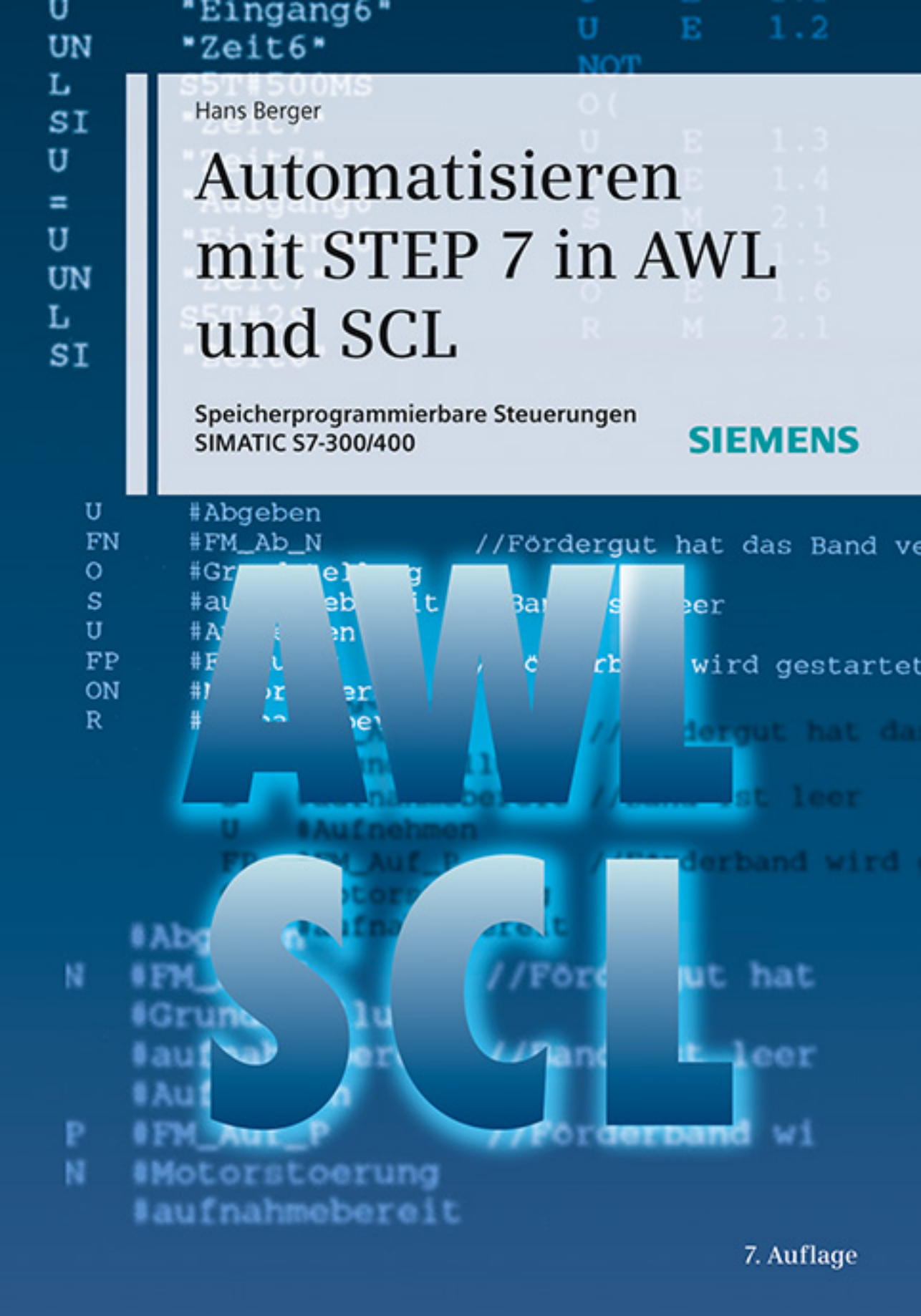

Hans Berger
Automatisieren mit STEP 7 in AWL und SCL
Speicherprogrammierbare Steuerungen
SIMATIC S7-300/400
SIEMENS
AWL
SCL
7. Auflage

Berger Automatisieren mit STEP 7 in AWL und SCL

Automatisieren mit STEP 7 in AWL und SCL

Speicherprogrammierbare Steuerungen
SIMATIC S7-300/400

von Hans Berger

7., überarbeitete und erweiterte Auflage, 2011

Publicis Publishing

Bibliografische Information der Deutschen Nationalbibliothek
Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über http://dnb.d-nb.de abrufbar.

Dem Buch liegt eine Demo DVD der Siemens AG bei.

„SIMATIC STEP 7 Professional, Edition 2006 SR5, Trial License" umfasst: SIMATIC STEP 7 V5.4 SP4, S7-GRAPH V5.3 SP6, S7-SCL V5.3 SP5, S7-PLCSIM V5.4 SP2 und ist 14 Tage zu Testzwecken nutzbar.

Die Software ist nur unter Microsoft Windows XP Professional Edition SP3 oder Microsoft Windows Vista 32 Bit Business SP1/SP2 oder Microsoft Windows Vista 32 Bit Ultimate SP1/SP2 ablauffähig.

Weitere Informationen erhalten Sie im Internet unter
http://www.siemens.de/sce/promotoren
http://www.siemens.de/sce/module
http://www.siemens.de/sce/tp

Die Programmierbeispiele sind dafür vorgesehen, schwerpunktmäßig die AWL- und SCL-Funktionen zu beschreiben und Anwendern von SIMATIC S7 Anhaltspunkte zu geben, wie bestimmte Aufgabenstellungen mit dieser Steuerung programmtechnisch gelöst werden können.

Bei den im Buch gezeigten Programmierbeispielen handelt es sich um Lösungsideen ohne Anspruch auf Vollständigkeit oder auf Ablauffähigkeit bei zukünftigen Ausgabeständen von STEP 7 oder S7-300/400. Für die Einhaltung entsprechender Sicherheitsvorschriften ist bei der Anwendung zusätzliche Sorge zu tragen.

Autor und Verlag haben alle Texte und Abbildungen in diesem Buch mit großer Sorgfalt erarbeitet. Dennoch können Fehler nicht ausgeschlossen werden. Eine Haftung des Verlags oder des Autors, gleich aus welchem Rechtsgrund, für durch die Verwendung der Programmierbeispiele verursachte Schäden ist ausgeschlossen.

Für Anregungen zum Inhalt weiterer Auflagen sind Autor und Verlag immer dankbar.
Publicis Publishing
Postfach 3240
91050 Erlangen
E-Mail: publishing-distribution@publicis.de

ISBN 978-3-89578-397-5

7. Auflage, 2011

Herausgeber Siemens Aktiengesellschaft, Berlin und München
Verlag: Publicis Publishing, Erlangen

Printed in Germany

Vorwort

Das Automatisierungssystem SIMATIC vereinigt alle Teilsysteme einer Automatisierungslösung unter einer einheitlichen Systemarchitektur zu einem homogenen Gesamtsystem von der Feldebene bis zur Leittechnik. Durch diese Totally Integrated Automation (TIA) wird eine durchgängige Projektierung, Programmierung, Datenhaltung und Kommunikation im gesamten Automatisierungssystem erreicht.

STEP 7 übernimmt als Basiswerkzeug für SIMATIC die Klammerfunktion bei Totally Integrated Automation. Mit STEP 7 erfolgt die Projektierung und Programmierung der Automatisierungssysteme SIMATIC S7, SIMATIC C7 und SIMATIC WinAC. Als Betriebssystem wurde Microsoft Windows gewählt und somit die Welt der Standard-PCs mit der auch in der Bürowelt weit verbreiteten und damit bekannten Bedienoberfläche erschlossen.

Für die Bausteinprogrammierung stellt STEP 7 nach DIN EN 6.1131-3 genormte Programmiersprachen zur Verfügung: AWL (Anweisungsliste, eine Assembler-ähnliche Sprache), KOP (Kontaktplan, eine Stromlaufplan-ähnliche Darstellung), FUP (Funktionsplan) und das Optionspaket S7-SCL (Structured Control Language, eine Pascal-ähnliche Hochsprache). Zusätzliche Optionspakete ergänzen diese Sprachen: S7-GRAPH (Ablaufkettensteuerung), S7-HiGraph (Programmierung als Zustandsgraph) und CFC (Funktionsplan-ähnliche Verschaltung von Bausteinen). Die verschiedenen Darstellungsarten geben jedem Anwender die Möglichkeit, sich die für ihn geeignete Beschreibung der Steuerungsfunktion auszuwählen. Die so gewonnene weitgehende Übereinstimmung mit der Darstellung der zu lösenden Steuerungsaufgabe vereinfacht wesentlich die Handhabung von STEP 7.

Dieses Buch beschreibt die Programmiersprachen AWL und SCL für S7-300/400. Als wertvolle Ergänzung zur Sprachbeschreibung bietet es nach einer Einführung in das Automatisierungssystem S7-300/400 praxisnahe Hinweise für die grundsätzliche Handhabung von STEP 7 bei der Projektierung der SIMATIC-Steuerungen, ihrer Vernetzung und Programmierung. Die Beschreibung der „Basisfunktionen“ einer binären Steuerung, wie z.B. logische Verknüpfungen oder Speicherfunktionen, erleichtern besonders für Anfänger oder Umsteiger von Schützensteuerungen den Einstieg in STEP 7. Die Digitalfunktionen erläutern die Verknüpfung digitaler Werte, z.B. Grundrechnungsarten, Vergleiche oder Datentypwandlung.

Das Buch zeigt, wie Sie die Programmbearbeitung (den Programmfluss) steuern und ein Programm strukturiert aufbauen können. Zusätzlich zum zyklisch bearbeiteten Hauptprogramm können Sie ereignisgesteuerte Programmteile einbinden sowie das Verhalten der Steuerung im Anlauf und im Fehlerfall beeinflussen.

Ein eigener Buchteil befasst sich mit der Beschreibung der Programmiersprache SCL. SCL eignet sich insbesondere für die Programmierung von komplexen Algorithmen oder für Aufgabenstellungen aus dem Bereich der Datenverwaltung und ergänzt AWL in Richtung höhere Programmiersprache. Die Beschreibung eines Konvertierungsprogramms zur Umsetzung von STEP-5- in STEP-7-Programme sowie eine Gesamtübersicht der Systemfunktionen und des Funktionsvorrats für AWL und SCL runden das Buch ab.

Der Inhalt des vorliegenden Buchs beschreibt die Programmiersoftware STEP 7 in der Version 5.5 und das Optionspaket S7-SCL in der Version 5.3 SP5.

Erlangen, im Oktober 2011

Hans Berger

Der Inhalt des Buchs auf einen Blick

Überblick über das Automatisierungssystem S7-300/400

SPS-Funktionen vergleichbar mit einer Schützensteuerung

Umgang mit Zahlen, Manipulation des Akkumulatorinhalts

Steuerung des Programmablaufs, Bausteinfunktionen

Bearbeitung des Anwenderprogramms

Komplexe Variablen hantieren, indirekte Adressierung

Beschreibung der Programmiersprache SCL

S5/S7-Konverter, Bausteinbibliotheken, Übersichten

Programmbearbeitung

20 Hauptprogramm
Programmstruktur; Zyklussteuerung (Zykluszeit, Reaktionszeit, Startinformation, Hintergrundbearbeitung); Programmfunktionen; DP-Kommunikation; GD-Kommunikation; S7- und S7-Basis-Kommunikation

21 Alarmbearbeitung
Uhrzeitalarme; Verzögerungsalarme; Weckalarme; Prozessalarme; DPV1-Alarme; Mehrprozessoralarm; Alarme hantieren

22 Anlaufverhalten
Kaltstart, Warmstart, Wiederanlauf; STOP, HALT, Urlöschen; Baugruppen parametrieren

23 Fehlerbehandlung
Synchronfehler; Asynchronfehler; Systemdiagnose

Variablenhantierung

24 Datentypen
Aufbau und Struktur (Ablage im Speicher), Deklaration und Anwendung; Elementare Datentypen; Zusammengesetzte Datentypen; anwenderdefinierte Datentypen UDT

25 Indirekte Adressierung
Bereichszeiger, DB-Zeiger, ANY-Zeiger; speicher- und registerindirekte Adressierung (bereichsintern und bereichsübergreifend); Arbeiten mit Adressregistern

26 Direkter Variablenzugriff
Variablenadresse laden Datenablage (Struktur) von Variablen im Speicher; Datenablage bei der Parameterübergabe; „Variabler“ ANY-Zeiger; Kurzbeschreibung „Beispiel Telegramm“

Structured Control Language SCL

27 Einführung, Sprachelemente
Adressierung, Operatoren, Ausdrücke, Wertzuweisungen

28 Kontrollanweisungen
IF, CASE, FOR, WHILE, REPEAT, CONTINUE, EXIT, GOTO, RETURN

29 SCL-Bausteinaufrufe
Funktionswert; OK-Variable, EN/ENO-Mechanismus, Beschreibung Beispiele

30 SCL-Standardfunktionen
Zeitfunktionen; Zählfunktionen; Konvertierungs- und Mathematische Funktionen; Schieben und Rotieren

31 IEC-Funktionen
Konvertierungs- und Vergleichsfunktionen; STRING-Funktionen; Datum/Uhrzeit-Funktionen; Numerische Funktionen

Anhang

32 S5/S7-Konverter
Konvertierung vorbereiten; STEP 5-Programme konvertieren; Nachbearbeiten

33 Bausteinbibliotheken
Organisationsbausteine; Systembausteine; IEC-Funktionen; S5/S7-Konverterfunktionen; TI/S7-Konverterfunktionen; Regelungsfunktionen; DP-Funktionen

34 Operationsübersicht AWL
Basisfunktionen; Digitalfunktionen; Programmfluss-Steuerung; Indirekte Adressierung

35 Anweisungs- und Funktionsübersicht SCL
Operatoren; Kontrollanweisungen; Bausteinaufrufe; Standardfunktionen

Die Programmierbeispiele auf einen Blick

Das vorliegende Buch bietet viele Abbildungen zur Darstellung und Anwendung der Programmiersprachen AWL und SCL. Alle im Buch gezeigten Programmteile sowie zusätzliche Beispiele finden Sie in den zwei Bibliotheken AWL_Buch und SCL_Buch, die Sie aus dem Download-Bereich von der Website des Verlags herunterladen können:

www.publicis.de/books

Die Bibliothek AWL_Buch enthält acht Programme, die im Wesentlichen Anschauungsbeispiele zur AWL-Darstellung sind. Zwei umfangreichere Beispiele zeigen die Programmierung von Funktionen, Funktionsbausteinen und Lokalinstanzen (Beispiel Fördertechnik) und den Umgang mit Daten (Beispiel Telegramm). Alle Beispiele liegen als Quelle vor und enthalten Symbolik und Kommentare.

Bibliothek AWL_Buch

Basisfunktionen Beispiele zur AWL-Darstellung	**Programmbearbeitung** Beispiele SFC-Aufrufe
FB 104 Kapitel 4: Binäre Verknüpfungen FB 105 Kapitel 5: Speicherfunktionen FB 106 Kapitel 6: Übertragungsfunktionen FB 107 Kapitel 7: Zeitfunktionen FB 108 Kapitel 8: Zählfunktionen	FB 120 Kapitel 20: Hauptprogramm FB 121 Kapitel 21: Alarmbearbeitung FB 122 Kapitel 22: Anlaufverhalten FB 123 Kapitel 23: Fehlerbehandlung
Digitalfunktionen Beispiele zur AWL-Darstellung	**Variablenhantierung** Beispiele zu Datentypen und Variablenbearbeitung
FB 109 Kapitel 9: Vergleichsfunktionen FB 110 Kapitel 10: Arithmetische Funktionen FB 111 Kapitel 11: Mathematische Funktionen FB 112 Kapitel 12: Umwandlungsfunktionen FB 113 Kapitel 13: Schiebefunktionen FB 114 Kapitel 14: Wortverknüpfungen	FB 124 Kapitel 24: Datentypen FB 125 Kapitel 25: Indirekte Adressierung FB 126 Kapitel 26: Direkter Variablenzugriff FB 101 Elementare Datentypen FB 102 Zusammengesetzte Datentypen FB 103 Parametertypen
Programmfluss-Steuerung Beispiele zur AWL-Darstellung	**Beispiel Fördertechnik** Beispiele zu Basisfunktionen und Lokalinstanzen
FB 115 Kapitel 15: Statusbits FB 116 Kapitel 16: Sprungfunktionen FB 117 Kapitel 17: Master Control Relay FB 118 Kapitel 18: Bausteinfunktionen FB 119 Kapitel 19: Bausteinparameter Quellprogramm Bausteinprogrammierung (Kapitel 3)	FC 11 Förderbandsteuerung FC 12 Fördergutzähler FB 20 Zuförderung FB 21 Förderband FB 22 Stückgutzähler
Beispiel Telegramm Beispiele zum Umgang mit Daten	**Beispiele allgemein**
UDT 51 Datenstruktur Header UDT 52 Datenstruktur Telegramm FB 51 Telegramm generieren FB 52 Telegramm speichern FC 61 Uhrzeit abfragen FC 62 Prüfsumme bilden FC 63 Datum konvertieren	FC 41 Bereichsüberwachung FC 42 Grenzwertmeldung FC 43 Zinseszins-Rechnung FC 44 Doppelwortweise Flankenauswertung FC 45 Wandlung S5-Gleitpunkt nach S7-REAL FC 46 Wandlung S7-REAL nach S5-Gleitpunkt FC 47 Datenbereich kopieren (ANY-Zeiger)

Die Bibliothek SCL_Buch enthält fünf Programme mit Darstellungen der SCL-Anweisungen und SCL-Funktionen. Die Programme „Beispiel Fördertechnik“ und „Beispiel Telegramm“ zeigen die gleichen Funktionen wie die gleichnamigen AWL-Beispiele. Das Programm „Beispiel allgemein“ enthält SCL-Funktionen zur Bearbeitung zusammengesetzter Datentypen, Datenspeicherung und – für SCL-Programmierer – eine Anweisung zum Programmieren einfacher AWL-Funktionen für SCL-Programme.

Zum Ausprobieren richten Sie ein Projekt ein, das der Ihnen vorliegenden Hardware-Konfiguration entspricht und kopieren das Programm einschließlich der Symboltabelle von der Bibliothek in das Projekt. Nun können Sie die Beispielprogramme aufrufen, für eigene Anwendungen abwandeln und online testen.

Bibliothek SCL_Buch

27 Sprachelemente Beispiele zur SCL-Darstellung (Kapitel 27)	**30 SCL-Funktionen** Beispiele zur SCL-Darstellung (Kapitel 30)
FC 271 Beispiel Begrenzer OB 1 Hauptprogramm zum Beispiel Begrenzer FB 271 Operatoren, Ausdrücke, Zuweisungen FB 272 Indirekte Adressierung	FB 301 Zeitfunktionen FB 302 Zählfunktionen FB 303 Konvertierungsfunktionen FB 304 Mathematische Funktionen FB 305 Schieben und Rotieren
28 Kontrollanweisungen Beispiele zur SCL-Darstellung (Kapitel 28)	**31 IEC-Funktionen** Beispiele zur SCL-Darstellung (Kapitel 31)
FB 281 IF-Anweisung FB 282 CASE-Anweisung FB 283 FOR-Anweisung FB 284 WHILE-Anweisung FB 285 REPEAT-Anweisung	FB 311 Konvertierungsfunktionen FB 312 Vergleichsfunktionen FB 313 Stringfunktionen FB 314 Datum/Uhrzeit-Funktionen FB 315 Numerische Funktionen
29 SCL-Bausteinaufrufe Beispiele zur SCL-Darstellung (Kapitel 29)	**Beispiele allgemein**
FC 291 FC-Baustein mit Funktionswert FC 292 FC-Baustein ohne Funktionswert FB 291 FB-Baustein FB 292 Beispielaufrufe für FC- und FB-Bausteine FC 293 FC-Baustein für EN/ENO-Beispiel FB 293 FB-Baustein für EN/ENO-Beispiel FB 294 Aufrufe für EN/ENO-Beispiele	FC 61 DT_TO_STRING FC 62 DT_TO_DATE FC 63 DT_TO_TOD FB 61 Variablenlänge FB 62 Prüfsumme FB 63 Ringpuffer FB 64 Fallregister AWL-Funktionen für SCL selbst programmiert
Beispiel Fördertechnik Beispiele zu Basisfunktionen und Lokalinstanzen	**Beispiel Telegramm** Beispiele zum Umgang mit Daten
FC 11 Förderbandsteuerung FC 12 Fördergutzähler FB 20 Zuförderung FB 21 Förderband FB 22 Stückgutzähler	UDT 51 Datenstruktur Header UDT 52 Datenstruktur Telegramm FB 51 Telegramm generieren FB 52 Telegramm speichern FC 51 Uhrzeit abfragen

Automatisieren mit STEP 7

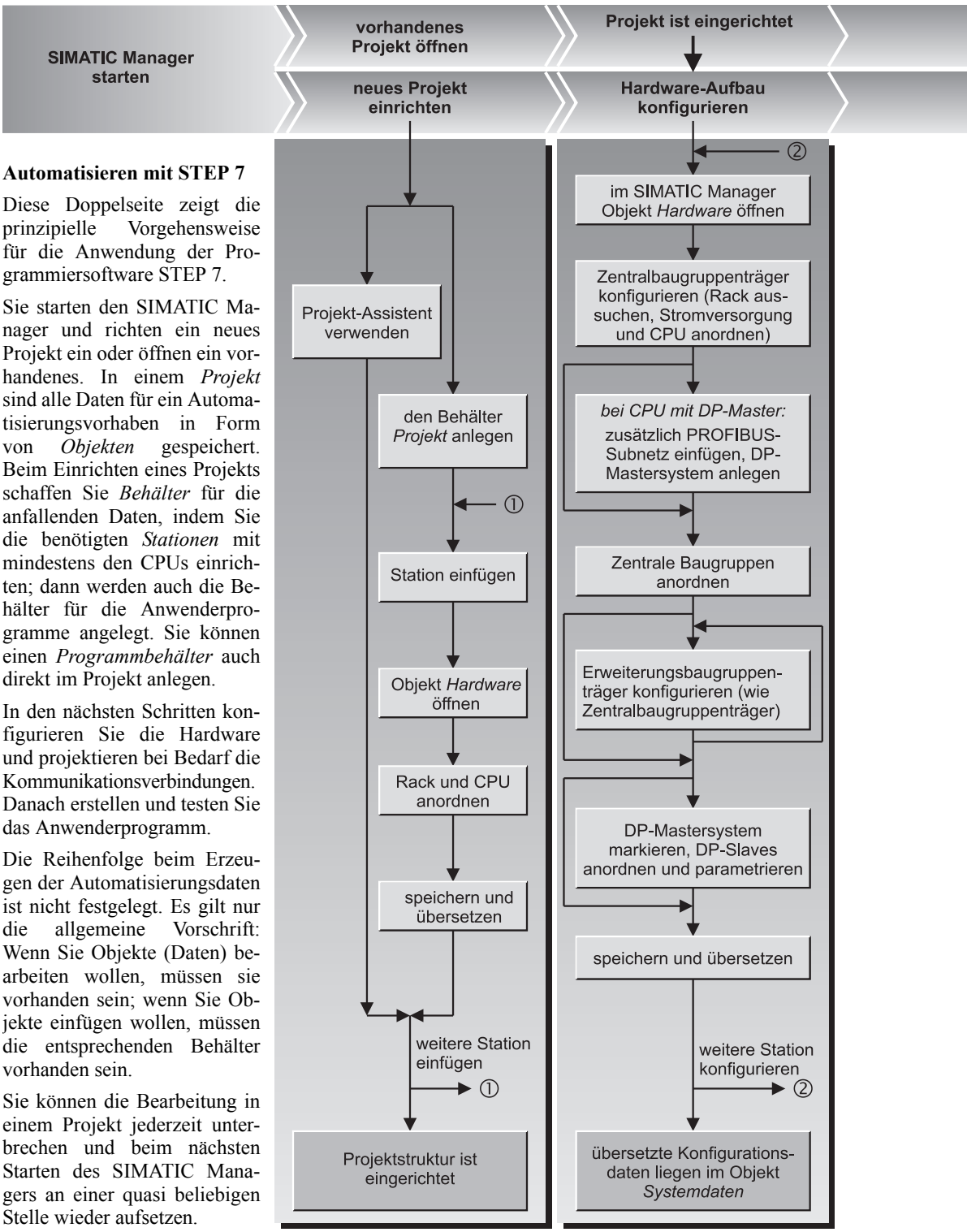
Diese Doppelseite zeigt die prinzipielle Vorgehensweise für die Anwendung der Programmiersoftware STEP 7.

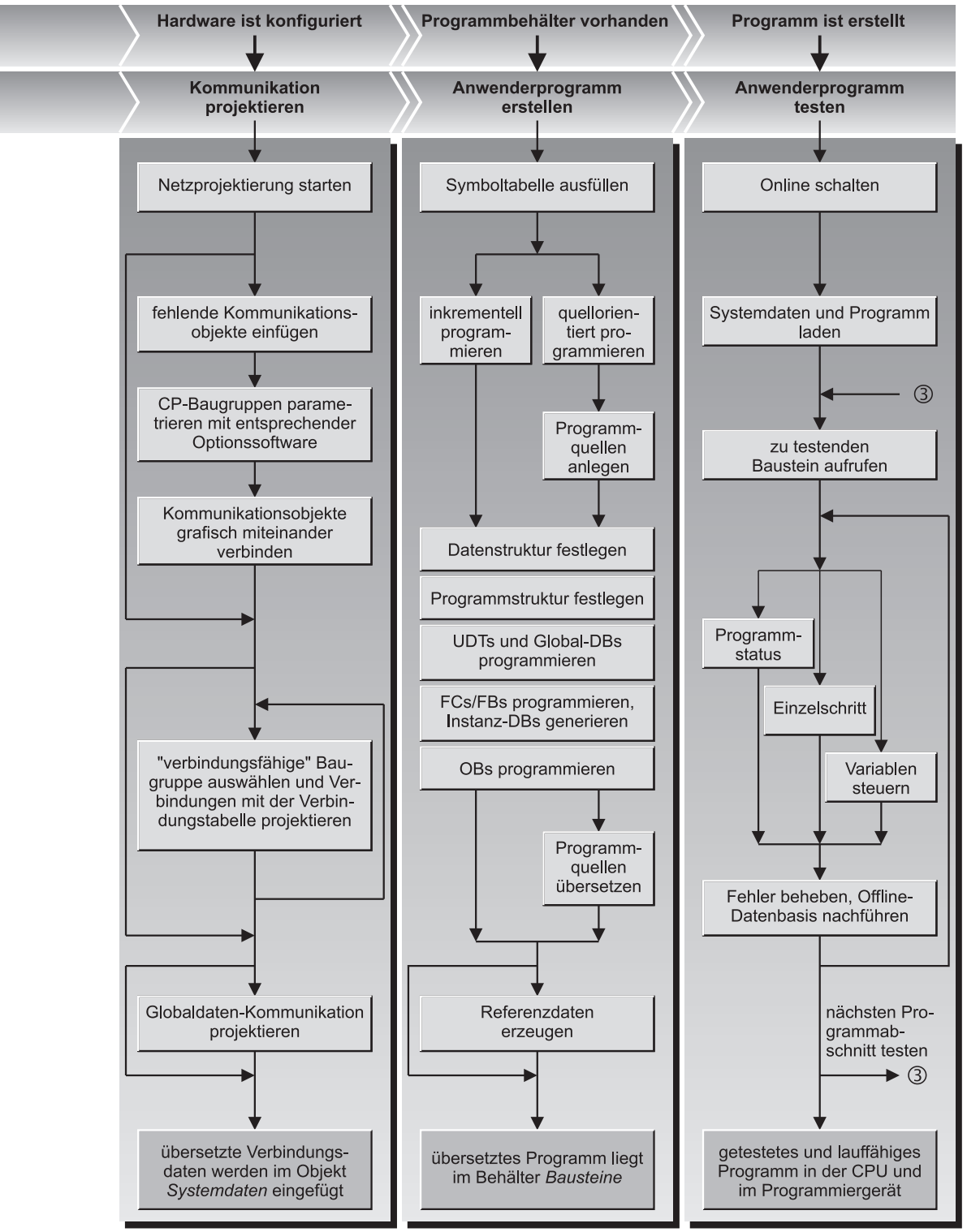
Sie starten den SIMATIC Manager und richten ein neues Projekt ein oder öffnen ein vorhandenes. In einem *Projekt* sind alle Daten für ein Automatisierungsvorhaben in Form von *Objekten* gespeichert. Beim Einrichten eines Projekts schaffen Sie *Behälter* für die anfallenden Daten, indem Sie die benötigten *Stationen* mit mindestens den CPUs einrichten; dann werden auch die Behälter für die Anwenderprogramme angelegt. Sie können einen *Programmbehälter* auch direkt im Projekt anlegen.

In den nächsten Schritten konfigurieren Sie die Hardware und projektieren bei Bedarf die Kommunikationsverbindungen. Danach erstellen und testen Sie das Anwenderprogramm.

Die Reihenfolge beim Erzeugen der Automatisierungsdaten ist nicht festgelegt. Es gilt nur die allgemeine Vorschrift: Wenn Sie Objekte (Daten) bearbeiten wollen, müssen sie vorhanden sein; wenn Sie Objekte einfügen wollen, müssen die entsprechenden Behälter vorhanden sein.

Sie können die Bearbeitung in einem Projekt jederzeit unterbrechen und beim nächsten Starten des SIMATIC Managers an einer quasi beliebigen Stelle wieder aufsetzen.




Hardware ist konfiguriert
Programmbehälter vorhanden
Programm ist erstellt
Kommunikation projektieren
Anwenderprogramm erstellen
Anwenderprogramm testen
Netzprojektierung starten
fehlende Kommunikations-objekte einfügen
CP-Baugruppen parame-trieren mit entsprechender Optionssoftware
Kommunikationsobjekte grafisch miteinander verbinden
"verbindungsfähige" Bau-gruppe auswählen und Ver-bindungen mit der Verbin-dungstabelle projektieren
Globaldaten-Kommunikation projektieren
übersetzte Verbindungs-daten werden im Objekt Systemdaten eingefügt
Symboltabelle ausfüllen
inkrementell program-mieren
quellorien-tiert pro-grammieren
Programm-quellen anlegen
Datenstruktur festlegen
Programmstruktur festlegen
UDTs und Global-DBs programmieren
FCs/FBs programmieren, Instanz-DBs generieren
OBs programmieren
Programm-quellen übersetzen
Referenzdaten erzeugen
übersetztes Programm liegt im Behälter Bausteine
Online schalten
Systemdaten und Programm laden
③
zu testenden Baustein aufrufen
Programm-status
Einzelschritt
Variablen steuern
Fehler beheben, Offline-Datenbasis nachführen
nächsten Pro-grammab-schnitt testen
③
getestetes und lauffähiges Programm in der CPU und im Programmiergerät

Inhaltsverzeichnis

Einführung

Dieser Teil des Buches stellt Ihnen das Automatisierungssystem SIMATIC S7-300/400 im Überblick vor.

Das **Automatisierungssystem S7-300/400** ist modular aus Baugruppen aufgebaut. Die Baugruppen können zentral (in der Nähe der CPU) oder dezentral vor Ort in der Anlage angeordnet sein, ohne dass Sie hierfür besondere Einstellungen und Parametrierungen benötigen. Die dezentrale Peripherie ist bei SIMATIC S7 integraler Bestandteil des Systems. Die Zentralbaugruppe mit ihren verschiedenen Speicherbereichen bildet die hardwaremäßige Grundlage für die Bearbeitung der Anwenderprogramme. Ein Ladespeicher enthält das komplette Anwenderprogramm; die ablaufrelevanten Programmteile befinden sich in einem Arbeitsspeicher, der mit kurzen Zugriffszeiten Voraussetzung für eine schnelle Programmbearbeitung ist.

STEP 7 ist die Programmiersoftware für S7-300/400, das Werkzeug zum Automatisieren ist der SIMATIC Manager. Er ist eine Applikation für die Windows-Betriebssysteme von Microsoft und enthält alle Funktionen zum Einrichten eines Projekts. Bei Bedarf startet der SIMATIC Manager weitere Werkzeuge, um z.B. Stationen zu konfigurieren, Baugruppen zu parametrieren, Programme zu erstellen und zu testen.

Mit den Programmiersprachen von STEP 7 formulieren Sie Ihre Automatisierungslösung. Das **SIMATIC-S7-Programm** ist strukturiert aufgebaut, d.h. es besteht aus Bausteinen mit abgegrenzter Funktion, unterteilt nach Netzwerken und Anweisungen. Verschiedene Prioritätsklassen gestatten eine abgestufte Unterbrechbarkeit des gerade laufenden Anwenderprogramms. STEP 7 arbeitet mit Variablen unterschiedlicher Datentypen, angefangen von Binärvariablen (Datentyp BOOL) über Digitalvariablen (z.B. mit den Datentypen INT oder REAL für Rechenaufgaben) bis zu zusammengesetzten Datentypen wie Feldern oder Strukturen (Zusammenfassung von Variablen verschiedener Datentypen zu einer einzigen Variablen).

Das erste Kapitel enthält einen Überblick über die Hardware des Automatisierungssystems S7-300/400, das zweite Kapitel den gleichen Überblick über die Programmiersoftware STEP 7. Grundlage der Beschreibung ist der Funktionsumfang für STEP 7 Version 5.5.

Das Kapitel 3 „SIMATIC S7-Programm“ führt Sie in die wesentlichen Elemente eines S7-Programms ein und zeigt die Programmierung einzelner Bausteine in den Programmiersprachen AWL und SCL. In den weiteren Kapiteln des Buchs werden dann ausführlich die Funktionen und Anweisungen von AWL und SCL beschrieben. Alle Beschreibungen sind durch kurze Darstellungsbeispiele erläutert.

1 Automatisierungssystem SIMATIC S7-300/400
Aufbau des Automatisierungssystems; dezentrale Peripherie; Kommunikation; Baugruppenadressen; Operandenbereiche

2 Programmiersoftware STEP 7
SIMATIC Manager; Projekt bearbeiten; Station konfigurieren; Netz projektieren; Programm erstellen (Symboltabelle, Programmeditor); Online schalten; Programm testen

3 SIMATIC S7-Programm
Programmbearbeitung mit Prioritätsklassen; Programmaufbau aus Bausteinen; Variablen adressieren; Bausteine mit AWL und SCL programmieren; Variablen und Konstanten; Datentypen (Übersicht)

1 Automatisierungssystem SIMATIC S7-300/400

1.1 Aufbau des Automatisierungssystems

1.1.1 Komponenten

Das Automatisierungssystem SIMATIC S7-300/400 ist eine modular aufgebaute speicherprogrammierbare Steuerung, die aus folgenden Komponenten besteht:

- Baugruppenträger (Racks);
  nehmen die Baugruppen auf und verbinden sie untereinander
- Stromversorgung (PS);
  liefert die internen Versorgungsspannungen
- Zentralbaugruppe (CPU);
  speichert und bearbeitet das Anwenderprogramm
- Anschaltungsbaugruppen (IM);
  verbinden die Baugruppenträger untereinander
- Signalbaugruppen (SM);
  passen die Signale der Anlage an den internen Signalpegel an oder steuern Stellgeräte über digitale und analoge Signale
- Funktionsbaugruppen (FM);
  bearbeiten komplexe oder zeitkritische Prozesse unabhängig von der Zentralbaugruppe
- Kommunikationsbaugruppen (CP)
  stellen den Anschluss zu Subnetzen her
- Subnetze
  verbinden die Automatisierungssysteme untereinander oder mit anderen Geräten

Ein Automatisierungssystem (eine Station) kann aus mehreren Baugruppenträgern bestehen, die untereinander über Buskabel verbunden sind. Im Zentralbaugruppenträger stecken die Stromversorgung, die CPU und Peripheriebaugruppen (SM, FM und CP). Reicht der Platz im Zentralbaugruppenträger für die Peripheriebaugruppen nicht aus oder möchte man räumlich entfernt zum Zentralbaugruppenträger Peripheriebaugruppen anordnen, stehen Erweiterungsbaugruppenträger zur Verfügung, die über Anschaltungsbaugruppen die Verbindung zum Zentralbaugruppenträger herstellen (Bild 1.1). Zusätzlich besteht die Möglichkeit, dezentrale Peripherie an eine Station anzuschließen (siehe Kapitel 1.2.1 „PROFIBUS DP“).

Die Baugruppenträger verbinden die Baugruppen mit zwei Bussen: dem Peripheriebus (P-Bus) und dem Kommunikationsbus (K-Bus). Der P-Bus ist für den schnellen Signalaustausch von Ein- und Ausgabesignalen ausgelegt, der K-Bus für den Austausch größerer Datenmengen. Der K-Bus verbindet die CPU und die Programmiergeräteschnittstelle (MPI) mit Funktionsbaugruppen und Kommunikationsbaugruppen.

1.1.2 S7-300-Station

Zentraler Aufbau

Bei S7-300 können bis zu 8 Peripheriebaugruppen im Zentralbaugruppenträger gesteckt werden. Reicht dieser einzeilige Aufbau nicht aus, können Sie ab CPU 313

- entweder einen zweizeiligen Aufbau wählen (mit IM 365 bis zu 1 m)
- oder einen bis zu vierzeiligen Aufbau wählen (mit IM 360 und IM 361 bis zu 10 m zwischen den Baugruppenträgern)

Sie können maximal 8 Baugruppen in einem Baugruppenträger betreiben. Die Anzahl der Baugruppen wird durch den maximal zulässigen Strom pro Baugruppenträger von 1,2 A begrenzt.

Die Baugruppen werden durch einen seriellen Rückwandbus miteinander verbunden, der die Funktionen des P-Busses und des K-Busses vereint.

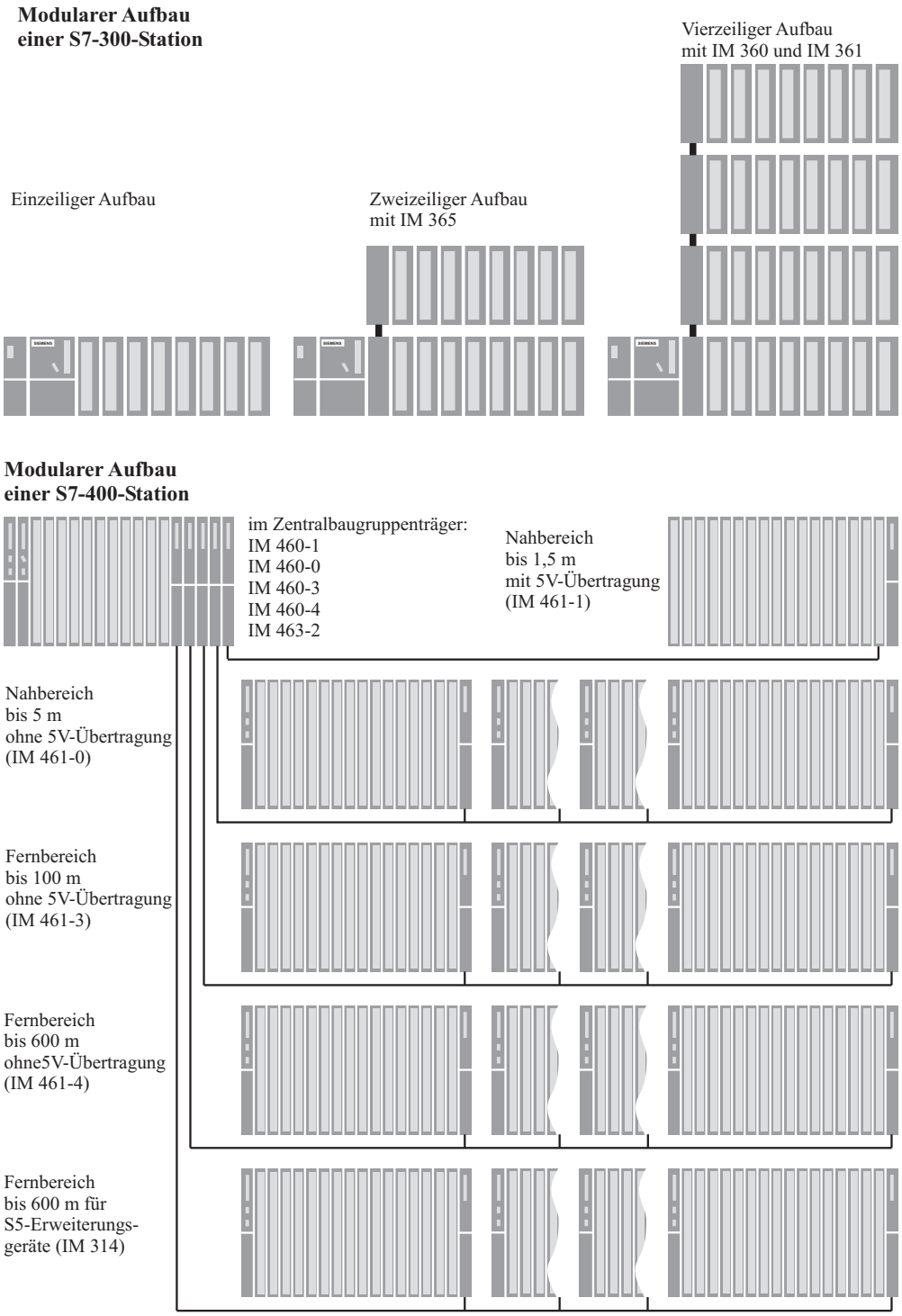


Bild 1.1 Hardware-Aufbau S7-300/400

Lokalbussegment

Eine Besonderheit bezüglich der Baugruppenanordnung stellt der Einsatz der Applikationsbaugruppe FM 356 dar. Eine FM 356 ist in der Lage, den Rückwandbus eines Baugruppenträgers „aufzutrennen“ und die Ansteuerung der restlichen Baugruppen im abgetrennten „Lokalbus-Segment“ selbst zu übernehmen. Für die Baugruppenanzahl und die Stromaufnahme gelten auch in diesem Fall die oben genannten Begrenzungen.

Standard-CPUs

Die Standard-CPUs gibt es in verschiedenen Ausführungen bezüglich Speicherumfang und Verarbeitungsgeschwindigkeit. Sie reichen von der „kleinsten“ CPU 312 für kleinere Anwendungen mit moderaten Anforderungen an die Verarbeitungsgeschwindigkeit bis zur CPU 319-3 PN/DP mit großem Programmspeicher und hoher Programmbearbeitungsleistung für branchenübergreifende Automatisierungsaufgaben. Mit den entsprechenden Schnittstellen ausgestattet können einige CPUs als Zentrale für die dezentrale Peripherie über PROFIBUS und PROFINET eingesetzt werden.

Zum Betrieb der Standard-CPUs ist – wie bei allen innovierten S7-300-CPUs – eine Micro Memory Card (MMC) erforderlich. Dieses Speichermedium eröffnet neue Anwendungsmöglichkeiten gegenüber der bisher eingesetzten Memory Card (siehe Kapitel 1.1.6 „Speicherbereiche der Zentralbaugruppe“).

Die mittlerweile abgekündigte CPU 318 kann durch die CPUs 317 oder 319 ersetzt werden.

Kompakt-CPUs

Die CPUs 3xxC gestatten einen kompakten Aufbau von Kleinsteuerungen. Je nach Ausführung enthalten sie bereits

- integrierte Peripherie
  Digital- und Analog-Ein/Ausgänge
- integrierte technologische Funktionen
  Zählen, Messen, Regeln, Positionieren
- integrierte Kommunikationsschnittstellen
  PROFIBUS-DP-Master oder -Slave, Punkt-zu-Punkt-Kopplung (PtP)

Die technologischen Funktionen sind Systembausteine, die die On-Board-Peripherie der Zentralbaugruppe verwenden.

Technologie-CPUs

Die CPUs 3xxT vereinen Steuerungsfunktionen mit einfachen Motion-Control-Funktionen. Der Steuerungsanteil ist wie bei einer Standard-CPU ausgelegt. Er wird mit STEP 7 projektiert, parametriert und programmiert. Die Technologieobjekte und der Motion-Control-Teil benötigen das Optionspaket S7-Technology, das nach der Installation im SIMATIC Manager integriert ist.

Die Technologie-CPUs besitzen eine PROFIBUS-DP-Schnittstelle, die den Betrieb als DP-Master oder DP-Slave gestattet. Die CPUs werden für branchenübergreifende Automatisierungsaufgaben im Serienmaschinen-, Sondermaschinen- und Anlagenbau eingesetzt.

Fehlersichere CPUs

Die CPUs 3xxF werden in Fertigungsanlagen mit erhöhten Sicherheitsanforderungen eingesetzt. Entsprechende PROFIBUS- und PROFINET-Schnittstellen gestatten das Betreiben von sicherheitsgerichteter dezentraler Peripherie über das Busprofil PROFIsafe (siehe „Safety Integrated für die Fertigungsindustrie“ unter 1.1.5 „Sicherheitsgerichtete SIMATIC“). Parallel zum sicherheitsgerichteten Betrieb können Standardbaugruppen für normale Anwendungen eingesetzt werden.

SIPLUS

Die Produktfamilie SIPLUS bietet Baugruppen, die in rauer Umgebung eingesetzt werden können. Die Basis der SIPLUS-Komponenten sind Standardgeräte, die speziell für den jeweiligen Einsatz umgerüstet werden, beispielsweise für einen erweiterten Temperaturbereich, erhöhte Rüttel- und Stoßfestigkeit oder verschiedene vom Standard abweichende Spannungsbereiche. Beachten Sie deshalb bitte die technischen Daten der jeweiligen SIPLUS-Baugruppe. Zur Projektierung mit STEP 7 verwenden Sie den Äquivalenztyp (die zugrunde liegende Standardbaugruppe), der beispielsweise auf dem Typenschild der Baugruppe angegeben ist.

1.1.3 S7-400-Station

Zentraler Aufbau

Die Zentralbaugruppenträger bei S7-400 gibt es in den Ausführungen UR1 (18 Steckplätze), UR2 (9 Steckplätze) und CR3 (4 Steckplätze). UR1 und UR2 können auch als Erweiterungsbaugruppenträger eingesetzt werden. In den Baugruppenträgern belegen auch die Stromversorgung und die CPU Steckplätze, evtl. sogar zwei oder mehr pro Baugruppe. Bei Bedarf wird mit Erweiterungsbaugruppenträgern die verfügbare Anzahl an Steckplätzen erhöht: UR1 und ER1 haben je 18 Steckplätze, UR2 und ER2 je 9.

Mit den Anschaltungen IM 460-1 und IM 461-1 kann pro Schnittstelle ein Erweiterungsbaugruppenträger bis 1,5 m entfernt vom Zentralbaugruppenträger angeordnet werden, hierbei wird die 5V-Versorgungsspannung mit übertragen. Ebenfalls im Nahbereich bis 5 m können bis zu 4 Erweiterungsbaugruppenträger über IM 460-0 und 461-0 betrieben werden. Im Fernbereich schließlich ist es mit IM 460-3 und IM 461-3 bzw. IM 460-4 und 461-4 möglich, bis zu 4 Erweiterungsbaugruppenträger bis zu 100 m bzw. 600 m entfernt zu betreiben.

Maximal sind bis zu 21 Erweiterungsbaugruppenträger an einen Zentralbaugruppenträger anschließbar. Zur Unterscheidung stellen Sie die Nummer des Baugruppenträgers am Kodierschalter der Empfangs-IM ein.

Der Rückwandbus besteht aus einem parallelen P-Bus und einem seriellen K-Bus. Die Erweiterungsbaugruppenträger ER1 und ER2 sind für „einfache“ Signalbaugruppen ausgelegt, die keine Prozessalarme auslösen, nicht mit 24 V über den P-Bus versorgt werden müssen, keine Pufferspannung benötigen und keinen K-Bus-Anschluss haben. Der K-Bus ist in den Baugruppenträgern UR1, UR2 und CR2 dann vorhanden, wenn sie entweder als Zentralbaugruppenträger oder als Erweiterungsbaugruppenträger mit den Nummern 1 bis 6 verwendet werden.

Segmentierter Baugruppenträger

Eine Besonderheit stellt der segmentierte Baugruppenträger CR2 dar. Sie können hier zwei CPUs in einem Baugruppenträger mit gemeinsamer Stromversorgung funktionsmäßig getrennt betreiben. Beide CPUs können über den K-Bus miteinander Daten austauschen, haben jedoch vollständig getrennte P-Busse für ihre eigenen Signalbaugruppen.

Mehrprozessorbetrieb

Bei S7-400 können sich in einem Zentralbaugruppenträger UR bis zu 4 entsprechend ausgelegte CPUs am Mehrprozessorbetrieb beteiligen. Jede Baugruppe in dieser Station wird einer einzigen CPU zugeordnet, sowohl mit ihrer Adresse als auch mit ihren Alarmen. Näheres siehe Kapitel 20.3.6 „Mehrprozessorbetrieb“ und 21.7 „Mehrprozessoralarm“.

Anschluss von SIMATIC S5-Baugruppen

Mit der Anschaltungsbaugruppe IM 463-2 können Sie S5-Erweiterungsgeräte (EG 183U, EG 185U, EG 186U sowie ER 701-2 und ER 701-3) an eine S7-400 anschließen und diese Erweiterungsgeräte wiederum zentral erweitern. Im S5-Erweiterungsgerät übernimmt eine IM 314 die Kopplung. Sie können alle in den angegebenen Erweiterungsgeräten zugelassenen Digital- und Analogbaugruppen betreiben. Eine S7-400 kann maximal 4 IM 463-2 aufnehmen; an jeder der beiden Schnittstellen einer IM 463-2 ist der Anschluss von maximal vier S5-Erweiterungsgeräten möglich.

1.1.4 Hochverfügbare SIMATIC

Für Anwendungen mit hohen Anforderungen an die Ausfallsicherheit von Maschinen und Prozessen gibt es bei SIMATIC S7 hochverfügbare Automatisierungssysteme mit redundant ausgelegtem Ausbau in zwei Ausführungen: Software-Redundanz und S7-400H/FH.

Software-Redundanz

Mit SIMATIC S7-300/400-Standard-Komponenten können Sie ein auf Softwarebasis redundantes System aufbauen, in dem eine Masterstation den Prozess steuert und, bei deren Ausfall, eine Reservestation die Steuerung übernimmt.

Hochverfügbarkeit durch Software-Redundanz eignet sich für langsame Prozesse, denn die

Umschaltung auf die Reservestation kann je nach Ausbau der Automatisierungsgeräte mehrere Sekunden benötigen. Während dieser Zeit sind die Prozesssignale „eingefroren“. Danach arbeitet die Reservestation mit den zuletzt in der Masterstation gültigen Daten weiter.

Die Redundanz der Ein-/Ausgabebaugruppen wird mit dezentraler Peripherie realisiert (ET 200M mit Anschaltung IM 153-2 für redundant aufgebautem PROFIBUS DP). Die Software-Redundanz kann mit STEP 7 ab Version 5.2 projektiert werden.

Hochverfügbare SIMATIC S7-400H

SIMATIC S7-400H ist ein hochverfügbares Automatisierungssystem mit redundant ausgelegtem Aufbau aus zwei Zentralbaugruppenträgern mit je einer H-CPU und einem Synchronisationsmodul zum Datenabgleich über Lichtwellenleiter. Beide Geräte arbeiten „hot standby“; im Fehlerfall übernimmt das intakte Gerät durch automatische stoßfreie Umschaltung die alleinige Bearbeitung. Der Baugruppenträger UR2-H mit zweimal neun Steckplätzen bietet die Möglichkeit, ein hochverfügbares System auch in einem einzigen Baugruppenträger aufzubauen.

Die Peripherie kann entweder normal verfügbar (einkanaliger einseitiger Aufbau) oder erhöht verfügbar (einkanaliger geschalteter Aufbau mit ET 200M) aufgebaut sein. Die Kommunikation erfolgt mit einfachem oder mit redundantem Bus.

Das Anwenderprogramm ist das gleiche wie für ein nicht redundantes Gerät; die Redundanzfunktion wird ausschließlich und vom Anwender verborgen durch die eingesetzte Hardware erbracht. Das für die Projektierung erforderliche Softwarepaket ist in STEP 7 ab V5.3 enthalten. Die mitgelieferten Standardbibliotheken *Redundant IO* enthalten Bausteine zur Unterstützung der redundanten Peripherie.

1.1.5 Sicherheitsgerichtete SIMATIC

Fehlersichere Automatisierungssysteme steuern Prozesse, in denen der sichere Zustand durch unmittelbare Abschaltung erreicht werden kann. Sie werden in Anlagen mit erhöhten Sicherheitsanforderungen eingesetzt.

Die Sicherheitsfunktionen liegen schwerpunktmäßig im sicherheitsgerichteten Anwenderprogramm einer entsprechend ausgelegten CPU und in den fehlersicheren Ein- und Ausgabe-Baugruppen. Eine F-CPU erfüllt die Sicherheitsanforderungen bis AK 6 nach DIN V 19250/ DIN V VDE 0801, bis SIL 3 nach IEC 61508 und bis Kategorie 4 nach EN 954-1. Die Sicherheitsfunktionen können in der gleichen CPU parallel zu einem nicht sicherheitsgerichtet ausgelegten Anwenderprogramm betrieben werden.

Die sicherheitsgerichtete Kommunikation über PROFIBUS DP – bei S7 Distributed Safety auch über PROFINET IO – verwendet das Busprofil PROFIsafe. Es erlaubt die Übertragung von sicherheitsgerichteten und nicht sicherheitsgerichteten Daten auf einer einzigen Busleitung.

Safety Integrated für die Fertigungsindustrie

S7 Distributed Safety ist ein fehlersicheres Automatisierungssystem mit dem Einsatzschwerpunkt im Bereich Maschinen- und Personalschutz für Maschinensteuerungen und in der Prozessindustrie.

Als F-CPUs stehen Controller aus dem Gerätespektrum von SIMATIC S7-300, S7-400 und ET 200S zur Verfügung. An S7-400 werden die sicherheitsgerichteten E/A-Baugruppen über PROFIBUS DP oder PROFINET IO mit dem sicherheitsgerichteten Busprofil PROFIsafe angeschlossen. Bei S7-300 und ET 200S ist der Betrieb von sicherheitsgerichteten E/A-Baugruppen auch im Zentralbaugruppenträger möglich.

Die Hardware-Projektierung und die Programmierung des nicht sicherheitsgerichteten Anwenderprogramms erfolgt mit den Standard-Applikationen von STEP 7.

Zur Programmierung der sicherheitsgerichteten Programmteile ist das Optionspaket *SIMATIC S7 Distributed Safety* erforderlich. Mit diesem Optionspaket erstellen Sie mit den Programmiersprachen F-KOP oder F-FUP die Bausteine, die das sicherheitsgerichtete Programm enthalten. Die Anbindung an die Peripherie erfolgt wie beim Standardprogramm über das Prozessabbild. S7 Distributed Safety enthält auch eine Bibliothek mit TÜV-zertifizierten Sicherheitsbausteinen. Zusätzlich gibt es eine Bibliothek mit F-Bausteinen für Pressen- und Brenner-Steuerungen.

Das sicherheitsgerichtete Anwenderprogramm kann parallel zum Standard-Anwenderprogramm laufen. Wird im sicherheitsgerichteten Teil des Programms ein Fehler entdeckt, geht die CPU in den Betriebszustand STOP.

Safety Integrated für die Prozessindustrie

S7 F/FH Systems ist ein fehlersicheres Automatisierungssystem auf der Basis von S7-400 mit dem Einsatzschwerpunkt in der Prozessindustrie. Die sicherheitsgerichteten E/A-Baugruppen werden über PROFIBUS DP mit dem sicherheitsgerichteten Busprofil PROFIsafe angeschlossen.

Eine S7-400-F-CPU erhält die sicherheitsgerichteten Steuerungsfunktionen durch den Einsatz einer *S7 F Systems Runtime-Lizenz*. Parallel zum sicherheitsgerichteten Anlagenteil kann ein nicht sicherheitsgerichtetes Anwenderprogramm laufen.

Zusätzlich zur Fehlersicherheit bietet S7-400FH eine erhöhte Verfügbarkeit. Führt ein erkannter Fehler zum STOP der Master-CPU, wird auf die im Hot-Stand-By laufende Reserve-CPU rückwirkungsfrei umgeschaltet. Für den Betrieb als S7-400FH ist zusätzlich das Optionspaket *S7 H Systems* erforderlich.

Die Hardware-Projektierung und die Programmierung des nicht sicherheitsgerichteten Anwenderprogramms erfolgt mit den Standard-Applikationen von STEP 7.

Für die Programmierung der sicherheitsgerichteten Programmteile ist das Optionspaket *S7 F Systems* erforderlich, zusätzlich das Optionspaket *CFC* ab V5.0 SP3 und das Optionspaket *S7-SCL* ab V5.0.

Die Programmierung des sicherheitsgerichteten Programms geschieht mit CFC (Continuous Function Chart, funktionsplanähnliche Verschaltung von Bausteinen). Hierbei werden sicherheitsgerichtet programmierte Funktionsbausteine aus der mitgelieferten F-Bibliothek aufgerufen und verschaltet. Sie enthalten neben Funktionen zur Programmierung von Sicherheitsfunktionen auch Funktionen zur Fehlererkennung und Fehlerreaktion. So wird sichergestellt, dass bei Ausfällen und Fehlern das F-System in einem sicheren Zustand gehalten wird oder in einen sicheren Zustand überführt wird. Wird im Sicherheitsprogramm ein Fehler entdeckt, schaltet sich der sicherheitsgerichtete Teil der Anlage ab, während der restliche Teil noch weiterlaufen kann.

Fehlersichere Peripherie

Für den Sicherheitsbetrieb sind fehlersichere Signalbaugruppen (F-Baugruppen bzw. F-Module) erforderlich. Die Fehlersicherheit wird durch die integrierten Sicherheitsfunktionen und eine entsprechende Verdrahtung der Sensoren und Aktoren erreicht.

Die F-Baugruppen sind auch im Standardbetrieb mit erhöhten Anforderungen an die Diagnose einsetzbar. Sowohl im Standard- als auch im Sicherheitsbetrieb können bei S7 F/FH Systems die F-Baugruppen zur Erhöhung der Verfügbarkeit redundant betrieben werden.

Die fehlersichere Peripherie gibt es in verschiedenen Ausführungen:

- Die fehlersicheren Signalbaugruppen in der S7-300-Bauform werden im dezentralen Peripheriegerät ET 200M oder – bei S7-Distributed Safety – auch zentral eingesetzt.
- Fehlersichere E/A-Module gibt es für die dezentralen Peripheriegeräte mit den Bauformen ET 200S, ET 200pro und ET 200eco.
- Für die dezentralen Peripheriegeräte ET 200S und ET 200pro sind auch fehlersichere Interface-Module als F-CPU erhältlich.
- Es können fehlersichere DP-Norm-Slaves und – bei S7-Distributed Safety auch IO-Norm-Devices –, die das Busprofil PROFIsafe beherrschen, eingesetzt werden.

Fehlersichere CPUs und Signalbaugruppen gibt es auch in SIPLUS-Ausführung.

1.1.6 Speicherbereiche der Zentralbaugruppe

Das Bild 1.2 zeigt die für Ihr Programm wichtigen Speicherbereiche im Programmiergerät, in der Zentralbaugruppe und in den Signalbaugruppen.

Das Programmiergerät enthält die *Offline-Daten*. Sie bestehen aus dem Anwenderprogramm (Programmcode und Anwenderdaten), aus den

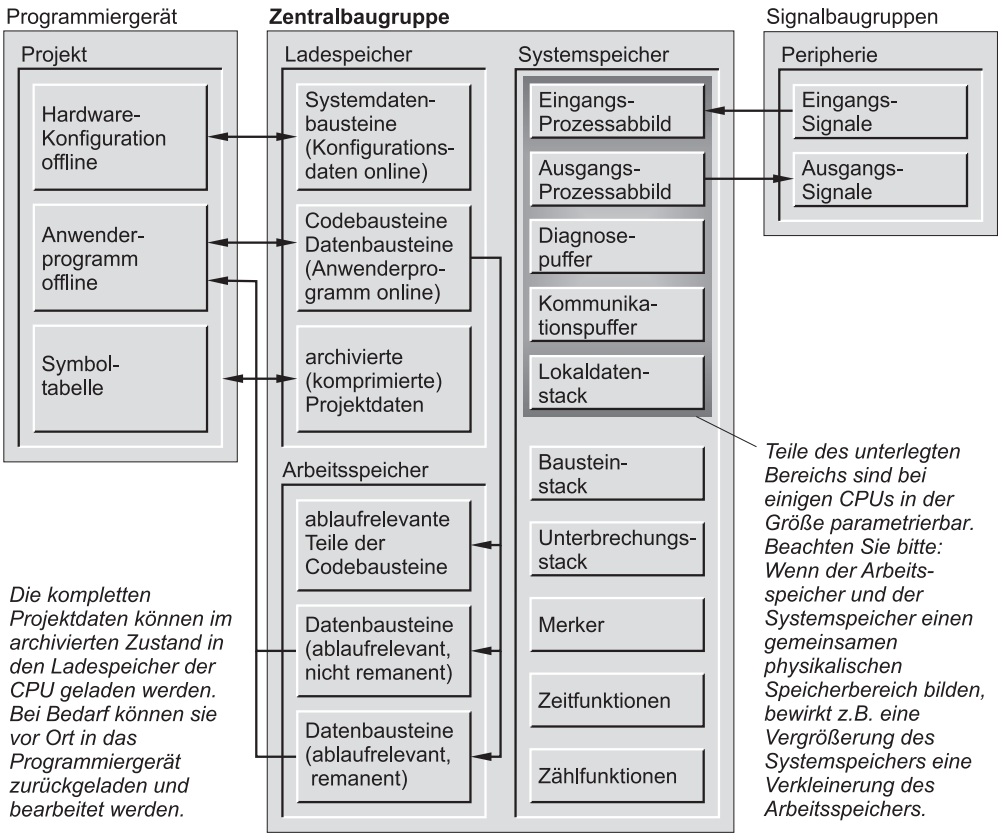


Bild 1.2 Speicherbereiche auf der Zentralbaugruppe

Systemdaten (z.B. Hardware-Projektierung, Netz- und Verbindungsprojektierung) und weiteren projektspezifischen Daten wie z.B. die Symboltabelle und die Kommentare.

Die *Online-Daten* bestehen aus dem Anwenderprogramm und den Systemdaten auf der Zentralbaugruppe, die in zwei Speicherbereichen, dem Ladespeicher und dem Arbeitsspeicher, untergebracht sind. Zusätzlich ist hier noch der Systemspeicher vorhanden.

Die Peripheriebaugruppen schließlich enthalten Speicher für die Signalzustände der Ein- und Ausgänge.

Die Zentralbaugruppen haben einen Schacht für ein steckbares *Speichermodul*. Auf diesem Speichermodul befinden sich der Ladespeicher oder Teile davon (siehe „Physikalische Ausführung des CPU-Speichers“, weiter unten). Das Speichermodul ist als Memory Card (S7-400-CPUs) oder als Micro Memory Card (S7-300-CPUs und davon abgeleitete ET 200-CPUs) ausgeführt. Über das Speichermodul kann auch ein Firmware-Update des CPU-Betriebssystems erfolgen.

Memory Card

Das Speichermodul für die S7-400-CPUs ist die Memory Card (MC). Es gibt zwei Arten: RAM Memory Cards und Flash EPROM Memory Cards.

Wenn Sie ausschließlich den Ladespeicher erweitern wollen, verwenden Sie eine RAM Memory Card. Mit einer RAM Memory Card können Sie das komplette Anwenderprogramm

online ändern. Das ist beispielsweise für größere Programme beim Test und bei der Inbetriebnahme erforderlich. RAM Memory Cards verlieren ihren Inhalt, wenn sie gezogen werden.

Wenn Sie nach Test und Inbetriebnahme das Anwenderprogramm einschließlich der Konfigurationsdaten und Baugruppenparameter auch ohne Pufferbatterie spannungsausfallsicher halten wollen, verwenden Sie eine Flash EPROM Memory Card. Hierbei laden Sie das gesamte Programm offline auf die Flash EPROM Memory Card, wenn sie im Programmiergerät steckt. Bei entsprechend ausgelegten CPUs können Sie das Programm auch online laden, wenn die Memory Card in der CPU steckt.

Micro Memory Card

Das Speichermodul bei den neueren S7-300-CPUs ist eine Micro Memory Card (MMC). Die Daten auf der MMC sind nullspannungsfest abgelegt, können jedoch wie bei einem RAM-Speicher gelesen, geschrieben und gelöscht werden. Dieses Verhalten erlaubt eine Datenpufferung ohne Batterie.

Auf der MMC befindet sich der komplette Ladespeicher, so dass zum Betrieb immer eine MMC erforderlich ist. Die MMC kann als portables Speichermedium für Anwenderprogramme oder Firmware-Updates genutzt werden. Mit speziellen Systemfunktionen lesen oder schreiben Sie vom Anwenderprogramm aus Datenbausteine auf der MMC und können so z.B. Rezepturen von der MMC lesen oder ein Messwertarchiv auf der MMC anlegen und mit Daten versorgen.

Ladespeicher

Im Ladespeicher steht das gesamte Anwenderprogramm einschließlich Konfigurationsdaten (Systemdaten). Vom Programmiergerät aus wird das Anwenderprogramm immer zuerst in den Ladespeicher übertragen und von hier aus in den Arbeitsspeicher. Das Programm im Ladespeicher wird nicht als Steuerungsprogramm bearbeitet.

Bei einer CPU 300 und einer CPU ET 200 liegt der Ladespeicher komplett auf der Micro Memory Card. Somit bleibt der Inhalt des Ladespeichers auch im spannungslosen Zustand der CPU erhalten.

Besteht bei einer CPU 400 der Ladespeicher aus integriertem RAM oder einer RAM Memory Card, ist eine Pufferbatterie erforderlich, um das Anwenderprogramm remanent zu halten. Bei integriertem EEPROM oder bei zusteckbarer Flash EPROM Memory Card als Ladespeicher kann die CPU batterielos betrieben werden.

Ab STEP 7 V5.1 können Sie bei entsprechend ausgelegten CPUs die gesamten Projektdaten als komprimierte Archivdatei im Ladespeicher ablegen (siehe Kapitel 2.2.2 „Verwalten, reorganisieren und archivieren“).

Arbeitsspeicher

Der Arbeitsspeicher ist als schneller, in vollem Umfang in der CPU integrierter RAM-Speicher ausgelegt. Das Betriebssystem der CPU kopiert den „ablaufrelevanten“ Programmcode und die Anwenderdaten in den Arbeitsspeicher. „Ablaufrelevant“ ist eine Eigenschaft der vorhandenen Objekte, nicht gleichbedeutend mit der Tatsache, dass ein bestimmter Codebaustein auch aufgerufen und bearbeitet wird. Das „eigentliche“ Steuerungsprogramm wird im Arbeitsspeicher bearbeitet.

Produktspezifisch kann der Arbeitsspeicher entweder als ein zusammenhängender Bereich oder unterteilt nach Programm- und Datenspeicher und letzterer auch unterteilt nach remanent und nicht remanent ausgebildet sein.

Beim Zurückladen des Anwenderprogramms in das Programmiergerät werden die Bausteine aus dem Ladespeicher geholt, ergänzt um die aktuellen Werte der Datenoperanden aus dem Arbeitsspeicher (weitere Informationen hierzu in den Kapiteln 2.6.4 „Anwenderprogramm in die CPU laden“ und 2.6.5 „Bausteinhantierung“).

Systemspeicher

Der Systemspeicher enthält die Operanden (Variablen), die Sie von Ihrem Programm aus ansprechen. Die Operanden sind zu Bereichen (Operandenbereiche) zusammengefasst, die eine für jede CPU spezifische Menge an Operan-