

EtherCAT®

TwinCAT 3 – Funktionsbausteine für Antriebsregler
der 6. Generation
Handbuch

de
05/2024
ID 443370.01

Inhaltsverzeichnis

Inhaltsverzeichnis	2
1 Vorwort	4
2 Benutzerinformationen	5
2.1 Aktualität	5
2.2 Originalsprache.....	5
2.3 Beschriebenes Produkt.....	5
2.4 Mitgeltende Dokumentationen.....	5
2.5 Haftungsausschluss	5
2.6 Auszeichnung von Textelementen	6
2.7 Marken	6
3 Sicherheitshinweise	7
4 Bibliothek installieren und einem Projekt hinzufügen	8
5 NC-Achse und SPS-Projekt verknüpfen	9
6 Funktionsbausteine	10
6.1 STOBER_BoxName.....	11
6.1.1 Beispiel-Code	12
6.2 STOBER_Backup_Restore_Initiator	12
6.2.1 Beispiel-Code	14
6.3 STOBER_Backup_Restore	14
6.3.1 Funktionsbaustein verwenden	16
6.3.2 Beispiel-Code	18
6.4 STOBER_MC_HOME	20
6.4.1 Ablauf der Referenzierung.....	21
6.4.2 Beispiel-Code	22
6.5 STOBER_MC_HOME_REF	23
6.5.1 Ablauf der Referenzierung.....	24
6.5.2 Beispiel-Code	25
6.6 STOBER_Action.....	26
6.6.1 Beispiele für Aktionen ohne erforderliche Freigabe.....	27
6.6.2 Beispiel-Code	27
6.7 STOBER_Power_Action.....	28
6.7.1 Beispiele für Aktionen mit erforderlicher Freigabe	29
6.7.2 Beispiel-Code	30
6.8 STOBER_Phase_Test.....	31
6.8.1 Ablauf des Phasentests.....	32
6.8.2 Beispiel-Code	33

6.9	STOBER_PRM_LoadMatrix	34
6.9.1	Ablauf des Auslesens und Schreibens.....	35
6.9.2	Beispiel-Code.....	35
6.10	STOBER_PRM_LoadMatrix_AMS.....	36
6.10.1	Ablauf des Auslesens und Schreibens.....	37
6.10.2	Beispiel-Code.....	37
6.11	STOBER_PRM_LoadMatrix_File.....	38
6.11.1	Ablauf des Auslesens und Schreibens.....	39
6.11.2	Beispiel-Code.....	40
6.12	STOBER_PRM_LoadMatrix_File_AMS.....	41
6.12.1	Ablauf des Auslesens und Schreibens.....	42
6.12.2	Beispiel-Code.....	43
6.13	STOBER_SDO_Info.....	44
6.13.1	Beispiel-Code.....	45
7	Berechnung des Index	46
8	Diagnose	47
8.1	eFBError (ENUM).....	48
9	Anhang.....	52
9.1	Weiterführende Informationen.....	52
9.2	Abkürzungen.....	53
10	Kontakt	54
10.1	Beratung, Service, Anschrift	54
10.2	Ihre Meinung ist uns wichtig	54
10.3	Weltweite Kundennähe.....	55
	Tabellenverzeichnis.....	56

1 Vorwort

Die Funktionsbausteine von STÖBER stellen kleine, funktionale Software-Einheiten dar, die Sie bei der Inbetriebnahme Ihrer Antriebsregler und im Service-Fall unterstützen. Sie können Sie in TwinCAT 3 in verschiedenen Projekten wiederverwenden.

Die verfügbaren Funktionsbausteine finden Sie in gepackter Form unter <http://www.stoeber.de/de/downloads/>.
Geben Sie im Suchfeld `TwinCAT 3 Bausteine` ein.

2 Benutzerinformationen

Um die von STÖBER zur Verfügung gestellten Funktionsbausteine effizient nutzen zu können, sollten Ihnen die Netzwerktechnologie EtherCAT und damit verbunden die Beckhoff Automatisierungssysteme, insbesondere die Programmierung mit TwinCAT 3 sowie die Erstellung und Bearbeitung der Hardware-Konfiguration bekannt sein.

2.1 Aktualität

Prüfen Sie, ob Ihnen mit diesem Dokument die aktuelle Version der Dokumentation vorliegt. Auf unserer Webseite stellen wir Ihnen die neuesten Dokumentversionen zu unseren Produkten zum Download zur Verfügung:

<http://www.stoeber.de/de/downloads/>.

2.2 Originalsprache

Die Originalsprache dieser Dokumentation ist Deutsch; alle anderssprachigen Fassungen sind von der Originalsprache abgeleitet.

2.3 Beschriebenes Produkt

Diese Dokumentation ist verbindlich für:

STÖBER Antriebsregler der 6. Generation.

2.4 Mitgeltende Dokumentationen

Diese Dokumentation ergänzt die Handbücher EtherCAT für SD6 oder EtherCAT für SC6 und SI6 sowie damit verbunden das Handbuch zur Applikation CiA 402. Sie dürfen die vorliegende Dokumentation nur in Verbindung mit den genannten Handbüchern verwenden (siehe [Weiterführende Informationen \[► 52\]](#)).

2.5 Haftungsausschluss

Bei der im STÖBER Download-Center zur Verfügung gestellten Bibliothek und den darin enthaltenen Funktionsbausteinen für TwinCAT 3 handelt es sich um einen kostenlosen Service.

Für deren Inhalt, Funktion und Anwendbarkeit in einer konkreten Maschine oder Anwendung übernimmt STÖBER keine Haftung.

2.6 Auszeichnung von Textelementen

Bestimmte Elemente des Fließtexts werden wie folgt ausgezeichnet.

Wichtige Information	Wörter oder Ausdrücke mit besonderer Bedeutung
Interpolated position mode	Optional: Datei-, Produkt- oder sonstige Namen
<u>Weiterführende Informationen</u>	Interner Querverweis
http://www.musterlink.de	Externer Querverweis

Software- und Display-Anzeigen

Um den unterschiedlichen Informationsgehalt von Elementen, die von der Software-Oberfläche oder dem Display eines Antriebsreglers zitiert werden sowie eventuelle Benutzereingaben entsprechend kenntlich zu machen, werden folgende Darstellungen verwendet.

Hauptmenü Einstellungen	Von der Oberfläche zitierte Fenster-, Dialog-, Seitennamen oder Schaltflächen, zusammengesetzte Eigennamen, Funktionen
Wählen Sie Referenziermethode A	Vorgegebene Eingabe
Hinterlegen Sie Ihre <Eigene IP-Adresse>	Benutzerdefinierte Eingabe
EREIGNIS 52: KOMMUNIKATION	Display-Anzeigen (Status, Meldungen, Warnungen, Störungen)

Tastenkürzel und Befehlsfolgen oder Pfade sind folgendermaßen dargestellt.

[Strg], [Strg] + [S]	Taste, Tastenkombination
Tabelle > Tabelle einfügen	Navigation zu Menüs/Untermenüs (Pfadangabe)

2.7 Marken

Die folgenden Namen, die in Verbindung mit dem Gerät, seiner optionalen Ausstattung und seinem Zubehör verwendet werden, sind Marken oder eingetragene Marken anderer Unternehmen:

CANopen [®] , CiA [®]	CANopen [®] und CiA [®] sind eingetragene Unionsmarken des CAN in AUTOMATION e.V., Nürnberg, Deutschland.
EtherCAT [®] , Safety over EtherCAT [®]	EtherCAT [®] und Safety over EtherCAT [®] sind eingetragene Marken und patentierte Technologien, lizenziert durch die Beckhoff Automation GmbH, Deutschland.
TwinCAT [®]	TwinCAT [®] ist eine eingetragene und lizenzierte Marke der Beckhoff Automation GmbH, Deutschland.
Windows [®] , Windows [®] 7, Windows [®] 10, Windows [®] 11	Windows [®] , das Windows [®] -Logo, Windows [®] XP, Windows [®] 7, Windows [®] 10 und Windows [®] 11 sind eingetragene Marken der Microsoft Corporation in den USA und/oder anderen Ländern.

Alle anderen, hier nicht aufgeführten Marken, sind Eigentum ihrer jeweiligen Inhaber.

Erzeugnisse, die als Marken eingetragen sind, sind in dieser Dokumentation nicht besonders kenntlich gemacht. Vorliegende Schutzrechte (Patente, Warenzeichen, Gebrauchsmusterschutz) sind zu beachten.

3 Sicherheitshinweise

WARNUNG!

Lebensgefahr bei Nichtbeachtung von Sicherheitshinweisen und Restrisiken!

Bei Nichtbeachtung der Sicherheitshinweise und Restrisiken in der Dokumentation des Antriebsreglers können Unfälle mit schweren Verletzungen oder Tod auftreten.

- Halten Sie die Sicherheitshinweise in der Antriebsregler-Dokumentation ein.
 - Berücksichtigen Sie bei der Risikobeurteilung für die Maschine oder Anlage die Restrisiken.
-

WARNUNG!

Fehlfunktion der Maschine infolge fehlerhafter oder veränderter Parametrierung!

Bei fehlerhafter oder veränderter Parametrierung können Fehlfunktionen an Maschinen oder Anlagen auftreten, die zu schweren Verletzungen oder Tod führen können.

- Beachten Sie die Security-Hinweise in der Antriebsregler-Dokumentation.
 - Schützen Sie z. B. die Parametrierung vor unbefugtem Zugriff.
 - Treffen Sie geeignete Maßnahmen für mögliche Fehlfunktionen (z. B. Not-Aus oder Not-Halt).
-

4 Bibliothek installieren und einem Projekt hinzufügen

Wenn Sie Funktionsbausteine von STÖBER verwenden möchten, müssen Sie diese in TwinCAT 3 als Bibliothek installieren und Ihrem Projekt hinzufügen.

Bibliothek installieren

1. Navigieren Sie im Solution Explorer zu Ihrem SPS-Projekt > References.
2. Klicken Sie im Hauptfenster auf **Add library**.
⇒ Das Fenster **Add library** öffnet sich.
3. Klicken Sie auf **Advanced...**
⇒ Ein weiteres Fenster **Add library** öffnet sich.
4. Klicken Sie auf **Library Repository...**
5. Das Fenster **Library Repository** öffnet sich.
6. Klicken Sie auf **Install...**, navigieren Sie zu der zu installierenden Bibliothek und klicken Sie auf **Open**.
⇒ Die ausgewählte Bibliothek wird in das Bibliotheksrepository installiert.

Bibliothek einem Projekt hinzufügen

1. Navigieren Sie im Solution Explorer zu Ihrem SPS-Projekt > References.
2. Klicken Sie im Hauptfenster auf **Add library**.
⇒ Das Fenster **Add library** öffnet sich.
3. Wählen Sie unter **Application > STÖBER Antriebstechnik GmbH + Co. KG** die Bibliothek aus, die Sie hinzufügen möchten und bestätigen Sie mit **OK**.
⇒ Die Bibliothek wird im Solution Explorer Ihrem SPS-Projekt unter **References** hinzugefügt.

5 NC-Achse und SPS-Projekt verknüpfen

Ist die Übergabe einer NC-Achse an den Funktionsbaustein erforderlich, müssen Sie eine Verknüpfung zwischen NC-Achse und SPS-Projekt erstellen. Ob eine Verknüpfung Voraussetzung ist, entnehmen Sie der Beschreibung des jeweiligen Funktionsbausteins.

Variable anlegen

Definieren Sie in Ihrem SPS-Projekt eine Variable vom Typ `AXIS_REF`.

Variable und NC-Achse verknüpfen

✓ Sie haben den Config-Modus aktiviert.

1. Navigieren Sie im Solution Explorer zu `Motion > NC-Task 1 SAF > Axes > Axis1`.
2. Wechseln Sie im Hauptfenster in das Register `Settings`.
3. Wählen Sie `Link To PLC ...`.
⇒ Das Fenster `Select Axis PLC Reference ('Axis 1')` öffnet sich.
4. Wählen Sie aus der Liste die zuvor im SPS-Projekt angelegte Variable vom Typ `AXIS_REF` aus und bestätigen Sie mit `OK`.
⇒ Variable und NC-Achse sind verknüpft.

6 Funktionsbausteine

Nachfolgende Tabelle liefert Ihnen eine Übersicht über die verfügbaren Funktionsbausteine.

Funktionsbaustein	Beschreibung	Software-Version	Bibliotheksversion
STOBER_BoxName [► 11]	Name des EtherCAT SubDevices in Parameter A251 des Antriebsreglers schreiben	Ab V 3.1.4022.22	Ab V 3.1.0.0
STOBER_Backup_Restore_Initiator [► 12]	Mögliche Services für den Funktionsbaustein STOBER_Backup_Restore ermitteln	Ab V 3.1.4024.40	Ab V 3.1.2.0
STOBER_Backup_Restore [► 14]	Projektierungen aus TwinCAT 3 in die Antriebsregler laden	Ab V 3.1.4022.22	Ab V 3.1.1.0
STOBER_MC_HOME [► 20]	Antriebsreglergeführte Referenzierung der Applikationen CiA 402 und CiA 402 HiRes Motion steuern (mit Übergabe einer NC-Achse)	Ab V 3.1.4022.22	Ab V 3.1.0.0
STOBER_MC_HOME_REF [► 23]	Antriebsreglergeführte Referenzierung der Applikationen CiA 402 und CiA 402 HiRes Motion steuern (mit Übergabe von Referenzposition, Referenziermethode sowie einer NC-Achse)	Ab V 3.1.4024.40	Ab V 3.1.2.0
STOBER_Action [► 26]	Aktionen auf dem Antriebsregler ausführen	Ab V 3.1.4024.40	Ab V 3.1.2.0
STOBER_Power_Action [► 28]	Aktionen auf dem Antriebsregler ausführen nach vorheriger Freigabe des Antriebsreglers (mit Übergabe einer NC-Achse)	Ab V 3.1.4024.40	Ab V 3.1.2.0
STOBER_Phase_Test [► 31]	Aktion Phasentest auf dem Antriebsregler ausführen (mit Übergabe einer NC-Achse)	Ab V 3.1.4024.40	Ab V 3.1.2.0
STOBER_PRM_LoadMatrix [► 34]	Last-Matrix aus dem Antriebsregler auslesen (R118) und in ein Array schreiben (mit Übergabe einer NC-Achse)	Ab V 3.1.4024.40	Ab V 3.1.2.0
STOBER_PRM_LoadMatrix_AMS [► 36]	Last-Matrix aus dem Antriebsregler auslesen (R118) und in ein Array schreiben	Ab V 3.1.4024.40	Ab V 3.1.2.0
STOBER_PRM_LoadMatrix_File [► 38]	Last-Matrix aus dem Antriebsregler auslesen (R118) und als Datei in ein Verzeichnis legen (mit Übergabe einer NC-Achse)	Ab V 3.1.4024.40	Ab V 3.1.2.0
STOBER_PRM_LoadMatrix_File_AMS [► 41]	Last-Matrix aus dem Antriebsregler auslesen (R118) und als Datei in ein Verzeichnis legen	Ab V 3.1.4024.40	Ab V 3.1.2.3
STOBER_SDO_Info [► 44]	Ermitteln, ob Service SDO Info im Antriebsregler aktiv ist	Ab V 3.1.4024.40	Ab V 3.1.2.0

Tab. 1: Funktionsbausteine für TwinCAT 3

6.1 STÖBER_BoxName

Durch das Ausführen des Funktionsbausteins STÖBER_BoxName in Ihrem TwinCAT-Projekt wird der in TwinCAT 3 vergebene Name des EtherCAT SubDevices automatisch in Parameter A251 des Antriebsreglers geschrieben. Dieser Vorgang wird für alle STÖBER Antriebsregler im Projekt mit einmaligem Ausführen des Funktionsbausteins durchgeführt. Dies erleichtert Ihnen bei der Konfiguration der Antriebsregler in der DriveControlSuite die Zuordnung zu den in TwinCAT 3 projektierten Antriebsreglern. Die Übertragung des Namens an das EtherCAT SubDevice erfolgt über den SDO-Datenaustausch.

Voraussetzungen

- TwinCAT 3 ab Version 3.1.4022.22
- Bibliothek STÖBER_G6_Util ab Version 3.1.0.0

Information

Achten Sie bei der Verwendung des Funktionsbausteins darauf, dass bei der Inbetriebnahme die tatsächlich verwendete Hardware mit der in TwinCAT 3 konfigurierten Topologie übereinstimmen muss. Stimmen Hardware und Netzwerk-Topologie in TwinCAT 3 nicht überein, führt dies zu Fehlfunktionen des Funktionsbausteins.

Parameter



Abb. 1: Funktionsbaustein STÖBER_BoxName: Ein- und Ausgangsparameter

Parameter	Datentyp	Deklaration	Beschreibung
Execute	BOOL	IN	Aktivieren des Funktionsbausteins mit steigender Flanke
AmsNetId_EtherCAT_MainDevice	T_AmsNetID	IN	AMS NetID des EtherCAT MainDevices
Busy	BOOL	OUT	Zustand des Funktionsbausteins (Busy = True: Schreiben noch nicht beendet)
Done	BOOL	OUT	Zustand des Funktionsbausteins (Done = True: Schreiben erfolgreich abgeschlossen)
Error	BOOL	OUT	Zustand des Funktionsbausteins (Error = True: Schreiben fehlerhaft)
ErrorID	UDINT	OUT	Beckhoff-spezifischer ADS Error Code der intern verwendeten Funktionsbausteine
FBEErrorID	eFBError	OUT	Bausteinspezifischer Fehler-Code (siehe eFBError (ENUM) [► 48])

Tab. 2: Funktionsbaustein STÖBER_BoxName: Parameter

Informationen zu den Datentypen entnehmen Sie der Dokumentation zu TwinCAT 3 unter https://infosys.beckhoff.com/content/1031/tc3_plc_intro/2529388939.html.

6.1.1 Beispiel-Code

Das folgende Beispiel dient der Implementierung in Strukturiertem Text (ST).

```
PROGRAM MAIN
VAR
    fbBoxname:STOBER_BoxName;
    bExecuteBox: BOOL;
    bError: BOOL;
    bBusy: BOOL;
    bDone: BOOL;
    uiErrorID: UDINT;
    uiFbErrorID: eFBError;
END_VAR

fbBoxname (
    AmsNetId_EtherCAT_MainDevice:='172.18.132.104.2.1',
    Execute:=bExecuteBox,
    Error=>bError,
    Busy=>bBusy,
    Done=>bDone,
    ErrorID=>uiErrorID,
    FBErrorID=>uiFbErrorID);
```

6.2 STOBER_Backup_Restore_Initiator

Über den Funktionsbaustein STOBER_Backup_Restore_Initiator können die möglichen Services für den Funktionsbaustein STOBER_Backup_Restore ermittelt werden. Der Funktionsbaustein überprüft, ob ein Backup durchgeführt werden kann oder ob ein Restore erforderlich ist. Anhand des Ausgangs iAction des Funktionsbausteins können Sie den Service für den Eingang Servicetype des Funktionsbausteins STOBER_Backup_Restore bestimmen.

Voraussetzungen

- TwinCAT 3 ab Version 3.1.4024.40
- Bibliothek STOBER_G6_Util ab Version 3.1.2.0
- Zielplattform: PC oder CX mit Betriebssystem Windows Embedded Standard (WES) 7, Windows 7 oder Windows 10
- Engineering-PC mit TwinCAT 3 Engineering-Umgebung (XAE) und externe EtherCAT-Steuerung mit TwinCAT 3 Laufzeitumgebung (XAR)

Parameter

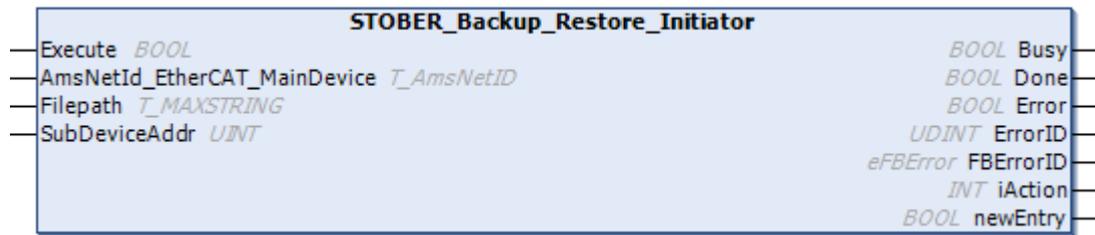


Abb. 2: Funktionsbaustein STOBER_Backup_Restore_Initiator: Ein- und Ausgangsparameter

Parameter	Datentyp	Deklaration	Beschreibung
Execute	BOOL	IN	Aktivieren des Funktionsbausteins mit steigender Flanke
AmsNetId_EtherCAT_MainDevice	T_AmsNetID	IN	AMS NetID des EtherCAT MainDevices
Filepath	T_MAXSTRING	IN	Dateipfad zum Verzeichnis der Projektdatei (*.ds6) auf der EtherCAT-Steuerung
SubDeviceAddr	UINT	IN	Adresse des EtherCAT SubDevices
Busy	BOOL	OUT	Zustand des Funktionsbausteins (Busy = True: Service noch nicht beendet)
Done	BOOL	OUT	Zustand des Funktionsbausteins (Done = True: Service erfolgreich abgeschlossen)
Error	BOOL	OUT	Zustand des Funktionsbausteins (Error = True: Service fehlerhaft)
ErrorID	UDINT	OUT	Beckhoff-spezifischer ADS Error Code der intern verwendeten Funktionsbausteine
FBErrorID	eFBError	OUT	Bausteinspezifischer Fehler-Code (siehe eFBError (ENUM) [► 48])
iAction	INT	OUT	Aktion, die vom Funktionsbaustein STOBER_Backup_Restore angefordert werden kann: <ul style="list-style-type: none"> ▪ iAction = 0 (Backup, Backup mit Rückdokumentation oder Restore) ▪ iAction = 2 (Restore)
newEntry	BOOL	OUT	Informationen zur DeviceInfo.txt: <ul style="list-style-type: none"> ▪ newEntry = True: Antriebsregler wurde in DeviceInfo.txt aufgenommen oder DeviceInfo.txt wurde neu erstellt ▪ newEntry = False: Antriebsregler ist in DeviceInfo.txt bereits enthalten

Tab. 3: Funktionsbaustein STOBER_Backup_Restore_Initiator: Parameter

Informationen zu den Datentypen entnehmen Sie der Dokumentation zu TwinCAT 3 unter https://infosys.beckhoff.com/content/1031/tc3_plc_intro/2529388939.html.

6.2.1 Beispiel-Code

Das folgende Beispiel dient der Implementierung in Strukturiertem Text (ST).

```
PROGRAM MAIN
VAR
    fbBackup_Init:STOBER_Backup_Restore_Initiator;
    bExecute: BOOL;
    bError: BOOL;
    bBusy: BOOL;
    bDone: BOOL;
    ErrorIDADS: UDINT;
    FbErrorID: STOBER_G6_Util.eFBERROR;
    iAction: INT;
END_VAR

fbBackup_Init(
    AmsNetId_EtherCAT_MainDevice:='192.168.12.50.3.1',
    Execute:=bExecute,
    Filepath:='C:\Transfer\ExampleProject\DS6-Projects',
    SubDeviceAddr:=1004,
    Error=>bError,
    Busy=>bBusy,
    Done=>bDone,
    ErrorID=>ErrorIDADS,
    FBErrorID=>FbErrorID,
    iAction=>iAction);
```

6.3 STOBER_Backup_Restore

Der Funktionsbaustein STOBER_Backup_Restore ermöglicht es, ausgewählte DriveControlSuite-Konfigurationen über TwinCAT 3 von der EtherCAT-Steuerung an den Antriebsregler zu senden oder aus dem Antriebsregler zu lesen. Für die korrekte Zuordnung der in der DriveControlSuite projektierten Antriebsregler zu den projektierten EtherCAT SubDevices in TwinCAT 3 benötigen Sie zusätzlich den Funktionsbaustein STOBER_BoxName.

Um zu ermitteln, welche Services möglich sind, führen Sie vorab den Funktionsbaustein STOBER_Backup_Restore_Initiator aus.

Der Funktionsbaustein STOBER_Backup_Restore greift auf den Skriptmodus der DriveControlSuite zu. Das Sichern oder Wiederherstellen der Konfiguration im Antriebsregler wird ausgeführt, sobald die DriveControlSuite gestartet und eine Online-Verbindung hergestellt ist.

Information

Der Funktionsbaustein führt die Aktion Werte speichern aus (A00).

Voraussetzungen

- TwinCAT 3 ab Version 3.1.4024.40
- Bibliothek STOBER_G6_Util ab Version 3.1.2.0
- Zielplattform: PC oder CX mit Betriebssystem Windows Embedded Standard (WES) 7, Windows 7 oder Windows 10
- Engineering-PC mit TwinCAT 3 Engineering-Umgebung (XAE) und externe EtherCAT-Steuerung mit TwinCAT 3 Laufzeitumgebung (XAR)
- DriveControlSuite ab Version 6.5-K, installiert auf der EtherCAT-Steuerung

Parameter

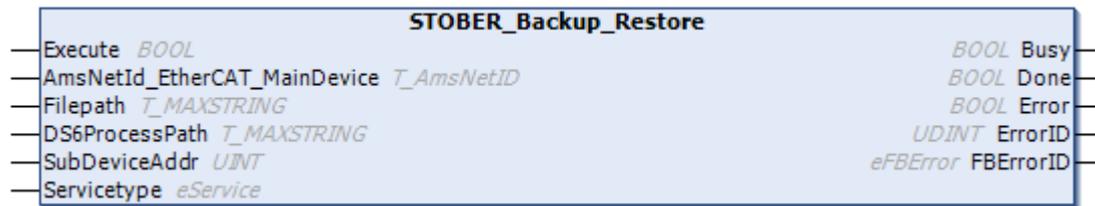


Abb. 3: Funktionsbaustein STOBBER_Backup_Restore: Ein- und Ausgangsparameter

Parameter	Datentyp	Deklaration	Beschreibung
Execute	BOOL	IN	Aktivieren des Funktionsbausteins mit steigender Flanke
AmsNetId_EtherCAT_MainDevice	T_AmsNetID	IN	AMS NetID des EtherCAT MainDevices
Filepath	T_MAXSTRING	IN	Dateipfad zum Verzeichnis der Projektdatei (*.ds6) auf der EtherCAT-Steuerung
DS6ProcessPath	T_MAXSTRING	IN	Dateipfad zur DriveControlSuite (*.exe) auf der EtherCAT-Steuerung, z. B.: C:\Program Files (x86)\STOBER\DriveControlSuite (6.X-X)\bin
SubDeviceAddr	UINT	IN	Adresse des EtherCAT SubDevices
Servicetype	eService	IN	Angefragter Service: <ul style="list-style-type: none"> Backup Backup_RevDocu (Backup mit Rückdokumentation) Restore
Busy	BOOL	OUT	Zustand des Funktionsbausteins (Busy = True: Service noch nicht beendet)
Done	BOOL	OUT	Zustand des Funktionsbausteins (Done = True: Service erfolgreich abgeschlossen)
Error	BOOL	OUT	Zustand des Funktionsbausteins (Error = True: Service fehlerhaft)
ErrorID	UDINT	OUT	Beckhoff-spezifischer ADS Error Code der intern verwendeten Funktionsbausteine
FBErrorID	eFBError	OUT	Bausteinspezifischer Fehler-Code (siehe eFBError (ENUM) [► 48])

Tab. 4: Funktionsbaustein STOBBER_Backup_Restore: Parameter

Informationen zu den Datentypen entnehmen Sie der Dokumentation zu TwinCAT 3 unter https://infosys.beckhoff.com/content/1031/tc3_plc_intro/2529388939.html.

Services

Servicetype	Beschreibung
Backup	Das Projekt wird aus dem Antriebsregler ausgelesen und im Verzeichnis auf der EtherCAT-Steuerung gespeichert.
Backup_RevDocu	Das Projekt wird mit Rückdokumentation aus dem Antriebsregler ausgelesen und im Verzeichnis auf der EtherCAT-Steuerung gespeichert.
Restore	Das Projekt im Verzeichnis auf der EtherCAT-Steuerung wird an den Antriebsregler übertragen und dort gespeichert.

Tab. 5: Funktionsbaustein STOBBER_Backup_Restore: Servicetype

Information

Wird ein Backup-Service ausgeführt und im Verzeichnis auf der EtherCAT-Steuerung ist bereits ein Projekt vorhanden, wird dieses in das Unterverzeichnis SaveOriginals verschoben und mit Datum und Uhrzeit versehen. Das Unterverzeichnis wird automatisch erstellt, sobald ein Backup-Service erstmals ausgeführt wird.

6.3.1 Funktionsbaustein verwenden

Projektieren Sie alle Antriebsregler, tragen Sie die SPS-Gerätenamen ein und speichern Sie Ihr DS6-Projekt auf der EtherCAT-Steuerung. Führen Sie im Anschluss die Funktionsbausteine STOBER_BoxName und STOBER_Backup_Restore aus.

Information

Für jeden Antriebsregler in Ihrem EtherCAT-Netzwerk benötigen Sie eine Instanz des Funktionsbausteins STOBER_Backup_Restore.

DS6-Projekt anlegen und Projektierung auf den Antriebsreglern speichern

1. Erstellen Sie ein neues Projektverzeichnis auf Ihrer EtherCAT-Steuerung.
2. Starten Sie die DriveControlSuite auf der EtherCAT-Steuerung.
3. Erstellen Sie ein Projekt und projektieren Sie alle Antriebsregler in Ihrem EtherCAT-Netzwerk.
4. Übertragen Sie das Projekt an die Antriebsregler und speichern Sie es auf diesen nichtflüchtig ab.

Information

Alternativ können Sie für jeden Antriebsregler in Ihrem EtherCAT-Netzwerk jeweils ein separates DS6-Projekt erstellen.

SPS-Gerätenamen eintragen und DS6-Projekt auf der EtherCAT-Steuerung speichern

1. Wechseln Sie zu TwinCAT XAE und navigieren Sie im Solution Explorer zu einem EtherCAT SubDevice.
2. Doppelklicken Sie auf das EtherCAT SubDevice, um es zu öffnen.
3. Hauptfenster > Register General > Feld Name:
Kopieren Sie den Namen des EtherCAT SubDevices in die Zwischenablage.
4. Wechseln Sie zur DriveControlSuite auf Ihrer EtherCAT-Steuerung.
5. Markieren Sie im Projektbaum den entsprechenden Antriebsregler und klicken Sie im Projektmenü > Bereich Parameterliste auf die gewünschte projektierte Achse.
6. Gruppe A > Parameter A251 SPS-Gerätename:
Fügen Sie den kopierten Namen aus der Zwischenablage ein.
7. Wiederholen Sie die Schritte für alle weiteren Antriebsregler in Ihrem Projekt.
8. Speichern Sie das Projekt in das zuvor auf der EtherCAT-Steuerung erstellte Verzeichnis.
9. Schließen Sie die DriveControlSuite auf der EtherCAT-Steuerung.

Information

Speichern Sie das Projekt erst ab, nachdem Sie mit den Antriebsreglern online verbunden waren. Stellen Sie sicher, dass beim Verbindungsaufbau die Produktionsnummer des jeweiligen Antriebsreglers in Parameter E52[2] eingetragen wurde.

Funktionsbaustein ausführen

Information

Bei der ersten Verwendung des Funktionsbausteins STÖBER_Backup_Restore wird der Service Restore ausgeführt, um das Projekt an die Antriebsregler zu übertragen.

Um zu ermitteln, welche Services möglich sind, führen Sie vorab den Funktionsbaustein STÖBER_Backup_Restore_Initiator aus.

Wird der falsche Service angefordert, gibt der Ausgang FBErrorID des Funktionsbausteins STÖBER_Backup_Restore den Fehler 109: WRONG_SERVICE_TYPE_INPUT aus.

- ✓ Am Eingang Filepath des Funktionsbausteins STÖBER_Backup_Restore ist der Dateipfad zum Verzeichnis der Projektdatei (*.ds6) auf der EtherCAT-Steuerung angegeben.
- ✓ Am Eingang DS6ProcessPath des Funktionsbausteins STÖBER_Backup_Restore ist der Dateipfad zur DriveControlSuite (*.exe) auf der EtherCAT-Steuerung angegeben.
- 1. Wechseln Sie zu TwinCAT XAE.
- 2. Stellen Sie sicher, dass sich alle Antriebsregler in Ihrem EtherCAT-Netzwerk im Zustand Operational befinden.
- 3. Führen Sie den Funktionsbaustein STÖBER_BoxName aus.
 - ⇒ Hat der Funktionsbaustein die Namen aller EtherCAT SubDevices in die Antriebsregler geschrieben, wird der Ausgang Done auf True gesetzt.
- 4. Rufen Sie im Anschluss die Instanzen des Funktionsbausteins STÖBER_Backup_Restore für jeden Antriebsregler nacheinander auf.
 - ⇒ Nach erfolgreichem Abschluss des Services wird der Ausgang Done auf True gesetzt.
 - ⇒ Die Daten werden im Antriebsregler nichtflüchtig gespeichert.

6.3.2 Beispiel-Code

Das folgende Beispielprojekt dient der Implementierung in Strukturiertem Text (ST). Es zeigt die serielle Ausführung des Funktionsbausteins STOBER_Backup_Restore für ein EtherCAT-Netzwerk mit zwei Antriebsreglern.

```
PROGRAM MAIN
VAR
    fbBoxname:STOBER_BoxName;
    fbBackup_Init1,fbBackup_Init2:STOBER_Backup_Restore_Initiator;
    fbBackup1,fbBackup2:STOBER_Backup_Restore;
    bExecuteBox: BOOL;
    bExecute_BR_init: ARRAY [0..1] OF BOOL;
    done_init: ARRAY [0..1] OF BOOL;
    busy_init: ARRAY [0..1] OF BOOL;
    Error_init: ARRAY [0..1] OF BOOL;
    BR_Init_ErrorIDADS: ARRAY [0..1] OF UDINT;
    BR_Init_FbErrorID: ARRAY [0..1] OF STOBER_G6_Util.eFBERROR;
    bExecute_BR: ARRAY [0..1] OF BOOL;
    done: ARRAY [0..1] OF BOOL;
    busy: ARRAY [0..1] OF BOOL;
    Error: ARRAY [0..1] OF BOOL;
    errorIDADS: ARRAY [0..1] OF UDINT;
    FBErrorID: ARRAY [0..1] OF STOBER_G6_Util.eFBERROR;
    iAction: ARRAY [0..1] OF INT;
    bError: BOOL;
    bBusy: BOOL;
    bDone: BOOL;
    uiErrorID: UDINT;
    uiFbErrorID: STOBER_G6_Util.eFBError;
    service: ARRAY [0..1] OF STOBER_G6_Util.eService;
END_VAR

fbBoxname (
    AmsNetId_EtherCAT_MainDevice:='192.168.12.50.3.1',
    Execute:=bExecuteBox,
    Error=>bError,
    Busy=>bBusy,
    Done=>bDone,
    ErrorID=>uiErrorID,
    FBErrorID=>uiFbErrorID);

fbBackup_Init1 (
    Execute:=bExecute_BR_init[0],
    AmsNetId_EtherCAT_MainDevice:='192.168.12.50.3.1',
    Filepath:='C:\Transfer\ExampleProject\DS6-Projects',
    SubDeviceAddr:=1004,
    Done=>done_init[0],
    Busy=>busy_init[0],
    Error=>Error_init[0],
    ErrorID=>BR_Init_ErrorIDADS[0],
    FBErrorID=>BR_Init_FbErrorID [0],
    iAction=>iAction[0]);
```

```
IF iAction[0] = 0 THEN
    service[0]:=STOBER_G6_Util.eService.Backup;
ELSE
    service[0]:=STOBER_G6_Util.eService.RESTORE;
END_IF
fbBackup1(
    Execute:=bExecute_BR[0] AND done_init[0],
    AmsNetId_EtherCAT_MainDevice:='192.168.12.50.3.1',
    Filepath:='C:\Transfer\ExampleProject\DS6-Projects',
    DS6ProcessPath:='C:\DS6\DriveControlSuite_Nightly\bin',
    SubDeviceAddr:=1004,
    Servicetype:=service[0],
    Done=>done[0],
    Busy=>busy[0],
    Error=>Error[0],
    ErrorID=>errorIDADS[0],
    FBErrorID=>FbErrorID[0]);
fbBackup_Init2(
    Execute:=bExecute_BR_init[1] AND done[0],
    AmsNetId_EtherCAT_MainDevice:='192.168.12.50.3.1',
    Filepath:='C:\Transfer\ExampleProject\DS6-Projects',
    SubDeviceAddr:=1005,
    Done=>done_init[1],
    Busy=>busy_init[1],
    Error=>Error_init[1],
    ErrorID=>BR_Init_ErrorIDADS[1],
    FBErrorID=>BR_Init_FbErrorID [1],
    iAction=>iAction[1]);
IF iAction[1] = 0 THEN
    service[1]:=STOBER_G6_Util.eService.Backup;
ELSE
    service[1]:=STOBER_G6_Util.eService.RESTORE;
END_IF
fbBackup2(
    Execute:=bExecute_BR[1] AND done_init[1],
    AmsNetId_EtherCAT_MainDevice:='192.168.12.50.3.1',
    Filepath:='C:\Transfer\ExampleProject\DS6-Projects',
    DS6ProcessPath:='C:\DS6\DriveControlSuite_Nightly\bin',
    SubDeviceAddr:=1005,
    Servicetype:=service[1],
    Done=>done[1],
    Busy=>busy[1],
    Error=>Error[1],
    ErrorID=>errorIDADS[1],
    FBErrorID=>FbErrorID[1]);
```

6.4 STOBER_MC_HOME

Der Funktionsbaustein steuert die antriebsreglergeführte Referenzierung der Applikationen CiA 402 und CiA 402 HiRes Motion. Bei der Ausführung des Funktionsbausteins wird die aktuelle Betriebsart ausgelesen. Danach wird die Referenziermethode für die Referenzfahrt aktiviert, die in Parameter A586 für den Antriebsregler definiert ist. Nach erfolgreicher Referenzierung wird die zuvor ausgelesene Betriebsart reaktiviert.

Voraussetzungen

- TwinCAT 3 ab Version 3.1.4024.40
- Bibliothek STOBER_G6_Util ab Version 3.1.2.0
- Bibliothek Tc2_MC2 von Beckhoff

Information

Da die Übergabe einer NC-Achse an den Funktionsbaustein erforderlich ist, müssen Sie eine Verknüpfung zwischen NC-Achse und SPS-Projekt erstellen (siehe [NC-Achse und SPS-Projekt verknüpfen \[▶ 9\]](#)).

Parameter



Abb. 4: Funktionsbaustein STOBER_MC_HOME: Ein- und Ausgangsparameter

Parameter	Datentyp	Deklaration	Beschreibung
Axis	AXIS_REF	IN/OUT	Achsdatenstruktur
Execute	BOOL	IN	Aktivieren des Funktionsbausteins mit steigender Flanke
Timeout	TIME	IN	Vorgegebene Zeitspanne, nach der eine Fehlermeldung ausgelöst wird, wenn die Referenzfahrt nicht zu einer Referenzierung führt
Busy	BOOL	OUT	Zustand des Funktionsbausteins (Busy = True: Referenzierung noch nicht beendet)
Done	BOOL	OUT	Zustand des Funktionsbausteins (Done = True: Referenzierung abgeschlossen)
Error	BOOL	OUT	Zustand des Funktionsbausteins (Error = True: Referenzierung fehlerhaft)
ErrorID	UDINT	OUT	Beckhoff-spezifischer ADS Error Code der intern verwendeten Funktionsbausteine
FBErrorID	eFBError	OUT	Bausteinspezifischer Fehler-Code (siehe eFBError (ENUM) [▶ 48])

Tab. 6: Funktionsbaustein STOBER_MC_HOME: Parameter

Informationen zu den Datentypen entnehmen Sie der Dokumentation zu TwinCAT 3 unter https://infosys.beckhoff.com/content/1031/tc3_plc_intro/2529388939.html.

6.4.1 Ablauf der Referenzierung

Voraussetzungen

- NC-Achse und SPS-Projekt sind verknüpft (siehe [NC-Achse und SPS-Projekt verknüpfen \[► 9\]](#))
- Betriebsart im Antriebsregler entspricht csp, csv, cst oder Homing mode (A541 = 8: Cyclic synchronous position mode, 9: Cyclic synchronous velocity mode, 10: Cyclic synchronous torque mode oder 6: Homing mode)
- Achse ist freigegeben

Ablauf

Während der Ausführung des Funktionsbausteins STÖBER_MC_HOME werden die folgenden Schritte durchlaufen:

1. Lesen der Achsdaten (ADS Zugriffsdaten, z.B. AMS NetID, SubDevice-Adresse, Achstyp, ...)
2. Auslesen der aktuellen Betriebsart der Steuerung für die Achse
3. Löschen des Referenzbit der NC-Achse
4. Setzen der Betriebsart auf Homing mode (A541 = 6: Homing mode)
5. Einlesen der Referenziermethode (A586) über CoE
6. Starten der Referenzfahrt
7. Warten, bis die Referenzierung beendet ist
8. Setzen der Referenz für die NC-Achse
9. Setzen der aktuellen Istposition als Sollposition für die NC-Achse
10. Setzen der Betriebsart (A541) auf die zu Beginn der Aktion ausgelesene Betriebsart

6.4.2 Beispiel-Code

Das folgende Beispiel dient der Implementierung in Strukturiertem Text (ST).

```
PROGRAM MAIN
VAR
    lSTOBER_MC_HOME: STOBER_MC_HOME;
    lEnable: MC_Power;
    lAxis: AXIS_REF;
    lExecute: BOOL;
    lBusy: BOOL;
    lDone: BOOL;
    lError: BOOL;
    lErrorID: UDINT;
    lFBErorID: STOBER_G6_Util.eFBEror;
    Enable: BOOL;
END_VAR

lEnable(Axis:=lAxis,
        Enable:=Enable,
        Enable_Positive:=Enable,
        Enable_Negative:=Enable);
lSTOBER_MC_HOME(Axis:=lAxis,
                Execute:=lExecute,
                Timeout:=T#180S,
                Busy=>lBusy,
                Done=>lDone,
                Error=>lError,
                ErrorID=>lErrorID,
                FBErorID=>lFBErorID);
```

6.5 STOBER_MC_HOME_REF

Der Funktionsbaustein steuert die antriebsreglergeführte Referenzierung der Applikationen CiA 402 und CiA 402 HiRes Motion. Bei der Ausführung des Funktionsbausteins wird die aktuelle Betriebsart ausgelesen. Danach wird die Referenziermethode für die Referenzfahrt in Parameter A586 geschrieben und aktiviert. Nach erfolgreicher Referenzierung wird die zuvor ausgelesene Betriebsart reaktiviert.

Dem Funktionsbaustein muss die Referenzposition (A569) übergeben werden, die nach der Referenzierung anliegen soll, und die Referenziermethode (A586), mit der der Antriebsregler referenzieren soll.

Voraussetzungen

- TwinCAT 3 ab Version 3.1.4024.40
- Bibliothek STOBER_G6_Util ab Version 3.1.2.0
- Bibliothek Tc2_MC2 von Beckhoff

Information

Da die Übergabe einer NC-Achse an den Funktionsbaustein erforderlich ist, müssen Sie eine Verknüpfung zwischen NC-Achse und SPS-Projekt erstellen (siehe [NC-Achse und SPS-Projekt verknüpfen \[▶ 9\]](#)).

Parameter



Abb. 5: Funktionsbaustein STOBER_MC_HOME_REF: Ein- und Ausgangsparameter

Parameter	Datentyp	Deklaration	Beschreibung
Axis	AXIS_REF	IN/OUT	Achsdatenstruktur
Execute	BOOL	IN	Aktivieren des Funktionsbausteins mit steigender Flanke
Timeout	TIME	IN	Vorgegebene Zeitspanne, nach der eine Fehlermeldung ausgelöst wird, wenn die Referenzfahrt nicht zu einer Referenzierung führt
ReferenceValue	LREAL	IN	Referenzposition nach der Referenzierung (A569)
HomingMethod	eHomingMethod	IN	Referenziermethode für die Referenzierung (A586)
Busy	BOOL	OUT	Zustand des Funktionsbausteins (Busy = True: Referenzierung noch nicht beendet)
Done	BOOL	OUT	Zustand des Funktionsbausteins (Done = True: Referenzierung abgeschlossen)
Error	BOOL	OUT	Zustand des Funktionsbausteins (Error = True: Referenzierung fehlerhaft)
ErrorID	UDINT	OUT	Beckhoff-spezifischer ADS Error Code der intern verwendeten Funktionsbausteine
FBErrorID	eFBError	OUT	Bausteinspezifischer Fehler-Code (siehe eFBError (ENUM) [▶ 48])

Tab. 7: Funktionsbaustein STOBER_MC_HOME_REF: Parameter

Informationen zu den Datentypen entnehmen Sie der Dokumentation zu TwinCAT 3 unter https://infosys.beckhoff.com/content/1031/tc3_plc_intro/2529388939.html.

6.5.1 Ablauf der Referenzierung

Voraussetzungen

- NC-Achse und SPS-Projekt sind verknüpft (siehe [NC-Achse und SPS-Projekt verknüpfen \[► 9\]](#))
- Betriebsart im Antriebsregler entspricht csp, csv, cst oder Homing mode (A541 = 8: Cyclic synchronous position mode, 9: Cyclic synchronous velocity mode, 10: Cyclic synchronous torque mode oder 6: Homing mode)
- Achse ist freigegeben

Ablauf

Während der Ausführung des Funktionsbausteins STÖBER_MC_HOME_REF werden die folgenden Schritte durchlaufen:

1. Schreiben der Referenziermethode in Parameter A586 (entspricht dem Kommunikationsobjekt Homing method nach CiA 402; Objekt 6098 hex, Objekt 6898 hex)
2. Schreiben der Referenzposition in Parameter A569 (entspricht dem Kommunikationsobjekt Home offset nach CiA 402; Objekt 607C hex, Objekt 687C hex)
3. Lesen der Achsdaten (ADS Zugriffsdaten, z.B. AMS NetID, SubDevice-Adresse, Achstyp, ...)
4. Auslesen der aktuellen Betriebsart der Steuerung für die Achse
5. Löschen des Referenzbit der NC-Achse
6. Setzen der Betriebsart auf Homing mode (A541 = 6: Homing mode)
7. Einlesen der Referenziermethode (A586) über CoE
8. Starten der Referenzfahrt
9. Warten, bis die Referenzierung beendet ist
10. Setzen der Referenz für die NC-Achse
11. Setzen der aktuellen Istposition als Sollposition für die NC-Achse
12. Setzen der Betriebsart (A541) auf die zu Beginn der Aktion ausgelesene Betriebsart

6.5.2 Beispiel-Code

Das folgende Beispiel dient der Implementierung in Strukturiertem Text (ST).

```
PROGRAM MAIN
VAR
    lSTOBER_MC_HOME: STOBER_MC_HOME_REF;
    lEnable: MC_Power;
    lAxis: AXIS_REF;
    lExecute: BOOL;
    lBusy: BOOL;
    lDone: BOOL;
    lError: BOOL;
    lErrorID: UDINT;
    lFBErrorID: STOBER_G6_Util.eFBError;
    Enable: BOOL;
    ReferenceValue: LREAL;
    HomingMethod: eHomingMethod;
END_VAR

lEnable(Axis:=lAxis,
        Enable:=Enable,
        Enable_Positive:=Enable,
        Enable_Negative:=Enable);
lSTOBER_MC_HOME(Axis:=lAxis,
                Execute:=lExecute,
                Timeout:=T#180S,
                ReferenceValue:=ReferenceValue,
                HomingMethod:=HomingMethod,
                Busy=>lBusy,
                Done=>lDone,
                Error=>lError,
                ErrorID=>lErrorID,
                FBErrorID=>lFBErrorID);
```

6.6 STOBER_Action

Über den Funktionsbaustein STOBER_Action können Aktionen auf dem Antriebsregler ausgeführt werden. Alle Voraussetzungen sowie nähere Informationen zu den einzelnen Aktionen entnehmen Sie den entsprechenden Parameterbeschreibungen in der DriveControlSuite. Über die Koordinaten des Parameters (Gruppe und Zeile) können Sie den Index für den Funktionsbaustein berechnen. Nach erfolgreicher Ausführung gibt der Funktionsbaustein das Ergebnis der Aktion aus.

Voraussetzungen

- TwinCAT 3 ab Version 3.1.4024.40
- Bibliothek STOBER_G6_Util ab Version 3.1.2.0

Parameter

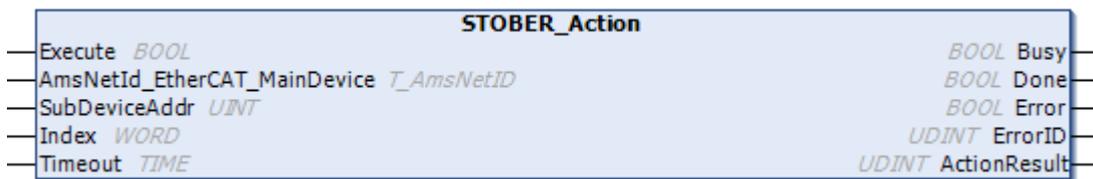


Abb. 6: Funktionsbaustein STOBER_Action: Ein- und Ausgangsparameter

Parameter	Datentyp	Deklaration	Beschreibung
Execute	BOOL	IN	Aktivieren des Funktionsbausteins mit steigender Flanke
AmsNetId_EtherCAT_MainDevice	T_AmsNetID	IN	AMS NetID des EtherCAT MainDevices
SubDeviceAddr	UINT	IN	Adresse des EtherCAT SubDevices
Index	WORD	IN	Index der Aktion, die ausgeführt werden soll (siehe Berechnung des Index [▶ 46])
Timeout	TIME	IN	Vorgegebene Zeitspanne, nach der eine Fehlermeldung ausgelöst wird, wenn die Aktion kein Ergebnis liefert
Busy	BOOL	OUT	Zustand des Funktionsbausteins (Busy = True: Aktion noch nicht beendet)
Done	BOOL	OUT	Zustand des Funktionsbausteins (Done = True: Aktion erfolgreich abgeschlossen)
Error	BOOL	OUT	Zustand des Funktionsbausteins (Error = True: Aktion fehlerhaft)
ErrorID	UDINT	OUT	Beckhoff-spezifischer ADS Error Code der intern verwendeten Funktionsbausteine
ActionResult	UDINT	OUT	Ergebnis der Aktion

Tab. 8: Funktionsbaustein STOBER_Action: Parameter

Informationen zu den Datentypen entnehmen Sie der Dokumentation zu TwinCAT 3 unter https://infosys.beckhoff.com/content/1031/tc3_plc_intro/2529388939.html.

6.6.1 Beispiele für Aktionen ohne erforderliche Freigabe

Nachfolgende Tabelle listet Beispiele für Aktionen, die über den Funktionsbaustein STÖBER_Action ausgeführt werden können, und die keine Freigabe des Antriebsreglers erfordern.

Parameter	Aktion	Index Achse A	Index Achse B
A00	Werte speichern	2000 hex	A000 hex
A09	Neu starten	2009 hex	A009 hex
B06	Typenschild auslesen	2006 hex	A206 hex
B30	Achsmodell aus Typenschild lesen	221E hex	A21E hex
I38	Referenz löschen	3026 hex	B026 hex

Tab. 9: Funktionsbaustein STÖBER_Action: Beispiele für Aktionen ohne erforderliche Freigabe

Prüfen Sie in der DriveControlSuite, welche Aktionen Ihnen in Abhängigkeit von Zugriffslevel, Hardware, Software und Applikation zur Verfügung stehen. Alle Voraussetzungen sowie nähere Informationen zu den einzelnen Aktionen entnehmen Sie den entsprechenden Parameterbeschreibungen.

6.6.2 Beispiel-Code

Das folgende Beispielprojekt dient der Implementierung in Strukturiertem Text (ST).

```
PROGRAM MAIN
VAR
    fbAction:STÖBER_Action;
    SubDevice_Address:UINT;
    bExecute: BOOL;
    bError: BOOL;
    bBusy: BOOL;
    bDone: BOOL;
    uiADSErrorID: UDINT;
    uiActionResult: UDINT;
    Net_ID: T_AmsNetID;
END_VAR
VAR CONSTANT
    SaveValuesIndex:WORD:=16#2000;
END_VAR

fbAction(
    AmsNetId_EtherCAT_MainDevice:=Net_ID,
    Execute:=bExecute,
    SubDevice_Addr:=SubDevice_Address,
    Index:=SaveValuesIndex,
    Timeout:=t#60s,
    Error=>bError,
    Busy=>bBusy,
    Done=>bDone,
    ErrorID=>uiADSErrorID,
    ActionResult=>uiActionResult);
```

6.7 STOBER_Power_Action

Über den Funktionsbaustein STOBER_Power_Action können Aktionen auf dem Antriebsregler ausgeführt werden, die eine Freigabe des Antriebsreglers benötigen. Alle Voraussetzungen sowie nähere Informationen zu den einzelnen Aktionen entnehmen Sie den entsprechenden Parameterbeschreibungen in der DriveControlSuite. Über die Koordinaten des Parameters (Gruppe und Zeile) können Sie den Index für den Funktionsbaustein berechnen. Nach erfolgreicher Ausführung gibt der Funktionsbaustein das Ergebnis der Aktion aus.

Voraussetzungen

- TwinCAT 3 ab Version 3.1.4024.40
- Bibliothek STOBER_G6_Util ab Version 3.1.2.0
- Bibliothek Tc2_MC2 von Beckhoff

Information

Alle MC_POWER-Bausteine Ihres SPS-Projekts dürfen die Freigabe dieses Funktionsbausteins nicht überschreiben und müssen vor diesem aufgerufen werden.

Information

Da die Übergabe einer NC-Achse an den Funktionsbaustein erforderlich ist, müssen Sie eine Verknüpfung zwischen NC-Achse und SPS-Projekt erstellen (siehe [NC-Achse und SPS-Projekt verknüpfen \[► 9\]](#)).

Parameter

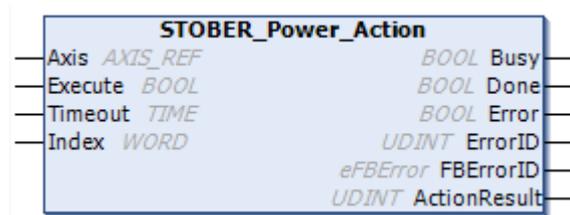


Abb. 7: Funktionsbaustein STOBER_Power_Action: Ein- und Ausgangsparameter

Parameter	Datentyp	Deklaration	Beschreibung
Axis	AXIS_REF	IN/OUT	Achsdatenstruktur
Execute	BOOL	IN	Aktivieren des Funktionsbausteins mit steigender Flanke
Timeout	TIME	IN	Vorgegebene Zeitspanne, nach der eine Fehlermeldung ausgelöst wird, wenn die Aktion kein Ergebnis liefert
Index	WORD	IN	Index der Aktion, die ausgeführt werden soll (siehe Berechnung des Index [► 46])
Busy	BOOL	OUT	Zustand des Funktionsbausteins (Busy = True: Aktion noch nicht beendet)
Done	BOOL	OUT	Zustand des Funktionsbausteins (Done = True: Aktion erfolgreich abgeschlossen)
Error	BOOL	OUT	Zustand des Funktionsbausteins (Error = True: Aktion fehlerhaft)
ErrorID	UDINT	OUT	Beckhoff-spezifischer ADS Error Code der intern verwendeten Funktionsbausteine
FBErrorID	eFBError	OUT	Bausteinspezifischer Fehler-Code (siehe eFBError (ENUM) [► 48])
ActionResult	UDINT	OUT	Ergebnis der Aktion

Tab. 10: Funktionsbaustein STOBER_Power_Action: Parameter

Informationen zu den Datentypen entnehmen Sie der Dokumentation zu TwinCAT 3 unter https://infosys.beckhoff.com/content/1031/tc3_plc_intro/2529388939.html.

6.7.1 Beispiele für Aktionen mit erforderlicher Freigabe

Nachfolgende Tabelle listet Beispiele für Aktionen, die über den Funktionsbaustein STOBER_Power_Action ausgeführt werden können, und die eine Freigabe des Antriebsreglers erfordern.

Parameter	Aktion	Index Achse A	Index Achse B
B41	Motor einmessen	2229 hex	A229 hex
B43	Wicklung testen	222B hex	A22B hex
B49	Stromregler optimieren (Stillstand)	2231 hex	A231 hex

Tab. 11: Funktionsbaustein STOBER_Power_Action: Beispiele für Aktionen mit erforderlicher Freigabe

Prüfen Sie in der DriveControlSuite, welche Aktionen Ihnen in Abhängigkeit von Zugriffslevel, Hardware, Software und Applikation zur Verfügung stehen. Alle Voraussetzungen sowie nähere Informationen zu den einzelnen Aktionen entnehmen Sie den entsprechenden Parameterbeschreibungen.

6.7.2 Beispiel-Code

Das folgende Beispielprojekt dient der Implementierung in Strukturiertem Text (ST).

```
PROGRAM MAIN
VAR
    lSTOBER_Power_Action: STOBER_Power_Action;
    lAxis: AXIS_REF;
    lExecute: BOOL;
    lBusy: BOOL;
    lDone: BOOL;
    lError: BOOL;
    lErrorID: UDINT;
    lFBErrorID: STOBER_G6_Util.eFBError;
    lTimeout: TIME;
END_VAR
VAR CONSTANT
    Phasetest_Index:WORD:=16#2228;
END_VAR

lSTOBER_Power_Action(Axis:=lAxis,
    Execute:=lExecute,
    Timeout:=lTimeout,
    Index:=Phasetest_Index,
    Busy=>lBusy,
    Done=>lDone,
    Error=>lError,
    ErrorID=>lErrorID,
    FBErrorID=>lFBErrorID);
```

6.8 STOBER_Phase_Test

Der Funktionsbaustein STOBER_Phase_Test startet die Aktion Phasentest auf dem Antriebsregler, die unter anderem den Kommutierungsoffset einmisst. Nach dem erfolgreichem Phasentest werden die Werte nichtflüchtig auf dem Antriebsregler gespeichert.



Lebensgefahr durch schwerkraftbelastete Vertikalachse!

Bei dieser Aktion werden die Bremsen lüftend angesteuert. Der Motor kann währenddessen nicht oder nur eingeschränkt Drehmoment/Kraft generieren. Somit kann eine schwerkraftbelastete Vertikalachse absinken.

- Führen Sie diese Aktion nur bei schwerkraftfreien Achsen aus.

Information

Der Funktionsbaustein führt die Aktion Werte speichern aus (A00).

Voraussetzungen

- TwinCAT 3 ab Version 3.1.4024.40
- Bibliothek STOBER_G6_Util ab Version 3.1.2.0
- Bibliothek Tc2_MC2 von Beckhoff

Information

Alle MC_POWER-Bausteine Ihres SPS-Projekts dürfen die Freigabe dieses Funktionsbausteins nicht überschreiben und müssen vor diesem aufgerufen werden.

Information

Da die Übergabe einer NC-Achse an den Funktionsbaustein erforderlich ist, müssen Sie eine Verknüpfung zwischen NC-Achse und SPS-Projekt erstellen (siehe [NC-Achse und SPS-Projekt verknüpfen \[► 9\]](#)).

Parameter

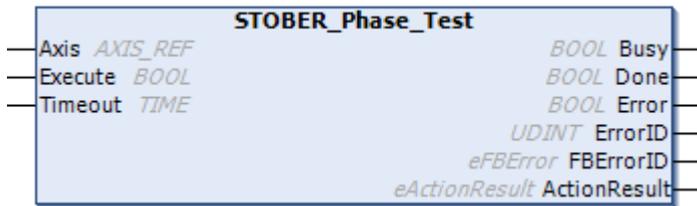


Abb. 8: Funktionsbaustein STOBER_Phase_Test: Ein- und Ausgangsparameter

Parameter	Datentyp	Deklaration	Beschreibung
Axis	AXIS_REF	IN/OUT	Achsdatenstruktur
Execute	BOOL	IN	Aktivieren des Funktionsbausteins mit steigender Flanke
Timeout	TIME	IN	Vorgegebene Zeitspanne, nach der eine Fehlermeldung ausgelöst wird, wenn die Aktion kein Ergebnis liefert
Busy	BOOL	OUT	Zustand des Funktionsbausteins (Busy = True: Aktion noch nicht beendet oder Werte speichern noch nicht ausgeführt)
Done	BOOL	OUT	Zustand des Funktionsbausteins (Done = True: Aktion erfolgreich abgeschlossen und Werte gespeichert)
Error	BOOL	OUT	Zustand des Funktionsbausteins (Error = True: Aktion fehlerhaft)
ErrorID	UDINT	OUT	Beckhoff-spezifischer ADS Error Code der intern verwendeten Funktionsbausteine
FBErrorID	eFBError	OUT	Bausteinspezifischer Fehler-Code (siehe eFBError (ENUM) [▶ 48])
ActionResult	eActionResult	OUT	Ergebnis der Aktion

Tab. 12: Funktionsbaustein STOBER_Phase_Test: Parameter

Informationen zu den Datentypen entnehmen Sie der Dokumentation zu TwinCAT 3 unter https://infosys.beckhoff.com/content/1031/tc3_plc_intro/2529388939.html.

6.8.1 Ablauf des Phasentests

Voraussetzungen

NC-Achse und SPS-Projekt sind verknüpft (siehe [NC-Achse und SPS-Projekt verknüpfen \[▶ 9\]](#))

Ablauf

Während der Ausführung des Funktionsbausteins STOBER_Phase_Test werden die folgenden Schritte durchlaufen:

1. Lesen der Achsdaten (ADS Zugriffsdaten, z.B. AMS NetID, SubDevice-Adresse, Achstyp, ...)
2. Ausführen des Funktionsbausteins STOBER_Power_Action mit der Aktion Phasentest:
 - 2.1. Prüfung der Freigabe und gegebenenfalls Freigabe der Achse
 - 2.2. Ausführen der Aktion Phasentest
 - 2.3. Entfernen der Freigabe der Achse
3. Ausführen der Aktion Werte speichern (A00)

6.8.2 Beispiel-Code

Das folgende Beispielprojekt dient der Implementierung in Strukturiertem Text (ST).

```
PROGRAM MAIN
VAR
    lSTOBER_LoadMatrix : STOBER_PRM_LoadMatrix;
    lAxis: AXIS_REF;
    lExecute: BOOL;
    lBusy: BOOL;
    lDone: BOOL;
    lError: BOOL;
    lErrorID: UDINT;
    lFBEErrorID: STOBER_G6_Util.eFBEError;
    PRMjson: ARRAY [0..16255] OF BYTE;
    StringLength: UINT;
END_VAR

lSTOBER_LoadMatrix (Axis:=lAxis, PRMjson:=PRMjson,
    Execute:=lExecute,
    Busy=>lBusy,
    Done=>lDone,
    Error=>lError,
    ErrorID=>lErrorID,
    FBEErrorID=>lFBEErrorID,
    StringLength=>StringLength);
```

6.9 STOBER_PRM_LoadMatrix

Der Funktionsbaustein liest die Last-Matrix aus dem Antriebsregler aus (R118) und schreibt sie in das übergebene Array.

Voraussetzungen

- TwinCAT 3 ab Version 3.1.4024.40
- Bibliothek STOBER_G6_Util ab Version 3.1.2.0
- Bibliothek Tc2_MC2 von Beckhoff

Information

Um die Last-Matrix auslesen zu können, muss die Predictive-Maintenance-Funktion im Antriebsregler aktiv sein.

Information

Da die Übergabe einer NC-Achse an den Funktionsbaustein erforderlich ist, müssen Sie eine Verknüpfung zwischen NC-Achse und SPS-Projekt erstellen (siehe [NC-Achse und SPS-Projekt verknüpfen \[▶ 9\]](#)).

Parameter

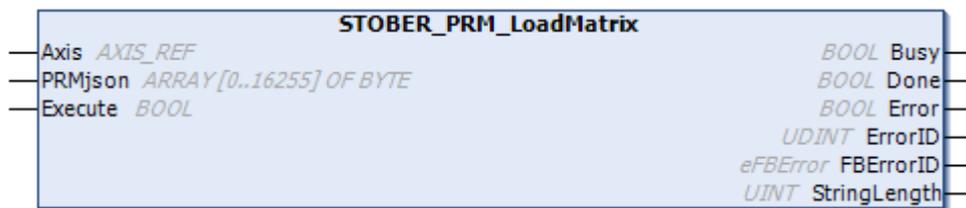


Abb. 9: Funktionsbaustein STOBER_PRM_LoadMatrix: Ein- und Ausgangsparameter

Parameter	Datentyp	Deklaration	Beschreibung
Axis	AXIS_REF	IN/OUT	Achsdatenstruktur
PRMjson	ARRAY [0..16255] OF BYTE	IN/OUT	Datenpuffer, in den die Last-Matrix geschrieben werden soll
Execute	BOOL	IN	Aktivieren des Funktionsbausteins mit steigender Flanke
Busy	BOOL	OUT	Zustand des Funktionsbausteins (Busy = True: Lesen und Schreiben noch nicht beendet)
Done	BOOL	OUT	Zustand des Funktionsbausteins (Done = True: Lesen und Schreiben erfolgreich abgeschlossen)
Error	BOOL	OUT	Zustand des Funktionsbausteins (Error = True: Lesen/Schreiben fehlerhaft)
ErrorID	UDINT	OUT	Beckhoff-spezifischer ADS Error Code der intern verwendeten Funktionsbausteine
FBErrorID	eFBError	OUT	Bausteinspezifischer Fehler-Code (siehe eFBError (ENUM) [▶ 48])
StringLength	UINT	OUT	Anzahl an beschriebenen Bytes im Array PRMjson

Tab. 13: Funktionsbaustein STOBER_PRM_LoadMatrix: Parameter

Informationen zu den Datentypen entnehmen Sie der Dokumentation zu TwinCAT 3 unter https://infosys.beckhoff.com/content/1031/tc3_plc_intro/2529388939.html.

6.9.1 Ablauf des Auslesens und Schreibens

Während der Ausführung des Funktionsbausteins STÖBER_PRM_LoadMatrix werden die folgenden Schritte durchlaufen:

1. Lesen der Achsdaten (ADS Zugriffsdaten, z.B. AMS NetID, SubDevice-Adresse, Achstyp, ...)
2. Auslesen des Predictive-Maintenance-Status (R100)
3. Auslesen der Last-Matrix und Schreiben in übergebenes Array PRMjson
4. StringLength der Last-Matrix berechnen

6.9.2 Beispiel-Code

Das folgende Beispielprojekt dient der Implementierung in Strukturiertem Text (ST).

```
PROGRAM MAIN
VAR
    lSTÖBER_LoadMatrix : STÖBER_PRM_LoadMatrix;
    lAxis: AXIS_REF;
    lExecute: BOOL;
    lBusy: BOOL;
    lDone: BOOL;
    lError: BOOL;
    lErrorID: UDINT;
    lFBErrorID: STÖBER_G6_Util.eFBError;
    PRMjson: ARRAY [0..16255] OF BYTE;
    StringLength: UINT;
END_VAR

lSTÖBER_LoadMatrix (Axis:=lAxis, PRMjson:=PRMjson,
    Execute:=lExecute,
    Busy=>lBusy,
    Done=>lDone,
    Error=>lError,
    ErrorID=>lErrorID,
    FBErrorID=>lFBErrorID,
    StringLength=>StringLength);
```

6.10 STOBER_PRM_LoadMatrix_AMS

Der Funktionsbaustein liest die Last-Matrix aus dem Antriebsregler aus (R118) und schreibt sie in das übergebene Array.

Voraussetzungen

- TwinCAT 3 ab Version 3.1.4024.40
- Bibliothek STOBER_G6_Util ab Version 3.1.2.0
- Bibliothek Tc2_MC2 von Beckhoff

Information

Um die Last-Matrix auslesen zu können, muss die Predictive-Maintenance-Funktion im Antriebsregler aktiv sein.

Parameter



Abb. 10: Funktionsbaustein STOBER_PRM_LoadMatrix_AMS: Ein- und Ausgangsparameter

Parameter	Datentyp	Deklaration	Beschreibung
PRMjson	ARRAY [0..16255] OF BYTE	IN/OUT	Datenpuffer, in den die Last-Matrix geschrieben werden soll
AmsNetId_EtherCAT_MainDevice	T_AmsNetID	IN	AMS NetID des EtherCAT MainDevices
SubDeviceAddr	UINT	IN	Adresse des EtherCAT SubDevices
AxisNumber	USINT	IN	Achse, aus der die Last-Matrix ausgelesen werden soll: <ul style="list-style-type: none"> ▪ 0 = Achse A ▪ 1 = Achse B
Execute	BOOL	IN	Aktivieren des Funktionsbausteins mit steigender Flanke
Busy	BOOL	OUT	Zustand des Funktionsbausteins (Busy = True: Lesen und Schreiben noch nicht beendet)
Done	BOOL	OUT	Zustand des Funktionsbausteins (Done = True: Lesen und Schreiben erfolgreich abgeschlossen)
Error	BOOL	OUT	Zustand des Funktionsbausteins (Error = True: Lesen/Schreiben fehlerhaft)
ErrorID	UDINT	OUT	Beckhoff-spezifischer ADS Error Code der intern verwendeten Funktionsbausteine
FBErrorID	eFBError	OUT	Bausteinspezifischer Fehler-Code (siehe eFBError (ENUM) [► 48])
StringLength	UINT	OUT	Anzahl an beschriebenen Bytes im Array PRMjson

Tab. 14: Funktionsbaustein STOBER_PRM_LoadMatrix_AMS: Parameter

Informationen zu den Datentypen entnehmen Sie der Dokumentation zu TwinCAT 3 unter https://infosys.beckhoff.com/content/1031/tc3_plc_intro/2529388939.html.

6.10.1 Ablauf des Auslesens und Schreibens

Während der Ausführung des Funktionsbausteins STÖBER_PRM_LoadMatrix_AMS werden die folgenden Schritte durchlaufen:

1. Auslesen des Predictive-Maintenance-Status (R100)
2. Auslesen der Last-Matrix und Schreiben in übergebenes Array PRMjson
3. StringLength der Last-Matrix berechnen

6.10.2 Beispiel-Code

Das folgende Beispielprojekt dient der Implementierung in Strukturiertem Text (ST).

```
PROGRAM MAIN
VAR
    lSTOBER_LoadMatrix : STOBER_PRM_LoadMatrix_AMS;
    lExecute: BOOL;
    lAMS_NetID: T_AmsNetID;
    lSubDeviceAddr: UINT;
    lBusy: BOOL;
    lDone: BOOL;
    lError: BOOL;
    lErrorID: UDINT;
    lFBErrorID: STOBER_G6_Util.eFBError;
    PRMjson: ARRAY [0..16255] OF BYTE;
    StringLength: UINT;
END_VAR

lSTOBER_LoadMatrix (PRMjson:=PRMjson,
    AmsNetId_EtherCAT_MainDevice:=lAMS_NetID,
    SubDeviceAddr:=lSubDeviceAddr,
    Execute:=lExecute,
    Busy=>lBusy,
    Done=>lDone,
    Error=>lError,
    ErrorID=>lErrorID,
    FBErrorID=>lFBErrorID,
    StringLength=>StringLength);
```

6.11 STOBER_PRM_LoadMatrix_File

Der Funktionsbaustein liest die Last-Matrix aus dem Antriebsregler aus (R118) und legt sie als Datei im Verzeichnis unter dem übergebenen Dateipfad ab.

Voraussetzungen

- TwinCAT 3 ab Version 3.1.4024.40
- Bibliothek STOBER_G6_Util ab Version 3.1.2.0
- Bibliothek Tc2_MC2 von Beckhoff

Information

Um die Last-Matrix auslesen zu können, muss die Predictive-Maintenance-Funktion im Antriebsregler aktiv sein.

Information

Da die Übergabe einer NC-Achse an den Funktionsbaustein erforderlich ist, müssen Sie eine Verknüpfung zwischen NC-Achse und SPS-Projekt erstellen (siehe [NC-Achse und SPS-Projekt verknüpfen \[▶ 9\]](#)).

Parameter



Abb. 11: Funktionsbaustein STOBER_PRM_LoadMatrix_File: Ein- und Ausgangsparameter

Parameter	Datentyp	Deklaration	Beschreibung
Execute	BOOL	IN	Aktivieren des Funktionsbausteins mit steigender Flanke
Filepath	T_MaxString	IN	Dateipfad zum Verzeichnis, in das die Datei geschrieben werden soll
Axis	AXIS_REF	IN/OUT	Achsdatenstruktur
Busy	BOOL	OUT	Zustand des Funktionsbausteins (Busy = True: Lesen und Schreiben noch nicht beendet)
Done	BOOL	OUT	Zustand des Funktionsbausteins (Done = True: Lesen und Schreiben erfolgreich abgeschlossen)
Error	BOOL	OUT	Zustand des Funktionsbausteins (Error = True: Lesen/Schreiben fehlerhaft)
ErrorID	UDINT	OUT	Beckhoff-spezifischer ADS Error Code der intern verwendeten Funktionsbausteine
FBErrorID	eFBError	OUT	Bausteinspezifischer Fehler-Code (siehe eFBError (ENUM) [▶ 48])

Tab. 15: Funktionsbaustein STOBER_PRM_LoadMatrix_File: Parameter

Informationen zu den Datentypen entnehmen Sie der Dokumentation zu TwinCAT 3 unter https://infosys.beckhoff.com/content/1031/tc3_plc_intro/2529388939.html.

Aufbau des Dateinamens

Beispiel 1: Achse 2_0_3B96214A.json

Wert im Beispiel	Bedeutung
Achse 2	TwinCAT-Achsidentifikation
0	Achse des Antriebsreglers (0 = Achse A, 1 = Achse B)
3B96214A	Dateiname aus Parameter R106

Tab. 16: Beispiel 1: Aufbau des Dateinamens

Beispiel 2: Achse 2_0_PRM.json

Wert im Beispiel	Bedeutung
Achse 2	TwinCAT-Achsidentifikation
0	Achse des Antriebsreglers (0 = Achse A, 1 = Achse B)
PRM	Automatisch vergebener Wert, wenn Parameter R106 nicht zur Verfügung steht

Tab. 17: Beispiel 2: Aufbau des Dateinamens

6.11.1 Ablauf des Auslesens und Schreibens

Während der Ausführung des Funktionsbausteins STÖBER_PRM_LoadMatrix_File werden die folgenden Schritte durchlaufen:

1. Lesen der Achsdaten (ADS Zugriffsdaten, z.B. AMS NetID, SubDevice-Adresse, Achstyp, ...)
2. Auslesen des Predictive-Maintenance-Status (R100)
3. Auslesen der Last-Matrix
4. StringLength der Last-Matrix berechnen
5. Dateinamen der aus dem Antriebsregler ausgelesenen Last-Matrix ermitteln (R106)
6. Last-Matrix mit StringLength als Datei im Verzeichnis unter dem übergebenen Dateipfad ablegen

6.11.2 Beispiel-Code

Das folgende Beispielprojekt dient der Implementierung in Strukturiertem Text (ST).

```
PROGRAM MAIN
VAR
    lSTOBER_LoadMatrix : STOBER_PRM_LoadMatrix_File;
    lAxis: AXIS_REF;
    lExecute: BOOL;
    lBusy: BOOL;
    lDone: BOOL;
    lError: BOOL;
    lErrorID: UDINT;
    lFLErrorID: STOBER_G6_Util.eFLError;
    Filepath: T_MaxString;
END_VAR

lSTOBER_LoadMatrix(Axis:=lAxis,
    Execute:=lExecute,
    Filepath:=Filepath,
    Busy=>lBusy,
    Done=>lDone,
    Error=>lError,
    ErrorID=>lErrorID,
    FLErrorID=>lFLErrorID);
```

6.12 STOBER_PRM_LoadMatrix_File_AMS

Der Funktionsbaustein liest die Last-Matrix aus dem Antriebsregler aus (R118) und legt sie als Datei im Verzeichnis unter dem übergebenen Dateipfad ab.

Voraussetzungen

- TwinCAT 3 ab Version 3.1.4024.40
- Bibliothek STOBER_G6_Util ab Version 3.1.2.3

Information

Um die Last-Matrix auslesen zu können, muss die Predictive-Maintenance-Funktion im Antriebsregler aktiv sein.

Parameter



Abb. 12: Funktionsbaustein STOBER_PRM_LoadMatrix_File_AMS: Ein- und Ausgangsparameter

Parameter	Datentyp	Deklaration	Beschreibung
Execute	BOOL	IN	Aktivieren des Funktionsbausteins mit steigender Flanke
Filepath	T_MaxString	IN	Dateipfad zum Verzeichnis, in das die Datei geschrieben werden soll
AmsNetId_EtherCAT_MainDevice	T_AmsNetID	IN	AMS NetID des EtherCAT MainDevices
SubDeviceAddr	UINT	IN	Adresse des EtherCAT SubDevices
AxisNumber	USINT	IN	Achse, aus der die Last-Matrix ausgelesen werden soll: <ul style="list-style-type: none"> ▪ 0 = Achse A ▪ 1 = Achse B
Busy	BOOL	OUT	Zustand des Funktionsbausteins (Busy = True: Lesen und Schreiben noch nicht beendet)
Done	BOOL	OUT	Zustand des Funktionsbausteins (Done = True: Lesen und Schreiben erfolgreich abgeschlossen)
Error	BOOL	OUT	Zustand des Funktionsbausteins (Error = True: Lesen/Schreiben fehlerhaft)
ErrorID	UDINT	OUT	Beckhoff-spezifischer ADS Error Code der intern verwendeten Funktionsbausteine
FBErrorID	eFBError	OUT	Bausteinspezifischer Fehler-Code (siehe eFBError (ENUM) [► 48])

Tab. 18: Funktionsbaustein STOBER_PRM_LoadMatrix_File_AMS: Parameter

Informationen zu den Datentypen entnehmen Sie der Dokumentation zu TwinCAT 3 unter https://infosys.beckhoff.com/content/1031/tc3_plc_intro/2529388939.html.

Aufbau des Dateinamens

Beispiel 1: Achse 2_0_3B96214A.json

Wert im Beispiel	Bedeutung
Achse 2	TwinCAT-Achsidentifikation
0	Achse des Antriebsreglers (0 = Achse A, 1 = Achse B)
3B96214A	Dateiname aus Parameter R106

Tab. 19: Beispiel 1: Aufbau des Dateinamens

Beispiel 2: Achse 2_0_PRM.json

Wert im Beispiel	Bedeutung
Achse 2	TwinCAT-Achsidentifikation
0	Achse des Antriebsreglers (0 = Achse A, 1 = Achse B)
PRM	Automatisch vergebener Wert, wenn Parameter R106 nicht zur Verfügung steht

Tab. 20: Beispiel 2: Aufbau des Dateinamens

6.12.1 Ablauf des Auslesens und Schreibens

Während der Ausführung des Funktionsbausteins STÖBER_PRM_LoadMatrix_File_AMS werden die folgenden Schritte durchlaufen:

1. Auslesen des Predictive-Maintenance-Status (R100)
2. Auslesen der Last-Matrix
3. StringLength der Last-Matrix berechnen
4. Dateinamen der aus dem Antriebsregler ausgelesenen Last-Matrix ermitteln (R106)
5. Last-Matrix mit StringLength als Datei im Verzeichnis unter dem übergebenen Dateipfad ablegen

6.12.2 Beispiel-Code

Das folgende Beispielprojekt dient der Implementierung in Strukturiertem Text (ST).

```
PROGRAM MAIN
VAR
    lSTOBER_LoadMatrix : STOBER_PRM_LoadMatrix_File_AMS;
    lAMS_NetID: T_AmsNetID;
    lSubDeviceAddr: UINT;
    lAxisNumber: USINT;
    lExecute: BOOL;
    lBusy: BOOL;
    lDone: BOOL;
    lError: BOOL;
    lErrorID: UDINT;
    lFBErrorID: STOBER_G6_Util.eFBError;
    Filepath: T_MaxString;
END_VAR

lSTOBER_LoadMatrix(AmsNetId_EtherCAT_MainDevice:=lAMS_NetID,
    SubDeviceAddr:=lSubDeviceAddr,
    Execute:=lExecute,
    AxisNumber:=lAxisNumber,
    Filepath:=Filepath,
    Busy=>lBusy,
    Done=>lDone,
    Error=>lError,
    ErrorID=>lErrorID,
    FBErrorID=>lFBErrorID);
```

6.13 STOBER_SDO_Info

Der Funktionsbaustein ermittelt, ob im Antriebsregler der Service SDO Info aktiv ist. Dies erleichtert Ihnen die Adressierung der Elemente von Array- und Record-Parametern. Bei inaktivem SDO Info entspricht der Subindex 0 eines SDO-Objekts dem Element 0 des Parameters. Bei aktivem SDO Info entspricht der Subindex 0 eines SDO-Objekts der Anzahl der Elemente des Parameters, der Subindex 1 dem Element 0 des Parameters.

Voraussetzungen

- TwinCAT 3 ab Version 3.1.4024.40
- Bibliothek STOBER_G6_Util ab Version 3.1.2.0

Parameter



Abb. 13: Funktionsbaustein STOBER_SDO_Info: Ein- und Ausgangsparameter

Parameter	Datentyp	Deklaration	Beschreibung
Execute	BOOL	IN	Aktivieren des Funktionsbausteins mit steigender Flanke
AmsNetId_EtherCAT_MainDevice	T_AmsNetID	IN	AMS NetID des EtherCAT MainDevices
SubDeviceAddr	UINT	IN	Adresse des EtherCAT SubDevices
Busy	BOOL	OUT	Zustand des Funktionsbausteins (Busy = True: Prüfung noch nicht beendet)
Done	BOOL	OUT	Zustand des Funktionsbausteins (Done = True: Prüfung abgeschlossen)
Error	BOOL	OUT	Zustand des Funktionsbausteins (Error = True: Prüfung fehlerhaft)
ErrorID	UDINT	OUT	Beckhoff-spezifischer ADS Error Code der intern verwendeten Funktionsbausteine
SDO_Info	USINT	OUT	Zustand Service SDO Info im Antriebsregler: <ul style="list-style-type: none"> ▪ 0 = inaktiv ▪ 1 = aktiv

Tab. 21: Funktionsbaustein STOBER_SDO_Info: Parameter

Informationen zu den Datentypen entnehmen Sie der Dokumentation zu TwinCAT 3 unter https://infosys.beckhoff.com/content/1031/tc3_plc_intro/2529388939.html.

6.13.1 Beispiel-Code

Das folgende Beispielprojekt dient der Implementierung in Strukturiertem Text (ST).

```
PROGRAM MAIN
VAR
    lSTOBER_SDO : STOBER_SDO_Info;
    AMS_NetID:T_AmsNetID;
    SubDeviceAddress: UINT;
    lExecute: BOOL;
    lBusy: BOOL;
    lDone: BOOL;
    lError: BOOL;
    lErrorID: UDINT;
    SDO_INFO: USINT;
END_VAR

lSTOBER_SDO(AmsNetId_EtherCAT_MainDevice:=AMS_NetID,
    SubDeviceAddr:=SubDeviceAddress,
    Execute:=lExecute,
    Busy=>lBusy,
    Done=>lDone,
    Error=>lError,
    ErrorID=>lErrorID,
    SDO_Info=>SDO_INFO);
```

7 Berechnung des Index

Über die Koordinaten des Parameters (Gruppe und Zeile) können Sie den Index für die Funktionsbausteine STÖBER_Action oder STÖBER_Power_Action berechnen.

Information

Der **Index** muss in dem von der Steuerung geforderten Format angegeben werden.

Information

Die nachfolgend beschriebene Berechnung ist nur gültig für die Umrechnung der herstellerspezifischen Parameter.

Die Achsen unterscheiden sich durch einen Offset von 8000 hex. Die Zeile des Parameters muss kleiner 512 sein. Der Index berechnet sich aus der Gruppe und Zeile des Parameters nach folgenden Formeln:

- Index Achse A = 8192 + (Nummer der Gruppe × 512) + Nummer der Zeile
- Index Achse B = 40960 + (Nummer der Gruppe × 512) + Nummer der Zeile

Berechnungsbeispiel für Achse A

Berechnung für Parameter I38:

Nummer der Gruppe = 8

Nummer der Zeile = 38

Index = 8192 + (8 × 512) + 38 = 12326 = 3026 hex

Nummer der Gruppe

Nachfolgender Tabelle entnehmen Sie die Nummer der Gruppe, die für die Berechnung des Index erforderlich ist.

Gruppe	Nummer
A: Antriebsregler	0
B: Motor	1
C: Maschine	2
D: Sollwert	3
E: Anzeigen	4
F: Klemmen	5
G: Technologie	6
H: Encoder	7
I: Motion	8
J: Fahrsätze	9
K: Steuertafel	10
M: Profile	12
P: Kundenspezifische Parameter	15
Q: Kundenspezifische Parameter, instanzabhängig	16
R: Fertigungsdaten	17
S: Sicherheit	18
T: Scope	19
U: Schutzfunktionen	20
Z: Störungszähler	25

Tab. 22: Funktionsbaustein STÖBER_Action: Parametergruppen und -nummern für die Indexberechnung

8 Diagnose

Zur Diagnose im Fehlerfall (Error = True) geben die Funktionsbausteine über die Ausgänge ErrorID und FLErrorID Fehler-Codes aus.

Handelt es sich um einen Fehler innerhalb der TwinCAT-spezifischen Funktionsblöcke, wird der ADS Error Code am Ausgang ErrorID ausgegeben. Diese Fehler-Codes können über die Dokumentation der Firma Beckhoff Automation GmbH & Co. KG referenziert werden.

Handelt es sich um einen bausteinspezifischen Fehler, wird dieser am Ausgang FLErrorID ausgegeben.

Log-Dateien zur erweiterten Diagnose

Bei der Ausführung der Funktionsbausteine STOBER_Backup_Restore_Initiator oder STOBER_Backup_Restore werden in Ihrem Projektverzeichnis verschiedene Log-Dateien erstellt, die im Fehlerfall der erweiterten Diagnose dienen.

Datei	Beschreibung
Datei im Verzeichnis log	Das Verzeichnis wird automatisch erstellt, sobald einer der beiden Funktionsbausteine das erste Mal ausgeführt wird. Jedes Mal, wenn der Funktionsbaustein STOBER_Backup_Restore ausgeführt wird, wird in diesem Verzeichnis eine Log-Datei mit den Informationen zum Skriptmodus gespeichert.
Datei Tc_Log.log	Bausteinspezifische Log-Datei mit Informationen, für welches EtherCAT SubDevice einer der beiden Funktionsbausteine ausgeführt wurde. Wurde der Funktionsbaustein STOBER_Backup_Restore ausgeführt, wird zusätzlich der Servicetype protokolliert. Ferner enthält die Datei Informationen, ob die Ausführung erfolgreich war.
Datei DeviceInfo.txt	Automatisch erstellte Datei mit allen SubDevice-Adressen und Produktionsnummern der Antriebsregler im EtherCAT-Netzwerk. ACHTUNG! Diese Datei darf nicht geändert oder gelöscht werden.
Dateien im Verzeichnis SaveOriginals	Das Verzeichnis wird automatisch erstellt, sobald ein Backup-Service das erste Mal ausgeführt wird. Bei jedem Backup-Service wird die aktuelle DS6-Projektdatei in das Verzeichnis SaveOriginals verschoben. Die Datei wird mit dem aktuellen Datum und der Uhrzeit versehen.

Tab. 23: Funktionsbaustein STOBER_Backup_Restore_Initiator oder STOBER_Backup_Restore: Log-Dateien im Projektverzeichnis

8.1 eFBError (ENUM)

Fehler (FBErrorID)	Nr.	Ursache	Prüfung und Maßnahmen
NO_ERROR	0	Kein bausteinspezifischer Fehler aufgetreten	—
HOMING_METHOD_INACTIVE	1	Parameter A586 = 0: Inaktiv	Wählen Sie in Parameter A586 die Referenziermethode aus, die beim Ausführen des Bausteins durchgeführt werden soll.
AXIS_NOT_ENABLED	2	Achse ist nicht freigegeben	Geben Sie die Achse des Antriebsreglers frei (Ursache Einschaltsperr: E47).
WRONG_MODE	3	Falsche Betriebsart für antriebsreglergeführte Referenzierung	Wechseln Sie zu einer der folgenden Betriebsarten (A541): <ul style="list-style-type: none"> ▪ 8: Cyclic synchronous position mode ▪ 9: Cyclic synchronous velocity mode ▪ 10: Cyclic synchronous torque mode ▪ 6: Homing mode
NO_DEVICE_LINK	4	Wechsel der Betriebsart zu 6: Homing mode hat nicht funktioniert	—
AXIS_ERROR	5	Antriebsregler oder Achse ist in Störung	Beheben Sie die Störungsursache und quittieren Sie die Störung.
COMMAND_ABORTED	6	Referenzierung wurde abgebrochen	Der Antriebsregler ist während der Referenzierung in Störung gegangen oder die Freigabe wurde entfernt. Beheben Sie die Störungsursache und quittieren Sie die Störung. Der Antriebsregler muss freigegeben sein, bis die Referenzierung abgeschlossen ist.
HOMING_TIMEOUT	7	Referenzierung konnte nicht in der angegebenen Zeit abgeschlossen werden	Verlängern Sie die angegebene Zeitspanne. Prüfen Sie, ob die korrekte Referenziermethode ausgewählt ist und die Bedingungen für die Referenzierung erfüllt werden können.
WRONG_AMS_NETID	100	Falsche AMS NetID	Verknüpfen Sie die AMS NetID und die zugehörige Variable vom Typ T_AmsNetId.
WRONG_FILE_PATH	101	Der angegebene Dateipfad zum Verzeichnis der Projektdatei konnte nicht gefunden werden	Prüfen und korrigieren Sie den Dateipfad zum Verzeichnis der Projektdatei auf der EtherCAT-Steuerung.
WRONG_DS6_PROCESS_PATH	103	Der angegebene Dateipfad zum Verzeichnis der DriveControlSuite konnte nicht gefunden werden	Prüfen und korrigieren Sie den Dateipfad zum Verzeichnis der DriveControlSuite auf der EtherCAT-Steuerung.
WRONG_ETHERCAT_REVISION	105	EtherCAT-Revisionsnummer < 6000	Revisionsnummer des Antriebsreglers < 6000 (Kommunikationsobjekt Revision Number nach CiA 301; Objekt 1018 hex, Subindex 3 hex). Erstellen Sie ein DS6-Projekt mit einem aktuellen EtherCAT-Template.
INVALID_BOXNAME	107	Parameter A251 hat keinen gültigen Wert	Führen Sie den Funktionsbaustein STÖBER_BoxName aus.

Fehler (FErrorID)	Nr.	Ursache	Prüfung und Maßnahmen
WRONG_SERVICE_TYPE_INPUT	109	Angefragter Service stimmt nicht mit dem vom Funktionsbaustein STÖBER_Backup_Restore angeforderten Servicetype überein	Wählen Sie für Servicetype den richtigen Service entsprechend Ausgang iAction des Funktionsbausteins STÖBER_Backup_Restore_Initiator. Bei der ersten Verwendung des Funktionsbausteins muss der Servicetype Restore gewählt werden (Servicetype = 2) .
TIMEOUT_EXCEED	111	Skriptmodus konnte nicht in vorgegebener Zeit ausgeführt werden	Stellen Sie sicher, dass die DriveControlSuite auf der EtherCAT-Steuerung nicht geöffnet ist, während Sie den Funktionsbaustein ausführen. Stellen Sie sicher, dass nicht mehrere Instanzen des Funktionsbausteins gleichzeitig ausgeführt werden, sondern führen Sie die Instanzen nacheinander aus.
NO_STÖBER_SUBDEVICE_IN_CONFIGURATION	113	In der TwinCAT-Konfiguration konnte kein STÖBER SubDevice gefunden werden	Stellen Sie sicher, dass es einen STÖBER Antriebsregler in der TwinCAT-Konfiguration gibt.
ERROR_SAVING_VALUES	115	Aktion Werte speichern auf dem Antriebsregler hat einen Fehler zurückgeliefert	Prüfen Sie über Parameter E61[0], ob eine SD-Karte bzw. ein Paramodul gesteckt ist und erkannt wurde. Prüfen Sie über E68, ob das Speichermedium gegebenenfalls schreibgeschützt ist.
UNEXPECTED_VALUE	117	Ein übergebener Parameter hat einen unerwarteten Wert	—
DEVICE_INFO_TXT_TOO_LONG	119	DeviceInfo.txt enthält mehr als 200 Antriebsregler	Reduzieren Sie die Anzahl der STÖBER Antriebsregler auf maximal 200.
MAX_SUBDEVICES_NUMBER_REACHED	200	Maximale Anzahl der verbundenen EtherCAT SubDevices überschritten	Reduzieren Sie die Anzahl der mit dem Funktionsbaustein verbundenen EtherCAT SubDevices auf maximal 2000.
PROJECT_NOT_FOUND	201	Antriebsregler wurde nicht im Projekt gefunden	Der Antriebsregler, für den der Funktionsbaustein STÖBER_Backup_Restore ausgeführt werden soll, konnte in keiner Datei (*.ds6) im Projektordner gefunden werden. Stellen Sie sicher, dass Sie den SPS-Gerätenamen aus Ihrem EtherCAT-Projekt in das DS6-Projekt eingetragen haben. Prüfen Sie die Log-Dateien in Ihrem Projektverzeichnis für weitere Informationen.
MORE_THAN_ONE_PROJECT_FOUND	203	Antriebsregler wurde in mehreren Projekten gefunden	Der Antriebsregler, für den der Funktionsbaustein STÖBER_Backup_Restore ausgeführt werden soll, wurde in mehreren Projekten gefunden. Prüfen Sie die Log-Dateien in Ihrem Projektverzeichnis für weitere Informationen.
OPEN_PROJECT_ERROR	205	Projekt konnte nicht geöffnet werden	Stellen Sie sicher, dass das angegebene DS6-Projekt nicht bereits geöffnet ist. Prüfen Sie die Log-Dateien in Ihrem Projektorder für weitere Informationen.

Fehler (FErrorID)	Nr.	Ursache	Prüfung und Maßnahmen
CONNECTION_ERROR	207	Verbindungsfehler	Verbindung zwischen Steuerungs-PC und Antriebsregler konnte nicht hergestellt werden. Prüfen Sie die Ethernet-Kabelverbindung zwischen Steuerungs-PC und Antriebsregler (Service-Schnittstelle X9).
ONLINE_ERROR	209	Fehlerhafte Projektierung	Stellen Sie sicher, dass Firmware-Version, Antriebsregler und Optionsmodul in Ihrem DS6-Projekt korrekt projektiert sind.
ALL_SUBDEVICES_NOT_IN_OPERATIONAL_MODE	300	Nicht alle STÖBER SubDevices im Zustand Operational	Stellen Sie sicher, dass sich alle EtherCAT SubDevices im Zustand Operational befinden. Der Fehler bleibt aktiv, bis alle SubDevices betriebsbereit sind.
INVALID_OFFSET	301	Ungültiger Offset beim Auslesen von R118	—
WRONG_AXIS	302	Falsche Antriebsregler-Achse verbunden	—
PREDICTIVE_MAINTENANCE_INACTIVE	303	Predictive Maintenance ist inaktiv	Aktivieren Sie für den Antriebsregler Predictive Maintenance und führen Sie die Aktion Werte speichern (A00) sowie einen Neustart des Antriebsreglers aus.
JSON_LENGTH_ZERO	304	Last-Matrix konnte nicht korrekt ausgelesen werden	Prüfen Sie, ob Predictive Maintenance aktiviert ist.
JSON_INCOMPLETE	305	Last-Matrix konnte nicht komplett ausgelesen werden	Prüfen Sie, ob es konkurrierende Zugriffe auf Parameter R118 gibt, z. B. durch die DS6.
MASTER_NOT_IN_OPERATIONAL_MODE	400	MainDevice nicht im Zustand Operational	Stellen Sie sicher, dass sich das EtherCAT MainDevice im Zustand Operational befindet.
ABORTED	401	Freigabe-Aus durch Anwender oder Gerätestörung	—
TIMEOUT	402	Freigabe-Ein nicht innerhalb von 30 s nach Start der Aktion aktiviert	Prüfen Sie die Rahmenbedingungen.
ILLEGAL	403	Aktivierung der Aktion im Zustand ≠ 2: Einschaltbereit (E48), Betrieb des Motors in Steuerart ohne Vektorregelung und Motorencoder (B20, B26) oder Verwendung der Bremsen mit inaktiver Bremsenansteuerung (F00)	Prüfen Sie die Rahmenbedingungen.
AXIS_LOAD	404	Achse hat sich bei Freigabe-Ein und lüftend angesteuerten Bremsen bewegt	—
PHASE_ORDER	405	Phasenfolge passt nicht zur Zählrichtung des Motorencoders	Prüfen Sie die Rahmenbedingungen.

Fehler (FErrorID)	Nr.	Ursache	Prüfung und Maßnahmen
MOTOR_POLES	406	Elektrisch vorgegebene Strecke passt nicht zur mechanisch zurückgelegten Strecke	Prüfen und korrigieren Sie gegebenenfalls folgende Einstellungen: <ul style="list-style-type: none"> ▪ Polzahl des Synchron-Servomotors oder Asynchronmotors und Einstellung in B10 ▪ Polteilung des Linearmotors und Einstellung in B16 ▪ Motorencoder-Skalierung (Wertepaar Zähler/Nenner) ▪ Keine mechanische Blockade des Abtriebs ▪ Ausreichendes Drehmoment/ausreichende Kraft für Testlauf
TEST_RUN	408	Testlauf mit ermitteltem Kommutierungsoffset nicht erfolgreich	—
REMOVE_ENABLE	498	Freigabe bereits bei Start der Aktion aktiv	Entfernen Sie die Freigabe der Achse.
AXIS_ENABLED	410	Freigegebene Achse beim Ausführen des Funktionsbausteins	Entfernen Sie die Freigabe der Achse.
GENERAL_ERROR	413	—	—

Tab. 24: Bausteinspezifische Fehler: eFError (ENUM)

9 Anhang

9.1 Weiterführende Informationen

Die nachfolgend gelisteten Dokumentationen liefern Ihnen weitere relevante Informationen zur 6. STÖBER Antriebsreglergeneration. Den aktuellen Stand der Dokumentationen finden Sie in unserem Download-Center unter: <http://www.stoeber.de/de/downloads/>.

Geben Sie die ID der Dokumentation in die Suche ein.

Antriebsregler SC6, SI6

Titel	Dokumentation	Inhalte	ID
Antriebsregler SC6	Handbuch	Systemaufbau, technische Daten, Projektierung, Lagerung, Einbau, Anschluss, Inbetriebnahme, Betrieb, Service, Diagnose	442789
Anreihetechnik mit SI6 und PS6	Handbuch	Systemaufbau, technische Daten, Projektierung, Lagerung, Einbau, Anschluss, Inbetriebnahme, Betrieb, Service, Diagnose	442727
Kommunikation EtherCAT – SC6, SI6	Handbuch	Elektrische Installation, Datentransfer, Inbetriebnahme, Diagnose, weiterführende Informationen	443024
Applikation CiA 402 – SC6, SI6	Handbuch	Projektierung, Konfiguration, Parametrierung, Funktionstest, weiterführende Informationen	443079

Antriebsregler SD6

Titel	Dokumentation	Inhalte	ID
Antriebsregler SD6	Handbuch	Systemaufbau, technische Daten, Projektierung, Lagerung, Einbau, Anschluss, Inbetriebnahme, Betrieb, Service, Diagnose	442425
Kommunikation EtherCAT – SD6	Handbuch	Einbau, elektrische Installation, Datentransfer, Inbetriebnahme, Diagnose, weiterführende Informationen	442515
Applikation CiA 402 – SD6	Handbuch	Projektierung, Konfiguration, Parametrierung, Funktionstest, weiterführende Informationen	443076

Eine kostenfreie Basisversion der Automatisierungssoftware TwinCAT 3 erhalten Sie unter <https://www.beckhoff.com/de-de/produkte/automation/twincat/texxxx-twincat-3-engineering/te1000.html>.

9.2 Abkürzungen

Abkürzung	Bedeutung
ADS	Automation Device Specification
AMS	Automation Message Specification
CiA	CAN in Automation
CoE	CANopen over EtherCAT
EtherCAT	Ethernet for Control Automation Technology
IP	Internet Protocol (dt.: Internetprotokoll)
MDevice	MainDevice
NC	Numerical Control (numerische Steuerung)
PDO	Process Data Objects (Prozessdaten-Objekte)
PLC	Programmable Logic Controller (speicherprogrammierbare Steuerung)
SDO	Service Data Objects (Servicedaten-Objekte)
SPS	Speicherprogrammierbare Steuerung
ST	Strukturierter Text
SubDevice	SubordinateDevice
TwinCAT	The Windows Control and Automation Technology

10 Kontakt

10.1 Beratung, Service, Anschrift

Wir helfen Ihnen gerne weiter!

Auf unserer Webseite stellen wir Ihnen zahlreiche Informationen und Dienstleistungen rund um unsere Produkte bereit:

<http://www.stoeber.de/de/service>

Für darüber hinausgehende oder individuelle Informationen, kontaktieren Sie unseren Beratungs- und Support-Service:

<http://www.stoeber.de/de/support>

Sie benötigen unseren System-Support:

Fon +49 7231 582-3060

systemsupport@stoeber.de

Sie benötigen ein Ersatzgerät:

Fon +49 7231 582-1128

replace@stoeber.de

So erreichen Sie unsere 24 h Service-Hotline:

Fon +49 7231 582-3000

Unsere Anschrift lautet:

STÖBER Antriebstechnik GmbH + Co. KG

Kieselbronner Straße 12

75177 Pforzheim, Germany

10.2 Ihre Meinung ist uns wichtig

Diese Dokumentation erstellen wir nach bestem Wissen mit dem Ziel, Sie beim Auf- und Ausbau Ihres Know-hows rund um unser Produkt nutzbringend und effizient zu unterstützen.

Ihre Anregungen, Meinungen, Wünsche und konstruktive Kritik helfen uns, die Qualität unserer Dokumentation sicherzustellen und weiterzuentwickeln.

Wenn Sie uns aus genannten Gründen kontaktieren möchten, freuen wir uns über eine E-Mail an:

documentation@stoeber.de

Vielen Dank für Ihr Interesse.

Ihr STÖBER Redaktionsteam

10.3 Weltweite Kundennähe

Wir beraten und unterstützen Sie mit Kompetenz und Leistungsbereitschaft in über 40 Ländern weltweit:

STOBER AUSTRIA

www.stoerber.at
+43 7613 7600-0
sales@stoerber.at

STOBER FRANCE

www.stoerber.fr
+33 478 98 91 80
sales@stoerber.fr

STOBER HUNGARY

www.stoerber.de
+36 53 5011140
info@emtc.hu

STOBER JAPAN

www.stoerber.co.jp
+81-3-5875-7583
sales@stoerber.co.jp

STOBER TAIWAN

www.stoerber.tw
+886 4 2358 6089
sales@stoerber.tw

STOBER UK

www.stoerber.co.uk
+44 1543 458 858
sales@stoerber.co.uk

STOBER CHINA

www.stoerber.cn
+86 512 5320 8850
sales@stoerber.cn

STOBER Germany

www.stoerber.de
+49 4 7231 582-0
sales@stoerber.de

STOBER ITALY

www.stoerber.it
+39 02 93909570
sales@stoerber.it

STOBER SWITZERLAND

www.stoerber.ch
+41 56 496 96 50
sales@stoerber.ch

STOBER TURKEY

www.stoerber.com
+90 216 510 2290
sales-turkey@stoerber.com

STOBER USA

www.stoerber.com
+1 606 759 5090
sales@stoerber.com

Tabellenverzeichnis

Tab. 1	Funktionsbausteine für TwinCAT 3	10
Tab. 2	Funktionsbaustein STOBER_BoxName: Parameter	11
Tab. 3	Funktionsbaustein STOBER_Backup_Restore_Initiator: Parameter	13
Tab. 4	Funktionsbaustein STOBER_Backup_Restore: Parameter	15
Tab. 5	Funktionsbaustein STOBER_Backup_Restore: Servicetype	15
Tab. 6	Funktionsbaustein STOBER_MC_HOME: Parameter	20
Tab. 7	Funktionsbaustein STOBER_MC_HOME_REF: Parameter	23
Tab. 8	Funktionsbaustein STOBER_Action: Parameter	26
Tab. 9	Funktionsbaustein STOBER_Action: Beispiele für Aktionen ohne erforderliche Freigabe	27
Tab. 10	Funktionsbaustein STOBER_Power_Action: Parameter	29
Tab. 11	Funktionsbaustein STOBER_Power_Action: Beispiele für Aktionen mit erforderlicher Freigabe	29
Tab. 12	Funktionsbaustein STOBER_Phase_Test: Parameter	32
Tab. 13	Funktionsbaustein STOBER_PRM_LoadMatrix: Parameter	34
Tab. 14	Funktionsbaustein STOBER_PRM_LoadMatrix_AMS: Parameter	36
Tab. 15	Funktionsbaustein STOBER_PRM_LoadMatrix_File: Parameter	38
Tab. 16	Beispiel 1: Aufbau des Dateinamens	39
Tab. 17	Beispiel 2: Aufbau des Dateinamens	39
Tab. 18	Funktionsbaustein STOBER_PRM_LoadMatrix_File_AMS: Parameter	41
Tab. 19	Beispiel 1: Aufbau des Dateinamens	42
Tab. 20	Beispiel 2: Aufbau des Dateinamens	42
Tab. 21	Funktionsbaustein STOBER_SDO_Info: Parameter	44
Tab. 22	Funktionsbaustein STOBER_Action: Parametergruppen und -nummern für die Indexberechnung	46
Tab. 23	Funktionsbaustein STOBER_Backup_Restore_Initiator oder STOBER_Backup_Restore: Log-Dateien im Projektverzeichnis	47
Tab. 24	Bausteinspezifische Fehler: eFBError (ENUM)	48



4 4 3 3 7 0 . 0 1

05/2024

STÖBER Antriebstechnik GmbH + Co. KG
Kieselbronner Str. 12
75177 Pforzheim
Germany
Tel. +49 7231 582-0
mail@stoeber.de
www.stober.com

24 h Service Hotline
+49 7231 582-3000

www.stober.com