

SIEMENS



Bibliotheksbeschreibung • 01/2017

SIMATIC EMS400S Funktionsbibliotheken

EMS400S

<http://support.automation.siemens.com/WW/view/de/89369337>

Gewährleistung und Haftung

Hinweis

Die Applikationsbeispiele sind unverbindlich und erheben keinen Anspruch auf Vollständigkeit hinsichtlich Konfiguration und Ausstattung sowie jeglicher Eventualitäten. Die Applikationsbeispiele stellen keine kundenspezifischen Lösungen dar, sondern sollen lediglich Hilfestellung bieten bei typischen Aufgabenstellungen. Sie sind für den sachgemäßen Betrieb der beschriebenen Produkte selbst verantwortlich. Diese Applikationsbeispiele entheben Sie nicht der Verpflichtung zu sicherem Umgang bei Anwendung, Installation, Betrieb und Wartung. Durch Nutzung dieser Applikationsbeispiele erkennen Sie an, dass wir über die beschriebene Haftungsregelung hinaus nicht für etwaige Schäden haftbar gemacht werden können. Wir behalten uns das Recht vor, Änderungen an diesen Applikationsbeispielen jederzeit ohne Ankündigung durchzuführen. Bei Abweichungen zwischen den Vorschlägen in diesem Applikationsbeispiel und anderen Siemens Publikationen, wie z.B. Katalogen, hat der Inhalt der anderen Dokumentation Vorrang.

Für die in diesem Dokument enthaltenen Informationen übernehmen wir keine Gewähr.

Unsere Haftung, gleich aus welchem Rechtsgrund, für durch die Verwendung der in diesem Applikationsbeispiel beschriebenen Beispiele, Hinweise, Programme, Projektierungs- und Leistungsdaten usw. verursachte Schäden ist ausgeschlossen, soweit nicht z.B. nach dem Produkthaftungsgesetz in Fällen des Vorsatzes, der groben Fahrlässigkeit, wegen der Verletzung des Lebens, des Körpers oder der Gesundheit, wegen einer Übernahme der Garantie für die Beschaffenheit einer Sache, wegen des arglistigen Verschweigens eines Mangels oder wegen Verletzung wesentlicher Vertragspflichten zwingend gehaftet wird. Der Schadensersatz wegen Verletzung wesentlicher Vertragspflichten ist jedoch auf den vertragstypischen, vorhersehbaren Schaden begrenzt, soweit nicht Vorsatz oder grobe Fahrlässigkeit vorliegt oder wegen der Verletzung des Lebens, des Körpers oder der Gesundheit zwingend gehaftet wird. Eine Änderung der Beweislast zu Ihrem Nachteil ist hiermit nicht verbunden.

Weitergabe oder Vervielfältigung dieser Applikationsbeispiele oder Auszüge daraus sind nicht gestattet, soweit nicht ausdrücklich von Siemens Industry Sector zugestanden.

Security-hinweise

Siemens bietet Produkte und Lösungen mit Industrial Security-Funktionen an, die den sicheren Betrieb von Anlagen, Lösungen, Maschinen, Geräten und/oder Netzwerken unterstützen. Sie sind wichtige Komponenten in einem ganzheitlichen Industrial Security-Konzept. Die Produkte und Lösungen von Siemens werden unter diesem Gesichtspunkt ständig weiterentwickelt. Siemens empfiehlt, sich unbedingt regelmäßig über Produkt-Updates zu informieren.

Für den sicheren Betrieb von Produkten und Lösungen von Siemens ist es erforderlich, geeignete Schutzmaßnahmen (z. B. Zellschutzkonzept) zu ergreifen und jede Komponente in ein ganzheitliches Industrial Security-Konzept zu integrieren, das dem aktuellen Stand der Technik entspricht. Dabei sind auch eingesetzte Produkte von anderen Herstellern zu berücksichtigen.

Weitergehende Informationen über Industrial Security finden Sie unter <http://www.siemens.com/industrialsecurity>.

Um stets über Produkt-Updates informiert zu sein, melden Sie sich für unseren produktspezifischen Newsletter an. Weitere Informationen hierzu finden Sie unter <http://support.automation.siemens.com>.

Inhaltsverzeichnis

Gewährleistung und Haftung	2
1 Übersicht der Funktionsbibliotheken	5
1.1 Anwenderszenarien der Funktionsbibliotheken	6
1.2 Anlagensegmentsteuerung	6
1.3 Fahrzeugsteuerung	7
1.4 Bibliotheken	8
1.4.1 Basis-Bausteinbibliothek „EMS400S“	8
1.4.2 Funktionsbibliotheken „EMS400S_xxx“	8
1.4.3 Einsatzmöglichkeit für die Verwendung der Funktionsbibliotheken „EMS400S_xxx“	9
1.4.4 Bibliotheksfunktionen in dieser Beschreibung	10
1.5 Hard- und Softwarevoraussetzungen.....	12
2 Funktionsbibliothek „EMS400S_Segment“	14
2.1 Bibliotheksinhalt	14
2.2 Erläuterung der Bausteine für die Anlagensegmentsteuerung	16
2.2.1 FC_ReadWriteModule (FC30).....	16
2.2.2 FC_Errors (FC50).....	18
2.2.4 FB_DataTransfer_SR (FB31).....	20
2.2.5 FB_ProcessImageCommands (FB38)	22
2.2.6 FB_ProcessImageStatus (FB39)	23
2.2.7 Datentypen (UDTs) für Kommunikationsschnittstellen	24
2.2.8 Variablen-tabelle (Segment tags)	26
3 Funktionsbibliothek „EMS400S_Communication“	27
3.1 Bibliotheksinhalt	27
3.2 Erläuterung der Kommunikations-Bausteine.....	28
3.2.1 FC_Call_InterfaceSegmentPLC (FC30).....	28
3.2.2 FB_DataTransferEMS (FB30).....	30
3.2.3 FB_DataTransferEMS_SR (FB31)	32
3.2.4 Datentypen (UDTs) für Kommunikationsschnittstellen	34
4 Funktionsbibliothek „EMS400S_Display_RemoteControl“	36
4.1 Bibliotheksinhalt	36
4.2 Erläuterung der Bausteine.....	37
4.2.1 FC_Call_RemoteControl_Display (FC20)	37
4.2.2 FC_DispErrorSubroutine (FC22).....	40
4.2.3 FC_StationaryRemoteControl (FC23).....	41
4.2.4 DB_GlobalDataEMS, Struktur Remote_Control (DB10)	42
4.2.5 FB_RemoteControl (FB21).....	43
4.2.6 FB_DispErrors (FB22).....	48
5 Funktionsbibliothek „EMS400S_Drive“	49
5.1 Bibliotheksinhalt	50
5.2 Erläuterung der Bausteine.....	51
5.2.1 FC_Drive (FC60)	51
5.2.2 DB_GlobalDataEMS, Struktur Drive (DB10)	54
5.2.3 FC_LinearMeasurementSensor (FC61).....	58
5.2.4 FC_StopPosition (FC62)	59
5.2.5 FC_SpeedControl (FC63)	60
5.2.6 FC_SeparatorBlocks (FC67).....	61
5.2.7 DB_Drive (DB60).....	63
5.2.8 DB_StopPositions (DB62)	64
5.2.9 DB_SpeedControl (DB63)	64
5.2.10 DB_SeparatorBlocks (DB67)	64

6	Funktionsbibliothek „EMS400S_Errors“	65
6.1	Bibliotheksinhalt	65
6.2	Erläuterung der Bausteine	66
6.2.1	FB_ErrorsAll (FB50)	66
6.2.2	FB_ErrorsDrive (FB54)	68
6.2.3	FB_ErrorsHoist (FB55)	69
6.2.4	DB_ErrorCodes (DB50)	70
6.2.5	DB_PNDevice (DB53)	72
7	Funktionsbibliothek „EMS400S_General“	73
7.1	Bibliotheksinhalt	73
7.2	Erläuterung der Bausteine	74
7.2.1	FC_General (FC10)	74
7.2.2	DB_GlobalDataEMS (DB10)	75
7.2.3	Variablentabelle (CAR tags)	75
8	Funktionsbibliothek „EMS400S_Hoist“	76
8.1	Bibliotheksinhalt	77
8.2	Erläuterung der Hubantriebs-Bausteine	78
8.2.1	FC_Hoist (FC70)	78
8.2.2	DB_GlobalDataEMS, Struktur Hoist (DB10)	82
8.2.3	FC_HoistEncoder (FC71)	84
8.2.4	FC_HoistEnablePosition (FC72)	85
8.2.5	DB_Hoist (DB70)	86
8.2.6	DB_HoistPositions (DB72)	87
9	Arbeiten mit den Funktionsbibliotheken	88
9.1	Integration der Funktionsbibliotheken in TIA STEP 7	88
9.2	Aufruf der Funktionsbibliotheksbausteine „EMS400S_Segment“ im TIA STEP 7 Programm	91
9.3	Aufruf der Funktionsbibliotheksbausteine „EMS400S_Communication“ im TIA STEP 7 Programm	98
9.4	Aufruf der Funktionsbibliotheksbausteine „EMS400S_Display_RemoteControl“ im TIA STEP 7 Programm	102
9.5	Aufruf der Funktionsbibliotheksbausteine „EMS400S_Drive“ im TIA STEP 7 Programm	106
9.6	Aufruf der Funktionsbibliotheksbausteine „EMS400S_Errors“ im TIA STEP 7 Programm	110
9.7	Aufruf der Funktionsbibliotheksbausteine „EMS400S_General“ im TIA STEP 7 Programm	113
9.8	Aufruf der Funktionsbibliotheksbausteine „EMS400S_Hoist“ im TIA STEP 7 Programm	117
10	Installation und Inbetriebnahme der Profinet (PN) -Geräte	120
10.1	Sick Linearmesssensor OLM200	121
10.2	TR Trommelgeber TR_C58M	128
11	Literaturhinweise	136
12	Historie	136

1 Übersicht der Funktionsbibliotheken

Was erhalten Sie?

Das vorliegende Dokument beschreibt die Funktionsbibliotheken

- EMS400S_Segment
- EMS400S_Communication
- EMS400S_Display_RemoteControl
- EMS400S_Drive
- EMS400S_Errors
- EMS400S_General
- EMS400S_Hoist

Mit den Funktionsbibliotheken erhalten Sie getesteten Code mit eindeutig definierten Schnittstellen. Auf diese können Sie entsprechend Ihrer zu realisierender Aufgabenstellung aufsetzen.

Kernanliegen des Dokuments ist die Beschreibung

- aller zu den Funktionsbibliotheken gehörenden Bausteine.
- der durch diese Bausteine realisierten Funktionalität.

Darüber hinaus hilft Ihnen diese Dokumentation mit Step-by-Step-Anweisungen, die Bibliotheken in Ihr TIA STEP 7-Projekt zu integrieren.

Gültigkeitsbereich der Bibliotheken

- TIA STEP 7 V14 und höher
- EMS400S
- S7-1200-CPU's

Abgrenzung

Diese Applikation enthält keine Beschreibung von:

- TIA-Portal
- Programmierung im TIA-Portal
- S7-1200-CPU's
- PSB-Modulen
- Basis-Bausteinbibliothek „EMS400S“, siehe \4\
- Frequenzumrichter V20 (FU V20)
- Hubantrieb Kettenzug

Grundlegende Kenntnisse über diese Themen werden vorausgesetzt.

1.1 Anwenderszenarien der Funktionsbibliotheken

Übersicht zum SIMATIC EMS400S System

Das SIMATIC EMS400S System (siehe [\3\](#)) ist für den Transport von Lasten in Elektrohängebahn-Anlagen (EHB) ausgelegt. Zu einer Anlagensteuerung mit dem EMS400S System gehören folgende Komponenten:

- PSB-S-Modul
- PSB-C-Modul
- IR-Fernbedienung
- Anzeigeeinheit

Das PSB-S-Modul bildet im Verbund mit einer S7-1200-Zentralbaugruppe die **Anlagensegmentsteuerung**. Eine ihrer spezifischen Funktionen ist die Verarbeitung der über PROFINET empfangenen Daten und deren Einspeisung auf die Schleifleiter RAIL.

Über die Schleifleiter wird die Kommunikationsverbindung mit der Fahrzeugsteuerung hergestellt.

Das PSB-C-Modul ist in die zum mobilen Anlagenteil gehörende **Fahrzeugsteuerung** integriert. Das PSB-C-Modul übernimmt zusammen mit einer S7-1200-Zentralbaugruppe die Steuerungsaufgaben, die das EHB-Fahrzeug betreffen.

PSB-S- und PSB-C-Modul kommunizieren über einen getakteten 16-Bit-Datenrahmen. Die Kommunikation kann quitiert und nicht quitiert erfolgen. Der Dateninhalt ist im TIA-Portal frei programmierbar.

Die Kommunikation der PSB-S- und PSB-C-Module wird durch die Basis-Kommunikationsbausteine der Basis-Bausteinbibliothek „EMS400S“ realisiert, siehe Kapitel [1.4.1 Basis-Bausteinbibliothek „EMS400S“](#) und [\4\](#).

1.2 Anlagensegmentsteuerung

Eine Anlagensegmentsteuerung hat die Funktion, die Kommunikation zu den Segmenten herzustellen- aufbauend auf den Basis-Kommunikationsbausteinen der Basis-Bausteinbibliothek EMS400S, siehe [\4\](#).

- Sie leitet die Kommandos der Steuerung der EHB-Anlage an die Segmente / RAILS weiter, und damit an die Fahrzeuge, die sich in den Segmenten befinden.
- Sie stellt die Statusmeldungen der Segmente für die Steuerung der EHB-Anlage bereit.
Diese Statusmeldungen sind die Statusmeldungen der Fahrzeuge.

Hinweis

Die Kommunikation mit der Steuerung der EHB-Anlage ist nicht Bestandteil dieser Beschreibung.

1.3 Fahrzeugsteuerung

Die Funktionen einer Fahrzeugsteuerung werden mit mehreren Funktionsgruppen realisiert- aufbauend auf den Basis-Kommunikationsbausteine der Basis-Bausteinbibliothek EMS400S, siehe \4\.

- Kommunikation mit der Anlagensegmentsteuerung.
- Allgemeine Funktionen.
- Ansteuerung des Displays, Auswertung der Fernbedienung (Remote Control).
- Ansteuerung des Fahrantriebs und Auswertung des Barcodes.
- Ansteuerung des Hubantriebs und Auswertung des Gebers für die Höhe.
- Überwachung der Sensoren und Aktoren, Generierung der Störungsmeldungen.

Diese Funktionsgruppen sind in den einzelnen Bibliotheken enthalten, siehe Kapitel 1.4.2 (Funktionsbibliotheken „EMS400S_xxx) und Kapitel 2 bis 8.

Hinweis

Sie müssen die Bausteine aus den jeweiligen Funktionsbibliotheken in Ihre Fahrzeugsteuerung kopieren und entsprechend den Anforderungen Ihrer Anlage anpassen.

1.4 Bibliotheken

1.4.1 Basis-Bausteinbibliothek „EMS400S“

Die Basis-Bausteinbibliothek „EMS400S“, siehe \4\, enthält die Basis-Kommunikationsbausteine für die Anlagensegmentsteuerung und die Fahrzeugsteuerung SIMATIC EMS400S, für den Betrieb der PSB-S- und PSC-C-Module.

Diese Basis-Kommunikationsbausteine sind die Grundlage für die nachfolgend beschriebenen Funktionsbibliotheken.

1.4.2 Funktionsbibliotheken „EMS400S_xxx“

Die in dem vorliegenden Dokument beschriebenen Funktionsbibliotheken „EMS400S_xxx“ enthalten Bausteine, um ein EHB-Fahrzeug mit Fahrtrieb und Hubtrieb zu betreiben und über die Segmentsteuerungen mit der Steuerung der EHB-Anlage zu kommunizieren.

Im Detail stehen folgende Funktionsbibliotheken zur Verfügung:

- EMS400S_Segment
- EMS400S_Communication
- EMS400S_Display_RemoteControl
- EMS400S_Drive
- EMS400S_Errors
- EMS400S_General
- EMS400S_Hoist

Hinweis

Jede Funktionsbibliothek enthält mehrere Bausteine. Die Auflistungen der Bausteine finden Sie in den Kapiteln 2 bis 8.

1.4.3 Einsatzmöglichkeit für die Verwendung der Funktionsbibliotheken „EMS400S_xxx“

Die folgende Abbildung zeigt schematisch ein vereinfachtes Schienensystem mit drei EHB-Fahrzeugen.

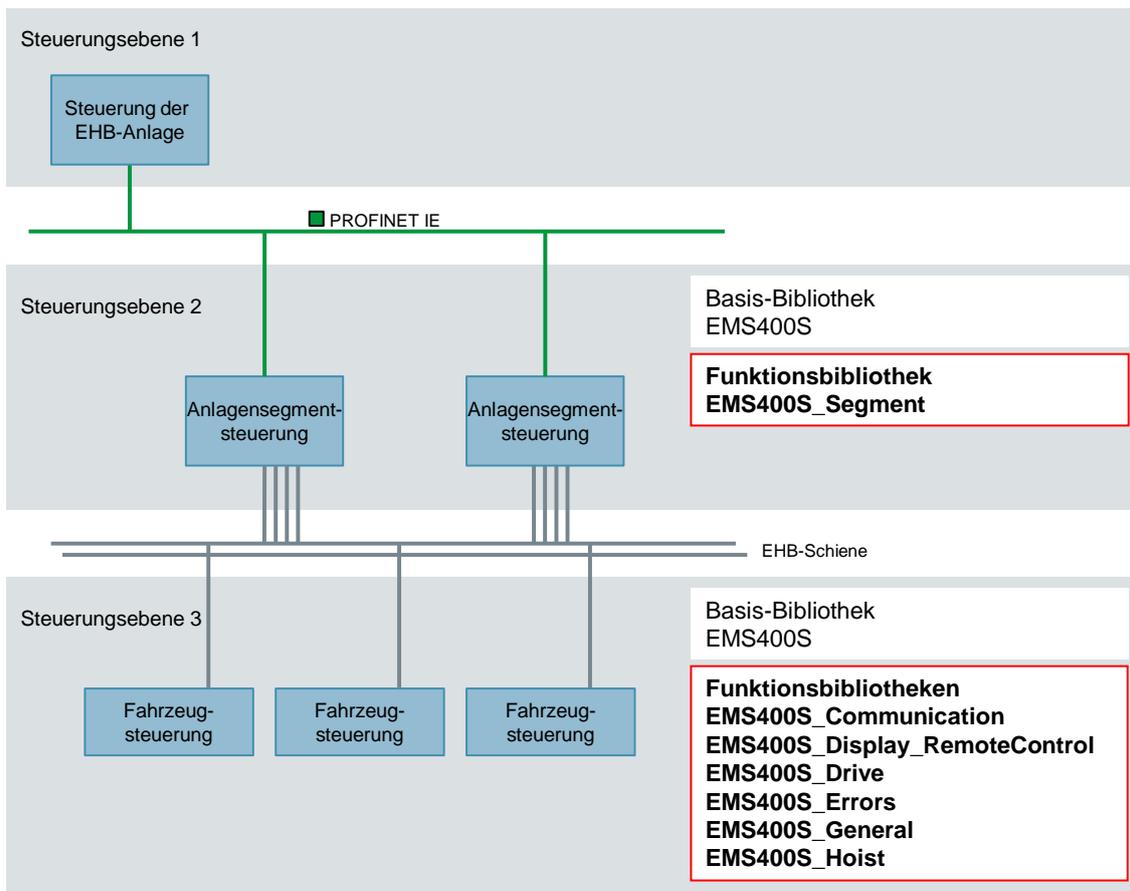
In der Steuerungsebene 2 (Anlagensegmentsteuerung) wird die Funktionsbibliothek „EMS400S_Segment“ eingesetzt.

In der Steuerungsebene 3 (EHB-Fahrzeugsteuerung) kommen die folgenden Funktionsbibliotheken zum Einsatz:

- EMS400S_Communication
- EMS400S_Display_RemoteControl
- EMS400S_Drive
- EMS400S_Errors
- EMS400S_General
- EMS400S_Hoist

Hinweis Die Bausteine aus der Basis-Bausteinbibliothek „EMS400S“ sind Grundlage für die Funktionsbibliotheken „EMS400S_xxx“.

Abbildung 1-1



1.4.4 Bibliotheksfunktionen in dieser Beschreibung

Anhand einer Anlagensegmentsteuerung und einer Fahrzeugsteuerung vom Typ S7-1200 werden die folgenden grundsätzlichen Funktionen mit den Funktionsbibliotheken „EMS400S_xxx“ erläutert.

Funktionsbibliothek für die Anlagensegmentsteuerung

- Funktionsbibliothek **EMS400S_Segment**
 - Kommandos an Fahrzeugsteuerung senden
 - Status von Fahrzeugsteuerung empfangen
 - Kommandos an RAILs senden
 - Status von RAILs empfangen
 - Daten an Fahrzeugsteuerung senden, quittierte Datenkommunikation

Funktionsbibliotheken für die Fahrzeugsteuerung

- Funktionsbibliothek **EMS400S_Communication**
 - Kommandos von Anlagensegmentsteuerung empfangen
 - Status an Anlagensegmentsteuerung senden
 - Daten von Anlagensegmentsteuerung empfangen, quittierte Datenkommunikation
 - Kommunikation zum PSB-C-Modul
- Funktionsbibliothek **EMS400S_Display_RemoteControl**
 - Kommunikation zum PSB-C-Modul
 - Signale der Infrarot-Fernbedienung empfangen und bereitstellen für das Anwenderprogramm
 - Signale der Stationären Fernbedienung empfangen und bereitstellen für das Anwenderprogramm
 - EMS-ID oder Fehlercode für das Display auswählen
- Funktionsbibliothek **EMS400S_Drive**
 - Barcode auswerten
 - Fahrtrieb ansteuern
 - Geschwindigkeit regeln
 - Positionieren
 - Isolationssegmente auswerten
- Funktionsbibliothek **EMS400S_Errors**
 - Störungsmeldungen für das PSB-C-Modul bilden
 - Diagnose für PN-Teilnehmer aufrufen
 - Störungsmeldungen für den Fahrtrieb bilden
 - Störungsmeldungen für den Hubtrieb bilden
 - Allgemeine Meldungen bilden
 - Sammelmeldung bilden

1.4 Bibliotheken

- Funktionsbibliothek **EMS400S_General**
 - Betriebsarten Hand / Automatik bilden
 - Quittiersignal bilden
 - Daten für die Antriebe bereitstellen
- Funktionsbibliothek **EMS400S_Hoist**
 - Trommelgeber
 - Hubantrieb ansteuern
 - Positionieren

Hinweis Jede Funktionsbibliothek enthält mehrere Bausteine. Die Auflistungen der Bausteine finden Sie in den Kapiteln 2 bis 8.

Hinweis Es werden in diesem Dokument nur die Schritte erläutert, die zur Realisierung der oben aufgeführten Punkte nötig sind. Für eine komplette Anlagenapplikation müssen noch zusätzlich anwenderdefinierte Bausteine erstellt werden. Dies ist kein Bestandteil dieser Bibliotheksbeschreibung.

1.5 Hard- und Softwarevoraussetzungen

Voraussetzungen für die Bibliotheken

Um die Funktionalität der hier beschriebenen Bibliotheken nutzen zu können, sind folgend genannte Hard- und Softwarevoraussetzungen einzuhalten.

Hardware zur Anlagensegmentsteuerung

Tabelle 1-1

Nr.	Komponente	Artikelnummer	Anzahl	Hinweis
1.	S7-1200 CPU1214C DC/DC/DC	6ES7214-1AG31-0XB0	1	Oder eine vergleichbare S7-1200 CPU ab FW 3.0
2.	S7-1200 PSB-S	6ES7228-1RC52-0AA0	1	Ab FW 2.0

Hardware zur Fahrzeugsteuerung

Tabelle 1-2

Nr.	Komponente	Artikelnummer	Anzahl	Hinweis
1.	S7-1200 CPU1212C DC/DC/DC	6ES7212-1AE31-0XB0	1	Oder eine vergleichbare S7-1200 CPU ab FW 3.0
2.	S7-1200 PSB-C	6ES7228-1RC51-0AA0	1	Ab FW 2.0

Hinweis Alternativ können für die in Tabelle 1-2 aufgeführten Komponenten auch ähnliche Komponenten verwendet werden.

Standard Software

Tabelle 1-3

Nr.	Komponente	Artikelnummer
1.	TIA STEP 7 Basic V14	6ES7822-0AA04-0YA5

Hinweis Alternativ können auch höhere Ausgabestände verwendet werden.

1 Übersicht der Funktionsbibliotheken

1.5 Hard- und Softwarevoraussetzungen

HSP und Bibliotheken

Die folgende Liste enthält alle Dateien und Projekte, die in diesem Beispiel verwendet werden.

Tabelle 1-4

Nr.	Komponente	Hinweis
1.	EMS400S_V14.zip	Diese gepackte Datei enthält die EMS400S Basis-Bibliothek, Kommunikationsbausteine für Anlagensegment- und Fahrzeugsteuerung SIMATIC EMS400S, siehe \4\.
2.	EMS400S_Communication_V14	Diese Datei enthält die EMS400S Bibliothek Anwender-Kommunikationsbausteine für Fahrzeugsteuerung SIMATIC EMS400S.
3.	EMS400S_Display_RemoteControl_V14	Diese Datei enthält die EMS400S Bibliothek Display- und Remote Control-Bausteine für Fahrzeugsteuerung SIMATIC EMS400S.
4.	EMS400S_Drive_V14	Diese Datei enthält die EMS400S Bibliothek Fahrtriebs- und Barcode-Bausteine für Fahrzeugsteuerung SIMATIC EMS400S.
5.	EMS400S_Errors_V14	Diese Datei enthält die EMS400S Bibliothek Störungsmeldung-Bausteine für Fahrzeugsteuerung SIMATIC EMS400S.
6.	EMS400S_General_V14	Diese Datei enthält die EMS400S Bibliothek Allgemeine Bausteine für Fahrzeugsteuerung SIMATIC EMS400S.
7.	EMS400S_Hoist_V14	Diese Datei enthält die EMS400S Bibliothek Hubantriebs- Bausteine für Fahrzeugsteuerung SIMATIC EMS400S.
8.	EMS400S_Segment_V14	Diese Datei enthält die EMS400S Bibliothek Anwender-Kommunikationsbausteine für Anlagensegmentsteuerung SIMATIC EMS400S.

2 Funktionsbibliothek „EMS400S_Segment“

Diese Funktionsbibliothek enthält Bausteine für die Anlagensegmentsteuerung. Diese Bausteine bearbeiten die Kommunikation mit der Steuerung, der EHB-Anlage und mit den Segmenten / Fahrzeugsteuerungen.

Sie benötigen die Bausteine, um

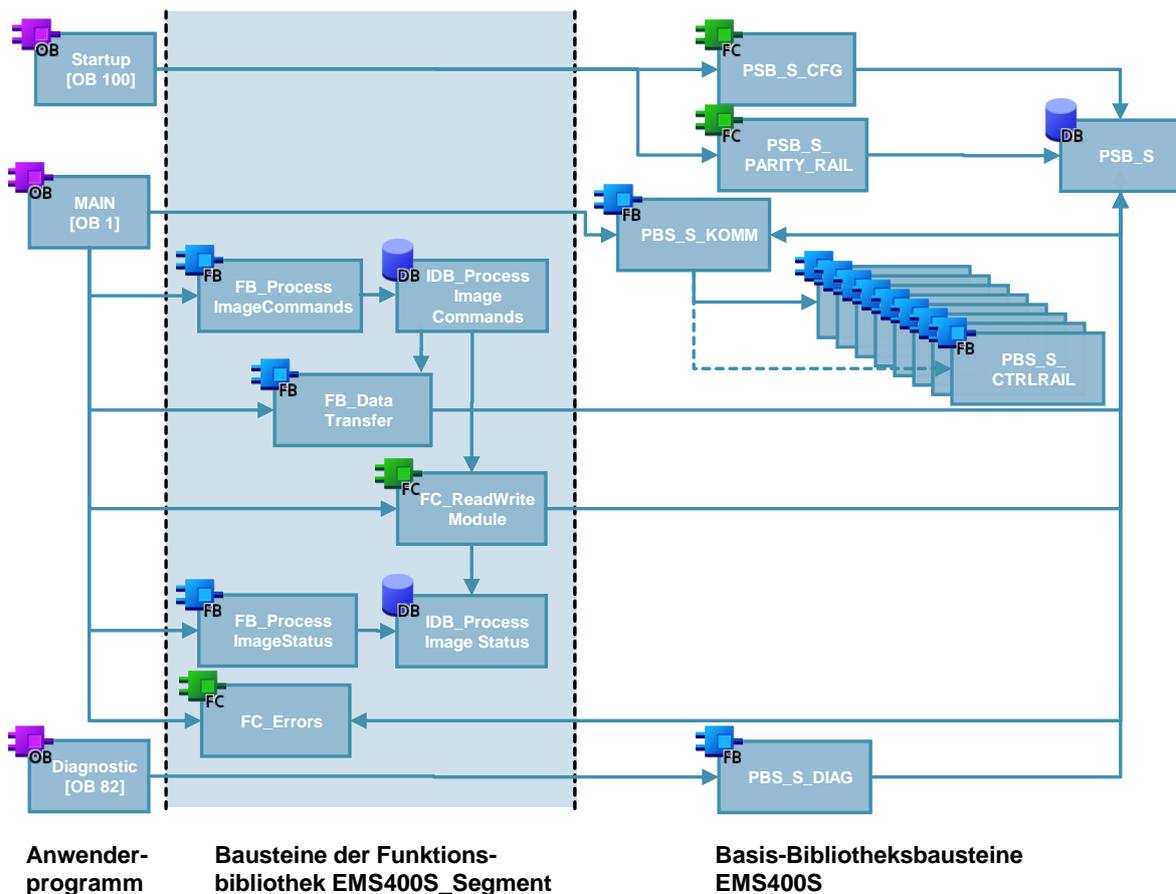
- Kommandos und Daten von der Steuerung der EHB-Anlage zu empfangen.
- Statusmeldungen an die Steuerung der EHB-Anlage zu senden.
- Kommandos und Daten über die Segmente an die Fahrzeugsteuerungen zu senden.
- Statusmeldungen über die Segmente von den Fahrzeugsteuerungen zu empfangen.

2.1 Bibliotheksinhalt

Schematische Darstellung

In der folgenden Abbildung sind die Bausteine markiert, die in der Bibliothek „EMS400S_Segment“ enthalten sind.

Abbildung 2-1



Auflistung der Bausteine

Die folgende Tabelle listet alle zur Bibliothek „EMS400S_Segment“ gehörenden Bausteine auf.

Tabelle 2-1

Baustein	Symbol	Kommentar
FC30	FC_ReadWriteModule	Kommunikation mit 1 PBS-S-Modul
FC50	FC_Errors	Sammelfehler der PBS-S-Module
FB30	FB_DataTransfer	Quitierte Datenkommunikation, Senden von Daten für 3 Datenbausteine
FB31	FB_DataTransfer_SR	Quitierte Datenkommunikation, Empfang von Daten und Senden von Daten
FB38	FB_ProcessImage Commands	Prozessabbild der Kommandos an die Fahrzeugsteuerung
FB39	FB_ProcessImageStatus	Prozessabbild der Statusinformationen der Fahrzeugsteuerung
	UDT_PSB_S_Command	Kommandos an die Segmente / RAILS
	UDT_PSB_S_Status	Statusinformationen der Segmente / RAILS
	UDT_PSB_SEG_OUT	Kommandos an die Fahrzeugsteuerungen in den Segmenten
	UDT_PSB_SEG_IN	Statusinformationen der Fahrzeugsteuerungen in den Segmenten
Variablen- tabelle	Segment tags	Alle Merker, die in der Anlagensegmentsteuerung SIMATIC EMS400S verwendet werden

2.2 Erläuterung der Bausteine für die Anlagensegmentsteuerung

Das folgende Kapitel erläutert alle zur Bibliothek „EMS400S_Segment“ gehörenden Bausteine einer Anlagensegmentsteuerung.

2.2.1 FC_ReadWriteModule (FC30)

Funktion

Der Baustein „FC_ReadWriteModule“ stellt die Kommunikation mit einem PSB-S-Modul der Anlagensegmentsteuerung her.

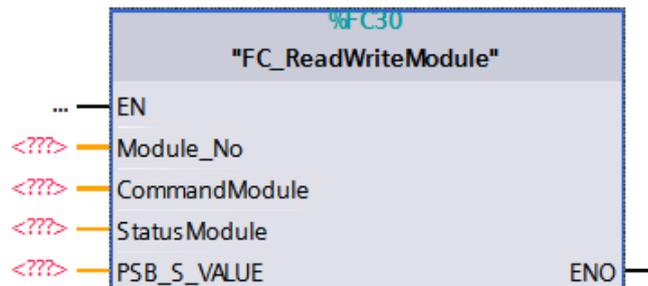
Rufen Sie den Baustein für jedes PSB-S-Modul im zyklischen Anwenderprogramm (OB1) auf.

Im Detail umfasst der Baustein die Funktionen:

- Kommunikation mit RAILS / Fahrzeugsteuerungen über DB „DB_PSB_S“
- Kommandos an die drei Segmente-/ Fahrzeugsteuerungen schreiben
- Kommandos für die drei Segmente-/ RAILS eines PSB-S-Moduls schreiben
- Statusmeldungen der drei Segmente-/ Fahrzeugsteuerungen lesen
- Statusmeldungen der drei Segmente/ RAILS eines PSB-S-Moduls lesen

Aufruf und Parameter

Die folgende Abbildung zeigt den Aufruf des Bausteins „FC_ReadWriteModule“. Abbildung 2-2



Eingangsparameter

Tabelle 2-2

Parameter	Datentyp	Beschreibung
Module_No	INT	Nummer des PSB-S Moduls, 1...8

Ein-/Ausgangsparameter

Tabelle 2-3

Parameter	Datentyp	Beschreibung
CommandModule	Struct	Übergabestruktur Kommandos, Struktur im InstanzDB des „FB_ProcessImageCommands“
Rail1	Word	Kommandos an Segment 1, siehe „UDT_PSB_SEG_OUT“ in Kapitel 2.2.7 Datentypen (UDTs)
Rail2	Word	Kommandos an Segment 2, siehe „UDT_PSB_SEG_OUT“ in Kapitel 2.2.7 Datentypen (UDTs)
Rail3	Word	Kommandos an Segment 3, siehe „UDT_PSB_SEG_OUT“ in Kapitel 2.2.7 Datentypen (UDTs)
Module	Word	Kommandos für 1 Modul, siehe „UDT_PSB_S_Command“ in Kapitel 2.2.7 Datentypen (UDTs)
StatusModule	Struct	Übergabestruktur Status, Struktur im InstanzDB des „FB_ProcessImageStatus“
Rail1	Word	Status von Segment 1, siehe „UDT_PSB_S_SEG_IN“ in Kapitel 2.2.7 Datentypen (UDTs)
Rail2	Word	Status von Segment 2, siehe „UDT_PSB_S_SEG_IN“ in Kapitel 2.2.7 Datentypen (UDTs)
Rail3	Word	Status von Segment 3, siehe „UDT_PSB_S_SEG_IN“ in Kapitel 2.2.7 Datentypen (UDTs)
Module	Word	Status von 1 Modul, siehe „UDT_PSB_S_Status“ in Kapitel 2.2.7 Datentypen (UDTs)
PSB_S_VALUE	UDT_PSB_S_Value	Übergabestruktur in DB „DB_PSB_S“ aus der Basis-Bibliothek, siehe 14

Hinweis Die Parameter EN und ENO sind nur in FUP/KOP Darstellung sichtbar. Sie müssen diese Parameter nicht beschalten.

Dies gilt für alle Bausteine in dieser Beschreibung.

2.2.2 FC_Errors (FC50)

Funktion

Der Baustein „FC_Errors“ bildet für jedes PSB-S-Modul einen Sammelfehler.

Rufen Sie den Baustein im zyklischen Anwenderprogramm (OB1) auf.

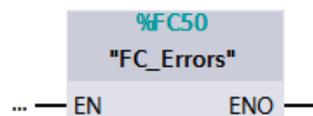
Im Detail realisiert der Baustein die Funktionen:

- Abfrage der Fehlercodes in DB „DB_PSB_S“ (aus der Basis-Bibliothek, siehe [4](#)) für jedes PSB-S- Modul.
- Bildung von Sammelfehler

Aufruf und Parameter

Die folgende Abbildung zeigt den Aufruf des Bausteins „FC_Errors“.

Abbildung 2-3



Der Baustein hat keine Parameter.

2.2.3 FB_DataTransfer (FB30)

Funktion

Der Baustein „FB_DataTransfer“ führt die quittierte Datenkommunikation aus.

Rufen Sie den Baustein „FB_DataTransfer“ im zyklischen Anwenderprogramm (OB1) auf.

Der Baustein realisiert die folgenden Funktionen:

- Daten an die Fahrzeugsteuerung senden.
- Kommunikation überwachen

Die Daten werden von „FB_ProcessImageCommands“ in den Bereich „Data_Segm_PLC[1..70]“ des InstanzDBs transferiert.

Der Baustein „FB_DataTransfer“ empfängt keine Daten von der Fahrzeugsteuerung.

Der Umfang der quittierten Datenkommunikation ist variabel, um verschiedene Arten von Daten zu übertragen. Dies wird durch „DB_Type_DT“ bestimmt.

Der Baustein „FB_DataTransfer“ sendet Daten für

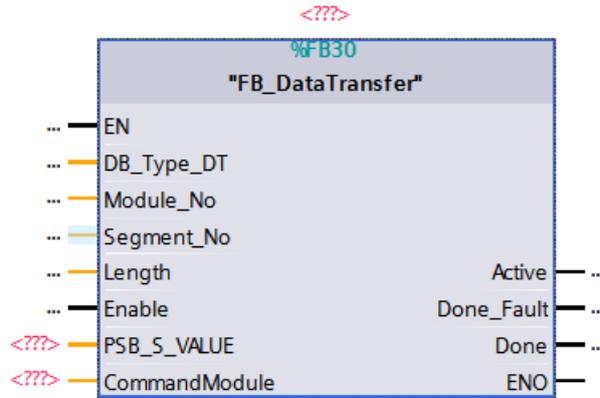
- DB_GlobalDataEMS; siehe Funktionsbibliothek „EMS400S_Global“
- DB_StopPositions; siehe Funktionsbibliothek „EMS400S_Drive“
- DB_HoistPositions; siehe Funktionsbibliothek „EMS400S_Hoist“

Diese globalen Daten sind in allen Fahrzeugsteuerungen einer Anlage identisch. Sie werden über die quittierte Datenkommunikation zentral aktualisiert.

Aufruf und Parameter

Die folgende Abbildung zeigt den Aufruf des Bausteins „FB_DataTransfer“.

Abbildung 2-4



Eingangsparameter

Tabelle 2-4

Parameter	Datentyp	Beschreibung
DB_Type_DT	Int	Auswahl des DBs für Datentransfer 1 = DB_GlobalDataEMS, 2 = DB_StopPositions, 3 = DB_HoistPositions
Module_No	Int	Modulnummer für Datentransfer (1..8)
Segment_No	Int	Segmentnummer für Datentransfer (1..3)
Length	Int	Datenlänge (DBB) für Datentransfer
Enable	Bool	Freigabe Datentransfer

Ausgangsparameter

Tabelle 2-5

Parameter	Datentyp	Beschreibung
Active	Bool	Datentransfer ist aktiv
Done_Fault	Bool	Datentransfer fertig mit Fehler
Done	Bool	Datentransfer fertig ohne Fehler

Ein-/Ausgangsparameter

Tabelle 2-6

Parameter	Datentyp	Beschreibung
PSB_S_VALUE	UDT_PSB_S_Value	Übergabestruktur in DB „DB_PSB_S“ aus der Basis-Bibliothek, siehe 4
CommandModule	Array [1..8] of Struct	Übergabestrukturen für Kommandos Struktur im InstanzDB des „FB_ProcessImageCommands“

Dauer der Kommunikation

In jedem Kommunikationszyklus werden 16 Bit / 1 Wort übertragen.

Zur Abschätzung des Zeitbedarfs für den Datenaustausch, kann die folgende Formel für die Zeitdauer des Transfers verwendet werden:

$$T = (\text{Anzahl Worte} * 0,57s) + 3s$$

Diese Berechnung gilt für die Frequenz des Synchronisationssignals
 $t_{\text{SYNC}} = 250 \text{ ms}$.

2.2.4 FB_DataTransfer_SR (FB31)

Funktion

Der Baustein „FB_DataTransfer“ hat zwei Funktionen:

- Empfang von Daten von der Fahrzeugsteuerung mit quittierter Datenkommunikation
- Senden von Daten zur Fahrzeugsteuerung mit quittierter Datenkommunikation.

Hinweis

Den Baustein „FB_DataTransfer_SR“ können Sie **alternativ** zu „FB_DataTransfer“ in OB1 aufrufen. In der Fahrzeugsteuerung müssen Sie entsprechend „FB_DataTransferEMS_SR“ anstelle von „FB_DataTransferEMS“ aufrufen, siehe [3.2.3](#) FB_DataTransferEMS_SR (FB31).

Beim Datenempfang empfängt der Baustein Daten von der Fahrzeugsteuerung und transferiert diese nach Abschluss der quittierten Datenkommunikation in den Bereich „Data_EMS_PLC[1..20]“ des InstanzDBs.

Sie müssen anschließend die Daten in Ihre Zieldatenbausteine oder Kommunikationsbereiche (Merker) transferieren.

Um Daten an die Fahrzeugsteuerung zu senden, müssen Sie vorher die zu sendenden Daten in den Bereich „Data_Segm_PLC[1..20]“ des InstanzDBs transferieren.

Der Umfang der Datenkommunikation ist variabel, um verschiedene Arten von Daten zu übertragen. Dies wird durch den Modus bestimmt.

In der folgenden Tabelle sind die programmierten Varianten der Datenkommunikation aufgelistet.

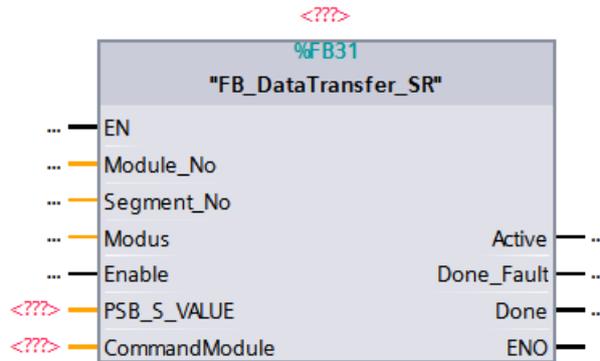
Tabelle 2-7

Modus	Anzahl Worte von Anlagensegmentsteuerung an Fahrzeugsteuerung	Anzahl Worte von Fahrzeugsteuerung an Anlagensegmentsteuerung	Anzahl Worte gesamt
1	10	10	20
2	2	0	2
3	0	2	2
4	4	0	4
5	0	4	4
6	20	0	20
7	0	20	20
8	20	20	40

Aufruf und Parameter

Die folgende Abbildung zeigt den Aufruf des Bausteins „FB_DataTransfer_SR“.

Abbildung 2-5



Eingangsparameter

Tabelle 2-8

Parameter	Datentyp	Beschreibung
Module_No	Int	Modulnummer für Datentransfer (1..8)
Segment_No	Int	Segmentnummer für Datentransfer (1..3)
Modus	Int	Datentransfermodus, Datenumfang (1..8)
Enable	Bool	Freigabe Datentransfer

Ausgangsparameter

Tabelle 2-9

Parameter	Datentyp	Beschreibung
Active	Bool	Datentransfer ist aktiv
Done_Fault	Bool	Datentransfer fertig mit Fehler
Done	Bool	Datentransfer fertig ohne Fehler

Ein-/Ausgangsparameter

Tabelle 2-10

Parameter	Datentyp	Beschreibung
PSB_S_VALUE	"UDT_PSB_S_Value"	Übergabestruktur in DB „DB_PSB_S“ aus der Basis-Bibliothek, siehe 4
CommandModule	Array [1..8] of Struct	Übergabestrukturen für Kommandos Struktur im InstanzDB des „FB_ProcessImageCommands“

2.2.5 FB_ProcessImageCommands (FB38)

Funktion

Der Baustein „FB_ProcessImageCommands“ kopiert Merkerbereiche, die von der Steuerung der EHB-Anlage gesendet werden, in ein Prozessabbild der Kommandos und Daten.

Rufen Sie den Baustein im zyklischen Anwenderprogramm (OB1) auf.

Im Detail realisiert der Baustein die Funktionen:

- Prozessabbild für Kommandos und Daten erstellen
- Kommunikation überwachen

ACHTUNG Die Steuerung der EHB-Anlage sendet Daten in Merkerbereiche. Diese Daten werden asynchron zum Programmzyklus empfangen. Dadurch kann es vorkommen, dass während der Bearbeitung Daten geändert werden.

Wenn die Bausteine, z.B. „FC_ReadWriteModule“, direkt auf die Merkerbereiche zugreifen, können sie mit inkonsistenten Daten arbeiten. Diese inkonsistenten Daten werden an die Segmente / Fahrzeugsteuerungen gesendet.

Um mit konsistenten Daten zu arbeiten, rufen Sie „FB_ProcessImageCommands“ einmal am Anfang des OB1 auf. Die Bausteine, z.B. „FC_ReadWriteModule“, greifen auf Strukturen im InstanzDB zu, und damit auf konsistente Daten.

Die Kommunikation wird überwacht. Die Steuerung der EHB-Anlage sendet ein taktendes Lebensbit / Heartbeat. Wenn „Heartbeat“ länger als 5s den Zustand nicht wechselt, wird das Prozessabbild der Kommandos mit NULL überschrieben.

Das Prozessabbild wird in Strukturen im InstanzDB gespeichert.

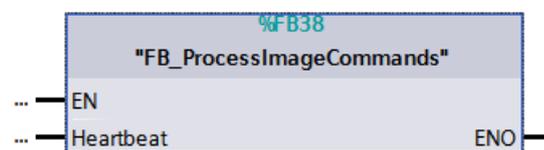
Die folgenden Bausteine werden mit diesen Strukturen parametrisiert und lesen Daten:

- FB_DataTransfer,
- FB_DataTransfer_SR
- FC_ReadWriteModule

Aufruf und Parameter

Die folgende Abbildung zeigt den Aufruf des Bausteins „FB_ProcessImageCommands“.

Abbildung 2-6



Eingangsparameter

Tabelle 2-11

Parameter	Datentyp	Beschreibung
Heartbeat	Bool	Lebensbit, Kommunikationsüberwachung

2.2.6 FB_ProcessImageStatus (FB39)

Funktion

Der Baustein „FB_ProcessImageStatus“ kopiert das Prozessabbild der Statusdaten in Merkerbereiche, die von der Steuerung der EHB-Anlage gelesen werden.

Rufen Sie den Baustein im zyklischen Anwenderprogramm (OB1) auf.

ACHTUNG	<p>Die Steuerung der EHB-Anlage liest Statusinformationen aus Merkerbereichen. Diese Daten werden asynchron zum Programmzyklus gelesen. Dadurch kann es vorkommen, dass während der Baustein-Bearbeitung zeitgleich Daten gelesen werden.</p> <p>In diesem Fall könnte die Steuerung der EHB-Anlage mit inkonsistenten Daten arbeiten.</p> <p>Um inkonsistente Daten zu verhindern, rufen Sie „FB_ProcessImageStatus“ einmal am Ende des OB1 auf. Die Bausteine, z.B. „FC_ReadWriteModule“, greifen auf Strukturen im InstanzDB zu.</p>
----------------	--

Das Prozessabbild ist in Strukturen im InstanzDB gespeichert.

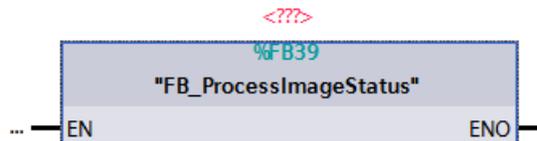
Die folgenden Bausteine werden mit diesen Strukturen parametrisiert und schreiben Statusdaten:

- FB_DataTransfer_SR
- FC_ReadWriteModule

Aufruf und Parameter

Die folgende Abbildung zeigt den Aufruf des Bausteins „FB_ProcessImageStatus“.

Abbildung 2-7



Der Baustein hat keine Parameter.

2.2.7 Datentypen (UDTs) für Kommunikationsschnittstellen

Die Kommunikationsschnittstellen zwischen der Steuerung der EHB-Anlage, der Anlagensegmentsteuerung und dem Segment / der Fahrzeugsteuerung bestehen aus jeweils 16 Bits = 1 Wort.

Die Belegungen sind in „UDT_PSB_SEG_OUT“ und in „UDT_PSB_SEG_IN“ definiert.

Die Kommunikationsschnittstellen zwischen der Steuerung der EHB-Anlage und dem PSB-S-Modul der Anlagensegmentsteuerung bestehen aus jeweils 16 Bits = 1 Wort.

Die Belegungen sind in „UDT_PSB_S_Command“ und in „UDT_PSB_S_STATUS“ definiert.

UDT_PSB_SEG_OUT

Die Kommandos an das Segment / die Fahrzeugsteuerung sind im „UDT_PSB_SEG_OUT“ definiert.

Die folgende Tabelle listet die Kommandos auf.

Tabelle 2-12

Name	Datentyp	Beschreibung
Enable_Drive	Bool	Antrieb x-Richtung ein
Enable_Drive_Non_Equ	Bool	Antrieb x-Richtung ein, invertiert
Bridge_Stoppos	Bool	Antrieb x-Richtung weiterfahren
Enable_Backward	Bool	Antrieb x-Richtung rückwärts
Enable_Hoist	Bool	Hub ein z-Richtung
Enable_Hoist_Up	Bool	Hub in obere Position z-Richtung
Reset	Bool	Fehler quittieren
Res07	Bool	Reserve
Speed_Slow	Bool	Antrieb x-Richtung langsam
Speed_Var	Bool	variable Antriebsgeschwindigkeit x-Richtung
Speed_1	Bool	variable Geschwindigkeit Bit 1 ($2^0/63$ von variabler Antriebsgeschwindigkeit)
Speed_2	Bool	variable Geschwindigkeit Bit 2 ($2^1/63$ von variabler Antriebsgeschwindigkeit)
Speed_3	Bool	variable Geschwindigkeit Bit 3 ($2^2/63$ von variabler Antriebsgeschwindigkeit)
Speed_4	Bool	variable Geschwindigkeit Bit 4 ($2^3/63$ von variabler Antriebsgeschwindigkeit)
Speed_5	Bool	variable Geschwindigkeit Bit 5 ($2^4/63$ von variabler Antriebsgeschwindigkeit)
Speed_6	Bool	variable Geschwindigkeit Bit 6 ($2^5/63$ von variabler Antriebsgeschwindigkeit)

Hinweis

Falls sich mehrere Fahrzeuge in einem Segment befinden, werden die Kommandos an alle Fahrzeuge im Segment gesendet.

UDT_PSB_SEG_IN

Die Statusinformationen des Segments / der Fahrzeugsteuerung sind im „UDT_PSB_SEG_IN“ definiert.

Die folgende Tabelle listet die Statusinformationen auf.

Tabelle 2-13

Name	Datentyp	Beschreibung
Carr_Number	Byte	EMS-ID
BB_Rail	Bool	Rails gebrückt
Drive_Inpos	Bool	Antrieb x-Richtung in Position
Hoist_Inpos	Bool	Hub z-Richtung in Position
Hoist_Inpos_up	Bool	Hub z-Richtung in Position, oben
Drive_forwards	Bool	Antrieb x-Richtung vorwärts
Stop_by_Initiator	Bool	Stopp durch Auffahrinitiator
Manual_Mode	Bool	Handbetrieb (IR)
Fault	Bool	Sammelstörung

Hinweis

Falls sich mehrere Fahrzeuge in einem Segment befinden, werden die Statusmeldungen der Fahrzeuge zusammengefasst (ODER-Funktion).

UDT_PSB_S_Command

Die Kommandos für das PSB-S-Modul sind im „UDT_PSB_S_Command“ definiert.

Die folgende Tabelle listet die Kommandos auf.

Tabelle 2-14

Name	Datentyp	Beschreibung
Segment1_valid	Bool	1=Kommunikation gültig
Segment2_valid	Bool	1=Kommunikation gültig
Segment3_valid	Bool	1=Kommunikation gültig
Segment1_idle_active	Bool	1=Freigabe Schiene (Reserve)
Segment2_idle_active	Bool	1=Freigabe Schiene (Reserve)
Segment3_idle_active	Bool	1=Freigabe Schiene (Reserve)
Segment1_Command_Data	Bool	1=quittierte Datenübertragung
Segment2_Command_Data	Bool	1=quittierte Datenübertragung
Segment3_Command_Data	Bool	1=quittierte Datenübertragung
Reserve9	Bool	Reserve
Reserve10	Bool	Reserve
Reserve11	Bool	Reserve
Reserve12	Bool	Reserve
Reserve13	Bool	Reserve
Reserve14	Bool	Reserve
Reserve15	Bool	Reserve

UDT_PSB_S_Status

Die Statusinformationen des PSB-S-Moduls sind im „UDT_PSB_S_Status“ definiert.

Die folgende Tabelle listet die Statusinformationen auf.

Tabelle 2-15

Name	Datentyp	Beschreibung
Segment1_valid	Bool	1=Kommunikation gültig
Segment2_valid	Bool	1=Kommunikation gültig
Segment3_valid	Bool	1=Kommunikation gültig
Segment1_Carr_presence	Bool	1=mindestens 1 Fahrzeug präsent
Segment2_Carr_presence	Bool	1=mindestens 1 Fahrzeug präsent
Segment3_Carr_presence	Bool	1=mindestens 1 Fahrzeug präsent
Segment1_Command_Data	Bool	1=quitierte Datenübertragung
Segment2_Command_Data	Bool	1=quitierte Datenübertragung
Segment3_Command_Data	Bool	1=quitierte Datenübertragung
Digital_Input1	Bool	Digitaler Eingang
Digital_Input2	Bool	Digitaler Eingang
Digital_Input3	Bool	Digitaler Eingang
Digital_Input4	Bool	Digitaler Eingang
Digital_Input5	Bool	Digitaler Eingang
Digital_Input6	Bool	Digitaler Eingang
Reserve15	Bool	Reserve

2.2.8 Variablen-tabelle (Segment tags)

Die Variablen-tabelle „Segment tags“ enthält die Deklaration der Merker, die für den Betrieb der Anlagensegmentsteuerung SIMATIC EMS400S notwendig sind.

Die Steuerung der EHB-Anlage sendet und liest Daten in Merkerbereichen.

Jeder Merkerbereich wird mit einer S7-Verbindung übertragen.

Die folgende Tabelle listet die Merkerbereiche auf.

Tabelle 2-16

Merkerbereich ab	Länge in Bytes	Verwendung
MW64	84	Kommandos von der Steuerung der EHB-Anlage
MW400	84	Statusinformationen an die Steuerung der EHB-Anlage
MW200	160	Daten für quitierte Datenübertragung, von der Steuerung der EHB-Anlage

Hinweis

Wenn Sie Daten von den Fahrzeugsteuerungen über die quitierte Datenübertragung empfangen und an die Steuerung der EHB-Anlage senden wollen, müssen Sie einen weiteren Merkerbereich definieren.

3 Funktionsbibliothek „EMS400S_Communication“

Diese Funktionsbibliothek enthält Bausteine für die Fahrzeugsteuerung. Diese Bausteine bearbeiten die Kommunikation mit der Anlagensegmentsteuerung.

Sie benötigen die Bausteine, um

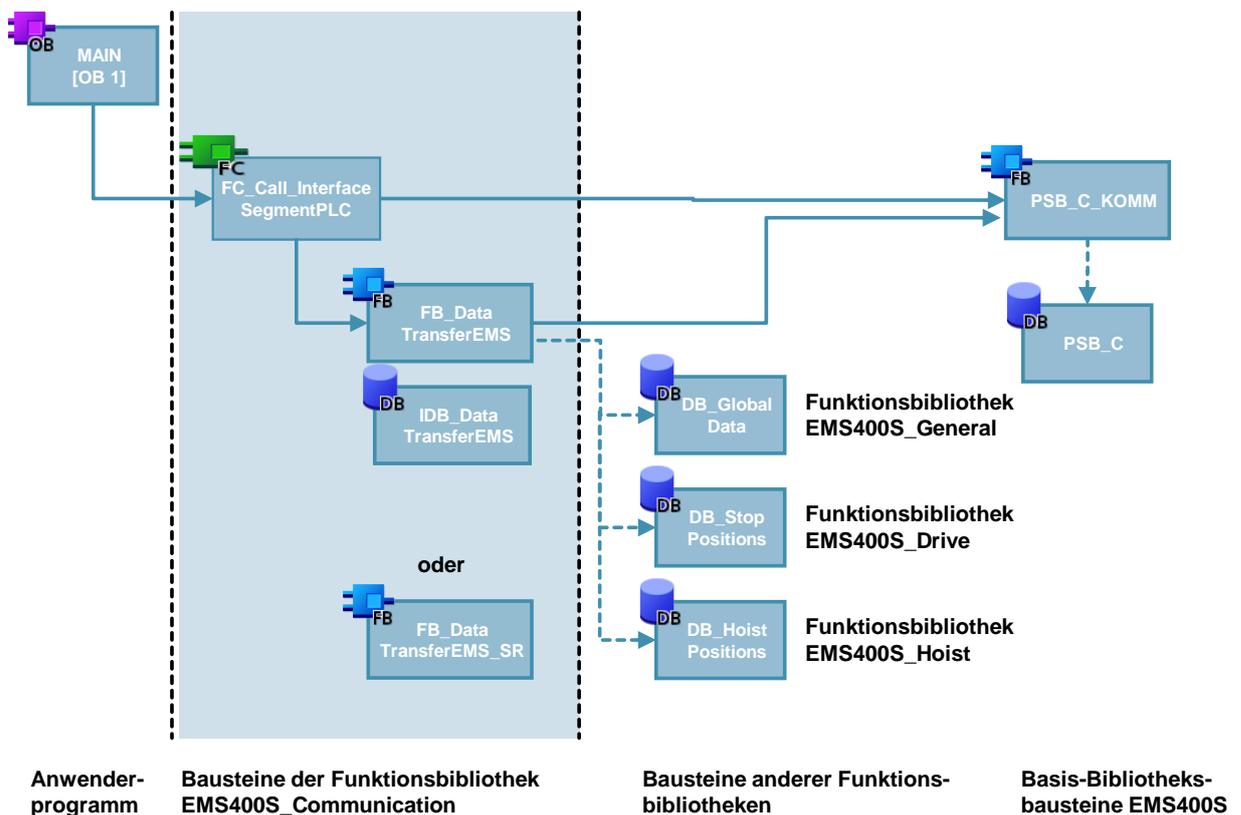
- Kommandos und Daten von der Anlagensegmentsteuerung zu empfangen.
- Statusmeldungen an die Anlagensegmentsteuerung zu senden.

3.1 Bibliotheksinhalt

Schematische Darstellung

In der folgenden Abbildung sind die Bausteine markiert, die in der Funktionsbibliothek „EMS400S_Communication“ enthalten sind.

Abbildung 3-1



Auflistung der Bausteine

Die folgende Tabelle listet alle zur Funktionsbibliothek „EMS400S_Communication“ gehörenden Bausteine auf.

Tabelle 3-1

Baustein	Symbol	Kommentar
FC30	FC_Call_Interface SegmentPLC	Aufruf der Bausteine, Generierung der Statusinformationen
FB30	FB_DataTransferEMS	Quitierte Datenkommunikation, Empfang von Daten für 3 Datenbausteine
FB31	FB_DataTransferEMS_ SR	Quitierte Datenkommunikation, Alternative zu „FB_DataTransferEMS“. Empfang und Senden von Daten
DB30	IDB_DataTransferEMS	InstanzDB für „FB_DataTransferEMS“
	UDT_PSB_SEG_IN	Statusinformationen der Fahrzeugsteuerung
	UDT_PSB_SEG_OUT	Kommandos an die Fahrzeugsteuerung

3.2 Erläuterung der Kommunikations-Bausteine

Das folgende Kapitel erläutert alle zur Funktionsbibliothek „EMS400S_Communication“ gehörenden Bausteine.

3.2.1 FC_Call_InterfaceSegmentPLC (FC30)

Funktion

Der Baustein „FC_Call_InterfaceSegmentPLC“ stellt die Kommunikation mit der Anlagensegmentsteuerung her.

Rufen Sie den Baustein im zyklischen Anwenderprogramm (OB1) auf.

Im Detail realisiert der Baustein die Funktionen:

- Aufruf des FC „PSB_C_KOMM“ zur Kommunikation mit RAIL / Anlagensegmentsteuerung über PSB-C-Modul, siehe [4](#)
- Kommandowort auf Merker transferieren, für das Anwenderprogramm
- Statusinformationen generieren
- Aufruf des „FB_DataTransferEMS“

Im Baustein werden Statusinformationen der Fahrzeugsteuerung gebildet. Die Belegung finden Sie im „UDT_PSB_SEG_IN“, in Kapitel [3.2.4](#) Datentypen (UDTs).

Hinweis

Die Statusinformationen „Hoist_Inpos“ und „Hoist_Inpos_up“ werden nur benötigt, wenn ein Hubantrieb / Kettenzug vorhanden ist.

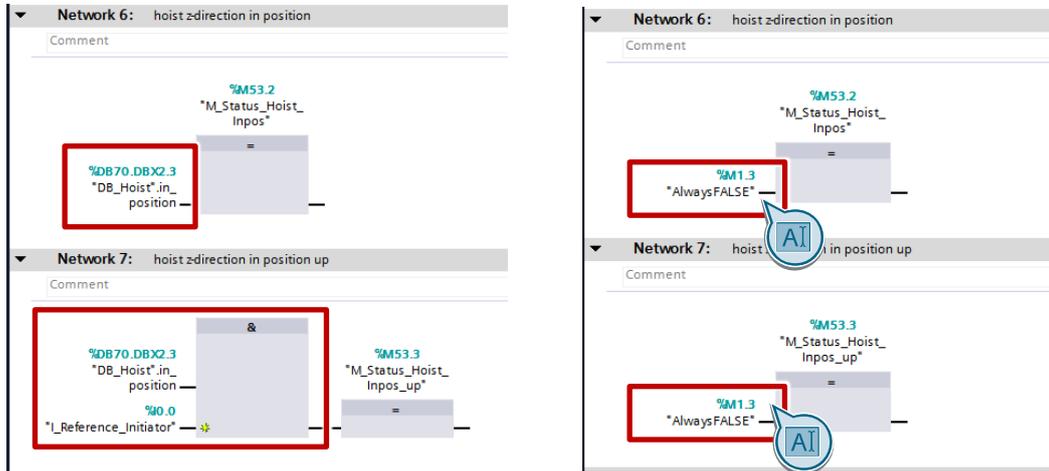
Besitzt Ihre Fahrzeugsteuerung keinen Hubantrieb, können Sie im Baustein „FC_Call_InterfaceSegmentPLC“ diesen Statusinformationen „AlwaysFALSE“ zuweisen.

In der folgenden Abbildung sehen Sie diese Statusinformationen des Hubantriebs.

Abbildung 3-2

Statusinformationen mit Hubantrieb,
Lieferzustand

Statusinformationen ohne Hubantrieb



Aufruf und Parameter

Die folgende Abbildung zeigt den Aufruf des Bausteins „FC_Call_InterfaceSegmentPLC“.

Abbildung 3-3



Der Baustein hat keine Parameter.

3.2.2 FB_DataTransferEMS (FB30)

Funktion

Der Baustein „FB_DataTransferEMS“ empfängt Daten von der Anlagensegmentsteuerung und transferiert die empfangenen Daten nach Abschluss der quittierten Datenkommunikation in festgelegte Datenbausteine.

Er sendet keine Daten an die Anlagensegmentsteuerung.

Im Detail umfasst der Baustein die Funktionen:

- Daten empfangen
- Daten speichern

Der Umfang der Datenkommunikation ist variabel, um verschiedene Arten von Daten zu übertragen. Dies wird durch den Modus bestimmt, der als 1. Wort übertragen wird.

Der Baustein „FB_DataTransferEMS“ empfängt Daten für

- DB_GlobalDataEMS in Funktionsbibliothek „EMS400S_Global“
- DB_StopPositions in Funktionsbibliothek „EMS400S_Drive“
- DB_HoistPositions in Funktionsbibliothek „EMS400S_Hoist“

Diese globalen Daten sind in allen Fahrzeugsteuerungen einer Anlage gleich. Sie werden über die quittierte Datenkommunikation zentral aktualisiert.

Der Baustein „FB_DataTransferEMS“ wird von „FC_Call_InterfaceSegmentPLC“ aufgerufen.

Hinweis

Die Datenkommunikation zu „DB_HoistPositions“ wird nur benötigt, wenn ein Hubantrieb / Kettenzug vorhanden ist. Besitzt Ihre Fahrzeugsteuerung keinen Hubantrieb, können Sie wahlweise

- „DB_HoistPositions“ aus der Funktionsbibliothek „EMS400S_Hoist“ in Ihre Fahrzeugsteuerung kopieren,
- die entsprechenden Anweisungen in „FB_DataTransferEMS“ auskommentieren oder löschen.

3 Funktionsbibliothek „EMS400S_Communication“

3.2 Erläuterung der Kommunikations-Bausteine

In der folgenden Abbildung sehen Sie diese Anweisungen.

Abbildung 3-4

Anweisungen mit Hubantrieb, Lieferzustand

```

EMS400S > Carrier_PLC [CPU 1212C DC/DC/DC] > Program blocks > 30_Communication > FB_DataTransferEMS [FB30]
Block interface
590 // Modus 3: 55 Worte an die EMS-Steuerung 55 Words to EMS PLC / DB_HoistPositions
591 3:
592     FOR #iTemp_index := 0 TO 10 DO
593         #iTemp_index_Rec := #iTemp_index*5;
594         #dTemp_DWord.W0 :=#Receive[#iTemp_index_Rec +2];
595         #dTemp_DWord.W1 :=#Receive[#iTemp_index_Rec +1];
596         "DB_HoistPositions".HoistDefinition[#iTemp_index].BarcodeStart:= DWORD_TO_DINT(#dTemp_DWord);
597         #dTemp_DWord.W0 :=#Receive[#iTemp_index_Rec +4];
598         #dTemp_DWord.W1 :=#Receive[#iTemp_index_Rec +3];
599         "DB_HoistPositions".HoistDefinition[#iTemp_index].BarcodeEnd:= DWORD_TO_DINT(#dTemp_DWord);
600         "DB_HoistPositions".HoistDefinition[#iTemp_index].Height:= WORD_TO_INT(#Receive[#iTemp_index_Rec +5]);
601     END_FOR;
602 END_CASE;
603 END_IF;
604
    
```

Anweisungen ohne Hubantrieb

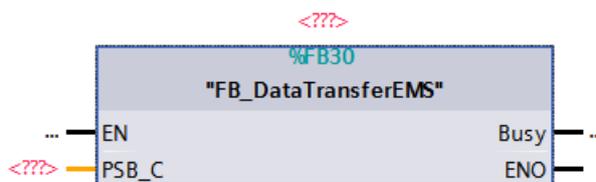
```

EMS400S > Carrier_PLC [CPU 1212C DC/DC/DC] > Program blocks > 30_Communication > FB_DataTransferEMS [FB30]
Block interface
589 END_FOR;
590 // Modus 3: 55 Worte an die EMS-Steuerung 55 Words to EMS PLC / DB_HoistPositions
591 3:
592 ;
593 END_CASE;
594 END_IF;
595
    
```

Aufruf und Parameter

Die folgende Abbildung zeigt den Aufruf des Bausteins „FB_DataTransferEMS“.

Abbildung 3-5



Ausgangsparameter

Tabelle 3-2

Parameter	Datentyp	Beschreibung
Busy	Bool	Datenkommunikation ist aktiv

Ein-/Ausgangsparameter

Tabelle 3-3

Parameter	Datentyp	Beschreibung
PSB_C	UDT_PSB_C	Übergabestruktur aus der Basis-Bibliothek, siehe 4\

Dauer der Kommunikation

In jedem Kommunikationszyklus werden 16 Bit = 1 Wort übertragen.

Zur Abschätzung des Zeitbedarfs für den Datenaustausch, kann die folgende Formel für die Zeitdauer des Transfers verwendet werden:

$$T = (\text{Anzahl Worte} * 0,57\text{s}) + 3\text{s}$$

Diese Berechnung gilt für die Frequenz des Synchronisationssignals
 $t_{\text{SYNC}} = 250\text{ms}$.

ACHTUNG

Falls das Fahrzeug während der quittierten Datenkommunikation das Segment / RAIL verlässt, besteht die Gefahr, dass die aktuellen Daten nicht übertragen und gespeichert sind.

Dadurch kann das Fahrzeug z.B. nicht an neuen StoppPositionen anhalten, die neuen Hubfreigaben nicht bearbeiten oder neue Geschwindigkeitsgrenzwerte nicht verarbeiten.

Die Gefahr kann vermieden werden, indem der Fahrantrieb des Fahrzeugs während der Datenkommunikation angehalten wird. Dies ist in „FC_Drive“ programmiert (in der Funktionsbibliothek „EMS400_Drive“), siehe Kapitel [5.2.1 FC_Drive \(FC60\)](#).

3.2.3 FB_DataTransferEMS_SR (FB31)

Funktion

Der Baustein „FB_DataTransferEMS_SR“ hat zwei Funktionen:

- Empfang von Daten von der Anlagensegmentsteuerung
- Senden von Daten zur Anlagensegmentsteuerung.

Hinweis

Den Baustein „FB_DataTransferEMS_SR“ können Sie **alternativ** zu „FB_DataTransferEMS“ in „FC_Call_InterfaceSegmentPLC“ aufrufen. In der Anlagensegmentsteuerung müssen Sie entsprechend „FB_DataTransfer_SR“ anstelle von „FB_DataTransfer“ aufrufen, siehe [2.2.4 FB_DataTransfer_SR \(FB31\)](#).

Beim Datenempfang empfängt der Baustein Daten von der Anlagensegmentsteuerung und transferiert diese nach Abschluss der quittierten Datenkommunikation in einen Bereich des InstanzDBs.

Sie müssen danach die Daten in Ihre Zieldatenbausteine transferieren.

Um Daten an die Anlagensegmentsteuerung zu senden, müssen Sie vorher die zu sendenden Daten in den Bereich „Data_EMS_PLC[1..20]“ des InstanzDBs transferieren.

Der Umfang der Datenkommunikation ist variabel, um verschiedene Arten von Daten zu übertragen. Dies wird durch den Modus bestimmt, der als 1. Wort übertragen wird.

In der folgenden Tabelle sind die programmierten Varianten der Datenkommunikation aufgelistet.

Tabelle 3-4

Modus	Anzahl Worte von Anlagensegmentsteuerung an Fahrzeugsteuerung	Anzahl Worte von Fahrzeugsteuerung an Anlagensegmentsteuerung	Anzahl Worte gesamt
1	10	10	20
2	2	0	2
3	0	2	2
4	4	0	4
5	0	4	4
6	20	0	20
7	0	20	20
8	20	20	40

Empfangene Daten

Die empfangenen Daten werden nach Abschluss des Transfers im InstanzDB gespeichert, damit sie konsistent sind.

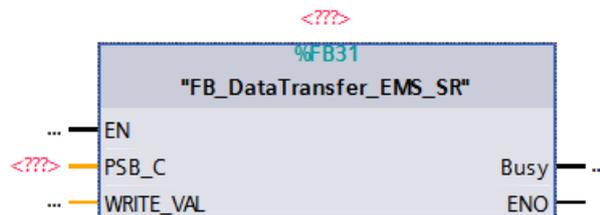
- Bei Modus 1, 6 und 8 werden die empfangenen Daten im Bereich „Data_Segm_PLC[1..20]“ als WORD gespeichert.
- Bei Modus 2 werden die empfangenen Daten gespeichert als
 - 2 * WORD Bereich Data_2Words_Rec
 - 1 * DINT Bereich Data_1Dint_Rec
 - 1 * DWORD Bereich Data_1DWord_Rec
- Bei Modus 4 werden die empfangenen Daten gespeichert als
 - 4 * WORD Bereich Data_4Words_Rec
 - 2 * DINT Bereich Data_2Dint_Rec
 - 2 * DWORD Bereich Data_2DWord_Rec

Sie müssen anschließend die Daten aus diesen Bereichen des InstanzDBs in Ihre Zieldatenbausteine transferieren.

Aufruf und Parameter

Die folgende Abbildung zeigt den Aufruf des Bausteins „FB_DataTransferEMS_SR“.

Abbildung 3-6



Ausgangsparameter

Tabelle 3-5

Parameter	Datentyp	Beschreibung
Busy	Bool	Datenkommunikation ist aktiv

Ein-/Ausgangsparameter

Tabelle 3-6

Parameter	Datentyp	Beschreibung
PSB_C	"UDT_PSB_C"	Übergabestruktur aus der Basis-Bibliothek, siehe 4
WRITE_VAL	Word	Sendedaten, Parameter für PSB_C_KOMM aus der Basis-Bibliothek, siehe 4

3.2.4 Datentypen (UDTs) für Kommunikationsschnittstellen

Die Kommunikationsschnittstellen zwischen der Anlagensegmentsteuerung und der Fahrzeugsteuerung bestehen aus jeweils 16 Bits = 1 Wort.

UDT_PSB_SEG_IN

Die Statusinformationen der Fahrzeugsteuerung sind im „UDT_PSB_SEG_IN“ definiert.

Die folgende Tabelle listet die Statusinformationen auf.

Tabelle 3-7

Name	Datentyp	Beschreibung
Carr_Number	Byte	EMS-Nummer
BB_Rail	Bool	Rails gebrückt
Drive_Inpos	Bool	Antrieb x-Richtung in Position
Hoist_Inpos	Bool	Hub z-Richtung in Position
Hoist_Inpos_up	Bool	Hub z-Richtung in Position, oben
Drive_forwards	Bool	Antrieb x-Richtung vorwärts
Stop_by_Initiator	Bool	Stopp durch Auffahrinitiator
Manual_Mode	Bool	Handbetrieb (IR)
Fault	Bool	Sammelstörung

UDT_PSB_SEG_OUT

Die Kommandos von der Anlagensegmentsteuerung sind im „UDT_PSB_SEG_OUT“ definiert.

Die folgende Tabelle listet die Kommandos auf.

Tabelle 3-8

Name	Datentyp	Beschreibung
Enable_Drive	Bool	Antrieb x-Richtung ein
Enable_Drive_Non_Equ	Bool	Antrieb x-Richtung ein, invertiert
Bridge_Stoppos	Bool	Antrieb x-Richtung weiterfahren
Enable_Backward	Bool	Antrieb x-Richtung rückwärts
Enable_Hoist	Bool	Hub ein z-Richtung
Enable_Hoist_Up	Bool	Hub in obere Position z-Richtung
Reset	Bool	Fehler quittieren
Res07	Bool	Reserve
Slow	Bool	Antrieb x-Richtung langsam
Speed_Var	Bool	variable Antriebsgeschwindigkeit x-Richtung
Speed_1	Bool	variable Geschwindigkeit Bit 1 ($2^{0/63}$ von variabler Antriebsgeschwindigkeit)
Speed_2	Bool	variable Geschwindigkeit Bit 2 ($2^{1/63}$ von variabler Antriebsgeschwindigkeit)
Speed_3	Bool	variable Geschwindigkeit Bit 3 ($2^{2/63}$ von variabler Antriebsgeschwindigkeit)
Speed_4	Bool	variable Geschwindigkeit Bit 4 ($2^{3/63}$ von variabler Antriebsgeschwindigkeit)
Speed_5	Bool	variable Geschwindigkeit Bit 5 ($2^{4/63}$ von variabler Antriebsgeschwindigkeit)
Speed_6	Bool	variable Geschwindigkeit Bit 6 ($2^{5/63}$ von variabler Antriebsgeschwindigkeit)

4 Funktionsbibliothek „EMS400S_Display_RemoteControl“

Diese Funktionsbibliothek enthält Bausteine für die Fahrzeugsteuerung. Diese Bausteine bearbeiten die Kommunikation mit der Infrarot-Fernbedienung und dem Display.

Sie benötigen die Bausteine, um

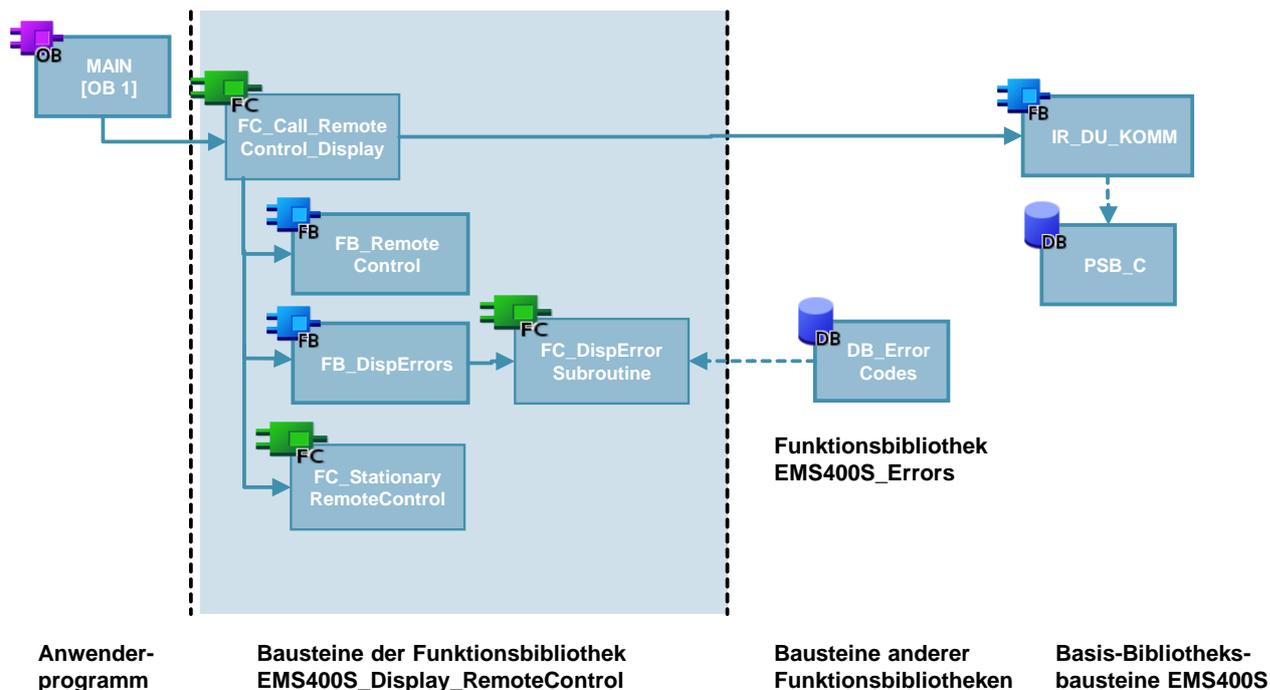
- Signale von der Infrarot-Fernbedienung zu empfangen.
- EMS-Nr. und Fehlernummern am Display anzuzeigen.

4.1 Bibliotheksinhalt

Schematische Darstellung

In der folgenden Abbildung sind die Bausteine markiert, die in der Funktionsbibliothek „EMS400S_Display_RemoteControl“ enthalten sind.

Abbildung 4-1



Auflistung der Bausteine

Die folgende Tabelle listet alle zur Funktionsbibliothek „EMS400S_Display_RemoteControl“ gehörenden Bausteine auf.

Tabelle 4-1

Baustein	Symbol	Kommentar
FC20	FC_Call_RemoteControl_Display	Aufruf der Bausteine
FC22	FC_DispErrorSubroutine	Unterprogramm für „FB_DispErrors“
FC23	FC_StationaryRemoteControl	Stationäre Fernbedienung auswerten
FB21	FB_RemoteControl	Fernbedienung auswerten, EMS-ID zuweisen
FB22	FB_DispErrors	Fehlercode für 32 Störungsmeldungen ermitteln und anzeigen
DB21	IDB_RemoteControl	InstanzDB für „FB_RemoteControl“
DB22	IDB_DispErrors	InstanzDB für „FB_DispErrors“

4.2 Erläuterung der Bausteine

Das folgende Kapitel erläutert alle zur Funktionsbibliothek „EMS400S_Display_RemoteControl“ gehörenden Bausteine.

4.2.1 FC_Call_RemoteControl_Display (FC20)**Funktion**

Der Baustein „FC_Call_RemoteControl_Display“ ruft „FC_IR_DU_KOMM“ aus der Basis-Bibliothek auf, siehe [4.1](#), und die Bausteine für die Fernbedienungen und das Display, siehe Tabelle 4-1.

Rufen Sie den Baustein im zyklischen Anwenderprogramm (OB1) auf.

Im Detail realisiert der Baustein die Funktionen:

- Kommunikation zum PSB-C-Modul
- Signale der Infrarot-Fernbedienung empfangen und bereitstellen für das Anwenderprogramm
- Signale der Stationären Fernbedienung empfangen und bereitstellen für das Anwenderprogramm
- EMS-ID oder Fehlercode für das Display auswählen

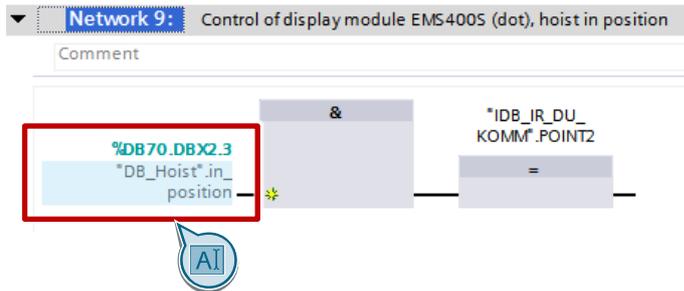
Hinweis

Wenn Ihre Fahrzeugsteuerung keinen Hubantrieb hat, können Sie in Netzwerk 9 dem Datenbit „POINT2“ das Signal „AlwaysFALSE“ zuweisen.

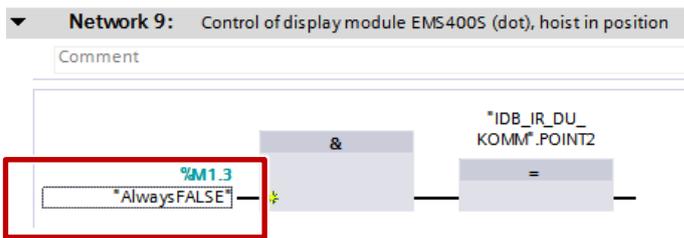
In der folgenden Abbildung sehen Sie dieses Netzwerk.

Abbildung 4-2

Ansteuerung mit Hubantrieb, Lieferzustand



Ansteuerung ohne Hubantrieb



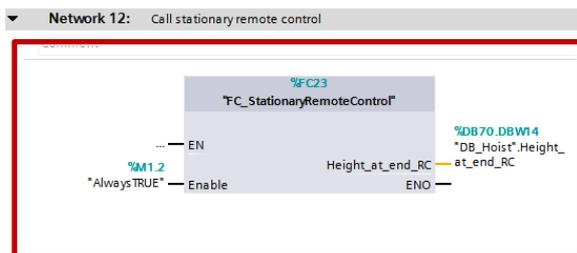
Hinweis

Wenn Ihre Fahrzeugsteuerung keine stationäre Fernbedienung hat, können Sie den Aufruf des „FC_StationalaryRemoteControl“ in Netzwerk 12 löschen, und den Merkern für die stationäre Fernbedienung das Signal „AlwaysFALSE“ zuweisen.

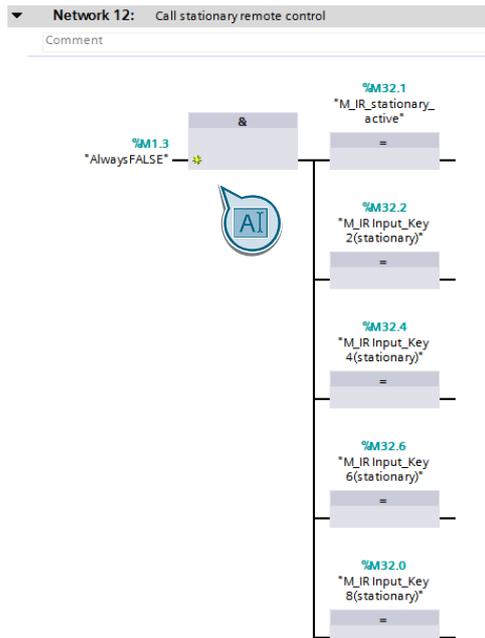
In der folgenden Abbildung sehen Sie dieses Netzwerk.

Abbildung 4-3

**Aufruf mit stationärer Fernbedienung ,
Lieferzustand**



Ohne stationäre Fernbedienung

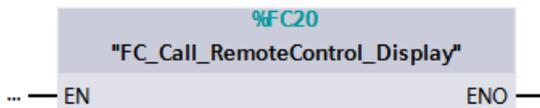


© Siemens AG Copyright-2014 All rights reserved

Aufruf und Parameter

Die folgende Abbildung zeigt den Aufruf des Bausteins „FC_Call_RemoteControl_Display“.

Abbildung 4-4



Der Baustein hat keine Parameter

4.2.2 FC_DispErrorSubroutine (FC22)

Funktion

Der Baustein „FC_DispErrorSubroutine“ kopiert die Störungsmeldungssignale aus „DB_ErrorCodes“.

Der Baustein wird in „FB_DispErrors“ aufgerufen.

Im Detail umfasst der Baustein die Funktionen:

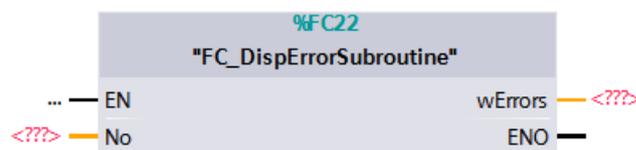
- Störungsmeldungswort auswählen
- Bytes tauschen

DB_ErrorCodes ist in der Funktionsbibliothek „EMS400S_Errors“ enthalten, die Beschreibung finden Sie in Kapitel [6.2.4 DB_ErrorCodes \(DB50\)](#).

Aufruf und Parameter

Die folgende Abbildung zeigt den Aufruf des Bausteins „FC_DispErrorSubroutine“.

Abbildung 4-5



Eingangsparameter

Tabelle 4-2

Parameter	Datentyp	Beschreibung
No	Int	Nummer des Störungsmeldungsworts

Ausgangsparameter

Tabelle 4-3

Parameter	Datentyp	Beschreibung
wErrors	Word	16 Störungsmeldungen

4.2.3 FC_StationaryRemoteControl (FC23)

Funktion

Der Baustein „FC_StationaryRemoteControl“ ermittelt, ob das Fahrzeug im definierten Bereich einer stationären Fernbedienung ist.

Der Baustein wird in „FC_Call_RemoteControl_Display“ aufgerufen.

Im Detail umfasst der Baustein die Funktionen

- Feststellen, ob eine stationäre Fernbedienung aktiv ist
- Signale für Hubantrieb bilden
- Signale für Fahrantrieb bilden

Die stationäre Fernbedienung ist am Streckenverlauf fest montiert. Der Anfang und das Ende der Arbeitsbereiche für zwei stationäre Fernbedienungen werden mit Barcode-Werten im „DB_GlobalDataEMS“ definiert, siehe [DB_GlobalDataEMS, Struktur Remote_Control \(DB10\)](#).

Innerhalb dieses Bereiches werden 4 Tasten ausgewertet, mit denen Sie folgendes bewirken:

- Heben
- Senken
- schneller Fahren
- langsamer Fahren

Die Signale werden in „FC_Drive“ und „FC_Hoist“ verarbeitet.

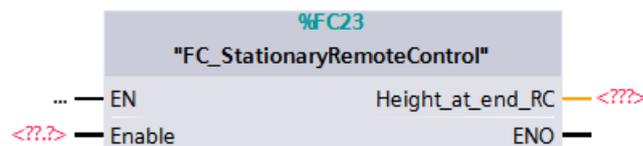
„FC_Drive“ ist in der Funktionsbibliothek „EMS400S_Drive“ enthalten, siehe Kapitel [5.2.1 FC_Drive \(FC60\)](#).

„FC_Hoist“ ist in der Funktionsbibliothek „EMS400S_Hoist“ enthalten, siehe Kapitel [8.2.1 FC_Hoist \(FC70\)](#).

Aufruf und Parameter

Die folgende Abbildung zeigt den Aufruf des Bausteins „FC_StationaryRemoteControl“.

Abbildung 4-6



4.2 Erläuterung der Bausteine

Eingangsparameter

Tabelle 4-4

Parameter	Datentyp	Beschreibung
Enable	Bool	Freigabe

Ausgangsparameter

Tabelle 4-5

Parameter	Datentyp	Beschreibung
Height_at_end_RC	Int	Soll-Höhe am Ende des Bereichs der stationären Fernbedienung

4.2.4 DB_GlobalDataEMS, Struktur Remote_Control (DB10)**Funktion**

Der Baustein „DB_GlobalDataEMS“ enthält die globalen Daten, die für den Betrieb des Fahrtriebs, des Hubantriebs und der stationären Fernbedienungen notwendig sind.

Er ist in der Funktionsbibliothek „EMS400S_General“ enthalten, siehe Kapitel [7.2.2 DB_GlobalDataEMS \(DB10\)](#).

In der folgenden Tabelle sind die Daten der Struktur „Remote_Control“ aufgelistet.

Inhalt der Struktur „Remote_Control“

Tabelle 4-6

Name	Datentyp	Beschreibung
Start_area1	DInt	Start der Strecke für festinstallierte Fernbedienung 1
End_area1	DInt	Ende der Strecke für festinstallierte Fernbedienung 1
Height_at_end_area1	Int	Höhe am Ende der Strecke für festinstallierte Fernbedienung 1
Start_area2	DInt	Start der Strecke für festinstallierte Fernbedienung 2
End_area2	DInt	Ende der Strecke für festinstallierte Fernbedienung 2
Height_at_end_area2	Int	Höhe am Ende der Strecke für festinstallierte Fernbedienung 2

4.2.5 FB_RemoteControl (FB21)

Funktion

Der Baustein „FB_RemoteControl“ wertet die Signale der Infrarot-Fernbedienung aus und weist dem Fahrzeug die EMS-ID zu.

Der Baustein wird in „FC_Call_RemoteControl_Display“ aufgerufen.

Im Detail sind es die Funktionen:

- Signale der Infrarot-Fernbedienung abfragen und ausgeben
- Kommunikation des Fahrzeugs mit der Infrarot-Fernbedienung freigeben
- EMS-ID zuweisen

Hinweis

Jedes Fahrzeug hat eine eigene EMS-ID.

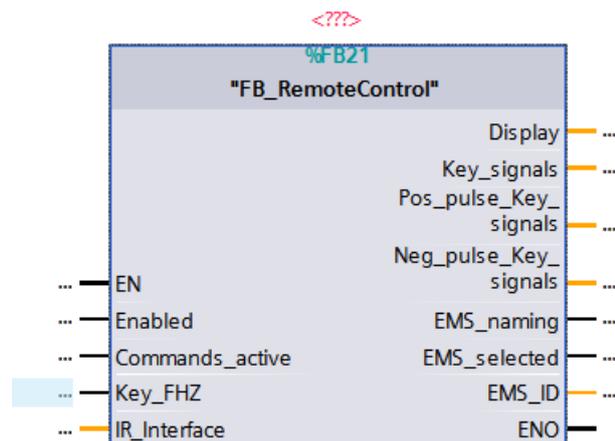
Die EMS-ID wird abwechselnd mit den Fehlercodes am Display angezeigt. Aus diesem Grund ist es ratsam, die EMS-ID außerhalb des Nummernbands der Fehlercodes zu wählen. Das Nummernband der Fehlercodes ist 600 bis 899.

Die EMS-ID wird in den Statusinformationen als Byte an die Anlagensegmentsteuerung übertragen. Wählen Sie somit die EMS-IDs aus dem Bereich 1 bis 255.

Aufruf und Parameter

Die folgende Abbildung zeigt den Aufruf des Bausteins „FB_RemoteControl“.

Abbildung 4-7



Eingangsparameter

Tabelle 4-7

Parameter	Datentyp	Beschreibung
Enabled	Bool	Freigabe
Commands_active	Bool	Steuerbefehle / keine Daten
Key_FHZ	Bool	Signal von Taste "FHZ"
IR_Interface	Word	Signale von IR-Fernbedienung, von „FC IR_DU_KOMM“ aus der Basis-Bibliothek, siehe 4

Ausgangsparameter

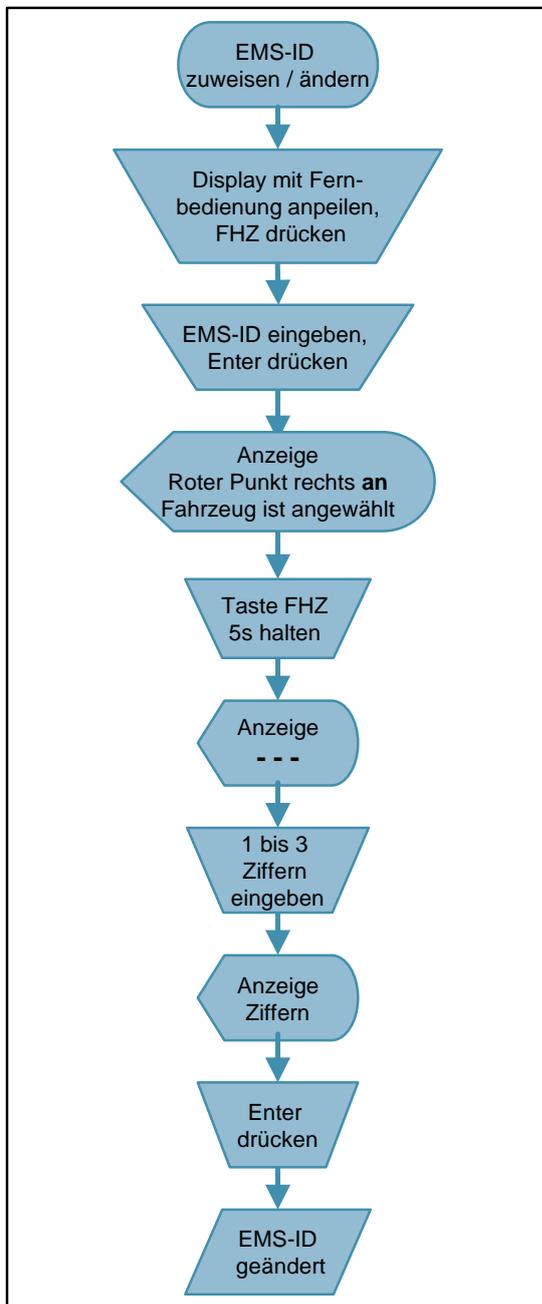
Tabelle 4-8

Parameter	Datentyp	Beschreibung
Display	Word	Ausgangswort für Anzeigeeinheit auf PSB-C Modul
Key_signals	Word	IR Tastersignale für Anwender
Pos_pulse_Key_signals	Word	IR Tastersignale für Anwender, positive Impulse
Neg_pulse_Key_signals	Word	IR Tastersignale für Anwender, negative Impulse
EMS_selection	Bool	EMS-Fahrzeugwahl aktiv
EMS_selected	Bool	EMS-Fahrzeug ist angewählt
EMS_ID	Int	EMS-ID, Fahrzeugnummer

Zuweisung der EMS-ID

Das folgende Schema stellt dar, wie Sie dem Fahrzeug die EMS-ID zuweisen.

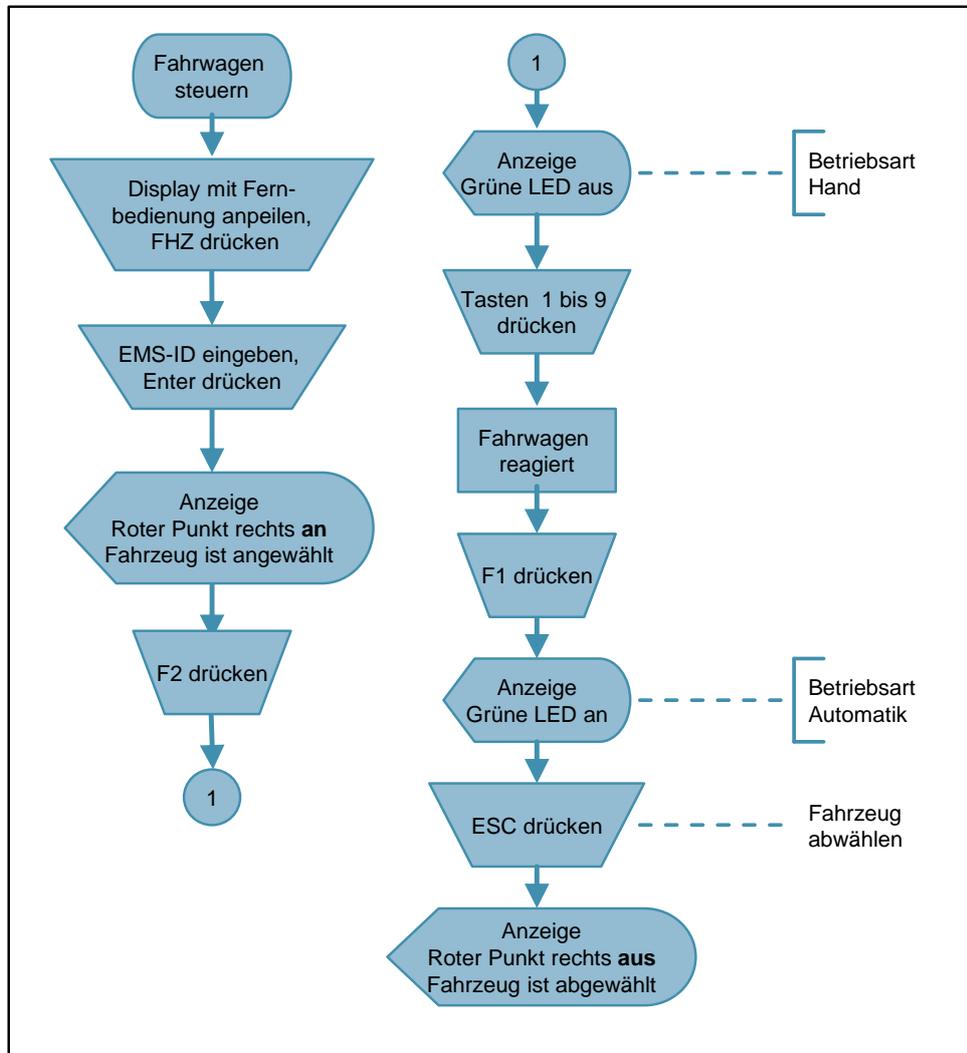
Abbildung 4-8



Steuerung mittels Infrarot

Das folgende Schema stellt dar, wie Sie das Fahrzeug mit der Infrarot-Fernbedienung steuern.

Abbildung 4-9



Tastenfunktionen

Die folgende Tabelle zeigt die Belegung der Tasten der Infrarot-Fernbedienung. Die rechten Spalte zeigt die Funktionen, wie sie in den EMS400-Funktionsbibliotheken verwendet sind.

Tabelle 4-9

Taste	Funktion	Funktion in EMS400-Funktionsbibliotheken
1	Ziffer 1	
2	Ziffer 2	Heben
3	Ziffer 3	Fehler quittieren
4	Ziffer 4	Fahren rückwärts (links)
5	Ziffer 5	schnell
6	Ziffer 6	Fahren vorwärts (rechts)
7	Ziffer 7	Trommelgeber justieren, mit Taste 2
8	Ziffer 8	Senken
9	Ziffer 9	Lampentest
Sun + 2	Anzeigehelligkeit erhöhen	
Sun + 8	Anzeigehelligkeit verringern	
	EHB-Fahrzeugwahl (FHZ)	EHB-Fahrzeugwahl
F1	Betriebsart Automatik	Betriebsart Automatik
F2	Betriebsart Hand	Betriebsart Hand
ESC	Eingabe abbrechen	
ENTER	Eingabe bestätigen	

Infrarot-Fernbedienung

Die folgende Abbildung zeigt die Infrarot-Fernbedienung.

Abbildung 4-10



4.2.6 FB_DisErrors (FB22)

Funktion

Der Baustein „FB_DisErrors“ wertet die 32 Störungsmeldungen in „DB_ErrorCodes“ aus und gibt nacheinander die Fehlercodes und die EMS-ID aus. Der Baustein „FB_DisErrors“ wird in „FC_Call_RemoteControl_Display“ aufgerufen.

Im Detail umfasst der Baustein die Funktionen:

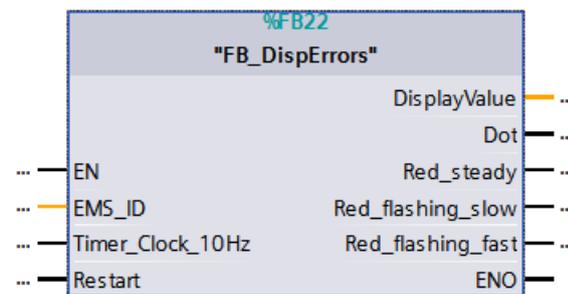
- EMS-ID anzeigen
- 32 Störungsmeldungen durchsuchen
- Fehlercodes und EMS-ID jeweils 2s lang anzeigen
- Rote LED ansteuern

„DB_ErrorCodes“ ist in der Funktionsbibliothek „EMS400S_Errors“ enthalten, die Beschreibung finden Sie in Kapitel [6.2.4 DB_ErrorCodes \(DB50\)](#).

Aufruf und Parameter

Die folgende Abbildung zeigt den Aufruf des Bausteins „FB_DisErrors“.

Abbildung 4-11



Eingangsparameter

Tabelle 4-10

Parameter	Datentyp	Beschreibung
EMS_ID	Int	EMS-ID, Fahrzeugnummer
Timer_Clock_10Hz	Bool	Taktimpuls 10 Hz für interne Zeitfunktion
Restart	Bool	Neustart der Anzeige

Ausgangsparameter

Tabelle 4-11

Parameter	Datentyp	Beschreibung
DisplayValue	Word	Ausgangswort für die Anzeigeeinheit auf PSB-C Modul
Dot	Bool	Punkt an
Red_steady	Bool	Rote LED, Dauerlicht
Red_flashing_slow	Bool	Rote LED, langsam blinken
Red_flashing_fast	Bool	Rote LED, schnell blinken

5 Funktionsbibliothek „EMS400S_Drive“

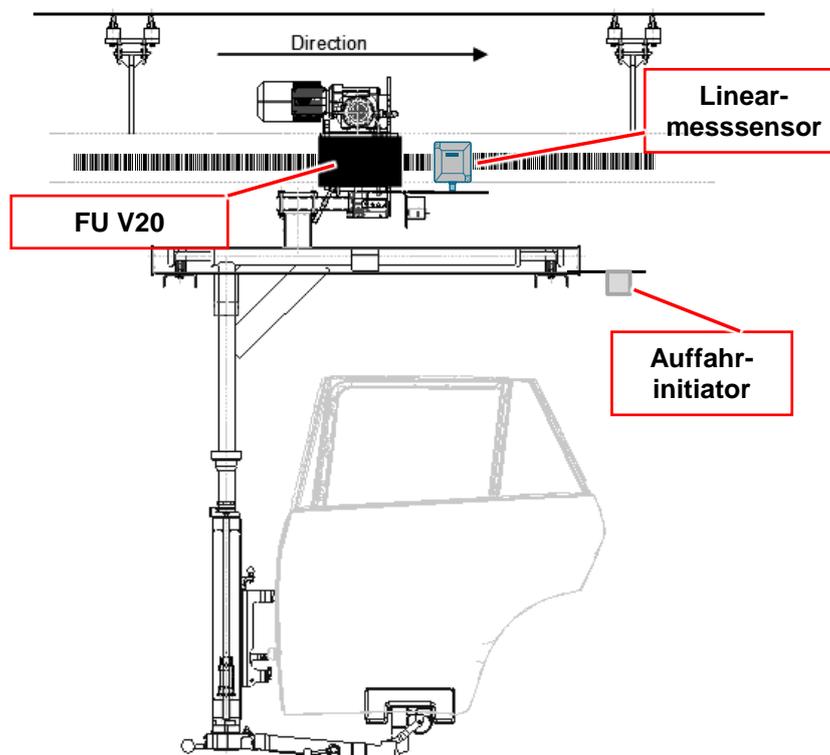
Diese Funktionsbibliothek enthält Bausteine für die Fahrzeugsteuerung. Diese Bausteine bearbeiten den Antrieb in x-Richtung, den Fahrtrieb.

Sie benötigen die Bausteine, um

- den Barcode mit dem Linearmesssensor zu erfassen und zu verarbeiten.
- den Fahrtrieb mit dem FU V20 anzusteuern.

Die folgende Abbildung zeigt schematisch die Aktoren und Sensoren für die Fahrtriebsfunktion.

Abbildung 5-1

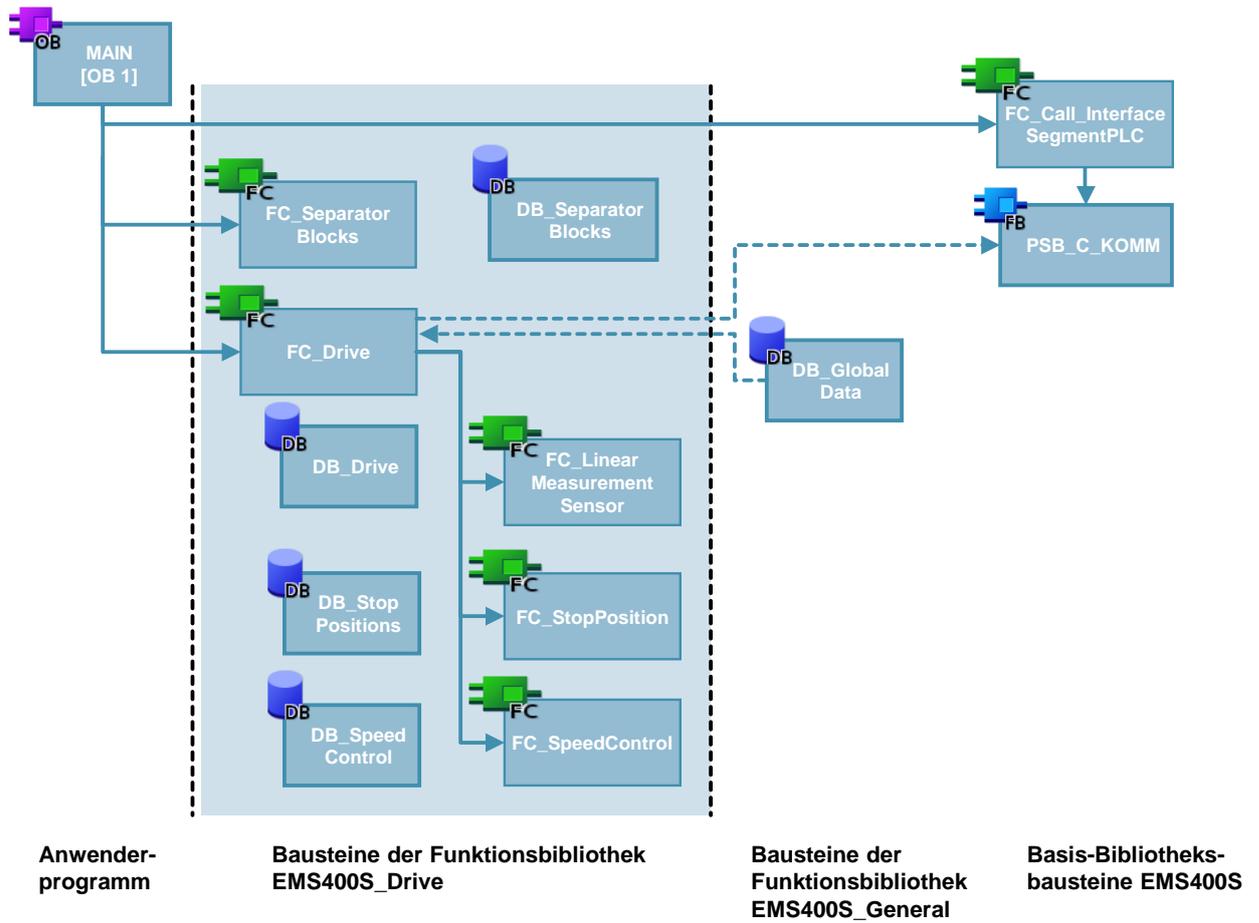


5.1 Bibliotheksinhalt

Schematische Darstellung

In der folgenden Abbildung sind die Bausteine markiert, die in der Funktionsbibliothek „EMS400S_Drive“ enthalten sind.

Abbildung 5-2



Auflistung der Bausteine

Die folgende Tabelle listet alle zur Funktionsbibliothek „EMS400S_Drive“ gehörenden Bausteine auf.

Tabelle 5-1

Baustein	Symbol	Kommentar
FC60	FC_Drive	Ansteuerung des FU V20
FC61	FC_LinearMeasurementSensor	Auswertung des Sick Linearmesssensoren OLM200
FC62	FC_SpeedControl	Geschwindigkeitsmessung und Regelung
FC63	FC_StopPosition	Ermittlung der nächsten Stopp-Position
FC67	FC_SeparatorBlocks	Ermittlung, ob Fahrzeug in Isolationssegment ist
DB60	DB_Drive	Datenbaustein für Fahrtrieb
DB62	DB_SpeedControl	Datenbaustein für Geschwindigkeitsmessung und Regelung
DB63	DB_StopPositions	Datenbaustein für die Stopp-Positionen
DB67	DB_SeparatorBlocks	Datenbaustein für die Isolationssegmente

5.2 Erläuterung der Bausteine

Das folgende Kapitel erläutert alle zur Funktionsbibliothek „EMS400S_Drive“ gehörenden Bausteine.

5.2.1 FC_Drive (FC60)**Funktion**

Der Baustein „FC_Drive“ steuert den Fahrtrieb der x-Richtung an.

Rufen Sie den Baustein im zyklischen Anwenderprogramm (OB1) auf.

Im Detail umfasst der Baustein die Funktionen:

- Freigabe des Fahrtriebs
- Auswertung der Betriebsarten Automatik / Hand
- Auswertung der Kommandos und der Geschwindigkeitsvorgabe aus der Schnittstelle zur Anlagensegmentsteuerung in Betriebsart Automatik
- Anhalten des Fahrtriebs während der Datenkommunikation
- Anhalten des Fahrtriebs, wenn der Hubtrieb am Ende des freigegebenen Hubbereichs die definierte Höhe nicht erreicht hat
- Langsam Weiterfahren, wenn das Fahrzeug in einem Isolationssegment ist und kein Kommando von der Anlagensegmentsteuerung ansteht
- Langsam Weiterfahren, wenn das Fahrzeug einen Lichttaster als Entfernungsmesser hat und dieser ein Signal gibt, weil das vorausfahrende Fahrzeug zu nah ist.
- Auswertung des Auffahrinitiators, Anhalten
- Positionieren auf die Stopp-Position in Betriebsart Automatik
- Auswertung der Signale der Fernbedienung in Betriebsart Hand
- Auswertung der Signale der stationären Fernbedienung
- Ermittlung der Sollgeschwindigkeit

5.2 Erläuterung der Bausteine

- Ansteuerung des FU V20

Der Baustein wertet folgende Parameter aus:

- die Betriebsarten,
- den Auffahrinitiator,
- den Lichttaster, falls vorhanden
- die Signale der Fernbedienung,
- den Barcodewert,
- das Fehlersignal des FU V20
- die Kommandos
- die Meldung „In_Separator_block“

Der Baustein gibt folgende Signale an den FU V20 aus:

- das Freigabesignal,
- das Richtungssignal
- das Quittiersignal
- einen analogen Geschwindigkeitswert.

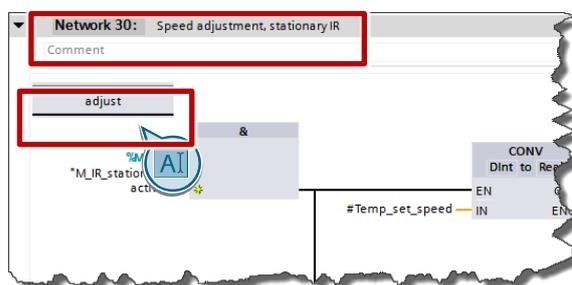
Hinweis

Die Geschwindigkeitsanpassung in Netzwerk 30 wird nur benötigt, wenn eine stationäre Fernbedienung vorhanden ist. Besitzt Ihre Fahrzeugsteuerung keine stationäre Fernbedienung, können Sie das Netzwerk 30 löschen. Die Sprungmarke „adjust“ müssen Sie in diesem Fall in das nächste Netzwerk verschieben.

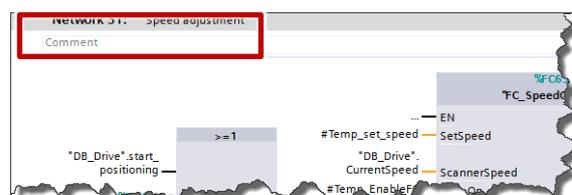
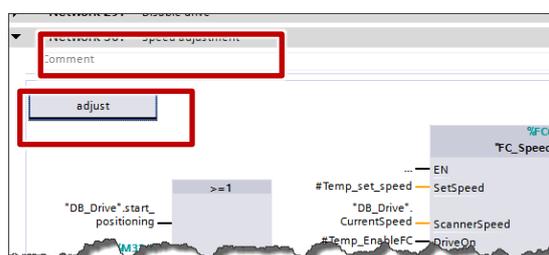
In der folgenden Abbildung sehen Sie diese Netzwerke.

Abbildung 5-3

Anweisungen mit stationärer Fernbedienung, Lieferzustand



Ohne stationäre Fernbedienung



5.2 Erläuterung der Bausteine

Aufruf und Parameter

Die folgende Abbildung zeigt den Aufruf des Bausteins „FC_Drive“.

Abbildung 5-4



Der Baustein hat keine Parameter.

Peripherie

„FC_Drive“ wertet folgende Eingänge aus:

Tabelle 5-2

Name	Datentyp	Beschreibung
I_Crash_Initiator_NC	Bool	Auffahrinitiator Öffner
I_Crash_Initiator_NO	Bool	Auffahrinitiator Schließer
I_FC_No_Error	Bool	FU V20 fehlerfrei
I_LinearMeasSensorValue	DInt	Linear-Messsensor, Positionswert
I_LinearMeasSensorSpeed	DInt	Linear-Messsensor, Geschwindigkeit
I_LinearMeasSensorStatus	Byte	Linear-Messsensor Status
I_Crash_Initiator_Slow	Bool	Lichttaster, Schleichfahrt Dieser Eingang steht nur zur Verfügung, wenn das Fahrzeug keinen Hubantrieb hat! Bei Fahrzeugen mit Hubantrieb ist der DI durch „I_HoistTemperature“ belegt.

„FC_Drive“ steuert folgende Ausgänge an:

Tabelle 5-3

Name	Datentyp	Beschreibung
Q_FC_DriveOn	Bool	Fahrtrieb ein
Q_FC_Reverse	Bool	Fahrtrieb reversieren
Q_FC_ResetError	Bool	Fehler des FU V20 quittieren
M_AnalogValue	Word	Analogwert, Geschwindigkeit für FU V20 Wert wird von PSB-C-Modul ausgegeben

Kommandos

„FC_Drive“ wertet folgende Kommandos aus der Schnittstelle zur Anlagensegmentsteuerung aus:

Tabelle 5-4

Symbol	Beschreibung
Enable_Drive	Antrieb x-Richtung ein
Enable_Drive_Non_Equ	Antrieb x-Richtung ein, invertiert
Bridge_Stoppos	Antrieb x-Richtung weiterfahren
Enable_Backward	Antrieb rückwärts
Speed_Slow	Antrieb x-Richtung langsam
Speed_Var	variable Antriebsgeschwindigkeit x-Richtung
Speed_1	variable Geschwindigkeit Bit 1 ($2^0/63$ von variabler Antriebsgeschwindigkeit)
Speed_2	variable Geschwindigkeit Bit 2 ($2^1/63$ von variabler Antriebsgeschwindigkeit)
Speed_3	variable Geschwindigkeit Bit 3 ($2^2/63$ von variabler Antriebsgeschwindigkeit)
Speed_4	variable Geschwindigkeit Bit 4 ($2^3/63$ von variabler Antriebsgeschwindigkeit)
Speed_5	variable Geschwindigkeit Bit 5 ($2^4/63$ von variabler Antriebsgeschwindigkeit)
Speed_6	variable Geschwindigkeit Bit 6 ($2^5/63$ von variabler Antriebsgeschwindigkeit)

5.2.2 DB_GlobalDataEMS, Struktur Drive (DB10)**Funktion**

Der Baustein „DB_GlobalDataEMS“ enthält die globalen Daten, die für den Betrieb des Fahrtriebs, des Hubtriebs und der stationären Fernbedienungen notwendig sind.

Er ist in der Funktionsbibliothek „EMS400S_General“ enthalten, siehe Kapitel [7.2.2 DB_GlobalDataEMS \(DB10\)](#).

In der folgenden Tabelle sind die Daten der Struktur „Drive“ aufgelistet, ihre Verarbeitung in „FC_Drive“ wird nachstehend erläutert.

Inhalt der Struktur „Drive“

Tabelle 5-5

Name	Datentyp	Beschreibung
SpeedMaximum	DInt	Maximalgeschwindigkeit, mm/min
SpeedSlow	DInt	langsame Geschwindigkeit, mm/min
SpeedMaxVariable	DInt	Maximale variable Geschwindigkeit, mm/min
SpeedMinVariable	DInt	Minimale variable Geschwindigkeit, mm/min
SpeedMinimum	DInt	Minimalgeschwindigkeit, mm/min
SpeedManual_slow	DInt	Handgeschwindigkeit, langsam, mm/min
SpeedManual_fast	DInt	Handgeschwindigkeit, schnell, mm/min

Name	Datentyp	Beschreibung
Acceleration	Real	Beschleunigung, für Berechnung des Bremswegs
DistanceBraking	Int	mm, Konstante für Bremsweg
Distance_spare	Int	mm, Reserve
DistanceSet_inPosition	Int	mm, Antrieb in Position
DistanceReset_inPosition	Int	mm, Antrieb verlässt Position
Delay_inPosition	Time	Verzögerung, Antrieb in Position
Delay_CrashInitiator	Time	Zeitwert für Auffahrinitiator
AdjustmentSpeed_RemoteControl	Int	Geschwindigkeitsanpassung (%) durch stationäre Bedieneinheit

Berechnung der Sollgeschwindigkeit

Die Kommandos für die Geschwindigkeit werden invertiert betrachtet, damit im Fall einer Brückung / Kommandoüberlagerung zweier RAILS eine geringe Geschwindigkeit gefahren wird.

- **DB_GlobalDataEMS.Drive.SpeedMaximum**
Maximale Geschwindigkeit in Anlage wird gefahren, wenn weder Steuerbit Slow noch Speed_Var ansteht.
- **DB_GlobalDataEMS.Drive.SpeedSlow**
Langsame Geschwindigkeit in Anlage wird gefahren, wenn Steuerbit Slow und nicht Speed_Var ansteht
- **DB_GlobalDataEMS.Drive.SpeedMaxVariable**
Maximale Variable Geschwindigkeit in den Arbeitstrecken wird gefahren, wenn Steuerbit Speed_Var ansteht und die Kommandos Speed_1 bis Speed_6 auf $0_H = 0_D$ stehen
- **DB_GlobalDataEMS.Drive.SpeedMinVariable**
Minimale Variable Geschwindigkeit in den Arbeitstrecken wird gefahren, wenn Steuerbit Speed_Var ansteht und die Kommandos Speed_1 bis Speed_6 auf $3F_H = 63_D$ stehen, sowie bei Schleichfahrt durch den Lichttaster.
- **DB_GlobalDataEMS.Drive.SpeedMinimum**
Minimale Geschwindigkeit in Anlage für Begrenzung der Geschwindigkeit beim Positionieren
- **DB_GlobalDataEMS.Drive.AdjustmentSpeed_RemoteControl**
Bei Verwendung einer stationären Fernbedienung wird die Sollgeschwindigkeit um diesen Prozentwert erhöht oder vermindert.

SpeedMaxVariable und **SpeedMinVariable** werden zur Berechnung der variablen Geschwindigkeit verwendet.

Beispiel:

$$\begin{aligned} \text{SpeedMaxVariable} &= 6000 \text{ [mm/min]} \\ \text{SpeedMinVariable} &= 2000 \text{ [mm/min]} \end{aligned}$$

Die Differenz ist 4000.

In den Arbeitstrecken ergibt sich ein Geschwindigkeitsintervall von $4000 \text{ [mm/min]}/63 = 63,5 \text{ [mm/min]}$.

5.2 Erläuterung der Bausteine

In der folgenden Tabelle sind Beispiele für die Berechnung der Sollgeschwindigkeit dargestellt.

Tabelle 5-6

Kommando Slow	Kommando Speed_Var	Kommando Speed_1 bis Speed_6	Sollgeschwindigkeit
0	0	0	SpeedMaximum
1	0	0	SpeedSlow
0	1	00 _H = 00 _D	SpeedMaxVariable
0	1	01 _H = 01 _D	SpeedMinVariable + 62* Geschwindigkeitsintervall
0	1	02 _H = 02 _D	SpeedMinVariable + 61* Geschwindigkeitsintervall
0	1	20 _H = 32 _D	SpeedMinVariable + 31* Geschwindigkeitsintervall
0	1	3E _H = 62 _D	SpeedMinVariable + 1* Geschwindigkeitsintervall
0	1	3F _H = 63 _D	SpeedMinVariable

Berechnung des Bremswegs

Der Bremsweg ist der Abstand vom Start des Positionierens bis zur Stopp-Position. Er ist abhängig von der aktuellen Geschwindigkeit **V** und dem Beschleunigungswert **a**.

Formel 1:

$$\text{Berechneter Bremsweg} = V^2 / 2a \text{ [mm]}$$

Beschleunigungswert **a**:

Aus den Rampen des FU errechnet sich abhängig von Motor und Getriebe eine Beschleunigung. In der Steuerung wird ein Beschleunigungswert eingestellt, der kleiner/gleich dem FU- Beschleunigungswert sein muss. Dieser Beschleunigungswert muss „DB_GlobalDataEMS.Drive.Acceleration“ eingetragen werden, sonst ist eine fehlerfreie Positionierung nicht möglich.

Bei langsamer Fahrt ist der Wert zu gering, das Fahrzeug ruckelt. Deshalb wird ein Zusatzwert addiert.

Formel 2:

$$\text{Zusatzwert} = 0.001 * V + \text{Konstante} \text{ [mm]}$$

Die **Konstante** ist ein Erfahrungswert und muss im „DB_GlobalDataEMS.Drive.DistanceBraking“ eingetragen werden.

Erfahrungswerte sind z.B.:

Heavy EMS = 40
Light EMS = 20

Formel 3:

$$\text{Gesamter Bremsweg} = \text{Berechneter Bremsweg} + \text{Zusatzwert} \text{ [mm]}$$

In Position

- Das Fahrzeug bleibt stehen, wenn er die mit einer Toleranz von DB_GlobalDataEMS.Drive.**DistanceSet_inPosition** erreicht. Das Fahrzeug läuft noch einige Millimeter nach, je nach Last. Deswegen wird vorher abgeschaltet
Das Statusbit „InPosition“ wird gesetzt.
Das Fahrzeug bleibt stehen, bis das Kommando „Weiterfahren“ kommt.
- Das Statusbit „InPosition“ wird rückgesetzt, wenn der Fahrtrieb weiterfährt und der Abstand zur StoppPosition größer ist als der Wert in DB_GlobalDataEMS.Drive.**DistanceReset_inPosition**.
- Das Statusbit „InPosition“ wird um die Zeit DB_GlobalDataEMS.Drive.**Delay_inPosition** verzögert, um das Positionieren zu beenden.

Auffahrinitiator

Wenn der Auffahrinitiator betätigt wird, dann wird die Freigabe des FU V20 rückgesetzt; das Fahrzeug bleibt stehen. Sie können ihn nur in Betriebsart Hand rückwärts verfahren.

Wenn der Auffahrinitiator wieder frei wird, dann wird der FU V20 erst nach Ablauf der Zeit DB_GlobalDataEMS.Drive.**Delay_CrashInitiator** für die Vorwärtsfahrt freigegeben.

5.2.3 FC_LinearMeasurementSensor (FC61)

Funktion

Der Baustein „FC_LinearMeasurementSensor“ wertet den Barcodewert, die Ist-Geschwindigkeit und das Statusbyte des Sick Linearmessgebers OLM200 aus.

Der Baustein wird in „FC_Drive“ aufgerufen.

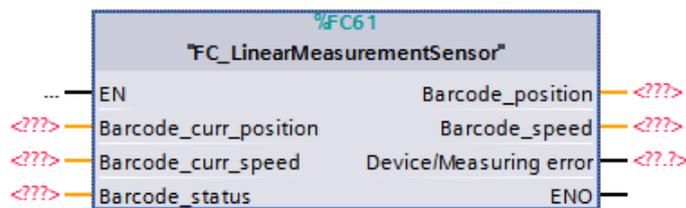
Im Detail umfasst der Baustein die Funktionen:

- Lesen und Speichern des Barcodewerts
- Lesen, Umrechnen und Speichern der Ist-Geschwindigkeit
- Auswertung des Statusbytes

Aufruf und Parameter

Die folgende Abbildung zeigt den Aufruf des Bausteins „FC_LinearMeasurementSensor“.

Abbildung 5-5



Eingangsparameter

Tabelle 5-7

Parameter	Datentyp	Beschreibung
Barcode_curr_position	DINT	Linear-Messsensor, aktuelle Position, von OLM200
Barcode_curr_speed	DINT	Linear-Messsensor, aktuelle Geschwindigkeit, von OLM200
Barcode_status	Byte	Linear-Messsensor, aktueller Status, von OLM200

Ausgangsparameter

Tabelle 5-8

Parameter	Datentyp	Beschreibung
Barcode_position	DINT	aktuelle Position in mm
Barcode_speed	DINT	aktuelle Geschwindigkeit, umgerechnet auf mm/min
Device/Measuring error	Bool	Linear-Messsensor-Fehler

5.2.4 FC_StopPosition (FC62)

Funktion

Der Baustein „FC_StopPosition“ wertet die aktuelle Barcode-Position aus und ermittelt aus den definierten Stopp-Positionen in „DB_StopPositions“ die nächstliegende Stopp-Position in Fahrtrichtung vorwärts.

Der Baustein wird in „FC_Drive“ aufgerufen.

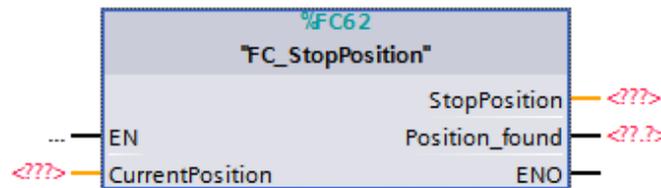
Im Detail umfasst der Baustein die Funktionen:

- Ermitteln der nächsten Stopp-Position in Fahrtrichtung vorwärts
- Auswerten, ob in Fahrtrichtung vorwärts eine Stopp-Position definiert ist

Aufruf und Parameter

Die folgende Abbildung zeigt den Aufruf des Bausteins „FC_StopPosition“.

Abbildung 5-6



Eingangsparameter

Tabelle 5-9

Parameter	Datentyp	Beschreibung
CurrentPosition	DINT	aktuelle Position in mm

Ausgangsparameter

Tabelle 5-10

Parameter	Datentyp	Beschreibung
StopPosition	DINT	Nächstliegende StoppPosition in Fahrtrichtung vorwärts
Position_found	Bool	StoppPosition in Fahrtrichtung vorwärts gefunden

5.2.5 FC_SpeedControl (FC63)

Funktion

Der Baustein „FC_SpeedControl“ wertet in Betriebsart Automatik die aktuelle Geschwindigkeit aus und regelt die Sollgeschwindigkeit.

Der Baustein wird in „FC_Drive“ aufgerufen.

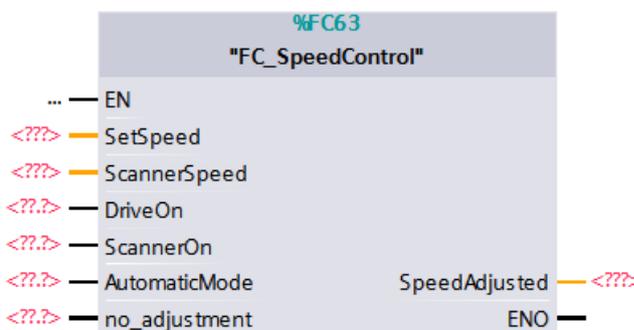
Im Detail umfasst der Baustein die Funktionen:

- aktuelle Geschwindigkeit speichern und einen Mittelwert bilden
- abhängig vom Mittelwert die Sollgeschwindigkeit erhöhen oder vermindern

Aufruf und Parameter

Die folgende Abbildung zeigt den Aufruf des Bausteins „FC_SpeedControl“.

Abbildung 5-7



Eingangsparameter

Tabelle 5-11

Parameter	Datentyp	Beschreibung
SetSpeed	DINT	Geschwindigkeitssollwert
ScannerSpeed	DINT	aktuelle Geschwindigkeit
DriveOn	Bool	Fahrtrieb ist freigegeben
ScannerOn	Bool	Linear-Messsensor ist ein
AutomaticMode	Bool	Automatikbetrieb
No_adjustment	Bool	Keine Geschwindigkeitsanpassung, z.B. : - beim Positionieren - wenn die Stationäre Fernbedienung aktiv ist - in Isolationssegmenten

Ausgangsparameter

Tabelle 5-12

Parameter	Datentyp	Beschreibung
SpeedAdjusted	DINT	angepasste Geschwindigkeit

5.2.6 FC_SeparatorBlocks (FC67)

Funktion

Der Baustein „FC_SeparatorBlocks“ wertet die aktuelle Barcode-Position aus und ermittelt aus den definierten Isolationssegmenten in „DB_SeparatorBlocks“, ob das Fahrzeug in einem Isolationssegment ist.

Rufen Sie den Baustein im zyklischen Anwenderprogramm (OB1) auf, wenn in Ihrer Anlage die Schienensegmente durch Isolationssegmente getrennt sind.

Im Detail umfasst der Baustein die Funktionen:

- aktuelle Barcode-Position auswerten
- Meldung „In_Separator_block“ ausgeben.

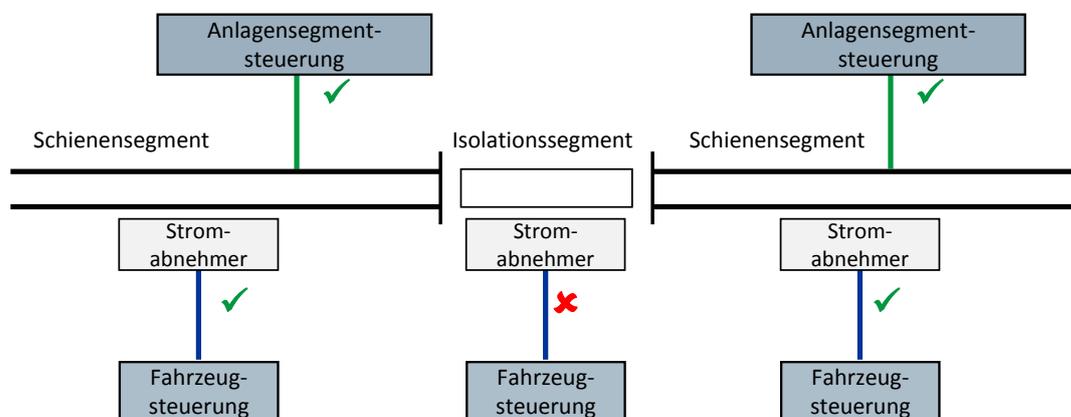
Die Meldung „In_Separator_block“ wird in „FC_Drive“ und in „FB_ErrorsAll“ (in der Funktionsbibliothek „EMS400S_Errors“) ausgewertet.

In Isolationssegmenten hat das Fahrzeug keine Kommunikation zur Anlagen-segmentsteuerung. Damit vermeiden Sie die Brückung zweier Schienensegmente beim Übergang eines Fahrzeugs von einem Schienensegment zum nächsten.

Die folgende Abbildung zeigt den prinzipiellen Einsatz von Isolationssegmenten.

Abbildung 5-8

Kommunikation bei Schienenübergängen



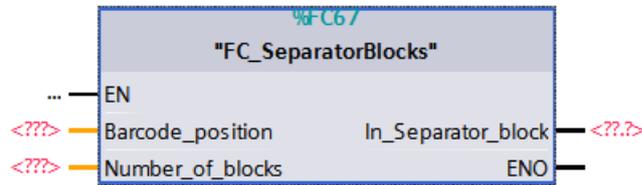
Legende:

- ✓ Kommunikation
- ✗ Keine Kommunikation

Aufruf und Parameter

Die folgende Abbildung zeigt den Aufruf des Bausteins „FC_SeparatorBlocks“.

Abbildung 5-9



Eingangsparameter

Tabelle 5-13

Parameter	Datentyp	Beschreibung
Barcode_position	DINT	aktuelle Position in mm
Number_of_blocks	INT	Anzahl der Isolationssegmente, max. 300

Ausgangsparameter

Tabelle 5-14

Parameter	Datentyp	Beschreibung
In_Separator_block	Bool	Fahrzeug ist in einem Isolationssegment

5.2.7 DB_Drive (DB60)

Der Datenbaustein „DB_Drive“ enthält Daten, die „FC_Drive“ benötigt.

In der folgenden Tabelle sind die Daten des DBs aufgelistet.

Tabelle 5-15

Name	Datentyp	Beschreibung
Stop_position	DInt	nächste StoppPosition
CurrentPosition	DInt	Linear-Messsensor, Positionswert [mm]
CurrentSpeed	DInt	Linear-Messsensor, Geschwindigkeit [mm/min]
Memory_speed	DInt	gespeicherter Geschwindigkeitssollwert
SetSpeed	DInt	Geschwindigkeitssollwert
CurrentSetSpeed	DInt	Geschwindigkeitssollwert zum Positionieren
SetSpeedOutput	DInt	Geschwindigkeitssollwert, Ausgabe
BrakingDistance	Int	Bremsweg, geschwindigkeitsabhängig
start_positioning	Bool	Start positionieren
stop_positioning	Bool	Ende Positionieren
command_positioning	Bool	Positionierbefehl
in_position	Bool	EMS in Position
in_position_delayed	Bool	EMS in Position, verzögert
Edge_start_positioning	Bool	Flanke Start Positionieren
StopPos_found	Bool	StoppPosition vor EMS gefunden
Error_LinearMeasSensor	Bool	Linear-Messsensor-Fehler
Drive_forwards	Bool	Fahren vorwärts
Drive_backwards	Bool	Fahren rückwärts

5.2.8 DB_StopPositions (DB62)

Der Datenbaustein „DB_StopPositions“ enthält die definierten Stopp-Positionen. Diese Daten sind in allen Fahrzeugsteuerungen einer Anlage gleich. Sie werden über die quitierte Datenkommunikation zentral aktualisiert. Die quitierte Datenkommunikation ist in der Funktionsbibliothek „EMS400S_Communication“ enthalten, siehe Kapitel [3.2.2](#) FB_DataTransferEMS (FB30).

In der folgenden Tabelle sind die Daten des DBs aufgelistet.

Tabelle 5-16

Name	Datentyp	Beschreibung
Position	Array [0..29] of DInt	StoppPositionen [mm]

5.2.9 DB_SpeedControl (DB63)

Der Datenbaustein „DB_SpeedControl“ enthält Daten, die „FC_SpeedControl“ benötigt.

In der folgenden Tabelle sind die Daten des DBs aufgelistet.

Tabelle 5-17

Name	Datentyp	Beschreibung
AverageSpeed	DInt	Mittlere Geschwindigkeit [mm/min]
NumberMeas	Int	Anzahl Messungen
Speed	Array [0..39] of DInt	Geschwindigkeitswerte [mm/min]

5.2.10 DB_SeparatorBlocks (DB67)

Der Datenbaustein „DB_SeparatorBlocks“ enthält die Definitionen der Isolationssegmente.

Jedes Isolationssegment wird durch „BarcodeStart“ und „BarcodeEnd“ definiert.

Der DB ist für maximal 300 Isolationssegmente ausgelegt.

Diese Daten sind in allen Fahrzeugsteuerungen einer Anlage gleich. Sie werden **nicht** über die quitierte Datenkommunikation zentral aktualisiert.

In der folgenden Tabelle sind die Daten des DBs aufgelistet.

Tabelle 5-18

Name	Datentyp	Beschreibung
Separatorblock	Array [0..300] of Struct	Isolationssegmente
Separatorblock [n]		
BarcodeStart	DInt	Start des Isolationssegments [mm]
BarcodeEnd	DInt	Ende des Isolationssegments [mm]

6 Funktionsbibliothek „EMS400S_Errors“

Diese Funktionsbibliothek enthält Bausteine für die Fahrzeugsteuerung. Diese Bausteine bearbeiten die Störungsmeldungen und Warnungen.

Sie benötigen die Bausteine, um

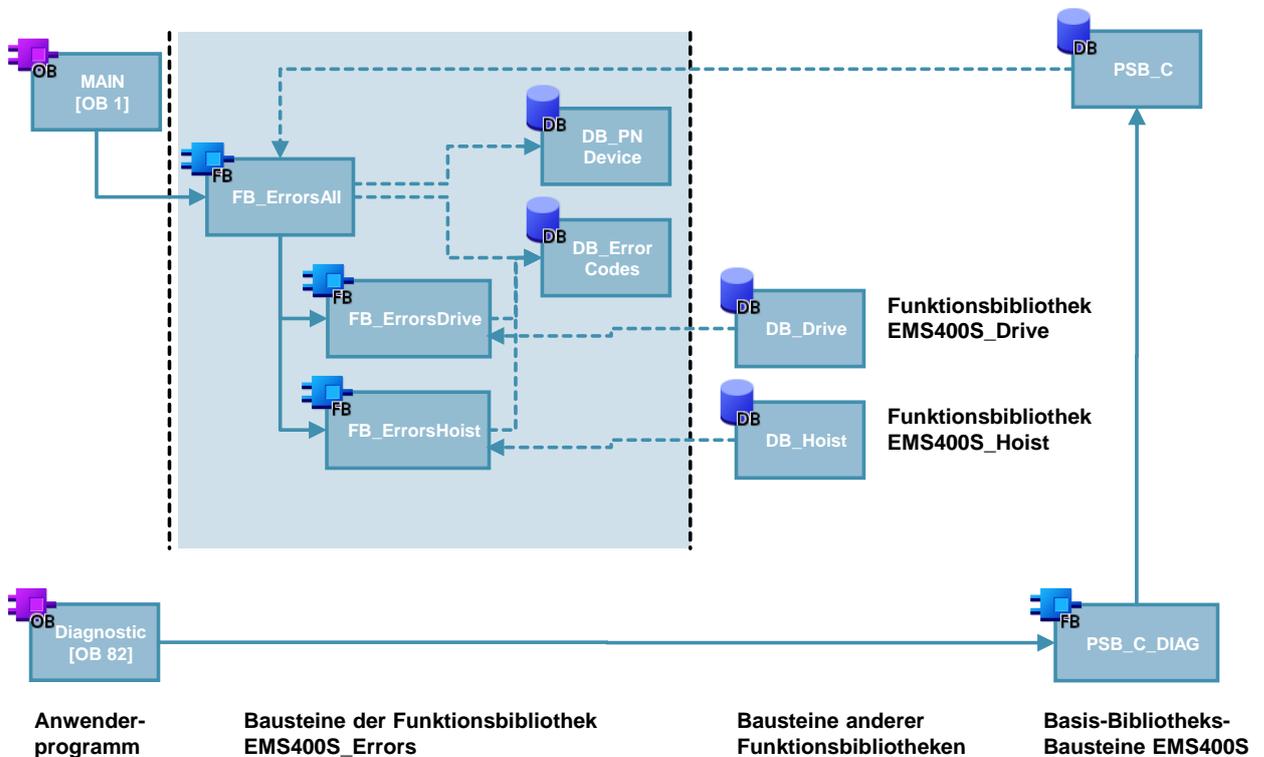
- die Störungsmeldungen und Warnungen zu erfassen.
- die Sammelmeldung zu generieren

6.1 Bibliotheksinhalt

Schematische Darstellung

In der folgenden Abbildung sind die Bausteine markiert, die in der Funktionsbibliothek „EMS400S_Errors“ enthalten sind.

Abbildung 6-1



Auflistung der Bausteine

Die folgende Tabelle listet alle zur Funktionsbibliothek „EMS400S_Errors“ gehörenden Bausteine auf.

Tabelle 6-1

Baustein	Symbol	Kommentar
FB50	FB_ErrorsAll	Störungsmeldungen, Aufrufe der „FB_ErrorsDrive“ und „FB_ErrorsHoist“
FB54	FB_ErrorsDrive	Störungsmeldungen des Fahrtriebs
FB55	FB_ErrorsHoist	Störungsmeldungen des Hubtriebs
DB50	DB_ErrorCodes	Störung- und Meldungssignale, und die zugeordneten Fehlercodes für das Display
DB53	DB_PNDevice	Globaler Datenbaustein für Diagnose der PN-Teilnehmer

6.2 Erläuterung der Bausteine

Das folgende Kapitel erläutert alle zur Funktionsbibliothek „EMS400S_Errors“ gehörenden Bausteine.

6.2.1 FB_ErrorsAll (FB50)**Funktion**

Der Baustein „FB_ErrorsAll“ bildet alle Störungsmeldungen der Fahrzeugsteuerung.

Rufen Sie den Baustein im zyklischen Anwenderprogramm (OB1) auf.

Im Detail umfasst der Baustein die Funktionen:

- Störungsmeldungen für das PSB-C-Modul bilden
- Diagnose für PN-Teilnehmer aufrufen (FC „DeviceStates“ mit Mode 2)
- Störungsmeldungen für den Fahrtrieb aufrufen („FB_ErrorsDrive“)
- Störungsmeldungen für den Hubtrieb aufrufen („FB_ErrorsHoist“)
- Allgemeine Meldungen bilden
- Sammelmeldung bilden

In Isolationssegmenten werden die Störungsmeldungen für das PSB-C-Modul unterdrückt.

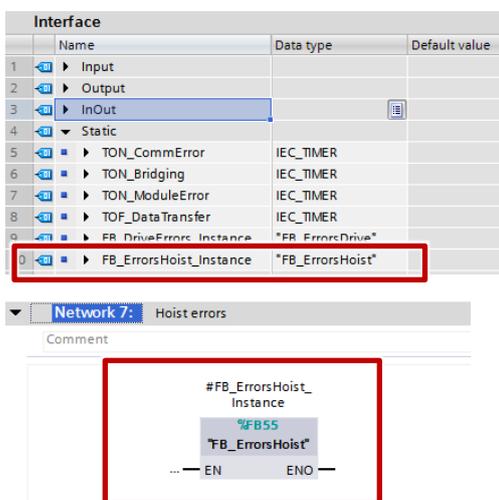
Hinweis

Die Störungsmeldungen für den Hubantrieb werden nur benötigt, wenn ein Hubantrieb / Kettenzug vorhanden ist. Besitzt Ihre Fahrzeugsteuerung keinen Hubantrieb, können Sie im Baustein „FB_ErrorsAll“, Netzwerk 7 den Aufruf des „FB_ErrorsHoist“ und die Instanz in der Schnittstelle löschen.

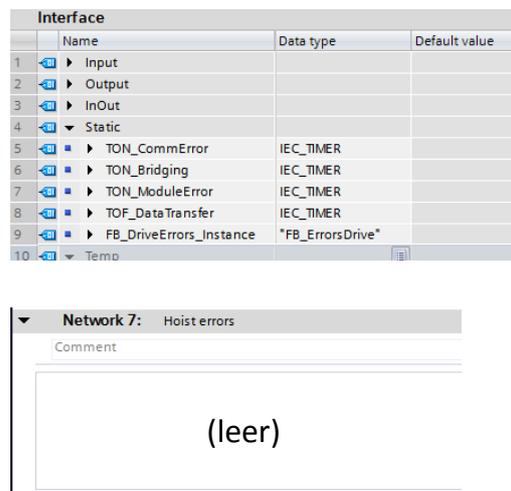
In der folgenden Abbildung sehen Sie die Schnittstelle und das Netzwerk.

Abbildung 6-2

Mit FB_ErrorsHoist, Lieferzustand



Ohne FB_ErrorsHoist

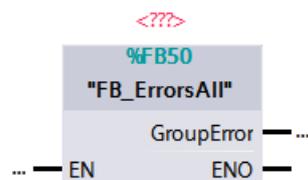


© Siemens AG Copyright-2014 All rights reserved

Aufruf und Parameter

Die folgende Abbildung zeigt den Aufruf des Bausteins „FB_ErrorsAll“.

Abbildung 6-3



Ausgangsparameter

Tabelle 6-2

Parameter	Datentyp	Beschreibung
GroupError	Bool	Sammelfehler

6.2.2 FB_ErrorsDrive (FB54)

Funktion

Der Baustein „FB_ErrorsDrive“ bildet die Störungsmeldungen des Fahrtriebs. Er wird als Multi-Instanz in „FB_ErrorsAll“ aufgerufen.

Im Detail realisiert der Baustein die Funktionen:

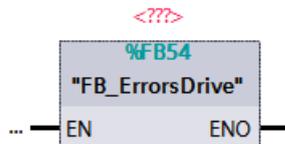
- Fehler des Frequenzumrichters speichern
- Fehler des Linear-Messensors speichern
- PN Diagnose des Linear-Messensors speichern
- Schwergängigkeit des Fahrtriebs feststellen
- Positionsänderung des Fahrtriebs ohne Kommando feststellen
- „StoppPosition überfahren“ feststellen

Der Baustein „FB_ErrorsDrive“ greift auf „DB_Drive“ zu. Dieser DB ist in der Funktionsbibliothek „EMS400S_Drive“ enthalten, siehe Kapitel [5.2.7 DB_Drive \(DB60\)](#).

Aufruf und Parameter

Die folgende Abbildung zeigt den Aufruf des Bausteins „FB_ErrorsDrive“.

Abbildung 6-4



Der Baustein hat keine Parameter

6.2.3 FB_ErrorsHoist (FB55)

Funktion

Der Baustein „FB_ErrorsHoist“ bildet die Störungsmeldungen des Hubantriebs. Er wird als Multi-Instanz in „FB_ErrorsAll“ aufgerufen.

Im Detail werden folgende Funktionen realisiert:

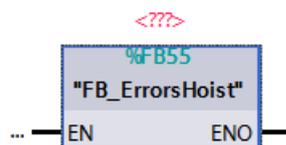
- Fehler des Kettenzugs speichern
- Fehler des TR Trommelgebers bei Justierung speichern
- PN Diagnose des TR Trommelgebers speichern
- Positionsänderung des Hubantriebs ohne Kommando feststellen
- Schwergängigkeit des Hubantriebs feststellen
- feststellen, ob am Ende des freigegebenen Hubbereichs die definierte Höhe erreicht wurde
- Hubbewegung in falsche Richtung feststellen
- Ansprechen der Endschalter feststellen

Der Baustein „FB_ErrorsHoist“ greift auf „DB_Hoist“ zu. Dieser DB ist in der Funktionsbibliothek „EMS400S_Hoist“ enthalten, siehe Kapitel [8.2.5 DB_Hoist \(DB70\)](#).

Aufruf und Parameter

Die folgende Abbildung zeigt den Aufruf des Bausteins „FB_ErrorsHoist“.

Abbildung 6-5



Der Baustein hat keine Parameter

6.2.4 DB_ErrorCodes (DB50)**Inhalt**

Der Baustein „DB_ErrorCodes“ enthält die Störung- und Meldungssignale, und die zugeordneten Fehlercodes für das Display.

Die Fehlercodes sind im Values-Array (Array[1..32] of WORD) gespeichert.

Die vorderste Stelle jedes WORDs hat folgende Bedeutung:

0xxx	rote LED ist aus
1xxx	Dauerlicht der roten LED
2xxx	schnelles Blinklicht der roten LED
4xxx	langsames Blinklicht der roten LED

In der folgenden Tabelle sind die Nummernbänder der Fehlercodes aufgelistet.

Tabelle 6-3

Nummernband	Funktionsbereich
500 - 507	PSB-C-Modul
600 - 607	Fahrtrieb
700 - 709	Hubtrieb
800 - 807	allgemein

In der folgenden Tabelle sind die Fehlercodes und die Störungssignale für das PSB-C-Modul aufgelistet.

Tabelle 6-4

Values[Nr]	Fehler code	Beschreibung	Störungssignal
1	500	PSB-C: Kommunikation Schiene (not valid)	ErrorWord0.Bit_00
2	501	PSB-C: Diagnose, Sammelfehler	ErrorWord0.Bit_01
3	502	PSB-C: Störung Idle-Bit (Idle liegt an)	ErrorWord0.Bit_02
4	503	PSB-C: Brückung	ErrorWord0.Bit_03
5	504	PSB-C: Temperaturwarnung	ErrorWord0.Bit_04
6	505	Reserve	ErrorWord0.Bit_05
7	506	Reserve	ErrorWord0.Bit_06
8	507	Reserve	ErrorWord0.Bit_07

6.2 Erläuterung der Bausteine

In der folgenden Tabelle sind die Fehlercodes und die Störungssignale für den Hubantrieb aufgelistet.

Tabelle 6-5

Values[Nr]	Fehler code	Beschreibung	Störungssignal
9	700	TR Trommelgeber, Fehler bei Justierung	ErrorWord0.Bit_10
10	701	Kettenzug Motorschutzschalter	ErrorWord0.Bit_11
11	702	Kettenzug Temperatur	ErrorWord0.Bit_12
12	703	Fehler: PN Slave TR Trommelgeber	ErrorWord0.Bit_13
13	704	Hubbewegung ohne Befehl	ErrorWord0.Bit_14
14	705	Hubgeschwindigkeit zu niedrig	ErrorWord0.Bit_15
15	706	Position nicht erreicht	ErrorWord0.Bit_16
16	707	Hub in falsche Richtung	ErrorWord0.Bit_17
28	708	Unterer Hub-Endschalter	ErrorWord1.Bit_13
29	709	Oberer Hub-Endschalter	ErrorWord1.Bit_14

In der folgenden Tabelle sind die Fehlercodes und die Störungssignale für den Fahrtrieb aufgelistet.

Tabelle 6-6

Values[Nr]	Fehler code	Beschreibung	Störungssignal
17	600	Sammelfehler FU	ErrorWord1.Bit_00
18	601	Linear-Messsensor-Fehler	ErrorWord1.Bit_01
19	602	Fehler: PN Slave SICK Linear-Messsensor	ErrorWord1.Bit_02
20	603	Gehänge schwergängig, zu langsam	ErrorWord1.Bit_03
21	604	Fahrbewegung ohne Befehl	ErrorWord1.Bit_04
22	605	StoppPosition überfahren	ErrorWord1.Bit_05
23	606	Stopp durch Auffahrinitiator (Meldung)	ErrorWord1.Bit_06
24	607	Reserve	ErrorWord1.Bit_07

In der folgenden Tabelle sind die Fehlercodes und die Störungssignale für die allgemeinen Meldungen aufgelistet.

Tabelle 6-7

Values[Nr]	Fehler code	Beschreibung	Störungssignal
25	800	Betriebsart Hand (Meldung)	ErrorWord1.Bit_10
26	801	Datentransfer ist aktiv (Meldung)	ErrorWord1.Bit_11
27	802	Keine Globale Daten für Fahrtrieb vorhanden	ErrorWord1.Bit_12
30	805	Reserve	ErrorWord1.Bit_15
31	806	Reserve	ErrorWord1.Bit_16
32	807	Reserve	ErrorWord1.Bit_17

6.2 Erläuterung der Bausteine

6.2.5 DB_PNDevice (DB53)

Inhalt

Der Baustein „DB_PNDevice“ enthält Daten für die Diagnose der PN-Teilnehmer mit FC „DeviceStates“.

FC „DeviceStates“ ist im TIA-Portal enthalten und wird von „FB_ErrorsAll“ mit Mode 2 aufgerufen.

In der folgenden Tabelle sind die Daten des DBs aufgelistet.

Tabelle 6-8

Name	Datentyp	Beschreibung
Device_State	Array [0..1023] of Bool	Status der PN-Teilnehmer
RetVal	WORD	Status des FC „DeviceStates“

7 Funktionsbibliothek „EMS400S_General“

Diese Funktionsbibliothek enthält Bausteine für die Fahrzeugsteuerung. Diese Bausteine bearbeiten die Betriebsarten und Daten für die Antriebe.

Sie benötigen die Bausteine, um

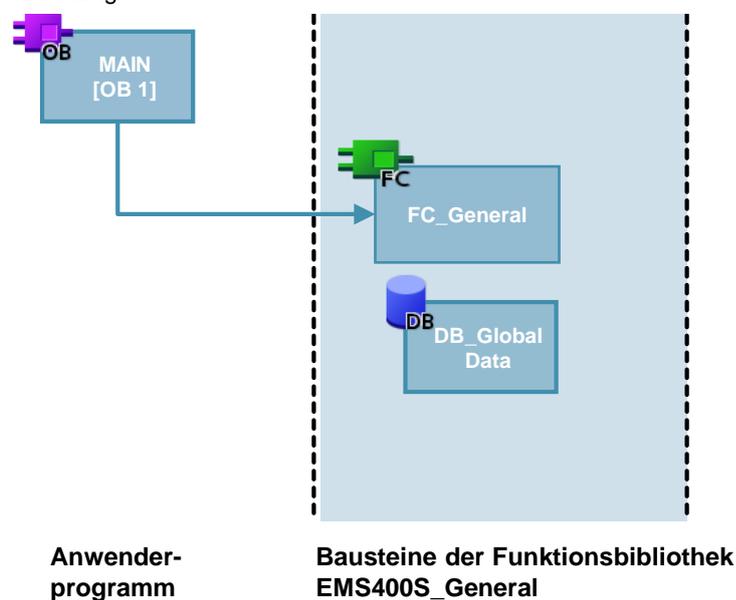
- die Betriebsarten zu bilden.
- Daten für Fahr- und Hubantriebe bereitzustellen

7.1 Bibliotheksinhalt

Schematische Darstellung

In der folgenden Abbildung sind die Bausteine markiert, die in der Funktionsbibliothek „EMS400S_General“ enthalten sind.

Abbildung 7-1



Auflistung der Bausteine

Die folgende Tabelle listet alle zur Funktionsbibliothek „EMS400S_General“ gehörenden Bausteine auf.

Tabelle 7-1

Baustein	Symbol	Hinweis
FC10	FC_General	Betriebsarten, Quittieren
DB10	DB_GlobalDataEMS	Globaler Datenbaustein, Daten für Fahr- und Hubantrieb und stationäre Fernbedienungen
Variablentabelle	CAR tags	Alle Eingänge, Ausgänge und Merker, die in der Fahrzeugsteuerung SIMATIC EMS400S verwendet werden

7.2 Erläuterung der Bausteine

Das folgende Kapitel erläutert alle zur Funktionsbibliothek „EMS400S_General“ gehörenden Bausteine.

7.2.1 FC_General (FC10)

Funktion

Der Baustein „FC_General“ bildet die Betriebsarten Automatik und Hand, und das Quittiersignal.

Rufen Sie den Baustein im zyklischen Anwenderprogramm (OB1) auf.

Im Detail realisiert der Baustein die folgenden Funktionen:

- Betriebsart Automatik (Hand = nicht Automatik)
- Verzögertes Signal nach Neustart der CPU
- Sammelsignal zum Quittieren von Störungen

Der Baustein wertet folgendes aus:

- Betriebsarten-Signale der Fernbedienung
- Neustart der CPU
- Quittierkommando aus der Kommandoschnittstelle
- Quittierkommando von der Fernbedienung

Aufruf und Parameter

Die folgende Abbildung zeigt den Aufruf des Bausteins „FC_General“.

Abbildung 7-2



Der Baustein hat keine Parameter.

7.2.2 DB_GlobalDataEMS (DB10)

Inhalt

Der Baustein „DB_GlobalDataEMS“ enthält in drei Strukturen die globalen Daten. Diese sind notwendig für den Betrieb des Fahrtriebs, des Hubantriebs und der stationären Fernbedienungen.

Diese globalen Daten sind in allen Fahrzeugsteuerungen einer Anlage gleich. Sie werden über die quitierte Datenkommunikation zentral aktualisiert. Die quitierte Datenkommunikation ist in der Funktionsbibliothek „EMS400S_Communication“ enthalten, siehe Kapitel [3.2.2](#) FB_DataTransferEMS (FB30).

Die folgende Tabelle zeigt die enthaltenen Strukturen:

Tabelle 7-2

Struktur	Verwendung	Beschreibung in Funktionsbibliothek
Drive	Fahrtrieb	„EMS400S_Drive“, siehe DB_GlobalDataEMS, Struktur Drive (DB10)
Hoist	Hubantrieb	„EMS400S_Hoist“, siehe DB_GlobalDataEMS, Struktur Hoist (DB10)
Remote_Control	Stationäre Fernbedienungen	„EMS400S_Display_RemoteControl“, siehe DB_GlobalDataEMS, Struktur Remote_Control (DB10)

Die Daten werden in den Kapiteln der entsprechenden Funktionsbibliotheken erläutert.

7.2.3 Variablen-tabelle (CAR tags)

Die Variablen-tabelle „CAR tags“ enthält die Deklaration der Eingänge, Ausgänge und Merker, die für den Betrieb der Fahrzeugsteuerung SIMATIC EMS400S notwendig sind.

Die Systemmerker und Taktmerker sind nicht enthalten.

8 Funktionsbibliothek „EMS400S_Hoist“

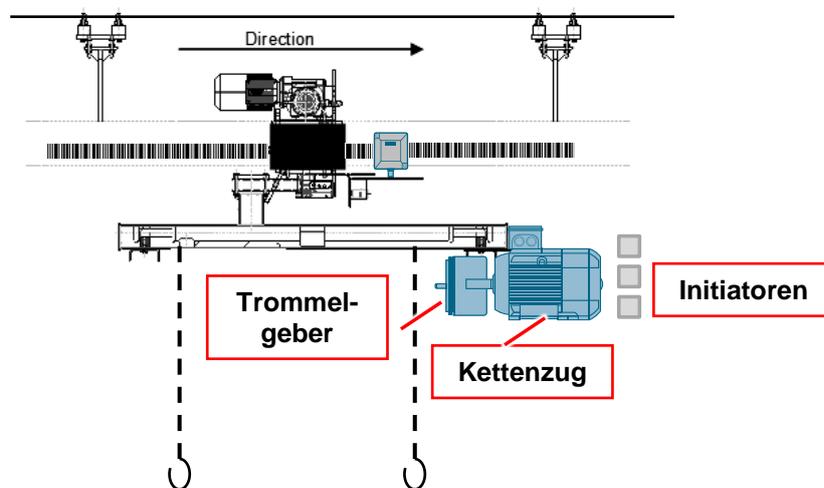
Diese Funktionsbibliothek enthält Bausteine für die Fahrzeugsteuerung. Diese Bausteine bearbeiten den Antrieb in z-Richtung, den Hubantrieb / Kettenzug.

Sie benötigen die Bausteine, um

- die Höhe mit dem Trommelgeber zu erfassen
- den Hubantrieb / Kettenzug anzusteuern.

Die folgende Abbildung zeigt schematisch die Aktoren und Sensoren für die Hubantriebsfunktion.

Abbildung 8-1

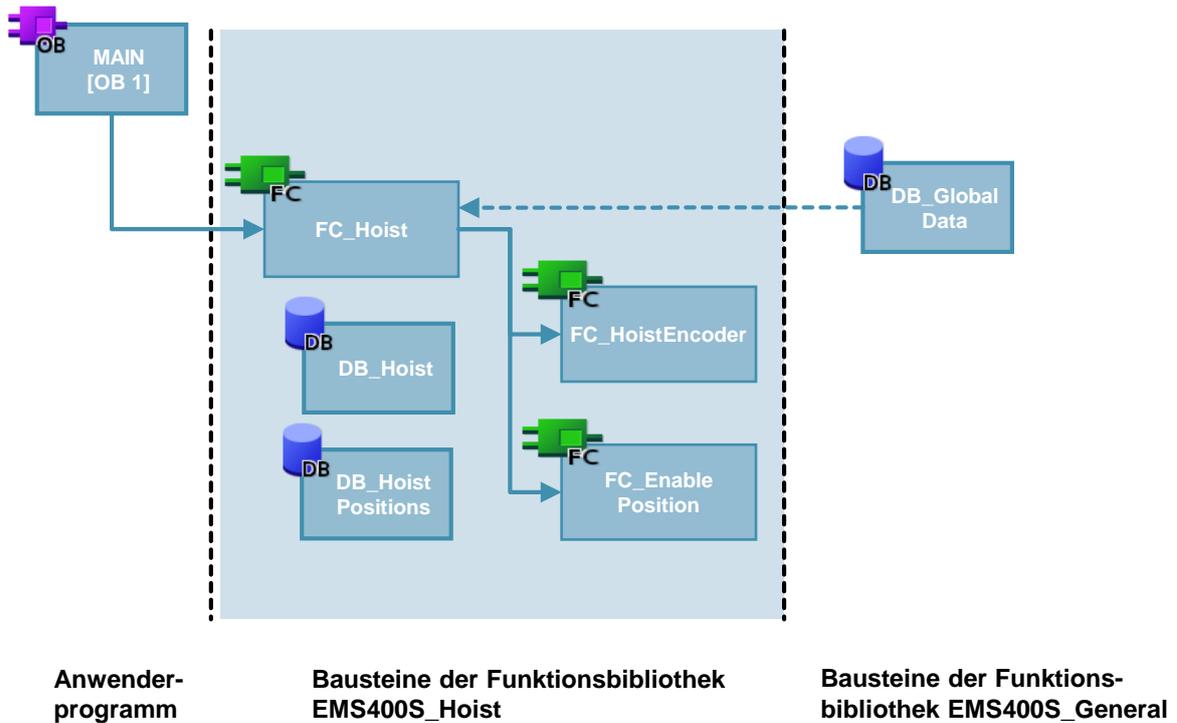


8.1 Bibliotheksinhalt

Schematische Darstellung

In der folgenden Abbildung sind die Bausteine markiert, die in der Funktionsbibliothek „EMS400S_Hoist“ enthalten sind.

Abbildung 8-2



Auflistung der Bausteine

Die folgende Tabelle listet alle zur Funktionsbibliothek „EMS400S_Hoist“ gehörenden Bausteine auf.

Tabelle 8-1

Baustein	Symbol	Hinweis
FC70	FC_Hoist	Ansteuerung des Hubantrieb / Kettenzugs
FC71	FC_HoistEncoder	Auswertung des TR Trommelgebers
FC72	FC_HoistEnable Position	Freigabe des Hubs, Ermittlung der Soll-Höhe
DB70	DB_Hoist	Datenbaustein für den Hubantrieb / Kettenzug
DB72	DB_HoistPositions	Datenbaustein für freigegebene Bereiche und Soll-Höhen, Hubtabelle

8.2 Erläuterung der Hubantriebs-Bausteine

Das folgende Kapitel erläutert alle zur Funktionsbibliothek „EMS400S_Hoist“ gehörenden Bausteine.

8.2.1 FC_Hoist (FC70)

Funktion

Der Baustein „FC_Hoist“ steuert den Hubantrieb / Kettenzug der z-Richtung an. Rufen Sie den Baustein im zyklischen Anwenderprogramm (OB1) auf.

Der Baustein realisiert die folgenden Funktionen:

- Freigabe des Kettenzugs
- Auswertung der Betriebsarten Automatik / Hand
- Auswertung der Kommandos aus der Schnittstelle zur Anlagensegmentsteuerung in Betriebsart Automatik
- Positionieren auf die Soll-Höhe in Betriebsart Automatik
- Auswertung der Signale der Fernbedienung in Betriebsart Hand
- Auswertung der Signale der stationären Fernbedienung
- Auswertung der drei Initiatoren
- Auswahl der Sollgeschwindigkeit
- Ansteuerung des Kettenzugs

Der Baustein wertet folgendes aus:

- die Betriebsarten
- die drei Initiatoren
- die Signale der Fernbedienungen
- den Wert des Trommelgebers
- die Fehlersignale des Kettenzugs
- die Kommandos aus der Schnittstelle zur Anlagensegmentsteuerung

Der Baustein gibt folgende Signale an den Kettenzug aus:

- das Freigabesignal
- das Richtungssignal
- das Geschwindigkeitssignal

Wird die Sollposition im Automatikbetrieb nicht rechtzeitig bis zum Ende der freigegebenen Strecke erreicht, bleibt der Fahrtrieb solange stehen, bis der Hub in Sollposition ist.

Der Fahrtrieb wird durch „FC_Drive“ angehalten, siehe Kapitel [5.2.1 FC_Drive \(FC60\)](#).

Eine Fehlermeldung wird durch „FB_ErrorsHoist“ angezeigt, siehe Kapitel [6.2.3 FB_ErrorsHoist \(FB55\)](#).

8.2 Erläuterung der Hubantriebs-Bausteine

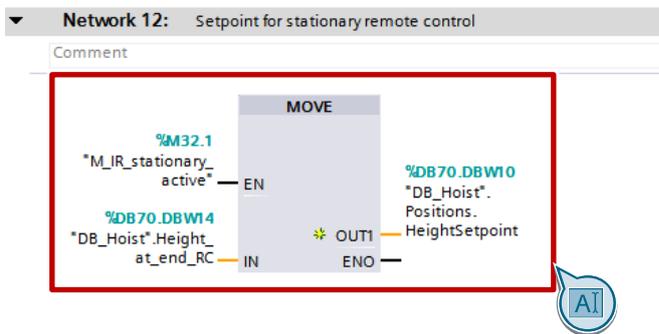
Im Handbetrieb hebt oder senkt der Hubantrieb bis zur gespeicherten Sollposition und bleibt dann stehen.
Wird die Taste der Fernbedienung erneut betätigt, hebt oder senkt der Hubantrieb weiter.

Hinweis Besitzt Ihre Fahrzeugsteuerung keine stationäre Fernbedienung, können Sie die Anweisung in Netzwerk 12 und die Abfragen der Merker für die stationäre Fernbedienung in den Netzwerken 19 bis 22 löschen.

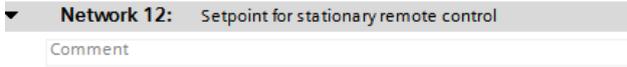
In der folgenden Abbildung sehen Sie Netzwerk 12.

Abbildung 8-3

mit stationärer Fernbedienung, Lieferzustand



Ohne stationäre Fernbedienung



(leer)

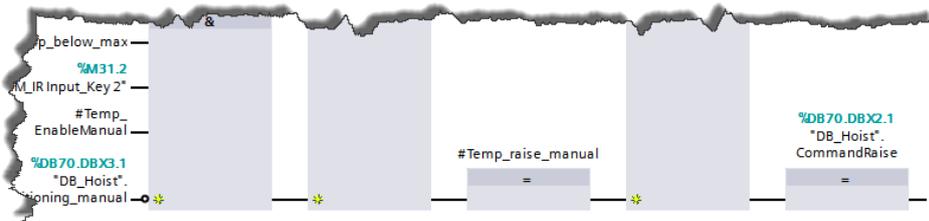
In der folgenden Abbildung sehen Sie als Beispiel Netzwerk 19. Netzwerk 21 ist entsprechend zu ändern.

Abbildung 8-4

mit stationärer Fernbedienung, Lieferzustand



Ohne stationäre Fernbedienung

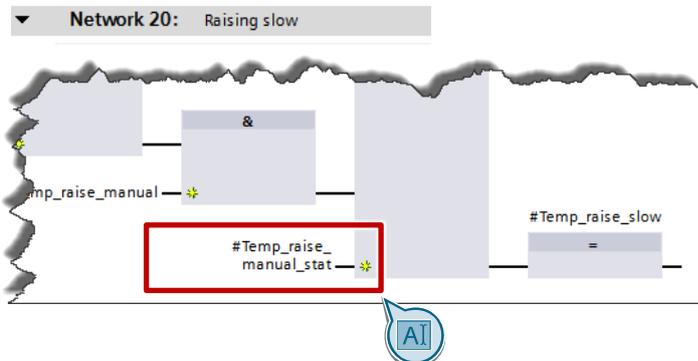


8.2 Erläuterung der Hubantriebs-Bausteine

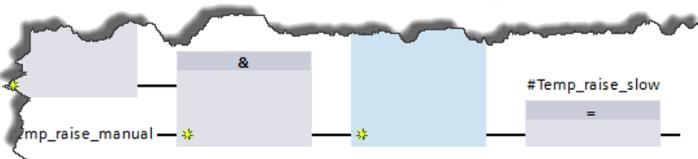
In der folgenden Abbildung sehen Sie als Beispiel Netzwerk 20. Netzwerk 22 ist entsprechend zu ändern.

Abbildung 8-5

mit stationärer Fernbedienung, Lieferzustand



Ohne stationäre Fernbedienung



© Siemens AG Copyright-2014 All rights reserved

Aufruf und Parameter

Die folgende Abbildung zeigt den Aufruf des Bausteins „FC_Hoist“.

Abbildung 8-6



Der Baustein hat keine Parameter.

Peripherie

„FC_Hoist“ wertet folgende Eingänge aus:

Tabelle 8-2

Name	Datentyp	Beschreibung
I_Reference_Initiator	Bool	Referenzinitiator
I_Hoist_top_limit	Bool	Oberer Endschalter
I_Hoist_lower_limit	Bool	Unterer Endschalter
I_HoistCircuitBreaker	Bool	Kettenzug Motorschutzschalter
I_HoistTemperature	Bool	Kettenzug Temperatur
I_HoistPosition	DInt	Trommelgeber, Höhe

8.2 Erläuterung der Hubantriebs-Bausteine

„FC_Hoist“ steuert folgende Ausgänge an:

Tabelle 8-3

Name	Datentyp	Beschreibung
Q_HoistUp	Bool	Hub auf
Q_HoistDown	Bool	Hub ab
Q_HoistFast	Bool	Hub schnell

Kommandos

„FC_Hoist“ wertet folgende Kommandos aus der Schnittstelle zur Anlagensegmentsteuerung aus:

Tabelle 8-4

Symbol	Beschreibung
Enable_Hoist	Hub ein z-Richtung
Enable_Hoist_Up	Hub in obere Position z-Richtung

8.2.2 DB_GlobalDataEMS, Struktur Hoist (DB10)**Funktion**

Der Baustein „DB_GlobalDataEMS“ enthält die globalen Daten, die für den Betrieb des Fahrtriebs, des Hubantriebs und der stationären Fernbedienungen notwendig sind.

Er ist in der Funktionsbibliothek „EMS400S_General“ enthalten, siehe Kapitel [7.2.2 DB_GlobalDataEMS \(DB10\)](#).

In der folgenden Tabelle sind die Daten der Struktur „Hoist“ aufgelistet, ihre Verarbeitung in „FC_Hoist“ wird nachstehend erläutert.

Inhalt der Struktur „Hoist“

Tabelle 8-5

Name	Datentyp	Beschreibung
Height_Reference	Int	obere Position / Höhe Referenzinitiator, mm
Height_maximum	Int	obere Position / Höhe, mm
Height_minimum	Int	untere Position / Höhe, mm
DistanceSet_inPosition	Int	Antrieb in Position, mm
DistanceReset_inPosition	Int	Antrieb verlässt Position, mm
Distance_slow_raising	Int	Umschalten auf langsam, mm
Distance_stop_raising	Int	Abschalten bei Positionieren, mm
Distance_slow_lowering	Int	Umschalten auf langsam, mm
Distance_stop_lowering	Int	Abschalten bei Positionieren, mm

Höhenwerte

- Der Trommelgeber muss justiert werden. Wenn der Referenzinitiator betätigt wird, dann wird der Wert „DB_GlobalDataEMS.Hoist.**Height_Reference**“3 an den Trommelgeber übertragen.
- DB_GlobalDataEMS.Hoist.**Height_maximum** ist der obere Grenzwert; bei Erreichen schaltet „FC_Hoist“ den Kettenzug ab.
- DB_GlobalDataEMS.Hoist.**Height_minimum** ist der untere Grenzwert; bei Erreichen schaltet „FC_Hoist“ den Kettenzug ab.

Umschalten der Sollgeschwindigkeit und Stopp

Der Kettenzug hat 2 Geschwindigkeiten. Im Allgemeinen steuert „FC_Hoist“ die schnelle Geschwindigkeit an.

- DB_GlobalDataEMS.Hoist.**Distance_slow_raising**
Wenn der Kettenzug beim Heben diese Distanz zur Soll-Höhe hat, steuert „FC_Hoist“ die langsame Geschwindigkeit an.
- DB_GlobalDataEMS.Hoist.**Distance_stop_raising**
Wenn der Kettenzug beim Heben diese Distanz zur Soll-Höhe hat, schaltet „FC_Hoist“ den Kettenzug ab. Der Kettenzug läuft - je nach Last - noch einige Millimeter nach. Deswegen wird vorher abgeschaltet.
- DB_GlobalDataEMS.Hoist.**Distance_slow_lowering**
Wenn der Kettenzug beim Senken diese Distanz zur Soll-Höhe hat, steuert „FC_Hoist“ die langsame Geschwindigkeit an.
- DB_GlobalDataEMS.Hoist.**Distance_stop_lowering**
Wenn der Kettenzug beim Senken diese Distanz zur Soll-Höhe hat, schaltet „FC_Hoist“ den Kettenzug ab. Der Kettenzug läuft - je nach Last - noch einige Millimeter nach. Deswegen wird vorher abgeschaltet.

In Position

- Das Statusbit „InPosition“ wird gesetzt, wenn der Kettenzug an der StoppPosition innerhalb der Distanz in „DB_GlobalDataEMS.Hoist.**DistanceSet_inPosition**“ steht. Der Kettenzug bleibt solange stehen, bis eine neue Soll-Höhe freigegeben ist und das Kommando Enable_Hoist ansteht.
- Das Statusbit „InPosition“ wird rückgesetzt, wenn der Kettenzug weiter als die Distanz in „DB_GlobalDataEMS.Hoist.**DistanceReset_inPosition**“ von der StoppPosition entfernt ist.

8.2.3 FC_HoistEncoder (FC71)

Funktion

Der Baustein „FC_HoistEncoder“ wertet den Absolutwert des TR Trommelgebers TR_C58M aus. Bei Erreichen des Referenzinitiators führt der FC die Pre-Justage-Funktion aus, wenn in Betriebsart Hand die Tasten 2 und 7 der Fernbedienung betätigt werden.

Er wird in „FC_Hoist“ aufgerufen.

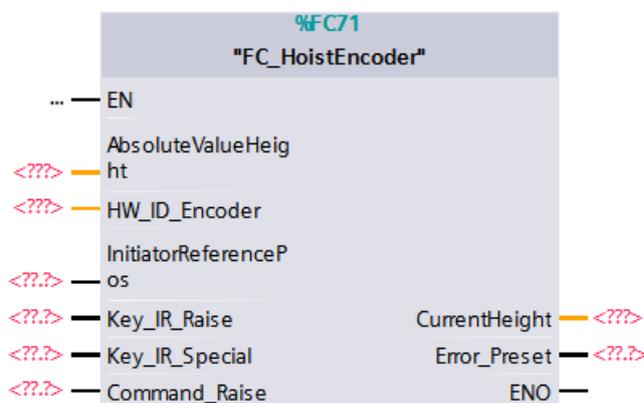
Im Detail umfasst der Baustein die Funktionen:

- Lesen und Speichern des Absolutwerts
- Justieren des TR Trommelgebers

Aufruf und Parameter

Die folgende Abbildung zeigt den Aufruf des Bausteins „FC_HoistEncoder“.

Abbildung 8-7



Eingangsparameter

Tabelle 8-6

Parameter	Datentyp	Beschreibung
AbsoluteValueHeight	DIInt	Absolutwert Höhe, von Trommelgeber TR C_H58M
HW_ID_Encoder	HW_ANY	HW_ID der TR-Geber-Daten
InitiatorReferencePos	Bool	Referenzinitiator
Key_IR_Raise	Bool	IR Fernbedienung Taste 2
Key_IR_Special	Bool	IR Fernbedienung Taste 7
Command_Raise	Bool	Befehl zum Heben

Ausgangsparameter

Tabelle 8-7

Parameter	Datentyp	Beschreibung
CurrentHeight	Int	aktuelle Höhe in mm
Error_Preset	Bool	Fehler beim Justieren /Voreinstellen des Gebers

8.2.4 FC_HoistEnablePosition (FC72)

Funktion

Der Baustein „FC_HoistEnablePosition“ wertet die aktuelle Barcode-Position aus. Er ermittelt aus den definierten Hubfreigabebereichen und Soll-Höhen in DB_HoistPositions, ob der Hub freigegeben ist und welche Soll-Höhe erreicht werden soll.

Der Baustein wird in „FC_Hoist“ aufgerufen.

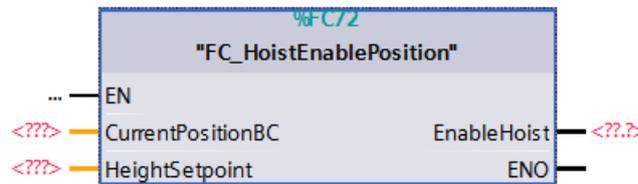
Im Detail realisiert der Baustein die folgenden Funktionen:

- Ermitteln der Hubfreigabe
- Ermitteln der Soll-Höhe

Aufruf und Parameter

Die folgende Abbildung zeigt den Aufruf des Bausteins „FC_HoistEnablePosition“.

Abbildung 8-8



Eingangsparameter

Tabelle 8-8

Parameter	Datentyp	Beschreibung
CurrentPositionBC	DINT	Aktuelle BarCode-Position

Ausgangsparameter

Tabelle 8-9

Parameter	Datentyp	Beschreibung
EnableHoist	Bool	Hubbewegung freigegeben

Ein-/Ausgangsparameter

Tabelle 8-10

Parameter	Datentyp	Beschreibung
HeightSetpoint	Int	Sollwert Höhe, mm

8.2.5 DB_Hoist (DB70)**Inhalt**

Der Datenbaustein „DB_Hoist“ enthält Daten, die „FC_Hoist“, „FC_HoistEncoder“ und „FC_HoistPositions“ benötigen.

In der folgenden Tabelle sind die Daten des DBs aufgelistet.

Tabelle 8-11

Name	Datentyp	Beschreibung
CurrentHeight	Int	aktuelle Höhe in mm
Error_Preset	Bool	Fehler beim Voreinstellen des Gebers
CommandRaise	Bool	Befehl: Hub heben
CommandLower	Bool	Befehl: Hub senken
in_position	Bool	Hub in Position
spare_2.4	Bool	Reserve
Edge_end_of_area	Bool	Freigegebene Strecke wurde verlassen
Error_not_in_position	Bool	Hub ist nicht in Position
Edge_Manual_raise	Bool	Flanke, IR-Taste heben
Edge_Manual_lower	Bool	Flanke, IR-Taste senken
Positioning_manual	Bool	Positionieren im Handbetrieb
Encoder	Struct	Daten für „FC_HoistEncoder“
PresetDrumEncoder	DInt	Wert für Preset, WRREC
EdgeInitiator Reference	Bool	Flanke Referenzinitiator
EnablePreset	Bool	Voreinstellen / Preset des Gebers freigeben
PresetManual	Bool	von Hand voreinstellen
REQ_Preset	Bool	Anforderung Preset, WRREC
Init_Top	Bool	Hub in oberer Position
Positions	Struct	Daten für „FC_HoistPositions“
HeightSetpoint	Int	Sollwert Höhe, mm
EnableHoist	Bool	Hubbewegung freigegeben
Height_at_end_RC	Int	Höhe am Ende der Strecke für stationäre Fernbedienung

8.2.6 DB_HoistPositions (DB72)**Inhalt**

Der Datenbaustein „DB_HoistPositions“ enthält die definierten Hubfreigabebereiche und die Soll-Höhen.

Diese Daten sind in allen Fahrzeugsteuerungen einer Anlage gleich. Sie werden über die quitierte Datenkommunikation zentral aktualisiert. Die quitierte Datenkommunikation ist in der Funktionsbibliothek „EMS400S_Communication“ enthalten, siehe Kapitel [3.2.2](#) FB_DataTransferEMS (FB30).

In der folgenden Tabelle sind die Daten des DBs aufgelistet.

Tabelle 8-12

Name	Datentyp	Beschreibung
HoistDefinition	Array [0..10] of Struct	Hubtabelle
HoistDefinition[n]		
BarcodeStart	DInt	Start der Hubstrecke, mm
BarcodeEnd	DInt	Ende der Hubstrecke, mm
Height	Int	Sollwert Höhe, mm

9 Arbeiten mit den Funktionsbibliotheken

Dieses Kapitel enthält Anleitungen zur Integration der Funktionsbibliotheken in Ihr TIA STEP 7-Projekt.

9.1 Integration der Funktionsbibliotheken in TIA STEP 7

Folgend sind die Schritte aufgeführt, um die Funktionsbibliothek „EMS400S_Segment“ in Ihr TIA STEP 7-Projekt zu integrieren. Anschließend können Sie die Bausteine der Funktionsbibliothek nutzen.

Auf die gleiche Weise integrieren Sie auch die Funktionsbibliotheken:

- EMS400S_Communication
- EMS400S_Display_RemoteControl
- EMS400S_Drive
- EMS400S_Errors
- EMS400S_General
- EMS400S_Hoist

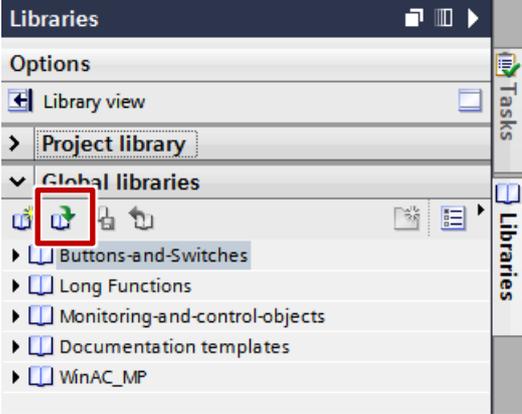
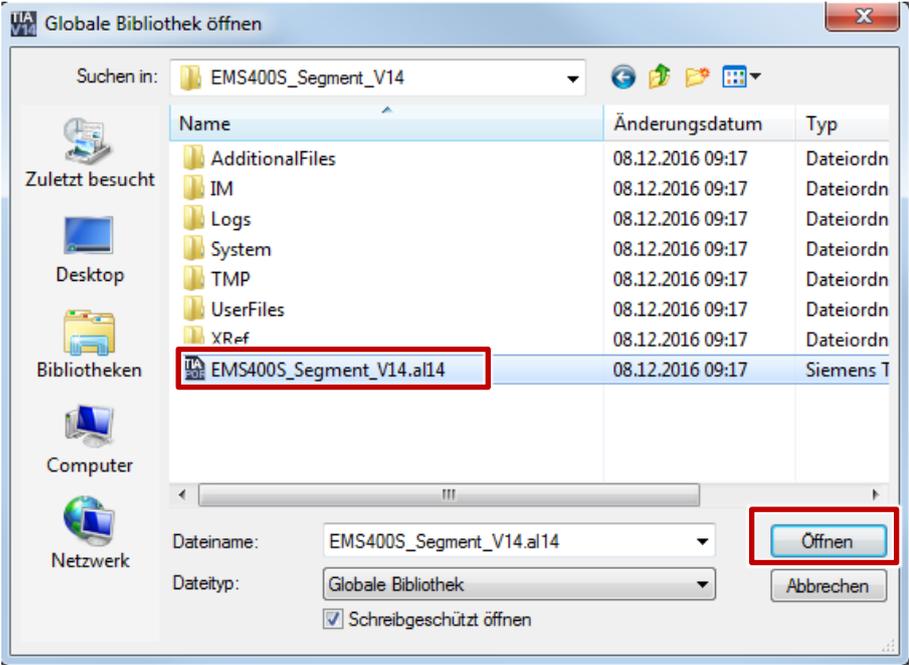
Hinweis

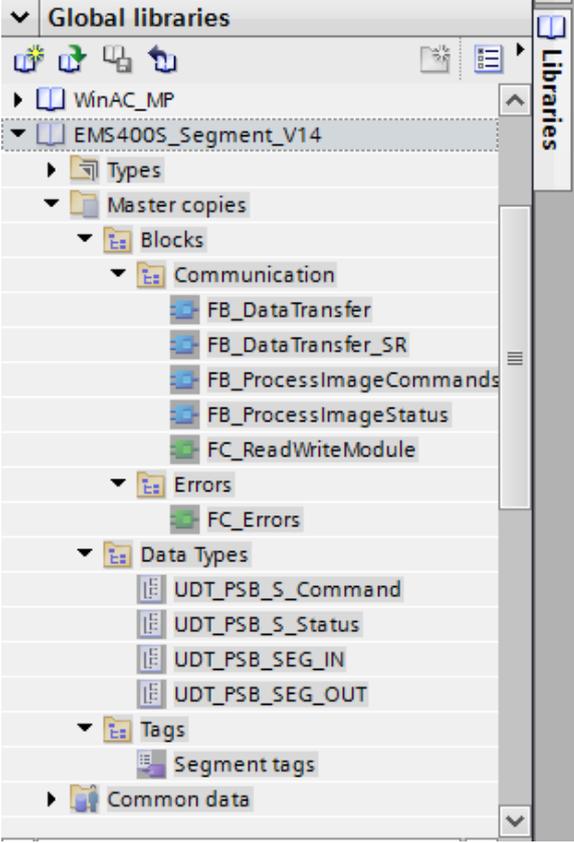
Das Vorhandensein eines TIA STEP 7-Projektes wird im Folgenden vorausgesetzt.

Inhalt

Die folgende Tabelle listet die Schritte zur Integration einer Funktionsbibliothek auf, am Beispiel der Funktionsbibliothek „EMS400S_Segment“.

Tabelle 9-1

Nr.	Aktion
1.	Entpacken Sie die Datei „EMS400S_Segment_V14“, die Sie von der HTML-Seite geladen haben, in einen beliebigen Ordner mit Schreib- und Lese-Rechten auf Ihrem lokalen Rechner.
2.	Öffnen Sie ihr bereits bestehendes TIA STEP 7-Projekt.
3.	<p>Klicken Sie in der Palette „Globale Bibliotheken“ (Global libraries) in der Funktionsleiste auf „Globale Bibliothek öffnen“ („Open global library“), oder wählen Sie im Menü „Optionen“ den Befehl „Globale Bibliotheken > Bibliothek öffnen“ („Options“ > „Global libraries“ > „Open library...“).</p>  <p>Der Dialog „Globale Bibliothek öffnen“ („Open global library“) öffnet sich.</p>
4.	<p>Selektieren Sie die Funktionsbibliothek „EMS400S_Segment_V14.al14“ und öffnen Sie die Datei.</p> 

Nr.	Aktion
5.	<p>Im Ordner „Kopiervorlagen“ („Master Copies“) befinden sich die Ordner</p> <ul style="list-style-type: none"> - Blocks, - Data Types und - Tags. <p>Aus diesen Ordnern können Sie die EMS400S Bibliotheksbausteine, die Datentypen und die Variablen-tabelle in Ihr Projekt kopieren.</p> 

Integrieren Sie die anderen Funktionsbibliotheken auf die gleiche Weise:

- EMS400S_Communication
- EMS400S_Display_RemoteControl
- EMS400S_Drive
- EMS400S_Errors
- EMS400S_General
- EMS400S_Hoist

9.2 Aufruf der Funktionsbibliotheksbausteine „EMS400S_Segment“ im TIA STEP 7 Programm

Hier sind die Schritte aufgeführt, um die Bausteine der Funktionsbibliothek „EMS400S_Segment“ in ein TIA STEP 7-Projekt einer S7-1200 Anlagensegmentsteuerung zu integrieren.

Hinweis Im Folgenden wird vorausgesetzt, dass Sie die Hardware einer Anlagensegmentsteuerung projektiert haben, siehe [3](#).

Ebenso wird vorausgesetzt, dass Sie die Systemmerker und die Taktmerker in den Eigenschaften der S7-1214 aktiviert haben.

Ebenso wird vorausgesetzt, dass Sie die Basis-Bibliothek EMS400S in Ihr TIA STEP 7-Projekt integriert haben, siehe [4](#).

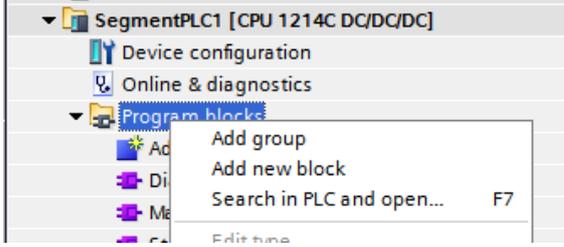
Hinweis Die Unterteilung der Programmbausteine in Gruppen ist prinzipiell nicht notwendig. Sie erleichtert aber die Übersicht im Programmbausteinordner.

Wir empfehlen für die Bausteine der Basis-Bibliothek EMS400S eine Gruppe „00_PSB“ einzufügen.

Variablentabelle, Datentypen und Bausteine kopieren

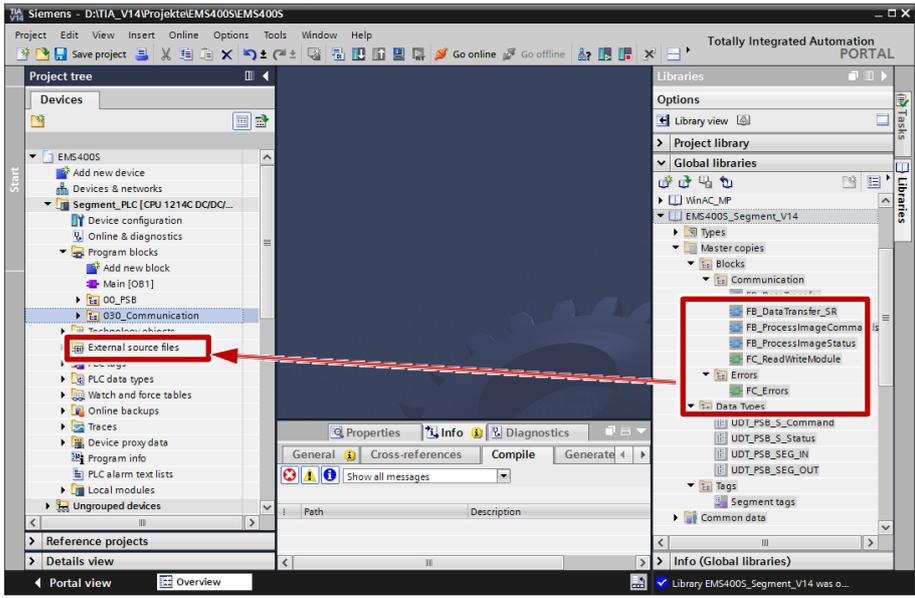
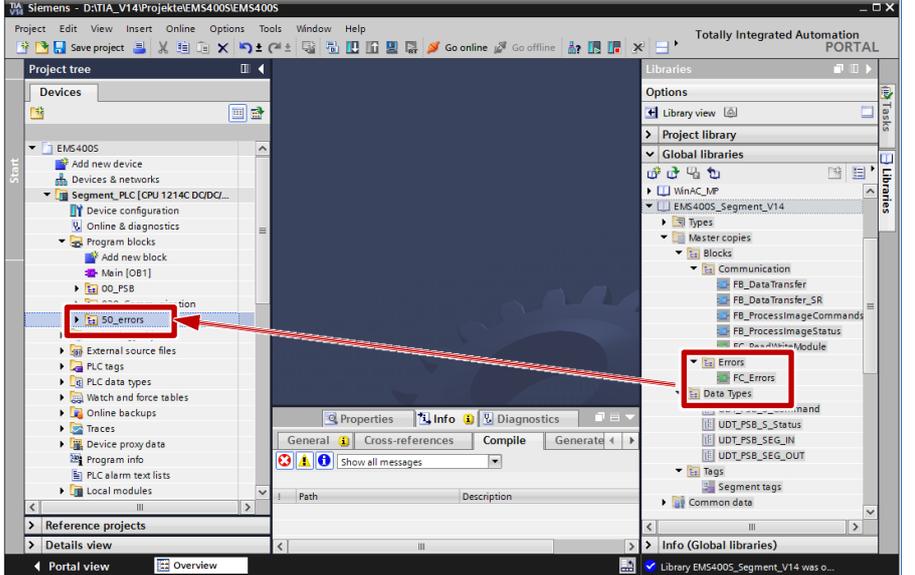
Die folgende Tabelle listet die Schritte zum Einfügen der Objekte der Funktionsbibliothek in die Anlagensegmentsteuerung auf.

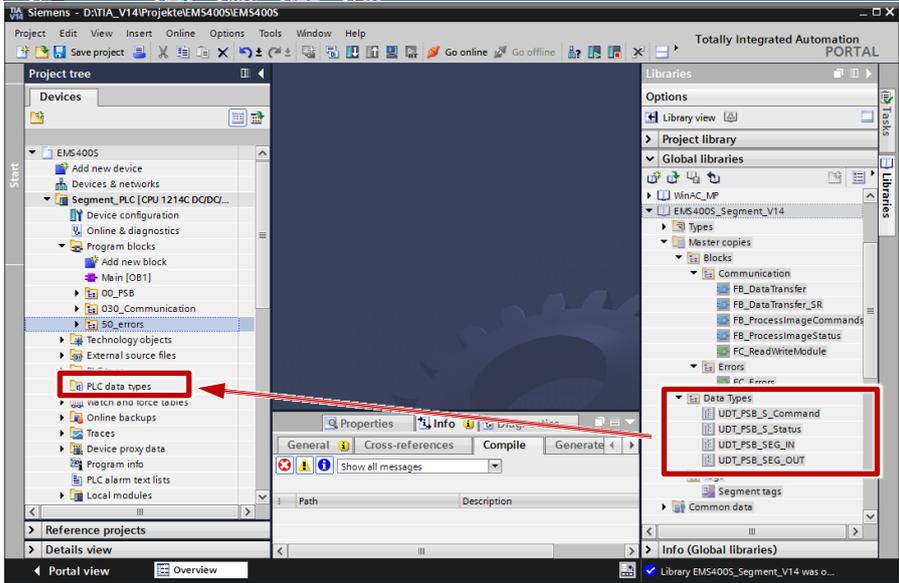
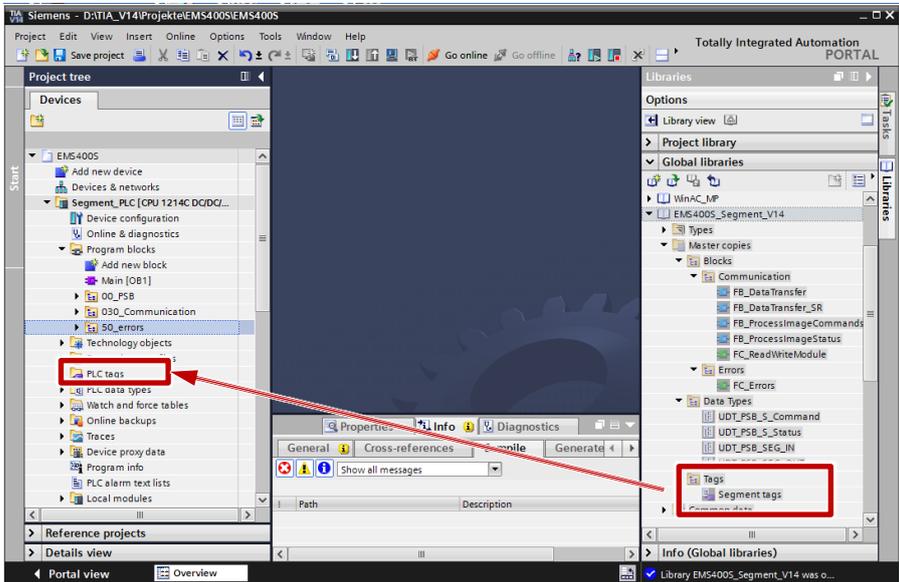
Tabelle 9-2

Nr	Aktion
1.	Fügen Sie dem Ordner „Programmbausteine“ eine neue Gruppe hinzu („Add group“) 
2.	Benennen Sie die Gruppe um, z.B. in „30_Communication“. Damit können Sie später sehen, dass sich hier die Bausteine für Kommunikations-Funktionen befinden, im Nummernband „30“.

9 Arbeiten mit den Funktionsbibliotheken

9.2 Aufruf der Funktionsbibliotheksbausteine „EMS400S_Segment“ im TIA STEP 7 Programm

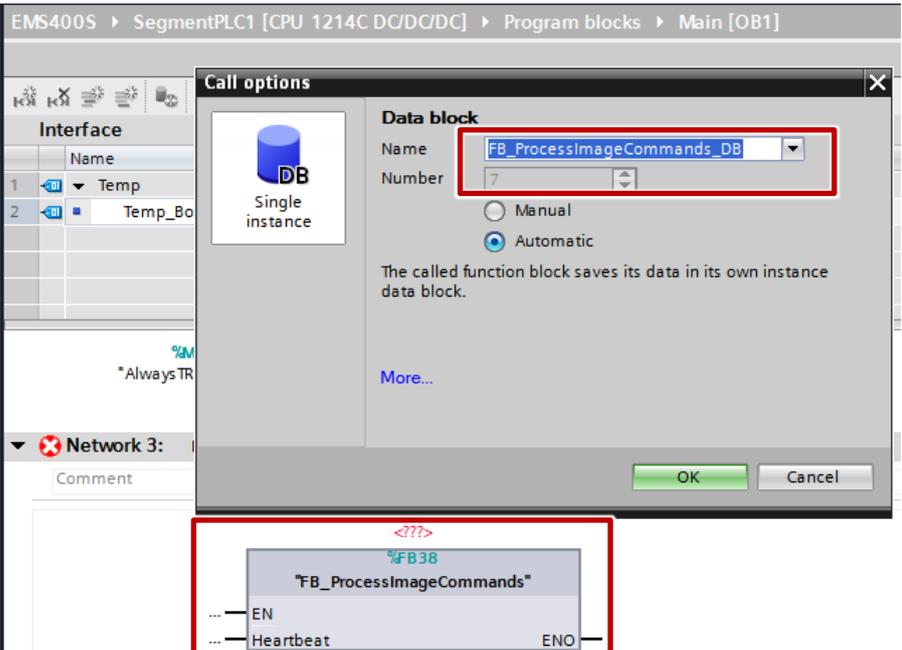
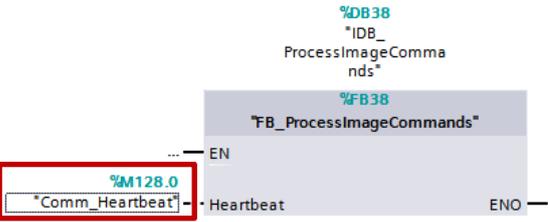
Nr	Aktion
3.	<p>Die Kommunikations Bausteine der Funktionsbibliothek „EMS400S_Segment“ befinden sich unter „Kopiervorlagen“ > „Blocks“ > „Communication“ („Master copies“ > „Blocks“ > „Communication“).</p> <p>Kopieren Sie die Bausteine per Drag&Drop in die Gruppe „30_Communication“ im Ordner „Programmbausteine“ („Program blocks“).</p> <p>Kopieren Sie „FB_DataTransfer_SR“ nur, wenn Sie Daten von der Fahrzeugsteuerung empfangen wollen.</p> 
4.	<p>Fügen Sie den Ordner „Programmbausteine“ eine neue Gruppe hinzu („Add group“)</p> <p>Benennen Sie die Gruppe um, z.B. in „50_Errors“.</p> <p>Damit können Sie später sehen, dass sich hier die Bausteine für Störungsmeldungs-Funktionen befinden, im Nummernband „50“</p>
5.	<p>Der Störungsmeldungs-Baustein der Funktionsbibliothek „EMS400S_Segment“ befindet sich unter „Kopiervorlagen“ > „Blocks“ > „Errors“ („Master copies“ > „Blocks“ > „Errors“).</p> <p>Kopieren Sie den Baustein per Drag&Drop in die Gruppe „50_Errors“ im Ordner „Programmbausteine“ („Program blocks“).</p> 

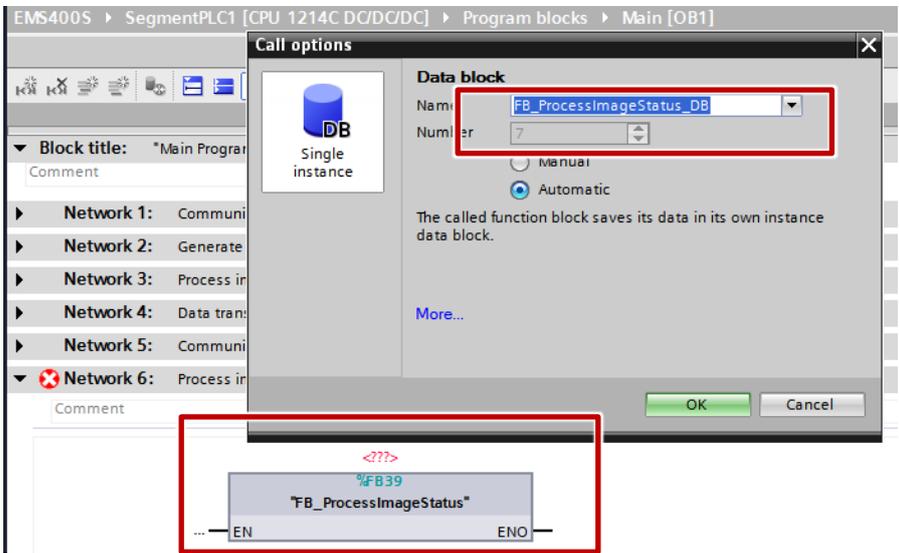
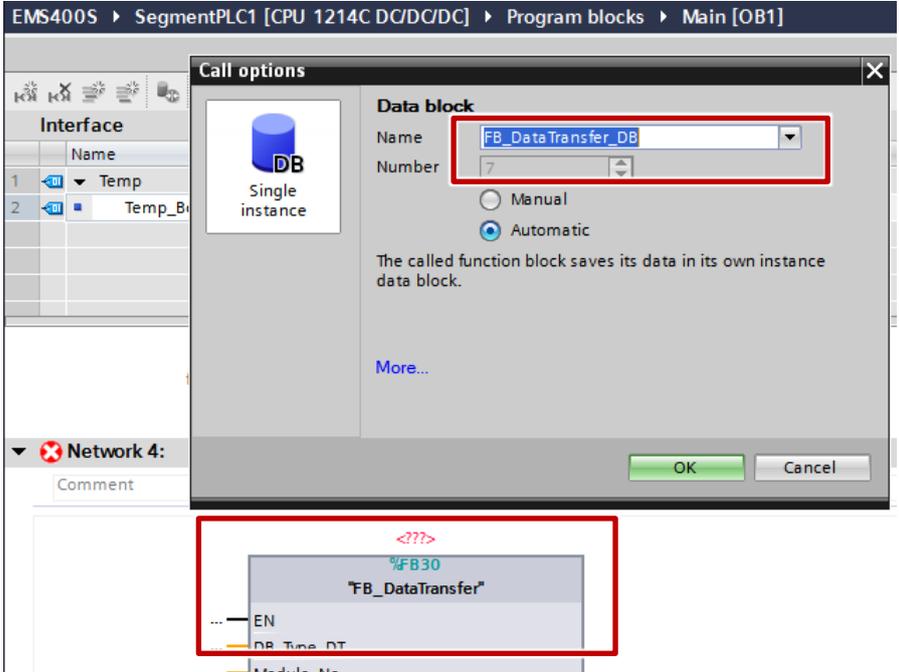
Nr	Aktion
6.	<p>Die Datentypen der Funktionsbibliothek „EMS400S_Segment“ befinden sich unter „Kopiervorlagen“ > „Data Types“ („Master copies“ > „Data Types“). Kopieren Sie alle Bausteine per Drag&Drop in den Ordner „PLC-Datentypen“ („PLC data types“).</p> 
7.	<p>Die Variablen-tabelle „Segment tags“ der Funktionsbibliothek „EMS400S_Segment“ befindet sich unter „Kopiervorlagen“ > „Tags“ („Master copies“ > „Tags“). Kopieren Sie die Variablen-tabelle per Drag&Drop in den Ordner „PLC-Variable“ („PLC tags“).</p> 

Zyklisches Programm

Die folgende Tabelle listet die Schritte zum Integrieren der Bausteine der Funktionsbibliothek in die Anlagensegmentsteuerung auf.

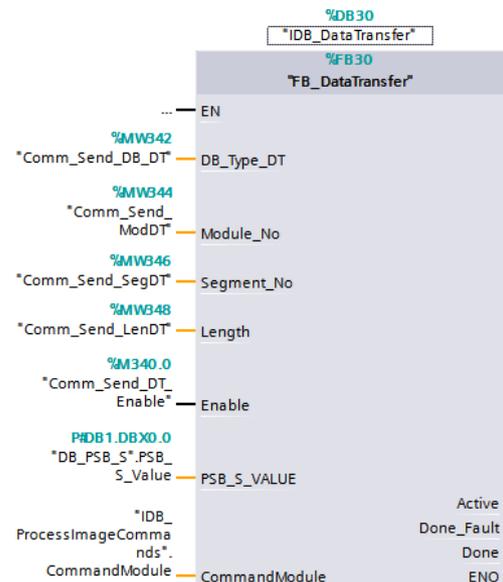
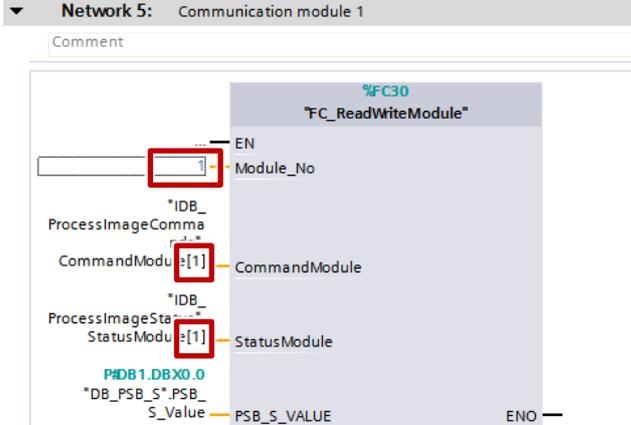
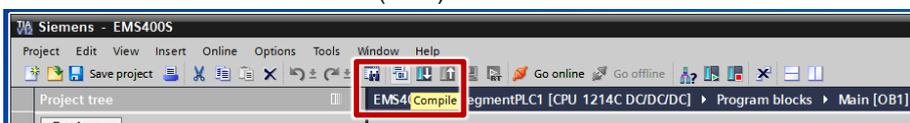
Tabelle 9-3

Nr.	Aktion
1.	Öffnen Sie den zyklischen OB1 (Main).
2.	<p>Rufen Sie am Anfang des OB1 den Baustein „FB_ProcessImageCommands“ auf und weisen Sie ihm einen InstanzDB zu.</p> 
3.	<p>Geben Sie z.B. die Bezeichnung „IDB_ProcessImageCommands“ ein. Übernehmen Sie die vorgeschlagene automatische Nummer, oder weisen Sie manuell eine Datenbausteinnummer zu, z.B. 38, um im Nummernband der Kommunikationsbausteine zu bleiben.</p>
4.	<p>Weisen Sie den Parameter „Heartbeat“ eine Variable zu.</p> 

Nr.	Aktion
5.	<p>Rufen Sie am Ende des OB1 den Baustein „FB_ProcessImageStatus“ auf und weisen Sie ihm einen InstanzDB zu.</p> 
6.	<p>Geben Sie z.B. die Bezeichnung „IDB_ProcessImageStatus“ ein. Übernehmen Sie die vorgeschlagene automatische Nummer, oder weisen Sie manuell eine Datenbausteinnummer zu, z.B. 39, um im Nummernband der Kommunikationsbausteine zu bleiben.</p>
7.	<p>Fügen Sie nach dem Aufruf des „FB_ProcessImageCommands“ ein Netzwerk ein. Rufen Sie den Baustein „FB_DataTransfer“ oder „FB_DataTransfer_SR“ auf und weisen Sie ihm einen InstanzDB zu.</p> 
8.	<p>Geben Sie z.B. die Bezeichnung „IDB_DataTransfer“ ein. Übernehmen Sie die vorgeschlagene automatische Nummer, oder weisen Sie manuell eine Datenbausteinnummer zu, z.B. 30, um im Nummernband der Kommunikationsbausteine zu bleiben.</p>

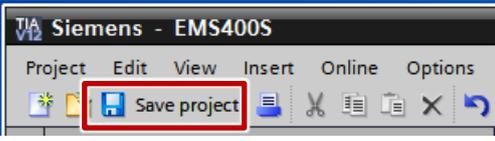
9 Arbeiten mit den Funktionsbibliotheken

9.2 Aufruf der Funktionsbibliotheksbausteine „EMS400S_Segment“ im TIA STEP 7 Programm

Nr.	Aktion
9.	<p>Weisen Sie die Parameter zu, z.B.:</p> 
10.	<p>Fügen Sie anschließend ein Netzwerk ein und rufen Sie den Baustein „FC_ReadWriteModule“ auf.</p>
11.	<p>Weisen Sie die Parameter zu, z.B. für PSB-S-Modul 1:</p> 
12.	<p>Wiederholen Sie die Schritte 10 und 11 für jedes PSB-S-Modul und passen Sie die Modulnummer in den Parametern an. In Schritt 11 ist jeweils an den Parametern die Nummer 1 markiert; Sie müssen an den jeweiligen Aufrufen die entsprechende Nummer eingeben.</p>
13.	<p>Fügen Sie am Ende des OB1 ein Netzwerk ein. Rufen Sie den Baustein „FC_Errors“ auf.</p>
14.	<p>Rufen Sie auch die Bausteine der Basis-Bibliothek EMS_400S (siehe 4) und Ihre individuellen, anwenderdefinierten Bausteine für die Anlagensegmentsteuerung auf (nicht Teil dieser Beschreibung).</p>
15.	<p>Verschieben Sie die InstanzDBs in die Gruppe „30_Communication“.</p>
16.	<p>Übersetzen Sie den Baustein OB1 (Main).</p> 

9 Arbeiten mit den Funktionsbibliotheken

9.2 Aufruf der Funktionsbibliotheksbausteine „EMS400S_Segment“ im TIA STEP 7 Programm

Nr.	Aktion
17.	Speichern Sie das TIA STEP 7-Projekt. 

Hinweis Der Datenbaustein „DB_PSB_S“ enthält die Übergabestrukturen der Basis-Bibliothek EMS400, siehe [4](#)

9.3 Aufruf der Funktionsbibliotheksbausteine „EMS400S_Communication“ im TIA STEP 7 Programm

Hier sind die Schritte aufgeführt, um die Bausteine der Funktionsbibliothek „EMS400S_Communication“ in ein TIA STEP 7-Projekt einer S7-1200 Fahrzeugsteuerung zu integrieren.

Hinweis Im Folgenden wird vorausgesetzt, dass Sie die Hardware einer Fahrzeugsteuerung projektiert haben, siehe [3](#).

Ebenso wird vorausgesetzt, dass Sie die Systemmerker und die Taktmerker in den Eigenschaften der S7-1212 aktiviert haben.

Es wird vorausgesetzt, dass Sie die Basis-Bibliothek EMS400S in Ihr TIA STEP 7-Projekt integriert haben, siehe [4](#).

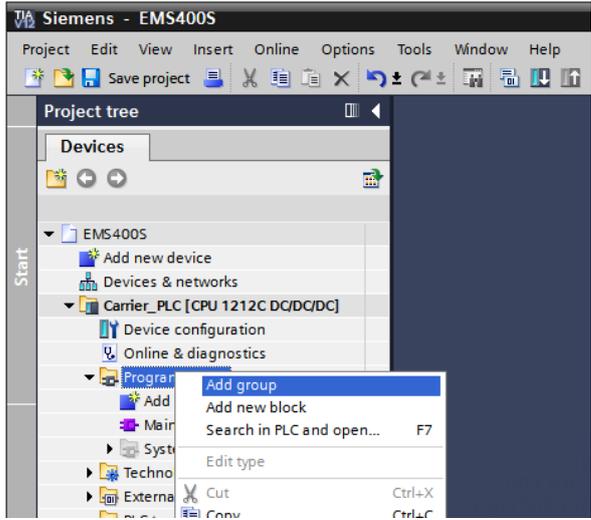
Hinweis Die Unterteilung der Programmbausteine in Gruppen ist nicht notwendig. Sie erleichtert die Übersicht im Programmbausteinordner.

Wir empfehlen, für die Bausteine der Basis-Bibliothek EMS400S eine Gruppe „00_PSB“ einzufügen.

Bausteine kopieren

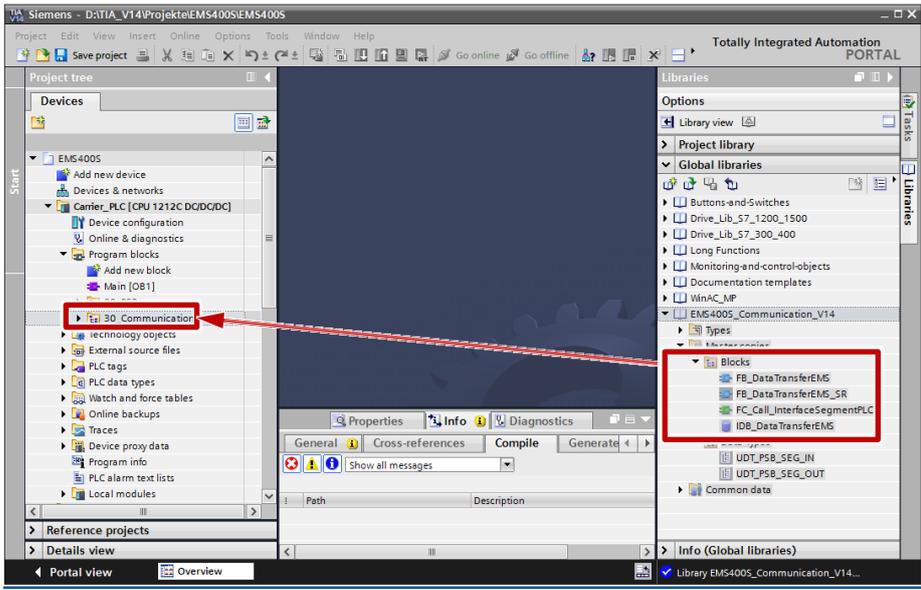
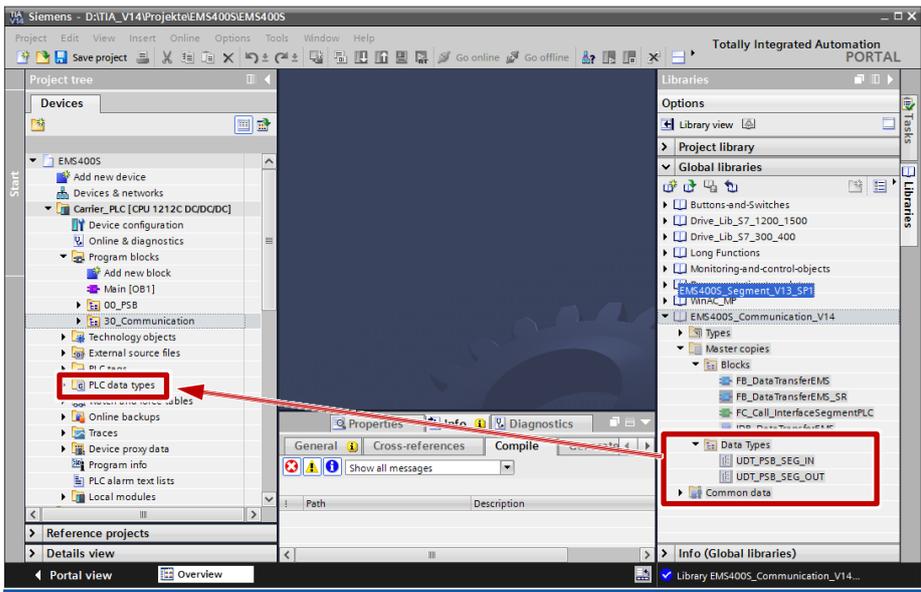
Die folgende Tabelle listet die Schritte zum Einfügen der Objekte der Funktionsbibliothek in die Fahrzeugsteuerung („Carrier PLC“) auf.

Tabelle 9-4

Nr	Aktion
1.	Fügen Sie dem Ordner „Programmbausteine“ eine neue Gruppe hinzu (Add group). 
2.	Benennen Sie die Gruppe um, z.B. in „30_Communication“. Damit können Sie später sehen, dass sich hier die Bausteine für Kommunikations-

9 Arbeiten mit den Funktionsbibliotheken

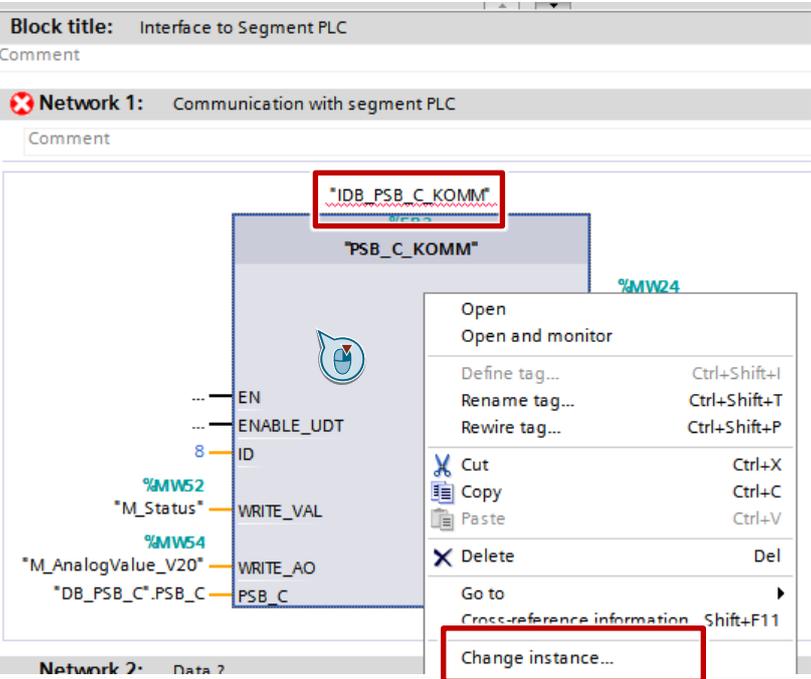
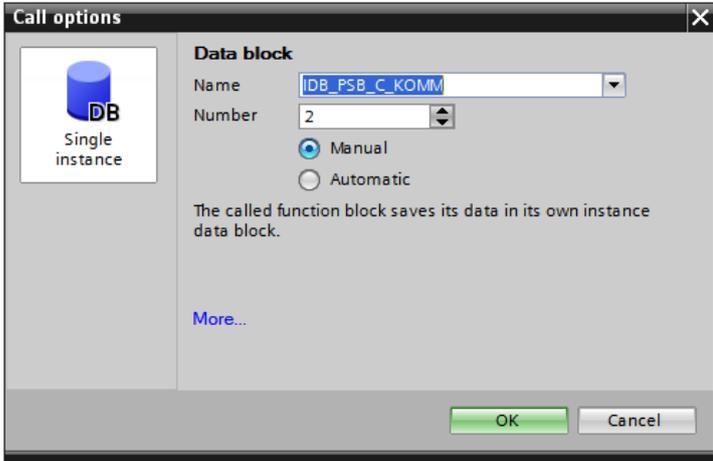
9.3 Aufruf der Funktionsbibliotheksbausteine „EMS400S_Communication“ im TIA STEP 7 Programm

Nr.	Aktion
3.	<p>Funktionen befinden, im Nummernband „30“</p> <p>Die Bausteine der Funktionsbibliothek „EMS400S_Communication“ befinden sich unter „Kopiervorlagen“ > „Blocks“ („Master copies“ > „Blocks“).</p> <p>Kopieren Sie die Bausteine per Drag&Drop in die Gruppe „30_Communication“ im Ordner „Programmbausteine“ („Program blocks“).</p> <p>Kopieren Sie „FB_DataTransferEMS_SR“ nur, wenn Sie Daten zur Anlagensegmentsteuerung senden wollen.</p> 
4.	<p>Die Datentypen der Funktionsbibliothek „EMS400S_Communication“ befinden sich unter „Kopiervorlagen“ > „Data Types“ („Master copies“ > „Data Types“).</p> <p>Kopieren Sie alle Bausteine per Drag&Drop in den Ordner „PLC-Datentypen“ („PLC data types“).</p> 

Zyklisches Programm

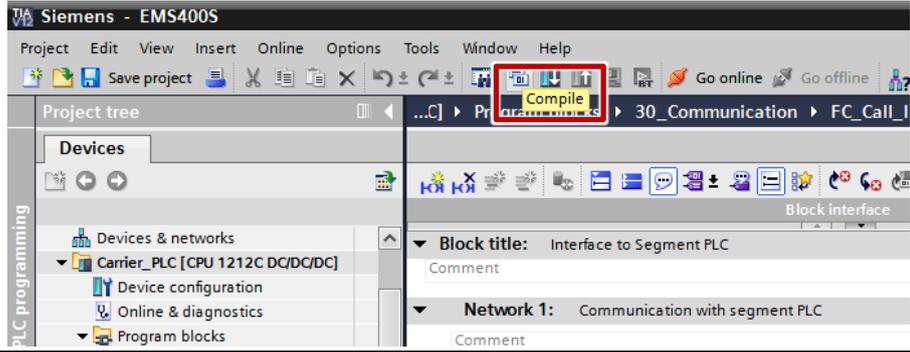
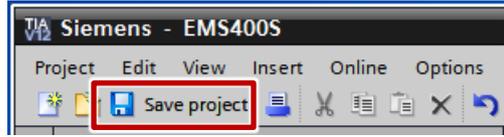
Die folgende Tabelle listet die Schritte zum Integrieren der Bausteine der Funktionsbibliothek in die Fahrzeugsteuerung auf.

Tabelle 9-5

Nr.	Aktion
1.	Öffnen Sie den zyklischen OB1. Rufen Sie den Baustein „FC_Call_InterfaceSegmentPLC“ auf. Rufen Sie auch die Bausteine aus den anderen EMS-Funktionsbibliotheken und Ihre individuellen, anwenderdefinierten Bausteine für die Fahrzeugsteuerung auf (nicht Teil dieser Beschreibung).
2.	Öffnen Sie den Baustein „FC_Call_InterfaceSegmentPLC“.
3.	Wählen Sie den Baustein „PSB_C_KOMM“ in Netzwerk 1 an und weisen Sie ihm einen InstanzDB zu mit „Instanz ändern“ („Change instance“). 
4.	Geben Sie die Bezeichnung „IDB_PSB_C_KOMM“ ein. Übernehmen Sie die vorgeschlagene automatische Nummer, oder weisen Sie manuell eine Datenbausteinnummer zu. 

9 Arbeiten mit den Funktionsbibliotheken

9.3 Aufruf der Funktionsbibliotheksbausteine „EMS400S_Communication“ im TIA STEP 7 Programm

Nr.	Aktion
5.	Verschieben Sie den InstanzDB in die Gruppe „00_PSB“
6.	Wenn Sie Daten zur Anlagensegmentsteuerung senden wollen, dann öffnen Sie den Baustein „FC_Call_InterfaceSegmentPLC“. Rufen Sie in Netzwerk 12 den Baustein „FB_DataTransferEMS_SR“ anstelle des Bausteins „FB_DataTransferEMS“ auf. Weisen Sie einen neuen InstanzDB zu.
7.	Übersetzen Sie den Baustein „FC_Call_InterfaceSegmentPLC“. 
8.	Speichern Sie das TIA STEP 7-Projekt. 

Hinweis

„FC_Call_InterfaceSegmentPLC“ verweist auf

- „IDB_RemoteControl“ aus der Funktionsbibliothek
- „EMS400S_Display_RemoteControl“ und
- „IDB_ErrorsAll“ aus der Funktionsbibliothek „EMS400S_Errors“.

Falls diese Bausteine noch nicht im TIA STEP 7-Projekt enthalten sind, erhalten Sie Fehlermeldungen beim Übersetzen.

„FC_Call_InterfaceSegmentPLC“ wird ohne Fehlermeldungen übersetzt, sobald die Funktionsbibliotheken „EMS400S_Display_RemoteControl“ und „EMS400S_Errors“ in Ihr TIA STEP 7-Projekt integriert sind.

9.4 Aufruf der Funktionsbibliotheksbausteine „EMS400S_Display_RemoteControl“ im TIA STEP 7 Programm

Hier sind die Schritte aufgeführt, um die Bausteine der Funktionsbibliothek EMS400S_Display_RemoteControl in ein TIA STEP 7-Projekt einer S7-1200 Fahrzeugsteuerung zu integrieren.

Hinweis

Im Folgenden wird vorausgesetzt, dass Sie die Hardware einer Fahrzeugsteuerung projektiert haben, siehe [31](#).

Ebenso wird vorausgesetzt, dass Sie die Systemmerker und die Taktmerker in den Eigenschaften der S7-1212 aktiviert haben.

Es wird vorausgesetzt, dass Sie die Basis-Bibliothek EMS400S in Ihr TIA STEP 7-Projekt integriert haben, siehe [4](#).

Ebenso wird vorausgesetzt, dass Sie die Funktionsbibliotheken

- „EMS400S_General“ und
- „EMS400_Drive“ in Ihr Projekt integriert haben.

„EMS400_Hoist“ müssen Sie nur integriert haben, wenn Ihre Fahrzeugsteuerung einen Hubantrieb enthält.

Hinweis

Die Unterteilung der Programmteile in Gruppen ist nicht notwendig. Sie erleichtert die Übersicht im Programmteilordner.

Wir empfehlen, für die Bausteine der Basis-Bibliothek EMS400S eine Gruppe „00_PSB“ einzufügen.

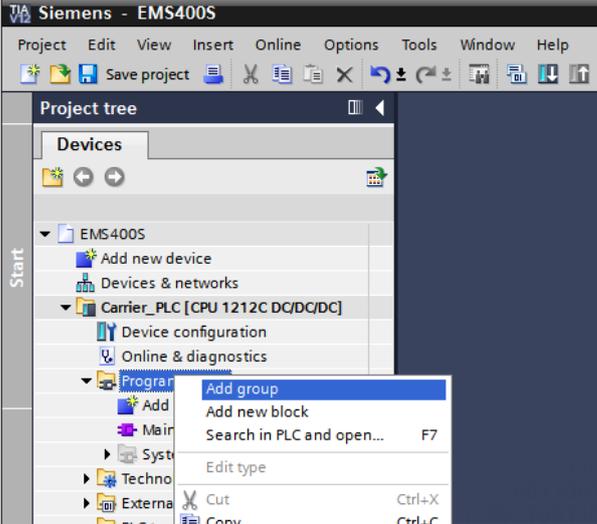
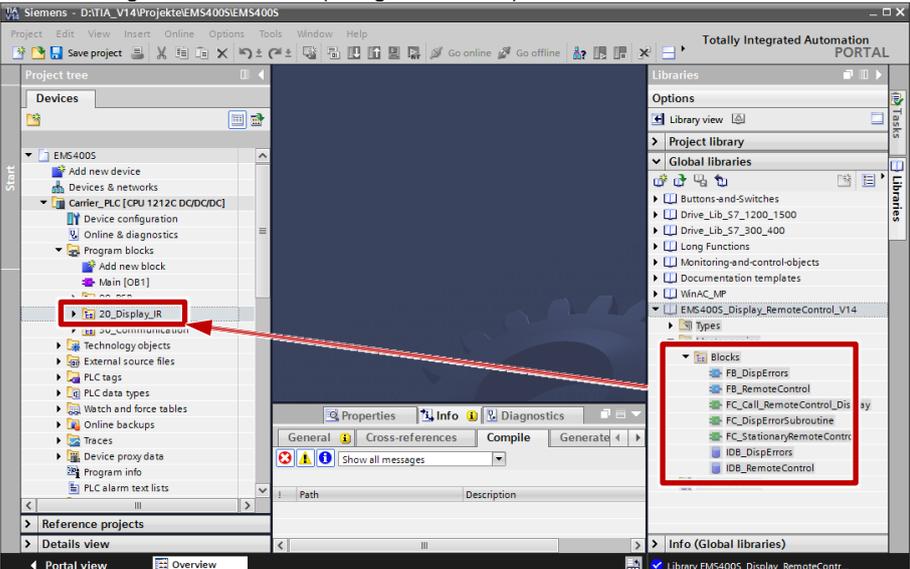
9 Arbeiten mit den Funktionsbibliotheken

9.4 Aufruf der Funktionsbibliotheksbausteine „EMS400S_Display_RemoteControl“ im TIA STEP 7 Programm

Bausteine kopieren

Die folgende Tabelle listet die Schritte zum Einfügen der Objekte der Funktionsbibliothek in die Fahrzeugsteuerung („Carrier PLC“) auf.

Tabelle 9-6

Nr	Aktion
1.	<p>Fügen Sie dem Ordner „Programmbausteine“ eine neue Gruppe hinzu (Add group)</p>  <p>The screenshot shows the Siemens TIA Portal interface. The 'Project tree' on the left is expanded to 'Carrier_PLC [CPU 1212C DC/DC/DC]' > 'Program blocks'. A right-click context menu is open, and the 'Add group' option is highlighted. Other options include 'Add new block', 'Search in PLC and open... F7', and 'Edit type'.</p>
2.	<p>Benennen Sie die Gruppe um, z.B. in „20_Display_IR“. Damit können Sie später sehen, dass sich hier die Bausteine für Display- und Fernbedienungs-Funktionen befinden, im Nummernband „20“</p>
3.	<p>Die Bausteine der Funktionsbibliothek „EMS400S_Display_RemoteControl“ befinden sich unter „Kopiervorlagen“ > „Blocks“ („Master copies“ > „Blocks“). Kopieren Sie alle Bausteine per Drag&Drop in die Gruppe „20_Display_IR“ im Ordner „Programmbausteine“ („Program blocks“).</p>  <p>The screenshot shows the Siemens TIA Portal interface. The 'Project tree' on the left is expanded to 'Carrier_PLC [CPU 1212C DC/DC/DC]' > 'Program blocks' > '20_Display_IR'. A red box highlights the '20_Display_IR' group. The 'Libraries' pane on the right shows the 'Global libraries' section expanded to 'EMS400S_Display_RemoteControl_V14'. Under the 'Blocks' sub-section, several blocks are listed, including 'FB_DispErrors', 'FB_RemoteControl', 'FC_Call_RemoteControl_Dis', 'FC_DispErrorSubroutine', 'FC_StationaryRemoteContr', 'IDB_DispErrors', and 'IDB_RemoteControl'. A red box highlights these blocks, and a red arrow points from this box to the '20_Display_IR' group in the Project tree.</p>

9 Arbeiten mit den Funktionsbibliotheken

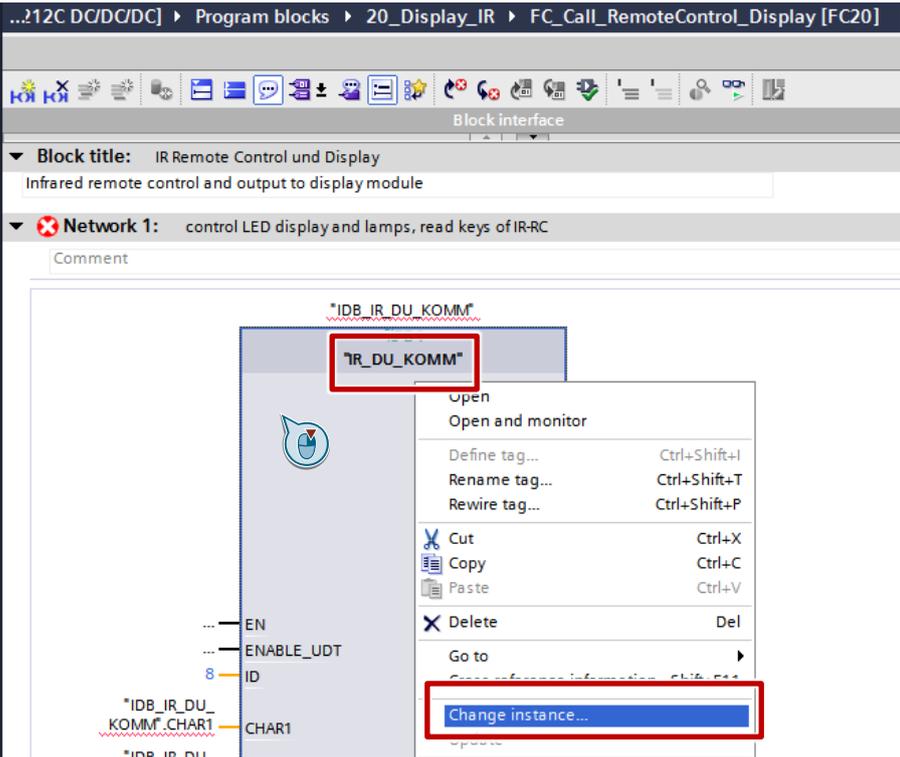
9.4 Aufruf der Funktionsbibliotheksbausteine „EMS400S_Display_RemoteControl“ im TIA STEP 7 Programm

Hinweis Der Baustein „FC_StationaryRemoteControl“ wird nur benötigt, wenn eine stationäre Fernbedienung vorhanden ist.

Zyklisches Programm

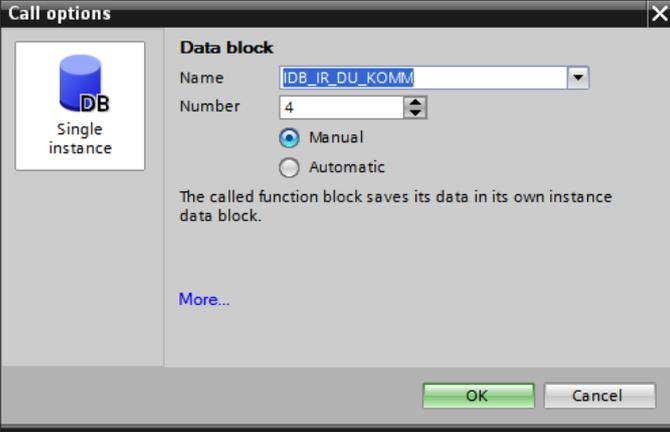
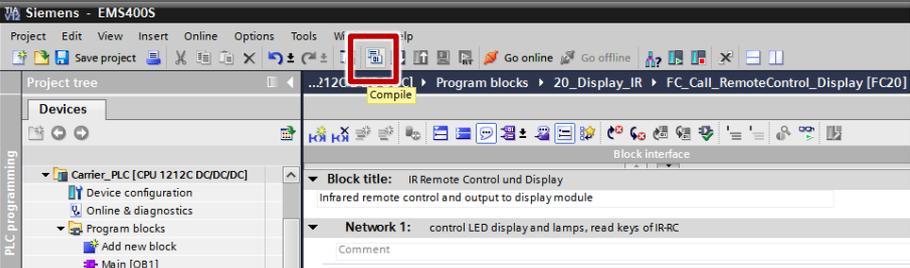
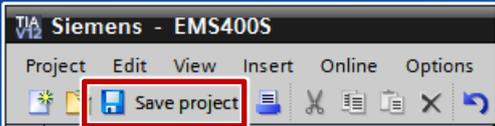
Die folgende Tabelle listet die Schritte zum Integrieren der Bausteine der Funktionsbibliothek in die Fahrzeugsteuerung auf.

Tabelle 9-7

Nr	Aktion
1.	Öffnen Sie den zyklischen OB1. Rufen Sie den Baustein „FC_Call_RemoteControl_Display“ auf. Rufen Sie auch die Bausteine aus den anderen EMS-Funktionsbibliotheken und Ihre individuellen, anwenderdefinierten Bausteine für die Fahrzeugsteuerung auf (nicht Teil dieser Beschreibung).
2.	Öffnen Sie den Baustein „FC_Call_RemoteControl_Display“.
3.	Wählen Sie den Baustein „IR_DU_KOMM“ in Netzwerk 1 an und weisen Sie ihm einen InstanzDB zu mit „Instanz ändern“ (Change instance). 

9 Arbeiten mit den Funktionsbibliotheken

9.4 Aufruf der Funktionsbibliotheksbausteine „EMS400S_Display_RemoteControl“ im TIA STEP 7 Programm

Nr	Aktion
4.	<p>Geben Sie die Bezeichnung „IDB_IR_DU_KOMM“ ein. Übernehmen Sie die vorgeschlagene automatische Nummer, oder weisen Sie manuell eine Datenbausteinnummer zu.</p> 
5.	<p>Verschieben Sie den InstanzDB in die Gruppe „00_PSB“</p>
6.	<p>Übersetzen Sie den Baustein „FC_Call_RemoteControl_Display“.</p> 
7.	<p>Speichern Sie das TIA STEP 7-Projekt.</p> 

Hinweis Wenn Ihre Fahrzeugsteuerung keine stationäre Fernbedienung hat, können Sie den Aufruf des „FC_StationaryRemoteControl“ in „FC_Call_RemoteControl_Display“ löschen, siehe [Abbildung 4-3](#)

9.5 Aufruf der Funktionsbibliotheksbausteine „EMS400S_Drive“ im TIA STEP 7 Programm

Hier sind die Schritte aufgeführt, um die Bausteine der Funktionsbibliothek EMS400S_Drive in ein TIA STEP 7-Projekt einer S7-1200 Fahrzeugsteuerung zu integrieren.

Hinweis

Im Folgenden wird vorausgesetzt, dass Sie die Hardware einer Fahrzeugsteuerung projektiert haben, siehe [13](#).

Ebenso wird vorausgesetzt, dass Sie die Systemmerker und die Taktmerker in den Eigenschaften der S7-1212 aktiviert haben.

Es wird vorausgesetzt, dass Sie die Basis-Bibliothek EMS400S in Ihr TIA STEP 7-Projekt integriert haben, siehe [14](#).

Ebenso wird vorausgesetzt, dass Sie die Funktionsbibliothek „EMS400S_General“ in Ihr Projekt integriert haben.

Hinweis

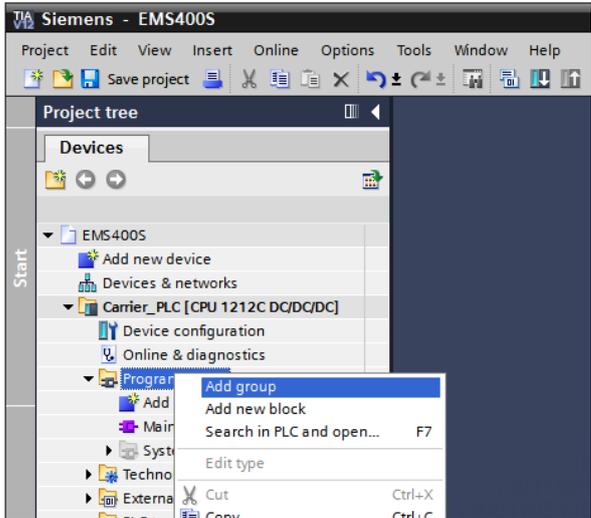
Die Unterteilung der Programmbausteine in Gruppen ist nicht notwendig. Sie erleichtert die Übersicht im Programmbausteinordner.

Wir empfehlen, für die Bausteine der Basis-Bibliothek EMS400S eine Gruppe „00_PSB“ einzufügen.

Bausteine kopieren

Die folgende Tabelle listet die Schritte zum Einfügen der Objekte der Funktionsbibliothek in die Fahrzeugsteuerung („Carrier PLC“) auf.

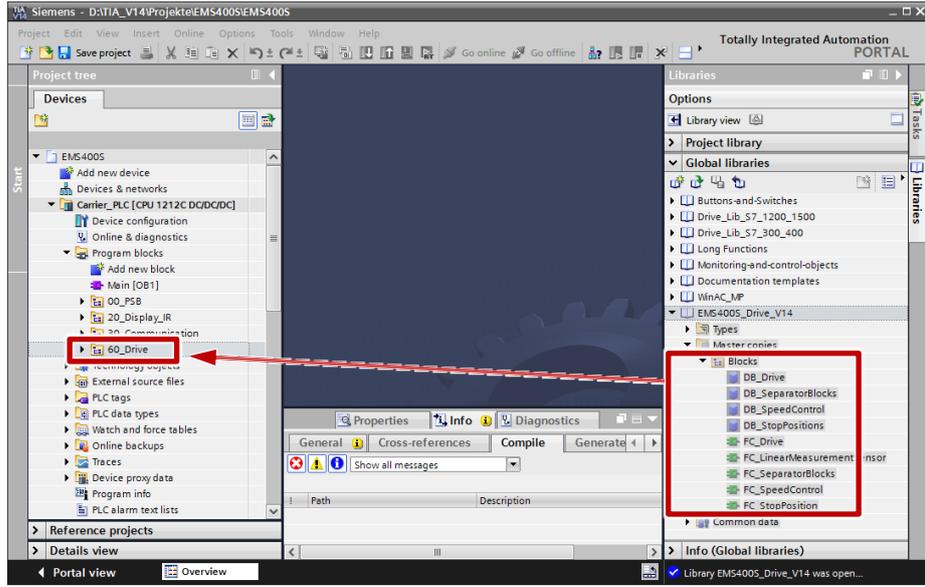
Tabelle 9-8

Nr	Aktion
1.	<p>Fügen Sie dem Ordner „Programmbausteine“ eine neue Gruppe hinzu (Add group)</p> 

9 Arbeiten mit den Funktionsbibliotheken

9.5 Aufruf der Funktionsbibliotheksbausteine „EMS400S_Drive“ im TIA STEP 7 Programm

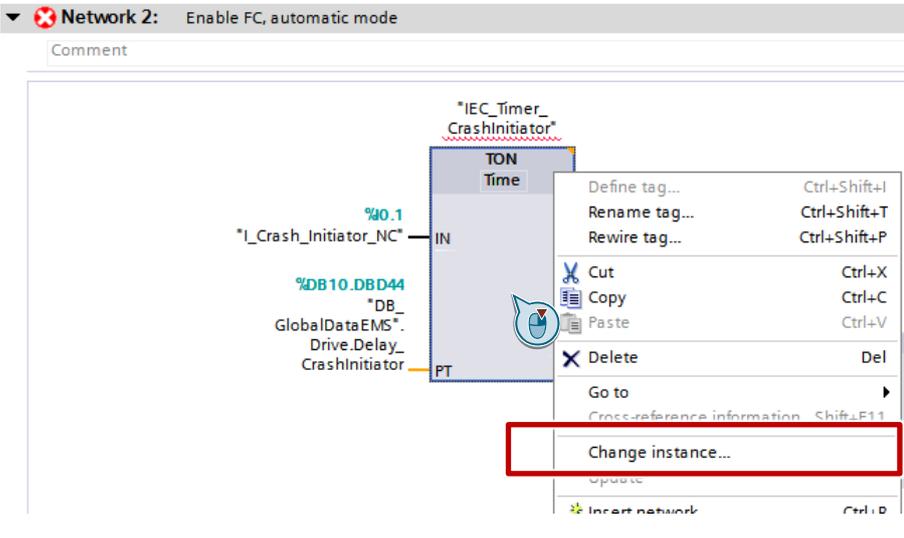
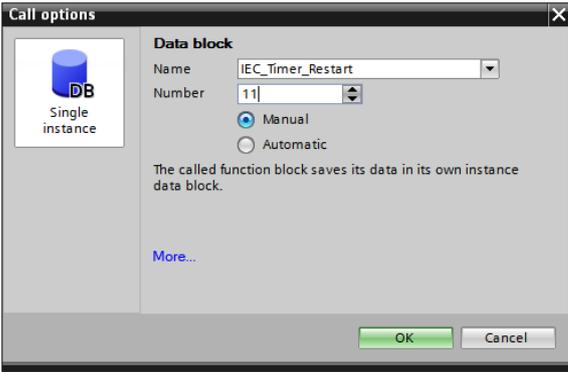
Nr.	Aktion
2.	Benennen Sie die Gruppe um, z.B. in „60_Drive“. Damit können Sie später sehen, dass sich hier die Bausteine für Fahrtriebsfunktionen befinden, im Nummernband „60“
3.	Die Bausteine der Funktionsbibliothek „EMS400S_Drive“ befinden sich unter „Kopiervorlagen“ > „Blocks“ („Master copies“ > „Blocks“). Kopieren Sie alle Bausteine per Drag&Drop in die Gruppe „60_Drive“ im Ordner „Programmbausteine“ („Program blocks“).



Zyklisches Programm

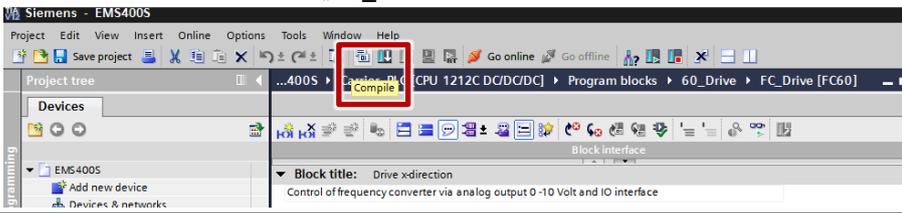
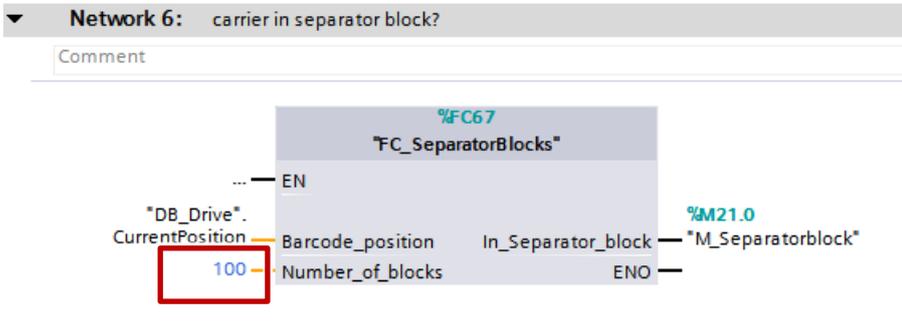
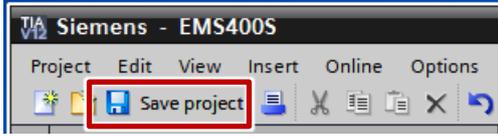
Die folgende Tabelle listet die Schritte zum Integrieren der Bausteine der Funktionsbibliothek in die Fahrzeugsteuerung auf.

Tabelle 9-9

Nr.	Aktion
1.	<p>Öffnen Sie den zyklischen OB1. Rufen Sie den Baustein „FC_Drive“ auf. Rufen Sie auch die Bausteine aus den anderen EMS-Funktionsbibliotheken und Ihre individuellen, anwenderdefinierten Bausteine für die Fahrzeugsteuerung auf (nicht Teil dieser Beschreibung).</p>
2.	<p>Öffnen Sie den Baustein „FC_Drive“. Wählen Sie den Timer in Netzwerk 2 an und weisen Sie dem Timer einen Instanz-Datenbaustein zu mit „Instanz ändern“ („Change instance“).</p> 
3.	<p>Übernehmen Sie die vorgeschlagene automatische Nummer, oder weisen Sie manuell eine Datenbausteinnummer, z.B. 65, um im Nummernband der Fahrtriebsbausteine zu bleiben.</p> 
4.	<p>Wählen Sie den Timer in Netzwerk 45 an und weisen Sie dem Timer einen Instanz-Datenbaustein zu mit „Instanz ändern“ („Change instance“). Übernehmen Sie die vorgeschlagene automatische Nummer, oder weisen Sie manuell eine Datenbausteinnummer, z.B. 66, um im Nummernband der Fahrtriebsbausteine zu bleiben.</p>

9 Arbeiten mit den Funktionsbibliotheken

9.5 Aufruf der Funktionsbibliotheksbausteine „EMS400S_Drive“ im TIA STEP 7 Programm

Nr.	Aktion
5.	<p>Übersetzen Sie den Baustein „FC_Drive“.</p> 
6.	<p>Wenn Ihre Anlage Isolationsblöcke enthält: Öffnen Sie den zyklischen OB1 (Main). Rufen Sie den Baustein „FC_SeparatorBlocks“ auf. Tragen Sie die Anzahl der Isolationsblöcke am Parameter „Number_of_blocks“ ein, und weisen Sie die Parameter zu.</p> <p>Network 6: carrier in separator block?</p> <p>Comment</p>  <p>Übersetzen Sie den Baustein OB1 (Main).</p>
7.	<p>Speichern Sie das TIA STEP 7-Projekt.</p> 

Hinweis

„FC_Drive“ verweist auf „DB_ErrorCodes“ aus der Funktionsbibliothek „EMS400S_Errors“. Falls „DB_ErrorCodes“ noch nicht im TIA STEP 7-Projekt enthalten ist, erhalten Sie Fehlermeldungen.

„FC_Drive“ wird ohne diese Fehlermeldungen übersetzt, sobald die Funktionsbibliothek „EMS400S_Errors“ in Ihr TIA STEP 7-Projekt integriert ist.

9.6 Aufruf der Funktionsbibliotheksbausteine „EMS400S_Errors“ im TIA STEP 7 Programm

Hier sind die Schritte aufgeführt, um die Bausteine der Funktionsbibliothek „EMS400S_Errors“ in ein TIA STEP 7-Projekt einer S7-1200 Fahrzeugsteuerung zu integrieren.

Hinweis

Im Folgenden wird vorausgesetzt, dass Sie die Hardware einer Fahrzeugsteuerung projektiert haben, siehe [31](#).

Ebenso wird vorausgesetzt, dass Sie die Systemmerker und die Taktmerker in den Eigenschaften der S7-1212 aktiviert haben.

Es wird vorausgesetzt, dass Sie die Basis-Bibliothek EMS400S in Ihr TIA STEP 7-Projekt integriert haben, siehe [41](#).

Ebenso wird vorausgesetzt, dass Sie die Funktionsbibliotheken

- „EMS400S_General“ und
- „EMS400_Drive“ in Ihr Projekt integriert haben.

„EMS400_Hoist“ müssen Sie nur integriert haben, wenn Ihre Fahrzeugsteuerung einen Hubantrieb enthält.

Hinweis

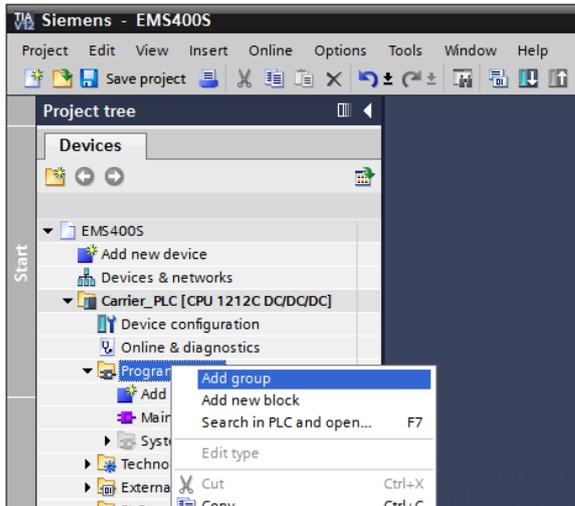
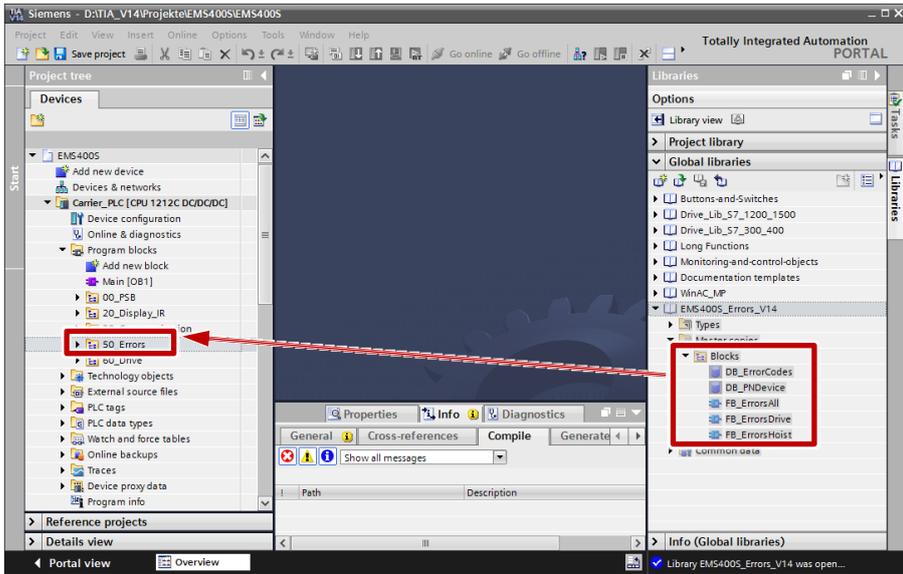
Die Unterteilung der Programmbausteine in Gruppen ist nicht notwendig. Sie erleichtert die Übersicht im Programmbausteinordner.

Wir empfehlen, für die Bausteine der Basis-Bibliothek EMS400S eine Gruppe „00_PSB“ einzufügen.

Variablen-tabelle und Bausteine kopieren

Die folgende Tabelle listet die Schritte zum Einfügen der Objekte der Funktionsbibliothek in die Fahrzeugsteuerung („Carrier PLC“) auf.

Tabelle 9-10

Nr.	Aktion
1.	<p>Fügen Sie dem Ordner „Programm-bausteine“ eine neue Gruppe hinzu („Add group“)</p> 
2.	<p>Benennen Sie die Gruppe um, z.B. in „50_Errors“. Damit können Sie später sehen, dass sich hier die Bausteine für Störungsmeldungs-Funktionen befinden, im Nummernband „50“</p>
3.	<p>Die Bausteine der Funktionsbibliothek „EMS400S_Errors“ befinden sich unter „Kopier-vorlagen“ > „Blocks“ („Master copies“ > „Blocks“). Kopieren Sie alle Bausteine per Drag&Drop in die Gruppe „50_Errors“ im Ordner „Programm-bausteine“ („Program blocks“).</p> 

© Siemens AG Copyright-2014 All rights reserved

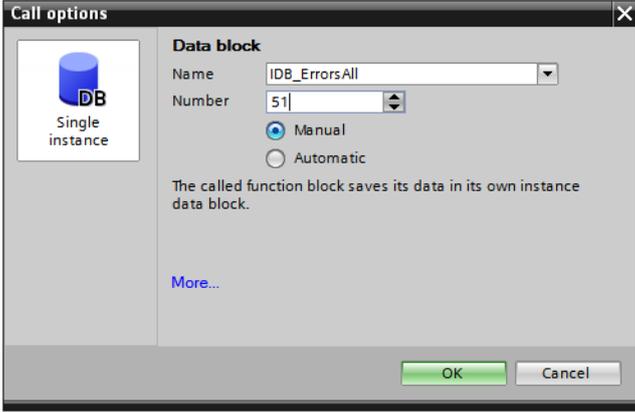
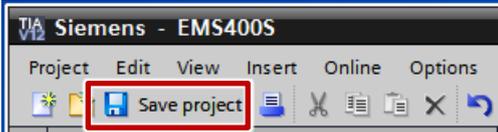
Hinweis

Der Baustein „FB_ErrorsHoist“ wird nur benötigt, wenn ein Hubantrieb / Kettenzug vorhanden ist.

Zyklisches Programm

Die folgende Tabelle listet die Schritte zum Integrieren der Bausteine der Funktionsbibliothek in die Fahrzeugsteuerung auf.

Tabelle 9-11

Nr.	Aktion
1.	Öffnen Sie den zyklischen OB1. Rufen Sie den Baustein „FB_ErrorsAll“ auf. Rufen Sie auch die Bausteine aus den anderen EMS-Funktionsbibliotheken und Ihre individuellen, anwenderdefinierten Bausteine für die Fahrzeugsteuerung auf (nicht Teil dieser Beschreibung).
2.	Weisen Sie dem Baustein „FB_ErrorsAll“ einen InstanzDB zu.
3.	Geben Sie z.B. die Bezeichnung „IDB_ErrorsAll“ ein. Übernehmen Sie die vorgeschlagene automatische Nummer, oder weisen Sie manuell eine Datenbausteinnummer, z.B. 51, um im Nummernband der Bausteine für die Störungsmeldungen zu bleiben. 
4.	Verschieben Sie den InstanzDB in die Gruppe „50_Errors“
5.	Übersetzen Sie den Baustein „FB_ErrorsAll“. 
6.	Speichern Sie das TIA STEP 7-Projekt. 

Hinweis

Wenn Ihre Fahrzeugsteuerung keinen Hubantrieb enthält, können Sie den Aufruf des Bausteins „FB_ErrorsHoist“ in „FB_ErrorsAll“ löschen, siehe [Abbildung 6-2](#).

9.7 Aufruf der Funktionsbibliotheksbausteine „EMS400S_General“ im TIA STEP 7 Programm

Hier sind die Schritte aufgeführt, um die Bausteine der Funktionsbibliothek „EMS400S_General“ in ein TIA STEP 7-Projekt einer S7-1200 Fahrzeugsteuerung zu integrieren.

Hinweis Im Folgenden wird vorausgesetzt, dass Sie die Hardware einer Fahrzeugsteuerung projektiert haben, siehe [13](#).

Ebenso wird vorausgesetzt, dass Sie die Systemmerker und die Taktmerker in den Eigenschaften der S7-1212 aktiviert haben.

Es wird vorausgesetzt, dass Sie die Basis-Bibliothek EMS400S in Ihr TIA STEP 7-Projekt integriert haben, siehe [14](#).

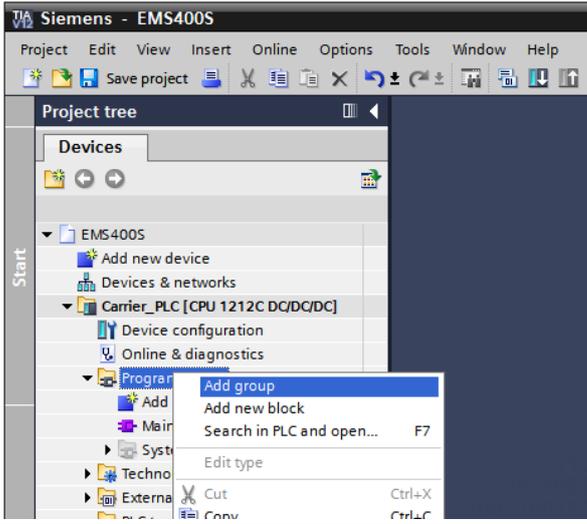
Hinweis Die Unterteilung der Programmbausteine in Gruppen ist nicht notwendig. Sie erleichtert die Übersicht im Programmbausteinordner.

Wir empfehlen, für die Bausteine der Basis-Bibliothek EMS400S eine Gruppe „00_PSB“ einzufügen.

Variablen-tabelle und Bausteine kopieren

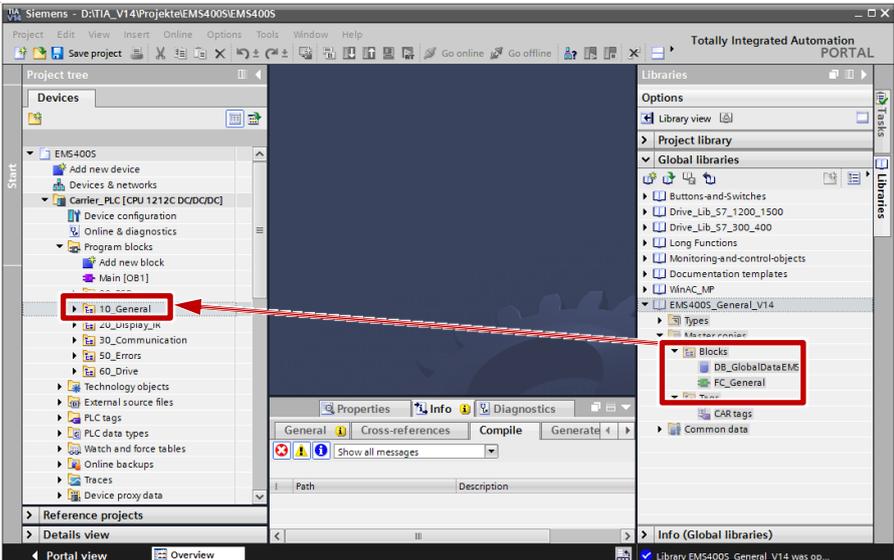
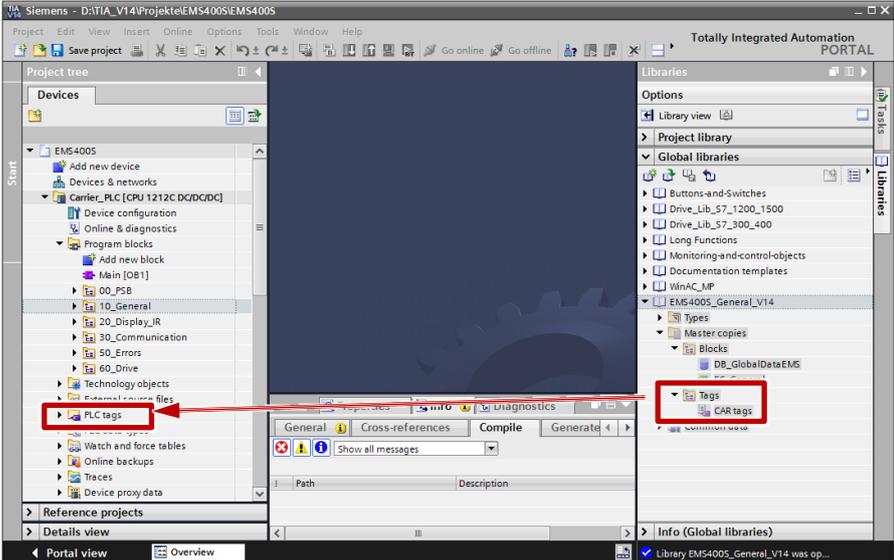
Die folgende Tabelle listet die Schritte zum Einfügen der Objekte der Funktionsbibliothek in die Fahrzeugsteuerung („Carrier PLC“) auf.

Tabelle 9-12

Nr.	Aktion
1.	<p>Fügen Sie dem Ordner „Programmbausteine“ eine neue Gruppe hinzu („Add group“)</p> 
2.	<p>Benennen Sie die Gruppe um, z.B. in „10_General“. Damit können Sie später sehen, dass sich hier die Bausteine für generelle Funktionen befinden, im Nummernband „10“</p>

9 Arbeiten mit den Funktionsbibliotheken

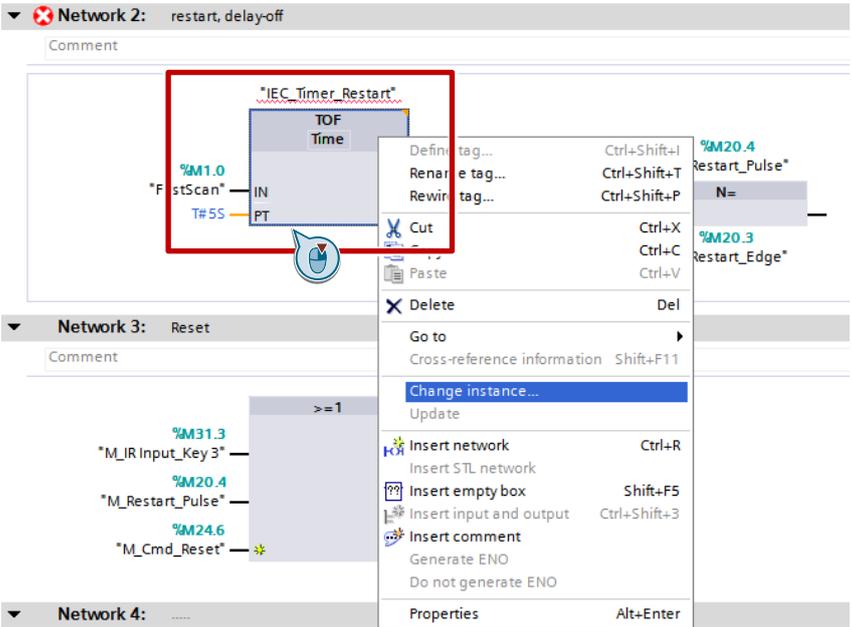
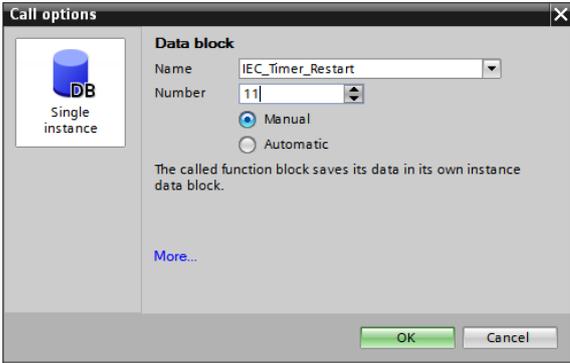
9.7 Aufruf der Funktionsbibliotheksbausteine „EMS400S_General“ im TIA STEP 7 Programm

Nr.	Aktion
3.	<p>Die Bausteine der Funktionsbibliothek „EMS400S_General“ befinden sich unter „Kopiervorlagen“ > „Blocks“ („Master copies“ > „Blocks“).</p> <p>Kopieren Sie alle Bausteine per Drag&Drop in die Gruppe „10_General“ im Ordner „Programmbausteine“ („Program blocks“).</p> 
4.	<p>Die Variablentabelle „CAR tags“ der Funktionsbibliothek „EMS400S_General“ befindet sich unter „Kopiervorlagen“ > „Tags“ („Master copies“ > „Tags“).</p> <p>Kopieren Sie die Variablentabelle per Drag&Drop in den Ordner „PLC-Variable“ („PLC tags“).</p> 

Zyklisches Programm

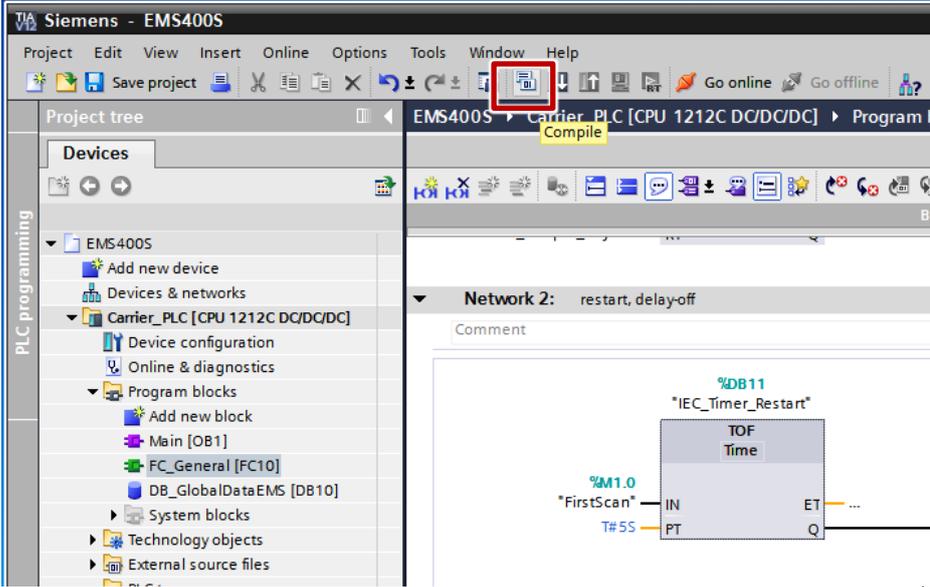
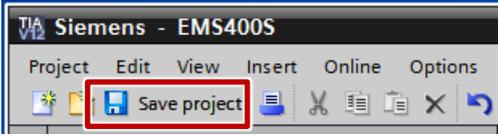
Die folgende Tabelle listet die Schritte zum Integrieren der Bausteine der Funktionsbibliothek in die Fahrzeugsteuerung auf.

Tabelle 9-13

Nr.	Aktion
1.	<p>Öffnen Sie den zyklischen OB1. Rufen Sie den Baustein „FC_General“ auf. Rufen Sie auch die Bausteine aus den anderen EMS-Funktionsbibliotheken und Ihre individuellen, anwenderdefinierten Bausteine für die Fahrzeugsteuerung auf (nicht Teil dieser Beschreibung).</p>
2.	<p>Öffnen Sie den Baustein „FC_General“. Wählen Sie den Timer in Netzwerk 2 an und weisen Sie dem Timer einen Instanz-Datenbaustein zu mit „Instanz ändern“ („Change instance“).</p> 
3.	<p>Übernehmen Sie die vorgeschlagene automatische Nummer, oder weisen Sie manuell eine Datenbausteinnummer, z.B. 11, um im Zahlenband für generelle Funktionen zu bleiben.</p> 

9 Arbeiten mit den Funktionsbibliotheken

9.7 Aufruf der Funktionsbibliotheksbausteine „EMS400S_General“ im TIA STEP 7 Programm

Nr.	Aktion
4.	<p>Übersetzen Sie den Baustein „FC_General“.</p>  <p>The screenshot shows the Siemens TIA Portal interface for project 'EMS400S'. The top menu bar includes 'Project', 'Edit', 'View', 'Insert', 'Online', 'Options', 'Tools', 'Window', and 'Help'. The 'Compile' button in the top toolbar is highlighted with a red box. The project tree on the left shows the following structure:</p> <ul style="list-style-type: none">EMS400S<ul style="list-style-type: none">Add new deviceDevices & networksCarrier_PLC [CPU 1212C DC/DC/DC]<ul style="list-style-type: none">Device configurationOnline & diagnosticsProgram blocks<ul style="list-style-type: none">Add new blockMain [OB1]FC_General [FC10]DB_GlobalDataEMS [DB10]System blocksTechnology objectsExternal source files <p>The main workspace displays a ladder logic network for 'Network 2: restart, delay-off'. It features a 'TOF Time' block with inputs '%M1.0' (labeled '*FirstScan*') and 'T#5S', and outputs 'ET' and 'Q'. The block is labeled '%DB11' and '*IEC_Timer_Restart*'.</p>
5.	<p>Speichern Sie das TIA STEP 7-Projekt.</p>  <p>The screenshot shows the Siemens TIA Portal interface for project 'EMS400S'. The top menu bar includes 'Project', 'Edit', 'View', 'Insert', 'Online', and 'Options'. The 'Save project' button in the top toolbar is highlighted with a red box.</p>

9.8 Aufruf der Funktionsbibliotheksbausteine „EMS400S_Hoist“ im TIA STEP 7 Programm

Hier sind die Schritte aufgeführt, um die Bausteine der Funktionsbibliothek „EMS400S_Hoist“ in ein TIA STEP 7-Projekt einer S7-1200 Fahrzeugsteuerung zu integrieren.

Hinweis

Im Folgenden wird vorausgesetzt, dass Sie die Hardware einer Fahrzeugsteuerung projektiert haben, siehe [13](#).

Ebenso wird vorausgesetzt, dass Sie die Systemmerker und die Taktmerker in den Eigenschaften der S7-1212 aktiviert haben.

Es wird vorausgesetzt, dass Sie die Basis-Bibliothek EMS400S in Ihr TIA STEP 7-Projekt integriert haben, siehe [14](#).

Ebenso wird vorausgesetzt, dass Sie die Funktionsbibliotheken
 - „EMS400S_General“ und
 - „EMS400_Drive“ in Ihr Projekt integriert haben.

Hinweis

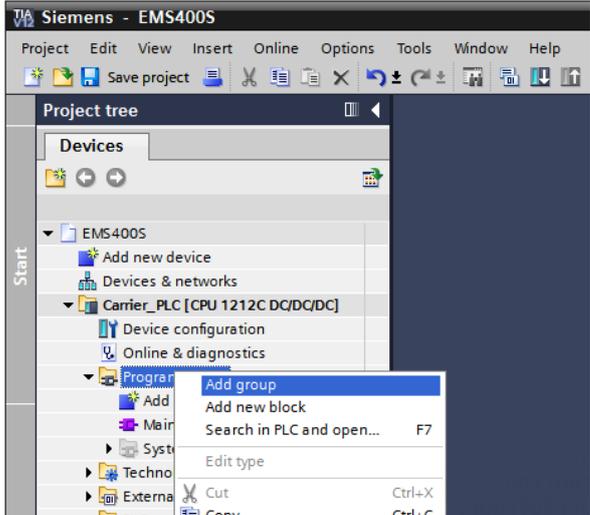
Die Unterteilung der Programmbausteine in Gruppen ist nicht notwendig. Sie erleichtert die Übersicht im Programmbausteinordner.

Wir empfehlen, für die Bausteine der Basis-Bibliothek EMS400S eine Gruppe „00_PSB“ einzufügen.

Bausteine kopieren

Die folgende Tabelle listet die Schritte zum Einfügen der Objekte der Funktionsbibliothek in die Fahrzeugsteuerung („Carrier PLC“) auf.

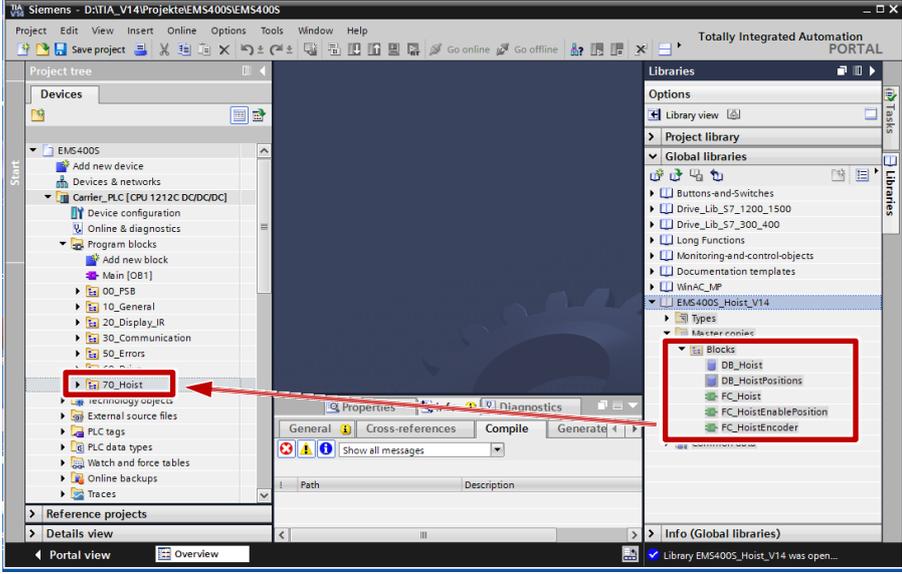
Tabelle 9-14

Nr.	Aktion
1.	<p>Fügen Sie dem Ordner „Programmbausteine“ eine neue Gruppe hinzu („Add group“)</p> 

9 Arbeiten mit den Funktionsbibliotheken

9.8 Aufruf der Funktionsbibliotheksbausteine „EMS400S_Hoist“ im TIA STEP 7 Programm

Nr.	Aktion
2.	Benennen Sie die Gruppe um, z.B. in „70_Hoist“. Damit können Sie später sehen, dass sich hier die Bausteine für Hubantriebsfunktionen befinden, im Nummernband „70“.
3.	Die Bausteine der Funktionsbibliothek „EMS400S_Hoist“ befinden sich unter „Kopiervorlagen“ > „Blocks“ („Master copies“ > „Blocks“). Kopieren Sie alle Bausteine per Drag&Drop in die Gruppe „70_Hoist“ im Ordner „Programmbausteine“ („Program blocks“).

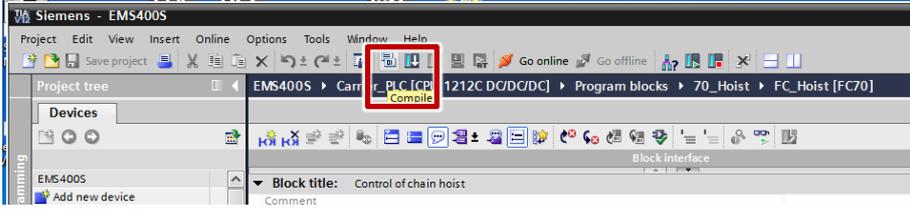
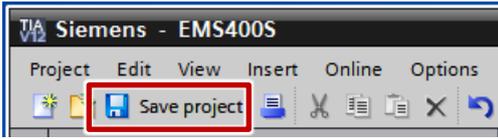


The screenshot shows the Siemens TIA Portal interface. The 'Project tree' on the left shows the 'Program blocks' folder expanded, with '70_Hoist' highlighted. The 'Libraries' panel on the right shows the 'EMS400S_Hoist_V14' library expanded, with the 'Blocks' folder highlighted. Red arrows indicate the relationship between these two elements.

Zyklisches Programm

Die folgende Tabelle listet die Schritte zum Integrieren der Bausteine der Funktionsbibliothek in die Fahrzeugsteuerung auf.

Tabelle 9-15

Nr.	Aktion
1.	Öffnen Sie den zyklischen OB1. Rufen Sie den Baustein „FC_Hoist“ auf. Rufen Sie auch die Bausteine aus den anderen EMS-Funktionsbibliotheken und Ihre individuellen, anwenderdefinierten Bausteine für die Fahrzeugsteuerung auf (nicht Teil dieser Beschreibung).
2.	Übersetzen Sie den Baustein „FC_Hoist“. 
3.	Speichern Sie das TIA STEP 7-Projekt. 

Hinweis

Beim Übersetzen des „FC_Hoist“ wird der Fehler „Network 1,Tag "TR_C58M not defined " angezeigt. Der Fehler wird behoben, indem das Gerät TR_C58M in das Projekt integriert wird, siehe Abschnitt „Konfiguration der Station TR Trommelgeber TR_C58M“.

Hinweis

„FC_Hoist“ verweist auf „DB_ErrorCodes“ aus der Funktionsbibliothek „EMS400S_Errors“. Falls „DB_ErrorCodes“ noch nicht im TIA STEP 7-Projekt enthalten ist, erhalten Sie Fehlermeldungen.

„FC_Hoist“ wird ohne diese Fehlermeldungen übersetzt, sobald die Funktionsbibliothek „EMS400S_Errors“ in Ihr TIA STEP 7-Projekt integriert ist.

10 Installation und Inbetriebnahme der Profinet (PN) -Geräte

Hier sind die Schritte aufgeführt, um die folgenden PN-Geräte in Ihr TIA STEP 7-Projekt zu integrieren:

- Sick Linearmesssensor OLM200
- TR Trommelgeber TR_C58M

Hinweis

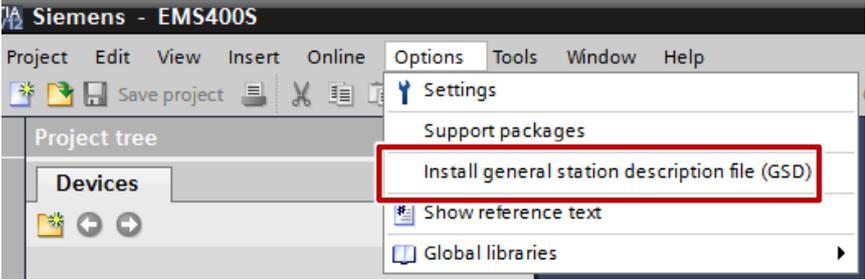
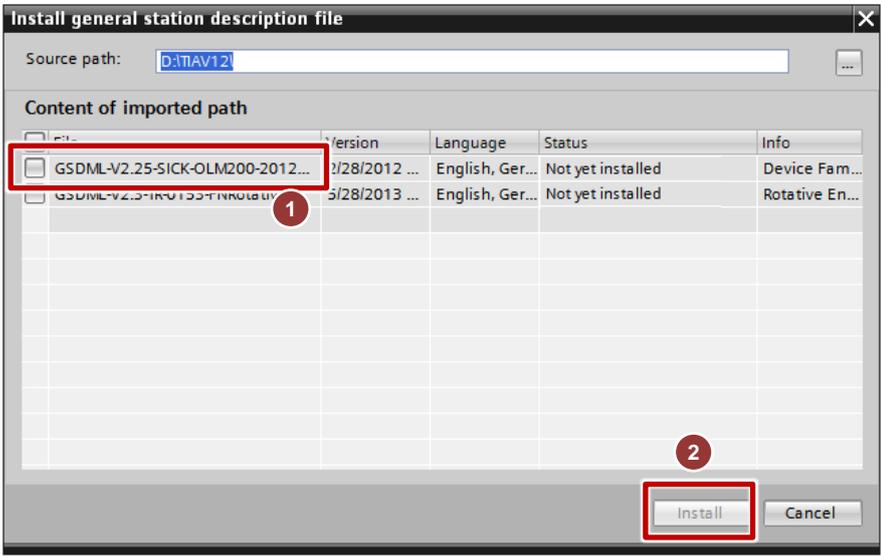
Die Aufbaurichtlinien sind generell zu beachten.

10.1 Sick Linearmesssensor OLM200

Integration der GSDML-Datei in TIA STEP7

In der folgenden Tabelle sind die Schritte aufgeführt, um die GSDML-Datei in Ihr TIA STEP 7-Projekt zu integrieren. Anschließend können Sie den Sick Linearmesssensor OLM200 in Ihrem TIA STEP 7-Projekt konfigurieren.

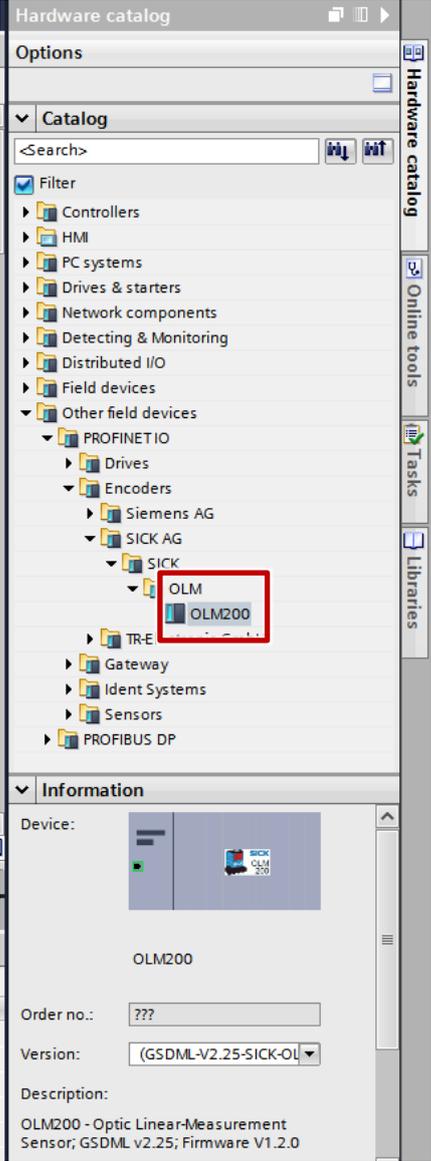
Tabelle 10-1

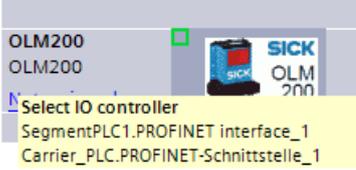
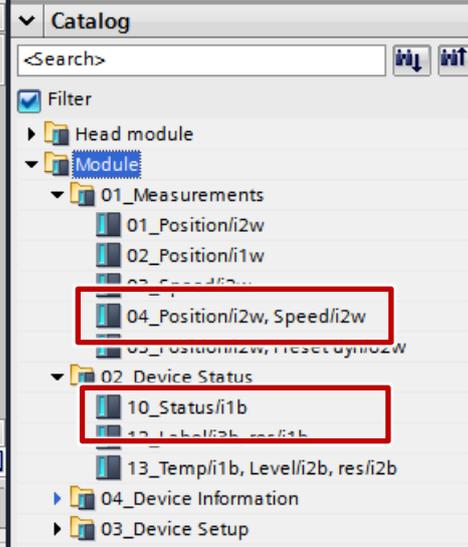
Nr.	Aktion
1.	Entpacken Sie die Datei „DC0003061.zip“, die Sie von der Hersteller-Seite 5 geladen haben, in einen beliebigen Ordner mit Schreib- und Lese-Rechten auf Ihrem lokalen Rechner.
2.	Öffnen Sie ihr bereits bestehendes TIA STEP 7 Projekt
3.	Wählen Sie im Menü „Extras“ den Befehl „Gerätebeschreibungdatei (GSD) installieren“ 
4.	Wählen Sie die GSD-Datei und klicken Sie auf „Installieren“ 

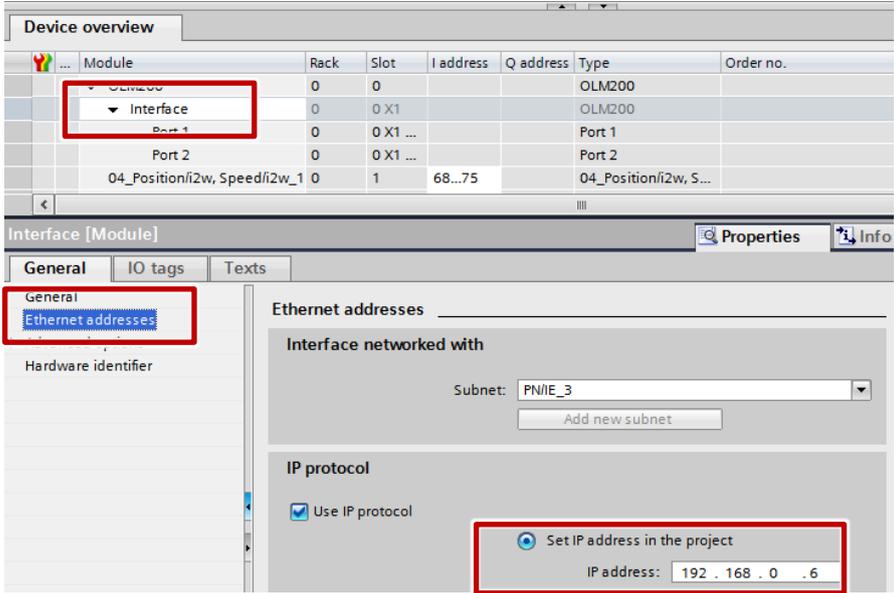
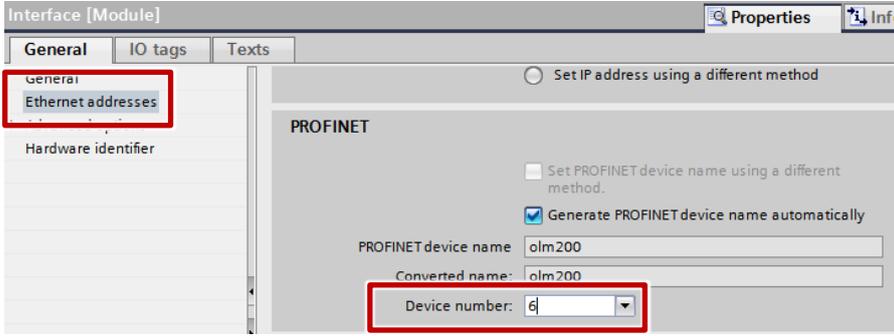
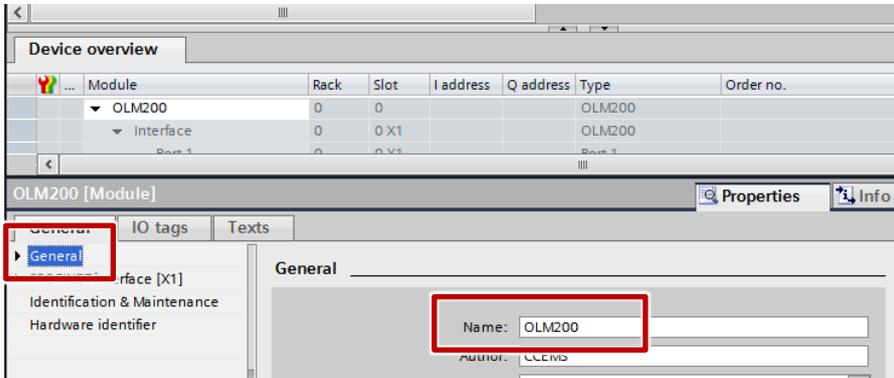
Konfiguration der Station Sick Linearmessensor

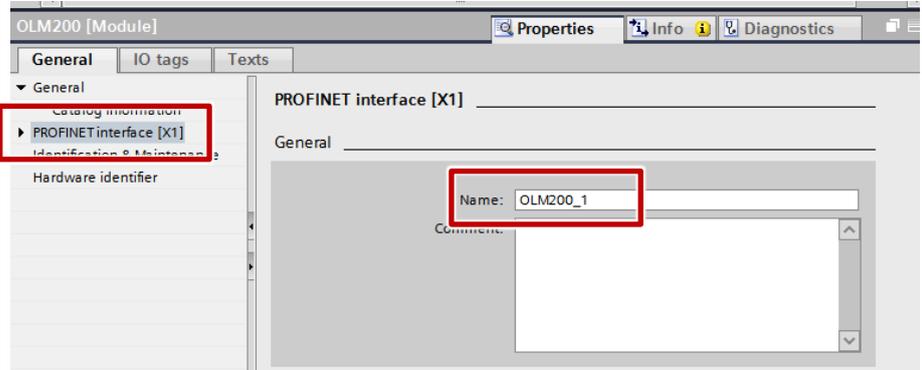
Die folgenden Tabelle zeigt die nötigen Schritte zur Konfiguration des Sick Linearmessensor OLM200 in Ihrem TIA STEP 7-Projekt.

Tabelle 10-2

Nr.	Aktion
1.	Öffnen Sie die Gerätekonfiguration / Netzsicht.
2.	<p>Wählen Sie im Hardwarekatalog / Weitere Feldgeräte / PROFINET IO / Encoders / SICK AG / OLM das Gerät „OLM200“.</p> <p>Ziehen Sie das Gerät per Drag&Drop in den graphischen Bereich der Netzsicht. Alternativ können Sie auf das Gerät doppelklicken.</p> 

Nr.	Aktion																																																
3.	<p>Weisen Sie dem Gerät OLM200 einen IO-Controller zu.</p>  <p>Ergebnis: Das Gerät ist der Carrier_PLC zugeordnet</p> 																																																
4.	<p>Öffnen Sie die Gerätesicht des OLM200.</p>																																																
5.	<p>Wählen Sie aus dem Hardwarekatalog die Module - „04_Position/i2w, Speed/i2w“ und - „10_Status/i1b“ und fügen Sie sie per Drag&Drop in das Gerät ein. Übernehmen Sie die vorgeschlagenen EA-Adressen „68 ... 75“ und „76“.</p>  <p>Ergebnis:</p> <table border="1" data-bbox="459 1503 1353 1691"> <thead> <tr> <th colspan="8">Device overview</th> </tr> <tr> <th>Module</th> <th>Rack</th> <th>Slot</th> <th>I address</th> <th>Q address</th> <th>Type</th> <th>Order no</th> <th></th> </tr> </thead> <tbody> <tr> <td>OLM200</td> <td>0</td> <td>0</td> <td></td> <td></td> <td>OLM200</td> <td></td> <td></td> </tr> <tr> <td> Interface</td> <td>0</td> <td>0 X1</td> <td></td> <td></td> <td>OLM200</td> <td></td> <td></td> </tr> <tr> <td> 04_Position/i2w, Speed/i2w_1</td> <td>0</td> <td>1</td> <td>68...75</td> <td></td> <td>04_Position/i2w, S...</td> <td></td> <td></td> </tr> <tr> <td> 10_Status/i1b_1</td> <td>0</td> <td>2</td> <td>76</td> <td></td> <td>10_Status/i1b</td> <td></td> <td></td> </tr> </tbody> </table>	Device overview								Module	Rack	Slot	I address	Q address	Type	Order no		OLM200	0	0			OLM200			Interface	0	0 X1			OLM200			04_Position/i2w, Speed/i2w_1	0	1	68...75		04_Position/i2w, S...			10_Status/i1b_1	0	2	76		10_Status/i1b		
Device overview																																																	
Module	Rack	Slot	I address	Q address	Type	Order no																																											
OLM200	0	0			OLM200																																												
Interface	0	0 X1			OLM200																																												
04_Position/i2w, Speed/i2w_1	0	1	68...75		04_Position/i2w, S...																																												
10_Status/i1b_1	0	2	76		10_Status/i1b																																												

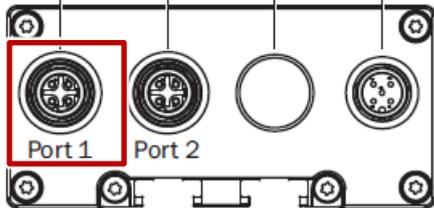
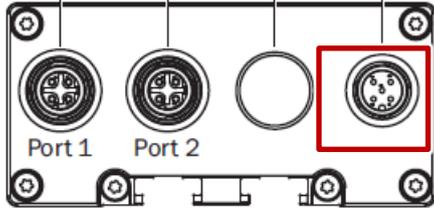
Nr.	Aktion
6.	<p>Wählen Sie die PROFINET-Schnittstelle an und weisen Sie eine IP-Adresse zu, entsprechend der Vorgabe des Kunden.</p> 
7.	<p>Weisen Sie die Gerätenummer zu, entsprechend der Vorgabe des Kunden. Hinweis: In der Störungsauswertung „FB_ErrorsDrive“, Netzwerk 3, wird die Diagnose für Gerät Nr. 6 abgefragt. Wenn Sie eine andere Gerätenummer einstellen, müssen Sie „FB_ErrorsDrive“ anpassen.</p> 
8.	<p>Weisen Sie einen Gerätenamen zu, entsprechend der Vorgabe des Kunden.</p> 

Nr.	Aktion
9.	<p>Weisen Sie der Schnittstelle einen Profinet-Namen zu, entsprechend der Vorgabe des Kunden.</p> 

Installation der Hardware

Schließen Sie den Sick Linearmesssensor OLM200 wie folgt an Ihre Fahrzeugsteuerung an.

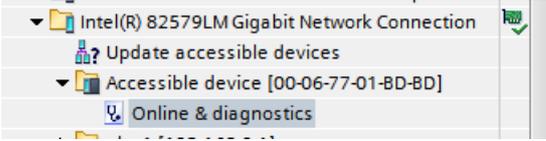
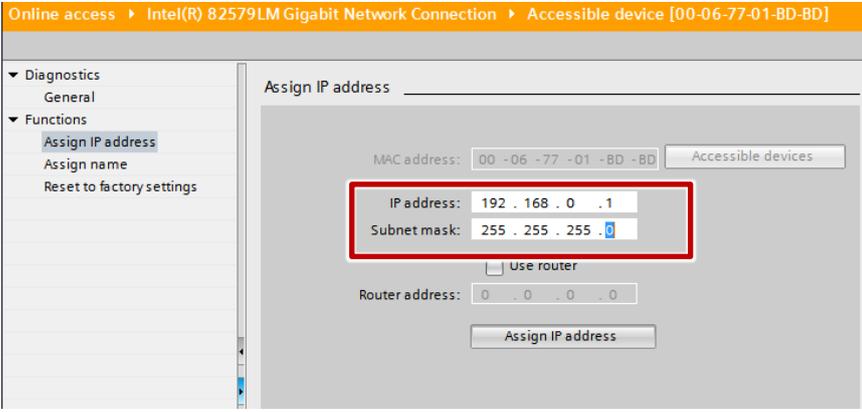
Tabelle 10-3

Nr.	Aktion
1.	<p>Schließen Sie die PROFINET-Verbindung an den Sick Linearmesssensor an.</p> 
2.	<p>Schließen Sie die Spannungsversorgung an den Sick Linearmesssensor an.</p> 

Taufe der Station Sick Linearmesssensor

Damit die Fahrzeugsteuerung mit dem Sick Linearmesssensor OLM200 kommunizieren kann, muss dieser getauft werden.

Tabelle 10-4

Nr.	Aktion
1.	Verbinden Sie Ihr Programmiergerät mit dem Sick Linearmesssensor OLM200.
2.	<p>Wählen Sie Ihre Schnittstelle und aktualisieren Sie die erreichbaren Teilnehmer.</p>  <p>The screenshot shows a network configuration menu with 'Online access' expanded. Underneath, several network interfaces are listed, including 'USB [S7USB]', 'COM <3> [RS232/PPI multi-master cable]', and 'Intel(R) 82579LM Gigabit Network Connection'. The 'Update accessible devices' option is highlighted with a red box.</p>
3.	<p>Wählen Sie das Gerät „Erreichbarer Teilnehmer“ („Accessible Device“) an und öffnen Sie „Online & Diagnose“ / Funktionen.</p>  <p>The screenshot shows the configuration for an 'Accessible device [00-06-77-01-BD-BD]'. The 'Update accessible devices' option is selected. Below it, the 'Online & diagnostics' option is highlighted.</p>
4.	<p>Weisen Sie die IP-Adresse zu.</p>  <p>The screenshot shows the 'Assign IP address' dialog box. The 'MAC address' is '00-06-77-01-BD-BD'. The 'IP address' field is set to '192.168.0.1' and the 'Subnet mask' is '255.255.255.0'. Both fields are highlighted with a red box. There is also a 'Use router' checkbox and a 'Router address' field set to '0.0.0.0'. An 'Assign IP address' button is at the bottom.</p>

10 Installation und Inbetriebnahme der Profinet (PN) -Geräte

10.1 Sick Linearmesssensor OLM200

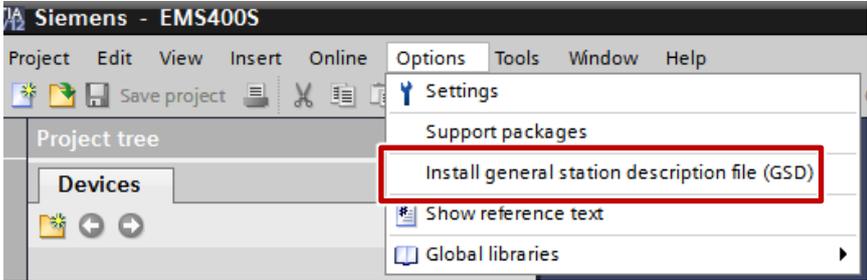
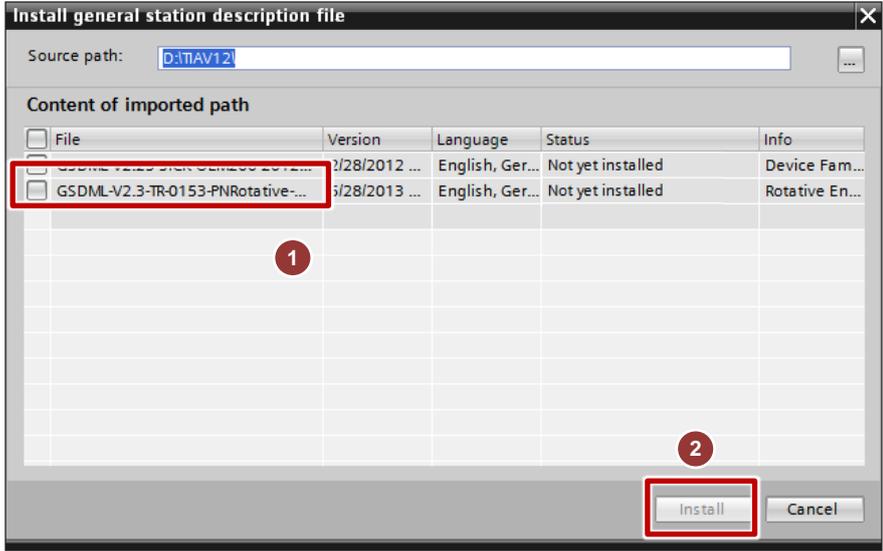
Nr.	Aktion																									
5.	<p data-bbox="469 309 762 336">Weisen Sie den Namen zu.</p> <div data-bbox="469 338 1342 922"><p data-bbox="469 338 1342 365">Online access ▶ Intel(R) 82579LM Gigabit Network Connection ▶ Accessible device [00-06-77-01-BD-BD]</p><p data-bbox="469 392 662 504">▼ Diagnostics General ▼ Functions Assign IP address Assign name Reset to factory settings</p><p data-bbox="683 392 1342 418">Assign name _____</p><p data-bbox="890 454 1326 495">PROFINET device name: <input type="text"/></p><p data-bbox="1034 495 1299 510">type: OLM200</p><p data-bbox="943 622 1206 689"><input type="checkbox"/> Only show devices of the same type <input type="checkbox"/> Only show devices with bad parameter settings <input type="checkbox"/> Only show devices without names</p><p data-bbox="810 719 1018 741">Accessible devices in the network: <input type="button" value="refresh"/></p><table border="1" data-bbox="730 745 1299 891"><thead><tr><th>IP address</th><th>MAC address</th><th>Type</th><th>Name</th><th>Status</th></tr></thead><tbody><tr><td> </td><td> </td><td> </td><td> </td><td> </td></tr><tr><td> </td><td> </td><td> </td><td> </td><td> </td></tr><tr><td> </td><td> </td><td> </td><td> </td><td> </td></tr><tr><td> </td><td> </td><td> </td><td> </td><td> </td></tr></tbody></table><p data-bbox="1038 898 1118 913"><input type="checkbox"/> LED flashes</p><p data-bbox="1166 875 1326 922"><input type="button" value="Assign name"/></p></div>	IP address	MAC address	Type	Name	Status																				
IP address	MAC address	Type	Name	Status																						

10.2 TR Trommelgeber TR_C58M

Integration der GSDML-Datei in TIA STEP7

In der folgenden Tabelle sind die Schritte aufgeführt, um die GSDML-Datei des TR Trommelgeber TR_C58M in Ihr TIA STEP 7-Projekt zu integrieren. Anschließend können Sie das Modul konfigurieren.

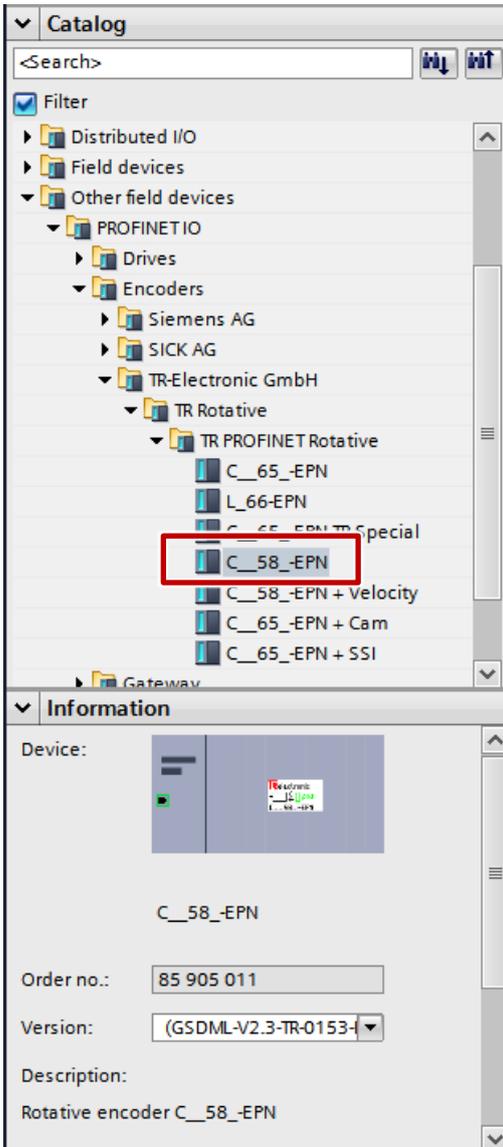
Tabelle 10-5

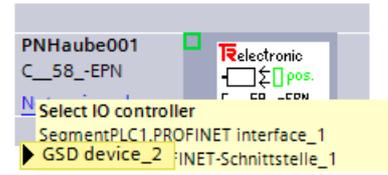
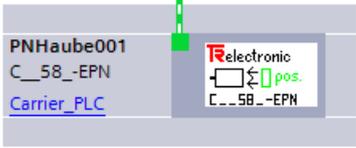
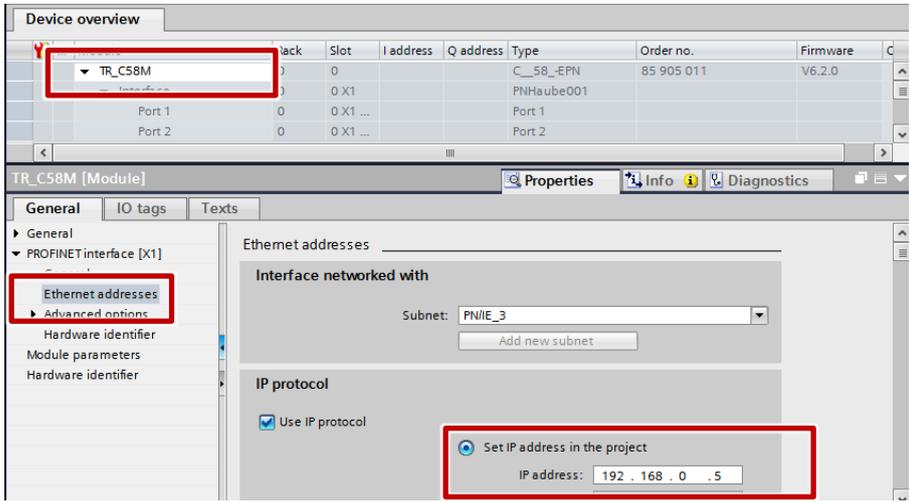
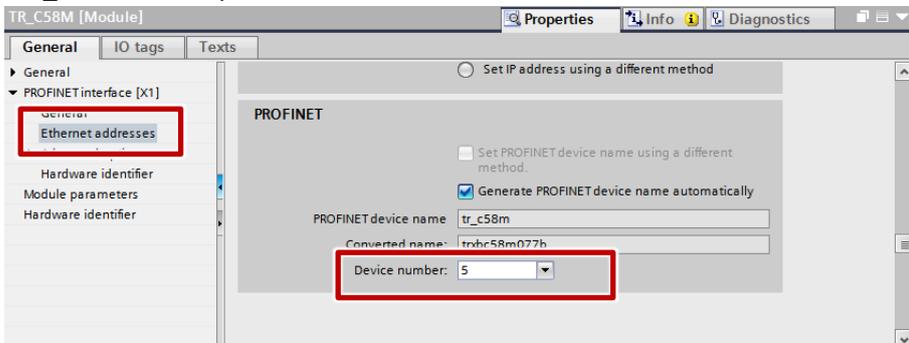
Nr.	Aktion
1.	Entpacken Sie die Datei „49000423.zip“, die Sie von der Hersteller-Seite 6 geladen haben, in einen beliebigen Ordner mit Schreib- und Lese-Rechten auf Ihrem lokalen Rechner.
2.	Öffnen Sie ihr bereits bestehendes TIA STEP 7 Projekt.
3.	Wählen Sie im Menü „Extras“ den Befehl „Gerätebeschreibungsdatei (GSD) installieren“. 
4.	Wählen Sie die GSD-Datei und klicken Sie auf „Installieren“. 

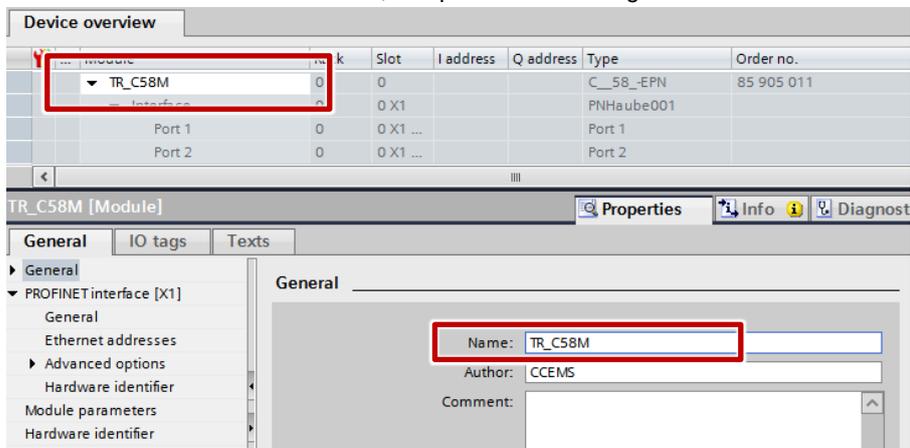
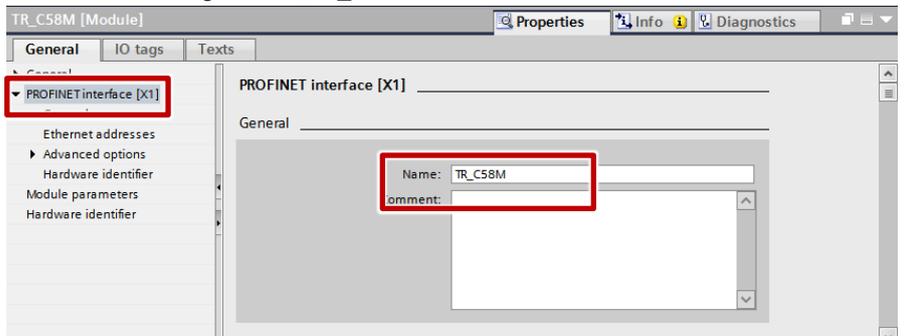
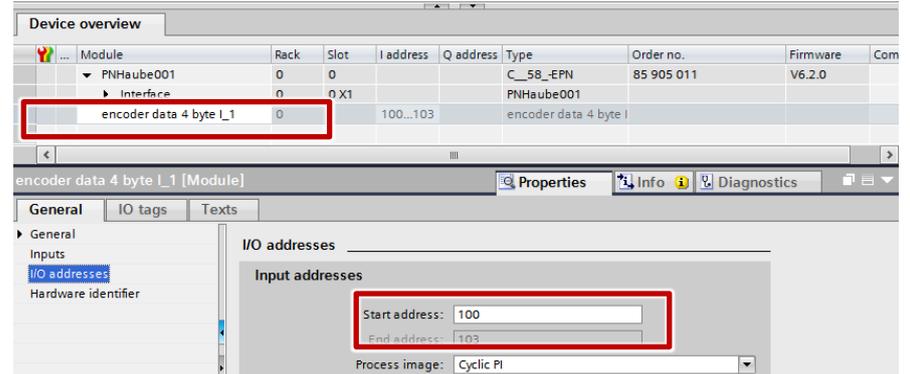
Konfiguration der Station TR Trommelgeber TR_C58M

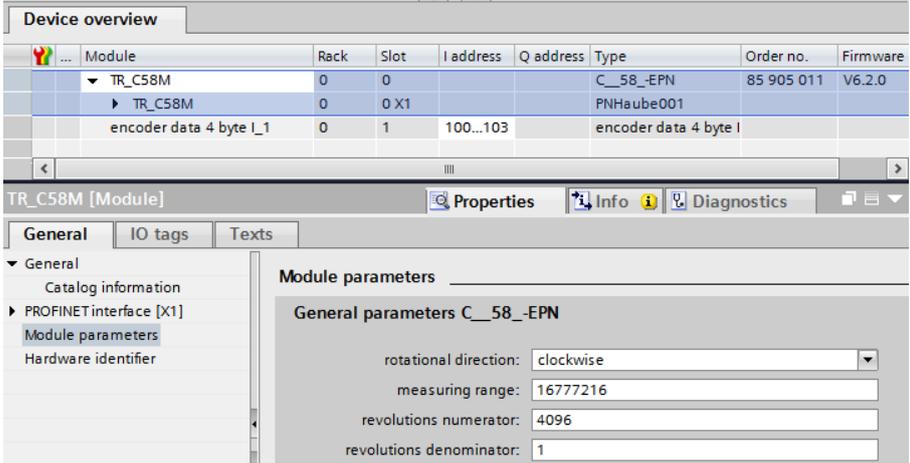
Die folgenden Tabelle zeigt die nötigen Schritte zur Konfiguration des TR Trommelgeber TR_C58M in Ihrem TIA STEP 7-Projekt.

Tabelle 10-6

Nr.	Aktion
1.	Öffnen Sie die Gerätekonfiguration / Netzsicht.
2.	<p>Wählen Sie im Hardwarekatalog /Weitere Feldgeräte / PROFINET IO / Encoders / TR-Electronic GmbH / TR Rotative / TR PROFINET Rotativ das Gerät „C__58_-EPN“.</p> <p>Ziehen Sie das Gerät per Drag&Drop in den graphischen Bereich der Netzsicht. Alternativ können Sie auf das Gerät doppelklicken.</p>  <p>The screenshot shows the 'Catalog' pane on the left with a tree view expanded to 'TR PROFINET Rotative' > 'C__58_-EPN', which is highlighted with a red box. The 'Information' pane on the right shows the device name 'C__58_-EPN', order number '85 905 011', version '(GSDML-V2.3-TR-0153-)', and description 'Rotative encoder C__58_-EPN'.</p>

Nr.	Aktion
3.	<p>Weisen Sie dem Gerät PNHaube001 einen IO-Controller zu.</p>  <p>Ergebnis: Das Gerät ist der Carrier_PLC zugeordnet.</p> 
4.	<p>Öffnen Sie die Gerätesicht des PNHaube001.</p>
5.	<p>Wählen Sie die PROFINET-Schnittstelle an und weisen Sie eine IP-Adresse zu, entsprechend der Vorgabe des Kunden.</p> 
6.	<p>Weisen Sie die Gerätenummer zu, entsprechend der Vorgabe des Kunden. Hinweis: In der Störungsauswertung „FB_ErrorsHoist“, Netzwerk 4, wird die Diagnose für Gerät Nr. 5 abgefragt. Wenn Sie eine andere Gerätenummer einstellen, müssen Sie „FB_ErrorsHoist“ anpassen.</p> 

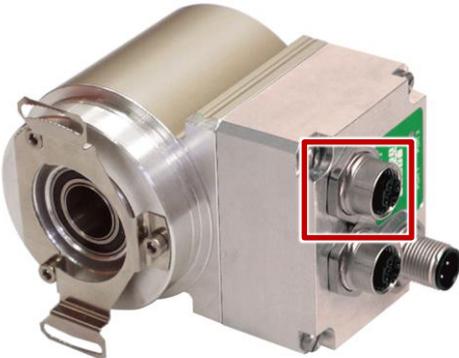
Nr.	Aktion
7.	<p>Weisen Sie einen Gerätenamen zu, entsprechend der Vorgabe des Kunden.</p>  <p>The screenshot shows the 'Device overview' table with columns: Module, Rack, Slot, I address, Q address, Type, Order no. The 'TR_C58M' module is selected. Below, the 'Properties' dialog for 'TR_C58M [Module]' is open, with the 'Name' field containing 'TR_C58M' and 'Author' set to 'CCEMS'.</p>
8.	<p>Weisen Sie der Schnittstelle den Profinet-Namen TR_C58M zu. Diese Bezeichnung wird in „FC_Hoist“ verwendet.</p>  <p>The screenshot shows the 'PROFINET interface [X1]' properties dialog. The 'Name' field is set to 'TR_C58M'.</p>
9.	<p>Übersetzen Sie „FC_Hoist“. Anschließend können Sie den Profinet-Namen entsprechend der Vorgabe des Kunden ändern, die Verwendung in „FC_Hoist“ wird von TIA aktualisiert.</p>
10.	<p>Stellen Sie die EA-Adressen auf „100 ... 103“ ein.</p>  <p>The screenshot shows the 'Device overview' table with columns: Module, Rack, Slot, I address, Q address, Type, Order no., Firmware, Com. The 'encoder data 4 byte I_1' module is selected. Below, the 'I/O addresses' dialog for 'encoder data 4 byte I_1 [Module]' is open, with 'Start address' set to 100 and 'End address' set to 103.</p>

Nr.	Aktion																																
11.	<p>Parametrieren Sie das Modul so, dass 1 Impuls / mm ausgegeben wird.</p> <p>Die Erläuterung zur Berechnung der Parameter finden Sie im Handbuch TR-ECE-BA-DGB-0088-05.pdf, Kapitel 7; siehe 6</p>  <p>Device overview</p> <table border="1"> <thead> <tr> <th>Module</th> <th>Rack</th> <th>Slot</th> <th>I address</th> <th>Q address</th> <th>Type</th> <th>Order no.</th> <th>Firmware</th> </tr> </thead> <tbody> <tr> <td>TR_C58M</td> <td>0</td> <td>0</td> <td></td> <td></td> <td>C__58_-EPN</td> <td>85 905 011</td> <td>V6.2.0</td> </tr> <tr> <td> TR_C58M</td> <td>0</td> <td>0 X1</td> <td></td> <td></td> <td>PNHaube001</td> <td></td> <td></td> </tr> <tr> <td> encoder data 4 byte I_1</td> <td>0</td> <td>1</td> <td>100...103</td> <td></td> <td>encoder data 4 byte I</td> <td></td> <td></td> </tr> </tbody> </table> <p>Module parameters</p> <p>General parameters C__58_-EPN</p> <ul style="list-style-type: none"> rotational direction: clockwise measuring range: 16777216 revolutions numerator: 4096 revolutions denominator: 1 	Module	Rack	Slot	I address	Q address	Type	Order no.	Firmware	TR_C58M	0	0			C__58_-EPN	85 905 011	V6.2.0	TR_C58M	0	0 X1			PNHaube001			encoder data 4 byte I_1	0	1	100...103		encoder data 4 byte I		
Module	Rack	Slot	I address	Q address	Type	Order no.	Firmware																										
TR_C58M	0	0			C__58_-EPN	85 905 011	V6.2.0																										
TR_C58M	0	0 X1			PNHaube001																												
encoder data 4 byte I_1	0	1	100...103		encoder data 4 byte I																												

Installation der Hardware

Schließen Sie den TR Trommelgeber TR_C58M wie folgt an Ihre Fahrzeugsteuerung an.

Tabelle 10-7

Nr.	Aktion
1.	<p>Schließen Sie die PROFINET-Verbindung an Port 1 des TR Trommelgebers TR_C58M an.</p> 
2.	<p>Schließen Sie die Spannungsversorgung an den TR Trommelgeber TR_C58M an.</p> 

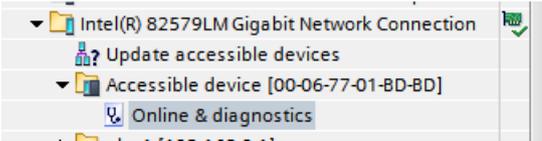
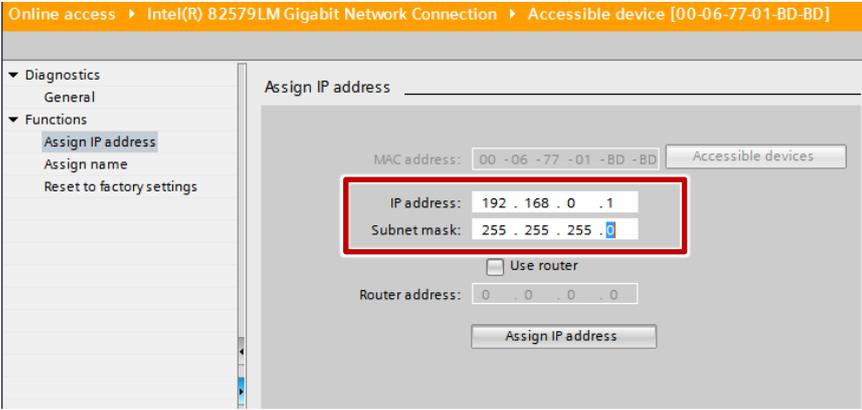
Hinweis

Die Abbildung ist ähnlich dem TR Trommelgeber TR_C58M.

Taufe der Station TR Trommelgeber TR_C58M

Damit die Fahrzeugsteuerung mit dem TR Trommelgeber TR_C58M kommunizieren kann, ist eine Taufe notwendig.

Tabelle 10-8

Nr.	Aktion
1.	Verbinden Sie Ihr Programmiergerät mit dem TR Trommelgeber TR_C58M.
2.	<p>Wählen Sie Ihre Schnittstelle und aktualisieren Sie die erreichbaren Teilnehmer.</p> 
3.	<p>Wählen Sie das Gerät „Erreichbarer Teilnehmer“ (Accessible Device) an und öffnen Sie „Online & Diagnose“ / Funktionen.</p> 
4.	<p>Weisen Sie die IP-Adresse zu.</p> 

10 Installation und Inbetriebnahme der Profinet (PN) -Geräte

10.2 TR Trommelgeber TR_C58M

Nr.	Aktion																									
5.	<p data-bbox="469 309 762 336">Weisen Sie den Namen zu.</p> <div data-bbox="469 336 1340 922"><p data-bbox="475 340 1334 362">Online access ▶ Intel(R) 82579LM Gigabit Network Connection ▶ Accessible device [00-06-77-01-BD-BD]</p><p data-bbox="475 389 667 501">▼ Diagnostics General ▼ Functions Assign IP address Assign name Reset to factory settings</p><p data-bbox="683 398 778 421">Assign name _____</p><div data-bbox="887 452 1327 510"><p data-bbox="938 470 1305 492">PROFINET device name: <input type="text"/></p><p data-bbox="1034 497 1305 519">type: OLM200</p></div><p data-bbox="941 627 1209 694"><input type="checkbox"/> Only show devices of the same type <input type="checkbox"/> Only show devices with bad parameter settings <input type="checkbox"/> Only show devices without names</p><p data-bbox="810 721 1018 743">Accessible devices in the network: <input type="button" value="refresh"/></p><table border="1" data-bbox="730 748 1305 855"><thead><tr><th>IP address</th><th>MAC address</th><th>Type</th><th>Name</th><th>Status</th></tr></thead><tbody><tr><td> </td><td> </td><td> </td><td> </td><td> </td></tr><tr><td> </td><td> </td><td> </td><td> </td><td> </td></tr><tr><td> </td><td> </td><td> </td><td> </td><td> </td></tr><tr><td> </td><td> </td><td> </td><td> </td><td> </td></tr></tbody></table><p data-bbox="1037 900 1295 922"><input type="checkbox"/> LED flashes <input type="button" value="Assign name"/></p></div>	IP address	MAC address	Type	Name	Status																				
IP address	MAC address	Type	Name	Status																						

11 Literaturhinweise

Tabelle 11-1

	Themengebiet	Titel
\1\	Siemens Industry Online Support	http://support.automation.siemens.com
\2\	Downloadseite des Beitrages	http://support.automation.siemens.com/WW/view/de/89369337
\3\	EMS400S Dokumentation	http://support.automation.siemens.com/WW/view/de/30360848/133300
\4\	Basis-Bibliothek EMS400S, Kommunikationsbausteine für die Fahrzeugsteuerung SIMATIC EMS400S	http://support.automation.siemens.com/WW/view/de/89369337
\5\	Sick Linear-Messsensor OLM200	http://www.sick.com/group/de/home/products/product_news/distance_sensors/seiten/olm200_linear_measurement_sensor.aspx
\6\	TR Trommelgeber TR_C58M	http://www.tr-electronic.de/produkte/drehgeber/absolutdrehgeber/standard-kompaktdrehgeber/c-58.html

12 Historie

Tabelle 12-1

Version	Datum	Änderung
V1.0	07/2014	Erste Ausgabe
V1.1	04/2015	Anpassung an TIA V13 SP1. Korrekturen: Kapitel 5.2.1: Symbol M_AnalogValue geändert Schleichfahrt und Lichttaster hinzugefügt Kapitel 8.2.2: Symbol Hoist_minimum geändert Kapitel 7.2.3 und 9: Hinweise auf System- und Taktmerker
V1.2	01/2017	Anpassung an TIA V14 Kapitel 1.5: Add-on entfällt Kapitel 8.2.3: Datentyp von HW_ID_Encoder geändert