATV

Funktionsbeschreibung Über den Funktionsbaustein wird ein Wert in einen bestimmten Parameter geschrieben.

Grafische Darstellung



Kompatible Geräte ATV31/ATV312 und ATV71/ATV32

Eingänge/Ausgänge Folgende Tabelle zeigt die Eingänge.

Eingang	Datentyp	Beschreibung
ParameterNumber	INT	Wertebereich: 1000 Initialwert: 1000 Reserviert.
Value	DINT	Wertebereich: -2147483648 ... +2147483647 Initialwert: 0 Neuer Wert der auf den Parameter geschrieben werden soll.
Index	UINT	Wertebereich: 0 ... 65535 Initialwert: 0 Index des Parameters der beschrieben werden soll. Eine Liste der Parameter mit der entsprechenden CANopen-Adresse finden Sie im Produkthandbuch.
Subindex	UINT	Wertebereich: 0 ... 255 Initialwert: 0 Subindex des Parameters der beschrieben werden soll. Eine Liste der Parameter mit der entsprechenden CANopen-Adresse finden Sie im Produkthandbuch.
Length	UINT	Wertebereich: 0 ... 4 Initialwert: 0 Länge des Parameters der geschrieben werden soll in Bytes.

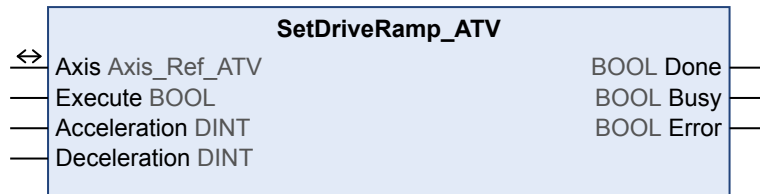
"2.2 Grundlegende Eingänge und Ausgänge"

Hinweise Werden die Eingänge `ParameterNumber`, `Index` oder `Subindex` während `Busy = TRUE` verändert, verwendet der Funktionsbaustein die vorherigen Werte. Mit der nächsten Ausführung des Funktionsbausteines werden die neuen Werte verwendet.

2.4.2.2 SetDriveRamp_ATV

Funktionsbeschreibung Der Funktionsbaustein konfiguriert die Beschleunigungsrampe und die Verzögerungsrampe des Gerätes.

Grafische Darstellung



Kompatible Geräte ATV31/ATV312 und ATV71/ATV32

Eingänge/Ausgänge Folgende Tabelle zeigt die Eingänge.

Eingang	Datentyp	Beschreibung
Acceleration	DINT	Wertebereich: 1 ... 9999 Initialwert: 30 Zeit für die Beschleunigungsrampe in 0,1 s. Beispiel: Bei einem Wert von 30 werden 3 Sekunden benötigt um von 0 auf Nennfrequenz des Motors zu beschleunigen. Der Wert muss mit dem verfügbaren Nennmoment des Motors erreicht werden können.
Deceleration	DINT	Wertebereich: 1 ... 9999 Initialwert: 30 Zeit für die Verzögerungsrampe in 0,1 s. Beispiel: Bei einem Wert von 30 werden 3 Sekunden benötigt um von Nennfrequenz des Motors auf 0 zu verzögern. Der Wert muss mit dem verfügbaren Nennmoment des Motors erreicht werden können.

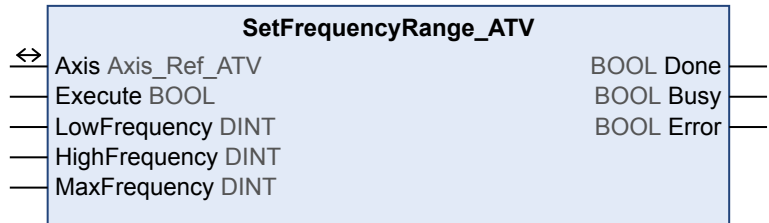
"2.2 Grundlegende Eingänge und Ausgänge"

Hinweise Bei großen externen Trägheitsmomenten oder bei hochdynamischen Anwendungen muss folgendes beachtet werden: Motoren speisen bei Verzögerung Energie zurück. Der DC-Bus kann eine begrenzte Energie in den internen Kondensatoren speichern. Durch den Anschluss zusätzlicher Kondensatoren am DC-Bus kann mehr Energie aufgenommen werden. Wird die Kapazität der Kondensatoren überschritten, muss die überschüssige Energie über interne oder externe Bremswiderstände abgeleitet werden. Wird die Energie nicht abgeleitet, schaltet eine Überspannungsüberwachung die Endstufe ab. Durch das Zuschalten eines Bremswiderstands mit entsprechender Ansteuerung kann eine Spannungsüberhöhung begrenzt werden. Dabei wird beim Verzögern die Rückspeisung in Wärmeenergie umgewandelt.

2.4.2.3 SetFrequencyRange_ATV

Funktionsbeschreibung Der Funktionsbaustein konfiguriert die Frequenzbereiche des Gerätes für die Funktionsbausteine MC_MoveVelocity und MC_Jog. Wird die Frequenz (Drehzahl) von LowFrequency unterschritten, verwendet das Gerät die Frequenz, die in LowFrequency festgelegt wurde, ohne eine Fehlermeldung auszugeben. Wird die Frequenz von HighFrequency überschritten, verwendet das Gerät die Frequenz, die in HighFrequency festgelegt wurde, ohne eine Fehlermeldung auszugeben.

Grafische Darstellung



Kompatible Geräte ATV31/ATV312 und ATV71/ATV32

Eingänge/Ausgänge Folgende Tabelle zeigt die Eingänge.

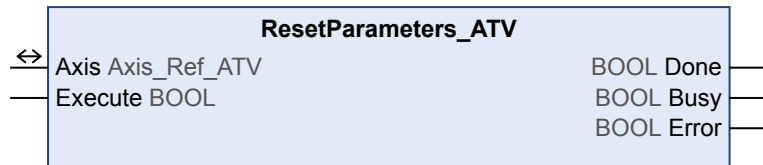
Eingang	Datentyp	Beschreibung
LowFrequency	DINT	Wertebereich: 0 ... HighFrequency Initialwert: 0 Motorfrequenz bei minimalem Sollwert.
HighFrequency	DINT	Wertebereich: LowFrequency ... MaxFrequency Initialwert: 500 Motorfrequenz bei maximalem Sollwert.
MaxFrequency	DINT	Wertebereich: 100 ... 5000/10000 (siehe Produkthandbuch) Initialwert: 600 Maximal zulässige Motorfrequenz. Passen Sie den Wert dem Motor und den mechanischen Gegebenheiten an. Die maximale Frequenz ist abhängig von bestimmten Parametern. Beachten Sie die Hinweise im Produkthandbuch.

"2.2 Grundlegende Eingänge und Ausgänge"

2.4.2.4 ResetParameters_ATV

Funktionsbeschreibung Der Funktionsbaustein setzt alle Parameter auf die Werkseinstellungen zurück.

Grafische Darstellung



Kompatible Geräte ATV31/ATV312 und ATV71/ATV32

Eingänge/Ausgänge "2.2 Grundlegende Eingänge und Ausgänge"

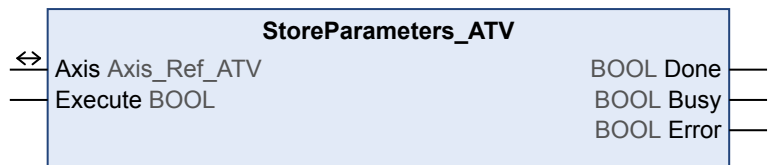
Hinweise Beachten Sie die Hinweise unter "Antriebsverstärker vorbereiten".

- Die neuen Einstellungen werden nicht in das EEPROM gespeichert. Verwenden Sie "2.4.2.5 StoreParameters_ATV" um die neuen Einstellungen im EEPROM zu speichern.

2.4.2.5 StoreParameters_ATV

Funktionsbeschreibung Der Funktionsbaustein speichert die Parameterwerte im EEPROM ab.

Grafische Darstellung



Kompatible Geräte ATV31/ATV312 und ATV71/ATV32

Eingänge/Ausgänge "2.2 Grundlegende Eingänge und Ausgänge"

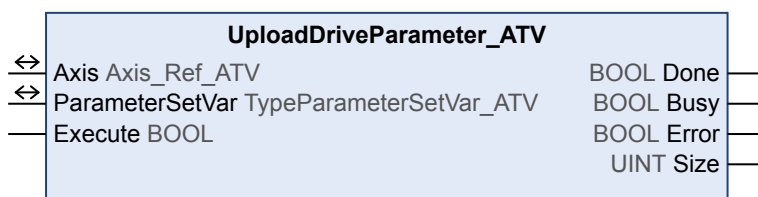
2.4.3 Gerätekonfiguration speichern und wiederherstellen

Die Gerätekonfiguration kann mit einem Funktionsbaustein vom Antriebsverstärker auf die Steuerung (upload) gespeichert werden. Mit einem weiteren Funktionsbaustein kann eine auf der Steuerung gespeicherte Gerätekonfiguration in einen Antriebsverstärker gespeichert werden (download).

2.4.3.1 UploadDriveParameter_ATV

Funktionsbeschreibung Der Funktionsbaustein liest die veränderbaren Parameterwerte aus dem Gerät aus. Siehe auch "2.4.3.2 DownloadDriveParameter_ATV".

Grafische Darstellung



Kompatible Geräte ATV31/ATV312 und ATV71/ATV32

Eingänge/Ausgänge Folgende Tabelle zeigt die Eingänge/Ausgänge.

Eingang/Ausgang	Datentyp	Beschreibung
ParameterSetVar	TypeParameterSetVar_ATV	Wertebereich: Initialwert: Liste der Geräteparameter. Vordefinierte Datenstruktur (Array of DINT).

Folgende Tabelle zeigt die Ausgänge.

Ausgang	Datentyp	Beschreibung
Size	UINT	Wertebereich: Initialwert: 0 Anzahl der gelesenen Parameter. Bei einem fehlerhaften Upload bleibt der Wert 0.

"2.2 Grundlegende Eingänge und Ausgänge"

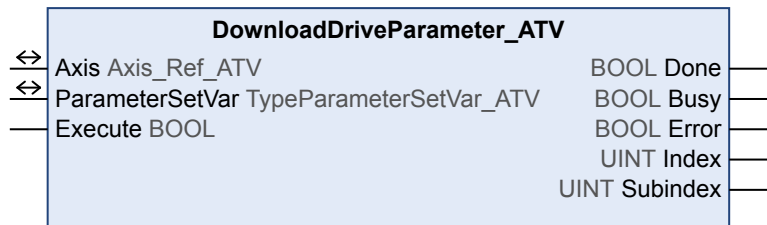
Hinweise

- Mit den beiden Funktionsbausteinen "2.4.3.2 DownloadDriveParameter_ATV" und "2.4.3.1 UploadDriveParameter_ATV" können die im Antriebsverstärker gespeicherte Parameter ohne Verwendung der Inbetriebnahmesoftware in einen baugleichen Antriebsverstärker gespeichert werden.

2.4.3.2 DownloadDriveParameter_ATV

Funktionsbeschreibung Der Funktionsbaustein schreibt die veränderbaren Parameter auf das Gerät. Vor Aufruf dieses Funktionsbausteins muss "2.4.3.1 UploadDriveParameter_ATV" ausgeführt werden. Ansonsten erscheint eine Fehlermeldung.

Grafische Darstellung



Kompatible Geräte ATV31/ATV312 und ATV71/ATV32

Eingänge/Ausgänge Folgende Tabelle zeigt die Eingänge/Ausgänge.

Eingang/Ausgang	Datentyp	Beschreibung
ParameterSetVar	TypeParameterSetVar_ATV	Wertebereich: Initialwert: Liste der Geräteparameter

Folgende Tabelle zeigt die Ausgänge.

Ausgang	Datentyp	Beschreibung
Index	UINT	Wertebereich: 0 ... 65535 Initialwert: Index des Parameters. Eine Übersicht der Parameter finden Sie im Produkthandbuch.
Subindex	UINT	Wertebereich: 0 ... 255 Initialwert: Subindex des Parameters. Eine Übersicht der Parameter finden Sie im Produkthandbuch.

"2.2 Grundlegende Eingänge und Ausgänge"

Hinweise

- Der Funktionsbaustein kann nur im Betriebszustand **3** Switch On Disabled (des Antriebsverstärkers) ausgeführt werden. Deaktivieren Sie hierzu die Endstufe mit dem Funktionsbaustein "2.3.1.1 MC_Power_ATV".
- Um die übertragenen Parameter dauerhaft zu speichern, müssen Sie über den Funktionsbaustein "2.4.2.5 StoreParameters_ATV" in das EEPROM gespeichert werden.
- Mit den beiden Funktionsbausteinen "2.4.3.2 DownloadDriveParameter_ATV" und "2.4.3.1 UploadDriveParameter_ATV" können die im Antriebsverstärker gespeicherte Parameter ohne Verwendung der Inbetriebnahmesoftware in einen baugleichen Antriebsverstärker gespeichert werden.

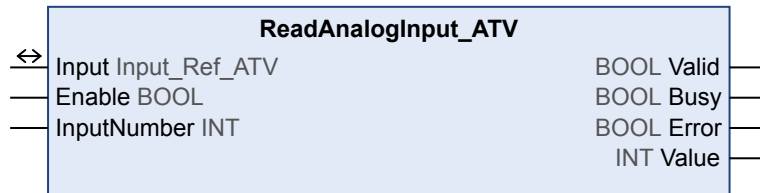
2.4.4 Eingänge und Ausgänge

Mit den folgenden Funktionsbausteinen können auf die digitalen und analogen Eingänge und Ausgänge jedes CAN-Knotens im System zugegriffen werden.

2.4.4.1 ReadAnalogInput_ATV

Funktionsbeschreibung Der Funktionsbaustein liefert den aktuellen Wert eines analogen Eingangs.

Grafische Darstellung



Kompatible Geräte ATV31/ATV312 und ATV71/ATV32

Eingänge/Ausgänge Folgende Tabelle zeigt die Eingänge.

Eingang	Datentyp	Beschreibung
InputNumber	INT	Wertebereich: 1 ... 4 Initialwert: 1 1: AI1 2: AI2 3: AI3 (bei ATV71 nur mit Erweiterungskarte) 4: AI4 (nur bei ATV71 mit Erweiterungskarte)

Folgende Tabelle zeigt die Ausgänge.

Ausgang	Datentyp	Beschreibung
Value	INT	Wertebereich: - Initialwert: 0 Entspricht der Eingangsspannung in mV oder dem Eingangsstrom in 0,001 mA Schritten am gewählten Analogeingang.

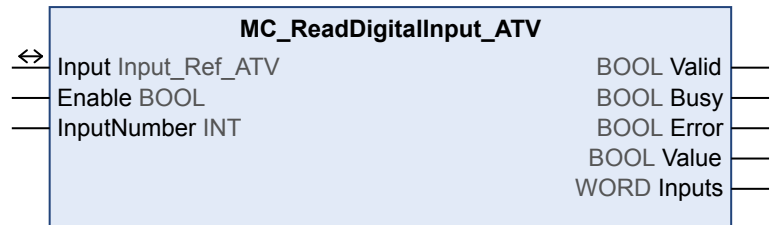
"2.2 Grundlegende Eingänge und Ausgänge"

- Hinweise*
- Die analogen Eingänge zwischen ATV31/ATV312 und ATV71/ATV32 unterscheiden sich. Weitere Informationen finden Sie im Produkthandbuch.

2.4.4.2 MC_ReadDigitalInput_ATV

Funktionsbeschreibung Lesen des aktuellen Zustands der digitalen Eingänge des Antriebsverstärkers.

Grafische Darstellung



Kompatible Geräte ATV31/ATV312 und ATV71/ATV32

Eingänge/Ausgänge Folgende Tabelle zeigt die Eingänge.

Eingang	Datentyp	Beschreibung
InputNumber	INT	<p>Wertebereich: 1 ... 14 (Produktabhängig) Initialwert: 1</p> <p>Nummer des Eingangs, der gelesen werden soll. Zuordnung der Eingänge des Antriebsverstärkers.</p> <p>1: IL1 2: IL2 3: IL3 4: IL4 5: IL5 6: IL6</p> <p>Eingänge der I/O Erweiterungskarte (ATV71): 7: IL7 8: IL8 9: IL9 10: IL10</p> <p>Eingänge der I/O Erweiterungskarte (ATV71): 11: IL11 12: IL12 13: IL13 14: IL14</p> <p>15: Reserviert. Wert = 0.</p>

Folgende Tabelle zeigt die Ausgänge.

Ausgang	Datentyp	Beschreibung
Value	BOOL	<p>Wertebereich: FALSE, TRUE Initialwert: FALSE</p> <p>FALSE: ausgewählter Eingang hat 0 V Pegel. TRUE: ausgewählter Eingang hat 24 V Pegel.</p>
Inputs	WORD	<p>Wertebereich: 0000_h ... 3FFF_h Initialwert: 0000_h</p> <p>Abbild der Eingänge als Bitmuster.</p> <p>Bit 0: IL1 Bit 1: IL2 Bit 2: IL3 Bit 3: IL4 Bit 4: IL5 Bit 5: IL6</p> <p>Eingänge der I/O Erweiterungskarte (ATV71): Bit 6: IL7 Bit 7: IL8 Bit 8: IL9 Bit 9: IL10</p> <p>Eingänge der I/O Erweiterung: Bit 10: IL11: Bit 11: IL12 Bit 12: IL13 Bit 13: IL14</p> <p>Bit 14 und Bit 15: Reserviert. Wert = 0.</p>

"2.2 Grundlegende Eingänge und Ausgänge"

Hinweise Eine Beschreibung der digitalen Eingänge finden Sie im Produkthandbuch.

2.4.4.3 MC_ReadDigitalOutput_ATV

Funktionsbeschreibung Der Funktionsbaustein dient zur Ausgabe des aktuellen Zustands der digitalen Ausgänge.

Grafische Darstellung



Kompatible Geräte ATV31/ATV312 und ATV71/ATV32

Eingänge/Ausgänge Folgende Tabelle zeigt die Eingänge.

Eingang	Datentyp	Beschreibung
OutputNumber	INT	<p>Wertebereich: 1 ...8 (Produktabhängig) Initialwert: 1</p> <p>Nummer des Ausgangs, der gelesen werden soll.</p> <p>ATV31/312/32: 1: Relay1 2: Relay2 3: LO</p> <p>ATV71: 1: Relay1 2: Relay2 3: Relay3 4: Relay4 5: LO1 6: LO2 7: LO3 8: LO4</p>

Folgende Tabelle zeigt die Ausgänge.

Ausgang	Datentyp	Beschreibung
Value	BOOL	<p>Wertebereich: FALSE, TRUE Initialwert: FALSE</p> <p>FALSE: ausgewählter Ausgang hat 0 V Pegel. TRUE: ausgewählter Ausgang hat 24 V Pegel.</p>
Outputs	WORD	<p>Wertebereich: 00h ...0Fh Initialwert: 00h</p> <p>Abbild der Ausgänge als Bitmuster.</p> <p>ATV31/312/32: Bit 0: Relay1 Bit 1: Relay2 Bit 2: LO</p> <p>ATV71: Bit 0: Relay1 Bit 1: Relay2 Bit 2: Relay3 Bit 3: Relay4 Bit 4: LO1 Bit 5: LO2 Bit 6: LO3 Bit 7: LO4</p> <p>Alle anderen Bits haben den Wert 0.</p>

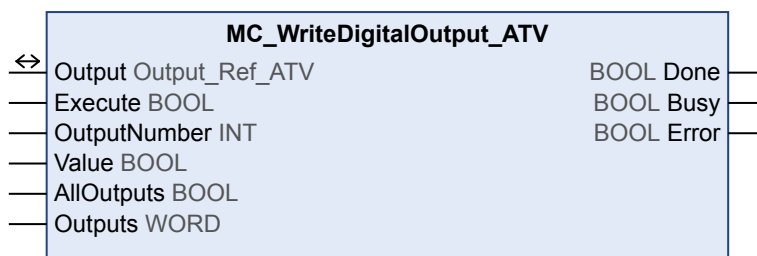
"2.2 Grundlegende Eingänge und Ausgänge"

Hinweise Eine Beschreibung der digitalen Ausgänge finden Sie im Produkthandbuch.

2.4.4.4 MC_WriteDigitalOutput_ATV

Funktionsbeschreibung Der Funktionsbaustein schreibt Werte auf die digitalen Ausgänge.

Grafische Darstellung



Kompatible Geräte ATV31/ATV312 und ATV71/ATV32

Eingänge/Ausgänge Folgende Tabelle zeigt die Eingänge.

Eingang	Datentyp	Beschreibung
OutputNumber	INT	Signalausgang, der beschrieben werden soll. ATV31/312/32: 1: Relay1 2: Relay2 3: LO ATV71: 1: Relay1 2: Relay2 3: Relay3 4: Relay4 5: LO1 6: LO2 7: LO3 8: LO4.
Value	BOOL	Wertebereich: FALSE, TRUE Initialwert: FALSE FALSE: Der gewählte Signalausgang wird mit 0V beschrieben. TRUE: Der gewählte Signalausgang wird mit 24V beschrieben.
AllOutputs	BOOL	Wertebereich: FALSE, TRUE Initialwert: FALSE: Der zu beschreibende Signalausgang wird über den Eingang OutputNumber eingestellt. TRUE: Die zu beschreibenden Signalausgänge werden über den Eingang Output eingestellt.
Outputs	WORD	Wertebereich: 0000 _h ... 0003 _h Initialwert: 0 Der Eingang definiert, welche Signalausgänge auf den unter dem Eingang Value angegebenen Wert geschrieben werden sollen. 0000 0000 0000 0000 ₂ (0000 _h) = Signalausgang/Relais 1 0000 0000 0000 0010 ₂ (0002 _h) = Signalausgang/Relais 2 0000 0000 0000 0011 ₂ (0003 _h) = Signalausgang/Relais 1 und Signalausgang/Relais 2

"2.2 Grundlegende Eingänge und Ausgänge"

2.4.5 Fehlerbehandlung

Zur Fehlerbehandlung hat jeder Funktionsbaustein einen Ausgang `Error`, der beim Auftreten eines synchronen oder asynchronen Fehler gesetzt wird.

Zur Analyse der Fehlerursache wird der Funktionsbaustein `MC_ReadAxisError_xxx` aufgerufen. Der Funktionsbaustein zeigt die gespeicherte Fehlerinformation.

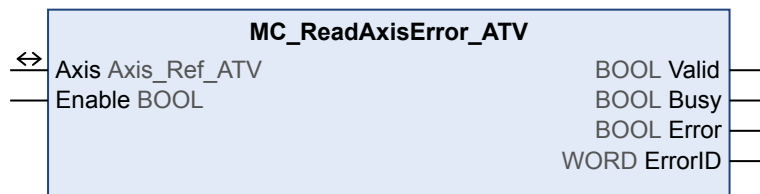
Mit dem Funktionsbaustein `MC_Reset_xxx` wird die eingetragene Fehlerinformation wieder gelöscht. Zukünftige Fehlerinformationen können wieder eingetragen werden.

Wenn ein weiterer Fehler auftritt, werden die Fehlerinformationen nur eingetragen, wenn noch keine Fehlerinformation eingetragen ist. Sind noch Fehlerinformationen eines vorangegangenen Fehler eingetragen, wird die neue Fehlermeldung ignoriert.

2.4.5.1 MC_ReadAxisError_ATV

Funktionsbeschreibung Der Funktionsbaustein dient zur Fehlerabfrage des Gerätes.

Grafische Darstellung



Kompatible Geräte ATV31/ATV312 und ATV71/ATV32

Eingänge/Ausgänge Folgende Tabelle zeigt die Ausgänge.

Ausgang	Datentyp	Beschreibung
ErrorID	WORD	Wertebereich: 0000 _h ... FFFF _h Initialwert: 0000 _h 0: kein Fehler gespeichert. >0: gespeicherte Fehlernummer. Eine Übersicht der Fehlernummern finden Sie im Produkthandbuch.

"2.2 Grundlegende Eingänge und Ausgänge"

Tabelle der Fehlernummern Folgende Tabelle zeigt die Fehlernummern der Bibliothek. Die Fehlernummern des Antriebsverstärkers finden sie im Produkthandbuch.

ErrorID Hexadezimal	ErrorID Dezimal	HMI	Beschreibung
1000 _h	4096	$\mathcal{L}rF$ $\alpha L F$ $5\alpha F$	Kondensator Fehler Überlast Motor Zu hohe Drehzahl
2310 _h	8976	$\alpha L F$	Überstrom Motor
2320 _h	8992	$5 L F$	Kurzschluss Motor
2330 _h	9008	SCF	Kurzschluss Motorphasen (Erdschluss)
2340 _h	9024	SCF	Kurzschluss Motorphasen (Phase-Phase)
3110 _h	12560	$\alpha 5 F$	Überspannung Netzversorgung
3120 _h	12576	$\underline{\alpha} 5 F$	Unterspannung Netzversorgung
3130 _h	12592	PhF	Netzphasenfehler
3310 _h	13072	$\alpha b F$ $\alpha P F$	Überspannung DC-Bus Fehler Motorphase
4210 _h	16912	$\alpha h F$	Übertemperatur
5520 _h	21792	EEF	EEPROM Fehler
6100 _h	24832	$, n F$	Internes Ereignis
6300 _h	25344	$\mathcal{L} L F$	Parameter außerhalb zulässigem Wertebereich
7300 _h	29440	$L F F$	Fehler an AI3
7510 _h	29968	$5 L F$	Modbus Kommunikationsfehler
8100 _h	33024	$\mathcal{L} \alpha F$	CANopen Kommunikationsfehler, Heartbeat oder Life Guard Fehler
9000 _h	36864	EPF	Externer Fehler
A309 _h	41737		Antrieb nicht im Betriebszustand 6 Operation Enabled
FE00 _h	65024	$\mathcal{L} n F$	Fehler beim Autotuning
FE01 _h	65025	$b L F$	Fehler Bremsenansteuerung
FF00 _h	65280	-	Toggle-Bit unverändert
FF01 _h	65281	-	SDO Zeitüberschreitung
FF02 _h	65282	-	server / client - command specifier ungültig oder nicht bekannt
FF03 _h	65283	-	Ungültige Blockgröße (nur im Block Mode)
FF04 _h	65284	-	Ungültige Sequenz-Nummer (nur im Block Mode)
FF05 _h	65285	-	CRC-Fehler (nur im Block Mode)
FF06 _h	65286	-	Kein Speicher frei
FF07 _h	65287	-	Kein Zugriff auf Objekt möglich
FF08 _h	65288	-	Kein Lesezugriff, da nur Schreib-Objekt (wo)
FF09 _h	65289	-	Kein Schreibzugriff, da Lese-Objekt (ro)
FF0A _h	65290	-	Objekt nicht im Objektverzeichnis vorhanden
FF0B _h	65291	-	Objekt unterstützt kein PDO-Mapping
FF0C _h	65292	-	Anzahl oder Länge der Objekte überschreiten die Byte-Länge des PDOs
FF0D _h	65293	-	Parameter sind nicht kompatibel
FF0E _h	65294	-	Gerät erkennt interne Inkompatibilität
FF0F _h	65295	-	Hardware-Fehler, Zugriff verweigert
FF10 _h	65296	-	Datentyp und Parameterlänge stimmen nicht überein
FF11 _h	65297	-	Datentyp stimmt nicht überein, Parameter zu lang

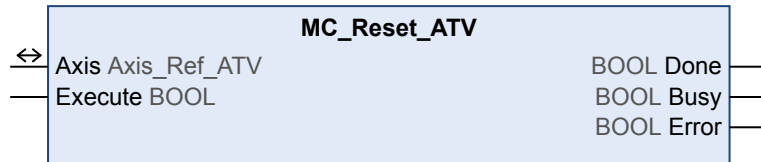
ErrorID Hexadezimal	ErrorID Dezimal	HMI	Beschreibung
FF12 _h	65298	-	Datentyp stimmt nicht überein, Parameter zu kurz
FF13 _h	65299	-	Subindex nicht unterstützt
FF14 _h	65300	-	Wertebereich des Parameters zu groß (nur für Schreibzugriff relevant)
FF15 _h	65301	-	Parameterwerte zu groß
FF16 _h	65302	-	Parameterwerte zu klein
FF17 _h	65303	-	Oberer Wert ist kleiner als unterer Wert
FF18 _h	65304	-	Allgemeiner Fehler
FF19 _h	65305	-	Daten können weder zur Anwendung übertragen noch gespeichert werden
FF1A _h	65306	-	Lokaler Zugriffskanal wird verwendet, Daten können nicht übertragen oder gespeichert werden
FF1B _h	65307	-	Gerätestatus sperrt das Übertragen und Speichern der Daten
FF1C _h	65308	-	Objektverzeichnis entweder nicht vorhanden oder nicht generierbar (zum Beispiel wenn Datenfehler bei Generierung aus Datei auftritt)
FF1D _h	65309	-	Reserviert
FF1E _h	65310	-	Reserviert
FF1F _h	65311	-	Reserviert
FF20 _h	65312	-	Unbekannter Status
FF21 _h	65313	-	Eingangsvariable wurde verändert bevor die Antwort empfangen wurde ("2.4.1.4 MC_ReadParameter_ATV", "2.4.2.1 MC_WriteParameter_ATV")
FF22 _h	65314	-	Versuch einen nicht unterbrechbaren Funktionsbaustein zu unterbrechen ("2.3.1.1 MC_Power_ATV", "2.3.5.1 MC_Stop_ATV")
FF34 _h	65332	-	Endstufe wechselt nicht in den Betriebszustand 6 Operation Enabled
FF37 _h	65335	-	Endstufe ist nicht im Betriebszustand 6 Operation Enabled
FF38 _h	65336	-	Parameterliste wurde noch nicht über "2.4.3.1 UploadDriveParameter_ATV" vom Gerät gelesen
FF39 _h	65337	-	Parameterliste passt nicht zum Gerät
FF3A _h	65338	-	Antriebsverstärker ist im Zustand PreOperational
FF3B _h	65339	-	Antriebsverstärker ist nicht im Betriebszustand 3 Switch On Disabled
FF3C _h	65340	-	STO "Safe Torque Off" (Power Removal) aktiv
FF3D _h	65341	-	Antriebsverstärker nicht kompatibel
FF3E _h	65342	-	Fehler im Mapping
FF50 _h	65360	-	Initialisierungsfehler des Funktionsbausteins "2.5.1.1 Altivar_Startup".
FF51 _h	65361	-	Der Funktionsbaustein "2.5.1.1 Altivar_Startup" kann nicht über die Anwendung gesteuert werden da <code>i_iControlMode = 1</code> .
FF52 _h	65362	-	Der Funktionsbaustein "2.5.1.1 Altivar_Startup" kann nicht über die Visualisierung gesteuert werden da <code>i_iControlMode = 0</code> .

ErrorID Hexadezimal	ErrorID Dezimal	HMI	Beschreibung
FF53 _h	65363	-	Der Wert am Eingang <code>i_iControlMode</code> ist außerhalb des gültigen Wertebereichs.
FF54 _h	65364	-	Der Wert am Eingang <code>iq_iCmd</code> ist außerhalb des gültigen Wertebereichs.

2.4.5.2 MC_Reset_ATV

Funktionsbeschreibung Der Funktionsbaustein dient zur Fehlerquittierung. Der Fehlerspeicher wird gelöscht und damit frei für zukünftige Fehlermeldungen. Wurde die Endstufe durch die automatische Fehlerreaktion deaktiviert, kann sie wieder aktiviert werden, wenn die Fehlerursache zum Zeitpunkt der Fehlerquittierung beseitigt ist.

Grafische Darstellung



Kompatible Geräte ATV31/ATV312 und ATV71/ATV32

Eingänge/Ausgänge "2.2 Grundlegende Eingänge und Ausgänge"

2.5 Device Function

2.5.1 Startup

Die Funktionsbausteine "Startup" unterstützen bei der Inbetriebnahme eines Antriebssystems an einer Steuerung. Um die Funktionsbausteine benutzen zu können, müssen die Kommunikationsparameter Baudrate und Knotenadresse im Antriebsverstärker und bei der Steuerung bereits eingestellt sein. Eine gleichzeitige Verwendung von Funktionsbaustein und Visualisierung ist nicht möglich.

Die Funktionsbausteine "Startup" haben mit den Visualisierungselementen folgende Funktionen:

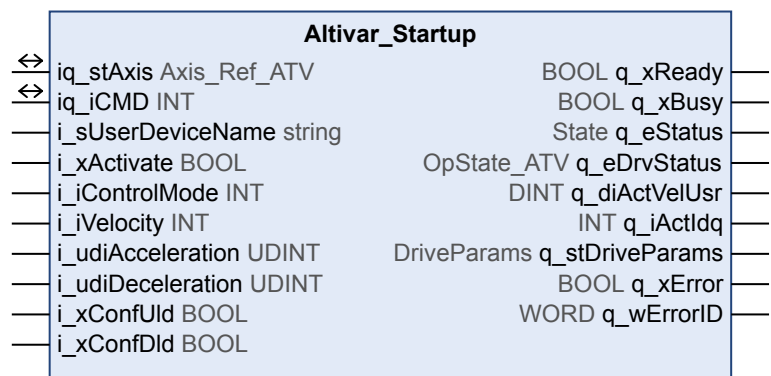
- Einschalten des Antriebssystems.
- Statusanzeige des Antriebssystems.
- Schneller Zugriff auf gängige Parameter.
- Zugriff auf einzelne Parameter durch Adressierung über Index und Subindex.
- Übertragen einer Geräteparameterliste vom Antriebsverstärker in die Steuerung und von der Steuerung auf einen Antriebsverstärker (Upload und Download).
- Ausführen der Betriebsart Jog.
- Ausführen der Betriebsart Profile Velocity (Bewegung mit konstanter Geschwindigkeit).
- Anzeigen und quittieren von Fehlermeldungen.

2.5.1.1 Altivar_Startup

Funktionsbeschreibung

Dieser Funktionsbaustein unterstützt Sie bei der ersten Inbetriebnahme eines Frequenzumrichters. Zu dem Funktionsbaustein gehören zwei Visualisierungen, um die Bedienung des Funktionsbausteins zu vereinfachen. Eine gleichzeitige Verwendung von Funktionsbaustein und Visualisierung ist nicht möglich.

Grafische Darstellung



Kompatible Geräte ATV31/ATV312 und ATV71/ATV32

Eingänge/Ausgänge Folgende Tabelle zeigt die Eingänge/Ausgänge.

Eingang/Ausgang	Datentyp	Beschreibung
iq_stAxis	Axis_Ref_ATV	Wertebereich: Initialwert: Entspricht dem Eingang/Ausgang Axis. Siehe "2.2 Grundlegende Eingänge und Ausgänge".
iq_iCMD	INT	Wertebereich: Initialwert: Befehle: -1: command is active 0: no ongoing command 1: ENABLE (Endstufe aktivieren) 2: DISABLE (Endstufe deaktivieren) 3: Reset 4: Stop 9: MoveVel Die auszuführende Funktion wird als Befehl von der Anwendung geschrieben und beim Bearbeiten vom Funktionsbaustein überschrieben. Bedingung: Der Eingang ist nur wirksam, wenn der Wert von ControlMode =1 ist. Um die gewählte Funktion zu starten, muss der Wert im Parameter CMD einmal geschrieben werden. Sobald der Befehl ausgeführt wird, wird der Wert mit -1 überschrieben. Wenn der Befehl beendet ist, wird der Wert mit 0 überschrieben.

Folgende Tabelle zeigt die Eingänge.

Eingang	Datentyp	Beschreibung
i_sUserDeviceName	string	Wertebereich: Initialwert: Name der Achse. Der Name wird vom Anwender definiert. Wenn kein Name eingetragen wird, wird die Knoten-ID angezeigt.
i_xActivate	BOOL	Wertebereich: FALSE, TRUE Initialwert: Der gewählte ControlMode wird mit einer steigenden Flanke aktiviert. Wenn alle Bedingungen für den gewählten ControlMode erfüllt sind, wird der gewählte ControlMode ausgeführt. Wenn nicht alle Bedingungen erfüllt sind, wird die Auswahl mit einer Fehlermeldung abgebrochen.
i_iControlMode	INT	Wertebereich: Initialwert: ControlMode = 0: Die Funktionen werden über Visualisierung gesteuert. ControlMode = 1: Die Funktionen werden über die Anwendung gesteuert. Die Visualisierung ist deaktiviert.
i_iVelocity	INT	Wertebereich: Initialwert: Zielgeschwindigkeit in usr
i_udiAcceleration	UDINT	Wertebereich: Initialwert: 30 Beschleunigungsrampe in [usr]
i_udiDeceleration	UDINT	Wertebereich: Initialwert: 30 Verzögerungsrampe in [usr]
i_xConfUld	BOOL	Wertebereich: FALSE, TRUE Initialwert: Mit steigender Flanke wird ein Upload (Parameter vom Gerät auf Controller speichern) gestartet.
i_xConfDld	BOOL	Wertebereich: FALSE, TRUE Initialwert: Mit steigender Flanke wird ein Download (gespeicherte Parameter vom Controller auf Gerät) gestartet.

Folgende Tabelle zeigt die Ausgänge.

Ausgang	Datentyp	Beschreibung
q_xReady	BOOL	Wertebereich: FALSE, TRUE Initialwert: Funktionsbaustein ist aktiviert und funktionsbereit.
q_xBusy	BOOL	Wertebereich: FALSE, TRUE Initialwert: Eine Funktion wird über den Funktionsbaustein ausgeführt. Wenn eine neue Funktion gestartet wird, wird die aktuell ausgeführte Funktion abgebrochen.
q_eStatus	State	Wertebereich: Initialwert: Zustand entsprechend PLCopen-Zustandsdiagramm: 0: Undefined 1: Errorstop 2: Disabled 3: Stopping 4: StandStill 5: DiscreteMotion 6: ContinuousMotion 7: SynchronizedMotion 8: Homing
q_eDrvStatus	OpState_ATV	Wertebereich: Initialwert: Betriebszustand des Antriebsverstärkers: 1: init 2: nrdy 3: dis 4: rdy 5: son 6: run 7: stop 8: flt
q_diActVelUsr	DINT	Wertebereich: Initialwert: Istgeschwindigkeit in [usr]
q_iActIdq	INT	Wertebereich: Initialwert: Iststrom in [A_{rms}]
q_stDriveParams	DriveParams	Wertebereich: Initialwert: Datenstruktur, besteht aus STRING: Geräteidentifikation REAL: Firmware-Version des Geräts
q_xError	BOOL	Wertebereich: FALSE, TRUE Initialwert: FALSE: es wurde kein Fehler erkannt. TRUE: Fehler wurde erkannt.
q_wErrorID	WORD	Wertebereich: Initialwert: Fehlernummer. Siehe "Tabelle der Fehlernummern"

Hinweise

⚠️ WARNUNG

UNBEABSICHTIGTES VERHALTEN DURCH INKONSISTENTE BEFEHLE

Wenn Sie diesen Funktionsbaustein aktiviert haben, führt die gleichzeitige Verwendung anderer Funktionsbausteine der Bibliothek zu unbeabsichtigtem Verhalten.

- Aktivieren Sie diesen Funktionsbaustein nur, wenn alle anderen Funktionsbausteine der Bibliothek inaktiv sind.
- Deaktivieren Sie diesen Funktionsbaustein, bevor Sie andere Funktionsbausteine der Bibliothek aktivieren.

Nichtbeachtung dieser Vorkehrungen kann zu Tod, schwerwiegenden Verletzungen oder Materialschäden führen.

Mit der Visualisierung Altivar_Startup_TuneVis kann auf viele Parameter des Antriebsverstärkers direkt zugegriffen werden. Die Adressierung der Parameter erfolgt über Index und Subindex. Parameterwerte können gelesen und geschrieben werden. Ein Abbild von fest definierten Parameterwerten kann vom Antriebsverstärker auf den Controller gespeichert werden. Die gespeicherten Parameterwerte können als Einheit vom Controller auf den Antriebsverstärker übertragen werden. "Conf Download" entspricht dem Funktionsbaustein

"2.4.3.2 DownloadDriveParameter_ATV". "Conf Upload" entspricht dem Funktionsbaustein "2.4.3.1 UploadDriveParameter_ATV".

Mit der Visualisierung `Altivar_Startup_ManVis` wird der Status des Antriebsverstärkers angezeigt. Es kann die Betriebsart Profile Velocity (Bewegung mit vorgegebener Geschwindigkeit) oder die Betriebsart Jog gestartet werden.

Tabelle der Fehlernummern

Folgende Tabelle zeigt die Fehlernummern der Bibliothek. Die Fehlernummern des Antriebsverstärkers finden sie im Produkthandbuch.

ErrorID Hexadezimal	ErrorID Dezimal	HMI	Beschreibung
1000 _h	4096	ŁrF oLF 5oF	Kondensator Fehler Überlast Motor Zu hohe Drehzahl
2310 _h	8976	oLF	Überstrom Motor
2320 _h	8992	5LF	Kurzschluss Motor
2330 _h	9008	SCF	Kurzschluss Motorphasen (Erdschluss)
2340 _h	9024	SCF	Kurzschluss Motorphasen (Phase-Phase)
3110 _h	12560	o5F	Überspannung Netzversorgung
3120 _h	12576	u5F	Unterspannung Netzversorgung
3130 _h	12592	PhF	Netzphasenfehler
3310 _h	13072	obF oPF	Überspannung DC-Bus Fehler Motorphase
4210 _h	16912	ohF	Übertemperatur
5520 _h	21792	EEF	EEPROM Fehler
6100 _h	24832	, nF	Internes Ereignis
6300 _h	25344	ŁLF	Parameter außerhalb zulässigem Wertebereich
7300 _h	29440	ŁFF	Fehler an AI3
7510 _h	29968	5LF	Modbus Kommunikationsfehler
8100 _h	33024	ŁoF	CANopen Kommunikationsfehler, Heartbeat oder Life Guard Fehler
9000 _h	36864	EPF	Externer Fehler
A309 _h	41737		Antrieb nicht im Betriebszustand 6 Operation Enabled
FE00 _h	65024	ŁnF	Fehler beim Autotuning
FE01 _h	65025	bLF	Fehler Bremsenansteuerung
FF00 _h	65280	-	Toggle-Bit unverändert
FF01 _h	65281	-	SDO Zeitüberschreitung
FF02 _h	65282	-	server / client - command specifier ungültig oder nicht bekannt
FF03 _h	65283	-	Ungültige Blockgröße (nur im Block Mode)
FF04 _h	65284	-	Ungültige Sequenz-Nummer (nur im Block Mode)
FF05 _h	65285	-	CRC-Fehler (nur im Block Mode)
FF06 _h	65286	-	Kein Speicher frei
FF07 _h	65287	-	Kein Zugriff auf Objekt möglich
FF08 _h	65288	-	Kein Lesezugriff, da nur Schreib-Objekt (wo)
FF09 _h	65289	-	Kein Schreibzugriff, da Lese-Objekt (ro)
FF0A _h	65290	-	Objekt nicht im Objektverzeichnis vorhanden
FF0B _h	65291	-	Objekt unterstützt kein PDO-Mapping
FF0C _h	65292	-	Anzahl oder Länge der Objekte überschreiten die Byte-Länge des PDOs
FF0D _h	65293	-	Parameter sind nicht kompatibel
FF0E _h	65294	-	Gerät erkennt interne Inkompatibilität
FF0F _h	65295	-	Hardware-Fehler, Zugriff verweigert
FF10 _h	65296	-	Datentyp und Parameterlänge stimmen nicht überein
FF11 _h	65297	-	Datentyp stimmt nicht überein, Parameter zu lang

ErrorID Hexadezimal	ErrorID Dezimal	HMI	Beschreibung
FF12 _h	65298	-	Datentyp stimmt nicht überein, Parameter zu kurz
FF13 _h	65299	-	Subindex nicht unterstützt
FF14 _h	65300	-	Wertebereich des Parameters zu groß (nur für Schreibzugriff relevant)
FF15 _h	65301	-	Parameterwerte zu groß
FF16 _h	65302	-	Parameterwerte zu klein
FF17 _h	65303	-	Oberer Wert ist kleiner als unterer Wert
FF18 _h	65304	-	Allgemeiner Fehler
FF19 _h	65305	-	Daten können weder zur Anwendung übertragen noch gespeichert werden
FF1A _h	65306	-	Lokaler Zugriffskanal wird verwendet, Daten können nicht übertragen oder gespeichert werden
FF1B _h	65307	-	Gerätestatus sperrt das Übertragen und Speichern der Daten
FF1C _h	65308	-	Objektverzeichnis entweder nicht vorhanden oder nicht generierbar (zum Beispiel wenn Datenfehler bei Generierung aus Datei auftritt)
FF1D _h	65309	-	Reserviert
FF1E _h	65310	-	Reserviert
FF1F _h	65311	-	Reserviert
FF20 _h	65312	-	Unbekannter Status
FF21 _h	65313	-	Eingangsvariable wurde verändert bevor die Antwort empfangen wurde ("2.4.1.4 MC_ReadParameter_ATV", "2.4.2.1 MC_WriteParameter_ATV")
FF22 _h	65314	-	Versuch einen nicht unterbrechbaren Funktionsbaustein zu unterbrechen ("2.3.1.1 MC_Power_ATV", "2.3.5.1 MC_Stop_ATV")
FF34 _h	65332	-	Endstufe wechselt nicht in den Betriebszustand 6 Operation Enabled
FF37 _h	65335	-	Endstufe ist nicht im Betriebszustand 6 Operation Enabled
FF38 _h	65336	-	Parameterliste wurde noch nicht über "2.4.3.1 UploadDriveParameter_ATV" vom Gerät gelesen
FF39 _h	65337	-	Parameterliste passt nicht zum Gerät
FF3A _h	65338	-	Antriebsverstärker ist im Zustand PreOperational
FF3B _h	65339	-	Antriebsverstärker ist nicht im Betriebszustand 3 Switch On Disabled
FF3C _h	65340	-	STO "Safe Torque Off" (Power Removal) aktiv
FF3D _h	65341	-	Antriebsverstärker nicht kompatibel
FF3E _h	65342	-	Fehler im Mapping
FF50 _h	65360	-	Initialisierungsfehler des Funktionsbausteins "2.5.1.1 Altivar_Startup".
FF51 _h	65361	-	Der Funktionsbaustein "2.5.1.1 Altivar_Startup" kann nicht über die Anwendung gesteuert werden da <code>i_iControlMode = 1</code> .
FF52 _h	65362	-	Der Funktionsbaustein "2.5.1.1 Altivar_Startup" kann nicht über die Visualisierung gesteuert werden da <code>i_iControlMode = 0</code> .

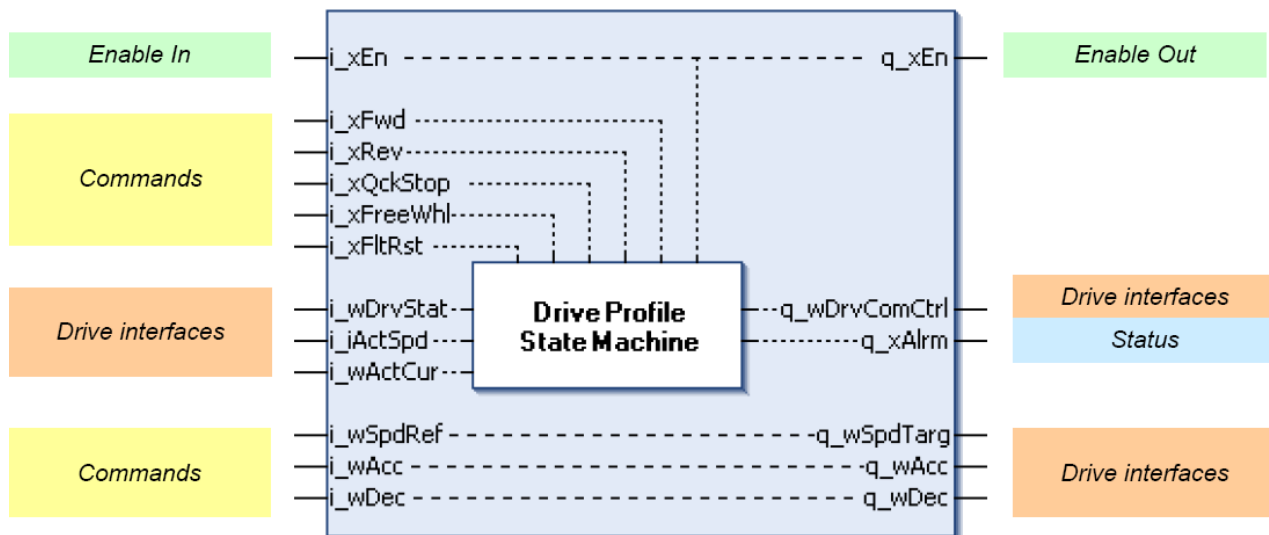
ErrorID Hexadezimal	ErrorID Dezimal	HMI	Beschreibung
FF53 _h	65363	-	Der Wert am Eingang <code>i_iControlMode</code> ist außerhalb des gültigen Wertebereichs.
FF54 _h	65364	-	Der Wert am Eingang <code>iq_iCmd</code> ist außerhalb des gültigen Wertebereichs.

2.5.2 Control

2.5.2.1 Altivar31_Control

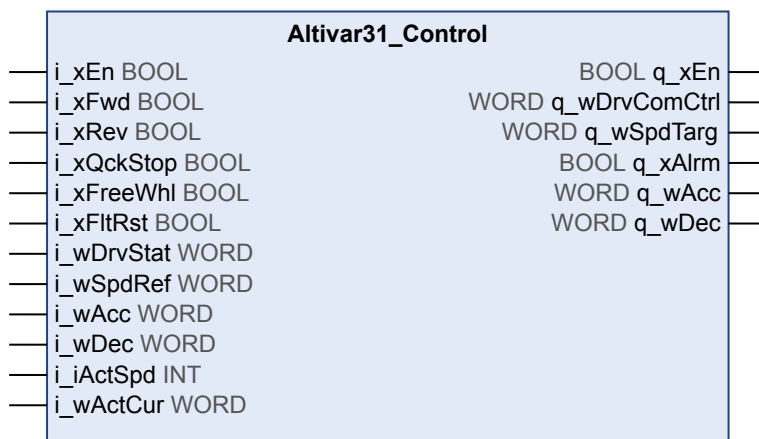
Funktionsbeschreibung Der Funktionsbaustein steuert das Steuerwort (q_wDrvComCtrl) des Antriebsverstärkers über das Statuswort (i_wDrvStat) und die anderen Eingänge.

Interner Aufbau des Funktionsbausteins Folgendes Bild zeigt eine Übersicht über den internen Aufbau des Funktionsbausteins.



Beschleunigung, Verzögerung und Sollgeschwindigkeit werden direkt vom Eingang auf den Ausgang kopiert. Die Sollgeschwindigkeit kann über die Visualisierung des Funktionsbausteins vorgegeben werden.

Grafische Darstellung



Kompatible Geräte ATV31/ATV312
Eingänge/Ausgänge Folgende Tabelle zeigt die Eingänge.

Eingang	Datentyp	Beschreibung
i_xEn	BOOL	<p>Wertebereich: FALSE, TRUE Initialwert: FALSE</p> <p>Befehl zum Aktivieren oder Deaktivieren des Funktionsbausteins.</p> <p>FALSE:</p> <ul style="list-style-type: none"> • Ausgang q_wDrvComCtrl wird auf 16#0000 gesetzt • Ausgang q_wSpdTarg wird auf 16#0000 gesetzt • Ausgang q_xAlrm wird auf FALSE gesetzt <p>TRUE: Funktionsbaustein aktiv</p>
i_xFwd	BOOL	<p>Wertebereich: FALSE, TRUE Initialwert: FALSE</p> <p>FALSE: Stoppt eine Bewegung in positive Richtung.</p> <p>TRUE: Wenn sich der Antriebsverstärker im Betriebszustand "Switched On" befindet und kein lokales Forcing aktiv ist, wird eine Bewegung in positive Richtung mit dem Geschwindigkeitssollwert i_wSpdRef gestartet.</p> <p>Der Befehl "Forward" wird mit einer steigenden Flanke ausgelöst. Beim Pegel FALSE wird die Bewegung beendet.</p>
i_xRev	BOOL	<p>Wertebereich: FALSE, TRUE Initialwert: FALSE</p> <p>FALSE: Stoppt eine Bewegung in negative Richtung.</p> <p>TRUE: Wenn sich der Antriebsverstärker im Betriebszustand "Switched On" befindet und kein lokales Forcing aktiv ist, wird eine Bewegung in negative Richtung mit dem Geschwindigkeitssollwert i_wSpdRef gestartet.</p> <p>Der Befehl "Reverse" wird mit einer steigenden Flanke ausgelöst. Beim Pegel FALSE wird die Bewegung beendet.</p>
i_xQckStop	BOOL	<p>Wertebereich: FALSE, TRUE Initialwert: FALSE</p> <p>FALSE: Wenn der Motor läuft, löst der Antriebsverstärker einen Quick Stop aus. Der Ausgang q_wDrvComCtrl wird auf 16#0002 gesetzt.</p> <p>TRUE: Normales Verhalten des Funktionsbausteins.</p> <p>Wenn nach einem Quick Stop die Istgeschwindigkeit und der Iststrom den Wert Null erreicht haben und wenn sowohl Forward als auch Backward FALSE sind, geht der Antriebsverstärker automatisch in den Betriebszustand "Switched On".</p> <p>Um den Motor neu zu starten, muss der Quick Stop deaktiviert werden (i_xQckStop auf TRUE setzen).</p>
i_xFreeWhl	BOOL	<p>Wertebereich: FALSE, TRUE Initialwert: FALSE</p> <p>FALSE: Wenn der Motor läuft, löst der Antriebsverstärker einen "Free Wheel Stop" aus. Der Ausgang q_wDrvComCtrl wird auf 16#0000 gesetzt.</p> <p>TRUE: Normales Verhalten des Funktionsbausteins.</p>
i_xFltRst	BOOL	<p>Wertebereich: FALSE, TRUE Initialwert: FALSE</p> <p>FALSE: Normales Verhalten des Funktionsbausteins.</p> <p>TRUE: Der Ausgang q_wDrvComCtrl wird auf 16#0080 gesetzt.</p>

Eingang	Datentyp	Beschreibung
i_wDrvStat	WORD	Wertebereich: Initialwert: 0 Muss direkt auf das Statuswort des Antriebsverstärkers gemappt werden (CANopen-Objekt 6041). Dieser Wert darf zwischen dem CANopen-Interface und dem Funktionsbaustein nicht geändert werden.
i_wSpdRef	WORD	Wertebereich: Initialwert: 0 Sollgeschwindigkeit für den Antriebsverstärker. Wird direkt in die Zielgeschwindigkeit $q_wSpdTarg$ kopiert, wenn der Funktionsbaustein aktiviert wird und wenn über die Visualisierung kein bestimmter Wert vorgegeben ist.
i_wAcc	WORD	Wertebereich: - Initialwert: - Beschleunigung: Wird direkt auf den Ausgang q_wAcc kopiert, wenn der Funktionsbaustein aktiviert wird.
i_wDec	WORD	Wertebereich: -- Initialwert: - Verzögerung: Wird direkt auf den Ausgang q_wDec kopiert, wenn der Funktionsbaustein aktiviert wird.
i_iActSpd	INT	Wertebereich: Initialwert: 0 Istgeschwindigkeit des Antriebs. Dieser Eingang muss direkt auf das PDO gemappt werden (CANopen-Objekt 2002 / Subindex 03, um die Geschwindigkeit in 0,1 Hz zu erhalten).
i_wActCur	WORD	Wertebereich: Initialwert: 16#FFFF Iststrom des Antriebsverstärkers. Dieser Eingang muss direkt auf das PDO gemappt werden (CANopen-Objekt 2002 / Subindex 05, Einheit 0,1 A). Dieser Eingang wird verwendet, um zu erkennen, ob der Motorstromwert gleich Null oder ungleich Null ist.

Folgende Tabelle zeigt die Ausgänge.

Ausgang	Datentyp	Beschreibung
q_xEn	BOOL	Wertebereich: FALSE, TRUE Initialwert: FALSE Funktionsbaustein aktiviert/deaktiviert. Direkte Kopie von i_xEn.
q_wDrvComCtrl	WORD	Wertebereich: Initialwert: Muss direkt auf das Steuerwort des Antriebsverstärkers gemappt werden (CANopen-Objekt 6040). Dieser Wert darf zwischen dem CANopen-Interface und dem Funktionsbaustein nicht geändert werden.
q_wSpdTarg	WORD	Wertebereich: Initialwert: Zielgeschwindigkeit für den Antriebsverstärker. Wird direkt von der Sollgeschwindigkeit i_wSpdRef kopiert, wenn der Funktionsbaustein aktiviert wird und wenn die Visualisierung keinen bestimmten Wert vorgibt. Die Zielgeschwindigkeit wird auf 0 gesetzt, wenn der Funktionsbaustein nicht aktiviert ist. Dieser Ausgang muss direkt auf das PDO gemappt werden (CANopen-Objekt 6042 zur Übertragung in Umdrehungen pro Minute).
q_xAlrm	BOOL	Wertebereich: FALSE, TRUE Initialwert: FALSE Wird auf FALSE gesetzt, wenn der Funktionsbaustein deaktiviert wird und wenn der Antriebsverstärker in den Betriebszustand "Switch On Disabled" geht (siehe Zustandsdiagramm des Antriebsprofils). Wird auf TRUE gesetzt, wenn ein Fehler vom Antriebsverstärker erkannt wird (Bit 3 des Statusworts).
q_wAcc	WORD	Wertebereich: - Initialwert: - Beschleunigung: Wird direkt vom Eingang i_wAcc kopiert, wenn der Funktionsbaustein aktiviert wird. Dieser Ausgang muss direkt auf das PDO gemappt werden (CANopen-Objekt 203C:2).
q_wDec	WORD	Wertebereich: - Initialwert: - Verzögerung: Wird direkt vom Eingang i_wDec kopiert, wenn der Funktionsbaustein aktiviert wird. Dieser Ausgang muss direkt auf das PDO gemappt werden (CANopen-Objekt 203C:3).

Hinweise

▲ WARNUNG**UNBEABSICHTIGTES VERHALTEN DURCH INKONSISTENTE BEFEHLE**

Wenn Sie diesen Funktionsbaustein aktiviert haben, führt die gleichzeitige Verwendung anderer Funktionsbausteine der Bibliothek zu unbeabsichtigtem Verhalten.

- Aktivieren Sie diesen Funktionsbaustein nur, wenn alle anderen Funktionsbausteine der Bibliothek inaktiv sind.
- Deaktivieren Sie diesen Funktionsbaustein, bevor Sie andere Funktionsbausteine der Bibliothek aktivieren.

Nichtbeachtung dieser Vorkehrungen kann zu Tod, schwerwiegenden Verletzungen oder Materialschäden führen.

Beachten Sie folgende Punkte:

- Der Antriebsverstärker verlässt nach einem "Quick Stop" automatisch den Betriebszustand "Quick Stop Active" (siehe Zustandsdiagramm weiter unten), sobald die Istgeschwindigkeit und der Iststrom den Wert Null erreicht haben und wenn sowohl Forward als auch Backward FALSE sind. Um den Motor neu zu starten, muss der Quick Stop deaktiviert werden (`i_xQckStop` auf TRUE setzen).
- Ein "Quick Stop" hat eine höhere Priorität als ein regulärer Stopp ("Forward" und "Reverse" auf FALSE).
- Ein "Free Wheel Stop" hat eine höhere Priorität als ein "Quick Stop".
- Wenn auf der 7-Segment-Anzeige des Antriebsverstärkers nach dem Download einer Anwendung auf den Antriebsverstärker die blinkende Meldung $\text{E}oF$ erscheint, ist eine steigende und dann eine fallende Flanke am Eingang Fault Reset (`i_xFltRst`) erforderlich, um die korrekte Kommunikation mit dem Antriebsverstärker neu zu starten.

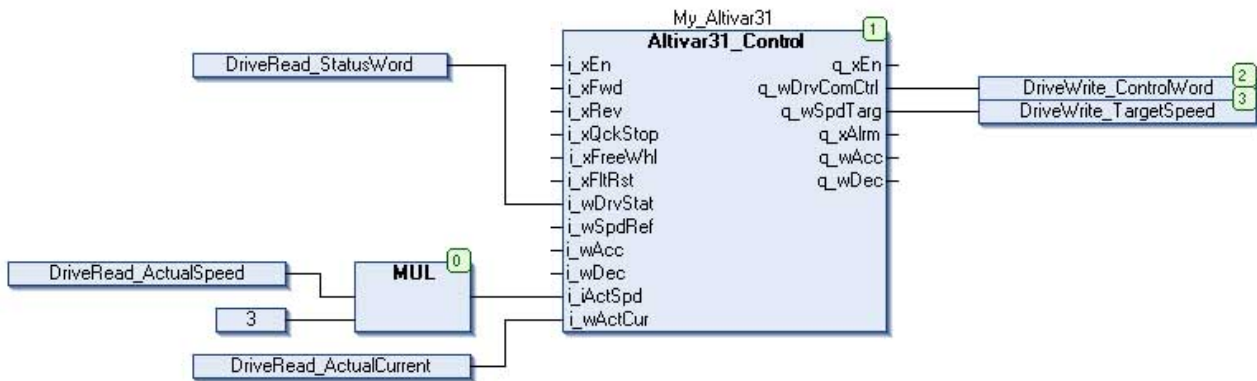
Verwendung des Funktionsbausteins

Vorgehen zum Starten des Funktionsbausteins mit den Voreinstellungen:

Schritt	Aktion
1	Statuswort und Steuerwort auf PDOs mappen: <ul style="list-style-type: none"> • <code>i_wDrvStat</code> auf PDO vom Antriebsverstärker zur Steuerung mappen • <code>q_wDrvComCtrl</code> auf PDO von Steuerung zum Antriebsverstärker mappen
2	Istgeschwindigkeit und Iststrom auf PDO vom Antriebsverstärker zur Steuerung mappen: <ul style="list-style-type: none"> • <code>i_wActCur</code> • <code>i_iActSpd</code>
3	"Free Wheel" deaktivieren: <code>i_xFreeWhl</code> auf TRUE setzen.
4	"Quick Stop" deaktivieren: <code>i_xQckStop</code> auf TRUE setzen.
5	Funktionsbaustein aktivieren: <code>i_xEn</code> auf TRUE setzen.
6	Sollgeschwindigkeit einstellen: <code>i_wSpdRef</code> auf einen Wert ungleich Null setzen.
7	Bewegung in positive ("Forward") oder negative ("Reverse") Richtung starten: <code>i_xFwd</code> oder <code>i_xRev</code> auf TRUE setzen.

Direktes Mapping PDOs - CANopen

Das Interface für das direkte Mapping von PDOs auf CANopen sieht folgendermaßen aus:



Die Istgeschwindigkeit wird in 0,1 Hz gelesen (von 0 bis 500, wenn HSP = 50 Hz)

Die Frequenz muss:

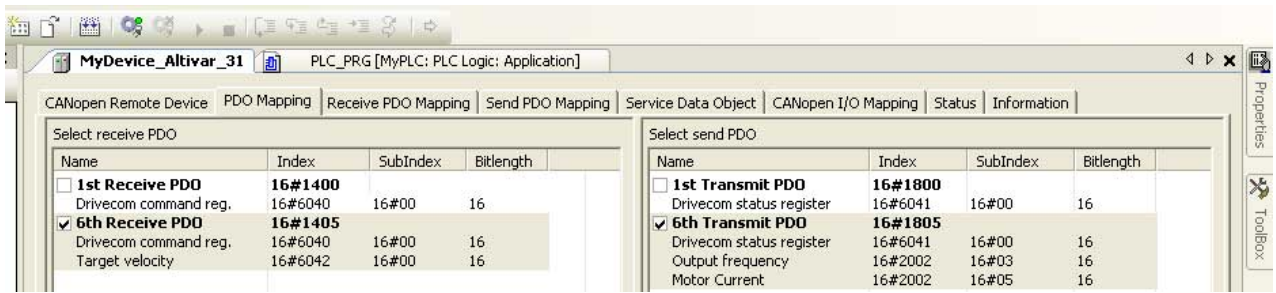
- durch 10 dividiert werden (um die Geschwindigkeit in Hz zu erhalten)
- durch 2 dividiert werden (2 Polpaare)
- mit 60 multipliziert werden (60 um die Geschwindigkeit in Umdrehungen pro Minute zu erhalten)

Bei Motoren mit 2 Polpaaren kann der Rechenweg verkürzt werden:

Die Istgeschwindigkeit multipliziert mit 3 konvertiert 0,1 Hz zu rpm (min⁻¹).

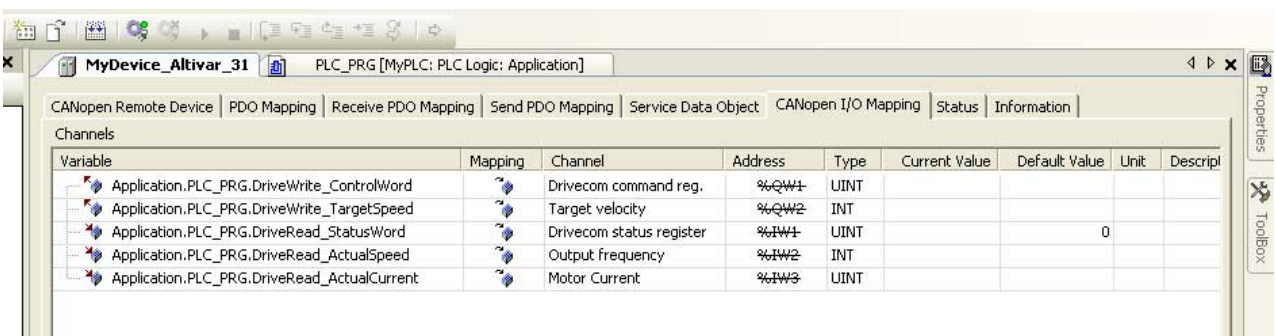
019844113879, V2.08, 04.2011

PDO Konfiguration:

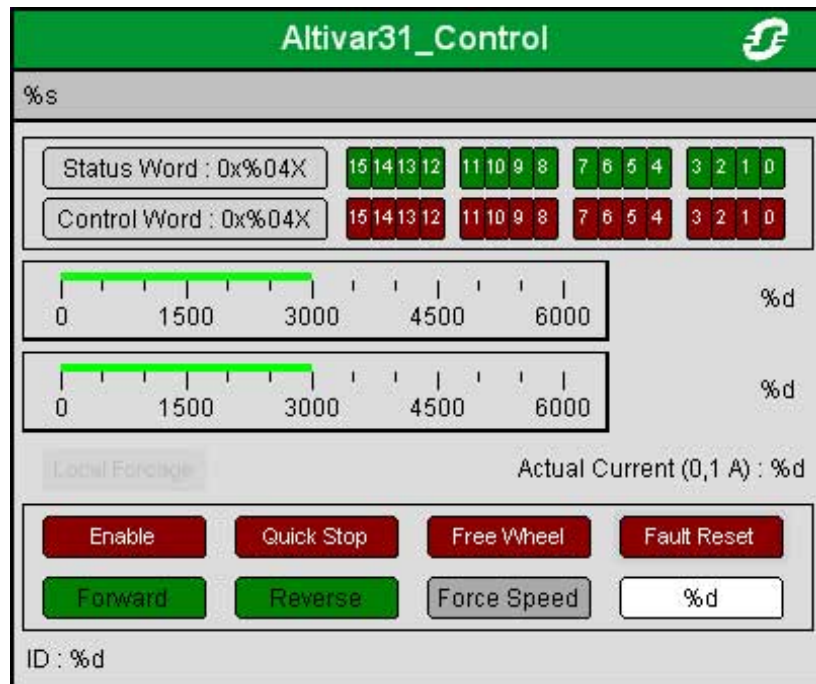


Das Bild zeigt eine Beispielkonfiguration. Andere Konfigurationen der PDOs sind möglich.

Mapping der Daten auf das PDO:

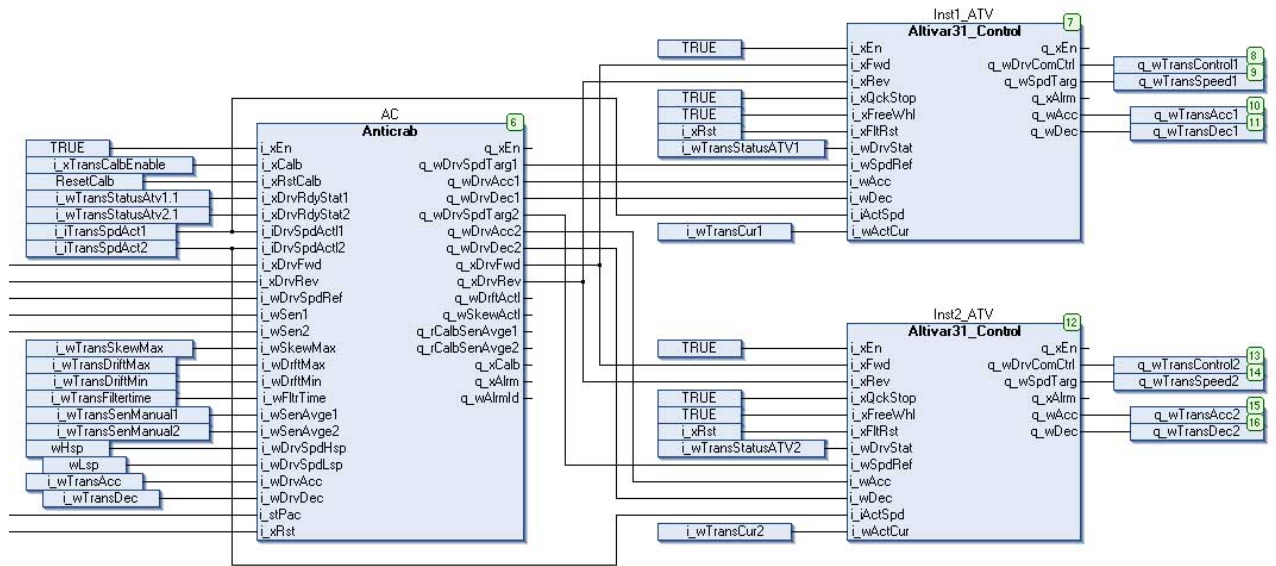


Visualisierung Mit der oben gezeigten Minimalkonfiguration kann die Visualisierung des Funktionsbausteins zur Steuerung des Antriebsverstärkers verwendet werden. Nach dem PDO-Mapping der 5 oben angegebenen Werte kann der Antrieb mit den folgenden Schritten gestartet werden:



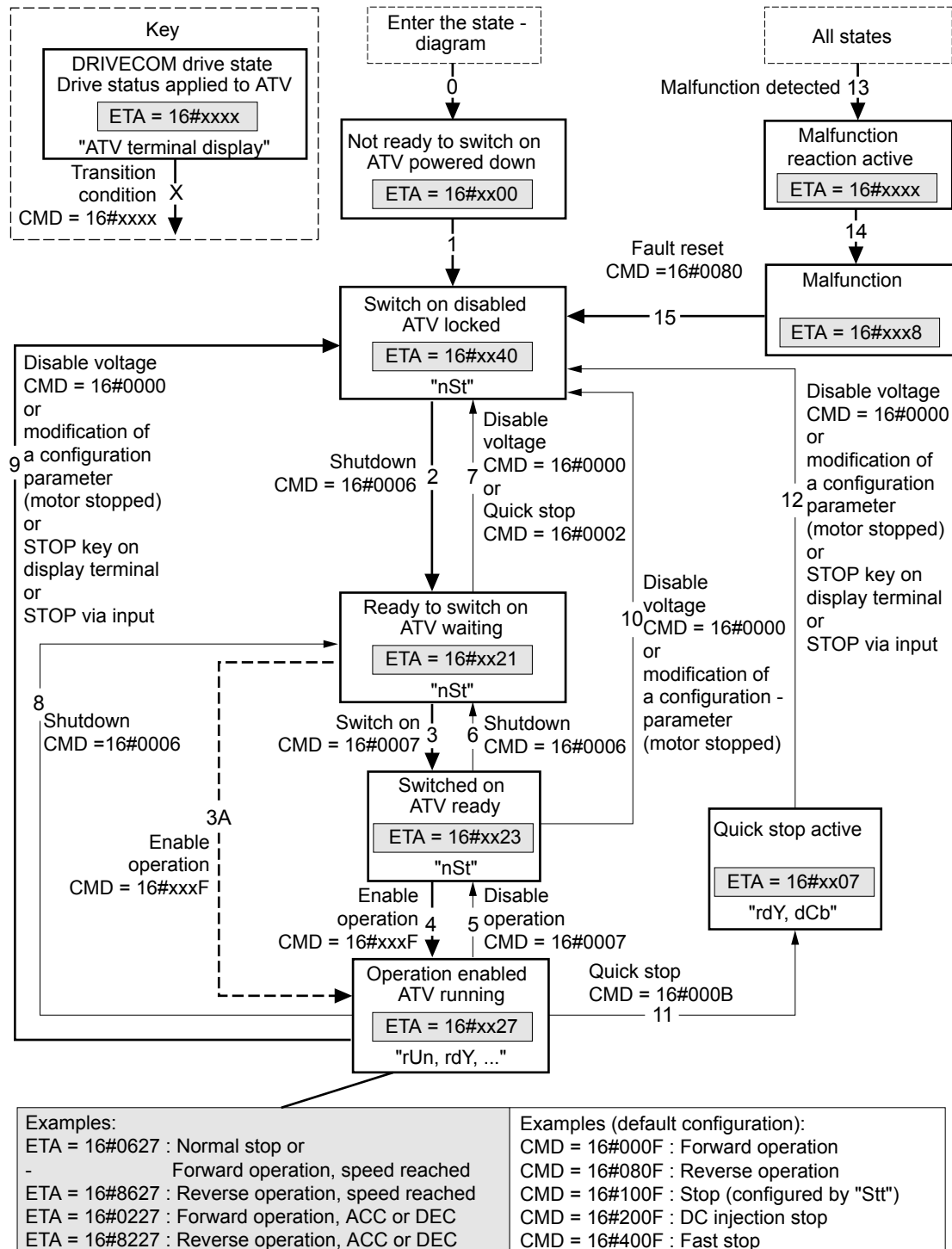
Schritt	Aktion
1	Klick auf Schaltfläche "Enable", um den Funktionsbaustein zu aktivieren
2	Klick auf Schaltfläche "Quick Stop", um "Quick Stop" zu deaktivieren
3	Klick auf Schaltfläche "Free Wheel", um "Free Wheel" zu deaktivieren
4	Geschwindigkeitswert ungleich Null (in Umdrehungen pro Minute) eingeben (im Feld neben der Schaltfläche Force Speed).
5	Klick auf Schaltfläche "Force Speed"
6	Klick auf Schaltfläche "Forward" oder "Reverse": Der Motor läuft

Beispiel einer Anwendung des Funktionsbausteins:



Altivar 31 Antriebsprofil CiA402 Zustandsdiagramm:

DRIVECOM state diagram

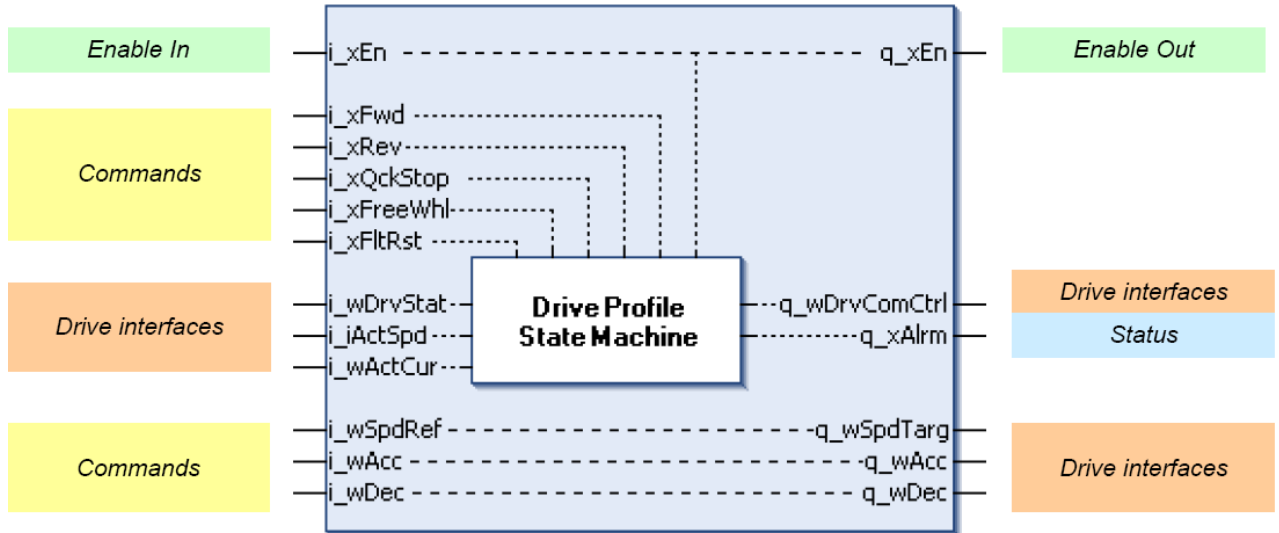


Verlassen des Zustands "Operation Enabled" über "Disable Voltage" (9) oder "Shutdown" (8) löst einen "Freewheel Stop" aus.

2.5.2.2 Altivar71_Control

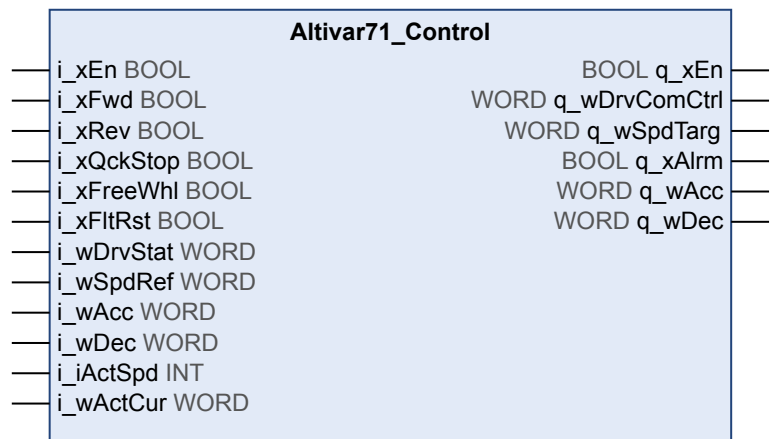
Funktionsbeschreibung Der Funktionsbaustein steuert das Steuerwort (q_wDrvComCtrl) des Antriebsverstärkers über das Statuswort (i_wDrvStat) und die anderen Eingänge.

Interner Aufbau des Funktionsbaustein Folgendes Bild zeigt eine Übersicht über den internen Aufbau des Funktionsbausteins.



Beschleunigung, Verzögerung und Sollgeschwindigkeit werden direkt vom Eingang auf den Ausgang kopiert. Die Sollgeschwindigkeit kann über die Visualisierung des Funktionsbausteins vorgegeben werden.

Grafische Darstellung



Kompatible Geräte ATV71

Eingänge/Ausgänge Folgende Tabelle zeigt die Eingänge.

Eingang	Datentyp	Beschreibung
i_xEn	BOOL	<p>Wertebereich: FALSE, TRUE Initialwert: FALSE</p> <p>Befehl zum Aktivieren oder Deaktivieren des Funktionsbausteins.</p> <p>FALSE:</p> <ul style="list-style-type: none"> • Ausgang q_wDrvComCtrl wird auf 16#0000 gesetzt • Ausgang q_wSpdTarg wird auf 16#0000 gesetzt • Ausgang q_xAlrm wird auf FALSE gesetzt <p>TRUE: Funktionsbaustein aktiv</p>
i_xFwd	BOOL	<p>Wertebereich: FALSE, TRUE Initialwert: FALSE</p> <p>FALSE: Stoppt eine Bewegung in positive Richtung.</p> <p>TRUE: Wenn sich der Antriebsverstärker im Betriebszustand "Switched On" befindet und kein lokales Forcing aktiv ist, wird eine Bewegung in positive Richtung mit dem Geschwindigkeitssollwert i_wSpdRef gestartet.</p> <p>Der Befehl "Forward" wird mit einer steigenden Flanke ausgelöst. Beim Pegel FALSE wird die Bewegung beendet.</p>
i_xRev	BOOL	<p>Wertebereich: FALSE, TRUE Initialwert: FALSE</p> <p>FALSE: Stoppt eine Bewegung in negative Richtung.</p> <p>TRUE: Wenn sich der Antriebsverstärker im Betriebszustand "Switched On" befindet und kein lokales Forcing aktiv ist, wird eine Bewegung in negative Richtung mit dem Geschwindigkeitssollwert i_wSpdRef gestartet.</p> <p>Der Befehl "Reverse" wird mit einer steigenden Flanke ausgelöst. Beim Pegel FALSE wird die Bewegung beendet.</p>
i_xQckStop	BOOL	<p>Wertebereich: FALSE, TRUE Initialwert: FALSE</p> <p>FALSE: Wenn der Motor läuft, löst der Antriebsverstärker einen Quick Stop aus. Der Ausgang q_wDrvComCtrl wird auf 16#0002 gesetzt.</p> <p>TRUE: Normales Verhalten des Funktionsbausteins.</p> <p>Wenn nach einem Quick Stop die Istgeschwindigkeit und der Iststrom den Wert Null erreicht haben und wenn sowohl Forward als auch Backward FALSE sind, geht der Antriebsverstärker automatisch in den Betriebszustand "Switched On".</p> <p>Um den Motor neu zu starten, muss der Quick Stop deaktiviert werden (i_xQckStop auf TRUE setzen).</p>
i_xFreeWhl	BOOL	<p>Wertebereich: FALSE, TRUE Initialwert: FALSE</p> <p>FALSE: Wenn der Motor läuft, löst der Antriebsverstärker einen "Free Wheel Stop" aus. Der Ausgang q_wDrvComCtrl wird auf 16#0000 gesetzt.</p> <p>TRUE: Normales Verhalten des Funktionsbausteins.</p>
i_xFltRst	BOOL	<p>Wertebereich: FALSE, TRUE Initialwert: FALSE</p> <p>FALSE: Normales Verhalten des Funktionsbausteins.</p> <p>TRUE: Der Ausgang q_wDrvComCtrl wird auf 16#0080 gesetzt.</p>

Eingang	Datentyp	Beschreibung
i_wDrvStat	WORD	Wertebereich: Initialwert: 0 Muss direkt auf das Statuswort des Antriebsverstärkers gemappt werden (CANopen-Objekt 6041). Dieser Wert darf zwischen dem CANopen-Interface und dem Funktionsbaustein nicht geändert werden.
i_wSpdRef	WORD	Wertebereich: Initialwert: 0 Sollgeschwindigkeit für den Antriebsverstärker. Wird direkt in die Zielgeschwindigkeit q_wSpdTarg kopiert, wenn der Funktionsbaustein aktiviert wird und wenn über die Visualisierung kein bestimmter Wert vorgegeben ist.
i_wAcc	WORD	Wertebereich: - Initialwert: - Beschleunigung: Wird direkt auf den Ausgang q_wAcc kopiert, wenn der Funktionsbaustein aktiviert wird.
i_wDec	WORD	Wertebereich: - Initialwert: - Verzögerung: Wird direkt auf den Ausgang q_wDec kopiert, wenn der Funktionsbaustein aktiviert wird.
i_iActSpd	INT	Wertebereich: Initialwert: 0 Istgeschwindigkeit des Antriebs. Dieser Eingang muss direkt auf das PDO gemappt werden (CANopen-Objekt 2002 / Subindex 03, um die Geschwindigkeit in 0,1 Hz zu erhalten).
i_wActCur	WORD	Wertebereich: Initialwert: 16#FFFF Iststrom des Antriebsverstärkers. Dieser Eingang muss direkt auf das PDO gemappt werden (CANopen-Objekt 2002 / Subindex 05, Einheit 0,1 A). Dieser Eingang wird verwendet, um zu erkennen, ob der Motorstromwert gleich Null oder ungleich Null ist.

Folgende Tabelle zeigt die Ausgänge.

Ausgang	Datentyp	Beschreibung
q_xEn	BOOL	Wertebereich: FALSE, TRUE Initialwert: FALSE Funktionsbaustein aktiviert/deaktiviert. Direkte Kopie von i_xEn.
q_wDrvComCtrl	WORD	Wertebereich: Initialwert: Muss direkt auf das Steuerwort des Antriebsverstärkers gemappt werden (CANopen-Objekt 6040). Dieser Wert darf zwischen dem CANopen-Interface und dem Funktionsbaustein nicht geändert werden.
q_wSpdTarg	WORD	Wertebereich: Initialwert: Zielgeschwindigkeit für den Antriebsverstärker. Wird direkt von der Sollgeschwindigkeit i_wSpdRef kopiert, wenn der Funktionsbaustein aktiviert wird und wenn die Visualisierung keinen bestimmten Wert vorgibt. Die Zielgeschwindigkeit wird auf 0 gesetzt, wenn der Funktionsbaustein nicht aktiviert ist. Dieser Ausgang muss direkt auf das PDO gemappt werden (CANopen-Objekt 6042 zur Übertragung in Umdrehungen pro Minute).
q_xAlrm	BOOL	Wertebereich: FALSE, TRUE Initialwert: FALSE Wird auf FALSE gesetzt, wenn der Funktionsbaustein deaktiviert wird und wenn der Antriebsverstärker in den Betriebszustand "Switch On Disabled" geht (siehe Zustandsdiagramm des Antriebsprofils). Wird auf TRUE gesetzt, wenn ein Fehler vom Antriebsverstärker erkannt wird (Bit 3 des Statusworts).
q_wAcc	WORD	Wertebereich: - Initialwert: - Beschleunigung: Wird direkt vom Eingang i_wAcc kopiert, wenn der Funktionsbaustein aktiviert wird. Dieser Ausgang muss direkt auf das PDO gemappt werden (CANopen-Objekt 203C:2).
q_wDec	WORD	Wertebereich: - Initialwert: - Verzögerung: Wird direkt vom Eingang i_wDec kopiert, wenn der Funktionsbaustein aktiviert wird. Dieser Ausgang muss direkt auf das PDO gemappt werden (CANopen-Objekt 203C:3).

Hinweise

⚠ WARNUNG**UNBEABSICHTIGTES VERHALTEN DURCH INKONSISTENTE BEFEHLE**

Wenn Sie diesen Funktionsbaustein aktiviert haben, führt die gleichzeitige Verwendung anderer Funktionsbausteine der Bibliothek zu unbeabsichtigtem Verhalten.

- Aktivieren Sie diesen Funktionsbaustein nur, wenn alle anderen Funktionsbausteine der Bibliothek inaktiv sind.
- Deaktivieren Sie diesen Funktionsbaustein, bevor Sie andere Funktionsbausteine der Bibliothek aktivieren.

Nichtbeachtung dieser Vorkehrungen kann zu Tod, schwerwiegenden Verletzungen oder Materialschäden führen.

Beachten Sie folgende Punkte:

- Der Antriebsverstärker verlässt nach einem "Quick Stop" automatisch den Betriebszustand "Quick Stop Active" (siehe Zustandsdiagramm weiter unten), sobald die Istgeschwindigkeit und der Iststrom den Wert Null erreicht haben und wenn sowohl Forward als auch Backward FALSE sind. Um den Motor neu zu starten, muss der Quick Stop deaktiviert werden (`i_xQckStop` auf TRUE setzen).
- Ein "Quick Stop" hat eine höhere Priorität als ein regulärer Stopp ("Forward" und "Reverse" auf FALSE).
- Ein "Free Wheel Stop" hat eine höhere Priorität als ein "Quick Stop".
- Wenn auf der 7-Segment-Anzeige des Antriebsverstärkers nach dem Download einer Anwendung auf den Antriebsverstärker die blinkende Meldung $\text{C}\alpha\text{F}$ erscheint, ist eine steigende und dann eine fallende Flanke am Eingang Fault Reset (`i_xFltRst`) erforderlich, um die korrekte Kommunikation mit dem Antriebsverstärker neu zu starten.

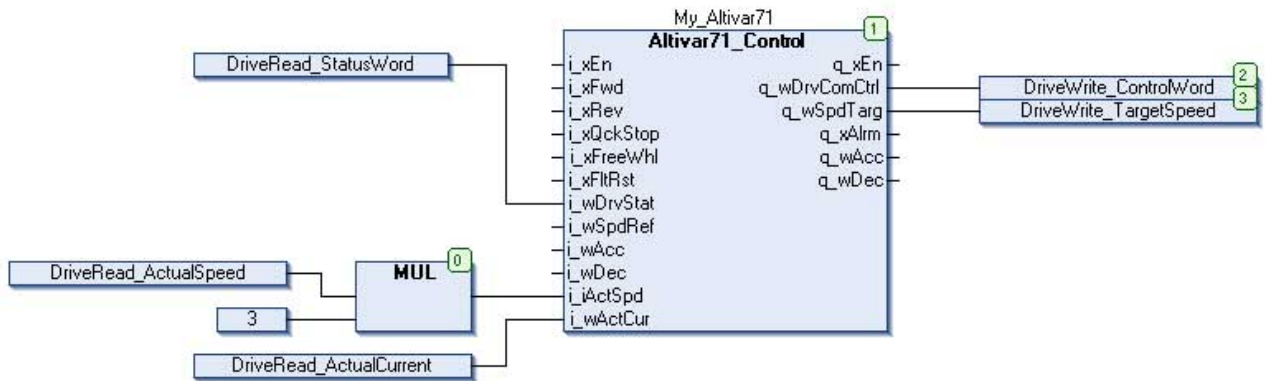
Verwendung des Funktionsbausteins

Vorgehen zum Starten des Funktionsbausteins mit den Voreinstellungen:

Schritt	Aktion
1	Statuswort und Steuerwort auf PDOs mappen: <ul style="list-style-type: none"> • <code>i_wDrvStat</code> auf PDO vom Antriebsverstärker zur Steuerung mappen • <code>q_wDrvComCtrl</code> auf PDO von Steuerung zum Antriebsverstärker mappen
2	Istgeschwindigkeit und Iststrom auf PDO vom Antriebsverstärker zur Steuerung mappen: <ul style="list-style-type: none"> • <code>i_wActCur</code> • <code>i_iActSpd</code>
3	"Free Wheel" deaktivieren: <code>i_xFreeWhl</code> auf TRUE setzen.
4	"Quick Stop" deaktivieren: <code>i_xQckStop</code> auf TRUE setzen.
5	Funktionsbaustein aktivieren: <code>i_xEn</code> auf TRUE setzen.
6	Sollgeschwindigkeit einstellen: <code>i_wSpdRef</code> auf einen Wert ungleich Null setzen.
7	Bewegung in positive ("Forward") oder negative ("Reverse") Richtung starten: <code>i_xFwd</code> oder <code>i_xRev</code> auf TRUE setzen.

Direktes Mapping PDOs - CANopen

Das Interface für das direkte Mapping von PDOs auf CANopen sieht folgendermaßen aus:



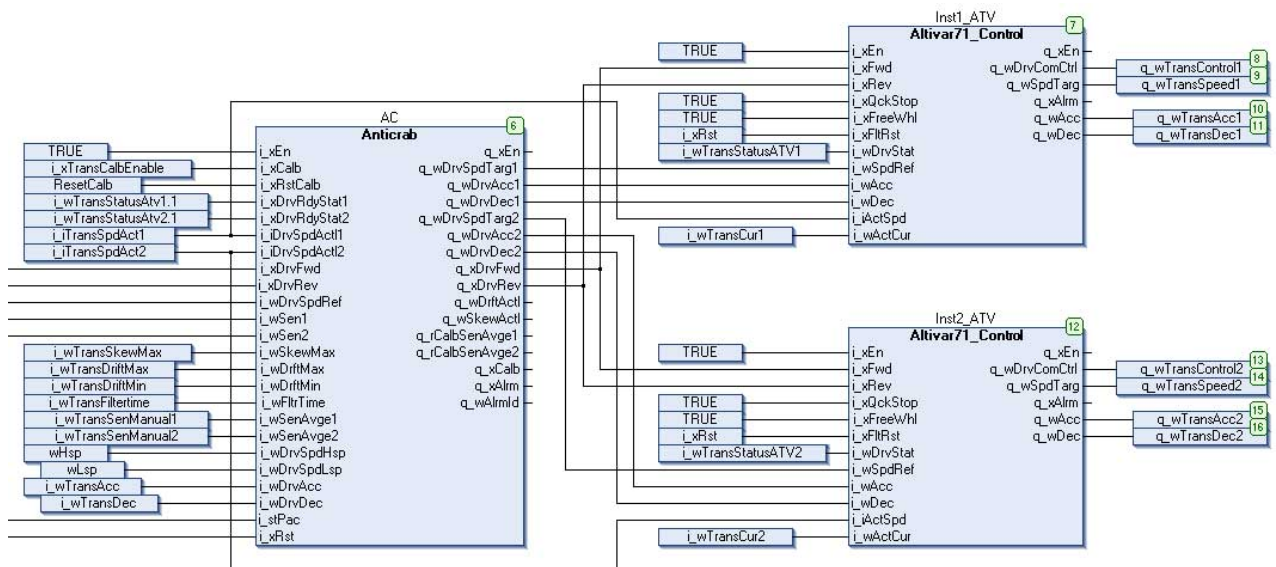
Die Istgeschwindigkeit wird in 0,1 Hz gelesen (von 0 bis 500, wenn HSP = 50 Hz)

Die Frequenz muss:

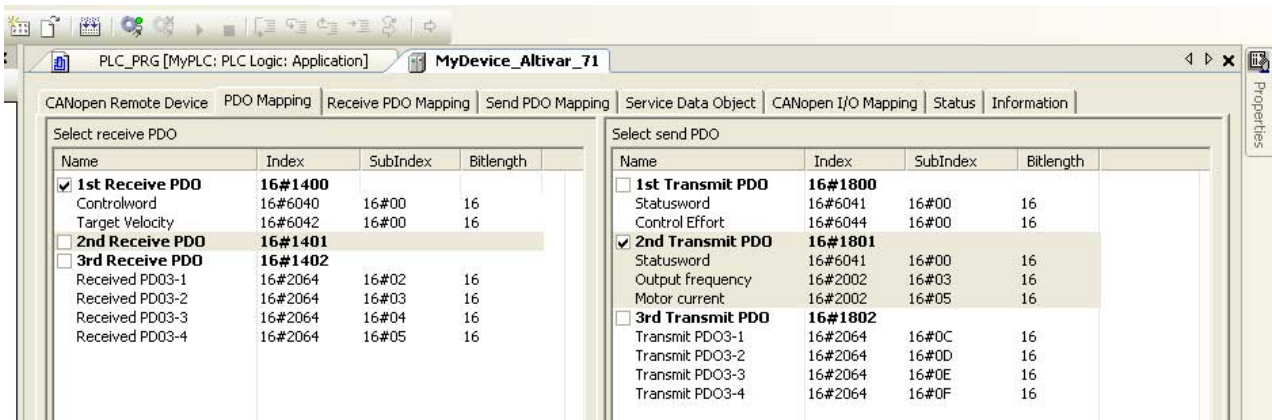
- durch 10 dividiert werden (um die Geschwindigkeit in Hz zu erhalten)
- durch 2 dividiert werden (2 Polpaare)
- mit 60 multipliziert werden (60 um die Geschwindigkeit in Umdrehungen pro Minute zu erhalten)

Bei Motoren mit 2 Polpaaren kann der Rechenweg verkürzt werden:

Die Istgeschwindigkeit multipliziert mit 3 konvertiert 0,1 Hz zu rpm (min⁻¹).

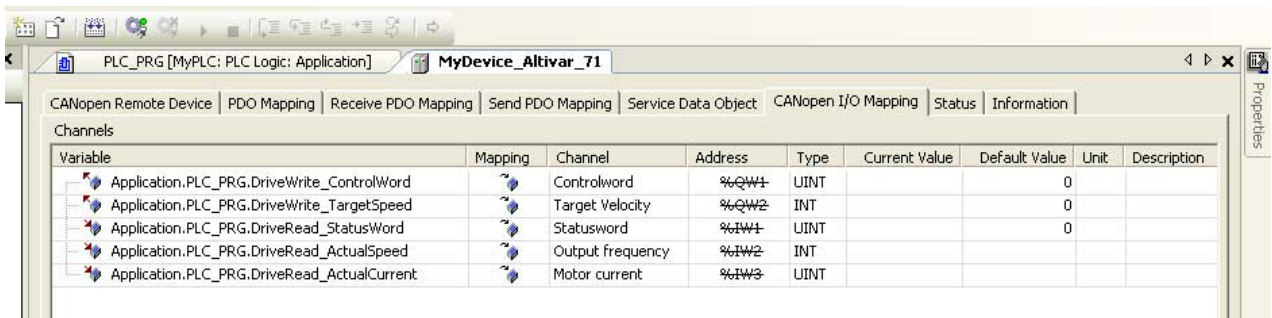


PDO Konfiguration:

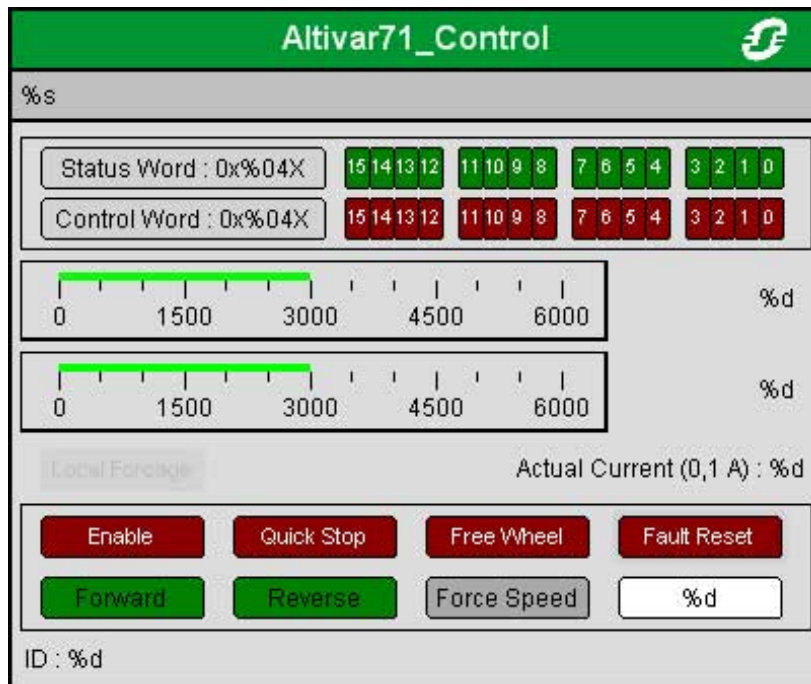


Das Bild zeigt eine Beispielkonfiguration. Andere Konfigurationen der PDOs sind möglich.

Mapping der Daten auf das PDO:

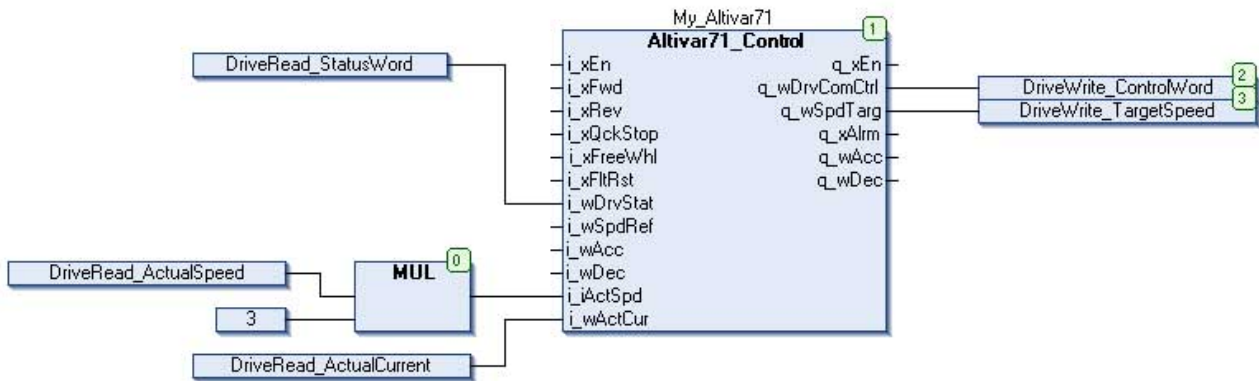


Visualisierung Mit der oben gezeigten Minimalkonfiguration kann die Visualisierung des Funktionsbausteins zur Steuerung des Antriebsverstärkers verwendet werden. Nach dem PDO-Mapping der 5 oben angegebenen Werte kann der Antrieb mit den folgenden Schritten gestartet werden:

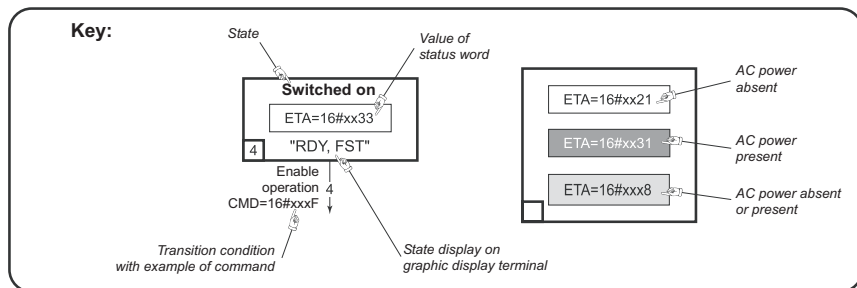
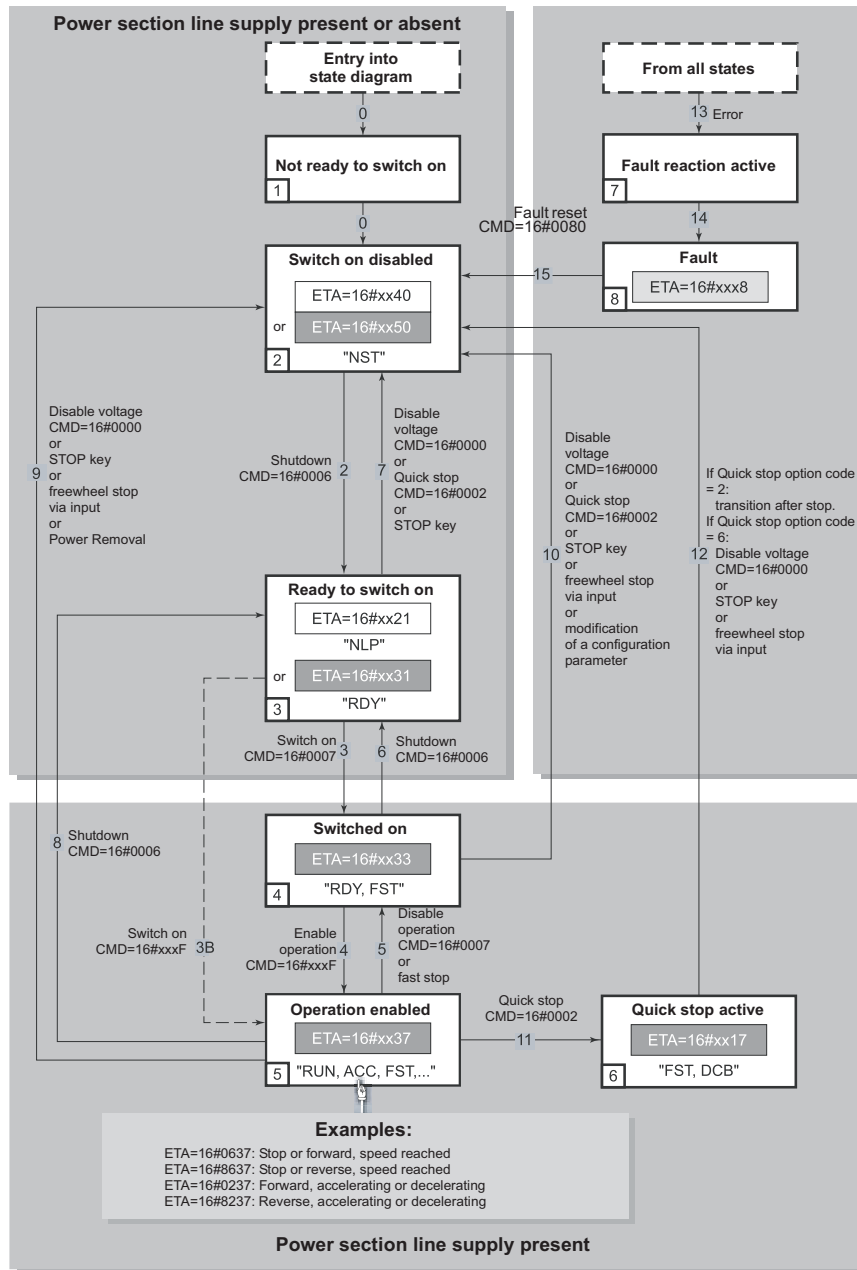


Schritt	Aktion
1	Klick auf Schaltfläche "Enable", um den Funktionsbaustein zu aktivieren
2	Klick auf Schaltfläche "Quick Stop", um "Quick Stop" zu deaktivieren
3	Klick auf Schaltfläche "Free Wheel", um "Free Wheel" zu deaktivieren
4	Geschwindigkeitswert ungleich Null (in Umdrehungen pro Minute) eingeben (im Feld neben der Schaltfläche Force Speed).
5	Klick auf Schaltfläche "Force Speed"
6	Klick auf Schaltfläche "Forward" oder "Reverse": Der Motor läuft

Beispiel einer Anwendung des Funktionsbausteins (wie bei Altivar31_Control):



Altivar 71 Antriebsprofil CiA402 Zustandsdiagramm:

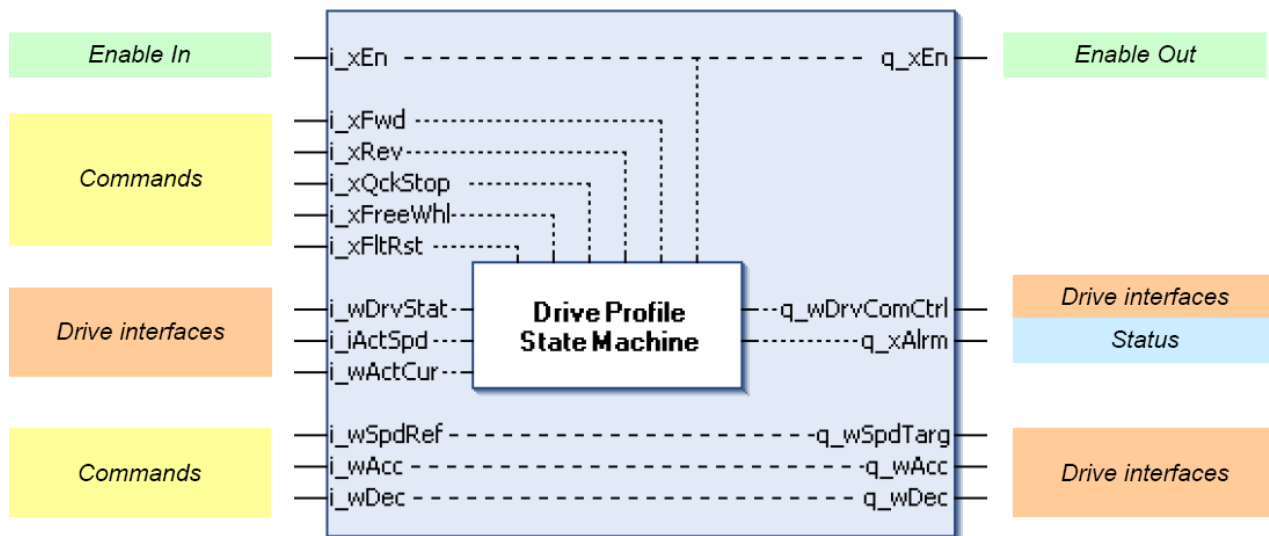


019844113879, V2.08, 04.2011

2.5.2.3 Altivar32_Control

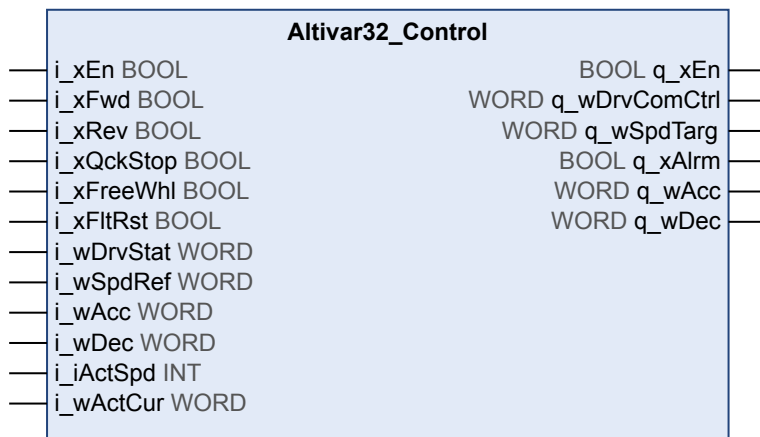
Funktionsbeschreibung Der Funktionsbaustein steuert das Steuerwort (q_wDrvComCtrl) des Antriebsverstärkers über das Statuswort (i_wDrvStat) und die anderen Eingänge.

Interner Aufbau des Funktionsbaustein Folgendes Bild zeigt eine Übersicht über den internen Aufbau des Funktionsbausteins.



Beschleunigung, Verzögerung und Sollgeschwindigkeit werden direkt vom Eingang auf den Ausgang kopiert. Die Sollgeschwindigkeit kann über die Visualisierung des Funktionsbausteins vorgegeben werden.

Grafische Darstellung



Kompatible Geräte ATV32

Eingänge/Ausgänge Folgende Tabelle zeigt die Eingänge.

Eingang	Datentyp	Beschreibung
i_xEn	BOOL	<p>Wertebereich: FALSE, TRUE Initialwert: FALSE</p> <p>Befehl zum Aktivieren oder Deaktivieren des Funktionsbausteins.</p> <p>FALSE:</p> <ul style="list-style-type: none"> • Ausgang q_wDrvComCtrl wird auf 16#0000 gesetzt • Ausgang q_wSpdTarg wird auf 16#0000 gesetzt • Ausgang q_xAlrm wird auf FALSE gesetzt <p>TRUE: Funktionsbaustein aktiv</p>
i_xFwd	BOOL	<p>Wertebereich: FALSE, TRUE Initialwert: FALSE</p> <p>FALSE: Stoppt eine Bewegung in positive Richtung.</p> <p>TRUE: Wenn sich der Antriebsverstärker im Betriebszustand "Switched On" befindet und kein lokales Forcing aktiv ist, wird eine Bewegung in positive Richtung mit dem Geschwindigkeitssollwert i_wSpdRef gestartet.</p> <p>Der Befehl "Forward" wird mit einer steigenden Flanke ausgelöst. Beim Pegel FALSE wird die Bewegung beendet.</p>
i_xRev	BOOL	<p>Wertebereich: FALSE, TRUE Initialwert: FALSE</p> <p>FALSE: Stoppt eine Bewegung in negative Richtung.</p> <p>TRUE: Wenn sich der Antriebsverstärker im Betriebszustand "Switched On" befindet und kein lokales Forcing aktiv ist, wird eine Bewegung in negative Richtung mit dem Geschwindigkeitssollwert i_wSpdRef gestartet.</p> <p>Der Befehl "Reverse" wird mit einer steigenden Flanke ausgelöst. Beim Pegel FALSE wird die Bewegung beendet.</p>
i_xQckStop	BOOL	<p>Wertebereich: FALSE, TRUE Initialwert: FALSE</p> <p>FALSE: Wenn der Motor läuft, löst der Antriebsverstärker einen Quick Stop aus. Der Ausgang q_wDrvComCtrl wird auf 16#0002 gesetzt.</p> <p>TRUE: Normales Verhalten des Funktionsbausteins.</p> <p>Wenn nach einem Quick Stop die Istgeschwindigkeit und der Iststrom den Wert Null erreicht haben und wenn sowohl Forward als auch Backward FALSE sind, geht der Antriebsverstärker automatisch in den Betriebszustand "Switched On".</p> <p>Um den Motor neu zu starten, muss der Quick Stop deaktiviert werden (i_xQckStop auf TRUE setzen).</p>
i_xFreeWhl	BOOL	<p>Wertebereich: FALSE, TRUE Initialwert: FALSE</p> <p>FALSE: Wenn der Motor läuft, löst der Antriebsverstärker einen "Free Wheel Stop" aus. Der Ausgang q_wDrvComCtrl wird auf 16#0000 gesetzt.</p> <p>TRUE: Normales Verhalten des Funktionsbausteins.</p>
i_xFltRst	BOOL	<p>Wertebereich: FALSE, TRUE Initialwert: FALSE</p> <p>FALSE: Normales Verhalten des Funktionsbausteins.</p> <p>TRUE: Der Ausgang q_wDrvComCtrl wird auf 16#0080 gesetzt.</p>

Eingang	Datentyp	Beschreibung
i_wDrvStat	WORD	Wertebereich: Initialwert: 0 Muss direkt auf das Statuswort des Antriebsverstärkers gemappt werden (CANopen-Objekt 6041). Dieser Wert darf zwischen dem CANopen-Interface und dem Funktionsbaustein nicht geändert werden.
i_wSpdRef	WORD	Wertebereich: Initialwert: 0 Sollgeschwindigkeit für den Antriebsverstärker. Wird direkt in die Zielgeschwindigkeit $q_wSpdTarg$ kopiert, wenn der Funktionsbaustein aktiviert wird und wenn über die Visualisierung kein bestimmter Wert vorgegeben ist.
i_wAcc	WORD	Wertebereich: - Initialwert: - Beschleunigung: Wird direkt auf den Ausgang q_wAcc kopiert, wenn der Funktionsbaustein aktiviert wird.
i_wDec	WORD	Wertebereich: - Initialwert: - Verzögerung: Wird direkt auf den Ausgang q_wDec kopiert, wenn der Funktionsbaustein aktiviert wird.
i_iActSpd	INT	Wertebereich: Initialwert: 0 Istgeschwindigkeit des Antriebs. Dieser Eingang muss direkt auf das PDO gemappt werden (CANopen-Objekt 2002 / Subindex 03, um die Geschwindigkeit in 0,1 Hz zu erhalten).
i_wActCur	WORD	Wertebereich: Initialwert: 16#FFFF Iststrom des Antriebsverstärkers. Dieser Eingang muss direkt auf das PDO gemappt werden (CANopen-Objekt 2002 / Subindex 05, Einheit 0,1 A). Dieser Eingang wird verwendet, um zu erkennen, ob der Motorstromwert gleich Null oder ungleich Null ist.

Folgende Tabelle zeigt die Ausgänge.

Ausgang	Datentyp	Beschreibung
q_xEn	BOOL	Wertebereich: FALSE, TRUE Initialwert: FALSE Funktionsbaustein aktiviert/deaktiviert. Direkte Kopie von i_xEn.
q_wDrvComCtrl	WORD	Wertebereich: Initialwert: Muss direkt auf das Steuerwort des Antriebsverstärkers gemappt werden (CANopen-Objekt 6040). Dieser Wert darf zwischen dem CANopen-Interface und dem Funktionsbaustein nicht geändert werden.
q_wSpdTarg	WORD	Wertebereich: Initialwert: Zielgeschwindigkeit für den Antriebsverstärker. Wird direkt von der Sollgeschwindigkeit i_wSpdRef kopiert, wenn der Funktionsbaustein aktiviert wird und wenn die Visualisierung keinen bestimmten Wert vorgibt. Die Zielgeschwindigkeit wird auf 0 gesetzt, wenn der Funktionsbaustein nicht aktiviert ist. Dieser Ausgang muss direkt auf das PDO gemappt werden (CANopen-Objekt 6042 zur Übertragung in Umdrehungen pro Minute).
q_xAlrm	BOOL	Wertebereich: FALSE, TRUE Initialwert: FALSE Wird auf FALSE gesetzt, wenn der Funktionsbaustein deaktiviert wird und wenn der Antriebsverstärker in den Betriebszustand "Switch On Disabled" geht (siehe Zustandsdiagramm des Antriebsprofils). Wird auf TRUE gesetzt, wenn ein Fehler vom Antriebsverstärker erkannt wird (Bit 3 des Statusworts).
q_wAcc	WORD	Wertebereich: - Initialwert: - Beschleunigung: Wird direkt vom Eingang i_wAcc kopiert, wenn der Funktionsbaustein aktiviert wird. Dieser Ausgang muss direkt auf das PDO gemappt werden (CANopen-Objekt 203C:2).
q_wDec	WORD	Wertebereich: - Initialwert: - Verzögerung: Wird direkt vom Eingang i_wDec kopiert, wenn der Funktionsbaustein aktiviert wird. Dieser Ausgang muss direkt auf das PDO gemappt werden (CANopen-Objekt 203C:3).

Hinweise

▲ WARNUNG**UNBEABSICHTIGTES VERHALTEN DURCH INKONSISTENTE BEFEHLE**

Wenn Sie diesen Funktionsbaustein aktiviert haben, führt die gleichzeitige Verwendung anderer Funktionsbausteine der Bibliothek zu unbeabsichtigtem Verhalten.

- Aktivieren Sie diesen Funktionsbaustein nur, wenn alle anderen Funktionsbausteine der Bibliothek inaktiv sind.
- Deaktivieren Sie diesen Funktionsbaustein, bevor Sie andere Funktionsbausteine der Bibliothek aktivieren.

Nichtbeachtung dieser Vorkehrungen kann zu Tod, schwerwiegenden Verletzungen oder Materialschäden führen.

Beachten Sie folgende Punkte:

- Der Antriebsverstärker verlässt nach einem "Quick Stop" automatisch den Betriebszustand "Quick Stop Active" (siehe Zustandsdiagramm weiter unten), sobald die Istgeschwindigkeit und der Iststrom den Wert Null erreicht haben und wenn sowohl Forward als auch Backward FALSE sind. Um den Motor neu zu starten, muss der Quick Stop deaktiviert werden (`i_xQckStop` auf TRUE setzen).
- Ein "Quick Stop" hat eine höhere Priorität als ein regulärer Stopp ("Forward" und "Reverse" auf FALSE).
- Ein "Free Wheel Stop" hat eine höhere Priorität als ein "Quick Stop".
- Wenn auf der 7-Segment-Anzeige des Antriebsverstärkers nach dem Download einer Anwendung auf den Antriebsverstärker die blinkende Meldung $\text{E}oF$ erscheint, ist eine steigende und dann eine fallende Flanke am Eingang Fault Reset (`i_xFltRst`) erforderlich, um die korrekte Kommunikation mit dem Antriebsverstärker neu zu starten.

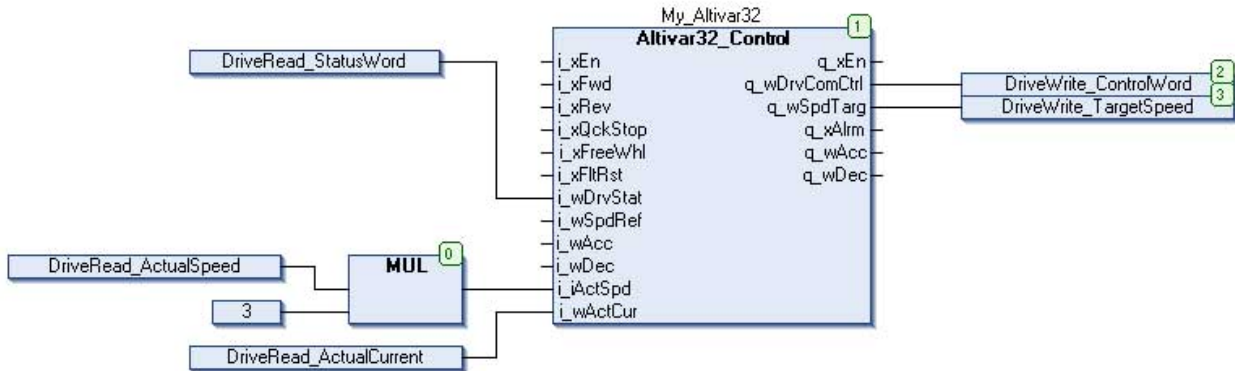
Verwendung des Funktionsbausteins

Vorgehen zum Starten des Funktionsbausteins mit den Voreinstellungen:

Schritt	Aktion
1	Statuswort und Steuerwort auf PDOs mappen: <ul style="list-style-type: none"> • <code>i_wDrvStat</code> auf PDO vom Antriebsverstärker zur Steuerung mappen • <code>q_wDrvComCtrl</code> auf PDO von Steuerung zum Antriebsverstärker mappen
2	Istgeschwindigkeit und Iststrom auf PDO vom Antriebsverstärker zur Steuerung mappen: <ul style="list-style-type: none"> • <code>i_wActCur</code> • <code>i_iActSpd</code>
3	"Free Wheel" deaktivieren: <code>i_xFreeWhl</code> auf TRUE setzen.
4	"Quick Stop" deaktivieren: <code>i_xQckStop</code> auf TRUE setzen.
5	Funktionsbaustein aktivieren: <code>i_xEn</code> auf TRUE setzen.
6	Sollgeschwindigkeit einstellen: <code>i_wSpdRef</code> auf einen Wert ungleich Null setzen.
7	Bewegung in positive ("Forward") oder negative ("Reverse") Richtung starten: <code>i_xFwd</code> oder <code>i_xRev</code> auf TRUE setzen.

Direktes Mapping PDOs - CANopen

Das Interface für das direkte Mapping von PDOs auf CANopen sieht folgendermaßen aus:



Die Istgeschwindigkeit wird in 0,1 Hz gelesen (von 0 bis 500, wenn HSP = 50 Hz)

Die Frequenz muss:

- durch 10 dividiert werden (um die Geschwindigkeit in Hz zu erhalten)
- durch 2 dividiert werden (2 Polpaare)
- mit 60 multipliziert werden (60 um die Geschwindigkeit in Umdrehungen pro Minute zu erhalten)

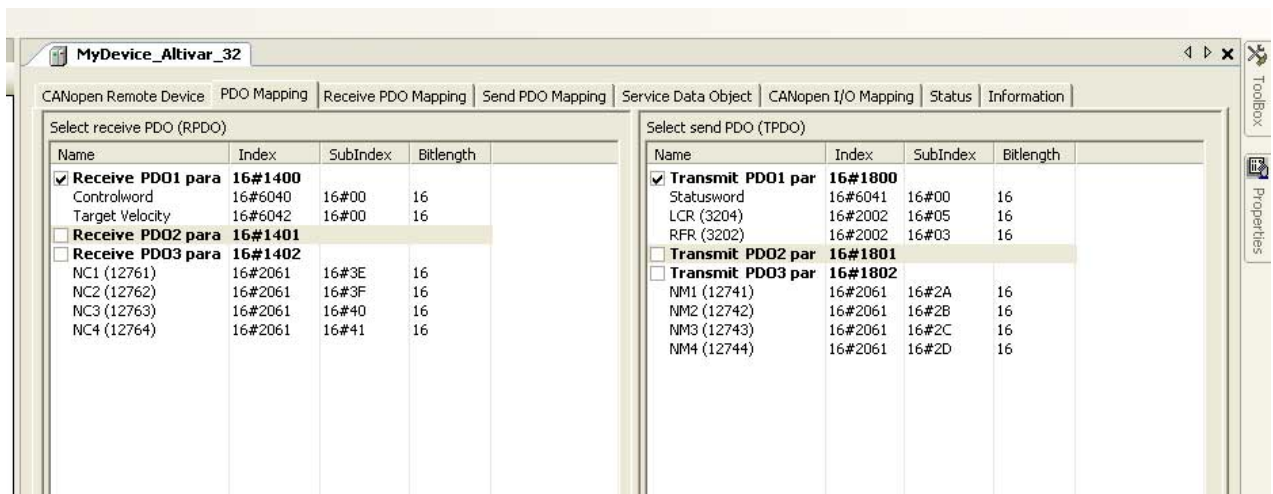
Bei Motoren mit 2 Polpaaren kann der Rechenweg verkürzt werden:

Die Istgeschwindigkeit multipliziert mit 3 konvertiert 0,1 Hz zu rpm (min⁻¹).

```

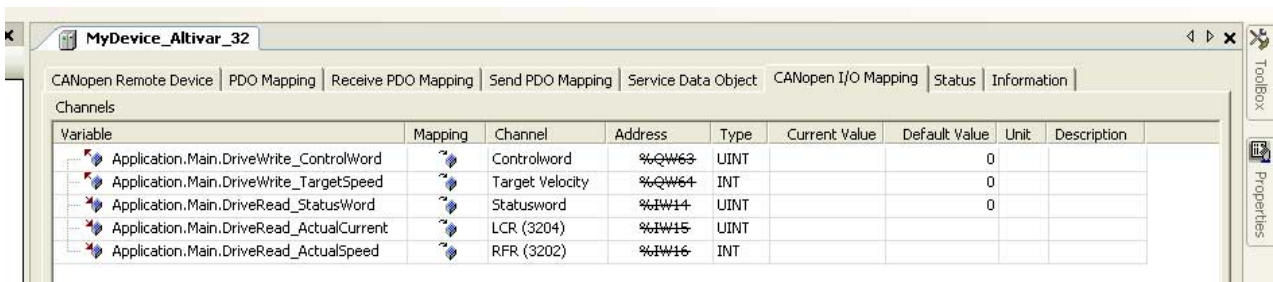
1 PROGRAM PLC_PRG
2
3 VAR
4   My_Altivar31: Altivar31_Control;
5   DriveRead_ActualSpeed: INT;
6   DriveRead_ActualCurrent: WORD;
7   DriveRead_StatusWord: WORD;
8   DriveWrite_ControlWord: WORD;
9   DriveWrite_TargetSpeed: WORD;
10 END_VAR
    
```

PDO Konfiguration:

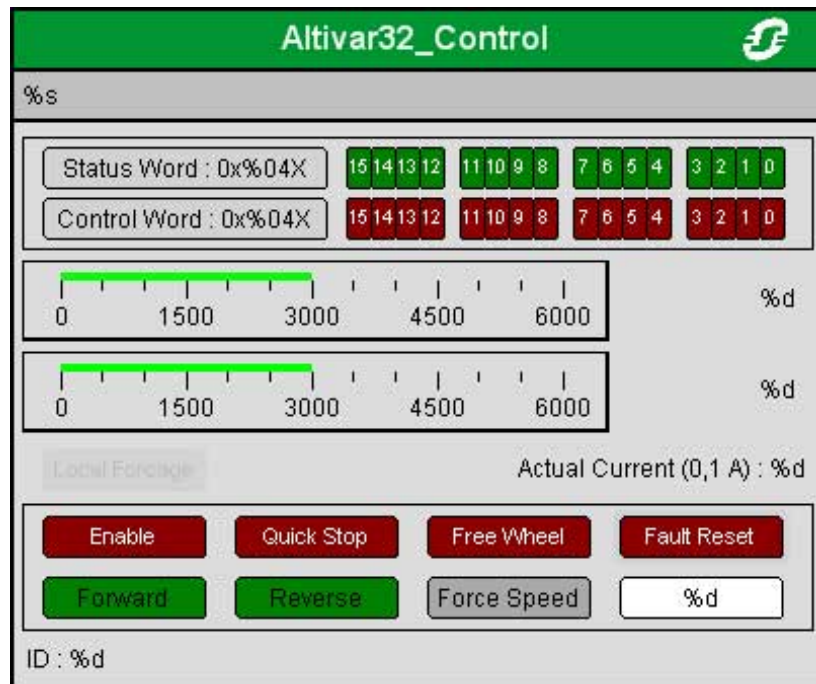


Das Bild zeigt eine Beispielkonfiguration. Andere Konfigurationen der PDOs sind möglich.

Mapping der Daten auf das PDO:

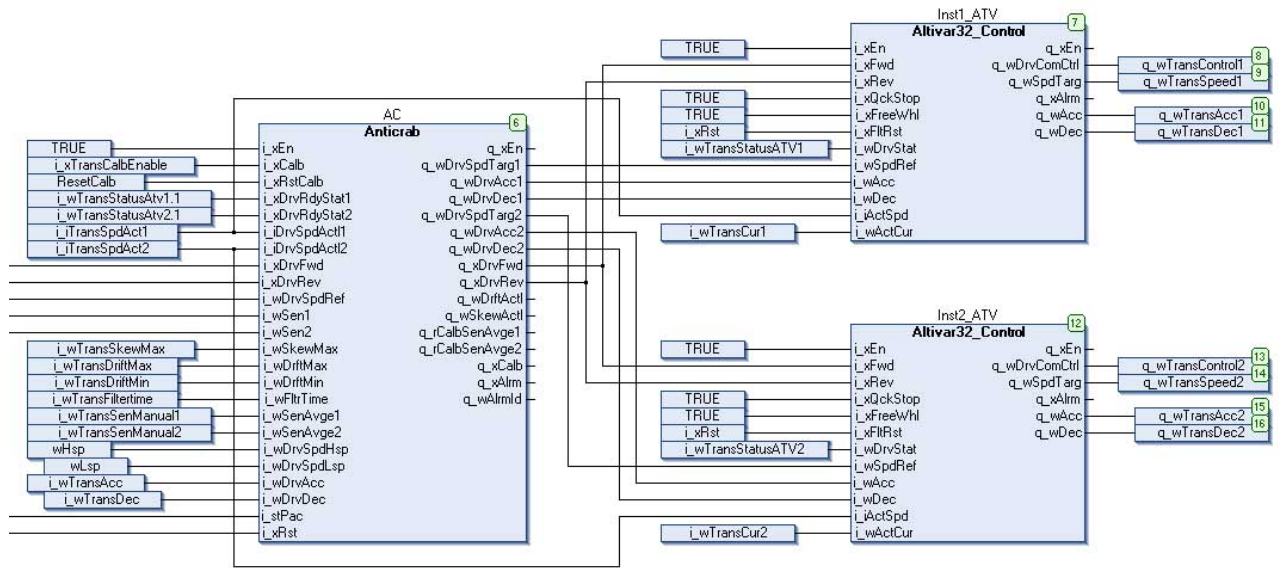


Visualisierung Mit der oben gezeigten Minimalkonfiguration kann die Visualisierung des Funktionsbausteins zur Steuerung des Antriebsverstärkers verwendet werden. Nach dem PDO-Mapping der 5 oben angegebenen Werte kann der Antrieb mit den folgenden Schritten gestartet werden:



Schritt	Aktion
1	Klick auf Schaltfläche "Enable", um den Funktionsbaustein zu aktivieren
2	Klick auf Schaltfläche "Quick Stop", um "Quick Stop" zu deaktivieren
3	Klick auf Schaltfläche "Free Wheel", um "Free Wheel" zu deaktivieren
4	Geschwindigkeitswert ungleich Null (in Umdrehungen pro Minute) eingeben (im Feld neben der Schaltfläche Force Speed).
5	Klick auf Schaltfläche "Force Speed"
6	Klick auf Schaltfläche "Forward" oder "Reverse": Der Motor läuft

Beispiel einer Anwendung des Funktionsbausteins:



3 Glossar

3

3.1 Einheiten und Umrechnungstabellen

Der Wert in der gegebenen Einheit (linke Spalte) wird mit der Formel (im Feld) für die gesuchte Einheit (obere Zeile) berechnet.

Beispiel: Umrechnung von 5 Meter [m] nach Yard [yd]
 $5 \text{ m} / 0,9144 = 5,468 \text{ yd}$

3.1.1 Länge

	in	ft	yd	m	cm	mm
in	-	/ 12	/ 36	* 0,0254	* 2,54	* 25,4
ft	* 12	-	/ 3	* 0,30479	* 30,479	* 304,79
yd	* 36	* 3	-	* 0,9144	* 91,44	* 914,4
m	/ 0,0254	/ 0,30479	/ 0,9144	-	* 100	* 1000
cm	/ 2,54	/ 30,479	/ 91,44	/ 100	-	* 10
mm	/ 25,4	/ 304,79	/ 914,4	/ 1000	/ 10	-

3.1.2 Masse

	lb	oz	slug	kg	g
lb	-	* 16	* 0,03108095	* 0,4535924	* 453,5924
oz	/ 16	-	* $1,942559 \cdot 10^{-3}$	* 0,02834952	* 28,34952
slug	/ 0,03108095	/ $1,942559 \cdot 10^{-3}$	-	* 14,5939	* 14593,9
kg	/ 0,453592370	/ 0,02834952	/ 14,5939	-	* 1000
g	/ 453,592370	/ 28,34952	/ 14593,9	/ 1000	-

3.1.3 Kraft

	lb	oz	p	dyne	N
lb	-	* 16	* 453,55358	* 444822,2	* 4,448222
oz	/ 16	-	* 28,349524	* 27801	* 0,27801
p	/ 453,55358	/ 28,349524	-	* 980,7	* $9,807 \cdot 10^{-3}$
dyne	/ 444822,2	/ 27801	/ 980,7	-	/ $100 \cdot 10^3$
N	/ 4,448222	/ 0,27801	/ $9,807 \cdot 10^{-3}$	* $100 \cdot 10^3$	-

3.1.4 Leistung

	HP	W
HP	-	* 746
W	/ 746	-

3.1.5 Rotation

	min ⁻¹ (RPM)	rad/s	deg./s
min ⁻¹ (RPM)	-	* $\pi / 30$	* 6
rad/s	* $30 / \pi$	-	* 57,295
deg./s	/ 6	/ 57,295	-

3.1.6 Drehmoment

	lb·in	lb·ft	oz·in	Nm	kp·m	kp·cm	dyne·cm
lb·in	-	/ 12	* 16	* 0,112985	* 0,011521	* 1,1521	* $1,129 \cdot 10^6$
lb·ft	* 12	-	* 192	* 1,355822	* 0,138255	* 13,8255	* $13,558 \cdot 10^6$
oz·in	/ 16	/ 192	-	* $7,0616 \cdot 10^{-3}$	* $720,07 \cdot 10^{-6}$	* $72,007 \cdot 10^{-3}$	* 70615,5
Nm	/ 0,112985	/ 1,355822	/ $7,0616 \cdot 10^{-3}$	-	* 0,101972	* 10,1972	* $10 \cdot 10^6$
kp·m	/ 0,011521	/ 0,138255	/ $720,07 \cdot 10^{-6}$	/ 0,101972	-	* 100	* $98,066 \cdot 10^6$
kp·cm	/ 1,1521	/ 13,8255	/ $72,007 \cdot 10^{-3}$	/ 10,1972	/ 100	-	* $0,9806 \cdot 10^6$
dyne·cm	/ $1,129 \cdot 10^6$	/ $13,558 \cdot 10^6$	/ 70615,5	/ $10 \cdot 10^6$	/ $98,066 \cdot 10^6$	/ $0,9806 \cdot 10^6$	-

3.1.7 Trägheitsmoment

	lb·in ²	lb·ft ²	kg·m ²	kg·cm ²	kp·cm·s ²	oz·in ²
lb·in ²	-	/ 144	/ 3417,16	/ 0,341716	/ 335,109	* 16
lb·ft ²	* 144	-	* 0,04214	* 421,4	* 0,429711	* 2304
kg·m ²	* 3417,16	/ 0,04214	-	* $10 \cdot 10^3$	* 10,1972	* 54674
kg·cm ²	* 0,341716	/ 421,4	/ $10 \cdot 10^3$	-	/ 980,665	* 5,46
kp·cm·s ²	* 335,109	/ 0,429711	/ 10,1972	* 980,665	-	* 5361,74
oz·in ²	/ 16	/ 2304	/ 54674	/ 5,46	/ 5361,74	-

3.1.8 Temperatur

	°F	°C	K
°F	-	(°F - 32) * 5/9	(°F - 32) * 5/9 + 273,15
°C	°C * 9/5 + 32	-	°C + 273,15
K	(K - 273,15) * 9/5 + 32	K - 273,15	-

3.1.9 Leiterquerschnitt

AWG	1	2	3	4	5	6	7	8	9	10	11	12	13
mm ²	42,4	33,6	26,7	21,2	16,8	13,3	10,5	8,4	6,6	5,3	4,2	3,3	2,6

AWG	14	15	16	17	18	19	20	21	22	23	24	25	26
mm ²	2,1	1,7	1,3	1,0	0,82	0,65	0,52	0,41	0,33	0,26	0,20	0,16	0,13

3.2 Begriffe und Abkürzungen

	Hinweise auf einschlägige Normen, die vielen Begriffen zugrunde liegen, finden Sie in Kapitel "1.5 Normen und Begrifflichkeiten". Einige Begriffe und Abkürzungen haben je nach Norm spezifische Bedeutungen.
<i>Asynchroner Fehler</i>	Asynchrone Fehler werden ohne Anfrage gemeldet. Beispiel für einen asynchronen Fehler: Übertemperatur der Endstufe.
<i>Drehrichtung</i>	Drehung der Motorwelle in positive oder negative Drehrichtung. Positive Drehrichtung gilt bei Drehung der Motorwelle im Uhrzeigersinn, wenn man auf die Stirnfläche der herausgeführten Motorwelle blickt.
<i>Endschalter</i>	Schalter, die das Verlassen des zulässigen Verfahrbereichs melden.
<i>Endstufe</i>	Hierüber wird der Motor angesteuert. Die Endstufe erzeugt entsprechend den Positionersignalen der Steuerung Ströme zur Ansteuerung des Motors.
<i>Fataler Fehler</i>	Bei einem fatalen Fehler ist das Produkt nicht mehr in der Lage, den Motor anzusteuern, so dass ein sofortiges Deaktivieren der Endstufe erforderlich wird.
<i>Fault</i>	Fault beschreibt einen Zustand, der durch einen Fehler hervorgerufen werden kann. Weitere Informationen finden Sie in entsprechende Normen und Standards, zum Beispiel IEC 61800-7, ODVA Common Industrial Protocol (CIP).
<i>Fault reset</i>	Eine Funktion, mit der ein Antrieb nach einem erkannten Fehler wieder in den regulären Betriebszustand versetzt wird, nachdem die Fehlerursache beseitigt worden ist und der Fehler nicht mehr ansteht.
<i>Fehler</i>	Diskrepanz zwischen einem erkannten (berechneten, gemessenen oder per Signal übermittelten) Wert oder Zustand und dem vorgesehenen oder theoretisch korrekten Wert beziehungsweise Zustand.
<i>Fehlerklasse</i>	Klassifizierung von Fehlern in Gruppen. Die Einteilung in unterschiedliche Fehlerklassen ermöglicht gezielte Reaktionen auf die Fehler einer Klasse, zum Beispiel nach Schwere eines Fehlers.
<i>Gerätedaten</i>	Gerätedaten sind Parameterwerte eines Gerätes. Die Daten sind persistent im EEPROM-Speicher des Gerätes abgelegt.
<i>LED</i>	Light Emitting Diode (engl.), Leuchtdiode
<i>Node Guarding</i>	(engl.: Knotenüberwachung), Verbindungsüberwachung mit dem Slave an einer Schnittstelle auf zyklischen Datenverkehr.
<i>Parameter</i>	Vom Anwender lesbare und teilweise einstellbare Gerätedaten und Gerätewerte.
<i>RS485</i>	Feldbusschnittstelle nach EIA-485, die eine serieller Datenübertragung mit mehreren Teilnehmern ermöglicht.
<i>Synchroner Fehler</i>	Fehler, der von der Steuerung gemeldet wird, wenn sie einen Befehl des Masters nicht ausführen kann.
<i>Warnung</i>	Bei einer Warnung außerhalb des Kontextes von Sicherheitshinweisen handelt es sich um einen Hinweis auf ein potentiell Problem, das durch eine Überwachungsfunktion erkannt wurde. Eine Warnung bewirkt keinen Wechsel des Betriebszustands.
<i>Werkseinstellung</i>	Einstellungen bei Auslieferung des Produkts.

4 Stichwortverzeichnis

4

A		H	
	Abkürzungen 101		Handbücher
	Altivar_Startup 57		Bezugsquelle 7
	Altivar31_Control 66	M	
	Altivar32_Control 88		MC_Jog_ATV 25
	Altivar71_Control 77		MC_MoveVelocity_ATV 29
B			MC_Power_ATV 24
	Begriffe 101		MC_ReadActualTorque_ATV 33
	Bestimmungsgemäße Verwendung 9		MC_ReadActualVelocity_ATV 32
	Bevor Sie beginnen		MC_ReadAxisError_ATV 52
	Sicherheitsinformationen 9		MC_ReadDigitalInput_ATV 47
	Bezugsquelle		MC_ReadDigitalOutput_ATV 49
	Handbücher 7		MC_ReadParameter_ATV 36
D			MC_ReadStatus_ATV 34
	DownloadDriveParameter_ATV 45		MC_Reset_ATV 56
E			MC_Stop_ATV 31
	Einheiten und Umrechnungstabellen 99		MC_WriteDigitalOutput_ATV 51
F		Q	
	Fehlercode 52, 62		Qualifikation des Personals 9
G		R	
	Gefahrenklassen 10		ReadAnalogInput_ATV 46
	Gerätedaten 101		Rechtliche Hinweise 8
	GetSupplierVersion 37		ResetParameters_ATV 42
	Glossar 99	S	
	Gültigkeitsbereich 7		SetDriveRamp_ATV 40

SetFrequencyRange_ATV	41	VelocityControlSelectAI_ATV	28
StoreParameters_ATV	43	Z	
U		Ziel dieses Dokuments	7
UploadDriveParameter_ATV	44		
V			
VelocityControlAnalogInput_ATV	27		