

Speicherprogrammierbare Steuerungen

SYSMAC CS-Serie

CS1G/H-CPU□□-EV1

CS1G/H-CPU□□H

CS1D-CPU□□H

CS1D-CPU□□S

SYSMAC CJ-Serie

CJ1G-CPU□□

CJ1G/H-CPU□□H

CJ1G-CPU□□P

CJ1M-CPU□□

PROGRAMMIERHANDBUCH



Kurzübersicht

- 1 Betrieb der CPU-Baugruppe
- 2 Programmierung
- 3 Befehlsfunktionen
- 4 Tasks

Advanced Industrial Automation

OMRON

SYSMAC CS-Serie

CS1G/H-CPU□□-EV1

CS1G/H-CPU□□H

CS1D-CPU□□H

CS1D-CPU□□S

SYSMAC CJ-Serie

CJ1G-CPU□□

CJ1G/H-CPU□□H

CJ1G-CPU□□P

CJ1M-CPU□□

Speicherprogrammierbare Steuerungen

Programmierhandbuch

Bearbeitungsstand Juli 2004

Hinweis:

OMRON-Produkte sind zum Gebrauch durch einen qualifizierten Bediener gemäß angemessenen Verfahren und nur zu den in diesem Handbuch beschriebenen Zwecken gefertigt.

In diesem Handbuch werden Sicherheitshinweise nach folgenden Konventionen gekennzeichnet und eingeteilt. Beachten Sie stets die in diesen Hinweisen enthaltenen Informationen. Eine Nichtbeachtung der Sicherheitshinweise kann zu Verletzungen oder Sachschäden führen.

-  **GEFAHR** Kennzeichnet eine bevorstehende gefährliche Situation, die zum Tod oder zu ernsthaften Verletzungen führt, wenn sie nicht vermieden wird.
-  **VORSICHT** Kennzeichnet eine potenziell gefährliche Situation, die zum Tod oder zu schweren Verletzungen führen kann, wenn sie nicht vermieden wird.
-  **Achtung** Kennzeichnet eine potenziell gefährliche Situation, die zu kleineren oder mittelschweren Verletzungen oder Sachschäden führen kann, wenn sie nicht vermieden wird.

OMRON-Produktreferenzen

Die in manchen Anzeigen und auf manchen OMRON-Produkten verwendete Abkürzung „Ch“ bedeutet häufig „Wort“. Die übliche Abkürzung für „Wort“ (im Sinne von 16 Bits) in Anzeigen und auf OMRON-Produkten ist „Wd“.

Die Abkürzung „SPS“ steht für speicherprogrammierbare Steuerung. In manchen Anzeigen von Programmierkonsolen wird jedoch noch die Abkürzung „PC“ für „Programmable Controller“ (Programmierbare Steuerung) verwendet. Dies ist nicht mit der üblichen Bedeutung von PC (Personal Computer) zu verwechseln.

Visuelle Hilfen

Die folgenden Überschriften tauchen in der linken Spalte des Handbuchs auf und helfen Ihnen, verschiedene Arten von Informationen zu finden.

Hinweis Kennzeichnet Informationen von besonderem Interesse für effizienten und zweckmäßigen Betrieb des Produkts.

1,2,3... 1. Kennzeichnet Auflistungen aller Art, z. B. Verfahren oder Checklisten.

© OMRON, 2003

Alle Rechte vorbehalten. Diese Publikation darf ohne vorherige schriftliche Genehmigung von OMRON weder als Ganzes noch in Auszügen in irgendeiner Form oder auf irgendeine Weise, sei es auf mechanischem oder elektronischem Wege oder durch Fotokopieren oder Aufzeichnen, reproduziert, auf einem Datensystem gespeichert oder übertragen werden.

In Bezug auf die hierin enthaltenen Informationen wird keine Patenthaftung übernommen. Da OMRON weiterhin an einer ständigen Verbesserung seiner Qualitätsprodukte arbeitet, sind Änderungen an den in diesem Handbuch enthaltenen Informationen ohne Ankündigung vorbehalten. Bei der Herstellung dieses Handbuchs wurden alle Vorsorgemaßnahmen ergriffen. Dennoch übernimmt OMRON keine Verantwortung für etwaige Fehler und Auslassungen. Es wird keine Haftung für Schäden übernommen, die aus der Nutzung von in diesem Dokument enthaltenen Informationen zurückzuführen sind.

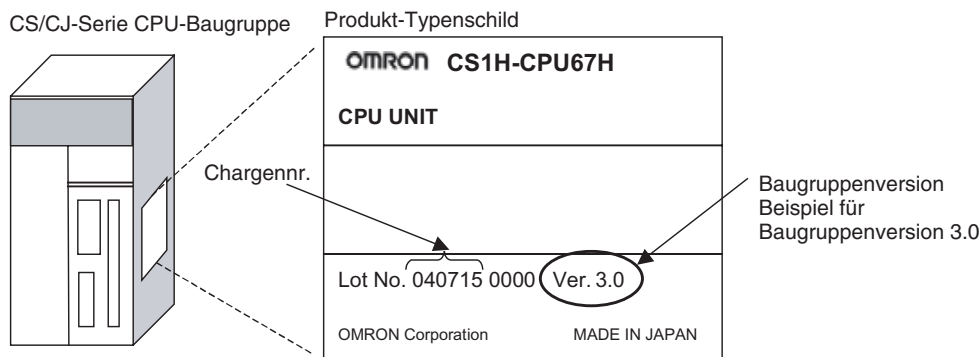
Baugruppenversionen von CPU-Baugruppen der CS/CJ-Serie

Baugruppenversionen

Kennzeichnung der Baugruppenversionen auf den Produkten

Eine „Baugruppenversion“ wurde eingeführt, um CPU-Baugruppen der CS/CJ-Serie entsprechend der Funktionalitätsunterschiede, sich durch Weiterentwicklungen der Baugruppen ergeben, zu kennzeichnen. Dies gilt für die CPU-Baugruppen CS1-H, CJ1-H, CJ1M und CS1D.

Die Baugruppen-Version ist, wie unten gezeigt, rechts von der Lot-Nummer auf dem Typenschild der Produkte angegeben.



- CS1-H, CJ1-H und CJ1M CPU-Baugruppen (außer Low-End-Modelle), die am oder vor dem 4. November 2003 hergestellt wurden, besitzen keine Angabe der Baugruppenversion (d. h. die Stelle für die oben gezeigte Baugruppenversion ist unbeschriftet).
- Die Baugruppen-Versionsnummern der CS1-H, CJ1-H und CJ1M CPU-Baugruppen sowie der CS1D CPU-Baugruppe für Einzel-CPU-Systeme beginnen mit der Version 2.0.
- Die Baugruppen-Versionsnummern der CS1D CPU-Baugruppe für Duplex-CPU-Systeme beginnt mit Version 1.1.
- CPU-Baugruppen, für die keine Baugruppenversion angegeben wird, werden als *Vor Version* □.□ CPU-Baugruppen bezeichnet, wie z. B. CPU-Baugruppen vor Version 2.0 und CPU-Baugruppen vor Version 1.1.

Überprüfung der Baugruppenversionen mit Hilfe von Programmiersoftware

CX-Programmer Version 4.0 kann zur Prüfung der Baugruppen-Version verwendet werden. Dazu gibt es zwei Möglichkeiten.

- Verwendung der **SPS-Informationen**
- Bei Verwendung der **Baugruppen-Herstellungsinformationen** (diese Methode kann auch für Spezial-E/A-Baugruppen und CPU-Bus-Baugruppen verwendet werden)

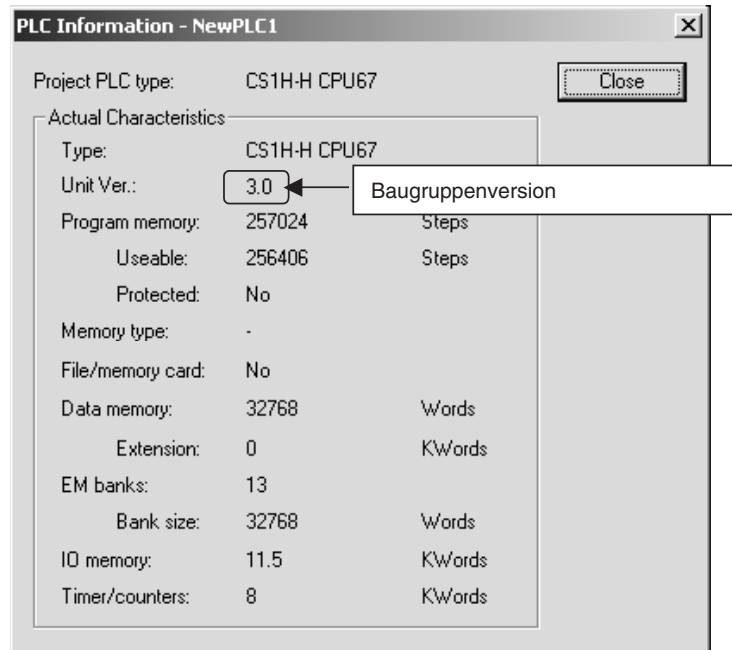
Hinweis

CX-Programmer Version 3.3 oder frühere Versionen können nicht zur Überprüfung von Baugruppenversionen verwendet werden.

SPS-Information

- Wenn der Geräte- und CPU-Typ bekannt sind, wählen Sie ihn im Dialogfeld *Ändern der SPS* aus, gehen Sie online, und wählen Sie in den Menüs **SPS - Bearbeiten - Information**.
- Wenn Geräte- und CPU-Typ nicht bekannt sind, diese aber direkt über ein serielles Kabel mit der CPU-Baugruppe verbunden sind, wählen Sie **SPS - Auto Online**, um online zu gehen, und wählen Sie dann in den Menüs **SPS - Bearbeiten - Information**.

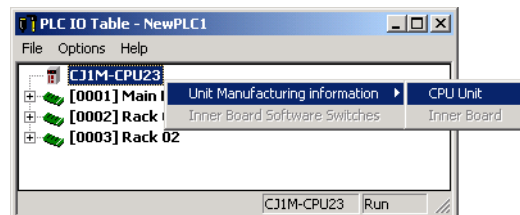
In beiden Fällen wird das nachfolgend abgebildete Dialogfeld *PLC Information (SPS Information)* angezeigt.



Verwenden Sie oben dargestellte Anzeige zur Prüfung der CPU-Baugruppenversion.

Baugruppen-Herstellungsinformationen

Klicken Sie im Fenster *IO Table (E/A-Tabelle)* mit der rechten Maustaste, und wählen Sie die Optionen ***Unit Manufacturing information - CPU Unit (Baugruppen-Herstellungsinformationen – CPU-Baugruppe)***.



Das nachfolgend abgebildete Dialogfeld *Unit Manufacturing information (Baugruppen-Herstellungsinformationen)* wird angezeigt.

Unit Manufacturing Information

File Help

Manufacturing Details

Revision: E

PCB Revision: ABD

Software Revision: AB 0

Lot Number: 040701

Manufacturing ID: E

Serial Number: 0001

Unit Ver.: 3.0

Unit Text

There is no Memory Card installed

CS1H H CPU67 Program

Verwenden Sie oben dargestellte Anzeige zur Prüfung der Baugruppen-Version der online verbundenen CPU-Baugruppe.

Die folgenden Baugruppen-Versionsetiketten werden mit der CPU-Baugruppe mitgeliefert.

Ver. 3.0 Ver.

Ver. 3.0 Ver.

バージョンアップによるユニットの搭載機能の差異を管理するためのラベルです。必要に応じて、製品の前面に貼り付けてご使用ください。

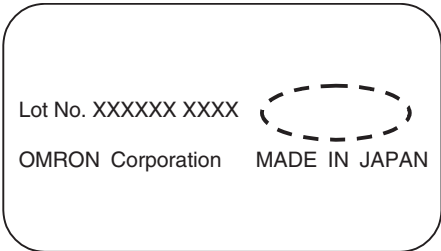
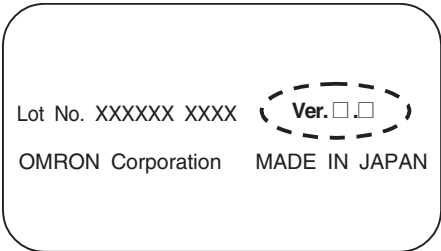
These Labels can be used to manage differences in the available functions among the Units. Place the appropriate label on the front of the Unit to show what Unit version is actually being used.

Diese Etiketten können an der Vorderseite von CPU-Baugruppen angebracht werden, um sie von anderen CPU-Baugruppen abweichender Baugruppenversionen unterscheiden zu können.

Verwendung der Baugruppen-Versionsetiketten

Angabe der Baugruppenversion

In diesem Handbuch wird die Baugruppenversion einer CPU-Baugruppe wie in der folgenden Tabelle dargestellt angegeben.

Produkt-Typenschild	CPU-Baugruppen ohne Angabe einer Baugruppenversion	Baugruppen mit Angabe einer Version (Ver. □.□)
Bedeutung		
Bezeichnung einzelner CPU-Baugruppen (z. B. CS1H-CPU67H)	CS1-H CPU-Baugruppen vor Version 2.0	CS1H-CPU67H CPU-Baugruppe Version □.□
Bezeichnung von Gruppen von CPU-Baugruppen (z. B. CS1-H CPU-Baugruppen)	CS1-H CPU-Baugruppen vor Version 2.0	CS1-H CPU-Baugruppen Version □.□
Bezeichnung einer ganzen Serie von CPU-Baugruppen (z. B. CPU-Baugruppen der CS-Serie)	CS-Serie CPU-Baugruppen vor Version 2.0	CS-Serie CPU-Baugruppen Version □.□

Baugruppenversionen und Chargennummern

Serie	Modell		Herstellungsdatum									
			Früher	Sep. 2003	Okt. 2003	Nov. 2003	Dez. 2003	Jun. 2004	Später			
CS-Serie	CS1 CPU-Baugruppen	CS1□-CPU□□										
			Keine Baugruppen-version									
	CS1-V1 CPU-Baugruppen	CS1□-CPU□□-V1										
			Keine Baugruppen-version									
	CS1-H CPU-Baugruppen	CS1□-CPU□□H										
			CPU-Baugruppen vor Ver. 2.0							CPU-Baugruppen Ver. 2.0 (Lot-Nr. ab 031105)		CPU-Baugruppen Ver. 3.0 (Lot-Nr. ab 040622)
	CS1D CPU-Baugruppen	CPU-Baugruppen für Duplex-CPU-Systeme	CS1D-CPU□□H	CPU-Baugruppen vor Ver. 1.1			CPU-Baugruppen Ver.1.1 (Lot-Nr. ab 031120)					
CPU-Baugruppen für Einzel-CPU-Systeme		CS1D-CPU□□S				CPU-Baugruppen Ver. 2.0 (Lot-Nr. ab 031215)						
CJ-Serie	CJ1 CPU-Baugruppen	CJ1G-CPU□□										
			CPU-Baugruppen vor Ver. 2.0									
	CJ1-H CPU-Baugruppen	CJ1□-CPU□□H										
			CPU-Baugruppen vor Ver. 2.0							CPU-Baugruppen Ver. 2.0 (Lot-Nr. ab 031105)		CPU-Baugruppen Ver. 3.0 (Lot-Nr. ab 040623)
	CJ1M CPU-Baugruppen (außer Low-End-Modelle)	CJ1M-CPU□□										
			CPU-Baugruppen vor Ver. 2.0							CPU-Baugruppen Ver. 2.0 (Lot-Nr. ab 031105)		CPU-Baugruppen Ver. 3.0 (Lot-Nr. ab 040624)
	CJ1M CPU-Baugruppen (Low-End-Modelle)	CJ1M-CPU11/21			CPU-Baugruppen Ver. 2.0 (Lot-Nr. ab 031002)			CPU-Baugruppen Ver. 3.0 (Lot-Nr. ab 040629)				
Programmier- software	CX-Programmer	WS02-CXPC1-EV□										
			Ver. 3.2		Ver. 3.3		Ver. 4.0		Ver. 5.0			

Unterstützung von Funktionen nach Baugruppenversion

CS1-H CPU-Baugruppen (CS1□-CPU□□H)

Funktion		Baugruppenversion	
		CPU-Baugruppen vor Version 2.0	CPU-Baugruppen Version 2.0
Herunterladen und Heraufladen von einzelnen Tasks		---	OK
Verbesserter Leseschutz durch Verwendung von Passwörtern		---	OK
Schreibschutz gegen FINS-Befehle, die über Netzwerke an die CPU-Baugruppen gesendet werden		---	OK
Online-Netzwerkverbindungen ohne E/A-Tabellen		---	OK
Kommunikation über bis zu 8 Netzwerkebenen		---	OK
Online-Verbindung zu SPS-Systemen über NS-Terminals		OK ab Lot-Nummer 030201	OK
Einstellung der Worte für den ersten Steckplatz		OK bei bis zu 8 Gruppen	OK bei bis zu 64 Gruppen
Automatische Übertragung bei Einschalten der Versorgungsspannung ohne Parameterdatei		---	OK
Automatische Erkennung der E/A-Zuweisungsmethode für automatische Übertragung bei Einschalten der Versorgungsspannung		---	---
Start-/Endzeiten des Betriebs		---	OK
Neue Anwendungsbefehle	MILH, MILR, MILC	---	OK
	=DT, <>DT, <DT, <=DT, >DT, >=DT	---	OK
	BCMP2	---	OK
	GRY	OK ab Lot-Nummer 030201	OK
	TPO	---	OK
	DSW, TKY, HKY, MTR, 7SEG	---	OK
	EXPLT, EGATR, ESATR, ECHRD, ECHWR	---	OK
	Lesen aus/Schreiben in CPU-Bus-Baugruppen mit IORD/IOWR	OK ab Lot-Nummer 030418	OK
	PRV2	---	---

CS1D CPU-Baugruppen

Funktion		CS1D CPU-Baugruppen für Duplex-CPU-Systeme (CS1D-CPU□□H)		CS1D CPU-Baugruppen für Einzel-CPU-Systeme (CS1D-CPU□□S)
		CPU-Baugruppen vor Version 1.1	CPU-Baugruppen Version 1.1	CPU-Baugruppen Version 2.0
Für CS1D CPU-Baugruppen spezifische Funktionen	Duplex-CPU-Baugruppen	OK	OK	---
	Online-Baugruppenaustausch	OK	OK	OK
	Duplex-Spannungsversorgungs-Baugruppen	OK	OK	OK
	Duplex-Controller-Link-Baugruppen	OK	OK	OK
	Duplex-Ethernet-Baugruppen	---	OK	OK
Herunterladen und Heraufladen von einzelnen Tasks		---	---	OK
Verbesserter Leseschutz durch Verwendung von Passwörtern		---	---	OK
Schreibschutz gegen FINS-Befehle, die über Netzwerke an die CPU-Baugruppen gesendet werden		---	---	OK
Online-Netzwerkverbindungen ohne E/A-Tabellen		---	---	OK
Kommunikation über bis zu 8 Netzwerkebenen		---	---	OK
Online-Verbindung zu SPS-Systemen über NS-Terminals		---	---	OK
Einstellung der Worte für den ersten Steckplatz		---	---	OK bei bis zu 64 Gruppen
Automatische Übertragung bei Einschalten der Versorgungsspannung ohne Parameterdatei		---	---	OK
Automatische Erkennung der E/A-Zuweisungsmethode für automatische Übertragung bei Einschalten der Versorgungsspannung		---	---	---
Start-/Endzeiten des Betriebs		---	OK	OK
Neue Anwendungsbefehle	MILH, MILR, MILC	---	---	OK
	=DT, <>DT, <DT, <=DT, >DT, >=DT	---	---	OK
	BCMP2	---	---	OK
	GRY	---	---	OK
	TPO	---	---	OK
	DSW, TKY, HKY, MTR, 7SEG	---	---	OK
	EXPLT, EGATR, ESATR, ECHRD, ECHWR	---	---	OK
	Lesen aus/Schreiben in CPU-Bus-Baugruppen mit IORD/IOWR	---	---	OK
	PRV2	---	---	---

CJ1-H/CJ1M CPU-Baugruppen

Funktion		CJ1-H CPU-Baugruppen (CJ1□-CPU□□H)		CJ1M CPU-Baugruppen, außer Low-End-Modelle (CJ1M-CPU□□)		CJ1M CPU- Baugruppen, Low-End- Modelle (CJ1M- CPU11/21)
		CPU-Bau- gruppen vor Version 2.0	CPU-Bau- gruppen Version 2.0	CPU-Bau- gruppen vor Version 2.0	CPU-Bau- gruppen Version 2.0	CPU-Bau- gruppen Version 2.0
Herunterladen und Heraufladen von einzelnen Tasks		---	OK	---	OK	OK
Verbesserter Leseschutz durch Verwendung von Passwörtern		---	OK	---	OK	OK
Schreibschutz gegen FINS- Befehle, die über Netzwerke an die CPU-Baugruppen gesendet werden		---	OK	---	OK	OK
Online-Netzwerkverbindungen ohne E/A-Tabellen		OK, aber nur, wenn E/A- Tabellenzuord- nung bei Ein- schalten der Spannungsver- sorgung einge- stellt ist	OK	OK, aber nur, wenn E/A- Tabellenzuord- nung bei Ein- schalten der Spannungsver- sorgung einge- stellt ist	OK	OK
Kommunikation über bis zu 8 Netzwerkebenen		OK bei bis zu 8 Gruppen	OK bei bis zu 64 Gruppen	OK bei bis zu 8 Gruppen	OK bei bis zu 64 Gruppen	OK bei bis zu 64 Gruppen
Online-Verbindung zu SPS- Systemen über NS-Terminals		OK ab Lot-Num- mer 030201	OK	OK ab Lot-Num- mer 030201	OK	OK
Einstellung der Worte für den ersten Steckplatz		---	OK	---	OK	OK
Automatische Übertragung bei Einschalten der Versorgungs- spannung ohne Parameterdatei		---	OK	---	OK	OK
Automatische Erkennung der E/A-Zuweisungsmethode für automatische Übertragung bei Einschalten der Versorgungs- spannung		---	OK	---	OK	OK
Start-/Endzeiten des Betriebs		---	OK	---	OK	OK
Neue Anwen- dungsbe- fehle	MILH, MILR, MILC	---	OK	---	OK	OK
	=DT, <>DT, <DT, <=DT, >DT, >=DT	---	OK	---	OK	OK
	BCMP2	---	OK	OK	OK	OK
	GRY	OK ab Lot-Num- mer 030201	OK	OK ab Lot-Num- mer 030201	OK	OK
	TPO	---	OK	---	OK	OK
	DSW, TKY, HKY, MTR, 7SEG	---	OK	---	OK	OK
	EXPLT, EGATR, ESATR, ECHRD, ECHWR	---	OK	---	OK	OK
	Lesen aus/Schrei- ben in CPU-Bus- Baugruppen mit IORD/IOWR	---	OK	---	OK	OK
	PRV2	---	---	---	OK, jedoch nur bei Modellen mit integrierten E/A	OK, jedoch nur bei Modellen mit integrierten E/A

Von Baugruppen der Version 3.0 und neuer unterstützte Funktionen

CS1-H CPU-Baugruppen (CS1□-CPU□□H)

Funktion		Baugruppenversion	
		bis einschließlich Ver. 2.0	Ver. 3.0
Funktionsblöcke (CX-Programmer Ver. 5.0 oder neuer erforderlich)		---	OK
Serial Gateway (Konvertierung von FINS-Befehlen in CompoWay/F-Befehle an der integrierten seriellen Schnittstelle)		---	OK
Kommentarspeicher (im internen Flash-Speicher)		---	OK
Erweiterte Easy-Backup-Datensicherung		---	OK
Neue Anwendungsbe- fehle	TXDU(256), RXDU(255) (ermöglichen die protokollfreie Kommunikation mittels serieller Kommunikationsbaugruppen ab Ver. 1.2)	---	OK
	Modell-Konvertierungsbefehle: XFERC(565), DISTC(566), COLLC(567), MOVBC(568), BCNTC(621)	---	OK
	Spezielle Funktionsblockbefehle: GETID(286)	---	OK
Weitere Befehle	TXD(236), RXD(235) (ermöglichen die protokollfreie Kommunikation mittels serieller Kommunikationssteckmodule ab Ver. 1.2)	---	OK

CS1D CPU-Baugruppen Baugruppenversion 3.0 wird nicht unterstützt.

CJ1-H/CJ1M CPU-Baugruppen (CJ1□-CPU□□H, CJ1G-CPU□□P, CJ1M-CPU□□)

Funktion		Baugruppenversion	
		bis einschließlich Ver. 2.0	Ver. 3.0
Funktionsblöcke (CX-Programmer Ver. 5.0 oder neuer erforderlich)		---	OK
Serial Gateway (Konvertierung von FINS-Befehlen in CompoWay/F-Befehle an der integrierten seriellen Schnittstelle)		---	OK
Kommentarspeicher (im internen Flash-Speicher)		---	OK
Erweiterte Easy-Backup-Datensicherung		---	OK
Neue Anwendungsbe- fehle	TXDU(256), RXDU(255) (ermöglichen die protokollfreie Kommunikation mittels serieller Kommunikationsbaugruppen ab Ver. 1.2)	---	OK
	Modell-Konvertierungsbefehle XFERC(565), DISTC(566), COLLC(567), MOVBC(568), BCNTC(621)	---	OK
	Spezielle Funktionsblockbefehle: GETID(286)	---	OK
Weitere Befehle	PRV(881) und PRV2(883): Neu hinzugekommene Berechnungsmethoden für die schnelle Berechnung von Impulsfrequenzen (nur CJ1M CPU-Baugruppen)	---	OK

Baugruppenversionen und Programmiergeräte

Um die bei den CPU-Baugruppen ab Ver. 2.0 neu hinzugekommenen Funktionen nutzen zu können, ist CX-Programmer Version 4.0 oder neuer erforderlich.

Um die bei den CPU-Baugruppen ab Ver. 3.0 neu hinzugekommenen Funktionsblöcke nutzen zu können, ist CX-Programmer Version 5.0 oder neuer erforderlich.

Die folgenden Tabellen zeigen die Beziehung zwischen Baugruppenversionen und Versionen von CX-Programmer.

Baugruppenversionen und Programmiergeräte

CPU-Baugruppe	Funktionen		CX-Programmer				Programmierkonsole
			bis einschließlich Ver. 3.2	Ver. 3.3	Ver. 4.0	ab Ver. 5.0	
CJ1M CPU-Baugruppen (Low-End-Modelle), Baugruppenversion 2.0	Neue Funktionen bei Baugruppenversion 2.0	Bei Verwendung neuer Funktionen	---	---	OK	OK	Keine Einschränkungen
		Ohne Verwendung neuer Funktionen	---	OK	OK	OK	
CS1-H, CJ1-H und CJ1M CPU-Baugruppen (außer Low-End-Modelle), Baugruppenversion 2.0	Neue Funktionen bei Baugruppenversion 2.0	Bei Verwendung neuer Funktionen	---	---	OK	OK	
		Ohne Verwendung neuer Funktionen	OK	OK	OK	OK	
CS1D CPU-Baugruppen für Einzel-CPU-Systeme, Baugruppenversion 2.0	Neue Funktionen bei Baugruppenversion 2.0	Bei Verwendung neuer Funktionen	---	---	OK	OK	
		Ohne Verwendung neuer Funktionen				OK	
CS1D CPU-Baugruppen für Duplex-CPU-Systeme, Baugruppenversion 1.1	Neue Funktionen bei Baugruppenversion 1.1	Bei Verwendung neuer Funktionen	---	---	OK	OK	
		Ohne Verwendung neuer Funktionen	OK	OK	OK	OK	
CS/CJ-Serie CPU-Baugruppen Ver. 3.0	Neue Funktionsblock-Funktionen bei Baugruppenversion 3.0	Bei Verwendung von Funktionsblöcken	---	---	---	OK	
		Ohne Verwendung von Funktionsblöcken	OK	OK	OK	OK	

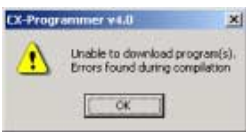
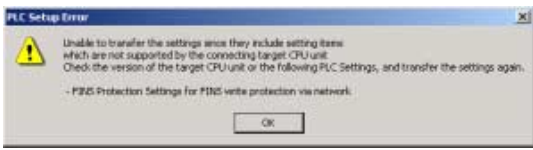
Hinweis Wie oben gezeigt, besteht keine Notwendigkeit zur Aktualisierung auf CX-Programmer Version 4.0, solange die in Baugruppenversion 2.0 bzw. 1.1 hinzugekommenen Funktionen nicht verwendet werden.

Gerätetyp-Einstellung

Die Baugruppenversion hat keinen Einfluss auf die mit CX-Programmierer vorgenommene Einstellung für den Gerätetyp. Wählen Sie den Gerätetyp entsprechend den Angaben in der folgenden Tabelle unabhängig von der Baugruppenversion der CPU-Baugruppe.

Serie	CPU-Baugruppengruppe	Produktbezeichnung der CPU-Baugruppe	Gerätetyp-Einstellung bei CX-Programmer Version 4.0 oder höher
CS-Serie	CS1-H CPU-Baugruppen	CS1G-CPU□□H	CS1G-H
		CS1H-CPU□□H	CS1H-H
	CS1D CPU-Baugruppen für Duplex-CPU-Systeme	CS1D-CPU□□H	CS1D-H (oder CS1H-H)
	CS1D CPU-Baugruppen für Einzel-CPU-Systeme	CS1D-CPU□□S	CS1D-S
CJ-Serie	CJ1-H CPU-Baugruppen	CJ1G-CPU□□H	CJ1G-H
		CJ1H-CPU□□H	CJ1H-H
	CJ1M CPU-Baugruppen	CJ1M-CPU□□	CJ1M

Behebung von Problemen mit Baugruppenversionen in CX-Programmer

Problem	Ursache	Lösung
 Nach Anzeige der oben abgebildeten Meldung wird im Meldungsfenster auf der Registerkarte <i>Kompilieren</i> ein Kompilierungsfehler angezeigt.	Es wurde der Versuch unternommen, mit Hilfe von CX-Programmer Version 4.0 oder höher ein Programm mit Befehlen, die nur von CPU-Baugruppen-Version 2.0 oder neuer unterstützt werden, auf eine CPU-Baugruppe vor Version 2.0 herunterzuladen.	Prüfen Sie das Programm oder wechseln Sie die CPU-Baugruppe, auf die das Programm heruntergeladen wird, zu einer CPU-Baugruppe der Version 2.0 oder neuer.
	Es wurde der Versuch unternommen, mit Hilfe von CX-Programmer Version 4.0 oder höher ein SPS-Setup mit Einstellungen (z. B. andere als Standardwerte), die nur von CPU-Baugruppen-Version 2.0 oder neuer unterstützt werden, auf eine CPU-Baugruppe vor Version 2.0 herunterzuladen.	Prüfen Sie die Einstellungen im SPS-Setup oder wechseln Sie die CPU-Baugruppe, auf die das Setup heruntergeladen wird, zu einer CPU-Baugruppe der Version 2.0 oder neuer.
„????“ wird in einem Programm angezeigt, das von der SPS zu CX-Programmer übertragen wurde.	CX-Programmer Version 3.3 oder früher wurde zum Heraufladen eines Programms von einer CPU-Baugruppe der Version 2.0 oder neuer verwendet, das Befehle enthält, die nur von CPU-Baugruppen-Version 2.0 oder neuer unterstützt werden.	Die neuen Befehle können nicht unter Verwendung von CX-Programmer Version 3.3 oder früher heraufgeladen werden. Verwenden Sie CX-Programmer Version 4.0 oder höher.

INHALTSVERZEICHNIS

SICHERHEITSHINWEISE xix

1	Zielgruppe	xx
2	Allgemeine Sicherheitshinweise	xx
3	Sicherheitshinweise	xx
4	Sicherheitshinweise für die Betriebsumgebung	xxii
5	Sicherheitshinweise zum Einsatz	xxii
6	Konformität mit EU-Richtlinien	xxvi

ABSCHNITT 1

Betrieb der CPU-Baugruppe 1

1-1	Inbetriebnahme - erste Schritte (nur CS1 CPU-Baugruppen)	2
1-2	Verwenden der internen Uhr (nur CS1 CPU-Baugruppen)	5
1-3	Interne Struktur der CPU-Baugruppe	6
1-4	Betriebsarten	9
1-5	Programme und Tasks	12
1-6	Beschreibung von Tasks	14

ABSCHNITT 2

Programmierung 19

2-1	Grundbegriffe	20
2-2	Sicherheitshinweise	55
2-3	Überprüfen von Programmen	64

ABSCHNITT 3

Befehlsfunktionen 71

3-1	Eingabebefehle	72
3-2	Ausgabebefehle	74
3-3	Sequenzsteuerbefehle	77
3-4	Zeitgeber- und Zählerbefehle	81
3-5	Vergleichsbefehle	85
3-6	Kopierbefehle	89
3-7	Datenverschiebepbefehle	92
3-8	Inkrementier- und Dekrementierbefehle	96
3-9	Arithmetische Befehle	97
3-10	Konvertierungsbefehle	102
3-11	Boolesche Befehle (Logikbefehle)	108
3-12	Spezielle arithmetische Befehle	110
3-13	Gleitkomma-Arithmetikbefehle	111
3-14	Gleitkomma-Arithmetikbefehle mit doppelter Präzision	115
3-15	Tabellendaten-Verarbeitungsbefehle	119
3-16	Datensteuerungsbefehle	123
3-17	Unterprogrammbefehle	127
3-18	Interrupt-Steuerungsbefehle	128
3-19	Befehle für schnelle Zähler und Impulsausgaben (nur CJ1M-CPU21/22/23)	130
3-20	Step-Befehle	132
3-21	Befehle für E/A-Baugruppen	132
3-22	Befehle für serielle Kommunikation	135
3-23	Netzwerkbefehle	136
3-24	Dateispeicherbefehle	139
3-25	Anzeigebefehle	140
3-26	Datum- und Uhrzeitbefehle	140

INHALTSVERZEICHNIS

3-27	Debugging-Befehle	141
3-28	Fehlerdiagnosebefehle	142
3-29	Sonstige Befehle	143
3-30	Blockprogrammierungsbefehle	144
3-31	Textzeichenketten-Verarbeitungsbefehle	150
3-32	Task-Steuerbefehle	153
3-33	Modell-Konvertierungsbefehle (ab CPU-Baugruppen Ver. 3.0)	154
3-34	Spezielle Funktionsblockbefehle	155

ABSCHNITT 4

Tasks	157
4-1 Merkmale von Tasks	158
4-2 Verwenden von Tasks	167
4-3 Interrupt-Tasks	177
4-4 Task-Programmierung mit Programmiergeräten	189

ABSCHNITT 5

Dateispeicherfunktionen	191
5-1 Dateispeicher.....	192
5-2 Arbeiten mit Dateien.....	211
5-3 Verwenden des Dateispeichers	247

ABSCHNITT 6

Erweiterte Funktionen	255
6-1 Zykluszeit/Hochgeschwindigkeitsverarbeitung	257
6-2 Indexregister	274
6-3 Serielle Kommunikation	284
6-4 Ändern des Modus der Zeitgeber/Zähler-Istwert-Aktualisierung	305
6-5 Verwenden eines zeitgesteuerten Interrupts als Präzisionszeitgeber (nur CJ1M)	313
6-6 Einstellungen für Start und Wartung	315
6-7 Diagnosefunktionen	326
6-8 CPU-Verarbeitungsbetriebsarten	331
6-9 Peripheriebedienungs-Prioritätsmodus	336
6-10 Betrieb ohne Batterie	342
6-11 Weitere Funktionen	344

ABSCHNITT 7

Programmübertragung, Testbetrieb und Debugging		347
7-1	Programmübertragung	348
7-2	Testbetrieb und Debugging	348

Anhang

A	SPS-Vergleichstabellen: SPS der CJ-Serie, CS-Serie, C200HG/HE/HX, CQM1H, CVM1 und der CV-Serie	357
B	Änderungen gegenüber älteren Host-Link-Systemen	383

Index	387
--------------------	------------

Revisionshistorie	393
--------------------------------	------------

Zu diesem Handbuch:

In diesem Handbuch wird die Programmierung von CPU-Baugruppen von speicherprogrammierbare Steuerungen (SPS) der CS/CJ-Serie beschrieben. Es besteht aus den auf der nächsten Seite aufgeführten Abschnitten. Die Serien CS und CJ werden wie in der folgenden Tabelle gezeigt unterteilt.

Baugruppe	CS-Serie	CJ-Serie
CPU-Baugruppen	CS1-H CPU-Baugruppen: CS1H-CPU□□H CS1G-CPU□□H	CJ1-H CPU-Baugruppen: CJ1H-CPU□□H CJ1G-CPU□□H
	CS1 CPU-Baugruppen: CS1H-CPU□□-EV1 CS1G-CPU□□-EV1	CJ1 CPU-Baugruppen: CJ1G-CPU□□-EV1 CJ1M CPU-Baugruppen: CJ1M-CPU□□
	CS1D CPU-Baugruppen: CS1D CPU-Baugruppen für Duplex-CPU-Systeme: CS1D-CPU□□H CS1D CPU-Baugruppen für Einzel-CPU-Systeme: CS1D-CPU□□S CS1D Prozess-CPU-Baugruppen: CS1D-CPU□□P	
E/A-Baugruppen	CS-Serie E/A-Baugruppen	CJ-Serie E/A-Baugruppen
Spezial-E/A-Baugruppen	CS-Serie Spezial-E/A-Baugruppen	CJ-Serie Spezial-E/A-Baugruppen
CPU-Bus-Baugruppen	CS-Serie CPU-Bus-Baugruppen	CJ-Serie CPU-Bus-Baugruppen
Spannungsversorgungs-Baugruppen	CS-Serie Spannungsversorgungs-Baugruppen	CJ-Serie Spannungsversorgungs-Baugruppen

Lesen Sie bitte dieses Handbuch und alle in der Tabelle auf der folgenden Seite aufgelisteten, dazugehörigen Handbücher und stellen Sie sicher, dass Sie alle Informationen verstanden haben, bevor Sie CPU-Baugruppen der CS/CJ-Serie in einem SPS-System installieren oder verwenden.

Dieses Handbuch besteht aus den folgenden Abschnitten.

Abschnitt 1 enthält eine Beschreibung der Grundstruktur und des Betrieb der CPU-Baugruppe.

Abschnitt 2 enthält grundlegende Informationen für das Schreiben, Überprüfen und Eingeben von Programmen.

Abschnitt 3 enthält eine Übersicht über die Befehle, die beim Schreiben von Anwenderprogrammen verwendet werden können.

Abschnitt 4 enthält eine Beschreibung der Funktionen von Tasks.

Abschnitt 5 enthält eine Beschreibung der Funktionen des Dateispeichers.

Abschnitt 6 bietet Details zu erweiterten Funktionen: Zykluszeit/Hochgeschwindigkeitsverarbeitung, Index-Register, serielle Kommunikation, Inbetriebnahme und Wartung, Diagnose und Debugging, Programmiergeräte sowie Einstellungen für die Antwortzeiten von E/A-Baugruppen der CJ-Serie.

Abschnitt 7 enthält eine Beschreibung der Vorgehensweisen für die Programmübertragung an die CPU-Baugruppe und der Funktionen zum Testen und zum Debugging des Programms.

In den **Anhängen** finden Sie einen Vergleich der CS/CJ-Serie, Informationen zu Einschränkungen bei der Verwendung von C200H Spezial-E/A-Baugruppen sowie zu an den Host-Link-Systemen vorgenommenen Änderungen.

Zu diesem Handbuch, Fortsetzung

Bezeichnung	Kat.-Nr.	Inhalt
SYSMAC CS/CJ-Serie CS1G/H-CPU□□-EV1, CS1G/H-CPU□□H, CS1D-CPU□□H, CS1D-CPU□□S, CJ1G-CPU□□, CJ1M-CPU□□, CJ1G-CPU□□P, CJ1G/H-CPU□□H SPS-Programmierhandbuch	W394	Programmierung und Nutzung der Funktionen der CS/CJ-Serie CPU-Baugruppen. (Dieses Handbuch)
SYSMAC CS-Serie CS1G/H-CPU□□-EV1, CS1G/H-CPU□□H Speicherprogrammierbare Steuerungen Technisches Handbuch	W339	Detaillierte Beschreibung des Konzepts, der Installation, der Wartung und anderer grundlegender Aspekte der CS-Serie CPU-Baugruppen.
SYSMAC CJ-Serie CJ1G-CPU□□, CJ1M-CPU□□, CJ1G-CPU□□P, CJ1G/H-CPU□□H SPS-Bediennerhandbuch	W393	Bietet einen Überblick und eine Beschreibung von Konstruktion, Installation, Wartung und grundlegender Funktionen von SPS-Systemen der CJ-Serie.
SYSMAC CJ-Serie CJ1M-CPU21/22/23 Handbuch der CPU-integrierten E/A	W395	Bietet eine Beschreibung der Funktionen der integrierten E/A bei CJ1M-CPU-Baugruppen.
SYSMAC CS-Serie CS1D-CPU□□H CPU-Baugruppen CS1D-CPU□□S CPU-Baugruppen CS1D-DPL01 Duplex-Baugruppe CS1D-PA207R Spannungsversorgungs-Baugruppe Duplex-Systeme Bediennerhandbuch	W405	Bietet einen Überblick und eine Beschreibung von Konstruktion, Installation, Wartung und grundlegender Funktionen eines auf CS1D CPU-Baugruppen basierenden Duplex-Systems.
SYSMAC CS/CJ-Serie CS1G/H-CPU□□-EV1, CS1G/H-CPU□□H, CS1D-CPU□□H, CS1D-CPU□□S, CJ1G-CPU□□, CJ1M-CPU□□, CJ1G-CPU□□P, CJ1G/H-CPU□□H Speicherprogrammierbare Steuerungen Befehle Referenzhandbuch	W340	Enthält Beschreibungen der von SPS-Systemen der CS/CJ-Serie unterstützten Befehle für die SPS-Programmierung.
SYSMAC CS/CJ-Serie CQM1H-PRO01-E, C200H-PRO27-E, CQM1-PRO01-E Programmierkonsolen Bediennerhandbuch	W341	Enthält Informationen darüber, wie SPS-Systeme der CS/CJ-Serie über eine Programmierkonsole programmiert und gesteuert werden.
SYSMAC CS/CJ-Serie CS1G/H-CPU□□-EV1, CS1G/H-CPU□□H, CS1D-CPU□□H, CS1D-CPU□□S, CJ1G-CPU□□, CJ1G/H-CPU□□H, CJ1G-CPU□□P, CJ1M-CPU□□, CS1W-SCB21-V1/41-V1, CS1W-SCU21-V1, CJ1W-SCU21-V1/41-V1 Kommunikationsbefehle Referenzhandbuch	W342	Enthält Beschreibungen der C-Serie- (Host-Link) und FINS-Kommunikationsbefehle, die mit SPS-Systemen der CS/CJ-Serie verwendet werden.
SYSMAC WS02-CXP□□-E CX-Programmer Handbuch Version 3.□	W414	Enthält Informationen zur Verwendung von CX-Programmer, einer Programmiersoftware, die SPS-Systeme der CS/CJ-Serie unterstützt, sowie der in CX-Programmer enthaltenen Software CX-Net.
SYSMAC WS02-CXP□□-E CX-Programmer Handbuch Version 4.□	W425	
SYSMAC WS02-CXP□□-E CX-Programmer Handbuch Version 5.□	W437	

Bezeichnung	Kat.-Nr.	Inhalt
SYSMAC WS02-CXP□□-E Bedienerhandbuch CX-Programmer (Funktionsblöcke)	W438	Funktionsblock-spezifische Spezifikationen und Verfahren. Diese Informationen sind nur für die Verwendung von Funktionsblöcken mit CX-Programmer Ver. 5.0 in Kombination mit einer CS1-H/CJ1-H/CJ1M CPU-Baugruppe Ver. 3.0 von Relevanz. Detaillierte Informationen zur sonstigen Verwendung von CX-Programmer Ver. 5.0 finden Sie im <i>CX-Programmer Handbuch Version 5.□ (W437)</i> .
SYSMAC CS/CJ-Serie CS1W-SCB21-V1/41-V1, CS1W-SCU21-V1, CJ1W-SCU21-V1/41-V1 Serielle Kommunikations-Module/-Baugruppen Handbuch	W336	Beschreibung der Nutzung von seriellen Kommunikations-Modulen und -Baugruppen zur Kommunikation mit externen Geräten, insbesondere auch der Verwendung von Standardsystemprotokollen für OMRON-Produkte.
SYSMAC WS02-PSTC1-E Bedienungsanleitung CX-Protocol	W344	Beschreibung der Nutzung von CX-Protocol zum Erstellen von Protokollmakros für die Kommunikation mit externen Geräten.



VORSICHT Falls Sie die in diesem Handbuch enthaltenen Informationen nicht durchlesen oder nicht verstehen, kann dies zur Verletzung oder zum Tod von Personen, zu einem Schaden am Produkt bzw. zu Fehlfunktionen führen. Lesen Sie jeden Abschnitt vollständig durch, und führen Sie die vorgestellten Maßnahmen und Bedienverfahren erst durch, wenn Sie sicher sind, dass Sie die im jeweiligen Abschnitt und den zugehörigen Abschnitten bereitgestellten Informationen verstanden haben.

SICHERHEITSHINWEISE

In diesem Abschnitt finden Sie allgemeine Sicherheitshinweise für die Nutzung der speicherprogrammierbaren Steuerungen (SPS) der CS/CJ-Serie sowie zugehöriger Geräte.

Die in diesem Abschnitt enthaltenen Informationen sind wichtig für die sichere und zuverlässige Anwendung von speicherprogrammierbaren Steuerungen. Bevor Sie versuchen, ein SPS-System einzurichten oder zu betreiben, müssen Sie diesen Abschnitt lesen und die hierin enthaltenen Informationen verstehen.

1	Zielgruppe	xx
2	Allgemeine Sicherheitshinweise	xx
3	Sicherheitshinweise	xx
4	Sicherheitshinweise für die Betriebsumgebung	xxii
5	Sicherheitshinweise zum Einsatz	xxii
6	Konformität mit EU-Richtlinien	xxvi
6-1	Anwendbare Richtlinien	xxvi
6-2	Konzepte	xxvi
6-3	Konformität mit EU-Richtlinien	xxvii
6-4	Methoden zur Entstörung von Relaisausgängen	xxvii

1 Zielgruppe

Das vorliegende Handbuch ist für den folgenden Personenkreis konzipiert, wobei Kenntnisse elektrischer Systeme vorausgesetzt werden (Elektrotechniker usw.).

- Personen, die mit der Installation von Fabrikautomations-Systemen befasst sind.
- Personen, die mit der Konstruktion von FA-Systemen befasst sind.
- Personen, die FA-Systeme und -Einrichtungen verwalten.

2 Allgemeine Sicherheitshinweise

Das Produkt muss vom Benutzer gemäß den in den Bedienerhandbüchern beschriebenen Leistungsspezifikationen betrieben werden.

Wenden Sie sich vor der Verwendung dieses Produktes an Ihre OMRON-Vertretung, falls Sie das Produkt unter Bedingungen verwenden, die nicht im Bedienerhandbuch aufgeführt sind bzw. wenn Sie das Produkt im Bereich der Nukleartechnik, im Eisenbahnverkehr, in der Luftfahrt, in Fahrzeugen, in Verbrennungssystemen, in medizinischen Geräten, in Spielautomaten, in Sicherheitsausrüstungen oder anderen Systemen, Geräten oder Ausrüstungen verwenden möchten, bei denen fehlerhafte Verwendung zu schwerwiegenden Gefahren für Leben und Sachgut führen kann.

Achten Sie darauf, dass die Nenn- und Leistungsdaten des Produkts für die Systeme, Maschinen und Geräte geeignet sind, und stellen Sie die Systeme, Maschinen und Geräte mit doppelten Sicherheitsmechanismen aus.

Dieses Handbuch enthält Informationen zu Programmierung und Betrieb des Produkts. Lesen Sie dieses Handbuch vor Verwendung des Produkts durch, und halten Sie dieses Handbuch während des Betriebs zu Referenzzwecken immer griffbereit.



VORSICHT

Es ist außerordentlich wichtig, dass SPS und alle SPS-Baugruppen nur für den vorgegebenen Einsatzzweck und unter den angegebenen Bedingungen verwendet werden. Dies gilt besonders für Anwendungen, bei denen direkt oder indirekt die Gefahr von Personenschäden besteht. Wenden Sie sich an den OMRON-Vertrieb, bevor Sie ein SPS-System für die oben aufgeführten Anwendungen einsetzen.

3 Sicherheitshinweise



VORSICHT

Die CPU-Baugruppe führt E/A-Aktualisierungen auch dann durch, wenn das Programm angehalten ist (d. h. insbesondere auch im PROGRAM-Modus). Überprüfen Sie die Sicherheit sorgfältig im Voraus, bevor Sie den Status von Bits oder Worten des E/A-Speichers ändern, die E/A-Baugruppen, Spezial-E/A-Baugruppen oder CPU-Bus-Baugruppen zugeordnet sind. Änderungen an den einer Baugruppe zugeordneten Daten können ein unvorhergesehenes Verhalten der an die Baugruppen angeschlossenen Lasten (Maschinen) zur Folge haben. Die folgenden Bedienvorgänge können zu Änderungen von Bits und Worten des E/A-Speichers führen.

- Übertragen von E/A-Speicherdaten von einem Programmiergerät in die CPU-Baugruppe.
- Ändern der aktuellen Werte im Speicher von einem Programmiergerät aus.
- Zwangsweises Setzen oder Rücksetzen von Bits über ein Programmiergerät.
- Übertragen von E/A-Speicherdateien von einer Speicherkarte oder dem EM-Dateispeicher in die CPU-Baugruppe.
- Übertragen von E/A-Speicherdaten von einem Host-Computer oder von einer anderen SPS im Netzwerk.



VORSICHT

Entnehmen Sie keine Baugruppe bei eingeschalteter Spannungsversorgung aus dem Rack. Es besteht die Gefahr eines elektrischen Schlags.

-  **VORSICHT** Berühren Sie Klemmen oder Klemmenblöcke nicht bei eingeschalteter Versorgungsspannung. Es besteht die Gefahr eines elektrischen Schlags.
-  **VORSICHT** Versuchen Sie nicht, Baugruppen zu zerlegen, instand zu setzen oder zu modifizieren. Bei jedem Versuch besteht das Risiko von Fehlfunktionen sowie die Gefahr eines elektrischen Schlags bzw. Brandgefahr.
-  **VORSICHT** Statten Sie externe Schaltkreise (d. h., nicht die in der SPS) mit Sicherheitsmaßnahmen aus, damit die Sicherheit im System gewährleistet bleibt, wenn aufgrund einer Fehlfunktion der SPS oder eines anderen externen Faktors mit Auswirkungen auf den SPS-Betrieb eine Abweichung auftritt. Dabei kann es sich z.B. um die im Folgenden genannten Maßnahmen handeln. Andernfalls besteht die Gefahr von schweren Unfällen.
- Externe Steuerschaltungen müssen mit Not-Aus-Schaltungen, Sperrschaltungen, Wegbegrenzungsschaltungen und ähnlichen Sicherheitsvorrichtungen ausgestattet werden.
 - Die SPS schaltet alle Ausgänge auf AUS, wenn durch die Selbstdiagnosefunktion ein Fehler (schwerwiegender Systemfehler) erkannt wird oder wenn ein FALS-Befehl (schwerwiegender anwenderdefinierter Fehler) ausgeführt wird. Zur Gewährleistung der Sicherheit im System müssen als Gegenmaßnahme für solche Fehler externe Sicherheitsmaßnahmen bereitgestellt werden.
 - Die SPS-Ausgänge können aufgrund von Ablagerungen oder Verbrennungen an den Ausgangsrelaiskontakten bzw. Zerstörung von Ausgangstristoren irrtümlicherweise EIN oder AUS geschaltet bleiben. Zur Gewährleistung der Sicherheit im System müssen als Gegenmaßnahme für solche Probleme externe Sicherheitsmaßnahmen bereitgestellt werden.
 - Wenn der 24-V-DC-Ausgang (Hilfsspannungsversorgung der SPS) überlastet oder kurzgeschlossen wird, fällt möglicherweise die Spannung ab, was dazu führt, dass die Ausgänge auf AUS geschaltet werden. Zur Gewährleistung der Sicherheit im System müssen als Gegenmaßnahme für solche Probleme externe Sicherheitsmaßnahmen bereitgestellt werden.
-  **Achtung** Überprüfen Sie die Sicherheit, bevor Sie unter Verwendung einer Programmiersoftware im Dateispeicher (Speicherkarte oder EM-Dateispeicher) gespeicherte Datendateien an den E/A-Bereich (CIO) der CPU-Baugruppe übertragen. Andernfalls können unabhängig von der Betriebsart der CPU-Baugruppe Fehlfunktionen an den mit der Ausgangsbaugruppe verbundenen Geräten auftreten.
-  **Achtung** Der Kunde muss durch entsprechende Vorkehrungen die Systemsicherheit für den Fall sicherstellen, dass bedingt durch unterbrochene Signalleitungen, vorübergehende Spannungsausfälle oder beliebige andere Ursachen falsche, fehlende oder ungewöhnliche Signale auftreten. Nicht ordnungsgemäßer Betrieb kann zu schweren Unfällen führen.
-  **Achtung** Das Anwenderprogramm und die Parameterdaten werden von den CS1-H, CJ1-H, CJ1M und CS1D CPU-Baugruppen beim Schreiben in die CPU-Baugruppe automatisch im Flash-Speicher gesichert. Der E/A-Speicher (einschließlich der DM-, EM- und HR-Bereiche) wird jedoch nicht in den Flash-Speicher geschrieben. Die Inhalte der DM-, EM- und HR-Bereiche können während eines Ausfalls der Versorgungsspannung durch eine Batterie erhalten werden. Wenn die Batterie nicht ordnungsgemäß funktioniert, sind die Inhalte dieser Bereiche nach einem Ausfall der Versorgungsspannung evtl. nicht mehr korrekt. Wenn der Inhalt der DM-, EM- und HR-Bereiche zur Steuerung externer Ausgänge verwendet wird, müssen geeignete Maßnahmen zur Vermeidung von falschen Ausgaben ergriffen werden, wenn der Batterie-Fehlermerker (A40204) auf EIN gesetzt ist.
-  **Achtung** Führen Sie die Online-Bearbeitung der Programme oder Parameter nur durch, nachdem Sie überprüft haben, dass eine Verlängerung der Zykluszeit keine negativen Auswirkungen mit sich bringt. Andernfalls sind die Eingangssignale möglicherweise nicht lesbar.

-  **Achtung** Stellen Sie die Sicherheit am Zielknotenpunkt sicher, bevor Sie ein Programm an einen anderen Knotenpunkt übertragen oder den Inhalt des E/A-Speicherbereichs verändern. Wenn Sie eine dieser Aktionen durchführen, ohne die Sicherheit zu überprüfen, kann dies zu Verletzungen führen.
-  **Achtung** Ziehen Sie die Schrauben am Klemmenblock speziell die der Spannungsversorgungsbaugruppe mit dem im Bedienerhandbuch angegebenen Drehmoment fest. Lose Schrauben können zu Schmorbränden oder Fehlfunktionen führen.
-  **Achtung** Berühren Sie die Spannungsversorgungsbaugruppe nicht bei eingeschalteter Versorgungsspannung bzw. direkt nach Ausschalten der Versorgungsspannung. Die Spannungsversorgungs-Baugruppe kann in diesen Fällen heiß sein, und Sie können sich Verbrennungen zuziehen.
-  **Achtung** Lassen Sie beim Anschließen eines PCs oder anderer Peripheriegeräte an eine SPS besondere Vorsicht walten, wenn an diese eine nicht isolierte, an eine externe Spannungsversorgung angeschlossene Baugruppe (CS1W-CLK12/52(-V1) oder CS1W-ETN01) angeschlossen ist. Wenn die 24-V-Seite der externen Spannungsversorgung und die 0-V-Seite des Peripheriegerätes zur gleichen Zeit geerdet sind, tritt ein Kurzschluss auf. Erden Sie beim Anschluss eines Peripheriegerätes an diese Art von SPS entweder die 0-V-Seite der externen Spannungsversorgung, oder erden Sie die externe Spannungsversorgung überhaupt nicht.

4 Sicherheitshinweise für die Betriebsumgebung

-  **Achtung** Betreiben Sie die SPS nicht an folgenden Orten:
- Orte, die direkter Sonneneinstrahlung ausgesetzt sind.
 - Orte, an denen Temperaturen oder Luftfeuchtigkeit außerhalb der in den technischen Daten angegebenen Bereiche herrschen.
 - Orte, die starken Temperaturschwankungen und damit Kondensatbildung ausgesetzt sind.
 - Orte, an denen korrosive oder entzündliche Gase vorhanden sind.
 - Orte, die dem Einfluss von Stäuben (besonders Eisenstaub) oder Salzen ausgesetzt sind.
 - Orte, die dem Einfluss von Feuchtigkeit, Öl oder Chemikalien ausgesetzt sind.
 - Orte, die Stößen oder Schwingungen ausgesetzt sind.
-  **Achtung** Ergreifen Sie beim Installieren von Systemen an folgenden Orten angemessene und geeignete Gegenmaßnahmen:
- Orte mit statischer Aufladung und anderen Störungen.
 - Orte mit starken elektromagnetischen Feldern.
 - Orte, die möglicherweise Radioaktivität ausgesetzt sind.
 - Orte in der Nähe von Spannungsversorgungseinrichtungen.
-  **Achtung** Die Betriebsumgebung des SPS-Systems kann große Auswirkungen auf die Lebensdauer und Zuverlässigkeit des Systems haben. Ungeeignete Betriebsumgebungen können Fehlfunktionen, Ausfälle und andere unvorhergesehene Probleme in Bezug auf das SPS-System zur Folge haben. Achten Sie darauf, dass die Betriebsumgebung zum Zeitpunkt der Installation die angegebenen Bedingungen erfüllt und dass diese Bedingungen während der Lebensdauer des Systems erfüllt bleiben.

5 Sicherheitshinweise zum Einsatz

Beachten Sie bei der Verwendung des SPS-Systems die folgenden Sicherheitshinweise.

- Sie müssen CX-Programmer (Programmiersoftware, die unter Windows läuft) verwenden, wenn Sie mehr als eine Task programmieren möchten.

Eine Programmierkonsole kann verwendet werden, um nur eine zyklische Task plus Interrupt-Tasks zu programmieren. Eine Programmierkonsole kann jedoch auch zum Bearbeiten von Multitask-Programmen verwendet werden, die ursprünglich mit CX-Programmer erstellt wurden.

 VORSICHT

Beachten Sie stets diese Sicherheitshinweise. Das Nichtbeachten der folgenden Sicherheitshinweise kann schwere und möglicherweise tödliche Verletzungen zur Folge haben.

- Achten Sie bei der Installation der SPS auf eine ordnungsgemäße Erdung, die den geltenden Vorschriften entspricht. Bei fehlender oder nicht vorschriftsmäßiger Erdung besteht die Gefahr eines elektrischen Schlags.
- Eine den Vorschriften entsprechende Erdung ist auch dann erforderlich, wenn die Schutzerdungsklemme (GR) und die Signalerdungsklemme (LG) der Spannungsversorgungs-Baugruppe miteinander verbunden werden.
- Schalten Sie die Spannungsversorgung zur SPS immer AUS, bevor Sie versuchen, eine der folgenden Arbeiten durchzuführen. Bei Nichtabschalten der Versorgungsspannung besteht das Risiko von Fehlfunktionen und die Gefahr eines elektrischen Schlags.
 - Anbringen oder Abbauen von Spannungsversorgungsbaugruppen, E/A-Baugruppen, CPU-Baugruppen, internen Modulen oder sonstigen Baugruppen.
 - Zusammensetzen der Baugruppen.
 - Einstellen der DIP-Schalter oder Drehschalter.
 - Anschließen von Kabeln oder Verdrahten des Systems.
 - Anschließen oder Lösen von Steckverbindungen.

 Achtung

Das Nichtbeachten der folgenden Sicherheitshinweise kann einen fehlerhaften Betrieb der SPS oder des Systems zur Folge haben, oder es könnten die SPS oder die SPS-Baugruppen beschädigt werden. Beachten Sie stets diese Sicherheitshinweise.

- Die Anwenderprogramm- und Parameterbereichsdaten in CS1-H, CS1D, CJ1-H und CJ1M CPU-Baugruppen werden im integrierten Flash-Speicher gesichert. Die BKUP-Anzeige auf der Vorderseite der CPU-Baugruppe leuchtet, während die Datensicherung ausgeführt wird. Schalten Sie die Versorgungsspannung der CPU-Baugruppe nicht aus, solange die BKUP-Anzeige leuchtet. Die Daten werden nicht gesichert, wenn die Versorgungsspannung ausgeschaltet wird.
- Wenn Sie eine CPU-Baugruppe der CS-Serie erstmalig verwenden, setzen Sie die mit Baugruppe mitgelieferte Batterie CS1W-BAT1 ein, und löschen Sie vor dem Programmieren sämtliche Speicherbereiche mittels eines Programmiergeräts. Wenn Sie die interne Uhr verwenden, schalten Sie nach dem Einsetzen der Batterie die Versorgungsspannung ein, und stellen Sie die Uhr mit Hilfe eines Programmiergeräts oder des Befehls DATE(735) ein. Die Uhr läuft erst, wenn die Zeit eingestellt wurde.
- Bei Auslieferung der CPU-Baugruppe ist das SPS-Setup so eingestellt, dass die CPU-Baugruppe in der Betriebsart startet, der mit dem Betriebsartenschalter der Programmierkonsole eingestellt ist. Wenn keine Programmierkonsole angeschlossen ist, startet eine CS1-CPU-Baugruppe der CS-Serie in der PROGRAM-Betriebsart, die CS1-H, CS1D, CJ1, CJ1-H oder CJ1M CPU-Baugruppen starten in der RUN-Betriebsart, so dass der Betrieb sofort aufgenommen wird. Starten Sie niemals absichtlich oder unabsichtlich den Betrieb, ohne sich zu vergewissern, dass dies sicher ist.
- Setzen Sie beim Erstellen einer AUTOEXEC.IOM-Datei über ein Programmiergerät (eine Programmierkonsole oder CX-Programmer) zum automatischen Übertragen von Daten beim Start die erste Schreibadresse auf D20000, und achten Sie darauf, dass die Größe der geschriebenen Daten nicht die Größe des DM-Bereichs übersteigt. Wenn die Datendatei beim Start von der Speicherkarte gelesen wird, werden beginnend bei D20000 Daten in die CPU-Baugruppe geschrieben, selbst wenn beim Erstellen der Datei AUTOEXEC.IOM eine andere Adresse eingestellt war. Wird ferner der DM-Bereich überschritten (dies ist beim Verwenden von CX-Programmer möglich), werden die übrigen Daten in den EM-Bereich geschrieben.

- Schalten Sie die Versorgungsspannung der SPS stets vor der der gesteuerten Anlage ein. Wird die Versorgungsspannung der SPS erst nach der der gesteuerten Anlage eingeschaltet, können zeitweilig Fehler in Steuerungssystemsignalen auftreten, da die Ausgangsklemmen an DC-Ausgangs-Baugruppen und anderen Baugruppen vorübergehend auf EIN geschaltet werden, wenn die Versorgungsspannung der SPS eingeschaltet wird.
- Der Kunde muss durch entsprechende Vorkehrungen die Systemsicherheit für den Fall sicherstellen, dass bedingt durch Schäden an internen Schaltkreisen (Relais, Transistoren oder andere Komponenten) von Ausgangs-Baugruppen intern auf AUS gesetzte Ausgänge weiterhin auf EIN geschaltet bleiben.
- Der Kunde muss durch entsprechende Vorkehrungen die Systemsicherheit für den Fall sicherstellen, dass bedingt durch unterbrochene Signalleitungen, vorübergehende Spannungsausfälle oder beliebige andere Ursachen falsche, fehlende oder ungewöhnliche Signale auftreten.
- Der Kunde muss Sperrschaltungen, Wegbegrenzungsschaltungen und ähnliche Sicherheitsmaßnahmen in externen Schaltungen (d. h. nicht die in der speicherprogrammierbaren Steuerung) vorsehen.
- Schalten Sie die Spannungsversorgung zur SPS nicht AUS, während Daten übertragen werden. Schalten Sie die Versorgungsspannung vor allem dann nicht aus, wenn eine Speicherkarte gelesen oder beschrieben wird. Entnehmen Sie die Speicherkarte auch nicht, wenn die Anzeige BUSY leuchtet. Drücken Sie zum Entnehmen einer Speicherkarte zunächst die Speicherkartentaste, und warten Sie dann, bis die BUSY-Anzeige erlischt, bevor Sie die Speicherkarte entnehmen.
- Wenn der E/A-Speicher-Haftmarker auf EIN gesetzt ist, werden die Ausgänge der SPS nicht auf AUS geschaltet und behalten ihren vorherigen Status bei, wenn die SPS von der RUN- oder MONITOR- in die PROGRAM-Betriebsart umgeschaltet wird. Stellen Sie sicher, dass in diesem Fall keine Gefahren von den angeschlossenen Geräten ausgehen. (Wenn der Betrieb wegen eines schwerwiegenden Fehlers (einschließlich FALS(007)) unterbrochen wird, werden alle Ausgänge aller Ausgangs-Baugruppen auf AUS geschaltet, und nur der interne Ausgabestatus wird aufrecht erhalten.)
- Der Speicherinhalt der DM-, EM- und HR-Bereiche in der CPU-Baugruppe wird durch eine Batterie gesichert. Wenn die Spannung der Batterie abfällt, können diese Daten verloren gehen. Ergreifen Sie mit Hilfe des Batterie-Fehlermerkers (A40204) Vorkehrungen für die Reinitialisierung von Daten oder andere Maßnahmen für den Fall, dass die Spannung der Batterie abfällt.
- Wenn eine SPS der CS-Serie mit 200 bis 240 V Wechselspannung versorgt wird, müssen Sie stets die Drahtbrücken von den Spannungsauswahlklemmen der Spannungsversorgungs-Baugruppe entfernen (mit Ausnahme von Spannungsversorgungs-Baugruppen mit weitem Eingangsspannungsbereich). Das Produkt wird zerstört, wenn bei angebrachten Drahtbrücken 200 bis 240 V AC angelegt werden.
- Verwenden Sie stets die in den Bedienerhandbüchern angegebenen Versorgungsspannungen. Eine falsche Spannung kann zu Fehlfunktionen oder Bränden führen.
- Ergreifen Sie geeignete Maßnahmen, um sicherzustellen, dass das Gerät mit der angegebenen Nennspannung und -frequenz versorgt wird. Lassen Sie an Installationsorten, an denen die Spannungsversorgung nicht stabil ist, besondere Vorsicht walten. Eine falsche Spannungsversorgung kann zu Fehlfunktionen führen.
- Installieren Sie externe Trennschalter, und ergreifen Sie weitere Sicherheitsmaßnahmen gegen Kurzschlüsse in externer Verdrahtung. Bei unzureichenden Sicherheitsmaßnahmen gegen Kurzschlüsse besteht Brandgefahr.
- Legen Sie keine Spannungen an die Eingangsbaugruppen an, die die Nenneingangsspannung übersteigen. Bei zu hohen Spannungen besteht Brandgefahr.

- Legen Sie keine Spannungen an die Ausgangsbaugruppen an und schließen Sie keine Lasten an die Ausgangsbaugruppen an, die die maximale Schaltleistung übersteigen. Bei übermäßigen Spannungen oder Lasten besteht Brandgefahr.
- Trennen Sie vor der Durchführung von Isolationswiderstandsmessungen oder Spannungsbeständigkeitstests die Signalmasse (LG) von der Gehäuseerde (GR). Andernfalls kann es zu Verbrennungen kommen.
- Installieren Sie die Baugruppen ordnungsgemäß, wie in den Bedienerhandbüchern angegeben. Bei unsachgemäßer Installation der Baugruppen besteht die Gefahr von Fehlfunktionen.
- Bei SPS-Systemen der CS-Serie muss sichergestellt werden, dass sämtliche Befestigungsschrauben der Baugruppen und des Baugruppenträgers mit dem in den entsprechenden Handbüchern angegebenen Drehmoment festgezogen werden. Ein falsches Anzugsdrehmoment kann zu Fehlfunktionen führen.
- Stellen Sie sicher, dass alle Klemmenschrauben und Schrauben von Steckverbindern mit dem in den entsprechenden Handbüchern angegebenen Drehmoment festgezogen werden. Ein falsches Anzugsdrehmoment kann zu Fehlfunktionen führen.
- Belassen Sie beim Verdrahten die Schutzfolie an der Baugruppe. Beim Entfernen der Schutzfolie besteht die Gefahr einer Fehlfunktion, wenn Fremdkörper in die Baugruppe eindringen.
- Entfernen Sie die Schutzfolie nach Abschluss der Verdrahtungsarbeiten, um eine ordnungsgemäße Wärmeableitung zu gewährleisten. Bei nicht entfernter Schutzfolie besteht die Gefahr von Fehlfunktionen.
- Verwenden Sie für die Verdrahtung Crimp-Kabelschuhe. Schließen Sie keine abisolierten Litzendrähte direkt an Klemmen an. Beim Anschließen von abisolierten Litzendrähten besteht die Gefahr von Schmorbränden.
- Verdrahten Sie alle Anschlüsse ordnungsgemäß.
- Überprüfen Sie die gesamte Verdrahtung und alle Schaltereinstellungen sorgfältig, bevor Sie die Spannungsversorgung einschalten. Bei unsachgemäßer Verdrahtung besteht Brandgefahr.
- Installieren Sie Baugruppen erst, nachdem Sie die Klemmenblöcke und Steckverbinder eingehend überprüft haben.
- Stellen Sie sicher, dass die Klemmenblöcke, Speicherbaugruppen, Verlängerungskabel und andere Teile mit Arretierungen ordnungsgemäß eingearbeitet sind. Bei fehlender Arretierung besteht die Gefahr von Fehlfunktionen.
- Überprüfen Sie vor Beginn des Betriebs die Schaltereinstellungen, den Inhalt des DM-Bereichs sowie andere Voreinstellungen. Bei Aufnahme des Betriebs ohne sachgemäße Einstellungen bzw. Daten besteht die Gefahr von unerwartetem Verhalten.
- Überprüfen Sie das Anwenderprogramm auf ordnungsgemäße Ausführung, bevor Sie es tatsächlich für die SPS einsetzen. Wird das Programm nicht überprüft, besteht die Gefahr von unerwartetem Verhalten.
- Vergewissern Sie sich, dass keine negativen Auswirkungen im System auftreten können, bevor Sie eine der folgenden Aktionen durchführen. Andernfalls besteht die Gefahr von unerwartetem Verhalten.
 - Ändern der Betriebsart der SPS.
 - Zwangsweises Setzen/Rücksetzen eines Bits im Speicher.
 - Ändern des aktuellen Wertes eines Wortes oder eines voreingestellten Wertes im Speicher.
- Ziehen Sie nicht an den Kabeln, und biegen Sie die Kabel nicht über das natürliche Maß hinaus. Andernfalls können die Kabel dabei brechen.
- Stellen Sie keine Gegenstände auf die Kabel oder andere Leitungen. Die Kabel können dabei brechen.
- Verwenden Sie keine handelsüblichen RS-232C-Computerkabel, sondern ausschließlich die hier aufgeführten Spezialkabel. Müssen Kabel angefertigt werden, so beachten Sie unbedingt die Angaben in diesem Handbuch. Bei Verwendung von handelsüblichen Kabeln besteht die Gefahr, dass externe Geräte oder die CPU-Baugruppe beschädigt werden.

- Verbinden Sie niemals Stift 6 (5-V-Spannungsversorgung) des RS-232C-Anschlusses der CPU-Baugruppe mit einem anderen Gerät als den Adaptern NT-AL001 oder CJ1W-CIF11. Das externe Gerät oder die CPU-Baugruppe können andernfalls beschädigt werden.
- Achten Sie beim Austauschen von Komponenten darauf, dass die neue Komponente die richtigen Nenndaten besitzt. Andernfalls besteht die Gefahr von Fehlfunktionen oder Brandgefahr.
- Achten Sie vor Berühren der SPS darauf, dass Sie zunächst einen geerdeten Metallgegenstand berühren, um statische Aufladung abzuleiten. Andernfalls besteht die Gefahr von Fehlfunktionen oder Beschädigungen.
- Bewahren Sie Leiterplatten bei Transport oder Lagerung in antistatischem Material auf, um sie vor statischer Elektrizität zu schützen. Achten Sie auf die Einhaltung der zulässigen Lagerungstemperatur.
- Berühren Sie Leiterplatten oder die darauf befindlichen Komponenten nicht mit bloßen Händen. Auf den Leiterplatten befinden sich scharfe Anschlussdrähte und andere Teile, die bei unsachgemäßer Handhabung zu Verletzungen führen können.
- Schließen Sie die Batterieanschlüsse niemals kurz. Die Batterie darf nicht aufgeladen, zerlegt, erhitzt oder verbrannt werden. Setzen Sie die Batterie keinen starken Erschütterungen aus. Andernfalls besteht die Gefahr, dass die Batterie undicht wird, bricht, Hitze erzeugt oder sich entzündet. Entsorgen Sie Batterien, die heruntergefallen sind oder auf sonstige Weise heftigen Erschütterungen ausgesetzt waren. Batterien, die Erschütterungen ausgesetzt waren, können beim Gebrauch möglicherweise undicht sein.
- Nach UL-Standards dürfen Batterien nur von erfahrenen Technikern ausgetauscht werden. Lassen Sie die Batterien nicht von Personen austauschen, die nicht dafür qualifiziert sind.
- Bei einer SPS der CJ-Serie müssen die Schieber an den Ober- und Unterseiten der Spannungsversorgungs-Baugruppen, CPU-Baugruppen, E/A-Baugruppen, Spezial-E/A-Baugruppen und CPU-Bus-Baugruppen vollständig verriegelt werden (sie müssen hörbar einrasten). Die SPS funktioniert möglicherweise nicht ordnungsgemäß, wenn die Schieber nicht eingerastet sind.
- Bringen Sie bei einer SPS der CJ-Serie stets die Endabdeckung an der letzten Baugruppe rechts an. Die SPS funktioniert ohne die Endabdeckung nicht ordnungsgemäß.
- Fehlerhaft eingerichtete Data-Link-Tabellen oder -Parameter können zu unerwartetem Systemverhalten führen. Stellen Sie auch bei korrekt eingerichteten Data-Link-Tabellen und -Parametern vor dem Starten und Stoppen von Data-Links sicher, dass dies keine unerwünschten Auswirkungen auf das System hat.
- CPU-Bus-Baugruppen werden neu gestartet, wenn Routing-Tabellen von einem Programmiergerät auf die CPU-Baugruppe übertragen werden. Das Neustarten der Baugruppen ist zum Lesen und Aktivieren der neuen Routing-Tabellen erforderlich. Stellen Sie vor dem Neustart der CPU-Bus-Baugruppen sicher, dass dies keine unerwünschten Auswirkungen auf das System hat.

6 Konformität mit EU-Richtlinien

6-1 Anwendbare Richtlinien

- EMV-Richtlinien
- Niederspannungsrichtlinie

6-2 Konzepte

EMV-Richtlinien

OMRON-Geräte, die den EU-Richtlinien entsprechen, entsprechen ebenfalls den damit zusammenhängenden EMV-Normen, so dass sie auf einfachere Weise in andere Geräte oder die Gesamtanlage integriert werden können. Die tatsächlichen Produkte wurden auf Konformität mit EMV-Normen geprüft (siehe folgender Hinweis). Ob die Produkte jedoch in dem vom Kunden verwendeten System den Normen entsprechen, muss der Kunde selbst prüfen.

Die Abstrahlungseigenschaften von OMRON-Produkten, die den EU-Richtlinien entsprechen, variieren je nach Konfiguration, Verdrahtung und anderen Bedingungen der Installation oder der Schaltschränke, in denen die OMRON-Geräte installiert sind. Der Kunde muss daher eine abschließende Prüfung durchführen, um die Bestätigung zu erhalten, dass die Geräte und die Gesamtanlage den EMV-Normen entsprechen.

Hinweis Die folgenden EMV-Normen (elektromagnetische Verträglichkeit) sind anwendbar:

EMS (Elektromagnetische Störfestigkeit):

CS-Serie: EN61131-2 und EN61000-6-2

CJ-Serie: EN61000-6-2

EMB (Elektromagnetische Beeinflussung):

EN61000-6-4

(Abstrahlungseigenschaften: 10-m-Bestimmungen)

Niederspannungsrichtlinie

Stellen Sie immer sicher, dass Geräte, die mit Spannungen von 50 bis 1.000 V AC bzw. 75 bis 1.500 V DC arbeiten, die erforderlichen Sicherheitsnormen für die SPS erfüllen (EN61131-2).

6-3 Konformität mit EU-Richtlinien

Die SPS der CS/CJ-Serie entsprechen den EU-Richtlinien. Um zu gewährleisten, dass die Anlage oder das Gerät, in dem die SPS der CS/CJ-Serie verwendet wird, den EU-Richtlinien entspricht, muss die SPS wie folgt installiert sein:

- 1,2,3...**
1. Die SPS der CS/CJ-Serie muss in einem Schaltschrank installiert sein.
 2. Sie müssen eine Schutzisolierung oder eine doppelte Isolierung für die DC-Spannungsversorgungen verwenden, die an DC-Spannungsversorgungs- oder E/A-Baugruppen angeschlossen werden.
 3. SPS der CS/CJ-Serie, die den EU-Richtlinien entsprechen, entsprechen ebenfalls der allgemeinen Emissionsnorm (EN50081-6-4). Die Abstrahlungseigenschaften (10-m-Bestimmungen) variieren je nach Konfiguration des verwendeten Schaltschranks, anderen an den Schaltschrank angeschlossenen Geräten, Verdrahtung und weiteren Bedingungen. Daher müssen Sie überprüfen, ob die Gesamtanlage oder -ausrüstung den EU-Richtlinien entspricht.

6-4 Methoden zur Entstörung von Relaisausgängen

Die SPS der CS/CJ-Serie entsprechen den allgemeinen Emissionsnormen (EN50081-6-4) der EMV-Richtlinien. Eine durch Schalten des Relaisausgangs erzeugte Störung erfüllt diese Norm jedoch möglicherweise nicht. In diesem Fall muss ein Entstörfilter an der Lastseite angeschlossen werden, oder für die SPS müssen andere geeignete externe Gegenmaßnahmen eingerichtet werden.

Die zur Erfüllung der Normen vorzunehmenden Gegenmaßnahmen variieren je nach Gerät auf der Lastseite, Verdrahtung, Konfiguration der Anlagen usw. Nachfolgend finden Sie Beispiele für Gegenmaßnahmen zur Reduzierung der erzeugten Störungen.

Gegenmaßnahmen

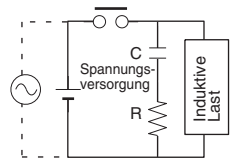
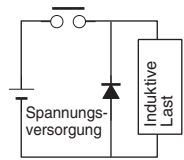
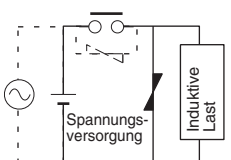
(Nähere Einzelheiten finden Sie in EN61000-6-4.)

Es sind keine Gegenmaßnahmen erforderlich, wenn die Frequenz der Lastschaltung im Gesamtsystem, in das die SPS eingebunden ist, unter fünf Schaltungen pro Minute liegt.

Es sind Gegenmaßnahmen erforderlich, wenn die Frequenz der Lastschaltung im Gesamtsystem, in das die SPS eingebunden ist, über fünf Schaltungen pro Minute liegt.

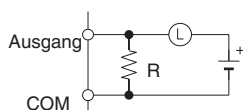
Beispiele für Gegenmaßnahmen

Wird eine induktive Last geschaltet, schließen Sie einen Überspannungsschutz, Dioden usw. parallel zu der Last oder dem Kontakt an, wie unten gezeigt.

Schaltung	Strom		Eigenschaften	Erforderliche Komponenten
	AC	DC		
RC-Glied 	Ja	Ja	Wenn es sich bei der Last um ein Relais oder eine Magnetspule handelt, gibt es eine Verzögerung zwischen dem Öffnen des Schaltkreises und dem Rücksetzen der Last. Wenn die Versorgungsspannung 24 oder 48 V beträgt, setzen Sie den Überspannungsschutz parallel zur Last ein. Wenn die Versorgungsspannung 100 bis 200 V beträgt, setzen Sie den Überspannungsschutz zwischen den Kontakten ein.	Kondensator: 0,5 bis 1 μF / A Laststrom; Widerstand: 0,5 bis 1 Ω / V Lastspannung. Diese Werte variieren jedoch je nach Last und den Eigenschaften des Relais. Ermitteln Sie experimentell die optimalen Werte. Berücksichtigen Sie dabei, dass der Kondensator die Funkenentladung beim Öffnen der Kontakte unterdrücken und der Widerstand den beim Schließen der Kontakte fließenden Ladestrom des Kondensators begrenzen soll. Die Durchschlagfestigkeit des Kondensators muss zwischen 200 und 300 V liegen. Wird eine AC-Last geschaltet, muss ein ungepolter Kondensator eingesetzt werden.
Freilaufdiode 	Nein	Ja	Die antiparallel zur Last geschaltete Diode (Freilaufdiode) leitet den beim Abbau des Magnetfelds entstehenden Strom wieder in die Last, wo er aufgrund des Ohmschen Widerstands der Last in Wärme umgewandelt wird. Diese durch diese Methode verursachte zeitliche Verzögerung zwischen dem Öffnen des Schaltkreises und dem Rücksetzen der Last ist länger als die durch die Kondensator/Widerstand-Methode verursachte.	Der Wert für die Durchschlagfestigkeit der Diode in Sperrrichtung muss mindestens zehnmal so hoch sein wie der Spannungswert in der Schaltung. Der Durchlassstrom der Diode muss gleich oder größer als der Laststrom sein. Bei Niederspannungsschaltkreisen genügt es, wenn die Spitzensperrspannung der Freilaufdiode dem Zwei- bis Dreifachen der Lastspannung entspricht.
Varistor 	Ja	Ja	Der Varistor wird bei Überschreitung einer bestimmten Spannung, die größer als die Lastspannung gewählt werden muss, niederohmig. Auf diese Weise wird der beim Abbau des Magnetfelds der induktiven Last entstehende Strom kurzgeschlossen und die Entstehung eines Schaltfunken an den Kontakten verhindert. Es tritt eine zeitliche Verzögerung zwischen dem Öffnen des Schaltkreises und dem Rücksetzen der Last auf. Beträgt die Versorgungsspannung 24 oder 48 V, so schalten Sie den Varistor parallel zur Last. Beträgt die Versorgungsspannung 100 oder 200 V, so schalten Sie den Varistor parallel zu den Schaltkontakten.	---

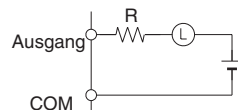
Beim Schalten einer Last mit einem hohen Einschaltstrom, etwa einer Glühlampe, reduzieren Sie den Einschaltstrom wie unten gezeigt.

Abhilfemaßnahme 1



Durch einen Vorschaltwiderstand fließt bereits ca. ein Drittel des Nennstroms der Glühlampe.

Abhilfemaßnahme 2



Verwendung eines Begrenzungswiderstands

ABSCHNITT 1

Betrieb der CPU-Baugruppe

In diesem Kapitel werden die Grundstruktur und der grundlegende Betrieb der CPU-Baugruppe beschrieben.

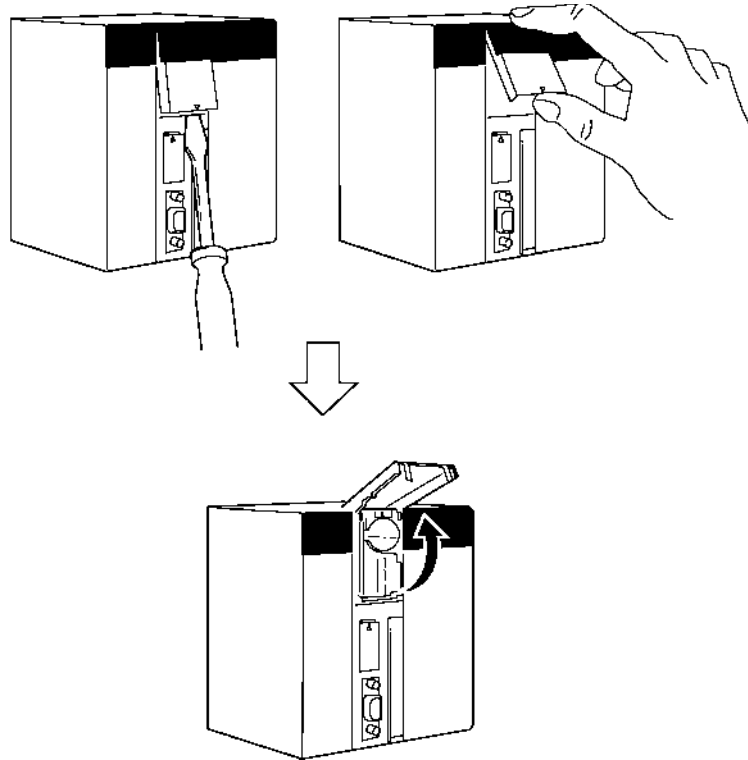
1-1	Inbetriebnahme - erste Schritte (nur CS1 CPU-Baugruppen)	2
1-2	Verwenden der internen Uhr (nur CS1 CPU-Baugruppen)	5
1-3	Interne Struktur der CPU-Baugruppe	6
1-3-1	Übersicht	6
1-3-2	Blockdiagramm des Speichers einer CPU-Baugruppe	7
1-4	Betriebsarten	9
1-4-1	Beschreibung der Betriebsarten	9
1-4-2	Initialisierung des E/A-Speichers	10
1-4-3	Betriebsart beim Einschalten	11
1-5	Programme und Tasks	12
1-6	Beschreibung von Tasks	14

1-1 Inbetriebnahme - erste Schritte (nur CS1 CPU-Baugruppen)

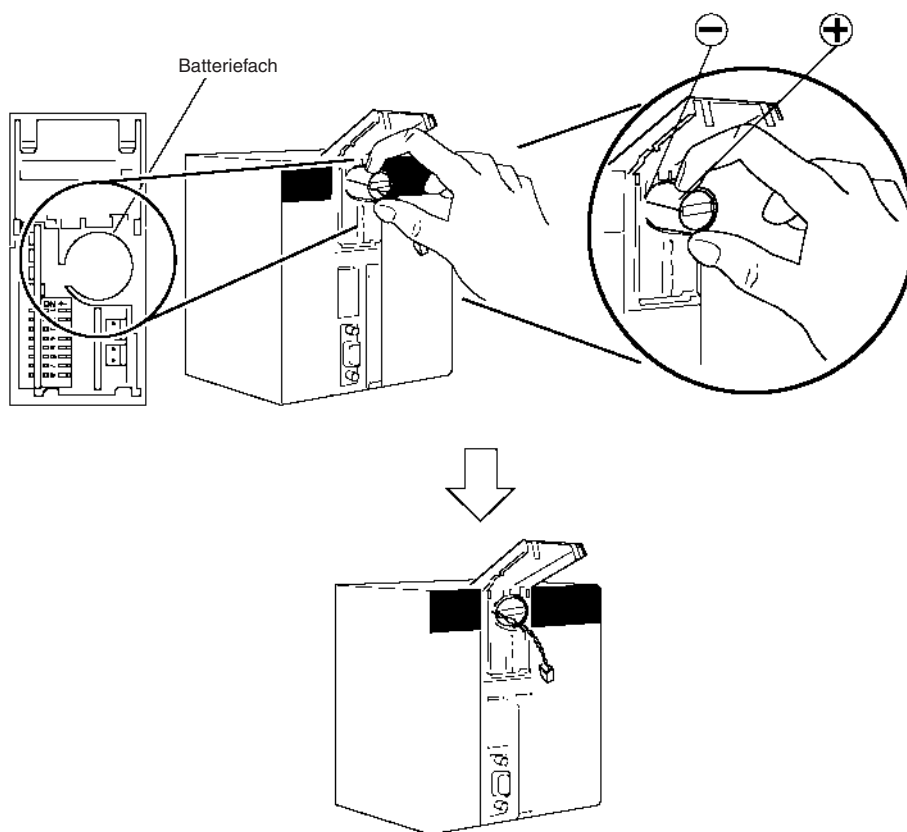
Einsetzen der Batterie

Vor Inbetriebnahme der CS1 CPU-Baugruppe müssen Sie den Batteriesatz in die CPU-Baugruppe einsetzen. Gehen Sie dazu wie folgt vor.

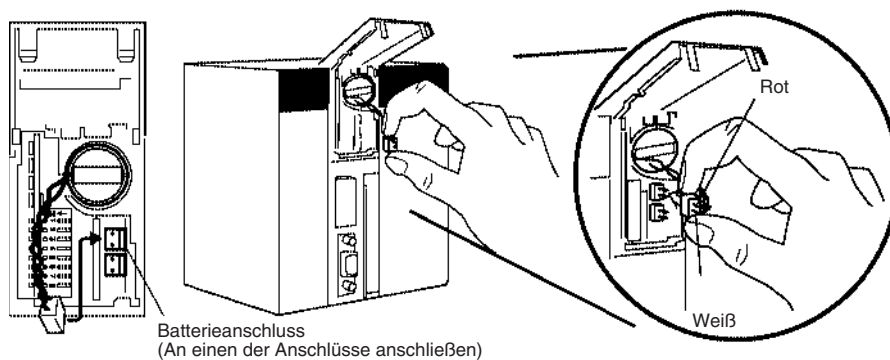
- 1,2,3...**
1. Führen Sie einen Schlitzschraubendreher in den schmalen Schlitz an der Unterkante des Batteriefachs ein, und drücken Sie den Deckel zum Öffnen nach oben.



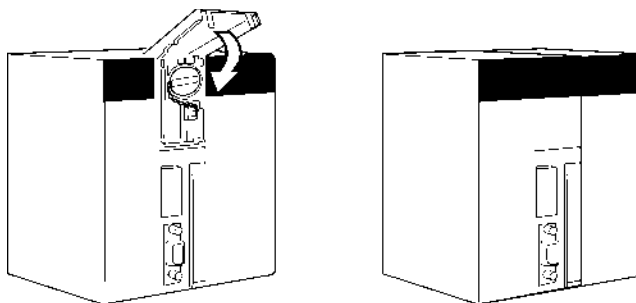
2. Setzen Sie den Batteriesatz mit dem Kabel nach außen in das Batteriefach ein.



3. Schließen Sie die Buchse des Batteriesatzes an den entsprechenden eingelöteten Batteriestecker im Batteriefach an. Der rote Draht muss an den oberen Stift und der weiße Draht an den unteren Stift angeschlossen werden. Im Gerät befinden sich zwei Batteriestecker. Schließen Sie die Batterie an einen der beiden Batteriestecker an. Es ist gleichgültig, ob Sie die Batterie an den oberen oder an den unteren Batteriestecker anschließen.



4. Verstecken Sie das Kabel im Fach, und schließen Sie den Deckel.



Löschen des Speichers

Löschen Sie nach dem Einsetzen der Batterie den Speicher mit Hilfe der Programmierkonsole. Dadurch wird der RAM-Speicher in der CPU-Baugruppe initialisiert.

Programmierkonsole

Geben Sie an der Programmierkonsole folgende Tastenfolge ein.



Hinweis

Beim Löschen von Speicherinhalten über die Programmierkonsole können Sie immer nur jeweils eine zyklische Task angeben. Sie können eine zyklische Task und eine Interrupt-Task oder eine zyklische Task und keine Interrupt-Task angeben. Weitere Informationen zum Löschen des Speichers finden Sie im *Bedienerhandbuch*. Weitere Informationen zu Tasks finden Sie in **ABSCHNITT 1 Betrieb der CPU-Baugruppe** und in **ABSCHNITT 4 Tasks**.

CX-Programmer

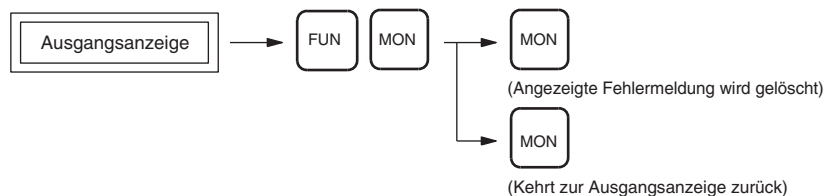
Der Speicher kann auch mit Hilfe von CX-Programmer gelöscht werden. Eine Beschreibung der Vorgehensweise hierzu finden Sie im *CX-Programmer Bedienerhandbuch*.

Löschen von Fehlern

Nach dem Löschen des Speichers müssen alle in der CPU-Baugruppe gespeicherten Fehler, einschließlich des Fehlers wegen zu niedriger Batteriespannung, gelöscht werden.

Programmierkonsole

Geben Sie an der Programmierkonsole folgende Tastenfolge ein.



CX-Programmer

Fehler können auch mit Hilfe von CX-Programmer gelöscht werden. Eine Beschreibung der Vorgehensweise hierzu finden Sie im *CX-Programmer Bedienerhandbuch*.

Hinweis

Falls ein internes Steckmodul installiert ist, bleibt möglicherweise weiterhin ein Routing-Tabellen-Fehler bestehen, obwohl Sie den Fehler mit Hilfe von CX-Programmer gelöscht haben. (Bei einem seriellen Kommunikationsmodul ist A42407 auf EIN gesetzt.) Falls dies der Fall ist, schalten Sie entweder das Gerät aus und wieder ein, oder führen Sie einen Neustart des internen Moduls durch. Löschen Sie den Fehler anschließend erneut.

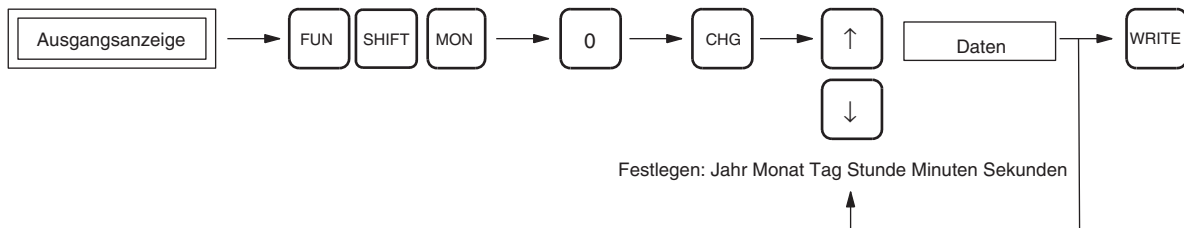
1-2 Verwenden der internen Uhr (nur CS1 CPU-Baugruppen)

Beim Einlegen der Batterie in die CPU-Baugruppe der CS-Serie ist die interne Uhr der CPU-Baugruppe auf „00 (Jahr), 01 (Monat), 01 (Tag), (00-01-01), 00 (Stunden), (00) Minuten, (00) Sekunden, (00:00:00), und Sonntag (So)“ eingestellt.

Wenn Sie die interne Uhr verwenden möchten, schalten Sie die Spannungsversorgung ein, nachdem Sie die Batterie eingelegt haben. Stellen Sie entweder mit Hilfe eines Programmiergeräts (Programmierkonsole oder CX-Programmer) die Zeit ein, oder führen Sie den Befehl zur Uhreinstellung CLOCK ADJUSTMENT (DATE) aus. Sie können die interne Uhr auch mit einem FINS-Befehl zeitgenau und mit dem richtigen Datum starten.

So stellen Sie die interne Uhr mit Hilfe der Programmierkonsole ein:

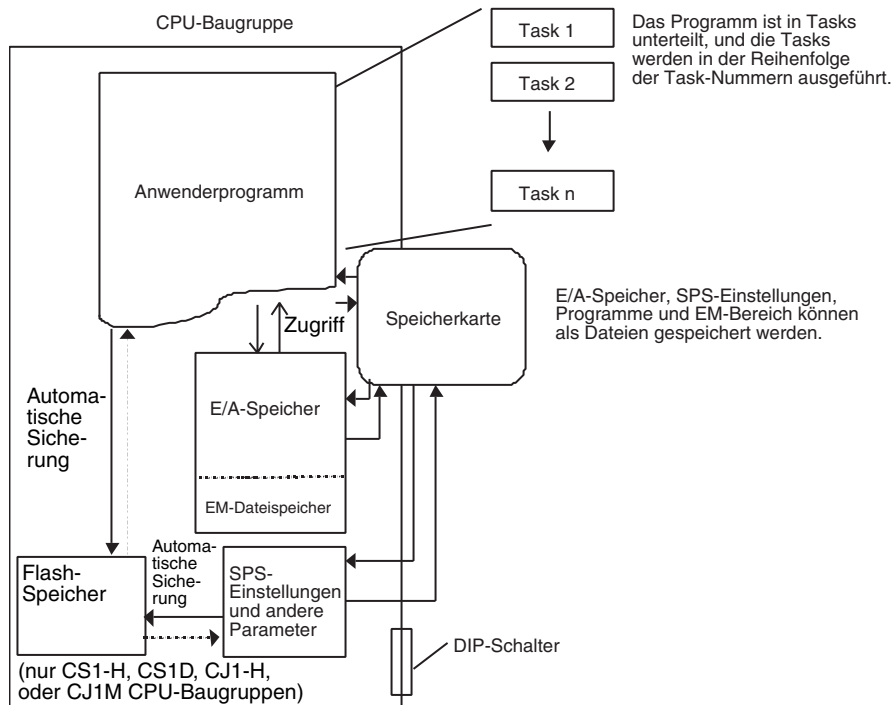
Tastenfolge



1-3 Interne Struktur der CPU-Baugruppe

1-3-1 Übersicht

In der folgenden Abbildung wird die interne Struktur der CPU-Baugruppe dargestellt.



Das Anwenderprogramm

Das Anwenderprogramm wird aus bis zu 288 Programm-Tasks, einschließlich Interrupt-Tasks, zusammengestellt. Die Tasks werden von der Programmiersoftware CX-Programmer in die CPU-Baugruppe übertragen.

Es gibt zwei Arten von Tasks: Die erste ist eine zyklische Task, die einmal pro Zyklus (maximal 32) ausgeführt wird, und die andere ist eine Interrupt-Task, die nur ausgeführt wird, wenn Interrupt-Bedingungen auftreten (maximal 256). Zyklische Tasks werden in numerischer Reihenfolge ausgeführt.

Hinweis

1. Mit den CS1-H, CJ1-H, CJ1M und CS1D CPU-Baugruppen können Interrupt-Tasks – genauso wie zyklische Tasks – auch zyklisch ausgeführt werden. Diese werden als „zyklische Zusatztasks“ bezeichnet. Die Gesamtzahl der Tasks, die zyklisch ausgeführt werden können, beträgt maximal 288.
2. Wenn Sie mit CX-Programmer arbeiten, müssen Sie für die CS1-H und CJ1-H CPU-Baugruppen Version 2.1 oder höher verwenden. Für die CJ1M CPU-Baugruppen (außer Low-End-Modelle) beziehungsweise CS1D bei Systemen mit Duplex-CPU muss Version 3.0 oder höher verwendet werden. Wenn Sie eine Low-End-CJ1M CPU-Baugruppe (CJ1M-CPU11/CPU21) einsetzen, müssen Sie die CX-Programmer-Version 3.3 oder höher verwenden. Wenn Sie eine CS1-H, CJ1-H, CJ1M bzw. CS1D CPU-Baugruppe Version 2.0 oder höher einsetzen, müssen Sie die CX-Programmer-Version 4.0 oder höher verwenden.

Programmbefehle, mit denen über Lese- und Schreibzugriff auf den E/A-Speicher zugegriffen wird, werden der Reihenfolge im Programm von oben beginnend ausgeführt. Nachdem alle zyklischen Tasks ausgeführt wurden, werden die E/A-Daten für alle Baugruppen aktualisiert, und der Zyklus wird, beginnend mit der zyklischen Task mit der niedrigsten Nummer, erneut gestartet.

Informationen zur E/A-Aktualisierung finden Sie auch im Abschnitt über den Betrieb der CPU-Baugruppe im *CS/CJ-Serie Bedienerhandbuch*.

E/A-Speicher	Der E/A-Speicher ist der Bereich des RAM, der für das Lesen und Schreiben aus dem Anwenderprogramm heraus verwendet wird. Er besteht aus einem Bereich, der beim Ein- beziehungsweise Ausschalten des Geräts gelöscht wird, und aus einem weiteren Bereich, dessen Daten in diesem Fall erhalten bleiben. Der E/A-Speicher ist zudem in einen Bereich, dessen Daten mit sämtlichen Baugruppen ausgetauscht werden, und in einen Bereich, der ausschließlich der internen Nutzung dient, unterteilt. Ein Datenaustausch mit allen Baugruppen erfolgt einmal pro Programmausführungszyklus oder wenn spezielle Befehle ausgeführt werden.
SPS-Setup	Das SPS-Setup dient dem Eingeben verschiedener Konfigurationseinstellungen oder anderer Einstellungen mit Hilfe von Software.
DIP-Schalter	DIP-Schalter dienen zur Festlegung von Konfigurationseinstellungen oder anderen Einstellungen.
Speicherkarten	Speicherkarten werden bei Bedarf dazu verwendet, um Daten, wie zum Beispiel Programme, E/A-Speicherdaten, das SPS-Setup und E/A-Kommentare, die mit Programmiergeräten erstellt wurden, zu speichern. Wenn die SPS eingeschaltet wird, können Programme und verschiedene Systemeinstellungen automatisch von der Speicherkarte übertragen werden (automatische Übertragung beim Einschalten).
Flash-Speicher (nur CS1-H, CJ1-H, CJ1M und CS1D CPU-Baugruppen)	Bei den CS1-H, CJ1-H, CJ1M und CS1D CPU-Baugruppen werden das Anwenderprogramm und die Daten der Parameterbereiche, wie zum Beispiel SPS-Setup, automatisch im integrierten Flash-Speicher gesichert, wenn der Anwender Daten an die CPU-Baugruppe überträgt. Damit ist ein Betrieb ohne Batterie und ohne Speicherkarte möglich. Der Inhalt des E/A-Speichers einschließlich des größten Teils des DM-Bereiches werden ohne Batterie nicht gesichert.

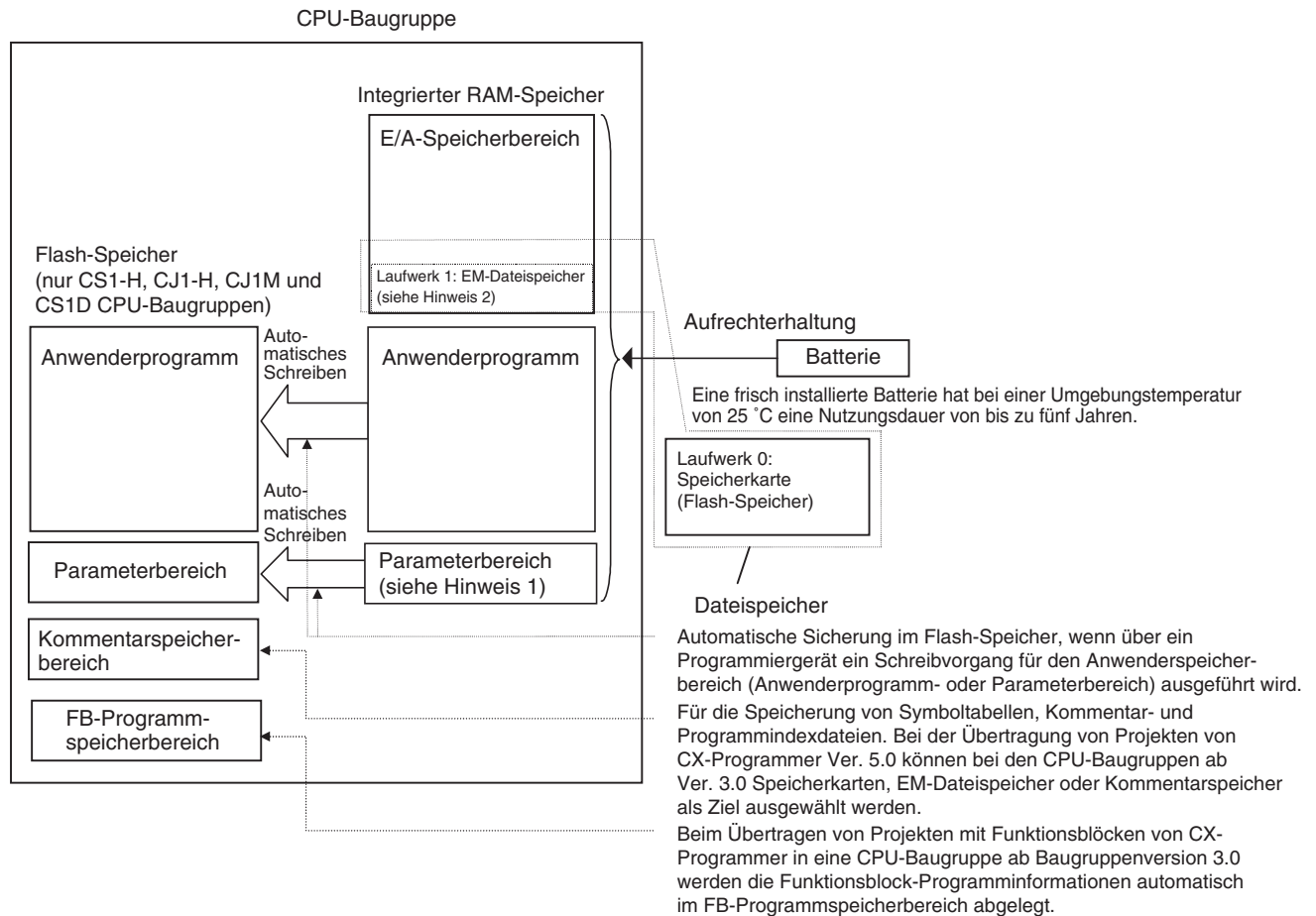
1-3-2 Blockdiagramm des Speichers einer CPU-Baugruppe

Bei der CS/CJ-Serie besteht der Speicher der CPU-Baugruppe (RAM) aus den folgenden Blöcken:

- Parameterbereich (SPS-Setup, registrierte E/A-Tabelle, Routing-Tabelle und CPU-Bus-Baugruppeneinstellungen)
- E/A-Speicherbereiche
- Anwenderprogramm

Daten aus dem Parameterbereich und dem E/A-Speicherbereich werden durch eine Batterie gesichert (CS-Serie: CS1W-BAT01, CJ1-H: CPM2A-BAT01) und gehen verloren, wenn die Batteriespannung zu gering ist.

Die CPU-Baugruppen CS1-H, CJ1-H, CJ1M und CS1D besitzen einen integrierten Flash-Speicher, in dem die Daten gesichert werden. Jedes Mal, wenn der Anwender über ein Programmiergerät (z. B. CX-Programmer oder Programmierkonsole) Daten in die CPU-Baugruppe schreibt, werden das Anwenderprogramm und Parameterbereichsdaten automatisch im eingebauten Flash-Speicher gesichert. Dies gilt auch bei folgenden Vorgängen: Datenübertragungen, Online-Bearbeitung, Übertragungen von Speicherkarten usw. Dies bedeutet, dass das Anwenderprogramm und die Daten des Parameterbereichs auch dann nicht verloren gehen, wenn die Batteriespannung abfällt.



- Hinweis**
1. Durch Umstellen von Schalter 1 des DIP-Schalterblocks an der Vorderseite der CPU-Baugruppe in die Stellung ON können der Parameterbereich und das Anwenderprogramm (d. h. der Anwenderspeicher) schreibgeschützt werden.
 2. Der EM-Dateispeicher ist Teil des EM-Bereichs, der im SPS-Setup in Dateispeicher umgewandelt wurde. Alle EM-Banken ab der spezifizierten Bank bis zum Ende des EM-Bereichs können nur als Dateispeicher für die Speicherung von Daten- und Programmdateien verwendet werden.
 3. Vergessen Sie nicht, die mitgelieferte Batterie (CS1W-BAT01) einzulegen, bevor Sie eine CS1 CPU-Baugruppe zum ersten Mal in Betrieb nehmen. Löschen Sie nach dem Einlegen der Batterie den RAM-Speicher der SPS (Parameterbereich, E/A-Speicherbereich und Anwenderprogramm) mit Hilfe eines Programmiergeräts.
 4. In den CPU-Baugruppen CS1-H, CJ1, CJ1-H, CJ1M und CS1D ist ab Werk eine Batterie eingesetzt. Es ist nicht nötig, den Speicher zu löschen und die Uhrzeit einzustellen.
 5. Die BKUP-Anzeige an der Vorderseite der CPU-Baugruppe leuchtet, während Daten in den Flash-Speicher geschrieben werden. Schalten Sie die Spannungsversorgung der CPU-Baugruppe erst aus, wenn die Datensicherung abgeschlossen ist (d. h. die BKUP-Anzeige erloschen ist). Weitere Informationen finden Sie im Abschnitt 6-6-11 *Flash-Speicher*.

1-4 Betriebsarten

1-4-1 Beschreibung der Betriebsarten

Die CPU-Baugruppe besitzt die folgenden Betriebsarten. Diese Betriebsarten gelten sowohl für das gesamte Anwenderprogramm als auch für alle Tasks.

PROGRAM-Betriebsart

In der PROGRAM-Betriebsart wird die Programmausführung angehalten. Die RUN-Anzeige leuchtet nicht. Diese Betriebsart wird verwendet, wenn das Programm bearbeitet wird oder andere Vorbereitungen ausgeführt werden, wie zum Beispiel:

- Registrieren der E/A-Tabelle.
- Ändern des SPS-Setups und anderer Einstellungen.
- Übertragen und Prüfen von Programmen.
- Zwangsweises Setzen beziehungsweise Rücksetzen von Bits, um die Verdrahtung und die Zuordnung von Bits zu prüfen.

In dieser Betriebsart werden keine zyklischen und Interrupt-Tasks ausgeführt, d. h. sie werden angehalten. Weitere Informationen zu Tasks finden Sie im Abschnitt 1-6 *Beschreibung von Tasks*.

Die E/A-Aktualisierung wird in der PROGRAM-Betriebsart ausgeführt. Weitere Informationen zur E/A-Aktualisierung finden Sie im *Bedienerhandbuch*.



VORSICHT

Die CPU-Baugruppe führt E/A-Aktualisierungen auch dann durch, wenn das Programm angehalten ist (d. h. in der PROGRAM-Betriebsart). Überprüfen Sie die Sicherheit sorgfältig im Voraus, bevor Sie den Status von Bits oder Worten des E/A-Speichers ändern, die E/A-Baugruppen, Spezial-E/A-Baugruppen oder CPU-Bus-Baugruppen zugeordnet sind. Änderungen an den einer Baugruppe zugeordneten Daten können ein unvorhergesehenes Verhalten der an die Baugruppen angeschlossenen Lasten (Maschinen) zur Folge haben. Die folgenden Bedienvorgänge können zu Änderungen von Bits und Worten des E/A-Speichers führen.

- Übertragen von E/A-Speicherdaten von einem Programmiergerät in die CPU-Baugruppe.
- Ändern der aktuellen Werte im Speicher von einem Programmiergerät aus.
- Zwangsweises Setzen oder Rücksetzen von Bits über ein Programmiergerät.
- Übertragen von E/A-Speicherdateien von einer Speicherkarte oder dem EM-Dateispeicher in die CPU-Baugruppe.
- Übertragen von E/A-Speicherdaten von einem Host-Computer oder von einer anderen SPS im Netzwerk.

MONITOR-Betriebsart

Die folgenden Vorgänge können mit Hilfe eines Programmiergerätes ausgeführt werden, während das Programm in der MONITOR-Betriebsart ausgeführt wird. Die RUN-Anzeige leuchtet. Diese Betriebsart wird dazu verwendet, Probeläufe des Programms auszuführen oder andere Anpassungen vorzunehmen.

- Online-Bearbeitung.
- Zwangsweises Setzen/Rücksetzen von Bits.
- Ändern von Werten im E/A-Speicher.

In dieser Betriebsart werden die für die Ausführung beim Systemstart ausgewählten Tasks (siehe Hinweis) und die mit dem Befehl TKON(820) ausführbar gemachten Tasks ausgeführt, sobald die Programmausführung bei den jeweiligen Tasknummern angelangt ist. Interrupt-Tasks werden ausgeführt, falls die für sie festgelegten Interrupt-Bedingungen vorliegen.

Hinweis

Die beim Systemstart auszuführenden Tasks werden in den Programmeigenschaften mit Hilfe von CX-Programmer spezifiziert.

RUN-Betriebsart

Diese Betriebsart wird für die normale Programmausführung verwendet. Die RUN-Anzeige leuchtet. In dieser Betriebsart sind einige Funktionen der Programmiergeräte, wie zum Beispiel Online-Bearbeitung, zwangsweises Setzen/Rücksetzen und Ändern von Werten im E/A-Speicher, deaktiviert, andere Funktionen der Programmiergeräte jedoch, wie zum Beispiel die Überwachung des Status der Programmausführung (Programmüberwachung und Überwachung des E/A-Speichers), stehen zur Verfügung.

Verwenden Sie diese Betriebsart für den normalen Systembetrieb. Die Ausführung der Tasks ist mit der in der MONITOR-Betriebsart identisch.

Weitere Informationen über die Funktionen, die in den einzelnen Betriebsarten zur Verfügung stehen, finden Sie im Abschnitt 10-2, *Betriebsarten der CPU-Baugruppe*, im *Bedienerhandbuch*.

1-4-2 Initialisierung des E/A-Speichers

In folgender Tabelle wird angezeigt, welche Datenbereiche gelöscht werden, wenn ein Wechsel der Betriebsart von der PROGRAM-Betriebsart zur RUN-/MONITOR-Betriebsart oder umgekehrt erfolgt.

Betriebsartenwechsel	Nicht erhaltene Bereiche (siehe Hinweis 1)	Erhaltene Bereiche (siehe Hinweis 2)
RUN/MONITOR → PROGRAM	Gelöscht (Hinweis 3)	Bleibt erhalten
PROGRAM → RUN/MONITOR	Gelöscht (Hinweis 3)	Bleibt erhalten
RUN ↔ MONITOR	Bleibt erhalten	Bleibt erhalten

Hinweis

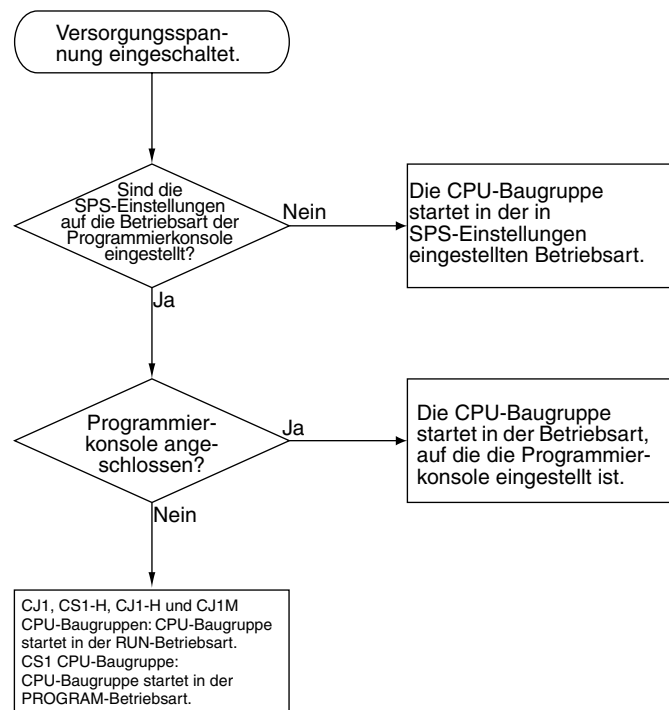
1. Nicht erhaltene Bereiche: CIO-Bereich, Arbeitsbereich, Zeitgeber-Istwerte, Zeitgeberablaufmerker, Indexregister, Datenregister, Task-Merker und Bedingungsmerker.
(Der Status einiger Adressen im Zusatz-Systembereich bleibt erhalten, während andere gelöscht werden.)
2. Erhaltene Bereiche: Haftmerker-Bereich, DM-Bereich, EM-Bereich, Zähler-Istwerte und Zählerablaufmerker.
3. Daten im E/A-Speicher bleiben erhalten, wenn der E/A-Speicher-Haftmerker (A50012) auf EIN gesetzt ist. Wenn der E/A-Speicher-Haftmerker (A50012) auf EIN gesetzt ist und der Betrieb aufgrund eines schwerwiegenden Fehlers (einschließlich FALS(007)) gestoppt wird, bleibt der Inhalt des E/A-Speichers erhalten, jedoch werden alle Ausgänge aller Ausgangsbaugruppen ausgeschaltet.

1-4-3 Betriebsart beim Einschalten

Weitere Informationen zu den Einstellungen für die Betriebsart beim Einschalten der CPU-Baugruppe finden Sie im *Bedienerhandbuch*.

Hinweis Die CJ1, CS1-H, CJ1-H, CJ1M und CS1D CPU-Baugruppen werden in der RUN-Betriebsart gestartet, sofern keine Programmierkonsole angeschlossen ist. Dies unterscheidet sie von einer CS1 CPU-Baugruppe, die in der werksseitigen Einstellung in der PROGRAM-Betriebsart gestartet wird, sofern keine Programmierkonsole angeschlossen ist.

Bedingungen	CS1 CPU-Baugruppe	CJ1, CS1-H, CJ1-H, CJ1M und CS1D CPU-Baugruppen
In den SPS-Einstellungen ist die Start-Betriebsart auf „Programmierkonsole verwenden“ festgelegt, es ist jedoch keine Programmierkonsole angeschlossen.	PROGRAM-Betriebsart	RUN-Betriebsart



1-5 Programme und Tasks

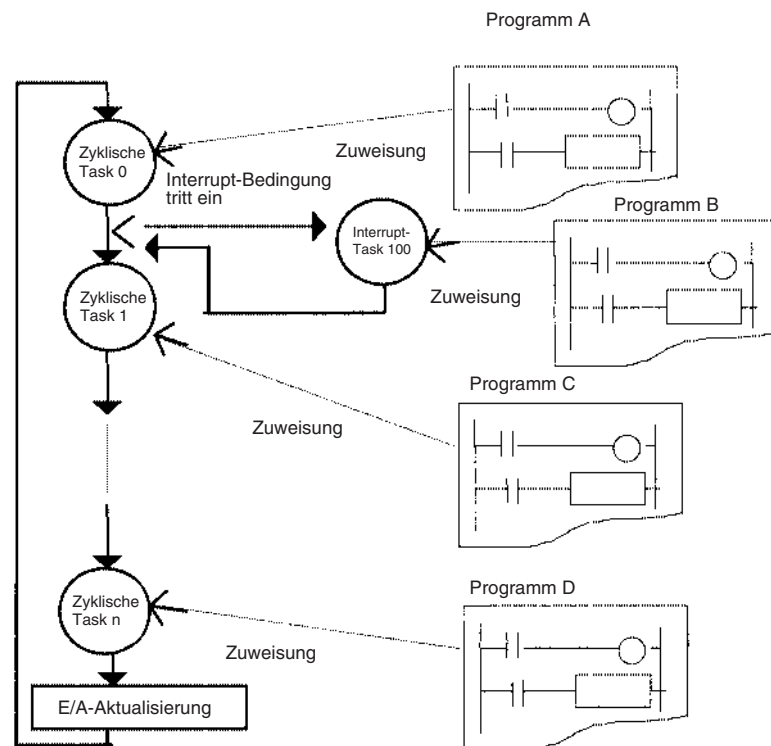
Tasks bestimmen die Abfolge und die Interrupt-Bedingungen, unter denen die einzelnen Programme ausgeführt werden. Sie können allgemein in die folgenden Typen unterteilt werden:

- 1,2,3...**
1. Nacheinander ausgeführte Tasks, die als zyklische Tasks bezeichnet werden.
 2. Beim Eintreten von Interrupt-Bedingungen ausgeführte Tasks, die als Interrupt-Tasks bezeichnet werden.

Hinweis Bei den CS1-H, CJ1-H, CJ1M und CS1D CPU-Baugruppen können Interrupt-Tasks – genauso wie zyklische Tasks – auch zyklisch ausgeführt werden. Diese werden als „zyklische Zusatztasks“ bezeichnet.

Die zyklischen Tasks zugeordneten Programme werden der Reihe nach entsprechend ihrer Tasknummer ausgeführt und die E/A-Daten werden einmal pro Zyklus aktualisiert, nachdem sämtliche Tasks (genauer gesagt, alle Tasks im ausführbaren Status) ausgeführt wurden. Falls während der Verarbeitung der zyklischen Tasks eine Interrupt-Bedingung eintritt, wird die Ausführung der zyklischen Task unterbrochen, und das der Interrupt-Task zugeordnete Programm wird ausgeführt.

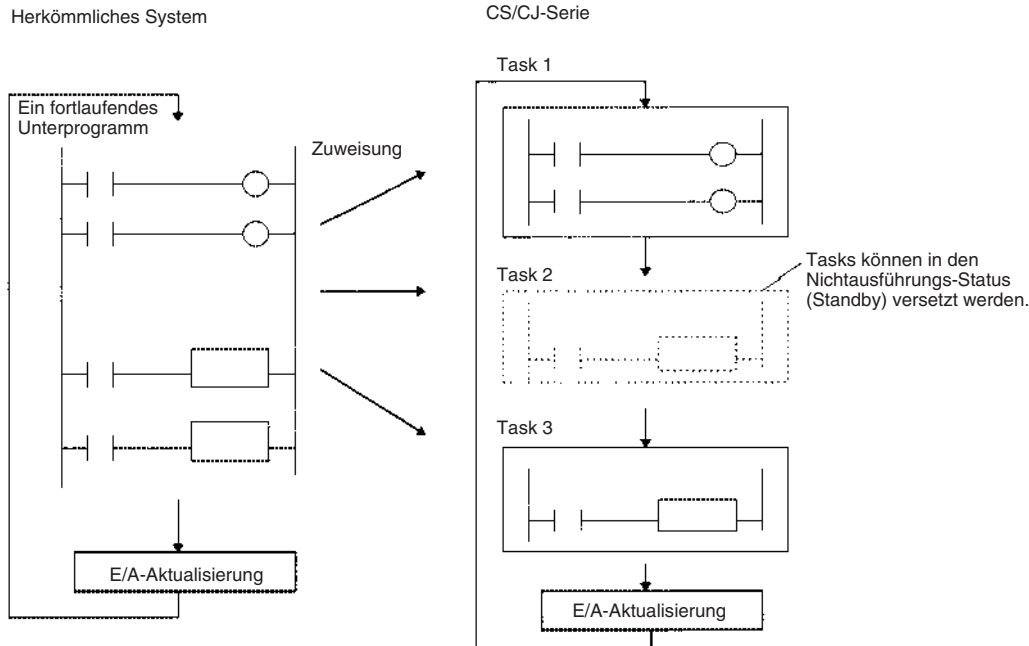
Weitere Informationen zur E/A-Aktualisierung finden Sie auch im Abschnitt über den Betrieb der CPU-Baugruppe im *CS/CJ-Serie Bedienerhandbuch*.



Bei dem oben aufgeführten Beispiel würde das Programm in der folgenden Reihenfolge ausgeführt: Start von A, B, Rest von A, C und dann D. Hierbei wird davon ausgegangen, dass während der Ausführung von Programm A die Interrupt-Bedingung die Interrupt-Task 100 erfüllt wird. Wenn die Ausführung von Programm B abgeschlossen ist, wird der Rest von Programm A ab dem Punkt ausgeführt, an dem die Ausführung unterbrochen wurde.

Bei älteren SPS von OMRON wird aus mehreren fortlaufenden Teilen ein kontinuierliches Programm gebildet. Die den einzelnen Tasks zugeordneten Programme sind Einzelprogramme, die mit einem END-Befehl enden, genau wie die Einzelprogramme früherer SPS.

Ein Merkmal der zyklischen Tasks ist, dass sie durch die Task-Steuerbefehle aktiviert (in den ausführbaren Status versetzt) und deaktiviert (in den Standby-Status versetzt) werden können. Dies bedeutet, dass mehrere Programmkomponenten zu einer Task zusammengestellt werden können und dass dann nur bestimmte Programme (Tasks), die für das jeweilig aktuelle Produktmodell oder den zu dem jeweiligen Zeitpunkt auszuführenden Prozess benötigt werden, ausgeführt werden können (Programm-Schritt-Schaltung). Da also nur die jeweils tatsächlich benötigten Programme ausgeführt werden, wird die Leistung (Zykluszeit) wesentlich gesteigert.



Eine Task, die ausgeführt wurde, wird in aufeinander folgenden Zyklen ausgeführt und eine Task im Standby-Status bleibt auch in den folgenden Zyklen im Standby-Status, sofern sie nicht von einer anderen Task ausgeführt wird.

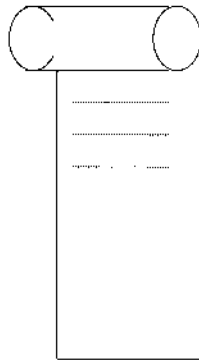
Hinweis

Im Gegensatz zu früheren Programmen, die mit dem Lesen einer Rolle verglichen werden können, kann das Ausführen von Tasks jetzt mit dem Lesen eines Satzes einzelner Karten verglichen werden.

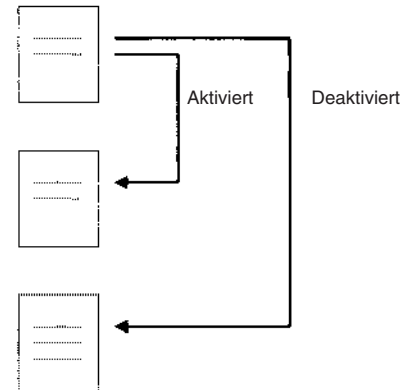
- Alle Karten werden in einer vorher festgelegten Reihenfolge beginnend mit der niedrigsten Nummer gelesen.
- Alle Karten sind entweder als aktiv oder als inaktiv gekennzeichnet. Als inaktiv gekennzeichnete Karten werden übersprungen. (Die Aktivierung beziehungsweise Deaktivierung der Karten erfolgt durch Task-Steuerbefehle.)

- Eine aktivierte Karte bleibt aktiviert und wird immer wieder gelesen. Eine deaktivierte Karte bleibt deaktiviert und wird übersprungen, bis sie durch eine andere Karte wieder aktiviert wird.

Herkömmliches Programm:
Wie eine Schriftrolle



Programm der CS/CJ-Serie:
Wie eine Reihe von Karten, die durch andere
Karten aktiviert oder deaktiviert werden können.



1-6 Beschreibung von Tasks

Tasks sind im Allgemeinen in die folgenden Typen unterteilt:

1,2,3...

1. Zyklische Tasks (max. 32)

Tasks, die einmal pro Zyklus ausgeführt werden, falls sie ausführbar sind. Bei Bedarf kann auch die Ausführung von zyklischen Tasks deaktiviert werden.

2. Interrupt-Tasks

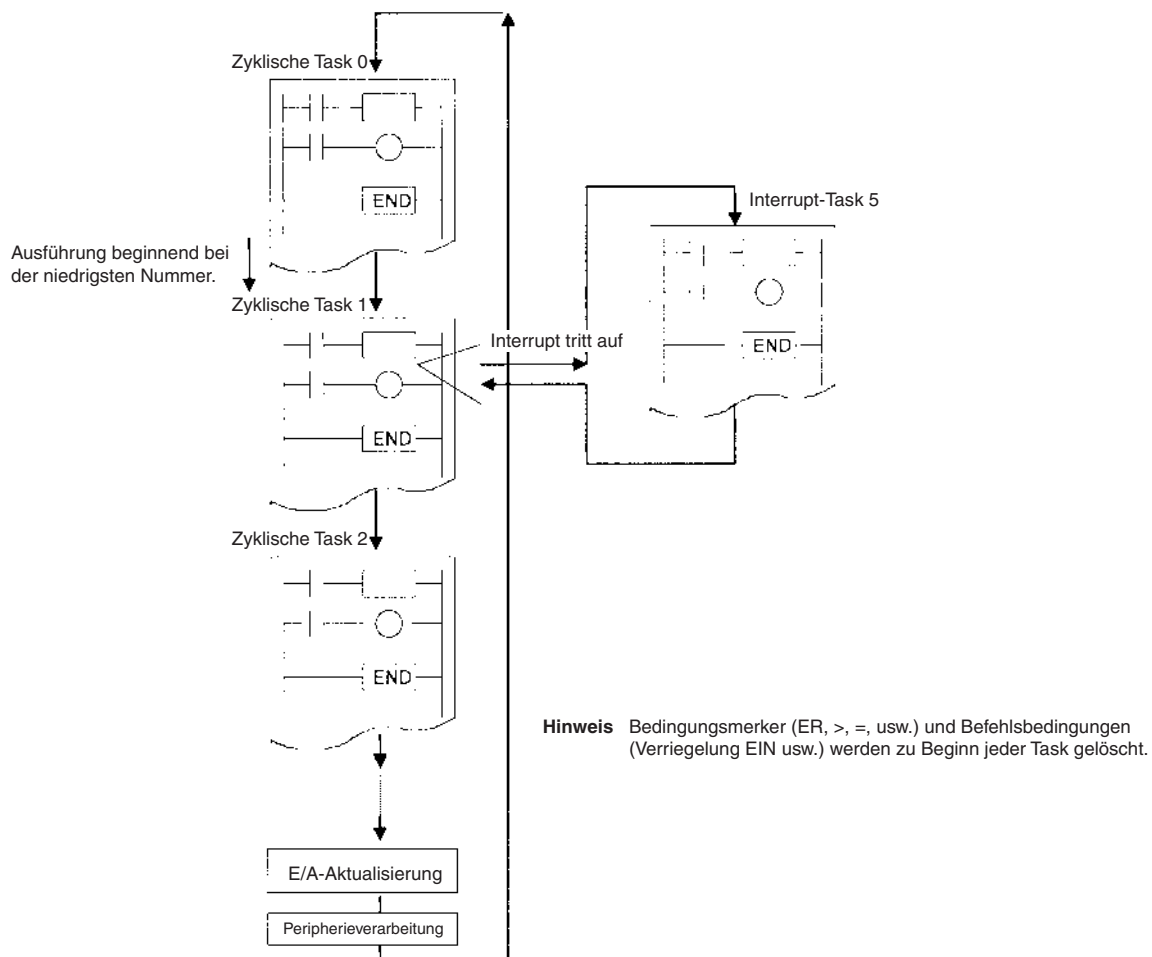
Tasks, die ausgeführt werden, wenn Interrupt-Bedingungen eintreten, unabhängig davon, ob gerade eine zyklische Task ausgeführt wird oder nicht. Interrupt-Tasks (siehe Hinweise 1 und 2) können in die folgenden vier Typen unterteilt werden (fünf Typen einschließlich der zyklischen Zusatz-Tasks bei den CS1-H, CJ1-H, CJ1M und CS1D CPU-Baugruppen):

- Interrupt-Task für Spannungsfehler (nicht unterstützt von den CS1D CPU-Baugruppen für Duplex-CPU-Systeme):
wird ausgeführt, wenn die Spannungsversorgung unterbrochen wird. (max. 1)
- Zeitgesteuerte Interrupt-Task für Intervallzeitgeber (nicht unterstützt von den CS1D CPU-Baugruppen für Duplex-CPU-Systeme):
wird in spezifizierten Zeitabständen ausgeführt. (max. 2)
- E/A-Interrupt-Task (nicht unterstützt von den CJ1 CPU-Baugruppen und CS1D CPU-Baugruppen für Duplex-CPU-Systeme):
wird ausgeführt, wenn ein Eingang einer Interrupt-Eingangsbau-
gruppe auf EIN wechselt (max. 32).
- Externe Interrupt-Task (nicht unterstützt von den CJ1 CPU-Baugrup-
pen und CS1D CPU-Baugruppen für Duplex-CPU-Systeme):
wird ausgeführt (max. 256), wenn er von einer Spezial-E/A-Bau-
gruppe, einer CPU-Bus-Baugruppe oder einem internen Modul
(nur CS-Serie) angefordert wird.
- Zyklische Zusatz-Task (nur unterstützt von den CS1-H, CJ1-H, CJ1M
und CS1D CPU-Baugruppen):
Interrupt-Tasks, die als zyklische Tasks behandelt werden. Zykli-
sche Zusatztasks werden, solange sie sich im ausführbaren Status
befinden, einmal pro Zyklus ausgeführt.

Mit CX-Programmer können insgesamt 288 Tasks mit 288 Programmen erstellt und gesteuert werden. Dazu gehören bis zu 32 zyklische Tasks und 256 Interrupt-Tasks.

- Hinweis**
1. Gegenwärtig werden von CJ1 CPU-Baugruppen keine E/A-Interrupt-Tasks und keine externen Interrupt-Tasks unterstützt. Demzufolge beträgt die Höchstzahl der Tasks für eine CJ1 CPU-Baugruppe 35, d. h. 32 zyklische Tasks und 3 Interrupt-Tasks. Die Gesamtzahl der Programme, die erstellt und gesteuert werden können, beträgt ebenfalls 35.
 2. Die CS1D CPU-Baugruppen unterstützen keine Interrupt-Tasks. Allerdings können bei CS1D CPU-Baugruppen Interrupt-Tasks als zyklische Zusatztasks verwendet werden.

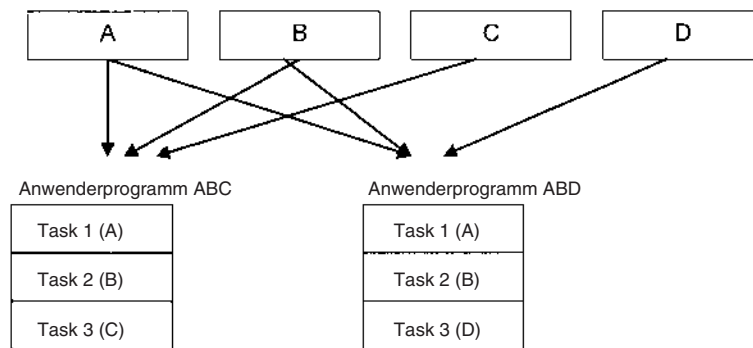
Jedes Programm ist durch eine individuelle Programmeigenschaft, die mit Hilfe von CX-Programmer eingestellt wird, 1:1 einer Task zugeordnet.



Programmstruktur

Für die Erstellung von Programmen können je nach Bedarf Standardunterprogramme erstellt und zu Tasks zugeordnet werden. Dies bedeutet, dass Programme in Modulen (Standardkomponenten) erstellt werden können und das Debugging bei jeder Task einzeln durchgeführt werden kann.

Standard-Unterprogramme



Beim Erstellen von modularen Programmen kann die Standardisierung durch die Angabe von Adressen mit Hilfe von Symbolen erleichtert werden.

Ausführbarer und Standby-Status

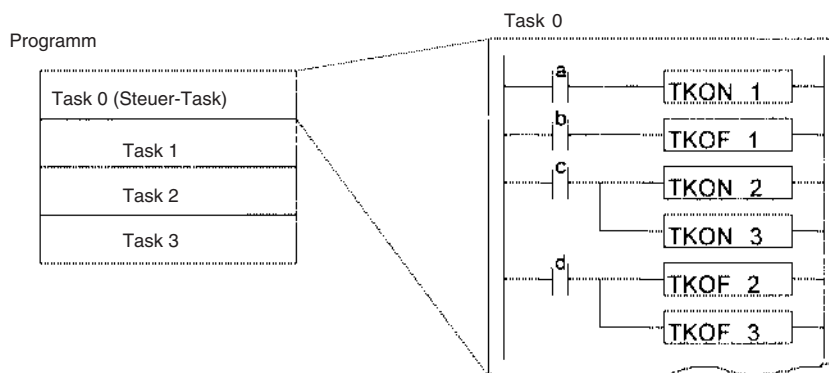
Um eine andere Task in den ausführbaren oder den Standby-Status zu versetzen, können in einer Task die Befehle TASK ON oder TASK OFF (TKON(820) bzw. TKOF(821)) ausgeführt werden.

Befehle in Tasks, die im Standby-Status sind, werden nicht ausgeführt, aber ihr E/A-Status bleibt erhalten. Wenn eine Task wieder in den ausführbaren Status versetzt wird, werden die Befehle mit dem erhalten gebliebenen E/A-Status ausgeführt.

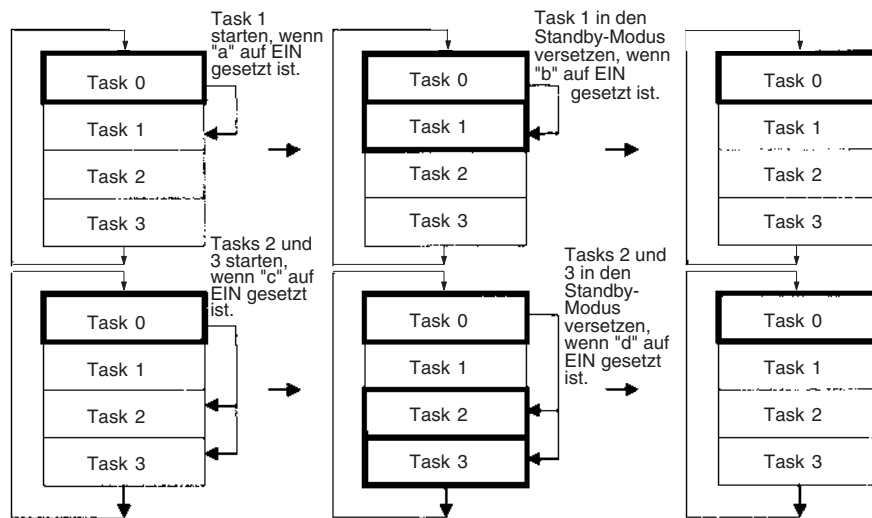
Beispiel: Programmierung mit einer Steuer-Task

In diesem Beispiel ist Task 0 eine Steuer-Task, die bei Beginn des Betriebs zuerst ausgeführt wird. Mit CX-Programmer (allerdings nicht mit einer Programmierkonsole) kann für weitere Tasks festgelegt werden, ob sie bei Beginn des Betriebs ausgeführt werden sollen oder nicht.

Wenn die Ausführung des Programms begonnen hat, können die Tasks durch die Befehle TKON(820) und TKOF(821) gesteuert werden.

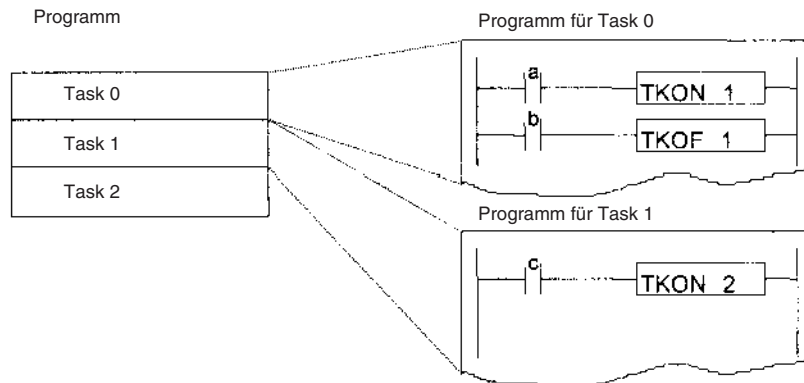


Beispiel: Task 0 ist auf die Ausführung bei Betriebsstart eingestellt (in den Programmeigenschaften mit CX-Programmer festgelegt).
 Task 1 ist ausführbar, wenn "a" auf EIN gesetzt ist.
 Task 1 wird in den Standby-Modus versetzt, wenn "b" auf EIN gesetzt ist.
 Tasks 2 und 3 sind ausführbar, wenn "c" auf EIN gesetzt ist.
 Tasks 2 und 3 werden in den Standby-Modus versetzt, wenn "d" auf EIN gesetzt ist.

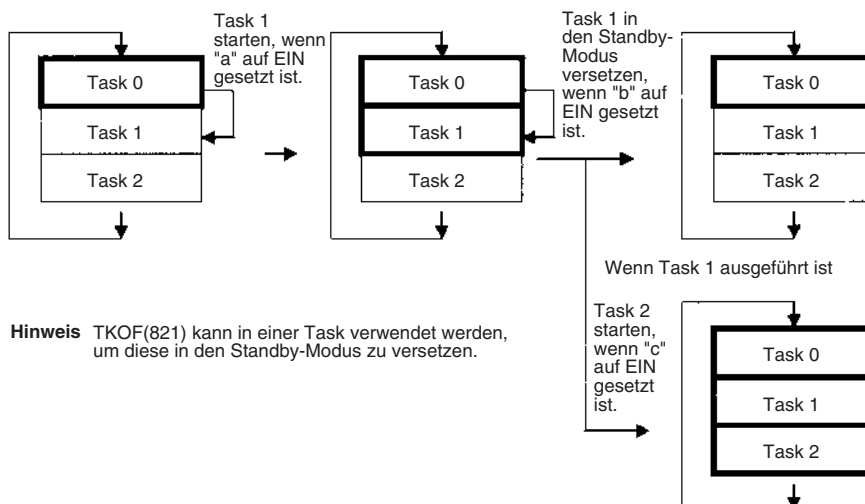


Beispiel: Jede Task wird von einer anderen Task gesteuert

In diesem Beispiel wird jede Task von einer anderen Task gesteuert.



Beispiel: Task 0 ist auf die bedingungslose Ausführung bei Betriebsstart eingestellt.
 Task 1 ist ausführbar, wenn "a" auf EIN gesetzt ist.
 Task 1 wird in den Standby-Modus versetzt, wenn "b" auf EIN gesetzt ist.
 Task 2 ist ausführbar, wenn "c" auf EIN gesetzt ist und Task 1 ausgeführt wird.



Hinweis TKOF(821) kann in einer Task verwendet werden, um diese in den Standby-Modus zu versetzen.

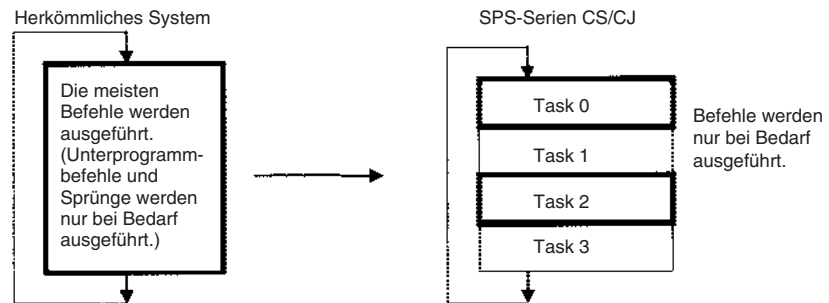
Task-Ausführungszeit

Während eine Task im Standby-Status ist, werden die in dieser Task enthaltenen Befehle nicht ausgeführt, deren Ausschaltbefehl-Ausführungszeit wird also nicht zur Zykluszeit addiert.

Hinweis

Aus dieser Sicht entsprechen Befehle in einer Task, die im Standby-Status ist, den Befehlen in einem übersprungenen Programmabschnitt (JMP-JME).

Da die Zykluszeit durch Befehle in einer nicht ausführbaren Task nicht verlängert wird, kann die Gesamtsystemleistung wesentlich erhöht werden, wenn das System in eine übergeordnete Steuer-Task und einzelne Tasks unterteilt wird, von denen die letzteren nur dann ausgeführt werden, wenn dies erforderlich ist.



ABSCHNITT 2

Programmierung

In diesem Abschnitt finden Sie grundlegende Informationen zum Schreiben, Überprüfen sowie zur Eingabe von Programmen.

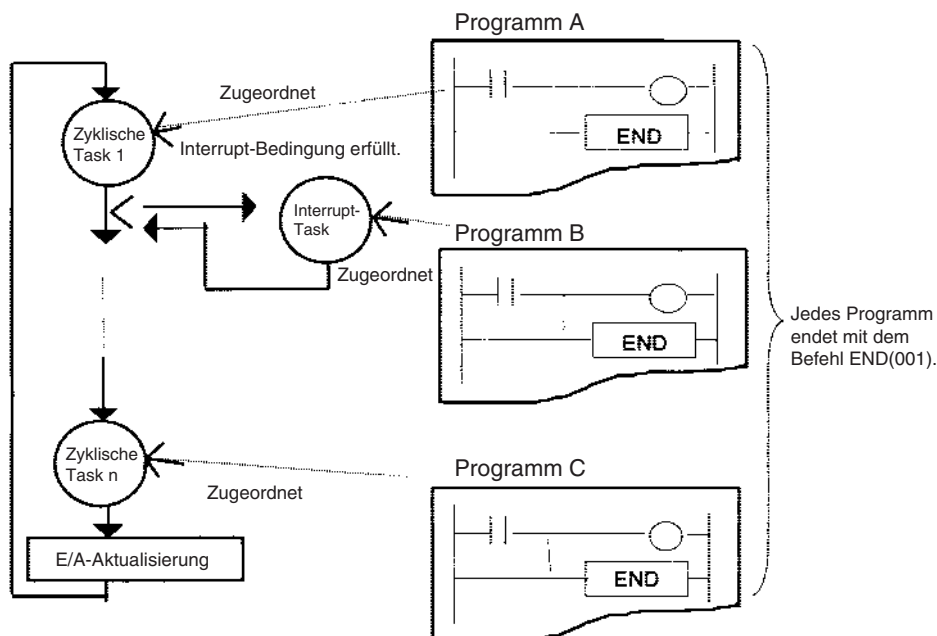
2-1	Grundbegriffe	20
2-1-1	Programme und Tasks.	20
2-1-2	Grundlegende Informationen zu Befehlen	21
2-1-3	Befehlsposition und Ausführungsbedingungen	23
2-1-4	Adressierung im E/A-Speicherbereich	24
2-1-5	Spezifizieren von Operanden	25
2-1-6	Datenformate	30
2-1-7	Befehlsvarianten	34
2-1-8	Ausführungsbedingungen	34
2-1-9	Zeitliches Verhalten von E/A-Befehlen	37
2-1-10	Aktualisierungszeitpunkt	39
2-1-11	Programmkapazität	42
2-1-12	Grundbegriffe der SPS-Programmierung	42
2-1-13	Eingabe in AWL	47
2-1-14	Programmierbeispiele	50
2-2	Sicherheitshinweise	55
2-2-1	Bedingungsmerker	55
2-2-2	Spezielle Programmabschnitte	60
2-3	Überprüfen von Programmen	64
2-3-1	Fehler während der Eingabe in das Programmiergerät	64
2-3-2	Programmprüfungen mit CX-Programmer	64
2-3-3	Prüfung der Programmausführung	66
2-3-4	Überprüfen schwerwiegender Fehler	68

2-1 Grundbegriffe

2-1-1 Programme und Tasks

SPS der CS/CJ-Serie führen SPS-Programme aus, die in Tasks enthalten sind. Das SPS-Programm in den einzelnen Tasks endet mit dem Befehl END(001), wie bei allen anderen SPS auch.

Tasks werden verwendet, um die Ausführungsreihenfolge der SPS-Programme und die Bedingungen für die Ausführung von Interrupts festzulegen.



In diesem Abschnitt werden die Grundlagen beschrieben, die zum Schreiben von Programmen für die CS/CJ-Serie erforderlich sind. Weitere Informationen zu Tasks in Verbindung mit SPS-Programmen finden Sie unter **ABSCHNITT 4 Tasks**.

Hinweis Tasks und Programmiergeräte

Tasks werden auf Programmiergeräten wie im Folgenden beschrieben gehandhabt. Weitere Informationen dazu finden Sie unter *4-4 Task-Programmierung mit Programmiergeräten* und im *CS/CJ-Serie Programmierkonsolen Bedienerhandbuch (W341)* sowie im *CX-Programmer Bedienerhandbuch*.

CX-Programmer

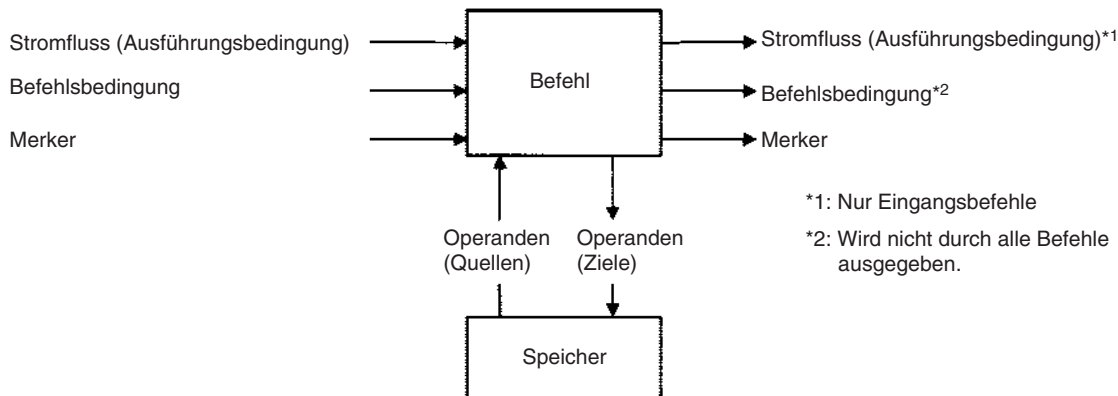
CX-Programmer wird zum Bestimmen von Tasktypen und Tasknummern als Attribute einzelner Programme verwendet.

Programmierkonsole

Über Programmierkonsolen kann auf Programme zum Bearbeiten zugegriffen werden, indem CT00 bis CT 31 für zyklische Tasks und IT00 bis IT255 für Interrupt-Tasks spezifiziert werden. Wenn der Speicher über eine Programmierkonsole gelöscht wird, kann in einem neuen Programm nur die zyklische Task 0 (CT00) geschrieben werden. Verwenden Sie CX-Programmer zum Erstellen der zyklischen Tasks 1 bis 31 (CT01 bis CT31).

2-1-2 Grundlegende Informationen zu Befehlen

Programme setzen sich aus Befehlen zusammen. In der folgenden Abbildung wird die Struktur der Eingaben und Ausgaben eines Befehls dargestellt.

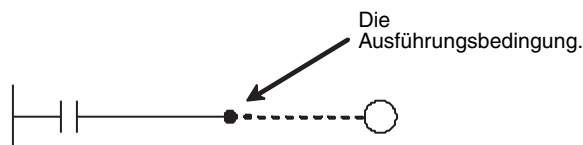


Ausführungsbedingung

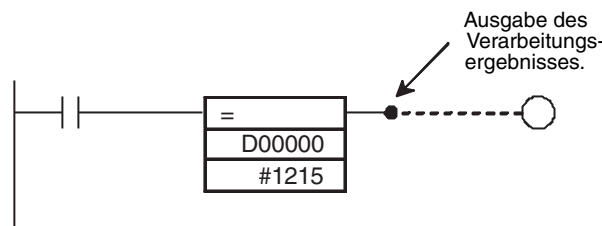
Die Ausführungsbedingung wird bei der normalen Ausführung von Programmen zur Steuerung der Ausführung der Befehle verwendet. In einem SPS-Programm ist der Status der Ausführungsbedingung entscheidend für die Ausführung des nachstehenden Befehls.

Eingabebefehle

- Der Befehl Laden kennzeichnet einen logischen Start und erzeugt die Ausführungsbedingung.

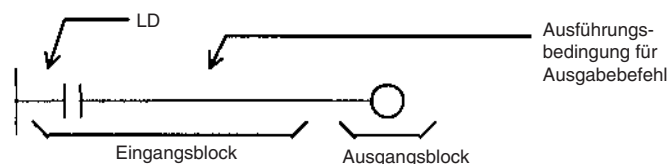


- Die Zwischenbefehle werden abhängig vom Status der Eingangs-Ausführungsbedingung verarbeitet und geben das Verarbeitungsergebnis an einen Zwischen- oder Ausgabebefehl weiter.



Ausgabebeefehle

Ausgabebeefehle führen alle Funktionen je nach Status der Ausführungsbedingung aus.



Sequenzsteuerbefehle

Folgende Sequenzsteuerbefehle steuern die Ausführung oder Nicht-Ausführung einer Sequenz von Befehlen in Programmabschnitten. Sequenzsteuerbefehle haben bei der Entscheidung zur Ausführung eines Befehls eine höhere Priorität als normale Befehls-Ausführungsbedingungen. Ein Befehl wird je nach Ausführung eines Sequenzsteuerbefehls möglicherweise nicht oder anders ausgeführt. Befehlsausführungsbedingungen aufgrund von Sequenzsteuerbefehlen werden zu Beginn jeder Task zurückgesetzt (aufgehoben), d. h., dass sie beim Wechsel der Task zurückgesetzt werden.

Die folgenden Befehle werden paarweise verwendet, um bestimmte Befehlsausführungsbedingungen zu setzen oder aufzuheben. Diese Befehlspaare müssen in derselben Task enthalten sein.

Sequenzsteuerbefehle	Beschreibung	Befehl am Anfang	Befehl am Ende
Ausführung gesperrt	Eine Sperre schaltet Teile des Programms aus. Spezielle Bedingungen, wie das Deaktivieren von Ausgabebits, das Zurücksetzen von Zeitgebern und das Halten von Zählerwerten, sind aktiviert.	IL(002)	ILC(003)
Ausführung von BREAK(514)	Beendet die Schleife FOR(512) - NEXT(513) während der Ausführung. (Verhindert die Ausführung aller Befehle bis zum nächsten NEXT(513)-Befehl.)	BREAK(514)	NEXT(513)
	Führt einen Sprungbefehl JMP0(515) zu JME0(516) aus.	JMP0(515)	JME0(516)
Blockprogramm-ausführung	Führt einen Programmblock von BPRG(096) bis BEND(801) aus.	BPRG(096)	BEND(801)

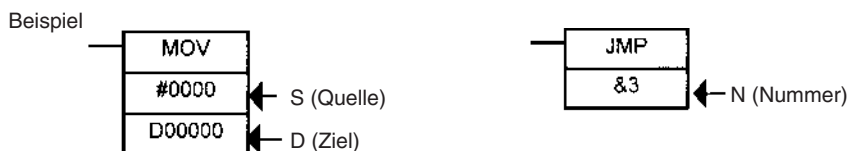
Merker

In diesem Zusammenhang ist ein Merker ein Bit, das als Schnittstelle zwischen Befehlen dient.

Eingabemerker	Ausgabemerker
- Flankenauswertungsmerker Flankenauswertungs-Ergebnismerker. Die Zustände der Merker dienen als automatische Eingaben für alle Befehle, die an der steigenden oder fallenden Signalfanke ausgeführt werden, sowie die Befehle DIFU(013)/DIFD(014). - Übertragsmerker (CY) Der Übertragsmerker wird als unbestimmter Operand bei Datenverschiebungsbefehlen sowie Additions-/Subtraktionsbefehlen verwendet. - Merker für Spezialbefehle Dazu gehören u. a. Teach-In-Merker für FPD(269)-Befehle und Merker für die Aktivierung der Netzwerk-kommunikation.	- Flankenauswertungsmerker Flankenauswertungs-Ergebnismerker. Die Zustände der Merker dienen als automatische flankengesteuerte Ausgaben von allen Befehlen, die an der steigenden oder fallenden Signalfanke ausgeführt werden, sowie der Befehle UP(521)/DOWN(522). - Bedingungsmerker Zu den Bedingungsmerker gehören die Immer-EIN/AUS-Merker sowie Merker, die durch die Ergebnisse einer Befehlsausführung aktualisiert werden. Bei Anwenderprogrammen können diese Merker anstelle von Adressen durch symbolische Bezeichnungen spezifiziert werden, wie z. B. P_ER, P_CY, P_GT, P_EQ, P_LT. - Merker für Spezialbefehle Zu diesen Merker gehören u. a. Speicherkarten-Befehlsmerker und Merker für abgeschlossene MSG(046)-Befehlsausführung.

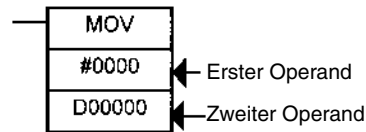
Operanden

Operanden legen vorgegebene Befehlsparameter fest (Felder in Kontaktplänen), die zur Spezifikation der Inhalte von E/A-Speicherbereichen oder Konstanten verwendet werden. Ein Befehl kann durch die Eingabe einer Adresse oder Konstante als Operand ausgeführt werden. Operanden werden als Quell-, Ziel- oder Nummer-Operanden klassifiziert.



Operandentypen		Operanden-symbol	Beschreibung	
Quelle (Source)	Spezifiziert die Adresse der zu lesenden Daten oder eine Konstante.	S	Quelloperand	Quelloperand außer Steuerdaten (C)
		C	Steuerdaten	Kombinierte Daten in einem Quell-operanden, die je nach Bitstatus eine andere Bedeutung besitzen.
Ziel (Ergebnisse)	Spezifiziert die Adresse, in die die Daten geschrieben werden.	D (R)	---	
Nummer	Spezifiziert eine bestimmte Nummer, die vom Befehl verwendet wird, beispielsweise eine Jump-Nummer oder die Nummer eines Unterprogramms.	N	---	

Hinweis Operanden werden auch, im Befehl von oben nach unten, als erster Operand, zweiter Operand usw. bezeichnet.



2-1-3 Befehlsposition und Ausführungsbedingungen

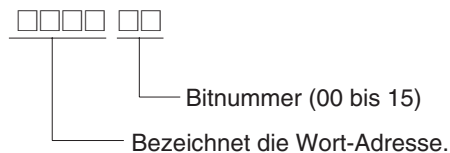
In der folgenden Tabelle werden die möglichen Positionen für Befehle im Kontaktplanprogramm aufgeführt. Befehle sind in Gruppen aufgeteilt, je nachdem, ob für sie Ausführungsbedingungen erforderlich sind oder nicht. Weitere Informationen zu einzelnen Befehlen finden Sie in *ABSCHNITT 3 Befehlsfunktionen*.

Befehlstyp		Mögliche Position	Ausführungsbedingung	Kontaktplan	Beispiele
Eingangsbefehle	Logischer Start (Lade-Befehl)	Direkt mit der linken Stromschiene verbunden oder am Anfang eines Befehlsblocks.	Nicht erforderlich		LD, LD TST(350), LD > (und andere Symbolvergleichsbefehle)
	Zwischenbefehle	Zwischen dem logischen Start und einem Ausgabebefehl.	Erforderlich		AND, OR, AND TEST(350), AND > (und andere AND-Symbolvergleichsbefehle), UP(521), DOWN(522), NOT(520) usw.
Ausgabebefehle		Direkt mit der rechten Stromschiene verbunden.	Erforderlich		Die meisten Befehle, einschließlich OUT und MOV(021).
			Nicht erforderlich		END(001), JME(005), FOR(512), ILC(003) usw.

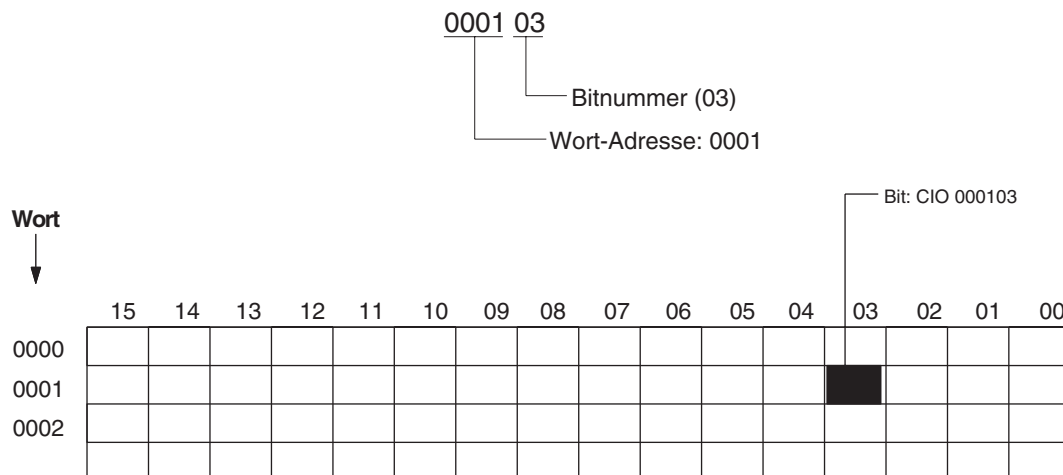
- Hinweis**
1. Es existiert eine weitere Gruppe von Befehlen, die eine ganze Folge von AWL-Befehlen auf Basis einer einzelnen Eingabe ausführen. Diese werden Blockprogrammierungsbefehle genannt. Weitere Informationen dazu finden Sie im *CS/CJ-Serie CPU-Baugruppen Befehlsreferenzhandbuch*.
 2. Wenn ein Befehl, der eine Ausführungsbedingung benötigt, ohne logischen Startbefehl direkt mit der linken Stromschiene verbunden ist, tritt bei der Überprüfung des Programms auf einem Programmiergerät (CX-Programmer oder Programmierkonsole) ein Fehler auf.

2-1-4 Adressierung im E/A-Speicherbereich

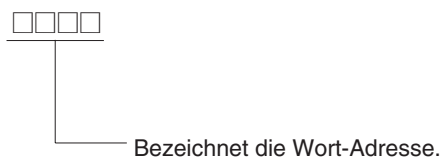
Bit-Adressen



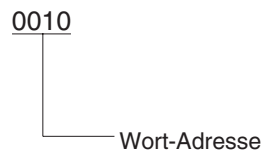
Beispiel: Die Adresse von Bit 03 in Wort 0001 im CIO-Bereich ergibt sich wie unten gezeigt. Diese Adresse wird in diesem Handbuch als „CIO 000103“ angegeben.



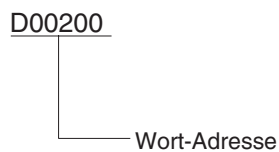
Wort-Adressen



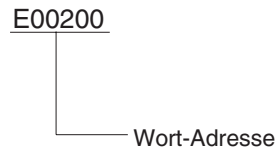
Beispiel: Die Adresse der Bits 00 bis 15 in Wort 0010 im CIO-Bereich ergibt sich wie unten gezeigt. Diese Adresse wird in diesem Handbuch als „CIO 0010“ angegeben.



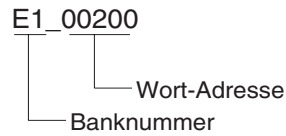
Adressen in den DM- und EM-Bereichen wird ein „D“ bzw. „E“ vorangestellt, wie im Folgenden für die Adresse D00200 dargestellt.



Beispiel: Die Adresse für Wort 200 in der Bank 0 des erweiterten Datenspeichers (EM) lautet folgendermaßen:



Die Adresse für Wort 200 in Bank 1 des erweiterten Datenspeichers lautet folgendermaßen:



2-1-5 Spezifizieren von Operanden

Operand	Beschreibung	Schreibweise	Anwendungsbeispiele
Spezifizieren von Bit-Adressen	Die Wort- und Bitnummern werden zur Spezifikation eines Bits direkt spezifiziert. Bezeichnet die Wort-Adresse. Hinweis Für den Zugriff auf Zeitgeber/Zähler-Ablaufmerker und -Istwerte werden dieselben Adressen verwendet. Für einen Task-Merker gibt es auch nur eine Adresse.	0001 02 Bitadresse (02) Wortadresse: 0001	0001 02 — —
Spezifizieren von Wort-Adressen	Die Wortadresse wird zur Spezifikation des 16-Bit-Worts direkt spezifiziert. Bezeichnet die Wort-Adresse.	0003 Wortadresse: 0003 D00200 Wortadresse: 00200	MOV 0003 D00200

Operand	Beschreibung	Schreibweise	Anwendungsbeispiele
Indirektes Spezifizieren DM/EM-Adressen im Binärmodus	Der Offset vom Bereichsanfang wird spezifiziert. Der Adressinhalt wird als Binärdaten (00000 bis 32767) behandelt, um die Wort-Adresse im Datenspeicher (DM) oder dem erweiterten Datenspeicher (EM) zu spezifizieren. Stellen Sie das @-Symbol voran, um eine indirekte Adresse im Binärmodus zu spezifizieren. @D□□□□□ ↓ Inhalt □ 00000 bis 32767 (0000 Hex bis 7FFF Hex in BIN) ↓ D □		
	1) D00000 bis D32767 werden spezifiziert, wenn @D(□□□□□) den Wert 0000 Hex bis 7FFF Hex (00000 bis 32767) enthält.	@D00300 0 1 0 0 Inhalt Binär: 256 ↓ Spezifiziert D00256 — Symbol @ hinzufügen	MOV #0001 @D00300
	2) E0_00000 bis E0_32767 von Bank 0 im erweiterten Datenspeicher werden spezifiziert, wenn @D(□□□□□) den Wert 8000 Hex bis FFFF Hex (32768 bis 65535) enthält.	@D00300 8 0 0 1 Inhalt Binär: 32769 ↓ Spezifiziert E0 00001	
	3) E□_00000 bis E□_32767 in der spezifizierten Bank werden spezifiziert, wenn @E□_□□□□□ den Wert 0000 Hex bis 7FFF Hex (00000 bis 32767) enthält.	@E1_00200 0 1 0 1 Inhalt Binär: 257 ↓ Spezifiziert E1_00257	MOV #0001 @E1_00200
	4) E(□+1)_00000 bis E(□+1)_32767 in der Bank, die auf die spezifizierte Bank □ folgt, werden spezifiziert, wenn @E□_□□□□□ den Wert 8000 Hex bis FFFF Hex (32768 bis 65535) enthält.	@E1_00200 8 0 0 2 Inhalt Binär: 32770 ↓ Spezifiziert E2_00002	
	Hinweis	Beim Spezifizieren einer indirekten Adresse im Binärmodus sollte der Datenspeicher (DM) und der erweiterte Datenspeicher (EM, Bänke 0 bis C) als eine Folge von Adressen behandelt werden. Wenn der Inhalt einer Adresse mit dem @-Symbol den Wert 32767 übersteigt, wird davon ausgegangen, dass sich diese Adresse im nächsten erweiterten Datenspeicher befindet (fortlaufend von 00000 in Bank Nr. 0).	
Beispiel:	Wenn das Datenspeicherwort den Wert 32768 enthält, wird E0_00000 in Bank 0 im erweiterten Datenspeicher (EM) spezifiziert.		
Hinweis	Wenn die Banknummer des erweiterten Datenspeichers (EM) als „n“ spezifiziert wird und der Inhalt des Worts den Wert 32767 übersteigt, wird davon ausgegangen, dass sich die Adresse im erweiterten Datenspeicher (EM) befindet (fortlaufend von 00000 in Bank n+1).		
Beispiel:	Wenn Bank 2 im erweiterten Datenspeicher den Wert 32768 enthält, wird E3_00000 in Bank Nr. 3 im erweiterten Datenspeicher spezifiziert.		

Operand	Beschreibung	Schreibweise	Anwendungsbeispiele
Spezifizieren indirekter DM/EM-Adressen im BCD-Modus	Der Offset vom Bereichsanfang wird spezifiziert. Der Adressinhalt wird als BCD-Daten (0000 bis 9999) behandelt, um die Wort-Adresse im Datenspeicher (DM) oder dem erweiterten Datenspeicher (EM) zu spezifizieren. Fügen Sie vorne ein Sternchen(*) hinzu, um eine indirekte Adresse im BCD-Modus zu spezifizieren. *D□□□□□ ↓ Inhalt □□□□□ 0000 bis 9999 (BCD) ↓ D □□□□□	*D00200 0 1 0 0 Inhalt ↓ Spezifiziert D0100 Sternchen (*) hinzufügen.	MOV #0001 *D00200

Operand	Beschreibung		Schreibweise	Anwendungsbeispiele
Direktes Spezifizieren eines Registers	Ein Indexregister (IR) oder ein Datenregister (DR) wird direkt durch die Angabe von IR□ (□: 0 bis 15) bzw. DR□ (□: 0 bis 15) spezifiziert.		IR0 IR1	MOVR 000102 IR0 Speichert die SPS-Speicheradresse für CIO 000102 in IR0. MOVR 0010 IR1 Speichert die SPS-Speicheradresse für CIO 0010 in IR1.
Spezifizieren einer indirekten Adresse mittels eines Registers	Indirekte Adresse (Kein Offset)	Das Bit oder Wort mit der SPS-Speicheradresse, die in IR□ enthalten ist, wird spezifiziert. Verwenden Sie ,IR□, um Bits und Worte für Befehlsoperanden zu spezifizieren.	,IR0 ,IR1	LD ,IR0 Lädt das Bit mit der SPS-Speicheradresse in IR0. MOV #0001 ,IR1 Speichert #0001 im Wort mit der SPS-Speicheradresse in IR1.
	Konstanter Offset	Das Bit oder Wort mit der SPS-Speicheradresse in IR□ + oder – der angegebenen Konstante wird spezifiziert. Verwenden Sie +/- <i>Konstante</i> ,IR□. Für konstante Offsets gilt der Bereich von – 2048 bis +2047 (dezimal). Der Offset wird in Binärdaten umgewandelt, wenn der Befehl ausgeführt wird.	+5,IR0 +31,IR1	LD +5 ,IR0 Lädt das Bit mit der SPS-Speicheradresse in IR0 + 5. MOV #0001 +31 ,IR1 Speichert #0001 im Wort mit der SPS-Speicheradresse in IR1 + 31
	Datenregister-Offset	Das Bit oder Wort mit der SPS-Speicheradresse in IR□ + der Inhalt von DR□ wird spezifiziert. Verwenden Sie DR□ ,IR□. DR-Inhalte (Datenregister) werden als Binärdaten mit Vorzeichen behandelt. Der Inhalt von IR□ ergibt einen negativen Offset, wenn der Binärwert mit Vorzeichen negativ ist.	DR0 ,IR0 DR0 ,IR1	LD DR0 ,IR0 Lädt das Bit mit der SPS-Speicheradresse in IR0 + den Wert in DR0. MOV #0001 DR0, IR1 Speichert #0001 im Wort mit SPS-Speicheradresse in IR1 + den Wert in DR0.
	Auto-Inkrement	Der Inhalt von IR□ wird um +1 oder +2 erhöht, nachdem der Wert als eine SPS-Speicheradresse referenziert wurde. +1: Verwenden Sie ,IR□+ +2: Verwenden Sie ,IR□++	,IR0 ++ ,IR1 +	LD ,IR0 ++ Inkrementiert den Inhalt von IR0 um 2, nachdem das Bit mit der SPS-Speicheradresse in IR0 geladen wurde. MOV #0001 ,IR1 + Inkrementiert den Inhalt von IR1 um 1, nachdem #0001 im Wort mit der SPS-Speicheradresse in IR1 gespeichert wurde.
	Auto-Dekrement	Der Inhalt von IR□ wird um –1 oder –2 dekrementiert, nachdem der Wert als eine SPS-Speicheradresse referenziert wurde. –1: Verwenden Sie ,–IR□ –2: Verwenden Sie ,– –IR□	,– –IR0 ,–IR1	LD ,– –IR0 Nachdem der Inhalt von IR0 um 2 dekrementiert wurde, wird das Bit mit der SPS-Speicheradresse in IR0 geladen. MOV #0001 ,–IR1 Nachdem der Inhalt von IR1 um 1 dekrementiert wurde, wird #0001 im Wort mit der SPS-Speicheradresse in IR1 gespeichert.

Daten	Operand	Art der Daten	Symbol	Bereich	Anwendungsbeispiel																																									
16-Bit-Konstante	Alle Binärdaten oder ein begrenzter Binärdatenbereich	Binär ohne Vorzeichen	#	#0000 bis #FFFF	---																																									
		Dezimal mit Vorzeichen	±	−32768 bis +32767	---																																									
		Dezimal ohne Vorzeichen	& (siehe Hinweis)	&0 bis &65535	---																																									
	Alle BCD-Daten oder ein begrenzter BCD-Datenbereich	BCD	#	#0000 bis #9999	---																																									
32-Bit-Konstante	Alle Binärdaten oder ein begrenzter Binärdatenbereich	Binär ohne Vorzeichen	#	#00000000 bis #FFFFFFFF	---																																									
		Dezimal mit Vorzeichen	+	−2147483648 bis +2147483647	---																																									
		Dezimal ohne Vorzeichen	& (siehe Hinweis)	&0 bis &429467295	---																																									
	Alle BCD-Daten oder ein begrenzter BCD-Datenbereich	BCD	#	#00000000 bis #99999999	---																																									
Textzeichenkette	Beschreibung		Symbol	Beispiele	---																																									
	Textzeichenkettendaten werden im ASCII-Format (1 Byte außer bei Sonderzeichen) in der Reihenfolge der Bytes von links nach rechts und der Worte von rechts (niedrigstes) nach links gespeichert. Bei einer ungeraden Zeichenanzahl wird im äußersten rechten Byte des letzten Words 00 Hex (NUL-Code) gespeichert. 0000 Hex (2 NUL-Codes) wird bei einer geraden Zeichenanzahl in den äußersten linken und rechten freien Bytes vom letzten Wort + 1 gespeichert.		---	'ABCDE' <table><tr><td>'A'</td><td>'B'</td></tr><tr><td>'C'</td><td>'D'</td></tr><tr><td>'E'</td><td>NUL</td></tr></table> II <table><tr><td>41</td><td>42</td></tr><tr><td>43</td><td>44</td></tr><tr><td>45</td><td>00</td></tr></table> 'ABCD' <table><tr><td>'A'</td><td>'B'</td></tr><tr><td>'C'</td><td>'D'</td></tr><tr><td>NUL</td><td>NUL</td></tr></table> II <table><tr><td>41</td><td>42</td></tr><tr><td>43</td><td>44</td></tr><tr><td>00</td><td>00</td></tr></table>	'A'	'B'	'C'	'D'	'E'	NUL	41	42	43	44	45	00	'A'	'B'	'C'	'D'	NUL	NUL	41	42	43	44	00	00	MOV\$ D00100 D00200 <table><tr><td>D00100</td><td>41</td><td>42</td></tr><tr><td>D00101</td><td>43</td><td>44</td></tr><tr><td>D00102</td><td>45</td><td>00</td></tr></table> ↓ <table><tr><td>D00200</td><td>41</td><td>42</td></tr><tr><td>D00201</td><td>43</td><td>44</td></tr><tr><td>D00202</td><td>45</td><td>00</td></tr></table>	D00100	41	42	D00101	43	44	D00102	45	00	D00200	41	42	D00201	43	44	D00202	45
'A'	'B'																																													
'C'	'D'																																													
'E'	NUL																																													
41	42																																													
43	44																																													
45	00																																													
'A'	'B'																																													
'C'	'D'																																													
NUL	NUL																																													
41	42																																													
43	44																																													
00	00																																													
D00100	41	42																																												
D00101	43	44																																												
D00102	45	00																																												
D00200	41	42																																												
D00201	43	44																																												
D00202	45	00																																												
Zu den ASCII-Zeichen, die in einer Textzeichenfolge verwendet werden können, gehören alphanumerische Zeichen, Katakana und Symbole (außer Sonderzeichen). Die Zeichen entnehmen Sie der folgenden Tabelle.																																														

Hinweis Die Schreibweise für Dezimal ohne Vorzeichen wird nur bei CX-Programmer verwendet.

ASCII-Zeichen

Bits 0 bis 3		Bits 4 bis 7															
Binär		0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
	Hex	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0000	0			Leer- zei- chen	0	@	P	`	p				—	ウ	ミ		
0001	1			!	1	A	Q	a	q			▯	ア	チ	ム		
0010	2			"	2	B	R	b	r			「	イ	ツ	ヌ		
0011	3			#	3	C	S	c	s			」	ウ	テ	モ		
0100	4			\$	4	D	T	d	t			「	エ	ト	フ		
0101	5			%	5	E	U	e	u			・	オ	ナ	ル		
0110	6			&	6	F	V	f	v			ヲ	カ	ニ	ヨ		
0111	7			'	7	G	W	g	w			フ	キ	ヌ	ラ		
1000	8			<	8	H	X	h	x			イ	ク	ネ	リ		
1001	9			>	9	I	Y	i	y			ロ	ケ	ル	ル		
1010	A			*	▯	J	Z	j	z			エ	コ	ハ	レ		
1011	B			+	;	K	[	k	[			オ	サ	ヒ	ロ		
1100	C			,	<	L	¥	l	l			ヤ	シ	フ	ワ		
1101	D			—	=	M	]	m	>			ユ	ス	ヘ	ン		
1110	E			.	>	N	^	n	~			ヨ	セ	ホ	ハ		
1111	F			/	?	O	_	o				ッ	リ	マ	▯		

2-1-6 Datenformate

In der folgenden Tabelle sind die von der CS/CJ-Serie unterstützten Datenformate aufgeführt.

Datentyp	Datenformat	Dezimal	4-stellig hexadezimal
Binär ohne Vorzeichen	15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0 Binär 2^{15} 2^{14} 2^{13} 2^{12} 2^{11} 2^{10} 2^9 2^8 2^7 2^6 2^5 2^4 2^3 2^2 2^1 2^0 Dezimal 32768 16384 8192 4092 2048 1024 512 256 128 64 12 16 8 4 2 1 Hex 2^3 2^2 2^1 2^0 2^3 2^2 2^1 2^0 2^3 2^2 2^1 2^0 2^3 2^2 2^1 2^0	0 bis 65535	0000 bis FFFF
Binär mit Vorzeichen	15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0 Binär 2^{15} 2^{14} 2^{13} 2^{12} 2^{11} 2^{10} 2^9 2^8 2^7 2^6 2^5 2^4 2^3 2^2 2^1 2^0 Dezimal 32768 16384 8192 4092 2048 1024 512 256 128 64 12 16 8 4 2 1 Hex 2^3 2^2 2^1 2^0 2^3 2^2 2^1 2^0 2^3 2^2 2^1 2^0 2^3 2^2 2^1 2^0 ↑ Vorzeichen-Bit: 0: Positiv, 1: Negativ	–32768 bis +32767	8000 bis 7FFF
BCD (Binary Coded Decimal)	15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0 Binärformat 2^3 2^2 2^1 2^0 2^3 2^2 2^1 2^0 2^3 2^2 2^1 2^0 2^3 2^2 2^1 2^0 Dezimal 0 bis 9 0 bis 9 0 bis 9 0 bis 9	0 bis 9999	0000 bis 9999

Datentyp	Datenformat	Dezimal	4-stellig hexadezimal
Gleitkomma- zimalwerte mit einfacher Präzision	Wert = $(-1)^{\text{Vorzeichen}} \times 1.[\text{Mantisse}] \times 2^{\text{Exponent}}$ Vorzeichen (Bit 31) 1: negativ oder 0: positiv Mantisse Die 23 Bits von Bit 00 bis Bit 22 enthalten die Mantisse, d. h., den Anteil hinter dem Dezimalkomma in 1,□□□..... im Binärformat. Exponent Die 8 Bits von Bit 23 bis Bit 30 enthalten den Exponenten. Der Exponent wird im Binärformat als 127 plus n in 2^n ausgedrückt. Hinweis Dieses Format entspricht den IEEE754-Standards für einfach genaue Fließkommaten und wird nur bei Befehlen verwendet, mit denen Fließkommaten konvertiert oder berechnet werden. Es kann zur Einstellung oder Überwachung im SPS-Speicher- und Datenansichts-Fenster im CX-Programmer verwendet werden (wird von den Programmierkonsolen nicht unterstützt). In dieser Eigenschaft müssen Anwender dieses Format nicht kennen, doch sie müssen wissen, dass die Formatierung zwei Worte belegt.	---	---
Gleitkomma- zimalwerte mit doppelter Präzision	Wert = $(-1)^{\text{Vorzeichen}} \times 1.[\text{Mantisse}] \times 2^{\text{Exponent}}$ Vorzeichen (Bit 63) 1: negativ oder 0: positiv Mantisse Die 52 Bits von Bit 00 bis Bit 51 enthalten die Mantisse, d.h., den Anteil hinter dem Dezimalkomma in 1,□□□..... im Binärformat. Exponent Die 11 Bits von Bit 52 bis 62 enthalten den Exponenten. Der Exponent wird im Binärformat als 1023 plus n in 2^n ausgedrückt. Hinweis Dieses Format entspricht den IEEE754-Standards für doppelt genaue Fließkommaten und wird nur bei Befehlen verwendet, mit denen Fließkommaten konvertiert oder berechnet werden. Es kann zur Einstellung oder Überwachung im SPS-Speicher- und Datenansichts-Fenster im CX-Programmer verwendet werden (wird von den Programmierkonsolen nicht unterstützt). In dieser Eigenschaft müssen Anwender dieses Format nicht kennen, doch sie müssen wissen, dass die Formatierung vier Worte belegt.	---	---

Binärdaten mit Vorzeichen

Bei Binärdaten mit Vorzeichen gibt das äußerste linke Bit das Vorzeichen von 16-Bit-Binärdaten an. Der Wert wird als 4-stelliger Hexadezimalwert ausgedrückt.

Positive Zahlen: Ein Wert ist positiv oder beträgt 0, wenn das äußerste linke Bit 0 (AUS) ist. Dies wird in 4-stelligen Hexadezimalwerten als 0000 bis 7FFF Hex ausgedrückt.

Negative Zahlen: Ein Wert ist negativ, wenn das äußerste linke Bit 1 (EIN) ist. Dies wird in 4-stelligen Hexadezimalwerten als 8000 bis FFFF Hex ausgedrückt. Der Betrag des negativen Wertes (dezimal) wird als Zweierkomplement ausgedrückt.

Beispiel: Wenn -19 in Dezimal als Binär mit Vorzeichen behandelt werden soll, wird 0013 Hex (der Betrag 19) von FFFF Hex subtrahiert und anschließend wird 0001 Hex addiert, so dass sich FFED Hex ergibt.

		F	F	F	F
		1111	1111	1111	1111
	Absolutwert	0	0	1	3
-)		0000	0000	0001	0011
		F	F	E	C
		1111	1111	1110	1100
		0	0	0	1
		0000	0000	0000	0001
+)					
	Zweierkomplement	F	F	E	D
		1111	1111	1110	1101

Komplemente

Im Allgemeinen bezieht sich das Komplement zur Basis x auf eine Zahl, die durch Subtrahieren aller Stellen einer gegebenen Zahl von $x - 1$ und anschließend Addieren von 1 zur äußersten rechten Stelle berechnet wird. (Beispiel: Das Zehnerkomplement von 7556 ist $9999 - 7556 + 1 = 2444$.) Ein Komplement wird verwendet, um eine Subtraktion und andere Funktionen als Addition auszudrücken.

Beispiel: Mit $8954 - 7556 = 1398$, $8954 +$ (das Zehnerkomplement von 7556) $= 8954 + 2444 = 11398$. Wenn das äußerste linke Bit ignoriert wird, wird bei Subtraktion ein Ergebnis von 1398 erzielt.

Zweierkomplemente

Ein Zweierkomplement ist ein Komplement mit Basis 2. Dabei werden alle Stellen von 1 ($2 - 1 = 1$) subtrahiert und 1 addiert.

Beispiel: Das Zweierkomplement der Binärzahl 1101 ist 1111 (F Hex) $- 1101$ (D Hex) $+ 1$ (1 Hex) $= 0011$ (3 Hex). Im Folgenden wird dieser Wert als 4-stelliger Hexadezimalwert ausgedrückt.

Das Zweierkomplement b Hex von a Hex ist $FFFF$ Hex $- a$ Hex $+ 0001$ Hex $= b$ Hex. Um das Zweierkomplement b Hex von „ a Hex“ zu bestimmen, wird b Hex $= 10000$ Hex $- a$ Hex verwendet.

Beispiel: Um das Zweierkomplement von 3039 Hex zu bestimmen, wird 10000 Hex $- 3039$ Hex $= CFC7$ Hex verwendet.

Gleichermaßen wird a Hex $= 10000$ Hex $- b$ Hex zur Bestimmung des Wertes a Hex vom Zweierkomplement b Hex verwendet.

Beispiel: Um den tatsächlichen Wert des Zweierkomplement CFC7 Hex zu bestimmen, wird 10000 Hex $- CFC7$ Hex $= 3039$ Hex verwendet.

Die CS/CJ-Serie verfügt über die folgenden zwei Befehle: NEG(160)(2'S COMPLEMENT) und NEGL(161) (DOUBLE 2'S COMPLEMENT), die verwendet werden können, um das Zweierkomplement aus der tatsächlichen Zahl oder die tatsächliche Zahl aus dem Zweierkomplement zu bestimmen.

BCD-Daten mit Vorzeichen

Bei BCD-Daten mit Vorzeichen handelt es sich um ein spezielles Datenformat, das verwendet wird, um negative Zahlen in BCD auszudrücken. Auch wenn dieses Format in einigen Anwendungen zu finden ist, ist es dennoch nicht genau definiert und von der spezifischen Anwendungen abhängig.

Die CS/CJ-Serie unterstützt die folgenden Befehle zur Konvertierung von Datenformaten: SIGNED BCD-TO-BINARY BINS(470), DOUBLE SIGNED BCD-TO-BINARY: BISL(472), SIGNED BINARY-TO-BCD: BCDS(471) und DOUBLE SIGNED BINARY-TO-BCD: BDSL(473). Weitere Informationen finden Sie im *CS/CJ-Serie Speicherprogrammierbare Steuerungen Befehlsreferenzhandbuch (W340)*.

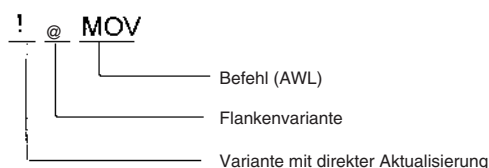
Dezimal	Hexadezimal	Binär	BCD	
0	0	0000		0000
1	1	0001		0001
2	2	0010		0010
3	3	0011		0011
4	4	0100		0100
5	5	0101		0101
6	6	0110		0110
7	7	0111		0111
8	8	1000		1000
9	9	1001		1001
10	A	1010	0001	0000
11	B	1011	0001	0001
12	C	1100	0001	0010
13	D	1101	0001	0011
14	E	1110	0001	0100
15	F	1111	0001	0101
16	10	10000	0001	0110

Dezimal	Binär ohne Vorzeichen (4-stellig hexadezimal)	Binär mit Vorzeichen (4-stellig hexadezimal)
+65.535	FFFF	Kann nicht ausgedrückt werden
+65534	FFFE	
.	.	
.	.	
.	.	
+32.769	8001	7FFF
+32.768	8000	
+32.767	7FFF	
+32.766	7FFE	
.	.	
.	.	0002
.	.	
+2	0002	
+1	0001	
0	0000	
-1	Kann nicht ausgedrückt werden	FFFF
-2		FFFE
.		8001
.		
.		
-32.767		
-32.768		

2-1-7 Befehlsvarianten

Die folgenden Befehlsvarianten stehen für Ausführungsbedingungen mit Flankenauswertung und zur Aktualisierung der Daten beim Ausführen des Befehls zur Verfügung (unmittelbare Aktualisierung).

Variante		Symbol	Beschreibung
Flanken-überwachung	EIN	@	Befehl, der ausgeführt wird, wenn die Ausführungsbedingung auf EIN wechselt.
	AUS	%	Befehl, der ausgeführt wird, wenn die Ausführungsbedingung auf AUS wechselt.
Direkte Aktualisierung		!	Aktualisiert Daten im E/A-Bereich, die durch Operanden spezifiziert werden, oder Worte einer Spezial-E/A-Baugruppe beim Ausführen des Befehls. (Direkte Aktualisierung wird von CS1D CPU-Baugruppen für Duplex-CPU-Systeme nicht unterstützt.)



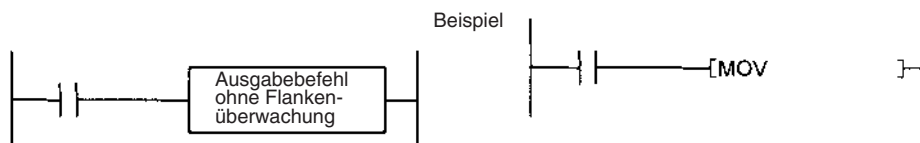
2-1-8 Ausführungsbedingungen

Die CS/CJ-Serie unterstützt die folgenden Arten von Grund- und Spezialbefehlen.

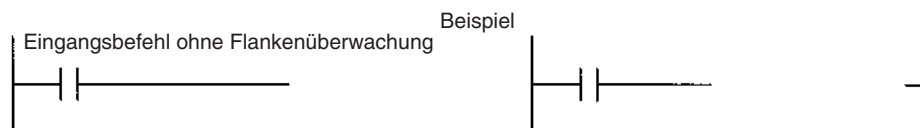
- Befehle ohne Flankenüberwachung, die in jedem Zyklus ausgeführt werden
- Befehle mit Flankenüberwachung, die nur einmalig ausgeführt werden

Befehle ohne Flankenüberwachung

Ausgabebefehle, die Ausführungsbedingungen benötigen, werden einmal in jedem Zyklus ausgeführt, solange die Ausführungsbedingung gültig ist (EIN oder AUS).



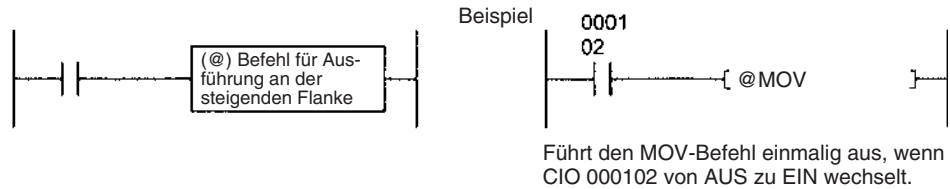
Eingangsbefehle, die logische Starts bewirken, sowie Zwischenbefehle lesen den Bitstatus, führen Vergleiche durch, testen Bits oder führen andere Verarbeitungen in jedem Zyklus aus. Wenn die Ergebnisse EIN lauten, wird die Ausführungsbedingung auf EIN gesetzt.



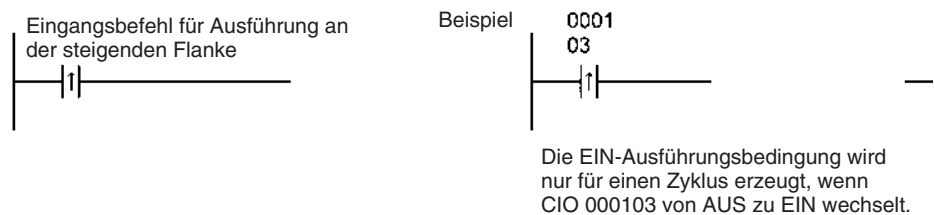
Befehle mit Eingangs-Flankenüberwachung

Befehle, die an der steigenden Flanke ausgeführt werden (Befehl mit vorangestelltem @)

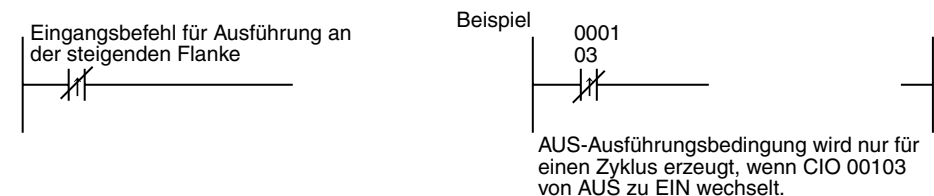
- **Ausgabebefehle:** Der Befehl wird nur in dem Zyklus ausgeführt, in dem die Ausführungsbedingung auf EIN gewechselt ist (AUS → EIN). In den darauffolgenden Zyklen wird der Befehl nicht ausgeführt.



- **Eingangsbefehle (logische Starts und Zwischenbefehle):** Der Befehl liest den Bitstatus, führt Vergleiche durch, testet Bits oder führt andere Verarbeitungen in jedem Zyklus durch. Auf dieser Basis gibt er eine EIN-Ausführungsbedingung aus, wenn die Ergebnisse von AUS zu EIN wechseln. Die Ausführungsbedingung wird im nächsten Zyklus auf AUS gesetzt.

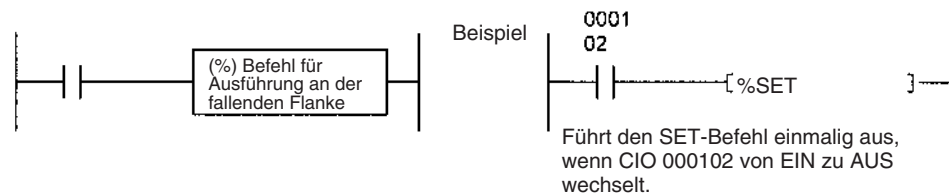


- **Eingangsbefehle (logische Starts und Zwischenbefehle):** Der Befehl liest den Bitstatus, führt Vergleiche durch, testet Bits oder führt andere Verarbeitungen in jedem Zyklus durch. Auf dieser Basis gibt er eine AUS-Ausführungsbedingung aus, wenn die Ergebnisse von AUS zu EIN wechseln. Die Ausführungsbedingung wird im nächsten Zyklus auf EIN gesetzt.

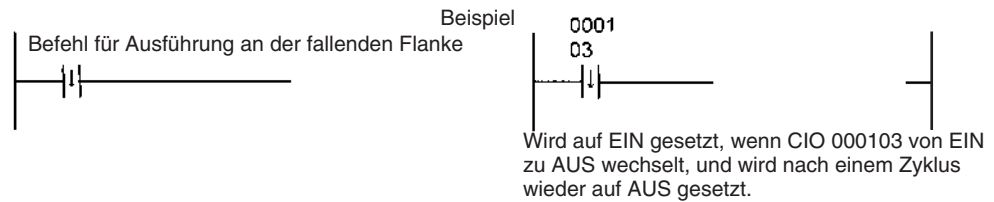


Befehle, die an der fallenden Flanke ausgeführt werden (Befehl mit vorangestelltem %)

- **Ausgabebefehle:** Der Befehl wird nur in dem Zyklus ausgeführt, in dem die Ausführungsbedingung auf AUS gewechselt ist (EIN → AUS). In den darauffolgenden Zyklen wird der Befehl nicht ausgeführt.

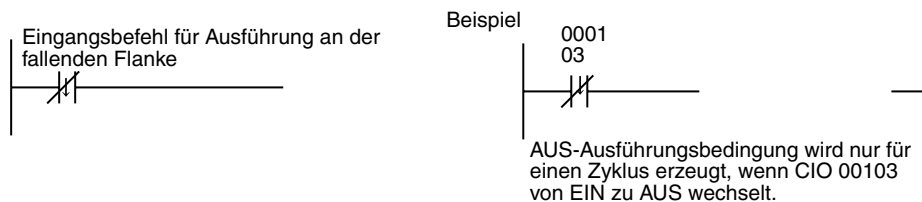


- **Eingangsbefehle (logische Starts und Zwischenbefehle):** Der Befehl liest den Bitstatus, führt Vergleiche durch, testet Bits oder führt andere Verarbeitungen in jedem Zyklus durch. Auf dieser Basis gibt er die Ausführungsbedingung aus, wenn die Ergebnisse von EIN zu AUS wechseln. Die Ausführungsbedingung wird beim nächsten Zyklus auf AUS gesetzt.



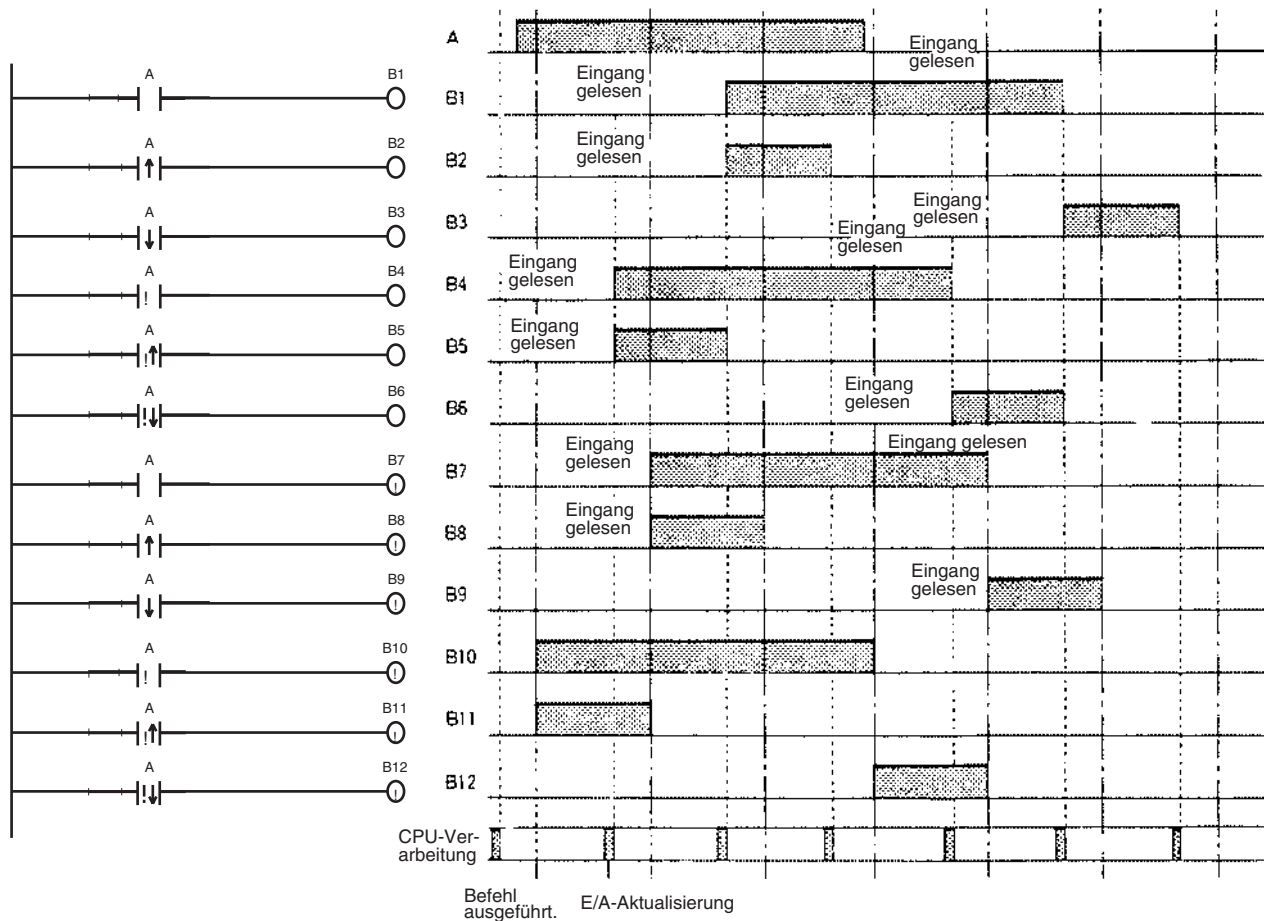
Hinweis Im Gegensatz zu den Befehlen, die an der steigenden Flanke ausgeführt werden, steht die Variante für die Ausführung an der fallenden Flanke (%) nur bei den Befehlen LD, AND, OR, SET und RSET zur Verfügung. Um eine Ausführung an der fallenden Flanke bei anderen Befehlen zu realisieren, müssen die Befehle mit den Befehlen DIFD oder DOWN kombiniert werden. Der NOT-Befehl kann nur zu Befehlen hinzugefügt werden, wenn eine CS1-H, CJ1-H, CJ1M oder CS1D CPU-Baugruppe verwendet wird.

- **Eingangsbefehle (logische Starts und Zwischenbefehle):** Der Befehl liest den Bitstatus, führt Vergleiche durch, testet Bits oder führt andere Verarbeitungen in jedem Zyklus durch. Auf dieser Basis gibt er eine AUS-Ausführungsbedingung (logischer Stromfluss wird unterbrochen) aus, wenn die Ergebnisse von EIN zu AUS wechseln. Die Ausführungsbedingung wird im nächsten Zyklus auf EIN gesetzt.



2-1-9 Zeitliches Verhalten von E/A-Befehlen

Im Folgenden finden Sie eine Darstellung der unterschiedlichen Zeitverhalten einzelner Befehle eines Programms, das nur die Befehle LD und OUT enthält.

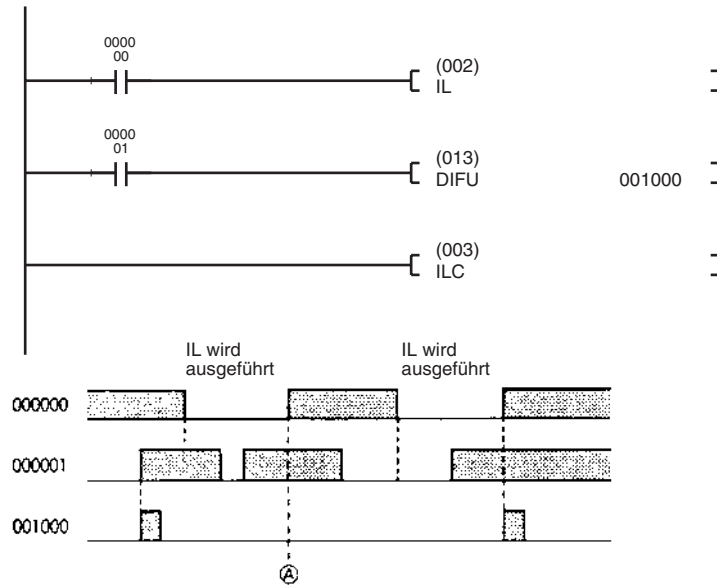


Befehle mit Flankendifferenzierung

- Ein Befehl mit Flankendifferenzierung verfügt über einen internen Merker, der angibt, ob der vorherige Wert EIN oder AUS ist. Zu Beginn des Betriebs werden die Merker für vorherigen Werte bei Befehlen zur Ausführung an der steigenden Flanke (DIFU- und @-Befehle) auf EIN gesetzt, und die Merker für vorherigen Werte bei Befehlen zur Ausführung an der fallenden Flanke (DIFD- und %-Befehle) werden auf AUS gesetzt. Dadurch wird verhindert, dass die Flankenüberwachungsergebnisse versehentlich zu Beginn des Betriebs ausgegeben werden.
- Ein Befehl zur Ausführung an der steigenden Flanke (DIFU- oder @-Befehl) gibt nur EIN aus, wenn die Ausführungsbedingung auf EIN und der Merker für den vorherigen Wert auf AUS gesetzt ist.

- **Verwendung in Sperren (IL - ILC-Befehle)**

Im folgenden Beispiel behält der Merker für den vorherigen Wert des Befehls mit Flankendifferenzierung den zuvor gesperrten Wert bei und gibt an Punkt A keinen Differenzierungsimpuls aus, da der Wert bei aktivierter Sperre nicht aktualisiert wird.



- **Verwendung in Sprungbefehlen (JMP - JME-Befehle):** Genau wie bei Sperren (Interlocks) wird der Merker für den vorherigen Wert eines Flankenüberwachungsbefehls nicht geändert, wenn dieser Befehl übersprungen, d.h. nicht ausgeführt wird, d. h., der vorherige Wert wird beibehalten. Befehle für zur Ausführung an der steigenden und fallenden Flanke geben die Ausführungsbedingung nur aus, wenn sich der Eingangsstatus vom vorherigen unterscheidet.

Hinweis a) Verwenden Sie nicht den Immer-EIN-Merker oder A20011 (Erster-Zyklus-Merker) als Eingangsbit für einen Befehl zur Ausführung an der steigenden Flanke. Dieser Befehl würde niemals ausgeführt.

- b)** Verwenden Sie nicht den Immer-AUS-Merker als Eingangsbit für einen Befehl zur Ausführung an der fallenden Flanke. Dieser Befehl würde niemals ausgeführt.

2-1-10 Aktualisierungszeitpunkt

Für die Aktualisierung externer E/A werden die folgenden Methoden verwendet.

- Zyklische Aktualisierung
- Direkte Aktualisierung (!-spezifizierte Befehle, IORF-Befehl)

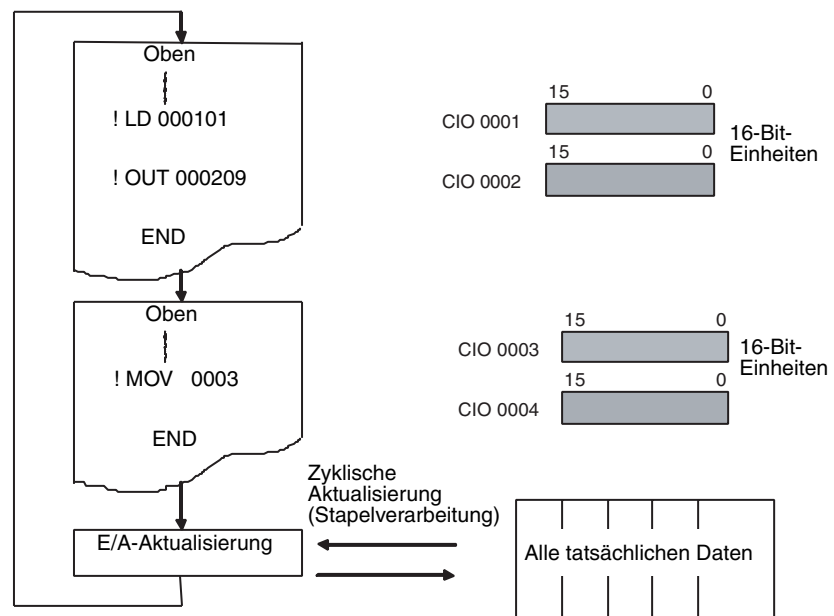
Informationen zur E/A-Aktualisierung finden Sie auch im Abschnitt über den Betrieb der CPU-Baugruppe im *CS/CJ-Serie Bedienerhandbuch*.

Zyklische Aktualisierung

Jedes Programm, das einer ausführungsbereiten zyklischen Task oder einer Task mit erfüllter Interrupt-Bedingung zugeordnet ist, wird beginnend mit der Programmstartadresse bis zum END(001)-Befehl ausgeführt. Nachdem alle ausführungsbereiten zyklischen Tasks oder Tasks mit erfüllter Interrupt-Bedingung ausgeführt wurden, werden alle Ein-/Ausgänge von der zyklischen Aktualisierung gleichzeitig aktualisiert.

Hinweis

Programme können in mehreren Tasks ausgeführt werden. Die E/A werden nach dem abschließenden END(001)-Befehl im Programm aktualisiert, dem die höchste Nummer zugewiesen ist (unter allen ausführungsbereiten zyklischen Tasks). Die E/A werden nicht nach dem END(001)-Befehl in Programmen aktualisiert, die anderen zyklischen Tasks zugeordnet sind.



Führen Sie einen IORF-Befehl für alle erforderlichen Worte vor dem END(001)-Befehl aus, wenn die zwischenzeitliche E/A-Aktualisierung in anderen Tasks benötigt wird.

Direkte Aktualisierung

Befehle mit Aktualisierungsvarianten (!)

Die E/A werden wie im Folgenden gezeigt aktualisiert, wenn ein Befehl ausgeführt wird und ein tatsächliches E/A-Bit als Operand spezifiziert ist.

Baugruppen	Aktualisierte Daten
C200H E/A-Baugruppen (nur CS-Serie)	E/A werden für die 16 Bits aktualisiert, die das Bit enthalten.
CJ E/A-Baugruppen	

- Wenn ein Wortoperand für einen Befehl spezifiziert ist, werden die E/A für die spezifizierten 16 Bits aktualisiert.
- Die Eingaben werden für die Eingangs- oder Quelloperanden aktualisiert, unmittelbar bevor ein Befehl ausgeführt wird.
- Die Ausgaben werden für die Ausgangs- oder Zieloperanden (D) aktualisiert, unmittelbar nachdem ein Befehl ausgeführt wurde.

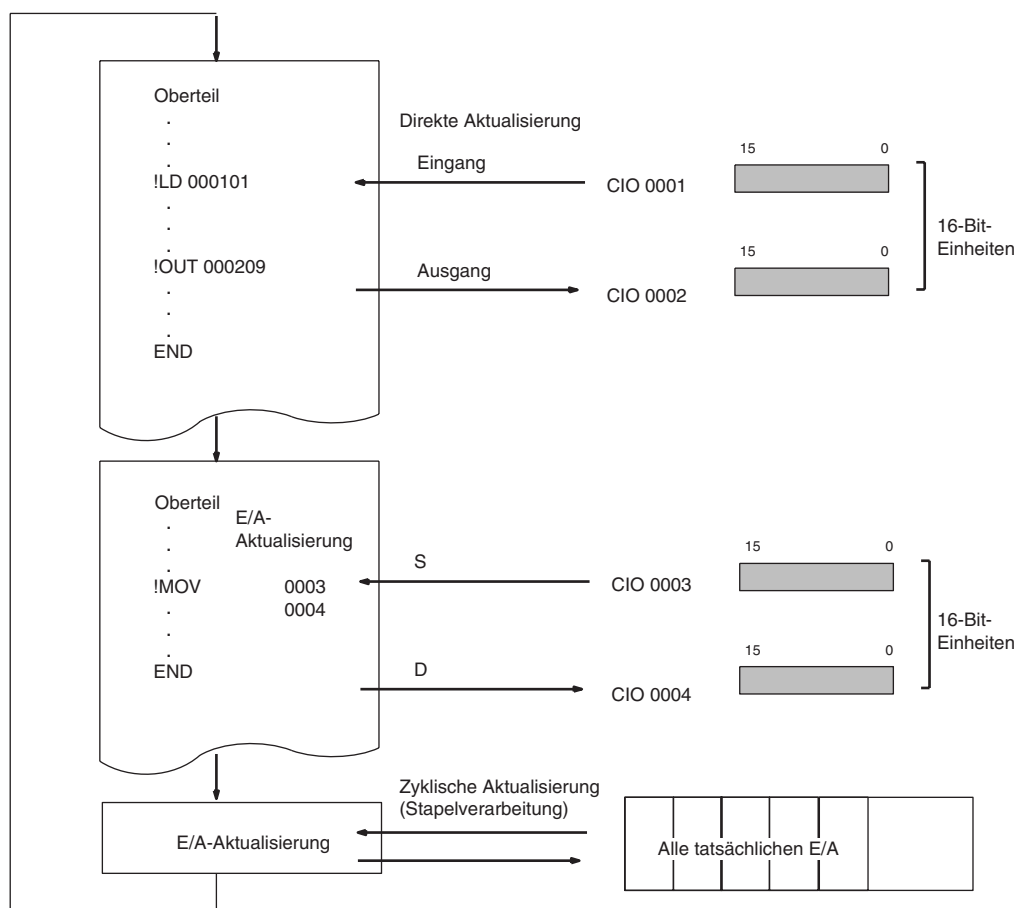
Fügen Sie eine Ausrufezeichen (!) (Direkte Aktualisierung) vor dem Befehl ein.

Hinweis Direkte Aktualisierung wird von den CS1D CPU-Baugruppen für Duplex-CPU-Systeme nicht unterstützt. Diese Systeme unterstützen jedoch die Aktualisierung durch die Befehle IORF(097) und DLNK(226).

Durch den Befehl I/O REFRESH aktualisierte Baugruppen

Position	CPU- oder E/A-Erweiterungs-Racks (jedoch keine SYSMAC BUS-Slave-Racks)		
Baugruppen	E/A-Baugruppen	CS/CJ-Serie E/A-Baugruppen	Aktualisiert
		C200H E/A-Baugruppe (siehe Hinweis)	Aktualisiert
		C200H Multipoint-E/A-Baugruppen der Gruppe 2 (siehe Hinweis)	Nicht aktualisiert
	Spezial-E/A-Baugruppen		Nicht aktualisiert

Hinweis C200H E/A-Baugruppen können nicht in SPS der CJ-Serie installiert werden.



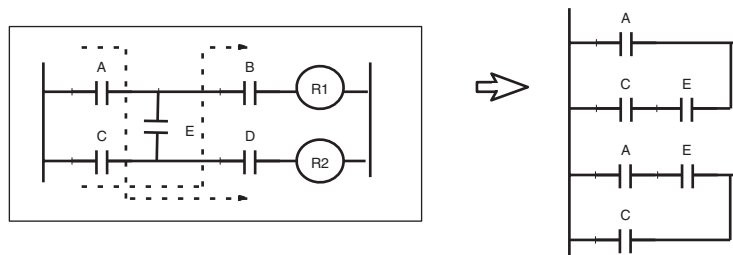
Durch IORF(097) oder DLNK(226) aktualisierte Baugruppen

Ein E/A-Auffrischungs-Befehl (IORF(097)), der die tatsächlichen E/A-Daten in einem spezifizierten Wortbereich aktualisiert, steht als Spezialbefehl zur Verfügung. Mit diesem Befehl können die gesamten oder nur ein spezifizierter Bereich der tatsächlichen E/A-Daten in einem Zyklus aktualisiert werden. Der IORF-Befehl kann auch dazu verwendet werden, Worte zu aktualisieren, die Spezial-E/A-Baugruppen zugewiesen sind.

Ein weiterer Befehl, CPU-Spezial-E/A-direkte-Auffrischung (DLNK(226)), steht zur Aktualisierung der Worte zur Verfügung, die den CPU-Bus-Baugruppen in den CIO- und DM-Bereichen zugeordnet sind. Mit diesem Befehl können außerdem spezielle Aktualisierungen bei den Baugruppen durchgeführt werden, wie z. B. für Data-Links. Der Befehl DLNK(226) wird nur von den CS1-H, CJ1-H, CJ1M und CS1D CPU-Baugruppen unterstützt.

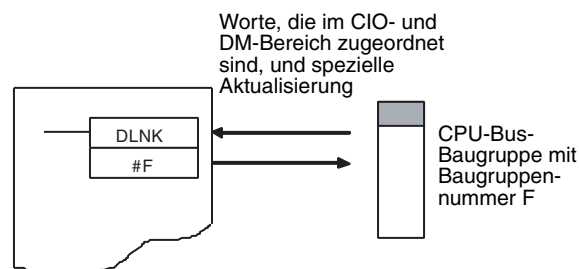
Durch IORF(097) aktualisierte Baugruppen

Position	CPU- oder E/A-Erweiterungs-Racks (jedoch keine SYSMAC BUS-Slave-Racks)		
Bau- gruppen	E/A-Baugruppen	CS/CJ-Serie E/A-Baugruppen	Aktualisiert
		C200H E/A-Baugruppen	Aktualisiert
		C200H Multipoint-E/A-Baugruppen der Gruppe 2	Aktualisiert
	Spezial-E/A-Baugruppen		Aktualisiert
	CPU-Bus-Baugruppen		Nicht aktualisiert



Durch DLNK(226) aktualisierte Baugruppen

Position	CPU- oder E/A-Erweiterungs-Racks (jedoch keine SYSMAC BUS-Slave-Racks)	
Bau- gruppen	E/A-Baugruppen	Nicht aktualisiert
	Spezial-E/A-Baugruppen	Nicht aktualisiert
	CPU-Bus-Baugruppen	Aktualisiert
	Worte, die der Baugruppe im CIO-Bereich zugeordnet sind Worte, die der Baugruppe im DM-Bereich zugeordnet sind Spezielle Aktualisierung bei der Baugruppe (Data-Links bei Controller-Link-Baugruppen und SYSMAC-Link-Baugruppen oder dezentrale E/A bei DeviceNet-Baugruppen)	



2-1-11 Programmkapazität

In der folgenden Tabelle sind die maximalen Programmkapazitäten von CPU-Baugruppen der CS/CJ-Serie für alle Anwenderprogramme (d. h., die Gesamtkapazität aller Tasks) aufgeführt. Alle Kapazitäten sind als maximale Anzahl von Steps (Programmschritten) angegeben. Die Kapazitäten dürfen nicht überschritten werden, und das Übertragen eines Programms in die SPS wird deaktiviert, wenn ein Versuch unternommen wird, die Kapazität zu überschreiten.

Ein einzelner Befehl ist 1 bis 7 Steps lang. Weitere Informationen zur jeweiligen Anzahl von Steps in den Befehlen finden Sie unter *10-5, Ausführungszeit und Step-Anzahl der Befehle* im *Bedienerhandbuch*. (Die Länge der einzelnen Befehle wird um 1 Step erhöht, wenn ein Operand mit doppelter Länge verwendet wird.)

Serie	CPU-Baugruppe	Max. Programmkapazität	Ein-/Ausgänge
CS-Serie	CS1H-CPU67H/CPU67-E	250 kSteps	5.120
	CS1D-CPU67H	250 kSteps	
	CS1D-CPU67S	250 kSteps	
	CS1H-CPU66H/CPU66-E	120 kSteps	
	CS1H-CPU65H/CPU65-E	60 kSteps	
	CS1D-CPU65H	60 kSteps	
	CS1D-CPU65S	60 kSteps	
	CS1H-CPU64H/CPU64-E	30 kSteps	
	CS1H-CPU63H/CPU63-E	20 kSteps	
	CS1G-CPU45H/CPU45-E	60 kSteps	
	CS1G-CPU44H/CPU44-E	30 kSteps	1.280
	CS1D-CPU44S	30 kSteps	
	CS1G-CPU43H/CPU43-E	20 kSteps	960
	CS1G-CPU42H/CPU42-E	10 kSteps	
	CS1D-CPU42S	10 kSteps	
CJ Serie	CJ1H-CPU67H	250KSteps	2.560
	CJ1H-CPU66H	120KSteps	
	CJ1H-CPU65H	60KSteps	
	CJ1G-CPU45H/CPU45	60KSteps	1280
	CJ1G-CPU44H/CPU44	30KSteps	
	CJ1G-CPU43H	20KSteps	960
	CJ1G-CPU42H	10KSteps	
	CJ1M-CPU23/CPU13	20KSteps	640
	CJ1M-CPU22/CPU12	10KSteps	320

Hinweis

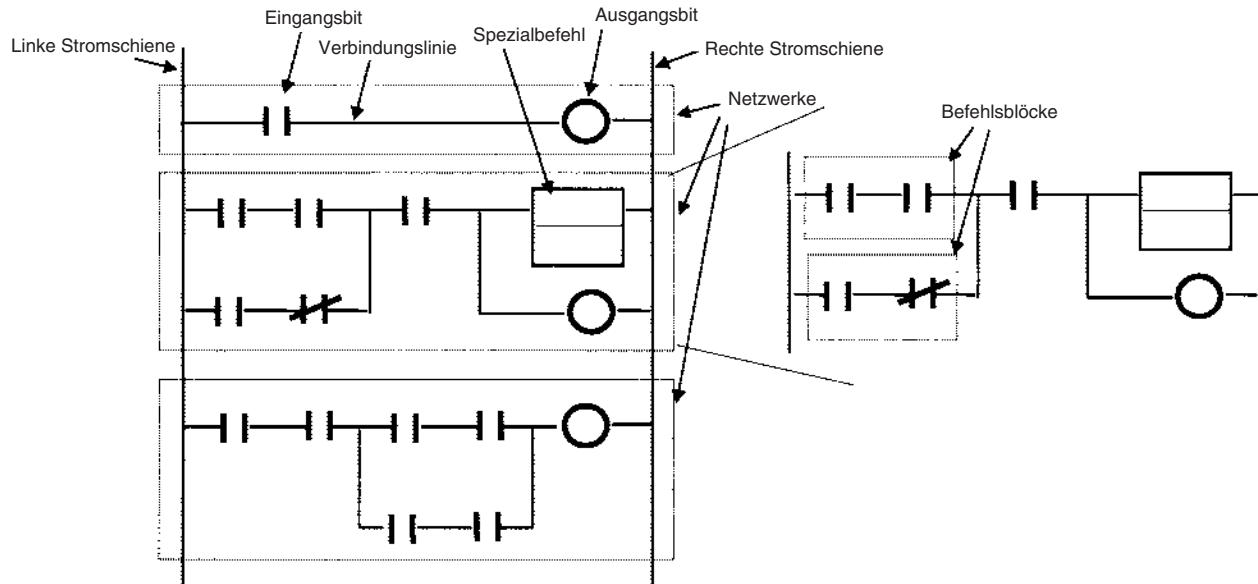
Anders als bei früheren OMRON SPS-Systemen (wie C200HX/HG/HE und CV-Serie SPS), bei denen die Programmkapazität in Worten angegeben wurde, erfolgt die Angabe der Programmkapazität bei CS/CJ-Serie SPS-Systemen in Steps. Weitere Informationen zu den Richtlinien für die Umrechnung von Programmkapazitäten früherer OMRON SPS finden Sie am Ende von Abschnitt *10-5, Ausführungszeit und Step-Anzahl der Befehle* im *Bedienerhandbuch* Ihrer SPS.

2-1-12 Grundbegriffe der SPS-Programmierung

Befehle werden in der Reihenfolge ausgeführt, in der sie im Speicher aufgelistet sind (AWL-Reihenfolge). Die Grundkonzepte der Programmierung und die Ausführungsreihenfolge müssen richtig sein.

Allgemeine Struktur des Kontaktplanprogramms

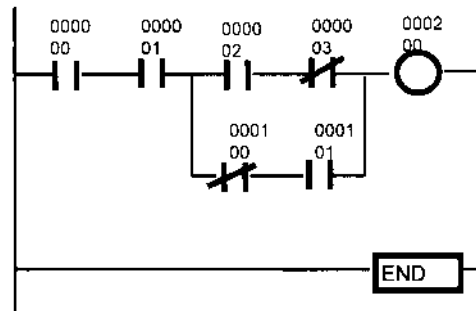
Ein Kontaktplan besteht aus den linken und rechten Stromschienen, Verbindungslinien, Eingangsbits, Ausgangsbits und Spezial-Befehlen. Ein Programm besteht aus mindestens einem Programm-Netzwerk. Ein Programm-Netzwerk ist eine Einheit, die durch horizontale Aufteilung des Kontaktplans erhalten wird. In Form von AWL-Code besteht ein Netzwerk aus allen Befehlen zwischen einem LD/LD NOT-Befehl bis zu dem Ausgabebefehl unmittelbar vor dem nächsten LD/LD NOT-Befehl. Ein Programm-Netzwerk besteht aus Befehlsblöcken, die mit einem LD/LD NOT-Befehl beginnen, der den logischen Start bildet.



AWL

Ein AWL-Programm (Anweisungsliste) besteht aus einer Folge von SPS-Befehlen, die in AWL-Form angegeben werden. Es besteht aus Programmadressen, und eine Programmadresse entspricht einem Befehl. Programmadressen bestehen aus sechs Stellen beginnend bei 000000.

Beispiel

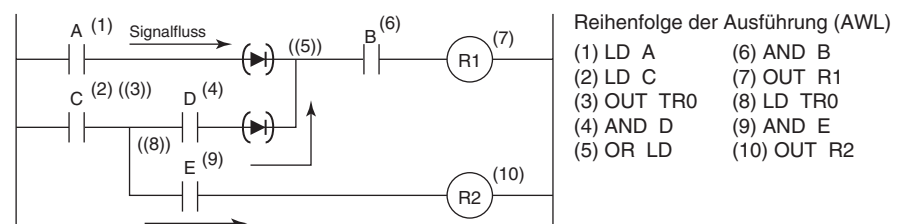


Programmadresse	Befehl (AWL)	Operand
000000	LD	000000
000001	AND	000001
000002	LDNOT	000002
000003	AND NOT	000003
000004	LD NOT	000100
000005	AND	000101
000006	OR LD	
000007	AND LD	
000008	OUT	000200
000009	END	

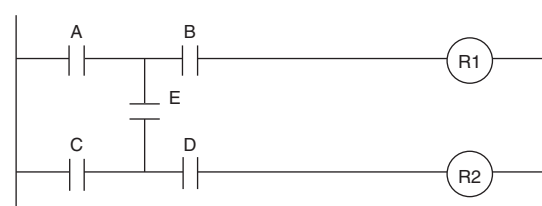
Grundbegriffe des Kontaktplanprogramms

- 1,2,3... 1. Wenn Kontaktplanprogramme von SPS ausgeführt werden, verläuft der Signalfluss immer von links nach rechts. Programme, die einen Signalfluss von rechts nach links benötigen, können nicht ausgeführt werden. Insofern unterscheidet sich dieser Fluss von dem Stromfluss in realen Schaltungen aus fest verdrahteten Steuerrelais. Wenn beispielsweise die Schaltung „a“ in ein SPS-Programm eingebunden wird, verläuft der Stromfluss, als ob die Dioden in Klammern eingesetzt wären, und die Spule R2 kann nicht über Kontakt D angesteuert werden. Die tatsächliche Ausführungsreihenfolge ist auf der rechten Seite in AWL angegeben. Um die Funktion ohne diese imaginären Dioden zu erzielen, muss die Schaltung neu geschrieben werden. Auch der Stromfluss in Schaltung „b“ kann nicht direkt programmiert werden und muss neu geschrieben werden.

Schaltung „a“

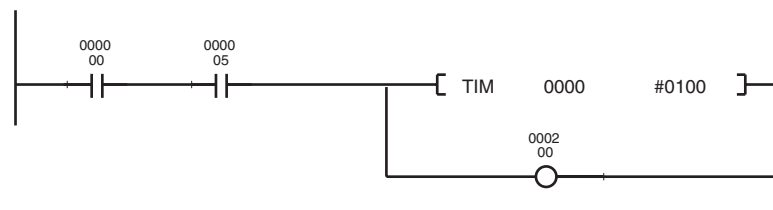


Schaltung „b“

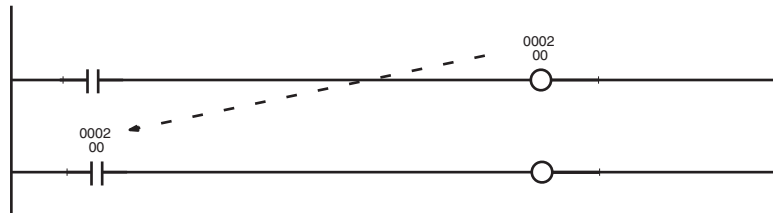


In Schaltung „a“ kann Spule R2 nicht über Kontakt D angesteuert werden. Bei Schaltung „b“ kann Kontakt E nicht in einen Kontaktplan geschrieben werden. Das Programm muss geändert werden.

- Es bestehen keine Beschränkungen der Anzahl von verwendbaren E/A-Bits, Arbeitsbits, Zeitgebern und anderen Eingangsbits. Netzwerke sollten jedoch so einfach und übersichtlich wie möglich sein, auch wenn dazu mehr Eingangsbits verwendet werden müssen.
- Es bestehen keine Einschränkungen bei der Anzahl von Eingangsbits, die in einer Reihe, parallelen Reihe oder in parallelen Strompfaden verbunden werden können.
- Es können zwei oder mehrere Ausgabebits parallel verbunden werden.



5. Ebenso lassen sich Ausgangsbits als Eingangsverknüpfungen verwenden.



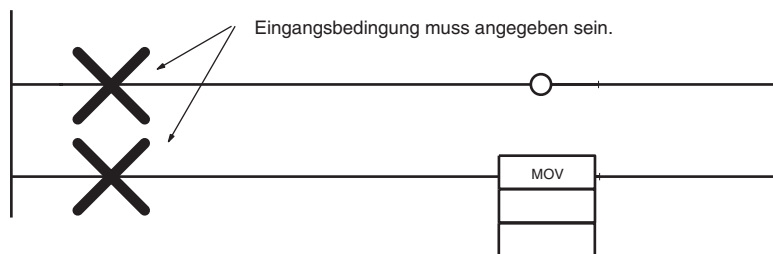
Beschränkungen

1,2,3...

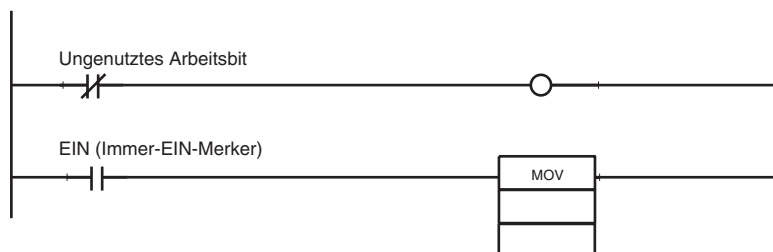
- Ein Kontaktplanprogramm-Netzwerk muss geschlossen werden, so dass die Signale (Stromfluss) von der linken Stromschiene zur rechten Stromschiene geleitet werden. Wenn das Programm-Netzwerk nicht geschlossen wird, tritt ein Kompilierungs-Fehler auf.



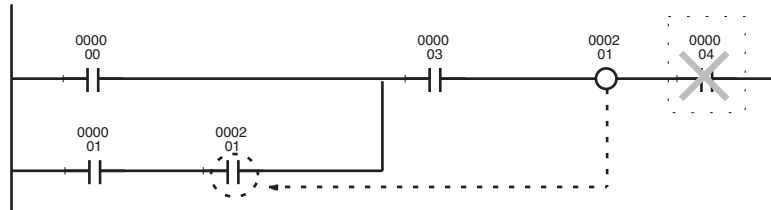
- Ausgangsbits, Zeitgeber, Zähler und andere Ausgabebefehle können nicht direkt mit der linken Stromschiene verbunden werden. Ist ein solcher Ausgabebefehl direkt mit der linken Stromschiene verbunden, wird bei der Programmierüberprüfung durch ein Programmiergerät ein Programmnetzwerk-Fehler angezeigt.



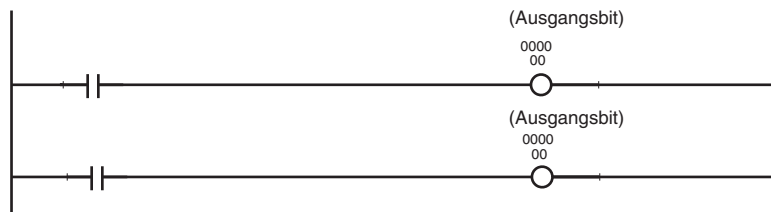
Fügen Sie als Öffner den Immer-Aus-Merker oder als Schließer den Immer-EIN-Merker ein, wenn die Eingangs-Ausführungsbedingung ständig auf EIN gehalten werden muss.



3. Ein Eingangsbit muss immer vor und nicht hinter einem Ausgabebefehl eingefügt werden wie ein Ausgangsbit. Wenn es nach einem Ausgabebefehl eingefügt wird, tritt während der Programmüberprüfung durch ein Programmiergerät ein Programmüberprüfungsfehler auf.



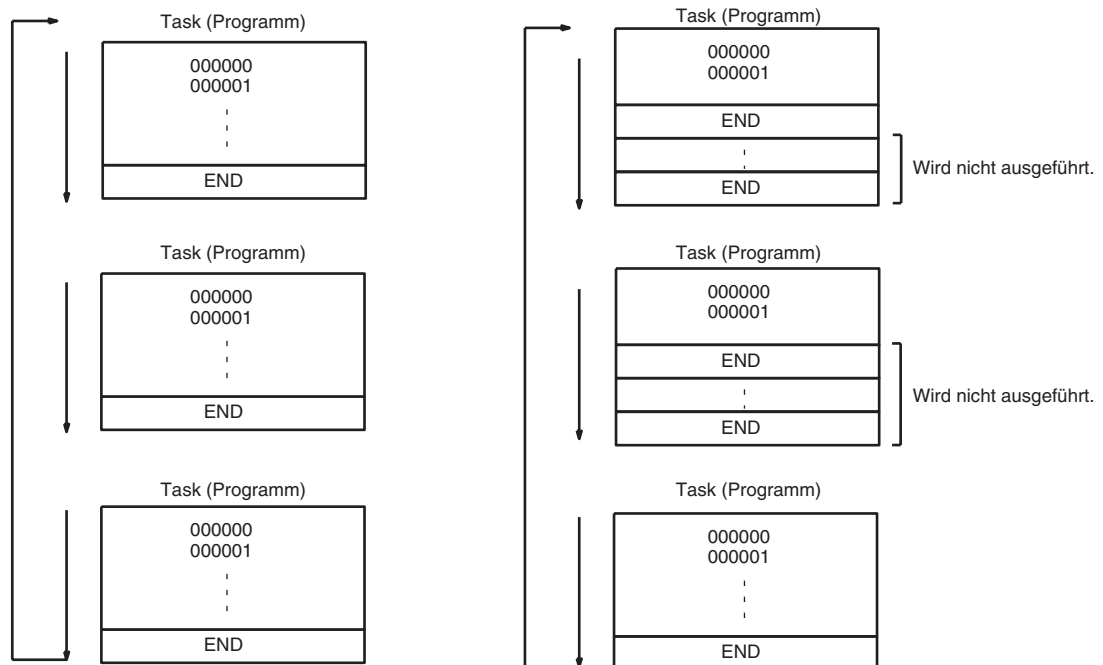
4. Das gleiche Ausgangsbit kann nur einmal in einem Ausgabebefehl programmiert werden. Befehle in einem Kontaktplanprogramm werden, mit dem obersten Strompfad beginnend, der Reihe nach in einem einzelnen Zyklus ausgeführt, daher beinhaltet das Ausgabebit letztendlich das Ergebnis des Ausgabebefehls des unteren Strompfades, und die Ergebnisse von vorherigen Befehlen, durch die das gleiche Bit gesteuert wird, werden überschrieben und nicht ausgegeben.



5. Ein Eingangsbit kann nicht in einem OUTPUT-Befehl (OUT) verwendet werden.



6. In jeder Task muss am Ende des Programms ein END(001)-Befehl eingefügt werden.
- Wenn ein Programm ohne einen END(001)-Befehl ausgeführt wird, tritt ein Programmfehler wegen eines fehlendes END-Befehls auf, die ERR/ALM-LED auf der Vorderseite der CPU-Baugruppe leuchtet und das Programm wird nicht ausgeführt.
 - Wenn in einem Programm mehr als ein END(001)-Befehl enthalten ist, wird es nur bis zum ersten END(001)-Befehl ausgeführt.

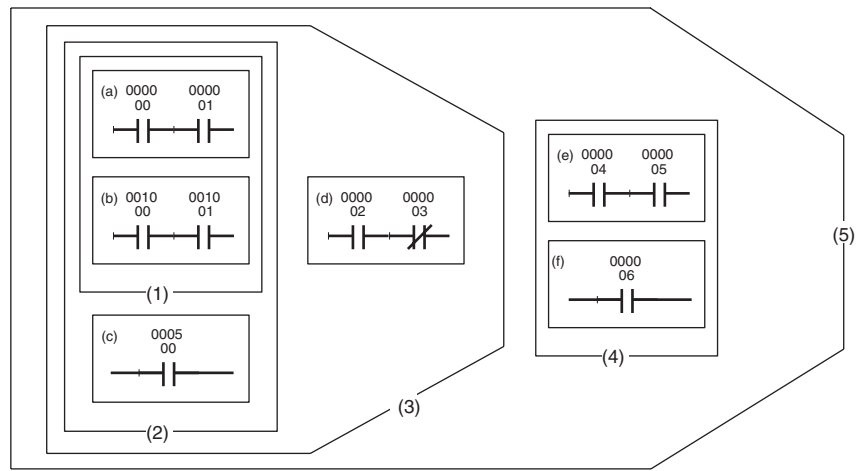
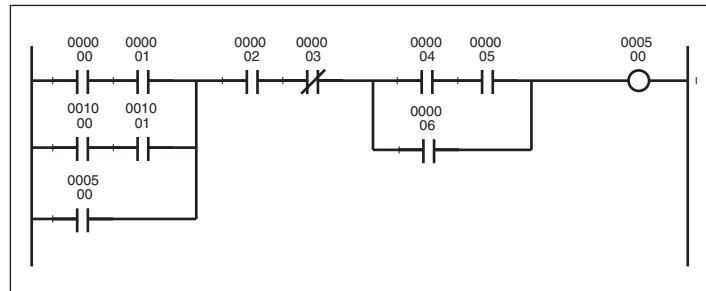


2-1-13 Eingabe in AWL

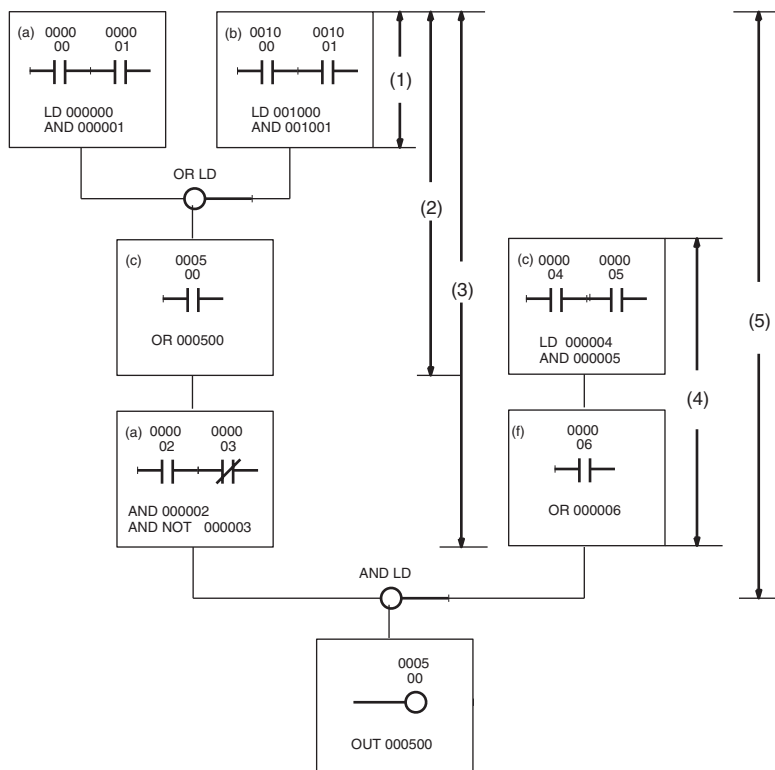
Ein logischer Start wird durch einen LD/LD NOT-Befehls gebildet. Der Bereich vom logischen Start bis zu dem Befehl unmittelbar vor dem nächsten LD/LD NOT-Befehl wird als einzelner Befehlsblock angesehen.

Erstellen Sie ein einzelnes Netzwerk, das aus zwei Befehlsblöcken besteht. Entweder mit einem AND LD-Befehl, um eine UND-Verknüpfung der beiden Blöcke zu erstellen, oder mit einem OR LD-Befehl, um eine ODER-Verknüpfung zwischen den beiden Blöcke zu erstellen. Das folgende Beispiel zeigt ein komplexes Netzwerk. An diesem wird die Vorgehensweise für die Eingabe in AWL beschrieben (Zusammenfassung und Reihenfolge).

- 1,2,3... 1. Teilen Sie das Netzwerk zunächst in kleine Blöcke (a) bis (f) auf.



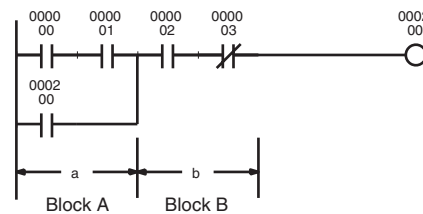
- Programmieren Sie die Blöcke von oben nach unten und dann von links nach rechts.



	Adresse	Befehl	Operand				
(a)	000200	LD	000000	(1)	(2)	(3)	(5)
	000201	AND	000001				
(b)	000202	LD	001000	(1)	(2)	(3)	(5)
	000203	AND	001001				
	000204	OR LD	---				
(c)	000205	OR	000500				
(d)	000206	AND	000002	(1)	(2)	(3)	(5)
	000207	AND NOT	000003				
(e)	000208	LD	000004	(1)	(2)	(3)	(5)
	000209	AND	000005				
(f)	000210	OR	000006	(1)	(2)	(3)	(5)
	000211	AND LD	---				
	000212	OUT	000500				

2-1-14 Programmierbeispiele

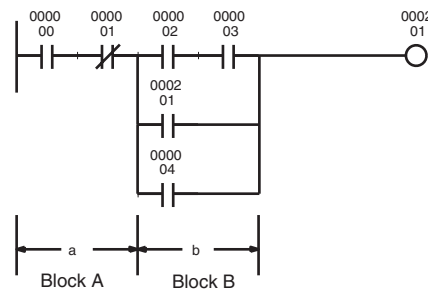
1,2,3... 1. Strompfade parallel/in Reihe



Befehl	Operanden
LD	000000
AND	000001
OR	000200
AND	000002
AND NOT	000003
OUT	000200

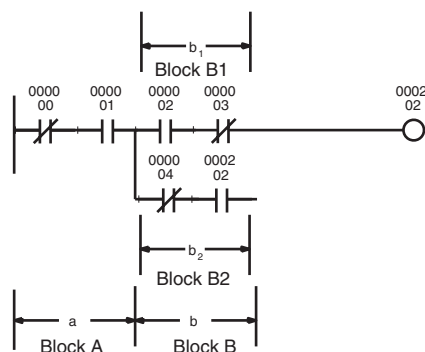
Programmieren Sie die parallele Befehlsausführung in Block A und dann in Block B.

2. Strompfade in Reihe/parallel



Befehl	Operanden
LD	000000
AND NOT	000001
LD	000002
AND	000003
OR	000201
OR	000004
AND LD	---
OUT	000201

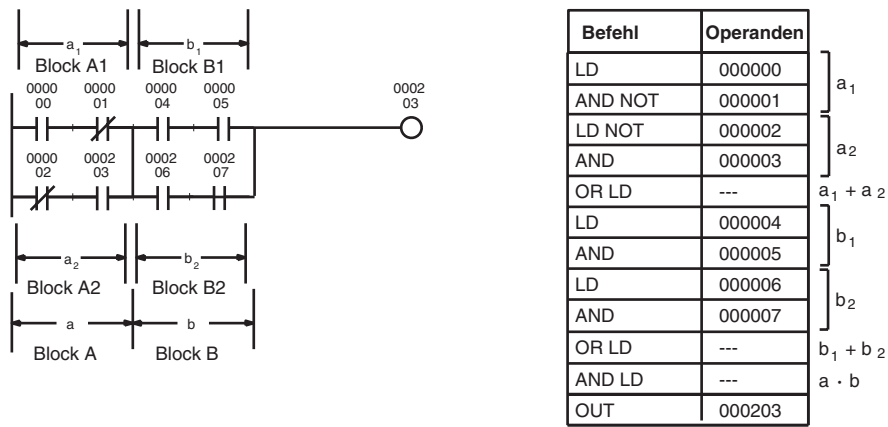
- Trennen Sie den Strompfad in die Blöcke A und B auf, und programmieren Sie sie einzeln.
- Verknüpfen Sie die Blöcke A und B durch einen AND LD-Befehl.
- Programmieren Sie Block A.



Befehl	Operanden
LD NOT	000000
AND	000001
LD	000002
AND NOT	000003
LD NOT	000004
AND	000202
OR LD	---
AND LD	---
OUT	000202

- Programmieren Sie Block B1 und danach Block B2.
- Verknüpfen Sie die Blöcke B1 und B2 durch einen OR LD-Befehl und dann die Blöcke A und B durch einen AND LD-Befehl.

3. Beispiel einer Reihenverbindung in einem Strompfad in Reihe

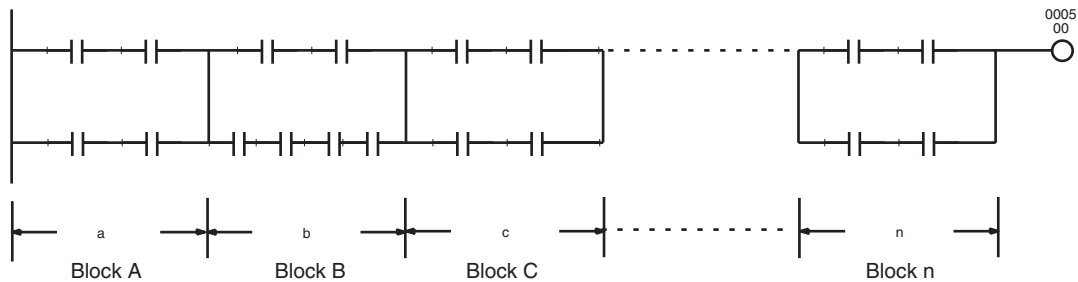


Programmieren Sie Block A₁ und Block A₂, und verknüpfen Sie dann Block A₁ mit Block A₂ durch einen OR LD-Befehl.

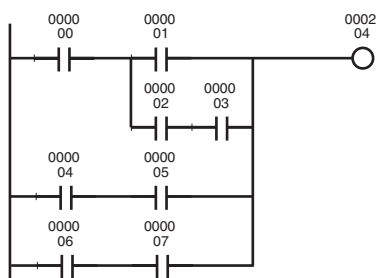
Programmieren Sie B₁ und B₂ in derselben Weise.

Verknüpfen Sie Block A und Block B durch einen AND LD-Befehl.

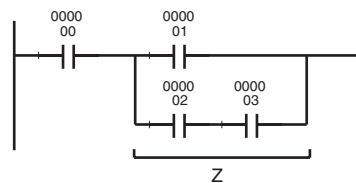
Wiederholen Sie den Vorgang für alle vorhandenen Blöcke A bis n.



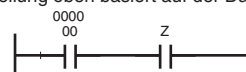
4. Komplexe Netzwerke



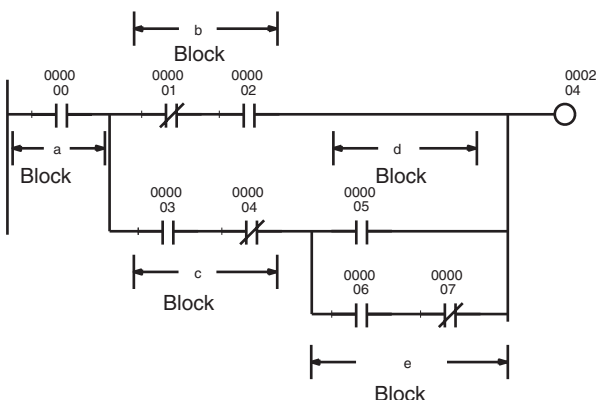
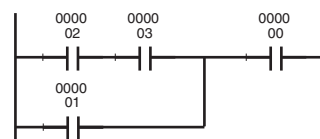
Befehl	Operand
LD	000000
LD	000001
LD	000002
AND	000003
OR LD	---
AND LD	---
LD	000004
AND	000005
OR LD	---
LD	000006
AND	000007
OR LD	---
OUT	000204



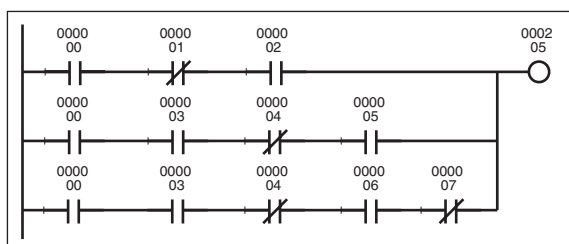
Die Darstellung oben basiert auf der Darstellung unten.



Wenn das Programm wie unten gezeigt umgeschrieben wird, kann es vereinfacht werden.

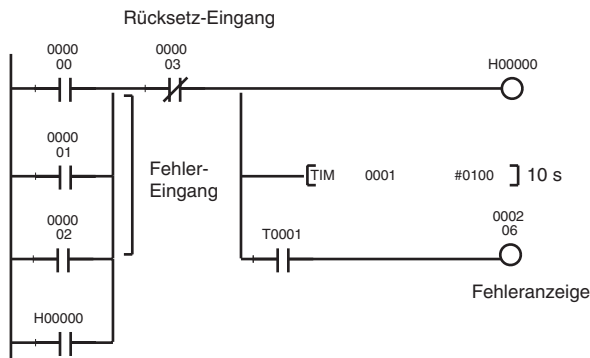


Der oben gezeigte Netzwerk kann wie folgt umgeschrieben werden:



Befehl	Operand
LD	000000
LD NOT	000001
AND	000002
LD	000003
AND NOT	000004
LD	000005
LD	000006
AND NOT	000007
OR LD	---
AND LD	---
OR LD	---
AND LD	---
OUT	000205

a
b
c
d
e
d + e
(d + e) · c
(d + e) · c + b
((d + e) · c + b) · a



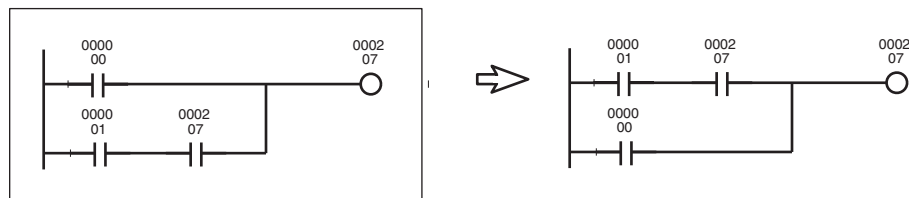
Befehl	Operand
LD	000000
OR	000001
OR	000002
OR	H00000
AND NOT	000003
OUT	H00000
TIM	0001
	0100
AND	T0001
OUT	000206

Wenn ein Haftmerker-Bit verwendet wird, bleibt der EIN/AUS-Status auch dann im Speicher erhalten, wenn die Spannungsversorgung ausgeschaltet wird, und das Fehlersignal ist noch aktiviert, wenn die Spannungsversorgung wieder eingeschaltet wird.

5. Netzwerke, die besondere Beachtung benötigen oder neu geschrieben werden müssen.

OR- und OR LD-Befehle

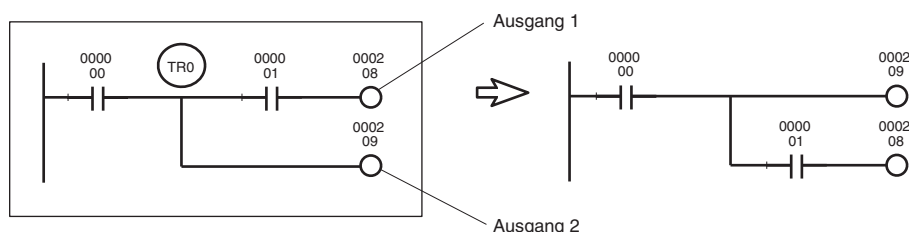
Bei einem OR oder OR NOT-Befehl wird der OR-Befehl auf das Ergebnis der Kontaktlogik vom LD- oder LD NOT-Befehl zum OR oder OR NOT-Befehl angewendet. Also können die Netzwerke so umgeschrieben werden, dass der OR LD-Befehl nicht mehr benötigt wird.



Beispiel: Ein OR LD-Befehl wird benötigt, wenn die Netzwerke wie gezeigt ohne Änderungen programmiert werden. Einige Schritte können ausgelassen werden, indem die Netzwerke wie folgt neu geschrieben werden.

Ausgabebefehlverzweigungen

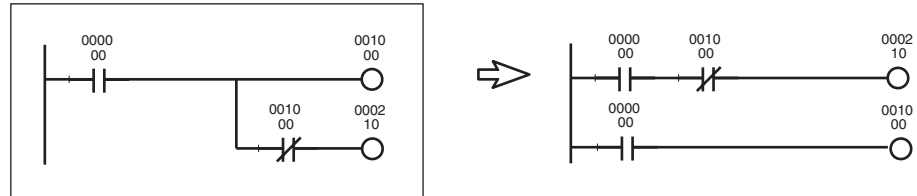
Sie benötigen ein TR-Bit, wenn vor einem AND- oder AND NOT-Befehl eine Verzweigung ist. Das TR-Bit wird nicht benötigt, wenn sich die Verzweigung an einem Punkt befindet, der direkt mit dem Ausgabebefehl und dem AND- bzw. AND NOT-Befehl verbunden ist, oder wenn der Ausgabebefehl so weitergeführt werden kann wie vorher.



Beispiel: Ein Ausgabebefehl und ein Ladebefehl (LD) für das temporäre Speicherbit TR0 werden an einem Verzweigungspunkt benötigt, wenn die Netzwerke ohne Änderung programmiert werden. Einige Schritte können ausgelassen werden, indem die Netzwerke umgeschrieben werden.

AWL-Ausführungsreihenfolge

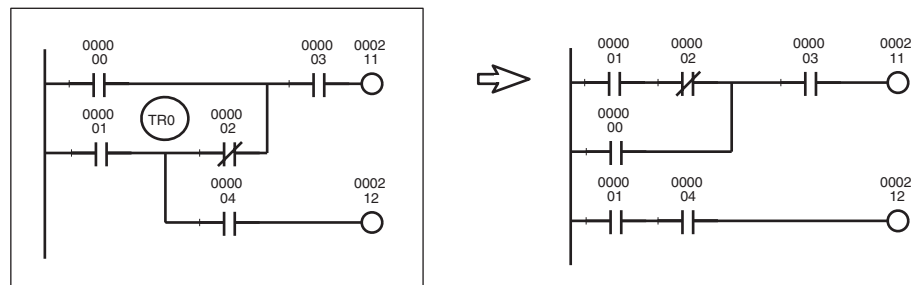
Eine SPS führt SPS-Programme in der Reihenfolge aus, in der die AWL eingegeben ist, daher werden Befehle, abhängig davon, wie die Netzwerke geschrieben wurden, möglicherweise nicht wie erwartet ausgeführt. Bedenken Sie beim Schreiben von SPS-Programmen immer die Ausführungsreihenfolge der Befehle.



Beispiel: Im oben aufgeführten Beispiel kann CIO 000210 nicht eingeschaltet werden. Durch das Neuschreiben des Netzwerks, wie oben angezeigt, kann CIO 000210 für einen Zyklus auf EIN gesetzt werden.

Netzwerke, die umgeschrieben werden müssen

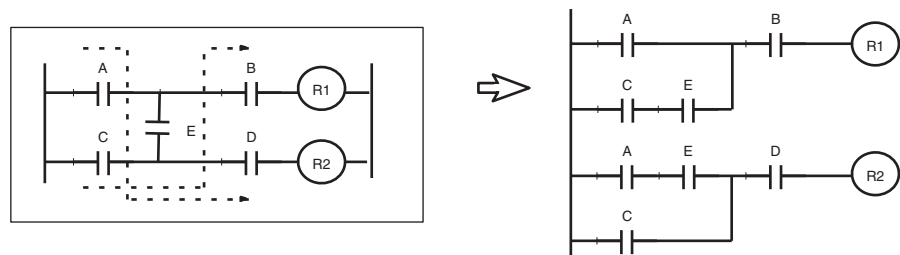
SPS führen Befehle in der Reihenfolge aus, so dass der Signalfluss (Stromfluss) im Kontaktplan von links nach rechts verläuft. Ein Stromfluss von rechts nach links kann nicht programmiert werden.



Beispiel: Das Programm kann auf die in der linken Abbildung dargestellte Weise geschrieben werden, bei der TR0 die Verzweigung erhält. Derselbe Wert wird jedoch auch von den Netzwerken auf der rechten Seite erhalten, die leichter nachvollziehbar sind. Aus diesem Grund empfiehlt es sich, das links dargestellte Netzwerk in die auf der rechten Seite gezeigten Netzwerke umzuschreiben.

Das Programm-Netzwerk unten links kann so nicht ausgeführt werden und muss umgeschrieben werden.

Die gestrichelten Pfeile zeigen den Signalfluss (Stromfluss), wenn der Schaltkreis aus Steuerrelais bestünde.



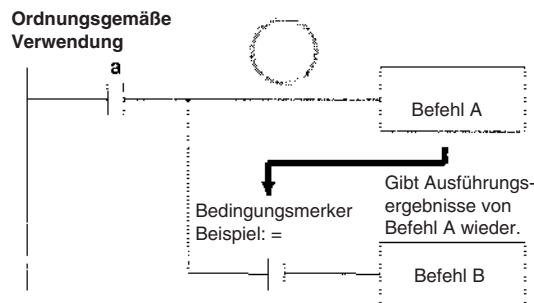
2-2 Sicherheitshinweise

2-2-1 Bedingungsmerker

Verwenden von Bedingungsmerkern

Bedingungsmerker werden gemeinsam von allen Befehlen verwendet und ändern ihren Status im Verlauf eines Zyklus entsprechend den Ausführungsergebnissen der einzelnen Befehle. Stellen Sie daher sicher, Bedingungsmerker mit der gleichen Ausführungsbedingung auf einem verzweigten Ausgang im unmittelbaren Anschluss an einen Befehl zu verwenden, um die Ergebnisse der Befehlsausführung zu erhalten. Verbinden Sie einen Bedingungsmerker niemals direkt mit der Stromschiene, da der Status des Merkers in diesem Fall die Ausführungsergebnisse anderer Befehle widerspiegelt.

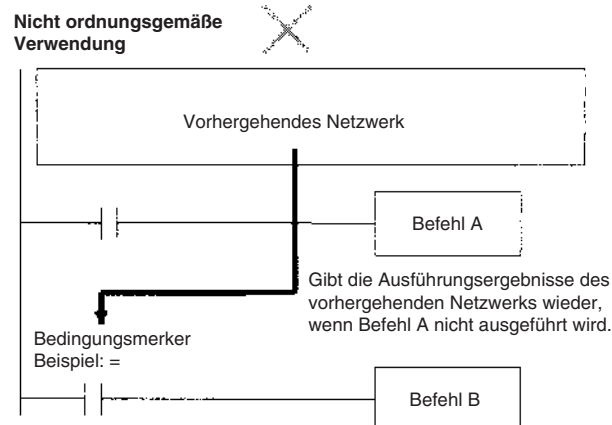
Beispiel: Verwenden der Ausführungsergebnisse von Befehl A



AWL

Befehl	Operand
LD	a
Befehl	A
AND	=
Befehl	B

Für die Befehle A und B wird die gleiche Ausführungsbedingung (a) verwendet, um Befehl B entsprechend den Ausführungsergebnissen von Befehl A auszuführen. Befehl B wird also nur entsprechend dem Bedingungsmerker ausgeführt, wenn Befehl A ausgeführt wurde.



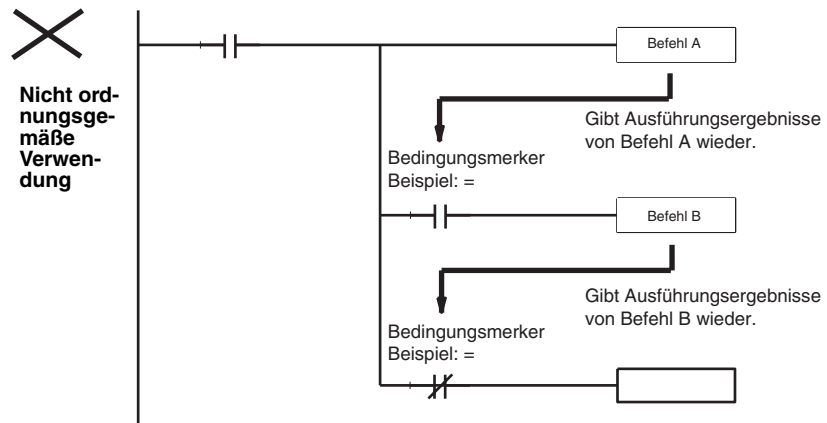
Wenn der Bedingungsmerker direkt mit der linken Stromschiene verbunden ist, wird Befehl B entsprechend den Ausführungsergebnissen eines früheren Strompfads ausgeführt, wenn Befehl A nicht ausgeführt wird.

Hinweis

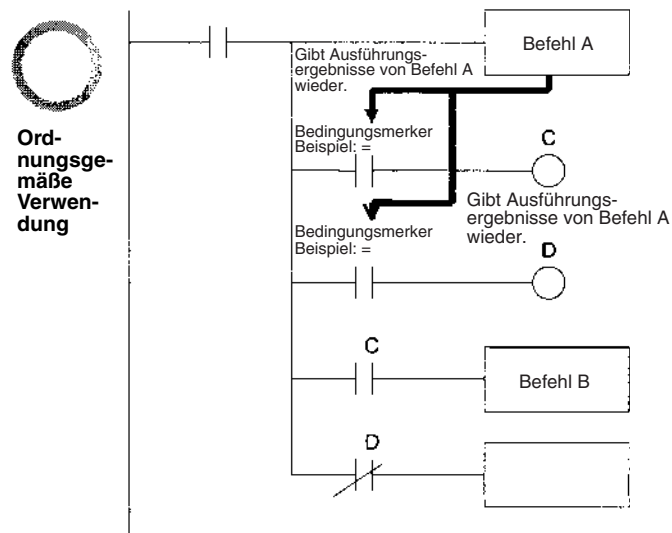
Bedingungsmerker werden von allen Befehlen innerhalb eines Einzelprogramms (Task) verwendet und gelöscht, wenn die Task wechselt. Daher gelten Ausführungsergebnisse der vorhergehenden Tasks nicht in späteren Tasks. Da Bedingungsmerker von allen Befehlen gemeinsam verwendet werden, muss sichergestellt werden, dass diese sich nicht innerhalb eines einzelnen SPS-Programms überschneiden. Im Folgenden finden Sie ein Beispiel.

Verwenden von Ausführungsergebnissen in Öffner- und Schließer-Eingangsbedingungen

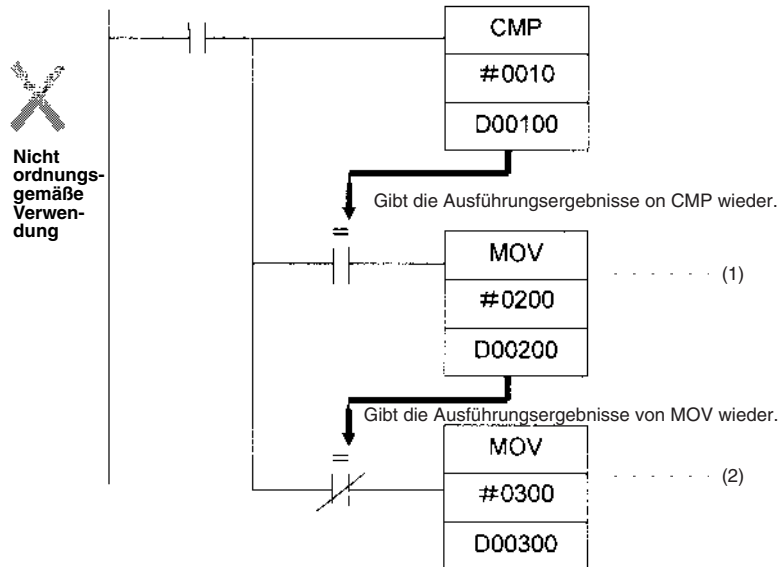
Die Bedingungsmerker übernehmen die Ausführungsergebnisse von Befehl B (siehe folgendes Beispiel), selbst wenn die Öffner- und Schließer-Eingangsbedingungen von der gleichen Verzweigung ausgeführt werden.



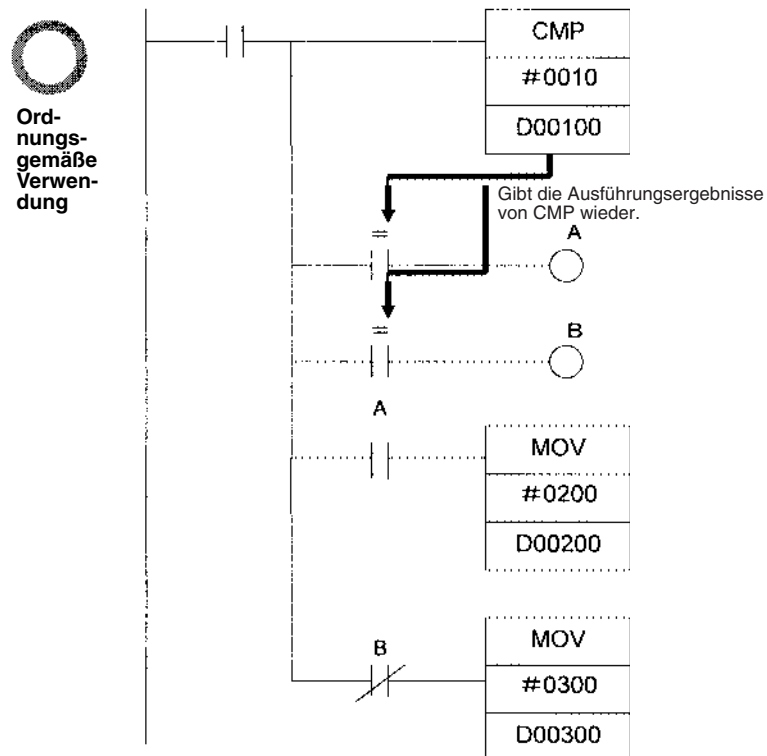
Stellen Sie sicher, dass jedes der Ergebnisse einmal von einem Ausgangsbefehl übernommen wird, damit die Ausführungsergebnisse für Befehl B nicht übernommen werden.



Beispiel: Im folgenden Beispiel wird #0200 nach D00200 verschoben, wenn D00100 den Wert #0010 enthält, und #0300 wird nach D00300 verschoben, wenn D00100 nicht den Wert #0010 enthält.



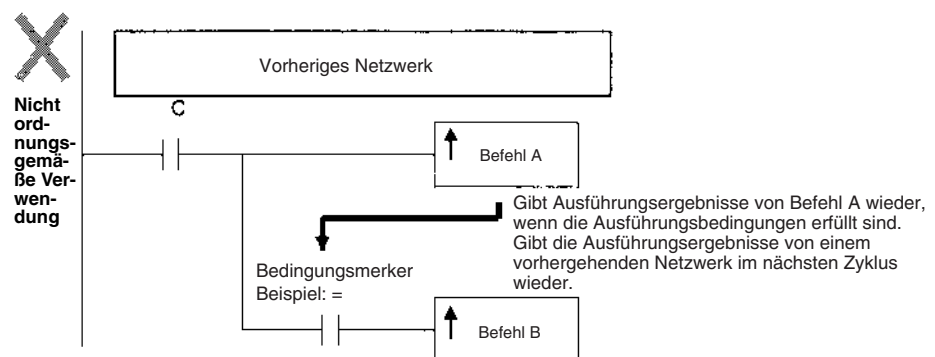
Der Gleich-Merker wird auf EIN gesetzt, wenn D00100 im vorherigen Strompfad #0010 enthält. #0200 wird für Befehl (1) nach D00200 verschoben, anschließend wird jedoch der Gleich-Merker auf AUS gesetzt, da die Quelldaten #0200 ungleich 0000 Hex sind. Der Befehl MOV bei (2) wird anschließend ausgeführt und #0300 wird nach D00300 verschoben. Daher muss, wie im Folgenden gezeigt, dieses Network erweitert werden, um zu verhindern, dass die Ausführungsergebnisse des ersten MOVE-Befehls übernommen werden.



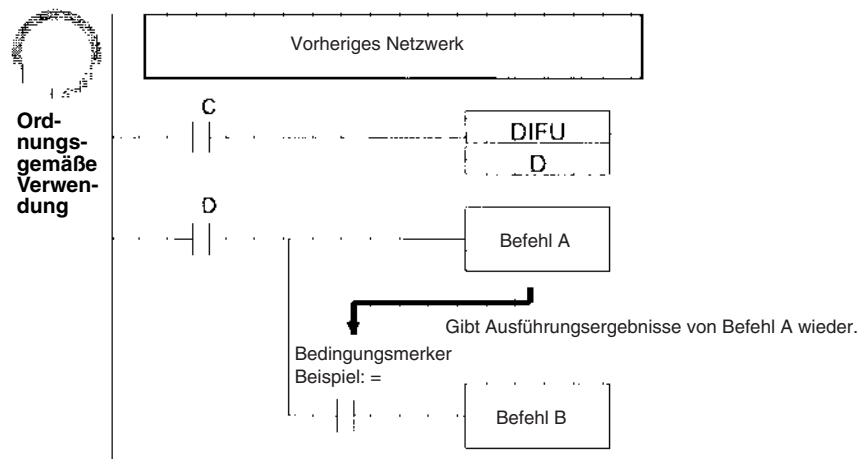
Verwenden der Ausführungsergebnisse von Befehlen mit Flankenauswertung

Bei Befehlen mit Flankenauswertung werden die Ausführungsergebnisse nur in Bedingungsmerkern wiedergegeben, wenn die Ausführungsbedingung erfüllt ist. Die Ergebnisse eines vorhergehenden Netzwerks (und nicht die Ausführungsergebnisse des Befehls mit Flankenauswertung) werden in Bedingungsmerkern des nächsten Zyklus verwendet. Daher muss auf die Bedingungsmerker des folgenden Zyklus geachtet werden, wenn die Ausführungsergebnisse von Befehlen mit Flankenauswertung verwendet werden sollen.

Im folgenden Beispiel werden die Befehle A und B nur ausgeführt, wenn die Ausführungsbedingung C erfüllt ist. Es treten jedoch folgende Probleme auf, wenn Befehl B Ausführungsergebnisse von Befehl A übernimmt. Wenn die Ausführungsbedingung C im an die Ausführung von Befehl anschließenden Zyklus weiter auf EIN gesetzt bleibt, wird Befehl B unerwartet ausgeführt (durch die Ausführungsbedingung), wenn der Bedingungsmerker aufgrund von Ergebnissen eines vorhergehenden Netzwerks von AUS auf EIN wechselt.



In diesem Fall handelt es sich bei den Befehlen A und B nicht um Befehle mit Flankenauswertung. Stattdessen wird der Befehl DIFU (von DIFD) wie im Folgenden gezeigt verwendet, und die Befehle A und B werden beide nur an der steigenden (oder fallenden) Flanke und somit nur in einem Zyklus ausgeführt.



Hinweis Die CS1-H, CJ1-H, CJ1M oder CS1D CPU-Baugruppen unterstützen Befehle zum Speichern (CCS(282)) und Laden (CCL(283)) der Bedingungsmerkerzustände. Mit diesen Befehlen können die Bedingungsmerkerzustände gespeichert und an anderer Stelle in der Task oder in einer anderen Task verwendet werden.

Hauptbedingungen, unter denen Bedingungsmerker auf EIN gesetzt werden

Fehlermerker

Der Fehlermerker (P_ER) wird unter bestimmten Bedingungen auf EIN gesetzt, z. B. wenn die Operandendaten eines Befehls ungültig sind. Der Befehl wird nicht ausgeführt, wenn der Fehlermerker P_ER auf EIN gesetzt wird.

Wenn der Fehlermerker P_ER auf EIN gesetzt ist, ändert sich der Zustand anderer Bedingungsmerker (z. B. der P_LT, P_GT, P_OF- und P_UF-Merker) nicht und der Zustand der Gleich- und Negativ-Merker hängt vom jeweiligen Befehl ab.

Weitere Informationen zu den einzelnen Befehlen und den Bedingungen, unter denen der Fehlermerker auf EIN gesetzt wird, finden Sie im *CS/CJ-Serie Speicherprogrammierbare Steuerungen Befehlsreferenzhandbuch (W340)*. Beachten Sie hierbei, dass einige Befehle den Fehlermerker unabhängig von den Bedingungen auf AUS setzen.

Hinweis

Die Einstellungen für das Auftreten eines Befehlsfehlers im SPS-Setup legen fest, ob der Betrieb angehalten wird, wenn der Fehlermerker auf EIN gesetzt wird. In der Standardeinstellung wird der Betrieb fortgeführt, wenn der Fehlermerker auf EIN gesetzt wird. Wenn eine Betriebsunterbrechung für den Fall, dass der Fehlermerker auf EIN gesetzt wird, festgelegt wurde, und der Betrieb angehalten wird (aufgrund eines Programmfehlers), wird die Programmadresse, an der der Betrieb abgebrochen wurde, in A298 und A299 gespeichert. Gleichzeitig wird A29508 auf EIN gesetzt.

Gleich-Merker

Beim Gleich-Merker handelt es sich um einen temporären Merker für alle Befehle, es sei denn die Vergleichsergebnisse sind gleich (=). Er wird automatisch vom System gesetzt und ändert seinen Status. Der Gleich-Merker kann von einem Befehl auf AUS (EIN) gesetzt werden, nachdem er von einem anderen Befehl auf EIN (AUS) gesetzt wurde. Der Gleich-Merker wird beispielsweise auf EIN gesetzt, wenn MOV oder ein anderer Kopierbefehl 0000 Hex als Quelldaten speichert und bleibt in allen anderen Fällen auf AUS gesetzt. Selbst wenn ein Befehl den Gleich-Merker auf EIN setzt, wird der Kopierbefehl unverzüglich ausgeführt und der Gleich-Merker wird, je nachdem, ob die Quelldaten gleich 0000 Hex sind oder nicht, auf EIN oder AUS gesetzt.

Übertragsmerker

Der Übertragsmerker wird für Verschiebepfehle, Additions- und Subtraktionsbefehle mit Übertrag, Additions- und Subtraktionsbefehlsüberträge mit negativen Überträgen, Spezial-E/A-Baugruppen-Befehle, PID-Befehle und FPD-Befehle verwendet. Beachten Sie die folgenden Sicherheitshinweise.

Hinweis

1. Der Übertragsmerker kann aufgrund der Ausführungsergebnisse eines bestimmten Befehls EIN (AUS) verbleiben und anschließend für einen anderen Befehl verwendet werden (einen Additions- und Subtraktionsbefehl mit Übertrag oder einen Verschiebepfehl). Stellen Sie sicher, den Übertragsmerker falls erforderlich zu löschen.
2. Der Übertragsmerker kann von den Ausführungsergebnissen eines bestimmten Befehls auf EIN (AUS) gesetzt und von einem anderen Befehl auf AUS (EIN) gesetzt werden. Stellen Sie sicher, die richtigen Ergebnisse zu erhalten, falls der Übertragsmerker berücksichtigt wird.

Kleiner-als- und Größer-als-Merker

Die Merker P_LT und P_GT werden für Vergleichsbefehle sowie u. a. die Befehle LMT, BAND, ZONE und PID verwendet.

Der P_LT oder P_GT Merker kann von einem anderen Befehl auf AUS (EIN) gesetzt werden, selbst wenn er von den Ausführungsergebnissen eines bestimmten Befehls auf EIN (AUS) gesetzt wurde.

Negativ-Merker

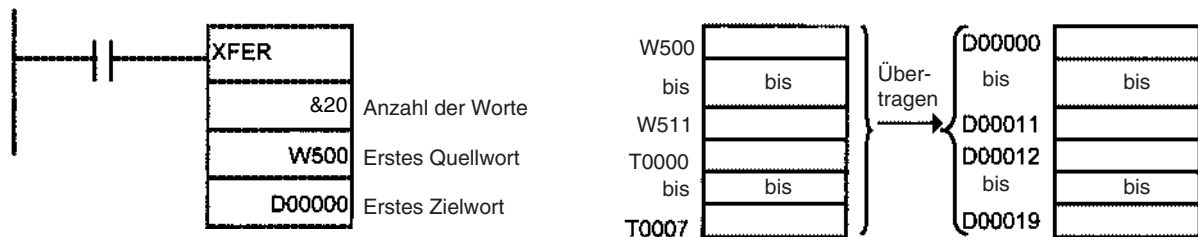
Der Negativ-Merker (P_N) wird auf EIN gesetzt, wenn das äußerste linke Bit des Worts mit den Befehlsausführungs-Ergebnissen bestimmter Befehle „1“ ist. Er ist weiterhin für alle anderen Befehle auf AUS gesetzt.

Spezifizieren von Operanden für mehrere Worte

Bei SPS der CS/CJ-Serie wird ein Befehl wie geschrieben ausgeführt, selbst wenn ein Operand spezifiziert wurde, der mehrere Worte erfordert, so dass nicht alle Worte des Operanden im selben Bereich liegen. In diesem Fall werden die Worte in der Reihenfolge der SPS-Speicheradressen verwendet. Der Fehlermerker wird **nicht** auf EIN gesetzt.

So kann beispielsweise eine Blockübertragung mit dem Befehl XFER(070) ausgeführt werden, bei der 20 Worte, beginnend mit W500, für die Übertragung spezifiziert wurden. In diesem Fall wird der bei W511 endende Arbeitsbereich erweitert und der Befehl ausgeführt, ohne den Fehlermerker auf EIN zu setzen. In den SPS-Speicheradressen werden die Zeitgeber-Istwerte im auf den Arbeitsbereich folgenden Speicher gespeichert. Daher werden durch den folgenden Befehl W500 bis W511 nach D00000 bis D00011 und die Istwerte für T0000 bis T0007 nach D00012 bis D00019 übertragen.

Hinweis Weitere Informationen zu den SPS-Speicheradressen finden Sie im Anhang *Aufteilung des SPS-Speicheradressbereichs*.



2-2-2 Spezielle Programmabschnitte

Programme für die CS/CJ-Serie verfügen über spezielle Programmabschnitte, die die Befehlsbedingungen steuern. Die folgenden speziellen Programmabschnitte sind verfügbar.

Programmabschnitt	Befehle	Befehlsbedingung	Status
Unterprogramm	Befehle SBS, SBN und RET	Das Unterprogramm wird ausgeführt.	Der Unterprogrammabschnitt zwischen den Befehlen SBN und RET wird ausgeführt.
Abschnitt IL - ILC	Befehle IL und ILC	Abschnitt ist gesperrt	Die Ausgangsbits werden auf AUS gesetzt und die Zeitgeber zurückgesetzt. Andere Befehle werden nicht ausgeführt und der vorherige Status wird beibehalten.
Schritt-Programmabschnitt	Befehle STEP und SNXT		
FOR-NEXT-Schleife	Befehle FOR und NEXT	Unterbrechung des sequenziellen Programm-Ablaufs:	Programmschleifen-Ausführung
Abschnitt JMP0 - JME0	Befehle JMP0 und JME0		Sprung
Blockprogrammabschnitt	Befehle BPRG und BEND	Blockprogramm wird ausgeführt.	Das in der AWL zwischen den Befehlen BPRG und BEND aufgelistete Blockprogramm wird ausgeführt.

Befehlskombinationen

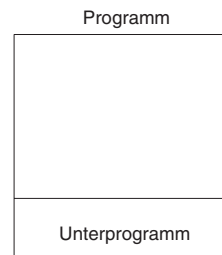
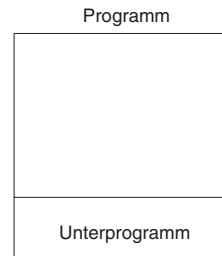
Der folgenden Tabelle können Sie entnehmen, welche Spezialbefehle innerhalb anderer Programmabschnitte verwendet werden können.

	Unterprogramm	Abschnitt IL - ILC	Schritt-Programmabschnitt	FOR-NEXT-Schleife	Abschnitt JMP0 - JME0	Blockprogrammabschnitt
Unterprogramm	Nicht möglich	Nicht möglich	Nicht möglich	Nicht möglich	Nicht möglich	Nicht möglich
IL - ILC	OK	Nicht möglich	Nicht möglich	OK	OK	Nicht möglich
Schritt-Programmabschnitt	Nicht möglich	OK	Nicht möglich	Nicht möglich	OK	Nicht möglich
FOR-NEXT-Schleife	OK	OK	Nicht möglich	OK	OK	Nicht möglich
JMP0 - JME0	OK	OK	Nicht möglich	Nicht möglich	Nicht möglich	Nicht möglich
Blockprogrammabschnitt	OK	OK	OK	Nicht möglich	OK	Nicht möglich

Hinweis Befehle, die Programmbereiche spezifizieren, können nicht für Programme in anderen Tasks verwendet werden. Ausführliche Informationen hierzu finden Sie im Abschnitt 4-2-2 *Einschränkungen bei Task-Befehlen*.

Unterprogramme

Platzieren Sie alle Unterprogramme in sämtlichen Programmen unmittelbar vor dem Befehl END(001), jedoch hinter anderen Programmabschnitten, die keine Unterprogramme darstellen. (Daher kann ein Unterprogramm nicht in einem Schritt-Programmabschnitt, Blockprogramm sowie einem FOR - NEXT- oder JMP0 - JME0-Abschnitt platziert werden.) Wenn ein Programmabschnitt, der kein Unterprogramm ist, hinter einem Unterprogramm (SBN bis RET) platziert wird, wird dieser Programmabschnitt nicht ausgeführt.



In Unterprogrammen nicht verfügbare Befehle

Folgende Befehle dürfen nicht in Unterprogrammen platziert werden.

Funktion	Befehl	Befehlsausführung
Prozess-Step-Steuerung	STEP(008)	Definieren des Schritt-Programmabschnitts
	SNXT(009)	Schritte durch Schritt-Programmabschnitte ausführen

Hinweis

Blockprogrammabschnitte

Ein Unterprogramm kann einen Blockprogrammabschnitt enthalten. Wenn sich das Blockprogramm jedoch im Status WAIT(Warten) befindet, während die Ausführung vom Unterprogramm zum Hauptprogramm zurückkehrt, verbleibt der Blockprogrammabschnitt beim nächsten Aufruf im Status WAIT(Warten).

**In Schritt-
Programmabschnitten
nicht verfügbare Befehle**

Funktion	AWL	Befehl
Sequenz- steuerung	FOR(512), NEXT(513) und BREAK(514)	FOR, NEXT und BREAK Programmschleifen
	END(001)	Programmende
	IL(002) und ILC(003)	INTERLOCK und INTERLOCK CLEAR (Verriegelung, Verriegelung löschen)
	JMP(004) und JME(005)	JUMP und JUMP END (Sprung, Sprungende)
	CJP(510) und CJPN(511)	CONDITIONAL JUMP und CONDI- TIONAL JUMP NOT (Bedingter Sprung, Bedingter negierter Sprung)
	JMP0(515) und JME0(516)	MULTIPLE JUMP und MULTIPLE JUMP END (Mehrfachsprung, Mehrfachsprung-Ende)
Unter- programme	SBN(092) und RET(093)	SUBROUTINE ENTRY und SUB- ROUTINE RETURN (Unterpro- gramm-Anfang, Unterprogramm- Ende)
Block- programme	IF(802) (NOT), ELSE(803) und IEND(804)	Verzweigungsbefehle
	BPRG(096) und BEND(801)	BLOCK PROGRAM BEGIN/END (Blockprogramm-Anfang, Blockprogramm-Ende)
	EXIT(806) (NOT)	CONDITIONAL BLOCK EXIT (NOT) (Bedingtes Blockende (NICHT))
	LOOP(809) und LEND(810) (NOT)	Schleifensteuerung (Schleife, Schleife beenden)
	WAIT(805) (NOT)	ONE CYCLE WAIT (NOT) (Ein Zyklus und warten (NICHT))
	TIMW(813)	TIMER WAIT (Zeitgeber warten)
	TMHW(815)	HIGH-SPEED TIMER WAIT (Schneller Zeitgeber warten)
	CNTW(814)	COUNTER WAIT (Zähler warten)
	BPPS(811) und BPRS(812)	BLOCK PROGRAM PAUSE und RESTART (Blockprogramm Pause, Blockprogramm Restart)

Hinweis

1. Ein Step-Programmabschnitt kann in einem Sperrabschnitt verwendet werden (zwischen IL und ILC). Der Step-Programmabschnitt wird vollständig zurückgesetzt, wenn die Sperre auf EIN gesetzt ist.
2. Ein Step-Programmabschnitt kann zwischen Mehrfachsprung (JMP0) und Mehrfachsprung-Ende (JME0) verwendet werden.

In Blockprogrammabschnitten nicht verfügbare Befehle

Folgende Befehle dürfen nicht in Blockprogrammabschnitten platziert werden.

Klassifizierung nach Funktion	AWL	Befehl
Sequenzsteuerung	FOR(512), NEXT(513) und BREAK(514)	FOR, NEXT und BREAK Programmschleifen
	END(001)	Programmende
	IL(002) und ILC(003)	INTERLOCK und INTERLOCK CLEAR (Verriegelung, Verriegelung löschen)
	JMP0(515) und JME0(516)	MULTIPLE JUMP und MULTIPLE JUMP END (Mehrfachsprung, Mehrfachsprung-Ende)
Sequenzeingang	UP(521)	CONDITION ON (Bedingung steigende Flanke)
	DOWN(522)	CONDITION OFF (Bedingung fallende Flanke)
Ausgabebefehle	DIFU	DIFFERENTIATE UP (Ausführung bei steigender Flanke)
	DIFD	DIFFERENTIATE DOWN (Ausführung bei fallender Flanke)
	KEEP	KEEP (Halten)
	OUT	OUTPUT (Ausgang)
	OUT NOT	OUTPUT NOT (Ausgang NICHT)
Zeitgeber/Zähler	TIM	TIMER
	TIMH	HIGH-SPEED TIMER
	TMHH(540)	ONE-MS TIMER (1-ms-Zeitgeber)
	TTIM(087)	ACCUMULATIVE TIMER (Akkumulierender Zeitgeber)
	TIML(542)	LONG TIMER (Langer Zeitgeber)
	MTIM(543)	MULTI-OUTPUT TIMER (Multi-Ausgabe-Zeitgeber)
	CNT	COUNTER (Zähler)
	CNTR	REVERSIBLE COUNTER (Umkehrbarer Zähler)
Unterprogramme	SBN(092) und RET(093)	SUBROUTINE ENTRY und SUBROUTINE RETURN (Unterprogramm-Anfang, Unterprogramm-Ende)
Datenverschiebung	SFT	SHIFT (Schieberegister)
Schrittprogramm-Steuerung	STEP(008) und SNXT(009)	STEP DEFINE und STEP START (Schritt, nächster Schritt)
Datensteuerung	PID	PID CONTROL (PID-Regelung)
Blockprogramm	BPRG(096)	BLOCK PROGRAM BEGIN (Blockprogramm-Anfang)
Fehlerdiagnose	FPD(269)	FAILURE POINT DETECTION (Fehlerpunkt-Erfassung)

Hinweis

1. Blockprogramme können in einem Step-Programmabschnitt verwendet werden.
2. Ein Blockprogramm kann in einem Sperrabschnitt verwendet werden (zwischen IL und ILC). Der Blockprogrammabschnitt wird nicht ausgeführt, wenn gesperrt zwischen IL und ILC.
3. Ein Blockprogrammabschnitt kann zwischen Mehrfachsprung (JMP0) und Mehrfachsprung-Ende (JME0) verwendet werden.
4. Der Sprung-Befehl (JMP) und der bedingte Sprung-Befehl (CJP/CJPN) können in einem Blockprogrammabschnitt verwendet werden. Die Befehle Sprung (JMP) und Sprung-Ende (JME), wie auch die Befehle bedingter Sprung (CJP/CJPN) und Sprung-Ende (JME), können in einem Blockprogrammabschnitt nicht verwendet werden, wenn sie nicht paarweise verwendet werden. Ein Programm kann nicht ordnungsgemäß ausgeführt werden, wenn die Befehle nicht als Paare verwendet werden.

2-3 Überprüfen von Programmen

Programme für die CS/CJ-Serie können in den folgenden Stadien überprüft werden.

- Überprüfung während der Eingabe mit der Programmierkonsole
- Programmüberprüfung durch CX-Programmer
- Befehlsüberprüfung während der Ausführung
- Überprüfung auf schwerwiegende Fehler (Programmfehler) während der Ausführung

2-3-1 Fehler während der Eingabe in das Programmiergerät

Programmierkonsole

Fehler in den folgenden Punkten werden von der Programmierkonsole während der Eingabe angezeigt.

Fehleranzeige	Ursache
CHK MEM	Schalter 1 des DIP-Schalterblocks der CPU-Baugruppe ist auf ON gestellt (Schreibschutz).
IO No. ERR	Eine unzulässige E/A-Eingabe ist aufgetreten.

CX-Programmer

Das Programm wird zu den folgenden Zeitpunkten automatisch von CX-Programmer überprüft.

Zeitpunkt	Überprüfter Inhalt
Bei Eingabe von SPS-Programm	Befehlseingaben, Operandeneingaben, Programmierungsschema
Beim Laden von Dateien	Alle Operanden aller Befehle und alle Programmierungsschemata
Beim Herunterladen von Dateien	Von der CS/CJ-Serie unterstützte Modelle und alle Operanden aller Befehle
Während der Online-Bearbeitung	Kapazität usw.

Die Ergebnisse der Überprüfung werden auf der Textregisterkarte des Ausgabefensters angezeigt. Die linke Stromschiene von unzulässigen Programmabschnitten wird in der Kontaktplanansicht rot dargestellt.

2-3-2 Programmprüfungen mit CX-Programmer

In der folgenden Tabelle sind die möglichen Fehler aufgeführt, die von der Programmüberprüfungs-Funktion in CX-Programmer gefunden werden können.

CX-Programmer führt keine Prüfung auf Bereichsfehler bei indirekt adressierten Operanden in Befehlen aus. Fehler bei der indirekten Adressierung werden während der Programmausführungsüberprüfung gefunden und der P_ER-Merker wird auf EIN gesetzt, wie im folgenden Abschnitt beschrieben. Ausführliche Informationen hierzu finden Sie im *CS/CJ-Serie Speicherprogrammierbare Steuerungen Befehlsreferenzhandbuch (W340)*.

Für die Programmüberprüfung durch CX-Programmer kann der Anwender die Überprüfungsstufen A, B oder C wählen (in Reihenfolge der Schwere des Fehlers) sowie eine anwenderdefinierte Überprüfungsstufe.

Bereich	Überprüfung
Unzulässige Daten: Kontaktplanprogrammierung	Befehlspositionen
	E/A-Linien
	Verbindungen
	Vollständigkeit von Befehl und Ausführung
Befehlsunterstützung durch SPS	Befehls- und Operandenunterstützung durch SPS
	Befehlsvarianten (NOT, !, @ und %)
	Integrität des Objektcodes

Bereich	Überprüfung
Bereiche der Operanden	Speicherbereiche der Operanden
	Datentypen der Operanden
	Zugriffsprüfung auf schreibgeschützte Worte
	Bereichsüberprüfung der Operanden, einschließlich des Folgenden.
	• Konstanten (#, &, +, -) • Steuercodes • Überprüfen der Bereichsgrenzen bei Multi-Wort-Operanden • Überprüfen der Größenverhältnisse bei Multi-Wort-Operanden • Überschneidungen der Operandenbereiche • Multi-Wort-Zuordnungen • Operanden doppelter Länge • Überprüfen der Bereichsgrenzen bei Offsets
	Anzahl der Steps
	Gesamtkapazität
	Anzahl der Tasks
Syntax	Aufrufsüberprüfung bei Befehlspaaren
	• IL-ILC • JMP-JME, CJP/CJPN-JME • SBS-SBN-RET, MCRO-SBN-RET • STEP-SNXT • BPRG-BEND • IF-IEND • LOOP-LEND
	Eingeschränkte Programmpositionen für BPRG-BEND
	Eingeschränkte Programmpositionen für SBN-RET
	Eingeschränkte Programmpositionen für STEP-SNXT
	Eingeschränkte Programmpositionen für FOR-NEXT
	Eingeschränkte Programmpositionen für Interrupt-Tasks
	Erforderliche Programmpositionen für BPRG-BEND
	Erforderliche Programmpositionen für FOR-NEXT
	Unzulässige Schachtelungen
	END(001)-Befehl
	Anzahlübereinstimmung
Kontaktplan-Struktur	Stapelüberläufe
Doppelte Ausgänge	Überprüfung auf doppelte Ausgänge
	• Nach Bit • Nach Wort • Zeitgeber- und Zähler-Befehle • Mehrere Worte (2 Worte und 4 Worte) • Mehrere zugeordnete Worte • Start/End-Bereiche • FAL-Nummern • Befehle mit mehreren Ausgabeoperanden
Tasks	Prüfung auf Tasks, die auf Starten zu Beginn des Betriebs eingestellt sind
	Task-Programmzuordnung

Hinweis

Zwischen den Tasks werden die doppelten Ausgänge nicht überprüft, lediglich innerhalb einzelner Tasks.

Multi-Wort-Operanden

Während der Programmüberprüfung werden Speicherbereichsgrenzen auf Multi-Wort-Operanden überprüft, wie in der folgenden Tabelle angegeben.

CX-Programmer	Programmierkonsolen
CX-Programmer bietet die folgenden Funktionen für Multi-Wort-Operanden, die eine Speicherbereichsgrenze überschreiten. • Das Programm kann nicht auf CPU-Baugruppe übertragen werden. • Ebenso kann das Programm nicht von der CPU-Baugruppe gelesen werden. • Kompilierungsfehler wurden für die Programmüberprüfung erzeugt. • Während der Offline-Programmierung werden Warnungen auf dem Bildschirm angezeigt. • Während der Online-Bearbeitung in der PROGRAM- oder MONITOR-Betriebsart werden Warnungen auf dem Bildschirm angezeigt.	Überprüfung während der Programmeingabe, d. h. Operanden, die eine Speicherbereichsgrenze überschreiten, können nicht geschrieben werden.

2-3-3 Prüfung der Programmausführung

Die Überprüfung von Operanden- und Befehlspositionen bei den Befehlen wird während der Eingabe durch ein Programmiergerät (einschließlich Programmierkonsole) und ebenso während der Programmüberprüfung durch Programmiergeräte (ausgenommen Programmierkonsole) durchgeführt. Dies sind jedoch nicht die abschließenden Überprüfungen.

Die folgenden Überprüfungen werden während der Befehlsausführung durchgeführt.

Fehlertyp	Merker, der im Fehlerfall auf EIN gesetzt wird	Betrieb anhalten/fortsetzen
1. Befehlsverarbeitungs-Fehler	P_ER-Merker Der Befehlsverarbeitungs-Fehlermerker (A29508) wird ebenfalls auf EIN gesetzt, wenn „Betrieb anhalten“ beim Auftreten eines Fehlers festgelegt wurde.	Eine Einstellung im SPS-Setup wird verwendet, um festzulegen, ob der Betrieb bei Befehlsverarbeitungsfehlern fortgesetzt oder angehalten wird. In der Standardeinstellung wird der Betrieb fortgesetzt. Ein Programmfehler tritt auf, und der Betrieb wird nur dann unterbrochen, wenn eine Betriebsunterbrechung festgelegt wurde.
2. Zugriffsfehler	P_AER-Merker Der Zugriffsfehlermerker (A29510) wird auf EIN gesetzt, wenn die Betriebsunterbrechung beim Auftreten eines Fehlers festgelegt wurde.	Eine Einstellung im SPS-Setup wird verwendet, um festzulegen, ob der Betrieb bei Befehlsverarbeitungsfehlern fortgesetzt oder angehalten wird. In der Standardeinstellung wird der Betrieb fortgesetzt. Ein Programmfehler tritt auf, und der Betrieb wird nur dann unterbrochen, wenn eine Betriebsunterbrechung festgelegt wurde.
3. Unzulässiger-Befehl-Fehler	Unzulässiger-Befehl-Fehlermerker (A29514)	Schwerwiegender Programmfehler
4. UM-Überlauffehler (Anwenderprogrammspeicher)	Anwenderprogrammspeicher-Überlauffehlermerker (A29515)	Schwerwiegender Programmfehler

Befehlsverarbeitungsfehler

Ein Befehlsverarbeitungsfehler tritt auf, wenn während der Befehlsausführung ungültige Daten zur Verfügung gestellt werden oder der Versuch unternommen wurde, einen Befehl außerhalb einer Task auszuführen. In diesem Fall werden die zu Beginn der Befehlsverarbeitung notwendigen Daten überprüft, als Ergebnis wurde der Befehl nicht ausgeführt und der P_ER-Merker auf EIN gesetzt. Abhängig vom Befehl werden die P_EQ- und P_N-Merker beibehalten oder auf AUS gesetzt.

Der P_ER-Merker (Fehlermerker) wird auf AUS gesetzt, wenn der Befehl (ausgenommen Eingangsbefehle) normal ausgeführt wurde. Die Bedingungen zur Steuerung des P_ER-Merkers unterscheiden sich bei den einzelnen Befehlen. Ausführliche Informationen zu den einzelnen Befehlen finden Sie im *CS/CJ-Serie Speicherprogrammierbare Steuerungen Befehlsreferenzhandbuch (W340)*.

Falls es im SPS-Setup entsprechend spezifiziert ist, wird der Betrieb beim Auftreten eines Befehlsverarbeitungsfehlers angehalten (schwerwiegender Fehler) und der Befehlsverarbeitungs-Fehlermerker (A29508) sowie der P_ER-Merker auf EIN gesetzt.

Fehler wegen unzulässigem Zugriff

Fehler wegen unzulässigem Zugriff zeigen an, dass der Zugriff auf falsche Bereiche auf eine der folgenden Arten erfolgte, während auf die den Befehlsoperanden spezifizierende Adresse zugegriffen wurde.

- Es wurde ein Schreib- oder Lesevorgang für einen Parameterbereich ausgeführt.
- Es wurde ein Schreibvorgang in einen Speicherbereich ausgeführt, der nicht vorhanden ist (siehe Hinweis).
- Es wurde ein Schreibvorgang in einen EM-Bereich ausgeführt, der als EM-Datenspeicher definiert ist.
- Es wurde ein Schreibvorgang in einen schreibgeschützten Bereich ausgeführt.
- Der in einer indirekten DM/EM-Adresse im BCD-Modus spezifizierte Wert war nicht in BCD (z. B. *D000001 enthält #A000).

Wenn ein Zugriffsfehler auftritt, wird die Befehlsverarbeitung fortgesetzt und der P_ER-Merker (Fehlermerker) nicht auf EIN gesetzt, der Zugriffsfehlermerker (P_AER-Merker) wird jedoch auf EIN gesetzt.

Hinweis

Ein Zugriffsfehler tritt in den folgenden Fällen auf:

- Wenn eine spezifizierte EM-Adresse 32767 der aktuellen Bank überschreitet (beispielsweise E32768).
- Die letzte Bank (beispielsweise C) wurde für eine indirekte EM-Adresse im Binär-Modus spezifiziert, und das spezifizierte Wort enthält 8000 bis FFFF Hex (beispielsweise @EC_00001 enthält #8000).
- Die aktuelle Bank (beispielsweise C) wurde für eine indirekte EM-Adresse im Binär-Modus spezifiziert, und die spezifizierten Worte enthalten 8000 bis FFFF Hex (beispielsweise @EC_00001 enthält #8000).
- Ein IR-Register, das die interne Speicheradresse eines Bits enthält, wird als Wortadresse verwendet, oder ein IR-Register, das die interne Speicheradresse eines Worts enthält, wird als Bit-Adresse verwendet.

Wenn die Befehlsfehler im SPS-Setup entsprechend definiert sind, wird der Betrieb beim Auftreten eines Zugriffsfehlers angehalten (schwerwiegender Fehler) und der Fehlermerker für unzulässigen Zugriff (A29510) sowie der P_AER-Merker auf EIN gesetzt.

Hinweis

Der Zugriffsfehlermerker (P_AER-Merker) wird nach der Taskausführung nicht zurückgesetzt. Wenn die Befehlsfehler auf ein Fortführen des Betriebs eingestellt sind, kann dieser Merker bis kurz vor den Befehl END(001) überwacht werden, um festzustellen, ob ein Fehler wegen unzulässigem Zugriff im Taskprogramm aufgetreten ist. (Der Status des letzten P_AER-Merkers nach der vollständigen Ausführung des Anwenderprogramms wird überwacht, wenn der P_AER-Merker auf einer Programmierkonsole angezeigt wird.)

Sonstige Fehler

Unzulässiger-Befehl-Fehler

Unzulässiger-Befehl-Fehler weisen auf Versuche hin, nicht auf dem System definierte Befehlsdaten auszuführen. Der Fehler wird gewöhnlich nicht auftreten, wenn das Programm mit einer Programmier-Software für die CS/CJ-Serie erstellt wurde (oder Programmierkonsole).

Sollte dieser Fehler dennoch auftreten wird er als Programmfehler behandelt, der Betrieb wird angehalten (schwerwiegender Fehler) und der Unzulässiger-Befehl-Fehlermerker (A29514) auf EIN gesetzt.

UM-Überlauf-Fehler (Anwenderprogrammspeicher)

UM-Überlauffehler deuten auf den Versuch hin, Befehlsdaten auszuführen, die jenseits der höchsten Adresse im definierten Anwenderprogrammspeicher (UM) liegen. Der Fehler wird gewöhnlich nicht auftreten, wenn das Programm mit einer Programmier-Software für die CS/CJ-Serie erstellt wurde (oder Programmierkonsole).

Tritt der Fehler dennoch auf, wird er als Programmfehler behandelt, der Betrieb wird unterbrochen (schwerwiegender Fehler) und der Anwenderprogrammspeicher-Überlauffehlermerker (A29515) wird auf EIN gesetzt.

2-3-4 Überprüfen schwerwiegender Fehler

Im Folgenden werden schwerwiegende Fehler aufgeführt, nach deren Auftreten die CPU-Baugruppe angehalten wird. Wenn der Betrieb aufgrund eines Programmfehlers angehalten wird, wird die Nummer der Task, in der der Fehler aufgetreten ist, in A294 und die Programmadresse in A298/A299 gespeichert. Die Ursache des Programmfehlers kann anhand dieser Informationen ermittelt werden.

Adresse	Beschreibung	Gespeicherte Daten
A294	Der Typ und die Nummer der Task, bei der der Betrieb unterbrochen wurde, werden hier gespeichert, wenn der Betrieb aufgrund eines Programmfehlers angehalten wird. Sind keine aktiven zyklischen Tasks in einem Zyklus vorhanden, d. h., wenn keine auszuführenden zyklischen Tasks vorhanden sind, wird FFFF Hex gespeichert.	Zyklische Task: 0000 bis 001F Hex (zyklische Tasks 0 bis 31) Interrupt-Task: 8000 bis 80FF Hex (Interrupt-Tasks 0 bis 255)
A298/A299	in dieser Adresse wird die Programmadresse, bei der der Betrieb angehalten wurde, in Binär gespeichert, wenn ein Programmfehler die Ursache für das Anhalten des Betriebs war. Wenn kein END(001)-Befehl vorhanden ist (A29511 ist auf EIN gesetzt), wird die Adresse gespeichert, an der der END(001)-Befehl erwartet wurde. Tritt ein Taskausführungsfehler auf (A29512 ist auf EIN gesetzt), wird FFFFFFFF Hex in A298 und A299 gespeichert.	A298: Der rechte Teil der Programmadresse A299: Der linke Teil der Programmadresse

Hinweis

Wird der Fehler- oder Zugriffsfehler-Merker auf EIN gesetzt, wird dies als Programmfehler behandelt und kann zum Anhalten der CPU verwendet werden. Bestimmen Sie den Betrieb (Fortsetzen/Abbruch) beim Auftreten von Programmfehlern im SPS-Setup.

Programmfehler	Beschreibung	Entsprechende Merker
Kein END-Befehl	Im Programm ist kein END-Befehl vorhanden.	Der END-Befehl-fehlt-Merker (A29511) wird auf EIN gesetzt.
Fehler während der Taskausführung	Im Zyklus ist keine ausführungsbereite Task vorhanden. Einer Task wurde kein Programm zugeordnet. Die entsprechende Interrupt-Tasknummer ist nicht vorhanden, obwohl die Ausführungsbedingungen für die Interrupt-Task erfüllt sind.	Der Task-Fehlermerker (29512) wird auf EIN gesetzt.
Befehlsverarbeitungs-Fehler (P_ER-Merker auf EIN gesetzt) und im SPS-Setup definierte Betriebsunterbrechung bei Befehlsfehlern	Im Operanden wurden bei einer Versuch der Befehlsausführung falsche Datenwerte bereitgestellt.	Wenn für Befehlsfehler im SPS-Setup eine Betriebsunterbrechung festgelegt wurde, werden der P_ER -Merker und der Befehlsverarbeitungs-Fehlermerker (A29508) auf EIN gesetzt.
Unzulässiger-Zugriff-Fehler (P_AER-Merker auf EIN gesetzt) und im SPS-Setup festgelegte Betriebsunterbrechung bei Befehlsfehlern	Es wurde ein Schreib- oder Lesevorgang für einen Parameterbereich ausgeführt. Es wurde ein Schreibvorgang in einen Speicherbereich ausgeführt, der nicht vorhanden ist (siehe Hinweis). Es wurde ein Schreibvorgang in einen EM-Bereich ausgeführt, der als EM-Dateispeicher definiert ist. Es wurde ein Schreibvorgang in einen schreibgeschützten Bereich ausgeführt. Der in einer indirekten DM/EM-Adresse im BCD-Modus spezifizierte Wert war nicht in BCD.	Wenn für Befehlsfehler im SPS-Setup eine Betriebsunterbrechung festgelegt wurde, werden der P_AER-Merker und der Unzulässiger-Zugriff-Fehlermerker (A29510) auf EIN gesetzt.
BCD-Fehler-bei-indirekter-DM/EM-Adressierung und im SPS-Setup festgelegte Betriebsunterbrechung bei Befehlsfehlern	Der in einer indirekten DM/EM-Adresse im BCD-Modus spezifizierte Wert ist nicht in BCD.	Wenn für Befehlsfehler im SPS-Setup eine Betriebsunterbrechung festgelegt wurde, werden der P_AER-Merker und der BCD-Fehler-bei-indirekter-DM/EM-Adressierung-Merker (A29509) auf EIN gesetzt.
Flankenauswertungs-Adressenüberlauf-Fehler	Bei der Online-Bearbeitung wurden mehr als 131.071 Befehle mit Flankenauswertung eingefügt oder gelöscht.	Der Flankenauswertungs-Überlauf-Fehlermerker (A29513) wird auf EIN gesetzt.
UM-Überlauf-Fehler (Anwenderprogrammspeicher)	Es wurde versucht, Befehlsdaten auszuführen, die jenseits der höchsten Adresse im Anwenderspeicher liegen, die als Programmspeicherbereich definiert ist.	Der Anwenderprogrammspeicher-Überlauffehlermerker (A29515) wird auf EIN gesetzt.
Unzulässiger-Befehl-Fehler	Es wurde versucht, einen nicht ausführbaren Befehl auszuführen.	Der Unzulässiger-Befehl-Merker (A29514) wird auf EIN gesetzt.

ABSCHNITT 3

Befehlsfunktionen

Dieser Abschnitt enthält eine Übersicht über die Befehle, die beim Schreiben von Anwenderprogrammen verwendet werden können.

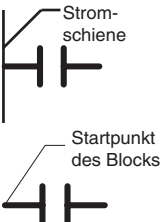
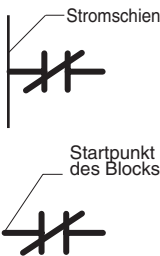
3-1	Eingabebefehle	72
3-2	Ausgabebefehle	74
3-3	Sequenzsteuerbefehle	77
3-4	Zeitgeber- und Zählerbefehle	81
3-5	Vergleichsbefehle	85
3-6	Kopierbefehle	89
3-7	Datenverschiebepbefehle	92
3-8	Inkrementier- und Dekrementierbefehle	96
3-9	Arithmetische Befehle	97
3-10	Konvertierungsbefehle	102
3-11	Boolesche Befehle (Logikbefehle)	108
3-12	Spezielle arithmetische Befehle	110
3-13	Gleitkomma-Arithmetikbefehle	111
3-14	Gleitkomma-Arithmetikbefehle mit doppelter Präzision	115
3-15	Tabellendaten-Verarbeitungsbefehle	119
3-16	Datensteuerungsbefehle	123
3-17	Unterprogrammbefehle	127
3-18	Interrupt-Steuerungsbefehle	128
3-19	Befehle für schnelle Zähler und Impulsausgaben (nur CJ1M-CPU21/22/23)	130
3-20	Step-Befehle	132
3-21	Befehle für E/A-Baugruppen	132
3-22	Befehle für serielle Kommunikation	135
3-23	Netzwerkbefehle	136
3-24	Dateispeicherbefehle	139
3-25	Anzeigebefehle	140
3-26	Datum- und Uhrzeitbefehle	140
3-27	Debugging-Befehle	141
3-28	Fehlerdiagnosebefehle	142
3-29	Sonstige Befehle	143
3-30	Blockprogrammierungsbefehle	144
3-31	Textzeichenketten-Verarbeitungsbefehle	150
3-32	Task-Steuerbefehle	153
3-33	Modell-Konvertierungsbefehle (ab CPU-Baugruppen Ver. 3.0)	154
3-34	Spezielle Funktionsblockbefehle	155

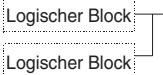
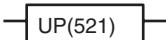
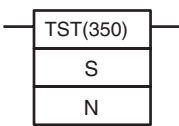
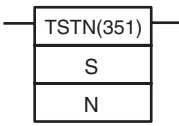
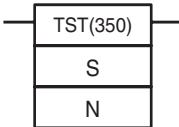
3-1 Eingabebefehle

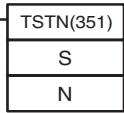
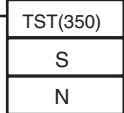
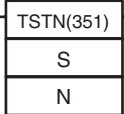
*1: Nicht unterstützt von CS1D CPU-Baugruppen für Duplex-CPU-Systeme.

*2: Nur unterstützt von CS1-H, CJ1-H, CJ1M und CS1D CPU-Baugruppen.

*3: Nur unterstützt durch CS1-H, CJ1-H und CJ1M CPU-Baugruppen.

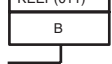
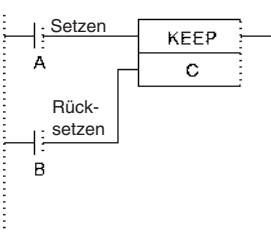
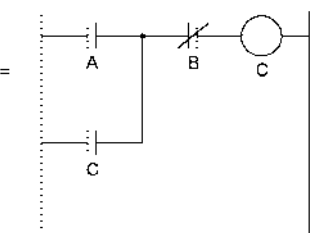
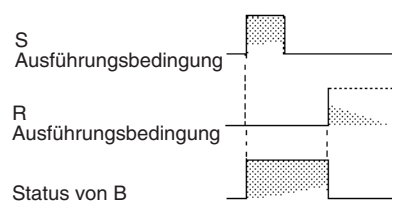
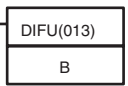
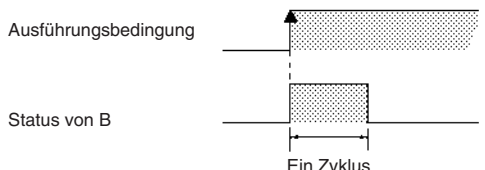
Befehl AWL Code	Symbol/Operand	Funktion	Position Ausführungs- bedingung
LOAD (Laden) LD @LD %LD !LD* ¹ !@LD* ¹ !%LD* ¹		Bildet den Anfang der logischen Verknüpfung und erzeugt eine EIN/AUS-Ausführungsbedingung auf der Grundlage des EIN/AUS-Status des spezifizierten Operanden-Bits.	Anfang der logischen Verknüpfung Nicht erforderlich
LOAD NOT (Laden Nicht) LD NOT @LD NOT* ² %LD NOT* ² !LD NOT* ¹ !@LD NOT* ³ !%LD NOT* ³		Bildet den Anfang der logischen Verknüpfung und erzeugt eine EIN/AUS-Ausführungsbedingung auf der Grundlage des invertierten EIN/AUS-Status des spezifizierten Operanden-Bits.	Anfang der logischen Verknüpfung Nicht erforderlich
AND (Und) AND @AND %AND !AND* ¹ !@AND* ¹ !%AND* ¹		Bildet eine logische UND-Verknüpfung zwischen dem Status des spezifizierten Operanden-Bits und der aktuellen Ausführungsbedingung.	Fortsetzung der logischen Verknüpfung Erforderlich
AND NOT (Und Nicht) AND NOT @AND NOT* ² %AND NOT* ² !AND NOT* ¹ !@AND NOT* ³ !%AND NOT* ³		Kehrt den Status des spezifizierten Operanden-Bits um und bildet ein logisches UND mit der aktuellen Ausführungsbedingung.	Fortsetzung der logischen Verknüpfung Erforderlich
OR (Oder) OR @OR %OR !OR* ¹ !@OR* ¹ !%OR* ¹		Bildet eine logische ODER-Verknüpfung zwischen dem EIN/AUS-Status des spezifizierten Operanden-Bits und der aktuellen Ausführungsbedingung.	Fortsetzung der logischen Verknüpfung Erforderlich
OR NOT (Oder Nicht) OR NOT @OR NOT* ² %OR NOT* ² !OR NOT* ¹ !@OR NOT* ³ !%OR NOT* ³		Kehrt den Status des spezifizierten Operanden-Bits um und bildet ein logisches ODER mit der aktuellen Ausführungsbedingung.	Fortsetzung der logischen Verknüpfung Erforderlich

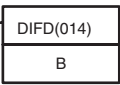
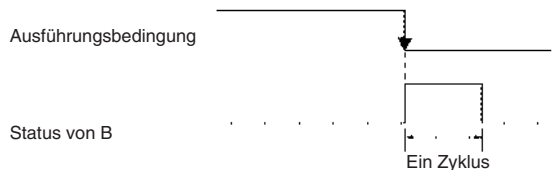
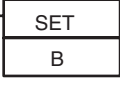
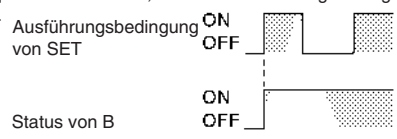
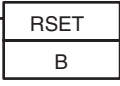
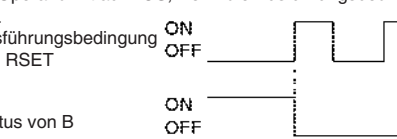
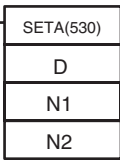
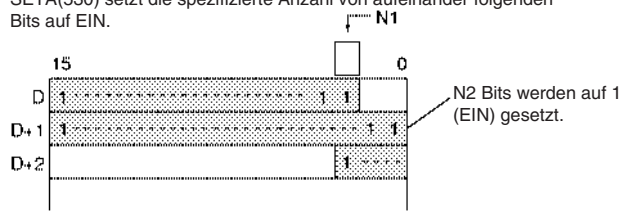
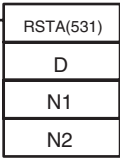
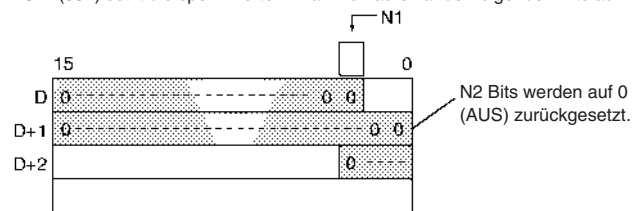
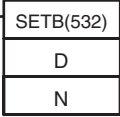
Befehl	AWL Code	Symbol/Operand	Funktion	Position Ausführungsbedingung
AND LOAD (Und Laden) AND LD			Führt eine logische AND-Bedingung zwischen logischen Blöcken ein. LD bis LD bis } } Logischer Block A Logischer Block B AND LD Reihenschaltung zwischen logischem Block A und logischem Block B	Fortsetzung der logischen Verknüpfung Erforderlich
OR LOAD (Oder Laden) OR LD			LD bis LD bis } } Logischer Block A Logischer Block B OR LD Parallelschaltung zwischen logischem Block A und logischem Block B.	Fortsetzung der logischen Verknüpfung Erforderlich
NOT NOT (Nicht) 520		---	Kehrt die Ausführungsbedingung um.	Fortsetzung der logischen Verknüpfung Erforderlich
CONDITION ON (Bedingung steigende Flanke) UP 521			UP(521) setzt die Ausführungsbedingung für einen Zyklus auf EIN, wenn die Ausführungsbedingung von AUS zu EIN wechselt.	Fortsetzung der logischen Verknüpfung Erforderlich
CONDITION OFF (Bedingung fallende Flanke) DOWN 522			DOWN(522) setzt die Ausführungsbedingung für einen Zyklus auf EIN, wenn die Ausführungsbedingung von EIN auf AUS wechselt.	Fortsetzung der logischen Verknüpfung Erforderlich
BIT TEST LD TST 350		 S: Quellwort N: Bit-Nummer	LD TST(350), AND TST(350) sowie OR TST(350) werden im Programm wie LD, AND und OR verwendet. Die Ausführungsbedingung ist EIN, wenn das spezifizierte Bit im spezifizierten Wort den Status EIN hat. Entsprechend ist sie AUS, wenn das Bit den Status AUS hat.	Fortsetzung der logischen Verknüpfung Nicht erforderlich
BIT TEST NICHT LD TSTN 351		 S: Quellwort N: Bit-Nummer	LD TSTN(351), AND TSTN(351) sowie OR TSTN(351) werden im Programm wie LD, AND und OR verwendet. Die Ausführungsbedingung ist AUS, wenn das spezifizierte Bit im spezifizierten Wort den Zustand EIN hat. Entsprechend ist sie EIN, wenn das Bit den Zustand AUS hat.	Fortsetzung der logischen Verknüpfung Nicht erforderlich
BIT TEST AND TST 350		 S: Quellwort N: Bit-Nummer	LD TST(350), AND TST(350) sowie OR TST(350) werden im Programm wie LD, AND und OR verwendet. Die Ausführungsbedingung ist EIN, wenn das spezifizierte Bit im spezifizierten Wort den Status EIN hat. Entsprechend ist sie AUS, wenn das Bit den Status AUS hat.	Fortsetzung der logischen Verknüpfung Erforderlich

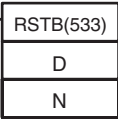
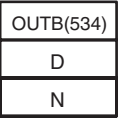
Befehl AWL Code	Symbol/Operand	Funktion	Position Ausführungs- bedingung
BIT TEST NICHT AND TSTN 351	 S: Quellwort N: Bit-Nummer	LD TSTN(351), AND TSTN(351) sowie OR TSTN(351) werden im Programm wie LD NOT, AND NOT und OR NOT verwendet. Die Ausführungsbedingung ist EIN, wenn das spezifizierte Bit im spezifizierten Wort den Zustand EIN hat. Entsprechend ist sie EIN, wenn das Bit den Zustand AUS hat.	Fortsetzung der logischen Verknüpfung Erforderlich
BIT TEST OR TST 350	 S: Quellwort N: Bit-Nummer	LD TST(350), AND TST(350) sowie OR TST(350) werden im Programm wie LD, AND und OR verwendet. Die Ausführungsbedingung ist EIN, wenn das spezifizierte Bit im spezifizierten Wort den Status EIN hat. Entsprechend ist sie AUS, wenn das Bit den Status AUS hat.	Fortsetzung der logischen Verknüpfung Erforderlich
BIT TEST NICHT OR TSTN 351	 S: Quellwort N: Bit-Nummer	LD TSTN(351), AND TSTN(351) sowie OR TSTN(351) werden im Programm wie LD NOT, AND NOT und OR NOT verwendet. Die Ausführungsbedingung ist EIN, wenn das spezifizierte Bit im spezifizierten Wort den Zustand EIN hat. Entsprechend ist sie EIN, wenn das Bit den Zustand AUS hat.	Fortsetzung der logischen Verknüpfung Erforderlich

3-2 Ausgabebefehle

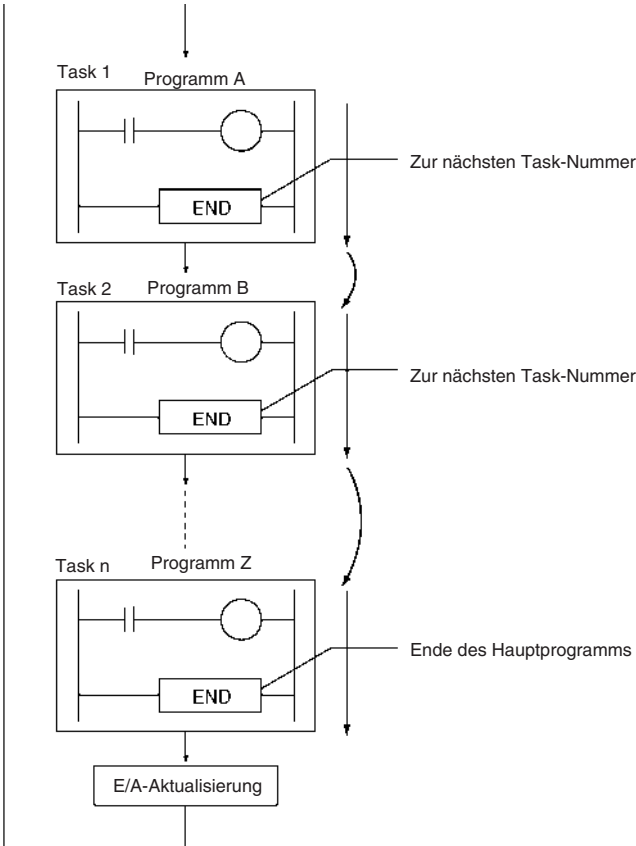
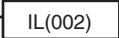
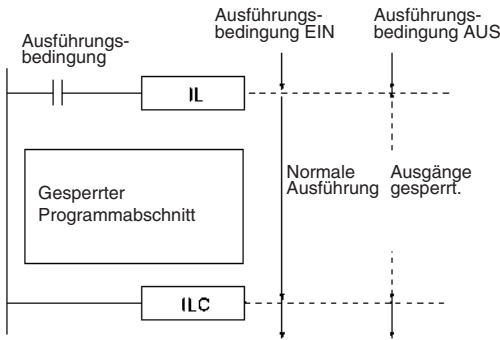
*1: Nicht unterstützt von CS1D CPU-Baugruppen für Duplex-CPU-Systeme.

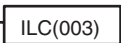
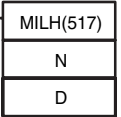
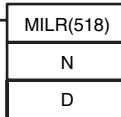
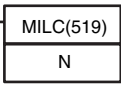
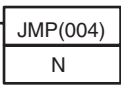
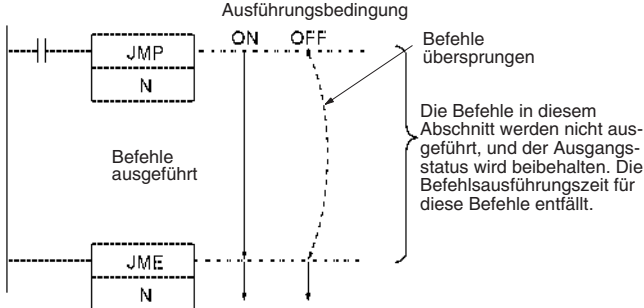
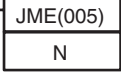
Befehl AWL Code	Symbol/Operand	Funktion	Position Ausführungs- bedingung
OUTPUT (Ausgang) OUT !OUT* ¹		Gibt das Ergebnis (Ausführungsbedingung) der logischen Verarbeitung an das spezifizierte Bit aus.	Ausgabe Erforderlich
OUTPUT NOT (Ausgang Nicht) OUT NOT !OUT NOT* ¹		Kehrt das Ergebnis (Ausführungsbedingung) der logischen Verarbeitung um und gibt es an das spezifizierte Bit aus.	Ausgabe Erforderlich
KEEP (Halten) KEEP !KEEP* ¹ 011	S (Setzen)  R (Zurücksetzen) B: Bit	Arbeitet als Halterelais.  =  	Ausgabe Erforderlich
DIFFERENTIATE UP (Ausführung bei steigender Flanke) DIFU !DIFU* ¹ 013	 B: Bit	DIFU(013) setzt das angegebene Bit für einen Zyklus auf EIN, wenn die Ausführungsbedingung von AUS zu EIN wechselt (steigende Flanke).  Ein Zyklus	Ausgabe Erforderlich

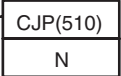
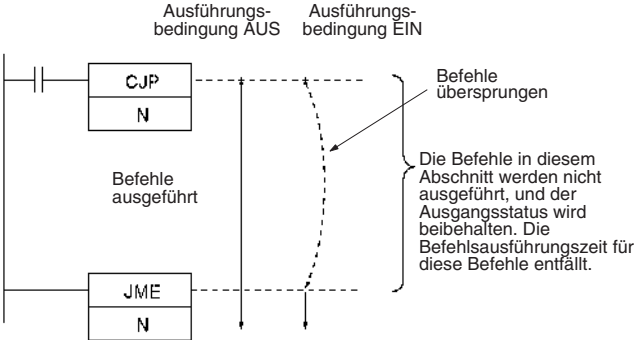
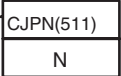
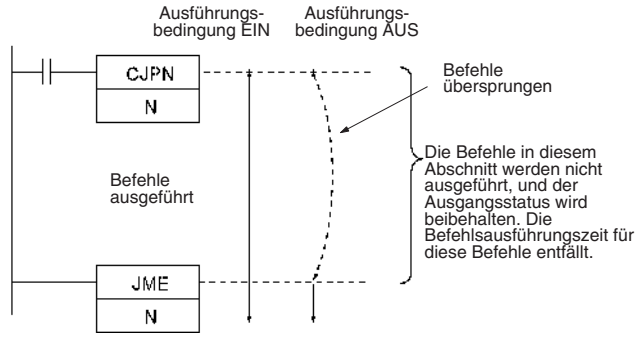
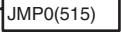
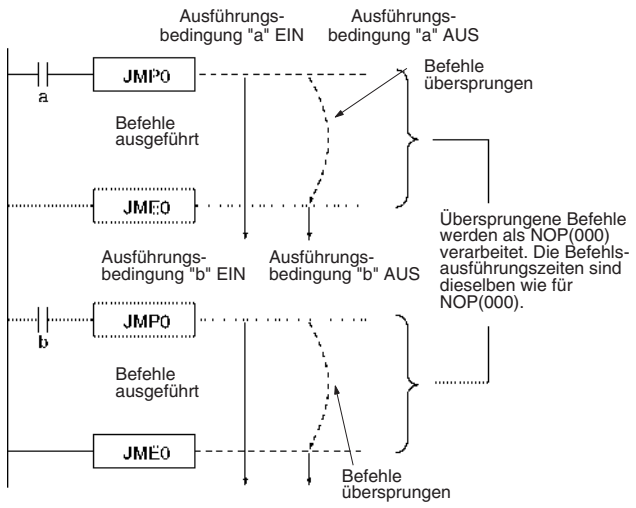
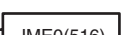
Befehl	Symbol/Operand	Funktion	Position Ausführungsbedingung
DIFFERENTIATE DOWN (Ausführung bei fallender Flanke) DIFD !DIFD ⁺¹ 014	 B: Bit	DIFD(014) setzt das angegebene Bit für einen Zyklus auf EIN, wenn die Ausführungsbedingung von EIN zu AUS wechselt (fallende Flanke). 	Ausgabe Erforderlich
SET (Setzen) SET @SET %SET !SET ⁺¹ !@SET ⁺¹ !%SET ⁺¹	 B: Bit	SET setzt das Operand-Bit auf EIN, wenn die Ausführungsbedingung den Zustand EIN hat. 	Ausgabe Erforderlich
RESET (Rücksetzen) RSET @RSET %RSET !RSET ⁺¹ !@RSET ⁺¹ !%RSET ⁺¹	 B: Bit	RSET setzt das Operand-Bit auf AUS, wenn die Ausführungsbedingung den Zustand EIN hat. 	Ausgabe Erforderlich
MULTIPLE BIT SET (Mehrfaches Bit-Setzen) SETA @SETA 530	 D: Anfangswort N1: Anfangsbit N2: Anzahl der Bits	SETA(530) setzt die spezifizierte Anzahl von aufeinander folgenden Bits auf EIN. 	Ausgabe Erforderlich
MULTIPLE BIT RESET (Mehrfaches Bit-Rücksetzen) RSTA @RSTA 531	 D: Anfangswort N1: Anfangsbit N2: Anzahl der Bits	RSTA(531) setzt die spezifizierte Anzahl von aufeinander folgenden Bits auf AUS. 	Ausgabe Erforderlich
SINGLE BIT SET (Bit-Setzen) (nur CS1-H, CJ1-H, CJ1M und CS1D) SETB @SETB !SETB ⁺¹ !@SETB ⁺¹	 D: Wort-Adresse N: Bit-Nummer	SETB(532) setzt das spezifizierte Bit im spezifizierten Wort auf EIN, wenn die Ausführungsbedingung den Zustand EIN hat. Anders als der SET-Befehl kann SETB(532) zum Setzen eines Bits in einem DM- oder EM-Wort verwendet werden.	Ausgabe Erforderlich

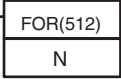
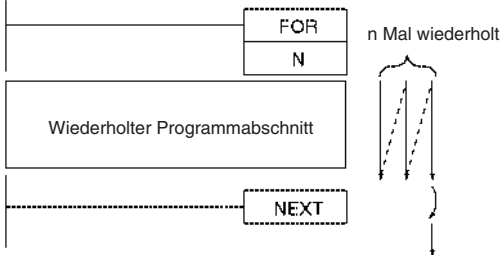
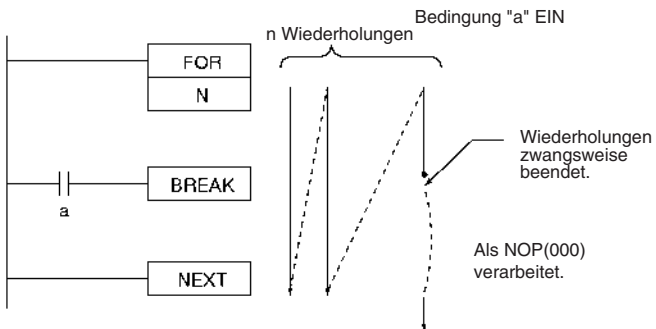
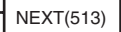
Befehl	Symbol/Operand	Funktion	Position Ausführungsbedingung
SINGLE BIT RESET (Bit-Rücksetzen) (nur CS1-H, CJ1-H, CJ1M oder CS1D) RSTB @RSTB !RSTB ^{*1} !@RSTB ^{*1}	 D: Wort-Adresse N: Bit-Nummer	RSTB(533) setzt das spezifizierte Bit im spezifizierten Wort auf AUS, wenn die Ausführungsbedingung den Zustand EIN hat. Anders als der RSET-Befehl kann RSTB(533) zum Rücksetzen eines Bits in einem DM- oder EM-Wort verwendet werden.	Ausgabe Erforderlich
SINGLE BIT OUT-PUT (Bit-Ausgabe) (nur CS1-H, CJ1-H, CJ1M oder CS1D) OUTB @OUTB !OUTB ^{*1}	 D: Wort-Adresse N: Bit-Nummer	OUTB(534) gibt das Ergebnis (Ausführungsbedingung) der logischen Verarbeitung an das spezifizierte Bit aus. Anders als der OUT-Befehl kann OUTB(534) zur Steuerung eines Bits in einem DM- oder EM-Wort verwendet werden.	Ausgabe Erforderlich

3-3 Sequenzsteuerbefehle

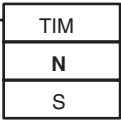
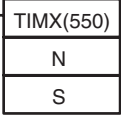
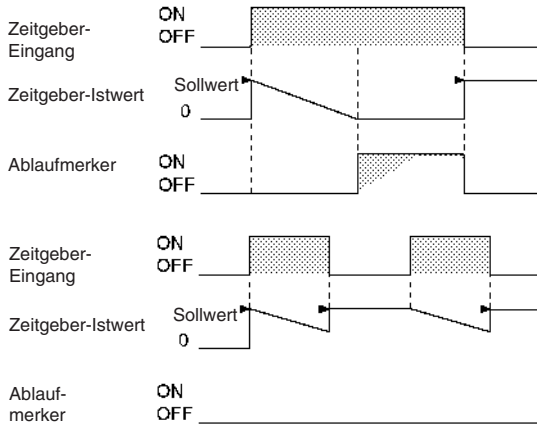
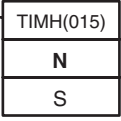
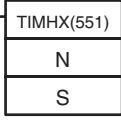
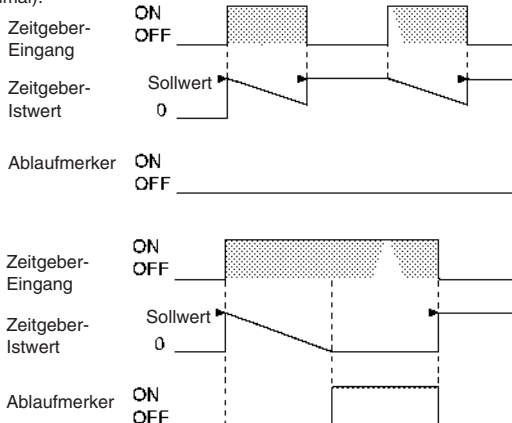
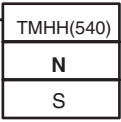
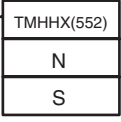
Befehl	AWL Code	Symbol/Operand	Funktion	Position Ausführungsbedingung
END	END 001		Kennzeichnet das Ende eines Programms. END(001) beendet die Ausführung eines Programms für diesen Zyklus. Nach END(001) geschriebene Befehle werden nicht ausgeführt. Die Ausführung fährt mit dem Programm mit der nächsten Task-Nummer fort. Wenn das aktuell ausgeführte Programm die höchste Task-Nummer im Programm hat, markiert END(001) das Ende des gesamten Hauptprogramms. 	Ausgabe Nicht erforderlich
NO OPERATION (Kein Vorgang) NOP 000			Dieser Befehl hat keine Funktion. (Durch NOP(000) erfolgt keine Verarbeitung.)	Ausgabe Nicht erforderlich
INTERLOCK (Verriegelung) IL 002			Sperrt alle Ausgänge zwischen IL(002) und ILC(003), wenn die Ausführungsbedingung für IL(002) den Zustand AUS hat. IL(002) und ILC(003) werden normalerweise als Paar verwendet. 	Ausgabe Erforderlich

Befehl AWL Code	Symbol/Operand	Funktion	Position Ausführungs- bedingung
INTERLOCK CLEAR (Verriegelung löschen) ILC 003		Alle Ausgaben zwischen IL(002) und ILC(003) sind gesperrt, wenn die Ausführungsbedingung für IL(002) den Zustand AUS hat. ILC(002) und ILC(003) werden normalerweise als Paar verwendet.	Ausgabe Nicht erforderlich
MULTI-INTER- LOCK DIFFE- RENTIATION HOLD (Mehrfach- verriegelung, Merker für Flanke halten) MILH 517 Nur CS/CJ-Serie CPU-Baugruppen Version 2.0 oder neuer.	 N: Verriegelungsnummer D: Verriegelungsstatus-Bit	Wenn die Ausführungsbedingung für MILH(517) den Zustand AUS hat, werden die Ausgaben für alle Befehle zwischen diesem MILH(517)-Befehl und dem nächsten MILC(519)-Befehl gesperrt. MILH(517) und MILC(519) werden als Paar verwendet. Die MILH(517)/MILC(519)-Sperrungen können geschachtelt werden (z. B. MILH(517) – MILH(517) – MILC(519) – MILC(519)). Bei einem Befehl mit Flankenauswertung (DIFU, DIFD bzw. ein Befehl mit einem Präfix @ oder %) zwischen MILH(517) und dem entsprechenden MILC(519)-Befehl wird dieser Befehl nach Aufhebung der Sperre ausgeführt, wenn die Flankenauswertungsbedingung des Befehls eingetreten war und blieb.	Ausgabe Erforderlich
MULTI-INTER- LOCK DIFFE- RENTIATION RELEASE (Mehrfachverrie- gelung, Merker für Flanke nicht beibehalten) MILR 518 Nur CS/CJ-Serie CPU-Baugruppen Version 2.0 oder neuer.	 N: Verriegelungsnummer D: Verriegelungsstatus-Bit	Wenn die Ausführungsbedingung für MILR(518) den Zustand AUS hat, werden die Ausgaben für alle Befehle zwischen diesem MILR(518)-Befehl und dem nächsten MILC(519)-Befehl gesperrt. MILR(518) und MILC(519) werden als Paar verwendet. Die MILR(518)/MILC(519)-Sperrungen können geschachtelt werden (z. B. MILR(518) – MILR(518) – MILC(519) – MILC(519)). Bei einem Befehl mit Flankenauswertung (DIFU, DIFD bzw. ein Befehl mit einem Präfix @ oder %) zwischen MILR(518) und dem entsprechenden MILC(519)-Befehl wird dieser Befehl nach Aufhebung der Sperre nicht ausgeführt, auch wenn die Flankenauswertungsbedingung des Befehls eingetreten war.	Ausgabe Erforderlich
MULTI-INTER- LOCK CLEAR (Mehrfachverrie- gelung löschen) MILC 519 Nur CS/CJ-Serie CPU-Baugruppen Version 2.0 oder neuer.	 N: Sperrnummer	Hebt eine Sperre auf, die durch einen MILH(517)- oder MILR(518)-Befehl mit der gleichen Sperrnummer gestartet wurde. Alle Ausgaben zwischen MILH(517)/MILR(518) und dem entsprechenden MILC(519)-Befehl mit der gleichen Sperrnummer werden gesperrt, wenn die Ausführungsbedingung für MILH(517)/MILR(518) den Zustand AUS hat.	Ausgabe Nicht erforderlich
JUMP (Sprung) JMP 004	 N: Sprung-Nummer	Wenn die Ausführungsbedingung für JMP(004) den Zustand AUS hat, springt die Programmausführung direkt zum ersten JME(005) im Programm mit derselben Sprung-Nummer. JMP(004) und JME(005) werden als Paar verwendet. 	Ausgabe Erforderlich
JUMP END (Sprungende) JME 005	 N: Sprung-Nummer	Zeigt das Ende eines durch JMP(004) oder CJP(510) eingeleiteten Sprungs an.	Ausgabe Nicht erforderlich

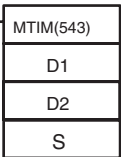
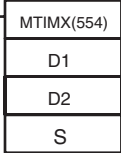
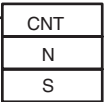
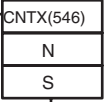
Befehl AWL Code	Symbol/Operand	Funktion	Position Ausführungs- bedingung
CONDITIONAL JUMP (Bedingter Sprung) CJP 510	 N: Sprung-Nummer	Die Funktion von CJP(510) ist im Prinzip die Umkehrung der Funktion von JMP(004). Wenn die Ausführungsbedingung für CJP(510) den Zustand EIN hat, springt die Programmausführung direkt zum ersten JME(005) im Programm mit derselben Sprung-Nummer. CJP(510) und JME(005) werden als Paar verwendet. 	Ausgabe Erforderlich
CONDITIONAL JUMP NOT (Bedingter negierter Sprung) CJPN 511	 N: Sprung-Nummer	Die Funktion von CJPN(511) ist annähernd dieselbe wie von JMP(004). Wenn die Ausführungsbedingung für CJPN(511) den Zustand AUS hat, springt die Programmausführung direkt zum ersten JME(005) im Programm mit derselben Sprung-Nummer. CJPN(511) und JME(005) werden als Paar verwendet. 	Ausgabe Nicht erforderlich
MULTIPLE JUMP (Mehrfachsprung) JMP0 515		Wenn die Ausführungsbedingung für JMP0(515) den Zustand AUS hat, werden alle Befehle zwischen JMP0(515) und dem nächsten JME0(516) im Programm als NOP(000) verarbeitet, also nicht verarbeitet. JMP0(515) und JME0(516) sind als Paare zu verwenden. Es gibt keine Begrenzung der Anzahl von Paaren, die im Programm verwendet werden können. 	Ausgabe Erforderlich
MULTIPLE JUMP END (Mehrfachsprung-Ende) JME0 516		Wenn die Ausführungsbedingung für JMP0(515) den Zustand AUS hat, werden alle Befehle zwischen JMP0(515) und dem nächsten JME0(516) im Programm als NOP(000) verarbeitet, also nicht verarbeitet. JMP0(515) und JME0(516) sind als Paare zu verwenden. Es gibt keine Begrenzung der Anzahl von Paaren, die im Programm verwendet werden können.	Ausgabe Nicht erforderlich

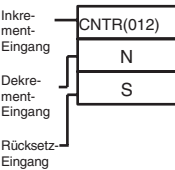
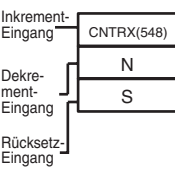
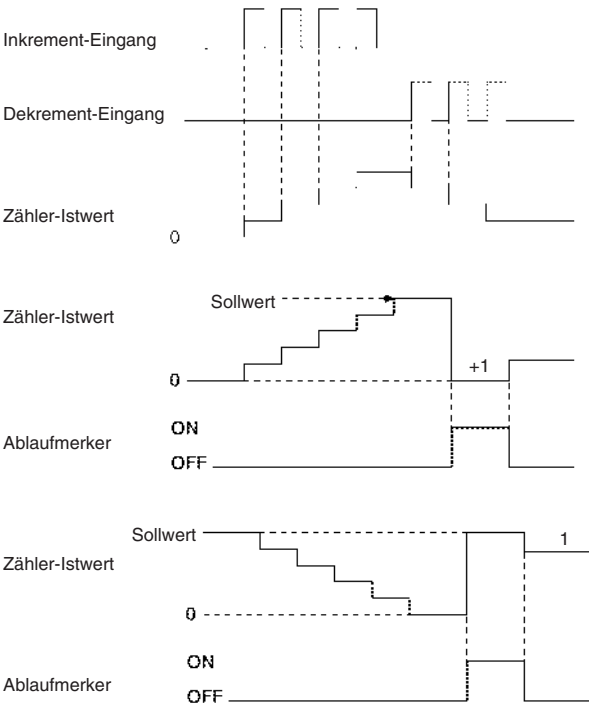
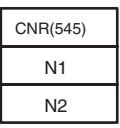
Befehl AWL Code	Symbol/Operand	Funktion	Position Ausführungs- bedingung
FOR-NEXT Programm Schleifen FOR 512	 N: Anzahl der Schleifen	Die Befehle zwischen FOR(512) und NEXT(513) werden die spezifizierte Anzahl von Ausführungen wiederholt. FOR(512) und NEXT(513) werden als Paar verwendet. 	Ausgabe Nicht erforderlich
BREAK LOOP (FOR/NEXT-Schleife vorzeitig verlassen) BREAK 514		In einer FOR-NEXT-Schleife programmiert, um die Schleife bei einer bestimmten Ausführungsbedingung zu unterbrechen. Die verbliebenen Befehle in der Schleife werden als NOP(000)-Befehle verarbeitet. 	Ausgabe Erforderlich
FOR-NEXT Programm Schleifen NEXT 513		Die Befehle zwischen FOR(512) und NEXT(513) werden die spezifizierte Anzahl von Ausführungen wiederholt. FOR(512) und NEXT(513) werden als Paar verwendet.	Ausgabe Nicht erforderlich

3-4 Zeitgeber- und Zählerbefehle

Befehl	Symbol/Operand	Funktion	Position Ausführungs- bedingung
TIMER TIM (BCD) TIMX (Binär) (nur CS1-H, CJ1- H, CJ1M und CS1D)	 N: Zeitgebernummer S: Sollwert  N: Zeitgebernummer S: Sollwert	TIM/TIMX(550) steuert einen dekrementierenden Zeitgeber mit Einheiten von 0,1 s. Der Einstellbereich für den Sollwert reicht von 0 bis 999,9 s bei BCD-Werten und von 0 bis 6.553,5 s bei Binärwerten (dezimal oder hexadezimal). 	Ausgabe Erforderlich
HIGH-SPEED TIMER (Schneller Zeitgeber) TIMH 015 (BCD) TIMHX 551 (Binär) (nur CS1-H, CJ1- H, CJ1M und CS1D)	 N: Zeitgebernummer S: Sollwert  N: Zeitgebernummer S: Sollwert	TIMH(015)/TIMHX(551) funktioniert als dekrementierender Zeitgeber mit 10-ms-Schritten. Der Einstellbereich für den Sollwert reicht von 0 bis 99,99 s bei BCD-Werten und von 0 bis 655,35 s bei Binärwerten (dezimal oder hexadezimal). 	Ausgabe Erforderlich
ONE-MS TIMER (1-ms-Zeitgeber) TMHH 540 (BCD) TMHHX 552 (Binär) (nur CS1-H, CJ1- H, CJ1M und CS1D)	 N: Zeitgebernummer S: Sollwert  N: Zeitgebernummer S: Sollwert	TMHH(540)/TMHHX(552) funktioniert als dekrementierender Zeitgeber mit 1-ms-Schritten. Der Einstellbereich für den Sollwert reicht von 0 bis 9,999 s bei BCD-Werten und von 0 bis 65,535 s bei Binärwerten (dezimal oder hexadezimal). Die Zeitablaufdiagramme für TMHH(540) und TMHHX(552) entsprechen denen, die oben für TIMH(015) dargestellt sind.	Ausgabe Erforderlich

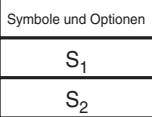
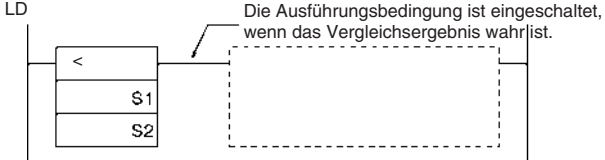
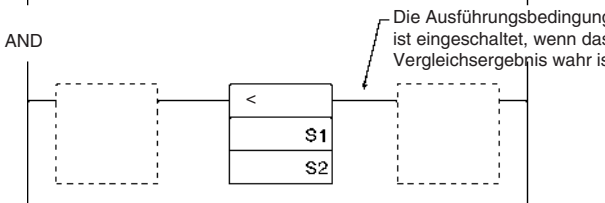
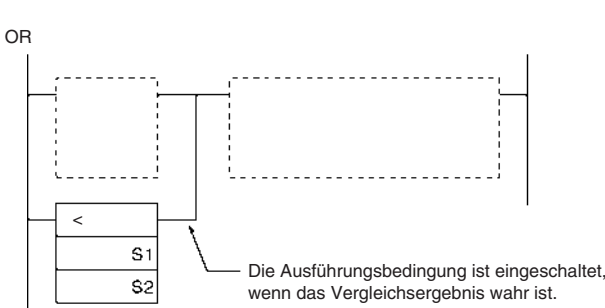
Befehl	Symbol/Operand	Funktion	Position Ausführungs- bedingung
ACCUMULATIVE TIMER (Akkumu- lierender Zeitge- ber) TTIM 087 (BCD) TTIMX 555 (Binär) (nur CS1-H, CJ1- H, CJ1M und CS1D)	Zeit- geber- Eingang TTIM(087) N S Rücksetz- Eingang N: Zeitgebern timer S: Sollwert Zeit- geber- Eingang TTIMX(555) N S Rücksetz- Eingang N: Zeitgebern timer S: Sollwert	TTIM(087)/TTIMX(555) steuert einen inkrementierenden Zeitgeber mit Einheiten von 0,1 s. Der Einstellbereich für den Sollwert reicht von 0 bis 999,9 s bei BCD-Werten und von 0 bis 6.553,5 s bei Binärwerten (dezimal oder hexadezimal). Zeitgeber-Eingang Zeitgeber-Istwert Ablaufmerker Rücksetz-Eingang Sollwert 0 Zeitmessung wird fortgesetzt. Istwert bleibt erhalten.	Ausgabe Erforderlich
LONG TIMER (Langer Zeitgeber) TIML 542 (BCD) TIMLX 553 (Binär) (nur CS1-H, CJ1- H, CJ1M und CS1D)	TIML(542) D1 D2 S D1: Ablaufmerker D2: Istwert-Wort S: Sollwert-Wort TIMLX(553) D1 D2 S D1: Ablaufmerker D2: Istwert-Wort S: Sollwert-Wort	TIML(542)/TIMLX(553) steuert einen dekrementierenden Zeitgeber mit Einheiten von 0,1 s, der eine Zeitmessung bis zu ca. 115 Tagen bei BCD-Werten und bis zu 49.710 Tagen bei Binärwerten (dezimal oder hexadezimal) durchführen kann. Zeitgeber-Eingang Zeitgeber-Istwert Ablaufmerker (Bit 00 von D1) Sollwert 0	Ausgabe Erforderlich

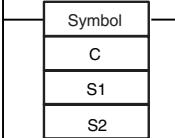
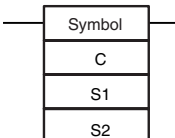
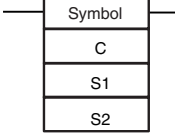
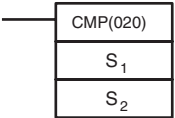
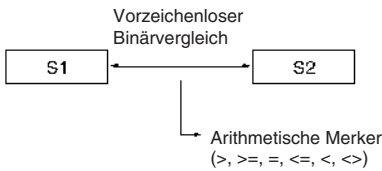
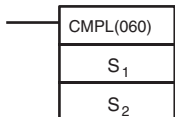
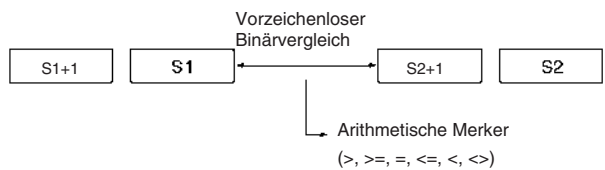
Befehl	Symbol/Operand	Funktion	Position Ausführungs- bedingung
MULTI-OUTPUT TIMER (Multi- Ausgabe-Zeitge- ber) MTIM 543 (BCD) MTIMX 554 (Binär) (nur CS1-H, CJ1- H, CJ1M und CS1D)	 D1: Ablaufmerker D2: Istwert-Wort S: 1. Sollwert-Wort  D1: Ablaufmerker D2: Istwert-Wort S: Erstes Sollwert-Wo	MTIM(543)/MTIMX(554) steuert einen 0,1-s-Inkrementierungs-Zeitgeber mit 8 unabhängigen Sollwerten und Ablaufmerkern. Der Einstellbereich für den Sollwert ist 0 bis 999,9 s bei BCD und 0 bis 6.553,5 s bei Binär (dezimal oder hexadezimal). Zeitgeber-Istwert Zeitgeber-Sollwerte Sollwert 7 bis Sollwert 0 Zeitgeber-Istwert (D2) Ablauf-Merker (D1) Bit 7 bis Bit 0 Zeitgeber-Istwert (D2) vs Sollwert 7 bis Sollwert 0 Ablauf-Merker (D1) vs Sollwert 7 bis Sollwert 0	Ausgabe Erforderlich
COUNTER (Zähler) CNT (BCD) CNTX 546 (Binär) (nur CS1-H, CJ1- H, CJ1M und CS1D)	 N: Zählernummer S: Sollwert  N: Zähler- nummer S: Sollwert	CNT/CNTX(546) steuert einen dekrementierenden Zähler. Der Einstellbereich für den Sollwert reicht von 0 bis 9.999 bei BCD-Werten und von 0 bis 65.535 bei Binärwerten (dezimal oder hexadezimal). Zähleingang Rücksetz-Eingang Zähler-Istwert Ablaufmerker Sollwert 0 ON OFF	Ausgabe Erforderlich

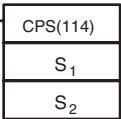
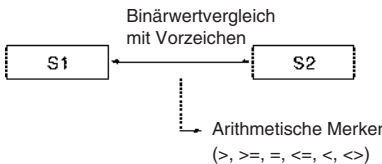
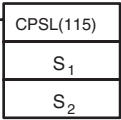
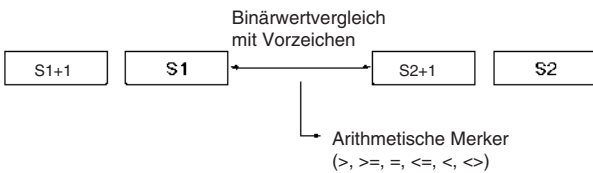
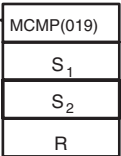
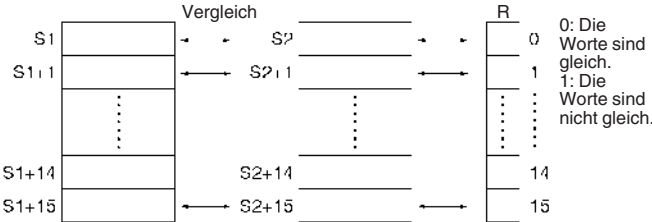
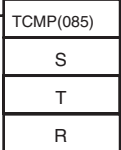
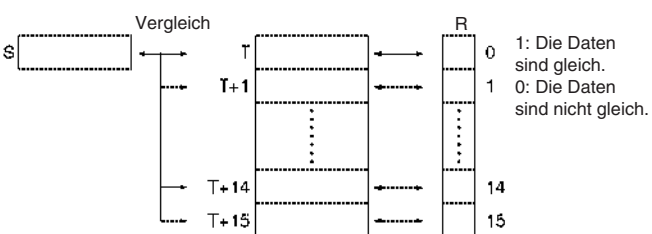
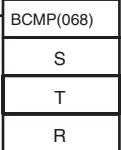
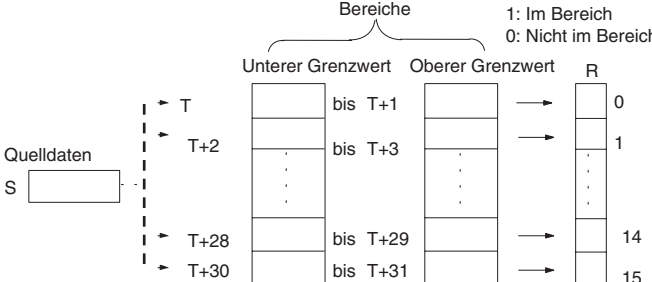
Befehl	Symbol/Operand	Funktion	Position Ausführungs- bedingung
REVERSIBLE COUNTER (Umkehrbarer Zähler) CNTR 012 (BCD) CNTRX 548 (Binär) (nur CS1-H, CJ1- H, CJ1M und CS1D)	 N: Zählnummer S: Sollwert  N: Zählnummer S: Sollwert	CNTR(012)/CNTRX(548) steuert einen reversierbaren Zähler. 	Ausgabe Erforderlich
RESET TIMER/ COUNTER (Zeit- geber/Zähler rücksetzen) CNR @CNR 545 (BCD) CNRX @CNRX 547 (Binär) (nur CS1-H, CJ1- H, CJ1M und CS1D)	 N1: Erste Zahl im Bereich N2: Letzte Zahl im Bereich  N1: Erste Nummer im Bereich N2: Letzte Nummer im Bereich	CNR(545)/CNRX(547) setzt die Zeitgeber oder Zähler innerhalb des spezifizierten Bereichs der Zeitgeber- bzw. Zählnummern zurück. Setzt den Istwert auf das Maximum: 9999.	Ausgabe Erforderlich

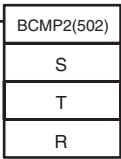
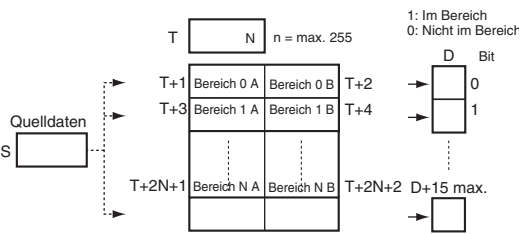
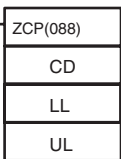
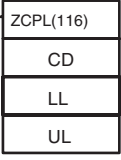
3-5 Vergleichsbefehle

*1: Nicht unterstützt von CS1D CPU-Baugruppen für Duplex-CPU-Systeme.

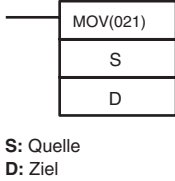
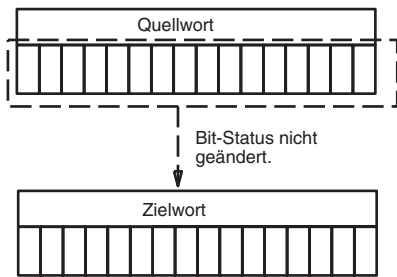
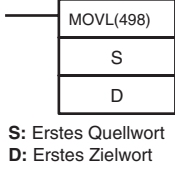
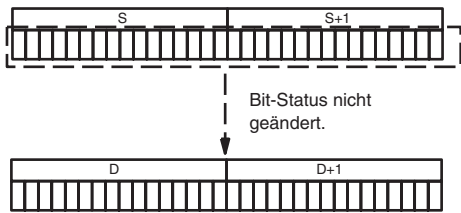
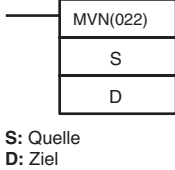
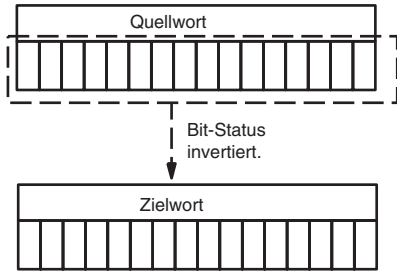
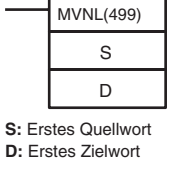
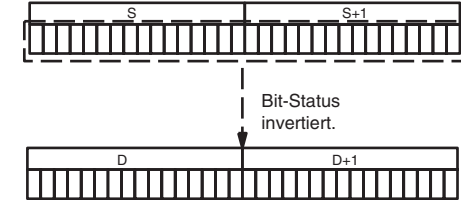
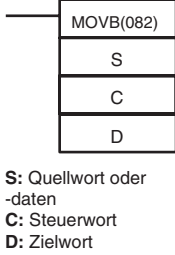
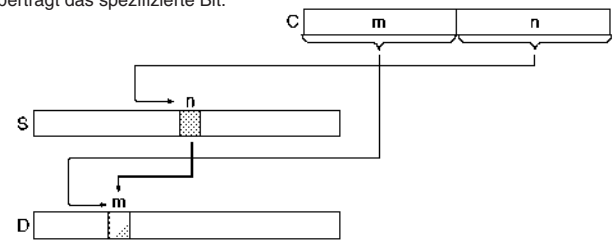
Befehl	Symbol/Operand	Funktion	Position Ausführungsbedingung
Symbolvergleich (ohne Vorzeichen) LD, AND, OR + =, <>, <, <=, >, >= 300 (=) 305 (<>) 310 (<) 315 (<=) 320 (>) 325 (>=)	 S₁ : Vergleichsdaten 1 S₂ : Vergleichsdaten 2	Symbolvergleichsbefehle (vorzeichenlos) vergleichen zwei Werte (Konstanten und/oder den Inhalt spezifizierter Worte) in 16-Bit-Binärdaten und erzeugen eine EIN-Ausführungsbedingung, wenn die Vergleichsbedingung wahr ist. Es gibt drei Arten von Symbolvergleichsbefehlen: LD (LOAD), AND und OR. LD  AND  OR 	LD: nicht erforderlich AND, OR: erforderlich
Symbolvergleich (Doppelwort, ohne Vorzeichen) LD, AND, OR + =, <>, <, <=, >, >= + L 301 (=) 306 (<>) 311 (<) 316 (<=) 321 (>) 326 (>=)	S₁ : Vergleichsdaten 1 S₂ : Vergleichsdaten 2	Symbolvergleichsbefehle (Doppelwort, ohne Vorzeichen) vergleichen zwei Werte (Konstanten und/oder den Inhalt spezifizierter Doppelwortdaten) in 32-Bit-Binärdaten ohne Vorzeichen und erzeugen eine EIN-Ausführungsbedingung, wenn die Vergleichsbedingung wahr ist. Es gibt drei Arten von Symbolvergleichsbefehlen: LD (LOAD), AND und OR.	LD: nicht erforderlich AND, OR: erforderlich
Symbolvergleich (mit Vorzeichen) LD, AND, OR + =, <>, <, <=, >, >= + S 302 (=) 307 (<>) 312 (<) 317 (<=) 322 (>) 327 (>=)	S₁ : Vergleichsdaten 1 S₂ : Vergleichsdaten 2	Symbolvergleichsbefehle (mit Vorzeichen) vergleichen zwei Werte (Konstanten oder den Inhalt spezifizierter Worte) in 16-Bit-Binärdaten mit Vorzeichen (4-stellige Hexadezimalzahlen) und erzeugen eine EIN-Ausführungsbedingung, wenn die Vergleichsbedingung wahr ist. Es gibt drei Arten von Symbolvergleichsbefehlen: LD (LOAD), AND und OR.	LD: nicht erforderlich AND, OR: erforderlich

Befehl AWL Code	Symbol/Operand	Funktion	Position Ausführungs- bedingung
Symbolvergleich (Doppelwort, mit Vorzeichen) LD, AND, OR +=, <>, <, <=, >, >= +SL 303 (=) 308 (<>) 313 (<) 318 (<=) 323 (>) 328 (>=)	S₁: Vergleichs- daten 1 S₂: Vergleichs- daten 2	Symbolvergleichsbefehle (Doppelwort, mit Vorzeichen) vergleichen zwei Werte (Konstanten oder den Inhalt spezifizierter Doppelwortdaten) in 32-Bit-Binärdaten mit Vorzeichen (8-stellige Hexadezimalzahlen) und erzeugen eine EIN-Ausführungsbedingung, wenn die Vergleichsbedingung wahr ist. Es gibt drei Arten von Symbolvergleichsbefehlen: LD (LOAD), AND und OR.	LD: nicht erforderlich AND, OR: erforderlich
Zeitvergleich LD, AND, OR += DT, <> DT, < DT, <= DT, > DT, >= DT 341 (= DT) 342 (<> DT) 343 (< DT) 344 (<= DT) 345 (> DT) 346 (>= DT) (nur CS/CJ-Serie CPU-Baugruppen Version 2.0 oder neuer)	LD (LOAD):  AND:  OR:  C: Steuerwort S₁: Erstes Wort der Ist-Zeit S₂: Erstes Wort der Vergleichszeit	Zeitvergleichsbefehle vergleichen zwei BCD-Zeitwerte und erzeugen eine EIN-Ausführungsbedingung, wenn die Vergleichsbedingung wahr ist. Es gibt drei Arten von Zeitvergleichsbefehlen: LD (LOAD), AND und OR. Zeitwerte (Jahr, Monat, Tag, Stunde, Minute und Sekunde) können beim Vergleich maskiert/demaskiert werden, so dass die Erstellung von Kalender-Timerfunktionen einfach ist.	LD: nicht erforderlich AND, OR: erforderlich
UNSIGNED COMPARE (Vergleich ohne Vorzeichen) CMP !CMP*1 020 S₁: Vergleichsdaten 1 S₂: Vergleichsdaten 2		Vergleicht zwei vorzeichenlose Binärwerte (Konstanten und/oder den Inhalt spezifizierter Worte) und gibt das Ergebnis an die arithmetischen Merker im Zusatz-Systembereich aus. 	Ausgabe Erforderlich
DOUBLE UNSIGNED COMPARE (Doppelwort- Vergleich ohne Vorzeichen) CMPL 060 S₁: Vergleichsdaten 1 S₂: Vergleichsdaten 2		Vergleicht zwei vorzeichenlose 2-Wort-Binärwerte (Konstanten und/oder den Inhalt spezifizierter Worte) und gibt das Ergebnis an die arithmetischen Merker im Zusatz-Systembereich aus. 	Ausgabe Erforderlich

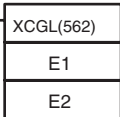
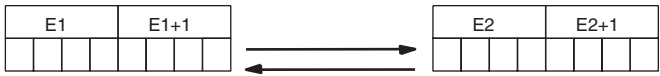
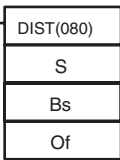
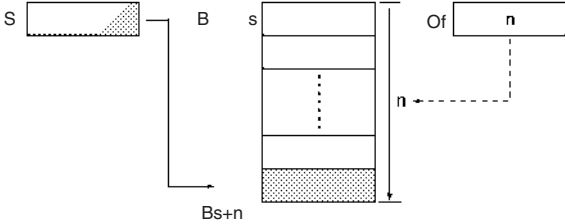
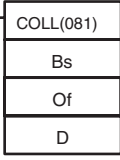
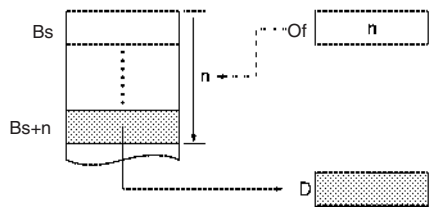
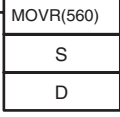
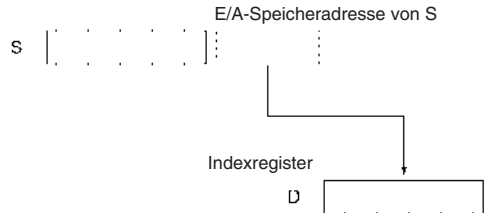
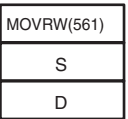
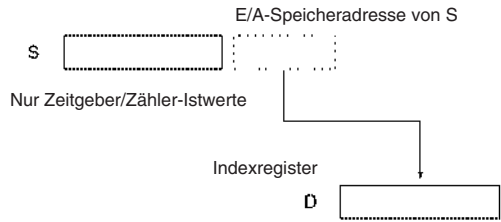
Befehl	Symbol/Operand	Funktion	Position Ausführungsbedingung
SIGNED BINARY COMPARE (Vorzeichenbehafteter Binärvergleich) CPS !CPS*1 114	 S1: Vergleichsdaten 1 S2: Vergleichsdaten 2	Vergleicht zwei Binärwerte mit Vorzeichen (Konstanten und/oder den Inhalt spezifizierter Worte) und gibt das Ergebnis an die arithmetischen Merker im Zusatz-Systembereich aus. 	Ausgabe Erforderlich
DOUBLE SIGNED BINARY COMPARE (Vorzeichenbehafteter Binär-Doppelwort-Vergleich) CPSL 115	 S1: Vergleichsdaten 1 S2: Vergleichsdaten 2	Vergleicht zwei 2-Wort-Binärwerte mit Vorzeichen (Konstanten und/oder den Inhalt spezifizierter Worte) und gibt das Ergebnis an die arithmetischen Merker im Zusatz-Systembereich aus. 	Ausgabe Erforderlich
MULTIPLE COMPARE (Mehrfach-Vergleich) MCMP @MCMP 019	 S1: Erstes Wort von Satz 1 S2: Erstes Wort von Satz 2 R: Ergebniswort	Vergleicht 16 aufeinander folgende Worte mit 16 anderen aufeinander folgenden Worten und setzt das entsprechende Bit im Ergebniswort auf EIN, wenn die Inhalte der Worte nicht gleich sind. 	Ausgabe Erforderlich
TABLE COMPARE (Tabellenvergleich) TCMP @TCMP 085	 S: Quelldaten T: Erstes Wort der Tabelle R: Ergebniswort	Vergleicht die Quelldaten mit dem Inhalt von 16 Worten und setzt das entsprechende Bit im Ergebniswort auf EIN, wenn die Inhalte identisch sind. 	Ausgabe Erforderlich
UNSIGNED BLOCK COMPARE (Vorzeichenloser Blockvergleich) BCMP @BCMP 068	 S: Quelldaten T: Erstes Wort der Tabelle R: Ergebniswort	Vergleicht die Quelldaten mit 16 Bereichen (durch 16 untere und 16 obere Grenzwerte definiert), und setzt das entsprechende Bit im Ergebniswort auf EIN, wenn sich die Quelldaten innerhalb des Bereichs befinden. 	Ausgabe Erforderlich

Befehl	Symbol/Operand	Funktion	Position Ausführungsbedingung
EXPANDED BLOCK COMPARE (Erweiterter Blockvergleich) BCMP2 @BCMP2 502 (nur CS1-H, CJ1-H und CS1D CPU-Baugruppen Version 2.0 oder neuer) CJ1M CPU-Baugruppe (vor Version 2.0 oder Baugruppenversion 2.0 oder neuer)	 S: Quelldaten T: Erstes Wort des Blocks R: Ergebniswort	Vergleicht die Quelldaten mit bis zu 256 Bereichen (durch obere und untere Grenzwerte definiert) und setzt das entsprechende Bit im Ergebniswort auf EIN, wenn die Quelldaten innerhalb eines Bereichs liegen.  Hinweis: A kann kleiner oder gleich B oder größer als B sein.	Ausgabe Erforderlich
AREA RANGE COMPARE (Bereichsvergleich) ZCP @ZCP 088 (nur CS1-H, CJ1-H, CJ1M und CS1D)	 CD: Vergleichsdaten (1 Wort) LL: Unterer Grenzwert des Bereichs UL: Oberer Grenzwert des Bereichs	Vergleicht den 16-Bit-Binärwert ohne Vorzeichen in CD (Wortinhalt oder Konstante) mit dem durch LL (unterer Grenzwert) und UL (oberer Grenzwert) definierten Bereich und gibt die Ergebnisse an Arithmetikmerker im Zusatzsystembereich aus.	Ausgabe Erforderlich
DOUBLE AREA RANGE COMPARE (Doppelwort-Bereichsvergleich) ZCPL @ZCPL 116 (nur CS1-H, CJ1-H, CJ1M und CS1D)	 CD: Vergleichsdaten (2 Worte) LL: Unterer Grenzwert des Bereichs UL: Oberer Grenzwert des Bereichs	Vergleicht den 32-Bit-Binärwert ohne Vorzeichen in CD und CD+1 (Wortinhalt oder Konstante) mit dem durch LL (unterer Grenzwert) und UL (oberer Grenzwert) definierten Bereich und gibt die Ergebnisse an Arithmetikmerker im Zusatzsystembereich aus.	Ausgabe Erforderlich

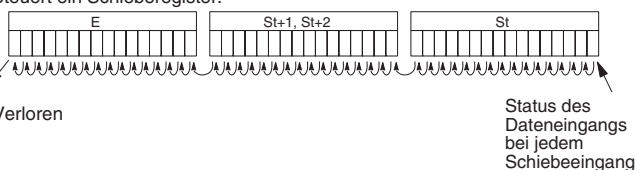
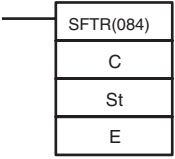
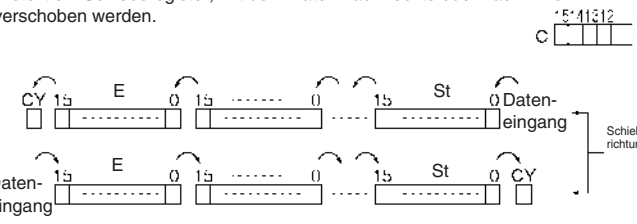
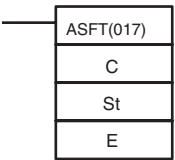
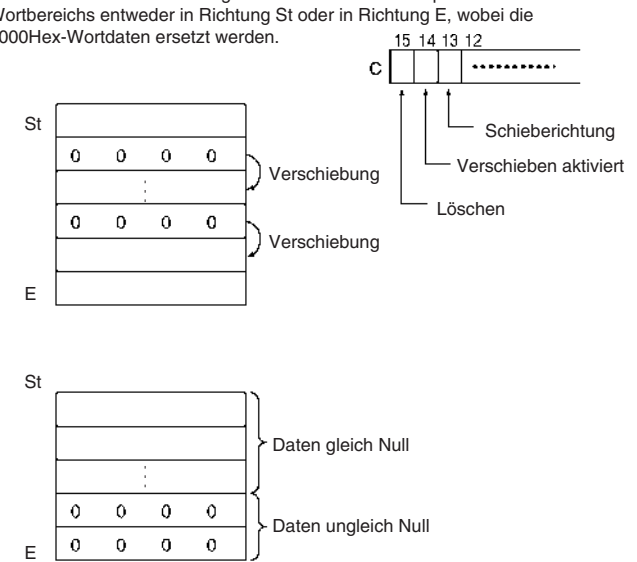
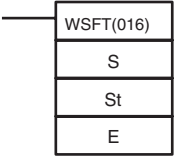
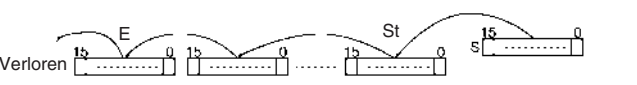
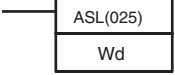
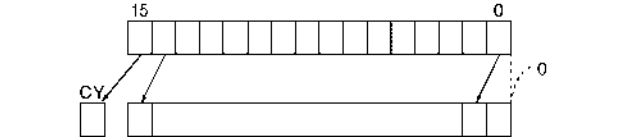
3-6 Kopierbefehle

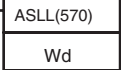
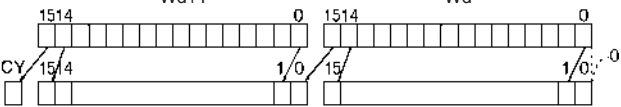
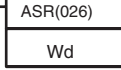
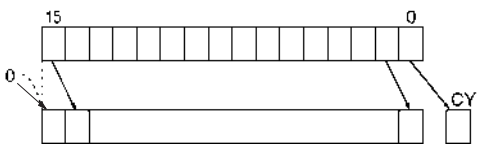
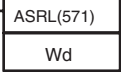
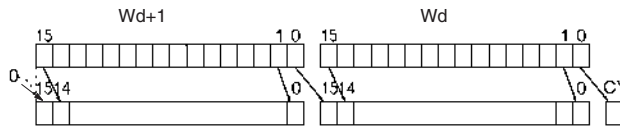
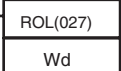
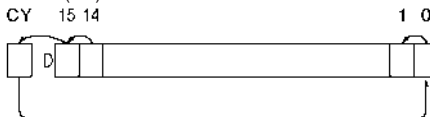
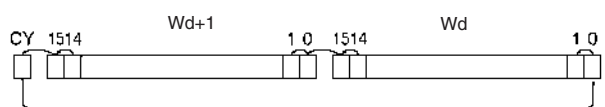
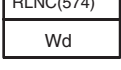
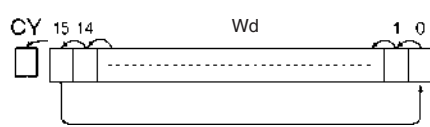
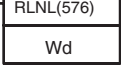
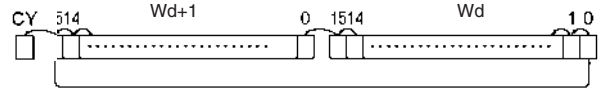
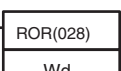
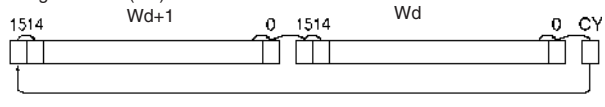
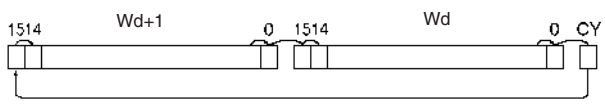
Befehl AWL Code	Symbol/Operand	Funktion	Position Ausführungs- bedingung
MOVE (Übertragen) MOV @MOV !MOV !@MOV 021		Überträgt ein Datenwort an das angegebene Wort. 	Ausgabe Erforderlich
DOUBLE MOVE (Doppelwort übertragen) MOVL @MOVL 498		Überträgt zwei Datenworte an die angegebenen Worte. 	Ausgabe Erforderlich
MOVE NOT (Übertragen invertiert) MVN @MVN 022		Überträgt das Komplement eines Datenworts an das angegebene Wort. 	Ausgabe Erforderlich
DOUBLE MOVE NOT (Doppelwort übertragen invertiert) MVNL @MVNL 499		Überträgt das Komplement von zwei Datenwörtern an die angegebenen Worte. 	Ausgabe Erforderlich
MOVE BIT (Bit übertragen) MOVB @MOVB 082		Überträgt das spezifizierte Bit. 	Ausgabe Erforderlich

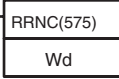
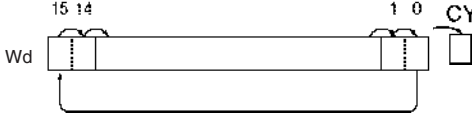
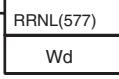
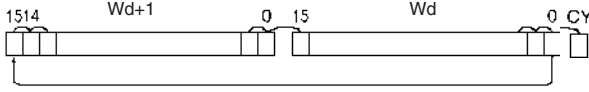
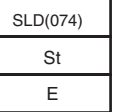
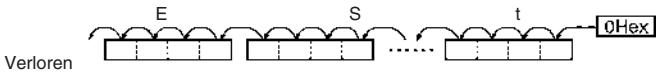
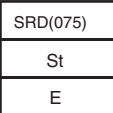
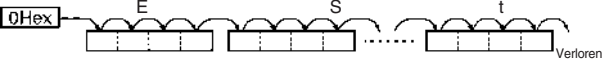
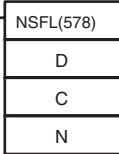
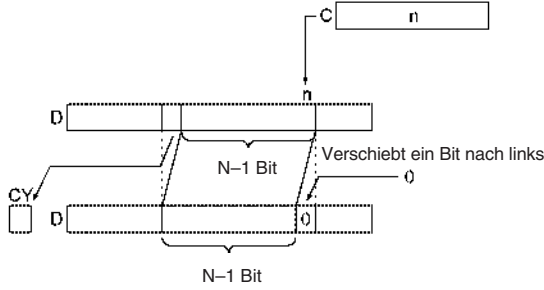
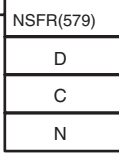
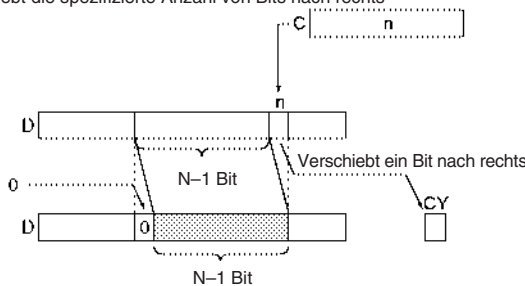
Befehl	Symbol/Operand	Funktion	Position Ausführungs- bedingung
MOVE DIGIT (Stelle übertragen) MOVD @MOVD 083	MOVD(083) S C D S: Quellwort oder -daten C: Steuerwort D: Zielwort	Überträgt die spezifizierte(n) Stelle oder Stellen. (Jede Stelle besteht aus 4 Bits.)	Ausgabe Erforderlich
MULTIPLE BIT TRANSFER (Mehrfache Bitübertragung) XFRB @XFRB 062	XFRB(062) C S D C: Steuerwort S: Erstes Quellwort D: Erstes Zielwort	Überträgt die spezifizierte Anzahl von aufeinander folgenden Bits.	Ausgabe Erforderlich
BLOCK TRANSFER (Block- übertragung) XFER @XFER 070	XFER(070) N S D N: Anzahl der Worte S: Erstes Quellwort D: Erstes Zielwort	Überträgt die spezifizierte Anzahl von aufeinander folgenden Worten.	Ausgabe Erforderlich
BLOCK SET (Block-Setzen) BSET @BSET 071	BSET(071) S St E S: Quellwort St: Anfangswort E: Endwort	Kopiert dasselbe Wort in einen Bereich aufeinander folgender Worte.	Ausgabe Erforderlich
DATA EXCHANGE (Datenaustausch) XCHG @XCHG 073	XCHG(073) E1 E2 E1: Erstes Austauschwort E2: Zweites Austauschwort	Tauscht den Inhalt der zwei spezifizierten Worte aus.	Ausgabe Erforderlich

Befehl	Symbol/Operand	Funktion	Position Ausführungsbedingung
DOUBLE DATA EXCHANGE (Doppelwort-Datenaustausch) XCGL @XCGL 562	 E1: Erstes Austauschwort E2: Zweites Austauschwort	Tauscht den Inhalt von zwei aufeinander folgenden Worten mit zwei anderen aufeinander folgenden Worten aus. 	Ausgabe Erforderlich
SINGLE WORD DISTRIBUTE (Einzelwort verteilen) DIST @DIST 080	 S: Quellwort Bs: Zielbasisadresse Of: Offset	Überträgt das Quellwort an ein durch Hinzufügen eines Offsetwerts zur Basisadresse errechneten Zielworts. 	Ausgabe Erforderlich
DATA COLLECT (Daten sammeln) COLL @COLL 081	 Bs: Quellbasisadresse Of: Offset D: Zielwort	Überträgt das Quellwort (errechnet durch Hinzufügen eines Offsetwerts zur Basisadresse) an das Zielwort. 	Ausgabe Erforderlich
MOVE TO REGISTER (zu Register übertragen) MOVR @MOVR 560	 S: Quelle (gewünschtes Wort oder Bit) D: Ziel (Indexregister)	Schreibt die interne E/A-Speicheradresse des spezifizierten Worts, Bits oder Zeitgeber/Zähler-Ablaufmerkers in das spezifizierte Indexregister. (Verwenden Sie den Befehl MOVRW(561), um die interne E/A-Speicheradresse eines Zeitgeber/Zähler-Istwerts in ein Indexregister zu schreiben) 	Ausgabe Erforderlich
MOVE TIMER/COUNTER PV TO REGISTER (Zeitgeber/Zähler zu Register übertragen) MOVRW @MOVRW 561	 S: Quelle (gewünschte Zeitgeber/Zähler-Nummer) D: Ziel (Indexregister)	Schreibt die interne E/A-Speicheradresse des spezifizierten Zeitgeber- oder Zähler-Istwerts in das spezifizierte Indexregister. (Verwenden Sie den Befehl MOVR(560), um die interne E/A-Speicheradresse eines Worts, Bits oder Zeitgeber/Zähler-Ablaufmerkers in ein Indexregister zu schreiben) 	Ausgabe Erforderlich

3-7 Datenverschiebepfehle

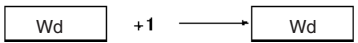
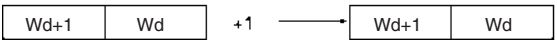
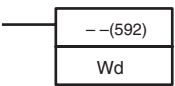
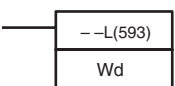
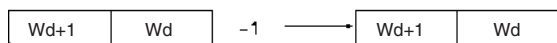
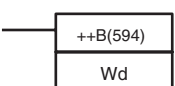
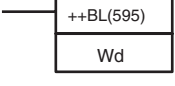
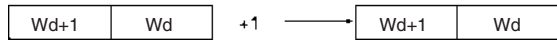
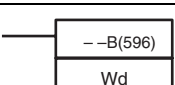
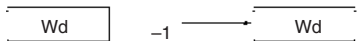
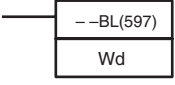
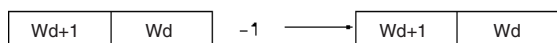
Befehl	Symbol/Operand	Funktion	Position Ausführungs- bedingung
SHIFT REGISTER (Schieberegister) SFT 010	 St: Anfangswort E: Endwort	Steuert ein Schieberegister.  Verloren Status des Dateneingangs bei jedem Schiebeeingang	Ausgabe Erforderlich
REVERSIBLE SHIFT REGISTER (Umkehrbares Schieberegister) SFTR @SFTR 084	 C: Steuerwort St: Anfangswort E: Endwort	Erstellt ein Schieberegister, mit dem Daten nach rechts oder nach links verschoben werden.  Schiebe- richtung	Ausgabe Erforderlich
ASYNCHRO- NOUS SHIFT REGISTER (Asynchrones Schieberegister) ASFT @ASFT 017	 C: Steuerwort St: Anfangswort E: Endwort	Verschiebt alle Wortdaten ungleich Null innerhalb des spezifizierten Wortbereichs entweder in Richtung St oder in Richtung E, wobei die 0000Hex-Wortdaten ersetzt werden.  Schieberichtung Verschieben aktiviert Löschen Daten gleich Null Daten ungleich Null	Ausgabe Erforderlich
WORD SHIFT (Wortverschiebung) WSFT @WSFT 016	 S: Quellwort St: Anfangswort E: Endwort	Verschiebt Daten zwischen St und E in Worteinheiten  Verloren	Ausgabe Erforderlich
ARITHMETIC SHIFT LEFT (Arithmetisches Verschieben nach links) ASL @ASL 025	 Wd: Wort	Verschiebt den Inhalt von Wd ein Bit nach links. 	Ausgabe Erforderlich

Befehl	Symbol/Operand	Funktion	Position Ausführungsbedingung
DOUBLE SHIFT LEFT (Doppelwort verschieben nach links) ASLL @ASLL 570	 Wd: Wort	Verschiebt den Inhalt von Wd und Wd +1 ein Bit nach links. 	Ausgabe Erforderlich
ARITHMETIC SHIFT RIGHT (Arithmetisches Verschieben nach rechts) ASR @ASR 026	 Wd: Wort	Verschiebt den Inhalt von Wd ein Bit nach rechts. 	Ausgabe Erforderlich
DOUBLE SHIFT RIGHT (Doppelwort verschieben nach rechts) ASRL @ASRL 571	 Wd: Wort	Verschiebt den Inhalt von Wd und Wd +1 ein Bit nach rechts. 	Ausgabe Erforderlich
ROTATE LEFT (Rotation nach links) ROL @ROL 027	 Wd: Wort	Verschiebt alle Wd-Bits ein Bit nach links einschließlich des Übertragsmerkers (CY). 	Ausgabe Erforderlich
DOUBLE ROTATE LEFT (Doppelwort rotieren nach links) ROLL @ROLL 572	 Wd: Wort	Verschiebt alle Bits von Wd und Wd +1 um ein Bit nach links einschließlich des Übertragsmerkers (CY). 	Ausgabe Erforderlich
ROTATE LEFT WITHOUT CARRY (Rotation nach links ohne Übertrag) RLNC @RLNC 574	 Wd: Wort	Verschiebt alle Bits von Wd ein Bit nach links ohne Übertragsmerker (CY). 	Ausgabe Erforderlich
DOUBLE ROTATE LEFT WITHOUT CARRY (Doppelwort rotieren nach links ohne Übertrag) RLNL @RLNL 576	 Wd: Wort	Verschiebt alle Bits von Wd und Wd +1 um ein Bit nach links ohne den Übertragsmerker (CY). 	Ausgabe Erforderlich
ROTATE RIGHT (Rotation nach rechts) ROR @ROR 028	 Wd: Wort	Verschiebt alle Wd-Bits ein Bit nach rechts einschließlich des Übertragsmerkers (CY). 	Ausgabe Erforderlich
DOUBLE ROTATE RIGHT (Doppelwort rotieren nach rechts) RORL @RORL 573	 Wd: Wort	Verschiebt alle Bits von Wd und Wd +1 um ein Bit nach rechts einschließlich des Übertragsmerkers (CY). 	Ausgabe Erforderlich

Befehl	Symbol/Operand	Funktion	Position Ausführungsbedingung
ROTATE RIGHT WITHOUT CARRY (Rotation nach rechts ohne Übertrag) RRNC @RRNC 575	 Wd: Wort	Verschiebt alle Bits von Wd ein Bit nach rechts ohne den Übertragsmerker (CY). Der Inhalt des niederwertigsten Bits (rechts) von Wd wird zum höchstwertigen Bit (links) und dem Übertragsmerker (CY) verschoben. 	Ausgabe Erforderlich
DOUBLE ROTATE RIGHT WITHOUT CARRY (Doppelwort rotieren nach rechts ohne Übertrag) RRNL @RRNL 577	 Wd: Wort	Verschiebt alle Bits von Wd und Wd + 1 um ein Bit nach rechts ohne den Übertragsmerker (CY). Der Inhalt des niederwertigsten Bits (rechts) von Wd + 1 wird zum höchstwertigen Bit (links) von Wd und dem Übertragsmerker (CY) verschoben. 	Ausgabe Erforderlich
ONE DIGIT SHIFT LEFT (Eine Stelle nach links verschieben) SLD @SLD 074	 St: Anfangswort E: Endwort	Verschiebt Daten um eine Stelle (4 Bits) nach links.  Verloren	Ausgabe Erforderlich
ONE DIGIT SHIFT RIGHT (Eine Stelle nach rechts verschieben) SRD @SRD 075	 St: Anfangswort E: Endwort	Verschiebt Daten um eine Stelle (4 Bits) nach rechts.  Verloren	Ausgabe Erforderlich
SHIFT N-BIT DATA LEFT (N-Bits verschieben nach links) NSFL @NSFL 578	 D: Anfangswort für Verschiebung C: Anfangsbit N: Datenlänge für Verschiebung	Verschiebt die spezifizierte Anzahl von Bits nach links  Verschiebt ein Bit nach links	Ausgabe Erforderlich
SHIFT N-BIT DATA RIGHT (N-Bits verschieben nach rechts) NSFR @NSFR 579	 D: Anfangswort für Verschiebung C: Anfangsbit N: Datenlänge für Verschiebung	Verschiebt die spezifizierte Anzahl von Bits nach rechts  Verschiebt ein Bit nach rechts	Ausgabe Erforderlich

Befehl	Symbol/Operand	Funktion	Position Ausführungsbedingung
SHIFT N-BITS LEFT (N-Bits verschieben nach links) NASL @NASL 580	NASL(580) D C D: Verschiebewort C: Steuerwort	Verschiebt die spezifizierten 16 Bits Wortdaten um die spezifizierte Anzahl von Bits nach links 	Ausgabe Erforderlich
DOUBLE SHIFT N-BITS LEFT (Doppelwort-N-Bits verschieben nach links) NSLL @NSLL 582	NSLL(582) D C D: Verschiebewort C: Steuerwort	Verschiebt die spezifizierten 32 Bits Wortdaten um die spezifizierte Anzahl von Bits nach links 	Ausgabe Erforderlich
SHIFT N-BITS RIGHT (N-Bits verschieben nach rechts) NASR @NASR 581	NASR(581) D C D: Verschiebewort C: Steuerwort	Verschiebt die spezifizierten 16Bits Wortdaten um die spezifizierte Anzahl von Bits nach rechts 	Ausgabe Erforderlich
DOUBLE SHIFT N-BITS RIGHT (Doppelwort-N-Bits verschieben nach rechts) NSRL @NSRL 583	NSRL(583) D C D: Verschiebewort C: Steuerwort	Verschiebt die spezifizierten 32 Bits Wortdaten um die spezifizierte Anzahl von Bits nach rechts 	Ausgabe Erforderlich

3-8 Inkrementier- und Dekrementierbefehle

Befehl AWL Code	Symbol/Operand	Funktion	Position Ausführungs- bedingung
INCREMENT BINARY (Binär- wert inkremen- tieren) ++ @++ 590	 Wd: Wort	Inkrementiert den 4-stelligen Hexadezimal-Inhalt des spezifizierten Worts um 1. 	Ausgabe Erforderlich
DOUBLE INCRE- MENT BINARY (Binär-Doppel- wort inkremen- tieren) ++L @++L 591	 Wd: Wort	Inkrementiert den 8-stelligen Hexadezimal-Inhalt der spezifizierten Worte um 1. 	Ausgabe Erforderlich
DECREMENT BINARY (Binär- wert dekremen- tieren) -- @-- 592	 Wd: Wort	Dekrementiert den 4-stelligen Hexadezimal-Inhalt des spezifizierten Worts um 1. 	Ausgabe Erforderlich
DOUBLE DECREMENT BINARY (Binär- Doppelwort dekrementieren) --L @--L 593	 Wd: Erstes Wort	Dekrementiert den 8-stelligen Hexadezimal-Inhalt der spezifizierten Worte um 1. 	Ausgabe Erforderlich
INCREMENT BCD (BCD-Wert inkrementieren) ++B @++B 594	 Wd: Wort	Inkrementiert den 4-stelligen BCD-Inhalt des spezifizierten Worts um 1. 	Ausgabe Erforderlich
DOUBLE INCRE- MENT BCD (BCD-Doppel- wort inkremen- tieren) ++BL @++BL 595	 Wd: Erstes Wort	Inkrementiert den 8-stelligen BCD-Inhalt der spezifizierten Worte um 1. 	Ausgabe Erforderlich
DECREMENT BCD (BCD-Wert dekrementieren) --B @--B 596	 Wd: Wort	Dekrementiert den 4-stelligen BCD-Inhalt des spezifizierten Worts um 1. 	Ausgabe Erforderlich
DOUBLE DECREMENT BCD (BCD- Doppelwort dekrementieren) --BL @--BL 597	 Wd: Erstes Wort	Dekrementiert den 8-stelligen BCD-Inhalt der spezifizierten Worte um 1. 	Ausgabe Erforderlich

3-9 Arithmetische Befehle

Befehl	Symbol/Operand	Funktion	Position Ausführungs- bedingung
SIGNED BINARY ADD WITHOUT CARRY (Vorzeichenbehaftete Binärwert-Addition ohne Übertrag) + @+ 400	+ (400) Au Ad R Au: Augend-Wort Ad: Addend-Wort R: Ergebniswort	Addiert 4-stellige (Einzelwort) hexadezimale Daten und/oder Konstanten. Au (Binär mit Vorzeichen) Ad (Binär mit Vorzeichen) + CY (Binär mit Vorzeichen) R (Binär mit Vorzeichen) CY wird auf EIN gesetzt, wenn ein Übertrag vorhanden ist.	Ausgabe Erforderlich
DOUBLE SIGNED BINARY ADD WITHOUT CARRY (Vorzeichenbehaftete Binär-Doppelwort-Addition ohne Übertrag) +L @+L 401	+L (401) Au Ad R Au: Erstes Augend-Wort Ad: Erstes Addend-Wort R: Erstes Ergebniswort	Addiert 8-stellige (Doppelwort) hexadezimale Daten und/oder Konstanten. Au+1 Au (Binär mit Vorzeichen) Ad+1 Ad (Binär mit Vorzeichen) + CY R+1 R (Binär mit Vorzeichen) CY wird auf EIN gesetzt, wenn ein Übertrag vorhanden ist.	Ausgabe Erforderlich
SIGNED BINARY ADD WITH CARRY (Vorzeichenbehaftete Binärwert-Addition mit Übertrag) +C @+C 402	+C (402) Au Ad R Au: Augend-Wort Ad: Addend-Wort R: Ergebniswort	Addiert 4-stellige (Einzelwort) hexadezimale Daten und/oder Konstanten mit Übertragungsmerker (CY). Au (Binär mit Vorzeichen) Ad (Binär mit Vorzeichen) + CY CY R (Binär mit Vorzeichen) CY wird auf EIN gesetzt, wenn ein Übertrag vorhanden ist.	Ausgabe Erforderlich
DOUBLE SIGNED BINARY ADD WITH CARRY (Vorzeichenbehaftete Binär-Doppelwort-Addition mit Übertrag) +CL @+CL 403	+CL (403) Au Ad R Au: Erstes Augend-Wort Ad: Erstes Addend-Wort R: Erstes Ergebniswort	Addiert 8-stellige (Doppelwort) hexadezimale Daten und/oder Konstanten mit Übertragungsmerker (CY). Au+1 Au (Binär mit Vorzeichen) Ad+1 Ad (Binär mit Vorzeichen) + CY CY R+1 R (Binär mit Vorzeichen) CY wird auf EIN gesetzt, wenn ein Übertrag vorhanden ist.	Ausgabe Erforderlich
BCD ADD WITHOUT CARRY (BCD-Addition ohne Übertrag) +B @+B 404	+B (404) Au Ad R Au: Augend-Wort Ad: Addend-Wort R: Ergebniswort	Addiert 4-stellige (Einzelwort) BCD-Daten und/oder Konstanten. Au (BCD) Ad (BCD) + CY (BCD) R (BCD) CY wird auf EIN gesetzt, wenn ein Übertrag vorhanden ist.	Ausgabe Erforderlich

Befehl	Symbol/Operand	Funktion	Position Ausführungs- bedingung
DOUBLE BCD ADD WITHOUT CARRY (BCD- Doppelwort- Addition ohne Übertrag) +BL @+BL 405	+BL(405) Au Ad R Au: Erstes Augend-Wort Ad: Erstes Addend-Wort R: Erstes Ergebniswort	Addiert 8-stellige (Doppelwort) BCD-Daten und/oder Konstanten. Au+1 Au (BCD) Ad+1 Ad (BCD) + CY R+1 R (BCD) CY wird auf EIN gesetzt, wenn ein Übertrag vorhanden ist.	Ausgabe Erforderlich
BCD ADD WITH CARRY (BCD- Addition mit Übertrag) +BC @+BC 406	+BC(406) Au Ad R Au: Augend-Wort Ad: Addend-Wort R: Ergebniswort	Addiert 4-stellige (Einzelwort) BCD-Daten und/oder Konstanten mit Übertragungsmerker (CY). Au (BCD) Ad (BCD) + CY CY R (BCD) CY wird auf EIN gesetzt, wenn ein Übertrag vorhanden ist.	Ausgabe Erforderlich
DOUBLE BCD ADD WITH CARRY (BCD-Doppel- wort-Addition mit Übertrag) +BCL @+BCL 407	+BCL(407) Au Ad R Au: Erstes Augend-Wort Ad: Erstes Addend-Wort R: Erstes Ergebniswort	Addiert 8-stellige (Doppelwort) BCD-Daten und/oder Konstanten mit Übertragungsmerker (CY). Au+1 Au (BCD) Ad+1 Ad (BCD) + CY CY R+1 R (BCD) CY wird auf EIN gesetzt, wenn ein Übertrag vorhanden ist.	Ausgabe Erforderlich
SIGNED BINARY SUBTRACT WITHOUT CARRY (Vorzei- chenbehaftete Binärwert-Sub- traktion ohne Übertrag) - @- 410	-(410) Mi Su R Mi: Minuend-Wort Su: Subtrahend-Wort R: Ergebniswort	Subtrahiert 4-stellige (Einzelwort) hexadezimale Daten und/oder Konstanten. Mi (Binär mit Vorzeichen) - Su (Binär mit Vorzeichen) + CY R (Binär mit Vorzeichen) CY wird auf EIN gesetzt, wenn ein negativer Übertrag vorhanden ist.	Ausgabe Erforderlich
DOUBLE SIGNED BINARY SUBTRACT WITHOUT CARRY (Vorzei- chenbehaftete Binär-Doppel- wort-Subtraktion ohne Übertrag) -L @-L 411	-L(411) Mi Su R Mi: Minuend-Wort Su: Subtrahend-Wort R: Ergebniswort	Subtrahiert 8-stellige (Doppelwort) hexadezimale Daten und/oder Konstanten. Mi+1 Mi (Binär mit Vorzeichen) - Su+1 Su (Binär mit Vorzeichen) + CY R+1 R (Binär mit Vorzeichen) CY wird auf EIN gesetzt, wenn ein negativer Übertrag vorhanden ist.	Ausgabe Erforderlich

Befehl AWL Code	Symbol/Operand	Funktion	Position Ausführungs- bedingung
SIGNED BINARY SUBTRACT WITH CARRY (Vorzeichenbehaftete Binärwert-Subtraktion mit Übertrag) -C @-C 412	-C(412) Mi Su R Mi: Minuend-Wort Su: Subtrahend-Wort R: Ergebniswort	Subtrahiert 4-stellige (Einzelwort) hexadezimale Daten und/oder Konstanten mit Übertragungsmerker (CY). Mi (Binär mit Vorzeichen) Su (Binär mit Vorzeichen) CY - CY R (Binär mit Vorzeichen) CY wird auf EIN gesetzt, wenn ein negativer Übertrag vorhanden ist.	Ausgabe Erforderlich
DOUBLE SIGNED BINARY SUBTRACT WITH CARRY (Vorzeichenbehaftete Binär-Doppelwort-Subtraktion mit Übertrag) -CL @-CL 413	-CL(413) Mi Su R Mi: Minuend-Wort Su: Subtrahend-Wort R: Ergebniswort	Subtrahiert 8-stellige (Doppelwort) hexadezimale Daten und/oder Konstanten mit Übertragungsmerker (CY). Mi+1 Mi (Binär mit Vorzeichen) Su+1 Su (Binär mit Vorzeichen) CY - CY R+1 R (Binär mit Vorzeichen) CY wird auf EIN gesetzt, wenn ein negativer Übertrag vorhanden ist.	Ausgabe Erforderlich
BCD SUBTRACT WITHOUT CARRY (BCD-Subtraktion ohne Übertrag) -B @-B 414	-B(414) Mi Su R Mi: Minuend-Wort Su: Subtrahend-Wort R: Ergebniswort	Subtrahiert 4-stellige (Einzelwort) BCD-Daten und/oder Konstanten. Mi (BCD) Su (BCD) - CY R (BCD) CY wird auf EIN gesetzt, wenn ein Übertrag vorhanden ist.	Ausgabe Erforderlich
DOUBLE BCD SUBTRACT WITHOUT CARRY (BCD-Doppelwort-Subtraktion ohne Übertrag) -BL @-BL 415	-BL(415) Mi Su R Mi: Erstes Minuend-Wort Su: Erstes Subtrahend-Wort R: Erstes Ergebniswort	Subtrahiert 8-stellige (Doppelwort) BCD-Daten und/oder Konstanten. Mi +1 Mi (BCD) Su+1 Su (BCD) - CY R+1 R (BCD) CY wird auf EIN gesetzt, wenn ein negativer Übertrag vorhanden ist.	Ausgabe Erforderlich
BCD SUBTRACT WITH CARRY (BCD-Subtraktion mit Übertrag) -BC @-BC 416	-BC(416) Mi Su R Mi: Minuend-Wort Su: Subtrahend-Wort R: Ergebniswort	Subtrahiert 4-stellige (Einzelwort) BCD-Daten und/oder Konstanten mit Übertragungsmerker (CY). Mi (BCD) Su (BCD) CY - CY R (BCD) CY wird auf EIN gesetzt, wenn ein negativer Übertrag vorhanden ist.	Ausgabe Erforderlich

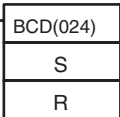
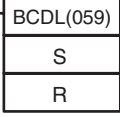
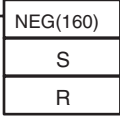
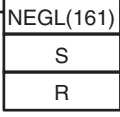
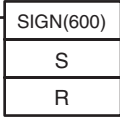
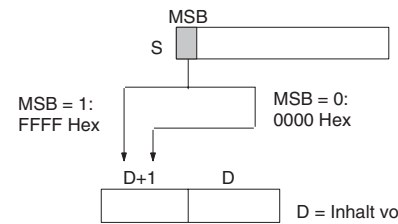
Befehl	Symbol/Operand	Funktion	Position Ausführungs- bedingung
DOUBLE BCD SUBTRACT WITH CARRY (BCD-Doppelwort-Subtraktion mit Übertrag) -BCL @-BCL 417	-BCL(417) Mi Su R Mi: Erstes Minuend-Wort Su: Erstes Subtrahend-Wort R: Erstes Ergebniswort	Subtrahiert 8-stellige (Doppelwort) BCD-Daten und/oder Konstanten mit Übertragungsmerker (CY). Mi + 1 Mi (BCD) Su + 1 Su (BCD) - CY CY R + 1 R (BCD) CY wird auf EIN gesetzt, wenn ein negativer Übertrag vorhanden ist.	Ausgabe Erforderlich
SIGNED BINARY MULTIPLY (Vorzeichenbehaf-tete Binärwert-Multiplikation) * @* 420	*(420) Md Mr R Md: Multiplikand-Wort Mr: Multiplikator-Wort R: Ergebniswort	Multipliziert 4-stellige vorzeichenbehaf-tete Hexadezimal-Daten und/oder Konstanten. Md (Binär mit Vorzeichen) x Mr (Binär mit Vorzeichen) R + 1 R (Binär mit Vorzeichen)	Ausgabe Erforderlich
DOUBLE SIGNED BINARY MULTIPLY (Vorzeichenbehaftete Binär-Doppelwort-Multiplikation) *L @*L 421	*L(421) Md Mr R Md: Erstes Multiplikand-Wort Mr: Erstes Multiplikator-Wort R: Erstes Ergebniswort	Multipliziert 8-stellige vorzeichenbehaf-tete Hexadezimal-Daten und/oder Konstanten. Md + 1 Md (Binär mit Vorzeichen) x Mr + 1 Mr (Binär mit Vorzeichen) R + 3 R + 2 R + 1 R (Binär mit Vorzeichen)	Ausgabe Erforderlich
UNSIGNED BINARY MULTIPLY (Vorzeichenlose Binärwert-Multiplikation) *U @*U 422	*U(422) Md Mr R Md: Multiplikand-Wort Mr: Multiplikator-Wort R: Ergebniswort	Multipliziert 4-stellige vorzeichenlose Hexadezimal-Daten und/oder Konstanten. Md (Binär ohne Vorzeichen) x Mr (Binär ohne Vorzeichen) R + 1 R (Binär ohne Vorzeichen)	Ausgabe Erforderlich
DOUBLE UNSIGNED BINARY MULTIPLY (Vorzeichenlose Binär-Doppelwort-Multiplikation) *UL @*UL 423	*UL(423) Md Mr R Md: Erstes Multiplikand-Wort Mr: Erstes Multiplikator-Wort R: Erstes Ergebniswort	Multipliziert 8-stellige vorzeichenlose Hexadezimal-Daten und/oder Konstanten. Md + 1 Md (Binär ohne Vorzeichen) x Mr + 1 Mr (Binär ohne Vorzeichen) R + 3 R + 2 R + 1 R (Binär ohne Vorzeichen)	Ausgabe Erforderlich

Befehl AWL Code	Symbol/Operand	Funktion	Position Ausführungs- bedingung
BCD MULTIPLY (BCD-Multiplikation) *B @*B 424	*B(424) Md Mr R Md: Multiplikand-Wort Mr: Multiplikator-Wort R: Ergebniswort	Multipliziert 4-stellige (Einzelwort) BCD-Daten und/oder Konstanten. Md (BCD) x Mr (BCD) R + 1 R (BCD)	Ausgabe Erforderlich
DOUBLE BCD MULTIPLY (BCD-Doppelwort-Multiplikation) *BL @*BL 425	*BL(425) Md Mr R Md: Erstes Multiplikand-Wort Mr: Erstes Multiplikator-Wort R: Erstes Ergebniswort	Multipliziert 8-stellige (Doppelwort) BCD-Daten und/oder Konstanten. Md + 1 Md (BCD) x Mr + 1 Mr (BCD) R + 3 R + 2 R + 1 R (BCD)	Ausgabe Erforderlich
SIGNED BINARY DIVIDE (Vorzeichenbehaftete Binär-Division) / @/ 430	/(430) Dd Dr R Dd: Dividend-Wort Dr: Divisor-Wort R: Ergebniswort	Dividiert 4-stellige (Einzelwort) Hexadezimal-Daten mit Vorzeichen und/oder Konstanten. Dd (Binär mit Vorzeichen) ÷ Dr (Binär mit Vorzeichen) R + 1 R (Binär mit Vorzeichen) Rest Quotient	Ausgabe Erforderlich
DOUBLE SIGNED BINARY DIVIDE (Vorzeichenbehaftete Binär-Doppelwort-Division) /L @/L 431	/L(431) Dd Dr R Dd: Erstes Dividend-Wort Dr: Erstes Divisor-Wort R: Erstes Ergebniswort	Dividiert 8-stellige (Doppelwort) Hexadezimal-Daten mit Vorzeichen und/oder Konstanten. Dd + 1 Dd (Binär mit Vorzeichen) ÷ Dr + 1 Dr (Binär mit Vorzeichen) R + 3 R + 2 R + 1 R (Binär mit Vorzeichen) Rest Quotient	Ausgabe Erforderlich
UNSIGNED BINARY DIVIDE (Vorzeichenlose Binär-Division) /U @/U 432	/U(432) Dd Dr R Dd: Dividend-Wort Dr: Divisor-Wort R: Ergebniswort	Dividiert 4-stellige (Einzelwort) Hexadezimal-Daten ohne Vorzeichen und/oder Konstanten. Dd (Binär ohne Vorzeichen) ÷ Dr (Binär ohne Vorzeichen) R + 1 R (Binär ohne Vorzeichen) Rest Quotient	Ausgabe Erforderlich

Befehl	AWL Code	Symbol/Operand	Funktion	Position Ausführungsbedingung
DOUBLE UNSIGNED BINARY DIVIDE (Vorzeichenlose Binär-Doppelwort-Division)	/UL @/UL 433	/UL(433) Dd Dr R Dd: Erstes Dividend-Wort Dr: Erstes Divisor-Wort R: Erstes Ergebniswort	Dividiert 8-stellige (Doppelwort) Hexadezimal-Daten ohne Vorzeichen und/oder Konstanten. Dd + 1 Dd (Binär ohne Vorzeichen) ÷ Dr + 1 Dr (Binär ohne Vorzeichen) R + 3 R + 2 R + 1 R (Binär ohne Vorzeichen) Rest Quotient	Ausgabe Erforderlich
BCD DIVIDE (BCD-Division)	/B @/B 434	/B(434) Dd Dr R Dd: Dividend-Wort Dr: Divisor-Wort R: Ergebniswort	Dividiert 4-stellige (Einzelwort) BCD-Daten und/oder Konstanten. Dd (BCD) ÷ Dr (BCD) R + 1 R (BCD) Rest Quotient	Ausgabe Erforderlich
DOUBLE BCD DIVIDE (BCD-Doppelwort-Division)	/BL @/BL 435	/BL(435) Dd Dr R Dd: Erstes Dividend-Wort Dr: Erstes Divisor-Wort R: Erstes Ergebniswort	Dividiert 8-stellige (Doppelwort) BCD-Daten und/oder Konstanten. Dd + 1 Dd (BCD) ÷ Dr + 1 Dr (BCD) R + 3 R + 2 R + 1 R (BCD) Rest Quotient	Ausgabe Erforderlich

3-10 Konvertierungsbefehle

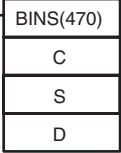
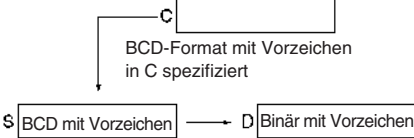
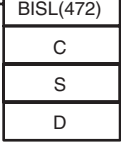
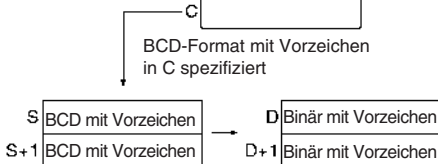
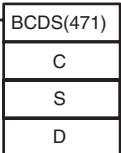
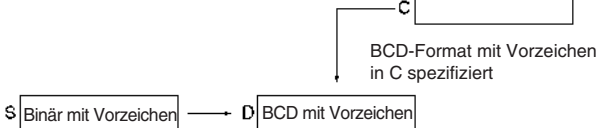
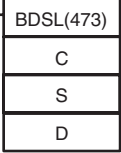
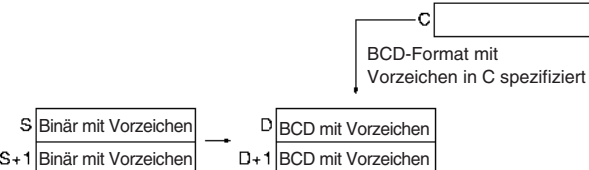
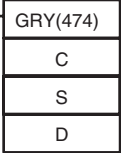
Befehl	AWL Code	Symbol/Operand	Funktion	Position Ausführungsbedingung
BCD-TO-BINARY (BCD zu Binär)	BIN @BIN 023	BIN(023) S R S: Quellwort R: Ergebniswort	Konvertiert BCD-Daten in Binärdaten. S (BCD) → R (BIN)	Ausgabe Erforderlich
DOUBLE BCD-TO-DOUBLE BINARY (BCD-Doppelwort zu Binär-Doppelwort)	BINL @BINL 058	BINL(058) S R S: Erstes Quellwort R: Erstes Ergebniswort	Konvertiert 8-stellige BCD-Daten in 8-stellige Hexadezimaldaten (32 Bit binär). S (BCD) → R (BIN) S+1 (BCD) → R+1 (BIN)	Ausgabe Erforderlich

Befehl	Symbol/Operand	Funktion	Position Ausführungsbedingung
BINARY-TO-BCD (Binär zu BCD) BCD @BCD 024	 S: Quellwort R: Ergebniswort	Konvertiert ein Wort Binärdaten in ein Wort BCD-Daten. $S \text{ (BIN)} \longrightarrow R \text{ (BCD)}$	Ausgabe Erforderlich
DOUBLE BINARY-TO-DOUBLE BCD (Binär-Doppelwort zu BCD-Doppelwort) BCDL @BCDL 059	 S: Erstes Quellwort R: Erstes Ergebniswort	Konvertiert 8-stellige Hexadezimaldaten (32 Bit binär) in 8-stellige BCD-Daten. $S \text{ (BIN)} \longrightarrow R \text{ (BCD)}$ $S+1 \text{ (BIN)} \longrightarrow R+1 \text{ (BCD)}$	Ausgabe Erforderlich
2'S COMPLEMENT (2er-Komplement) NEG @NEG 160	 S: Quellwort R: Ergebniswort	Berechnet das Zweierkomplement eines Worts Hexadezimaldaten. $\overline{(S)} \longrightarrow (R)$ Zweierkomplement (Komplement + 1)	Ausgabe Erforderlich
DOUBLE 2'S COMPLEMENT (Doppelwort-2er-Komplement) NEGL @NEGL 161	 S: Erstes Quellwort R: Erstes Ergebniswort	Berechnet das Zweierkomplement von zwei Wörtern Hexadezimaldaten. $\overline{(S+1, S)} \longrightarrow (R+1, R)$ Zweierkomplement (Komplement + 1)	Ausgabe Erforderlich
16-BIT TO 32-BIT SIGNED BINARY (Vorzeichenbehaftete 16-Bits in 32-Bits Binärwerte) SIGN @SIGN 600	 S: Quellwort R: Erstes Ergebniswort	Erweitert einen 16-Bit-Binärwert mit Vorzeichen in sein 32-Bit-Äquivalent.  MSB = 1: FFFF Hex MSB = 0: 0000 Hex D+1 D D = Inhalt von S	Ausgabe Erforderlich

Befehl	Symbol/Operand	Funktion	Position Ausführungs- bedingung
DATA DECODER (Daten-Dekoder) MLPX @MLPX 076	MLPX(076) S C R S: Quellwort C: Steuerwort R: Erstes Ergebniswort	Liest den numerischen Wert der spezifizierten Stelle (oder des Bytes) im Quellwort, setzt das entsprechende Bit im Ergebniswort (oder dem 16-Wort-Bereich) auf EIN und setzt alle anderen Bits im Ergebniswort (oder dem 16-Wort-Bereich) auf AUS. Konvertierung von 4 zu 16 Bit Konvertierung von 8 zu 256 Bit	Ausgabe Erforderlich

Befehl	Symbol/Operand	Funktion	Position Ausführungsbedingung
DATA ENCODER (Daten-Encoder) DMPX @DMPX 077	DMPX(077) S R C S: Erstes Quellwort R: Ergebniswort C: Steuerwort	Sucht den Ort des ersten oder letzten auf EIN gesetzten Bits des Quellworts (oder des 16-Wort-Bereichs) und schreibt diesen Wert in die angegebene Stelle (oder das Byte) im Ergebniswort Konvertierung von 16 zu 4 Bit C 0 1;0 1 n Sucht Bit höchster Wertigkeit (Höchste Bit-Adresse) S 15 p n 0 1 l = 1 (Konvertiert 2 Worte) Dekodierung von 16 zu 4 Bit (Position des äußerst links gelegenen Bits (m) wird in R geschrieben.) Bit höchster Wertigkeit (links) Bit niedrigster Wertigkeit (rechts) n=2 (Start mit Stelle 2) R p m Konvertierung von 256 zu 8 Bit C 0 1;0 1 n l = 0 (Einen 16-Wort-Bereich konvertieren) S 15 0 31 16 S+1 239 224 255 240 S+14 S+15 1 Sucht Bit höchster Wertigkeit (Höchste Bit-Adresse) Dekodierung von 256 zu 8 Bits (Die Position des im 16-Wort-Bereich (m) äußerst links liegenden Bits wird in R geschrieben.) n=1 (Start mit Byte 1) R m	Ausgabe Erforderlich
ASCII CONVERT (ASCII-Konvertierung) ASC @ASC 086	ASC(086) S Di D S: Quellwort Di: Stellen-Steuerwort D: Erstes Zielwort	Konvertiert 4-Bit-Hexadezimalstellen im Quellwort in die entsprechenden 8-Bit ASCII-Äquivalente. Di 0 1;0 n m Erste zu konvertierende Stelle S 1 2 3 m HEX ↓ ASCII Anzahl der Stellen (n+1) Links (1) Rechts (0) D 33 31 32	Ausgabe Erforderlich

Befehl	Symbol/Operand	Funktion	Position Ausführungsbedingung
ASCII TO HEX (ASCII-zu-HEX) HEX @HEX 162	HEX(162) S Di D S: Erstes Quellwort Di: Stellen-Steuerwort D: Zielwort	Konvertiert bis zu 4 Bytes ASCII-Daten im Quellwort in die Hexadezimal-Äquivalente und schreibt diese Stellen in das spezifizierte Zielwort. C: 0021 Di: 0 0/1 n m S: 33 32 34 D: 4 3 2 m	Ausgabe Erforderlich
COLUMN TO LINE (Spalte zu Zeile) LINE @LINE 063	LINE(063) S N D S: Erstes Quellwort N: Bit-Nummer D: Zielwort	Konvertiert eine Spalte Bits eines 16-Worte-Bereichs (die gleiche Bit-Nummer in 16 aufeinander folgenden Worten) in die 16 Bits des Zielworts. S: 0 0 0 1 1 1 1 0 0 0 1 0 0 0 0 1 S+1: 1 1 0 1 0 0 1 0 0 1 1 1 0 0 0 1 S+2: 0 0 0 1 1 0 1 1 0 0 1 0 0 1 1 1 S+3: 1 0 0 0 0 0 1 1 0 0 0 0 0 1 1 1 S+15: 0 1 1 0 0 0 0 1 1 0 0 0 1 0 1 0 D: 0 . . . 0 1 1 1	Ausgabe Erforderlich
LINE TO COLUMN (Zeile zu Spalte) COLM @COLM 064	COLM(064) S D N S: Quellwort D: Erstes Zielwort N: Bit-Nummer	Konvertiert die 16 Bits eines Quellworts in eine Spalte mit Bits in einem 16-Worte-Bereich von Zielworten (die gleiche Bit-Nummer in 16 aufeinander folgenden Worten). S: 0 0 1 1 1 D: 0 0 0 0 1 1 1 0 0 0 1 0 0 0 0 1 D+1: 1 1 0 1 0 0 1 0 0 1 1 1 0 0 0 1 D+2: 0 0 0 1 1 0 1 1 0 0 1 0 0 1 1 1 D+3: 1 0 0 0 0 0 1 1 0 0 0 0 0 1 1 1 D+15: 0 1 1 1 0 0 0 1 1 0 0 0 1 0 1 0	Ausgabe Erforderlich

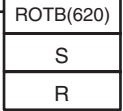
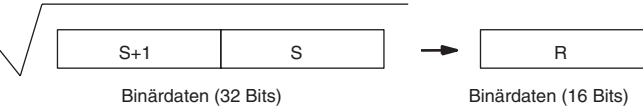
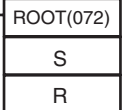
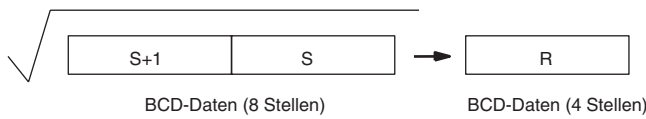
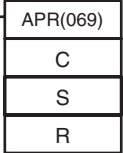
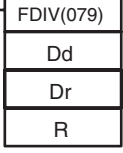
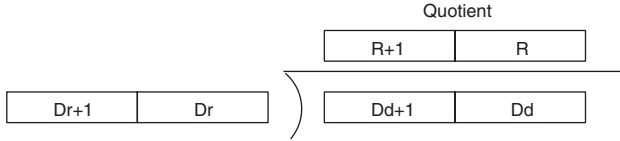
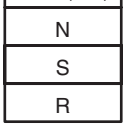
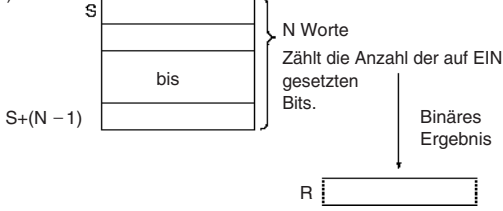
Befehl	Symbol/Operand	Funktion	Position Ausführungs- bedingung
SIGNED BCD-TO-BINARY (Vorzeichenbehaftetes BCD zu Binär) BINS @BINS 470	 C: Steuerwort S: Quellwort D: Zielwort	Konvertiert ein Wort BCD-Daten mit Vorzeichen in ein Wort Binärdaten mit Vorzeichen. 	Ausgabe Erforderlich
DOUBLE SIGNED BCD-TO-BINARY (Vorzeichenbehaftetes BCD-Doppelwort zu Binär) BISL @BISL 472	 C: Steuerwort S: Erstes Quellwort D: Erstes Zielwort	Konvertiert BCD-Doppelwort-Daten mit Vorzeichen zu Binär-Doppelwort-Daten mit Vorzeichen. 	Ausgabe Erforderlich
SIGNED BINARY-TO-BCD (Vorzeichenbehafteter Binärwert zu BCD) BCDS @BCDS 471	 C: Steuerwort S: Quellwort D: Zielwort	Konvertiert ein Wort Binärdaten mit Vorzeichen in ein Wort BCD-Daten mit Vorzeichen. 	Ausgabe Erforderlich
DOUBLE SIGNED BINARY-TO-BCD (Vorzeichenbehaftetes Binär-Doppelwort zu BCD) BDSL @BDSL 473	 C: Steuerwort S: Erstes Quellwort D: Erstes Zielwort	Konvertiert Binär-Doppelwort-Daten mit Vorzeichen zu BCD-Doppelwort-Daten mit Vorzeichen. 	Ausgabe Erforderlich
GRAY CODE CONVERSION (Gray-Code konvertieren) GRY 474 (nur CS/CJ-Serie CPU-Baugruppen Version 2.0 oder neuer, einschließlich CS1-H, CJ1-H und CJ1M CPU-Baugruppen ab Lot-Nummer 030201)	 C: Steuerwort S: Quellwort D: Erstes Zielwort	Konvertiert Gray-Code-Daten im spezifizierten Wort in Binär-, BCD- oder Winkeldaten (°) mit der spezifizierten Auflösung.	Ausgabe Erforderlich

3-11 Boolesche Befehle (Logikbefehle)

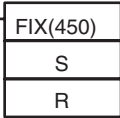
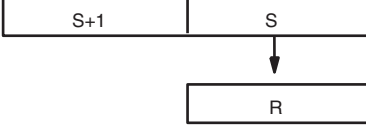
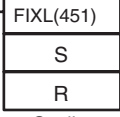
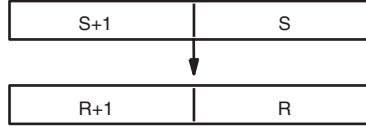
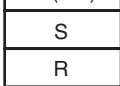
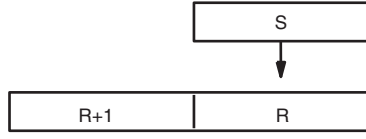
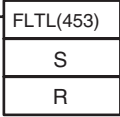
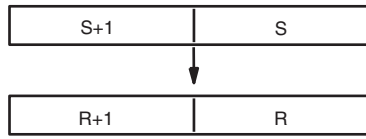
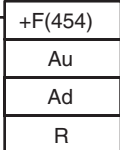
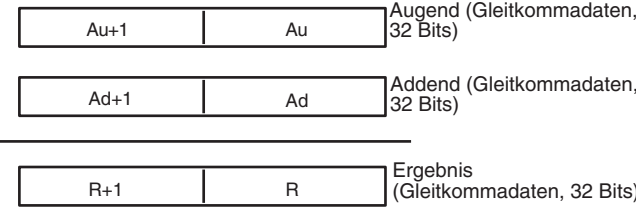
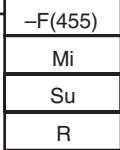
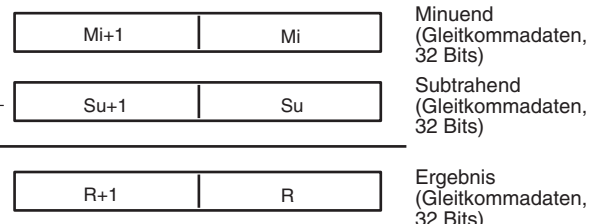
Befehl	AWL Code	Symbol/Operand	Funktion	Position Ausführungs- bedingung															
LOGICAL AND (Logisches AND- Wort)	ANDW @ANDW 034	ANDW(034) I₁ I₂ R I₁: Eingang 1 I₂: Eingang 2 R: Ergebniswort	Bildet eine logische AND-Verknüpfung zwischen entsprechenden Bits in Einzelworten von Wortdaten und/oder Konstanten. $I_1 \cdot I_2 \rightarrow R$ <table><tr><th>I₁</th><th>I₂</th><th>R</th></tr><tr><td>1</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>0</td><td>0</td><td>0</td></tr></table>	I ₁	I ₂	R	1	1	1	1	0	0	0	1	0	0	0	0	Ausgabe Erforderlich
I ₁	I ₂	R																	
1	1	1																	
1	0	0																	
0	1	0																	
0	0	0																	
DOUBLE LOGICAL AND (Logisches Doppelwort AND)	ANDL @ANDL 610	ANDL(610) I₁ I₂ R I₁: Eingang 1 I₂: Eingang 2 R: Ergebniswort	Bildet eine logische AND-Verknüpfung zwischen entsprechenden Bits in Doppelworten von Wortdaten und/oder Konstanten. $(I_1 \cdot I_1+1) \cdot (I_2 \cdot I_2+1) \rightarrow (R, R+1)$ <table><tr><th>I₁ · I₁+1</th><th>I₂ · I₂+1</th><th>R, R+1</th></tr><tr><td>1</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>0</td><td>0</td><td>0</td></tr></table>	I ₁ · I ₁ +1	I ₂ · I ₂ +1	R, R+1	1	1	1	1	0	0	0	1	0	0	0	0	Ausgabe Erforderlich
I ₁ · I ₁ +1	I ₂ · I ₂ +1	R, R+1																	
1	1	1																	
1	0	0																	
0	1	0																	
0	0	0																	
LOGICAL OR (Logisches OR)	ORW @ORW 035	ORW(035) I₁ I₂ R I₁: Eingang 1 I₂: Eingang 2 R: Ergebniswort	Bildet eine logische OR-Verknüpfung zwischen entsprechenden Bits in Einzelworten von Wortdaten und/oder Konstanten. $I_1 + I_2 \rightarrow R$ <table><tr><th>I₁</th><th>I₂</th><th>R</th></tr><tr><td>1</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>0</td><td>0</td><td>0</td></tr></table>	I ₁	I ₂	R	1	1	1	1	0	1	0	1	1	0	0	0	Ausgabe Erforderlich
I ₁	I ₂	R																	
1	1	1																	
1	0	1																	
0	1	1																	
0	0	0																	
DOUBLE LOGICAL OR (Logisches Doppelwort AND)	ORWL @ORWL 611	ORWL(611) I₁ I₂ R I₁: Eingang 1 I₂: Eingang 2 R: Ergebniswort	Bildet eine logische OR-Verknüpfung zwischen entsprechenden Bits in Doppelworten von Wortdaten und/oder Konstanten. $(I_1 \cdot I_1+1) + (I_2 \cdot I_2+1) \rightarrow (R, R+1)$ <table><tr><th>I₁ · I₁+1</th><th>I₂ · I₂+1</th><th>R, R+1</th></tr><tr><td>1</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>0</td><td>0</td><td>0</td></tr></table>	I ₁ · I ₁ +1	I ₂ · I ₂ +1	R, R+1	1	1	1	1	0	1	0	1	1	0	0	0	Ausgabe Erforderlich
I ₁ · I ₁ +1	I ₂ · I ₂ +1	R, R+1																	
1	1	1																	
1	0	1																	
0	1	1																	
0	0	0																	
EXCLUSIVE OR (Logisches XOR)	XORW @XORW 036	XORW(036) I₁ I₂ R I₁: Eingang 1 I₂: Eingang 2 R: Ergebniswort	Bildet eine logische Exklusiv-OR-Verknüpfung zwischen entsprechenden Bits in Einzelworten von Wortdaten und/oder Konstanten. $I_1 \cdot I_2 + I_1 \cdot \bar{I}_2 \rightarrow R$ <table><tr><th>I₁</th><th>I₂</th><th>R</th></tr><tr><td>1</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>0</td><td>0</td><td>0</td></tr></table>	I ₁	I ₂	R	1	1	0	1	0	1	0	1	1	0	0	0	Ausgabe Erforderlich
I ₁	I ₂	R																	
1	1	0																	
1	0	1																	
0	1	1																	
0	0	0																	

Befehl	AWL Code	Symbol/Operand	Funktion	Position Ausführungs- bedingung															
DOUBLE EXCLUSIVE OR (Doppelwort Exklusiv OR)	XORL @XORL 612	XORL(612) I₁ I₂ R I₁: Eingang 1 I₂: Eingang 2 R: Ergebniswort	Bildet eine logische exklusive OR-Verknüpfung zwischen entsprechenden Bits in Doppelworten von Wortdaten und/oder Konstanten. $(I_1 \cdot I_1 + 1) \cdot (\overline{I_2 \cdot I_2 + 1}) + (\overline{I_1 \cdot I_1 + 1}) \cdot (I_2 \cdot I_2 + 1) \rightarrow (R, R+1)$ <table><tr><th>I₁ · I₁ + 1</th><th>I₂ · I₂ + 1</th><th>R, R+1</th></tr><tr><td>1</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>0</td><td>0</td><td>0</td></tr></table>	I ₁ · I ₁ + 1	I ₂ · I ₂ + 1	R, R+1	1	1	0	1	0	1	0	1	1	0	0	0	Ausgabe Erforderlich
I ₁ · I ₁ + 1	I ₂ · I ₂ + 1	R, R+1																	
1	1	0																	
1	0	1																	
0	1	1																	
0	0	0																	
EXCLUSIVE NOR (Logisches XNOR)	XNRW @XNRW 037	XNRW(037) I₁ I₂ R I₁: Eingang 1 I₂: Eingang 2 R: Ergebniswort	Bildet eine logische exklusive NOR-Verknüpfung zwischen entsprechenden Einzelworten von Wortdaten und/oder Konstanten. $I_1 \cdot I_2 + \overline{I_1} \cdot \overline{I_2} \rightarrow R$ <table><tr><th>I₁</th><th>I₂</th><th>R</th></tr><tr><td>1</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>0</td><td>0</td><td>1</td></tr></table>	I ₁	I ₂	R	1	1	1	1	0	0	0	1	0	0	0	1	Ausgabe Erforderlich
I ₁	I ₂	R																	
1	1	1																	
1	0	0																	
0	1	0																	
0	0	1																	
DOUBLE EXCLUSIVE NOR (Doppelwort Exklusiv NOR)	XNRL @XNRL 613	XNRL(613) I₁ I₂ R I₁: Eingang 1 I₂: Eingang 2 R: Erstes Ergebniswort	Bildet eine logische exklusive NOR-Verknüpfung zwischen entsprechenden Bits in Doppelworten von Wortdaten und/oder Konstanten. $(I_1 \cdot I_1 + 1) \cdot (I_2 \cdot I_2 + 1) + (\overline{I_1 \cdot I_1 + 1}) \cdot (\overline{I_2 \cdot I_2 + 1}) \rightarrow (R, R+1)$ <table><tr><th>I₁ · I₁ + 1</th><th>I₂ · I₂ + 1</th><th>R, R+1</th></tr><tr><td>1</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>0</td><td>0</td><td>1</td></tr></table>	I ₁ · I ₁ + 1	I ₂ · I ₂ + 1	R, R+1	1	1	1	1	0	0	0	1	0	0	0	1	Ausgabe Erforderlich
I ₁ · I ₁ + 1	I ₂ · I ₂ + 1	R, R+1																	
1	1	1																	
1	0	0																	
0	1	0																	
0	0	1																	
COMPLEMENT (Komplement)	COM @COM 029	COM(029) Wd Wd: Wort	Setzt in Wd alle auf EIN gesetzten Bits auf AUS und alle auf AUS gesetzten Bits auf EIN. $\overline{Wd} \rightarrow Wd: 1 \rightarrow 0 \text{ und } 0 \rightarrow 1$	Ausgabe Erforderlich															
DOUBLE COMPLEMENT (Doppelwort- Komplement)	COML @COML 614	COML(614) Wd Wd: Wort	Setzt in Wd und Wd+1 alle auf EIN gesetzten Bits auf AUS und alle auf AUS gesetzten Bits auf EIN. $(\overline{Wd+1}, \overline{Wd}) \rightarrow (Wd+1, Wd)$	Ausgabe Erforderlich															

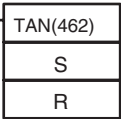
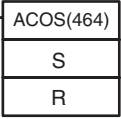
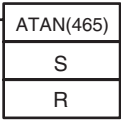
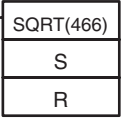
3-12 Spezielle arithmetische Befehle

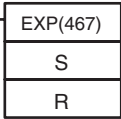
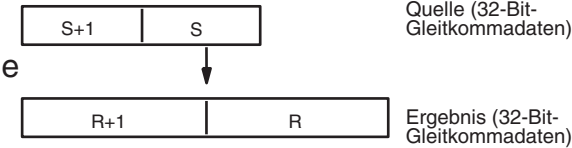
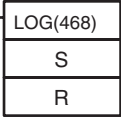
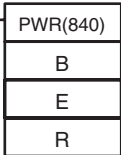
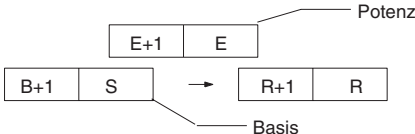
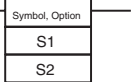
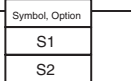
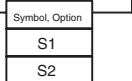
Befehl	Symbol/Operand	Funktion	Position Ausführungs- bedingung
BINARY ROOT (Binärwurzel) ROTB @ROTB 620	 S: Erstes Quellwort R: Ergebniswort	Berechnet die Quadratwurzel aus dem 32-Bit-Binärinhalt der spezifizierten Worte und gibt den ganzzahligen Teil des Ergebnisses in das spezifizierte Ergebniswort aus. 	Ausgabe Erforderlich
BCD SQUARE ROOT (BCD- Quadratwurzel) ROOT @ROOT 072	 S: Erstes Quellwort R: Ergebniswort	Berechnet die Quadratwurzel aus einer 8-stelligen BCD-Zahl und gibt den ganzzahligen Teil des Ergebnisses in das spezifizierte Ergebniswort aus. 	Ausgabe Erforderlich
ARITHMETIC PROCESS (Arithmetische Verarbeitung) APR @APR 069	 C: Steuerwort S: Quelldaten R: Ergebniswort	Berechnet den Sinus, Kosinus oder eine lineare Extrapolation der Quelldaten. Mit der linearen Extrapolation können alle Beziehungen zwischen X und Y durch Geradensegmente angenähert werden.	Ausgabe Erforderlich
FLOATING POINT DIVIDE (Fließkomma- Division) FDIV @FDIV 079	 Dd: Erstes Dividend- Wort Dr: Erstes Divisor- Wort R: Erstes Ergebniswort	Dividiert eine 7-stellige Gleitkommazahl durch eine andere. Die Gleitkommazahlen werden in wissenschaftlicher Schreibweise ausgedrückt (Mantisse und Exponent). 	Ausgabe Erforderlich
BIT COUNTER (Bit-Zähler) BCNT @BCNT 067	 N: Anzahl der Worte S: Erstes Quellwort R: Ergebniswort	Zählt die Gesamtanzahl der auf EIN gesetzten Bits in dem/den spezifizierten Wort(en). 	Ausgabe Erforderlich

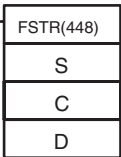
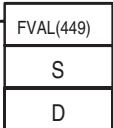
3-13 Gleitkomma-Arithmetikbefehle

Befehl	Symbol/Operand	Funktion	Position Ausführungs- bedingung
FLOATING TO 16-BIT (Fließkomma zu 16 Bits) FIX @FIX 450	 S: Erstes Quellwort R: Ergebniswort	Konvertiert einen 32-Bit-Gleitkommazahl-Wert in 16-Bit-Binärdaten mit Vorzeichen, und gibt das Ergebnis in das spezifizierte Ergebniswort aus. 	Ausgabe Erforderlich
FLOATING TO 32-BIT (Fließkomma zu 32 Bits) FIXL @FIXL 451	 S: Erstes Quellwort R: Erstes Ergebniswort	Konvertiert einen 32-Bit-Gleitkommazahl-Wert in 32-Bit-Binärdaten mit Vorzeichen, und gibt das Ergebnis in die spezifizierten Ergebnisworte aus. 	Ausgabe Erforderlich
16-BIT TO FLOATING (16 Bits zu Fließkomma) FLT @FLT 452	 S: Quellwort R: Erstes Ergebniswort	Konvertiert einen 16-Bit-Binärwert mit Vorzeichen in 32-Bit-Gleitkommadaten und gibt das Ergebnis in die spezifizierten Ergebnisworte aus. 	Ausgabe Erforderlich
32-BIT TO FLOATING (32 Bits zu Fließkomma) FLTL @FLTL 453	 S: Erstes Quellwort R: Erstes Ergebniswort	Konvertiert einen 32-Bit-Binärwert mit Vorzeichen in 32-Bit-Gleitkommadaten und gibt das Ergebnis in die spezifizierten Ergebnisworte aus. 	Ausgabe Erforderlich
FLOATING-POINT ADD (Fließkomma-Addition) +F @+F 454	 Au: Erstes Augend-Wort Ad: Erstes Addend-Wort R: Erstes Ergebniswort	Addiert zwei 32-Bit-Gleitkommazahlen und gibt das Ergebnis in die spezifizierten Ergebnisworte aus. 	Ausgabe Erforderlich
FLOATING-POINT SUBTRACT (Fließkomma-Subtraktion) -F @-F 455	 Mi: Erstes Minuend-Wort Su: Erstes Subtrahend-Wort R: Erstes Ergebniswort	Subtrahiert eine 32-Bit-Gleitkommazahl von einer anderen und gibt das Ergebnis in die spezifizierten Ergebnisworte aus. 	Ausgabe Erforderlich

Befehl	Symbol/Operand	Funktion	Position Ausführungs- bedingung
FLOATING- POINT MULTIPLY (Fließkomma- Multiplikation) *F @ *F 456	*F(456) Md Mr R Md: Erstes Multiplikand-Wort Mr: Erstes Multiplikator-Wort R: Erstes Ergebniswort	Multipliziert zwei 32-Bit-Gleitkommazahlen miteinander und gibt das Ergebnis in die spezifizierten Ergebnisworte aus. Md+1 Md Mr+1 Mr R+1 R Multiplikand (Gleitkommandaten, 32 Bits) Multiplikator (Gleitkommandaten, 32 Bits) Ergebnis (Gleitkommandaten, 32 Bits)	Ausgabe Erforderlich
FLOATING- POINT DIVIDE (Fließkomma- Division) /F @ /F 457	/F(457) Dd Dr R Dd: Erstes Dividend-Wort Dr: Erstes Divisor- Wort R: Erstes Ergebniswort	Dividiert eine 32-Bit-Gleitkommazahl durch eine andere und gibt das Ergebnis in die spezifizierten Ergebnisworte aus. Dd+1 Dd Dr+1 Dr R+1 R Dividend (Gleitkommandaten, 32 Bits) Divisor (Gleitkommandaten, 32 Bits) Ergebnis (Gleitkommandaten, 32 Bits)	Ausgabe Erforderlich
DEGREES TO RADIANS (Grad in Radianen) RAD @ RAD 458	RAD(458) S R S: Erstes Quellwort R: Erstes Ergebniswort	Konvertiert eine 32-Bit-Gleitkommazahl von Grad zu Radianen und gibt das Ergebnis in die spezifizierten Ergebnisworte aus. S+1 S R+1 R Quelle (Grad, 32-Bit-Gleitkommandaten) Ergebnis (Radianen, 32-Bit-Gleitkommandaten)	Ausgabe Erforderlich
RADIANS TO DEGREES (Radi- anten in Grad) DEG @ DEG 459	DEG(459) S R S: Erstes Quellwort R: Erstes Ergebniswort	Konvertiert eine 32-Bit-Gleitkommazahl von Radianen zu Grad und gibt das Ergebnis in die spezifizierten Ergebnisworte aus. S+1 S R+1 R Quelle (Radianen, 32-Bit-Gleitkommandaten) Ergebnis (Grad, 32-Bit-Gleitkommandaten)	Ausgabe Erforderlich
SINE (Sinus) SIN @ SIN 460	SIN(460) S R S: Erstes Quellwort R: Erstes Ergebniswort	Berechnet den Sinus einer 32-Bit-Gleitkommazahl (in Radianen) und gibt das Ergebnis in die spezifizierten Ergebnisworte aus. SIN (S+1 S) R+1 R Quelle (32-Bit- Gleitkommandaten) Ergebnis (32-Bit- Gleitkommandaten)	Ausgabe Erforderlich
COSINE (Kosinus) COS @ COS 461	COS(461) S R S: Erstes Quellwort R: Erstes Ergebniswort	Berechnet den Kosinus einer 32-Bit-Gleitkommazahl (in Radianen) und gibt das Ergebnis in die spezifizierten Ergebnisworte aus. COS (S+1 S) R+1 R Quelle (32-Bit- Gleitkommandaten) Ergebnis (32-Bit- Gleitkommandaten)	Ausgabe Erforderlich

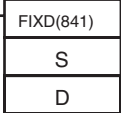
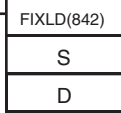
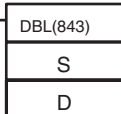
Befehl	Symbol/Operand	Funktion	Position Ausführungs- bedingung
TANGENT (Tangens) TAN @ TAN 462	 S: Erstes Quellwort R: Erstes Ergebniswort	Berechnet den Tangens einer 32-Bit-Gleitkommazahl (in Radianten) und gibt das Ergebnis in die spezifizierten Ergebnisworte aus. $\text{TAN} \left(\begin{array}{ c c } \hline S+1 & S \\ \hline \end{array} \right) \quad \text{Quelle (32-Bit-Gleitkommaten)}$ $\begin{array}{ c c } \hline R+1 & R \\ \hline \end{array} \quad \text{Ergebnis (32-Bit-Gleitkommaten)}$	Ausgabe Erforderlich
ARC SINE (Arkussinus) ASIN @ ASIN 463	 S: Erstes Quellwort R: Erstes Ergebniswort	Berechnet den Arkussinus einer 32-Bit-Gleitkommazahl und gibt das Ergebnis in die spezifizierten Ergebnisworte aus. (Die Arkussinusfunktion ist die Umkehrung der Sinusfunktion. Sie ergibt wieder den Winkel, der einem gegebenen Sinuswert zwischen -1 und 1 entspricht.) $\text{SIN}^{-1} \left(\begin{array}{ c c } \hline S+1 & S \\ \hline \end{array} \right) \quad \text{Quelle (32-Bit-Gleitkommaten)}$ $\begin{array}{ c c } \hline R+1 & R \\ \hline \end{array} \quad \text{Ergebnis (32-Bit-Gleitkommaten)}$	Ausgabe Erforderlich
ARC COSINE (Arkuskosinus) ACOS @ ACOS 464	 S: Erstes Quellwort R: Erstes Ergebniswort	Berechnet den Arkuskosinus einer 32-Bit-Gleitkommazahl und gibt das Ergebnis in die spezifizierten Ergebnisworte aus. (Die Arkuskosinusfunktion ist die Umkehrung der Kosinusfunktion. Sie ergibt wieder den Winkel, der einem gegebenen Kosinuswert zwischen -1 und 1 entspricht.) $\text{COS}^{-1} \left(\begin{array}{ c c } \hline S+1 & S \\ \hline \end{array} \right) \quad \text{Quelle (32-Bit-Gleitkommaten)}$ $\begin{array}{ c c } \hline R+1 & R \\ \hline \end{array} \quad \text{Ergebnis (32-Bit-Gleitkommaten)}$	Ausgabe Erforderlich
ARC TANGENT (Arkustangens) ATAN @ ATAN 465	 S: Erstes Quellwort R: Erstes Ergebniswort	Berechnet den Arkustangens einer 32-Bit-Gleitkommazahl und gibt das Ergebnis in die spezifizierten Ergebnisworte aus. (Die Arkustangensfunktion ist die Umkehrung der Tangensfunktion. Sie ergibt wieder den Winkel, der einem gegebenen Tangenswert entspricht.) $\text{TAN}^{-1} \left(\begin{array}{ c c } \hline S+1 & S \\ \hline \end{array} \right) \quad \text{Quelle (32-Bit-Gleitkommaten)}$ $\begin{array}{ c c } \hline R+1 & R \\ \hline \end{array} \quad \text{Ergebnis (32-Bit-Gleitkommaten)}$	Ausgabe Erforderlich
SQUARE ROOT (Quadratwurzel) SQRT @ SQRT 466	 S: Erstes Quellwort R: Erstes Ergebniswort	Berechnet die Quadratwurzel einer 32-Bit-Gleitkommazahl und gibt das Ergebnis in die spezifizierten Ergebnisworte aus. $\sqrt{\begin{array}{ c c } \hline S+1 & S \\ \hline \end{array}} \quad \text{Quelle (32-Bit-Gleitkommaten)}$ $\begin{array}{ c c } \hline R+1 & R \\ \hline \end{array} \quad \text{Ergebnis (32-Bit-Gleitkommaten)}$	Ausgabe Erforderlich

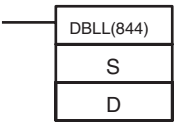
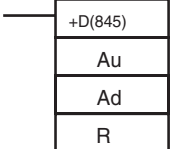
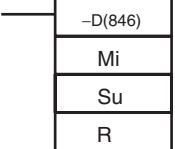
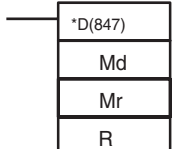
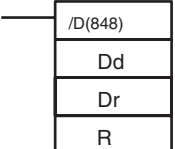
Befehl	Symbol/Operand	Funktion	Position Ausführungsbedingung
EXPONENT (Exponent) EXP @EXP 467	 S: Erstes Quellwort R: Erstes Ergebniswort	Berechnet die natürliche Exponentialgröße (Basis e) einer 32-Bit-Gleitkommazahl und gibt das Ergebnis in die spezifizierten Ergebnisworte aus.  Quelle (32-Bit-Gleitkommandaten) Ergebnis (32-Bit-Gleitkommandaten)	Ausgabe Erforderlich
LOGARITHM (Logarithmus) LOG @LOG 468	 S: Erstes Quellwort R: Erstes Ergebniswort	Berechnet den natürlichen Logarithmus (Basis e) einer 32-Bit-Gleitkommazahl und gibt das Ergebnis in die spezifizierten Ergebnisworte aus.  Quelle (32-Bit-Gleitkommandaten) Ergebnis (32-Bit-Gleitkommandaten)	Ausgabe Erforderlich
EXPONENTIAL POWER (Potenzfunktion) PWR @PWR 840	 B: Erstes Basiswort E: Erstes Exponenten-Wort R: Erstes Ergebniswort	Erhebt eine 32-Bit-Gleitkommazahl in die Potenz einer anderen 32-Bit-Gleitkommazahl.  Potenz Basis	Ausgabe Erforderlich
FLOATING SYMBOL COMPARISON (Fließkomma-Symbolvergleich) (nur CS1-H, CJ1-H, CJ1M und CS1D) LD, AND oder OR + =F (329), <>F (330), <F (331), <=F (332), >F (333), oder >=F (334)	Verwenden von LD:  Verwenden von AND:  Verwenden von OR:  S1: Vergleichsdaten 1 S2: Vergleichsdaten 2	Vergleicht die spezifizierten Daten einfacher Präzision (32 Bit) oder Konstanten und erzeugt eine EIN-Ausführungsbedingung, wenn das Vergleichsergebnis wahr ist. Mit den Gleitkomma-Symbolvergleichsbefehlen können drei Symbolarten verwendet werden: LD (Laden), AND und OR.	LD: nicht erforderlich AND oder OR: erforderlich

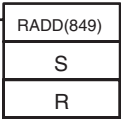
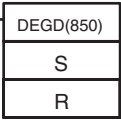
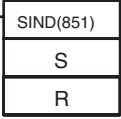
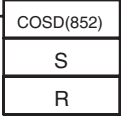
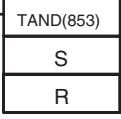
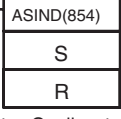
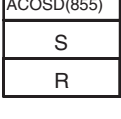
Befehl AWL Code	Symbol/Operand	Funktion	Position Ausführungs- bedingung
FLOATING- POINT TO ASCII (Fließkomma zu Zeichenkette) (nur CS1-H, CJ1-H, CJ1M und CS1D) FSTR @FSTR 448	 S: Erstes Quellwort C: Steuerwort D: Zielwort	Konvertiert die spezifizierten Gleitkommadaten einfacher Präzision (32-Bit Dezimalkomma- oder Exponentialformat) in Textzeichenketten-daten (ASCII) und gibt das Ergebnis in das Zielwort aus.	Ausgabe erforderlich
ASCII TO FLOATING- POINT (Zeichen- kette zu Fließ- komma) (nur CS1-H, CJ1-H, CJ1M und CS1D) FVAL @FVAL 449	 S: Quellwort D: Erstes Zielwort	Konvertiert die spezifizierte Textzeichenketten-Schreibweise (ASCII) der Gleitkommadaten einfacher Präzision (Dezimalkomma- oder Exponentialformat) zu 32-Bit Gleitkommadaten einfacher Genauigkeit und gibt das Ergebnis in die Zielworte aus.	Ausgabe erforderlich

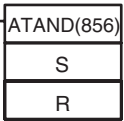
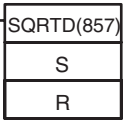
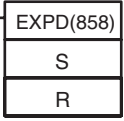
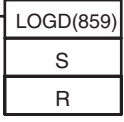
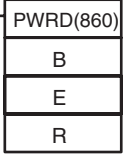
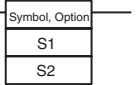
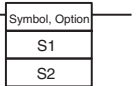
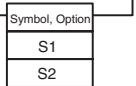
3-14 Gleitkomma-Arithmetikbefehle mit doppelter Präzision

Die Gleitkomma-Arithmetikbefehle mit doppelter Präzision werden nur durch die CPU-Baugruppen CS1-H, CJ1-H, CJ1M und CS1D unterstützt.

Befehl AWL Code	Symbol/Operand	Funktion	Position Ausführungs- bedingung
DOUBLE FLOATING TO 16-BIT BINARY (Doppeltgenaue Fließkomma zu 16 Bits) FIXD @FIXD 841	 S: Erstes Quellwort D: Zielwort	Konvertiert die spezifizierten Gleitkommadaten doppelter Präzision (64 Bit) zu 16-Bit-Daten mit Vorzeichen und gibt das Ergebnis in das Zielwort aus.	Ausgabe Erforderlich
DOUBLE FLOATING TO 32-BIT BINARY (Doppeltgenaue Fließkomma zu 32 Bits) FIXLD @FIXLD 842	 S: Erstes Quellwort D: Erstes Zielwort	Konvertiert die spezifizierten Gleitkommadaten doppelter Präzision (64 Bit) zu 32-Bit-Daten mit Vorzeichen und gibt das Ergebnis in die Zielworte aus.	Ausgabe Erforderlich
16-BIT BINARY TO DOUBLE FLOATING (16 Bits zu Doppeltgenaue Fließkomma) DBL @DBL 843	 S: Quellwort D: Erstes Zielwort	Konvertiert die spezifizierten 16-Bit-Binärdaten mit Vorzeichen zu Gleitkommadaten doppelter Präzision (64 Bit) und gibt das Ergebnis in die Zielworte aus.	Ausgabe Erforderlich

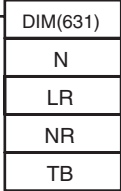
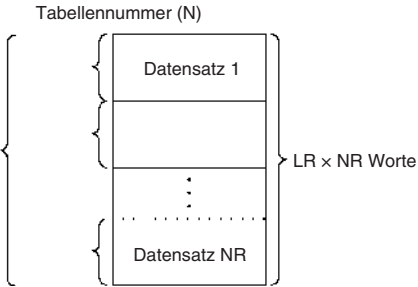
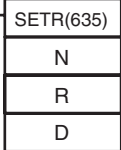
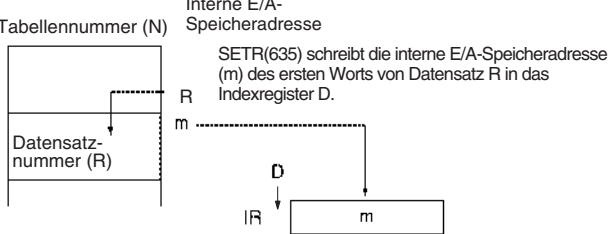
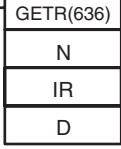
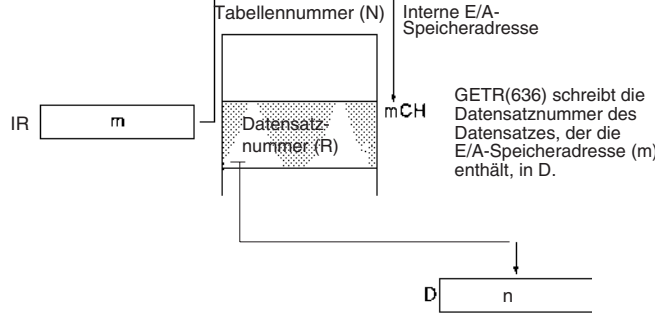
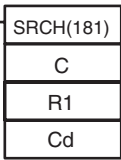
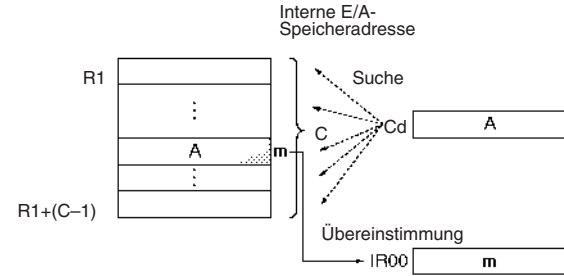
Befehl AWL Code	Symbol/Operand	Funktion	Position Ausführungs- bedingung
32-BIT BINARY TO DOUBLE FLOATING (32 Bits zu Doppeltgenaue Fließkomma) DBLL @DBLL 844	 S: Erstes Quellwort D: Erstes Zielwort	Konvertiert die spezifizierten 32-Bit-Binärdaten mit Vorzeichen zu Gleitkommadaten doppelter Präzision (64 Bit) und gibt das Ergebnis in die Zielworte aus.	Ausgabe Erforderlich
DOUBLE FLOATING- POINT ADD (Doppeltgenaue Fließkomma- Addition) +D @+D 845	 Au: Erstes Augend- Wort Ad: Erstes Addend- Wort R: Erstes Ergebnis- wort	Addiert die spezifizierten Gleitkommawerte doppelter Präzision (je 64 Bit) und gibt das Ergebnis in die Ergebnisworte aus.	Ausgabe Erforderlich
DOUBLE FLOATING- POINT SUB- TRACT (Doppeltgenaue Fließkomma- Subtraktion) -D @-D 846	 Mi: Erstes Minuend- Wort Su: Erstes Sub- trahendwort R: Erstes Ergebnis- wort	Subtrahiert die spezifizierten Gleitkommawerte doppelter Präzision (je 64 Bit) voneinander und gibt das Ergebnis in die Ergebnisworte aus.	Ausgabe Erforderlich
DOUBLE FLOATING- POINT MULTI- PLY (Doppeltgenaue Fließkomma- Multiplikation) *D @*D 847	 Md: Erstes Multiplikand-Wort Mr: Erstes Multiplikator-Wort R: Erstes Ergebniswort	Multipliziert die spezifizierten Gleitkommawerte doppelter Präzision (je 64 Bit) miteinander und gibt das Ergebnis in die Ergebnisworte aus.	Ausgabe Erforderlich
DOUBLE FLOATING- POINT DIVIDE (Doppeltgenaue Fließkomma- Division) /D @/D 848	 Dd: Erstes Dividend-Wort Dr: Erstes Divisor-Wort R: Erstes Ergebniswort	Dividiert die spezifizierten Gleitkommawerte doppelter Präzision (je 64 Bit) durcheinander und gibt das Ergebnis in die Ergebnisworte aus.	Ausgabe Erforderlich

Befehl AWL Code	Symbol/Operand	Funktion	Position Ausführungs- bedingung
DOUBLE DEGREES TO RADIANS (Grad nach Radiant Doppeltgenaue Fließkomma) RADD @ RADD 849	 S: Erstes Quellwort R: Erstes Ergebniswort	Konvertiert die spezifizierten Gleitkommatdaten doppelter Präzision (64 Bit) von Grad in Radianten und gibt das Ergebnis in die Ergebnisworte aus.	Ausgabe Erforderlich
DOUBLE RADIANS TO DEGREES (Radiant zu Grad, Doppeltgenaue Fließkomma) DEGD @ DEGD 850	 S: Erstes Quellwort R: Erstes Ergebniswort	Konvertiert die spezifizierten Gleitkommatdaten doppelter Präzision (64 Bit) von Radianten in Grad und gibt das Ergebnis in die Ergebnisworte aus.	Ausgabe Erforderlich
DOUBLE SINE (Sinus, Doppeltgenaue Fließkomma) SIND @ SIND 851	 S: Erstes Quellwort R: Erstes Ergebniswort	Berechnet den Sinus des Winkels (Radianten) in den spezifizierten Gleitkommatdaten doppelter Präzision (64 Bit) und gibt das Ergebnis in die Ergebnisworte aus.	Ausgabe Erforderlich
DOUBLE COSINE (Kosinus, Doppeltgenaue Fließkomma) COSD @ COSD 852	 S: Erstes Quellwort R: Erstes Ergebniswort	Berechnet den Kosinus des Winkels (Radianten) in den spezifizierten Gleitkommatdaten doppelter Präzision (64 Bit) und gibt das Ergebnis in die Ergebnisworte aus.	Ausgabe Erforderlich
DOUBLE TANGENT (Tangens, Doppeltgenaue Fließkomma) TAND @ TAND 853	 S: Erstes Quellwort R: Erstes Ergebniswort	Berechnet den Tangens des Winkels (Radianten) in den spezifizierten Gleitkommatdaten doppelter Präzision (64 Bit) und gibt das Ergebnis in die Ergebnisworte aus.	Ausgabe Erforderlich
DOUBLE ARC SINE (Arkus-Sinus, Doppeltgenaue Fließkomma) ASIND @ ASIND 854	 S: Erstes Quellwort R: Erstes Ergebniswort	Berechnet den Winkel (in Radianten) aus dem Sinuswert der spezifizierten Gleitkommatdaten doppelter Präzision (64 Bit) und gibt das Ergebnis in die Ergebnisworte aus. (Die Arkussinusfunktion ist die Umkehrung der Sinusfunktion. Sie ergibt wieder den Winkel, der einem gegebenen Sinuswert zwischen -1 und 1 entspricht.)	Ausgabe Erforderlich
DOUBLE ARC COSINE (Arkus-Kosinus, Doppeltgenaue Fließkomma) ACOSD @ ACOSD 855	 S: Erstes Quellwort R: Erstes Ergebniswort	Berechnet den Winkel (in Radianten) aus dem Kosinuswert der spezifizierten Gleitkommatdaten doppelter Präzision (64 Bit) und gibt das Ergebnis in die Ergebnisworte aus. (Die Arkuskosinusfunktion ist die Umkehrung der Kosinusfunktion. Sie ergibt wieder den Winkel, der einem gegebenen Kosinuswert zwischen -1 und 1 entspricht.)	Ausgabe Erforderlich

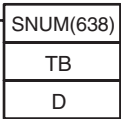
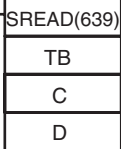
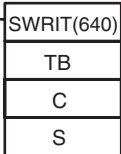
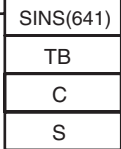
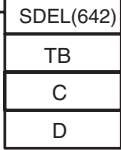
Befehl AWL Code	Symbol/Operand	Funktion	Position Ausführungs- bedingung
DOUBLE ARC TANGENT (Arkus-Tangens, Doppeltgenaue Fließkomma) ATAND @ ATAND 856	 S: Erstes Quellwort R: Erstes Ergebniswort	Berechnet den Winkel (in Radianten) aus dem Tangenswert der spezifizierten Gleitkommadaten doppelter Präzision (64 Bit) und gibt das Ergebnis in die Ergebnisworte aus. (Die Arkustangensfunktion ist die Umkehrung der Tangensfunktion. Sie ergibt wieder den Winkel, der einem gegebenen Tangenswert entspricht.)	Ausgabe Erforderlich
DOUBLE SQUARE ROOT (Quadratwurzel, Doppeltgenaue Fließkomma) SQRTD @ SQRTD 857	 S: Erstes Quellwort R: Erstes Ergebniswort	Berechnet die Quadratwurzel aus den spezifizierten Gleitkommadaten doppelter Präzision (64 Bit) und gibt das Ergebnis in die Ergebnisworte aus.	Ausgabe Erforderlich
DOUBLE EXPONENT (Exponent, Doppeltgenaue Fließkomma) EXPD @ EXPD 858	 S: Erstes Quellwort R: Erstes Ergebniswort	Berechnet den natürlichen Exponenten (zur Basis e) der spezifizierten Gleitkommadaten doppelter Präzision (64 Bit) und gibt das Ergebnis in die Ergebnisworte aus.	Ausgabe Erforderlich
DOUBLE LOGARITHM (Logarithmus, Doppeltgenaue Fließkomma) LOGD @ LOGD 859	 S: Erstes Quellwort R: Erstes Ergebniswort	Berechnet den natürlichen Logarithmus (Basis e) aus den spezifizierten Gleitkommadaten doppelter Präzision (64 Bit) und gibt das Ergebnis in die Ergebnisworte aus.	Ausgabe Erforderlich
DOUBLE EXPONENTIAL POWER (Potenzfunktion, Doppeltgenaue Fließkomma) PWRD @ PWRD 860	 B: Erstes Basiswort E: Erstes Exponenten-Wort R: Erstes Ergebniswort	Erhebt eine Gleitkommazahl doppelter Präzision (64 Bit) in die Potenz einer anderen Gleitkommazahl doppelter Präzision und gibt das Ergebnis in die Ergebnisworte aus.	Ausgabe Erforderlich
DOUBLE SYMBOL COMPARISON (Symbolvergleich, Doppeltgenaue Fließkomma) LD, AND oder OR + =D (335), <>D (336), <D (337), <=D (338), >D (339), oder >=D (340)	Verwenden von LD:  Verwenden von AND:  Verwenden von OR:  S1: Vergleichsdaten 1 S2: Vergleichsdaten 2	Vergleicht die spezifizierten Daten doppelter Präzision (64 Bit) und erzeugt eine EIN-Ausführungsbedingung, wenn das Vergleichsergebnis wahr ist. Mit den Gleitkomma-Symbolvergleichsbefehlen können drei Symbolarten verwendet werden: LD (Load), AND und OR.	LD: nicht erforderlich AND oder OR: erforderlich

3-15 Tabellendaten-Verarbeitungsbefehle

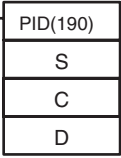
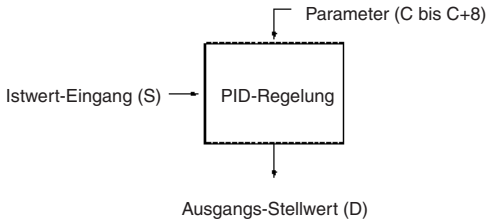
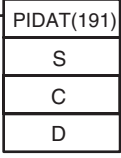
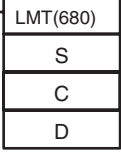
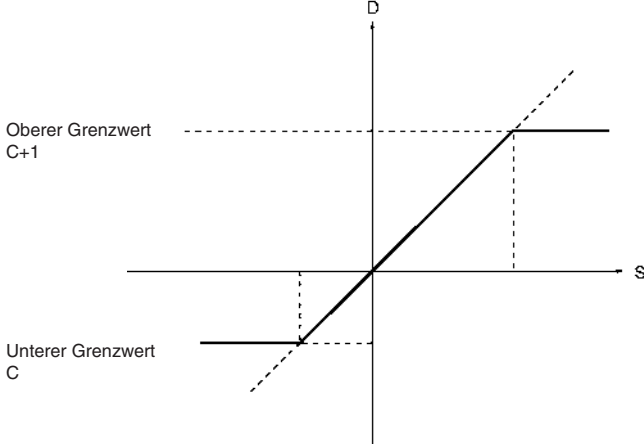
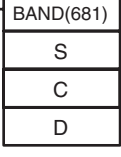
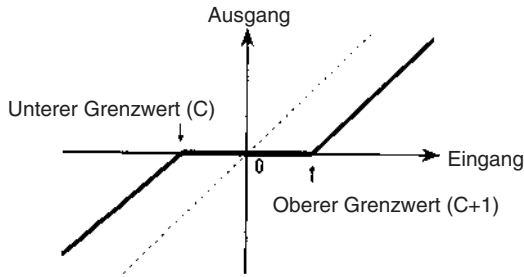
Befehl	Symbol/Operand	Funktion	Position Ausführungs- bedingung
SET STACK (Stack-Bereich spezifizieren) SSET @SSET 630	SSET(630) TB N TB: Erste Stapeladresse N: Anzahl der Worte	Definiert einen Stapel der spezifizierten Länge beginnend mit dem spezifizierten Wort und initialisiert die Worte im Datenbereich alle auf Null. Interne E/A-Speicheradresse Letztes Wort im Stapel Stapel-zeiger N Worte im Stapel	Ausgabe Erforderlich
PUSH ONTO STACK (Auf Stack schieben) PUSH @PUSH 632	PUSH(632) TB S TB: Erste Stapeladresse S: Quellwort	Schreibt ein Datenwort in das spezifizierte Wort im Stapel. Interne E/A-Speicheradresse Interne E/A-Speicheradresse PUSH(632)	Ausgabe Erforderlich
LAST IN FIRST OUT (LIFO-Speicher) LIFO @LIFO 634	LIFO(634) TB D TB: Erste Stapeladresse D: Zielwort	Liest das zuletzt in den spezifizierten Stapel geschriebene Datenwort (die neuesten Daten im Stapel) Stapelzeiger Interne E/A-Speicheradresse Interne E/A-Speicheradresse Neueste Daten Stapel-zeiger Der Zeiger wird dekrementiert LIFO-Prinzip (zuletzt geschriebene Daten werden zuerst ausgelesen) A bleibt unverändert	Ausgabe Erforderlich
FIRST IN FIRST OUT (FIFO-Speicher) FIFO @FIFO 633	FIFO(633) TB D TB: Erste Stapeladresse D: Zielwort	Liest das zuerst in den spezifizierten Stapel geschriebene Datenwort (die ältesten Daten im Stapel) Stapel-zeiger Interne E/A-Speicheradresse Interne E/A-Speicheradresse Älteste Daten Stapel-zeiger FIFO-Prinzip (zuerst geschriebene Daten werden auch zuerst ausgelesen)	Ausgabe Erforderlich

Befehl	Symbol/Operand	Funktion	Position Ausführungs- bedingung
DIMENSION RECORD TABLE (Tabellen-definition) DIM @DIM 631	 N: Tabellennummer LR: Länge jedes Datensatzes NR: Anzahl der Datensätze TB: Erstes Wort der Tabelle	Definiert eine Datensatztabelle durch Angabe der Länge der einzelnen Datensätze und der Anzahl der Datensätze. Es können bis zu 16 Datensatztabellen definiert werden. Tabellennummer (N) Anzahl der Datensätze 	Ausgabe Erforderlich
SET RECORD LOCATION (Einstellung der Datensatzposition) SETR @SETR 635	 N: Tabellennummer R: Datensatznummer D: Ziel-Indexregister	Schreibt die Position des spezifizierten Datensatzes (die interne E/A-Speicheradresse, an der der Datensatz beginnt) in das spezifizierte Indexregister. Tabellennummer (N) Interne E/A-Speicheradresse  SETR(635) schreibt die interne E/A-Speicheradresse (m) des ersten Worts von Datensatz R in das Indexregister D.	Ausgabe Erforderlich
GET RECORD NUMBER (Datensatznummer erhalten) GETR @GETR 636	 N: Tabellennummer IR: Indexregister D: Zielwort	Gibt die Datensatznummer des Datensatzes in der Speicheradresse, die im spezifizierten Indexregister enthalten ist, zurück. Tabellennummer (N) Interne E/A-Speicheradresse  GETR(636) schreibt die Datensatznummer des Datensatzes, der die E/A-Speicheradresse (m) enthält, in D.	Ausgabe Erforderlich
DATA SEARCH (Datensuche) SRCH @SRCH 181	 C: Erstes Steuerwort R1: Erstes Wort im Bereich Cd: Vergleichsdaten	Sucht nach einem Datenwort innerhalb eines Wortbereichs. Interne E/A-Speicheradresse 	Ausgabe Erforderlich

Befehl	Symbol/Operand	Funktion	Position Ausführungs- bedingung
SWAP BYTES (Bytes tauschen) SWAP @SWAP 637	SWAP(637) N R1 N: Anzahl der Worte R1: Erstes Wort im Bereich	Vertauscht die Bytes höchster und niedrigster Wertigkeit in allen Worten innerhalb des Bereichs. Byte-Position wird ausgetauscht.	Ausgabe Erforderlich
FIND MAXIMUM (Finde Maximum) MAX @MAX 182	MAX(182) C R1 D C: Erstes Steuerwort R1: Erstes Wort im Bereich D: Zielwort	Sucht den maximalen Wert innerhalb des Bereichs. Speicheradresse	Ausgabe Erforderlich
FIND MINIMUM (Finde Minimum) MIN @MIN 183	MIN(183) C R1 D C: Erstes Steuerwort R1: Erstes Wort im Bereich D: Zielwort	Sucht den geringsten Wert innerhalb des Bereichs. Speicheradresse	Ausgabe Erforderlich
SUM (Summen- bildung) SUM @SUM 184	SUM(184) C R1 D C: Erstes Steuerwort R1: Erstes Wort im Bereich D: Erstes Zielwort	Addiert die Bytes oder Worte innerhalb des Bereichs und gibt das Ergebnis in zwei Worte aus.	Ausgabe Erforderlich
FRAME CHECKSUM (Rahmenprüf- summe) FCS @FCS 180	FCS(180) C R1 D C: Erstes Steuerwort R1: Erstes Wort im Bereich D: Erstes Zielwort	Berechnet den ASCII FCS-Wert für den spezifizierten Bereich	Ausgabe Erforderlich

Befehl AWL Code	Symbol/Operand	Funktion	Position Ausführungs- bedingung
STACK SIZE READ (Stack- größe lesen) (nur CS1-H, CJ1-H, CJ1M und CS1D) SNUM @SNUM 638	 TB: Erste Stapel- adresse D: Zielwort	Zählt die Menge der Stapeldaten (Anzahl der Worte) im spezifizierten Stapel.	Ausgabe erforderlich
STACK DATA READ (Stack lesen) (nur CS1-H, CJ1-H, CJ1M und CS1D) SREAD @SREAD 639	 TB: Erste Stapel- adresse C: Offsetwert D: Zielwort	Liest die Daten ab dem spezifizierten Datenelement in dem Stapel. Der Offset-Wert bestimmt die Stelle des gewünschten Datenelements (Anzahl der Datenelemente vor der aktuellen Zeigerposition).	Ausgabe erforderlich
STACK DATA OVERWRITE (Stack schreiben) (nur CS1-H, CJ1-H, CJ1M und CS1D) SWRIT @SWRIT 640	 TB: Erste Stapel- adresse C: Offsetwert S: Quelldaten	Schreibt die Eingangsdaten für das spezifizierte Datenelement in den Stapel (überschreibt vorhandene Daten). Der Offset-Wert bestimmt die Stelle des gewünschten Datenelements (Anzahl der Datenelemente vor der aktuellen Zeigerposition).	Ausgabe erforderlich
STACK DATA INSERT (Stack einfügen) (nur CS1-H, CJ1-H, CJ1M und CS1D) SINS @SINS 641	 TB: Erste Stapel- adresse C: Offsetwert S: Quelldaten	Fügt die Quelldaten an der spezifizierten Stelle in den Stapel ein und verschiebt den Rest der Daten im Stapel nach unten. Der Offset-Wert bestimmt die Stelle des Einfügekpunktes (Anzahl der Datenelemente vor der aktuellen Zeigerposition).	Ausgabe erforderlich
STACK DATA DELETE (Stack löschen) (nur CS1-H, CJ1-H, CJ1M und CS1D) SDEL @SDEL 642	 TB: Erste Stapel- adresse C: Offsetwert D: Zielwort	Löscht das Datenelement an der spezifizierten Stelle im Stapel und verschiebt den Rest der Daten im Stapel nach oben. Der Offset-Wert bestimmt die Stelle des Löschpunktes (Anzahl der Datenelemente vor der aktuellen Zeigerposition).	Ausgabe erforderlich

3-16 Datensteuerungsbefehle

Befehl	Symbol/Operand	Funktion	Position Ausführungs- bedingung
PID CONTROL (PID-Regelung) PID 190	 S: Eingangswort C: Erstes Parameterwort D: Ausgangswort	Führt eine PID-Regelung anhand der spezifizierten Parameter aus. 	Ausgabe Erforderlich
PID CONTROL WITH AUTOTU- NING (PID-Rege- lung mit Auto- Tuning) PIDAT 191 (nur CS1-H, CJ1- H und CJ1M)	 S: Eingangswort C: Erstes Parameterwort D: Ausgangswort	Führt eine PID-Regelung anhand der spezifizierten Parameter aus. Die PID-Konstanten können mit PIDAT(191) durch Autotuning eingestellt werden.	Ausgabe erforderlich
LIMIT CONTROL (Begrenzungs- steuerung) LMT @LMT 680	 S: Eingangswort C: Erstes Grenzwert-Wort D: Ausgabewort	Steuert Ausgabedaten in Abhängigkeit davon, ob sich Eingangsdaten innerhalb des Bereichs zwischen oberem und unterem Grenzwert befinden. 	Ausgabe Erforderlich
DEAD BAND CONTROL (Totband- Steuerung) BAND @BAND 681	 S: Eingangswort C: Erstes Grenzwert-Wort D: Ausgabewort	Steuert Ausgabedaten in Abhängigkeit davon, ob sich Eingangsdaten innerhalb der Totzone befinden. 	Ausgabe Erforderlich

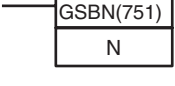
Befehl	Symbol/Operand	Funktion	Position Ausführungs- bedingung				
DEAD ZONE CONTROL (Totzonen-Steuerung) ZONE @ZONE 682	<table><tr><td>ZONE(682)</td></tr><tr><td>S</td></tr><tr><td>C</td></tr><tr><td>D</td></tr></table> S: Eingangswort C: Erstes Grenzwert-Wort D: Ausgabewort	ZONE(682)	S	C	D	Addiert den spezifizierten Offset zu den Eingangsdaten und gibt das Ergebnis aus. 	Ausgabe Erforderlich
ZONE(682)							
S							
C							
D							
TIME-PROPORTIONAL OUTPUT (Zeit-proportionaler Ausgang) TPO 685 (nur CS/CJ-Serie CPU-Baugruppen Version 2.0 oder neuer)	<table><tr><td>TPO(685)</td></tr><tr><td>S</td></tr><tr><td>C</td></tr><tr><td>R</td></tr></table> S: Eingangswort C: Erstes Parameterwort R: Impulsausgabe-Bit	TPO(685)	S	C	R	Verwendet das Tastverhältnis oder den Stellwert aus dem spezifizierten Wort als Eingabe, konvertiert das Tastverhältnis in eine zeitproportionale Ausgabe auf der Basis der spezifizierten Parameter und gibt das Ergebnis über den spezifizierten Ausgang aus.	Ausgabe Erforderlich
TPO(685)							
S							
C							
R							
SCALING (Skalierung) SCL @SCL 194	<table><tr><td>SCL(194)</td></tr><tr><td>S</td></tr><tr><td>S1</td></tr><tr><td>R</td></tr></table> S: Quellwort P1: Erstes Parameterwort R: Ergebniswort	SCL(194)	S	S1	R	Konvertiert Binärdaten ohne Vorzeichen in BCD-Daten ohne Vorzeichen entsprechend der spezifizierten linearen Funktion. 	Ausgabe Erforderlich
SCL(194)							
S							
S1							
R							

Befehl	AWL Code	Symbol/Operand	Funktion	Position Ausführungsbedingung													
SCALING 2 (Skalierung 2)	SCL2 @SCL2 486	<table><tr><td>SCL2(486)</td></tr><tr><td>S</td></tr><tr><td>S1</td></tr><tr><td>R</td></tr></table> S: Quellwort P1: Erstes Parameterwort R: Ergebniswort	SCL2(486)	S	S1	R	Konvertiert Binärdaten mit Vorzeichen in BCD-Daten mit Vorzeichen entsprechend der spezifizierten linearen Funktion. Ein Offset kann bei der Definition der linearen Funktion eingegeben werden. Positiver Offset R (BCD mit Vorzeichen) S (Binär mit Vorzeichen) Negativer Offset R (BCD mit Vorzeichen) S (Binär mit Vorzeichen) Offset von 0000 R (BCD mit Vorzeichen) S (Binär mit Vorzeichen) Offset = 0000 Hex <table><tr><td>P1</td><td>Offset</td><td>(Binär mit Vorzeichen)</td></tr><tr><td>P1 + 1</td><td>ΔY</td><td>(Binär mit Vorzeichen)</td></tr><tr><td>P1 + 2</td><td>ΔX</td><td>(BCD mit Vorzeichen)</td></tr></table> <td>Ausgabe Erforderlich</td>	P1	Offset	(Binär mit Vorzeichen)	P1 + 1	ΔY	(Binär mit Vorzeichen)	P1 + 2	ΔX	(BCD mit Vorzeichen)	Ausgabe Erforderlich
SCL2(486)																	
S																	
S1																	
R																	
P1	Offset	(Binär mit Vorzeichen)															
P1 + 1	ΔY	(Binär mit Vorzeichen)															
P1 + 2	ΔX	(BCD mit Vorzeichen)															

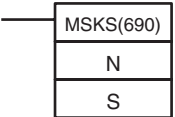
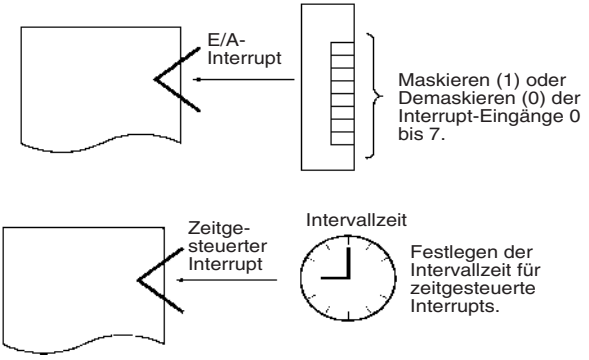
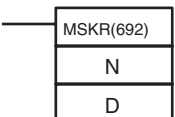
Befehl	Symbol/Operand	Funktion	Position Ausführungs- bedingung
SCALING 3 (Skalierung 3) SCL3 @ SCL3 487	SCL3(487) S P1 R S: Quellwort P1: Erstes Parameterwort R: Ergebniswort	Konvertiert BCD-Daten mit Vorzeichen in Binärdaten mit Vorzeichen entsprechend der spezifizierten linearen Funktion. Ein Offset kann bei der Definition der linearen Funktion eingegeben werden. Positiver Offset Negativer Offset Offset von 0000	Ausgabe Erforderlich
AVERAGE (Mittelwert) AVG 195	AVG(195) S N R S: Quellwort N: Anzahl der Zyklen R: Ergebniswort	Berechnet den Mittelwert bei einem Eingangswort über die spezifizierte Anzahl von Zyklen. S: Quellwort N: Anzahl der Zyklen	Ausgabe Erforderlich

3-17 Unterprogrammbefehle

Befehl	Symbol/Operand	Funktion	Position Ausführungs- bedingung
SUBROUTINE CALL (Unterprogramm aufrufen) SBS @ SBS 091	SBS(091) N N: Unterprogramm-Nummer	Ruft das Unterprogramm mit der spezifizierten Unterprogrammnummer auf und führt das Programm aus.	Ausgabe Erforderlich
MACRO (Makro) MCRO @ MCRO 099	MCRO(099) N S D N: Unterprogramm-Nummer S: Erstes Eingangs-Parameterwort D: Erstes Ausgangs-Parameterwort	Ruft das Unterprogramm mit der spezifizierten Unterprogramm-Nummer auf und führt das Programm unter Verwendung der Eingangsparameter in S bis S+3 und der Ausgangsparameter in D bis D+3 aus.	Ausgabe Erforderlich
SUBROUTINE ENTRY (Unterprogramm-Anfang) SBN 092	SBN(092) N N: Unterprogramm-Nummer	Kennzeichnet den Beginn eines Unterprogramms mit der spezifizierten Unterprogramm-Nummer.	Ausgabe Nicht erforderlich
SUBROUTINE RETURN (Unterprogramm verlassen) RET 093	RET(093)	Kennzeichnet das Ende eines Unterprogramms.	Ausgabe Nicht erforderlich

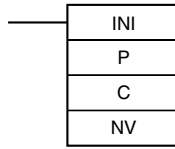
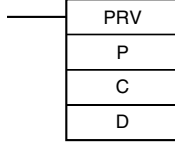
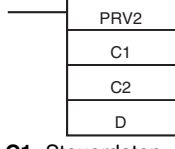
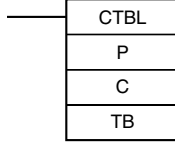
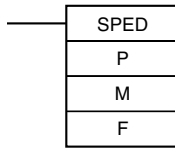
Befehl AWL Code	Symbol/Operand	Funktion	Position Ausführungs- bedingung
GLOBAL SUB-ROUTINE CALL (Globales Unterprogramm aufrufen) (nur CS1-H, CJ1-H, CJ1M und CS1D) GSBS 750	 N: Unterprogramm-Nummer	Ruft das Unterprogramm mit der angegebenen Unterprogrammnummer auf und führt das Programm aus.	Ausgabe Nicht erforderlich
GLOBAL SUB-ROUTINE ENTRY (Globales Unterprogramm-Anfang) (nur CS1-H, CJ1-H, CJ1M und CS1D) GSBN 751	 N: Unterprogramm-Nummer	Kennzeichnet den Beginn eines Unterprogramms mit der spezifizierten Unterprogrammnummer.	Ausgabe Nicht erforderlich
GLOBAL SUBROUTINE RETURN (Globales Unterprogramm verlassen) (nur CS1-H, CJ1-H, CJ1M und CS1D) GRET 752		Kennzeichnet das Ende eines Unterprogramms.	Ausgabe Nicht erforderlich

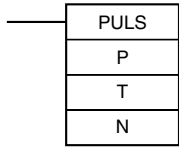
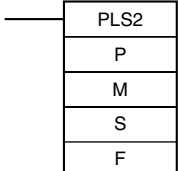
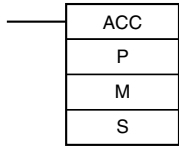
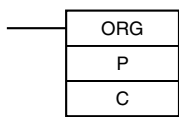
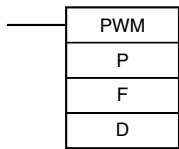
3-18 Interrupt-Steuerungsbefehle

Befehl AWL Code	Symbol/Operand	Funktion	Position Ausführungs- bedingung
SET INTERRUPT MASK (Interrupt-Maske einstellen) (Nicht unterstützt von CS1D CPU-Baugruppen für Duplex-CPU-Systeme.) MSKS @MSKS 690	 N: Interrupt-Bezeichner S: Interrupt-Daten	Richtet Interrupt-Verarbeitung für E/A-Interrupts oder zeitgesteuerte Interrupts ein. Beide Interrupt-Tasks (E/A und zeitgesteuert) werden maskiert (deaktiviert), wenn die SPS erstmalig eingeschaltet wird. MSKS(690) kann zum Maskieren oder Demaskieren von E/A-Interrupts verwendet werden sowie zum Einstellen der Intervallzeit für zeitgesteuerte Interrupts. Interrupt-Eingangsbaugruppe 0 bis 3  Maskieren (1) oder Demaskieren (0) der Interrupt-Eingänge 0 bis 7. Intervallzeit Festlegen der Intervallzeit für zeitgesteuerte Interrupts.	Ausgabe Erforderlich
READ INTERRUPT MASK (Interrupt-Maske lesen) (Nicht unterstützt von CS1D CPU-Baugruppen für Duplex-CPU-Systeme.) MSKR @MSKR 692	 N: Interrupt-Bezeichner D: Zielwort	Liest die aktuellen Interrupt-Verarbeitungseinstellungen die mit MSKS(690) eingestellt wurden.	Ausgabe Erforderlich

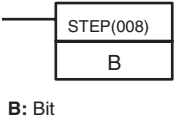
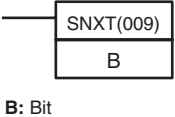
Befehl	Symbol/Operand	Funktion	Position Ausführungsbedingung
CLEAR INTERRUPT (Interrupt zurücksetzen) (Nicht unterstützt von CS1D CPU-Baugruppen für Duplex-CPU-Systeme) CLI @CLI 691	CLI(691) N S N: Interrupt-Bezeichner S: Interrupt-Daten	Löscht oder erhält aufgezeichnete Interrupt-Eingänge für E/A-Interrupts oder legt die Zeit bis zum ersten Interrupt bei zeitgesteuerten Interrupts fest. N = 0 bis 3 N = 4 bis 5	Ausgabe Erforderlich
DISABLE INTERRUPTS (Interrupts deaktivieren) DI @DI 693	DI(693)	Deaktiviert die Ausführung sämtlicher Interrupt-Tasks mit Ausnahme der Ausschalt-Interrupt-Task. Deaktiviert die Ausführung sämtlicher Interrupt-Tasks (mit Ausnahme der Ausschalt-Interrupt-Task).	Ausgabe Erforderlich
ENABLE INTERRUPTS (Interrupts aktivieren) EI 694	EI(694)	Aktiviert die Ausführung aller mit DI(693) deaktivierten Interrupt-Tasks. Deaktiviert die Ausführung sämtlicher Interrupt-Tasks (mit Ausnahme der Ausschalt-Interrupt-Task). Aktiviert die Ausführung aller deaktivierten Interrupt-Tasks.	Ausgabe Nicht erforderlich

3-19 Befehle für schnelle Zähler und Impulsausgaben (nur CJ1M-CPU21/22/23)

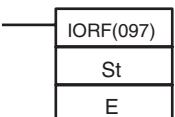
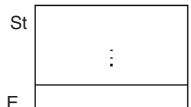
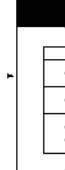
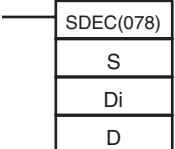
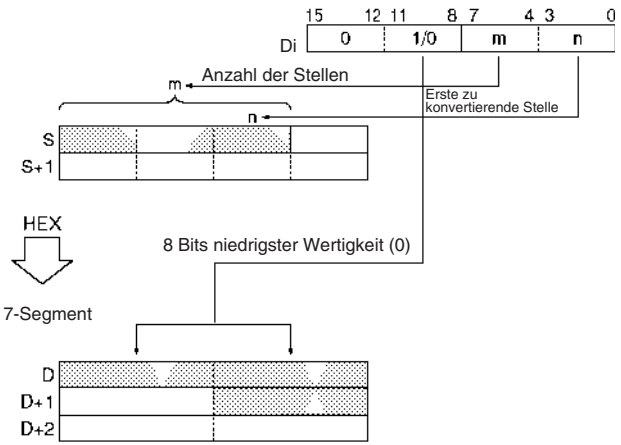
Befehl	Symbol/Operand	Funktion	Position Ausführungs- bedingung
MODE CON- TROL (Betriebs- artensteuerung) INI @ INI 880	 P: Ausgangsbe- zeichner C: Steuerdaten NV: Erstes Wort mit neuem Istwert	INI(880) wird verwendet zum Starten und Anhalten des Zielwertvergleichs, zur Änderung des Istwerts eines schnellen Zählers, zur Änderung des Istwerts eines Interrupt-Eingangs (Zählermodus), zur Änderung des Istwerts einer Impulsausgabe oder zum Anhalten einer Impulsausgabe.	Ausgabe Erforderlich
HIGH-SPEED COUNTER PV READ (Schneller Zähler-Istwert lesen) PRV @ PRV 881	 P: Ausgangsbe- zeichner C: Steuerdaten D: Erstes Zielwort	PRV(881) wird verwendet zum Lesen des Istwerts eines schnellen Zählers, einer Impulsausgabe oder eines Interrupt-Eingangs (Zählermodus).	Ausgabe Erforderlich
COUNTER FREQUENCY CONVERT (Impulsfrequenz- konvertierung) PRV2 883 (nur CJ1M CPU- Baugruppen Ver- sion 2.0 oder neuer)	 C1: Steuerdaten C2: Impulse/ Umdrehung D: Erstes Zielwort	Liest den Impulsfrequenzeingang eines schnellen Zählers und wandelt entweder die Frequenz in eine Drehzahl (Umdrehungen/Minute) oder den aktuellen Wert des Zählers in eine Gesamtanzahl von Umdrehungen um. Das Ergebnis wird als 8-stellige Hexadezimalzahl an die Zielworte ausgegeben. Impulsfrequenzen können nur vom schnellen Zähler 0 gelesen werden.	Ausgabe Erforderlich
COMPARISON TABLE LOAD (Tabellen-Regi- stervergleich) CTBL @ CTBL 882	 P: Schnnittstellen- bezeichner C: Steuerdaten TB: Erstes Ver- gleichstabellen- wort	CTBL(882) wird zur Registrierung einer Vergleichstabelle und zur Ausführung von Zielwert- oder Bereichsvergleichen mit dem Istwert eines schnellen Zählers verwendet.	Ausgabe Erforderlich
SPEED OUTPUT (Geschwindig- keitssteuerung) SPED @ SPED 885	 P: Ausgangsbe- zeichner M: Ausgangsbe- triebsart F: Erstes Impuls- frequenzwort	SPED(885) wird zur Spezifikation der Frequenz und zur Durchführung der Impulsausgabe ohne Beschleunigung oder Verzögerung verwendet.	Ausgabe Erforderlich

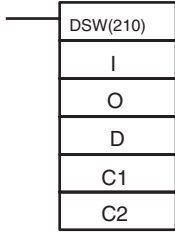
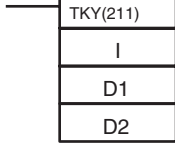
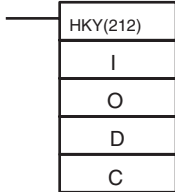
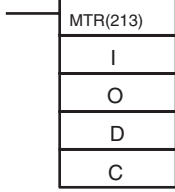
Befehl AWL Code	Symbol/Operand	Funktion	Position Ausführungs- bedingung
SET PULSES (Impulsausgabe) PULS @PULS 886	 P: Ausgangsbezeichner T: Impulstyp N: Anzahl der Impulse	PULS(886) wird verwendet, um die Anzahl der Impulse für die Impulsausgabe festzulegen.	Ausgabe Erforderlich
PULSE OUTPUT (Impuls- ausgabe 2) PLS2 @PLS2 887	 P: Ausgangsbezeichner M: Ausgangs- triebsart S: Erstes Wort der Einstellungen- tabelle F: Erstes Wort der Startfrequenz	PLS2(887) wird verwendet, um die Impulsfrequenz und die Beschleunigungs-/Verzögerungsraten einzustellen und die Impulsausgabe mit Beschleunigung/Verzögerung (mit unterschiedlichen Beschleunigungs-/Verzögerungsraten) durchzuführen. Nur Positionierung ist möglich.	Ausgabe Erforderlich
ACCELERATION CONTROL (Beschleuni- gungssteuerung) ACC @ACC 888	 P: Ausgangsbezeichner M: Ausgangs- triebsart S: Erstes Wort der Einstellungen- tabelle	ACC(888) wird verwendet, um die Impulsfrequenz und die Beschleunigungs-/Verzögerungsraten einzustellen und die Impulsausgabe mit Beschleunigung/Verzögerung (mit gleichen Beschleunigungs-/Verzögerungsraten) durchzuführen. Sowohl Positionierung als auch Geschwindigkeitssteuerung ist möglich	Ausgabe Erforderlich
ORIGIN SEARCH (Nullpunktsuche) ORG @ORG 889	 P: Ausgangsbezeichner C: Steuerdaten	ORG(889) wird zur Durchführung von Nullpunktsuche und Nullpunkt-rückkehrfunktionen verwendet.	Ausgabe Erforderlich
PULSE WITH VARIABLE DUTY FACTOR (Impuls mit variablem Tastverhältnis) PWM @ 891	 P: Ausgangsbezeichner F: Frequenz D: Tastverhältnis	PWM(891) wird zur Ausgabe von Impulsen mit variablem Tastverhältnis verwendet.	Ausgabe Erforderlich

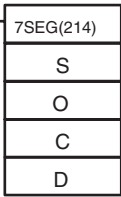
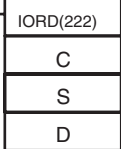
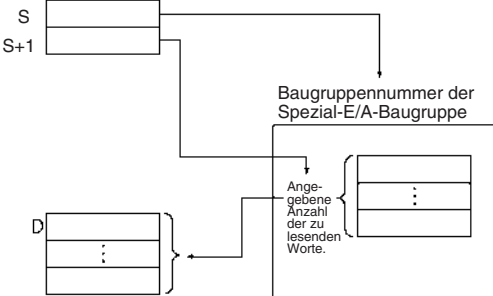
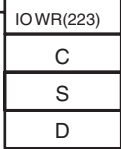
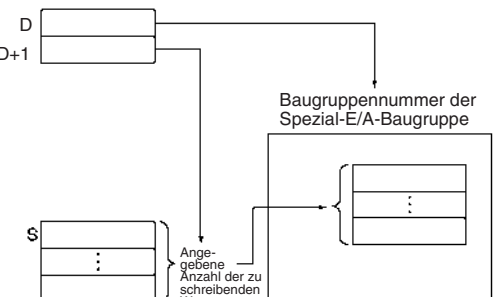
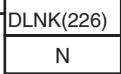
3-20 Step-Befehle

Befehl	Symbol/Operand	Funktion	Position Ausführungs- bedingung
STEP DEFINE (Schritt) STEP 008	 B: Bit	STEP(008) funktioniert je nach seiner Position und ob ein Steuerbit angegeben wurde oder nicht, auf die folgenden 2 Arten: (1) Startet einen bestimmten Step. (2) Beendet den Step-Programmierungsbereich (d. h. die Step-Ausführung).	Ausgabe Erforderlich
STEP START (nächster Schritt) SNXT 009	 B: Bit	SNXT(009) wird auf die folgenden drei Arten verwendet: (1) Zum Start der Ausführung der Step-Programmierung, (2) zum Fortfahren mit dem nächsten Step-Steuerbit, (3) für die Beendigung der Step-Programmierung.	Ausgabe Erforderlich

3-21 Befehle für E/A-Baugruppen

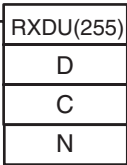
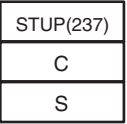
Befehl	Symbol/Operand	Funktion	Position Ausführungs- bedingung
I/O REFRESH (E/A-Auffrischung) IORF @ IORF 097	 St: Anfangswort E: Endwort	Aktualisiert die spezifizierten E/A-Worte E/A-Bitbereich oder Spezial-E/A-Baugruppen- Bitbereich  E/A-Aktualisierung  E/A-Baugruppe oder Spezial-E/A- Baugruppe	Ausgabe Erforderlich
7-SEGMENT DECODER (7 Segment- Dekoder) SDEC @ SDEC 078	 S: Quellwort Di: Stellen- bezeichner D: Erstes Zielwort	Konvertiert den hexadezimalen Inhalt der angegebenen Stelle(n) in 8-Bit-7-Segment-Anzeigecode und gibt ihn in die höher- oder niederwertigen 8 Bits der spezifizierten Zielworte aus 15 12 11 8 7 4 3 0 Di 0 1/0 m n ← Anzahl der Stellen ← Erste zu konvertierende Stelle  8 Bits niedrigster Wertigkeit (0) 7-Segment D D+1 D+2	Ausgabe Erforderlich

Befehl AWL Code	Symbol/Operand	Funktion	Position Ausführungs- bedingung
DIGITAL SWITCH INPUT (Digital-schaltereingabe) DSW 210 (nur CS/CJ-Serie CPU-Baugruppen Version 2.0 oder neuer)	 I: Dateneingangswort (D0 bis D3) O: Ausgangswort D: Erstes Ergebniswort C1: Anzahl der Stellen C2: Systemwort	Liest den Wert, der auf einem an eine Eingangs- oder Ausgangsbaugruppe angeschlossenen externen Digital- oder Daumenradschalter eingestellt ist und speichert die 4- bzw. 8-stellige Zahl in den spezifizierten Worten.	Ausgabe Erforderlich
TEN KEY INPUT (Zehnertasteneingabe) TKY 211 (nur CS/CJ-Serie CPU-Baugruppen Version 2.0 oder neuer)	 I: Dateneingangswort D1: Erstes Registerwort D2: Tasteneingabewort	Liest numerische Daten von einem an eine Eingangsbaugruppe angeschlossenen numerischen Tastenfeld und speichert bis zu 8 Stellen BCD-Daten in den spezifizierten Worten.	Ausgabe Erforderlich
HEXADECIMAL KEY INPUT (Hexadezimale Tastatureingabe) HKY 212 (nur CS/CJ-Serie CPU-Baugruppen Version 2.0 oder neuer)	 I: Dateneingangswort O: Ausgangswort D: Erstes Registerwort C: Systemwort	Liest numerische Daten von einem an eine Eingangs- und Ausgangsbaugruppe angeschlossenen Hexadezimal-Tastenblock und speichert bis zu 8 Stellen Hexadezimaldaten in den spezifizierten Worten.	Ausgabe Erforderlich
MATRIX INPUT (Matrizeingabe) MTR 213 (nur CS/CJ-Serie CPU-Baugruppen Version 2.0 oder neuer)	 I: Dateneingangswort O: Ausgangswort D: Erstes Zielwort C: Systemwort	Gibt bis zu 64 Signale von einer an eine Eingangs- und Ausgangsbaugruppe (mit 8 Eingängen und 8 Ausgängen) angeschlossenen 8 × 8-Matrix ein und speichert diese 64-Bit-Daten in den 4 Zielworten.	Ausgabe Erforderlich

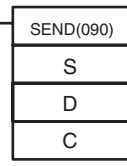
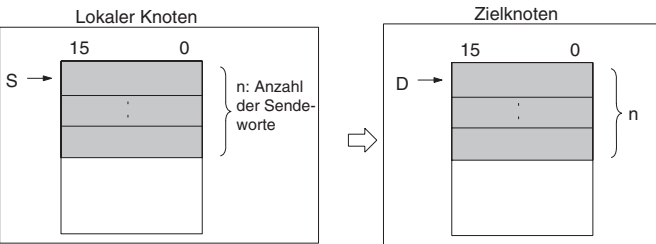
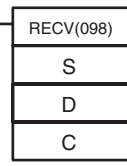
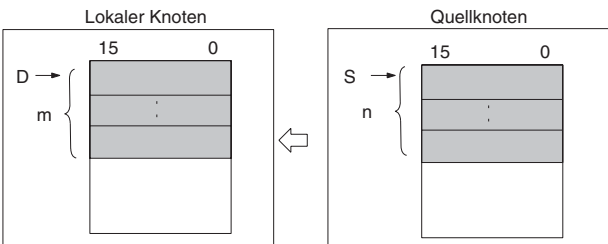
Befehl	Symbol/Operand	Funktion	Position Ausführungs- bedingung
7-SEGMENT DISPLAY OUT- PUT (7-Segment- anzeigen- Ausgabe) 7SEG 214 (nur CS/CJ-Serie CPU-Baugruppen Version 2.0 oder neuer)	 S: Erstes Quellwort O: Ausgangswort C: Steuerdaten D: Systemwort	Konvertiert die Eingangsdaten (4-stellige oder 8-stellige BCD-Daten) in Daten für eine 7-Segment-Anzeige und gibt diese Daten in das spezifizierte Ausgangswort aus.	Ausgabe Erforderlich
INTELLIGENT I/O READ (Intelligen- tes E/A-Lesen) IORD @IORD 222	 C: Steuerdaten S: Übertragungs- quelle und Anzahl der Worte D: Übertragungsziel und Anzahl der Worte	 Hinweis: CS/CJ-Serie CPU-Baugruppen Version 2.0 oder neuer (einschließlich CS1-H, CJ1-H und CJ1M CPU-Baugruppen ab Lot-Nummer 030418) können von CPU-Bus-Baugruppen lesen.	Ausgabe Erforderlich
INTELLIGENT I/O WRITE (Intelligentes E/A-Schreiben) IOWR @IOWR 223	 C: Steuerdaten S: Übertragungs- quelle und Anzahl der Worte D: Übertragungsziel und Anzahl der Worte	 Hinweis: CS/CJ-Serie CPU-Baugruppen Version 2.0 oder neuer (einschließlich CS1-H, CJ1-H und CJ1M CPU-Baugruppen ab Lot-Nummer 030418) können in CPU-Bus-Baugruppen schreiben.	Ausgabe Erforderlich
CPU BUS UNIT I/O REFRESH (CPU-Bus- Baugruppe E/A- Auffrischung) (nur CS1-H, CJ1-H, CJ1M und CS1D) DLNK @DLNK 226	 N: Baugruppen- nummer	Aktualisiert die E/A in der CPU-Bus-Baugruppe mit der spezifizierten Baugruppennummer unmittelbar.	Ausgabe erforderlich

3-22 Befehle für serielle Kommunikation

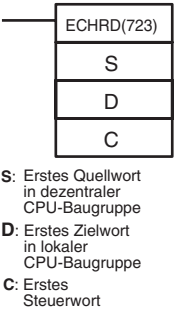
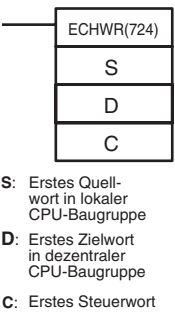
Befehl	Symbol/Operand	Funktion	Position Ausführungsbedingung
PROTOCOL MACRO (Protokoll-Makro) PMCR @ PMCR 260	PMCR(260). C1 C2 S R C1: Steuerwort 1 C2: Steuerwort 2 S: Erstes Sendewort R: Erstes Empfangswort	Ruft eine bei einem seriellen Kommunikationsmodul (nur CS-Serie) oder einer seriellen Kommunikationsbaugruppe registrierte Kommunikationssequenz auf und führt diese aus	Ausgabe Erforderlich
TRANSMIT (Senden) TXD @ TXD 236	TXD(236) S C N S: Erstes Quellwort C: Steuerwort N: Anzahl der Bytes 0000 bis 0100 hex (0 bis 256 dezimal)	Gibt die spezifizierte Anzahl Daten-Bytes über die integrierte RS-232C-Schnittstelle der CPU-Baugruppe aus.	Ausgabe Erforderlich
RECEIVE (Empfangen) RXD @ RXD 235	RXD(235) D C N D: Erstes Zielwort C: Steuerwort N: Anzahl der Bytes zum Speichern von 0000 bis 0100 hex (0 bis 256 dezimal)	Liest die spezifizierte Anzahl Daten-Bytes über die integrierte RS-232C-Schnittstelle der CPU-Baugruppe.	Ausgabe Erforderlich
TRANSMIT VIA SERIAL COMMUNICATIONS UNIT (CPU-Bus-Baugruppe - Senden) TXDU @ TXDU 256	TXDU(256) S C N S: Erstes Quellwort C: Erstes Steuerwort N: Anzahl der Bytes (0000 bis 0256 BCD)	Gibt die angegebene Anzahl Datenbytes ohne jegliche Umwandlung über die serielle Schnittstelle einer seriellen Kommunikationsbaugruppe ab Ver. 1.2 aus. Die Ausgabe der Daten erfolgt protokollfrei. Sind im zugeordneten DM-Bereich Start- und/oder Endcode festgelegt, werden diese verwendet.	Ausgabe erforderlich

Befehl	Symbol/Operand	Funktion	Position Ausführungs- bedingung
Daten über eine CPU-Bus-Baugruppe empfangen RXDU @RXDU 255	 D: Erstes Zielwort C: Erstes Steuerwort N: Anzahl der zu speichernden Bytes (0000 bis 0100 hex)	Empfängt beginnend mit dem festgelegten ersten Wort die angegebene Anzahl Datenbytes über die serielle Schnittstelle einer seriellen Kommunikationsbaugruppe ab Ver. 1.2. Der Datenempfang erfolgt protokollfrei. Sind im zugeordneten DM-Bereich Start- und/oder Endcode festgelegt, werden diese verwendet.	
CHANGE SERIAL PORT SETUP (RS232-Konfigurationseinstellung) STUP @STUP 237	 C: Steuerwort (Schnittstelle) S: Erstes Quellwort	Ändert die Kommunikationsparameter einer seriellen Schnittstelle der CPU-Baugruppe, einer seriellen Kommunikationsbaugruppe (CPU-Bus-Baugruppe) oder eines seriellen Kommunikationsmoduls. Damit ermöglicht STUP(237) die Änderung des Protokollmodus während des SPS-Betriebs.	Ausgabe Erforderlich

3-23 Netzwerkbeefhle

Befehl	Symbol/Operand	Funktion	Position Ausführungs- bedingung
NETWORK SEND (Netzwerk Senden) SEND @SEND 090	 S: Erstes Quellwort D: Erstes Zielwort C: Erstes Steuerwort	Überträgt Daten zu einem Knoten im Netzwerk. 	Ausgabe Erforderlich
NETWORK RECEIVE (Netzwerk empfangen) RECV @RECV 098	 S: Erstes Quellwort D: Erstes Zielwort C: Erstes Steuerwort	Fordert Daten an, die von einem Knoten im Netzwerk übertragen werden, und empfängt die Daten. 	Ausgabe Erforderlich

Befehl	Symbol/Operand	Funktion	Position Ausführungs- bedingung
DELIVER COMMAND (FINS-Befehl zustellen) CMND @CMND 490	CMND(490) S D C S: Erstes Befehls- wort D: Erstes Antwort- Wort C: Erstes Steuerwort	Sendet FINS-Befehle und empfängt die Antwort.	Ausgabe Erforderlich
EXPLICIT MESSAGE SEND (Allgemeine Explicit Message-Kommunikation) EXPLT 720 (nur CS/CJ-Serie CPU-Baugruppen Version 2.0 oder neuer)	EXPLT(720) S D C S: Erstes Wort der Sendemeldung D: Erstes Wort der empfangenen Meldung C: Erstes Steuerwort	Sendet eine explizite Meldung mit einem beliebigen Service-Code.	Ausgabe Erforderlich
EXPLICIT GET ATTRIBUTE (Explicit-Einfaches Attribut abrufen) EGATR 721 (nur CS/CJ-Serie CPU-Baugruppen Version 2.0 oder neuer)	EGATR(721) S D C S: Erstes Wort der Sendemeldung D: Erstes Wort der empfangenen Meldung C: Erstes Steuerwort der Meldung	Liest Statusinformationen mit einer expliziten Meldung (Get Attribute Single, Service-Code: 0E hex).	Ausgabe Erforderlich
EXPLICIT SET ATTRIBUTE (Explicit-Einfaches Attribut setzen) ESATR 722 (nur CS/CJ-Serie CPU-Baugruppen Version 2.0 oder neuer)	ESATR(722) S C S: Erstes Wort der Sendemeldung C: Erstes Steuerwort	Schreibt Statusinformationen mit einer expliziten Meldung (Set Attribute Single, Service-Code: 0E hex)	Ausgabe Erforderlich

Befehl	Symbol/Operand	Funktion	Position Ausführungs- bedingung
EXPLICIT WORD READ (Explicit- SPS-Wort lesen) ECHRD 723 (nur CS/CJ-Serie CPU-Baugruppen Version 2.0 oder neuer)	 S: Erstes Quellwort in dezentraler CPU-Baugruppe D: Erstes Zielwort in lokaler CPU-Baugruppe C: Erstes Steuerwort	Liest Daten für die lokale CPU-Baugruppe von einer dezentralen CPU-Baugruppe im Netzwerk. (Die dezentrale CPU-Baugruppe muss explizite Meldungen unterstützen.)	Ausgabe Erforderlich
EXPLICIT WORD WRITE (Explicit- SPS-Wort schreiben) ECHWR 724 (nur CS/CJ-Serie CPU-Baugruppen Version 2.0 oder neuer)	 S: Erstes Quell- wort in lokaler CPU-Baugruppe D: Erstes Zielwort in dezentraler CPU-Baugruppe C: Erstes Steuerwort	Schreibt Daten von der lokalen CPU-Baugruppe in eine dezentrale CPU-Baugruppe im Netzwerk. (Die dezentrale CPU-Baugruppe muss explizite Meldungen unterstützen.)	Ausgabe Erforderlich

3-24 Dateispeicherbefehle

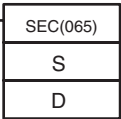
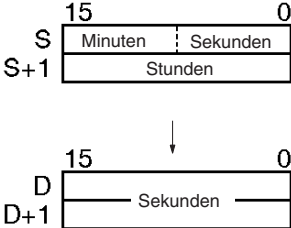
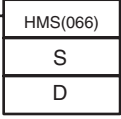
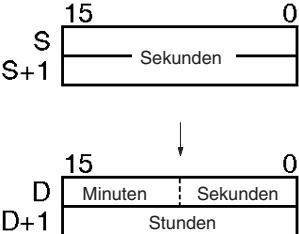
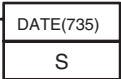
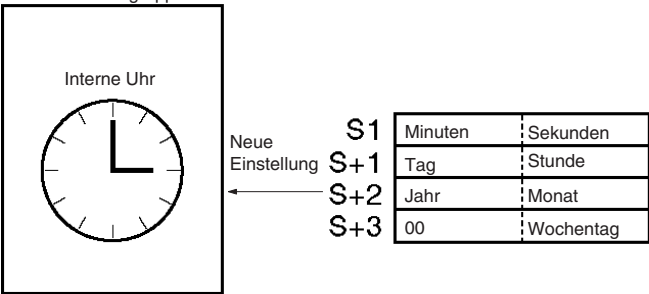
Befehl	Symbol/Operand	Funktion	Position Ausführungsbedingung
READ DATA FILE (Datei lesen) FREAD @FREAD 700	FREAD(700) C S1 S2 D C: Steuerwort S1: Erstes Quellwort S2: Dateiname D: Erstes Zielwort	Liest die spezifizierten Daten oder Datenmenge aus der spezifizierten Datendatei im Dateispeicher in den spezifizierten Datenbereich der CPU-Baugruppe. Startet Lesen der in S1+2 und S1+3 spezifizierten Adresse In S2 spezifizierte Datei CPU-Baugruppe In S1 und S1+1 spezifizierte Anzahl der Worte Speicherkarte oder EM-Dateispeicher (Spezifiziert durch die 4. Stelle von C) Anzahl der in D und D+1 geschriebenen Worte	Ausgabe Erforderlich
WRITE DATA FILE (Datei schreiben) FWRIT @FWRIT 701	FWRIT(701) C D1 D2 S C: Steuerwort D1: Erstes Zielwort D2: Dateiname S: Erstes Quellwort	Überschreibt Daten der spezifizierten Datendatei im Dateispeicher mit den spezifizierten Daten aus dem Datenbereich der CPU-Baugruppe, oder hängt diese Daten an. Wenn die spezifizierte Datei nicht vorhanden ist, wird eine neue Datei mit diesem Dateinamen erstellt. CPU-Baugruppe In S spezifizierte Anfangsadresse In D1+2 und D1+3 spezifiziertes Anfangswort In S2 spezifizierte Datei In D1 und D1+1 spezifizierte Anzahl der Worte Überschreiben Speicherkarte oder EM-Dateispeicher (Spezifiziert durch die 4. Stelle von C.) Dateiende Vorhandene Daten Anhängen Anfang der Datei In D2 spezifizierte Datei Neue Datei erstellt Speicherkarte oder EM-Dateispeicher (Spezifiziert durch die 4. Stelle von C.)	Ausgabe Erforderlich

3-25 Anzeigebefehle

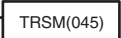
Befehl	AWL Code	Symbol/Operand	Funktion	Position Ausführungs- bedingung
DISPLAY MESSAGE (Meldungs- anzeige)	MSG @MSG 046	MSG(046) N M N: Meldungsnummer M: Erstes Meldungswort	Liest die spezifizierten sechzehn Worte der erweiterten ASCII-Daten und zeigt die Meldung auf einer Programmierkonsole an.	Ausgabe Erforderlich

3-26 Datum- und Uhrzeitbefehle

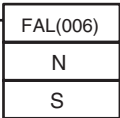
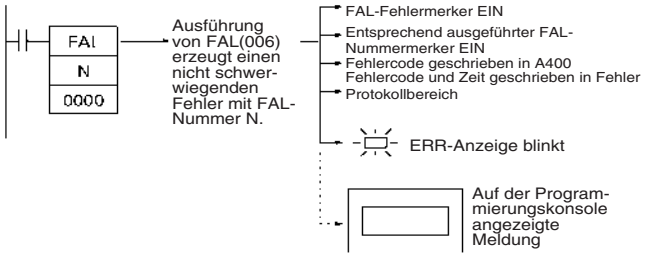
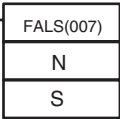
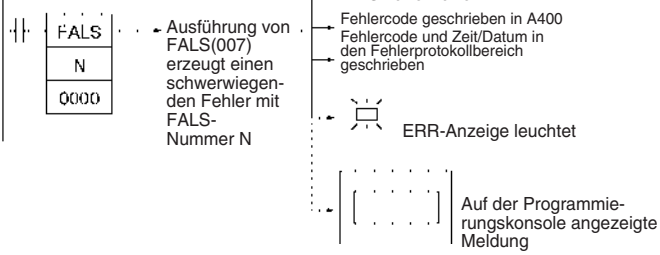
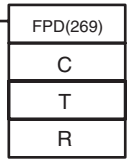
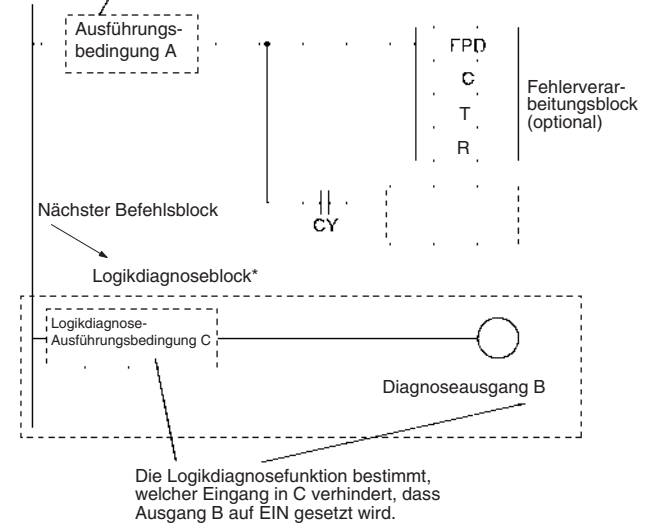
Befehl	AWL Code	Symbol/Operand	Funktion	Position Ausführungs- bedingung
CALENDAR ADD (Kalender- Addition)	CADD @CADD 730	CADD(730) C T R C: Erstes Kalenderwort T: Erstes Zeit-Wort R: Erstes Ergebniswort	Addiert Zeit zu den Kalenderdaten in den spezifizierten Worten. 15870 C Minuten Sekunden C+1 Tag Stunde C+2 Jahr Monat + 15870 T Minuten Sekunden T+1 Stunden ↓ 15870 R Minuten Sekunden R+1 Tag Stunde R+2 Jahr Monat	Ausgabe Erforderlich
CALENDAR SUBTRACT (Kalender- Subtraktion)	CSUB @CSUB 731	CSUB(731) C T R C: Erstes Kalenderwort T: Erstes Zeit-Wort R: Erstes Ergebniswort	Subtrahiert Zeit von den Kalenderdaten in den spezifizierten Worten. 15870 C Minuten Sekunden C+1 Tag Stunde C+2 Jahr Monat − 15870 T Minuten Sekunden T+1 Stunden ↓ 15870 R Minuten Sekunden R+1 Tag Stunde R+2 Jahr Monat	Ausgabe Erforderlich

Befehl AWL Code	Symbol/Operand	Funktion	Position Ausführungs- bedingung
HOURS TO SECONDS (Stunden zu Sekunden) SEC @SEC 065	 S: Erstes Quellwort D: Erstes Zielwort	Wandelt Zeitdaten aus Stunden/Minuten/Sekunden in die entsprechende Zeit nur in Sekunden um. 	Ausgabe Erforderlich
SECONDS TO HOURS (Sekunden zu Stunden) HMS @HMS 066	 S: Erstes Quellwort D: Erstes Zielwort	Wandelt Sekunden in die entsprechende Zeit in Stunden/Minuten/Sekunden um. 	Ausgabe Erforderlich
CLOCK ADJUSTMENT (Uhreinstellung) DATE @DATE 735	 S: Erstes Quellwort	Ändert die Einstellung der internen Uhr auf die Einstellung in den spezifizierten Worten. CPU-Baugruppe 	Ausgabe Erforderlich

3-27 Debugging-Befehle

Befehl AWL Code	Symbol/Operand	Funktion	Position Ausführungs- bedingung
TRACE MEMORY SAMPLING (Aufzeichnungsspeicher-Stichprobe) TRSM 045		Wenn TRSM(045) ausgeführt wird, wird der Status eines vorausgewählten Bits oder Wortes erfasst und im Aufzeichnungsspeicher gespeichert. TRSM(045) kann überall im Programm beliebig oft verwendet werden.	Ausgabe Nicht erforderlich

3-28 Fehlerdiagnosebefehle

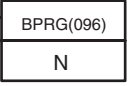
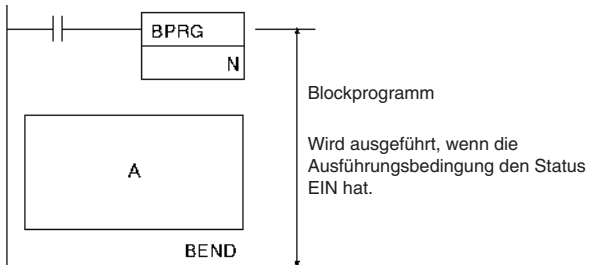
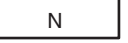
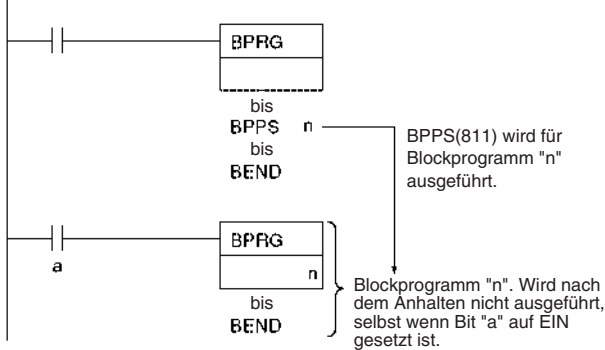
Befehl	Symbol/Operand	Funktion	Position Ausführungs- bedingung
FAILURE ALARM (Geringfügiger Fehleralarm) FAL @ FAL 006	 N: FAL-Nummer S: Erstes Meldungs- Wort oder zu generierender Fehlercode	Erzeugt oder löscht anwenderdefinierte nicht schwerwiegende Fehler. Durch nicht schwerwiegende Fehler wird der SPS-Betrieb nicht angehalten. Erzeugt auch nicht schwerwiegende Fehler im System. 	Ausgabe Erforderlich
SEVERE FAILURE ALARM (Schwerwiegender Fehleralarm) FALS 007	 N: FALS-Nummer S: Erstes Meldungs- wort oder zu generierender Fehlercode	Erzeugt schwerwiegende anwenderdefinierte Fehler. Durch schwerwiegende Fehler wird der SPS-Betrieb angehalten. Erzeugt auch schwerwiegende Fehler im System. 	Ausgabe Erforderlich
FAILURE POINT DETECTION (Fehlerpunkt- Erfassung) FPD 269	 C: Steuerwort T: Überwachungszeit R: Erstes Registerwort	Diagnostiziert einen Fehler in einem Befehlsblock durch Überwachen der Zeit zwischen der Ausführung des Befehls FPD(269) und der Ausführung einer Diagnoseausgabe und durch Suche nach dem Eingang, der verhindert, dass der Ausgang auf EIN geschaltet wird. Zeitüberwachungsfunktion: Startet die Zeitmessung, wenn Ausführungsbedingung A zu EIN wechselt. Erzeugt einen nicht schwerwiegenden Fehler, wenn Ausgang B innerhalb des Überwachungszeitraums nicht auf EIN gesetzt wird. 	Ausgabe Erforderlich

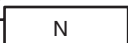
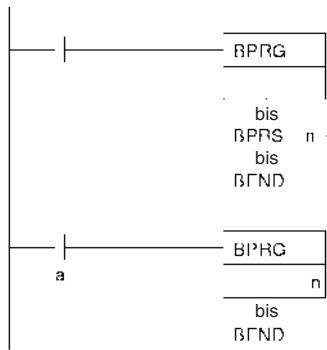
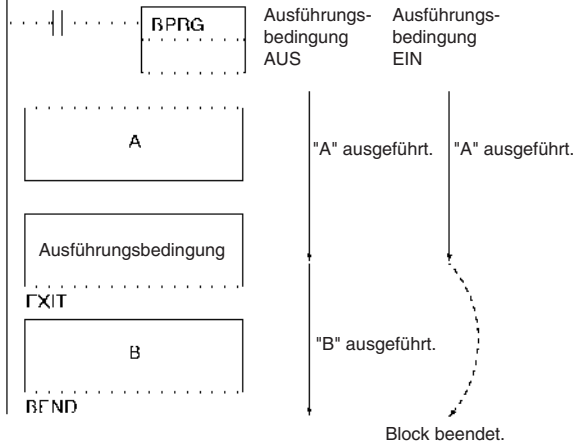
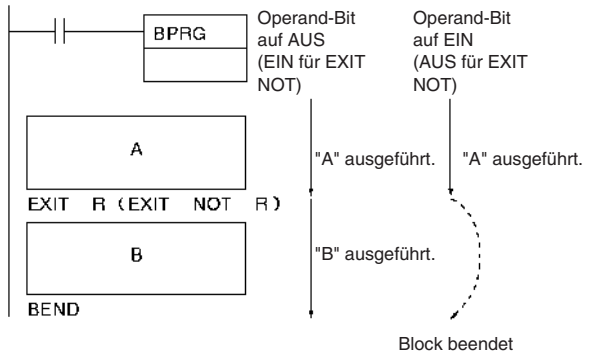
3-29 Sonstige Befehle

Befehl AWL Code	Symbol/Operand	Funktion	Position Ausführungs- bedingung
SET CARRY (Übertrags- merker setzen) STC @STC 040		Setzt den Übertragsmerker (CY).	Ausgabe Erforderlich
CLEAR CARRY (Übertragsmer- ker löschen) CLC @CLC 041		Setzt den Übertragsmerker (CY) auf AUS.	Ausgabe Erforderlich
SELECT EM BANK (EM-Bank auswählen) EMBC @EMBC 281	 N: Nummer der EM-Bank	Wechselt die aktuelle EM-Bank.	Ausgabe Erforderlich
EXTEND MAXIMUM CYCLE TIME (Maximale Zykluszeitüber- wachung erhöhen) WDT @WDT 094	 T: Zeitgebereinstellung	Verlängert die maximale Zykluszeit, allerdings nur für den Zyklus, in dem dieser Befehl ausgeführt wird.	Ausgabe Erforderlich
SAVE CONDITION FLAGS (Merker speichern) (nur CS1-H, CJ1-H, CJ1M und CS1D) CCS @CCS 282		Speichert den Status der Bedingungsmerker.	Ausgabe Erforderlich
LOAD CONDI- TION FLAGS (Merker laden) (nur CS1-H, CJ1-H, CJ1M und CS1D) CCL @CCL 283		Liest den Status der gespeicherten Bedingungsmerker.	Ausgabe Erforderlich
CONVERT ADDRESS FROM CV (Adresskon- vertierung von CV) (nur CS1-H, CJ1-H, CJ1M und CS1D) FRMCV @FRMCV 284	 S: Wort, das eine Speicheradresse der CV-Serie enthält D: Ziel- Indexregister	Konvertiert die Speicheradresse einer SPS der CV-Serie zur entsprechenden Speicheradresse einer SPS der CS/CJ-Serie.	Ausgabe Erforderlich
CONVERT ADDRESS TO CV (Adresskonver- tierung zu CV) (nur CS1-H, CJ1-H, CJ1M und CS1D) TOCV @TOCV 285	 S: Indexregister, das eine Speicheradresse der CS-Serie enthält D: Zielwort	Konvertiert die Speicheradresse einer SPS der CS/CJ-Serie zur entsprechenden Speicheradresse einer SPS der CV-Serie.	Ausgabe Erforderlich

Befehl AWL Code	Symbol/Operand	Funktion	Position Ausführungs- bedingung
DISABLE PERIPHERAL SERVICING (Peripherieservice deaktivieren) (nur CS1D CPU-Baugruppen für Einzel-CPU-Systeme, CS1-H, CJ1-H, und CJ1M) IOSP @ IOSP 287		Deaktiviert die Peripheriebedienung während der Programmausführung in einem der Parallelverarbeitungsmodi bzw. im Peripherieprioritätsmodus.	Ausgabe Erforderlich
ENABLE PERIPHERAL SERVICING (Peripherieservice aktivieren) (nur CS1D CPU-Baugruppen für Einzel-CPU-Systeme, CS1-H, CJ1-H, und CJ1M) IORS 288		Aktiviert die Peripheriebedienung, die durch IOSP(287) für die Programmausführung in einem der Parallelverarbeitungsmodi bzw. im Peripherieprioritätsmodus deaktiviert wurde.	Ausgabe Nicht erforderlich

3-30 Blockprogrammierungsbefehle

Befehl AWL Code	Symbol/Operand	Funktionen	Position Ausführungs- bedingung
BLOCK PROGRAM BEGIN (Blockprogramm Anfang) BPRG 096	 N: Blockprogramm-Nummer	Definiert einen Blockprogrammierungsbereich. Für jeden BPRG(096)-Befehl muss es einen entsprechenden BEND(801)-Befehl geben. 	Ausgabe Erforderlich
BLOCK PROGRAM END (Blockprogramm Ende) BEND 801		Definiert einen Blockprogrammierungsbereich. Für jeden BPRG(096)-Befehl muss es einen entsprechenden BEND(801)-Befehl geben.	Blockprogramm Erforderlich
BLOCK PROGRAM PAUSE (Blockprogramm-Pause) BPPS 811	 N: Blockprogramm-Nummer	Anhalten des spezifizierten Blockprogramms und Neustart des Blockprogramms aus einem anderen Blockprogramm heraus. 	Blockprogramm Erforderlich

Befehl AWL Code	Symbol/Operand	Funktionen	Position Ausführungs- bedingung
BLOCK PROGRAM RESTART (Blockpro- gramm-Restart) BPRS 812	BPRS (812)  N: Blockprogramm- Nummer	Anhalten des spezifizierten Blockprogramms und Neustart des Blockprogramms aus einem anderen Blockprogramm heraus. 	Blockprogramm Erforderlich
CONDITIONAL BLOCK EXIT (Bedingtes Blokke, Block verlassen) EXIT 806	EXIT(806) B: Bit-Operand	EXIT(806) ohne ein Operand-Bit verlässt das Programm, wenn die Ausführungsbedingung den Zustand EIN hat. 	Blockprogramm Erforderlich
CONDITIONAL BLOCK EXIT (Bedingtes Blokke, Block verlassen) EXIT 806	EXIT(806)B B: Bit-Operand	EXIT(806) ohne ein Operand-Bit verlässt das Programm, wenn die Ausführungsbedingung den Zustand EIN hat. 	Blockprogramm Erforderlich
CONDITIONAL BLOCK EXIT NOT (Bedingtes Blokke, Block verlassen (Nicht)) EXIT NOT 806	EXITNOT(806) B B: Bit-Operand	EXIT(806) ohne ein Operanden-Bit verlässt das Programm, wenn die Ausführungsbedingung den Zustand AUS hat.	Blockprogramm Erforderlich

Befehl AWL Code	Symbol/Operand	Funktionen	Position Ausführungs- bedingung
CONDITIONAL BLOCK BRAN- CHING (Bedingte Blockverzwei- gung) IF 802	IF(802)	Wenn die Ausführungsbedingung auf EIN gesetzt ist, werden die Befehle zwischen IF(802) und ELSE(803) ausgeführt. Wenn die Ausführungsbedingung auf AUS gesetzt ist, werden die Befehle zwischen ELSE(803) und IEND(804) ausgeführt. <pre> graph TD A[Ausführungs- bedingung] --> B{Ausführungs- bedingung EIN?} B -- YES --> C["A" ausgeführt (zwischen IF und ELSE).] B -- NO --> D["B" ausgeführt (nach ELSE)."] C --> E[IEND] D --> E </pre> Ausführungs- bedingung IF A ELSE B IEND	Blockprogramm Erforderlich
CONDITIONAL BLOCK BRAN- CHING (Bedingte Blockverzwei- gung) IF 802	IF(802) B B: Bit-Operand	Wenn das Operand-Bit auf EIN gesetzt ist, werden die Befehle zwischen IF(802) und ELSE(803) ausgeführt. Wenn das Operand-Bit auf AUS gesetzt ist, werden die Befehle zwischen ELSE(803) und IEND(804) ausgeführt. <pre> graph TD A[IF R (IF NOT R)] --> B{Operand-Bit EIN?} B -- YES --> C["A" ausgeführt (zwischen IF und ELSE)."] B -- NO --> D["B" ausgeführt (nach ELSE)."] C --> E[IEND] D --> E </pre> IF R (IF NOT R) A ELSE B IEND	Blockprogramm Erforderlich
CONDITIONAL BLOCK BRANCHING (NOT) (Bedingte Blockverzwei- gung (Nicht)) IF NOT 802	IFNOT (802) B B: Bit-Operand	Die Befehle zwischen IF(802) und ELSE(803) werden ausgeführt, wenn das Operanden-Bit EIN ist, und die Befehle zwischen ELSE(803) und IEND(804) werden ausgeführt, wenn das Operanden-Bit AUS ist.	Blockprogramm Erforderlich
CONDITIONAL BLOCK BRANCHING (ELSE) (Bedingte Blockverzwei- gung (ELSE)) ELSE 803	---	Wenn der Befehl ELSE(803) nicht vorhanden ist und das Operanden-Bit den Zustand EIN hat, werden die Befehle zwischen IF(802) und IEND(804) ausgeführt.	Blockprogramm Erforderlich
CONDITIONAL BLOCK BRANCHING END (Bedingte Blockverzwei- gung ENDE) IEND 804	---	Wenn das Operanden-Bit den Zustand AUS hat, werden nur die Befehle hinter IEND(804) ausgeführt.	Blockprogramm Erforderlich

Befehl AWL Code	Symbol/Operand	Funktionen	Position Ausführungs- bedingung
ONE CYCLE AND WAIT (Ein Zyklus und warten) WAIT 805	WAIT(805)	Wenn die Ausführungsbedingung für WAIT(805) auf EIN gesetzt ist, werden die restlichen Befehle im Blockprogramm übersprungen. Ausführungs- bedingung AUS Ausführungs- bedingung AUS Ausführungs- bedingung EIN "A" ausgeführt. "B" ausgeführt. "C" ausgeführt. Warten	Blockprogramm Erforderlich
ONE CYCLE AND WAIT (Ein Zyklus und warten) WAIT 805	WAIT(805) B B: Bit-Operand	Wenn das Operanden-Bit den Zustand AUS hat (EIN bei WAIT-NOT(805)), werden die restlichen Befehle im Blockprogramm übersprungen. Im nächsten Zyklus wird kein Teil der Blockprogramms ausgeführt, außer der Teil für die Ausführungsbedingung für WAIT(805) bzw. WAITNOT (805). Wenn die Ausführungsbedingung auf EIN wechselt (AUS bei WAITNOT (805)), werden die Befehle von WAIT(805) bzw. WAITNOT (805) bis zum Ende des Programms ausgeführt.	Blockprogramm Erforderlich
ONE CYCLE AND WAIT (NOT) WAITNOT 805	WAITNOT (805) B B: Bit-Operand	Wenn das Operanden-Bit den Zustand AUS hat (EIN bei WAIT-NOT(805)), werden die restlichen Befehle im Blockprogramm übersprungen. Im nächsten Zyklus wird kein Teil der Blockprogramms ausgeführt, außer der Teil für die Ausführungsbedingung für WAIT(805) bzw. WAITNOT (805). Wenn die Ausführungsbedingung auf EIN wechselt (AUS bei WAIT(805) NOT), werden die Befehle von WAIT(805) bzw. WAITNOT (805) bis zum Ende des Programms ausgeführt.	Blockprogramm Erforderlich
TIMER WAIT (Zeitgeber warten) TIMW 813 (BCD) TIMWX 816 (Binär) (nur CS1-H, CJ1-H, CJ1M und CS1D)	TIMW(813) N SV N: Zeitgeber- nummer SV: Sollwert TIMWX(816) N SV N: Zeitgeber- nummer SV: Sollwert	Verzögert die Ausführung des Blockprogramms, bis die spezifizierte Zeitspanne verstrichen ist. Die Ausführung fährt mit dem nächsten Befehl nach TIMW(813)/TIMWX(816) fort, wenn der Zeitgeber abgelaufen ist. Sollwert: 0 bis 999,9 s bei BCD und 0 bis 6.553,5 s bei binär "A" ausgeführt. Sollwert (Voreinstellung) Zeit abgelaufen. "B" ausgeführt. "C" ausgeführt.	Blockprogramm Erforderlich

Befehl	Symbol/Operand	Funktionen	Position Ausführungs- bedingung
COUNTER WAIT (Zähler warten) CNTW 814 (BCD) CNTWX 818 (Binär) (nur CS1-H, CJ1- H, CJ1M und CS1D)	CNTW(814) N SV N: Zähler- nummer SV: Sollwert I: Zählengang CNTWX(818) N SV N: Zähler- nummer SV: Sollwert I: Zählengang	Verzögert die Ausführung des restlichen Blockprogramms, bis der spezifizierte Zählwert erreicht ist. Die Ausführung fährt mit dem nächsten Befehl nach CNTW(814)/CNTWX(818) fort, wenn der Zähler abgelaufen ist. Sollwert: 0 bis 9.999 bei BCD und 0 bis 65.535 bei binär	Blockprogramm Erforderlich
HIGH-SPEED TIMER WAIT (Schneller Zeit- geber warten) TMHW 815 (BCD) TMHWX 817 (Binär) (nur CS1-H, CJ1- H, CJ1M und CS1D)	TMHW(815) N SV N: Zeitgeber- nummer SV: Sollwert TMHWX(817) N SV N: Zeitgeber- nummer SV: Sollwert	Verzögert die Ausführung des restlichen Blockprogramms, bis die angegebene Zeitspanne verstrichen ist. Die Ausführung fährt mit dem nächsten Befehl nach TMHW(815) fort, wenn der Zeitgeber abgelaufen ist. Sollwert: 0 bis 99,99 s bei BCD und 0 bis 655,35 s bei binär	Blockprogramm Erforderlich

Befehl AWL Code	Symbol/Operand	Funktionen	Position Ausführungs- bedingung
LOOP (Pro- grammschleife) LOOP 809	---	LOOP(809) bezeichnet den Anfang einer Programmschleife.	Blockprogramm Erforderlich
LEND (Programm- schleifenende) LEND 810	LEND(810)	LEND(810) bzw. LENDNOT (810) geben das Ende der Schleife an. Wenn LEND(810) bzw. LENDNOT (810) erreicht ist, geht die Programmausführung zurück zum vorhergehenden Befehl LOOP(809), bis das Operanden-Bit für LEND(810) bzw. LENDNOT (810) auf EIN bzw. AUS wechselt oder bis die Ausführungsbedingung für LEND(810) auf EIN wechselt.	Blockprogramm Erforderlich
LEND (Programm- schleifenende) LEND 810	LEND(810) B B: Bit-Operand	Wenn das Operand-Bit für LEND(810) auf AUS gesetzt ist (oder auf EIN bei LENDNOT(810)) wird die Ausführung der Schleife beginnend mit dem nächsten Befehl nach LOOP(809) wiederholt. Wenn das Operand-Bit für LEND(810) auf EIN gesetzt ist (oder auf AUS bei LENDNOT(810)) wird die Schleife beendet, und die Ausführung wird mit dem nächsten Befehl nach LEND(810) bzw. LENDNOT (810) fortgeführt. Hinweis Der Status des Operand-Bits wird im Falle von LENDNOT(810) umgekehrt.	Blockprogramm Erforderlich
LENDNOT LENDNOT 810	LENDNOT (810) B: Bit-Operand	LEND(810) bzw. LENDNOT (810) geben das Ende der Schleife an. Wenn LEND(810) bzw. LENDNOT (810) erreicht ist, geht die Programmausführung zurück zum vorhergehenden Befehl LOOP(809), bis das Operanden-Bit für LEND(810) bzw. LENDNOT (810) auf EIN bzw. AUS wechselt oder bis die Ausführungsbedingung für LEND(810) auf EIN wechselt.	Blockprogramm Erforderlich

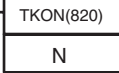
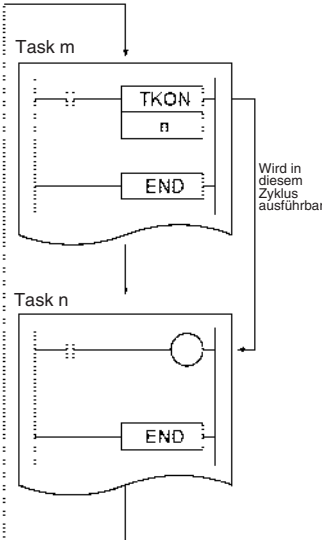
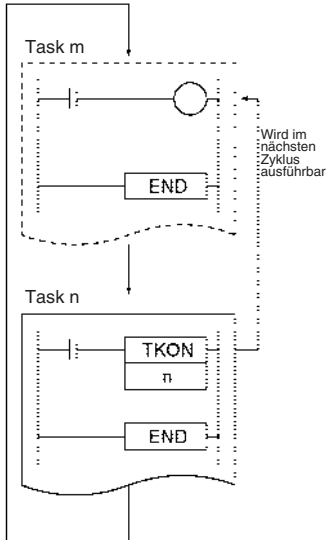
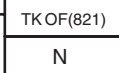
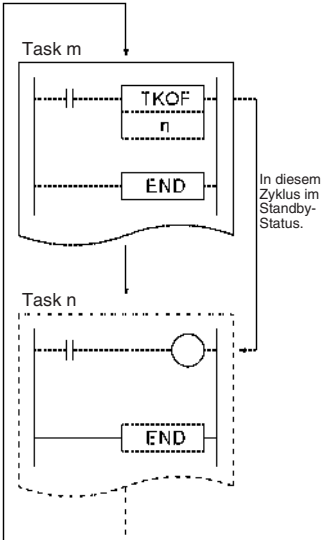
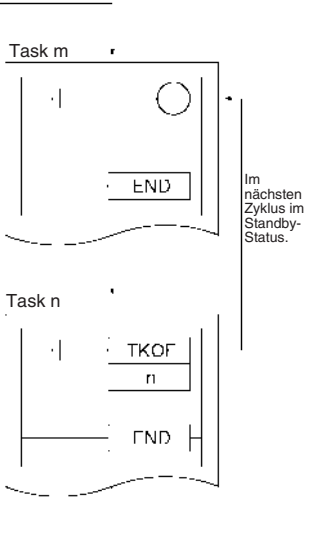
3-31 Textzeichenketten-Verarbeitungsbefehle

Befehl AWL Code	Symbol/Operand	Funktion	Position Ausführungs- bedingung
MOV STRING (Zeichenkette übertragen) MOV\$ @MOV\$ 664	S: Erstes Quellwort D: Erstes Zielwort	Überträgt eine Textzeichenkette.	Ausgabe Erforderlich
CONCATENATE STRING (Zeichenketten aneinander- hängen) +\$ @+\$ 656	S1: Textzeichen- kette 1 S2: Textzeichen- kette 2 D: Erstes Zielwort	Verknüpft eine Textzeichenkette mit einer anderen Textzeichenkette	Ausgabe Erforderlich
GET STRING LEFT (Zeichen- kette links extrahieren) LEFT\$ @LEFT\$ 652	S1: Erstes Wort der Textzeichenkette S2: Anzahl Zeichen D: Erstes Zielwort	Liest eine angegebene Anzahl von Zeichen von der linken Seite (dem Anfang) einer Textzeichenkette.	Ausgabe Erforderlich
GET STRING RIGHT (Zeichen- kette rechts extrahieren) RGHT\$ @RGHT\$ 653	S1: Erstes Wort der Textzeichenkette S2: Anzahl Zeichen D: Erstes Zielwort	Liest eine angegebene Anzahl von Zeichen von der rechten Seite (dem Ende) einer Textzeichenkette	Ausgabe Erforderlich
GET STRING MIDDLE (Zeichenkette Auszug von einer beliebigen Position) MID\$ @MID\$ 654	S1: Erstes Wort der Textzeichenkette S2: Anzahl Zeichen S3: Anfangsposition D: Erstes Zielwort	Liest eine spezifizierte Anzahl von Zeichen ab einer beliebigen Stelle innerhalb einer Textzeichenkette	Ausgabe Erforderlich

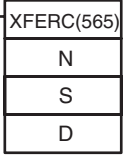
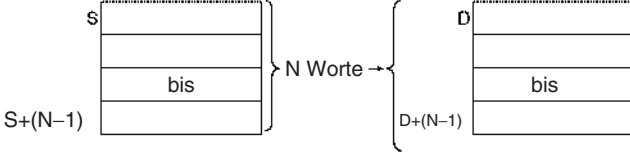
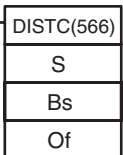
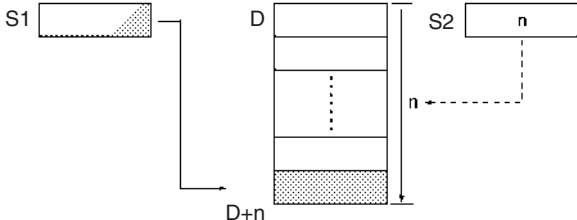
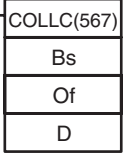
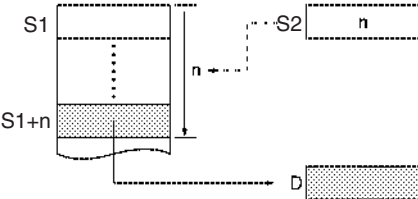
Befehl	AWL Code	Symbol/Operand	Funktion	Position Ausführungsbedingung																							
FIND IN STRING (Auffinden in Zeichenkette) FIND @FIND\$ 660	FIND\$(660) S1 S2 D S1: Erstes Wort der Quelltextzeichenkette S2: Erstes Wort der gefundenen Textzeichenkette D: Erstes Zielwort	Sucht eine bestimmte Textzeichenkette innerhalb einer Textzeichenkette. Gefundene Daten S1 → <table><tr><td>A</td><td>B</td></tr><tr><td>C</td><td>D</td></tr><tr><td>E</td><td>F</td></tr><tr><td>NUL</td><td>NUL</td></tr></table> S2 → <table><tr><td>C</td><td>NUL</td></tr></table> D → <table><tr><td>00</td><td>03</td></tr><tr><td></td><td></td></tr></table>	A	B	C	D	E	F	NUL	NUL	C	NUL	00	03			Ausgabe Erforderlich										
A	B																										
C	D																										
E	F																										
NUL	NUL																										
C	NUL																										
00	03																										
STRING LENGTH (Zeichenkettenlänge ermitteln) LEN\$ @LEN\$ 650	LEN\$(650) S D S: Erstes Wort der Textzeichenkette D: Erstes Zielwort	Berechnet die Länge einer Textzeichenkette. S1 → <table><tr><td>1</td><td>2</td></tr><tr><td>3</td><td>4</td></tr><tr><td>5</td><td>NUL</td></tr></table> D → <table><tr><td>00</td><td>05</td></tr></table>	1	2	3	4	5	NUL	00	05	Ausgabe Erforderlich																
1	2																										
3	4																										
5	NUL																										
00	05																										
REPLACE IN STRING (Zeichenkette ersetzen) RPLC\$ @RPLC\$ 661	RPLC\$(661) S1 S2 S3 S4 D S1: Erstes Wort der Textzeichenkette S2: Erstes Wort der Austausch-Textzeichenkette S3: Anzahl Zeichen S4: Anfangsposition D: Erstes Zielwort	Ersetzt eine Textzeichenkette durch eine angegebene Textzeichenkette ab einer bestimmten Position. S1 → <table><tr><td>A</td><td>B</td></tr><tr><td>C</td><td>D</td></tr><tr><td>E</td><td>F</td></tr><tr><td>G</td><td>H</td></tr><tr><td>I</td><td>NUL</td></tr></table> S2 → <table><tr><td>K</td><td>NUL</td></tr></table> S3 → <table><tr><td>00</td><td>04</td></tr></table> S4 → <table><tr><td>00</td><td>05</td></tr></table> D → <table><tr><td>A</td><td>B</td></tr><tr><td>C</td><td>D</td></tr><tr><td>K</td><td>I</td></tr><tr><td>NUL</td><td>NUL</td></tr></table>	A	B	C	D	E	F	G	H	I	NUL	K	NUL	00	04	00	05	A	B	C	D	K	I	NUL	NUL	Ausgabe Erforderlich
A	B																										
C	D																										
E	F																										
G	H																										
I	NUL																										
K	NUL																										
00	04																										
00	05																										
A	B																										
C	D																										
K	I																										
NUL	NUL																										
DELETE STRING (Zeichenkette löschen) DEL\$ @DEL\$ 658	DEL\$(658) S1 S2 S3 D S1: Erstes Wort der Textzeichenkette S2: Anzahl Zeichen S3: Anfangsposition D: Erstes Zielwort	Löscht eine spezifizierte Textzeichenkette aus der Mitte einer Textzeichenkette. Anzahl der zu löschenden Zeichen (spezifiziert durch S2). S1 → <table><tr><td>A</td><td>B</td></tr><tr><td>C</td><td>D</td></tr><tr><td>E</td><td>F</td></tr><tr><td>G</td><td>H</td></tr><tr><td>I</td><td>NUL</td></tr></table> S2 → <table><tr><td>00</td><td>05</td></tr></table> D → <table><tr><td>A</td><td>B</td></tr><tr><td>C</td><td>D</td></tr><tr><td>H</td><td>I</td></tr><tr><td>NUL</td><td>NUL</td></tr></table>	A	B	C	D	E	F	G	H	I	NUL	00	05	A	B	C	D	H	I	NUL	NUL	Ausgabe Erforderlich				
A	B																										
C	D																										
E	F																										
G	H																										
I	NUL																										
00	05																										
A	B																										
C	D																										
H	I																										
NUL	NUL																										

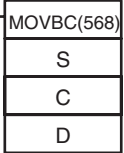
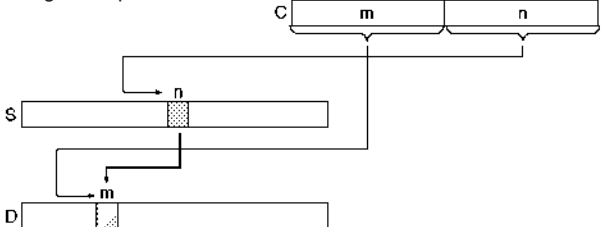
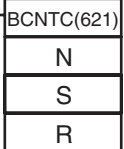
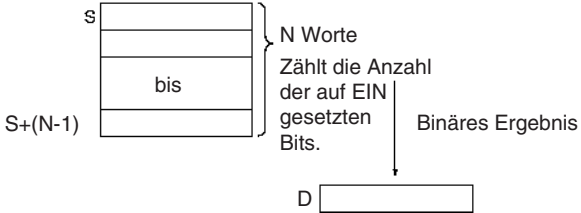
Befehl	Symbol/Operand	Funktion	Position Ausführungs- bedingung
EXCHANGE STRING (Zeichenkette austauschen) XCHG\$ @XCHG\$ 665	XCHG\$(665) Ex1 Ex2 Ex1: Erstes Austauschwort 1 Ex2: Erstes Austauschwort 2	Ersetzt eine spezifizierte Textzeichenkette durch eine andere spezifizierte Textzeichenkette. 	Ausgabe Erforderlich
CLEAR STRING (Zeichenkette auf ASCII Null zurücksetzen) CLR\$ @CLR\$ 666	CLR\$(666) S S: Erstes Wort der Textzeichenkette	Löscht eine komplette Textzeichenkette mit NUL (00 Hex) 	Ausgabe Erforderlich
INSERT INTO STRING (Zeichenkette einfügen) INS\$ @INS\$ 657	INS\$(657) S1 S2 S3 D S1: Erstes Wort der Basis-Textzeichenkette S2: Erstes Wort der eingefügten Textzeichenkette S3: Anfangsposition D: Erstes Zielwort	Fügt eine spezifizierte Textzeichenkette in die Mitte einer Textzeichenkette ein. 	Ausgabe Erforderlich
Zeichenkettenvergleich LD, AND, OR + =\$, <>\$, <\$, <=\$, >\$, >=\$ 670 (=)\$ 671 (<>\$) 672 (<\$) 673 (<=\$) 674 (>\$) 675 (>=\$)	LD Symbol S1 S2 AND Symbol S1 S2 OR Symbol S1 S2 S1: Textzeichenkette 1 S2: Textzeichenkette 2	Zeichenketten-Vergleichsbefehle (=, <>, <, <=, >, >=) vergleichen zwei Textzeichenketten, an deren Anfang beginnend. Dabei werden die Werte der ASCII-Codes verglichen. Wenn das Ergebnis des Vergleichs wahr ist, wird eine EIN-Ausführungsbedingung für LOAD, AND oder OR erzeugt.	LD: nicht erforderlich AND, OR: erforderlich

3-32 Task-Steuerbefehle

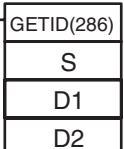
Befehl	Symbol/Operand	Funktion	Position Ausführungs- bedingung
TASK ON (Programm-Task EIN) TKON @TKON 820	 N: Task-Nummer	Macht die spezifizierte Task ausführbar. Die Nummer der spezifizierten Task ist höher als die Nummer der lokalen Task ($m < n$). Die Nummer der spezifizierten Task ist niedriger als die Nummer der lokalen Task ($m > n$).  	Ausgabe Erforderlich
TASK OFF (Programm-Task AUS) TKOF @TKOF 821	 N: Task-Nummer	Versetzt die angegebene Task in den Standby-Status. Die Nummer der spezifizierten Task ist höher als die Nummer der lokalen Task ($m < n$). Die Nummer der spezifizierten Task ist niedriger als die Nummer der lokalen Task ($m > n$).  	Ausgabe Erforderlich

3-33 Modell-Konvertierungsbefehle (ab CPU-Baugruppen Ver. 3.0)

Befehl Mnemo- code	Symbol/Operand	Funktion	Mit/ohne Ausführungs- bedingung
Block Übertragung XFERC @XFERC 565	 N: Anzahl der Worte S: Erstes Quellwort D: Erstes Zielwort	Überträgt die spezifizierte Anzahl von aufeinanderfolgenden Worten. 	Ausführungs- bedingung erforderlich
Einzelwort- verteilung DISTC @DISTC 566	 S: Quellwort Bs: Ziel-Basisadresse Of: Offset	Überträgt das Quellwort an das durch Addition eines Offsetwerts zur Basisadresse errechnetes Zielwort.  Kann auch auf einen Stack schreiben (Push-Operation).	Ausführungs- bedingung erforderlich
Daten sammeln COLLC @COLLC 567	 Bs: Quellen-Basisadresse Of: Offset (BCD) D: Zielwort	Überträgt das Quellwort (errechnet durch Addition eines Offsetwerts zur Basisadresse) an das Zielwort.  Kann auch von einem Stack lesen (Pop-Operation).	Ausführungs- bedingung erforderlich

Befehl Mnemo- code	Symbol/Operand	Funktion	Mit/ohne Ausführungs- bedingung
Bit übertragen MOVBC @MOVBC 568	 S: Quellwort oder -daten C: Steuerwort (BCD) D: Zielwort	Überträgt das spezifizierte Bit. 	Ausführungs- bedingung erforderlich
Bit-Zähler BCNTC @BCNTC 621	 N: Anzahl der Worte (BCD) S: Erstes Quellwort R: Ergebniswort	Zählt die Gesamtanzahl der auf EIN gesetzten Bits in den spezifizierten Worten. 	Ausführungs- bedingung erforderlich

3-34 Spezielle Funktionsblockbefehle

Befehl Mnemo- code	Symbol/Operand	Funktion	Ort Ausführungs- bedingung
ID-Typ bestimmen GETID @GETID 286	 S: Variable oder Adresse D1: ID-Code D2: Zielwort	Gibt den FINS-Variablentyp-Code (Datenbereich) und die Wortadresse der angegebenen Variable oder Adresse aus. Dieser Befehl wird im allgemeinen verwendet, um die einer Variable in einem Funktionsblock zugeordnete Adresse zu ermitteln.	Ausführungs- bedingung erforderlich

ABSCHNITT 4

Tasks

Dieser Abschnitt enthält eine Beschreibung der Funktionen von Tasks.

4-1	Merkmale von Tasks	158
4-1-1	Übersicht.	158
4-1-2	Tasks und Programme.	159
4-1-3	Prinzipieller Betriebsablauf der CPU-Baugruppe	160
4-1-4	Arten von Tasks	162
4-1-5	Task-Ausführungsbedingungen und Einstellungen	164
4-1-6	Status der zyklischen Tasks	165
4-1-7	Statuswechsel	166
4-2	Verwenden von Tasks	167
4-2-1	TASK ON und TASK OFF.	167
4-2-2	Einschränkungen bei Task-Befehlen	170
4-2-3	Zu Tasks gehörige Merker	171
4-2-4	Erstellen von Tasks	175
4-2-5	Globale Unterprogramme	176
4-3	Interrupt-Tasks	177
4-3-1	Arten von Interrupt-Tasks	177
4-3-2	Prioritäten von Interrupt-Tasks	184
4-3-3	Merker und Worte für Interrupt-Tasks	185
4-3-4	Sicherheitshinweise zum Einsatz	186
4-4	Task-Programmierung mit Programmiergeräten.	189
4-4-1	Verwenden mehrerer zyklischer Tasks	189
4-4-2	Bedienverfahren bei Programmiergeräten	189

4-1 Merkmale von Tasks

4-1-1 Übersicht

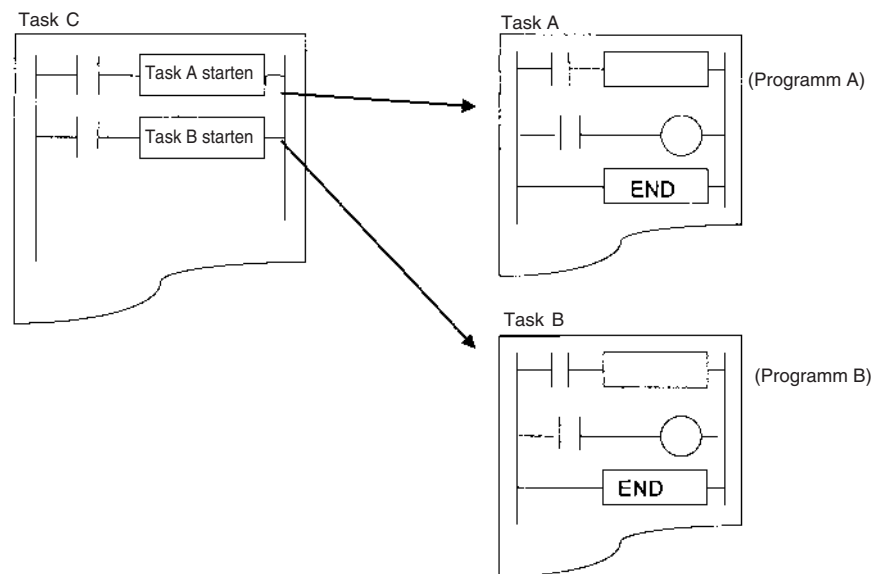
Steuervorgänge können bei SPS der CS/CJ-Serie nach Funktionen, gesteuerten Geräte, Prozessen, Entwicklern oder beliebigen anderen Kriterien eingeteilt werden. Jeder dieser Vorgänge kann in separaten Einheiten, den so genannten Tasks programmiert werden. Das Verwenden von Tasks bietet folgende Vorteile:

1,2,3...

1. Programme können von mehreren Personen gleichzeitig entwickelt werden. Individuell entwickelte Programmteile können ohne größeren Aufwand zu einem einzelnen Anwenderprogramm zusammengestellt werden.
2. Programme können in Modulen standardisiert werden.
Darüber hinaus werden folgende Programmiersoftwarefunktionen kombiniert, um Programme zu entwickeln, bei denen es sich nicht um Programme für bestimmte Systeme (Maschinen, Geräte) sondern um eigenständige Standardmodule handelt. Daher lassen sich von mehreren Personen separat entwickelte Programme problemlos kombinieren.
 - Programmierung unter Verwendung von Symbolen
 - Globale und lokale Bestimmung von Symbolen
 - Automatische Zuordnung von lokalen Symbolen zu Adressen
3. Insgesamt verbesserte Ansprechleistung.
Die Gesamt-Ansprechleistung wird verbessert, da das System in ein übergeordnetes Steuerungsprogramm und in individuelle Steuerungsprogramme aufgeteilt wird, und lediglich bestimmte erforderliche Programme ausgeführt werden.
4. Einfache Überarbeitung und Fehlerbehebung.
 - Das Debugging ist effizienter, da Tasks von mehreren Personen separat entwickelt werden können und die Überarbeitung und die Fehlerbehebung anschließend Task für Task durchgeführt werden kann.
 - Die Wartung ist einfach, da lediglich die zu überarbeitende Task geändert werden muss, um Änderungen an den Spezifikationen oder sonstige Änderungen vorzunehmen.
 - Die Fehlerbehebung ist effizienter, da problemlos festgestellt werden kann, ob eine Adresse spezifisch oder global ist. Adressen zwischen Programmen müssen lediglich einmal während der Ablaufkontrolle überprüft werden, da Symbole global oder lokal bestimmt und lokale Symbole von Programmiersoftware automatisch zu Adressen zugeordnet werden.
5. Einfacher Wechsel zwischen Programmen.
Wenn eine Änderung des Betriebs erforderlich ist, kann ein Task-Steuerbefehl des Programms zur Ausführung produktspezifischer Tasks (Programme) verwendet werden.

6. Leicht verständliche Anwenderprogramme.

Programme werden in Blöcken strukturiert, die diese bei Abschnitten besser verständlich machen, die gewöhnlich z. B. mit Sprungbefehlen gehandhabt werden.



4-1-2 Tasks und Programme

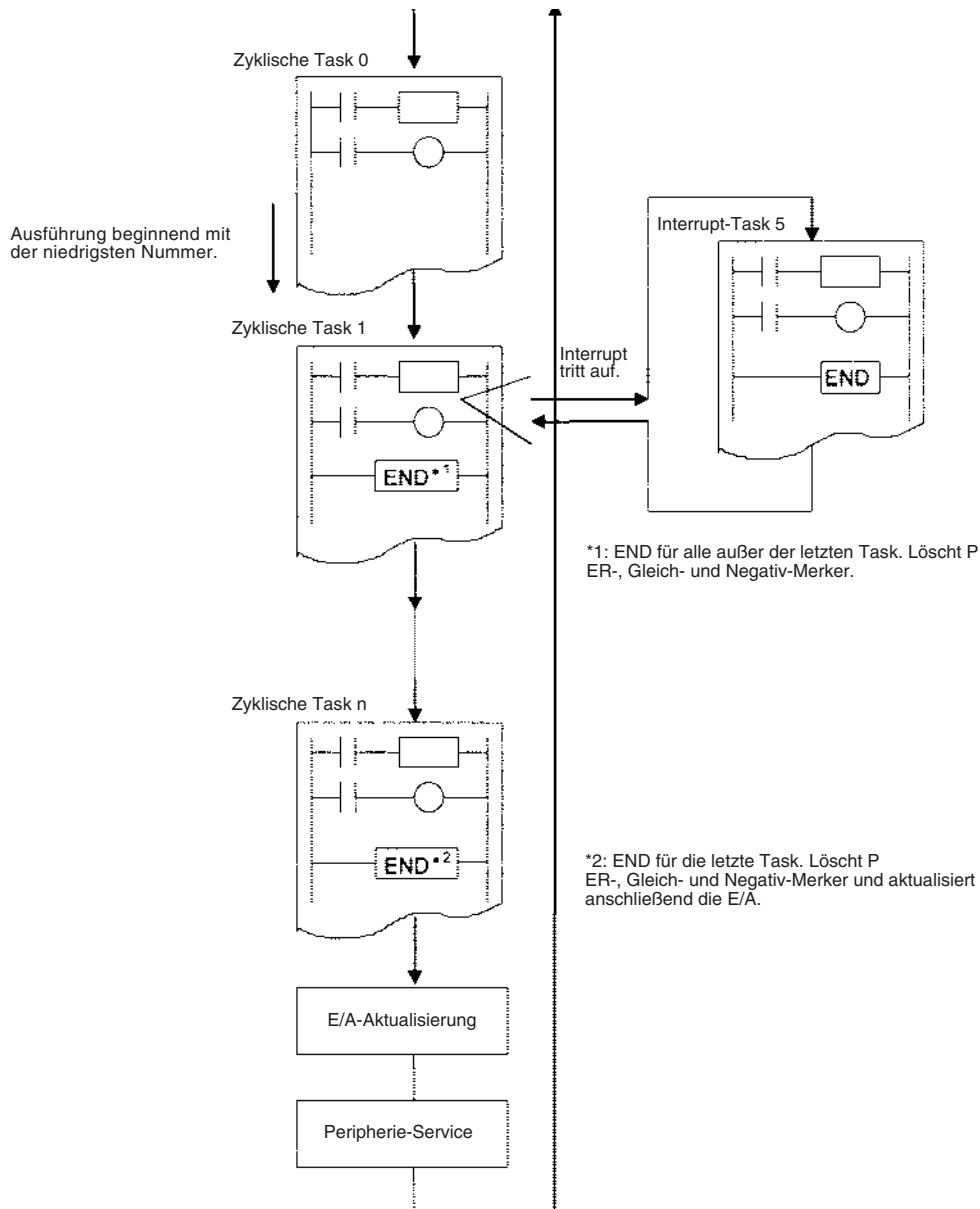
- Es können bis zu 288 Programme (Tasks) gesteuert werden. Individuelle Programme werden 1:1 zu Tasks zugeordnet. Tasks sind im Allgemeinen in die folgenden Typen unterteilt:
- Zyklische Tasks
- Interrupt-Tasks

- Hinweis**
1. Es können bis zu 32 zyklische Tasks und 256 Interrupt-Tasks (also insgesamt 288 Tasks) erstellt werden. Jede Task verfügt über eine eindeutige Nummer zwischen 0 bis 31 bei zyklischen Tasks und 0 bis 255 bei Interrupt-Tasks.
 2. Bei den CPU-Baugruppen CS1-H, CJ1-H, CJ1M oder CS1D können Interrupt-Tasks (Interrupt-Task-Nummern 0 bis 255) als zyklische Tasks ausgeführt werden, indem diese mit dem Befehl TKON gestartet werden. Diese werden als „zyklische Zusatztasks“ bezeichnet. Durch die Verwendung von zyklischen Zusatztasks können insgesamt 288 zyklische Tasks verwendet werden.
 3. Interrupt-Eingangs-Baugruppen können nur von CJ1-H oder CJ1M CPU-Baugruppen unterstützt werden, nicht jedoch von CJ1 CPU-Baugruppen. Demzufolge beträgt die Höchstzahl der Tasks für eine CJ1 CPU-Baugruppe 35, d. h. 32 zyklische Tasks und 3 Interrupt-Tasks. Die Gesamtzahl der Programme, die erstellt und gesteuert werden können, beträgt ebenfalls 35.

Jedes einer Task zugeordnete Programm muss mit dem Befehl END(001) beendet werden. Eine E/A-Aktualisierung wird erst ausgeführt, nachdem sämtliche Taskprogramme in einem Zyklus ausgeführt wurden.

4-1-3 Prinzipieller Betriebsablauf der CPU-Baugruppe

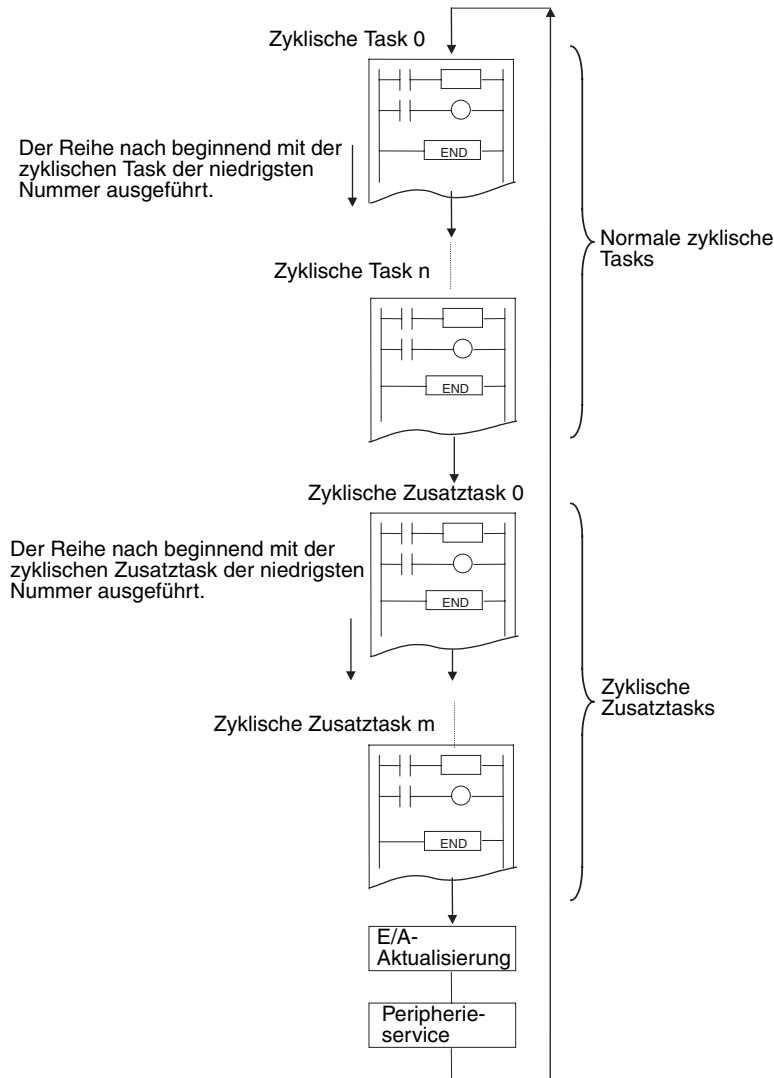
Die CPU-Baugruppe führt zyklische Tasks, beginnend mit der kleinsten Nummer, aus (bei den CPU-Baugruppen CS1-H, CJ1-H, CJ1M oder CS1D einschließlich zyklischer Zusatztasks). Die Ausführung der zyklischen Tasks wird unterbrochen, um eine Interrupt-Task auszuführen, wenn ein Interrupt auftritt.



Hinweis

Alle Bedingungsmerker (P_ER, P_CY, P_EQ, P_AER, usw.) sowie Befehlsbedingungen (Sperre (IL) EIN, usw.) werden beim Start einer Task gelöscht. Daher können Bedingungsmerker nicht Task-übergreifend gelesen und die Befehle INTERLOCK/INTERLOCK CLEAR (IL/ILC), JUMP/JUMP END (JMP/JME) oder SUBROUTINE CALL/SUBROUTINE ENTRY (SBS/SBN) nicht zwischen zwei Tasks aufgeteilt werden.

Bei den CPU-Baugruppen CS1-H, CJ1-H, CJ1M oder CS1D können Interrupt-Tasks als zyklische Tasks ausgeführt werden, indem diese mit dem Befehl TKON gestartet werden. Diese werden als „zyklische Zusatztasks“ bezeichnet. Zyklische Zusatztasks (Interrupt-Task-Nummern 0 bis 255) werden beginnend mit der niedrigsten Tasknummer nach der vollständigen Ausführung der zyklischen Tasks (Haupt-Tasks 0 bis 31) ausgeführt.



4-1-4 Arten von Tasks

Tasks werden entweder als zyklische Tasks oder als Interrupt-Tasks klassifiziert. Interrupt-Tasks werden weiter in Ausschalt-, zeitgesteuerte, E/A- und externe Interrupt-Tasks untergeteilt. Interrupt-Tasks können auch als zyklische Zusatztasks ausgeführt werden.

Hinweis Bei den CPU-Baugruppen CS1-H, CJ1-H, CJ1M oder CS1D können Interrupt-Tasks als zyklische Tasks ausgeführt werden, indem diese mit dem Befehl TKON gestartet werden. Diese werden als „zyklische Zusatztasks“ bezeichnet.

Zyklische Tasks

Ausführbereite (Aktive) Tasks werden in numerischer Reihenfolge, beginnend mit der Task mit der niedrigsten Nummer, einmal pro Zyklus ausgeführt (vom Beginn des Programms bis zum Befehl END(001)). Die maximale Anzahl zyklischer Tasks beträgt 32. (Zyklische Task-Nummern: 00 bis 31).

Hinweis Bei den CPU-Baugruppen CS1-H, CJ1-H, CJ1M oder CS1D können Interrupt-Tasks (Interrupt-Task-Nummern 0 bis 255) auf die gleiche Weise wie normale zyklische Tasks (Task-Nummern 0 bis 31) ausgeführt werden. Durch die Verwendung von zyklischen Zusatztasks können insgesamt 288 zyklische Tasks verwendet werden.

Interrupt-Tasks

Eine Interrupt-Task wird beim Auftreten eines Interrupts selbst dann ausgeführt, wenn gerade eine zyklische Task (einschließlich zyklischer Zusatztasks) ausgeführt wird. Die Interrupt-Task wird zu einem beliebigen Zeitpunkt im Zyklus ausgeführt, auch während der Ausführung des Anwenderprogramms, der E/A-Aktualisierung oder Peripherbedienung, wenn die Ausführungsbedingung für den Interrupt erfüllt ist.

Bei den CPU-Baugruppen CS1-H, CJ1-H, CJ1M oder CS1D können Interrupt-Tasks als zyklische Tasks ausgeführt werden. (Die CS1D CPU-Baugruppen für Duplex-CPU-Systeme unterstützen keine Interrupts. Bei den CS1D CPU-Baugruppen für Duplex-CPU-Systeme können Interrupt-Tasks lediglich als zyklische Zusatztasks verwendet werden.)

Zu Aktivierung von Interrupt-Tasks können die integrierten Interrupt- und Schnelle-Zähler-Eingänge einer CJ1M CPU-Baugruppe verwendet werden. Detaillierte Informationen hierzu finden Sie im *CJ-Serie Integrierte E/A CJ1M-CPU21/22/23 Bedienerhandbuch*.

Ausschalt-Interrupt-Task

Die Ausschalt-Interrupt-Task wird ausgeführt, wenn die Spannungsversorgung der CPU-Baugruppe ausgeschaltet wird. Es kann lediglich eine Ausschalt-Interrupt-Task programmiert werden (Interrupt-Task-Nummer: 1).

Hinweis Die Ausschalt-Interrupt-Task muss vor Ablauf folgender Zeitspanne ausgeführt werden, andernfalls wird die Task zwangsweise beendet.

10 ms – (Ausschalterkennungs-Verzögerungszeit)

Die Ausschalterkennungs-Verzögerungszeit wird im SPS-Setup festgelegt.

Zeitgesteuerte Interrupt-Tasks

Zeitgesteuerte Interrupt-Tasks werden entsprechend einem im internen Zeitgeber der CPU-Baugruppe eingestellten festen Intervall ausgeführt. Die maximale Anzahl zeitgesteuerter Interrupt-Tasks beträgt 2 (Interrupt-Task-Nummern: 2 und 3).

Hinweis Der Befehl SET INTERRUPT MASK (Interrupt-Maske)(MSKS(690)) legt den Interrupt für eine zeitgesteuerte Interrupt-Task fest. Die Interrupt-Zeiten können im SPS-Setup in 10-ms- oder 1,0-ms-Schritten festgelegt werden.

E/A-Interrupt-Tasks

Eine E/A-Interrupt-Task wird ausgeführt, wenn ein Eingang einer Interrupt-Eingangsbaugruppe eingeschaltet wird. Die maximale Anzahl von E/A-Interrupt-Tasks beträgt 32 (Interrupt-Task-Nummern: 100 bis 131). Die Interrupt-Eingangsbaugruppe muss im CPU-Rack installiert sein. Bei CJ1-H CPU-Baugruppen muss die Baugruppe als eine der fünf Baugruppen, die der CPU-

Baugruppe am nächsten liegen, installiert sein (Steckplätze 0 bis 4). Bei CJ1M CPU-Baugruppen muss die Baugruppe als eine der drei Baugruppen, die der CPU-Baugruppe am nächsten liegen, installiert sein (Steckplätze 0 bis 2). In anderen Steckplätzen installierte E/A-Interrupt-Baugruppen können nicht für die Ausführungsanforderung von E/A-Interrupt-Tasks verwendet werden.

E/A-Interrupts werden von CJ1 CPU-Baugruppen nicht unterstützt.

Externe Interrupt-Tasks

Eine externe Interrupt-Task wird ausgeführt, wenn diese von Anwenderprogrammen einer Spezial-E/A-Baugruppe, CPU-Bus-Baugruppe oder einem internen Steckmodul (nur CS-Serie) angefordert wird. Spezial-E/A-Baugruppen und CPU-Bus-Baugruppen müssen jedoch im CPU-Rack installiert sein. Bei CJ1-H-CPU-Baugruppen muss die Baugruppe als eine der fünf Baugruppen, die der CPU-Baugruppe am nächsten liegen, installiert sein (Steckplätze 0 bis 4). Bei CJ1M CPU-Baugruppen muss die Baugruppe als eine der drei Baugruppen, die der CPU-Baugruppe am nächsten liegen, installiert sein (Steckplätze 0 bis 2). In anderen Steckplätzen installierte Baugruppen können nicht für das Erzeugen externer Interrupts verwendet werden.

Die maximale Anzahl externer Interrupt-Tasks beträgt 256 (Interrupt-Task-Nummern: 0 bis 255). Wenn eine externe Interrupt-Task die gleiche Nummer wie eine Ausschalt-, zeitgesteuerte oder E/A-Interrupt-Task besitzt, wird die Interrupt-Task beim Eintreten einer der beiden Bedingungen ausgeführt (die beiden Bedingungen arbeiten mit ODER-Logik). Grundsätzlich sollten Tasknummern jedoch nicht doppelt vergeben werden.

E/A-Interrupts werden von CJ1 CPU-Baugruppen nicht unterstützt.

Zyklische Zusatztasks (nur CS1-H, CJ1-H, CJ1M und CS1D CPU- Baugruppen)

Eine Interrupt-Tasks kann, wie normale zyklische Tasks, in jedem Zyklus ausgeführt werden. Zyklische Zusatztasks (Interrupt-Task-Nummern 0 bis 255) werden beginnend mit der niedrigsten Tasknummer im Anschluss an die vollständige Ausführung der zyklischen Tasks (zyklische Task-Nummern 0 bis 31) ausgeführt. Die maximale Anzahl zyklischer Zusatztasks beträgt 256 (Interrupt-Task-Nummern: 0 bis 255). Zyklische Interrupt-Tasks unterscheiden sich jedoch insofern von zyklischen Tasks, dass sie mit dem Befehl TKON(820) gestartet werden. Auch können die Befehle TKON(820) und TKOF nicht in zyklischen Zusatztasks verwendet werden, was bedeutet, dass zyklische Tasks und andere zyklische Zusatztasks nicht aus einer zyklischen Zusatztasks heraus gesteuert werden können.

Wenn eine zyklische Zusatztask die gleiche Nummer wie eine Ausschalt-, zeitgesteuerte oder E/A-Interrupt-Task besitzt, wird die Interrupt-Task beim Eintreten einer der beiden Bedingungen ausgeführt (die beiden Bedingungen arbeiten mit ODER-Logik). Verwenden Sie Interrupt-Tasks nicht gleichzeitig als Interrupt-Tasks und zyklische Zusatztasks.

Hinweis

1. Die Ausschalt-Interrupt-Task (siehe oben unter 1) hat Vorrang und wird beim Ausschalten der Spannungsvorgang selbst dann ausgeführt, wenn gleichzeitig eine andere Interrupt-Task ausgeführt wird.
2. Wenn während der Ausführung einer Interrupt-Task ein zeitgesteuerter, E/A- oder externer Interrupt auftritt, werden diese Interrupt-Tasks erst nach dem Beenden dieser Task ausgeführt. Wenn gleichzeitig mehrere Interrupts auftreten, werden die Interrupt-Tasks nacheinander beginnend mit der niedrigsten Interrupt-Tasknummer ausgeführt.
3. In der folgenden Tabelle sind die Unterschiede zwischen zyklischen Tasks und zyklischen Zusatztasks aufgeführt.

Beschreibung	Zyklische Zusatztasks	Normale zyklische Tasks
Aktivierung beim Start	Einstellung ist nicht möglich.	Einstellung mit CX-Programmer
Verwendung der Befehle TKON/TKOF	Nicht möglich	Möglich
Task-Merker	Nicht unterstützt	Unterstützt (Die zyklischen Tasks 00 bis 31 korrespondieren mit den Task-Merkern TK00 bis TK31.)

Beschreibung	Zyklische Zusatztasks	Normale zyklische Tasks
Task-Erstausführungs-Merker (A20015) und Task-Start-Merker (A20014)	Nicht unterstützt	Unterstützt
Indexregister- (IR) und Datenregister- (DR) Werte	Beim Start der Task nicht definiert (wie bei normalen Interrupt-Tasks). Die Werte zu Beginn eines Zyklus sind nicht definiert. Stellen Sie stets Werte ein, bevor Sie diese verwenden. Die im vorherigen Zyklus gesetzten Werte können nicht gelesen werden.	Am Anfang des Vorgangs nicht definiert. Die im vorherigen Zyklus gesetzten Werte können gelesen werden.

- Die CJ1 CPU-Baugruppen unterstützen keine E/A-Interrupt-Tasks und externen Interrupt-Tasks.

4-1-5 Task-Ausführungsbedingungen und Einstellungen

In der folgenden Tabelle finden Sie Informationen zu den Task-Ausführungsbedingungen, zugehörigen Einstellungen und Zuständen.

Task		Nr.	Ausführungsbedingung	Zugehörige Einstellung
Zyklische Tasks		0 bis 31	Werden einmal pro Zyklus ausgeführt, wenn ausführungsbereit (Aktiv) (Einstellung auf automatischen Start oder Start mit Hilfe des Befehls TKON(820)) , wenn das Ausführungsrecht erteilt wurde.	Keine
Interrupt-Tasks	Ausschalt-Interrupt-Task	Interrupt-Task 1	Wird ausgeführt, wenn die Spannungsversorgung der CPU-Baugruppe ausgeschaltet wird.	Ausschalt-Interrupt im SPS-Setup aktiviert.
	Zeitgesteuerte Interrupt-Tasks 0 und 1	Interrupt-Tasks 2 und 3	Wird bei jedem Ablauf des im internen Zeitgeber der CPU-Baugruppe voreingestellten Intervalls ausgeführt.	- Die zeitgesteuerte Interrupt-Zeit (0 bis 9999) wird mit dem Befehl SET INTERRUPT MASK (Interrupt-Maske setzen)(MSKS) eingestellt. - Die Zeiteinheit (10 ms oder 1,0 ms) für die Einstellung des Intervalls wird im SPS-Setup spezifiziert.
	E/A-Interrupt-Tasks 00 bis 31	Interrupt-Tasks 100 bis 131	Werden ausgeführt, wenn ein Eingang einer Interrupt-Eingangsbaugruppe im CPU-Rack eingeschaltet wird.	Maskierungen für bestimmte Eingänge können durch den Befehl SET INTERRUPT MASK (Interrupt-Maske setzen) (MSKS) aufgehoben werden.
	Externe Interrupt-Tasks 0 bis 255	Interrupt-Tasks 0 bis 255	Werden ausgeführt, wenn sie vom Anwenderprogramm einer Spezial-E/A-Baugruppe oder CPU-Bus-Baugruppe im CPU-Rack oder einem internen Steckmodul (nur CS-Serie) angefordert werden.	Keine (stets aktiviert)
Zyklische Zusatztasks (nur CS1-H, CJ1-H, CJ1M und CS1D CPU-Baugruppen)		Interrupt-Tasks 0 bis 255	Werden einmal pro Zyklus ausgeführt, wenn ausführungsbereit (Aktiv) (Start mit Hilfe des Befehls TKON(820)) , wenn das Ausführungsrecht erteilt wurde.	Keine (stets aktiviert)

- Hinweis**
- Die Interrupt-Eingangsbaugruppe muss im CPU-Rack installiert sein. Bei CJ1-H-CPU-Baugruppen muss die Baugruppe als eine der fünf Baugruppen, die der CPU-Baugruppe am nächsten liegen, installiert sein (Steckplätze 0 bis 4). Bei CJ1M CPU-Baugruppen muss die Baugruppe als eine der drei Baugruppen, die der CPU-Baugruppe am nächsten liegen, installiert sein (Steckplätze 0 bis 2). In anderen Steckplätzen installierte E/A-Interrupt-Baugruppen können nicht für die Ausführungsanforderung von E/A-Interrupt-Tasks verwendet werden.

2. Die Spezial-E/A-Baugruppe oder CPU-Bus-Baugruppe muss im CPU-Rack installiert sein. Bei CJ1-H-CPU-Baugruppen muss die Baugruppe als eine der fünf Baugruppen, die der CPU-Baugruppe am nächsten liegen, installiert sein (Steckplätze 0 bis 4). Bei CJ1M CPU-Baugruppen muss die Baugruppe als eine der drei Baugruppen, die der CPU-Baugruppe am nächsten liegen, installiert sein (Steckplätze 0 bis 2). In anderen Steckplätzen installierte Baugruppen können nicht für das Erzeugen externer Interrupts verwendet werden.
3. Die Anzahl an zyklischen Tasks und Interrupt-Tasks ist begrenzt, wenn mit Hilfe einer Programmierkonsole programmiert wird.
 - Nur die zyklische Task 0 kann erstellt werden. Die zyklischen Tasks 1 bis 31 können nicht mit Hilfe einer Programmierkonsole erstellt werden. Sie können jedoch bearbeitet werden, wenn sie bereits mit CX-Programmer erstellt wurden.
 - Es können nur die Interrupt-Tasks 1, 2, 3 und 100 bis 131 erstellt werden. Die Interrupt-Tasks 0, 4 bis 99 und 132 bis 255 können nicht mit Hilfe einer Programmierkonsole erstellt (Ausnahme: 140 bis 143 bei CJ1M-CPU-Baugruppen), jedoch bearbeitet werden, wenn sie bereits mit CX-Programmer erstellt wurden.

4-1-6 Status der zyklischen Tasks

In diesem Abschnitt finden Sie Informationen über den Status zyklischer Tasks, einschließlich zyklischer Zusatztasks (nur von CS1-H, CJ1-H, CJ1M und CS1D CPU-Baugruppen unterstützt).

Zyklische Tasks verfügen stets über einen von vier Zuständen: Deaktiviert, READY, RUN (ausführbar) und Standby (WAIT).

Deaktivierter Status (INI)

Eine Task im deaktivierten Status wird nicht ausgeführt. Alle zyklischen Tasks sind in der PROGRAM-Betriebsart deaktiviert. Sämtliche Tasks, die in einen anderen Status gewechselt sind, können nur durch Umschalten in die PROGRAM-Betriebsart wieder in diesen Status zurückkehren.

READY-Status (ausführbereit)

Es kann ein Task-Attribut eingestellt werden, um zu steuern, wann die Task in den READY-Status (ausführbereit) wechselt. Das Attribut kann entweder auf die Aktivierung der Task mit Hilfe des Befehls TASK ON oder durch den Start der RUN-Betriebsart eingestellt werden.

Befehlsaktivierte Tasks

Der Befehl TASK ON (TKON(820)) wird verwendet, um eine Task aus dem Status „Deaktiviert“ oder „Standby“ in den Status READY (ausführbereit) umzuschalten.

Betriebsartaktivierte Tasks

Der Status einer betriebsartaktivierten zyklischen Task wechselt von „Deaktiviert“ zu „READY“, wenn die Betriebsart von PROGRAM zu RUN oder MONITOR umgeschaltet wird. Dies gilt nur für normale zyklische Tasks.

Hinweis

Mit Hilfe der Programmiersoftware kann für die Tasks 0 bis 31 eingestellt werden, dass sie beim Start des Betriebs automatisch in den READY-Status wechseln. Diese Einstellung ist jedoch für zyklische Zusatztasks nicht möglich.

RUN-Status

Eine ausführbereite (READY) zyklische Task wechselt in den RUN (ausführbar) -Status und wird ausgeführt, sobald sie das Recht zur Ausführung erhält.

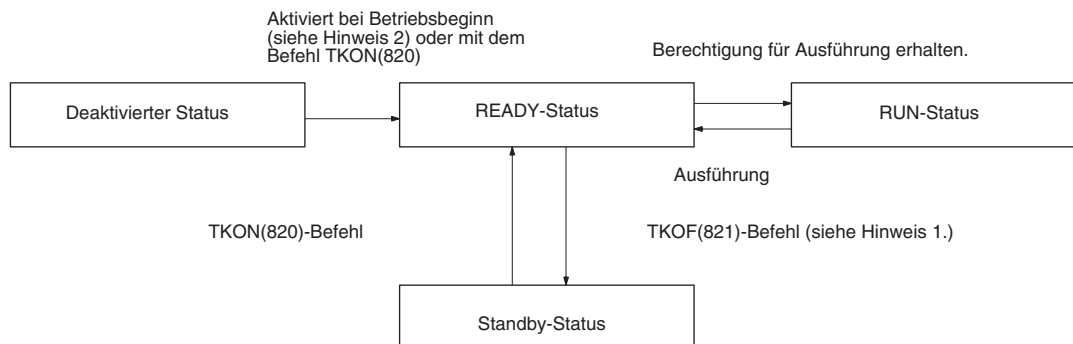
Standby-Status

Der Befehl TASK OFF (Programm-Task AUS) (TKOF(821)) kann verwendet werden, um den Status einer zyklischen Task von „Ausführungsbereit“ zu „Standby“ zu ändern.

Hinweis Mit CX-Programmer Version 4.0 oder höher können die Taskprogramme bei der SPS der CS/CJ-Serie online überwacht werden, um zu überprüfen, ob diese ausgeführt werden oder gestoppt sind. In CX-Programmer werden die folgenden Statusangaben verwendet:

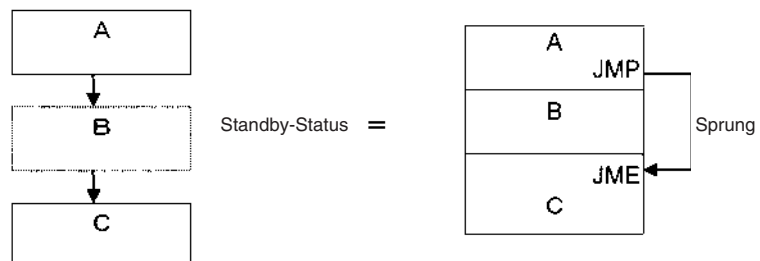
- Aktiv: Die Task hat den Status READY (ausführungsbereit) oder RUN (ausführbar). (Ohne Unterscheidung zwischen diesen beiden Zuständen.)
- Angehalten: Die Task hat den Status INI oder WAIT. (Ohne Unterscheidung zwischen diesen beiden Zuständen.)

4-1-7 Statuswechsel

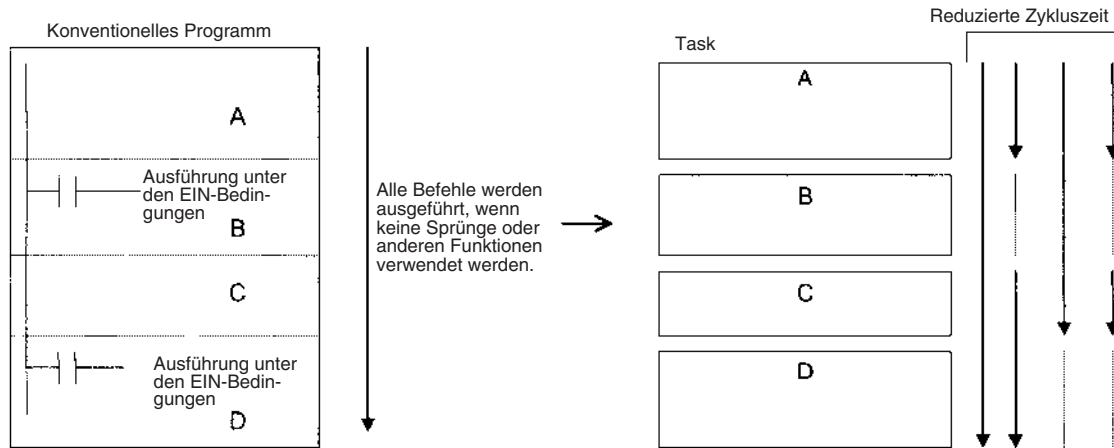


- Hinweis**
1. Eine Task im ausführbaren Status RUN wird mit Hilfe des Befehls TKOF(821) in den Standby-Status versetzt, selbst wenn der Befehl TKOF(821) innerhalb der Task ausgeführt wird.
 2. Die Aktivierung beim Betriebsstart ist nur für normale zyklische Tasks möglich, nicht jedoch für zyklische Zusatztasks.

Der Standby-Status funktioniert auf gleiche Weise wie ein Sprung (JMP-JME). Der Ausgangsstatus der Standby-Task wird beibehalten.



Befehle werden im Standby-Status nicht ausgeführt, sodass die Befehlsausführungszeit nicht verlängert wird. Programmabschnitte, die nicht ständig ausgeführt werden müssen, können in Tasks erstellt werden, denen der Standby-Status zugewiesen wird, um die Zykluszeit zu verringern.



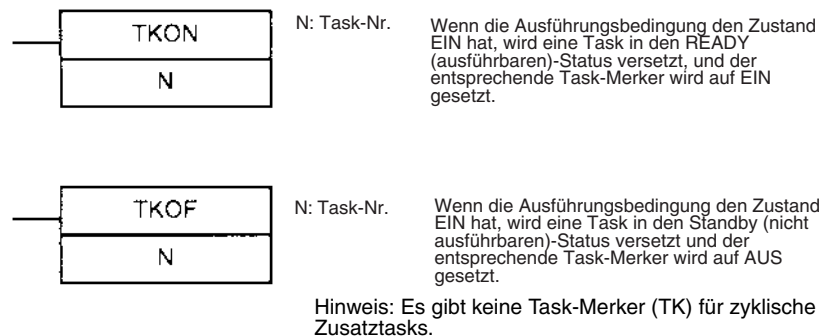
Hinweis Standby-Status bedeutet lediglich, dass eine Task bei der Taskausführung überspringen wird. Der Wechsel in den Standby-Status beendet nicht das Programm.

4-2 Verwenden von Tasks

4-2-1 TASK ON und TASK OFF

Mit Hilfe der Befehle TASK ON (Programm-Task EIN) (TKON(820)) und TASK OFF (Programm-Task AUS) (TKOF(821)) wird der Status zyklischer Tasks (einschließlich zyklischer Zusatztasks) aus dem Programm heraus zwischen ausführbereit (READY) und Standby umgeschaltet.

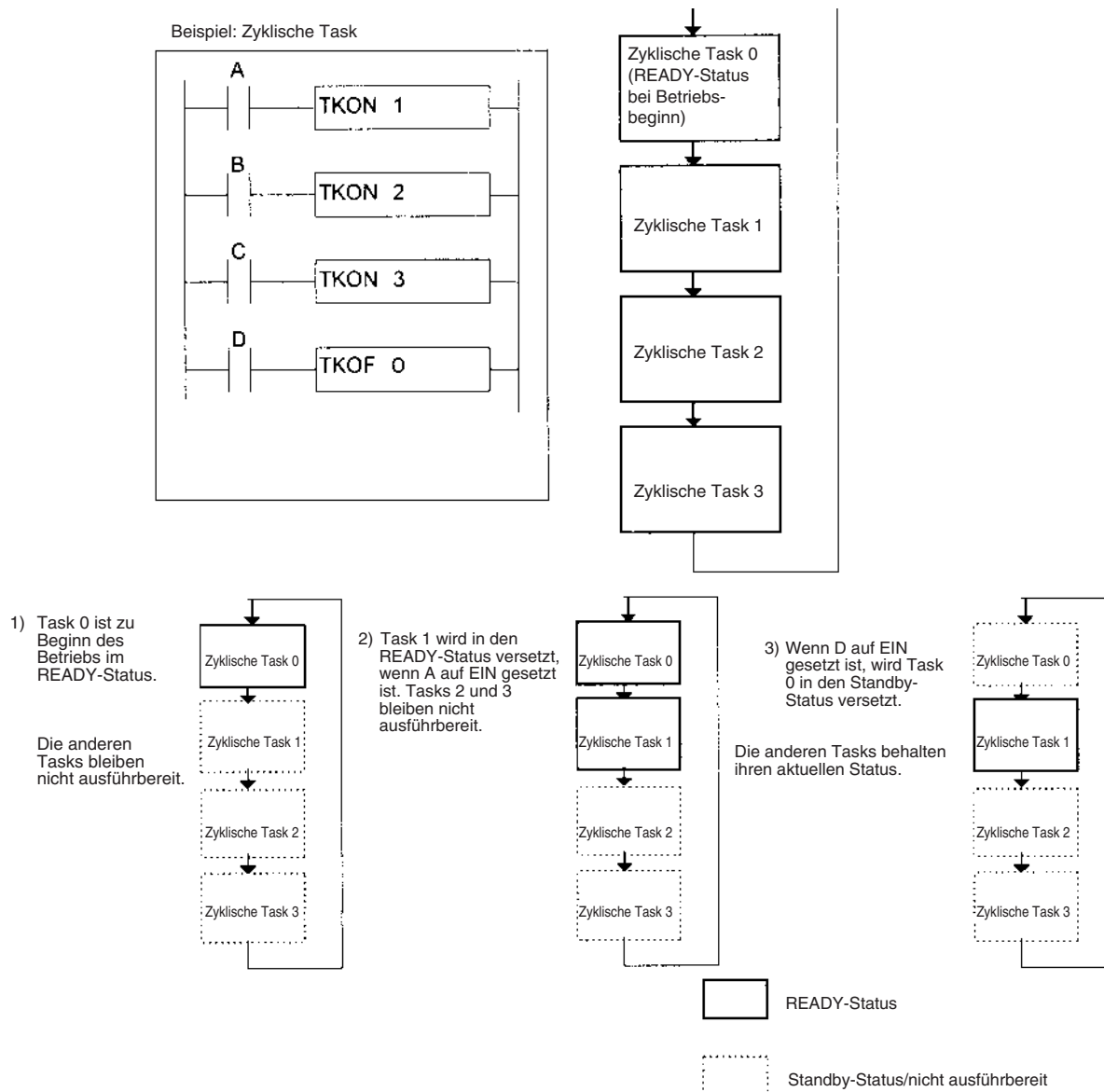
Hinweis Zyklische Zusatztasks werden nur von den CPU-Baugruppen CS1-H, CJ1-H, CJ1M und CS1D unterstützt.



Mit Hilfe der Befehle TASK ON (Programm-Task EIN) und TASK OFF (Programm-Task AUS) kann der Status zyklischer Tasks zu jedem beliebigen Zeitpunkt zwischen ausführbereit (READY) und Standby geändert werden. Eine zyklische Task im ausführbereiten READY-Status behält diesen Status in darauffolgenden Zyklen ebenso bei, wie eine zyklische Task im Standby-Status auch ihren Status beibehält.

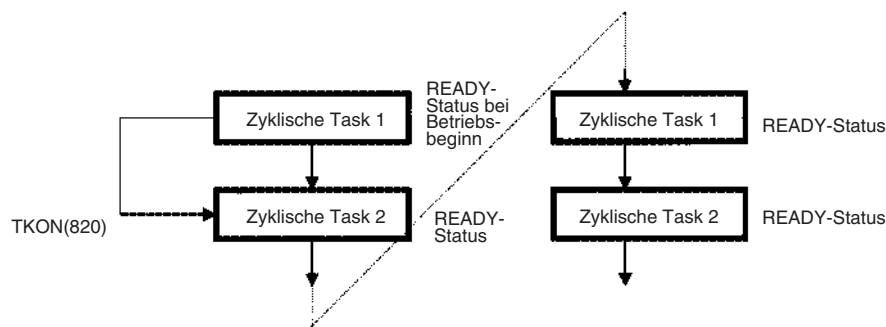
Die Befehle TASK ON und TASK OFF können nur auf zyklische Tasks und zyklische Zusatztasks angewendet, nicht jedoch auf Interrupt-Tasks.

Hinweis Pro Zyklus muss mindestens eine zyklische Task im READY-Status (ausführbereit) sein. Wenn keine ausführbare Task vorhanden ist, wird der Task-Fehlermerker (A29512) auf EIN gesetzt, und die CPU-Baugruppe wird angehalten.

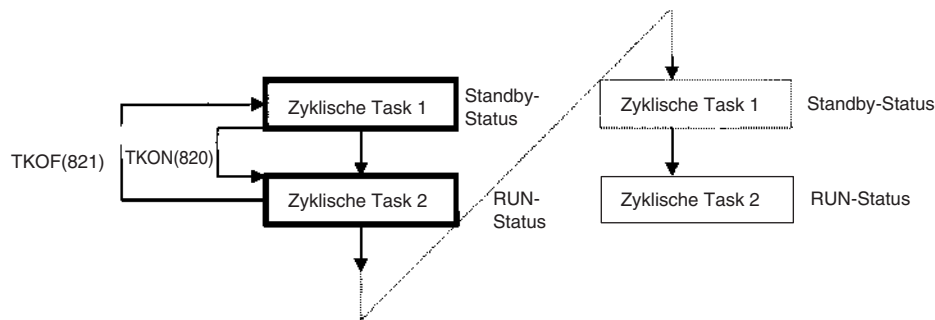


Tasks und der Ausführungszyklus

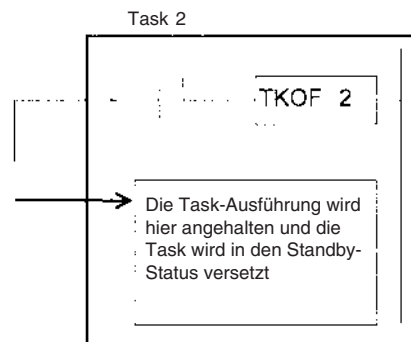
Eine ausführbbereite (READY) zyklische Task (einschließlich zyklischer Zusatztasks) behält diesen Status in den darauffolgenden Zyklen bei.



Das Gleiche gilt für zyklische Tasks im Standby-Status. Die Task muss mit Hilfe des Befehls TKON(820) aktiviert werden, wodurch ihr Status von Standby zu ausführbbereit (READY) geändert wird.



Wenn der Befehl TKOF(821) für Task ausgeführt wird, in der er sich befindet, wird die Ausführung der Task mit der Ausführung des Befehls angehalten und die Task wechselt in den Standby-Status.



Nummern zyklischer Tasks und der Ausführungszyklus (einschließlich zyklischer Zusatztasks)

Wenn Task m Task n aktiviert und $m > n$ ist, hat Task n im nächsten Zyklus den ausführbereiten (READY)-Status.

Beispiel: Wenn Task 5 Task 2 aktiviert, hat Task 2 im nächsten Zyklus den READY-Status.

Wenn Task m Task n aktiviert und $m < n$ ist, wechselt Task n im gleichen Zyklus in den READY-Status.

Beispiel: Wenn Task 2 Task 5 aktiviert, wechselt Task 5 im gleichen Zyklus in den READY-Status.

Wenn Task m Task n in den Standby-Status setzt und $m > n$ ist, hat Task n im nächsten Zyklus den Standby-Status.

Beispiel: Wenn Task 5 Task 2 in den Standby-Status setzt, hat Task 2 im nächsten Zyklus den Standby-Status.

Wenn Task m Task n in den Standby-Status setzt und $m < n$ ist, wechselt Task n im gleichen Zyklus in den Standby-Status.

Beispiel: Wenn Task 2 Task 5 in den Standby-Status setzt, wechselt Task 5 im gleichen Zyklus in den Standby-Status.

Beziehung zwischen Tasks und E/A-Speicher

Es gibt zwei verschiedene Verwendungsmöglichkeiten für Index-Register (IR) und Daten-Register (DR): 1) Unabhängig durch einzelne Tasks oder 2) gemeinsam durch all Tasks (nur von CS1-H, CJ1-H, CJ1M und CS1D CPU-Baugruppen unterstützt).

Bei unabhängigen Registern unterscheidet sich z. B. das von der zyklischen Task 1 verwendete IR0 vom IR0 der zyklischen Task 2. Bei gemeinsam genutzten Registern verwenden z. B. die zyklischen Tasks 1 und 2 das gleiche IR0.

Die Einstellung, ob Register unabhängig oder gemeinsam verwendet werden, wird in CX-Programmer vorgenommen.

- Andere Worte und Bits im E/A-Speicher werden von allen Tasks gemeinsam verwendet. CIO 001000 ist beispielsweise das gleiche Bit für die zyklischen Tasks 1 und 2. Programmieren Sie daher sehr sorgfältig, wenn andere E/A-Speicherbereiche als die IR- und DR-Bereiche verwendet werden, da die für eine Task geänderten Werte auch von anderen Tasks verwendet werden.

E/A-Speicher	Verhältnis zu Tasks
CIO, Zusatz-, Datenspeicher und alle weiteren Speicherbereiche mit Ausnahme der IR- und DR-Bereiche. (siehe Hinweis 1)	Gemeinsam mit anderen Tasks genutzt.
Index-Register (IR) und Daten-Register (DR) (siehe Hinweis 2)	Von einzelnen Tasks separat genutzt.

Hinweis

1. Die aktuelle EM-Bank wird von den Tasks ebenfalls gemeinsam verwendet. Wenn die aktuelle Nummer der EM-Bank z.B. durch die zyklische Task 1 geändert wird, gilt die geänderte EM-Bank-Nummer daher auch für die zyklische Task 2.
2. Die IR- und DR-Werte sind beim Starten von Interrupt-Tasks (einschließlich zyklischer Zusatztasks) nicht gesetzt. Wenn IR und DR in einer Interrupt-Task verwendet werden, müssen diese Werte mit Hilfe der Befehle MOVR/MOVRW (MOVE TO REGISTER (Zu Register übertragen) und MOVE TIMER/COUNTER PV TO REGISTER (Zeitgeber/Zähler-Istwert-adresse zu Register übertragen) innerhalb der Interrupt-Task gesetzt werden. Nach der Ausführung einer Interrupt-Task werden IR und DR automatisch auf die ursprünglichen Werte (Werte vor Ausführung der Interrupt-Task) zurückgesetzt.

**Beziehung zwischen
Tasks und
Zeitgeberbetrieb**

Die Istwerte für TIM, TIMX, TIMH, TIMHX, TMHH, TMHHX, TIMW, TIMWX, TMHW und TMHWX mit Zeitgebernummern von 0000 bis 2047 werden selbst dann aktualisiert, wenn die Task wechselt oder die Task, die den Zeitgeber enthält, in den Standby-Status oder zurück in den ausführungsbereiten (READY)-Status gesetzt wird.

Wenn die Task, die den TIM-Befehl enthält, ihren Status zu Standby und anschließend wieder zu READY wechselt, wird der Ablaufmerker auf EIN gesetzt, wenn der Befehl TIM bei einem Istwert von 0 ausgeführt wird. (Ablaufmerker für Zeitgeber werden nur während der Ausführung des Befehls aktualisiert.) Wenn der Befehl TIM ausgeführt wird, solange der Istwert noch nicht gleich 0 ist, wird der Istwert weiterhin wie mit der Task im READY-Status aktualisiert.

- Die Istwerte für Zeitgeber mit den Zeitgebernummern 2048 bis 4098 werden beibehalten, wenn sich die Task im Standby-Status befindet.

**Beziehung zwischen
Tasks und
Bedingungsmerkern**

Sämtliche Bedingungsmerker werden vor der Ausführung jeder Task gelöscht. Daher kann der Bedingungsmerker am Ende von Task 1 in Task 2 nicht gelesen werden. Bei den CS1-H, CJ1-H, CJ1M und CS1D CPU-Baugruppen können jedoch die Befehle CCS(282) und CCL(283) verwendet werden, um den Bedingungsmerkerzustand aus anderen Teilen des Programms, z. B. aus einer anderen Task zu lesen.

Hinweis

Wenn der Bedingungsmerkerzustand mit Hilfe einer Programmierkonsole überwacht wird, zeigt diese den Merkerzustand am Ende des Zyklus an, d. h. ihren Zustand am Ende der letzten Task im Zyklus.

4-2-2 Einschränkungen bei Task-Befehlen

In der gleichen Task erforderliche Befehle

Folgende zusammengehörige Befehle müssen in der gleichen Task enthalten sein. Wenn der Versuch unternommen wird, zusammengehörige Befehle zwischen zwei Tasks aufzuteilen, wird der Fehlermerker auf EIN gesetzt und die Befehle werden nicht ausgeführt.

AWL	Befehl
JMP/JME	JUMP/JUMP END (Sprung/Sprungende)
CJP/JME	CONDITIONAL JUMP/JUMP END (Bedingter Sprung/Sprungende)
CJPN/JME	CONDITIONAL JUMP NOT/CONDITIONAL JUMP END (Bedingter negierter Sprung/Sprungende)

AWL	Befehl
JMP0/JME0	MULTIPLE JUMP/ MULTIPLE JUMP END (Mehrfachsprung/ Mehrfachsprung-Ende)
FOR/NEXT	FOR/NEXT
IL/ILC	INTERLOCK/INTERLOCK CLEAR (Verriegelung/Verriegelung löschen)
SBS/SBN/RET	SUBROUTINE CALL/SUBROUTINE ENTRY/SUBROUTINE RETURN (Unterprogramm aufrufen/ Unterprogramm-Anfang/ Unterprogramm verlassen)
MCRO/SBN/ RET	MACRO/SUBROUTINE ENTRY/SUBROUTINE RETURN (Makro/ Unterprogramm-Anfang/ Unterprogramm verlassen)
BPRG/BEND	BLOCK PROGRAM BEGIN/BLOCK PROGRAM END (Blockprogramm Anfang/Blockprogramm Ende)
STEP /SNXT	Schritt/ nächster Schritt

In Interrupt-Tasks nicht zulässige Befehle

Die folgenden Befehle dürfen nicht in Interrupt-Tasks verwendet werden. Wenn der Versuch unternommen wird, einen dieser Befehle in einer Interrupt-Task auszuführen, wird der Fehlermerker auf EIN gesetzt und der Befehl wird nicht ausgeführt. Die folgenden Befehle können verwendet werden, wenn eine Interrupt-Task als Zusatztask verwendet wird.

AWL	Befehl
TKON(820)	TASK ON (Programm-Task EIN)
TKOF(821)	TASK OFF (Programm-Task AUS)
STEP	STEP DEFINE (Schritt)
SNXT	STEP NEXT (nächster Schritt)
STUP	CHANGE SERIAL PORT SETUP (Änderung der seriellen Schnittstellen-Konfiguration)
DI	DISABLE INTERRUPT (Interrupt deaktivieren)
EI	ENABLE INTERRUPT (Interrupt aktivieren)

Die Ausführung folgender Befehle in Interrupt-Tasks kann zu nicht vorhersehbaren Ergebnissen führen: TIMER (Zeitgeber): TIM und TIMX((550), HIGH-SPEED TIMER (Schneller Zeitgeber): TIMH(015) und TIMHX(551), ONE-MS TIMER (1-ms-Zeitgeber): TMHH(540) und TMHHX(552), ACCUMULATIVE TIMER (Akkumulierender Zeitgeber): TTIM(087) und TTIMX(555), MULTIPLE OUTPUT TIMER (Multi-Ausgabe-Zeitgeber): MTIM(543) und MTIMX(554), LONG TIMER (Langer Zeitgeber): TIML(542) und TIMLX(553), TIMER WAIT (Zeitgeber warten): TIMW(813) und TIMWX(816), HIGH-SPEED TIMER WAIT (Schneller Zeitgeber warten): TMHW(815) und TMHWX(817), PID CONTROL (PID-Regelung): PID(190), FAILURE POINT DETECTION (Fehlerpunkt-Erfassung): FPD(269) und CHANGE SERIAL PORT SETUP (Änderung der seriellen Schnittstellen-Konfiguration): STUP(237).

Die folgenden Befehle können nicht in der Ausschalt-Interrupt-Task verwendet werden (sollten Sie in der Ausschalt-Interrupt-Task verwendet werden, werden sie nicht ausgeführt und der Fehlermerker wird **nicht** auf EIN gesetzt):

READ DATA FILE (Datei lesen): FREAD(700), WRITE DATA FILE (Datei schreiben): FWRT(701), NETWORK SEND (Netzwerk Senden): SEND(090), NETWORK RECEIVE (Netzwerk Empfangen): RECV(098), DELIVER COMMAND (FINS-Befehl zustellen): CMND(490), PROTOCOL MACRO (Protokoll-Makro): PMCR(260).

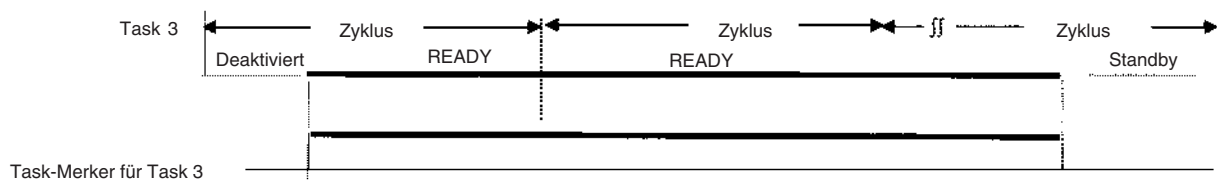
4-2-3 Zu Tasks gehörige Merker

Zu zyklischen Tasks gehörige Merker

Die folgenden Merker funktionieren nur für normale zyklische Tasks. Sie können nicht für zyklische Zusatztasks verwendet werden.

Task-Merker (TK00 bis TK31)

Ein Task-Merker wird auf EIN gesetzt, wenn eine zyklische Task ausführbar (READY) ist, und auf AUS, wenn eine Task deaktiviert (INI) oder im Standby-Status (WAIT) ist. Die Tasks 00 bis 31 entsprechen den Task-Merkern TK00 bis TK31.

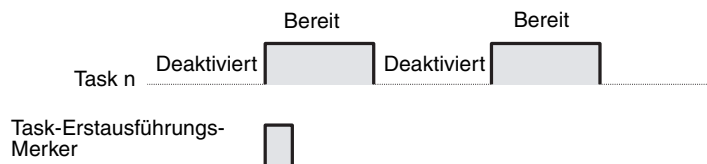


Hinweis

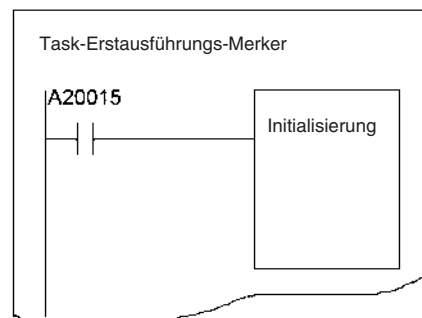
Task-Merker werden nur für zyklische Tasks, nicht jedoch für Interrupt-Tasks verwendet. Bei einer Interrupt-Task wird A44115 auf EIN gesetzt, wenn die Interrupt-Task nach dem Start des Betriebs ausgeführt wird. Die Nummer der Interrupt-Task, die die längste Zeit zur Ausführung in Anspruch nimmt, wird als 2-stellige Hexadezimalzahl in A44100 bis A44107 gespeichert.

Task-Erstaussführungs-Merker (A20015)

Der Task-Erstaussführungs-Merker wird auf EIN gesetzt, wenn der Status zyklischer Tasks von deaktiviert (INI) zu ausführbereit (READY) wechselt, die Tasks das Ausführungsrecht erhalten und erstmals ausgeführt werden. Der Merker wird auf AUS gesetzt, wenn die erste Ausführung der Tasks abgeschlossen ist.



Der Task-Erstaussführungs-Merker gibt an, ob die zyklischen Tasks erstmals ausgeführt werden oder nicht. Dieser Merker kann daher für die Ausführung eines Initialisierungsvorgangs innerhalb der Tasks verwendet werden.

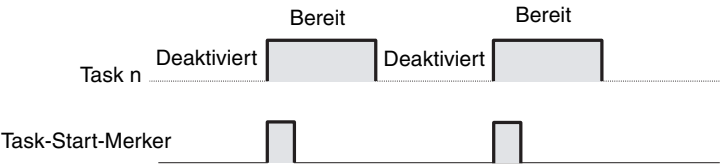


Hinweis

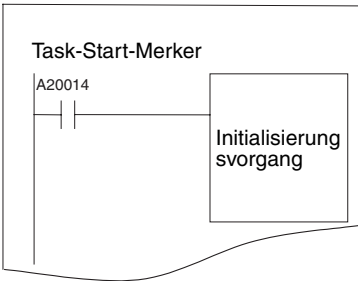
Obwohl eine zyklische Task im Standby-Status durch den Befehl TKON(820) wieder ausführbar (READY) wird, gilt dies nicht als Erstaussführung, so dass der Task-Erstaussführungs-Merker (20015) nicht auf EIN gesetzt wird. Der Task-Erstaussführungs-Merker (20015) wird ebenfalls nicht auf EIN gesetzt, wenn der Status einer zyklische Task von deaktiviert zu ausführbar (RUN) wechselt oder durch eine andere Task mit Hilfe des Befehls TKOF(821) in den Standby-Status gesetzt wird, bevor das Recht zur Ausführung tatsächlich erteilt wurde.

Task-Start-Merker (A20014, nur CS1-H, CJ1-H, CJ1M und CS1D CPU-Baugruppen)

Mit Hilfe von Task-Start-Merkern kann bei jedem Start eines Taskzyklus ein Initialisierungsvorgang durchgeführt werden. Der Task-Start-Merker wird auf AUS gesetzt, wenn der Status einer zyklischen Task von deaktiviert (INI) oder Standby (WAIT) in ausführbereit (READY) wechselt (wohingegen der Task-Erstaussführungs-Merker nur dann auf EIN gesetzt wird, wenn von deaktiviert (INI) zu ausführbereit (READY) gewechselt wird).



Der Task-Start-Merker kann zur Durchführung eines Initialisierungsvorgangs verwendet werden, wenn eine Task vom Standby-Status in den Status ausführbar (RUN) wechselt, d. h. wenn eine Task im Standby-Status mit Hilfe des Befehls TKON(820) aktiviert wird.



Merker, die alle Tasks betreffen

- Task-Fehlermerker (A29512)**
- Der Task-Fehlermerker wird auf EIN gesetzt, wenn einer der folgenden Taskfehler auftritt.
- Keine zyklischen Tasks (einschließlich zyklischer Zusatztasks) sind während eines Zyklus ausführbereit (READY).
 - Das einer zyklischen Task (einschließlich zyklischer Zusatztasks) zugeordnete Programm ist nicht vorhanden. (Dies tritt nicht auf, wenn CX-Programmer oder eine Programmierkonsole verwendet werden.)
 - Einer aktivierten Interrupt-Task ist kein Programm zugeordnet.

Tasknummer bei Programmabbruch (A294)

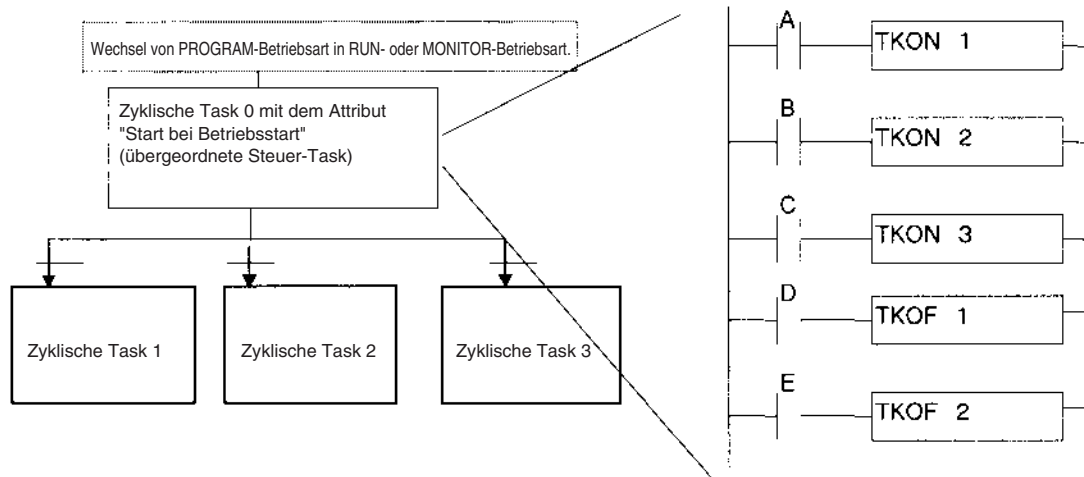
Der Tasktyp und die aktuelle Tasknummer werden bei einem Ausführungsabbruch aufgrund eines Programmfehlers wie folgt gespeichert:

Typ	A294
Zyklische Task	0000 bis 001F Hex (entspricht den Tasknummern 0 bis 31)
Interrupt-Task	8000 bis 80FF Hex (entspricht den Interrupt-Task-Nummern 0 bis 255)

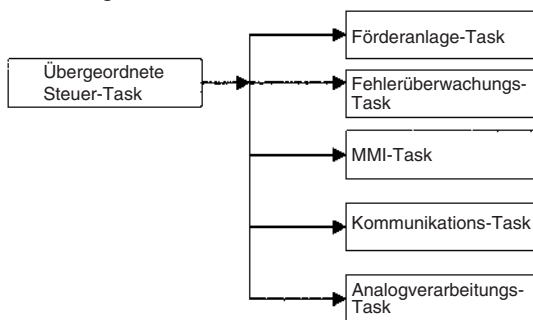
Mit Hilfe dieser Informationen lässt sich einfacher ermitteln, wo der schwerwiegende Fehler aufgetreten ist. Die Informationen werden nach der Fehlerbehebung gelöscht. Die Programmadresse, an der die Taskausführung abgebrochen wurde, wird in A298 (rechte Bits der Programmadresse) und in A299 (linke Bits der Programmadresse) gespeichert.

Beispiele für Tasks

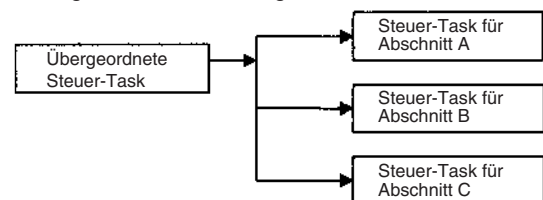
Eine übergeordnete Steuer-Task, die darauf eingestellt ist, beim Betriebsstart in den Status ausführbereit (READY) zu wechseln, wird im Allgemeinen verwendet, um den Status READY/Standby aller anderen zyklischen Tasks (einschließlich zyklischer Zusatztasks) zu steuern. Natürlich kann jede zyklische Task den READY/Standby-Status beliebiger anderer zyklischer Tasks entsprechend den Anwendungsanforderungen steuern.



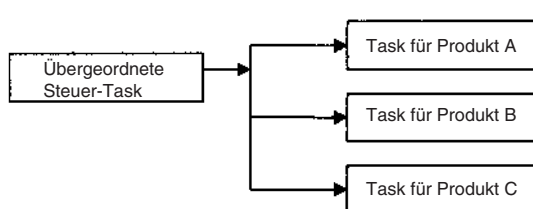
Tasks getrennt nach Funktion



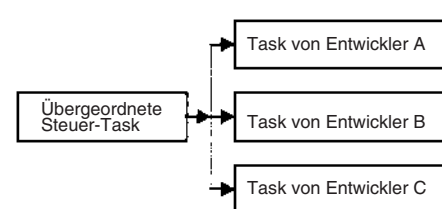
Tasks getrennt nach Steuerungsabschnitt



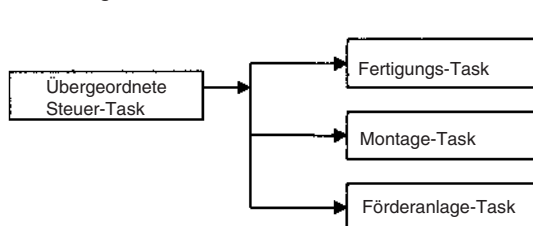
Tasks getrennt nach Produkt



Tasks getrennt nach Entwickler



Tasks getrennt nach Prozess

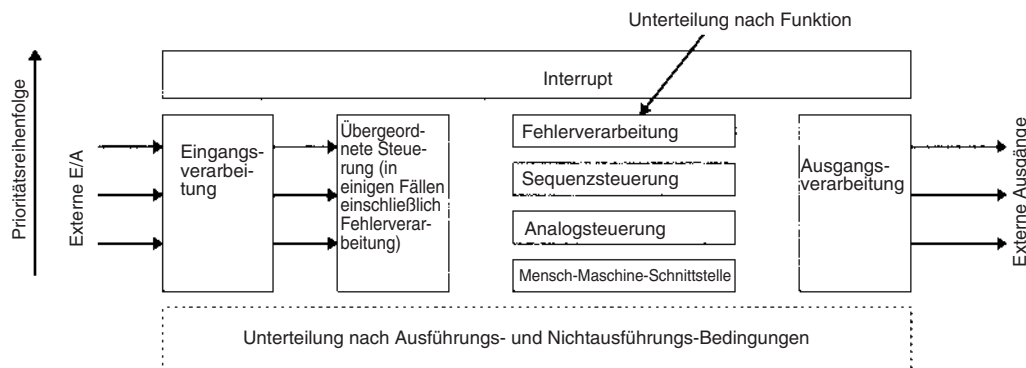


Kombinationen der genannten Einteilungen sind ebenfalls möglich, z. B. Einteilung nach Funktion und Prozess.

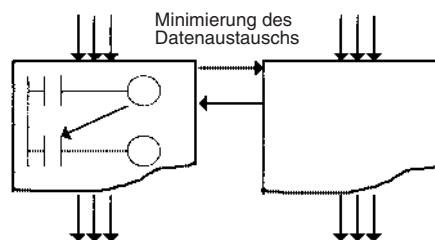
4-2-4 Erstellen von Tasks

Wir empfehlen, bei der Erstellung von Tasks nach folgenden Richtlinien vorzugehen.

- 1,2,3...**
1. Verwenden Sie folgende Standards, wenn Sie sich mit dem Separieren von Tasks befassen.
 - a) Fassen Sie spezifische Bedingungen für die Ausführung oder Nicht-Ausführung zusammen.
 - b) Fassen Sie das Vorhandensein oder Fehlen von externen E/A zusammen.
 - c) Fassen Sie die Funktionen zusammen.
Halten Sie den Datenaustausch zwischen Tasks für die Sequenzsteuerung, analoge Steuerung, Anwender-Maschinen-Schnittstellen, Fehlerbearbeitung und andere Prozesse möglichst gering, um einen hohen Autonomiegrad zu gewährleisten.
 - d) Fassen Sie die Ausführung in Reihenfolge der Priorität zusammen.
Teilen Sie die Verarbeitung in zyklische und Interrupt-Tasks auf.



2. Unterteilen und entwickeln Sie Programme so, dass die Autonomie gewährleistet ist und möglichst wenige Daten zwischen Tasks (Programmen) ausgetauscht werden.



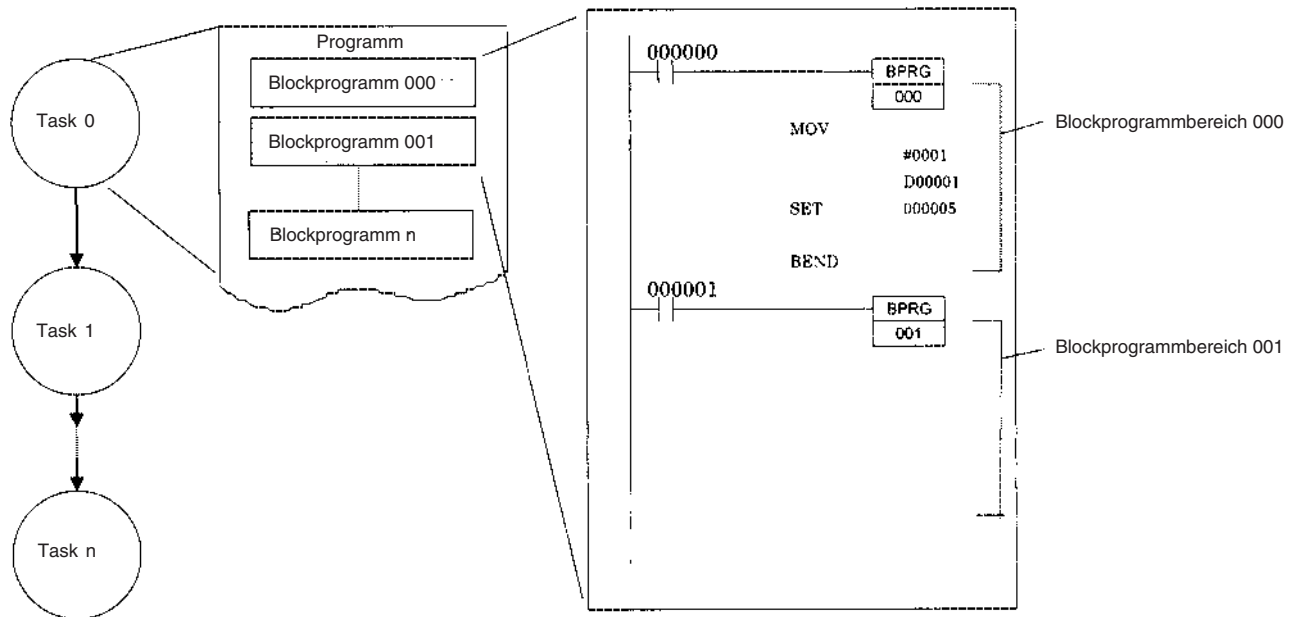
3. Verwenden Sie eine übergeordnete Steuer-Task, um den READY/Standby-Status anderer Tasks zu steuern.
4. Ordnen Sie Tasks mit den höchsten Prioritäten die niedrigsten Nummern zu. Beispiel: Ordnen Sie der Steuer-Task eine niedrigere Nummer als den Verarbeitungs-Tasks zu.
5. Ordnen Sie Interrupt-Tasks mit höherer Priorität niedrigere Nummern zu.
6. Eine ausführfbereite (READY-Status) Task wird in aufeinander folgenden Zyklen ausgeführt, solange sie nicht durch sich selbst oder eine andere Task in den Standby-Status gesetzt wird. Fügen Sie für die anderen Tasks den Befehl TKOF(821) (TASK OFF) ein, wenn die Verarbeitung zwischen Tasks verzweigt wird.

7. Verwenden Sie in der Ausführungsbedingung für Befehle zur Initialisierung von Tasks den Task-Erstausführungs-Merker (A20015) oder den Task-Start-Merker (A20014). Der Task-Erstausführungs-Merker ist bei der Erstauführung jeder Task auf EIN gesetzt. Für den Task-Start-Merker gilt dies immer dann, wenn eine Task in den READY-Status gesetzt wird.
8. Ordnen Sie E/A-Speicher den gemeinsam von Tasks verwendeten und dem von einzelnen Tasks verwendeten Speicher zu, und gruppieren Sie anschließend den von individuellen Tasks verwendeten E/A-Speicher nach den einzelnen Tasks.

Beziehung zwischen Tasks und Blockprogrammen

In den Tasks können bis zu 128 Blockprogramme erstellt werden. Dies ist die für sämtliche Tasks mögliche Gesamtzahl. Die Ausführung jedes vollständigen Blockprogramms wird vom Kontaktplan gesteuert, die Befehle im Blockprogramm werden jedoch mittels AWL geschrieben. Ein Blockprogramm besteht also aus einer Kombination von Kontaktplanbefehlen und AWL-Code.

Die Verwendung eines Blockprogramms erleichtert das Schreiben eines logischen Ablaufs, wie z. B. von Bedingungsverzweigungen und Verarbeitungsschritten, was mit einem Kontaktplan schwer zu schreiben ist. Blockprogramme sind unten in der Programmhierarchie angesiedelt. Die nächst größeren Programmeinheiten, die Tasks, können in kleinere Programme, die Blockprogramme, aufgeteilt werden, die mit der gleichen Ausführungsbedingung (EIN-Bedingung) arbeiten.



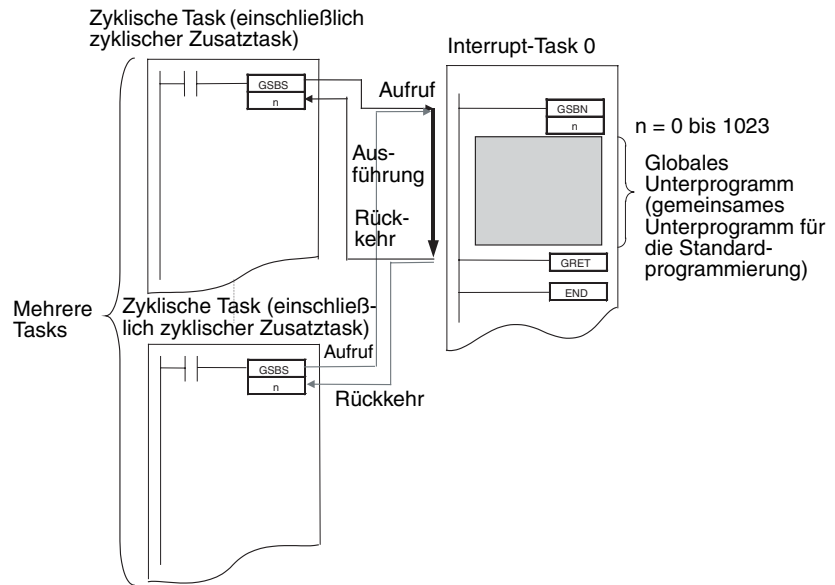
4-2-5 Globale Unterprogramme

Globale Unterprogramme können von mehreren Tasks aufgerufen werden. Sie werden nur von den CS1-H, CJ1-H, CJ1M und CS1D CPU-Baugruppen unterstützt.

Bei den CS1 und CJ1 CPU-Baugruppen kann ein Unterprogramm in einer Task nicht von einer anderen Task aufgerufen werden. Bei den CS1-H, CJ1-H, CJ1M und CS1D CPU-Baugruppen können globale Unterprogramme jedoch in Interrupt-Task 0 erstellt und von zyklischen Tasks (einschließlich zyklischer Zusatztasks) aufgerufen werden.

Zum Aufruf eines globalen Unterprogramms wird der Befehl GSBS verwendet. Die Nummer des Unterprogramms muss im Bereich zwischen 0 und 1023 liegen. Das globale Unterprogramm wird am Ende von Interrupt Task 0 (unmittelbar vor dem END(001)-Befehl) zwischen den Befehlen GSBN und GRET definiert.

Globale Unterprogramme können für das Erstellen einer Bibliothek von jederzeit aufrufbaren Standardprogrammabschnitten verwendet werden.



4-3 Interrupt-Tasks

4-3-1 Arten von Interrupt-Tasks

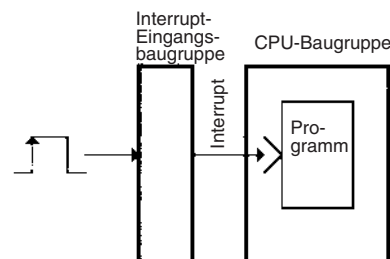
Interrupt-Tasks können zu einem beliebigen Zeitpunkt innerhalb des Zyklus ausgeführt werden, wenn eine der folgenden Bedingungen erfüllt ist.

Zu Aktivierung von Interrupt-Tasks können die integrierten Interrupt- und Schnelle-Zähler-Eingänge einer CJ1M CPU-Baugruppe verwendet werden. Detaillierte Informationen hierzu finden Sie im *CJ-Serie Integrierte E/A Bedienerhandbuch*.

Hinweis Die CS1D CPU-Baugruppen für Duplex-CPU-Systeme unterstützen keine Interrupts. Für CS1D CPU-Baugruppen können Interrupt-Tasks nur als zyklische Zusatztasks verwendet werden.

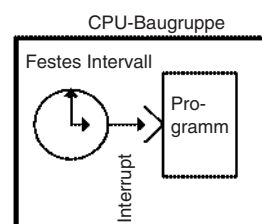
E/A-Interrupts

Die E/A-Interrupt-Task wird ausgeführt, wenn der entsprechende Eingang der Interrupt-Eingangsbaugruppe auf EIN gesetzt wird.



Zeitgesteuerte Interrupts

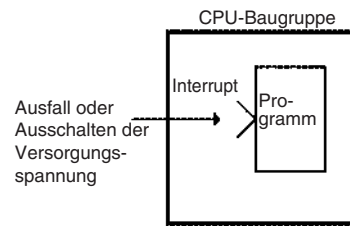
Eine zeitgesteuerte Interrupt-Task wird in festgelegten Intervallen ausgeführt.



Ausschalt-Interrupt

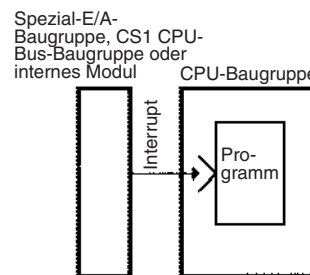
Die Ausschalt-Interrupt-Task wird ausgeführt, wenn die Spannungsversorgung ausgeschaltet wird.

Hinweis Die Ausführungszeit einer Ausschalt-Task muss unter 10 ms liegen – (Ausschalterkennungs-Verzögerungszeit).



Externe Interrupts

Eine externe Interrupt-Task wird ausgeführt, wenn von einer Spezial-E/A-Baugruppe, CPU-Bus-Baugruppe oder einem internen Steckmodul (nur CS-Serie) ein Interrupt angefordert wird. Die Spezial-E/A- oder CJ-Bus-Baugruppe muss jedoch im CPU-Rack installiert sein, um die Ausführung einer externen Interrupt-Task anfordern zu können.



Liste der Interrupt-Tasks

Typ	Task-Nr.	Ausführungsbedingung	Einstellungsverfahren	Anzahl der Interrupts	Anwendungsbeispiele
E/A-Interrupts 00 bis 31	100 bis 131	EIN-Eingang der Interrupt-Eingangsbaugruppe im CPU-Rack (Siehe Hinweis 1).	Verwenden Sie den Befehl MSKS (SET INTERRUPT MASK-(Interrupt-Maske setzen)), um Eingänge der Interrupt-Eingangsbaugruppe entsprechend der Position im CPU-Rack maskieren/demaskieren.	32	Verbesserte Antwortzeit für bestimmte Eingänge
Zeitgesteuerte Interrupts 0 und 1	2 und 3	Zeitgesteuert (feste Intervalle)	Verwenden Sie den Befehl MSKS (Interrupt-Maske setzen), um das Interrupt-Intervall festzulegen. Siehe Zeiteinheit des zeitgesteuerten Interrupts im SPS-Setup.	2	Überwachen des Betriebsstatus in festgelegten Intervallen.
Ausschalt-Interrupt	1	Wenn die Spannungsversorgung ausgeschaltet wird (nach der standardmäßigen Ausschalterkennungs-Verzögerungszeit und der Ausschalt-Erfassungsverzögerungszeit)	Siehe Ausschalt-Interrupt-Task und Ausschalterkennungs-Verzögerungszeit im SPS-Setup.	1	Ausführen der Notfallverarbeitung bei einer Unterbrechung der Spannungsversorgung.
Externe Interrupts 0 bis 255	0 bis 255	Wenn von einer Spezial-E/A- oder CPU-Bus-Baugruppe des CPU-Racks oder einem internen Steckmodul (nur CS-Serie) angefordert. (Siehe Hinweis 2).	Keine (immer gültig)	256	Ausführung von Verarbeitungsvorgängen auf Anforderung von Spezial E/A-Baugruppen, CPU-Bus-Baugruppen oder vom internen Modul.

Hinweis 1. Die Interrupt-Eingangsbaugruppe muss im CPU-Rack installiert sein. Bei CJ1-H-CPU-Baugruppen muss die Baugruppe als eine der fünf Baugruppen, die der CPU-Baugruppe am nächsten liegen, installiert sein (Steckplätze 0 bis 4). Bei CJ1M CPU-Baugruppen muss die Baugruppe als eine der drei Baugruppen, die der CPU-Baugruppe am nächsten liegen, installiert sein (Steckplätze 0 bis 2). In anderen Steckplätzen installierte E/A-Interrupt-Baugruppen können nicht für die Ausführungsanforderung von E/A-Interrupt-Tasks verwendet werden.

2. Die Spezial-E/A-Baugruppe oder CPU-Bus-Baugruppe muss im CPU-Rack installiert sein. Bei CJ1-H-CPU-Baugruppen muss die Baugruppe als eine der fünf Baugruppen, die der CPU-Baugruppe am nächsten liegen, installiert sein (Steckplätze 0 bis 4). Bei CJ1M CPU-Baugruppen muss die Baugruppe als eine der drei Baugruppen, die der CPU-Baugruppe am nächsten liegen, installiert sein (Steckplätze 0 bis 2). In anderen Steckplätzen installierte Baugruppen können nicht für das Erzeugen externer Interrupts verwendet werden.
3. Die CJ1 CPU-Baugruppen für Duplex-CPU-Systeme unterstützen keine E/A-Interrupt-Tasks und externen Interrupt-Tasks.
4. Die CS1D CPU-Baugruppen für Duplex-CPU-Systeme unterstützen keine Interrupts. Für CS1D CPU-Baugruppen können Interrupt-Tasks nur als zyklische Zusatztasks verwendet werden, d. h., dass andere Arten von Interrupt-Tasks nicht verwendet werden können.

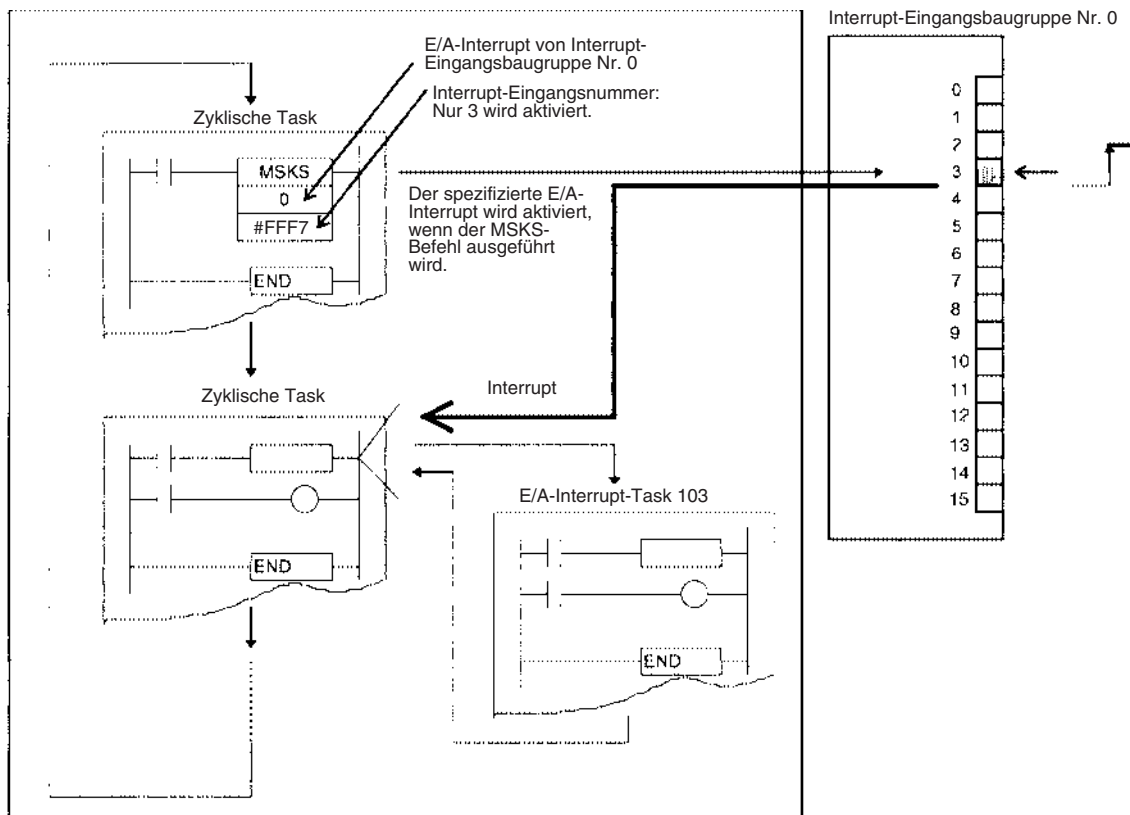
E/A-Interrupt-Tasks: Tasks 100 bis 131

E/A-Interrupt-Tasks werden beim Start der Ausführung von zyklischer Task standardmäßig deaktiviert. Führen Sie zum Aktivieren von E/A-Interrupts in einer zyklischen Task den Befehl MSKS (Interrupt-Maske setzen) für die Interrupt-Nummer der Interrupt-Eingangsbaugruppe durch.

Beispiel: Das folgende Beispiel zeigt die Ausführung der E/A-Interrupt-Task 103 wenn der Interrupt-Eingang Nr. 3 der Interrupt-Eingangsbaugruppe Nr. 0 (die linke der beiden Baugruppen 0 und 1) auf EIN gesetzt wird.

Hinweis

Aktivieren Sie keine unnötigen E/A-Interrupt-Tasks. Wenn der Interrupt durch eine elektrische Störung ausgelöst wird und keine entsprechende Interrupt-Task vorhanden ist, wird das Programm aufgrund eines schwerwiegenden Fehlers (Taskfehler) beendet.

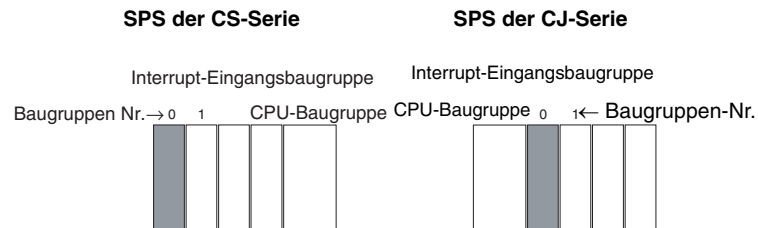


Interrupt-Eingangsbaugruppen-Nummern, Eingang-Nummern und E/A-Interrupt-Task-Nummern

Nummer der Interrupt-Eingangsbaugruppe (siehe Hinweis).	Eingang Nr.	E/A-Interrupt-Task
0	0 bis 15	100 bis 115
1	0 bis 15	116 bis 131

Hinweis

Bei SPS der CS-Serie erfolgt die Nummerierung der Interrupt-Eingangsbaugruppen in der Reihenfolge von 0 bis 1 beginnend auf der linken Seite des CPU-Racks. Bei SPS der CJ-Serie erfolgt die Nummerierung der Interrupt-Eingangsbaugruppen in der Reihenfolge von 0 bis 1 beginnend an der CPU-Baugruppe.



Operand S (der zweite Operand) von MSKS: Die Bits von FFF7 Hex entsprechen den Interrupt-Eingängen der Interrupt-Eingangsbaugruppe. Die Interrupt-Eingangsnummern 0 bis 15 entsprechen den Bits 0 bis 15.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	1	1	1	0	1	1	1	0
F Hex				F Hex				F Hex				7 Hex			

Zeitgesteuerte Interrupt-Tasks: Tasks 2 und 3

Zeitgesteuerte Interrupt-Tasks sind im Standard-SPS-Setup beim Start der Ausführung zyklischer Task deaktiviert. Gehen Sie nach folgenden Schritte vor, um zeitgesteuerte Interrupt-Tasks zu aktivieren.

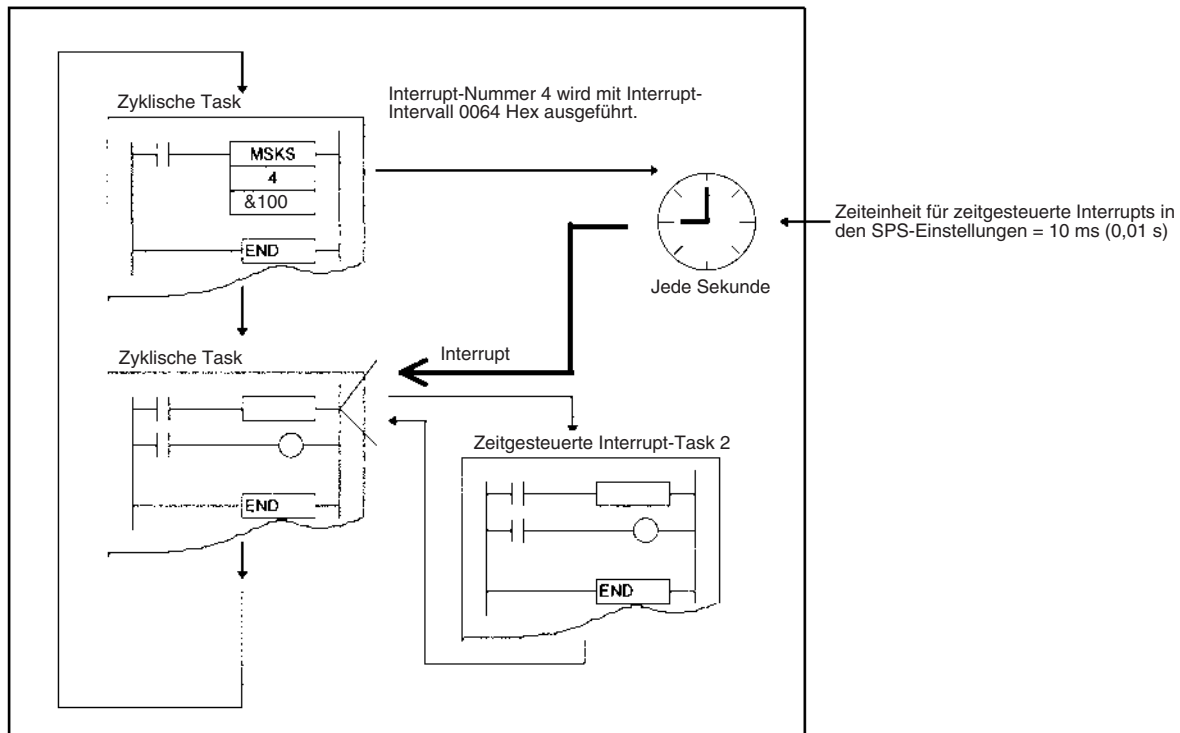
1,2,3...

1. Führen Sie in einer zyklischen Task den Befehl MSKS (Interrupt-Maske setzen) aus, und legen Sie die Zeit (das Intervall) für die Ausführung des spezifizierten zeitgesteuerten Interrupts fest.
2. Stellen Sie im SPS-Setup die Zeiteinheit für den zeitgesteuerten Interrupt ein.

Hinweis

Die Einstellung der Interrupt-Zeit beeinflusst die zyklische Task, da kürzere Interrupt-Zeiten zu einer häufigeren Ausführung der Task und somit zu längeren Zykluszeiten führen.

Beispiel: Das folgende Beispiel zeigt die sekundliche Ausführung der zeitgesteuerten Interrupt-Task 2.



Interrupt-Nummern und Nummern der zeitgesteuerten Interrupt-Tasks

Interrupt-Nr.	Zeitgesteuerte Interrupt-Task
4	2
5	3

SPS-Setup-Einstellungen

Adresse	Bezeichnung	Beschreibung	Einstellungen	Standardeinstellung
Bits 0 bis 3 von 195	Zeiteinheiten für zeitgesteuerten Interrupt	Legt die Zeiteinheit für zeitgesteuerte Interrupts fest, damit Interrupt-Tasks in bestimmten Intervallen ausgeführt werden.	00 Hex: 10 ms 01 Hex: 1,0 ms 02 Hex: 0,1 ms (nur CJ1M CPU-Baugruppen)	00 Hex

Ausschalt-Interrupt-Task: Task 1

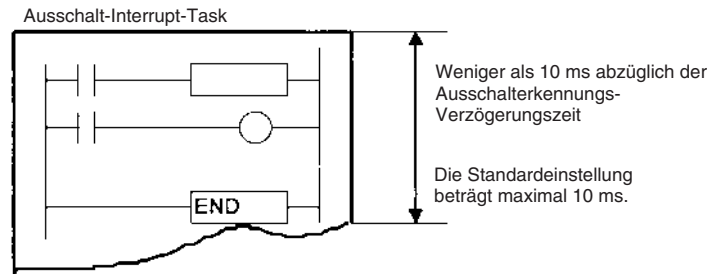
Die Ausschalt-Interrupt-Task ist in der Standardeinstellung des SPS-Setup beim Start der Ausführung zyklischer Task deaktiviert.

Sie kann jedoch im SPS-Setup aktiviert werden.

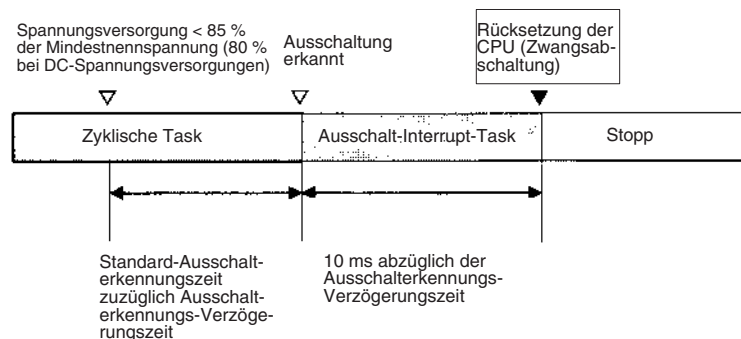
Im standardmäßigen SPS-Setup wird die Ausschalt-Interrupt-Task nach 10 ms gestoppt. Sie muss also in weniger als 10 ms ausgeführt werden.

Wenn im SPS-Setup eine Ausschalterkennungs-Verzögerungszeit eingestellt ist, wird die Ausschalt-Interrupt-Task nach 10 ms abzüglich der Ausschalterkennungs-Verzögerungszeit beendet. In diesem Fall muss die Ausschalt-Interrupt-Task in weniger als 10 ms abzüglich der im SPS-Setup eingestellten Ausschalterkennungs-Verzögerungszeit ausgeführt werden.

Beispiel: Wenn die Ausschalterkennungs-Verzögerungszeit im SPS-Setup auf 4 ms eingestellt ist, muss die Ausführungszeit weniger als 10 minus 4 ms (also 6 ms) betragen.

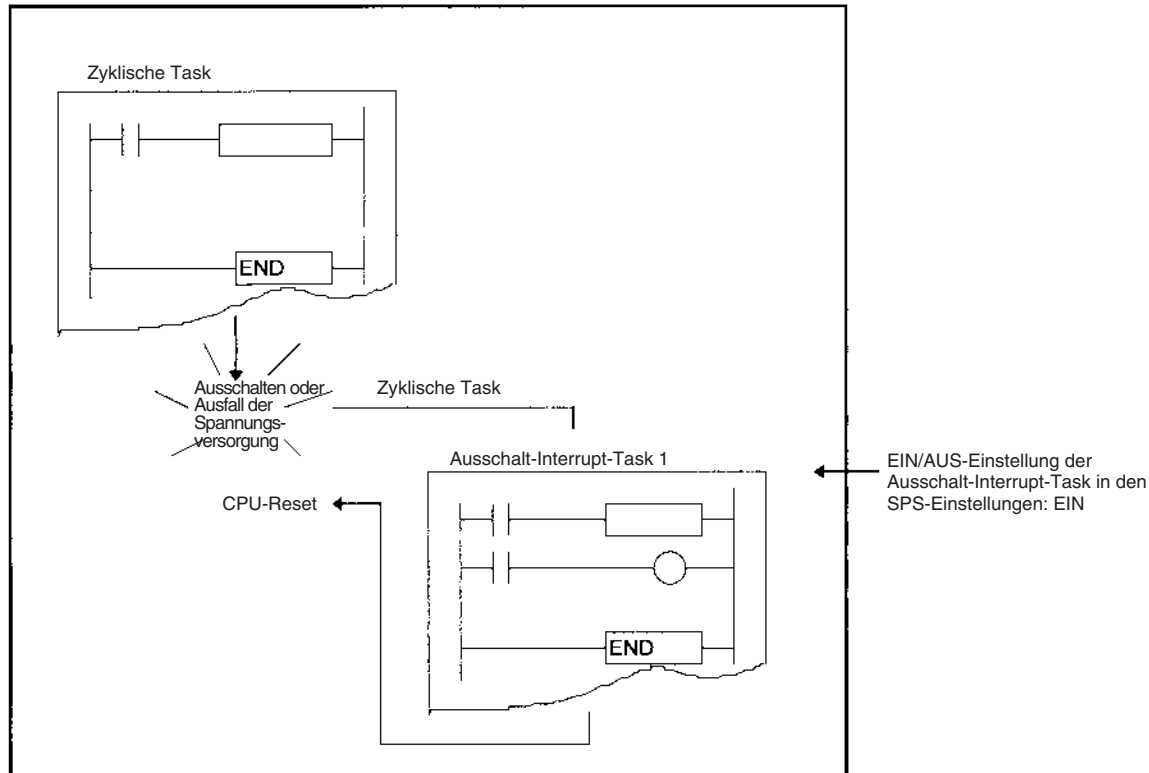


Hinweis Ein Ausschaltzustand wird erkannt, wenn die Spannungsversorgung unter 85 % der Mindest-Nennspannung (80 % bei Gleichspannungsversorgung) fällt. Die Zeitspanne bis zur Ausführung der Ausschalt-Interrupt-Task ist die standardmäßige Ausschalt-Erkennungszeit (10 bis 25 ms bei AC-Spannungsversorgung und 2 bis 5 ms bei DC-Spannungsversorgung) plus die im SPS-Setup eingestellte Ausschalterkennungs-Verzögerungszeit (0 bis 10 ms). Zyklische Tasks werden innerhalb dieses Zeitraums ausgeführt.



Hinweis Stellen Sie sicher, dass die Ausschalt-Interrupt-Task in weniger als 10 ms abzüglich der im SPS-Setup eingestellten Ausschalterkennungs-Verzögerungszeit ausgeführt werden kann. Nach Ablauf dieser Zeitspanne werden verbleibende Befehle nicht mehr ausgeführt. Die Ausschalt-Interrupt-Task wird nicht ausgeführt, wenn die Spannungsversorgung während der Online-Bearbeitung unterbrochen wird. Zusätzlich zu den in sämtlichen Interrupt-Tasks nicht verwendbaren Befehlen (weitere Informationen hierzu finden Sie im *Befehls-Referenzhandbuch*) können folgende Befehle in der Ausschalt-Interrupt-Task nicht verwendet werden: READ DATA FILE (Datei lesen): FREAD(700), WRITE DATA FILE (Datei schreiben): FWRT(701), NETWORK SEND (Netzwerk Senden): SEND(090), NETWORK RECEIVE (Netzwerk Empfangen): RECV(098), DELIVER COMMAND (FINS-Befehl zustellen): CMND(490), TRANSMIT (Senden): TXD(236), RECEIVE (Empfangen): RXD(235) und PROTOCOL MACRO (Protokoll-Makro): PMCR(260).

Ausführung der Ausschalt-Interrupt-Task



SPS-Setup-Einstellungen für die Ausschalt-Interrupt-Task (Task-Nr: 1)

Adresse	Bezeichnung	Beschreibung	Einstellungen	Standard-einstellung
Bit 15 bei 225	Ausschalt-Interrupt-Task	Wenn Bit 15 von 225 auf EIN gesetzt ist, wird eine Ausschalt-Interrupt-Task gestartet, wenn die Spannungsversorgung ausgeschaltet wird.	0: AUS, 1: EIN	0
Bits 0 bis 7 bei 225	Ausschalte-kennungs-Ver-zögerungszeit	Eine Unterbrechung der Spannungsversorgung wird nach Ablauf dieser Zeitspanne zuzüglich der standardmäßigen Ausschalt-Erkennungszeit (10 bis 25 ms bei AC- und 2 bis 5 ms bei DC-Spannungsversorgung) erkannt.	00 bis 0A Hex: 0 bis 10 ms (1-ms-Schritte)	00 Hex

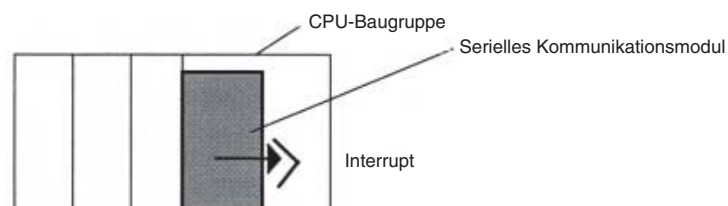
Externe Interrupt-Tasks: Tasks 0 bis 255

Externe Interrupt-Tasks können jederzeit empfangen werden.

Die Interrupt-Verarbeitung wird in der CPU-Baugruppe einer SPS ausgeführt, die über ein internes Steckmodul (nur CS-Serie), Spezial-E/A-Baugruppen oder CPU-Bus-Baugruppen verfügt. Für die CPU-Baugruppe müssen keine Einstellungen vorgenommen werden, sofern das Programm keine externe Interrupt-Task für eine besondere Tasknummer enthält.

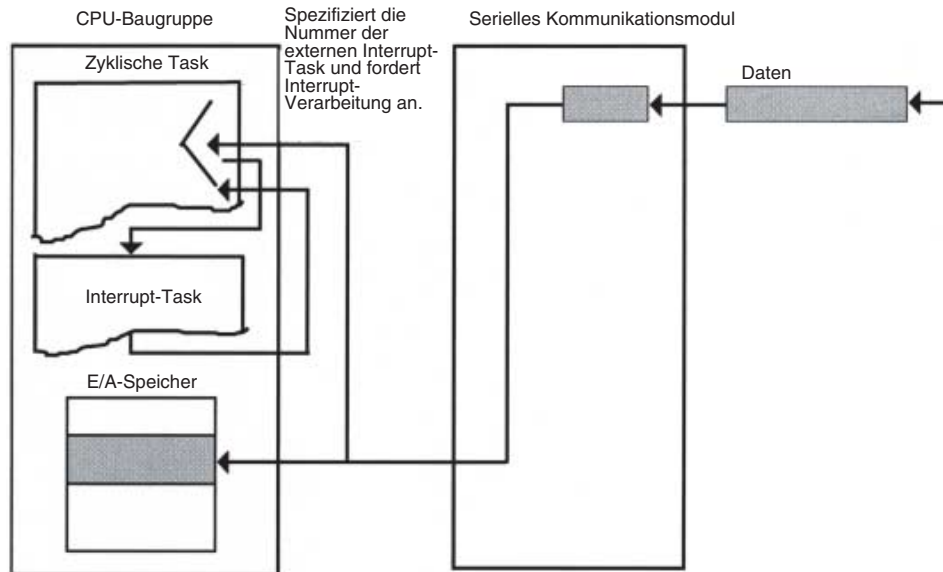
Externe Interrupts werden von CJ1 CPU-Baugruppen nicht unterstützt.

Beispiel: Das folgende Beispiel zeigt einen externen Interrupt, der von dem seriellen Kommunikationsmodul CS1W-SCB□1 erzeugt wird.



Wenn die Antworttype des seriellen Kommunikationsmoduls auf Interrupt mit fester Nummer oder Interrupt mit empfangener Nummer eingestellt ist, fordert

das Modul die Ausführung einer externen Interrupt-Task in der CPU-Baugruppe an, nachdem die Daten von der seriellen Schnittstelle empfangen und in den E/A-Speicher der CPU-Baugruppe geschrieben wurden.



Hinweis

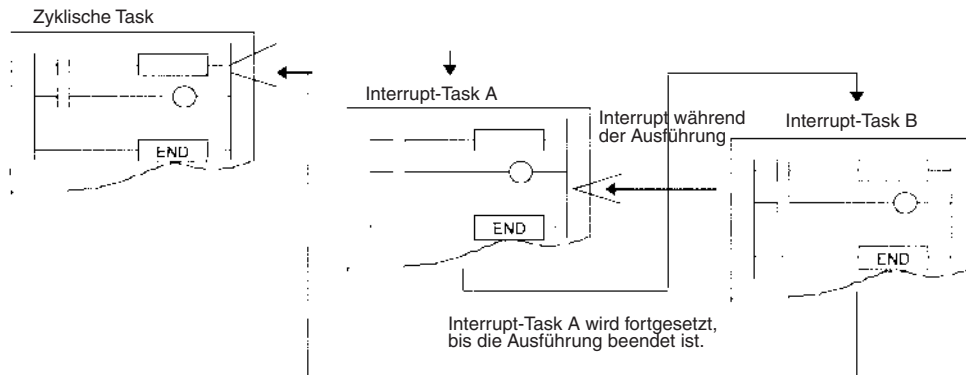
1. Wenn die Antworttype auf Interrupt mit fester Nummer eingestellt ist, wird die Ausführung der Interrupt-Task mit der voreingestellten Tasknummer angefordert.
2. Wenn die Antworttype auf Interrupt mit empfangener Nummer eingestellt ist, wird die Nummer der externen Interrupt-Task mit der spezifizierten Formel errechnet und das Modul fordert die Ausführung der Interrupt-Task mit dieser Nummer an.
3. Wenn eine externe Interrupt-Task (0 bis 255) über die gleiche Nummer verfügt wie die Ausschalt-Task (Task 1), zeitgesteuerte Interrupt-Tasks (Task 2 oder 3) oder E/A-Interrupt-Tasks (100 bis 131), wird die Interrupt-Task bei beiden Interrupt-Bedingungen (externer Interrupt oder die andere Interrupt-Bedingung) ausgeführt. Grundsätzlich sollten Tasknummern jedoch nicht doppelt vergeben werden.

4-3-2 Prioritäten von Interrupt-Tasks

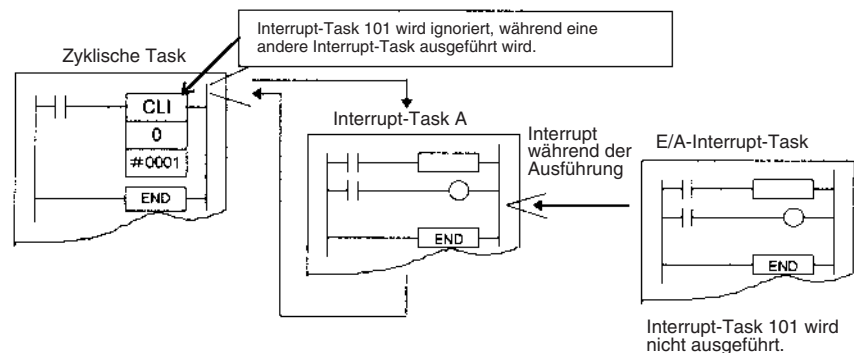
Die Ausführung einer anderen Interrupt-Task wird beendet, um die Ausführung der Ausschalt-Interrupt-Task zu ermöglichen. Es erfolgt eine Rücksetzung der CPU, und die beendete Interrupt-Task wird nicht im Anschluss an die Ausführung der Ausschalt-Interrupt-Task ausgeführt.

Interrupt während der Ausführung von Interrupt-Tasks

Wenn ein Interrupt während der Ausführung einer anderen Interrupt-Task auftritt, wird die Task für den Interrupt nicht ausgeführt, bevor die Ausführung der ursprünglichen Interrupt-Task abgeschlossen ist.

**Hinweis**

Wenn eine E/A-Interrupt-Task mit einer bestimmten Nummer bei einer CPU-Baugruppe der CS-Serie nicht gespeichert und ausgeführt werden soll, wenn diese während der Ausführung einer anderen Interrupt-Task auftritt, führen Sie in der anderen Interrupt-Task den Befehl CLI (Interrupt löschen) aus, um die intern gespeicherte Interrupt-Nummer zu löschen. Zeitgesteuerte Interrupts und externe Interrupts können nicht abgebrochen werden.

**Gleichzeitiges Auftreten mehrerer Interrupts**

Interrupt-Tasks (mit Ausnahme der Ausschalt-Interrupt-Task) werden beim gleichzeitigen Auftreten mehrerer Interrupts in der folgenden Prioritätsreihenfolge ausgeführt.

E/A-Interrupt-Tasks > externe Interrupt-Tasks > zeitgesteuerte Interrupt-Tasks

Jeder der verschiedenen Arten von Interrupt-Tasks wird in der Reihenfolge beginnend mit der jeweils niedrigsten Nummer ausgeführt.

Hinweis

Im Speicher wird für jede Interrupt-Task jeweils nur ein Interrupt aufgezeichnet. Ein Interrupt für einen bereits ausgeführten Interrupt wird nicht aufgezeichnet. Aufgrund der niedrigen Priorität von zeitgesteuerten Interrupts, und da nur jeweils ein Interrupt aufgezeichnet wird, werden zeitgesteuerte Interrupts möglicherweise übersprungen.

4-3-3 Merker und Worte für Interrupt-Tasks**Längste Interrupt-Task-Ausführungsdauer (A440)**

Die längste Interrupt-Task-Ausführungsdauer wird als Binärwert in 0,1-ms-Einheiten gespeichert und beim Betriebsstart gelöscht.

Interrupt-Task mit der längsten Ausführungsdauer (A441)

Die Nummer der Interrupt-Task mit der längsten Ausführungsdauer wird als Binärwert gespeichert. Die Werte 8000 bis 80FF Hex entsprechen hierbei den Tasknummern 00 bis FF Hex.

A44115 wird auf EIN gesetzt, wenn der erste Interrupt nach dem Betriebstart auftritt. Die Tasknummer mit der längsten Ausführungsdauer für aufeinander folgende Interrupt-Tasks wird in den beiden rechten Stellen als Hexadezimalwert gespeichert und beim Start des Betriebs gelöscht.

Interrupt-Task-Fehlermerker (nicht schwerwiegender Fehler) (A40213)

Wenn die Erkennung von Interrupt-Task-Fehlern im SPS-Setup auf EIN gesetzt ist, wird der Interrupt-Task-Fehlermerker auf EIN gesetzt, wenn ein Interrupt-Task-Fehler auftritt.

Interrupt-Task-Fehlermerker (A42615)/Tasknummer, die den Interrupt-Task-Fehler verursacht (A42600 bis 42611)

Wenn A40213 auf EIN gesetzt wird, werden die folgenden Daten in A42615 und A42600 bis A42611 gespeichert.

A40213	Interrupt-Task-Fehlerbeschreibung	A42615	A42600 bis 42611
Interrupt-Task-Fehler (sofern die Interrupt-Task-Fehlererkennung im SPS-Setup auf EIN gesetzt ist)	Wenn eine Interrupt-Task während einer Aktualisierung von C200H Spezial-E/A-Baugruppen oder dezentralen SYSMAC BUS-E/A (nur CS-Serie) länger als 10 ms ausgeführt wird.	AUS	Die Nummer der Interrupt-Task wird in 12 binären Datenbits gespeichert (Interrupt-Task 0 bis 255: 000 bis 0FF Hex).
	Wenn versucht wird, aus einer Interrupt-Task heraus mit Hilfe des Befehls IORF eine E/A-Aktualisierung für eine große Anzahl an Worten durchzuführen, während eine Spezial-E/A-Baugruppe durch die zyklische E/A-Aktualisierung aktualisiert wird.	EIN	Die Baugruppennummer der aktualisierten Spezial-E/A-Baugruppe wird in 12 binären Datenbits gespeichert (Baugruppennr. 0 bis 95: 000 bis 05F Hex).

Tasknummer bei Programmabbruch (A294)

Der Tasktyp und die Nummer der aktuellen Task werden bei einem Programmabbruch aufgrund eines Programmfehlers in folgenden Adressen gespeichert:

Typ	A294
Interrupt-Task	8000 bis 80FF Hex (entspricht den Interrupt-Task-Nummern 0 bis 255)
Zyklische Task	0000 bis 001F Hex (entspricht den Task-Nummern 0 bis 31)

4-3-4 Sicherheitshinweise zum Einsatz

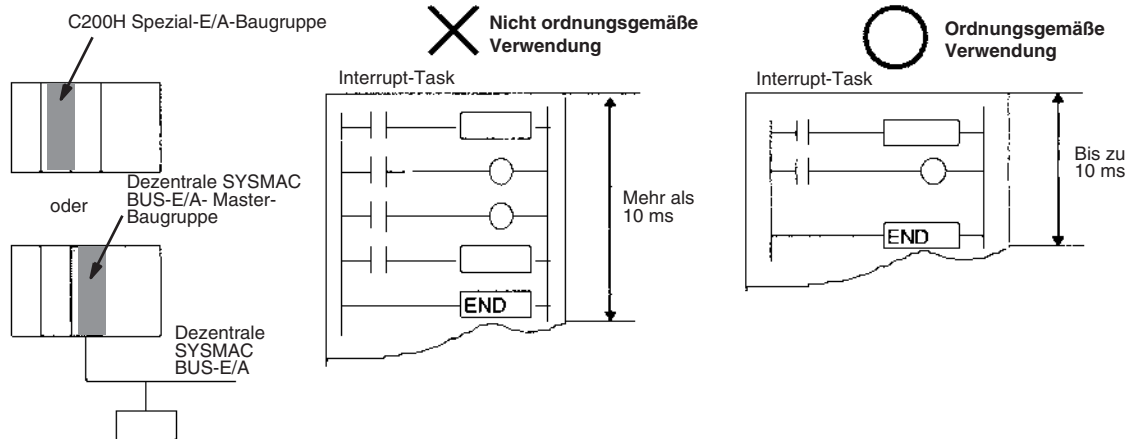
Lange Ausführungszeiten bei C200H Spezial-E/A-Baugruppen oder SYSMAC BUS (nur CS-Serie)

Stellen Sie sicher, dass sämtliche Interrupt-Tasks (E/A-, zeitgesteuerte, Ausschalt- und externe Interrupt-Tasks) innerhalb von 10 ms ausgeführt werden, wenn C200H Spezial-E/A-Baugruppen oder dezentrale SYSMAC BUS-E/A verwendet werden.

Wenn eine Interrupt-Task während einer Aktualisierung einer C200H Spezial-E/A-Baugruppe oder dezentralen SYSMAC BUS-E/A länger als 10 ms ausgeführt wird, tritt ein Taskfehler auf. A40206 (Spezial-E/A-Baugruppen-Fehlermerker) wird auf EIN gesetzt und die E/A-Aktualisierung der Spezial-E/A-Baugruppen wird beendet. Die CPU-Baugruppe setzt den Betrieb jedoch fort.

Wenn die Erkennung von Interrupt-Task-Fehlern im SPS-Setup auf EIN gesetzt ist, wird A40213 (Interrupt-Task-Fehlermerker) auf EIN gesetzt, wenn

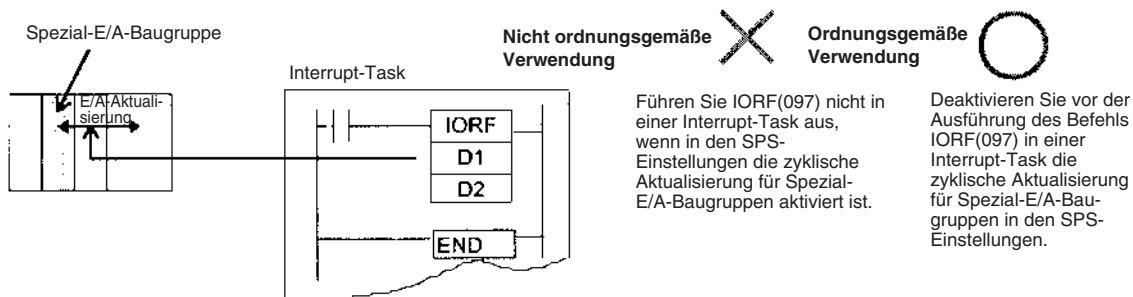
ein Taskfehler auftritt und die Nummer der betroffenen Baugruppe wird in A426 (Interrupt-Task-Fehler, Baugruppen-Nummer) gespeichert. Die CPU-Baugruppe setzt den Betrieb jedoch fort.



Ausführen von IORF für eine Spezial-E/A-Baugruppe

Wenn der Befehl IORF(097) für eine Spezial-E/A-Baugruppe von einer Interrupt-Task ausgeführt werden muss, sollten Sie sicherstellen, dass die zyklische Aktualisierung für die Spezial-E/A-Baugruppe (unter Verwendung der Baugruppennummer) im SPS-Setup auf AUS gesetzt ist.

Ein Interrupt-Task-Fehler tritt auf, wenn versucht wird, eine Spezial-E/A-Baugruppe durch den Befehl IORF(097) in einer Interrupt-Task zu aktualisieren, während diese Baugruppe gleichzeitig durch zyklische E/A-Aktualisierung oder durch einen E/A-Aktualisierungsbefehl (IORF(097) oder einen Befehl für direkte Aktualisierung (vorangestelltes Ausrufezeichen) aktualisiert wird. Wenn die Erkennung von Interrupt-Task-Fehlern im SPS-Setup auf EIN gesetzt ist, wird A40213 (Interrupt-Task-Fehlermerker) auf EIN gesetzt, wenn ein Taskfehler auftritt, und die Baugruppennummer der doppelt aktualisierten Spezial-E/A-Baugruppe wird in A426 (Interrupt-Task-Fehler, Baugruppen-Nummer) gespeichert. Die CPU-Baugruppe setzt den Betrieb fort.



Hinweis Das linke Bit von A426 (Interrupt-Task-Fehler, Baugruppen-Nummer) kann zur Ermittlung des aufgetretenen Interrupt-Task-Fehlers verwendet werden. (Bit 15: Fehler durch Ausführungszeit über 10 ms bei 0, Fehler durch Mehrfach-Aktualisierung bei 1)

SPS-Setup-Einstellungen

Adresse	Bezeichnung	Beschreibung	Einstellungen	Standard-einstellung
Bit 14 bei 128	Interrupt-Task-Fehlererkennung	Legt fest, ob Interrupt-Task-Fehler erkannt werden sollen oder nicht. Wenn die Erkennung aktiviert ist, wird der Interrupt-Task-Fehlermerker (A40213) verwendet.	0: Erkennung aktiviert, 1: Erkennung deaktiviert	0

Zugehörige Merker/Worte im Zusatz-Systembereich

Bezeichnung	Adresse	Beschreibung
Interrupt-Task-Fehler-merker	A40213	Wird auf EIN gesetzt, wenn eine Interrupt-Task während einer Aktualisierung von C200H Spezial-E/A-Baugruppen oder dezentralen SYSMAC BUS-E/A (nur CS-Serie) länger als 10 ms ausgeführt wird. Die CPU-Baugruppe setzt den Betrieb fort. Die ERR/ALM LED-Anzeige an der Gerätefront leuchtet. Wird auf EIN gesetzt, wenn versucht wird, eine Spezial-E/A-Baugruppe mit dem Befehl IORF in einer Interrupt-Task zu aktualisieren, während diese Baugruppe gleichzeitig durch zyklischer E/A-Aktualisierung aktualisiert wird.
Interrupt-Task-Fehler, Baugruppen-Nummer	A426	Enthält die Nummer der aktualisierten Spezial-E/A-Baugruppe. (Bit 15 wird auf AUS gesetzt, wenn die Ausführung einer Interrupt-Task 10 ms oder länger dauert, und auf EIN, wenn eine doppelte Aktualisierung einer Spezial-E/A-Baugruppe aufgetreten ist.)

Deaktivieren von Interrupts

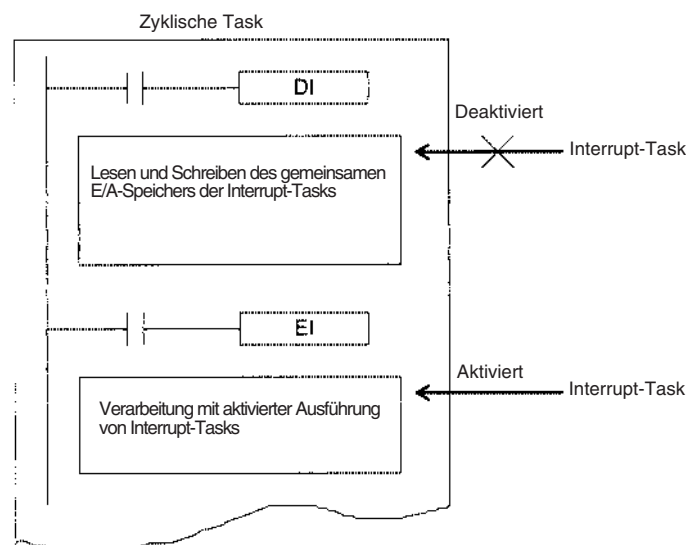
In folgenden Fällen wird die Verarbeitung unterbrochen und die Interrupt-Task ausgeführt.

- Während ein Befehl ausgeführt wird
- Während der Aktualisierung von E/A-Baugruppen, CPU-Bus-Baugruppen, internen Steckmodulen (nur CS-Serie) oder dezentralen SYSMAC BUS-E/A (nur CS-Serie)
- Während der Host-Link-Bedienung

Datenübereinstimmung zwischen zyklischen und Interrupt-Tasks

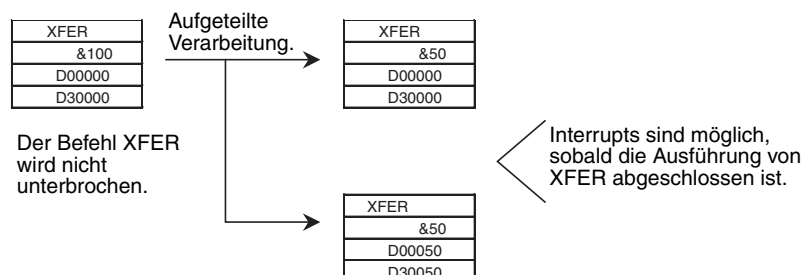
Die Daten stimmen möglicherweise nicht überein, wenn eine zyklische (einschließlich zyklischer Zusatztasks) und eine Interrupt-Task die gleichen E/A-Speicheradressen lesen und schreiben. Verwenden Sie folgendes Verfahren, um Interrupts während des Speicherzugriffs durch Befehle zyklischer Tasks zu verhindern.

- Verwenden Sie unmittelbar vor dem Lesen oder Schreiben durch einen Befehl einer zyklischen Task den Befehl DI (Interrupts deaktivieren), um die Ausführung von Interrupt-Tasks zu deaktivieren.
- Verwenden Sie unmittelbar im Anschluss an die Verarbeitung den Befehl EI (Interrupts aktivieren), um die Ausführung von Interrupt-Tasks wieder zu aktivieren.



Es können jedoch selbst dann Probleme mit der Datenübereinstimmung auftreten, wenn die Befehle DI(693) und EI(694) zur Deaktivierung der Ausführung von Interrupt-Tasks während der Ausführung eines Befehls verwendet werden, der Antwortempfang und -verarbeitung erfordert (beispielsweise Netzwerkbefehle oder Befehle für serielle Kommunikation).

Hinweis Bei den CS1-H, CJ1-H, CJ1M und CS1D CPU-Baugruppen wird die Ausführung der Befehle BIT COUNTER (BCNT), BLOCK SET (BSET) und BLOCK TRANSFER (XFER) nicht für die Ausführung von Interrupt-Tasks unterbrochen. Die Ausführung des Befehls wird abgeschlossen, bevor die Interrupt-Task ausgeführt wird, so dass die Antwort des Interrupts verzögert wird. Um dies zu vermeiden, teilen Sie die Datenverarbeitung für diese Befehle in mehrere Befehle auf, wie im folgenden Beispiel für den Befehl XFER gezeigt.



4-4 Task-Programmierung mit Programmiergeräten

4-4-1 Verwenden mehrerer zyklischer Tasks

Verwenden Sie CX-Programmer, um mehrere zyklische Tasks (einschließlich zyklischer Zusatztasks) zu erstellen. Mit einer Programmierkonsole können keine neuen zyklischen Tasks erstellt werden. Verwenden Sie in jedem Fall CX-Programmer, um erstellten Programmen einen Tasktyp und die Task-Nummer zuzuordnen.

- Mehrere zyklische Tasks, die mit CX-Programmer erstellt und an eine CPU-Baugruppe übertragen wurden, können mit einer Programmierkonsole überwacht und bearbeitet werden.
- Die Programmierkonsole kann verwendet werden, um eine zyklische Task und eine oder mehrere Interrupt-Tasks zu erstellen, indem die Funktion zum vollständigen Löschen verwendet wird und Interrupt-Tasks spezifiziert werden. Mit einer Programmierkonsole können nur die Interrupt-Tasks 1 (Ausschalt-Interrupt), 2 und 3 (zeitgesteuerte Interrupts) sowie 100 bis 131 (E/A-Interrupts) erstellt werden. Bei CJ1M CPU-Baugruppen können jedoch auch die Interrupt-Tasks 140 bis 143 (für integrierte Eingänge) erstellt werden. Die zyklische Task 0 startet mit dem SPS-Betrieb.

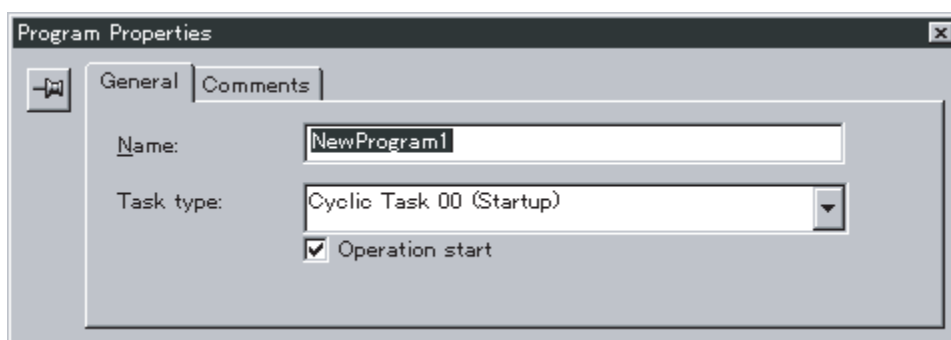
4-4-2 Bedienverfahren bei Programmiergeräten

CX-Programmer

Geben Sie für jedes Programm Tasktyp und Tasknummer als Attribute an.

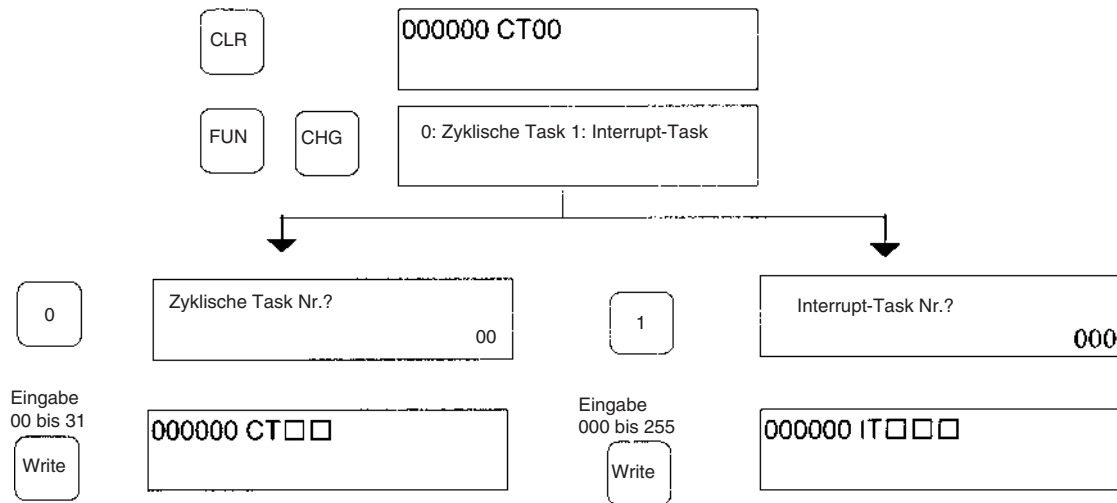
1,2,3...

1. Wählen Sie **Anzeigen/Eigenschaften**, oder klicken Sie mit der rechten Maustaste, und wählen Sie im Popup-Menü **Eigenschaften**, um die einer Task zuzuordnende Programmeigenschaften anzuzeigen.
2. Wählen Sie die Registerkarte **Allgemeines** und anschließend **Tasktyp** und **Task-Nr.** aus. Klicken Sie zur Aktivierung der zyklischen Task auf das Kontrollkästchen **Start**.



Programmierkonsole

Eine Task wird von der Programmierkonsole als gesamtes Programm behandelt. Programme werden über eine Programmierkonsole aufgerufen und bearbeitet, indem CT00 bis CT31 für zyklische Tasks und IT00 bis IT255 für Interrupt-Tasks spezifiziert werden.

**Hinweis**

1. Mit einer Programmierkonsole können keine neuen zyklischen Tasks erstellt werden.
2. CPU-Baugruppen der CJ-Serie unterstützen keine E/A-Interrupt-Tasks und externe Interrupt-Tasks. Es können nur IT001 bis IT003 spezifiziert werden.

ABSCHNITT 5

Dateispeicherfunktionen

Dieses Kapitel enthält eine Beschreibung der Funktionen, die für die Bearbeitung des Dateispeichers verwendet werden.

5-1	Dateispeicher.	192
5-1-1	Arten von Dateispeicher	193
5-1-2	Dateidaten	195
5-1-3	Dateien	196
5-1-4	Beschreibung von Dateioperationen	207
5-1-5	Anwendungen	209
5-2	Arbeiten mit Dateien	211
5-2-1	Programmiergeräte (einschließlich Programmierkonsolen)	211
5-2-2	FINS-Befehle	215
5-2-3	FREAD(700), FWRIT(701) und CMND(490).	216
5-2-4	Austausch des gesamten Programms während des Betriebs	221
5-2-5	Automatische Übertragung beim Start	228
5-2-6	Easy-Backup-Funktion	234
5-3	Verwenden des Dateispeichers	247
5-3-1	Initialisieren von Medien	247
5-3-2	Bedienung	249
5-3-3	Unterbrechungen der Versorgungsspannung während des Zugriffs auf den Dateispeicher . .	254

5-1 Dateispeicher

Der Dateispeicher wird von der CS/CJ-Serie unterstützt. Die folgenden Medien können zur Speicherung von Dateien verwendet werden.

1,2,3...

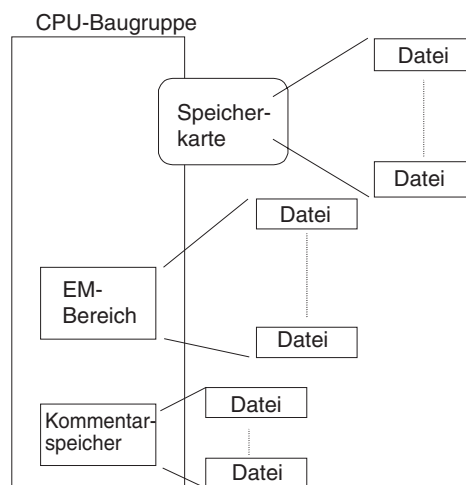
1. Speicherkarten
2. Ein spezifizierter Bereich im EM-Bereich, der EM-Dateispeicher

Hinweis CJ1M CPU-Baugruppen haben keinen EM-Bereich, daher kann kein EM-Datenspeicher verwendet werden.

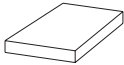
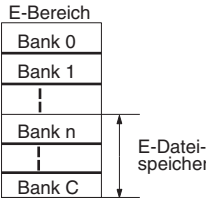
Beide Arten von Speicher können zum Speichern des gesamten Anwenderprogramms, des E/A-Speichers und der Parameterbereiche in Dateiform verwendet werden.

3. Kommentarspeicher (im internen Flash-Speicher der CPU)

Detailinformationen zur Auswahl von Dateispeicher finden Sie im Abschnitt 5-1-5 Anwendungen auf Seite 209.



5-1-1 Arten von Dateispeicher

Kategorie	Typ	Kapazität	Produktbezeichnung	Von der CPU-Baugruppe erkannte Dateidaten	Zulässige Dateioperationen
Speicherkarten 	Flash-Speicher	30 MB 64 MB	HMC-EF372 HMC-EF672	1) Gesamtes Anwenderprogramm 2) Spezifizierter Bereich des E/A-Speichers 3) Parameterbereichsdaten (SPS-Setup und andere Einstellungen) Siehe Hinweis 4	Alle Operationen sind möglich. (Detaillierte Informationen hierzu finden Sie auf Seite 207.)
E-Dateispeicher 	RAM	Kapazität des EM-Bereichs von CPU-Baugruppen CS-Serie CS1H-CPU67H: 832 KB (Bänke 0 bis C: E0_00000 bis EC_00000) CJ-Serie CJ1H-CPU66H: 448 KB (Bänke 0 bis 6: E0_00000 bis E6_00000)	Von der spezifizierten Bank im EM-Bereich des E/A-Speichers bis zur letzten Bank (im SPS-Setup spezifiziert).		Mit der Funktion zur automatischen Übertragung beim Start können keine Daten aus dem EM-Dateispeicher übertragen werden. (Detaillierte Informationen hierzu finden Sie auf Seite 207.)
Kommentarspeicher	Interner Flash-Speicher der CPU-Baugruppe	Kommentardateien: CPU66H/67H: 128 kByte Andere CPU-Baugruppen: 64 kByte Programmindexdateien: CPU66H/67H: 128 kByte Andere CPU-Baugruppen: 64 kByte Symboltabellendateien: CPU45H/65H/66H/67H: 128 kByte Andere CPU-Baugruppen: 64 kByte	CPU-Baugruppen mit Baugruppen-Version 3.0 oder höher	CX-Programmer Kommentare Abschnittsbezeichnungen, Abschnittskommentare, Programmkommentare von CX-Programmer. Globale Symboltabellen, lokale Symboltabellen, Setup-Daten für automatischen Zuweisungsbereich von CX-Programmer.	Aktiviert beim Übertragen von Projekten mit CX-Programmer Version 5.0 oder höher, und bei einfachen Datensicherungen.

Hinweis

- Nähere Einzelheiten zum Einsetzen und Entnehmen von Speicherkarten finden Sie unter *5-2 Arbeiten mit Dateien*.
- Initialisieren Sie die Speicherkarte bzw. den EM-Dateispeicher vor der ersten Verwendung. Detaillierte Informationen zur Initialisierung finden Sie unter *5-3 Verwenden des Dateispeichers*.
- Mit dem Speicherkartenadapter HMC-AP001 kann eine Speicherkarte in den PCMCIA-Kartensteckplatz eines Computers eingesteckt werden, um sie als Speichermedium zu nutzen.
- Wenn CX-Programmer verwendet wird, können Symboltabellen (einschließlich E/A-Kommentare) und Kommentare von der CPU-Baugruppe erkannt werden. Das Übertragungsziel ist die Speicherkarte, oder falls eine solche nicht eingesetzt ist, der EM-Dateispeicher.

Vorsichtsmaßnahmen bei Speicherkarten

Prüfen Sie vor der Verwendung einer Speicherkarte Folgendes:

Format

Speicherkarten werden vor der Auslieferung formatiert. Nach dem Kauf muss keine Formatierung erfolgen. Wenn Sie Speicherkarten formatieren möchten, nachdem diese verwendet wurden, führen Sie den Vorgang immer in der CPU-Baugruppe mittels CX-Programmer oder einer Programmierkonsole durch.

Wenn eine Speicherkarte direkt in einem Notebook-Computer oder einem anderen Computer formatiert wird, kann sie von der CPU-Baugruppe möglicherweise nicht erkannt werden. Sie können die Speicherkarte in diesem Fall nicht mehr verwenden, selbst dann nicht, wenn Sie in der CPU-Baugruppe neu formatiert wird.

Anzahl der Dateien im Dateiverzeichnis

Die Anzahl der Dateien, die im Dateiverzeichnis einer Speicherkarte gespeichert werden können, ist begrenzt (wie es auch für Festplatten eine Begrenzung gilt). Die Begrenzung ist zwar vom Typ und Format der Speicherkarte abhängig, jedoch liegt die Anzahl der Dateien zwischen 128 und 512. Bei Anwendungen, die in bestimmten Abständen Protokolldateien oder andere Dateien schreiben, sollten die Dateien in ein Unterverzeichnis und nicht in das Hauptverzeichnis geschrieben werden.

Unterverzeichnisse können auf dem Computer oder mit Hilfe des Befehls CMND(490) erstellt werden. Ein Beispiel zur Verwendung des Befehls CMND(490) finden Sie unter 3-25-5 *DELIVER COMMAND: CMND(490)* im *CS/CJ-Serie Befehls-Referenzhandbuch*.

Anzahl der Schreibvorgänge

Es gibt im Allgemeinen keine Begrenzung hinsichtlich der Anzahl der Schreibvorgänge, die bei einem Flash-Speicher ausgeführt werden können. Für Speicherkarten wurde aus Gewährleistungsgründen jedoch eine Begrenzung auf 100.000 Schreibvorgänge festgelegt. Wenn beispielsweise alle 10 Minuten auf eine Speicherkarte geschrieben wird, werden innerhalb von 2 Jahren mehr als 100.000 Schreibvorgänge ausgeführt.

Mindestdateigröße

Wenn auf einer Speicherkarte viele kleine Dateien, wie die nur wenige Worte beinhaltenden Dateien der Daten im DM-Bereich, gespeichert werden, ist es nicht möglich, die gesamte Kapazität der Speicherkarte zu nutzen. Wenn beispielsweise eine Speicherkarte mit einer Zuordnungseinheitsgröße von 4.096 Bytes verwendet wird, werden unabhängig von der Größe der Datei mindestens 4.096 Byte Speicherplatz je Datei verwendet. Wenn Sie 10 Worte Daten des DM-Bereichs auf der Speicherkarte speichern, werden 4.096 Byte Speicherplatz in Anspruch genommen, obwohl die tatsächliche Dateigröße lediglich 68 Byte beträgt. Die Verwendung kleiner Dateigrößen reduziert die Ausnutzung der Speicherkarte erheblich. Wenn die Größe der Zuordnungseinheit jedoch reduziert wird, um die Ausnutzung zu erhöhen, verringert sich dadurch die Zugriffsgeschwindigkeit.

Die Größe der Zuordnungseinheit auf der Speicherkarte kann in der DOS-Eingabeaufforderung mittels CHKDSK überprüft werden. Die genaue Vorgehensweise wird hier nicht ausgeführt. Weitere Informationen zur Größe von Zuordnungseinheiten finden Sie in den allgemeinen Computerunterlagen.

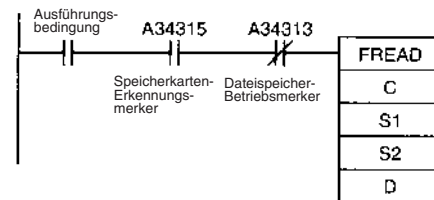
Vorsichtsmaßnahmen zum Speicherkartenzugriff

Beim Zugriff der SPS auf die Speicherkarte leuchtet die BUSY-Anzeige auf der CPU-Baugruppe. Beachten Sie die folgenden Vorsichtsmaßnahmen.

1,2,3...

1. Schalten Sie die Spannungsversorgung der CPU-Baugruppe niemals AUS, während die BUSY-Anzeige leuchtet. Die Speicherkarte kann dadurch unbrauchbar werden.
2. Entnehmen Sie die Speicherkarte niemals aus der CPU-Baugruppe, während die BUSY-Anzeige leuchtet. Drücken Sie zunächst die Speicherkartentaste der CPU-Baugruppe, und warten Sie, bis die Anzeige BUSY erloschen ist, bevor Sie die Speicherkarte entnehmen. Die Speicherkarte kann unbrauchbar werden, wenn Sie nicht in dieser Weise vorgehen.
3. Setzen Sie die Speicherkarte so in den Speicherkartensteckplatz ein, dass die beschriftete Seite der Speicherkarte nach rechts weist. Versuchen Sie nicht, die Karte in eine andere Ausrichtung einzuführen. Die Speicherkarte und die CPU-Baugruppe könnten beschädigt werden.

4. Nach dem Einsetzen der Speicherkarte dauert es einige Sekunden, bis sie von der CPU-Baugruppe erkannt wird. Wenn Sie auf eine Speicherkarte sofort nach dem Einschalten der Spannungsversorgung oder nach dem Einsetzen der Karte zugreifen, müssen Sie eine Schließer-Bedingung für den Speicherkarte-erkannt-Merker (A34315) als Eingabebedingung programmieren, wie im Folgenden dargestellt.



5-1-2 Dateidaten

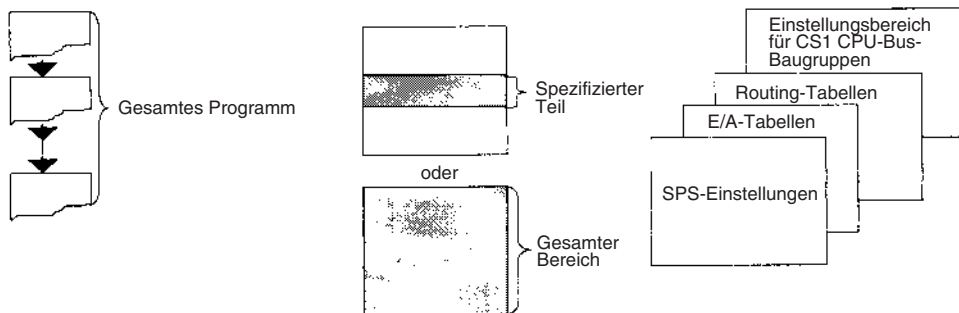
Die folgende Tabelle gibt die Typen der schreibbaren Dateien an.

Dateityp	Operation
• Programmdateien • Datendateien • Parameterdateien	Auf diese Dateien kann mittels eines Programmiergeräts (CX-Programmer oder Programmierkonsole), FINS-Befehlen, Kontaktplanbefehlen, Easy Backup und speziellen Steuerbits im Speicher der CPU-Baugruppe zugegriffen (Lesen, Schreiben usw.) werden.
• Symboltabellendateien • Kommentardateien • Programmindexdateien	Auf diese Dateien kann mittels CX-Programmer und mittels Easy Backup zugegriffen (Lesen, Schreiben usw.) werden.

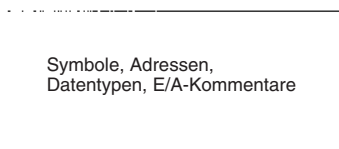
Anwenderprogramm:
Programmdatei des gesamten Programms einschließlich Task-Attribute

Spezifizierter Bereich des Speichers:
Datendateien
Gesamter Bereich oder ein spezifizierter Teil eines Speicherbereichs

Parameterbereichsdaten:
Parameterdatei
Grundeinstellungen der CX-Programmer CPU-Baugruppe.



Symboltabellendateien
Von CX-Programmer verwendete Variablentabellen

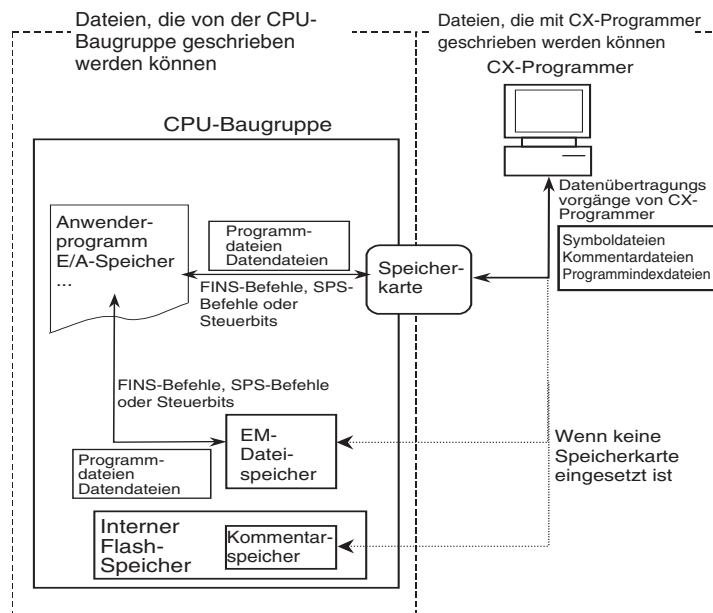


Von CX-Programmer verwendete Kommentare in **Kommentardateien**



Programmindexdatei
Abschnittsinformationen (von CX-Programmer verwendet)

Netzwerk-Kommentare
Abschnittsbezeichnungen, Abschnittsbefehle
Kommentare (Anmerkungen)



Hinweis Symboltabellen (Symbole, Adressen und E/A-Kommentare) können von CX-Programmer als Dateien behandelt werden.

Datei	Dateiname	Erweiterung	Inhalt
Symboleibellendatei	SYMBOLS	.SYM	Globale und lokale Symbole
Kommentardatei	COMMENTS	.CMT	Kommentare (Anmerkungen)
Programmindexdatei	PROGRAM	.IDX	Abschnittsbezeichnungen und Abschnittsbefehle

Datenübertragungsvorgänge für Projekte können mittels CX-Programmer durchgeführt werden. Dabei werden alle der oben genannten Dateitypen (Symboleibellendateien, Kommentardateien und Programmindexdateien) zwischen der CPU-Baugruppe und einer Speicherkarte oder dem EM-Dateispeicher übertragen. (Das Übertragen von Programmindexdateien wird ab Version 2.0 unterstützt.) Die Symboleibellendateien und Kommentardateien können auch zwischen CX-Programmer, dem RAM-Speicher des Computers und einem Datenspeichergerät mit CX-Programmer ab Version 1.2 übertragen werden.

5-1-3 Dateien

Die Dateien sind in DOS formatiert und können daher auf Windows-Computern wie normale Dateien verwendet werden.

Die Dateien besitzen, wie in der folgenden Tabelle dargestellt, einen Dateinamen mit Namensenerweiterung. Ein Dateiname kann folgende Zeichen beinhalten: Die Buchstaben A bis Z, die Zahlen 0 bis 9, !, &, \$, #, ` , {, }, -, ^, (,) und _

Die folgenden Zeichen dürfen in Dateinamen nicht verwendet werden: ,, , /, ¥, ?, *, ", ;, :, <, >, =, +, Leerzeichen und 2-Byte-Zeichen.

Die Dateinamenerweiterungen sind von dem zu speichernden Dateityp abhängig. Folgende Erweiterungen sind für Datendateien zulässig: IOM, TXT, CSV und IOR. (Erweiterungen TXT, CSV und IOR werden von CS1 CPU-Baugruppen der CS-Serie vor EV1 nicht unterstützt.) Programmdateien tragen die Erweiterung OBJ und Parameterdateien die Erweiterung STD. Der Speicherort einer Datei im Speicher kann im Verzeichnis spezifiziert werden, wobei ein Verzeichnis bis zu fünf Unterverzeichnisebenen enthalten kann (einschließlich des Hauptverzeichnisses).

Dateitypen, Namen und Erweiterungen

Von der CPU-Baugruppe können drei verschiedene Dateitypen verwaltet werden (Lese-/Schreibzugriff).

- **Universell nutzbare Dateien**

Auf diese Dateien kann über Programmiergeräte, FINS-Befehle, Befehle oder Steuerbits im Zusatz-Systembereich zugegriffen werden (Lese-/ Schreibzugriff). Die Dateinamen können vom Anwender frei bestimmt werden.

- **Dateien für die automatische Übertragung beim Start**

Diese Dateien werden von der Speicherkarte beim Einschalten der Spannungsversorgung automatisch an die CPU-Baugruppe übertragen. Der Dateiname lautet AUTOEXEC oder ATEXC□□, wenn Parameterdateien übertragen werden. Der Dateiname lautet REPLACE, wenn keine Parameterdateien übertragen werden. (Nur Baugruppe der CS/CJ-Serie ab Version 2.0)

- **Sicherungsdateien** (Nicht unterstützt von CS1 CPU-Baugruppen der CS-Serie vor EV1).

Diese Dateien werden zwischen der Speicherkarte und der CPU-Baugruppe durch die Sicherungsfunktion übertragen. Die Dateinamen sind festgelegt als BACKUP□□.

Hinweis Verwenden Sie für die Speicherung von Symboltabellen, Kommentar- und Programmindexdateien die Speicherkarte, EM-Dateispeicher oder Kommentarspeicher.

- **CX-Programmer-Systemdateien**

Diese Dateien werden bei der Übertragung mittels CX-Programmer automatisch generiert. Die Dateinamen sind fix.

Universell nutzbare Dateien

In der folgenden Tabelle sind die Dateinamen und Namensweiterungen für universell nutzbare Dateien aufgeführt.

Typ	Name ¹	Erweiterung	Beschreibung	Erläuterung	
Programmdatei	*****	.OBJ	Gesamtes Anwenderprogramm	• Alle zyklischen und Interrupt-Tasks sowie Task-Daten für eine CPU-Baugruppe	
Parameterbereichsdatei	*****	.STD	SPS-Setup, registrierte E/A-Tabelle, Routing-Tabellen, CPU-Bus-Baugruppeneinstellungen ³ usw.	• Enthält alle Grundeinstellungen für eine CPU-Baugruppe • Die in der Datei enthaltenen Parameterdaten müssen nicht nach Typ unterschieden werden.	
Datendatei	*****	.IOM	Spezifizierter Bereich des E/A-Speichers	• Daten innerhalb eines Bereichs vom ersten bis zum letzten Wort in Worteinheiten (16 Bits) • Mögliche Bereiche sind der CIO-, HR-, WR-, AR-, DM- und der EM-Bereich.	Binärformat
		.TXT			TXT-Format ² (nicht getrennt oder durch Tabulator getrennt)
		.CSV			CSV-Format ² (durch Komma getrennt)

Hinweis

1. Dateinamen, die in der Tabelle mit "*****" angegeben sind, bestehen aus bis zu 8 ASCII-Zeichen.
2. Die TXT- und CSV-Dateiformate werden von CS1 CPU-Baugruppen der CS-Serie vor EV1 nicht unterstützt.
3. Die Data-Link-Tabellen stellen ein Beispiel für die CPU-Bus-Baugruppeneinstellungen dar. Informationen zu anderen Setup-Daten finden Sie in den Bedienerhandbüchern der jeweiligen Baugruppen.

Beim Starten der SPS automatisch übertragene Dateien

In der Spalte *Datei* sind die Dateien aufgeführt, die für die automatische Übertragung beim Start auf der Speicherkarte vorhanden sein müssen.

Es gibt zwei verschiedene Arten der automatischen Übertragung von Dateien beim Systemstart: Die Übertragung mit und die Übertragung ohne Parameterbereichsdatei.

Übertragung mit Parameterbereichsdatei

Typ	Name ¹	Erweiterung	Beschreibung	Erläuterung	Datei
Programm-datei	AUTOEXEC	.OBJ	Gesamtes Anwenderprogramm	- Die Datei muss sich nicht auf der Speicherkarte befinden, auch wenn die automatische Übertragung beim Starten spezifiziert wird. - Alle Programme zyklischer Tasks und Interrupt-Tasks sowie Task-Daten für eine CPU-Baugruppe - Eine Übertragung ist nur möglich, wenn auf der Speicherkarte auch eine Parameterbereichsdatei (AUTOEXEC.STD) vorhanden ist.	Erforderlich
Parameterbereichs-datei	AUTOEXEC	.STD	SPS-Setup, registrierte E/A-Tabelle, Routing-Tabellen, CPU-Bus-Baugruppeneinstellungen ³ usw.	Die Datei muss auf der Speicherkarte vorhanden sein, wenn die automatische Übertragung beim Start spezifiziert wird. Enthält alle Grundeinstellungen für eine CPU-Baugruppe Die in der Datei enthaltenen Parameterdaten müssen nicht nach Typ unterschieden werden. Die Grundeinstellungs-Daten werden beim Starten automatisch an bestimmten Stellen in der CPU-Baugruppe gespeichert. Die Parameterbereichsdatei wird nicht übertragen, wenn sich auf der Speicherkarte eine Programmdatei mit Namen REPLACE.OBJ befindet.	Erforderlich
Datendatei	AUTOEXEC	.IOM	E/A-Speicherdaten (Enthält die spezifizierte Anzahl von Datenworten beginnend bei D20000.)	- Speicherung von Daten im DM-Bereich beginnend bei D20000 in einer Datei mit Namen AUTOEXEC.IOM. - Beim Starten werden alle Daten in der Datei an den DM-Bereich, beginnend mit D20000, übertragen. - Die Datei muss sich nicht auf der Speicherkarte befinden, wenn die Funktion zur automatischen Übertragung beim Start verwendet wird.	---
	ATEXEC DM	.IOM	E/A-Speicherdaten ² (Enthält die spezifizierte Anzahl von Datenworten beginnend bei D00000.)	- Speicherung von Daten im DM-Bereich beginnend bei D00000 in einer Datei mit Namen ATEXEC DM.IOM. - Beim Starten werden alle Daten in der Datei an den DM-Bereich, beginnend mit D00000, übertragen. - Die Datei muss sich nicht auf der Speicherkarte befinden, wenn die Funktion zur automatischen Übertragung beim Start verwendet wird. Hinweis Die Daten in dieser Datei sind bei einer Überlappung mit den Daten des DM-Bereichs in der Datei AUTOEXEC.IOM von höherer Priorität.	---
	ATEXECE□	.IOM	Daten des EM-Bereichs (Bank □) ² (Enthält die spezifizierte Anzahl von Datenworten beginnend bei E□_00000.)	- Speicherung der Daten für EM-Bank □ beginnend bei E□_00000 in einer Datei mit Namen ATEXECE□.IOM. Die höchste verwendbare Nummer für die Bank hängt von der eingesetzten CPU-Baugruppe ab. - Beim Starten werden alle Daten in der Datei in die EM-Bank □, beginnend bei E□_00000, übertragen. - Die Datei muss sich nicht auf der Speicherkarte befinden, wenn die Funktion zur automatischen Übertragung beim Start verwendet wird.	---

Übertragung ohne Parameterbereichsdatei

Typ	Name ¹	Erweiterung	Beschreibung	Erläuterung	Datei
Programmdatei	REPLACE Hinweis: Nur CS/CJ-Serie CPU-Baugruppen ab Version 2.0.	.OBJ	Gesamtes Anwenderprogramm	- Der Inhalt ist identisch mit dem von AUTOEXEC.OBJ. - Diese Datei wird beim Start übertragen, auch wenn keine Parameterbereichsdatei (AUTOEXEC.STD) vorhanden ist.	Erforderlich
Parameterbereichsdatei	Nicht erforderlich	---	---	Die Parameterbereichsdatei wird unabhängig von ihrem Dateinamen nicht übertragen.	---
Datendatei	REPLACE Hinweis: Nur CS/CJ-Serie CPU-Baugruppen ab Version 2.0.	.IOM	E/A-Speicherdaten (Enthält die spezifizierte Anzahl von Datenworten beginnend bei D20000.)	- Der Inhalt ist identisch mit dem von AUTOEXEC.IOM. - Diese Datei wird beim Start übertragen, wenn auf der Speicherkarte die Programmdatei REPLACE.OBJ vorhanden ist.	---
	REPLCDM Hinweis: Nur CS/CJ-Serie CPU-Baugruppen ab Version 2.0.	.IOM	E/A-Speicherdaten (Enthält die spezifizierte Anzahl von Datenworten beginnend bei D00000.)	- Der Inhalt ist identisch mit dem von ATEXECDM.IOM. - Diese Datei wird beim Start übertragen, wenn auf der Speicherkarte die Programmdatei REPLACE.OBJ vorhanden ist.	---
	REPLCE□ Hinweis: Nur CS/CJ-Serie CPU-Baugruppen ab Version 2.0.	.IOM	Daten des EM-Bereichs (Bank □) (Enthält die spezifizierte Anzahl von Datenworten beginnend bei E□_00000.)	- Der Inhalt ist identisch mit dem von ATEXECE□.IOM. - Diese Datei wird beim Start übertragen, wenn auf der Speicherkarte die Programmdatei REPLACE.OBJ vorhanden ist.	---

Hinweis

1. Stellen Sie sicher, dass die Dateien, die beim Starten automatisch übertragen werden, die Namen AUTOEXEC oder ATEXEC□□ tragen.
2. Die Dateien ATEXECDM.IOM und ATEXECE□.IOM werden von CS1 CPU-Baugruppen der CS-Serie vor EV1 nicht unterstützt.
3. Die Data-Link-Tabellen stellen ein Beispiel für die CPU-Bus-Baugruppeneinstellungen dar. Informationen zu anderen Setup-Daten finden Sie in den Bedienerhandbüchern der jeweiligen Baugruppen.

Sicherungsdateien (nicht unterstützt von CS1 CPU-Baugruppen der CS-Serie vor EV1.)

Die in der folgenden Tabelle aufgeführten Dateien werden bei der Übertragung von der und auf die Speicherkarte während des Sicherungsvorgangs automatisch erstellt.

Typ	Name ¹	Erweiterung	Beschreibung	Erläuterung
Datendatei	BACKUP	.IOM	Worte im DM-Bereich, die Spezial-E/A-Baugruppen, CPU-Bus-Baugruppen und internen Steckmodulen (nur CS-Serie) zugeordnet sind.	- Enthält DM-Daten von D20000 bis D32767. - Diese Datei muss auf der Speicherkarte vorhanden sein, wenn während der Sicherung Dateien von der Speicherkarte gelesen werden.
	BACKUIO	.IOR	E/A-Speicher-Datenbereiche	- Enthält alle Daten der Datenbereiche CIO, WR, HR und AR sowie Ablaufmerker und Istwerte von Zeitgebern/Zählern.² - Diese Datei muss auf der Speicherkarte vorhanden sein, wenn während der Sicherung Dateien von der Speicherkarte gelesen werden.
	BACKUPDM	.IOM	Universell nutzbarer DM-Bereich	- Enthält DM-Daten von D00000 bis D19999. - Diese Datei muss auf der Speicherkarte vorhanden sein, wenn während der Sicherung Dateien von der Speicherkarte gelesen werden.
	BACKUPE□	.IOM	Universell nutzbarer EM-Bereich	Enthält alle EM-Daten der EM-Bank □ mit einer Adressen im Bereich von E□_00000 bis E□_32767. (Die höchst mögliche Banknummer hängt von der eingesetzten CPU-Baugruppe ab.) Diese Datei muss auf der Speicherkarte vorhanden sein, wenn während der Sicherung Dateien von der Speicherkarte gelesen werden. Wenn Daten auf der Speicherkarte gesichert werden, werden alle Daten der einzelnen EM-Bänke automatisch in einzelne Dateien geschrieben.
Programmdatei	BACKUP	.OBJ	Gesamtes Anwenderprogramm	- Enthält alle Programme zyklischer Tasks und Interrupt-Tasks sowie Task-Daten für eine CPU-Baugruppe. - Diese Datei muss auf der Speicherkarte vorhanden sein, wenn während der Sicherung Dateien von der Speicherkarte gelesen werden.
Parameterdatei		.STD	SPS-Setup, registrierte E/A-Tabelle, Routing-Tabellen, CPU-Bus-Baugruppeneinstellungen ³ usw.	- Enthält alle Grundeinstellungen für eine CPU-Baugruppe. - Die in der Datei enthaltenen Parameterdaten müssen nicht nach Typ unterschieden werden. - Diese Datei muss auf der Speicherkarte vorhanden sein, wenn während der Sicherung Dateien von der Speicherkarte gelesen werden.
Baugruppen-/Modul-Sicherungsdateien (nur CS1-H, CJ1-H und CJ1M CPU-Baugruppen)	BACKUP□□ (wobei □□ für die Baugruppenadresse der/des zu sichernden Baugruppe/Moduls steht.)	.PRM	Daten einer bestimmten Baugruppe bzw. eines bestimmten Moduls	Steuerungs-Sicherungsdaten einer Baugruppe oder eines Moduls. Weitere Informationen finden Sie unter 5-2-6 <i>Easy-Backup-Funktion</i>.
Symboltabelledateien (siehe Hinweis 1)	BKUPSYM	.SYM	Globale und lokale Symboltabelle wie automatische SPS-Speicherzuweisung	Umfasst die folgenden Daten in der globalen bzw. lokalen Symboltabelle von CX-Programmer: Variablen, Adressen, Datentypen, E/A-Kommentare Umfasst die in den Einstellungen der automatischen SPS-Adresszuweisung von CX-Programmer eingestellten Daten.
Kommentar-dateien (siehe Hinweis 1)	BKUPCMT	.CMT	Netzwerkkommentare und Kommentare	CX-Programmer Netzwerkkommentare und Kommentare
Programmin-dexdateien (siehe Hinweis 1)	BKUPPRG	.IDX	CX-Programmer Sektionsnamen, Sektionskommentare und Programmkommentare	CX-Programmer Sektionsbegrenzerdaten (Die Begrenzerposition hängt jedoch vom Sektionsbegrenzer innerhalb des Programms ab).

- Hinweis**
1. Die folgenden Sicherungsdateien können nur bei Verwendung von CS/CJ-Serie CPU-Baugruppen ab Baugruppenversion 3.0 erstellt werden.
Symboltabellen-, Kommentar- und Programmindexdateien
Diese Dateien werden automatisch auf der Speicherkarte, im EM-Dateispeicher oder im Kommentarspeicher erzeugt.
 2. Die Data-Link-Tabellen stellen ein Beispiel für die CPU-Bus-Baugruppen-einstellungen dar. Informationen zu anderen Einstellungs-Daten finden Sie in den Bedienerhandbüchern der jeweiligen Baugruppen.

CX-Programmer-Systemdateien

Diese Dateien werden beim Herunterladen mittels CX-Programmer ab Ver. 5.0 automatisch generiert. Die Dateinamen sind fix.

Bei Verwendung von CX-Programmer ab Ver. 5.0 mit CPU-Baugruppen ab Ver. 3.0 können beim Übertragen von Projekten die folgenden Optionen als Übertragungsziel ausgewählt werden:

- Speicherkarte
- EM-Dateispeicher
- Kommentarspeicher (im internen Flash-Speicher der CPU)

Hinweis Bei CX-Programmer bis Ver. 4.0 können diese Dateien nicht im Kommentarspeicher gespeichert werden, auch wenn eine CPU-Gruppe mit der Baugruppenversion 3.0 oder neuer verwendet wird.

Typ	Name	Namenserweiterung	Beschreibung	Erläuterung	
Symboltabelledateien	SYMBOLS	.SYM	Globale und lokale Symboltabellen	Diese Dateien werden beim Herunterladen von Daten mittels CX-Programmer automatisch generiert.	Umfasst die folgenden Daten in der globalen bzw. lokalen Symboltabelle von CX-Programmer: Variablen, Adressen, Datentypen, E/A-Kommentare. Umfasst die in den Einstellungen der automatischen SPS-Adresszuordnung von CX-Programmer eingestellten Daten.
Kommentardateien	COMMENTS	.CMT	Netzwerkcommentare und Kommentare (Anmerkungen)		CX-Programmer Netzwerkcommentare und Kommentare
Programmindexdateien	PROGRAMS	.IDX	Sektionsnamen, Sektionskommentare		CX-Programmer Sektionsbegrenzerdaten (Die Begrenzerposition hängt jedoch vom Sektionsbegrenzer innerhalb des Programms ab). Hinweis: nur CX-Programmer ab Version 2.0.

Hinweis Mit dem CX-Programmer ab Version 1.2 können die in der obigen Tabelle aufgeführten Symboltabelle- und Kommentardateien online auf Speicherkarten oder auf EM-Dateispeicher übertragen werden.

Verzeichnisse

Mit den SPS-Systemen der CS/CJ-Serie kann auf Dateien in Unterverzeichnissen zugegriffen werden. Programmierkonsolen bieten jedoch nur Zugriff auf Dateien im Hauptverzeichnis. Die maximale Länge für einen Verzeichnispfad beträgt 65 Zeichen. Achten Sie darauf, dass die maximal zulässige Zei-

chenanzahl beim Erstellen von Unterverzeichnissen auf der Speicherkarte mit einem Programm wie Windows nicht überschritten wird.

Dateigrößen

Mit Hilfe der Gleichungen in der folgenden Tabelle kann die Dateigröße in Byte errechnet werden.

Dateityp	Dateigröße
Datendateien (.IOM)	(Anzahl der Worte $\times$ 2) + 48 Byte Beispiel: Gesamter DM-Bereich (D00000 bis D32767) (32.768 Worte $\times$ 2) + 48 = 65.584 Byte
Datendateien (.TXT oder .CSV)	Die Dateigröße hängt von der Anzahl der verwendeten Trennzeichen und Zeilenumbrüche ab. Der Trennzeichen-code belegt ein Byte und der Wagenrücklauf (CR) zwei Byte. Beispiel 1: Nicht getrennte Worte, kein Wagenrücklauf 123456789ABCDEF012345678 belegt 24 Byte. Beispiel 2: Getrennte Worte, Wagenrücklauf jedes zweite Feld 1234,5678.␣ 9ABC,DEF0.␣ 1234,5678.␣ belegt 33 Byte. Beispiel 3: Getrennte Doppelworte, Wagenrücklauf jedes zweite Feld 56781234,DEF01234.␣ 56781234.␣ belegt 29 Byte.
Programmdateien (.OBJ)	(Anzahl der verwendeten Steps $\times$ 4) + 48 Byte (siehe Hinweis)
Parameterdateien (.STD)	16.048 Byte

Hinweis

Berechnen Sie die Step-Anzahl in der Programmdatei, indem Sie die verfügbaren UM-Steps von der Gesamtzahl der UM-Steps abziehen. Diese Werte werden im „Querverweis-Report“ von CX-Programmer angegeben. Weitere Informationen hierzu finden Sie im *CX-Programmer Bedienerhandbuch*.

Datendateien

Universell nutzbare Dateien

1,2,3...

1. Universell nutzbare Dateien haben die Dateinamenerweiterungen IOM, TXT oder CSV. (TXT- und CSV-Dateien werden von CS1 CPU-Baugruppen der CS-Serie vor EV1 nicht unterstützt.)

Erweiterung	Datenformat	Inhalt	Worte/Feld
.IOM	Binär	Datenformat der CS/CJ-Serie	---
.TXT (siehe Hinweise)	Nicht getrennte Worte	ASCII- Format Dieses Format wird erstellt, indem Wort-Felder des E/A-Speichers (4-stellige Hexadezimalwerte) in ASCII konvertiert und die Felder ohne Trennzeichen gespeichert werden. Datensätze können durch Wagenrückläufe (CR) getrennt werden.	1 Wort
	Nicht getrennte Doppelworte	Dieses Format wird erstellt, indem Doppelwort-Felder des E/A-Speichers (8-stellige Hexadezimalwerte) in ASCII konvertiert und die Felder ohne Trennzeichen gespeichert werden. Datensätze können durch Wagenrückläufe (CR) getrennt werden.	2 Worte
	Durch Tabulator getrennte Worte	Dieses Format wird erstellt, indem Wort-Felder des E/A-Speichers (4-stellige Hexadezimalwerte) in ASCII konvertiert werden und die Felder durch Tabulatoren voneinander getrennt werden. Datensätze können durch Wagenrückläufe (CR) getrennt werden.	1 Wort
	Durch Tabulator getrennte Doppelworte	Dieses Format wird erstellt, indem Doppelwort-Felder des E/A-Speichers (8-stellige Hexadezimalwerte) in ASCII konvertiert werden und die Felder durch Tabulatoren voneinander getrennt werden. Datensätze können durch Wagenrückläufe (CR) getrennt werden.	2 Worte
.CSV (siehe Hinweise)	Durch Komma getrennte Worte	Dieses Format wird erstellt, indem Wort-Felder des E/A-Speichers (4-stellige Hexadezimalwerte) in ASCII konvertiert werden und die Felder durch Kommata voneinander getrennt werden. Datensätze können durch Wagenrückläufe (CR) getrennt werden.	1 Wort
	Durch Komma getrennte Doppelworte	Dieses Format wird erstellt, indem Doppelwort-Felder des E/A-Speichers (8-stellige Hexadezimalwerte) in ASCII konvertiert werden und die Felder durch Kommata voneinander getrennt werden. Datensätze können durch Wagenrückläufe (CR) getrennt werden.	2 Worte

- Hinweis a)** Lesen und Schreiben von TXT- und CSV-Datendateien:
TXT- und CSV-Datendateien können nur mit den Befehlen FREAD(700) und FWRITE(701) gelesen und geschrieben werden.
- b)** Hinweis zu Zeichen:
Daten können nicht ordnungsgemäß in den E/A-Speicher geschrieben werden, wenn die TXT- oder CSV-Datei andere als hexadezimale Zeichen enthält (0 bis 9, A bis F oder a bis f).
- c)** Hinweise zur Feldgröße:
Wenn Worte verwendet werden, können die Daten nur korrekt in den E/A-Speicher geschrieben werden, wenn die TXT- oder CSV-Datei Felder enthält, die 4-stellige Hexadezimalwerte darstellen. Ebenso können die Daten bei Verwendung von Doppelworten nur korrekt geschrieben werden, wenn die Datei Felder enthält, die 8-stellige Hexadezimalwerte darstellen.
- d)** Speicherreihenfolge:
Wenn Worte verwendet werden, werden die E/A-Speicherdaten in ASCII konvertiert und in ein Wort enthaltende Felder, sortiert von der niedrigsten zur höchsten E/A-Speicheradresse, gespeichert. Wenn Doppelworte verwendet werden, werden die E/A-Speicherdaten in ASCII konvertiert und in zwei Worte enthaltende Felder, sortiert von der niedrigsten zur höchsten E/A-Speicheradresse, gespeichert. (Innerhalb der zwei Worte enthaltenden Felder wird zuerst (links) das Wort der höheren Adresse und dann (rechts) das Wort der niedrigeren Adresse gespeichert.)
- e)** Trennzeichen:
Wenn keine Trennzeichen vorhanden sind, werden die Felder fortlaufend aneinandergereiht und dann gespeichert. Bei einer Trennung durch Komma werden vor dem Speichern Kommata zwischen den

Feldern eingefügt. Bei einer Trennung durch Tabulatoren werden vor dem Speichern Tabulatorzeichen zwischen den Feldern eingefügt. Wenn im Befehl FREAD(700) Trennzeichen (Kommas oder Tabulatoren) spezifiziert werden, werden die Daten als durch Trennzeichen (Kommas oder Tabulatoren) getrennte Ein-Wort-Einheiten gelesen.

f) Wagenrückläufe:

Die Daten werden fortlaufend aneinandergereiht, wenn keine Wagenrückläufe verwendet werden.

Bei der Verwendung von Wagenrückläufen wird nach einer spezifizierten Anzahl von Feldern ein Wagenrücklaufzeichen eingefügt. In den Befehlen FREAD(700)/FWRITE(701) kann kein Offset vom Anfang der Datei (Startwort zum Lesen oder Startwort zum Schreiben) spezifiziert werden, wenn in der Datei Wagenrücklaufzeichen verwendet werden.

g) Anzahl der Felder:

Die Gesamtmenge der Daten einer Datei ist von der Anzahl der Felder (Anzahl der Schreibeinträge) abhängig, die im Befehl FWRITE(701) spezifiziert sind, und von der Anzahl von Worten pro Feld. Bei der Verwendung von Worten als Einheiten wird 1 Wort/Feld gespeichert, bei der Verwendung von Doppelworten 2 Worte pro Feld.

2. Datendateien enthalten keine Informationen über die Art der gespeicherten Daten, d. h., welcher Speicherbereich gespeichert ist. Achten Sie darauf, dass, wie in den folgenden Beispielen gezeigt, Dateinamen vergeben werden, die den Inhalt angeben. Auf diese Weise lässt sich die Dateiverwaltung vereinfachen.

Beispiele: D00100.IOM, CIO0020.IOM

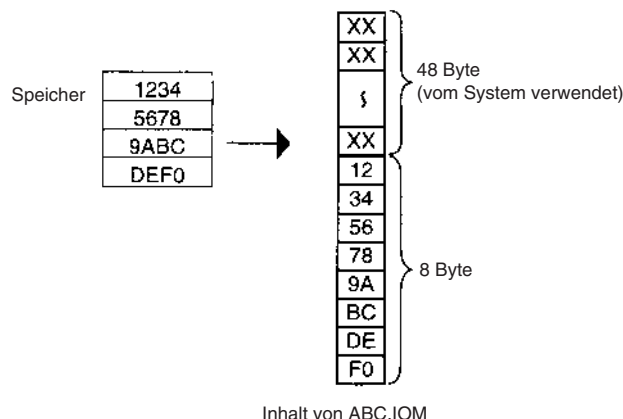
Daten vom Anfang der Datei werden beginnend mit der im E/A-Speicher spezifizierten Adresse geschrieben, auch wenn die ursprünglich in die Datendatei (IOM, TXT oder CSV) geschriebenen Daten nicht aus demselben Bereich stammen. Wenn beispielsweise CIO-Daten in einer Datei in den DM-Bereich eines Programmiergeräts geschrieben werden, werden die Daten ohne Hinweis darauf, dass der Bereich unterschiedlich ist, in den DM-Bereich der CPU-Baugruppe eingelesen.

Hinweis

Datendateien im TXT- und CSV-Format enthalten hexadezimale Daten (0 bis 9, A bis F). Dies ermöglicht den Austausch numerischer E/A-Speicherdaten mit Tabellenkalkulationsprogrammen.

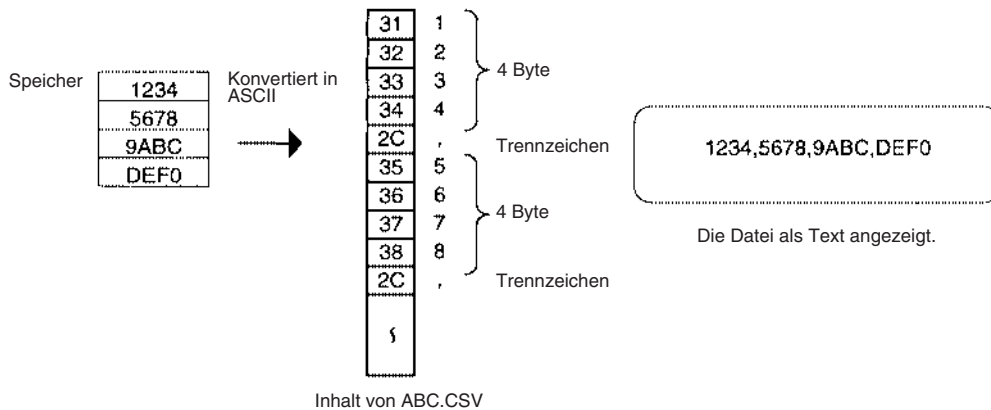
IOM-Datendateistruktur

Folgende Abbildung zeigt die Binärdatenstruktur einer Datendatei (ABC.IOM), die vier Worte aus dem E/A-Speicher enthält: 1234 Hex, 5678 Hex, 9ABC Hex und DEF0 Hex. Beim normalen Betrieb muss das Datenformat vom Anwender jedoch nicht berücksichtigt werden.



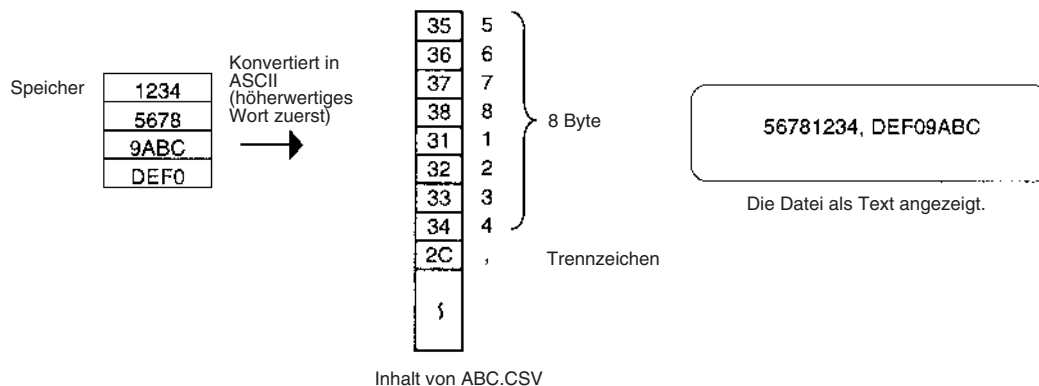
**CSV/TXT-
Datendateistruktur
(Einzelwort)**

Folgende Abbildung zeigt die Datenstruktur einer CSV-Datendatei (ABC.CSV) mit Einzelwort-Feldern, die vier Worte aus dem E/A-Speicher enthält: 1234 Hex, 5678 Hex, 9ABC Hex und DEF0 Hex. Die Struktur einer TXT-Datei mit Einzelwort-Feldern ist identisch.



CSV/TXT-Datendateistruktur (Doppelwort)

Folgende Abbildung zeigt die Datenstruktur einer CSV-Datendatei (ABC.CSV) mit Doppelwort-Feldern, die vier Worte aus dem E/A-Speicher enthält: 1234 Hex, 5678 Hex, 9ABC Hex und DEF0 Hex. Die Struktur einer TXT-Datei mit Doppelwort-Feldern ist identisch.



Erstellen von Datendateien mit Tabellenkalkulationsprogrammen

Gehen Sie wie folgt vor, um TXT- und CSV-Datendateien mit Hilfe eines Tabellenkalkulationsprogramms wie Microsoft Excel zu erstellen.

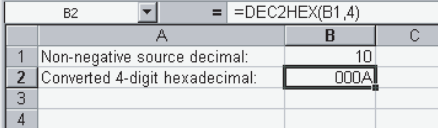
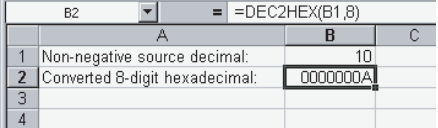
- Setzen Sie den Zelleninhalt auf Text.
- Geben Sie bei Verwendung von Einzelwort-Feldern 4 Zeichen in jedes Feld ein und 8 Zeichen bei Doppelwort-Feldern. Geben Sie bei Einzelwort-Feldern beispielsweise „000A“ ein und nicht nur „A“.
- Achten Sie darauf, nur Hexadezimalzeichen (0 bis 9, A bis F oder a bis f) in die Zellen einzugeben. Andere Zeichen oder Codes können nicht verwendet werden.

Wenn Sie Hexadezimalstellen im E/A-Speicher speichern möchten, ist es hilfreich, die Dezimaleingaben in der Kalkulationstabelle in Hexadezimalzahlen zu konvertieren. Gehen Sie zur Konvertierung in Hexadezimalzahlen wie folgt vor.

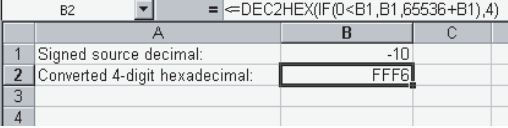
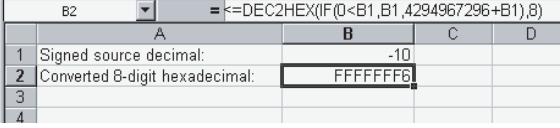
1,2,3...

1. Wählen Sie im Menü „Extras“ **Add-Ins-Manager...** aus.
2. Wählen Sie im Menü „Add-Ins“ **Analyse-Funktionen - VBA** aus.
3. Wählen Sie im Menü „Einfügen“ für die Zelle, für die die Funktion verwendet werden soll, die Option **Funktion...** aus.
4. Wählen Sie im Feld „Funktionskategorie“ unter „Engineering“ **Dec2Hex (number; places)** aus.
5. Geben Sie bei der Konvertierung in eine vierstellige Hexadezimalzahl als Variable "Number" Folgendes ein: WENN(0<=Zelle;Zelle;65535+Zelle)
Geben Sie bei der Konvertierung in eine achtstellige Hexadezimalzahl als Variable "Number" Folgendes ein: WENN(0<=Zelle;Zelle;4294967296+Zelle)

• **Beispiel 1:** Eingabe nicht negativer Dezimalwerte

Beschreibung	Konvertieren von dezimal (ohne Vorzeichen) zu 4-stellig hexadezimal	Konvertieren von dezimal (ohne Vorzeichen) zu 8-stellig hexadezimal
Verwendete Funktion	DEC2HEX(Zelle;4)	DEC2HEX(Zelle;8)
Beispiel	10 in dezimal eingegeben und zu 000A als 4-stellige Hexadezimalzahl konvertieren. 	10 in dezimal eingegeben und zu 0000000A als 8-stellige Hexadezimalzahl konvertieren. 

• **Beispiel 2:** Eingabe von Dezimalwerten mit Vorzeichen

Beschreibung	Konvertieren von dezimal (mit Vorzeichen) zu 4-stellig hexadezimal	Konvertieren von dezimal (mit Vorzeichen) zu 8-stellig hexadezimal
Verwendete Funktion	DEC2HEX(WENN(0<=Zelle;Zelle;65536+Zelle);4)	DEC2HEX(WENN(0<=Zelle;Zelle;4294967296+Zelle);8)
Beispiel	-10 in dezimal eingegeben und zu FFF6 als 4-stellige Hexadezimalzahl konvertieren. 	-10 in dezimal eingegeben und zu FFFFFFF6 als 8-stellige Hexadezimalzahl konvertieren. 

Beim Starten der SPS automatisch übertragene Datendateien

Es gibt drei Arten von Dateien, die beim Starten der SPS automatisch übertragen werden, wenn die Funktion zum automatischen Übertragen beim Start verwendet wird.

- **AUTOEXEC.IOM** oder **REPLACE.IOM**: Spezial-E/A-Baugruppen und internen Steckmodulen zugeordnete DM-Worte.
Der Inhalt dieser Datei wird beim Einschalten der Spannungsversorgung, beginnend bei D20000, in den DM-Bereich übertragen.
- **ATEXEC.DM.IOM** oder **REPLCDM.IOM**: Universell nutzbare DM-Worte
Der Inhalt dieser Datei wird beim Einschalten der Spannungsversorgung, beginnend bei D00000, in den DM-Bereich übertragen.
- **ATEXECE□.IOM** oder **REPLCE□.IOM**: Universell nutzbare EM-Worte
Der Inhalt dieser Datei wird beim Einschalten der Spannungsversorgung, beginnend bei E□_00000, in den EM-Bereich übertragen.

Geben Sie beim Erstellen der oben aufgeführten Datendateien immer die oben angegebene erste Adresse (D20000, D00000 oder E□_00000) an, und achten Sie darauf, dass die Dateigröße die Kapazität des angegebenen Datenbereichs nicht überschreitet.

Alle Daten der Dateien werden immer von der ersten angegebenen Adresse (D20000, D00000 oder E□_00000) an übertragen.

Hinweis

1. Geben Sie beim Erstellen der Dateien **AUTOEXEC.IOM**, **ATEXEC.DM.IOM** und **ATEXECE□.IOM** oder der Dateien **REPLACE.IOM**, **REPLCDM.IOM** oder **REPLCE□.IOM** mit einem Programmiergeräts (Programmierkonsole oder CX-Programmer) immer die richtige erste Adresse an (D20000, D00000 oder E□_00000), und achten Sie darauf, dass die Dateigröße die Kapazität des DM-Bereichs bzw. der spezifizierten EM-Bank nicht überschreitet. Der Inhalt der Datei wird immer von der ordnungsgemäßen ersten Adresse (D20000, D00000 oder E□_00000) an übertragen, auch wenn ein anderes Anfangswort spezifiziert wird, was zu einem ungewollten Überschreiben des Inhalts eines Teils des DM-Bereichs bzw. der EM-Bank führen kann. Sollte

überdies die Kapazität des DM-Bereichs oder der EM-Bank überschritten werden (was möglich ist, wenn in CX-Programmer Einstellungen vorgenommen werden), werden die verbleibenden Daten in die EM-Bank 0 geschrieben, wenn der DM-Bereich überschritten wird oder in die folgende EM-Bank, wenn eine EM-Bank überschritten wird.

2. Wenn CX-Programmer verwendet wird, kann eine Datendatei spezifiziert werden, die über den maximalen DM-Adressbereich (D32767) oder den maximalen EM-Adressbereich von E□_32767 hinausgeht. Wenn die Datei AUTOEXEC.IOM die DM-Bereichsgrenze überschreitet, werden alle noch zu übertragenden Daten in den EM-Bereich, beginnend bei E0_00000 und weiter in der Reihenfolge der Speicheradresse und Banken bis zur letzten Bank, geschrieben. Es ist daher möglich, beim Starten der SPS Daten automatisch in die DM- und auch in die EM-Bereiche zu übertragen. Wenn die Datei ATEXECE□.IOM größer ist als eine EM-Bank, werden die noch zu übertragenden Daten in gleicher Weise in nachfolgende EM-Bänke geschrieben.
3. Die System-Setups für Spezial-E/A-Baugruppen, CPU-Bus-Baugruppen und das interne Modul (nur CS-Serie) können durch Verwendung verschiedener AUTOEXEC.IOM-Dateien geändert werden, die verschiedene Einstellungen für den Bereich der Spezial-E/A-Baugruppen (D20000 bis D29599), den Bereich der CPU-Bus-Baugruppe (D30000 bis D31599) und den Bereich des internen Moduls (nur CS-Serie, D32000 bis D32099) enthalten. Speicherkarten können dann zum Erstellen von Bibliotheken von System-Setup-Daten für Spezial-E/A-Baugruppen, CPU-Bus-Baugruppen und interne Module (nur CS-Serie) für verschiedene Systeme oder Geräte verwendet werden.

Sicherungsdatendateien

Von der Sicherungsfunktion werden, wie im Folgenden beschrieben, vier Arten von Datendateien erstellt.

Schalten Sie zum Sichern von Daten Schalter 7 des DIP-Schalterblocks der CPU-Baugruppe auf ON und Schalter 8 auf OFF, setzen Sie die Speicherkarte ein, drücken Sie die Speicherkartentaste und halten Sie diese drei Sekunden lang gedrückt. Die vier Sicherungsdateien (BACKUP.IOM, BACKUPIO.IOR, BACKUPDM.IOM und BACKUPE□.IOM) werden automatisch erstellt und auf die Speicherkarte geschrieben.

Die vier Sicherungsdateien werden ausschließlich von der Sicherungsfunktion verwendet, obwohl drei der Dateien (BACKUP.IOM, BACKUPDM.IOM und BACKUPE□.IOM) auch mit Hilfe eines Programmiergeräts erstellt werden können. (BACKUPIO.IOR kann nicht mit einem Programmiergerät erstellt werden.)

5-1-4 Beschreibung von Dateioperationen

In der folgenden Tabelle sind die sechs Methoden zum Lesen und Schreiben von Dateien zusammengefasst.

Lesen: Überträgt Dateien vom Dateispeicher in die CPU-Baugruppe.

Schreiben: Überträgt Dateien von der CPU-Baugruppe in den Dateispeicher.

OK: Möglich; ---: Nicht möglich

Vorgehensweise		Medium	Dateiname	Beschreibung	Gesamtes Programm	Datenbereichsdaten (siehe Hinweis 3)	Parameterbereichsdaten	Symboltabellen, Kommentardateien, Programmindexdateien (siehe Hinweis 6)
Programmiergerät	CX-Programmer oder Programmierkonsole	Speicherkarte EM-Dateispeicher	Alle gültigen Dateinamen	Lesen	OK	OK	OK	---
				Schreiben	OK	OK	OK	---
				Andere Operationen (siehe Hinweis 2)	OK	OK	OK	---

Vorgehensweise		Medium	Dateiname	Beschreibung	Gesamtes Programm	Datenbereichsdaten (siehe Hinweis 3)	Parameterbereichsdaten	Symboltabellen, Kommentardateien, Programmindexdateien (siehe Hinweis 6)
Programmiergerät	nur CX-Programmer	Speicherkarte EM-Dateispeicher, Kommentarspeicher	CX-Programmer-Systemdateien (fester Dateiname)	Lesen	---	---	---	OK
				Schreiben	---	---	---	OK (siehe Hinweis 6)
				Andere Operationen (siehe Hinweis 2)	---	---	---	---
FINS-Befehl (siehe Hinweis 1)		Speicherkarte EM-Dateispeicher	Alle gültigen Dateinamen	Lesen	OK	OK	OK	---
				Schreiben	OK	OK	OK	---
				Andere Operationen (siehe Hinweis 2)	OK (siehe Hinweis 4)	OK	OK	---
Programmierbefehle FREAD(700) und FWRIT(701)		Speicherkarte EM-Dateispeicher	Alle gültigen Dateinamen	Daten aus einer Datei lesen	---	OK	---	---
				Daten in eine Datei schreiben	---	OK	---	---
Durch Operation mit Steuerbit im Zusatz-Systembereich wird das gesamte Programm während des laufenden Betriebs ausgetauscht. (Nicht unterstützt von CS1 CPU-Baugruppen der CS-Serie vor EV1)		Speicherkarte	Alle gültigen Dateinamen	Lesen	OK	---	---	---
Automatische Übertragung beim Start		Speicherkarte	Automatische Übertragung der Startdateien (AUTOEXEC, ATEXEC□□ oder REPLACE) (siehe Hinweis 5)	Lesen	OK	OK	OK	---
				Schreiben	---	---	---	---
Backup-Vorgang (Nicht unterstützt von CS1 CPU-Baugruppen der CS-Serie vor EV1)		Speicherkarte	Easy-Backup-Dateien (BACKUP□□)	Lesen	OK	OK	OK	OK
				Schreiben	OK	OK	OK	OK (siehe Hinweis 6)

- Hinweis**
1. FINS-Befehle für Dateispeichervorgänge können von Host-Computern mit einer Host-Link-Verbindung gesendet werden, von einer anderen mit einem Netzwerk verbunden SPS (durch CMND(490)) oder dem Programm der lokalen SPS (durch CMND(490)). (Bei CS1 CPU-Baugruppen der CS-Serie vor EV1 können Dateispeichervorgänge unter Verwendung von CMND(490) nicht in derselben CPU-Baugruppe durchgeführt werden, bei der die Dateispeichervorgänge ausgeführt werden.)
 2. Andere Vorgänge: Dateispeicher formatieren, Dateidaten lesen, Dateidaten schreiben, Dateinamen ändern, Dateispeicherdaten lesen, Datei löschen, Datei kopieren und Unterverzeichnis erstellen.
 3. Datendateien im TXT- oder CSV-Format können nur mit Hilfe der Befehle FREAD(700) und FWRITE(701) gelesen und geschrieben werden. Das Lesen und Schreiben ist mit einem Programmiergerät nicht möglich.

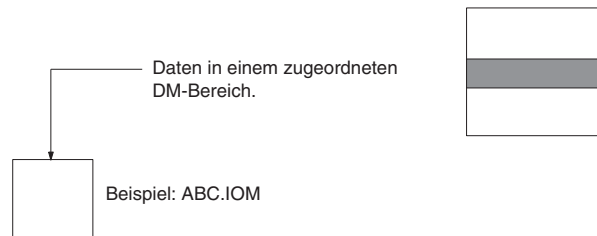
4. CX-Programmer ab Version V1.2 kann für die Übertragung von Programmdateien (.OBJ) zwischen dem RAM-Speicher des Computers und einem Speichergerät verwendet werden.
5. Bei einer CPU-Baugruppe der CS/CJ-Serie in der Version 2.0 können Dateien beim Starten der SPS automatisch an die CPU-Baugruppe übertragen werden, ohne dass eine Parameterdatei auf der Speicherkarte vorhanden sein muss. Dies wird durch Ändern des Programmdateinamen in REPLACE.OBJ erreicht. Datendateien können auch zusammen mit REPLACE.OBJ übertragen werden, wenn folgende Dateinamen verwendet werden: REPLACE.IOM, REPLCDM.IOM und REPLCE□.IOM.
6. Beim Übertragen von Projekten von CX-Programmer ab Version 5.0 in eine CS/CJ-Serie CPU-Baugruppe ab Version 3.0 können die Symboltabellen sowie die Kommentar- und Programmindexdateien im Kommentarspeicher im internen Flash-Speicher der CPU-Baugruppe gespeichert werden (wenn keine Speicherkarte und kein EM-Dateispeicher oder keine Bereiche verfügbar sind).
Es werden automatisch Sicherungsdateien der Symboltabellen-, Kommentar- und Programmindexdateien in der Speicherkarte, dem EM-Dateispeicher oder dem Kommentarspeicher erstellt und auf der Speicherkarte gespeichert.

5-1-5 Anwendungen

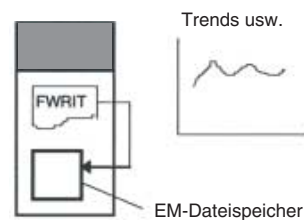
Der Datenspeicher kann für folgende Anwendungen verwendet werden.

Datendateien

Bei dieser Anwendung werden DM-Bereichs-Dateneinstellungen (für Spezial-E/A-Baugruppen, CPU-Bus-Baugruppen und interne Steckmodule (nur CS-Serie)) auf der Speicherkarte gespeichert. Trägt die Datendatei den Namen AUTOEXEC.IOM, werden die in der Datei gespeicherten Einstellungen beim Einschalten der Spannungsversorgung automatisch übertragen.



Bei dieser Anwendung werden Betriebsdaten (Trend-, Qualitätskontroll- und andere Daten), die während der Programmausführung erzeugt wurden, mit dem Befehl WRITE DATA FILE (Datei schreiben) (FWRIT(701)) im EM-Dateispeicher gespeichert.

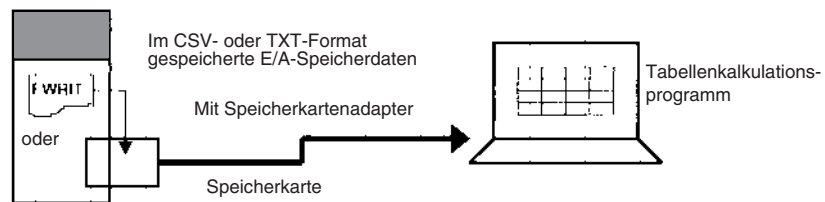


Hinweis

Daten, auf die häufig zugegriffen wird, wie Trenddaten, sollten vorzugsweise im EM-Dateispeicher und nicht auf einer Speicherkarte gespeichert werden.

ASCII-Datendateien (.TXT und .CSV)

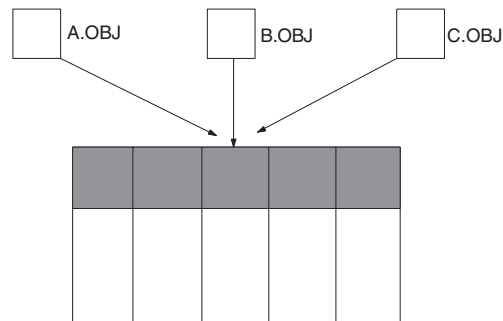
Produktionsdaten, die auf einer der Speicherkarte im TXT- oder CSV-Format gespeichert wurde, können durch Verwendung eines Speicherkartenadapters auf einen Computer übertragen und mit einem Tabellenkalkulationsprogramm bearbeitet werden (nicht unterstützt von CS1 CPU-Baugruppen der CS-Serie vor EV1).



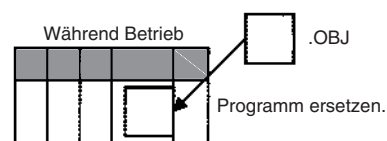
Umgekehrt können Daten, wie die Einstellungen der Spezial-E/A-Baugruppe, mit einem Tabellenkalkulationsprogramm im TXT- oder CSV-Format erstellt werden, auf eine Speicherkarte gespeichert werden und mit FREAD(700) in die CPU-Baugruppe geschrieben werden. (Nicht unterstützt von CS1 CPU-Baugruppen der CS-Serie vor EV1.)

Programmdateien (.OBJ)

Bei dieser Anwendung werden Programme, mit denen verschiedene Prozesse gesteuert werden, auf einzelnen Speicherkarten gespeichert. Die gesamte SPS-Konfiguration (Programm, SPS-Setup usw.) kann geändert werden, indem eine andere Speicherkarte und die Funktion für das automatische Übertragen beim Start verwendet werden.

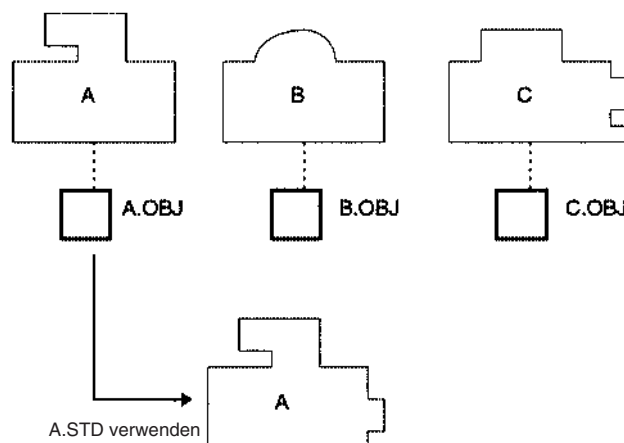


Das gesamte Programm kann während des Betriebs mit Hilfe von Steuerbits im Zusatz-Systembereich vom Programm selbst (ohne Programmiergerät) ausgetauscht werden (nicht unterstützt von CS1 CPU-Baugruppen der CS-Serie vor EV1).



Parameterbereichsdateien (.STD)

Bei dieser Anwendung werden das SPS-Setup, Routing-Tabellen, E/A-Tabellen und andere Daten für bestimmte Geräte oder Maschinen auf Speicherkarten gespeichert. Die Daten können durch Austausch der Speicherkarte einfach auf ein anderes Gerät übertragen werden.



Sicherungsdateien

Mit Hilfe der Sicherungsfunktion können alle Daten der CPU-Baugruppe (gesamter E/A-Speicher, Programm und Parameterbereich) ohne Programmiergerät auf der Speicherkarte gespeichert werden. Wenn mit den Daten der CPU-Baugruppe ein Problem auftritt, können die gesicherten Daten sofort wiederhergestellt werden. (Nicht unterstützt von CS1 CPU-Baugruppen der CS-Serie vor EV1.)

Symboltabellendateien

Mit CX-Programmer können Programmsymbole und E/A-Kommentare in Symboltabellendateien, genannt SYMBOLS.SYM, auf Speicherkarten oder im EM-Dateispeicher gespeichert werden.

Beim Übertragen von Projekten von CX-Programmer ab Version 5.0 in eine CS/CJ-Serie CPU-Baugruppe ab Version 3.0 können die Symboltabellendateien im Kommentarspeicher im internen Flash-Speicher der CPU-Baugruppe gespeichert werden, wenn keine Speicherkarte, kein EM-Dateispeicher oder keine freien Speicherbereiche verfügbar sind.

Kommentardateien

Mit CX-Programmer können Programm-Kommentare in Kommentardateien, genannt COMMENTS.CMT, auf Speicherkarten oder im EM-Dateispeicher gespeichert werden.

Beim Übertragen von Projekten in eine CS/CJ-Serie CPU-Baugruppe ab Version 3.0 können die Kommentardateien im Kommentarspeicher im internen Flash-Speicher der CPU-Baugruppe gespeichert werden, unabhängig davon, ob eine Speicherkarte oder EM-Dateispeicher verfügbar ist.

5-2 Arbeiten mit Dateien

Mit den folgenden Vorgehensweisen können Dateien gelesen, geschrieben und anderweitig verwendet werden.

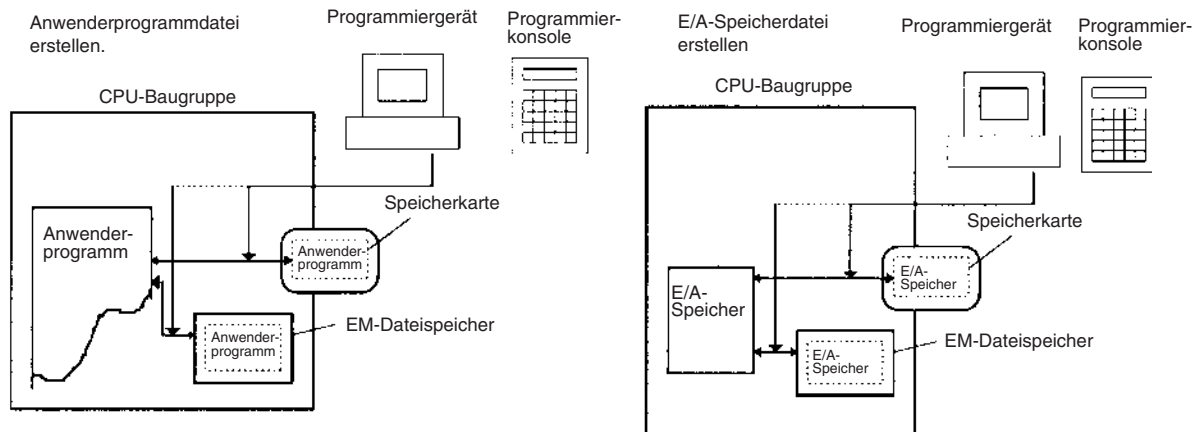
- Programmiergeräte
- FINS-Befehle
- Befehle FREAD(700), FWRT(701) und CMND(490) im Anwenderprogramm (CMND(490) wird von CS1 CPU-Baugruppen der CS-Serie vor EV1 nicht unterstützt.)
- Austausch des gesamten Programms mit Hilfe von Steuerbits im Zusatz-Systembereich (nicht unterstützt von CS1 CPU-Baugruppen der CS-Serie vor EV1)
- Automatische Übertragung beim Start
- Sicherungsfunktion (nicht unterstützt von CS1 CPU-Baugruppen der CS-Serie vor EV1)

5-2-1 Programmiergeräte (einschließlich Programmierkonsolen)

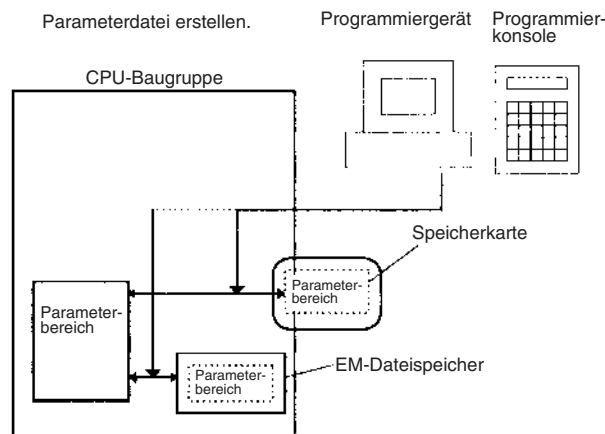
Folgende Operationen stehen mit Hilfe von Programmiergeräten zur Verfügung.

Vorgang		CX-Programmer	Programmier-konsole
Lesen von Dateien (Übertragung vom Dateispeicher in die CPU-Baugruppe)		OK	OK
Schreiben von Dateien (Übertragung von der CPU-Baugruppe in den Dateispeicher)		OK (siehe Hinweis)	OK (siehe Hinweis)
Vergleichen von Dateien (Dateien in der CPU-Baugruppe und im Dateispeicher vergleichen)		OK	OK
Formatieren des Dateispeichers	Speicherkarten	OK	OK
	EM-Dateien	OK	OK
Ändern von Dateinamen		OK	Nicht möglich
Lesen von Dateispeicherdaten		OK	Nicht möglich
Löschen von Dateien		OK	OK
Kopieren von Dateien		OK	Nicht möglich
Löschen/Erstellen von Unterverzeichnissen		OK	Nicht möglich

Hinweis Bei einer CPU-Baugruppe der CS/CJ-Serie ab der Version 2.0 kann ein Leseschutz-Passwort verwendet werden um zu verhindern, dass Programmdateien in den Dateispeicher geschrieben werden (d. h. eine Speicherkarte oder den EM-Dateispeicher).

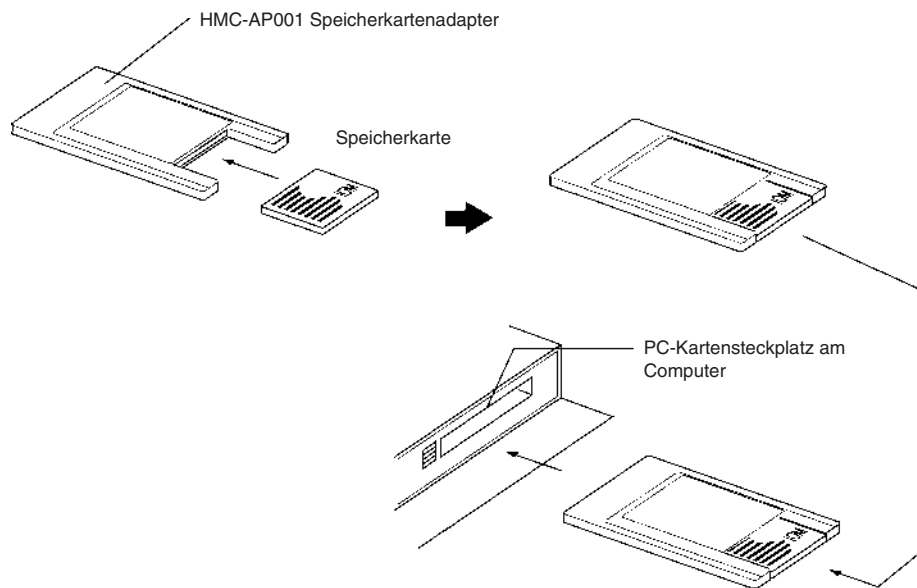


- Hinweis**
1. Erstellen Sie alle benötigten Datenträgerbezeichnungen mit Hilfe von Windows Explorer.
 2. Vom Dateispeicher wird die Windows-Schnellformatierung verwendet. Wenn bei Speicherkarten Formatierungsfehler auftreten, können Speicherkarten mit dem normalen Windows-Formatierungsbefehl formatiert werden.
 3. Datum und Uhrzeit für die Dateien, die bei Übertragungen von der CPU-Einheit in den Dateispeicher geschrieben werden, werden von der Uhr in der CPU-Baugruppe übernommen.



Hinweis Bei einer CPU-Baugruppe der CS/CJ-Serie ab der Version 2.0 und CX-Programmer ab Version 4.0 kann das Erstellen einer Backup-Programmdatei (.OBJ) als Option beim Registrieren eines Passworts für das gesamte Anwenderprogramm oder für bestimmte Tasks gesperrt werden. Weitere Informationen finden Sie im Abschnitt *Programmschreibschutz* unter 1-4-2 *Verbesserter Leseschutz durch Passwörter* im *CS-Serie Speicherprogrammierbare Steuerungen Bedienerhandbuch* oder im *CJ-Serie Speicherprogrammierbare Steuerungen Bedienerhandbuch*.

Eine Speicherkarte kann mit Hilfe des HMC-AP001-Speicherkartenadapters (separat erhältlich) in den PC-Kartensteckplatz eines Computers eingesetzt werden. Durch das Einsetzen der Speicherkarte in den Computer können die Dateien auf der Karte von anderen Programmen, wie Windows Explorer, gelesen und geschrieben werden.

**CX-Programmer**

Verwenden Sie für Dateispeichervorgänge die folgende Vorgehensweise.

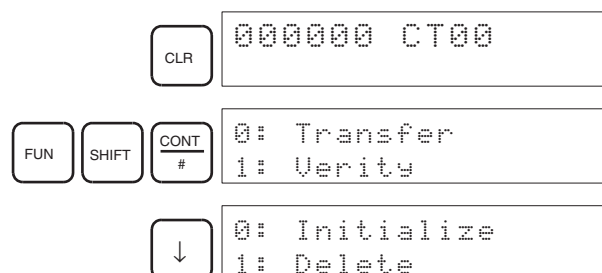
1,2,3...

1. Doppelklicken Sie im Projektfenster auf das Speichermodul-Symbol, während die CPU-Baugruppe online ist. Das Fenster „Speichermodul“ wird angezeigt.
2. Wählen Sie für die Übertragung von der CPU-Baugruppe zum Dateispeicher den Programmbereich, den E/A-Speicherbereich oder den Parameterbereich im Projektarbeitsbereich aus, wählen Sie im Dateispeicher **Übertragen** aus, und wählen Sie dann die Übertragung zur Speicherkarte oder dem EM-Dateispeicher.

oder Wählen Sie für die Übertragung vom Dateispeicher zur CPU-Baugruppe die Datei im Dateispeicher aus, und ziehen Sie sie in den Programmbereich, den E/A-Speicherbereich oder in den Parameterbereich im Projektarbeitsbereich, und lassen Sie die Maustaste los.

Hinweis

Verwenden Sie Projektübertragungsvorgänge, um mit CX-Programmer Symboltabellendateien (SYMBOLS.SYM) und Kommentardateien (COMMENTS.CMT) zu erstellen und zu lesen.

Programmierkonsole

Folgende Vorgänge können durchgeführt werden.

Eintrag 1	Eintrag 2	Eintrag 3	Eintrag 4	Eintrag 5
0: Senden	0: SPS an Speicherkarte	Wählen Sie OBJ, CIO, HR, WR, AR, DM, EM oder STD.	Legen Sie die Anfangs- und die Endadressen für die Übertragung fest.	Medientyp, Dateiname
	1: Speicherkarte an SPS	Wählen Sie OBJ, CIO, HR, WR, AR, DM, EM oder STD.	Legen Sie die Anfangs- und die Endadressen für die Übertragung fest.	Medientyp, Dateiname
1: Überprüfen		Wählen Sie OBJ, CIO, HR, WR, AR, DM, EM oder STD.	Legen Sie die Anfangs- und die Endadressen für den Vergleich fest.	Medientyp, Dateiname

Eintrag 1	Eintrag 2	Eintrag 3	Eintrag 4	Eintrag 5
2: Initialisieren		Geben Sie 9713 (Speicherkarte) oder 8426 (EM-Dateispeicher) ein.	---	---
3: Löschen		Wählen Sie OBJ, CIO, HR, WR, AR, DM, EM oder STD.	Medientyp, Dateiname	---

Hinweis Die Dateitypen sind in der folgenden Tabelle aufgeführt.

Symbol	Dateityp	
OBJ	Programmdatei (.OBJ)	
CIO	Datendatei (.IOM)	CIO-Bereich
HR		HR-Bereich
WR		WR-Bereich
AR		Zusatz-Systembereich
DM		DM-Bereich
EM0_		EM-Bereich
STD	Parameterdatei (.STD)	

Spezieller Hinweis zur Verifizierung von Daten nach dem Übertragen von Parameterdateien

Beim Dateivergleich von Parameterdaten vor und nach dem Übertragen können an der Programmierkonsole Verifizierungsfehler auftreten, wenn die Parameterdateien (.STD) in einer CJ-Serie CPU-Baugruppe erstellt und in einer CJ-Serie CPU-Baugruppe einer anderen Baugruppenversion auf Speicherkarte gespeichert werden. Die nachstehende Tabelle zeigt, in welchen Fällen es zum Auftreten solcher Verifizierungsfehler kommen kann.

Quell-CPU-Baugruppe		Ursprünglicher E/A-Zuordnungsmodus	Ziel-CPU-Baugruppe		
			Baugruppenversion der CPU-Baugruppe, in die die Parameterdateien übertragen werden		
			bis einschließlich Ver. 2.0	Baugruppenversion 2.0	ab Baugruppenversion 3.0
Baugruppenversion der CPU-Baugruppe, in der die Parameterdatei erzeugt wurde	bis einschließlich Ver. 2.0	Automatische Zuordnung	Verifizierung möglich	Verifizierung möglich	Verifizierungsfehler
		Benutzerdefiniert			Verifizierungsfehler
	Baugruppenversion 2.0	Automatische Zuordnung			Verifizierungsfehler
		Benutzerdefiniert			Verifizierung möglich
	ab Baugruppenversion 3.0	Automatische Zuordnung		Verifizierungsfehler	Verifizierung möglich
		Benutzerdefiniert		Verifizierung möglich	

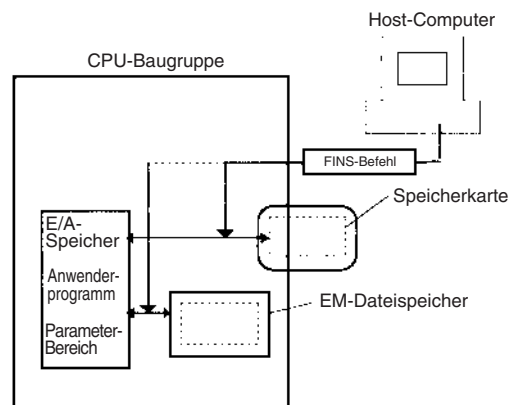
Quell-CPU-Baugruppe		Ursprünglicher E/A-Zuordnungsmodus	Ziel-CPU-Baugruppe		
			Baugruppenversion der CPU-Baugruppe, in die die beim Start automatisch zu übertragenden Dateien übertragen werden		
			bis einschließlich Ver. 2.0	Baugruppenversion 2.0	ab Baugruppenversion 3.0
Baugruppenversion der CPU-Baugruppe, in der die beim Start automatisch zu übertragenden Dateien erzeugt wurden	bis einschließlich Ver. 2.0	Automatische Zuordnung	Wechsel zu benutzerdefiniert	(Identisch) Automatische Zuordnung	Wechsel zu benutzerdefiniert
		Benutzerdefiniert	(Identisch) Benutzerdefiniert	Wechsel zu automatischer Zuordnung (siehe Hinweis 1)	(Identisch) Benutzerdefiniert
	Baugruppenversion 2.0	Automatische Zuordnung	Wechsel zu benutzerdefiniert	(Identisch) Automatische Zuordnung	Wechsel zu benutzerdefiniert
		Benutzerdefiniert	(Identisch) Benutzerdefiniert	(Identisch) Benutzerdefiniert	(Identisch) Benutzerdefiniert
	ab Baugruppenversion 3.0	Automatische Zuordnung	Wechsel zu benutzerdefiniert	(Identisch) Automatische Zuordnung	(Identisch) Automatische Zuordnung
		Benutzerdefiniert	(Identisch) Benutzerdefiniert	(Identisch) Benutzerdefiniert	(Identisch) Benutzerdefiniert

5-2-2 FINS-Befehle

Wenn die CPU-Baugruppe die korrekten FINS-Befehle empfängt, können folgende Dateispeichervorgänge von der Baugruppe ausgeführt werden. Diese sind den Programmiergerätfunktionen ähnlich.

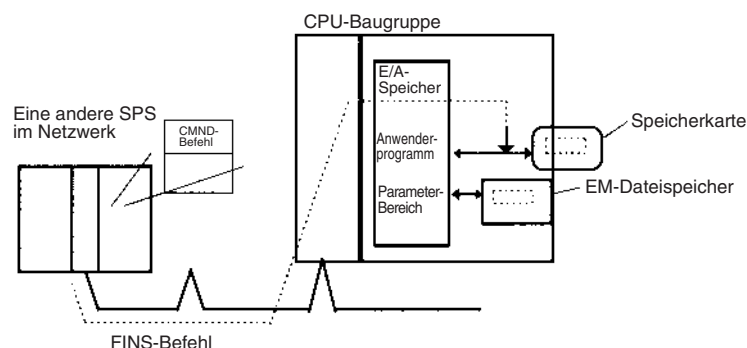
FINS-Befehle über Host-Link

Von einem über ein Host-Link-System verbundenen Computer können FINS-Befehle mit Host-Link-Header und -Terminator gesendet werden.

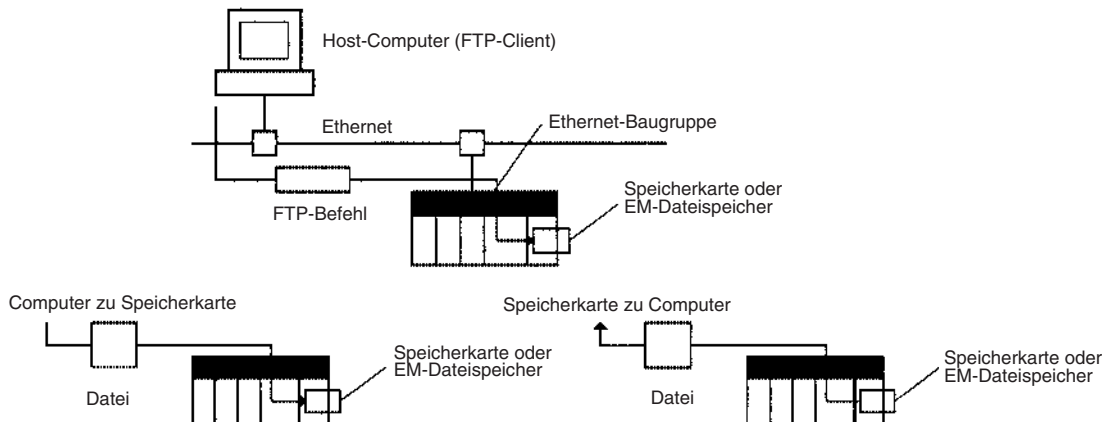


FINS-Befehl von einem anderen SPS-System im Netzwerk

Eine andere SPS im Netzwerk kann mit FINS-Befehlen mit dem Befehl CMND(490) senden.



Hinweis Von Computern in einem Ethernet-Netzwerk kann der Dateispeicher (Speicherkarten oder EM-Dateispeicher) einer CPU-Baugruppe mittels einer Ethernet-Baugruppe geschrieben und gelesen werden. Daten in Dateien können ausgetauscht werden, wenn der Host-Computer als FTP-Client und die SPS der CS/CJ-Serie als FTP-Server dient.



Die folgenden FINS-Befehle können zur Ausführung einer Reihe von Funktionen verwendet werden, z. B. dem Lesen und Schreiben von Dateien.

Befehl	Name	Beschreibung
2201 Hex	FILE NAME READ	Liest Dateispeicherdaten.
2202 Hex	SINGLE FILE READ	Liest Dateidaten einer angegebenen Länge ab einer spezifizierten Position in einer einzelnen Datei.
2203 Hex	SINGLE FILE WRITE	Schreibt Dateidaten einer angegebenen Länge ab einer spezifizierten Position in eine einzelne Datei.
2204 Hex	FILE MEMORY FORMAT	Formatiert (initialisiert) den Dateispeicher.
2205 Hex	FILE DELETE	Löscht spezifizierte Dateien aus dem Dateispeicher.
2207 Hex	FILE COPY	Kopiert Dateien von einem Dateispeicher in einen anderen Dateispeicher.
2208 Hex	FILE NAME CHANGE	Ändert einen Dateinamen.
220A Hex	MEMORY AREA-FILE TRANSFER	Überträgt oder vergleicht Daten zwischen dem E/A-Speicherbereich und dem Dateispeicher.
220B Hex	PARAMETER AREA-FILE TRANSFER	Überträgt oder vergleicht Daten zwischen dem Parameterbereich und dem Dateispeicher.
220C Hex	PROGRAM AREA-FILE TRANSFER	Überträgt oder vergleicht Daten zwischen dem Anwenderspeicherbereich und dem Dateispeicher.
2215 Hex	CREATE/DELETE SUBDIRECTORY	Erstellt und löscht Unterverzeichnisse.

Hinweis Für das Datieren der im Dateispeicher mit den Befehlen 220A, 220B, 220C und 2203 erstellten Dateien wird die Zeit der internen Uhr der CPU-Baugruppe verwendet.

5-2-3 FREAD(700), FWRIT(701) und CMND(490)

Der Befehl FWRIT(701) (WRITE DATA FILE) kann zum Erstellen einer Datendatei, die sie spezifizierten E/A-Speicherdaten enthält, auf einer Speicherkarte oder im EM-Dateispeicher verwendet werden. Das Hinzufügen oder Überschreiben in vorhandenen Dateien ist von jeder Position aus ebenfalls möglich.

Mit dem Befehl FREAD(700) (READ DATA FILE) werden E/A-Speicherdaten ab einer spezifizierten Position aus einer Datendatei auf einer Speicherkarte oder im EM-Dateispeicher gelesen und anschließend in den spezifizierten Bereich des

E/A-Speichers geschrieben. Das Lesen kann ab jeder Stelle in der spezifizierten Datei erfolgen.

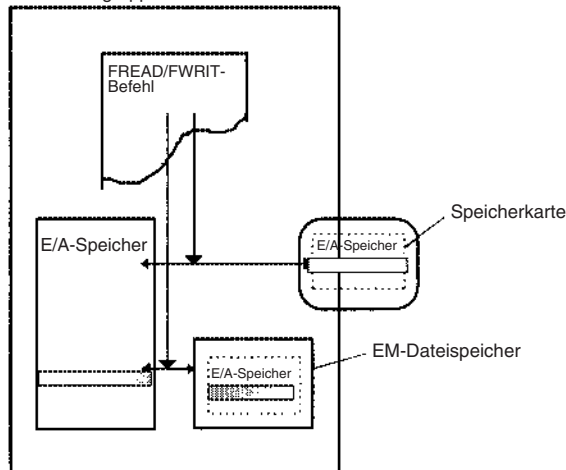
Hinweis

Mit diesen Befehlen wird nicht die spezifizierte Datei übertragen, sondern nur die spezifizierte Menge von Daten, beginnend an der spezifizierten Startposition in der Datei.

Der Befehl CMND(490) (DELIVER COMMAND) kann zur Ausgabe eines FINS-Befehls an die CPU-Baugruppe selbst ausgeführt werden, um Dateioperationen zu ermöglichen. Dateioperationen wie das Formatieren, Löschen, Kopieren und Umbenennen von Dateien können mit den Dateien auf der Speicherkarte oder im EM-Dateispeicher durchgeführt werden. (Nicht unterstützt von CS1 CPU-Einheiten der CS-Serie vor EV1.)

FREAD(700)/FWRIT(701): Übertragungen zwischen E/A-Speicher und Dateispeicher

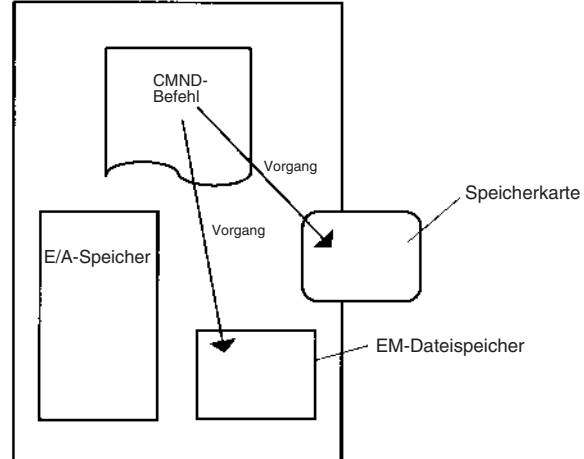
CPU-Baugruppe



CMND(490): Dateispeichervorgänge

(Nicht möglich für CPU-Baugruppen der CS-Serie vor EV1)

CPU-Baugruppe



Befehle FREAD(700)/FWRIT(701)

Die Befehle FREAD(700) und FWRIT(701) übertragen Daten zwischen E/A- und Dateispeicher. Von allen CPU-Baugruppen der CJ-Serie können Binärdaten (.IOM-Dateien) übertragen werden, und bei den CPU-Baugruppen der Version V1 können auch ASCII-Dateien (.TXT- und .CSV-Dateien) übertragen werden.

Bezeichnung	AWL	Beschreibung
READ DATA FILE	FREAD(700)	Liest spezifizierte Dateidaten oder Datenelemente in einen spezifizierten E/A-Speicher.
WRITE DATA FILE	FWRIT(701)	Verwendet spezifizierte E/A-Speicherbereichsdaten zum Erstellen einer spezifizierten Datendatei.

Übertragen von ASCII-Dateien
(Nicht unterstützt von CS1 CPU-Baugruppen der CS-Serie vor EV1)

ASCII-Dateien können ebenso wie Binärdateien übertragen werden. Die dritte und vierte Stelle des Wort-Operanden (C) der Befehlssteuerung gibt den Typ der Datendatei an, die übertragen wird, und die Feldanzahl zwischen Wagenrückläufen.

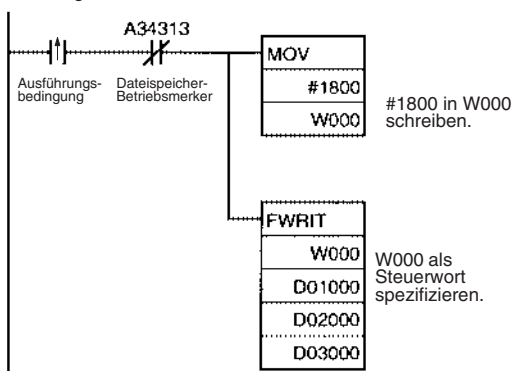
Bits in C	Einstellungen	Einschränkungen bei Programmiergeräten
12 bis 15	Datentyp 0: Binär (.IOM) 1: Nicht getrennte Worte (.TXT) 2: Nicht getrennte Doppelworte (.TXT) 3: Durch Komma getrennte Worte (.CSV) 4: Durch Komma getrennte Doppelworte (.CSV) 5: Durch Tabulator getrennte Worte (.TXT) 6: Durch Tabulator getrennte Doppelworte (.TXT)	Wenn CX-Programmer V1.1 oder eine niedrigere Version verwendet wird, kann nur 0 Hex (.IOM-Dateien) direkt spezifiziert werden. Wenn CX-Programmer V1.2 oder eine höhere Version (oder eine Programmierkonsole) verwendet wird, können die Steuerwort-Bits auf 0 bis 6 Hex gesetzt werden.
08 bis 11	Wagenrückläufe 0: Keine 8: Nach jeweils 10 Feldern 9: Nach jedem Feld A: Nach jeweils 2 Feldern B: Nach jeweils 4 Feldern C: Nach jeweils 5 Feldern D: Nach jeweils 16 Feldern	Wenn CX-Programmer V1.1 oder eine niedrigere Version (oder eine Programmierkonsole) verwendet wird, kann nur 0 Hex (kein Wagenrücklauf) direkt spezifiziert werden. Wenn CX-Programmer V1.2 oder eine höhere Version verwendet wird, können die Steuerwort-Bits auf 0 Hex oder von 8 bis D Hex gesetzt werden.

CX-Programmer V1.1 oder niedriger: Indirekte Einstellung des Steuerworts

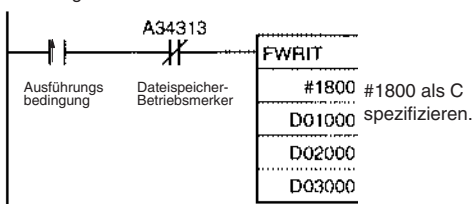
Wenn V1.1. oder eine niedrigere Version von CX-Programmer verwendet wird, können ASCII-Dateien nicht mit den Befehlen FREAD(700) und FWRIT(701) übertragen werden, wenn eine Konstante für das Steuerwort eingegeben wird, um den Datentyp und die Wagenrückläufe zu spezifizieren. Wenn eine Konstante verwendet wird, können nur Binärdaten ohne Wagenrückläufe übertragen werden.

ASCII-Dateien können mit den Befehlen FREAD(700) und FWRIT(701) jedoch übertragen werden, wenn eine indirekte Einstellung des Steuerworts vorgenommen wird. Schreiben Sie die gewünschte Steuerwordeinstellung in ein Wort, und spezifizieren Sie dieses Wort in FREAD(700) oder FWRIT(701) als Steuerwort, wie in der folgenden Abbildung auf der linken Seite gezeigt.

CX-Programmer bis Version V1.1



CX-Programmer ab Version V1.2



Hinweis Für das Datieren der im Dateispeicher mit FWRIT(701) erstellten Dateien wird die Zeit der internen Uhr der CPU-Baugruppe verwendet.

Es kann immer nur eine Dateispeicheroperation durchgeführt werden. Daher dürfen die Befehle FREAD(700) und FWRIT(701) nicht ausgeführt werden, während eine der folgenden Dateispeicheroperationen ausgeführt wird:

- 1,2,3...**
1. Ausführung des Befehls FREAD(700) oder FWRIT(701)
 2. Ausführung des Befehls CMND(490) zum Senden eines FINS-Befehls an die CPU-Baugruppe selbst
 3. Austausch des gesamten Programms durch Steuerbit-Operationen des Zusatz-Systembereichs
 4. Ausführen eines Easy-Backup-Sicherungsvorgangs

Verwenden Sie den Dateispeicher-Vorgangsmerker (A34313), um Dateispeicherbefehle ausschließlich zu steuern und damit deren Ausführung während eines anderen Dateispeichervorgangs zu verhindern.

Wenn FREAD(700) ausgeführt wird, wird der Dateilesefehlermerker (A34310) auf EIN gesetzt, und der Befehl wird nicht ausgeführt, wenn die spezifizierte Datei den falschen Datentyp enthält oder die Dateidaten beschädigt sind. Bei Text- oder CSV-Dateien muss der Zeichencode aus hexadezimalen Daten bestehen, und die Trennzeichen müssen alle 4 Stellen bei Wortdaten und alle 8 Stellen bei Doppelwortdaten gesetzt werden. Die Daten werden bis zu der Stelle gelesen, an der ein ungültiges Zeichen erkannt wird.

Zugehörige Bits/Worte im Zusatz-Systembereich

Bezeichnung	Adresse	Funktion
Speicherkartentyp	A34300 bis A34302	Bezeichnet den Typ der Speicherkarte, sofern eine solche in die CPU-Baugruppe eingesetzt ist.
EM-Dateispeicher-Formatierungs-Fehlermerker	A34306	EIN, wenn ein Formatierungsfehler in der ersten für Dateispeicher zugeordneten EM-Bank auftritt. Dieser Merker wird nach fehlerfreier Formatierung auf AUS gesetzt.
Speicherkarten-Formatierungs-Fehlermerker	A34307	EIN bei unformatierter Speicherkarte oder Auftreten eines Formatierungsfehlers.
Dateischreibfehlermerker	A34308	EIN, wenn beim Schreiben in die Datei ein Fehler auftrat.
Schreiben-in-Datei-unmöglich-Merker	A34309	EIN, wenn die Daten nicht geschrieben werden konnten, weil die Datei schreibgeschützt war oder nicht ausreichend freier Speicherplatz zur Verfügung stand.
Dateilesefehlermerker	A34310	EIN, wenn eine Datei aufgrund beschädigter Daten oder des falschen Datentyps nicht gelesen werden konnte.
Keine-Datei-Merker	A34311	EIN, wenn Daten nicht gelesen werden konnten, weil die spezifizierte Datei nicht vorhanden ist.
Dateispeicher-Vorgangsmerker	A34313	EIN, falls eine der folgenden Bedingungen zutrifft: Die CPU-Baugruppe verarbeitet einen mit CMND(490) an sich selbst gesendeten FINS-Befehl. Der Befehl FREAD(700) oder FWRIT(701) wird ausgeführt. Das Programm wird mittels eines Steuerbits des Zusatz-Systembereichs überschrieben. Ein Easy-Backup-Sicherungsvorgang wird ausgeführt.
Dateizugriffsmerker	A34314	EIN, während ein Zugriff auf Dateidaten erfolgt.
Speicherkarten-Erkennungsmerker	A34315	EIN, wenn eine Speicherkarte erkannt wurde. (Nicht unterstützt von CS1 CPU-Baugruppen der CS-Serie vor EV1)
Anzahl der noch zu übertragenden Elemente	A346 bis A347	Diese Worte geben die Anzahl der noch zu übertragenden Worte bzw. Felder an (32 Bits). Wenn eine Binärdatei (.IOM) übertragen wird, verringert sich die Zahl mit jedem gelesenen Wort. Wenn eine Text- oder CSV-Datei übertragen wird, verringert sich die Zahl mit jedem übertragenen Feld.

CMND(490): DELIVER COMMAND

CMND(490) kann zur Ausgabe eines FINS-Befehls an die lokale CPU-Baugruppe selbst verwendet werden, um Dateispeicheroperationen wie das Formatieren oder Löschen von Dateien durchzuführen. Nehmen Sie die folgenden Einstellungen in den Steuerworten des Befehls CMND(490) vor, wenn Sie einen Dateispeicher-FINS-Befehl an die lokale SPS ausgeben möchten:

- 1,2,3...**
1. Setzen Sie die Zielnetzwerkadresse in C+2 auf 00 (lokales Netzwerk).
 2. Setzen Sie die Zielnetzwerkadresse in C+3 auf 00 (CPU-Baugruppe des SPS-Systems) und den Zielknotenpunkt auf 00 (innerhalb des lokalen Knotens).
 3. Setzen Sie die Anzahl der Wiederholungen in C+4 auf 0. (diese Einstellung ist ungültig; Stellen Sie diese auf 0.)

FINS-Befehle für den Dateispeicher

Weitere Informationen über FINS-Befehle finden Sie unter 5-2-2 *FINS-Befehle*.

Hinweis

In der folgenden Tabelle sind nicht alle ausführbaren FINS-Befehle für den Dateispeicher aufgeführt. Weitere Informationen zu FINS-Befehlen finden Sie im *Kommunikationsbefehle Referenzhandbuch* (W342).

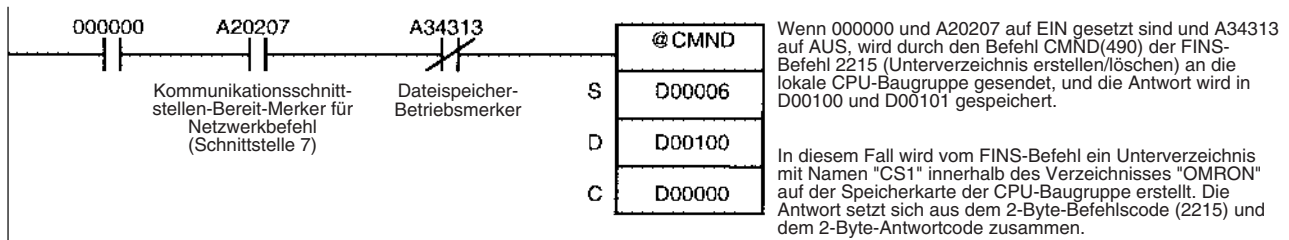
Der Befehl CMND(490) kann für die lokale CPU-Baugruppe nicht ausgeführt werden, wenn ein anderer CMND(490)-Befehl für eine andere CPU-Baugruppe ausgeführt wird, die Befehle FREAD(700) oder FWRT(701) ausgeführt werden, das Programm durch Steuerbit-Operationen des Zusatz-Systembereichs ausgetauscht wird, oder wenn ein Easy-Backup-Sicherungsvorgang ausgeführt wird. Stellen Sie sicher, dass der Dateispeicher-Vorgangsmerker (A34313) als invertierte Bedingungsabfrage gilt, um zu verhindern, dass CMND(490) während eines anderen Dateispeichervorgangs ausgeführt wird.

Wenn der Befehl CMND(490) für die lokale CPU-Baugruppe nicht ausgeführt werden kann, wird der Fehlermerker auf EIN gesetzt.

Zugehörige Bits/Worte im Zusatz-Systembereich

Bezeichnung	Adresse	Funktion
Dateispeicher-Vorgangsmerker	A34313	EIN, falls eine der folgenden Bedingungen zutrifft: • Die CPU-Baugruppe verarbeitet einen mit CMND(490) an sich selbst gesendeten FINS-Befehl. • Der Befehl FREAD(700) oder FWRT(701) wird ausgeführt. • Das Programm wird mittels eines Steuerbits des Zusatz-Systembereichs überschrieben. • Ein Easy-Backup-Sicherungsvorgang wird ausgeführt.
Speicherkarten-Erkennungsmerker	A34315	EIN, wenn eine Speicherkarte erkannt wurde. (Nicht unterstützt von CS1 CPU-Baugruppen der CS-Serie vor EV1)

Das folgende Beispiel zeigt, wie der Befehl CMND(490) zum Erstellen eines Unterverzeichnisses auf der Speicherkarte verwendet wird.



		15	8	7	0	
S:	D00006	2	2	1	5	Befehlscode: 2215 Hex (Unterverzeichnis erstellen/löschen)
S+1:	D00007	8	0	0	0	Datenträgernummer: 8000 Hex (Speicherkarte)
S+2:	D00008	0	0	0	0	Parameter: 0000 Hex (Unterverzeichnis erstellen.)
S+3:	D00009	4	3	5	3	Name des Unterverzeichnisses: CS1□□□□□□□□ (□: ein Leerzeichen)
S+4:	D00010	3	1	2	0	
S+5:	D00011	2	0	2	0	
S+6:	D00012	2	0	2	0	
S+7:	D00013	2	E	2	0	
S+8:	D00014	2	0	2	0	Verzeichnislänge: 0006 Hex (6 Zeichen)
S+9:	D00015	0	0	0	6	
S+10:	D00016	5	C	4	F	
S+11:	D00017	4	D	5	2	Verzeichnispfad: \OMRON
S+12:	D00018	4	F	4	E	

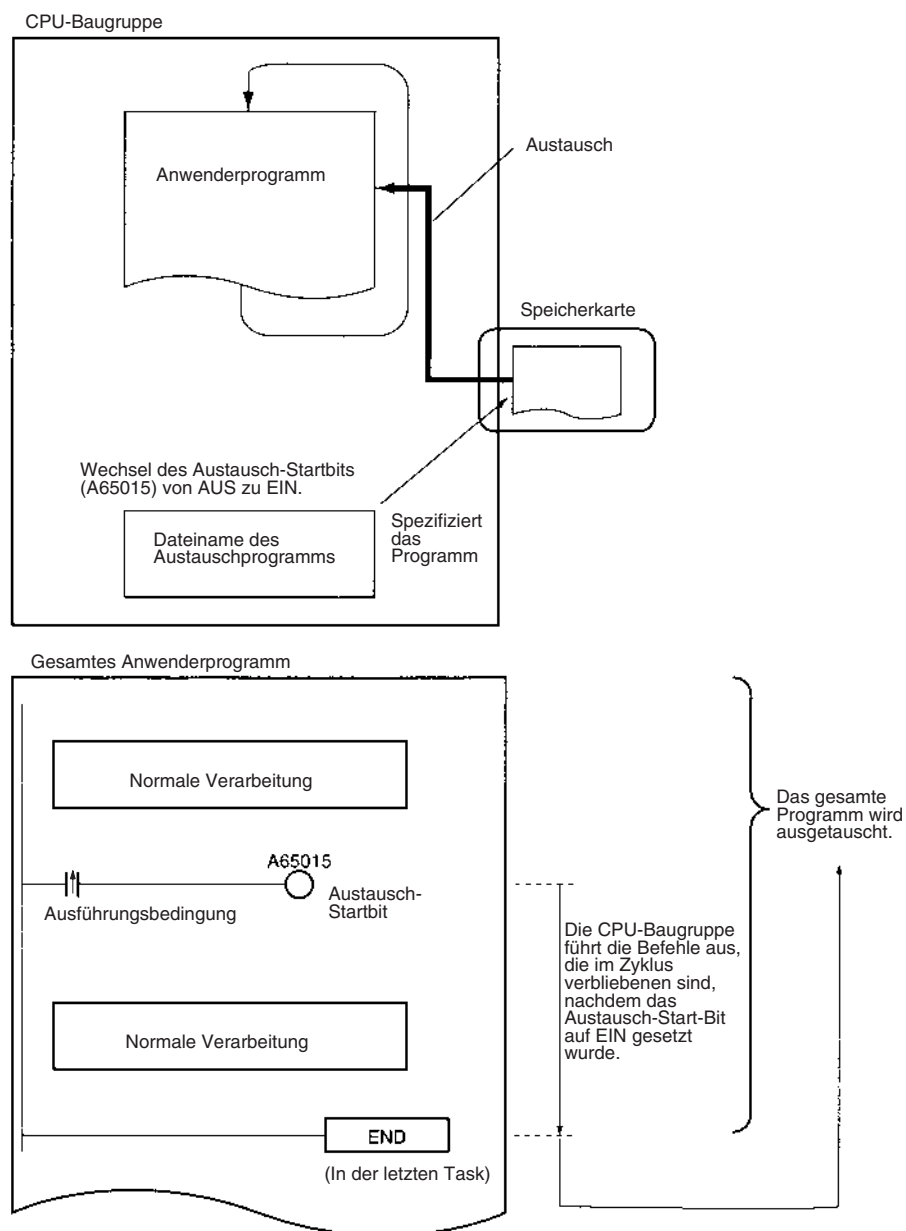
		15	8	7	0	
C:	D00000	0	0	1	A	Byte-Anzahl der Befehlsdaten: 001A Hex (26 Byte)
C+1:	D00001	0	0	0	4	Byte-Anzahl der Antwortdaten: 0004 Hex (4 Byte)
C+2:	D00002	0	0	0	0	Zieladresse: 0000 Hex (lokales Netzwerk)
C+3:	D00003	0	0	0	0	00 Hex (lokaler Knoten) und 00 Hex (CPU-Baugruppe)
C+4:	D00004	0	7	0	0	Antwort angefordert, Kommunikationsschnittstelle 7, 0 Wiederholungen
C+5:	D00005	0	0	0	0	Antwortüberwachungszeit: FFFF Hex (6.553,5 s)

Hinweis Zusätzlich zu den in der obigen Liste aufgeführten Befehlen für Dateispeicheroperationen gibt es weitere FINS-Befehle, die an die lokale SPS gesendet werden können. Der Dateispeicher-Vorgangsmerker muss verwendet werden, um das gleichzeitige Ausführen dieser anderen FINS-Befehle zu verhindern.

5-2-4 Austausch des gesamten Programms während des Betriebs

(Nicht unterstützt von CS1 CPU-Baugruppen der CS-Serie vor EV1)

Das gesamte Programm kann während des Betriebs ausgetauscht werden (in der RUN- oder MONITOR-Betriebsart), indem das Austausch-Startbit (A65015) auf EIN gesetzt wird. Die spezifizierte Datei wird von der Speicherkarte gelesen, und das ausführbare Programm wird am Ende des aktuellen Zyklus ausgetauscht. Das Programmaustauschpasswort (A651) und der Programmdateiname (A654 bis A657) müssen im Vorfeld aufgezeichnet werden, und die angegebene Programmdatei muss auf der Speicherkarte vorhanden sein, damit das Programm während des Betriebs ausgetauscht werden kann.



Das Programm kann auch ausgetauscht werden, wenn die Programmausführung angehalten wird (PROGRAM-Betriebsart), indem das Austausch-Startbit von einem Programmiergerät auf EIN gesetzt wird.

Hinweis

Die Datei für den Programmaustausch kann nicht aus dem EM-Dateispeicher gelesen werden.

Das Austausch-Startbit (A65015) kann an jeder Position im Programm (Programmadresse) auf EIN gesetzt werden. Die CPU-Baugruppe führt nach dem Setzen des Austausch-Startbits von AUS auf EIN die im Zyklus verbliebenen Befehle aus.

Das Programm wird während des Austauschs nicht ausgeführt. Nach dem Austausch des Programms wird der Betrieb wieder gestartet. Der Vorgang ist derselbe wie beim Wechsel der CPU-Baugruppe von der PROGRAM-Betriebsart in die RUN- oder MONITOR-Betriebsart.

Das Programm wird am Ende des Zyklus ausgetauscht, bei dem das Austausch-Startbit von AUS auf EIN gesetzt wurde, d. h., nachdem END(001) bei der letzten Task im Programm ausgeführt wurde.

Hinweis

1. Setzen Sie den E/A-Speicher-Haftmerker (A50012) auf EIN, wenn Sie den Status der E/A-Speicherdaten beibehalten möchten, während das Programm ausgetauscht wird.

Setzen Sie den Zwangssetzungs-Haftmerker (A50013) auf EIN, wenn Sie den Status der Bits, die zwangsweise gesetzt oder rückgesetzt wurden, beibehalten möchten, während das Programm ausgetauscht wird.

2. Wenn der E/A-Speicher-Haftmerker (A50012) auf EIN gesetzt ist, bevor das Programm ausgetauscht wird, bleibt der Status der Bits im E/A-Speicherbereich nach dem Programmaustausch erhalten. Stellen Sie sicher, dass die angeschlossenen Geräte mit denselben E/A-Speicherdaten ordnungsgemäß funktionieren.

Ebenso wird der Status der zwangsweise gesetzten oder rückgesetzten Bits nach dem Programmaustausch beibehalten, wenn der Zwangssetzungs-Haftmerker (A50013) auf EIN gesetzt ist, bevor das Programm ausgetauscht wird. Stellen Sie sicher, dass die angeschlossenen Geräte mit denselben zwangsweise gesetzten oder rückgesetzten Bits ordnungsgemäß funktionieren.

Austauschdatei

Die im Programmdateinamen (A654 bis A657) spezifizierte Programmdatei wird von der Speicherkarte gelesen, und das vorhandene Programm wird am Ende des Zyklus ausgetauscht, in dem das Austausch-Startbit (A65015) von AUS auf EIN gesetzt wird.

Datei	Dateiname und Namensweiterung	Spezifizierung des Austauschdateinamens (*****)
Programmdatei	*****.OBJ	Schreiben Sie den Namen der Programmaustauschdatei in A654 bis A657, bevor das Programm ausgetauscht wird.

Voraussetzungen, die für den Programmaustausch erfüllt sein müssen

Folgende Voraussetzungen müssen erfüllt sein, damit das Programm während des Betriebs ausgetauscht werden kann.

- Das Programmpasswort (A5A5) wurde in A651 geschrieben.
- Die in den Programmdateinamen-Worten (A654 bis A657) spezifizierte Programmdatei ist im Verzeichnis der Speicherkarte vorhanden.
- Die Speicherkarte wurde von der CPU-Baugruppe erkannt. (A34315 EIN)
- Es sind keine schwerwiegenden Fehler aufgetreten.
- Es werden keine Dateispeichervorgänge ausgeführt. (A34313 AUS)
- Es werden keine Daten in den Programmbereich geschrieben.
- Das Zugriffsrecht besteht. (Z. B., wenn Daten von CX-Programmer nicht zur SPS übertragen werden.)

Hinweis

Das Programm kann in jeder Betriebsart übertragen werden.

Betrieb der CPU-Baugruppe während des Programmaustauschs

Der Betrieb der CPU-Baugruppe ist während des Programmaustauschs wie folgt:

- Programmausführung: Angehalten
- Zykluszeitüberwachung: Keine Überwachung

Operationen, die während und nach dem Programmaustausch fortgesetzt werden

Wenn der E/A-Speicher-Haftmerker (A50012) auf EIN gesetzt ist, bleiben die Daten in den folgenden Speicherbereichen erhalten: CIO-Bereich, Arbeitsbereich (W), Zeitgeber-Ablaufmerker (T), Index-Register (IR), Daten-Register (DR) und die aktuelle EM-Banknummer.

Hinweis

Die Zeitgeber-Istwerte werden beim Programmaustausch gelöscht.

Wenn der E/A-Speicher-Haftmerker bei der Programmübertragung auf EIN gesetzt ist, werden die vor dem Programmaustausch bestehenden Schaltzu-

stände der Lastausgänge nach dem Austausch weiterhin ausgegeben. Stellen Sie sicher, dass die angeschlossenen Geräte nach dem Programmaustausch ordnungsgemäß funktionieren.

Der Status der zwangsweise gesetzten oder rückgesetzten Bits bleibt während des Programmaustauschs erhalten, wenn der Zwangssetzungs-Haftmerker (A50013) auf EIN gesetzt ist.

Interrupts werden maskiert.

Wenn Daten aufgezeichnet werden, wird die Aufzeichnung angehalten.

Befehlsbedingungen (Sperren, Unterbrechungen und Blockprogrammausführung) werden initialisiert.

Flankenauswertungs-Merker werden unabhängig davon, ob der E/A-Speicher-Haftmerker auf EIN oder auf AUS gesetzt ist, initialisiert.

Betrieb nach Programmaustausch

Der Status der zyklischen Tasks hängt von deren Eigenschaften bei Betriebsbeginn ab. (Der Status ist derselbe wie beim Umschalten der SPS von der PROGRAM- in die RUN/MONITOR-Betriebsart.)

Der Erster-Zyklus-Merker (A20011) wird einen Zyklus lang auf EIN gesetzt, wenn die Programmausführung wieder aufgenommen wird. (Der Status ist derselbe wie beim Umschalten der SPS von der PROGRAM- in die RUN/MONITOR-Betriebsart.)

Für den Programmaustausch erforderliche Zeit

Größe des gesamten Programms	Im SPS-Setup eingestellte Peripherie-Bedienungszeit	Für den Programmaustausch etwa erforderliche Zeit
60 kSteps	Standard (4 % der Zykluszeit)	6 s
250 kSteps		25 s

Zugehörige Bits/Worte im Zusatz-Systembereich

Bezeichnung	Adresse	Funktion
Dateispeicher-Vorgangsmarker	A34313	EIN, falls eine der folgenden Bedingungen zutrifft: Die CPU-Baugruppe hat mit CMND(490) einen FINS-Befehl an sich selbst gesendet. Der Befehl FREAD(700) oder FWRT(701) wird ausgeführt. Das Programm wird mittels eines Steuerbits des Zusatz-Systembereichs (A65015) überschrieben. Ein Easy-Backup-Sicherungsvorgang wird ausgeführt.
Speicherkarten-Erkennungsmarker (Nicht unterstützt von CS1 CPU-Baugruppen der CS-Serie vor EV1)	A34315	EIN, wenn eine Speicherkarte erkannt wurde.
E/A-Speicher-Haftmerker	A50012	Wenn dieses Bit auf EIN gesetzt ist, bleibt der Inhalt des E/A-Speichers während des Programmaustauschs erhalten.
Zwangssetzungs-Haftmerker	A50013	Wenn dieses Bit auf EIN gesetzt ist, bleibt der Status der zwangsweise gesetzten und rückgesetzten Bits während des Programmaustauschs erhalten.

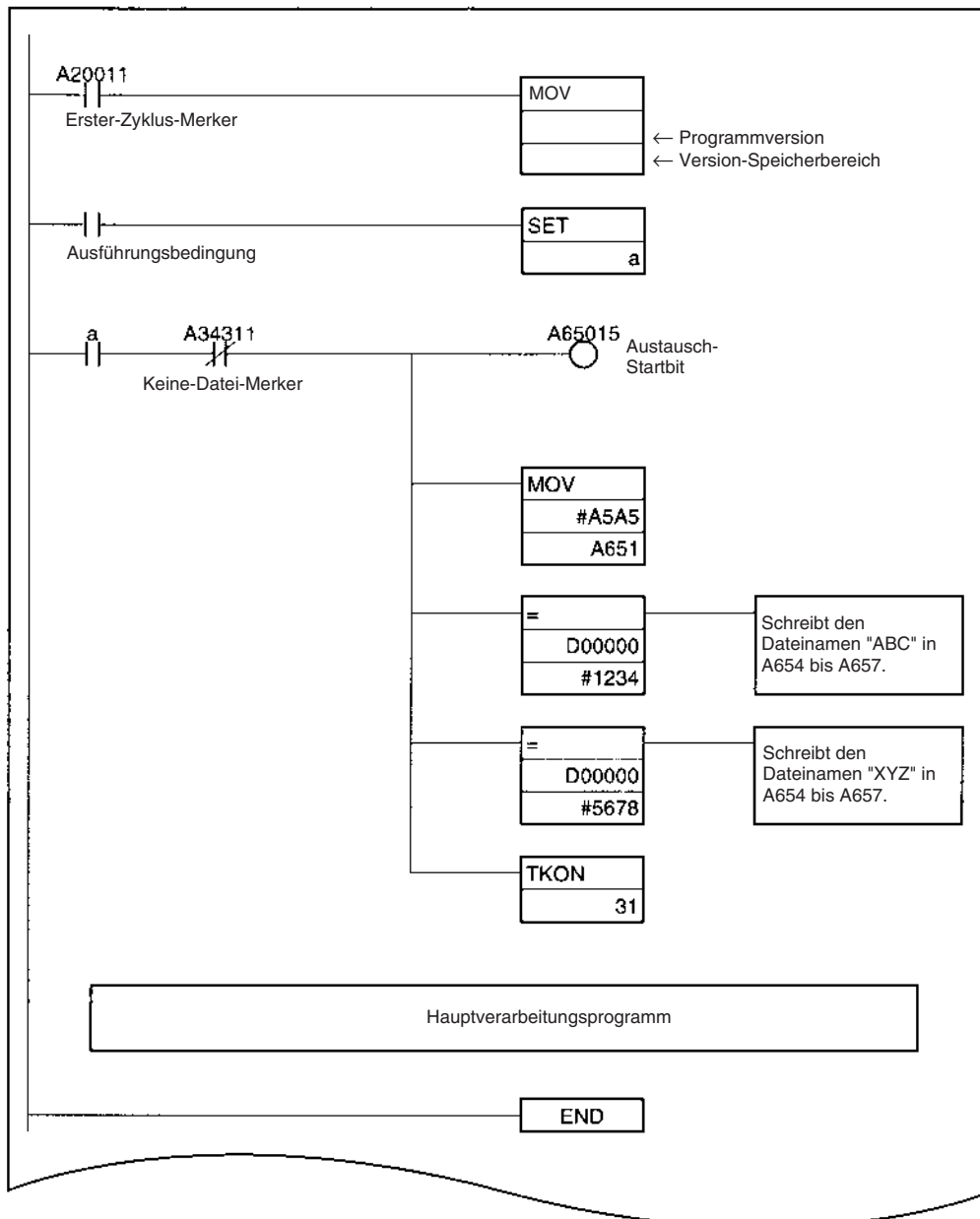
Bezeichnung	Adresse	Funktion															
Programmaustausch-Abschluss-Code (Nicht unterstützt von CS1 CPU-Baugruppen der CS-Serie vor EV1)	A65000 bis A65007	Codes für normalen Programmaustausch (A65014 AUS): 01 Hex: Die Programmdatei (.OBJ) ersetzt das Programm. Codes für unvollständigen Programmaustausch (A65014 EIN): 00 Hex: Ein schwerwiegender Fehler ist aufgetreten. 01 Hex: Ein Speicherfehler ist aufgetreten. 11 Hex: Das Programm ist schreibgeschützt. 12 Hex: Das Programmpasswort in A651 ist ungültig. 21 Hex: Es ist keine Speicherkarte eingesetzt. 22 Hex: Die spezifizierte Datei ist nicht vorhanden. 23 Hex: Die spezifizierte Datei ist zu groß (Speicherfehler). 31 Hex: Einer der folgenden Vorgänge wurde ausgeführt: - Ein Dateispeichervorgang wurde ausgeführt. - Das Programm wurde übertragen. - Die Betriebsart wurde geändert.															
Programmaustausch-Fehlermerker (Nicht unterstützt von CS1 CPU-Baugruppen der CS-Serie vor EV1)	A65014	Auf EIN gesetzt, wenn beim Programmaustausch ein Fehler auftrat, nachdem A65015 von AUS auf EIN gesetzt wurde. Auf AUS gesetzt, nachdem A65015 das nächste Mal von AUS auf EIN gesetzt wurde.															
Austausch-Startbit (Nicht unterstützt von CS1 CPU-Baugruppen der CS-Serie vor EV1)	A65015	Wenn dieses Bit durch die Einstellung des Programmpassworts (A651) auf A5A5 Hex aktiviert wurde, wird der Programmaustausch gestartet, wenn dieses Bit von AUS auf EIN gesetzt wird. Setzen Sie dieses Bit während des Programmaustauschs nicht nochmals von AUS auf EIN. Dieses Bit wird automatisch auf AUS gesetzt, wenn der Programmaustausch abgeschlossen ist (normal oder mit einem Fehler), oder wenn die Versorgungsspannung aus- und wieder eingeschaltet wird. Der Status dieses Bits kann durch ein Programmiergerät, ein NT/NS-Terminal oder einen Host-Computer ausgelesen werden, um festzustellen, ob der Programmaustausch abgeschlossen ist oder nicht.															
Programmpasswort (Nicht unterstützt von CS1 CPU-Baugruppen der CS-Serie vor EV1)	A651	Für einen Programmaustausch muss in dieses Wort das Passwort eingegeben werden. A5A5 Hex: Ermöglicht den Austausch mit dem Austausch-Startbit (A65015). Anderer Wert: Sperrt das Austausch-Startbit (A65015). Dieses Bit wird automatisch auf AUS gesetzt, wenn der Programmaustausch abgeschlossen ist (normal oder mit einem Fehler), oder wenn die Versorgungsspannung aus- und wieder eingeschaltet wird.															
Programmdateiname (Nicht unterstützt von CS1 CPU-Baugruppen der CS-Serie vor EV1)	A654 bis A657	Schreiben Sie den Dateinamen der Austauschprogrammdatei in ASCII vor dem Programmaustausch in diese Worte. Schreiben Sie nur den aus 8 Zeichen bestehenden Dateinamen, die OBJ-Erweiterung wird automatisch hinzugefügt. Schreiben Sie die Zeichen mit A654 beginnend (das höchstwertige Byte zuerst). Wenn der Dateiname aus weniger als 8 Zeichen besteht, füllen Sie die übrigen Bytes mit Leerzeichen (20 Hex) auf. Verwenden Sie innerhalb des Dateinamens keine NULL-Zeichen oder Leerzeichen. Das folgende Beispiel zeigt die Daten für die Programmdatei ABC.OBJ: <table> <tr> <td></td><td>15</td><td>0</td></tr> <tr> <td>A654</td><td>41</td><td>42</td></tr> <tr> <td>A655</td><td>43</td><td>20</td></tr> <tr> <td>A656</td><td>20</td><td>20</td></tr> <tr> <td>A657</td><td>20</td><td>20</td></tr> </table>		15	0	A654	41	42	A655	43	20	A656	20	20	A657	20	20
	15	0															
A654	41	42															
A655	43	20															
A656	20	20															
A657	20	20															

Beispielprogramm 1

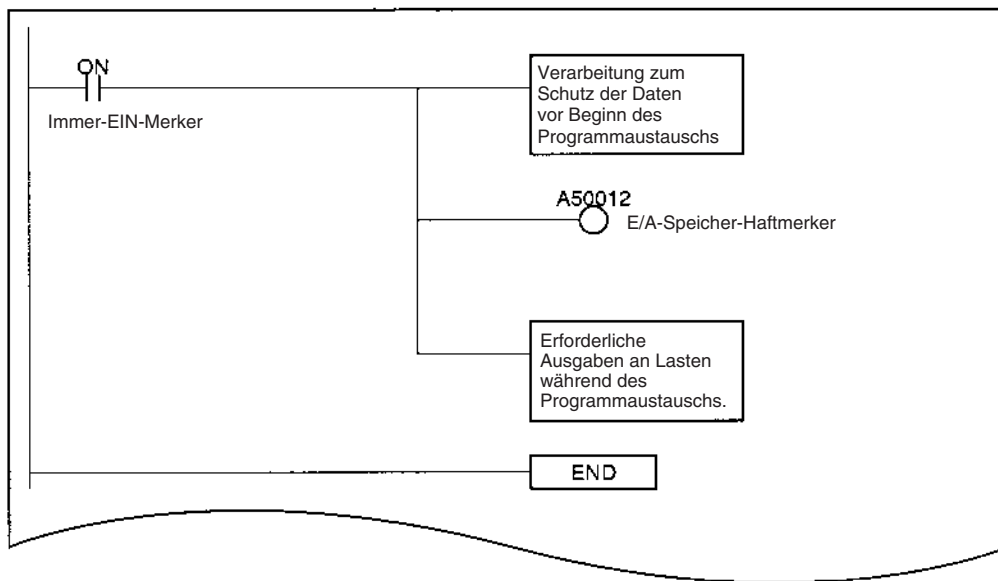
Speichern Sie die Programmdateien ABC.OBJ und XYZ.OBJ auf der Speicherkarte, und wählen Sie dem Wert von D00000 entsprechend eines der Programme aus. Setzen Sie D00000 auf #1234, um ABC.OBJ auszuwählen, oder auf #5678 zur Auswahl von XYZ.OBJ.

Starten Sie eine andere Task und führen Sie diese aus, um jegliche vor dem Programmaustausch oder Verarbeitung des E/A-Speicher-Haftmerkers erforderlichen Vorgänge auszuführen.

Haupt-Task (Zyklische Task Nummer 0)



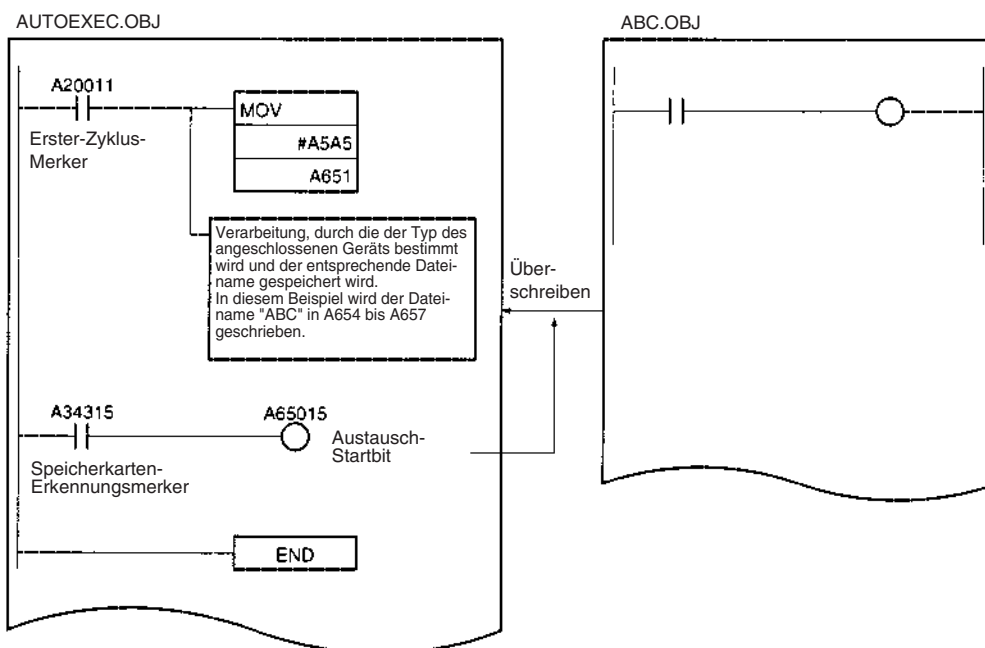
Task zum Schutz der Daten während des Programmaustauschs
(Zyklische Tasks Nummer 31, Standby(nicht ausführbarer)-Status beim Start)



Beispielprogramm 2

Speichern Sie die Programmdateien für mehrere Geräte und die Programmdatei für den automatischen Startvorgang beim Starten (AUTOEXEC.OBJ oder REPLACE.OBJ (siehe Hinweis)) auf eine Speicherkarte. Beim Starten der SPS wird die Datei für den automatischen Startvorgang gelesen, und das Programm wird zu einem späteren Zeitpunkt mit einer Programmdatei für ein anderes Gerät ausgetauscht.

Hinweis REPLACE.OBJ wird nur von CPU-Baugruppen der CS/CJ-Serie ab Version 2.0 unterstützt.



5-2-5 Automatische Übertragung beim Start

Mit der automatischen Übertragung beim Start werden beim Einschalten der Versorgungsspannung das Anwenderprogramm, die Parameter und die E/A-Speicherdaten von einer Speicherkarte in die CPU-Baugruppe geschrieben. Folgende Dateien können automatisch in den Speicher der CPU-Baugruppe gelesen werden.

Hinweis Diese Funktion kann nicht für das Lesen des EM-Dateispeichers verwendet werden.

Der Name der Programmdatei hängt davon ab, ob auch eine Parameterbereichsdatei übertragen wird.

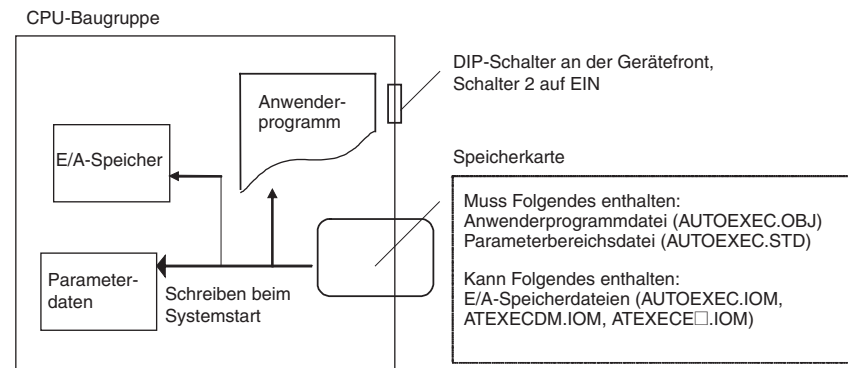
Verwenden Sie folgende Dateinamen.

Programmdatei: AUTOEXEC.OBJ

Parameterbereichsdatei: AUTOEXEC.STD

Datendateien: AUTOEXEC.IOM, ATEXECMD.IOM, ATEXECE□.IOM

Übertragung einer Parameterbereichsdatei



Datei	Dateiname	Beim Startvorgang	Für automatische Übertragung erforderlich
Programmdatei	AUTOEXEC.OBJ	Der Inhalt dieser Datei wird automatisch übertragen und überschreibt das gesamte Anwenderprogramm einschließlich der Taskattribute der CPU-Baugruppe.	Auf Speicherkarte erforderlich.
Parameterbereichsdatei	AUTOEXEC.STD	Der Inhalt dieser Datei wird automatisch übertragen und überschreibt alle Grundeinstellungsdaten der CPU-Baugruppe.	Auf Speicherkarte erforderlich.
Datendatei	AUTOEXEC.IOM	DM-Worte, die Spezial-E/A-Baugruppen, CPU-Bus-Baugruppen und internen Steckmodulen (nur CS-Serie) zugeordnet sind. Der Inhalt dieser Datei wird beim Einschalten der Spannungsversorgung, beginnend bei D20000, automatisch in den DM-Bereich übertragen. (siehe Hinweis 1)	Nicht auf Speicherkarte erforderlich.
	ATEXECMD.IOM	Universell nutzbare DM-Worte Der Inhalt dieser Datei wird beim Einschalten der Spannungsversorgung, beginnend bei D00000, automatisch in den DM-Bereich übertragen. (Nicht unterstützt von CS1 CPU-Baugruppen der CS-Serie vor EV1) (siehe Hinweis 1)	
	ATEXECE□.IOM	Universell nutzbare DM-Worte Der Inhalt dieser Datei wird beim Einschalten der Spannungsversorgung, beginnend bei E□_00000, automatisch in den EM-Bereich übertragen. (Nicht unterstützt von CS1 CPU-Baugruppen der CS-Serie vor EV1)	

- Hinweis**
- Wenn sich die Daten in AUTOEXEC.IOM und ATEXECMD.IOM überschneiden, werden alle von AUTOEXEC.IOM übertragene Daten von den Daten in ATEXECMD.IOM überschrieben, da ATEXECMD.IOM später geschrieben wird.
 - Die Programmdatei (AUTOEXEC.OBJ) und die Parameterdatei (AUTOEXEC.STD) müssen auf der Speicherkarte vorhanden sein. Ist dies nicht der Fall, schlägt die automatische Übertragung fehl, ein Speicherfeh-

- ler tritt ein, und A40115 (Speicherfehlermerker: Schwerwiegender Fehler) wird auf EIN gesetzt. (Die E/A-Speicherdatei (AUTOEXEC.IOM) muss nicht vorhanden sein.)
- Die Dateien AUTOEXEC.IOM, ATEXECMD.IOM und ATEXECE□.IOM können mittels eines Programmiergeräts (Programmierkonsole oder CX-Programmer) auch mit von D20000, D00000 und E□_00000 abweichenden Anfangsadressen erstellt werden. Die Daten werden in jedem Fall beginnend mit der richtigen Anfangsadresse geschrieben, geben Sie jedoch keine anderen Anfangsadressen an.
 - Wenn Schalter 7 des DIP-Schalterblocks auf EIN und Schalter 8 auf AUS gestellt ist, um die Easy-Backup-Funktion auszuführen, hat diese Funktion Vorrang, auch wenn Schalter 2 ebenfalls auf EIN gestellt ist. In diesem Fall werden die BACKUP□□-Dateien in die CPU-Baugruppen übertragen, nicht jedoch die Dateien zur automatischen Übertragung beim Start. (Nicht unterstützt von CS1 CPU-Baugruppen der CS-Serie vor EV1.)
 - Die Funktion zum automatischen Übertragen beim Startvorgang kann gemeinsam mit der Programmaustauschfunktion verwendet werden. Das Austausch-Startbit (A65015) kann von dem Programm auf EIN gesetzt werden, das beim Hochladen automatisch übertragen wird, um es durch ein anderes Programm zu ersetzen.
 - Die Funktion zum automatischen Übertragen beim Starten und die Programmaustauschfunktion, die Bits des Zusatz-Systembereichs verwendet, können gemeinsam ausgeführt werden, d.h., das Programm, das beim Startvorgang automatisch an die CPU-Baugruppe übertragen wird, kann Programmierungen enthalten, mit denen die Bits des Zusatz-Systembereichs manipuliert werden, um einen Programmaustausch zu bestimmen.

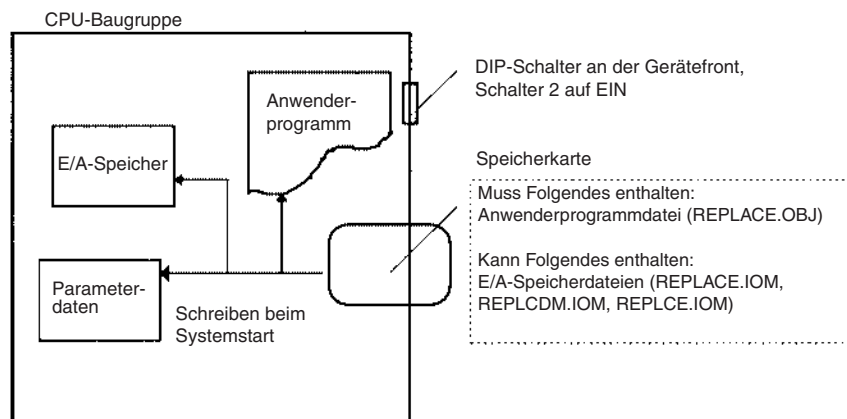
**Übertragen ohne
Parameterbereichsdatei
(nur CPU-Baugruppe der
CS/CJ-Serie ab Version
2.0.)**

Verwenden Sie folgende Dateinamen.

Programmdatei: REPLACE.OBJ

Parameterbereichsdatei: Ist nicht erforderlich und wird unabhängig vom Dateinamen nicht übertragen.

Datendateien: REPLACE.IOM, REPLCDM.IOM, REPLCE□.IOM



Datei	Dateiname	Beim Startvorgang	Für automatische Übertragung erforderlich
Programmdatei	REPLACE.OBJ Hinweis: Nur CS/CJ-Serie CPU-Baugruppen ab Version 2.0.	Der Inhalt dieser Datei wird automatisch übertragen und überschreibt das gesamte Anwenderprogramm einschließlich der Taskattribute der CPU-Baugruppe.	Auf Speicherkarte erforderlich.
Parameterbereichsdatei		Wird unabhängig vom Dateinamen nicht übertragen.	Nicht erforderlich

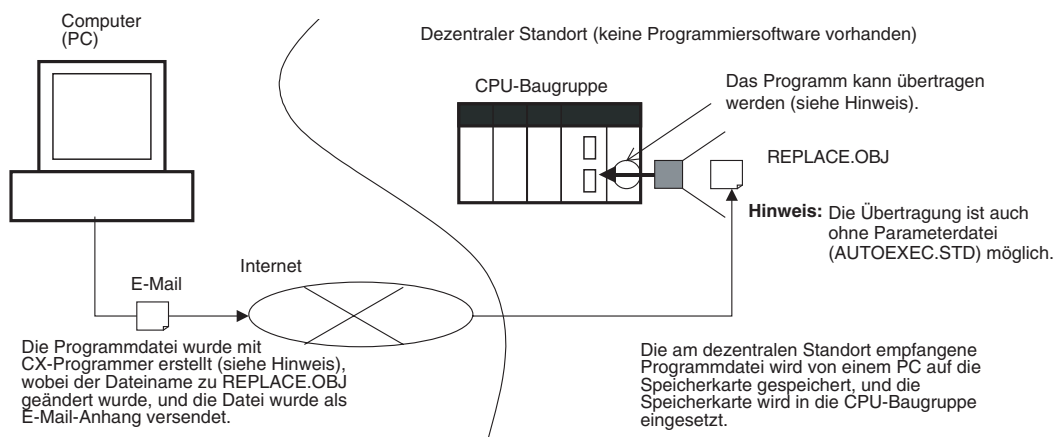
Datei	Dateiname	Beim Startvorgang	Für automatische Übertragung erforderlich
Datendatei	REPLACE.IOM Hinweis: Nur CS/CJ-Serie CPU-Baugruppen ab Version 2.0.	DM-Worte, die Spezial-E/A-Baugruppen, CPU-Bus-Baugruppen und internen Steckmodulen (nur CS-Serie) zugeordnet sind. Der Inhalt dieser Datei wird beim Einschalten der Spannungsversorgung, beginnend bei D20000, automatisch in den DM-Bereich übertragen.	Nicht auf Speicherkarte erforderlich.
	REPLCDM.IOM Hinweis: Nur CS/CJ-Serie CPU-Baugruppen ab Version 2.0.	Universell nutzbare DM-Worte Der Inhalt dieser Datei wird beim Einschalten der Spannungsversorgung, beginnend bei D00000, automatisch in den DM-Bereich übertragen.	
	REPLCE□.IOM Hinweis: Nur CS/CJ-Serie CPU-Baugruppen ab Version 2.0.	Universell nutzbare DM-Worte Der Inhalt dieser Datei wird beim Einschalten der Spannungsversorgung, beginnend bei E□_00000, automatisch in den EM-Bereich übertragen. □ gibt die Banknummer an.	

Hinweis

1. Wenn der Programmdateiname REPLACE.OBJ lautet (nur CS/CJ-Serie CPU-Baugruppe ab Version 2.0), wird die Parameterbereichsdatei unabhängig von ihrem Namen nicht übertragen, auch wenn sie sich auf der Speicherkarte befindet.
2. Wenn die Dateien REPLACE.IOM, REPLCDM.IOM oder REPLCE□.IOM mittels eines Programmiergeräts (Programmierkonsole oder CX-Programmer) erstellt werden, spezifizieren Sie immer die korrekte Anfangsadresse (D20000, D00000 oder E□_00000). Der Inhalt der Datei wird immer von der ordnungsgemäßen ersten Adresse (D20000, D00000 oder E□_00000) an übertragen, auch wenn ein anderes Anfangswort spezifiziert wird, was zu einem ungewollten Überschreiben des Inhalts eines Teils des DM-Bereichs bzw. der EM-Bank führen kann.
3. Wenn Schalter 7 des DIP-Schalterblocks auf EIN und Schalter 8 auf AUS gestellt ist, um die Easy-Backup-Funktion auszuführen, hat diese Funktion Vorrang, auch wenn Schalter 2 ebenfalls auf EIN gestellt ist. In diesem Fall werden die BACKUP□□-Dateien in die CPU-Baugruppen übertragen, nicht jedoch die Dateien zur automatischen Übertragung beim Start.

Anwendungsbeispiel**Automatisches Übertragen von Dateien ohne Parameterbereichsdatei**

Eine Programmdatei (.OBJ) kann offline in einem Büro erstellt und ohne Parameterbereichsdatei (.STD) an einen anderen Ort übertragen werden. Die Programmdatei kann an diesem entfernten Ort ohne Programmiergerät auf einer Speicherkarte gespeichert werden, und die Speicherkarte kann dafür verwendet werden, das Programm beim Starten automatisch zur CPU-Baugruppe zu übertragen.



Unterstützte Kombinationen bei der Datenübertragung

In den folgenden Tabellen ist aufgelistet, ob Dateien beim Starten automatisch zur CPU-Baugruppe übertragen werden, abhängig davon, welche Dateien auf der Speicherkarte vorhanden sind.

■ Programmdatei: AUTOEXEC.OBJ

Programmdatei	Parameterbereichsdatei	Datendateien	Übertragen/ Nicht übertragen
AUTOEXEC.OBJ	AUTOEXEC.STD	Eine oder mehrere der Folgenden: AUTOEXEC.IOM, ATEXECDM.IOM, ATEXECE□.IOM	Übertragen
		Keine	
	Keine	Eine oder mehrere der Folgenden: AUTOEXEC.IOM, ATEXECDM.IOM, ATEXECE□.IOM	Nicht übertragen
		Keine	

■ Programmdatei: REPLACE.OBJ

Programmdatei	Parameterbereichsdatei	Datendateien	Übertragen/ Nicht übertragen
REPLACE.OBJ	Vorhanden	Eine oder mehrere der Folgenden: REPLACE.IOM, REPLCDM.IOM, REPLCE□.IOM	Übertragen, jedoch nicht die Parameterbereichsdatei
		Keine	
	Keine	Eine oder mehrere der Folgenden: REPLACE.IOM, REPLCDM.IOM, REPLCE□.IOM	Übertragen
		Keine	

■ Keine Programmdatei

Programmdatei	Parameterbereichsdatei	Datendateien	Übertragen/ Nicht übertragen
Keine	AUTOEXEC.STD	Eine oder mehrere der Folgenden: AUTOEXEC.IOM, ATEXECDM.IOM, ATEXECE□.IOM	Nicht übertragen
		Eine oder mehrere der Folgenden: REPLACE.IOM, REPLCDM.IOM, REPLCE□.IOM	
		Keine	
	Keine	Eine oder mehrere der Folgenden: AUTOEXEC.IOM, ATEXECDM.IOM, ATEXECE□.IOM	
		Eine oder mehrere der Folgenden: REPLACE.IOM, REPLCDM.IOM, REPLCE□.IOM	
		Keine	

■ Sowohl die AUTOEXEC- als auch die REPLACE-Datei

Mehr als eine Programmdatei

Programmdateien		Parameterbereichsdatei	Datendateien	Übertragen/ Nicht übertragen
AUTOEXEC.OBJ	REPLACE.OBJ	AUTOEXEC.STD	Eine oder mehrere der Folgenden: AUTOEXEC.IOM, ATEXECDM.IOM, ATEXECE□.IOM	Nicht übertragen
			Eine oder mehrere der Folgenden: REPLACE.IOM, REPLCDM.IOM, REPLCE□.IOM	
			Keine	
		Keine	Eine oder mehrere der Folgenden: AUTOEXEC.IOM, ATEXECDM.IOM, ATEXECE□.IOM	
			Eine oder mehrere der Folgenden: REPLACE.IOM, REPLCDM.IOM, REPLCE□.IOM	
			Keine	

Mehr als ein Datendateityp

Programmdatei	Parameterbereichsdatei	Datendateien		Übertragen/ Nicht übertragen
AUTOEXEC.OBJ	AUTOEXEC.STD	Eine oder mehrere der Folgenden: AUTOEXEC.IOM, ATEXECMD.IOM, ATEXECE□.IOM	Mindestens eine der Folgenden: REPLACE.IOM, REPLCDM.IOM, REPLCE□.IOM	Folgende Datendateien werden übertragen: AUTOEXEC.IOM, ATEXECMD.IOM, ATEXECE□.IOM
	Keine			Nicht übertragen
REPLACE.OBJ	Ignoriert			Folgende Datendateien werden übertragen: REPLACE.IOM, REPLCDM.IOM, REPLCE□.IOM

Vorgehensweise**1,2,3...**

1. Schalten Sie die Spannungsversorgung der SPS AUS.
2. Stellen Sie Schalter 2 des DIP-Schalterblocks an der Vorderseite der CPU-Baugruppe auf EIN. Stellen Sie sicher, dass sowohl Schalter 7 als auch Schalter 8 auf AUS gestellt sind.

Hinweis Die Easy-Backup-Funktion hat Vorrang vor der Funktion der automatischen Übertragung beim Start. Achten Sie daher darauf, dass die Schalter 7 und 8 auf AUS gestellt sind.

3. Bereiten Sie eine Speicherkarte wie folgt vor:
 - a) Übertragung mit Parameterbereichsdatei
Setzen Sie eine Speicherkarte ein, die die mit CX-Programmer erstellte Anwenderprogrammdatei (AUTOEXEC.OBJ), die Parameterbereichsdatei (AUTOEXEC.STD) und/oder die E/A-Speicherdateien (AUTOEXEC.IOM, ATEXECMD.IOM und ATEXECE□.IOM) enthält. (Die Programmdatei und die Parameterbereichsdatei müssen sich auf der Speicherkarte befinden. Die E/A-Speicherdateien sind optional.)
 - b) Übertragung ohne Parameterbereichsdatei
Setzen Sie eine Speicherkarte ein, die die mit CX-Programmer erstellte Anwenderprogrammdatei (REPLACE.OBJ) und/oder die E/A-Speicherdateien (REPLACE.IOM, REPLCDM.IOM und REPLCE□.IOM) enthält. (Die Programmdatei muss sich auf der Speicherkarte befinden. Die E/A-Speicherdateien sind optional.)
4. Schalten Sie die Spannungsversorgung der SPS ein.

Hinweis**Fehler bei der automatischen Übertragung beim Start**

Wenn die automatische Übertragung beim Start fehlschlägt, tritt ein Speicherfehler auf, A40115 wird auf EIN gesetzt, und die CPU-Baugruppe wird angehalten. Schalten Sie die Spannungsversorgung beim Auftreten eines Fehlers aus, um den Fehler zu löschen. (Der Fehler kann nicht ohne die Spannungsversorgung auszuschalten gelöscht werden.)

DIP-Schalter an der Gerätefront der CPU-Baugruppe

Schalter	Bezeichnung	Einstellung
2	Schalter für automatische Übertragung beim Start	EIN: Automatische Übertragung beim Start ausführen. AUS: Automatische Übertragung beim Start nicht ausführen.
7 und 8	Easy-Backup-Schalter	Stellen Sie beide Schalter auf AUS.

Zugehörige Bits/Worte im Zusatz-Systembereich

Bezeichnung	Adresse	Einstellung
Speicherfehlermerker (Schwerwiegender Fehler)	A40115	EIN, wenn ein Speicherfehler oder ein Fehler beim automatischen Übertragen von der Speicherkarte beim Start der SPS auftrat (automatische Übertragung beim Start). Die CPU-Baugruppe wird angehalten, und die ERR/ALM-Anzeige an der Gerätefront der CPU-Baugruppe leuchtet. Hinweis: A40309 wird auf EIN gesetzt, wenn der Fehler bei der automatischen Übertragung beim Start auftrat. (In diesem Fall kann der Fehler nicht gelöscht werden.)
Speicherkarten-Startübertragungs-Fehlermerker	A40309	EIN, wenn die automatische Übertragung beim Start ausgewählt wurde und bei der automatischen Übertragung ein Fehler auftritt (Schalter 2 des DIP-Schalterblocks auf EIN). Es kommt zu einem Fehler, wenn ein Übertragungsfehler auftritt, die angegebene Datei nicht vorhanden oder die Speicherkarte nicht eingesetzt ist. Hinweis: Der Fehler kann gelöscht werden, indem die Spannungsversorgung ausgeschaltet wird. (Solange die Spannungsversorgung eingeschaltet ist, kann der Fehler nicht gelöscht werden.)

Spezieller Hinweis zu möglichen Änderungen des E/A-Zuordnungsmodus bei der automatischen Übertragung beim Start

Werden die Parameterdateien für die automatische Übertragung beim Start auf einer CJ-Serie CPU-Baugruppe erstellt und für die Verwendung mit einer anderen CJ-Serie CPU-Baugruppe auf einer Speicherkarte gespeichert, hängt der E/A-Zuordnungsmodus von den Baugruppenversionen der Quell- und Ziel-CPU-Baugruppe ab. Die nachstehende Tabelle zeigt die möglichen Änderungen des E/A-Zuordnungsmodus für die verschiedenen Kombinationen von Baugruppenversionen.

Quell-CPU-Baugruppe		Ursprünglicher E/A-Zuordnungsmodus	Ziel-CPU-Baugruppe		
			Baugruppenversion der CPU-Baugruppe, in die die beim Start automatisch zu übertragenden Dateien übertragen werden		
			bis einschließlich Ver. 2.0	Baugruppenversion 2.0	ab Baugruppenversion 3.0
Baugruppenversion der CPU-Baugruppe, in der die beim Start automatisch zu übertragenden Dateien erzeugt wurden	bis einschließlich Ver. 2.0	Automatische Speicherzuordnung	Wechsel zu benutzerdefiniert	(Identisch) Automatische Zuordnung	Wechsel zu benutzerdefiniert
		Benutzerdefiniert	(Identisch) Benutzerdefiniert	Wechsel zu automatischer Zuordnung (siehe Hinweis 1)	(Identisch) Benutzerdefiniert
	Baugruppenversion 2.0	Automatische Zuordnung	Wechsel zu benutzerdefiniert	(Identisch) Automatische Zuordnung	Wechsel zu benutzerdefiniert
		Benutzerdefiniert	(Identisch) Benutzerdefiniert	(Identisch) Benutzerdefiniert	(Identisch) Benutzerdefiniert
	ab Baugruppenversion 3.0	Automatische Zuordnung	Wechsel zu benutzerdefiniert	(Identisch) Automatische Zuordnung	(Identisch) Automatische Zuordnung
		Benutzerdefiniert	(Identisch) Benutzerdefiniert	(Identisch) Benutzerdefiniert	(Identisch) Benutzerdefiniert

Hinweis

1. Werden die Dateien für die automatische Übertragung beim Start (AUTOEXEC.STD) mittels einer CJ-Serie CPU-Baugruppe vor Baugruppenversion 2.0 erstellt und auf Speicherkarte gespeichert, erfolgt bei der automatischen Übertragung von der Speicherkarte automatisch ein Wechsel von einer möglichen benutzerdefinierten E/A-Zuordnung zur automatischen E/A-Zuordnung.
2. Werden die Dateien für die automatische Übertragung beim Start (AUTOEXEC.STD) mittels einer CJ-Serie CPU-Baugruppe Version 2.0 erstellt und auf Speicherkarte gespeichert, erfolgt bei der automatischen Übertra-

gung von der Speicherkarte in eine CJ-Serie CPU-Baugruppe ab Version 3.0 automatisch ein Wechsel zur benutzerdefinierten E/A-Zuordnung.

Spezieller Hinweis zur Verifizierung der Parameterdateien für die automatische Übertragung beim Start

Beim Dateivergleich von Parameterdaten vor und nach dem Übertragen können an der Programmierkonsole Verifizierungsfehler auftreten, wenn die Parameterdateien für die automatische Übertragung beim Start (AUTOEXEC.STD) in einer CJ-Serie CPU-Baugruppe erstellt und in einer CJ-Serie CPU-Baugruppe einer anderen Baugruppenversion verwendet werden.

Quell-CPU-Baugruppe		Ursprünglicher E/A-Zuordnungsmodus	Ziel-CPU-Baugruppe		
			Baugruppenversion der CPU-Baugruppe, in die die Dateien automatisch übertragen werden		
			bis einschließlich Ver. 2.0	ab Baugruppenversion 2.0	ab Baugruppenversion 3.0
Baugruppenversion der CPU-Baugruppe, auf der die Dateien erstellt wurden	bis einschließlich Ver. 2.0	Automatische Zuordnung	Verifizierung möglich	Verifizierung möglich	Verifizierungsfehler
		Benutzerdefiniert			Verifizierungsfehler
	ab Baugruppenversion 2.0	Automatische Zuordnung			Verifizierungsfehler
		Benutzerdefiniert			Verifizierung möglich
	ab Baugruppenversion 3.0	Automatische Zuordnung		Verifizierungsfehler	Verifizierung möglich
		Benutzerdefiniert		Verifizierung möglich	

5-2-6 Easy-Backup-Funktion

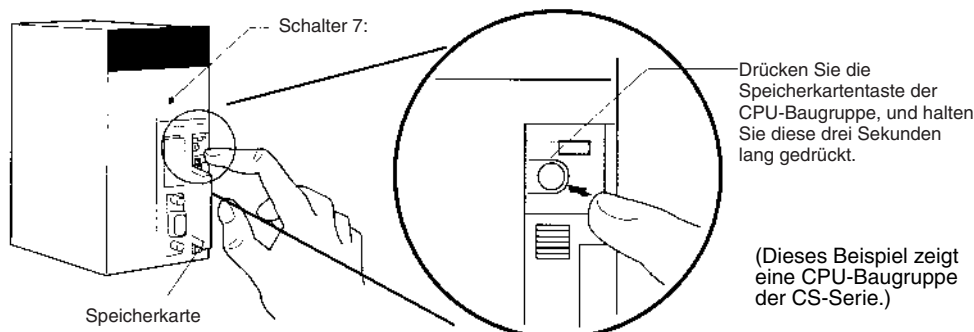
Diese Funktion wird von CS-Serie CS1 CPU-Baugruppen vor Version EV1 nicht unterstützt.

Erstellen einer Datensicherung der CPU-Baugruppe auf der Speicherkarte

Schalten Sie zum Sichern von Daten Schalter 7 des DIP-Schalterblocks der CPU-Baugruppe auf EIN. Drücken Sie dann die Speicherkartentaste, und halten Sie diese drei Sekunden lang gedrückt. Die Sicherungsfunktion erstellt automatisch Sicherungsdateien mit festen, vordefinierten Dateinamen und schreibt diese in die Speicherkarte. Die Sicherungsdateien beinhalten das Programm, die Parameterbereichsdaten, die E/A-Speicherdaten, die Symboltabellen (siehe Hinweis), die Kommentardateien (siehe Hinweis) und die Programmindexdateien (siehe Hinweis). Diese Funktion kann in jeder Betriebsart ausgeführt werden.

Hinweis

Diese Daten werden erst von CS/CJ-Serie CPU-Baugruppen ab Baugruppenversion 3.0 unterstützt. Diese Sicherungsdateien werden automatisch aus den Dateien auf der Speicherkarte, im EM-Dateispeicher oder im Kommentarspeicher erzeugt.



Wiederherstellen von Daten von der Speicherkarte in die CPU-Baugruppe

Um die Sicherungsdateien in der CPU-Baugruppe wiederherzustellen, überprüfen Sie, dass Schalter 7 auf EIN gestellt ist, und schalten Sie die Spannungsversorgung der SPS aus und dann wieder ein. Die Sicherungsdateien mit dem Programm, den Parameterbereichsdateien und den E/A-Speicherdaten werden von der Speicherkarte in die CPU-Baugruppe gelesen.

*1 Diese Daten werden erst von CS/CJ-Serie CPU-Baugruppen ab Baugruppenversion 3.0 unterstützt.

Hinweis

1. Die Sicherungsfunktion hat Priorität gegenüber der Funktion zur automatischen Übertragung beim Start. Daher werden die Sicherungsdateien beim Einschalten der SPS in die CPU-Baugruppe geschrieben, auch wenn Schalter 2 des DIP-Schalterblocks auf EIN gestellt ist.
2. Wenn Schalter 1 des DIP-Schalterblocks auf EIN gestellt ist, werden keine Daten von der Speicherkarte in die CPU-Baugruppe gelesen (schreibgeschützter Programmspeicher).
3. Wenn die Sicherungsdateien auf der Speicherkarte von der Sicherungsfunktion gelesen werden, werden die Zustände des E/A-Speichers und der Bits, die zwangsweise gesetzt/rückgesetzt werden, gelöscht, es sei denn, im Zusatz-Systembereich und SPS-Setup werden die erforderlichen Einstellungen vorgenommen.

Wenn der E/A-Speicher-Haftmarker (A50012) auf EIN gesetzt ist, und im SPS-Setup die Einstellung zur Beibehaltung des E/A-Speicher-Haftmarkers beim Start beim Schreiben der Sicherungsdateien aktiviert ist, wird der Status der E/A-Speicherdaten beibehalten, wenn Dateien von der Speicherkarte gelesen werden.

Wenn der Zwangssetzungs-Haftmarker (A50013) auf EIN gesetzt ist, und im SPS-Setup die Einstellung zur Beibehaltung des Zwangssetzungs-Haftmarkers beim Start beim Schreiben der Sicherungsdateien aktiviert ist, wird der Status der zwangsweise gesetzten und zurückgesetzten Bits beibehalten, wenn Dateien von der Speicherkarte gelesen werden.

4. CS1-H, CJ1-H, CJ1M oder CS1D CPU-Baugruppen verbleiben nach einer Easy-Backup-Sicherung in der PROGRAM-Betriebsart und können nur durch ein Aus- und Einschalten der SPS wieder in die MONITOR- oder RUN-Betriebsart versetzt werden. Schalten Sie daher nach Durchführung der Easy-Backup-Sicherung die Spannungsversorgung der CPU-Baugruppe aus, ändern Sie die Einstellung von Schalter 7, und schalten Sie die Spannungsversorgung anschließend wieder ein.
5. Dateisicherungen können einige Sekunden bis mehrere Minuten Zeit in Anspruch nehmen. Weitere Informationen über Ausführungszeiten finden Sie auf Seite 252.

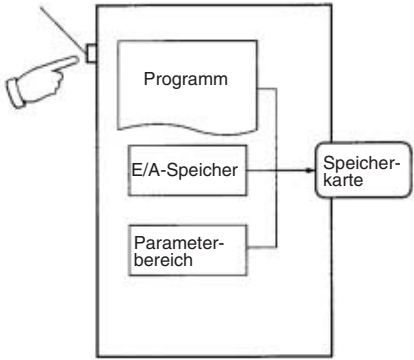
Vergleichen von Daten auf der Speicherkarte und in der CPU-Baugruppe

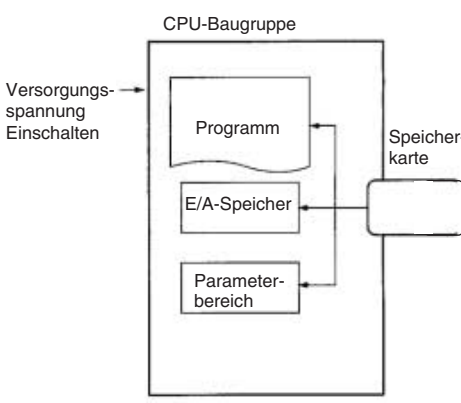
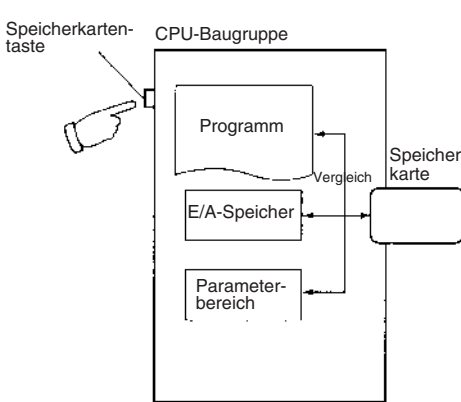
Schalten Sie zum Vergleichen der Sicherungsdateien in der Speicherkarte mit den Daten in der CPU-Baugruppe Schalter 7 des DIP-Schalterblocks der CPU-Baugruppe auf AUS, drücken Sie die Speicherkartentaste; und halten Sie diese drei Sekunden lang gedrückt. Die Sicherungsfunktion vergleicht das Programm, die Parameterbereichsdateien, die E/A-Speicherdaten, die Symboltabellen (siehe Hinweis), die Kommentardateien (siehe Hinweis) und die Programmindexdateien (siehe Hinweis) auf der Speicherkarte mit den entsprechenden Daten in der CPU-Baugruppe. Diese Funktion kann in jeder Betriebsart ausgeführt werden.

Hinweis

Diese Daten werden erst von CS/CJ-Serie CPU-Baugruppen ab Baugruppenversion 3.0 unterstützt.

Die folgende Tabelle enthält eine Zusammenfassung der Easy-Backup-Funktionen.

Sicherungsvorgang	Schalter- stellung	Vorgehensweise
	Schalter 7	
Erstellen einer Datensicherung der CPU-Baugruppe auf der Speicherkarte Sichern von Daten auf der Speicherkarte Speicherkarten-Taste der CPU-Baugruppe  Schalter 7: EIN	EIN	Drücken Sie die Speicherkartentaste der CPU-Baugruppe, und halten Sie diese drei Sekunden lang gedrückt.

Sicherungsvorgang	Schalter- stellung	Vorgehensweise
	Schalter 7	
Wiederherstellen der Daten von der Speicherkarte in die CPU-Baugruppe Wiederherstellen von Daten von der Speicherkarte  Schalter 7: EIN	EIN	Schalten Sie die Versorgungsspannung der SPS aus und wieder ein. (siehe Hinweis 1)
Vergleichen der Daten zwischen CPU-Baugruppe und Speicherkarte Vergleich mit Daten auf der Speicherkarte  Schalter 7: AUS	AUS	Drücken Sie die Speicherkartentaste der CPU-Baugruppe, und halten Sie diese drei Sekunden lang gedrückt.

Hinweis

1. Weitere Informationen über die Ergebnisse der Lese-, Schreib- und Vergleichsvorgänge finden Sie unter *Überprüfen von Sicherungsvorgängen mit Hilfe der Anzeigen* auf Seite 240.
2. Richtlinien für die Zeiten, die für Sicherungsvorgänge auf der Speicherkarte erforderlich sind, finden Sie unter *5-3-2 Bedienung*.

Sicherungsdateien**Datendateien**

Dateiname und Namens- erweiterung	Datenbereich und gespeicherter Adressenbereich		Sicherung aus dem E/A-Speicher auf Speicherkarte (Erstellen von Dateien)	Lesen von der Speicherkarte in die CPU-Baugruppe	Vergleichen der Speicherkarte mit der CPU-Baugruppe		Für die Datenwiederherstellung erforderliche Dateien
CPU-Baugruppe	CS/CJ				CS1/CJ1	CS1-H/CJ1-H	
BACKUP.IOM	DM	D20000 bis D32767	Ja	Ja	Ja	---	Auf Speicherkarte erforderlich
BACKUPIO.IOR	CIO	0000 bis 6143 (einschließlich Bit-Zwangssetzungsstatus)	Ja	--- ⁴	Ja	---	Auf Speicherkarte erforderlich
	WR	W000 bis W511 (einschließlich Bit-Zwangssetzungsstatus)	Ja	--- ⁴	Ja	---	
	HR	H000 bis H511	Ja	Ja	Ja	---	
	AR	A000 bis A447	Ja	---	---	---	
		A448 bis A959	Ja	Ja	Ja	---	
	Zeitgeber	T0000 bis T4095	Ja	Ja ⁴	Ja	---	
	Zähler ¹	C0000 bis C4095	Ja	Ja	Ja	---	
BACKUPDM.IOM	DM	D00000 bis D19999	Ja	Ja	Ja	---	Auf Speicherkarte erforderlich
BACKUPE□.IOM ^{2, 3}	EM	E□_00000 bis E□_32767	Ja	Ja	Ja	---	Auf Speicherkarte erforderlich (muss mit CPU-Baugruppe übereinstimmen)

Hinweis

1. Die Ablaufmerker und die Istwerte werden gesichert.
2. Das □ bezeichnet die Banknummer, und die Anzahl der Bänke ist abhängig von der verwendeten CPU-Baugruppe.
Bei der Wiederherstellung der BACKUPE□.IOM-Dateien von der Speicherkarte in die CPU-Baugruppe werden die Dateien in der Reihenfolge, beginnend mit Bank 0 und mit der höchsten Banknummer in der CPU-Baugruppe endend, gelesen. Überschüssige BACKUPE□.IOM-Dateien werden nicht gelesen, wenn die Anzahl der gesicherten Bänke die Anzahl der Bänke in der CPU-Baugruppe übertrifft. Umgekehrt werden die restlichen EM-Bänke in der CPU-Baugruppe nicht geändert, wenn die Anzahl der gespeicherten Bänke geringer ist als die Anzahl der Bänke in der CPU-Baugruppe.
Wenn eine BACKUPE□.IOM-Datei fehlt (z. B.: 0, 1, 2, 4, 5, 6) werden nur die direkt aufeinander folgenden Dateien gelesen. In diesem Fall würden nur die Daten von Bank 0, 1 und 2 gelesen.
3. Die Daten des EM-Bereichs werden als Binärdaten gesichert. EM-Bänke, die zu Dateispeicher umgewandelt wurden, werden zusammen mit den EM-Bänken gesichert, die nicht umgewandelt wurden.
Ein EM-Dateispeicher kann nur im EM-Bereich einer anderen CPU-Baugruppe wiederhergestellt werden, wenn die BACKUPE□.IOM-Dateien direkt aufeinander folgen und die Anzahl der gesicherten EM-Bänke mit der Anzahl der Bänke in der CPU-Baugruppe übereinstimmt. Wenn die BACKUPE□.IOM-Dateien nicht aufeinander folgen oder die Anzahl der EM-Bänke nicht mit der Anzahl der Bänke in der CPU-Baugruppe übereinstimmt, wird der EM-Dateispeicher wieder in sein unformatiertes Format zurückgesetzt und die Dateien im Dateispeicher werden ungültig. (Die regulären Bänke des EM-Bereichs werden normal gelesen.)

4. Normalerweise werden die Daten des CIO-Bereichs, des WR-Bereichs, die Zeitgeber-Ablaufmerker und die Zeitgeber-Istwerte sowie der Status der Bits, die zwangsweise gesetzt/rückgesetzt werden, gelöscht, wenn die SPS eingeschaltet wird und die Datei BACKUPIO.IOR von der Speicherkarte gelesen wird.

Wenn der E/A-Speicher-Haftmerker (A50012) auf EIN gesetzt ist, und im SPS-Setup die Einstellung zur Beibehaltung des E/A-Speicher-Haftmerkers beim Start beim Schreiben der Sicherungsdateien aktiviert ist, wird der Status der E/A-Speicherdaten beibehalten, wenn Dateien von der Speicherkarte gelesen werden.

Wenn der Zwangssetzungs-Haftmerker (A50013) auf EIN gesetzt ist, und im SPS-Setup die Einstellung zur Beibehaltung des Zwangssetzungs-Haftmerkers beim Start beim Schreiben der Sicherungsdateien aktiviert ist, wird der Status der zwangsweise gesetzten und zurückgesetzten Bits beibehalten, wenn Dateien von der Speicherkarte gelesen werden.

Programmdateien

Dateiname und Namens-erweiterung	Inhalt	Sicherung aus dem E/A-Speicher auf Speicherkarte (Erstellen von Dateien)	Lesen von der Speicherkarte in die CPU-Baugruppe	Vergleichen der Speicherkarte mit der CPU-Baugruppe	Für die Datenwiederherstellung erforderliche Dateien
CPU-Baugruppe	CS/CJ				
BACKUP.OBJ	Gesamtes Anwenderprogramm	Ja	Ja	Ja	Auf Speicherkarte erforderlich

Parameterdateien

Dateiname und Namens-erweiterung	Inhalt	Sicherung aus dem E/A-Speicher auf Speicherkarte (Erstellen von Dateien)	Lesen von der Speicherkarte in die CPU-Baugruppe	Vergleichen der Speicherkarte mit der CPU-Baugruppe	Für die Datenwiederherstellung erforderliche Dateien
CPU-Baugruppe	CS/CJ				
BACKUP.STD	SPS-Setup Registrierte E/A-Tabellen Routing-Tabellen Setup für CPU-Bus-Baugruppe usw.	Ja	Ja	Ja	Auf Speicherkarte erforderlich

Sicherungsdateien für Baugruppen/Module (Nur CS1-H, CS1D, CJ1-H, CJ1M oder CS1D CPU-Baugruppen)

Dateiname und Namens-erweiterung	Inhalt	Sicherung aus dem E/A-Speicher auf Speicherkarte (Erstellen von Dateien)	Lesen von der Speicherkarte in die CPU-Baugruppe	Vergleichen der Speicherkarte mit der CPU-Baugruppe	Für die Datenwiederherstellung erforderliche Dateien
CPU-Baugruppe	Nur CS1-H, CS1D, CJ1-H, CJ1M oder CS1D CPU-Baugruppen				
BACKUP□□.PRM (wobei □□ für die Baugruppenadresse der/des zu sichernden Baugruppe/ Moduls steht.)	Sicherungsdaten von der Baugruppe oder des Moduls mit der spezifizierten Baugruppenadresse (Der angegebene Inhalt ist abhängig von Baugruppe oder Modul.)	Ja	Ja	Ja	Auf Speicherkarte erforderlich (siehe Hinweis 2)

Hinweis

1. Baugruppenadressen werden wie folgt angegeben:
CPU-Bus-Baugruppen: Baugruppennummer + 10 Hex
Spezial-E/A-Baugruppen: Baugruppennummer + 20 Hex
Internes Modul: E1 Hex
2. In der CPU-Baugruppe tritt kein Fehler auf, wenn diese Datei bei der Übertragung von Daten von der Speicherkarte auf die E/A-Speicherkarte fehlt. In der Baugruppe oder dem Modul tritt jedoch ein Fehler auf, wenn die Daten nicht wiederhergestellt werden. Weitere Informationen zu Baugruppen- oder Modulfehlern finden Sie im Bedienerhandbuch der jeweiligen Baugruppe bzw. des jeweiligen Moduls.

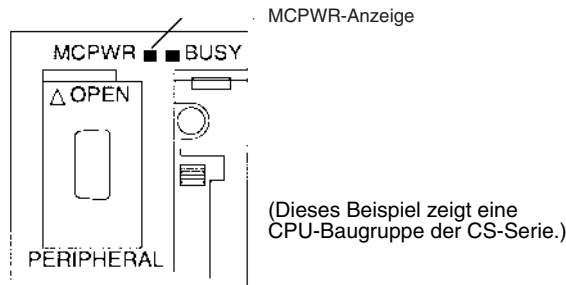
Symoltabellen, Kommentardateien, Programmindexdateien (nur CS1-H, CJ1-H, CJ1M und CS1D CPU-Baugruppen ab Baugruppenversion 3.0)

Dateiname und Namens-erweiterung	Inhalt	Sicherung aus der CPU-Bau-Gruppe auf Speicherkarte (Erstellen von Dateien)	Lesen von der Speicherkarte in die CPU-Baugruppe	Vergleichen der Speicherkarte mit der CPU-Bau-Gruppe	Für die Datenwiederherstellung erforderliche Dateien
CPU-Baugruppe	nur CS1-H, CJ1-H, CJ1M und CS1D CPU-Baugruppen ab Baugruppenversion 3.0				
BKUPSYM.SYM	Symoltabellendateien	Ja (Schreiben, wenn der Kommentarspeicher der CPU-Baugruppe Dateien enthält)	Ja (Lesen in den Kommentarspeicher der CPU-Baugruppe, wenn die Speicherkarte Dateien enthält)	Ja (Vergleichen mit den Dateien im Kommentarspeicher der CPU-Baugruppe)	Diese Dateien sind nicht unbedingt auf der Speicherkarte erforderlich.
BKUPCMT.CMT	Kommentardateien				
BKUPPRG.IDX	Programmindexdateien				

Hinweis Die in der obigen Tabelle aufgeführten Sicherungs-, Lese- und Vergleichsoperationen werden nur für im Kommentarspeicher enthaltene Dateien durchgeführt.

Überprüfen von Sicherungsvorgängen mit Hilfe der Anzeigen

Der Status der Speicherkarten-Betriebsanzeige (MCPWR) zeigt an, ob eine Easy-Backup-Sicherung fehlerfrei abgeschlossen wurde oder nicht.



Sicherungsvorgang	Fehlerfreier Abschluss (siehe Hinweis 1)	Fehler aufgetreten	
	MCPWR-Status	MCPWR-Status	Fehler
Erstellen einer Datensicherung der CPU-Baugruppe auf der Speicherkarte	Leuchtet → Leuchtet so lange, wie die Speicherkartentaste gedrückt wird. → Blinkt einmal → Leuchtet während des Schreibvorgangs → AUS nach Beendigung des Schreibvorgangs.	Leuchtet → Leuchtet so lange, wie die Speicherkartentaste gedrückt wird. → Blinkt fortwährend → Leuchtet so lange, wie die Speicherkartentaste gedrückt wird.	Bei den folgenden Fehlern werden keine Dateien erstellt: Unzureichender Speicherplatz auf der Speicherkarte (siehe Hinweis 2) Speicherfehler in der CPU-Baugruppe E/A-Bus-Fehler (beim Schreiben von Daten in eine Baugruppe oder ein Modul, nur bei CS1-H, CS1D oder CJ1-H CPU-Baugruppen)

Sicherungsvorgang	Fehlerfreier Abschluss (siehe Hinweis 1)	Fehler aufgetreten	
	MCPWR-Status	MCPWR-Status	Fehler
Wiederherstellen der Daten von der Speicherkarte in die CPU-Baugruppe	Leuchtet, wenn die Spannungsversorgung eingeschaltet ist. → Blinkt einmal → Leuchtet während des Lesevorgangs → AUS nach Beendigung des Lesevorgangs.	Leuchtet, wenn die Spannungsversorgung eingeschaltet ist. → Blinkt fünf Mal → Erlischt.	Bei den folgenden Fehlern können Daten nicht gelesen werden: Programm auf der Speicherkarte übersteigt die Speicherkapazität der CPU-Baugruppe. Die erforderlichen Sicherungsdateien sind nicht auf der Speicherkarte vorhanden. Das Programm kann nicht geschrieben werden, da es schreibgeschützt ist (Schalter 1 des DIP-Schalterblocks steht auf EIN.)
		Leuchtet, wenn die Spannungsversorgung eingeschaltet ist. → Blinkt einmal → Leuchtet während des Lesevorgangs → Blinkt drei Mal → AUS nach Beendigung des Lesevorgangs.	Achtung: Bei den folgenden Fehlern können Daten gelesen werden: EM-Dateien und EM-Bänke der CPU-Baugruppen stimmen nicht überein (nicht aufeinander folgende Banknummern oder Nichtübereinstimmung der max. Anzahl an Bänken).
Vergleichen der Daten zwischen CPU-Baugruppe und Speicherkarte	Leuchtet → Leuchtet so lange, wie die Speicherkartentaste gedrückt wird. → Blinkt einmal → Leuchtet während des Vergleichens → AUS nach Beendigung des Vergleichens.	Leuchtet → Leuchtet so lange, wie die Speicherkartentaste gedrückt wird. → Blinkt fortwährend → Leuchtet so lange, wie die Speicherkartentaste gedrückt wird.	Die folgenden Fehler können beim Vergleichen auftreten (Siehe Hinweis 3): Die Daten von Speicherkarte und CPU-Baugruppe stimmen nicht überein. Die erforderlichen Sicherungsdateien sind nicht auf der Speicherkarte vorhanden. EM-Dateien und EM-Bänke der CPU-Baugruppen stimmen nicht überein (nicht aufeinander folgende Banknummern oder Nichtübereinstimmung der max. Anzahl an Bänken). Speicherfehler in der CPU-Baugruppe E/A-Bus-Fehler (beim Vergleichen von Daten mit denen einer Baugruppe oder eines Moduls, nur bei CS1-H, CS1D oder CJ1-H CPU-Baugruppen)
Bei allen drei Sicherungsvorgängen	---	Lesen: Blinkt fünf Mal → Erlischt. Schreiben oder Vergleichen: Blinkt fortwährend → Leuchtet so lange, wie die Speicherkartentaste gedrückt wird.	Speicherkarten-Zugriffsfehler (Formatfehler oder Fehler beim Lesen/Schreiben)

Hinweis

1. Bei fehlerfreiem Abschluss des Sicherungsvorgangs wird die Spannungsversorgung der Speicherkarte ausgeschaltet, wenn die MCPWR-Anzeige erlischt. Wenn die Speicherkarte erneut verwendet werden soll, drücken Sie die Speicherkartentaste, um die Spannungsversorgung für die Speicherkarte einzuschalten, und führen Sie den gewünschten Vorgang aus.
2. Wenn bei einem Easy-Backup-Sicherungsvorgang die Daten von einer CS1-H, CS1D, CJ1-H, CJ1M oder CS1D CPU-Baugruppe geschrieben werden, können Fehlermeldungen über unzureichenden Speicherplatz auf der Speicherkarte in A397 (Easy-Backup-Schreibkapazität) überprüft werden. Wenn A397 nach der Ausführung des Schreibvorgangs einen anderen Wert als 0000 Hex enthält, gibt dieser Wert den auf der Speicherkarte benötigten Speicherplatz in kByte an.
3. Bei CS1-H, CS1D, CJ1-H, CJ1M oder CS1D CPU-Baugruppen werden auch die Sicherungsdateien für die Baugruppen und Module verglichen.

Zugehörige Bits/Worte im Zusatz-Systembereich

Bezeichnung	Adresse	Beschreibung
Dateispeicher-Vorgangsmerker	A34313	EIN, wenn einer der folgenden Vorgänge ausgeführt wird. AUS, wenn die Ausführung abgeschlossen ist. - Erkennung der Speicherkarte - Ausführung eines CMND-Befehls für die lokale CPU-Baugruppe - Ausführung eines FREAD- oder FWRT-Befehls - Programmaustausch mittels spezieller Steuerbits - Easy-Backup-Sicherungsvorgang Während dieser Merker auf EIN gesetzt ist, können keine Daten auf die Speicherkarte übertragen oder Inhalte der Speicherkarte überprüft werden.
EM-Dateispeicher-Anfangsbank	A344	Wenn die CPU-Baugruppe mit dem Lesen von der Speicherkarte beginnt, wird die Zuordnung von diesem Wert bestimmt. Wenn die höchste EM-Banknummer der BACKUPE□.IOM-Dateien (die höchste, aufeinander folgende Banknummer ab 0) der höchsten Banknummer der CPU-Baugruppe entspricht, wird der EM-Bereich anhand des Wertes dieses Wortes formatiert. Wenn die höchsten EM-Banknummern nicht übereinstimmen, wird der EM-Bereich wieder in sein unformatiertes Format zurückgesetzt.
Netzwerkkommunikationsbefehl-Aktiviert-Merker (nur CS1-H, CJ1-H, CJ1M oder CS1D CPU-Baugruppen) (siehe Hinweis)	A20200 bis A20207	- Wird auf AUS gesetzt, wenn ein Schreib- oder Vergleichsvorgang der Speicherkartendaten startet. - Wird auf EIN gesetzt, wenn ein Schreib- oder Vergleichsvorgang der Speicherkartendaten abgeschlossen ist. Daten von Baugruppen und Modulen können nicht geschrieben oder verglichen werden, wenn bei Beginn des Schreib- oder Vergleichsvorgangs mit der Speicherkarte alle Netzwerkkommunikationsbefehl-Aktiviert-Merker auf AUS gesetzt sind. Wird trotzdem versucht, diesen Befehl auszuführen, wird eine Fehlermeldung angezeigt.
Netzwerkkommunikations-Abschluss-Code (nur CS1-H, CJ1-H, CJ1M oder CS1D CPU-Baugruppen) (siehe Hinweis)	A203 bis A210	Enthält die Ergebnisse der Datenübertragung mit Baugruppe oder Modul, wenn Schreib- oder Vergleichsvorgänge mit der Speicherkarte ausgeführt werden.
Netzwerkkommunikations-Fehlermerker (nur CS1-H, CJ1-H, CJ1M oder CS1D CPU-Baugruppen) (siehe Hinweis)	A21900 bis A21907	- Wird auf EIN gesetzt, wenn während der Ausführung von Schreib- oder Vergleichsvorgängen mit der Speicherkarte ein Fehler bei der Datenübertragung mit der Baugruppe oder dem Modul auftritt. - Bleibt auf AUS gesetzt (oder wird auf AUS gesetzt) wenn während der Ausführung von Schreib- oder Vergleichsvorgängen mit der Speicherkarte kein Fehler bei der Datenübertragung mit der Baugruppe oder dem Modul auftritt.
Easy-Backup-Schreibkapazität (nur CS1-H, CJ1-H, CJ1M oder CS1D CPU-Baugruppen)	A397	Zeigt die Speicherkapazität in kByte an, die auf der Speicherkarte benötigt wird, wenn der Schreibvorgang für eine Easy-Backup-Sicherung fehlschlägt. Zeigt an, dass ein Schreibfehler aufgetreten ist, da nicht genug Speicherkapazität vorhanden war. 0001 bis FFFF Hex: Schreibfehler (Zeigt an, dass der notwendige Speicherplatz auf Speicherkarte zwischen 1 und 65.535 kByte beträgt.) (Wird auf 0000 Hex gesetzt, wenn der Schreibvorgang erfolgreich ausgeführt wurde.) 0000 Hex: Schreibvorgang normal abgeschlossen.

Hinweis Diese Merker für CS1-H, CJ1-H, CJ1M oder CS1D CPU-Baugruppen ähneln sich, da die CPU-Baugruppe beim Schreiben oder Vergleichen von Daten für eine Speicherkarte automatisch eine verfügbare Kommunikationsschnittstelle verwendet.

Sichern von Baugruppen- und Moduldaten

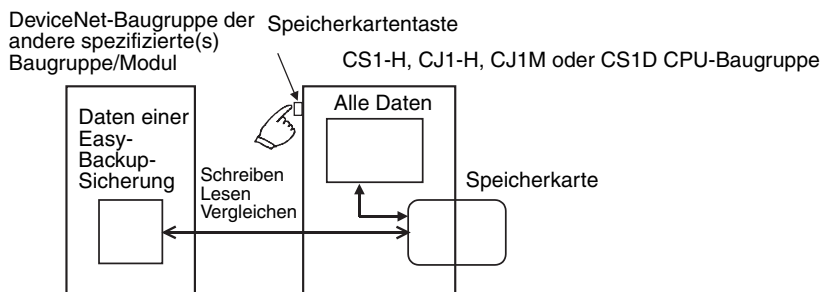
Diese Funktion wird nur von den CS1-H, CJ1-H, CJ1M und CS1D CPU-Baugruppen unterstützt.

Einführung

Bei einer Easy-Backup-Sicherung werden bei CS1 und CJ1 CPU-Baugruppen die folgenden Daten der CPU-Baugruppe gesichert: Anwenderprogramm, Parameterbereich, gesamter E/A-Speicher. Zusätzlich zu den oben genannten Daten werden bei CS1-H, CJ1-H, CJ1M oder CS1D CPU-Baugruppen außerdem die folgenden Daten gesichert: Daten von spezifizierten Baugruppen oder Modulen, die in der SPS installiert sind.

Übersicht

Bei Verwendung einer Easy-Backup-Sicherung bei einer CS1-H, CJ1-H, CJ1M oder CS1D CPU-Baugruppe wird eine Baugruppen-/Modul-Sicherungsdatei mit den Daten der spezifizierten Baugruppen und Module auf die Speicherkarte geschrieben. Die Daten werden für jede Baugruppe und jedes Modul separat gesichert.



Anwendung

Diese Funktion kann zum Speichern der Daten der vollständigen SPS verwendet werden, inklusive CPU-Baugruppe, DeviceNet-Baugruppen, serielle Kommunikationsbaugruppen/-module usw. Außerdem kann sie beim Ersetzen von Baugruppen verwendet werden.

Sicherungsdateien für Baugruppen/Module

Die Daten von jeder Baugruppe/jedem Modul werden auf der Speicherkarte unter Verwendung der folgenden Dateinamen gespeichert: BACKUP□□.PRM. wobei „□□“ für die Baugruppenadresse der/des zu sichernden Baugruppe/Moduls in Hexadezimal steht.

Hinweis

Baugruppenadressen werden wie folgt angegeben:
 CPU-Bus-Baugruppen: Baugruppennummer + 10 Hex
 Spezial-E/A-Baugruppen: Baugruppennummer + 20 Hex
 Internes Modul: E1 hex

Diese Dateien werden auch beim Lesen von der Speicherkarte oder Vergleichen mit den Daten von der Speicherkarte verwendet.

Geeignete Baugruppen und Module

Zu Sicherung von Baugruppen- und Moduldaten muss die/das Baugruppe/Modul die Sicherungsfunktion ebenfalls unterstützen. Weitere Informationen hierzu finden Sie im Bedienerhandbuch zur/zum Baugruppe/Modul.

Baugruppe/ Modul	Produkt- bezeichnungen	Durch Easy-Backup-Sicherung gespeicherte Daten bei Verwendung mit CS1-H/CJ1-H CPU-Baugruppe	Auf der Speicherkarte belegte Datenkapazität bei Easy-Backup- Sicherung
DeviceNet- Baugruppen	CJ1W-DRM21-V1 CJ1W-DRM21	Geräteparameter (alle Daten in EEPROM der Baugruppe) (Obwohl hier die gleichen Daten wie zuvor bei der Speicherkarten-Backup-Funktion gespeichert werden, die von der Baugruppe oder der Konfiguration von DeviceNet (Ver. 2.0 unterstützt werden, ist eine Kompatibilität der Dateien nicht gegeben.)	7 kByte
Serielle Kommuni- kationsbaugruppen	CS1W-SCU21-V1 CJ1W-SCU41	Protokollmakro-Daten (Dazu zählen sowohl Standardsystemprotokolle als auch anwenderdefinierte Protokolle aus dem Flash-Speicher der Baugruppe bzw. des Moduls)	129 kByte
Serielle Kommuni- kationsmodule	CS1W-SCB21-V1 CS1W-SCB41-V1		129 kByte

Baugruppe/ Modul	Produkt- bezeichnungen	Durch Easy-Backup-Sicherung gespeicherte Daten bei Verwendung mit CS1-H/CJ1-H CPU-Baugruppe	Auf der Speicherkarte belegte Datenkapazität bei Easy-Backup- Sicherung
Programmierbare Zählerbaugruppen	CS1W-HIO01-V1 CS1W-HCP22-V1 CS1W-HCA22-V1 CS1W-HCA12-V1	- Anwenderprogramm - Universell nutzbare, schreibgeschützter DM-Bereich - Baugruppenfunktion-Einstellungsbereich - Informationen zu Erweiterungsbefehlen - Programmbibliothek	64 kByte
Motion-Controller- Baugruppen	CS1W-MCH71	• Positionierungsdaten	8.192 kByte
	CS1W-MC221-V1 CS1W-MC421-V1	• Systemparameter • Programme für G-Sprachen	142 kByte
Positionier- Baugruppen	CS1W- NC113/133/213/2 33/413/433 Ver. 2.0 oder höher CJ1W- NC113/133/213/2 33/413/433 Ver. 2.0 oder höher	- Achsenparameter - Sequenzdaten - Geschwindigkeitsdaten - Daten zu Beschleunigungs-/Verzögerungszeit - Daten für Pausen-Zeiten - Zonendaten	7 kByte

Hinweis Daten von den oben aufgeführten Baugruppen und Modulen werden bei Easy-Backup-Sicherung automatisch gespeichert. Sie können diese Daten nicht durch eine Einstellung einschließen oder ausschließen. Bei Verwendung einer Programmierkonsole werden jedoch Sicherungsvorgänge für den Anwenderprogramm-Bereich, Parameter-Bereich und E/A-Speicher-Bereiche einzeln unterstützt. Weitere Informationen hierzu finden Sie im *Programmierkonsole Bedienerhandbuch (W314)*.

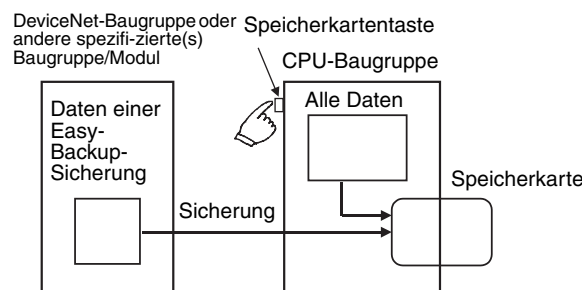
Vorgehensweise

Die Vorgehensweise bei einer Easy-Backup-Sicherung ist immer die Gleiche, unabhängig davon, ob Daten von bestimmten Baugruppen oder Modulen gesichert werden oder nicht (dazu zählt auch Schreiben, Lesen und Vergleichen).

■ Sichern von Daten

- 1,2,3...**
- Stellen Sie Schalter 7 des DIP-Schalterblocks der CPU-Baugruppe auf EIN.
 - Drücken Sie die Speicherkartentaste der CPU-Baugruppe, und halten Sie diese drei Sekunden lang gedrückt.

Die Sicherungsdaten für die Baugruppen und Module werden in einer Datei erstellt und zusammen mit anderen Sicherungsdaten auf der Speicherkarte gesichert.

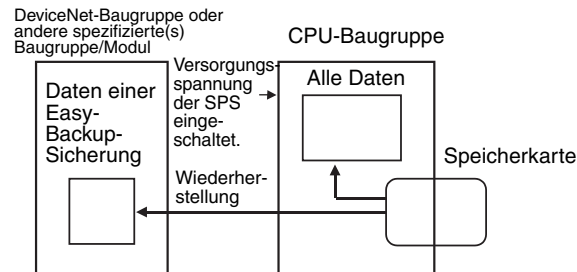


Wenn die Speicherkartentaste gedrückt wird, blinkt die MCPWR-Anzeige einmal. Während des Schreibvorgangs leuchtet sie durchgängig und sie erlischt, wenn der Schreibvorgang ohne Fehler abgeschlossen wird.

■ Wiederherstellen von Daten

- 1,2,3... 1. Stellen Sie Schalter 7 des DIP-Schalterblocks der CPU-Baugruppe auf EIN.
2. Schalten Sie die SPS ein. Die Sicherungsdateien werden in den Baugruppen und Modulen wiederhergestellt.

Die Sicherungsdaten der Baugruppen und Module werden von der Speicherkarte auf den Baugruppen und Modulen wiederhergestellt.

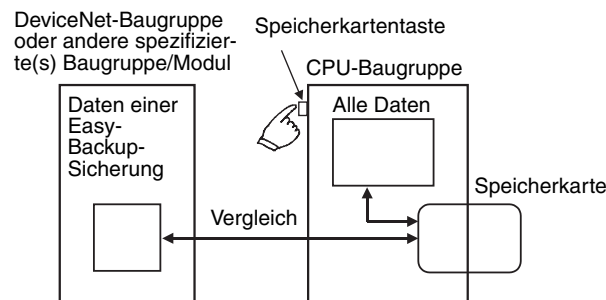


Wenn die Speicherkartentaste gedrückt wird, blinkt die MCPWR-Anzeige einmal. Während des Lesevorgangs leuchtet sie durchgängig und sie erlischt, wenn der Lesevorgang ohne Fehler abgeschlossen wird.

■ Vergleichen von Daten

- 1,2,3... 1. Stellen Sie Schalter 7 des DIP-Schalterblocks der CPU-Baugruppe auf AUS.
2. Drücken Sie die Speicherkartentaste der CPU-Baugruppe, und halten Sie diese drei Sekunden lang gedrückt.

Die Sicherungsdaten auf der Speicherkarte werden mit den Daten in den Baugruppen und Modulen verglichen.



Wenn die Speicherkartentaste gedrückt wird, blinkt die MCPWR-Anzeige einmal. Während des Vergleichsvorgangs leuchtet sie durchgängig und sie erlischt, wenn der Vergleichsvorgang ohne Fehler abgeschlossen wird und die Daten identisch sind.

Hinweis Wenn die serielle Kommunikationsbaugruppe CS1W-SCU21-V1 verwendet wird, ist die für eine Easy-Backup-Sicherung erforderliche Zeit länger als bei der Nicht-Verwendung der CS1W-SCU21-V1. Die entsprechenden Zeitunterschiede sind in der folgenden Tabelle aufgeführt.

Zusätzlich benötigte Zeit, wenn die CPU-Bus-Baugruppen-Einstellungsdatei (BACKUP□□.PRM) auf der Speicherkarte eine Größe von 60 kByte hat

Betriebsart	Zusätzlich benötigte Zeit beim Schreiben auf eine Speicherkarte	Zusätzlich benötigte Zeit beim Überprüfen einer Speicherkarte	Zusätzlich benötigte Zeit beim Lesen von einer Speicherkarte
PROGRAM	ca. 25 s	ca. 10 s	ca. 4 s
RUN	ca. 1 Min. 30 s	ca. 30 s	ca. 4 s

Zusätzlich benötigte Zeit, wenn die CPU-Bus-Baugruppen-Einstellungsdatei (BACKUP□□.PRM) auf der Speicherkarte eine Größe von 128 kByte hat

Betriebsart	Zusätzlich benötigte Zeit beim Schreiben auf eine Speicherkarte	Zusätzlich benötigte Zeit beim Überprüfen einer Speicherkarte	Zusätzlich benötigte Zeit beim Lesen von einer Speicherkarte
PROGRAM	ca. 40 s	ca. 14 s	ca. 8 s
RUN	ca. 2 Min. 30 s	ca. 1 Min.	ca. 8 s

Hinweis

- Überprüfen Sie, ob die Baugruppen und Module ordnungsgemäß funktionieren, bevor Sie die oben genannten Vorgänge ausführen. Die Schreib-, Lese- und Vergleichsvorgänge werden nicht ausgeführt, wenn die Baugruppen und Module nicht ordnungsgemäß funktionieren.
- Stellen Sie vor der Programmierung einer Easy-Backup-Sicherung für ein(e) spezifische(s) Baugruppe/Modul sicher, dass entweder die CPU-Baugruppe auf den Modus PROGRAM eingestellt ist oder dass die Ausführung der Easy-Backup-Sicherung sich nicht nachteilig auf Befehle auswirkt, die Kommunikations-Schnittstellennummern verwenden. Wenn Daten von spezifischen Baugruppen/Modulen gesichert werden, wird nach einer Kommunikationsschnittstelle gesucht, beginnend mit Schnittstelle 0. Die erste freie Schnittstelle wird verwendet. Wenn eine Schnittstellennummer mit einer anderen, von einem Netzwerkkommunikationsbefehl verwendeten übereinstimmt, wird der Netzwerkkommunikationsbefehl nicht ausgeführt, bis die Easy-Backup-Sicherung fertig gestellt wurde.

Spezieller Hinweis zu möglichen Änderungen des E/A-Zuordnungsmodus bei der Sicherung/Rücksicherung

Werden die Sicherungsparameterdateien (BKUP.STD) auf einer CJ-Serie CPU-Baugruppe erstellt und für die Verwendung mit einer anderen CJ-Serie CPU-Baugruppe auf einer Speicherkarte gespeichert, hängt der E/A-Zuordnungsmodus von den Baugruppenversionen der Quell- und Ziel-CPU-Baugruppe ab. Die nachstehende Tabelle zeigt die möglichen Änderungen des E/A-Zuordnungsmodus für die verschiedenen Kombinationen von Baugruppenversionen.

Quell-CPU-Baugruppe		Ursprünglicher E/A-Zuordnungsmodus	Ziel-CPU-Baugruppe		
			Baugruppenversion der CPU-Baugruppe, auf der die Daten gesichert werden/ die Rücksicherung erfolgt		
			bis einschließlich Ver. 2.0	Baugruppenversion 2.0	ab Baugruppenversion 3.0
Baugruppenversion der CPU-Baugruppe, von der die Daten gesichert werden	bis einschließlich Ver. 2.0	Automatische Zuordnung	(Identisch) automatische Zuordnung		
		Benutzerdefiniert	(Identisch) Benutzerdefiniert		
	Baugruppenversion 2.0	Automatische Zuordnung	(Identisch) automatische Zuordnung		
		Benutzerdefiniert	(Identisch) Benutzerdefiniert		
	ab Baugruppenversion 3.0	Automatische Zuordnung	(Identisch) automatische Zuordnung		
		Benutzerdefiniert	(Identisch) Benutzerdefiniert		

Spezieller Hinweis zu möglichen Abweichungen beim Vergleich der Sicherungsdateien mit den rückgesicherten Daten

Beim Vergleich der Parameterdaten in der Easy-Backup-Sicherungsdatei mit den rückgesicherten Daten können Verifizierungsfehler auftreten, wenn die Quell-CPU-Baugruppe und Ziel-CPU-Baugruppe unterschiedliche Versionsnummern aufweisen.

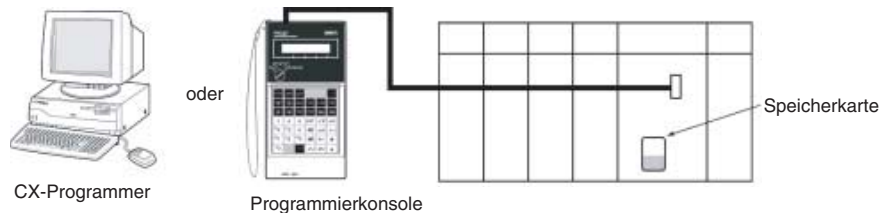
Quell-CPU-Baugruppe		Vorheriger E/A-Zuordnungsmodus	Ziel-CPU-Baugruppe		
			Baugruppenversion der CPU-Baugruppe, die Ziel der Sicherung/Rücksicherung ist		
			bis einschließlich Ver. 2.0	Baugruppenversion 2.0	ab Baugruppenversion 3.0
Baugruppenversion der CPU-Baugruppe, von der die Daten gesichert werden	bis einschließlich Ver. 2.0	Automatische Zuordnung	Verifizierung möglich	Verifizierung möglich	Verifizierungsfehler
		Benutzerdefiniert			
	Baugruppenversion 2.0	Automatische Zuordnung	Verifizierung möglich		Verifizierung möglich
		Benutzerdefiniert			
	ab Baugruppenversion 3.0	Automatische Zuordnung	Verifizierungsfehler	Verifizierungsfehler	Verifizierung möglich
		Benutzerdefiniert	Verifizierung möglich	Verifizierung möglich	

5-3 Verwenden des Dateispeichers

5-3-1 Initialisieren von Medien

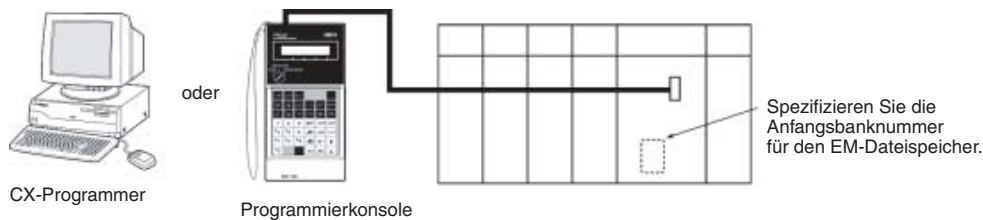
Speicherkarten

- 1,2,3...** 1. Verwenden Sie zur Initialisierung von Speicherkarten ein Programmiergerät, wie z. B. eine Programmierkonsole.

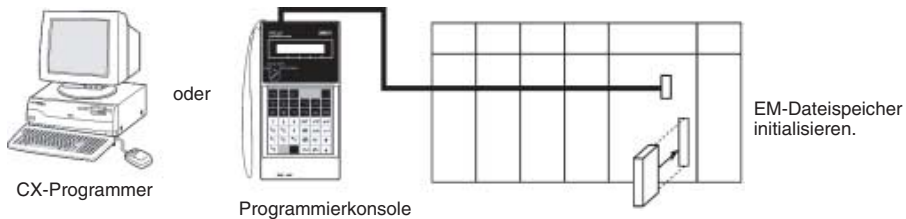


EM-Dateispeicher

- 1,2,3...** 1. Verwenden Sie ein Programmiergerät, wie z. B. eine Programmierkonsole, und legen Sie in den Einstellungen des EM-Dateispeichers im SPS-Setup fest, dass der EM-Dateispeicher aktiviert wird. Legen Sie anschließend die spezifizierte Banknummer für den EM-Dateispeicher zwischen 0 und C Hex fest.



2. Verwenden Sie zur Initialisierung des EM-Dateispeichers einen FINS-Befehl oder ein Programmiergerät, nicht jedoch eine Programmierkonsole.

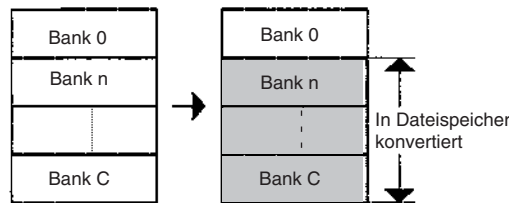


Initialisieren eines einzelnen EM-Dateispeichers

Eine spezifizierte EM-Bank kann aus normalem EM-Speicher in Dateispeicher umgewandelt werden.

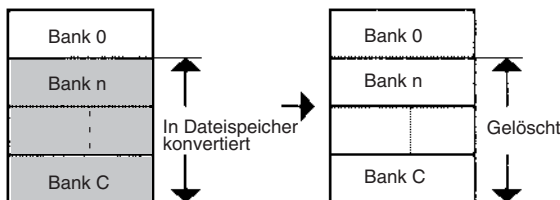
Hinweis

Die maximale Banknummer für CPU-Baugruppen der CJ-Serie ist 6.



1. Legen Sie "n" in den SPS-Einstellungen fest.
2. Verwenden Sie ein Programmiergerät oder einen FINS-Befehl zum Formatieren ab "n".
3. "n" wird in A344 gespeichert.

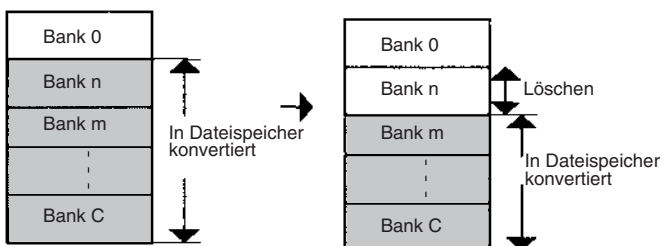
Als Dateispeicher verwendeter EM-Bereich kann wieder zu normalem EM-Status zurück konvertiert werden.



1. Setzen Sie die Dateispeicherzuweisung in den SPS-Einstellungen auf AUS.
2. Wenn für die Formatierung ein Programmiergerät oder ein FINS-Befehl verwendet wird, wird der Speicher ab "n" auf 0000 Hex gelöscht.
3. FFFF Hex wird in A344 gespeichert, um anzuzeigen, dass kein EM-Dateispeicher vorhanden ist.

Hinweis: 1. Alle vorhandenen Dateidaten werden zu diesem Zeitpunkt gelöscht.
2. Bei einer CPU-Baugruppe der CJ-Serie können nur die Bänke 0 bis 6 spezifiziert werden.

Die Nummer der ersten für Dateispeicher verwendeten Bank kann geändert werden.



1. Ändern Sie in den SPS-Einstellungen "n" zu "m".
2. Verwenden Sie ein Programmiergerät oder einen FINS-Befehl zum Konvertieren der Bänke, beginnend ab "m" in Dateispeicher.
3. "m" wird in A344 gespeichert.

Hinweis: 1. Alle vorhandenen Dateidaten werden zu diesem Zeitpunkt gelöscht.
2. Bei einer CPU-Baugruppe der CJ-Serie können nur die Bänke 0 bis 6 spezifiziert werden.

SPS-Setup

Adresse	Bezeichnung	Beschreibung	Anfängliche Einstellung
136	EM-Dateispeicher-Anfangsbank	0000 Hex: Keine 0080 Hex: Beginn bei Bank Nr. 0 008C Hex: Bank Nr. C Der EM-Bereich wird ab der spezifizierten Banknummer zu Dateispeicher umgewandelt.	0000 Hex

Zugehörige Spezial-Zusatzmerker

Bezeichnung	Adresse	Beschreibung
EM-Dateispeicher-Anfangsbank	A344	Die Banknummer, mit der der EM-Dateispeicherbereich tatsächlich zu diesem Zeitpunkt beginnt, wird gespeichert. Die EM-Datei von der ersten bis zur letzten Banknummer wird zu Dateispeicher umgewandelt. FFFF Hex gibt an, dass kein EM-Dateispeicher vorhanden ist.

Lesen/Schreiben von
Symboltabellen und
Kommentaren mit
CX-Programmer

Gehen Sie wie folgt vor, um Symboltabellen oder Kommentare, die mit CX-Programmer erstellt wurden, auf oder von Speicherkarte oder EM-Dateispeicher zu übertragen.

1,2,3...

1. Setzen Sie eine formatierte Speicherkarte in die CPU-Baugruppe ein, oder formatieren Sie einen EM-Dateispeicher.
2. Gehen Sie mit CX-Programmer in den Onlinemodus.
3. Wählen Sie **Übertragung** und anschließend im SPS-Menü die Option **Zur SPS** oder **Von der SPS**.
4. Wählen Sie als zu übertragende Daten entweder **Symbole** oder **Programmkommentare** aus.

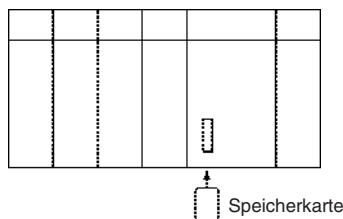
5-3-2 Bedienung

Speicherkarten

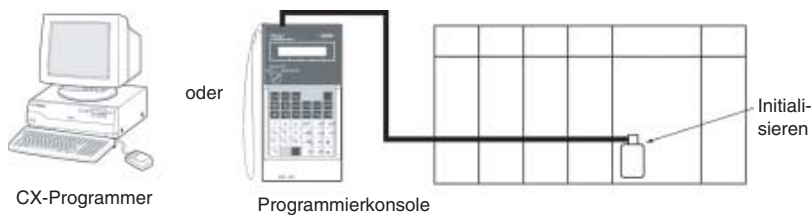
Verwenden eines Programmiergeräts

1,2,3...

1. Setzen Sie eine Speicherkarte in die CPU-Baugruppe ein.



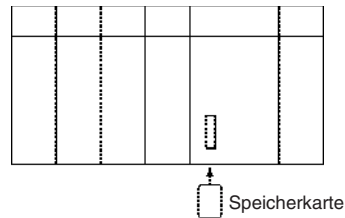
2. Initialisieren Sie die Speicherkarte mit Hilfe eines Programmiergeräts.



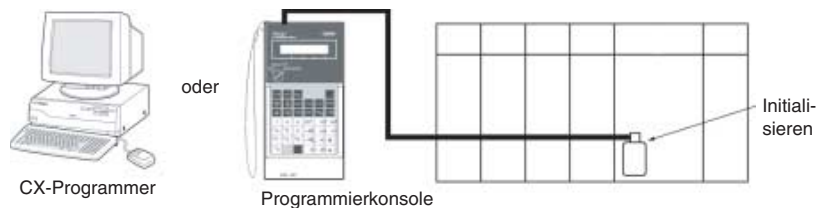
3. Verwenden Sie ein Programmiergerät, um die CPU-Baugruppendaten zu benennen (Anwenderprogramm, E/A-Speicher, Parameter-Bereich), und speichern Sie die Daten anschließend auf der Speicherkarte. (Verwenden Sie ein Programmiergerät, um die Speicherkartendatei in die CPU-Baugruppe einzulesen.)

Automatische Übertragung von Dateien beim Start**Vorgehensweise, wenn eine Parameterbereichsdatei übertragen wird**

- 1,2,3...** 1. Setzen Sie eine initialisierte Speicherkarte in die CPU-Baugruppe ein.

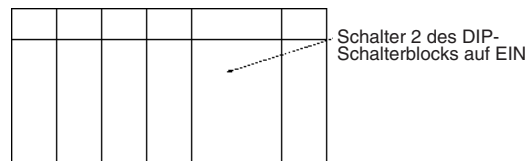


2. Verwenden Sie ein Programmiergerät, um die Dateien zur automatischen Übertragung beim Starten auf die Speicherkarte zu schreiben. Diese Dateien umfassen die Programmdatei (AUTOEXEC.OBJ), die Parameterbereichsdatei (AUTOEXEC.STD) und die E/A-Speicherdateien (AUTOEXEC.IOM, ATEXECMD.IOM und/oder ATEXECE□.IOM).

**Hinweis**

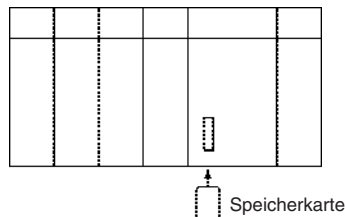
Auf der Speicherkarte müssen ein Anwenderprogramm und eine Parameterbereichsdatei vorhanden sein.

3. Schalten Sie die Spannungsversorgung der SPS AUS.
 4. Stellen Sie Schalter 2 des DIP-Schalterblocks auf EIN (Automatische Übertragung beim Start).



Hinweis Wenn Schalter 7 auf EIN steht und Schalter 8 auf AUS, wird die Sicherungsfunktion aktiviert, und diese hat Priorität gegenüber der Funktion zur automatischen Übertragung beim Start. Stellen Sie die Schalter 7 und 8 zur automatischen Übertragung beim Start auf AUS.

5. Setzen Sie die Speicherkarte in die CPU-Baugruppe ein.



6. Schalten Sie die Spannungsversorgung der SPS ein, um die Datei zu lesen.

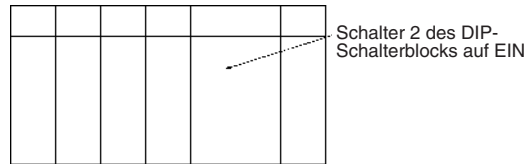
Vorgehensweise, wenn keine Parameterbereichsdatei übertragen wird

- 1,2,3...** 1. Setzen Sie eine initialisierte Speicherkarte in die CPU-Baugruppe ein.
 2. Verwenden Sie ein Programmiergerät, um die Dateien zur automatischen Übertragung beim Start auf die Speicherkarte zu schreiben. Diese Dateien umfassen die Programmdatei (REPLACE.OBJ), die Parameterbereichs-

datei (AUTOEXEC.STD) und die E/A-Speicherdateien (REPLACE.IOM, REPLCDM.IOM und/oder REPLCDE□.IOM).

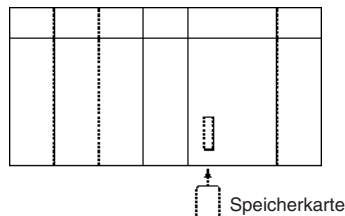
Hinweis Ein Parameterbereich wird nicht übertragen, auch wenn er sich auf der Speicherkarte befindet.

3. Schalten Sie die Spannungsversorgung der SPS AUS.
4. Stellen Sie Schalter 2 des DIP-Schalterblocks auf EIN (Automatische Übertragung beim Start).



Hinweis Wenn Schalter 7 auf EIN steht und Schalter 8 auf AUS, wird die Sicherungsfunktion aktiviert, und diese hat Priorität gegenüber der Funktion zur automatischen Übertragung beim Start. Stellen Sie die Schalter 7 und 8 zur automatischen Übertragung beim Start auf AUS.

5. Setzen Sie die Speicherkarte in die CPU-Baugruppe ein.



6. Schalten Sie die Spannungsversorgung der SPS ein, um die Datei zu lesen.

Verwenden der Befehle FREAD(700)/FWRITE(701)/CMND(490)

1,2,3...

1. Setzen Sie eine Speicherkarte in die CPU-Baugruppe ein. (Bereits initialisiert)
2. Verwenden Sie den Befehl FWRITE(701), um die Dateien im angegebenen Bereich des E/A-Speichers zu benennen und die Datei auf der Speicherkarte zu speichern.

Hinweis Eine Speicherkarte, die Datendateien im TXT- oder CSV-Format enthält, kann mit einem Speicherkartenadapter HMC-AP001 in den PC-Karten-Steckplatz eines Computers eingesetzt werden, und die Datendateien können mit Hilfe der Standardfunktionen von Windows in einem Tabellenkalkulationsprogramm gelesen werden (nicht unterstützt von CS1 CPU-Baugruppen der CS-Serie vor EV1).

3. Verwenden Sie den Befehl FREAD(700), um die Datei von der Speicherkarte in den E/A-Speicher der CPU-Baugruppe zu lesen.

Speicherkarten-Dateivorgänge können ausgeführt werden, indem mit CMND(490) FINS-Befehle an die lokale CPU-Baugruppe gesendet werden. (Nicht unterstützt von CS1 CPU-Baugruppen der CS-Serie vor EV1)

Programmaustausch während des Betriebs

1,2,3...

1. Setzen Sie eine Speicherkarte in die CPU-Baugruppe ein. (Bereits initialisiert)
2. Schreiben Sie das Programmpasswort (A5A5 Hex) in A651 und den Programmdateinamen in A654 bis A657.
3. Setzen Sie das Austausch-Start-Bit (A65015) von AUS auf EIN.

Easy-Backup-Funktion

Es stehen drei Sicherungsvorgänge zur Verfügung: Erstellung einer Sicherung von Daten auf der Speicherkarte, Wiederherstellen der Daten von der Speicherkarte und Vergleichen von Daten mit denen auf der Speicherkarte.

Erstellen einer Datensicherung der CPU-Baugruppe auf der Speicherkarte

- 1,2,3...**
1. Setzen Sie eine Speicherkarte in die CPU-Baugruppe ein. (Bereits initialisiert)
 2. Stellen Sie Schalter 7 des DIP-Schalterblocks der CPU-Baugruppe auf EIN und Schalter 8 auf AUS.
 3. Drücken Sie die Speicherkartentaste der CPU-Baugruppe, und halten Sie diese drei Sekunden lang gedrückt.
 4. Überprüfen Sie, ob die MCPWR-Anzeige einmal blinkt und dann erlischt. (Ein anderes Verhalten der Anzeige weist darauf hin, dass bei der Erstellung der Datensicherung ein Fehler aufgetreten ist.)

Wiederherstellen von Daten von der Speicherkarte in die CPU-Baugruppe

- 1,2,3...**
1. Setzen Sie die Speicherkarte mit den Sicherungsdateien in die CPU-Baugruppe ein.
 2. Stellen Sie Schalter 7 des DIP-Schalterblocks der CPU-Baugruppe auf EIN und Schalter 8 auf AUS.
 3. Die Sicherungsdateien werden wiederhergestellt, sobald die SPS eingeschaltet wird.
 4. Überprüfen Sie, ob die MCPWR-Anzeige einmal blinkt und dann erlischt. (Ein anderes Verhalten der Anzeige weist darauf hin, dass bei der Wiederherstellung der Daten ein Fehler aufgetreten ist.)

Vergleichen von Daten auf der Speicherkarte und in der CPU-Baugruppe

- 1,2,3...**
1. Setzen Sie die Speicherkarte mit den Sicherungsdateien in die CPU-Baugruppe ein.
 2. Stellen Sie die Schalter 7 und 8 des DIP-Schalterblocks der CPU-Baugruppe auf AUS.
 3. Drücken Sie die Speicherkartentaste der CPU-Baugruppe, und halten Sie diese drei Sekunden lang gedrückt.
 4. Die Daten stimmen überein, wenn die MCPWR-Anzeige einmal blinkt und dann erlischt.

Hinweis Die MCPWR-Anzeige blinkt, wenn beim Schreiben oder Vergleichen der Daten ein Fehler auftritt. Wenn die Speicherkartentaste gedrückt wird, hört die MCPWR-Anzeige auf zu blinken und leuchtet durchgängig.

In der folgenden Tabelle finden Sie Angaben zu den Zeiten, die für Sicherungsvorgänge bei Programmen mit 20 kSteps und einer Zykluszeit von 10 ms in der RUN-Betriebsart erforderlich sind:

Betriebsart	Sicherung	Wiederherstellung	Vergleich
PROGRAM	ca. 50 s	ca. 30 s	ca. 7 s
RUN	ca. 5 Min.	ca. 2 Min.	ca. 7 s

In der folgenden Tabelle finden Sie Angaben zu den Zeiten, die für Sicherungsvorgänge bei Programmen mit 30 kSteps und einer Zykluszeit von 10 ms in der RUN-Betriebsart erforderlich sind:

Betriebsart	Sicherung	Wiederherstellung	Vergleich
PROGRAM	ca. 50 s	ca. 30 s	ca. 7 s
RUN	ca. 5 Min. 30 s	ca. 2 Min. 40 s	ca. 7 s

In der folgenden Tabelle finden Sie Angaben zu den Zeiten, die für Sicherungsvorgänge bei Programmen mit 250 kSteps und einer Zykluszeit von 12 ms in der RUN-Betriebsart erforderlich sind:

Betriebsart	Sicherung	Wiederherstellung	Vergleich
PROGRAM	ca. 1 Min. 30 s	ca. 1 Min. 30 s	ca. 20 s
RUN	ca. 13 Min.	ca. 7 Min. 30 s	ca. 20 s

Erstellen von Variablen- und Kommentartabellen

Verwenden Sie die folgende Vorgehensweise bei CX-Programmer zur Erstellung von Variablen- oder Kommentartabellen auf Speicherkarten oder in einem EM-Dateispeicher.

- 1,2,3...**
1. Setzen Sie eine formatierte Speicherkarte in die CPU-Baugruppe ein, oder formatieren Sie einen EM-Dateispeicher.
 2. Gehen Sie mit CX-Programmer in den Onlinemodus.
 3. Wählen Sie **Übertragung** und anschließend im SPS-Menü die Option **Zur SPS** oder **Von der SPS**.
 4. Wählen Sie als zu übertragende Daten entweder **Symbole** oder **Programmkommentare** aus.

Hinweis Wenn eine Speicherkarte in die CPU-Baugruppe eingesetzt ist, können Daten nur mit der Speicherkarte übertragen werden. (Dies ist nicht mit EM-Dateispeicher möglich.)

EM-Dateispeicher

Verwenden eines Programmiergeräts

- 1,2,3...**
1. Verwenden Sie das SPS-Setup, um die Anfangs-EM-Bank für die Umwandlung zu Dateispeicher zu spezifizieren.
 2. Verwenden Sie ein Programmiergerät, um den EM-Dateispeicher zu initialisieren.
 3. Verwenden Sie ein Programmiergerät, um die Daten der CPU-Baugruppe zu benennen (Anwenderprogramm, E/A-Speicher, Parameterbereich), und speichern Sie die Daten anschließend im EM-Dateispeicher.
 4. Verwenden Sie ein Programmiergerät, um die Datei im EM-Dateispeicher in die CPU-Baugruppe einzulesen.

Verwenden der Befehle FREAD(700)/FWRIT(701)/CMND(490)

- 1,2,3...**
1. Verwenden Sie das SPS-Setup, um die Anfangs-EM-Bank für die Umwandlung zu Dateispeicher zu spezifizieren.
 2. Verwenden Sie ein Programmiergerät, um den EM-Dateispeicher zu initialisieren.
 3. Verwenden Sie den Befehl FWRIT(701), um die Dateien im spezifizierten Bereich des E/A-Speichers zu benennen und die Datei im EM-Dateispeicher zu speichern.
 4. Verwenden Sie den Befehl FREAD(700), um die Datei vom EM-Dateispeicher in den E/A-Speicher der CPU-Baugruppe einzulesen.

EM-Dateispeicher-Vorgänge können ausgeführt werden, indem mit CMND(490) FINS-Befehle an die lokale CPU-Baugruppe gesendet werden.

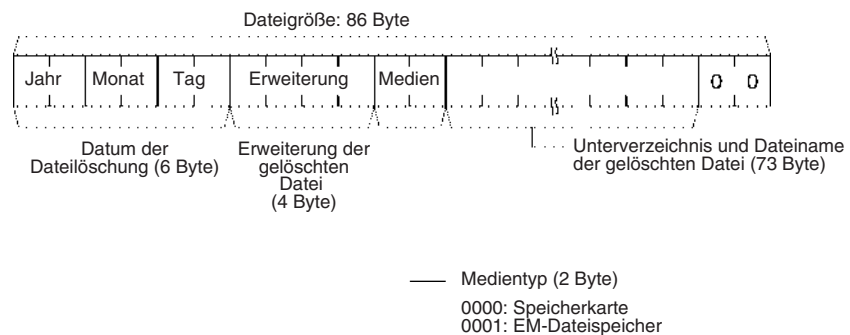
5-3-3 Unterbrechungen der Versorgungsspannung während des Zugriffs auf den Dateispeicher

Wenn eine Unterbrechung der Versorgungsspannung auftritt, während die CPU auf den Dateispeicher zugreift (Speicherkarte oder EM-Dateispeicher) sind die Inhalte der Speicherkarte möglicherweise nicht korrekt. Die aktualisierte Datei wird möglicherweise nicht ordnungsgemäß überschrieben und in manchen Fällen kann auch die Speicherkarte selbst beschädigt werden.

Die betroffene Datei wird beim nächsten Einschalten automatisch vom System gelöscht. Der entsprechende Datei-gelöscht-Merker (A39507 für die Speicherkarte, A39506 für EM-Dateispeicher) wird auf EIN gesetzt. Der Merker wird beim nächsten Ausschalten auf AUS gesetzt.

Beim Löschen einer Datei wird eine Löschprotokoll-Datei (DEL_FILE.IOM) im Verzeichnis der Speicherkarte bzw. des EM-Dateispeichers erstellt. Die Löschprotokolldatei kann mit CX-Programmer oder FREAD(700) gelesen werden. Überprüfen Sie dabei die folgenden Informationen: Das Löschdatum der Datei, den vorhandenen Dateispeichertyp (Medien), das Unterverzeichnis sowie Dateiname und -Erweiterung. Bei Bedarf können Sie die gelöschte Datei neu erstellen oder erneut kopieren.

Die folgende Abbildung zeigt die Struktur der Löschprotokolldatei.



ABSCHNITT 6

Erweiterte Funktionen

Dieser Abschnitt enthält Detailinformationen zu den folgenden erweiterten Funktionen: Funktionen für Zykluszeit/Hochgeschwindigkeitsverarbeitung, Indexregister-Funktionen, Funktionen für serielle Kommunikation, Funktionen für Inbetriebnahme und Wartung, Funktionen für Diagnose und Debugging, Programmiergerätefunktionen und die Einstellungen für die Eingangs-Ansprechzeit von E/A-Baugruppen.

6-1	Zykluszeit/Hochgeschwindigkeitsverarbeitung	257
6-1-1	Minimale Zykluszeit.	257
6-1-2	Maximale Zykluszeit (Watchdog)	258
6-1-3	Zykluszeitüberwachung	258
6-1-4	Schnelle Eingänge	259
6-1-5	Interrupt-Funktionen.	259
6-1-6	E/A-Aktualisierungsmethoden	260
6-1-7	Deaktivieren der zyklischen Aktualisierung von Spezial-E/A-Baugruppen	262
6-1-8	Verbessern der Aktualisierungs-Ansprechgeschwindigkeit für CPU-Bus-Baugruppendaten	262
6-1-9	Maximale Data-Link-E/A-Ansprechzeit	264
6-1-10	Befehlsausführung im Hintergrund	266
6-1-11	Gemeinsame Nutzung von Index- und Datenregistern in verschiedenen Tasks	273
6-2	Indexregister	274
6-2-1	Zweck von Indexregistern.	274
6-2-2	Verwenden von Indexregistern	275
6-2-3	Verarbeitung mit Indexregistern	278
6-3	Serielle Kommunikation	284
6-3-1	Host-Link-Kommunikation	286
6-3-2	Protokollfreie Kommunikation	291
6-3-3	NT-Link (1:N-Modus)	292
6-3-4	Die Serial Gateway-Funktion der CPU-Baugruppe	293
6-3-5	Serielle SPS-Links (nur CJ1M CPU-Baugruppen).	299
6-4	Ändern des Modus der Zeitgeber/Zähler-Istwert-Aktualisierung	305
6-4-1	Übersicht.	305
6-4-2	Funktionsspezifikationen	306
6-4-3	Auswahl und Überprüfung des BCD-/Binärmodus	307
6-4-4	AWL-Codes und Daten des BCD-/Binärmodus	308
6-4-5	Beschränkungen	309
6-4-6	Befehle und Operanden	310
6-5	Verwenden eines zeitgesteuerten Interrupts als Präzisionszeitgeber (nur CJ1M).	313
6-5-1	Einstellen des zeitgesteuerten Interrupts auf 0,1-ms-Schritte.	313
6-5-2	Spezifizieren einer Rücksetzung mit MSKS(690)	314
6-5-3	Lesen des Istwerts des internen Zeitgebers mit MSKR(692)	314
6-6	Einstellungen für Start und Wartung	315
6-6-1	Warmstart-/Warmstopp-Funktionen	315
6-6-2	Einstellung der Betriebsart beim Einschalten.	316

6-6-3	RUN-Ausgang	317
6-6-4	Einstellung der Ausschalterkennungs-Verzögerungszeit	317
6-6-5	Deaktivieren der Ausschalt-Interrupts.	317
6-6-6	Uhrfunktionen	318
6-6-7	Programmschutz	319
6-6-8	Schreibschutz gegen FINS-Befehle, die über Netzwerke an die CPU-Baugruppen gesendet werden.	321
6-6-9	Dezentrale Programmierung und Überwachung	322
6-6-10	Baugruppen-Profile	322
6-6-11	Flash-Speicher.	323
6-6-12	Startbedingungen (Einstellung im SPS-Setup)	324
6-7	Diagnosefunktionen.	326
6-7-1	Fehlerprotokoll	326
6-7-2	Ausgang-AUS-Funktion	327
6-7-3	Fehleralarmfunktionen	327
6-7-4	Fehlerpunkt-Erkennung.	328
6-7-5	Simulieren von Systemfehlern	329
6-7-6	Deaktivieren der Fehlerprotokollierung für anwenderdefinierte FAL-Fehler.	330
6-8	CPU-Verarbeitungsbetriebsarten	331
6-8-1	CPU-Verarbeitungsbetriebsarten	331
6-8-2	Parallelverarbeitungs-Betriebsart und minimale Zykluszeiten	336
6-8-3	Datenübereinstimmung bei Parallelverarbeitung mit asynchronem Speicherzugriff	336
6-9	Peripheriebedienungs-Prioritätsmodus	336
6-9-1	Peripheriebedienungs-Prioritätsmodus	337
6-9-2	Vorübergehendes Deaktivieren der Peripheriebedienungs-Priorität	339
6-10	Betrieb ohne Batterie	342
6-11	Weitere Funktionen	344
6-11-1	E/A-Ansprechzeit-Einstellungen.	344
6-11-2	E/A-Bereichszuordnung	345

6-1 Zykluszeit/Hochgeschwindigkeitsverarbeitung

Die folgenden Funktionen werden in diesem Abschnitt erläutert:

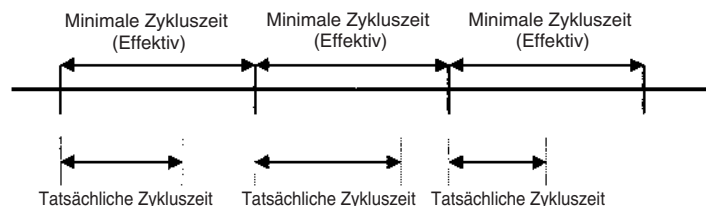
- Minimale Zykluszeit
- Maximale Zykluszeit (Watchdog)
- Zykluszeitüberwachung
- Eingänge mit kurzer Ansprechzeit
- Interrupt-Funktionen
- E/A-Aktualisierungsmethoden
- Deaktivierung der zyklischen E/A-Aktualisierung von Spezial-E/A-Baugruppen
- Verbessern des Aktualisierungsverhaltens für Data-Links und andere CPU-Bus-Baugruppen-Daten (nur CS1-H, CJ1-H, CJ1M und CS1D CPU-Baugruppen)
- Reduzieren von Schwankungen in der Zykluszeit durch Hintergrundausführung von Datenmanipulationen (nur CS1-H, CJ1-H, CJ1M und CS1D CPU-Baugruppen)

6-1-1 Minimale Zykluszeit

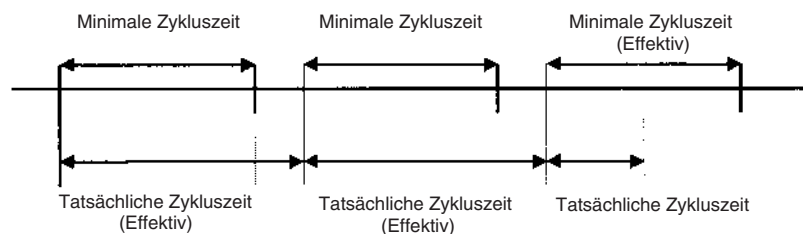
Bei SPS-Systemen der CS/CJ-Serie kann eine minimale (oder feste) Zykluszeit eingestellt werden. (Siehe Hinweis.) Unterschiede in den E/A-Ansprechzeiten können bei wiederholter Abarbeitung des Programms mit einer festen Zykluszeit unterdrückt werden.

Hinweis Sie können die Zykluszeit auch bei einer CS1D CPU-Baugruppe für Einzel-CPU-Systeme, einer CS1-H, CJ1-H oder CJ1M CPU-Baugruppe oder durch Verwenden eines Parallelverarbeitungsmodus fest einstellen.

Die minimale Zykluszeit (1 bis 32.000 ms) wird im SPS-Setup in 1-ms-Schritten spezifiziert.



Wenn die tatsächliche Zykluszeit länger ist als die minimale Zykluszeit, ist die Funktion für die minimale Zykluszeit unwirksam und die Zykluszeit variiert von Zyklus zu Zyklus.



SPS-Setup

Adresse	Bezeichnung	Einstellung	Standard-einstellung
208 Bits: 0 bis 15	Minimale Zykluszeit	0001 bis 7D00: 1 bis 32.000 ms (1-ms-Schritte)	0000 (kein Minimum)

6-1-2 Maximale Zykluszeit (Watchdog)

Überschreitet die Zykluszeit (siehe Hinweis) die Watchdog-Zykluszeit, wird der Zykluszeit-zu-lang-Merker (A40108) auf EIN gesetzt und der SPS-Betrieb angehalten.

Hinweis In diesem Falle entspricht die Zykluszeit der Programmausführungsdauer, wenn Sie bei CS1-H, CJ1-H oder CJ1M CPU-Baugruppen sowie CS1D CPU-Baugruppen für Einzel-CPU-Systeme einen Parallelverarbeitungsmodus verwenden.

SPS-Setup

Adresse	Bezeichnung	Einstellung	Standard-einstellung
209 Bit: 15	Einstellung der Watchdog-Zykluszeit aktivieren	0: Standardeinstellung (1 s) 1: Bits 0 bis 14	0001 (1 s)
209 Bits: 0 bis 14	Watchdog-Zykluszeit-einstellung (Aktiviert, wenn Bit 15 auf 1 gesetzt ist.)	001 bis FA0: 10 bis 40.000 ms (10-ms-Schritte)	

Merker und Worte des Zusatz-Systembereichs

Bezeichnung	Adresse	Beschreibung
Zykluszeit-zu-lang-Merker	A40108	Wenn die Zykluszeit die eingestellte Watchdog-Zykluszeit überschreitet, wird A40108 auf EIN gesetzt und die CPU-Baugruppe stoppt den Betrieb. Die Zykluszeit entspricht der Programmausführungsdauer, wenn Sie bei CS1-H, CJ1-H oder CJ1M CPU-Baugruppen sowie CS1D CPU-Baugruppen für Einzel-CPU-Systeme einen Parallelverarbeitungsmodus verwenden.

Hinweis Wenn bei CS1-H, CJ1-H oder CJ1M CPU-Baugruppen sowie CS1D CPU-Baugruppen für Einzel-CPU-Systeme im Parallelverarbeitungsmodus die Peripheriebedienungszykluszeit 2,0 s überschreitet, tritt ein Fehler wegen Überschreiten der Peripheriebedienungszykluszeit auf und die CPU-Baugruppe stoppt den Betrieb. In diesem Falle wird A40515 (Peripheriebedienungszykluszeit-abgelaufen-Merker) auf EIN gesetzt.

6-1-3 Zykluszeitüberwachung

Die maximale und die aktuelle Zykluszeit werden jeden Zyklus im Zusatz-Systembereich gespeichert. Bei CS1-H, CJ1-H CPU-Baugruppen oder CS1D CPU-Baugruppen für Einzel-CPU-Systeme im Parallelverarbeitungsmodus werden die Programmausführungsdauern gespeichert.

Merker und Worte des Zusatz-Systembereichs

Bezeichnung	Adresse	Beschreibung
Maximale Zykluszeit (Programmausführungsdauer bei CS1-H, CJ1-H oder CJ1M CPU-Baugruppen im Parallelverarbeitungsmodus)	A262 und A263	Wird jeden Zyklus als 32-Bit-Binärwert im folgenden Bereich gespeichert: 0 bis 429.496.729,5 ms in 0,1-ms-Schritten (0 bis FFFF FFFF)
Aktuelle Zykluszeit (Programmausführungsdauer bei CS1-H, CJ1-H oder CJ1M CPU-Baugruppen im Parallelverarbeitungsmodus)	A264 und A265	Wird jeden Zyklus als 32-Bit-Binärwert im folgenden Bereich gespeichert: 0 bis 429.496.729,5 ms in 0,1-ms-Schritten (0 bis FFFF FFFF)

Mit einem Programmiergerät (CX-Programmer oder Programmierkonsole) können Sie den Mittelwert der Zykluszeiten der letzten acht Zyklen lesen.

Reduzieren der Zykluszeit

Mit den folgenden Methoden können Sie bei SPS-Systemen der CS/CJ-Serie die Zykluszeit effektiv reduzieren:

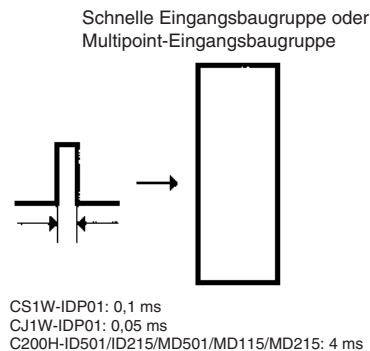
- 1,2,3...**
1. Versetzen Sie nicht ausgeführte Tasks in den Standby-Modus (gestoppt).
 2. Überspringen Sie nicht ausgeführte Programmabschnitte mit JMP(004) und JME(005).

Bei CS1-H oder CJ1-H CPU-Baugruppen sowie CS1D CPU-Baugruppen für Einzel-CPU-Systeme im Parallelverarbeitungsmodus wird die Peripheriebedienungszykluszeit in jeden Bedienungszyklus in der Adresse A268 (Peripheriebedienungszykluszeit) gespeichert.

6-1-4 Schnelle Eingänge

Wenn Sie Impulse empfangen möchten, die kürzer als die Zykluszeit sind, verwenden Sie die schnelle Eingangsbaugruppe CS1W-IDP01 oder die schnellen Eingänge der C200H-ID501/ID215 und C200H-MD501/MD115/MD215 Multi-Point-E/A-Baugruppen.

Die schnellen Eingänge können Impulse mit einer Impulsdauer (EIN-Dauer) von 1 ms oder 4 ms bei C200H Multi-Point-Eingangsbaugruppen bzw. 0,1 ms bei der schnellen Eingangsbaugruppe CS1W-IDP01 erkennen.

**6-1-5 Interrupt-Funktionen**

Interrupt-Tasks können unter den folgenden Bedingungen ausgeführt werden. Weitere Detailinformationen finden Sie unter *4-3 Interrupt-Tasks*.

Hinweis Die CS1D CPU-Baugruppen für Duplex-CPU-Systeme unterstützen keine Interrupts. Für CS1D CPU-Baugruppen können Interrupt-Tasks nur als zyklische Zusatztasks verwendet werden, d. h., dass andere Arten von Interrupt-Tasks nicht verwendet werden können.

E/A-Interrupts (Interrupt-Tasks 100 bis 131)

Eine E/A-Interrupt-Task wird ausgeführt, wenn von einer Interrupt-Eingangsbaugruppe das entsprechende Signal empfangen wird (an der ansteigenden Flanke des Signals oder bei CS/CJ-Serie Interrupt-Eingangsbaugruppen an der ansteigenden oder fallenden Flanke).

Zeitgesteuerte Interrupts (Interrupt-Tasks 2 und 3)

Eine zeitgesteuerte Interrupt-Task wird in regelmäßigen Intervallen ausgeführt.

Ausschalt-Interrupt-Task (Interrupt-Task 1)

Diese Task wird bei einer Unterbrechung der Spannungsversorgung ausgeführt.

Externe Interrupts (Interrupt-Tasks 0 bis 255)

Eine externe Interrupt-Task wird beim Empfang eines Interrupts von einer Spezial-E/A-Baugruppe, CPU-Bus-Baugruppe oder einem internen Modul ausgeführt.

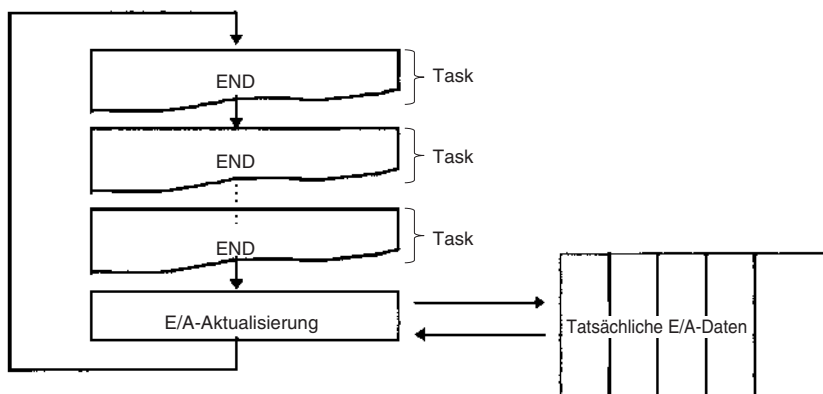
Hinweis Zu Aktivierung von Interrupt-Tasks können die integrierten Interrupt- und Schnelle-Zähler-Eingänge einer CJ1M CPU-Baugruppe verwendet werden. Detaillierte Informationen hierzu finden Sie im *CJ-Serie Integrierte E/A Bedienerhandbuch*.

6-1-6 E/A-Aktualisierungsmethoden

Es gibt drei Möglichkeiten, wie CPU-Baugruppen der CS/CJ-Serie Daten mit E/A-Baugruppen und Spezial-E/A-Baugruppen aktualisieren können: Zyklische Aktualisierung, sofortige Aktualisierung und Ausführung von IORF(097).

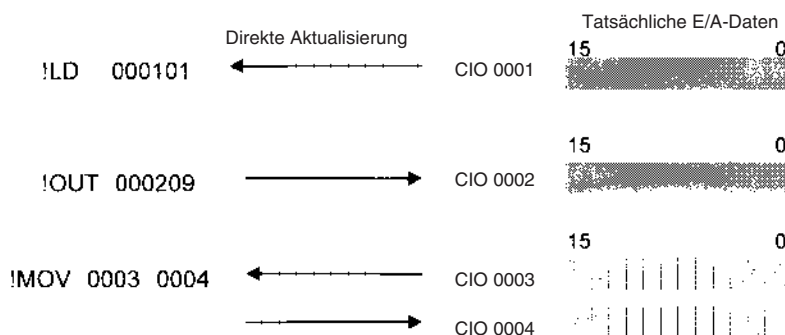
1. Zyklische Aktualisierung

Die E/A-Aktualisierung wird durchgeführt, nachdem alle Befehle der ausführbaren Tasks ausgeführt wurden. (Im SPS-Setup kann die zyklische Aktualisierung einzelner Spezial-E/A-Baugruppen deaktiviert werden.)



2. Direkte Aktualisierung

Wenn bei der Befehlsvariante mit direkter Aktualisierung eine Adresse im E/A-Bereich als Operand angegeben wird, werden die Operandendaten aktualisiert, wenn der Befehl ausgeführt wird. Befehle für direkte Aktualisierung können Daten aktualisieren, die E/A-Baugruppen zugeordnet sind. Direkte Aktualisierung ist auch für die integrierten E/A von CJ1M CPU-Baugruppen möglich.

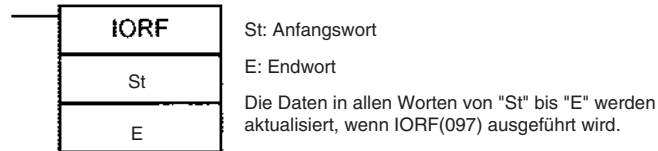


- Hinweis**
1. Wenn der Befehl einen Bit-Operanden enthält, wird das gesamte dazugehörige Wort aktualisiert. Wenn der Befehl einen Wort-Operanden enthält, wird das Wort aktualisiert.
 2. Die Eingangs- und Quelldaten werden direkt vor der Ausführung des Befehls aktualisiert. Die Ausgangs- und Zieldaten werden direkt nach der Ausführung des Befehls aktualisiert.
 3. Die Ausführungszeiten der Befehlsvarianten mit direkter Aktualisierung sind länger als die der regulären Varianten der Befehle. Somit verlängert sich auch die Zykluszeit. Ausführliche Informationen hierzu finden Sie im *Bedienerhandbuch* unter *10-5 Ausführungszeit und Step-Anzahl der Befehle*.
 4. CS1D CPU-Baugruppen für Duplex-CPU-Systeme unterstützen keine direkte Aktualisierung.

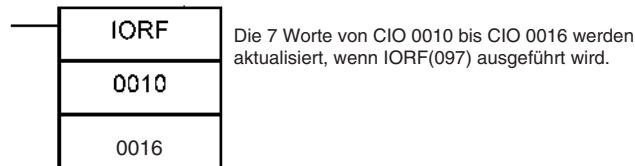
3. Ausführung von IORF(097) und DLNK(226)

■ IORF(097): I/O REFRESH

Mit dem Befehl IORF(097) können Sie durch Ausführung eines Befehls einen Bereich von E/A-Worten aktualisieren. Der Befehl IORF(097) kann auch zur Aktualisierung der E/A-Baugruppen und Spezial-E/A-Baugruppen zugeordneten Daten verwendet werden.



Im folgenden Beispiel wird gezeigt, wie Sie mit IORF(097) acht Worte E/A-Daten aktualisieren.



Wenn für die Ein- und Ausgabe einer Berechnung ein schnelles Ansprechen erforderlich ist, verwenden Sie IORF(097) direkt vor und nach dem Berechnungsbefehl.

Hinweis

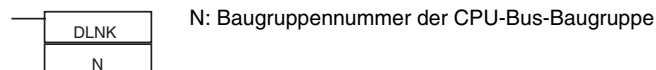
Die relativ lange Befehlsausführungszeit von IORF(097) steigt proportional mit der Anzahl der zu aktualisierenden Worte, so dass seine Ausführung zu einem deutlichen Ansteigen der Zykluszeit führen kann. Ausführliche Informationen hierzu finden Sie im *Bedienerhandbuch* unter *10-5 Ausführungszeit und Step-Anzahl der Befehle*.

■ DLNK(226): CPU-Bus-Baugruppen-E/A-Aktualisierung (nur CS1-H, CJ1-H, CJ1M und CS1D CPU-Baugruppen)

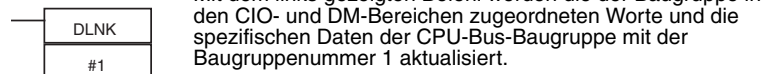
Mit dem Befehl DLNK(226) können Sie Daten einer anhand ihrer Baugruppennummer spezifizierte CPU-Bus-Baugruppe aktualisieren. Die folgenden Daten werden aktualisiert.

- Worte, die der Baugruppe im CIO-Bereich zugeordnet sind
- Worte, die der Baugruppe im DM-Bereich zugeordnet sind
- Daten, die spezifisch für diese Baugruppe sind (siehe Hinweis)

Hinweis Zu den Daten, die spezifisch zu einer CPU-Bus-Baugruppe gehören, zählen bei Controller-Link- oder SYSMAC LINK-Baugruppen die Data-Links und bei DeviceNet-Baugruppen die dezentralen E/A.



Beispiel:



Anwendungsbeispiel: Bei einer langen Zykluszeit kann das Aktualisierungsintervall für Controller-Link Data-Links sehr lang sein. Sie können dieses Intervall verkürzen, indem Sie den Befehl DLNK(226) für die Controller-Link-Baugruppe ausführen und dadurch die Frequenz der Data-Link-Aktualisierung erhöhen.

6-1-7 Deaktivieren der zyklischen Aktualisierung von Spezial-E/A-Baugruppen

Spezial-E/A-Baugruppen werden entsprechend ihrer Baugruppennummer (an der Gerätefront der Baugruppe einstellbar) zehn Worte im Spezial-E/A-Baugruppenbereich (CIO 2000 bis CIO 2959) zugeordnet. Die Daten werden in jedem Zyklus während der E/A-Aktualisierung zwischen diesem Bereich und der CPU-Baugruppe aktualisiert, jedoch können Sie diese zyklische Aktualisierung im SPS-Setup für einzelne Baugruppen deaktivieren.

Drei Gründe sprechen für die Deaktivierung der zyklischen Aktualisierung:

1,2,3...

1. Die zyklische Aktualisierung der Spezial-E/A-Baugruppen kann deaktiviert werden, wenn die Zykluszeit zu lang ist, da zu viele Spezial-E/A-Baugruppen installiert sind.
2. Erfolgen die E/A-Aktualisierungen in zu kurzem Abstand hintereinander, kann die interne Verarbeitung der Spezial-E/A-Baugruppe möglicherweise nicht Schritt halten. Dies hat zur Folge, dass der Spezial-E/A-Baugruppen-Fehlermerker (A40206) auf EIN gesetzt wird und die Spezial-E/A-Baugruppe möglicherweise nicht ordnungsgemäß funktioniert.
In diesem Fall kann wahlweise die Zykluszeit durch Einstellung einer Mindestzykluszeit im SPS-Setup verlängert oder die zyklische E/A-Aktualisierung der betroffenen Spezial-E/A-Baugruppen deaktiviert werden.
3. Deaktivieren Sie die zyklische Aktualisierung einer Spezial-E/A-Baugruppe immer, wenn sie in einer Interrupt-Task durch den Befehl IORF(097) aktualisiert wird. Wenn für dieselbe Baugruppe gleichzeitig eine zyklische Aktualisierung und eine Aktualisierung durch den Befehl IORF(097) ausgeführt werden, tritt ein Interrupt-Task-Fehler auf und der Interrupt-Task-Fehlermerker (A40213) wird auf EIN gesetzt.

Bei deaktivierter zyklischer E/A-Aktualisierung kann die E/A-Aktualisierung der betroffenen Spezial-E/A-Baugruppen im Rahmen der Programmausführung durch Ausführung des Befehls IORF(097) erfolgen.

SPS-Setup

Die Bits für die Deaktivierung der zyklischen Aktualisierung der Spezial-E/A-Baugruppen 0 bis 95 entsprechen direkt den 96 Bits in den Adressen 226 bis 231.

Adresse	Bezeichnung	Einstellung	Standard-einstellung
226 Bit 0	Bit für Deaktivierung der zyklischen Aktualisierung von Spezial-E/A-Baugruppe 0	0: Aktiviert 1: Deaktiviert	0 (Aktiviert)
:	:	:	:
231 Bit 15	Bit für Deaktivierung der zyklischen Aktualisierung von Spezial-E/A-Baugruppe 95	0: Aktiviert 1: Deaktiviert	0 (Aktiviert)

6-1-8 Verbessern der Aktualisierungs-Ansprechgeschwindigkeit für CPU-Bus-Baugruppendaten

Diese Funktion wird nur von den CS1-H, CJ1-H, CJ1M und CS1D CPU-Baugruppen unterstützt.

Normalerweise werden Data-Links und andere spezielle Daten für CPU-Bus-Baugruppen zusammen mit den Worten der CIO- und DM-Bereiche, die den Baugruppen zugeordnet sind, während der E/A-Aktualisierung nach der Programmausführung aktualisiert.

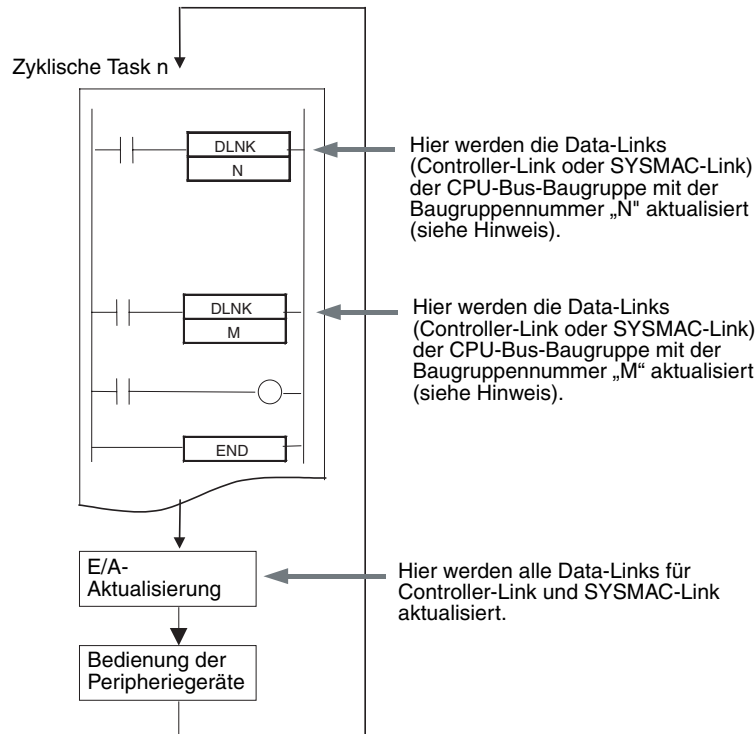
In der folgenden Tabelle sind einige Beispiele spezieller Daten für CPU-Bus-Baugruppen aufgelistet.

Baugruppen	Spezielle Daten
Controller-Link-Baugruppen und SYSMAC LINK-Baugruppen	Controller-Link- und SYSMAC LINK-Data-Links (einschließlich automatischer und anwenderdefinierter Links)
DeviceNet-Baugruppen der CS/CJ-Serie	Dezentrale DeviceNet-E/A-Kommunikation (einschließlich fester und anwenderdefinierter Zuordnungen)

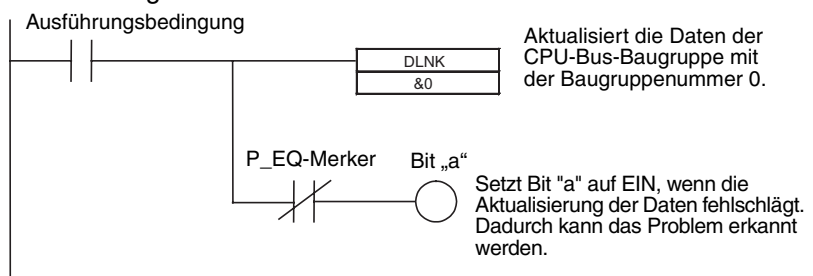
Mit den folgenden Funktionen können Sie die Aktualisierungs-Ansprechgeschwindigkeit für spezielle CPU-Bus-Baugruppendaten der CS1-H, CJ1-H, CJ1M oder CS1D CPU-Baugruppen verbessern.

- Reduzieren der Zykluszeit durch den Parallelverarbeitungsmodus oder Verwendung von Hochgeschwindigkeitsbefehlen (der Parallelverarbeitungsmodus wird von CS1D CPU-Baugruppen für Duplex-CPU-Systeme nicht unterstützt.)
- Ausführen des Befehls DLNK(226), um bestimmte CPU-Bus-Baugruppen unter Spezifikation ihrer Baugruppennummern zu aktualisieren. Der Befehl (DLNK(226) kann in einem Programm mehrfach verwendet werden.)

- Hinweis** 1. Durch längere Zykluszeiten (z. B. 100 ms) wird das Intervall zwischen den Aktualisierungen der Data-Links erhöht. In diesem Fall kann, wie im folgenden Beispiel dargestellt, der Befehl DLNK(226) verwendet werden.



- Hinweis** Wenn für eine CPU-Bus-Baugruppe, die gerade Daten aktualisiert, der Befehl DLNK(226) ausgeführt wird, werden die Daten nicht aktualisiert und der Gleich-Merker (P_EQ) wird auf AUS gesetzt. Normalerweise sollte der Gleich-Merker (P_EQ) wie unter gezeigt programmiert werden, um sicherzustellen, dass die Aktualisierung fehlerfrei abgeschlossen wurde.

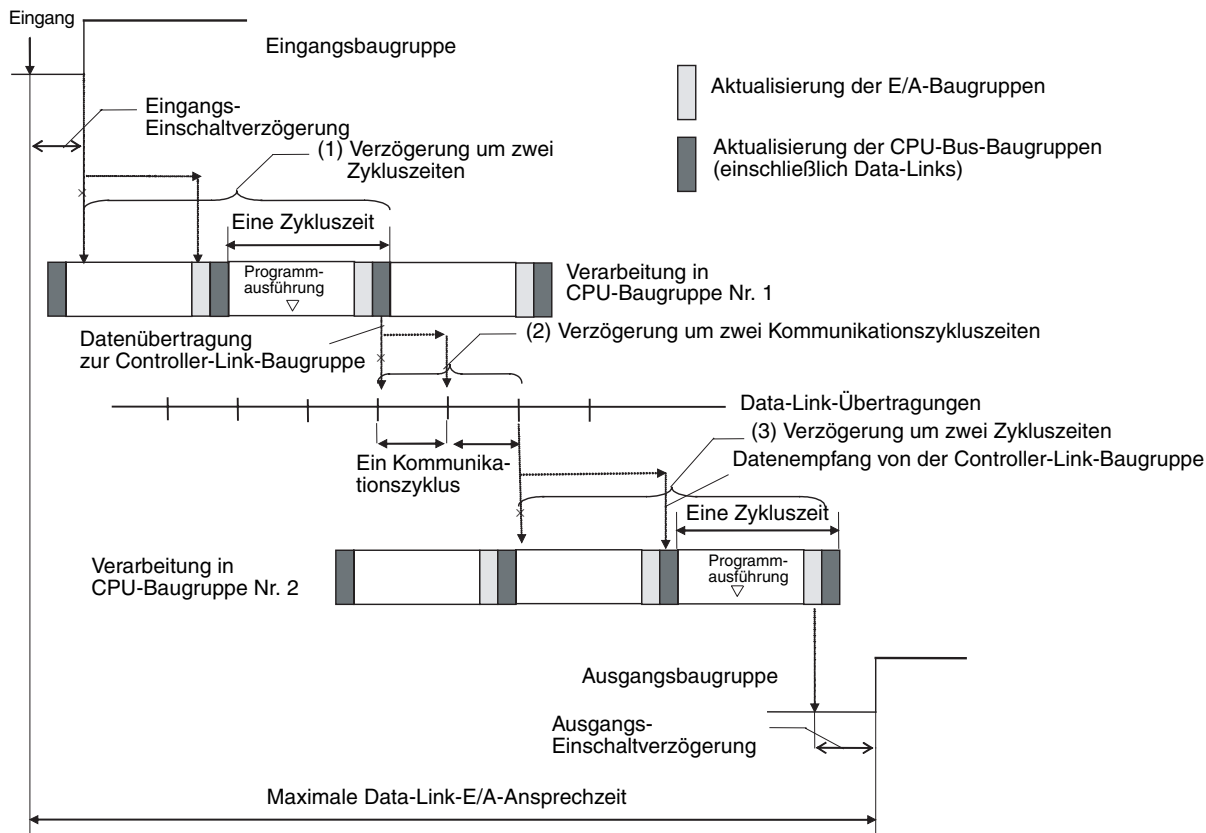


2. Mit dem Befehl IORF(097) werden die Daten der E/A-Baugruppen und Spezial-E/A-Baugruppen aktualisiert. Mit dem Befehl DLNK(226) werden CPU-Bus-Baugruppen aktualisiert (die den Baugruppen zugeordneten Worte im CIO- und DM-Bereich und die speziellen Daten der Baugruppen).

6-1-9 Maximale Data-Link-E/A-Ansprechzeit

Normale Verarbeitung

In der folgenden Abbildung wird der Datenfluss dargestellt, der zu der maximalen Data-Link-E/A-Ansprechzeit führt, wenn DLNK(226) nicht verwendet wird.



In der Abbildung werden drei Punkte dargestellt, an denen die Verarbeitung verzögert wird. Dadurch steigt die Data-Link-E/A-Ansprechzeit.

1,2,3...

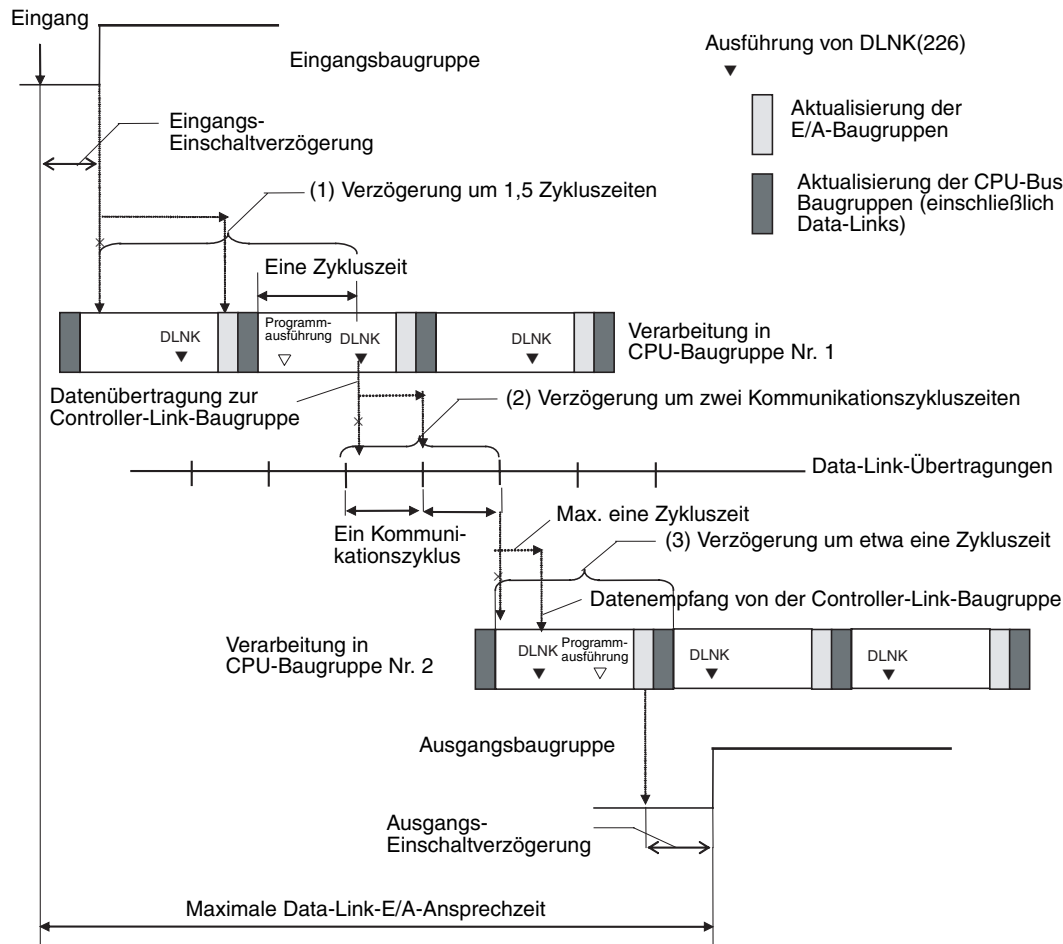
1. Der Eingang erreicht das SPS-System (CPU-Baugruppe Nr. 1) erst nach der E/A-Aktualisierung. Dadurch wird eine Verzögerung um einen Zyklus verursacht, bevor der Eingang in das SPS-System eingelesen wird. CPU-Bus-Baugruppen werden nach der Programmausführung aktualisiert. Dadurch wird eine Gesamtverzögerung um zwei Zykluszeiten verursacht.
2. Der Datenaustausch erfolgt erst, nachdem das SPS-System das Token weitergereicht hat, das es zum abfragenden Knoten macht. Dadurch wird eine Verzögerung von bis zu einer Kommunikationszykluszeit verursacht, bevor die Daten bei der Data-Link-Verarbeitung übertragen werden. Außerdem tritt nach dem Empfang des Tokens eine Verzögerung von bis zu einer Kommunikationszykluszeit auf. Dadurch beträgt die Gesamtverzögerung bis zu zwei Kommunikationszykluszeiten.
3. Die bei der Data-Link-Verarbeitung übertragenen Daten erreichen das SPS-System (CPU-Baugruppe Nr. 2) nach dem Datenaustausch, so dass die Daten bis zum nächsten Datenaustausch nicht vom SPS-System eingelesen werden. Dadurch wird eine Verzögerung von bis zu einem Zyklus verursacht. CPU-Bus-Baugruppen werden nach der Programmausführung aktualisiert. Dadurch wird eine Gesamtverzögerung um zwei Zykluszeiten verursacht.

Die maximale Data-Link-E/A-Ansprechzeit lässt sich wie folgt berechnen:

Eingangs-Einschaltverzögerung	1,5 ms
Zykluszeit der SPS mit CPU-Baugruppe Nr. 1 $\times 2$	25 ms $\times 2$
Kommunikationszykluszeit $\times 2$	10 ms $\times 2$
Zykluszeit der SPS mit CPU-Baugruppe Nr. 2 $\times 2$	20 ms $\times 2$
Ausgangs-Einschaltverzögerung	15 ms
Summe (Data-Link-E/A-Ansprechzeit)	126,5 ms

Verwendung von DLNK(226)

In der folgenden Abbildung wird der Datenfluss dargestellt, der zu der maximalen Data-Link-E/A-Ansprechzeit führt, wenn der Befehl DLNK(226) verwendet wird.



In der Abbildung werden drei Punkte dargestellt, an denen die Verarbeitung verzögert wird. Dadurch steigt die Data-Link-E/A-Ansprechzeit.

Hinweis

In diesem Beispiel wird davon ausgegangen, dass im Programm beider CPU-Baugruppen der Befehl DNLK(226) im Anschluss an die anderen Befehlen angegeben ist.

1,2,3...

1. Der Eingang erreicht das SPS-System (CPU-Baugruppe Nr. 1) erst nach der E/A-Aktualisierung. Dadurch wird eine Verzögerung um einen Zyklus verursacht, bevor der Eingang in das SPS-System eingelesen wird. CPU-Bus-Baugruppen werden während der Programmausführung aktualisiert. Dadurch wird die Gesamtverzögerung auf ungefähr 1,5 Zykluszeiten reduziert.
2. Der Datenaustausch erfolgt erst, nachdem das SPS-System das Token weitergereicht hat, das es zum abfragenden Knoten macht. Dadurch wird eine Verzögerung von bis zu einer Kommunikationszykluszeit verursacht, bevor die Daten bei der Data-Link-Verarbeitung übertragen werden. Außerdem tritt nach dem Empfang des Tokens eine Verzögerung von bis zu einer Kommunikationszykluszeit auf. Dadurch beträgt die Gesamtverzögerung bis zu zwei Kommunikationszykluszeiten.
3. Die bei der Data-Link-Verarbeitung übertragenen Daten erreichen die SPS (CPU-Baugruppe Nr. 2) nach der E/A-Aktualisierung, jedoch aktualisiert DLNK(226) die Daten, so dass sie ohne eine Verzögerung von bis zu einem Zyklus in die SPS eingelesen werden. Die E/A-Baugruppen werden nach der Programmausführung aktualisiert. Dadurch wird eine Gesamtverzögerung von ungefähr einer Zykluszeit verursacht.

Die maximale Data-Link-E/A-Ansprechzeit lässt sich wie folgt berechnen:

Eingangs-Einschaltverzögerung	1,5 ms	---
Zykluszeit der SPS mit CPU-Baugruppe Nr. 1 $\times 1,5$	$25 \text{ ms} \times 1,5$	Um 12,5 ms schneller (25 ms $\times 0,5$)
Kommunikationszykluszeit $\times 2$	$10 \text{ ms} \times 2$	---
Zykluszeit der SPS mit CPU-Baugruppe Nr. 2 $\times 1$	$20 \text{ ms} \times 1$	Um 20 ms schneller (20 ms $\times 1$)
Ausgangs-Einschaltverzögerung	15 ms	---
Summe (Data-Link-E/A-Ansprechzeit)	94 ms	Um 32,5 ms schneller (26 %)

6-1-10 Befehlsausführung im Hintergrund

Die Befehlsausführung im Hintergrund kann verwendet werden, um Schwankungen in der Zykluszeit zu reduzieren. Die Hintergrundausführung wird nur von CS1-H, CJ1-H und CJ1M CPU-Baugruppen sowie CS1D CPU-Baugruppen für Einzel-CPU-Systeme unterstützt.

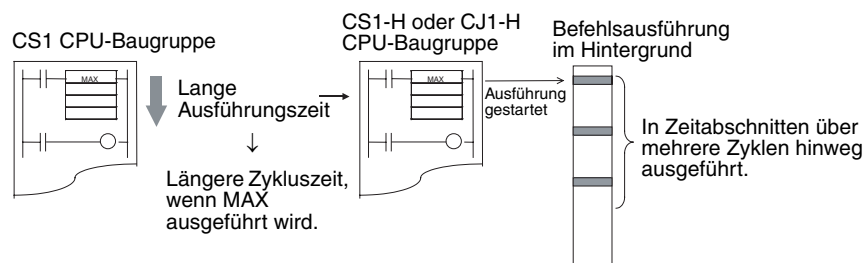
Tabellendatenverarbeitung (z. B. Suche nach Daten) und Textzeichenkettenverarbeitung (z. B. Suche nach Textzeichenketten) erfordern zur Ausführung eine bestimmte Zeit und können durch diese zusätzliche Ausführungszeit große Schwankungen in der Zykluszeit verursachen.

Jedoch können Sie mit den CS1-H, CJ1-H und CJ1M CPU-Baugruppen sowie CS1D CPU-Baugruppen für Einzel-CPU-Systeme (siehe Hinweis) die Ausführung der folgenden Befehle durch Hintergrundausführung (Ausführung in Zeitabschnitten) über mehrere Zyklen verteilen und dadurch Schwankungen in der Zykluszeit besser steuern. Sie können die Hintergrundausführung für die Befehlstypen im SPS-Setup einstellen.

Hinweis Die CS1D CPU-Baugruppen für Duplex-CPU-Systeme unterstützen keine Hintergrundausführung.

- Tabellendaten-Verarbeitungsbefehle
- Textzeichenketten-Verarbeitungsbefehle
- Datenverschiebungsbefehle (nur ASYNCHRONOUS SHIFT REGISTER)

Durch Ausführen der oben genannten Befehle im Hintergrund können Sie temporäre Zunahmen der Zykluszeit besser steuern.



Anwendungen

Verwenden Sie die Befehlsausführung im Hintergrund bei der Verarbeitung großer Datenmengen, z. B. beim Verknüpfen oder Verarbeiten von Daten, die nur zu bestimmten Zeitpunkten erforderlich sind (z. B. einmal am Tag), wenn die Reduzierung des Einflusses auf die Zykluszeit wichtiger als die Datenverarbeitungsgeschwindigkeit ist.

Vorgehensweise

1,2,3...

1. Aktivieren Sie im SPS-Setup die Hintergrundverarbeitung für die gewünschten Befehle.
2. Stellen Sie im SPS-Setup die Nummer der Kommunikationsschnittstelle (logische Schnittstellennummer) für die Hintergrundverarbeitung ein. Diese Schnittstellennummer wird für alle im Hintergrund ausgeführten Befehle verwendet.

Hinweis Für alle Hintergrundausführungen wird dieselbe Schnittstelle verwendet. Folglich kann kein neuer Befehl im Hintergrund ausgeführt werden, während ein anderer Befehl im Hintergrund ausgeführt wird. Steuern Sie Befehle, die im Hintergrund ausgeführt werden sollen, mit dem Kommunikationsschnittstellen-Frei-Merker, so dass nie mehrere Befehle gleichzeitig ausgeführt werden.

3. Wenn ein im Hintergrund auszuführender Befehl ausgeführt wird, beginnt die Ausführung in dem Zyklus, in dem die Ausführungsbedingung erfüllt ist, die Ausführung wird jedoch nicht in diesem Zyklus abgeschlossen.
4. Wenn die Hintergrundausführung startet, wird der Kommunikationsschnittstellen-Frei-Merker der entsprechenden Schnittstelle auf AUS gesetzt.
5. Die Hintergrundausführung wird über mehrere Zyklen hinweg fortgesetzt.
6. Wenn die Verarbeitung abgeschlossen ist, wird der Kommunikationsschnittstellen-Frei-Merker der entsprechenden Schnittstelle auf EIN gesetzt. Dadurch können nun andere Befehle im Hintergrund ausgeführt werden.

Anwendbare Befehle

■ Tabellendaten-Verarbeitungsbefehle

Befehl	AWL	Funktionscode
DATA SEARCH	SRCH	181
SWAP BYTES	SWAP	637
FIND MAXIMUM	MAX	182
FIND MINIMUM	MIN	183
SUM	SUM	184
FRAME CHECKSUM	FCS	180

■ Textzeichenketten-Verarbeitungsbefehle

Befehl	AWL	Funktionscode
MOVE STRING	MOV\$	664
CONCATENATE STRING	+\$	656
GET STRING LEFT	LEFT\$	652
GET STRING RIGHT	RIGHT\$	653
GET STRING MIDDLE	MID\$	654
FIND IN STRING	FIND\$	660
STRING LENGTH	LEN\$	650
REPLACE IN STRING	RPLC\$	661
DELETE STRING	DEL\$	658
EXCHANGE STRING	XCHG\$	665
CLEAR STRING	CLR\$	666
INSERT INTO STRING	INS\$	657

■ Datenverschiebepbefehle

Befehl	AWL	Funktionscode
ASYNCHRONOUS SHIFT REGISTER	ASFT	017

Unterschiede zwischen normaler und Hintergrundausführung von Befehlen

Im Folgenden werden die Unterschiede zwischen der normalen Befehlsausführung und der im Hintergrund aufgelistet.

■ Ausgabe in Indexregister (IR)

Wenn Sie die E/A-Speicherzuordnungsadresse des Worts, das den kleinsten oder größten Wert enthält, mit MAX(182) oder MIN(183) in ein Indexregister ausgeben möchten, wird die Adresse nicht in das Indexregister, sondern in A595 und A596 geschrieben. Kopieren Sie die Adresse gegebenenfalls mit einem Kopierbefehl (z. B. MOVL(498)) von A595 und A596 in ein Indexregister.

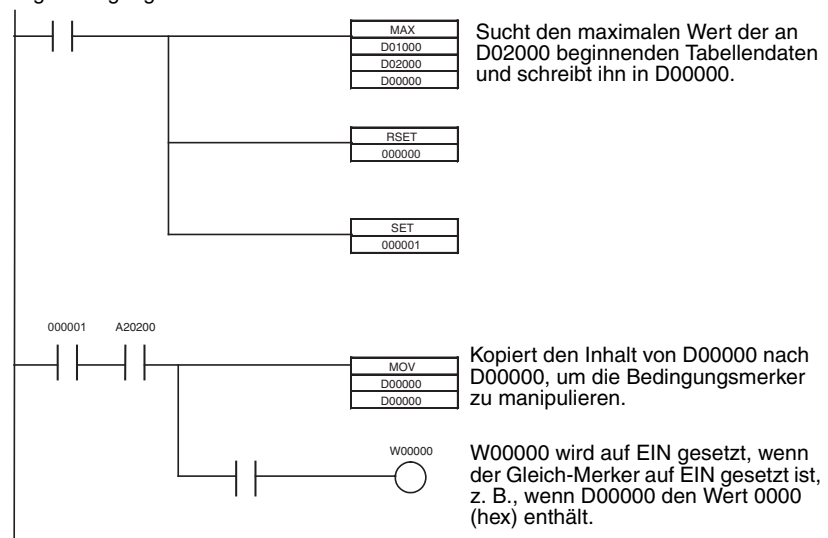
■ Bedingungsmerker

Bei der Ausführung von Befehlen im Hintergrund werden Bedingungsmerker nicht aktualisiert. Wenn Sie auf den Status eines Bedingungsmerkers zugreifen möchten, führen Sie einen Befehl aus, der die Bedingungsmerker auf dieselbe Weise beeinflusst, und greifen Sie anschließend auf die Bedingungsmerker zu. Im folgenden Beispiel wird dieses Vorgehen dargestellt.

Beispiel:

MOV(021) beeinflusst die Gleich- und Negativ-Merker auf dieselbe Weise wie MAX(182), beispielsweise setzen beide den Gleich-Merker für 0 auf EIN und den Negativ-Merker auf EIN, wenn MSB (das äußerstes linkes Bit) auf EIN gesetzt ist. Somit können Sie MOV(021) verwenden, um die Ergebnisse von MAX(182) an dieselbe Adresse zu kopieren, um die Bedingungsmerker zu manipulieren, so dass der Status abgefragt werden kann.

Ausführungsbedingung



■ Ausgabe in Indexregister IR00

Wenn Sie die E/A-Speicherzuordnungsadresse des Wortes, das den gesuchten Wert (das erste Wort, wenn mehrere vorhanden sind) enthält, mit SRCH(181) in ein Indexregister ausgeben möchten, wird die Adresse nicht in das Indexregister, sondern in A595 und A596 geschrieben.

■ **Ausgabe in Datenregister (DR) für SRCH(181)**

Wenn Sie die gesuchten Daten mit SRCH(181) in ein Datenregister ausgeben möchten, werden die Daten nicht in das Datenregister, sondern in A597 geschrieben.

■ Gesuchte Textzeichenketten

Wenn SRCH(181) die gesuchten Daten findet, wird nicht der Gleich-Merker, sondern A59801 auf EIN gesetzt.

■ Befehlsfehler

Wenn bei der Hintergrundaufführung eines Befehls ein Befehls- oder Zugriffsfehler auftritt, wird nicht der P_ER- bzw. P_AER-Merker, sondern A39510 auf EIN gesetzt. A39510 bleibt bis zur nächsten Hintergrundaufführung eines Befehls auf EIN gesetzt.

■ Ausgabe in Datenregister (DR) für MAX(182) oder MIN(183)

Wenn Sie mit MAX(182) oder MIN(183) den größten bzw. kleinsten Wert in ein Datenregister ausgeben möchten, löst das einen Befehlsfehler aus und der P_ER-Merker wird auf EIN gesetzt.

SPS-Setup

Wort	Bits	Bezeichnung	Einstellung	Standard-einstellung und Aktualisierungszeitpunkt
198	15	Tabellendaten-Befehlsausführung im Hintergrund	0: Keine Verarbeitung im Hintergrund 1: Verarbeitung im Hintergrund	0: Keine Verarbeitung im Hintergrund Zu Beginn der Programmausführung
	14	Textzeichenketten-Befehlsausführung im Hintergrund	0: Keine Verarbeitung im Hintergrund 1: Verarbeitung im Hintergrund	
	13	Datenverschiebe-Befehlsausführung im Hintergrund	0: Keine Verarbeitung im Hintergrund 1: Verarbeitung im Hintergrund	
	00 bis 03	Nummer der Kommunikationsschnittstelle für Hintergrundausführung	0 bis 7 Hex: Kommunikationsschnittstelle 0 bis 7 (interne logische Schnittstellen)	0 Hex: Schnittstelle 0 Zu Beginn der Programmausführung

Merker und Worte des Zusatz-Systembereichs

Bezeichnung	Adresse	Beschreibung
Kommunikationsschnittstellen-Frei-Merker	A20200 bis A20207	Wird auf EIN gesetzt, wenn die jeweilige Kommunikationsschnittstelle bereit (frei) für einen Netzwerkbefehl (SEND, RECV, CMND oder PMCR) oder eine Hintergrundausführung (nur CS1D CPU-Baugruppen für Einzel-CPU-Systeme und CS1-H, CJ1-H oder CJ1M CPU-Baugruppen) ist. Die Bits 00 bis 07 entsprechen den Kommunikationsschnittstellen 0 bis 7. Nur CS1-H, CJ1-H, CJ1M und CS1D CPU-Baugruppen: Bei Verwendung der Easy-Backup-Funktion zum Beschreiben oder Vergleichsfunktion einer Speicherkarte wird automatisch eine Kommunikationsschnittstelle belegt und der entsprechende Merker auf AUS gesetzt. Nach Abschluss des Vorgangs wird der Merker wieder auf EIN gesetzt.
Kommunikationsschnittstellen-Fehlermerker	A21900 bis A21907	Wird auf EIN gesetzt, wenn bei der Ausführung eines Netzwerkbefehls (SEND, RECV, CMND oder PMCR) ein Fehler auftritt. Die Bits 00 bis 07 entsprechen den Kommunikationsschnittstellen 0 bis 7. Nur CS1-H, CJ1-H, CJ1M und CS1D CPU-Baugruppen: Bei Verwendung der Easy-Backup-Funktion zum Beschreiben oder Überprüfen einer Speicherkarte wird automatisch eine Kommunikationsschnittstelle belegt. Beim Auftreten eines Fehler wird der jeweilige Merker auf EIN gesetzt. Wenn der Sicherungsvorgang normal abgeschlossen wird, wird der Merker auf AUS gesetzt.

Bezeichnung	Adresse	Beschreibung
Kommunikationsschnittstellen-Abschluss-Codes	A203 bis A210	Nach der Ausführung eines Netzwerkbefehls (SEND, RECV, CMND oder PMCR) mit der jeweiligen Kommunikationsschnittstelle wird in diesen Worten ein Abschluss-Code für den Vorgang gespeichert. Die Inhalte werden gelöscht, wenn die Hintergrundausführung normal abgeschlossen ist (bei CS1D CPU-Baugruppen für Einzel-CPU-Systeme und CS1-H, CJ1-H und CJ1M CPU-Baugruppen). Die Worte A203 bis A210 entsprechen den Kommunikationsschnittstellen 0 bis 7. Nur CS1-H, CJ1-H, CJ1M und CS1D CPU-Baugruppen: Bei Verwendung der Easy-Backup-Funktion zum Beschreiben oder Überprüfen einer Speicherkarte wird automatisch eine Kommunikationsschnittstelle belegt. Nach Abschluss des Vorgangs wird der Abschluss-Code im entsprechenden Wort gespeichert.
Hintergrundausführung P_ER/ P_AER-Merker	A39510	Wird auf EIN gesetzt, wenn bei einem im Hintergrund ausgeführten Befehl ein Befehls- oder Zugriffsfehler auftritt. Wird auf AUS gesetzt, wenn die Spannungsversorgung eingeschaltet oder der Betrieb gestartet wird.
Hintergrundausführungs-IR00-Ausgabe	A595 und A596	Wenn Sie als Ausgabeziel für einen im Hintergrund ausgeführten Befehl ein Indexregister angeben, wird die Ausgabe in diese Worte geschrieben. Es erfolgt keine Ausgabe in IR00. Bereich: 0000 0000 bis FFFF FFFF Hex Die niederwertigen 4 Stellen: A595, die höherwertigen 4 Stellen: A596
Hintergrundausführungs-DR00-Ausgabe	A597	Wenn Sie als Ausgabeziel für einen im Hintergrund ausgeführten Befehl ein Datenregister angeben, wird die Ausgabe in dieses Wort geschrieben. Es erfolgt keine Ausgabe in DR00. Bereich: 0000 bis FFFF hex
Hintergrundausführungs-Gleich-Merker-Ausgabe	A59801	Wird auf EIN gesetzt, wenn bei einer im Hintergrund ausgeführten Suche mit SRCH(181) übereinstimmende Daten gefunden werden.

Hinweis Die Kommunikationsschnittstellen (interne logische Schnittstellen) der CPU-Baugruppe werden bei der Hintergrundausführung und bei den folgenden Befehlen verwendet:

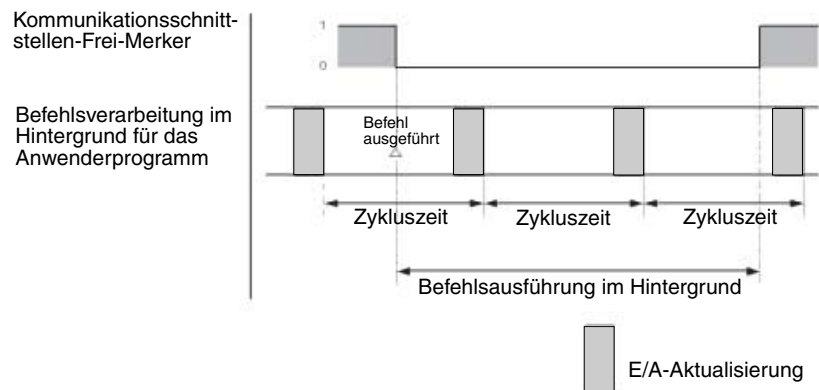
- SEND(090), RECV(098) und CMND(490) (Netzwerkkommunikationsbefehle)
- PMCR(260) (PROTOCOL MACRO)

Sie können Hintergrundbefehle und die oben aufgeführten Befehle nicht gleichzeitig mit der selben Schnittstelle ausführen. Stellen Sie mit Hilfe der Kommunikationsschnittstellen-Frei-Merker sicher, dass an jeder Schnittstelle jeweils nur ein Befehl ausgeführt wird.

Hinweis Wenn Sie einen Hintergrundbefehl mit einer Schnittstelle ausführen möchten, für die der Kommunikationsschnittstellen-Frei-Merker auf AUS gesetzt ist, wird der P_ER-Merker auf EIN gesetzt und der Hintergrundbefehl wird nicht ausgeführt.

Kommunikationsschnittstellen-Frei-Merker

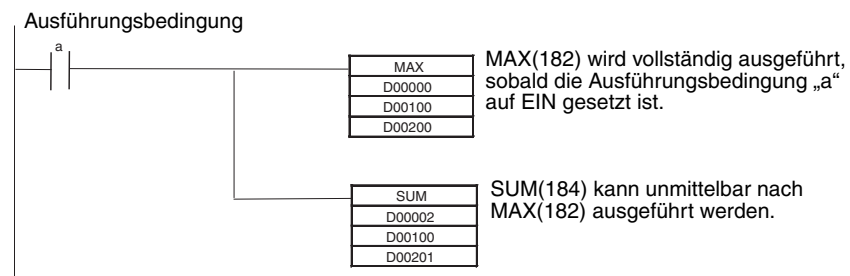
Ein Kommunikationsschnittstellen-Frei-Merker ist auf EIN gesetzt, wenn die entsprechende Schnittstelle nicht verwendet wird und auf AUS, wenn an der Schnittstelle eine Verarbeitung stattfindet.



Programmierbeispiel 1

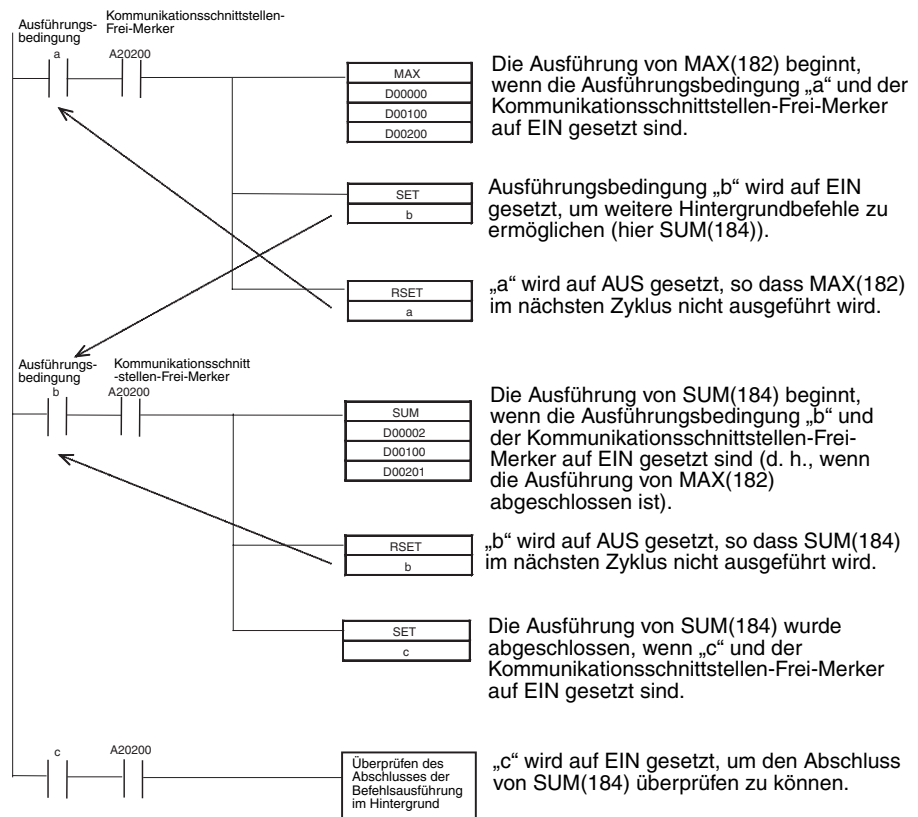
■ Konventionelle Programmierung ohne Hintergrundauführung

Wie unten gezeigt, wird bei der Ausführung eines Befehls die Verarbeitung vollständig abgeschlossen.



■ Programmierung mit Hintergrundauführung

Mit Hintergrundauführung wird das Programm so geändert, dass MAX(182) nur ausgeführt wird, wenn der entsprechende Kommunikationsschnittstellen-Frei-Merker auf EIN gesetzt ist (d. h., nur wenn die Schnittstelle nicht bereits für eine Befehlsausführung im Hintergrund oder zur Netzwerkkommunikation verwendet wird). Außerdem werden Eingangsbedingungen mit SET- und RSET-Befehlen gesteuert, um sicherzustellen, dass die Verarbeitung in der richtigen Reihenfolge stattfindet. (Im folgenden Beispiel wird für die Befehlsausführung im Hintergrund die Kommunikationsschnittstelle 0 verwendet.)

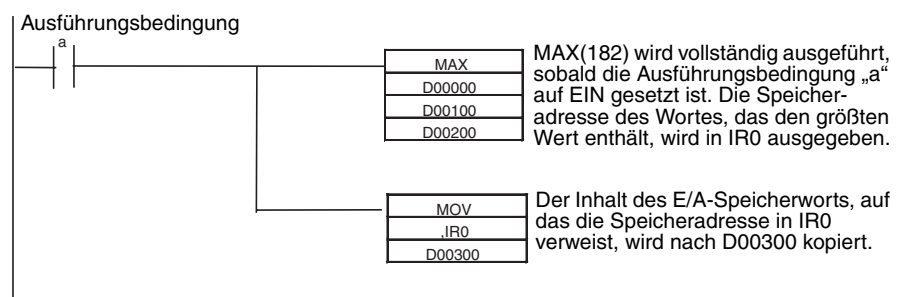


Programmierbeispiel 2

In diesem Beispiel wird die Hintergrundauführung mit einer Ausgabe in ein Indexregister dargestellt. Dies ist für die Befehle MAX(182), MIN(183) und SRCH(181) möglich.

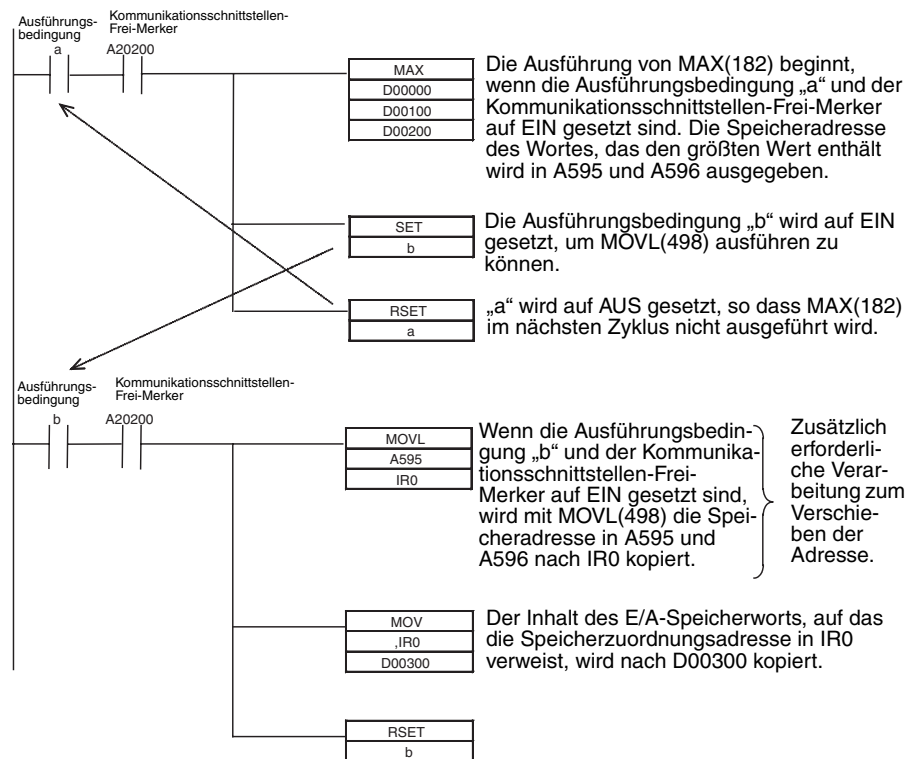
■ Konventionelle Programmierung ohne Hintergrundauführung

Wie unten gezeigt, wird die Speicheradresse des Wortes, das den größten Wert enthält, in ein Indexregister ausgegeben.



■ Programmierung mit Hintergrundauführung

Mit Hintergrundauführung wird die Speicherzuordnungsadresse des Wortes, das den größten Wert enthält, in A595 und A596 ausgegeben. Anschließend wird die Speicherzuordnungsadresse mit MOVL(498) in das Indexregister kopiert.



6-1-11 Gemeinsame Nutzung von Index- und Datenregistern in verschiedenen Tasks

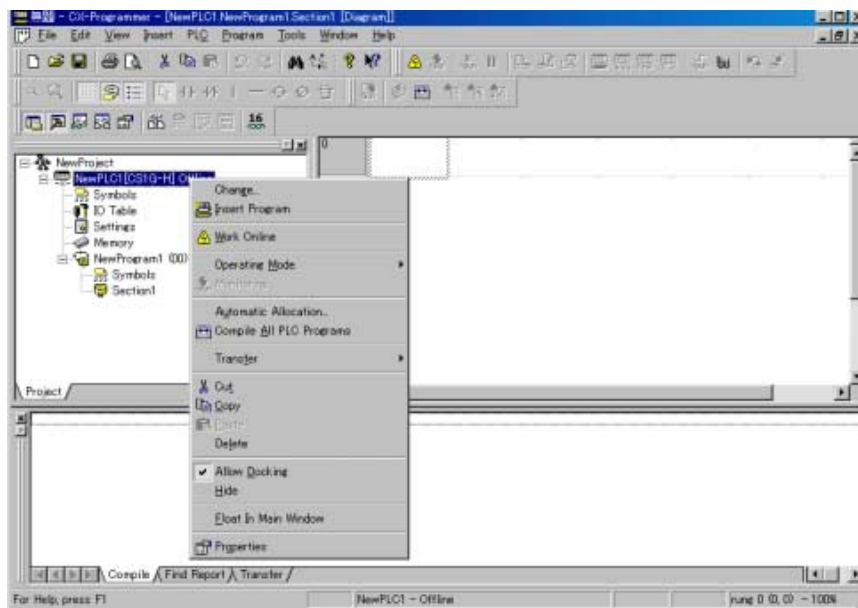
Die gemeinsame Nutzung der Index- und Datenregister (IR/DR) in verschiedenen Tasks wird nur von den CS1-H, CJ1-H, CJ1M und CS1D CPU-Baugruppen unterstützt. In der Standardeinstellung verwenden alle Tasks unabhängig die Register. Die aktuelle Einstellung können Sie in A09914 überprüfen.

- Hinweis**
1. Mit gemeinsam genutzten Index- und Datenregistern können Sie das Speichern und Laden der Registerinhalte beim Task-Wechsel überflüssig machen, wenn Sie in mehreren Tasks dieselben Inhalte verwenden möchten. Weitere Informationen über das Speichern und Laden der Inhalte von Indexregistern finden Sie im *CS-Serie Bedienerhandbuch* (W339) oder im *CJ-Serie Bedienerhandbuch* (W393) im jeweiligen Abschnitt über Indexregister.
 2. Die Umschaltzeit beim Task-Wechsel ist etwas kürzer, wenn die Index- und Datenregister gemeinsam genutzt werden. Es wird empfohlen, die gemeinsame Nutzung der Register einzustellen, wenn die Register nicht verwendet werden oder wenn in den einzelnen Tasks keine getrennten Register erforderlich sind.

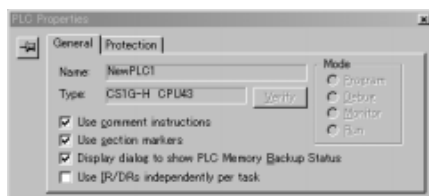
Einstellverfahren

Stellen Sie die gemeinsame Nutzung der Index- und Datenregister mittels CX-Programmer ein. Mittels einer Programmierkonsole ist dies nicht möglich.

- 1,2,3...**
1. Klicken Sie in der Projektstruktur von CX-Programmer mit der rechten Maustaste auf eine SPS.



2. Wählen Sie **Eigenschaften** aus. Das folgende Dialogfeld wird angezeigt.



3. Aktivieren Sie das Kontrollkästchen „I/R/DRs für jede Task allein verwenden“, wenn für jede Task getrennte Index und Datenregister erforderlich sind. Deaktivieren Sie das Kontrollkästchen, um die Index- und Datenregister in allen Tasks gemeinsam zu nutzen.

Merker und Worte des Zusatz-Systembereichs

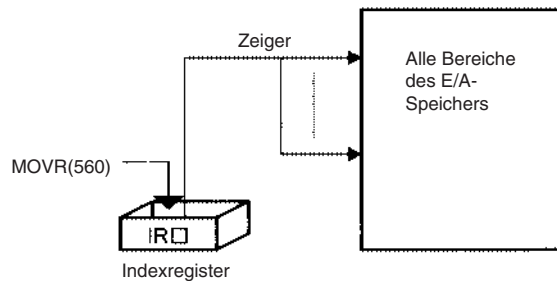
Bezeichnung	Adresse	Beschreibung
Task-übergreifende Index- und Datenregister	A09914	Gibt an, ob die Index- und Datenregister von verschiedenen Tasks gemeinsam genutzt werden. 0: Getrennte Benutzung der Register für jede Task (Standardeinstellung) 1: Von allen Tasks gemeinsam genutzte Register.

6-2 Indexregister

6-2-1 Zweck von Indexregistern

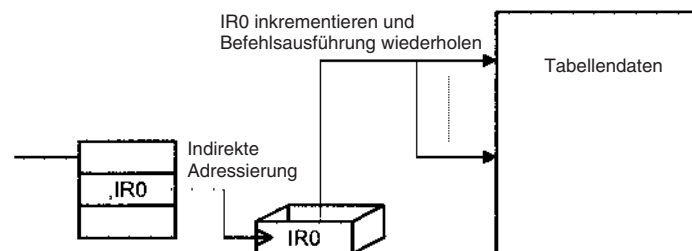
Indexregister dienen als Zeiger zur Spezifikation von SPS-Speicheradressen, die absolute Speicheradressen im E/A-Speicher sind. Geben Sie ein Indexregister nach dem Speichern einer SPS-Speicheradresse in dem Indexregister mit MOVR(560) oder MOVRW(561) als Operand eines anderen Befehls an, um die gespeicherte SPS-Speicheradresse indirekt zu adressieren.

Der Vorteil von Indexregistern ist, dass Sie mit ihnen jedes Bit oder Wort im E/A-Speicher spezifizieren können, einschließlich Zeitgeber- und Zähler-Istwerten.



6-2-2 Verwenden von Indexregistern

Indexregister können in Kombination mit Schleifen (z. B. FOR-NEXT) eine leistungsfähige Funktion bilden. Die Inhalte der Indexregister können sehr einfach erhöht, verringert und mit einem Offset versehen werden. Dadurch können Schleifen mit wenigen Befehlen Tabellen mit aufeinander folgenden Daten sehr effizient verarbeiten.



Grundfunktion

Im Wesentlichen werden Indexregister wie folgt verwendet:

1,2,3...

1. Speichern Sie die SPS-Speicheradresse des gewünschten Bits oder Wortes mit MOVR(560) in ein Indexregister.
2. Geben Sie das Indexregister als Operand eines Befehls an, um das gewünschte Bit oder Wort indirekt zu adressieren. Dies wird von fast allen Befehlen unterstützt.
3. Erhöhen Sie die ursprüngliche SPS-Speicheradresse oder geben Sie einen Offsetwert an, um den Zeiger auf eine andere Adresse umzuleiten (siehe unten).
4. Wiederholen Sie die Schritte 2 und 3, um den Befehl für eine beliebige Anzahl an Adressen auszuführen.

Arbeiten mit indirekter Adressierung (Erhöhen, Verringern und Offsetwerte)

In der folgenden Tabelle werden die verfügbaren Varianten der indirekten Adressierung dargestellt.

Variante	Syntax
Indirekte Adressierung	,IR□
Indirekte Adressierung mit konstantem Offset	Konstante ,IR□ (Geben Sie ggf. ein Vorzeichen für die Konstante an.)
Indirekte Adressierung mit Datenregister-Offset	DR□,IR□
Indirekte Adressierung mit Auto-Inkrement	Inkrementierung um 1: ,IR□+ Inkrementierung um 2: ,IR□++
Indirekte Adressierung mit Auto-Dekrement	Dekrementierung um 1: ,-IR□ Dekrementierung um 2: ,- -IR□

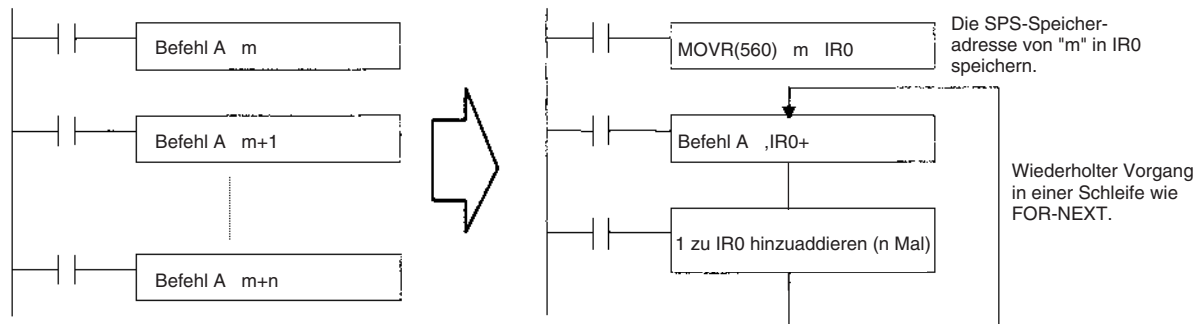
Befehle mit direkter Adressierung von Indexregistern

Mit den folgenden Befehlen können Sie Indexregister direkt adressieren.

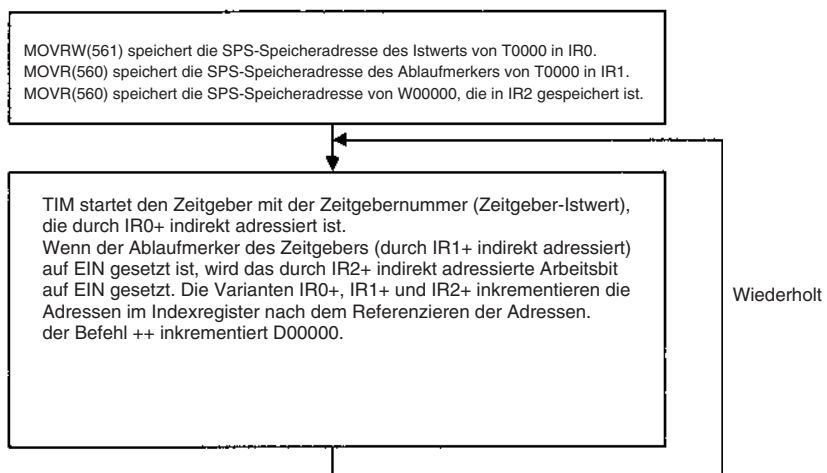
DOUBLE SIGNED BINARY ADD WITHOUT CARRY: +L(401), DOUBLE SIGNED BINARY SUBTRACT WITHOUT CARRY: -L(411), DOUBLE INCREMENT BINARY: ++L(591) und DOUBLE DECREMENT BINARY: --L(593)

Beispiel 1

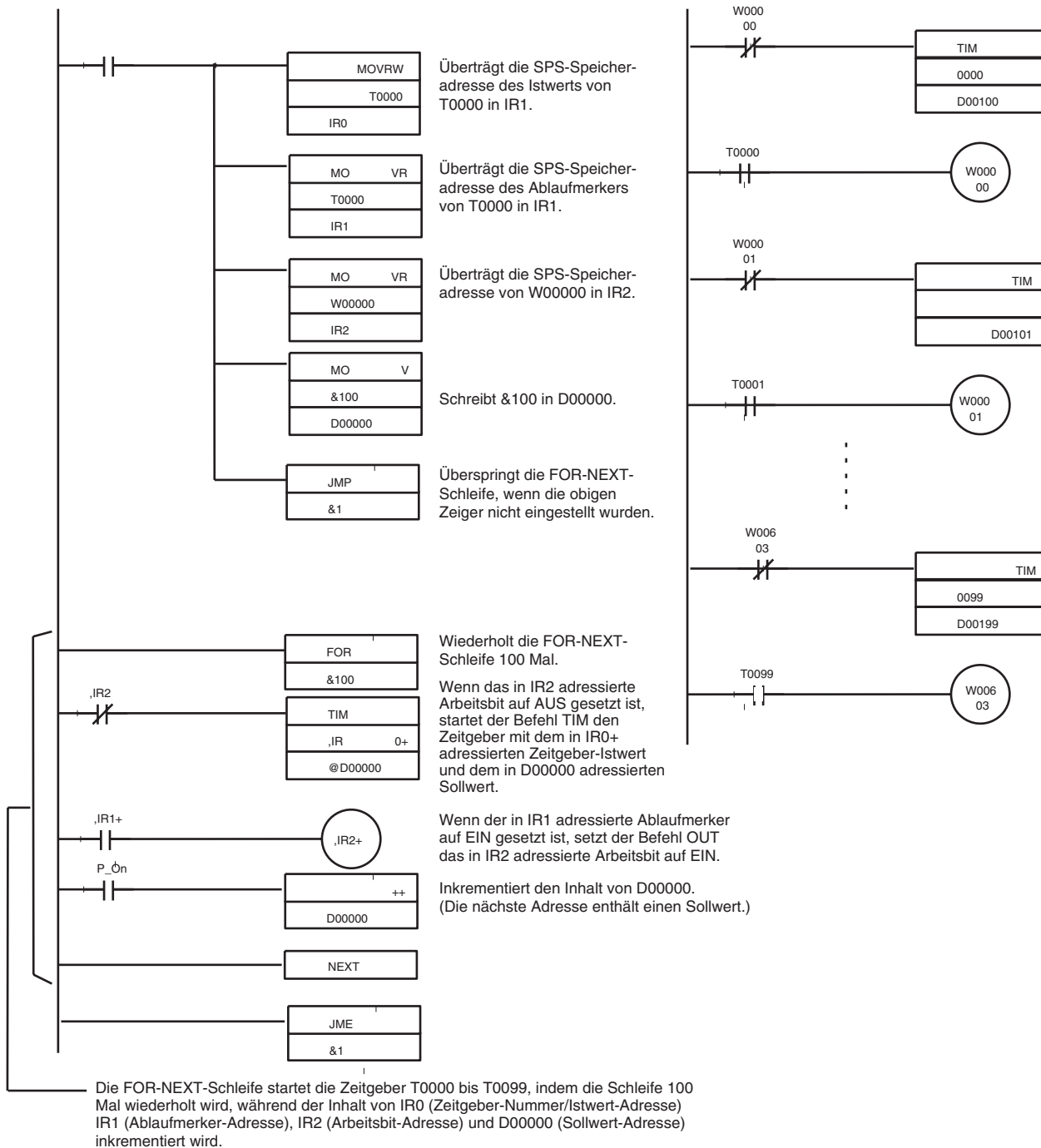
Im folgenden Beispiel wird dargestellt, wie Sie eine lange Befehlsfolge durch ein Indexregister in einer Programmschleife ersetzen können. In diesem Fall wird Befehl A n+1-mal wiederholt, um bestimmte Operationen durchzuführen, wie z. B. die Werte einer Tabelle zu lesen oder zu vergleichen.

**Beispiel 2**

Im folgenden Beispiel werden mit Indexregistern in einer FOR-NEXT-Schleife 100 Zeitgeber (T0000 bis T0099) mit den in D00100 bis D00199 enthaltenen Sollwerten definiert und gestartet. Für jeden Zeitgeber werden die Zeitgebernummer und der Ablauf-Merker in einem Indexregister angegeben. Bei jeder Wiederholung der Schleife werden die Indexregister um eins inkrementiert.



Das aus nur 11 Befehlen bestehende Unterprogramm auf der linken Seite leistet dasselbe wie das aus 200 Befehlen bestehende Unterprogramm aus der rechten Seite.



Direkte Adressierung von Indexregistern

Indexregister können nur mit den in der folgenden Tabelle aufgeführten Befehlen direkt adressiert werden.

Befehlstyp	Bezeichnung des Befehls	AWL	Hauptfunktion
Kopierbefehle	MOVE TO REGISTER	MOVR(560)	Speichert die SPS-Speicheradresse eines Bits oder Wortes in einem Indexregister.
	MOVE TIMER/COUNTER PV TO REGISTER	MOVW(561)	
Tabellendaten-Verarbeitungsbefehle	SET RECORD LOCATION	SETR(635)	Gibt die in einem Indexregister gespeicherte SPS-Speicheradresse aus.
	GET RECORD NUMBER	GETR(636)	
Kopierbefehle	DOUBLE MOVE	MOVL(498)	Überträgt zwischen Indexregistern. Für Datenaustausch und Vergleich.
	DOUBLE DATA EXCHANGE	XCGL(562)	
Vergleichsbefehle	DOUBLE EQUAL	=L(301)	
	DOUBLE NOT EQUAL	< >L(306)	
	DOUBLE LESS THAN	< L(311)	
	DOUBLE LESS THAN OR EQUAL	< =L(316)	
	DOUBLE GREATER THAN	>L(321)	
	DOUBLE GREATER THAN OR EQUAL	> =L(326)	
	DOUBLE COMPARE	CMPL(060)	
Inkrementier- und Dekrementierbefehle	DOUBLE INCREMENT BINARY	++L(591)	Ändert die SPS-Speicheradresse im Indexregister durch Inkrementieren, Dekrementieren oder durch einen Offsetwert.
	DOUBLE DECREMENT BINARY	--L(593)	
Arithmetische Befehle	DOUBLE SIGNED BINARY ADD WITHOUT CARRY	+L(401)	
	DOUBLE SIGNED BINARY SUBTRACT WITHOUT CARRY	-L(411)	
Spezialbefehle	CONVERT ADDRESS FROM CV	FRMCV(284)	Konvertiert die tatsächlichen SPS-Speicheradressen zwischen Adressen der CV-Serie und Adressen der CS/CJ-Serie. (nur CS1-H, CJ1-H, CJ1M und CS1D CPU-Baugruppen)
	CONVERT ADDRESS TO CV	TOCV(285)	

Hinweis Befehle für Operanden doppelter Länge (d. h. solche, die mit einem „L“ enden) werden bei den Indexregister IR0 bis IR15 verwendet, da jedes Register zwei Worte enthält.

6-2-3 Verarbeitung mit Indexregistern

Die Tabellendaten-Verarbeitungsbefehle der CS/CJ-Serie CPU-Baugruppe ergänzen die Funktionen der Indexregister. Diese Befehle können in Stapelverarbeitungs- und Tabellenverarbeitungsbefehle unterteilt werden.

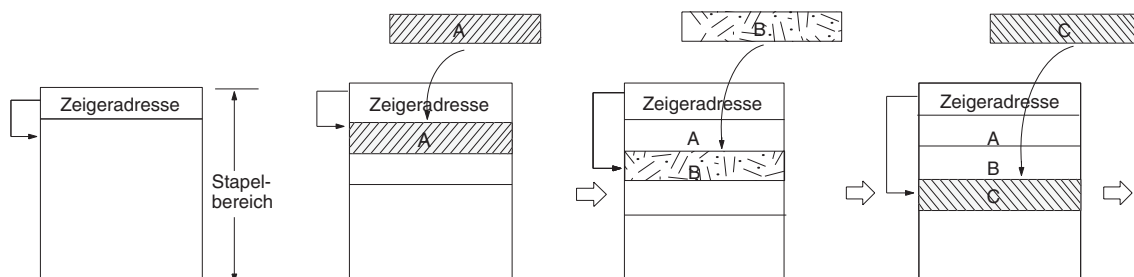
Details	Zweck	Befehle
Stapelverarbeitung	Verarbeiten FIFO- oder LIFO-Datentabellen (First-In First-Out bzw. Last-In First-Out) und können Dateneinträge in Datentabellen lesen, schreiben, einfügen, löschen oder zählen.	SSET(630), PUSH(632), FIFO(633), LIFO(634) und (nur bei CS1-H, CJ1-H, CJ1M und CS1D CPU-Baugruppen) SREAD(639), SWRITE(640), SINS(641), SDEL(642), SNUM(638).

Details			Zweck	Befehle
Tabellen- verarbei- tung	Tabellen mit Ein-Wort- Datensätzen (Bereichs- befehle)	Grund- befehle	Suchen nach Werten wie der Prüfsumme, einem bestimmten Wert, dem größten Wert oder dem kleinsten Wert im Bereich.	FCS(180), SRCH(181), MAX(182), MIN(183) und SUM(184).
		Spezielle Befehle	Durchführen weiterer Tabellenverarbeitungen, wie Vergleichen oder Sortieren.	Kombination von Indexregistern mit Befehlen wie SRCH(181), MAX(182), MIN(183) und Vergleichsbefehlen.
	Tabellen mit Datensätzen aus mehreren Worten (Datensatztabellen-Befehle)		Verarbeiten von Daten in Datensätzen, die aus mehreren Worten bestehen.	Kombination von Indexregistern mit Befehlen wie DIM(631), SETR(635), GETR(636) und Vergleichsbefehlen.

Stapelverarbeitung

Stapelbefehle verarbeiten speziell definierte Datentabellen, die Stapel genannt werden. Daten können nach dem FIFO- oder dem LIFO-Prinzip (First-In First-Out bzw. Last-In First-Out) aus einem Stapel gezogen werden.

Ein bestimmter Bereich des E/A-Speichers muss dafür als Stapel definiert werden. Die ersten Worte des Stapels geben die Länge des Stapels an und enthalten den Stapelzeiger. Bei jedem Schreiben von Daten in den Stapel wird der Stapelzeiger inkrementiert, wodurch er die nächste Adresse angibt, an der Daten gespeichert werden können.



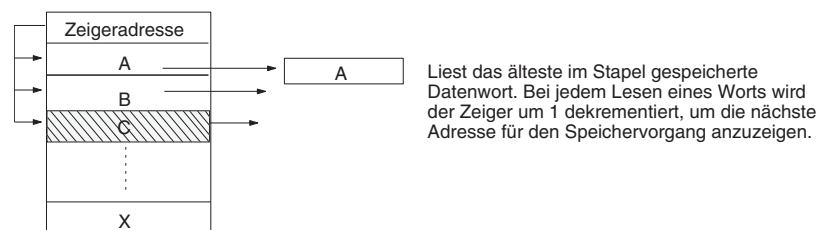
(In der Abbildung oben wird der Status des Zeigers vor dem Hinzufügen der Daten dargestellt.)

Hinweis

Tatsächlich enthalten die ersten beiden Worte des Stapels die SPS-Speicheradresse des letzten Wortes im Stapel und das folgende Doppelwort enthält den Stapelzeiger.

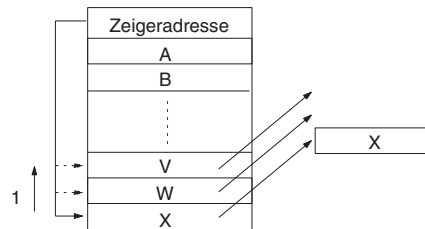
FIFO-Verarbeitung (First-In First-Out)

In der folgenden Abbildung wird die Funktionsweise eines FIFO-Stapels dargestellt.



LIFO-Verarbeitung (Last-In First-Out)

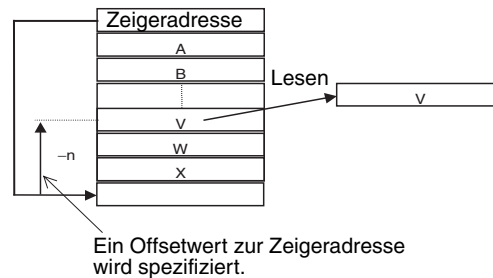
In der folgenden Abbildung wird die Funktionsweise eines LIFO-Stapels dargestellt.



Liest das neueste im Stapel gespeicherte Datenwort. Bei jedem Lesen eines Worts wird der Zeiger um 1 dekrementiert, um die nächste Adresse für den Speichervorgang anzuzeigen. Die Daten an der gelesenen Position bleiben unverändert.

Bearbeiten bestimmter Tabellendaten

Sie können einzelne Einträge in einer Tabelle lesen, schreiben, einfügen oder löschen. In der folgenden Abbildung wird ein Lesevorgang dargestellt.



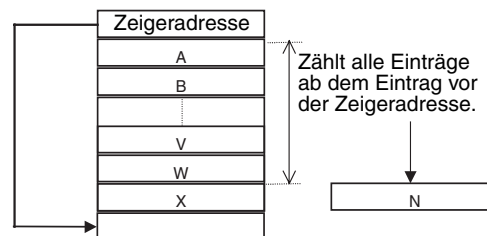
Daten werden ausgehend von der Zeigeradresse mit einem bestimmten Offset aus der Tabelle gelesen.

Die Bearbeitung bestimmter Tabellendaten kann beispielsweise bei der Objektverfolgung auf einem Förderband verwendet werden.

Ein Offsetwert zur Zeigeradresse wird spezifiziert.

Zählen von Tabellendaten

In der folgenden Abbildung wird dargestellt, wie in einer Datentabelle gezählt wird.



Die Anzahl der Einträge in der Datentabelle wird ab dem Eintrag unmittelbar vor der Zeigeradresse bis zum Anfang der Tabelle gezählt.

Dadurch können Sie beispielsweise die Anzahl der Objekte auf einem Förderband zählen.

Stapelbefehle

In der folgenden Tabelle werden die Stapelbefehle und ihre Funktionen aufgelistet. Zu den typischen Anwendungsgebieten für Stapel zählen die Verarbeitung von Regalinformationen für automatische Lagerhaussysteme, die Verarbeitung von Testergebnissen und die Verwaltung von Informationen über die Werkstücke auf einem Förderband.

Befehl	Funktion
SSET(630)	Definiert einen Stapelbereich.
PUSH(632)	Schreibt Daten in die mit dem Stapelzeiger angezeigte Adresse.
FIFO(633)	Liest Daten nach dem FIFO-Prinzip (also den ersten, ältesten Datensatz zuerst) aus dem Stapel.
LIFO(634)	Liest Daten nach dem LIFO-Prinzip (also den letzten, neuesten Datensatz zuerst) aus dem Stapel.
SREAD(639)	Liest einen bestimmten Eintrag aus der Tabelle (nur CS1-H, CJ1-H, CJ1M und CS1D CPU-Baugruppen).
SWRITE(640)	Schreibt einen bestimmten Eintrag in die Tabelle (nur CS1-H, CJ1-H, CJ1M und CS1D CPU-Baugruppen).
SINS(641)	Fügt einen bestimmten Eintrag in die Tabelle ein (nur CS1-H, CJ1-H, CJ1M und CS1D CPU-Baugruppen).
SDEL(642)	Löscht einen bestimmten Eintrag aus der Tabelle (nur CS1-H, CJ1-H, CJ1M und CS1D CPU-Baugruppen).
SNUM(638)	Zählt die Anzahl der Einträge in der Tabelle (nur CS1-H, CJ1-H, CJ1M und CS1D CPU-Baugruppen).

Tabellenverarbeitung (Bereichsbefehle)

Die Bereichsbefehle verarbeiten einen Bereich von Worten, der als Tabelle von Ein-Wort-Datensätzen betrachtet werden kann. Mit diesen Befehlen können Sie Grundfunktionen durchführen, z. B. in dem Bereich nach dem größten bzw. kleinsten oder einem bestimmten Wert suchen oder die Summe oder Prüfsumme berechnen.

Die SPS-Speicheradresse des Ergebnisworts (das z. B. den größten, kleinsten oder gesuchten Wert enthält) wird automatisch in IR0 gespeichert. Sie können das Indexregister (IR0) in späteren Befehlen als Operand verwenden, um beispielsweise den Inhalt des Wortes mit dem Befehl MOV(021) zu lesen.



In der folgenden Tabelle sind die Bereichsbefehle und ihre Funktionen aufgelistet.

Befehl	Funktion	Beschreibung
SRCH(181)	Sucht nach bestimmten Daten.	Sucht die entsprechenden Daten im spezifizierten Bereich, und gibt die SPS-Speicheradresse des Wortes, das den Wert enthält, in IR0 aus.
MAX(182)	Sucht den größten Wert.	Sucht den größten Wert im spezifizierten Bereich, und gibt die SPS-Speicheradresse des Wortes, das diesen Wert enthält, in IR0 aus.
MIN(183)	Sucht den kleinsten Wert.	Sucht den kleinsten Wert im spezifizierten Bereich, und gibt die SPS-Speicheradresse des Wortes, das diesen Wert enthält, in IR0 aus.
SUM(184)	Berechnet die Summe.	Berechnet die Summe der Daten im spezifizierten Bereich.
FCS(180)	Berechnet die Prüfsumme.	Berechnet die Prüfsumme der Daten im spezifizierten Bereich.

Sie können die Indexregister in FOR-NEXT-Schleifen mit anderen Befehlen kombinieren (z. B. mit Vergleichsbefehlen), um komplizierte Operationen für ganze Wortbereiche durchzuführen.

Tabellenverarbeitung (Datensatztabellen-Befehle)

Die Datensatztabellen-Befehle verarbeiten speziell definierte Datentabellen, die aus Datensätzen gleicher Länge bestehen. Dadurch kann bei der Verarbeitung einfach direkt über die Datensatznummer auf die Datensätze zugegriffen werden.

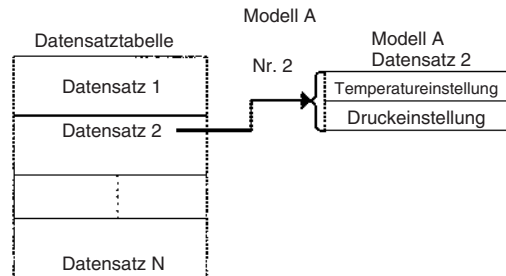
Befehl	Funktion	Beschreibung
DIM(631)	Definiert eine Datensatztable.	Legt die Länge der einzelnen Datensätze und die Anzahl der Datensätze fest.
SETR(635)	Setzt die Datensatzposition.	Schreibt die Position des spezifizierten Datensatzes (die SPS-Speicheradresse, an der der Datensatz beginnt) in das angegebene Indexregister.
GETR(636)	Ruft die Datensatzposition ab.	Gibt die Datensatznummer des Datensatzes, der die SPS-Speicheradresse enthält, in das spezifizierte Indexregister zurück.

Hinweis

Datensatznummern und Wortadressen werden durch die Indexregister in Beziehung gesetzt. Geben Sie in SETR(635) eine Datensatznummer an, um die SPS-Speicheradresse, an der der Datensatz beginnt, in einem Indexregister zu speichern. Um auf ein bestimmtes Wort des Datensatzes zuzugreifen, fügen Sie dem Indexregister den entsprechenden Offsetwert hinzu.

Sie können mit Datensatztabellen-Befehlen und Indexregistern die folgenden Funktionsarten durchführen: Lesen und Schreiben von Datensatzdaten, Suchen nach Datensätzen, Sortieren von Datensatzdaten, Vergleichen von Datensatzdaten und Durchführen von Berechnungen mit Datensatzdaten.

Zu den typischen Anwendungsgebieten für Datensatztabellen zählt die Speicherung der Fertigungsdaten verschiedener Modelle eines Produkts (z. B. die Temperatur- und Druckeinstellungen) in Form von Datensätzen. Dadurch kann einfach durch Ändern der Datensatznummer zwischen den Modellen umgeschaltet werden.



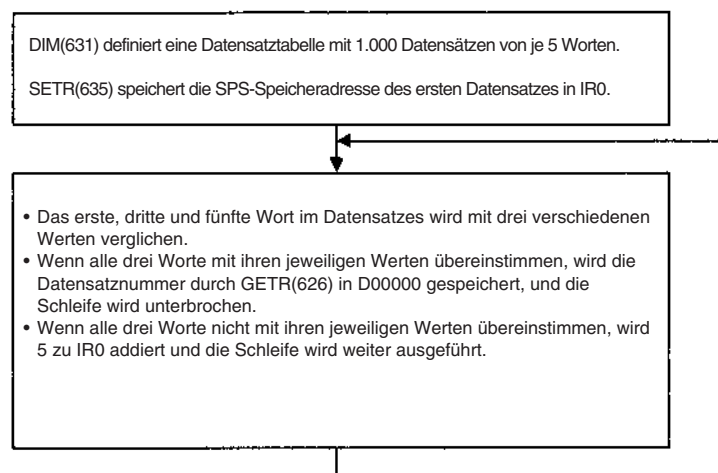
Im Wesentlichen werden Datensatztabellen wie folgt verwendet:

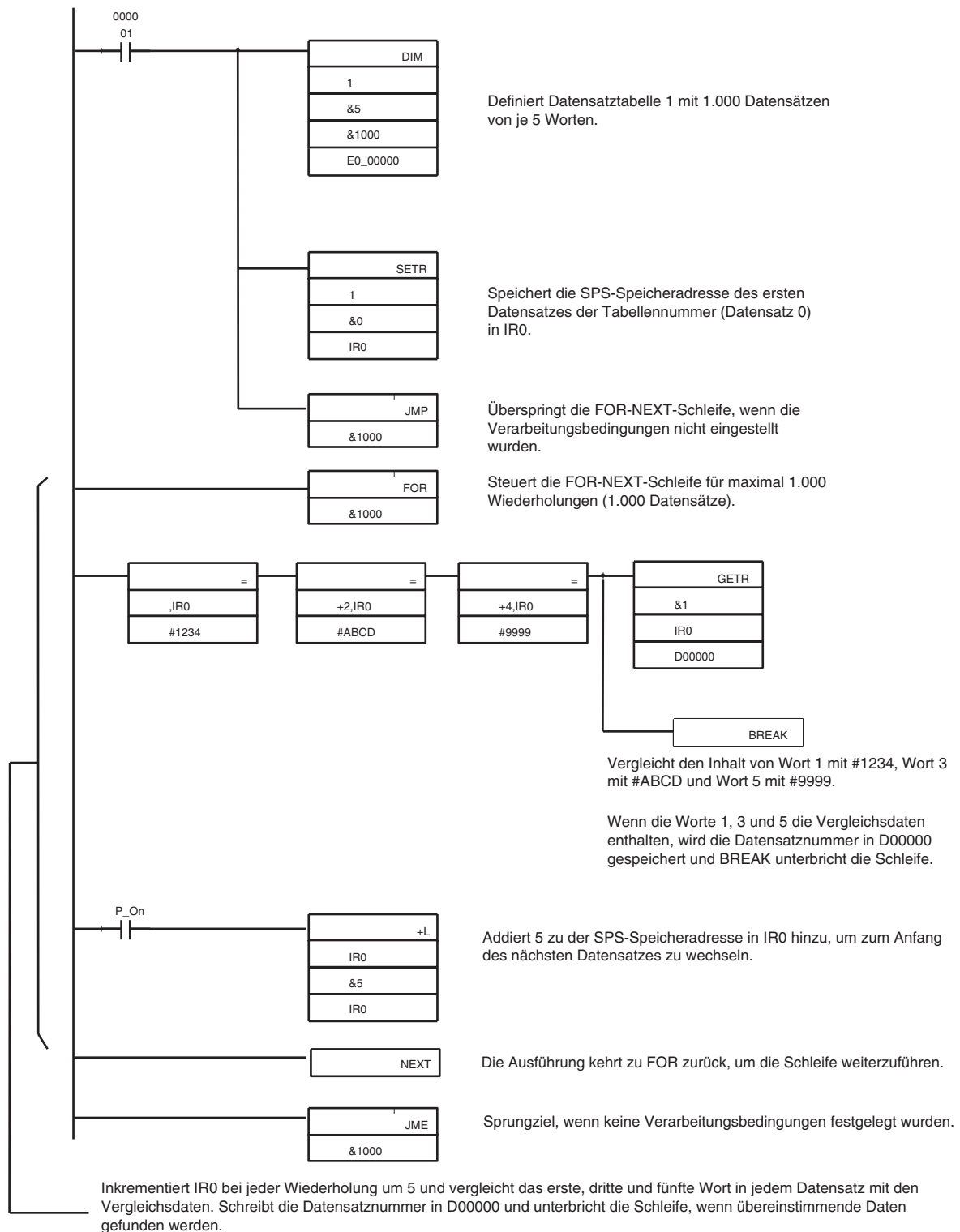
1,2,3...

1. Definieren Sie mit DIM(631) die Struktur der Datensatztabelle, und schreiben Sie mit SETR(635) die SPS-Speicheradresse eines Datensatzes in ein Indexregister.
2. Wenden Sie eine Inkrementierung oder ein Offset auf die SPS-Speicheradresse im Indexregister an, um die Worte im Datensatz zu lesen oder zu vergleichen.
3. Wenden Sie eine Inkrementierung oder ein Offset auf die SPS-Speicheradresse im Indexregister an, um zu einem anderen Datensatz zu wechseln.
4. Wiederholen Sie die Schritte 2 und 3 nach Bedarf.

Beispiel

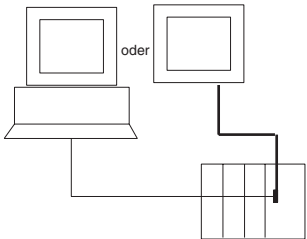
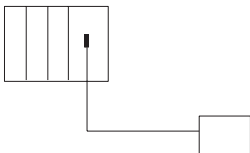
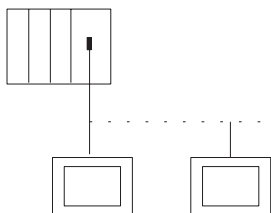
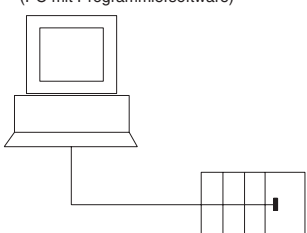
Im folgenden Beispiel werden mit Hilfe der Indexregister und der Datensatztabellen-Befehle drei Werte mit den Worten 1, 3 und 5 jedes Datensatzes verglichen. Bei einer gefundenen Übereinstimmung wird die entsprechende Datensatznummer in D00000 gespeichert.

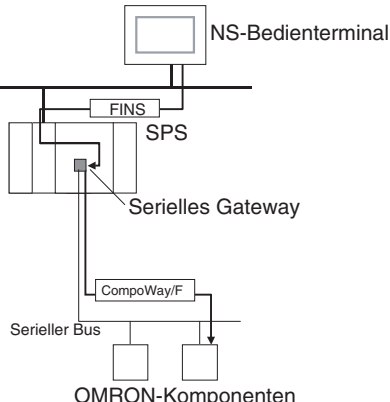
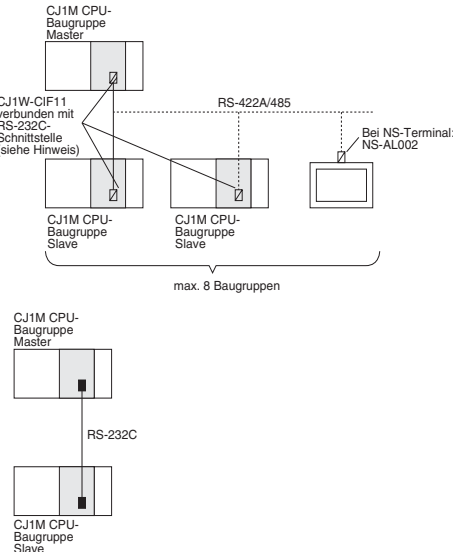




6-3 Serielle Kommunikation

Die CS/CJ-Serie CPU-Baugruppen unterstützen die folgenden seriellen Kommunikationsfunktionen. Host-Link-Kommunikation und protokollfreie Kommunikation werden weiter unten in diesem Abschnitt ausführlich beschrieben.

Protokoll	Verbindungen	Beschreibung	Schnittstellen	
			Peripherie	RS-232C
Host-Link	Host-Computer OMRON Bedienterminal 	1) Verschiedene Steuerbefehle, wie Lesen und Schreiben des E/A-Speichers, Ändern der Betriebsart und zwangsweises Setzen oder Rücksetzen von Bits, können ausgeführt werden, indem der CPU-Baugruppe vom Host-Computer aus Host-Link- oder FINS-Befehle erteilt werden. 2) Außerdem können dem Host-Computer von der CPU-Baugruppe aus FINS-Befehle erteilt werden, um Daten oder Informationen zu senden. Verwenden Sie Host-Link-Kommunikation, um Daten im SPS-System zu überwachen, wie Betriebsstatus, Fehlerinformationen und Qualitätsdaten, oder um Daten zum SPS-System zu senden, z. B. Produktionsplanungsinformationen.	OK	OK
Protokollfrei	Externes Standardgerät 	Für die Kommunikation mit an der RS-232C-Schnittstelle angeschlossenen Standardgeräten ohne festes Befehl/Antwort-Format. Stattdessen werden vom Programm die Befehle TXD(236) und RXD(235) ausgeführt, um Daten von der Übertragungsschnittstelle zu übertragen oder von der Empfangsschnittstelle zu lesen. Die Frame-Header und Endcodes können spezifiziert werden.	Nicht zulässig	OK
NT-Link 1:N oder 1:1	OMRON Bedienterminals 	Daten können mit programmierbaren Bedienterminals ausgetauscht werden, ohne dass in der CPU-Baugruppe ein Kommunikationsprogramm verwendet werden muss.	OK	OK
Peripheriebus	Programmiergeräte (PC mit Programmiersoftware) 	Bietet Hochgeschwindigkeitskommunikation mit Programmiergeräten (mit Ausnahme von Programmierkonsolen). (Dezentrale Programmierung über Modems wird nicht unterstützt.)	OK	OK

Protokoll	Verbindungen	Beschreibung	Schnittstellen	
			Peripherie	RS-232C
Seriell Gateway (Konvertierung in CompoWay/F) (nur CPU-Baugruppen ab Baugruppenversion 3.0)	OMRON-Komponenten (CompoWay/F-kompatible Produkte) 	Konvertiert die empfangenen FINS-Befehle in CompoWay/F-Befehle und gibt diese an den seriellen Bus aus.	OK	OK
Serielle SPS-Links (nur CJ1M)		Bis zu neun CPU-Baugruppen (ein Masters und bis zu acht Slaves) können bis zu zehn Worte pro Baugruppe gemeinsam nutzen. Sie können die RS-232C-Schnittstelle jeder CPU-Baugruppe mit einem RS-422A-Konverter verbinden, um über RS-422A/485 zu kommunizieren, oder zwei CPU-Baugruppen können über eine RS-232C-Verbindung kommunizieren. Die seriellen SPS-Links können in Kombination mit CJ1M CPU-Baugruppen auch über NT-Links (1:N) angeschlossene programmierbare Bedienterminals als Slaves enthalten.	Nicht zulässig	OK

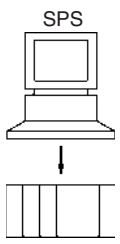
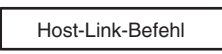
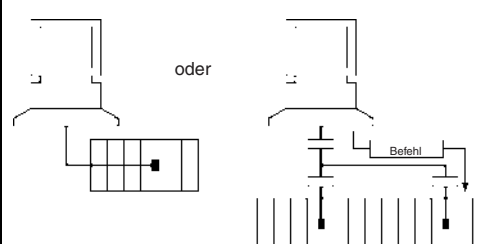
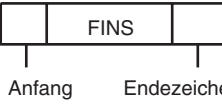
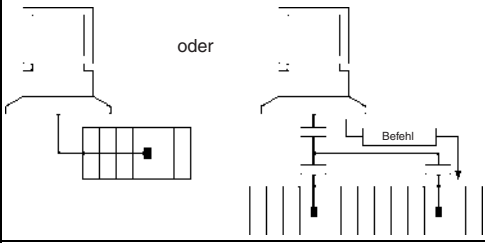
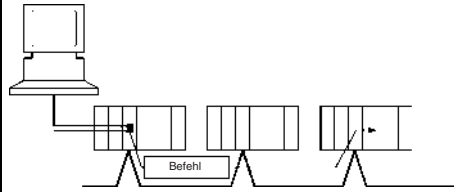
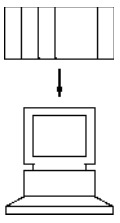
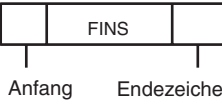
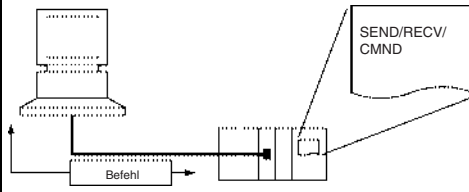
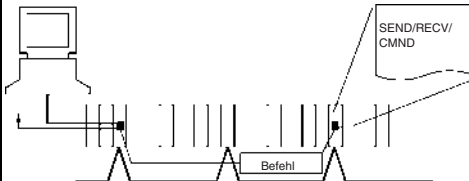
Im Folgenden werden die Host-Link-Kommunikation und die protokollfreie Kommunikation beschrieben.

Hinweis

Der CJ1W-CIF11 ist nicht galvanisch getrennt und die Gesamtkabellänge darf höchstens 50 Meter betragen. Wenn die Gesamtkabellänge länger als 50 Meter ist, müssen Sie den galvanisch getrennten NT-AL001 statt des CJ1W-CIF11 verwenden. Wenn nur der NT-AL001 verwendet wird, kann die Gesamtkabellänge bis zu 500 Meter betragen.

6-3-1 Host-Link-Kommunikation

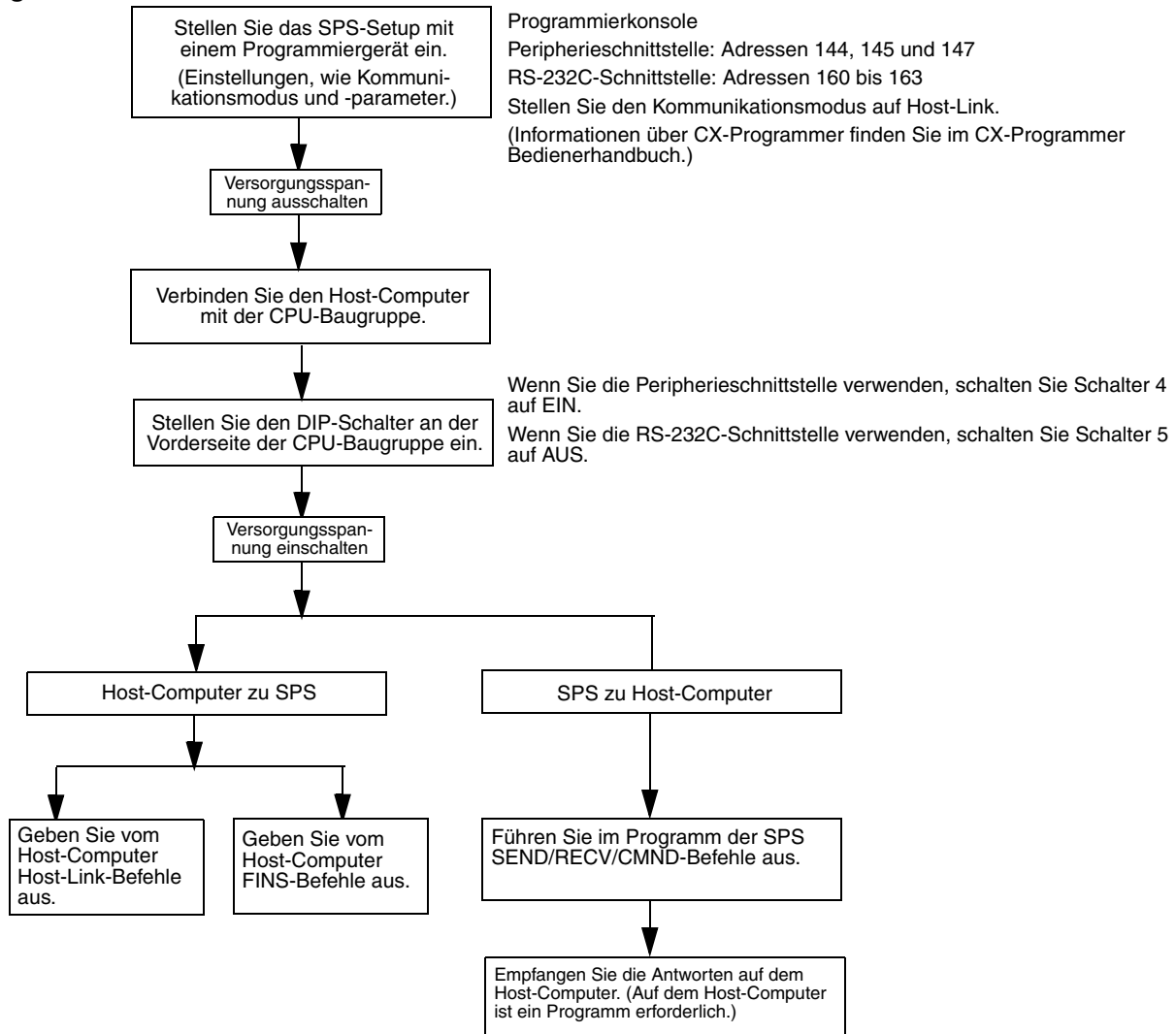
In der folgenden Tabelle sind die auf CS/CJ SPS-Systemen verfügbaren Host-Link-Kommunikationsfunktionen dargestellt. Wählen Sie die für Ihre Anwendung geeignete Methode aus.

Befehlsfluss	Befehlsart	Kommunikationsmethode	Konfiguration
Host-Computer 	Host-Link-Befehl 	Frame im Host-Computer erstellen und Befehl zur SPS ausgeben. Die Antwort von der SPS empfangen. Anwendung: Verwenden Sie diese Methode, wenn Sie hauptsächlich vom Host-Computer zum SPS-System kommunizieren.	Direkte Verbindung mit dem Host-Computer in einem 1:1 oder 1:N-System. 
	FINS-Befehl¹ (mit Host-Link-Anfangs- und -Endezeichen) 	Frame im Host-Computer erstellen und Befehl zur SPS ausgeben. Die Antwort von der SPS empfangen. Anwendung: Verwenden Sie diese Methoden, wenn Sie hauptsächlich vom Host-Computer zu SPS-Systemen im Netzwerk kommunizieren.	Direkte Verbindung mit dem Host-Computer in einem 1:1 oder 1:N-System. 
			Kommunikation vom Host-Computer aus mit anderen SPS-Systemen im Netzwerk. (Konvertierung von Host-Link in Netzwerkprotokoll.) 
SPS Host-Computer 	FINS-Befehl² (mit Host-Link-Anfangs- und -Endezeichen) 	Frame von der CPU-Baugruppe mit den SEND/RECV/CMND-Befehlen ausgeben. Die Antwort vom Host-Computer empfangen. Anwendung: Verwenden Sie diese Methode, wenn Sie hauptsächlich von der SPS zum Host-Computer kommunizieren, um Statusinformationen (z. B. Fehlerinformationen) zu übertragen.	Direkte Verbindung mit dem Host-Computer in einem 1:1-System. 
			Kommunikation mit dem Host-Computer über andere SPS-Systeme im Netzwerk. (Konvertierung von Host-Link in Netzwerkprotokoll.) 

Hinweis 1. Der FINS-Befehl muss zwischen einem Host-Link-Header und -Abschlusscode eingebettet werden, bevor er vom Host-Computer übertragen wird.

2. Der FINS-Befehl wird von der SPS mit einem angehängten Host-Link-Header und -Abschlusscode übertragen. Auf dem Host-Computer muss ein Programm darauf vorbereitet sein, die FINS-Befehle zu analysieren und die richtigen Antworten zurückzugeben.

Vorgehensweise



Host-Link-Befehle

In der folgenden Tabelle sind die Host-Link-Befehle aufgelistet. Weitere Detailinformationen finden Sie im *C-Serie Host-Link-Baugruppen Systemhandbuch (W143)*.

Header-Code	Bezeichnung	Funktion
RR	CIO-Bereich lesen	Liest die Inhalte der spezifizierten Anzahl an Worten im CIO-Bereich ab dem spezifizierten Wort.
RL	LINK-Bereich lesen	Liest die Inhalte der spezifizierten Anzahl an Worten im Link-Bereich ab dem spezifizierten Wort.
RH	H-Bereich lesen	Liest die Inhalte der spezifizierten Anzahl an Worten im Haftmerkerbereich ab dem spezifizierten Wort.
RC	Istwert lesen	Liest die Inhalte der spezifizierten Anzahl von Zeitgeber-/Zähler-Istwerten ab dem spezifizierten Zeitgeber/Zähler.
RG	T/C-Status lesen	Liest den Status der Ablaufmerker der spezifizierten Anzahl von Zeitgebern/Zählern ab dem spezifizierten Zeitgeber/Zähler.
RD	D-Bereich lesen	Liest die Inhalte der spezifizierten Anzahl an Worten im DM-Bereich ab dem spezifizierten Wort.
RJ	A-Bereich lesen	Liest die Inhalte der spezifizierten Anzahl an Worten im Zusatzsystembereich ab dem spezifizierten Wort.

Header-Code	Bezeichnung	Funktion
RE	E-Bereich lesen	Liest die Inhalte der spezifizierten Anzahl an Worten im EM-Bereich ab dem spezifizierten Wort.
WR	CIO-Bereich schreiben	Schreibt die spezifizierten Daten (nur ganze Worte) ab dem spezifizierten Wort in den CIO-Bereich.
WL	LINK-Bereich schreiben	Schreibt die spezifizierten Daten (nur ganze Worte) ab dem spezifizierten Wort in den Link-Bereich.
WH	H-Bereich schreiben	Schreibt die spezifizierten Daten (nur ganze Worte) ab dem spezifizierten Wort in den Haftmerkerbereich.
WC	Istwert schreiben	Schreibt die Istwerte der spezifizierten Anzahl von Zeitgebern/Zählern ab dem spezifizierten Zeitgeber/Zähler.
WD	D-Bereich schreiben	Schreibt die spezifizierten Daten (nur ganze Worte) ab dem spezifizierten Wort in den DM-Bereich.
WJ	A-Bereich schreiben	Schreibt die spezifizierten Daten (nur ganze Worte) ab dem spezifizierten Wort in den Zusatz-Systembereich.
WE	E-Bereich schreiben	Schreibt die spezifizierten Daten (nur ganze Worte) ab dem spezifizierten Wort in den EM-Bereich.
R#	Sollwert lesen 1	Liest die vierstellige BCD-Konstante oder Wortadresse des Sollwerts des spezifizierten Zeitgeber-/Zählerbefehls.
R\$	Sollwert lesen 2	Sucht ab der spezifizierten Programmadresse nach dem spezifizierten Zeitgeber-/Zählerbefehl, und ließt die vierstellige Konstante oder Wortadresse des Sollwerts.
R%	Sollwert lesen 3	Sucht ab der spezifizierten Programmadresse nach dem spezifizierten Zeitgeber-/Zählerbefehl, und ließt die vierstellige BCD-Konstante oder Wortadresse des Sollwerts.
W#	Sollwert ändern 1	Ändert die vierstellige BCD-Konstante oder Wortadresse des Sollwerts des spezifizierten Zeitgeber-/Zählerbefehls.
W\$	Sollwert ändern 2	Sucht ab der spezifizierten Programmadresse nach dem spezifizierten Zeitgeber-/Zählerbefehl, und ändert die vierstellige Konstante oder Wortadresse des Sollwerts.
W%	Sollwert ändern 3	Sucht ab der spezifizierten Programmadresse nach dem spezifizierten Zeitgeber-/Zählerbefehl, und ändert die vierstellige BCD-Konstante oder Wortadresse des Sollwerts.
MS	Status lesen	Liest den Betriebsstatus der CPU-Baugruppe (Betriebsart, Status für zwangsweises Setzen/Rücksetzen, Status schwerwiegender Fehler).
SC	Status ändern	Ändert die Betriebsart der CPU-Baugruppe.
MF	Fehler lesen	Liest und löscht (nicht schwerwiegende und schwerwiegende) Fehler in der CPU-Baugruppe.
KS	Zwangsweises Setzen	Setzt das spezifizierte Bit zwangsweise.
KR	Zwangsweises Rücksetzen	Setzt das spezifizierte Bit zwangsweise zurück.
FK	Mehrfaches zwangsweises Setzen/Rücksetzen	Zwangsweises Setzen bzw. Rücksetzen oder Löschen des Zwangssetzungsstatus der spezifizierten Bits.
KC	Zwangsweises Setzen/Rücksetzen aufheben	Löscht den Zwangssetzungsstatus aller Bits, die mit zwangsweisem Setzen bzw. Rücksetzen gesetzt wurden.
MM	SPS-Typ lesen	Liest die Produktbezeichnung des SPS-Systems.
TS	Test	Gibt einen vom Host-Computer übertragenen Datenblock unverändert zurück.
RP	Programm lesen	Liest den Inhalt des Anwenderprogrammbereichs der CPU-Baugruppe in Maschinensprache (Objektcode).
WP	Programm schreiben	Schreibt das vom Host-Computer übertragene Maschinenspracheprogramm (Objektcode) in den Anwenderprogrammbereich der CPU-Baugruppe.
MI	E/A-Tabelle generieren	Erstellt aus der E/A-Tabelle eine registrierte E/A-Tabelle.
QQMR	Zusammengefasste Lesedaten	Registriert die gewünschten Bits und Worte in einer Tabelle.
QQIR	Zusammengefasste Leseaufforderung	Liest die registrierten Worte und Bits aus dem E/A-Speicher.
XZ	Abbruch (nur Befehl)	Bricht den aktuell verarbeiteten Host-Link-Befehl ab.

Header-Code	Bezeichnung	Funktion
**	INITIALIZE (nur Befehl)	Initialisiert den Übertragungssteuerungsvorgang aller mit dem Host-Computer verbundenen SPS-Systeme.
IC	Undefinierter Befehl (nur Antwort)	Diese Antwort wird zurückgegeben, wenn der Header-Code eines Befehls unbekannt ist.

FINS-Befehle

In der folgenden Tabelle sind die FINS-Befehle aufgelistet. Ausführliche Informationen finden Sie im *FINS-Befehle Referenzhandbuch (W227)*.

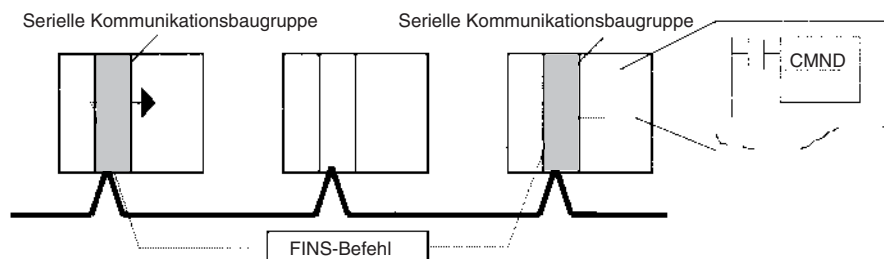
Typ	Befehls-code		Bezeichnung	Funktion
Zugriff auf den E/A-Speicherbereich	01	01	MEMORY AREA READ	Liest aufeinander folgende Daten aus dem E/A-Speicherbereich.
	01	02	MEMORY AREA WRITE	Schreibt aufeinander folgende Daten in den E/A-Speicherbereich.
	01	03	MEMORY AREA FILL	Füllt den spezifizierten Bereich des E/A-Speichers mit denselben Daten.
	01	04	MULTIPLE MEMORY AREA READ	Liest nicht aufeinander folgende Daten aus dem E/A-Speicherbereich.
	01	05	MEMORY AREA TRANSFER	Kopiert und überträgt aufeinander folgende Daten von einem Teil des E/A-Speicherbereichs in einen anderen.
Zugriff auf den Parameterbereich	02	01	PARAMETER AREA READ	Liest aufeinander folgende Daten aus dem Parameterbereich.
	02	02	PARAMETER AREA WRITE	Schreibt aufeinander folgende Daten in den Parameterbereich.
	02	03	PARAMETER AREA FILL	Füllt den spezifizierten Bereich des Parameterbereichs mit denselben Daten.
Zugriff auf den Programmbereich	03	06	PROGRAM AREA READ	Liest Daten aus dem Anwenderprogrammbereich.
	03	07	PROGRAM AREA WRITE	Schreibt Daten in den Anwenderprogrammbereich.
	03	08	PROGRAM AREA CLEAR	Löscht den spezifizierten Bereich des Anwenderprogrammbereichs.
Ausführungssteuerung	04	01	RUN	Schaltet die CPU-Baugruppe in die RUN-, MONITOR- oder DEBUG-Betriebsart.
	04	02	STOP	Schaltet die CPU-Baugruppe in die PROGRAM-Betriebsart.
Lesen der Konfiguration	05	01	CONTROLLER DATA READ	Liest die Informationen der CPU-Baugruppe.
	05	02	CONNECTION DATA READ	Liest die Produktbezeichnungen der spezifizierten Baugruppen.
Lesen des Status	06	01	CONTROLLER STATUS READ	Liest die Statusinformationen der CPU-Baugruppe.
	06	20	CYCLE TIME READ	Liest die mittlere, maximale und minimale Zykluszeit.
Zugriff auf die Uhr	07	01	CLOCK READ	Liest die Uhr.
	07	02	CLOCK WRITE	Stellt die Uhr.
Zugriff auf Meldungen	09	20	MESSAGE READ/CLEAR	Liest bzw. löscht Meldungen und FAL(S)-Meldungen.
Zugriffsrechte	0C	01	ACCESS RIGHT ACQUIRE	Nimmt das Zugriffsrecht an, wenn es nicht bei einem anderen Gerät liegt.
	0C	02	ACCESS RIGHT FORCED ACQUIRE	Nimmt das Zugriffsrecht an, auch wenn es gerade bei einem anderen Gerät liegt.
	0C	03	ACCESS RIGHT RELEASE	Gibt das Zugriffsrecht frei, unabhängig davon, bei welchem Gerät es gerade liegt.
Zugriff auf Fehler	21	01	ERROR CLEAR	Löscht die Fehler und Fehlermeldungen.
	21	02	ERROR LOG READ	Liest das Fehlerprotokoll.
	21	03	ERROR LOG CLEAR	Setzt den Fehlerprotokollzeiger zurück auf null.

Typ	Befehls-code		Bezeichnung	Funktion
Dateispeicher	22	01	FILE NAME READ	Liest die Dateiinformationen des Dateispeichers.
	22	02	SINGLE FILE READ	Liest die spezifizierte Datenmenge ab dem spezifizierten Punkt aus einer Datei.
	22	03	SINGLE FILE WRITE	Schreibt die spezifizierte Datenmenge ab dem spezifizierten Punkt in eine Datei.
	22	04	FILE MEMORY FORMAT	Formatiert den Dateispeicher.
	22	05	FILE DELETE	Löscht die spezifizierte Dateien aus dem Dateispeicher.
	22	07	FILE COPY	Kopiert eine Datei innerhalb des Dateispeichers oder zwischen zwei Dateispeichergeräten eines Systems.
	22	08	FILE NAME CHANGE	Ändert einen Dateinamen.
	22	0A	I/O MEMORY AREA FILE TRANSFER	Überträgt oder vergleicht Daten zwischen dem E/A-Speicherbereich und dem Dateispeicher.
	22	0B	PARAMETER AREA-FILE TRANSFER	Überträgt oder vergleicht Daten zwischen dem Parameterbereich und dem Dateispeicher.
	22	0C	PROGRAM AREA-FILE TRANSFER	Überträgt oder vergleicht Daten zwischen dem Programmbereich und dem Dateispeicher.
	22	15	CREATE/DELETE DIRECTORY	Erstellt oder löscht ein Verzeichnis.
Zwangssetzungsstatus	23	01	FORCED SET/RESET	Zwangswises Setzen bzw. Rücksetzen oder Löschen des Zwangssetzungsstatus der spezifizierten Bits.
	23	02	FORCED SET/RESET CANCEL	Löscht den Zwangssetzungsstatus aller Bits, die mit zwangsweisem Setzen bzw. Rücksetzen gesetzt wurden.

Meldungskommunikationsfunktionen

Die in der Tabelle oben aufgelisteten FINS-Befehle können auch von anderen SPS-Systemen über das Netzwerk zur CPU-Baugruppe gesendet werden. Beachten Sie die folgenden Punkte, wenn Sie FINS-Befehle über das Netzwerk senden.

- Um FINS-Befehle senden zu können, müssen in der lokalen und in der Ziel-SPS CPU-Bus-Baugruppen (wie Controller-Link-Baugruppen oder Ethernet-Baugruppen) installiert sein.
- FINS-Befehle werden vom Programm der CPU-Baugruppe mit CMND(490) erteilt.
- FINS-Befehle können für die CPU-Baugruppen der CS/CJ-Serie ab Version 2.0 über bis zu acht, bei anderen CPU-Baugruppen über bis zu drei Netzwerke hinweg gesendet werden. Die Netzwerke müssen nicht vom selben Typ sein.



Weitere Detailinformationen zu den Meldungskommunikationsfunktionen finden Sie im Bedienerhandbuch der CPU-Bus-Baugruppen.

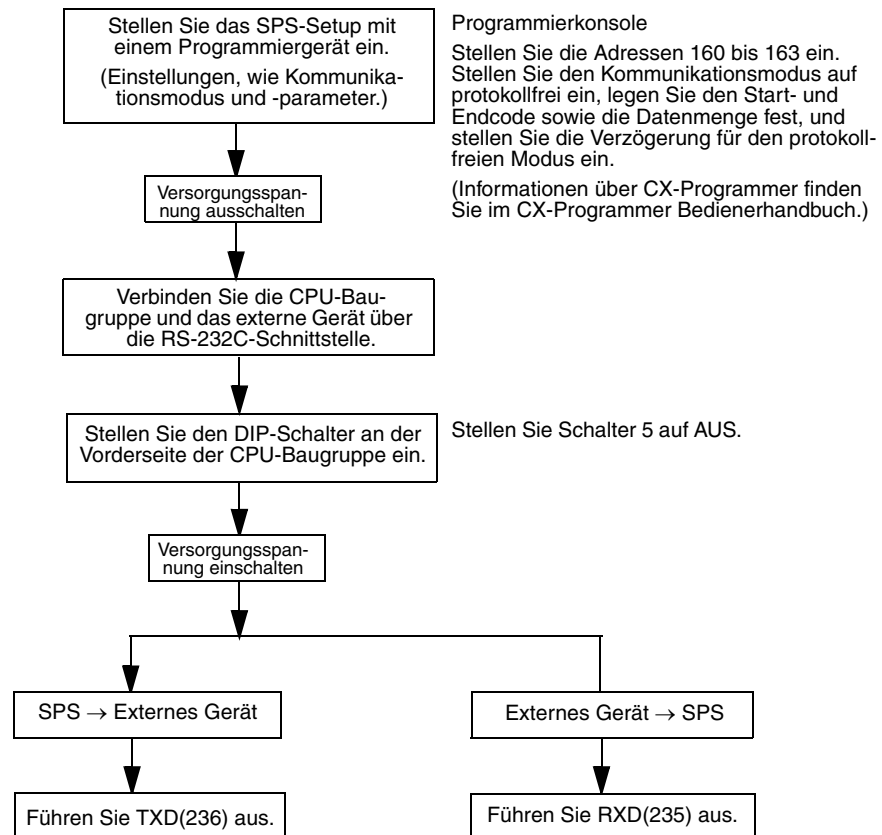
6-3-2 Protokollfreie Kommunikation

In der folgenden Tabelle sind die in CS/CJ SPS-Systemen verfügbaren Funktionen zur protokollfreien Kommunikation dargestellt.

Übertragungsrichtung	Verfahren	Maximale Datenmenge	Frame-Format		Weitere Funktionen
			Startcode	Endcode	
Datenübertragung (SPS → Externes Gerät)	Ausführung von TXD(236) im Programm*	256 Byte	Ja: 00 bis FF Nein: Kein	Ja: 00 bis FF oder CR+LF Nein: Kein	Sendeverzögerungszeit (Verzögerung zwischen Ausführung von TXD und Senden der Daten über die spezifizierte Schnittstelle): 0 bis 99.990 ms (in 10-ms-Schritten einstellbar)
Datenempfang (Externes Gerät → SPS)	Ausführung von RXD(235) im Programm	256 Byte			---

Hinweis Eine Übertragungsverzögerung oder „Verzögerung im protokollfreien Modus“ kann im SPS-Setup spezifiziert werden (Adresse 162). Mit dieser Einstellung können Sie die Übertragung der Daten über die spezifizierte angegebenen Schnittstelle nach der Ausführung von TXD(236) um bis zu 30 Sekunden verzögern.

Vorgehensweise



Meldungsframe-Formate

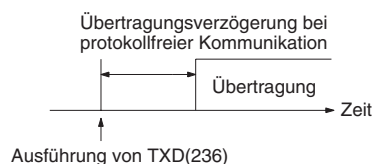
Zur Übertragung durch TXD(236) können Daten zwischen einem Start- und Endcode eingeschlossen und mit RXD(235) empfangen werden. Bei der Übertragung mit TXD(236) werden nur die Daten aus dem E/A-Speicher übertragen, und beim Empfang mit RXD(235) werden nur die Daten an sich im E/A-Speicher gespeichert. Im protokollfreien Modus können höchstens 256 Byte (einschließlich Start- und Endcode) übertragen werden.

In der folgenden Tabelle sind die Meldungsformate aufgeführt, die für Übertragung und Empfang im protokollfreien Modus festgelegt werden können. Das Format wird durch die Einstellungen für den Startcode (ST) und den Endcode (ED) im SPS-Setup bestimmt.

Startcode-Einstellung	Endcode-Einstellung		
	Nein	Ja	CR+LF
Nein	Daten (Daten: max. 256 Byte)	Daten+ED (Daten: max. 255 Byte)	Daten+CR+LF (Daten: max. 254 Byte)
Ja	ST+Daten (Daten: max. 255 Byte)	ST+Daten+ED (Daten: max. 254 Byte)	ST+Daten+CR+LF (Daten: max. 253 Byte)

- Wenn Sie mehr als einen Startcode verwenden, ist der erste Startcode wirksam.
- Wenn Sie mehr als einen Endcode verwenden, ist der erste Endcode wirksam.

- Hinweis**
1. Wenn die übertragenen Daten den Endcode enthalten, bricht die Übertragung an dieser Stelle ab. Stellen Sie in diesem Fall den Endcode auf CR+LF ein.
 2. Im SPS-System steht eine Einstellung zur Verfügung (Adresse 162: Verzögerung im protokollfreien Modus), mit der Sie eine Verzögerung zwischen der Ausführung von TXD(236) und der Datenübertragung festlegen können.



Weitere Detailinformationen zu den Befehlen TXD(236) und RXD(235) finden Sie im *CJ-Serie Speicherprogrammierbare Steuerungen Befehle Referenzhandbuch (W340)*.

6-3-3 NT-Link (1:N-Modus)

Bei der CS/CJ-Serie ist die Kommunikation mit NS/NT-Terminals über 1:N-NT-Link möglich.

- Hinweis** Eine Kommunikation über das 1:1-NT-Link-Protokoll ist nicht möglich.

Neben dem bisherigen Standard-NT-Link sind mit dem Systemmenü des NT/NS-Terminals und den folgenden Einstellungen im SPS-Setup zusätzlich Hochgeschwindigkeits-NT-Links möglich (nicht unterstützt von CS-Serie CS1 CPU-Baugruppen vor EV1). Hochgeschwindigkeits-NT-Links sind jedoch nur mit den NT-Terminals NT31(C)-V2 oder NT631(C)-V2 möglich.

SPS-Setup

Kommunikations-schnittstelle	Einstellungs-adresse der Pro-grammierkonsole	Bezeichnung	Inhalt der Einstellungen	Vorgabewerte	Weitere Bedingungen
Peripherieschnitt-stelle	144 Bits: 8 bis 11	Serieller Kommu-nikationsmodus	02 Hex: NT-Link (1:N-Modus)	00 Hex: Host-Link	Stellen Sie Schal-ter 4 des DIP-Schalterblocks der CPU-Baugruppe auf EIN.
	145 Bits: 0 bis 7	Baudrate	00 bis 09 Hex: Standard-NT-Link 0A Hex: Hochge-schwindigkeits-NT-Link (siehe Hinweis 1)	00 Hex: Standard-NT-Link	
	150 Bits: 0 bis 3	Höchste Geräte-nummer im NT-Link-Modus	0 bis 7 Hex	0 Hex (höchste Gerätenummer: 0)	
RS-232C-Schnittstelle	160 Bits: 8 bis 11	Serieller Kommu-nikationsmodus	02 Hex: NT-Link (1:N-Modus)	00 Hex: Host-Link	Stellen Sie Schalter 5 des DIP-Schalter-blocks der CPU-Baugruppe auf AUS.
	161 Bits: 0 bis 7	Baudrate	00 bis 09 Hex: Standard-NT-Link 0A Hex: Hochge-schwindigkeits-NT-Link (siehe Hinweis 1)	00 Hex: Standard-NT-Link	
	166 Bits: 0 bis 3	Höchste Geräte-nummer im NT-Link-Modus	0 bis 7 Hex	0 Hex (höchste Gerätenummer: 0)	

Hinweis Wenn Sie Einstellungen mit dem CX-Programmer vornehmen, stellen Sie die Baudrate auf 115.200 Bit/s.

Systemmenü des NT/NS-Terminals

Stellen Sie das NT/NS-Terminal wie folgt ein:

- 1,2,3...**
1. Wählen Sie im Systemmenü des Bedienterminals im Menü „Memory Switch“ unter „Comm. A Method“ oder „Comm. B Method“ die Option „NT Link (1:N)“ aus.
 2. Stellen Sie die Kommunikationsgeschwindigkeit mit der SET-Touch-Taste auf „High Speed“.

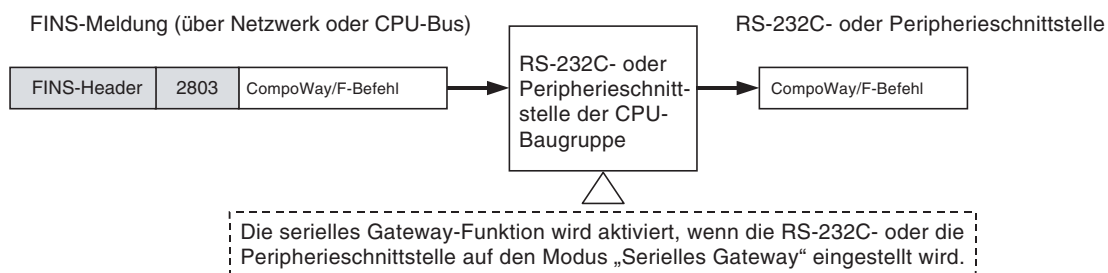
6-3-4 Die Serial Gateway-Funktion der CPU-Baugruppe

Seriellles Gateway – Übersicht

Empfangene FINS-Nachrichten (Befehle) werden automatisch in das angegebene Protokoll konvertiert und an den seriellen Bus ausgegeben. Die Antworten werden ebenfalls automatisch konvertiert. Bei Verwendung der RS-232C- oder der Peripherieschnittstelle der CPU-Baugruppe können FINS-Befehle in das folgende Protokoll konvertiert werden:

- CompoWay/F

Seriellles Gateway wird aktiviert, wenn der serielle Kommunikationsmodus auf „Seriellles Gateway“ eingestellt wird.



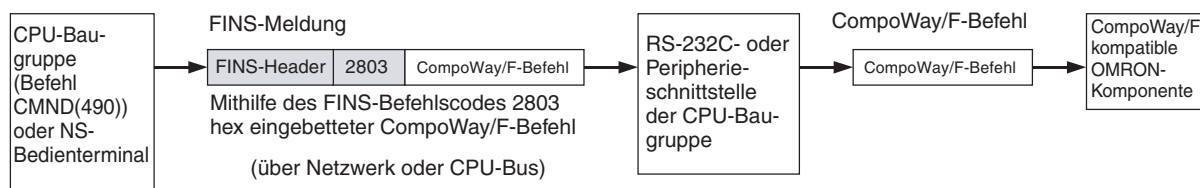
Seriell Gateway – Spezifikationen

Angabe	Details
Konvertierungsquelle	FINS-Befehl (empfangen über das FINS-Netzwerk, Host Link FINS, Peripheriebus, NT-Link oder CPU-Bus)
Konvertierungsfunktion	Empfangene FINS-Befehle werden entsprechend den folgenden Werten konvertiert und dann über die serielle Schnittstelle (Peripherie- oder RS-232C-Schnittstelle) der CPU-Baugruppe ausgegeben. 2803 hex: Entfernt den FINS-Header und konvertiert den Befehl in einen CompoWay/F-Befehl
Nach Konvertierung	CompoWay/F-Befehle
Serielle Kommunikation	1:N-Halbduplex
Maximale Anzahl angeschlossener Baugruppen	31 Slaves
Unterstützte serielle Kommunikationsmodi	Seriell Gateway
Antwortzeit-Überwachung	Die Überwachung der Antwortzeit erfolgt von dem Zeitpunkt, an dem eine mithilfe des seriellen Gateway in das CompoWay/F-Protokoll konvertierte Meldung ausgegeben wurde, bis zum Eintreffen einer Antwort. Die Antwortzeit-Überwachung ist sowohl im seriellen Gateway- als auch im Protokollmakro-Modus. Standardeinstellung: 5 s; Einstellbereich: 0,1 bis 25,5 s Hinweis Tritt eine Zeitüberschreitung auf, wird an die Quelle des FINS-Befehls der FINS-End-Code zurückgegeben (0205 hex: Antwort-Zeitüberschreitung).
Sendeverzögerung	Keine

Konvertierung von FINS nach CompoWay/F

Seriell über CompoWay/F an die Peripherie- oder RS-232C-Schnittstelle der CPU-Baugruppe angeschlossene Geräte können von der SPS oder dem NS-Bedienterminal über in FINS-Meldungen eingebettete CompoWay/F-Befehle angesprochen werden.

- Aufbau der FINS-Meldung: FINS-Header + FINS-Befehlscode 2803 hex + CompoWay/F-Befehl
- Meldung nach der Konvertierung: CompoWay/F-Befehl



Weitere Informationen zum FINS-Befehlscode 2803 hex finden Sie im *Kommunikationsbefehle Referenzhandbuch (W342)*.

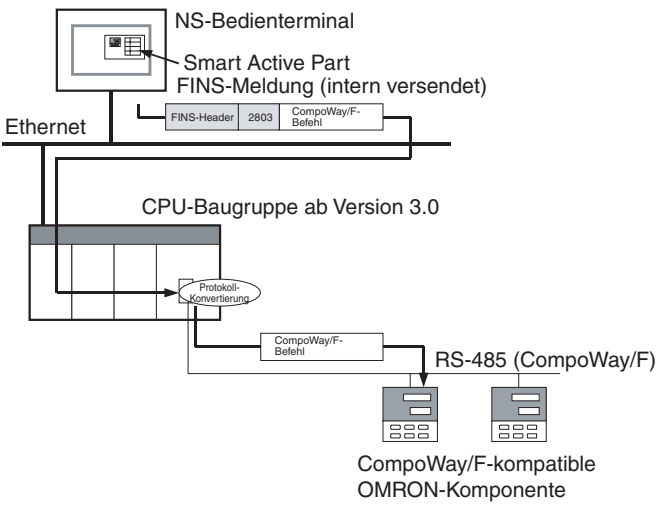
CompoWay/F-Slave-kompatible Komponenten

Komponente		Modellserie
Temperaturregler	Digitale Temperaturregler	E5GN
		E5CN
		E5EN
		E5AN
	Digitale Temperaturregler	E5AR
		E5ER
	Modulare Temperaturregler	E5ZN
	Digitale Regler-Boards	E5ZM
	Digitalregler	ES100X
Zeitgeber/Zähler	Zeitgeber/Zähler	H8GN
Digitalanzeigen	Digitale Prozessanzeigen	K3GN
		K3NX
	Digitale Wägezellenanzeigen	K3HB-V
	Digitale Frequenz-/Drehzahlanzeigen	K3HB-R
	Digitale Inkrementanzeigen	K3HB-P
	Digitale Periodenanzeigen	K3HB-C
	Digitale Temperatur/Prozessanzeigen	K3HB-H
Intelligente Sensoren	ZX-Kommunikations-Schnittstelleneinheit	ZX-SF11
Nockenschaltwerke	---	3F88L-160, 3F88L-162
Sicherheits-Controller	---	F3SX

Systemkonfigurationsmuster

Ausführen von Smart Active Parts unter Verwendung eines NS-Bedienterminals (Senden von internen FINS-Meldungen)

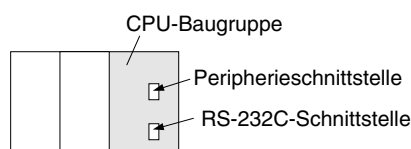
Für diese Operation konvertiert die CPU-Baugruppe die gesendeten FINS-Meldungen in das CompoWay/F-Protokoll.

Zugriff vom NS-Bedienterminal über Ethernet oder einen seriellen NT-Link	Details	Routing-Tabellen für die Behandlung des seriellen Busses als Netzwerk
 Hinweis Ist das NS-Bedienterminal unter Verwendung eines seriellen Kommunikationsmodus (1:N-NT-Link) seriell an die SPS angeschlossen und sendet das NS-Bedienterminal mittels Smart Active Parts in NT-Link-Befehl eingekapselte FINS-Befehle, entfernt die CPU-Baugruppe den NT-Link-Header usw. von dem empfangenen Befehl, konvertiert ihn in einen FINS-Befehl und sendet ihn über ihre serielle Schnittstelle an den seriellen Bus. Dabei nutzt die CPU-Baugruppe das Serial Gateway zum Konvertieren des Befehls in das angegebene Protokoll. Auf diese Weise kann man unter Verwendung der Smart Active Parts eines NS-Bedienterminals über die serielle Schnittstelle der CPU-Baugruppe auf angeschlossene Geräte zugreifen.	Von einem an das Netzwerk angeschlossenen NS-Bedienterminal kann im CompoWay/F-Protokoll auf die an den seriellen Bus angeschlossenen Geräte zugegriffen werden, indem ein Smart Active Part ausgeführt wird, das automatisch einen internen FINS-Befehl sendet.	Optional

Hinweis

1. Der FINS-Header enthält die folgenden Informationen:

- Netzwerkadresse des Remote-Ziels (DNA)
 - Bei Verwendung von Routing-Tabellen für die Behandlung des seriellen Busses als Netzwerk: Der seriellen Schnittstelle in den Routingtabellen entsprechende Netzwerkadresse.
 - Ohne Verwendung von Routing-Tabellen für die Behandlung des seriellen Busses als Netzwerk: Netzwerkadresse für die konkrete Remote-SPS.
- Teilnehmeradresse des Remote-Ziels (DA1)
 - Bei Verwendung von Routing-Tabellen für die Behandlung des seriellen Busses als Netzwerk: 00 hex (interne Kommunikation der lokalen SPS)
 - Ohne Verwendung von Routing-Tabellen für die Behandlung des seriellen Busses als Netzwerk: Teilnehmeradresse für die konkrete Remote-SPS.
- Netzwerkadresse der Remote-Baugruppe (DA2)
Baugruppenadresse der seriellen Schnittstelle



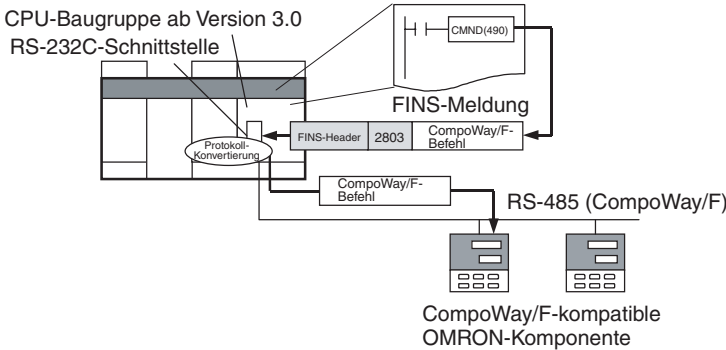
Serielle Schnittstelle der CPU-Baugruppe	Baugruppenadresse der seriellen Schnittstelle
Peripherieschnittstelle	FD hex (253 decimal)
RS-232C-Schnittstelle	FC hex (252 decimal)

2. Der in der gesendeten FINS-Meldung eingebettete CompoWay/F-Befehl setzt sich wie folgt zusammen:
Teilnehmernummer + Subadresse + SID + Befehltext (ASCII)
STX, ETX+BCC sind beim Senden von FINS-Meldungen nicht erforderlich, sie werden bei serieller Kommunikation automatisch hinzugefügt.

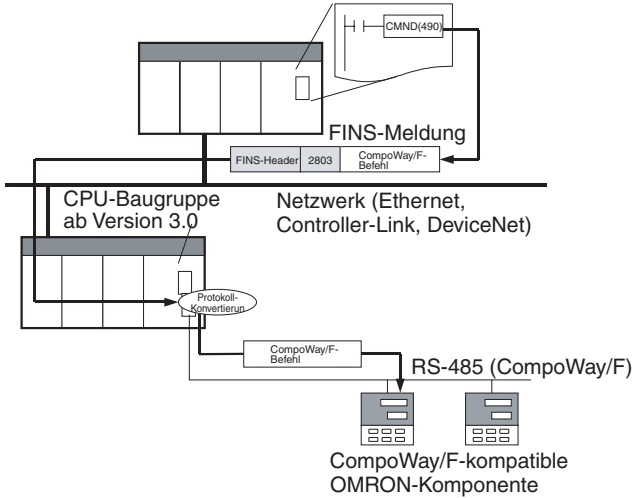
Senden von FINS-Meldungen unter Verwendung des Befehls CMND(490) im SPS-Programm der CPU-Baugruppe

Für diese Operation konvertiert die CPU-Baugruppe die gesendeten FINS-Meldungen in das CompoWay/F-Protokoll.

Zugriff von der CPU-Baugruppe (derselben SPS)

Serielle Konvertierung	Details	Routing-Tabellen für die Behandlung des seriellen Busses als Netzwerk
	An die RS-232C-Schnittstelle der CPU-Baugruppe angeschlossene Geräte können von der CPU-Baugruppe dieser SPS unter Verwendung des CompoWay/F-Protokolls angesprochen werden.	Optional

Zugriff von der CPU-Baugruppe (SPS im Netzwerk)

Konvertierung der FINS-Meldung in einen CompoWay/F-Befehl	Details	Routing-Tabellen für die Behandlung des seriellen Busses als Netzwerk
	An die RS-232C-Schnittstelle der CPU-Baugruppe angeschlossene Geräte können von der CPU-Baugruppe einer im Netzwerk eingebundenen SPS unter Verwendung des CompoWay/F-Protokolls angesprochen werden.	Optional

Kommunikations-Frames

Befehls-Frame

Frame vor der Konvertierung

FINS-Header				FINS-Befehl		CompoWay/F (siehe Hinweis)				
Netzwerk- adresse des Remote-Ziels (DNA)	Teilnehmer- adresse des Remote- Ziels (DA1)	Adresse der Remote- Baugruppe (DA2)	...	MRC	SRC					
Der seriellen Schnittstelle zugeordnete Adresse oder lokale Netzwerk- adresse	00 Hex oder lokale Netzwerk- teilnehmer- adresse	Bau- gruppen- adresse der seriellen Schnitt- stelle		28	03	Teilnehmernr. ($\times 10^1$) ($\times 10^2$) (2 Bytes ASCII)	Subadresse "00" (ASCII 3030 hex) usw.	SID "0" (ASCII 30 hex)	Befehl (MRC, SRC) (4 Bytes ASCII)	Text (ASCII)

Frame nach der Konvertierung

CompoWay/F							
STX (02 hex)	Teilnehmernr. ($\times 10^1$) ($\times 10^2$) (2 Bytes ASCII)	Subadresse "00" (ASCII 3030 hex) usw.	SID "0" (ASCII 0 hex)	Befehl (MRC, SRC) (4 Bytes ASCII)	Text (ASCII)	ETX (03 hex)	BCC

Hinweis CompoWay/F-Befehle verwenden ASCII als Übertragungscode. Achten Sie daher darauf, dass Sie für den CompoWay/F-Befehl, der in der mittels CMND(490) oder durch einen anderen Befehl abgesetzten FINS-Meldung dem FINS-Befehlscode 2803 hex folgt, den ASCII-Code verwenden.

Beispiel: Haben die Komponenten MRC und SRC des CompoWay/F-Befehls beispielsweise den Wert „01“ und „02“ (die Anführungszeichen weisen hier darauf hin, dass es sich um ASCII-Zeichen handelt), müssen 0, 1 und 2 als Zeichen und nicht als Zahlen behandelt werden. „01“ muss daher als 3031 hex (und nicht als 01 hex), „02“ als 3032 hex (nicht als 02 hex) kodiert werden.

Außerdem benötigt, um den CompoWay/F-Befehl mittels CMND(490) in der Frame-Reihenfolge (ohne leere Bytes zu erzeugen) in den Befehlsspeicherbereich zu schreiben, die SID-Komponente des CompoWay/F-Befehls ein Byte ASCII-Code (30 hex). Die folgenden Komponenten (ab s+3) müssen daher jeweils in einzelnen Bytes untergebracht werden.

Frame vor der Konvertierung

CompoWay/F							
STX (02 hex)	Teilnehmernr. ($\times 10^1$) ($\times 10^2$) (2 Bytes ASCII)	Subadresse "00" (ASCII 3030 hex) usw.	Endcode (2 Bytes ASCII)	Befehl (MRC, SRC) (4 Bytes ASCII)	Antwort (MRES, SRES) (4 Bytes ASCII)	Text (ASCII)	ETX (03 hex)

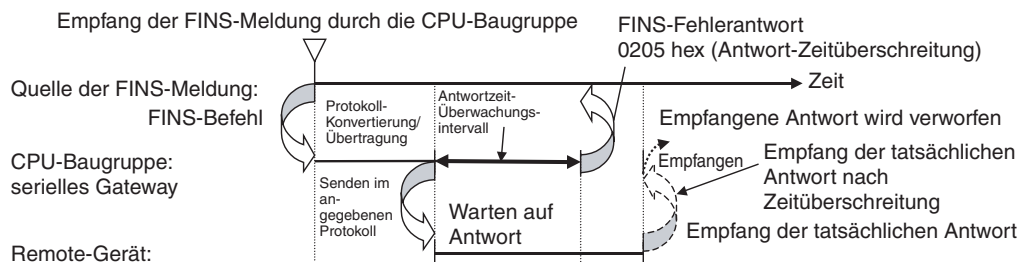
Frame nach der Konvertierung

FINS-Header			FINS-Befehl		FINS-Endcode		CompoWay/F (siehe Hinweis)				
Netzwerk- adresse des Remote- Ziels (DNA)	Teilnehmer- adresse des Remote- Ziels (DA1)	Adresse der Remote- Baugruppe (DA2) usw.	MRC	SRC	MRES	SRES					
Der seriellen Schnittstelle zugeordnete Adresse	00 Hex	Bau- gruppen- adresse der seriellen Schnittstelle	28	03	Benutzer- definiert	Benutzer- definiert	Teilnehmernr. ($\times 10^1$) ($\times 10^2$) (2 Bytes ASCII)	Subadresse "00" (ASCII 3030 hex) usw.	Endcode (2 Bytes ASCII)	Befehl (MRC, SRC) (4 Bytes ASCII)	Antwort (MRES, SRES) (4 Bytes ASCII)

Antwortzeit-Überwachung (Serieller Gateway-Modus)

Die Überwachung der Antwortzeit im seriellen Gateway-Modus erfolgt von dem Zeitpunkt, an dem eine mithilfe des seriellen Gateway in das angegebene Protokoll konvertierte Meldung ausgegeben wurde, bis zum Eintreffen einer Antwort von dem Remote-Gerät. (Die Standardeinstellung beträgt 5 s, der Einstellbereich liegt zwischen 0,1 und 25,5 s.)

Wird an der seriellen Schnittstelle innerhalb der eingestellten Zeit keine Antwort empfangen, wird eine FINS-Fehlerantwort an die Quelle des FINS-Befehls (0205 hex: Antwort-Zeitüberschreitung). Wird nach Auftreten der Zeitüberschreitung dennoch eine Antwort empfangen, wird diese verworfen, es wird also keine FINS-Antwort an die Quelle des FINS-Befehls übermittelt.



6-3-5 Serielle SPS-Links (nur CJ1M CPU-Baugruppen)

Übersicht

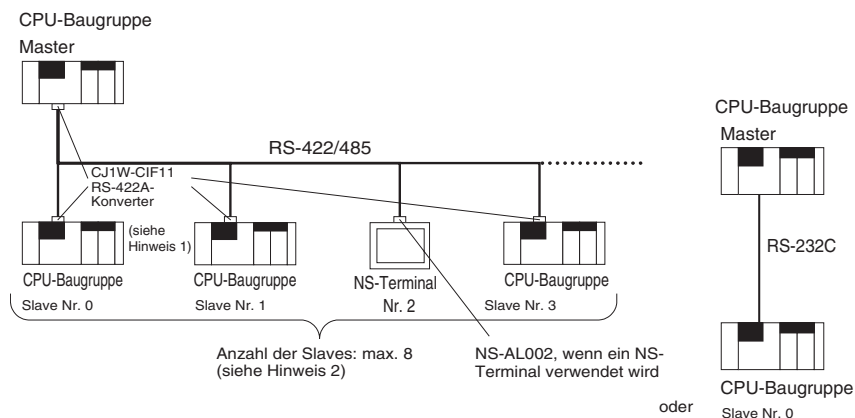
Serielle SPS-Links werden nur von CJ1M CPU-Baugruppen unterstützt. Mit ihrer Hilfe können Daten über die integrierten RS-232C-Schnittstellen zwischen CJ1M CPU-Baugruppen ausgetauscht werden, ohne dass eine spezielle Programmierung erforderlich ist. Worte werden im Speicher aus dem Bereich der Seriell-SPS-Link-Worte (CIO 3100 bis CIO 3199) zugeordnet. Zwischen CPU-Baugruppen können RS-232C-Verbindungen oder, wenn Sie an die RS-232C-Schnittstellen RS-232C-zu-RS-422A/485-Konverter anschließen, RS-422A/485-Verbindungen verwendet werden. Zur Umwandlung zwischen RS-232C und RS-422A/485 können Sie den RS-422A-Konverter CJ1W-CIF11 verwenden.

Im selben Netzwerk kann auch ein NT/NS-Terminal verwendet werden, das auf 1:N-NT-Link-Kommunikation eingestellt ist. Das NT/NS-Terminal als Slave kommuniziert in einer 1:N-NT-Link-Verbindung über das Netzwerk mit der Master-CPU-Baugruppe. Wenn ein NT/NS-Terminal verbunden ist, sind die Adressen im Seriell-SPS-Link-Wortebereich undefiniert, die der Gerätenummer des NT-Terminals entsprechen.

Spezifikationen

Kenndaten	Spezifikationen
Anschlussart	RS-232C- oder RS-422A/485-Verbindung über die RS-232C-Schnittstelle der CPU-Baugruppe.
Zugeordneter Datenbereich	Seriell-SPS-Link-Worte: CIO 3100 bis CIO 3199 (Pro CPU-Baugruppe können höchstens zehn Worte zugeordnet werden.)
Anzahl der Geräte	Höchstens neun Geräte: ein Master und acht Slaves. (Im selben Netzwerk kann ein NT/NS-Terminal mit einer 1:N-NT-Link-Verbindung vorhanden sein, muss dann jedoch als einer der 8 Slaves gezählt werden.)

Systemkonfiguration



- Hinweis**
1. Der CJ1W-CIF11 ist nicht galvanisch getrennt und die Gesamtkabellänge darf höchstens 50 Meter betragen. Wenn die Gesamtkabellänge länger als 50 Meter ist, müssen Sie den galvanisch getrennten NT-AL001 statt des CJ1W-CIF11 verwenden. Wenn nur der NT-AL001 verwendet wird, kann die Gesamtkabellänge bis zu 500 Meter betragen.
 2. Mit dem Master können maximal 8 Geräte, einschließlich NT/NS-Terminal und Slaves, verbunden werden, wenn sich ein für die serielle SPS-Link-Kommunikation konfiguriertes NT/NS-Terminal im selben Netzwerk befindet.

Datenaktualisierungsmethoden

Zur Aktualisierung von Daten können die beiden folgenden Methoden verwendet werden.

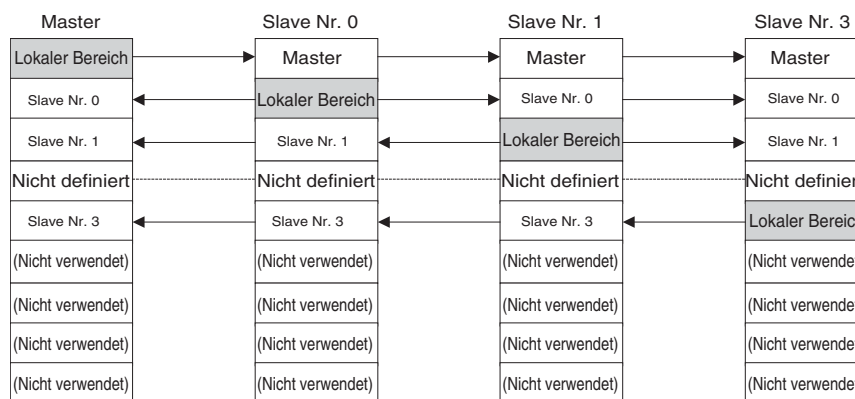
- Gesamter-Link-Verfahren
- Master-Link-Verfahren

Gesamter-Link-Verfahren

Die Daten aller Knoten im seriellen SPS-Link werden im Master und in den Slaves abgebildet. (Die einzigen Ausnahmen sind die der Gerätenummer des angeschlossenen NT/NS-Terminals zugeordnete Adresse und die Adressen der Slaves, die im Netzwerk nicht präsent sind. Diese Datenbereiche sind in allen Knoten undefiniert.)

Beispiel: Gesamter-Link-Verfahren, höchste Gerätenummer: 3.

In der folgenden Abbildung ist Slave Nr. 2 ein NT-Terminal oder im Netzwerk nicht präsent, so dass dem Slave Nr. 2 zugeordnete Bereich in allen Knoten undefiniert ist.

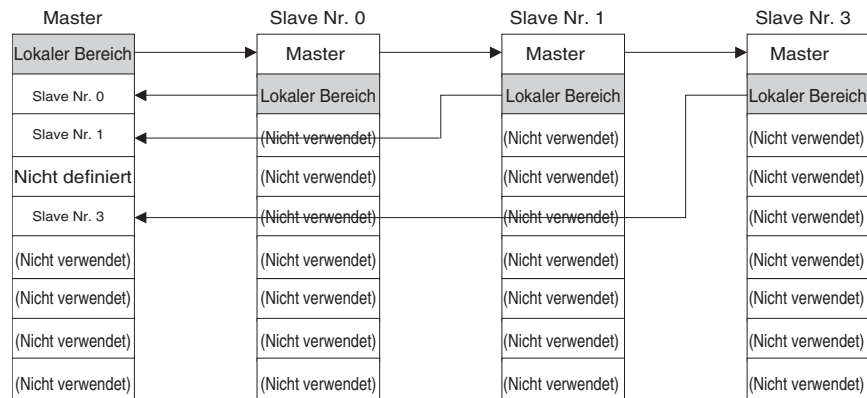


Master-Link-Verfahren

Die Daten aller Slaves im seriellen SPS-Link werden nur im Master abgebildet, und in allen Slaves werden nur die Daten des Masters abgebildet. Der Vorteil des Master-Link-Verfahrens besteht darin, dass die zugeordneten Adressen der Link-Daten des lokalen Slaves in allen Slaves dieselben sind, so dass auf die Link-Daten auf die gleichen Adressen im Slave-Programm zugegriffen werden kann. Nur im Master sind die Bereiche undefiniert, die den Gerätenummern im Netzwerk nicht präsenter Slaves oder NT/NS-Terminals zugeordnet sind.

Beispiel: Master-Link-Verfahren, höchste Gerätenummer: 3.

In der folgenden Abbildung ist Slave Nr. 2 ein NT/NS-Terminal oder ein Gerät, das nicht am Netzwerk teilnimmt, so dass der entsprechende Bereich im Master undefiniert ist.



Zugeordnete Worte

Gesamter-Link-Verfahren

Adresse

CIO 3100

Seriell-
SPS-Link-
Worte

CIO 3199

Link-Worte	1 Wort	2 Worte	3 Worte	bis	10 Worte
Master	CIO 3100	CIO 3100 bis CIO 3101	CIO 3100 bis CIO 3102		CIO 3100 bis CIO 3109
Slave Nr. 0	CIO 3101	CIO 3102 bis CIO 3103	CIO 3103 bis CIO 3105		CIO 3110 bis CIO 3119
Slave Nr. 1	CIO 3102	CIO 3104 bis CIO 3105	CIO 3106 bis CIO 3108		CIO 3120 bis CIO 3129
Slave Nr. 2	CIO 3103	CIO 3106 bis CIO 3107	CIO 3109 bis CIO 3111		CIO 3130 bis CIO 3139
Slave Nr. 3	CIO 3104	CIO 3108 bis CIO 3109	CIO 3112 bis CIO 3114		CIO 3140 bis CIO 3149
Slave Nr. 4	CIO 3105	CIO 3110 bis CIO 3111	CIO 3115 bis CIO 3117		CIO 3150 bis CIO 3159
Slave Nr. 5	CIO 3106	CIO 3112 bis CIO 3113	CIO 3118 bis CIO 3120		CIO 3160 bis CIO 3169
Slave Nr. 6	CIO 3107	CIO 3114 bis CIO 3115	CIO 3121 bis CIO 3123		CIO 3170 bis CIO 3179
Slave Nr. 7	CIO 3108	CIO 3116 bis CIO 3117	CIO 3124 bis CIO 3126		CIO 3180 bis CIO 3189
Nicht verwendet.	CIO 3109 bis CIO 3199	CIO 3118 bis CIO 3199	CIO 3127 bis CIO 3199		CIO 3190 bis CIO 3199

Master-Link-Verfahren

Adresse

CIO 3100

Seriell-
SPS-Link-
Worte

CIO 3199

Link-Worte	1 Wort	2 Worte	3 Worte	bis	10 Worte
Master	CIO 3100	CIO 3100 bis CIO 3101	CIO 3100 bis CIO 3102		CIO 3100 bis CIO 3109
Slave Nr. 0	CIO 3101	CIO 3102 bis CIO 3103	CIO 3103 bis CIO 3105		CIO 3110 bis CIO 3119
Slave Nr. 1	CIO 3101	CIO 3102 bis CIO 3103	CIO 3103 bis CIO 3105		CIO 3110 bis CIO 3119
Slave Nr. 2	CIO 3101	CIO 3102 bis CIO 3103	CIO 3103 bis CIO 3105		CIO 3110 bis CIO 3119
Slave Nr. 3	CIO 3101	CIO 3102 bis CIO 3103	CIO 3103 bis CIO 3105		CIO 3110 bis CIO 3119
Slave Nr. 4	CIO 3101	CIO 3102 bis CIO 3103	CIO 3103 bis CIO 3105		CIO 3110 bis CIO 3119
Slave Nr. 5	CIO 3101	CIO 3102 bis CIO 3103	CIO 3103 bis CIO 3105		CIO 3110 bis CIO 3119
Slave Nr. 6	CIO 3101	CIO 3102 bis CIO 3103	CIO 3103 bis CIO 3105		CIO 3110 bis CIO 3119
Slave Nr. 7	CIO 3101	CIO 3102 bis CIO 3103	CIO 3103 bis CIO 3105		CIO 3110 bis CIO 3119
Nicht verwendet.	CIO 3102 bis CIO 3199	CIO 3104 bis CIO 3199	CIO 3106 bis CIO 3199		CIO 3120 bis CIO 3199

Vorgehensweise

Die seriellen SPS-Links arbeiten entsprechend der folgenden Einstellungen im SPS-Setup.

Einstellungen bei dem Master

- 1,2,3...**
1. Stellen Sie unter dem Karteireiter „Host-Link-Port“ die Betriebsart der RS-232C-Schnittstelle auf „PC-Link (Master)“.
 2. Stellen Sie den PC-Link-Modus auf „Alle“ oder „Master“.
 3. Stellen Sie die Anzahl der Link-Worte ein (bis zu 10 Worte bei jedem Master/Slave).
 4. Stellen Sie die höchste Gerätenummer im seriellen SPS-Link ein (0 bis 7).

Einstellungen bei den Slaves

- 1,2,3...**
1. Stellen Sie unter dem Karteireiter „Host-Link-Port“ die Betriebsart der RS-232C-Schnittstelle auf „PC-Link (Slave)“.
 2. Stellen Sie die „PC-Link-Baugr.-Nr.“ für den Slave im seriellen SPS-Link ein.

SPS-Setup

Einstellungen bei dem Master

Beschreibung		SPS-Adresse		Einstellwert	Standard-einstellung	Aktualisierungszeitpunkt
		Wort	Bit			
Einstellungen für RS-232C-Schnittstelle	Serieller Kommunikationsmodus	160	11 bis 08	8 Hex: Seriell-SPS-Link-Master	0 Hex	In jedem Zyklus
	Schnittstellen-Baudrate	161	07 bis 00	00 bis 09 Hex: Standard 0A Hex: Hochgeschwindigkeit (siehe Hinweis 2)	00 Hex	
	Link-Verfahren	166	15	0: Gesamter Link 1: Master-Link	0	
	Anzahl der Link-Worte		07 bis 04	1 bis A Hex	0 Hex (siehe Hinweis 1)	
	Höchste Gerätenummer		03 bis 00	0 bis 7 Hex	0 Hex	

- Hinweis**
1. Automatische Zuordnung von 10 Worten (A Hex), wenn die Standardeinstellung 0 Hex verwendet wird.
 2. Spezifizieren Sie bei der Einstellung von CX-Programmer 115.200 Bits/s.

Einstellungen bei dem Slave

Beschreibung		SPS-Adresse		Einstellwert	Standard-einstellung	Aktualisierungszeitpunkt
		Wort	Bit			
Einstellungen für RS-232C-Schnittstelle	Serieller Kommunikationsmodus	160	11 bis 08	7 Hex: Seriell-SPS-Link-Slave	0 Hex	In jedem Zyklus
	Schnittstellen-Baudrate	161	07 bis 00	00 bis 09 Hex: Standard 0A Hex: Hochgeschwindigkeit (siehe Hinweis 2)	00 Hex	
	Gerätenummer des Slaves	167	03 bis 00	0 bis 7 Hex	0 Hex	

- Hinweis** Spezifizieren Sie bei der Einstellung von CX-Programmer 115.200 Bits/s.

Zugehörige Merker im Zusatz-Systembereich

Bezeichnung	Adresse	Details	Lesen/ Schreiben	Aktualisierungszeitpunkt
RS-232C-Schnittstelle Kommunikations-Fehlermerker	A39204	EIN, wenn an der RS-232C-Schnittstelle ein Kommunikationsfehler auftritt. 1: Fehler 0: Normal	Lesen	- Wird beim Einschalten der Spannungsversorgung gelöscht. - EIN, wenn an der RS-232C-Schnittstelle ein Kommunikationsfehler auftritt. - Wird beim Neustart der Schnittstelle auf AUS gesetzt. - Im Peripheriebus- und NT-Link-Modus deaktiviert.
Merker: RS-232C-Schnittstelle kommuniziert mit NT-Terminal (siehe Hinweis)	A39300 bis A39307	Wenn die RS-232C-Schnittstelle im NT-Link-Modus verwendet wird, ist das Bit für das kommunizierende Gerät auf EIN gesetzt. Die Bits 00 bis 07 entsprechen den Gerätenummern 0 bis 7. 1: Kommuniziert 0: Keine Kommunikation	Lesen	- Wird beim Einschalten der Spannungsversorgung gelöscht. - Setzt das zur Gerätenummer des NT-Terminals/Slaves gehörige Bit auf EIN, das oder der im NT-Link- oder Seriell-SPS-Link-Modus über die RS-232C-Schnittstelle kommuniziert. - Die Bits 00 bis 07 entsprechen den Gerätenummern 0 bis 7.
RS-232C-Schnittstellen-Neustartbit	A52600	Wird dieses Bit auf EIN gesetzt, wird die RS-232C-Schnittstelle neu gestartet (initialisiert).	Lesen/ Schreiben	- Wird beim Einschalten der Spannungsversorgung gelöscht. - Wird beim Neustart der RS-232C-Schnittstelle auf EIN gesetzt (außer bei Kommunikation im Peripheriebusmodus). Hinweis: Je nach System wird das Bit möglicherweise automatisch auf AUS gesetzt, wenn der Neustart abgeschlossen ist.
RS-232C-Schnittstellen-Fehlermerker	A52800 bis A52807	Wenn an der RS-232C-Schnittstelle ein Fehler auftritt, wird der entsprechende Fehlercode gespeichert. Bit 00: Nicht verwendet. Bit 01: Nicht verwendet. Bit 02: Paritätsfehler Bit 03: Rahmen-Fehler Bit 04: Überlauf-Fehler Bit 05: Zeitüberschreitungsfehler Bit 06: Nicht verwendet. Bit 07: Nicht verwendet.	Lesen/ Schreiben	- Wird beim Einschalten der Spannungsversorgung gelöscht. - Wenn an der RS-232C-Schnittstelle ein Fehler auftritt, wird der entsprechende Fehlercode gespeichert. - Je nach System wird der Merker beim Neustart der RS-232C-Schnittstelle möglicherweise gelöscht. - Ist im Peripheriebusmodus deaktiviert. - Im NT-Link-Modus ist nur Bit 05 (Zeitüberschreitungsfehler) aktiviert. Im seriellen SPS-Link-Modus sind nur die folgenden Bits aktiviert. Fehler im Master: Bit 05: Zeitüberschreitungsfehler Fehlerprüfung im Slave: Bit 05: Zeitüberschreitungsfehler Bit 04: Überlauf-Fehler Bit 03: Rahmen-Fehler
RS-232C-Schnittstellen-Einstellungs-geändert-Merker	A61902	EIN, wenn die Kommunikationseinstellungen der RS-232C-Schnittstelle gerade geändert werden. 1: Während der Änderung 0: Keine Änderung	Lesen/ Schreiben	- Wird beim Einschalten der Spannungsversorgung gelöscht. - EIN, während die Kommunikationseinstellungen für die RS-232C-Schnittstelle geändert werden. - Wird auf EIN gesetzt, wenn der Befehl CHANGE SERIAL PORT SETUP (STUP(237)) ausgeführt wird. - Wird wieder auf AUS gesetzt, wenn die Änderungen an den Einstellungen abgeschlossen sind.

Hinweis

Genauso wie für den vorhandenen 1:N-NT-Link, kann der Status (Kommunikation/keine Kommunikation) der NT/NS-Terminals im seriellen SPS-Link vom Master (CPU-Baugruppe) durch Lesen des RS-232C-Schnittstelle-kommuniziert-mit-Terminal-Merkers (A393, Bits 00 bis 07 für Gerätenummern 0 bis 7) überprüft werden.

6-4 Ändern des Modus der Zeitgeber/Zähler-Istwert-Aktualisierung

6-4-1 Übersicht

Bisher wurde bei CS1 CPU-Baugruppen für die Zeitgeber/Zähler-Istwert-Aktualisierung nur das BCD-Format verwendet. Dadurch mussten alle Zeitgeber- und Zählereinstellungen als BCD-Werte eingegeben werden. Andere CPU-Baugruppen (siehe Hinweise 1 und 2) können die Istwerte der Zeitgeber- und Zählerbefehle im BCD-Modus oder Binärmodus aktualisieren (siehe Hinweis 3).

Im Binärmodus kann statt des bisherigen Einstellbereichs für Zeitgeber und Zähler von 0 bis 9999 der erweiterte Einstellbereich von 0 bis 65535 verwendet werden. Auch mit anderen Befehlen berechnete Binärdaten können als Sollwerte für Zeitgeber und Zähler verwendet werden. Der Zeitgeber/Zähler-Istwert-Aktualisierungsmodus kann auch spezifiziert werden, wenn der Zeitgeber-/Zähler-Sollwert als Adresse spezifiziert ist (indirekte Spezifikation). (Durch die Einstellung auf den BCD- oder Binärmodus wird auch festgelegt, ob die Inhalte der adressierten Worte als BCD- oder Binärwerte interpretiert werden.)

Jedoch unterscheiden sich im BCD- und Binärmodus die Befehlsoperanden. Daher müssen Sie sich mit den Unterschieden zwischen BCD- und Binärmodus vertraut machen, bevor Sie den Istwert-Aktualisierungsmodus für Zeitgeber und Zähler ändern.

Hinweis

1. Diese „anderen als CS1“ CPU-Baugruppen sind die Folgenden:
 - CS1-H CPU-Baugruppen
 - CJ1-H CPU-Baugruppen
 - CJ1M CPU-Baugruppen
 - CS1D CPU-Baugruppen
2. Wenn Sie auf der Programmierkonsole bei einer CS1-H, CJ1-H, CJ1M oder CS1D CPU-Baugruppe, die vor dem 1. Juni 2002 hergestellt wurde und bei der für die Zeitgeber/Zähler-Istwert-Aktualisierung der Binärmodus eingestellt ist, den AWL-Code eines Befehls anzeigen, wird als AWL-Code des Binärbefehls der AWL-Code des BCD-Befehls angezeigt (Beispiel: „TIMX #0000 &16“ wird angezeigt als „TIM #0000 &16“), die Operationen werden jedoch im Binärmodus durchgeführt.
3. Der Istwert-Aktualisierungsmodus kann nur mit CX-Programmer ab Version 3.0 ausgewählt werden. Die Auswahl des Modus wird von CX-Programmer bis einschließlich Version 2.1 und den Programmierkonsolen nicht unterstützt.
4. CX-Programmer bis einschließlich Version 2.1 kann Anwenderprogramme für die CPU-Baugruppe nicht lesen, wenn diese Binärmodus-Befehle enthalten. Anwenderprogramme, in denen BCD-Modus-Befehle verwendet werden, kann die Software jedoch lesen.

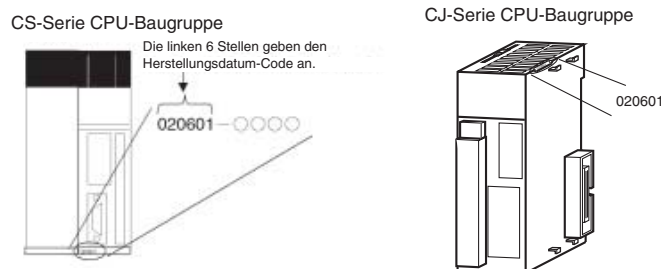
6-4-2 Funktionsspezifikationen

Beschreibung	Details		
Einstellverfahren für den Zeitgeber/Zähler-Istwert-Aktualisierungsmodus	Kann nur mit CX-Programmer Version 3.0 eingestellt werden (nicht unterstützt von CX-Programmer Version 2.1 oder niedriger). Einstellen in den SPS-Eigenschaften von CX-Programmer Version 3.0.		
Unterstützte CPU-Baugruppen	CS1-H/CJ1-H CPU-Baugruppen ab Lot-Nr. 020601 (hergestellt ab 1. Juni 2002, siehe Hinweis 1) sowie CJ1M und CS1D CPU-Baugruppen.		
Modus	BCD-Modus	Binärmodus	
AWL	Wie bei den vorherigen Modellen Beispiel: TIM	Wie im BCD-Modus, jedoch mit angefügtem „X“ Beispiel: TIMX	
Funktionscode	Wie bei den vorherigen Modellen	Neue Codes	
Istwert-/Sollwert-Bereich	#0000 bis #9999	&0 bis &65536	#0000 bis #FFFF
Istwert-Anzeige auf dem Programmiergerät (CX-Programmer Version 3.0 oder Programmierkonsole)	BCD Beispiel: #0100	Dezimal Beispiel: &100	Hexadezimal Beispiel: #64

Hinweis Wenn Sie auf der Programmierkonsole bei einer CS1-H/CJ1-H CPU-Baugruppe, die vor dem 1. Juni 2002 hergestellt wurde und bei der für die Zeitgeber/Zähler-Istwert-Aktualisierung der Binärmodus eingestellt ist, den AWL-Code eines Befehls anzeigen, wird als AWL-Code des Binärbefehls der AWL-Code des BCD-Befehls angezeigt (Beispiel: „TIMX #0000 &16“ wird angezeigt als „TIM #0000 &16“), die Operationen werden jedoch im Binärmodus durchgeführt.

Überprüfen der CPU-Baugruppen-Lot-Nummer

- 1,2,3...**
- Die Lot-Nummer ist unten an der Gerätefront (CS-Serie) oder in der rechten Ecke auf der Oberseite der Baugruppe aufgedruckt (CJ-Serie) und setzt sich aus den letzten zwei Stellen der Jahreszahl, dem Monat und dem Tag zusammen, wie in der folgenden Abbildung dargestellt.
Beispiel: 020601 (Hergestellt am 1. Juni 2002.)

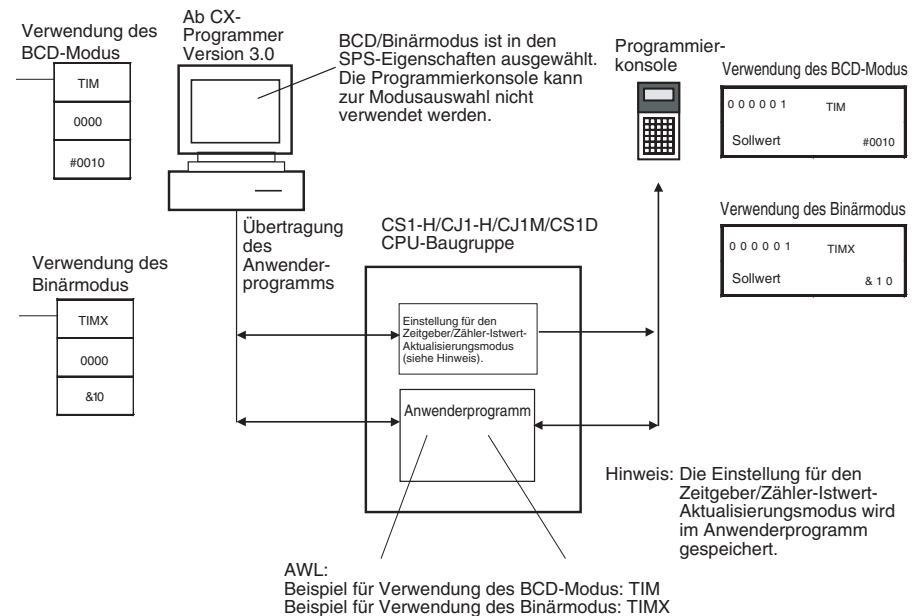


- Überprüfen Sie, welcher Modus gerade eingestellt ist, indem Sie mit CX-Programmer eine Online-Verbindung herstellen, das E/A-Tabellenfenster öffnen und **Baugruppen-Herstellungsinformationen - CPU-Baugruppe** auswählen. Die Lot-Nummer wird im selben Format angezeigt, wie in der Abbildung oben (letzte zwei Stellen des Jahres, Monat, Tag).

6-4-3 Auswahl und Überprüfung des BCD-/Binärmodus

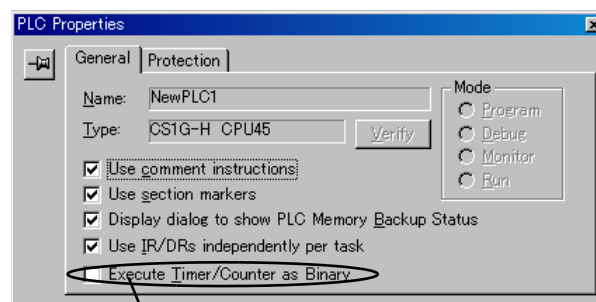
Wenn Sie ein neues Programm schreiben, können Sie den BCD-/Binärmodus in den Einstellungen der SPS-Eigenschaften in CX-Programmer Version 3.0 einstellen.

Hinweis Die Auswahl des BCD-/Binärmodus wird nur von CX-Programmer ab Version 3.0 unterstützt. In CX-Programmer Version 2.1 oder niedriger ist diese Modusauswahl nicht möglich.



Auswahl des BCD-/Binärmodus

- 1,2,3... 1. Klicken Sie mit der rechten Maustaste auf den SPS-Namen, und wählen Sie **Eigenschaften** aus.

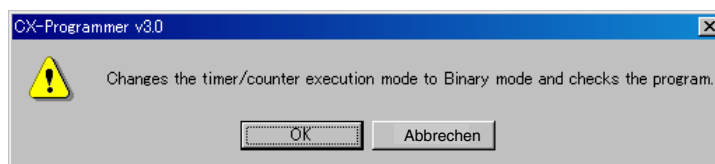


Aktivieren Sie die Einstellung mit diesem

2. Klicken Sie auf die Registerkarte **Allgemeines**, und wählen Sie **Zeitgeber/Zähler als Binärwert ausführen** aus.
- Nicht ausgewählt (Standardeinstellung): BCD-Modus
 - Ausgewählt: Binärmodus

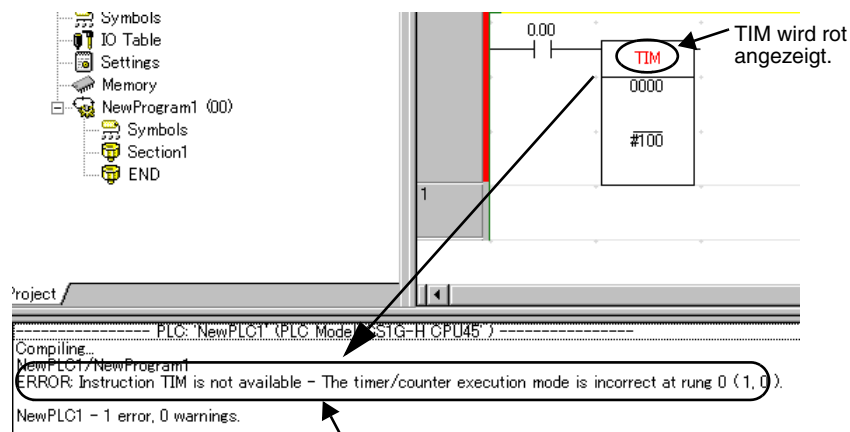
Der Einstellwert für den Modus der Zeitgeber/Zähler-Istwert-Aktualisierung, einzustellen in den SPS-Eigenschaften, wird beim Übertragen des Anwenderprogramms von CX-Programmer zur CPU-Baugruppe im Anwenderspeicher der CPU-Baugruppe gespeichert.

Wenn Sie die Einstellung ändern, wird automatisch das folgende Dialogfeld angezeigt.



Klicken Sie auf die Schaltfläche **OK**, um die Programmüberprüfung auszuführen. Die Ergebnisse der Programmüberprüfung werden im Meldungsfenster angezeigt.

Beispiel: Der Befehl TIM wurde verwendet, obwohl der Binärmodus eingestellt wurde.



Die Ergebnisse der Programmüberprüfung werden im Meldungsfenster angezeigt.

Beispiel: Der Modus für Zeitgeber/Zähler widerspricht der Verwendung des Befehls TIM.

Prüfung des BCD-/Binärmodus

Mit A09915 im Zusatz-Systembereich (Merker für Zeitgeber/Zähler-Istwert-Aktualisierung) können Sie überprüfen, ob eine CPU-Baugruppe im BCD- oder im Binärmodus arbeitet.

Bezeichnung	Adresse	Details
Merker für Zeitgeber/Zähler-Istwert-Aktualisierung	A09915	0: BCD-Modus 1: Binärmodus

6-4-4 AWL-Codes und Daten des BCD-/Binärmodus

AWL-Codes des BCD-/Binärmodus

Die AWL-Codes des Binärmodus bestehen aus dem AWL-Code des BCD-Modus und einem am Ende hinzugefügten „X“.

Beispiel: AWL-Codes für den Befehl TIMER

BCD-Modus: TIM

Binärmodus: TIMX

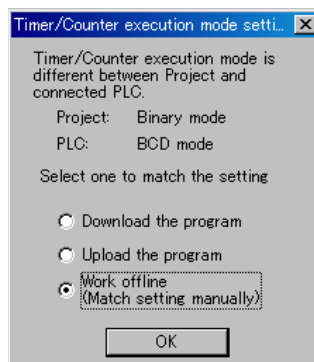
Datenanzeige des BCD-/Binärmodus

SPS-Eigen-schaften	Bedeutung der Eingabe- und Anzeigesymbole	Einstellbereich	Beispiel: Zeitgebernummer: 0000, Sollwert: 1,0 s
BCD-Modus	Das Symbol „#“ kennzeichnet eine Konstante (im BCD-Modus ein BCD-Wert).	#0000 bis #9999 oder #00000000 bis #99999999	
Binärmodus	Das Symbol „&“ kennzeichnet einen Dezimalwert.	&0 bis &65535 oder &0 bis &4294967295	
	Das Symbol „#“ kennzeichnet den Befehls-wert (ein Hexadezimalwert im Binärmodus).	#0000 bis #FFFF oder #0000 bis #FFFFFFFF	

Hinweis Wenn Sie einen numerischen Wert ohne ein Eingabe-/Anzeigesymbol („#“ oder „&“) eingeben, das darauf hinweist, dass eine Konstante folgt (z. B. „TIM 0000 0010“), wird der Zeitgeber/Zähler-Sollwert als Adresse interpretiert (im Beispiel wird der Wert im CIO-Wort 0010 als Sollwert verwendet), unabhängig davon, ob Sie CX-Programmer im BCD- oder Binärmodus verwenden.

6-4-5 Beschränkungen

- Sie können den BCD-Modus und den Binärmodus nicht gleichzeitig in einer CPU-Baugruppe verwenden.
- Wenn Sie mit der Programmierkonsole ein neues Anwenderprogramm erstellen oder Speicher löschen, erfolgt die Zeitgeber/Zähler-Istwert-Aktualisierung immer im BCD-Modus.
- Wenn Sie die CPU-Baugruppe mit CX-Programmer ab Version 3.0 in den Online-Betrieb setzen, wird automatisch der Einstellwert verwendet, der im Anwenderspeicher der CPU-Baugruppe für die Zeitgeber/Zähler-Istwert-Aktualisierung gespeichert ist. Wenn die CPU-Einstellung von der Einstellung des CX-Programmer-Projekts abweicht, tritt ein Fehler auf und die Online-Verbindung wird nicht hergestellt. Folgende Meldung wird angezeigt.



Wählen Sie aus, welche Einstellung übernommen werden soll. Die andere Einstellung wird dementsprechend angepasst.

- Mit CX-Programmer Version 2.1 oder niedriger können nur in den BCD-Modus gesetzte Anwenderprogramme von der CPU-Baugruppe gelesen werden. Der Binärmodus wird nicht unterstützt.

- CX-Programmer und Programmierkonsole reagieren auf die Eingabe eines falschen Zeitgeber/Zähler-Istwert-Aktualisierungsbefehls unterschiedlich:
 - CX-Programmer:
Ein Fehler tritt auf, wenn Sie einen Befehl für einen anderen Modus eingeben, als der in den *SPS-Eigenschaften* für die Zeitgeber/Zähler-Istwert-Aktualisierung spezifizierte Modus.
Beispiel: Wenn das SPS-System im Projekt auf den Binärmodus eingestellt ist, tritt ein Fehler auf, wenn Sie als AWL-Code „TIM“ eingeben. Andererseits tritt ein Fehler auf, wenn Sie im BCD-Modus „TIMX“ als AWL-Code eingeben.
 - Programmierkonsole:
Wenn Sie den Funktionscode eines Befehls für einen anderen Modus als den für die Zeitgeber/Zähler-Istwert-Aktualisierung in der CPU-Baugruppe eingestellten Modus eingeben, wird der AWL-Code automatisch an den eingestellten Modus angepasst.

6-4-6 Befehle und Operanden

Befehle

Befehlstyp	Bezeichnung	AWL	
		BCD-Modus	Binärmodus
Zeitgeber- und Zählerbefehle	TIMER (100 ms)	TIM	TIMX(550)
	HIGH-SPEED TIMER (10 ms)	TIMH(015)	TIMHX(551)
	ONE-MS TIMER (1 ms)	TMHH(540)	TMHHX(552)
	ACCUMULATIVE TIMER (100 ms)	TTIM(087)	TTIMX(555)
	LONG TIMER (100 ms)	TIML(542)	TIMLX(553)
	MULTI-OUTPUT TIMER (100 ms)	MTIM(543)	MTIMX(554)
	COUNTER	CNT	CNTX(546)
	REVERSIBLE COUNTER	CNTR(012)	CNTRX(548)
	RESET TIMER/COUNTER	CNR(545)	CNRX(547)
Blockprogramm-befehle	TIMER WAIT (100 ms)	TIMW(813)	TIMWX(816)
	HIGH-SPEED TIMER WAIT (10 ms)	TMHW(815)	TMHWX(817)
	COUNTER WAIT	CNTW(814)	CNTWX(818)

Befehle und Operanden**Zeitgeber- und Zählerbefehle****TIMER (100 ms)**

Bezeichnung des Befehls	BCD-Modus	Binärmodus
AWL	TIM	TIMX(550)
S (Zeitgeber-Sollwert)	#0000 bis #9999 (BCD)	&0 bis &65535 (dezimal) oder #0000 bis #FFFF (hexadezimal)
Zeiteinstellung (in 0,1-s-Schritten)	0 bis 999,9 s	0 bis 6.553,5 s

HIGH-SPEED TIMER (10 ms)

Bezeichnung des Befehls	BCD-Modus	Binärmodus
AWL	TIMH(015)	TIMHX(551)
S (Zeitgeber-Sollwert)	#0000 bis #9999 (BCD)	&0 bis &65535 (dezimal) oder #0000 bis #FFFF (hexadezimal)
Zeiteinstellung (in 0,01-s-Schritten)	0 bis 99,99 s	0 bis 655,35 s

ONE-MS TIMER (1 ms)

Bezeichnung des Befehls	BCD-Modus	Binärmodus
AWL	TMHH(540)	TMHHX(552)
S (Zeitgeber-Sollwert)	#0000 bis #9999 (BCD)	&0 bis &65535 (dezimal) oder #0000 bis #FFFF (hexadezimal)
Zeiteinstellung (in 0,001-s-Schritten)	0 bis 9,999 s	0 bis 65,535 s

ACCUMULATIVE TIMER (100 ms)

Bezeichnung des Befehls	BCD-Modus	Binärmodus
AWL	TTIM(087)	TTIMX(555)
S (Zeitgeber-Sollwert)	#0000 bis #9999 (BCD)	&0 bis &65535 (dezimal) oder #0000 bis #FFFF (hexadezimal)
Zeiteinstellung (in 0,1-s-Schritten)	0 bis 999,9 s	0 bis 6.553,5 s

LONG TIMER (100 ms)

Bezeichnung des Befehls	BCD-Modus	Binärmodus
AWL	TIML(542)	TIMLX(553)
S, S+1 (Zeitgeber-Sollwerte)	#00000000 bis #99999999 (BCD)	&0 bis &4294967295 (dezimal) oder #0000 bis #FFFFFFFF (hexadezimal)
Zeiteinstellung (in 0,1-s-Schritten)	0 bis 999,9 s	0 bis 6.553,5 s

MULTI-OUTPUT TIMER (100 ms)

Bezeichnung des Befehls	BCD-Modus	Binärmodus
AWL	MTIM(543)	MTIMX(554)
S bis S-7 (Sollwerte für jeden Zeitgeber)	#0000 bis #9999 (BCD)	&0 bis &65535 (dezimal) oder #0000 bis #FFFF (hexadezimal)
Zeiteinstellung (in 0,1-s-Schritten)	0 bis 999,9 s	0 bis 6.553,5 s

COUNTER

Bezeichnung des Befehls	BCD-Modus	Binärmodus
AWL	CNT	CNTX(546)
S (Zähler-Sollwert)	#0000 bis #9999 (BCD)	&0 bis &65535 (dezimal) oder #0000 bis #FFFF (hexadezimal)
Einstellung	0 bis 9.999	0 bis 65.535

REVERSIBLE COUNTER

Bezeichnung des Befehls	BCD-Modus	Binärmodus
AWL	CNTR(012)	CNTRX(548)
S (Zähler-Sollwert)	#0000 bis #9999 (BCD)	&0 bis &65535 (dezimal) oder #0000 bis #FFFF (hexadezimal)
Einstellung	0 bis 9.999	0 bis 65.535

RESET TIMER/COUNTER

Bezeichnung des Befehls	BCD-Modus	Binärmodus
AWL	CNR(545)	CNRX(547)

Blockprogrammbefehle**TIMER WAIT (100 ms)**

Bezeichnung des Befehls	BCD-Modus	Binärmodus
AWL	TIMW(813)	TIMWX(816)
S (Zeitgeber-Sollwert)	#0000 bis #9999 (BCD)	&0 bis &65535 (dezimal) oder #0000 bis #FFFF (hexadezimal)
Zeiteinstellung (in 0,1-s-Schritten)	0 bis 999,9 s	0 bis 6.553,5 s

HIGH-SPEED TIMER WAIT (10 ms)

Bezeichnung des Befehls	BCD-Modus	Binärmodus
AWL	TMHW(815)	TMHWX(817)
S (Zeitgeber-Sollwert) Einheit: 0,01 s	#0000 bis #9999 (BCD)	&0 bis &65535 (dezimal) oder #0000 bis #FFFF (hexadezimal)
Zeiteinstellung (in 0,01-s-Schritten)	0 bis 999,9 s	0 bis 655,35 s

COUNTER WAIT

Bezeichnung des Befehls	BCD-Modus	Binärmodus
AWL	CNTW(814)	CNTWX(818)
S (Zähler-Sollwert)	#0000 bis #9999 (BCD)	&0 bis &65535 (dezimal) oder #0000 bis #FFFF (hexadezimal)
Einstellung	0 bis 9.999	0 bis 65.535

6-5 Verwenden eines zeitgesteuerten Interrupts als Präzisionszeitgeber (nur CJ1M)

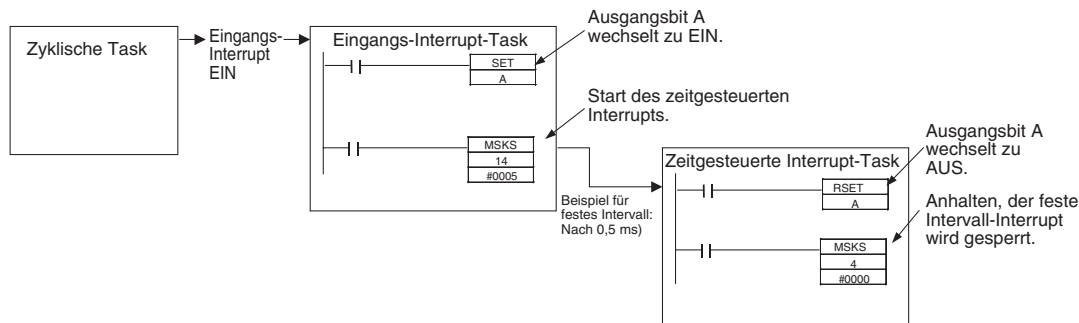
Mit CJ1M CPU-Baugruppen können Sie einen zeitgesteuerten Interrupt mit den folgenden Funktionen als Präzisionszeitgeber verwenden.

- Sie können den zeitgesteuerten Interrupt-Zeitgeber in 0,1-ms-Schritten eingeben (Präzisions-Intervallzeitgeber).
- Mit dem Befehl MSKS(690) ist ein Rücksetzen (d. h. ein Neustart) möglich (feste Zeit bis zum ersten Interrupt).
- Der Istwert von Intervallzeitgebern kann mit dem Befehl MSKR(692) gelesen werden.

Mit diesen Funktionen können Sie Anwendungen wie das folgende Beispiel entwickeln: Eine Impulsdauer-Zeitfunktion mit Präzisionszeitgeber, in der das auf EIN gesetzte Eingangsbit als Trigger für das Ausgangsbit dient, das auf EIN und nach einem festen Intervall wieder auf AUS gesetzt wird.

Beispiel:

- 1,2,3...**
1. Die Eingangs-Interrupt-Task startet, wenn das Bit des integrierten Eingangs auf EIN gesetzt wird.
 2. In der Eingangs-Interrupt-Task wird Ausgangsbit A auf EIN gesetzt und der Befehl MSKS(690) ausgeführt, um einen Start des zeitgesteuerten Interrupts durchzuführen.
 3. Nach einem festen Intervall startet die zeitgesteuerte Interrupt-Task und das Ausgangsbit A der zeitgesteuerten Interrupt-Task wird auf AUS gesetzt und der Befehl MSKS(690) wird ausgeführt, um einen zeitgesteuerten Interrupt zu deaktivieren.



6-5-1 Einstellen des zeitgesteuerten Interrupts auf 0,1-ms-Schritte

Das Intervall für zeitgesteuerte Interrupts wird im SPS-Setup mit der Einstellung für die Zeiteinheit des zeitgesteuerten Interrupts und dem Befehl MSKS(690) gesetzt.

Bei CJ1M CPU-Baugruppen können Sie das Zeitintervall für zeitgesteuerte Interrupts im Bereich von 0,5 ms bis 999,9 ms in 0,1-ms-Schritten einstellen.

SPS-Setup

Beschreibung	SPS-Adresse		Einstellwert	Standard-einstellung	Aktualisierungs-zeitpunkt
	Wort	Bit			
Einstellung der Zeiteinheit des zeitgesteuerten Interrupts	195	00 bis 03	0 Hex: 10-ms-Schritte 1 Hex: 1-ms-Schritte 2 Hex: 0,1-ms-Schritte (nur CJ1M CPU-Baugruppen)	0 Hex	Bei Betriebsstart

6-5-2 Spezifizieren einer Rücksetzung mit MSKS(690)

Wenn Sie bei einer CJ1M CPU-Baugruppe mit dem Befehl MSKS(690) einen zeitgesteuerten Interrupt starten, kann der interne Zeitgeber vor dem Starten des Interrupts zurückgesetzt werden (dies wird "Rücksetzung" genannt).

Mit dieser Methode können Sie die Zeit für den ersten Interrupt angeben, ohne den Befehl CLI(691) zu verwenden.

Zeitgesteuerte Interrupts werden mit dem Befehl MSKS(690) gestartet, um die Zeit für zeitgesteuerte Interrupts (das Intervall zwischen zwei Interrupts) zu setzen. Nach der Ausführung des Befehls MSKS(690) kann die vor dem ersten Start der zeitgesteuerten Interrupt-Task erforderliche Zeit (erste Interrupt-Startzeit) jedoch nur mit dem Befehl CLI(691) festgelegt werden. Die CJ1M CPU-Baugruppen bieten eine interne Zeitgeber-Rücksetzung, mit der die Zeit bis zum ersten Interrupt ohne den Befehl CLI(691) gesetzt werden kann.

MSKS(690)-Befehlsoperand (nur, wenn zeitgesteuerter Interrupt spezifiziert wurde)

Operand	Einstellwert
N (Interrupt-Bezeichner)	4: Zeitgesteuerter Interrupt 0, normale Einstellung (interner Zeitgeber nicht zurückgesetzt)
	5: Zeitgesteuerter Interrupt 1, normale Einstellung (interner Zeitgeber nicht zurückgesetzt)
	14: Zeitgesteuerter Interrupt 0, spezifiziert Rücksetzung (nur CJ1M CPU-Baugruppen)
	15: Zeitgesteuerter Interrupt 1, spezifiziert Rücksetzung (nur CJ1M CPU-Baugruppen)

6-5-3 Lesen des Istwerts des internen Zeitgebers mit MSKR(692)

Bei CJ1M CPU-Baugruppen können Sie den Istwert des internen Zeitgebers auslesen, mit dem die Zeit für zeitgesteuerte Interrupts gemessen wird. Die Zeit wird vom Startpunkt oder vom vorherigen Interrupt-Punkt des zeitgesteuerten Interrupts gelesen. Der Istwert des internen Zeitgebers wird mit dem Befehl MSKR(692) gelesen. Die Zeiteinheit und das Intervall für zeitgesteuerte Interrupts variieren je nach der entsprechenden Einstellung im SPS-Setup.

MSKR(692)-Operand (nur, wenn zeitgesteuerter Interrupt spezifiziert wurde)

Operand	Einstellwert
N (Interrupt-Bezeichner)	4: Zeitgesteuerter Interrupt 0, liest Intervall für zeitgesteuerte Interrupts (Sollwert)
	5: Zeitgesteuerter Interrupt 1, liest Intervall für zeitgesteuerte Interrupts (Sollwert)
	14: Zeitgesteuerter Interrupt 0, liest Istwert des internen Zeitgebers (nur CJ1M CPU-Baugruppen)
	15: Zeitgesteuerter Interrupt 1, liest Istwert des internen Zeitgebers (nur CJ1M CPU-Baugruppen)

6-6 Einstellungen für Start und Wartung

In diesem Abschnitt werden die folgenden Funktionen für Start und Wartung beschrieben.

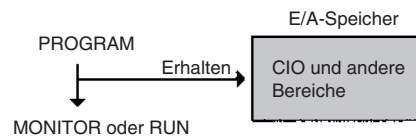
- Warmstart-/Warmstopp-Funktionen
- Einstellung der Betriebsart beim Einschalten
- Einstellung der Verzögerungszeit bei der Erkennung des Spannungsausfalls
- Deaktivieren der Ausschalt-Interrupts
- RUN-Ausgang
- Uhr
- Programmschutz
- Dezentrale Programmierung und Überwachung
- Flash-Speicher
- Einstellung der Startbedingungen

6-6-1 Warmstart-/Warmstopp-Funktionen

Betriebsart-Wechsel

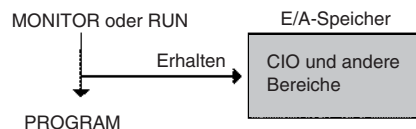
Warmstart

Setzen Sie den E/A-Speicher-Haftmerker (A50012) auf EIN, um alle Daten im E/A-Speicher zu erhalten*, wenn die CPU-Baugruppe für die Ausführung des Programms von der PROGRAM-Betriebsart in die RUN-/MONITOR-Betriebsart umgeschaltet wird.



Warmstopp

Wenn Sie den E/A-Speicher-Haftmerker (A50012) auf EIN setzen, bleiben auch alle Daten im E/A-Speicher erhalten*, wenn die CPU-Baugruppe zum Anhalten des Programms von der RUN-/MONITOR-Betriebsart in die PROGRAM-Betriebsart umgeschaltet wird.



Hinweis

*Die folgenden Bereiche im E/A-Speicher werden beim Umschalten zwischen den Betriebsarten gelöscht (PROGRAM ↔ RUN/MONITOR), wenn der E/A-Speicher-Haftmerker nicht auf EIN gesetzt ist: CIO-Bereich (E/A-Bereich, Data-Link-Bereich, CPU-Bus-Baugruppen-Bereich, Spezial-E/A-Baugruppen-Bereich, Bereich für internes Modul, SYSMAC BUS-Bereich, Bereich für dezentrale E/A-Module, DeviceNet-Bereich (CompoBus/D) und interne E/A-Bereiche), Arbeitsbereich, Zeitgeber-Ablaufmerker und Zeitgeber-Istwerte. (Bereich für internes Modul, SYSMAC BUS-Bereich und Bereich für dezentrale E/A-Module werden nur von CPU-Baugruppen der CS-Serie unterstützt.)

Merker und Worte des Zusatz-Systembereichs

Bezeichnung	Adresse	Beschreibung
E/A-Speicher-Haftmerker	A50012	Wenn dieses Bit auf EIN gesetzt ist, bleibt beim Ändern der Betriebsart (PROGRAM ↔ RUN/MONITOR) der gesamte E/A-Speicher erhalten.

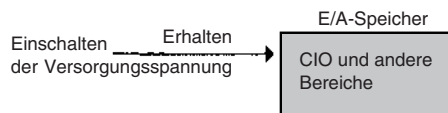
Wenn der E/A-Speicher-Haftmerker auf EIN gesetzt ist, bleiben beim Anhalten des Programms alle Zustände der Ausgänge von Ausgängen-Baugruppen erhalten.

Einschalten des SPS-Systems

ten. Wird das Programm wieder gestartet, haben die Ausgänge denselben Status wie beim Anhalten des Programms.

(Wenn der E/A-Speicher-Haftmerker auf AUS gesetzt ist, werden die Ausgangszustände vor dem Ausführen der Befehle gelöscht.)

Um beim Einschalten des SPS-Systems (AUS → EIN) alle Daten im E/A-Speicher zu erhalten*, muss der E/A-Speicher-Haftmerker auf EIN gesetzt und im SPS-Setup geschützt sein (Adresse 80, IOM Hold Bit Status at Startup (E/A-Speicher-Haftmerker beim Start)).



Merker und Worte des Zusatz-Systembereichs

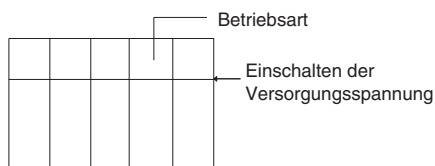
Bezeichnung	Adresse	Beschreibung
E/A-Speicher-Haftmerker	A50012	Wenn dieses Bit auf EIN gesetzt ist, bleibt beim Ändern der Betriebsart (PROGRAM ↔ RUN/MONITOR) der gesamte E/A-Speicher erhalten.

SPS-Setup

Adresse der Programmierkonsole	Bezeichnung	Einstellung	Standard-einstellung
80 Bit 15	IOM Hold Bit Status at Startup (E/A-Speicher-Haftmerker beim Start)	0: Der E/A-Speicher-Haftmerker wird beim Einschalten auf 0 zurückgesetzt. 1: Der E/A-Speicher-Haftmerker bleibt beim Einschalten erhalten.	0 (gelöscht)

6-6-2 Einstellung der Betriebsart beim Einschalten

Sie können in den SPS-Einstellungen die Betriebsart der CPU-Baugruppe (Betriebsart nach dem Einschalten der Spannungsversorgung) festlegen.



SPS-Setup

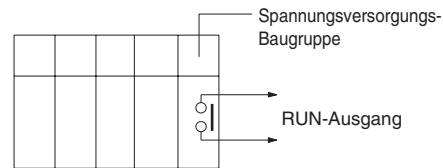
Adresse der Programmierkonsole	Bezeichnung	Bedeutung	Einstellung	Standard-einstellung
81	Betriebsart beim Einschalten	Spezifiziert die Betriebsart beim Einschalten.	PRCN: Betriebsarten-Schaltereinstellung der Programmierkonsole PRG: PROGRAM-Betriebsart MON: MONITOR-Betriebsart RUN: RUN-Betriebsart	PRCN: Betriebsarten-Schaltereinstellung der Programmierkonsole

Hinweis

Wenn die Startbetriebsart auf „PRCN“ (Betriebsarten-Schaltereinstellung der Programmierkonsole) gesetzt ist, jedoch keine Programmierkonsole angeschlossen ist, startet die CPU-Baugruppe in der RUN-Betriebsart. Ändern Sie die Standardeinstellung im SPS-Setup, um nach dem Einschalten automatisch in die MONITOR- oder PROGRAM-Betriebsart zu wechseln. (Eine CS1 CPU-Baugruppe der CS-Serie startet unter denselben Bedingungen in der PROGRAM-Betriebsart.)

6-6-3 RUN-Ausgang

Einige Spannungsversorgungs-Baugruppen (C200HW-PA204R, C200HW-PA209R, CJ1W-PA205R und CS1D-PA207R) sind mit einem RUN-Ausgang ausgestattet. Dieser Ausgangskontakt ist in der RUN- oder MONITOR-Betriebsart EIN (geschlossen), in der PROGRAM-Betriebsart AUS (offen).



Dieser RUN-Ausgang kann dazu verwendet werden, einen externen Sicherheitsschaltkreis aufzubauen, z. B. einen NOT-AUS-Schaltkreis, mit dem sichergestellt wird, dass eine externe Spannungsversorgung der Ausgangs-Baugruppen nur Spannung liefert, wenn das SPS-System eingeschaltet ist.

Hinweis Sie können einen vergleichbaren Ausgang erstellen, indem Sie bei Verwendung einer Spannungsversorgungs-Baugruppe ohne RUN-Ausgang als Ausführungsbedingung für einen Ausgang einer Ausgangsbaugruppe den Immer-EIN-Merker (P_On) im Programm verwenden.

Achtung Wenn die externe Spannungsversorgung der Ausgangs-Baugruppe vor der Spannungsversorgung des SPS-Systems funktionsbereit ist, kann es beim ersten Einschalten des SPS-Systems möglicherweise zu einer vorübergehenden Fehlfunktion der Ausgangs-Baugruppen kommen. Richten Sie zur Vermeidung von Fehlfunktionen eine externe Schaltung ein, um zu verhindern, dass die Spannungsversorgung der Ausgangs-Baugruppe vor der Spannungsversorgung des SPS-Systems eingeschaltet wird. Erstellen Sie eine ausfallsichere Schaltung wie die oben beschriebene, um sicherzustellen, dass eine externe Spannungsversorgung nur Spannung liefert, wenn das SPS-System in der RUN- oder MONITOR-Betriebsart betrieben wird.

6-6-4 Einstellung der Ausschalterkennungs-Verzögerungszeit

Normalerweise wird eine Spannungsunterbrechung ungefähr 10 bis 25 ms (2 bis 5 ms bei DC-Spannungsquellen) nach dem Absinken der Eingangsspannung unter 85 % des Mindestnennwerts (80 % bei DC-Spannungsquellen) erkannt. Es gibt eine Einstellung im SPS-Setup (Adresse 225, Bits 0 bis 7, Ausschalterkennungszeit), mit der Sie diese Zeit bis auf 10 ms (2 ms bei DC-Spannungsquellen) verlängern können.

Wenn die Ausschalt-Interrupt-Task aktiviert ist, wird diese zum Zeitpunkt der Bestätigung der Spannungsunterbrechung ausgeführt, andernfalls wird die CPU zurückgesetzt und der Betrieb angehalten.

Zugehörige Einstellungen

Adresse	Bezeichnung	Bedeutung	Einstellung	Standard-einstellung
CIO 256, Bits 00 bis 07	Ausschalterkennungszeit	Stellen Sie die Zeit ein, um die die Erkennung einer Spannungsunterbrechung verzögert werden soll.	00 bis 0A (Hex): 0 bis 10 ms	00 (Hex): 0 ms

6-6-5 Deaktivieren der Ausschalt-Interrupts

Diese Funktion wird nur von den CS1-H, CJ1-H, CJ1M und CS1D CPU-Baugruppen unterstützt.

Bei CS1-H, CJ1-H, CJ1M und CS1D CPU-Baugruppen können Programmbereiche vor Spannungsunterbrechungen geschützt werden, so dass sie sogar bei einem Ausfall der Spannungsversorgung von der CPU-Baugruppe ausge-

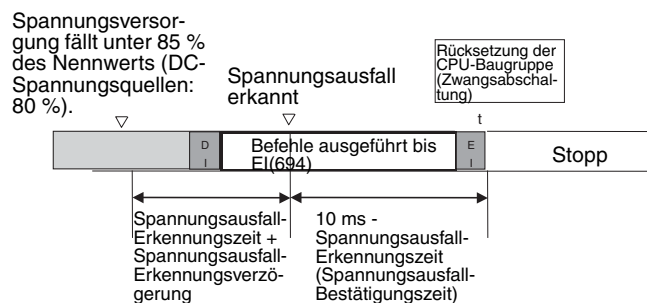
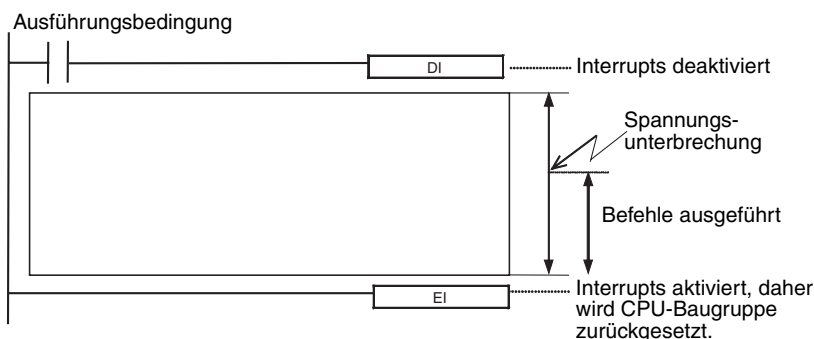
führt werden. Dies geschieht mit den Befehlen DISABLE INTERRUPTS (DI(693)) und ENABLE INTERRUPTS (EI(694)).

Verwenden Sie diese Funktion für einen Satz von Befehlen, die als Gruppe ausgeführt werden müssen, z. B. damit die Ausführung beim nächsten Einschalten nicht mit zwischengespeicherten Daten startet.

Vorgehensweise

1,2,3...

1. Setzen Sie zur Deaktivierung von Ausschalt-Interrupts die Einstellung in A530 auf A5A5 Hex, um eine Unterbrechung geschützter Programmabschnitte bei Spannungsausfall zu verhindern.
2. Aktivieren Sie im SPS-Setup das Deaktivieren von Ausschalt-Interrupts (Standardeinstellung).
3. Leiten Sie den zu schützenden Programmabschnitt mit dem Befehl DI(693) ein, um die Ausführung der Interrupt-Tasks zu verhindern. Schließen Sie den zu schützenden Programmabschnitt mit dem Befehl EI(694) ab, um die Ausführung von Interrupt-Tasks wieder zuzulassen. Tritt nun während der Ausführung eines geschützten Programmabschnitts ein Spannungsausfall auf, werden zunächst alle Befehle zwischen DI(693) und EI(694) ausgeführt, bevor die CPU angehalten wird.



Zugehörige Einstellungen

Bezeichnung	Adresse	Bedeutung
Einstellung zur Deaktivierung von Ausschalt-Interrupts	A530	Ermöglicht den Schutz von Programmabschnitten vor der Verarbeitung von Ausschalt-Interrupts mit DI(693) und EI(694) (außer Ausführung der Ausschalt-Interrupt-Task). A5A5 Hex: Aktiviert den Schutz vor Verarbeitung von Ausschalt-Interrupts mit DI(693). Jeder andere Wert: Deaktiviert den Schutz vor Verarbeitung von Ausschalt-Interrupts mit DI(693).

6-6-6 Uhrfunktionen

Die SPS-Systeme der CS/CJ-Serie verfügen über die folgenden Uhrfunktionen:

- Überwachen der Uhrzeit, zu der ein Spannungsausfall aufgetreten ist.

- Überwachen der Uhrzeit, zu der das SPS-System eingeschaltet wurde.
- Überwachen der Gesamtzeit, die das SPS-System eingeschaltet war.

Hinweis

Die CS-Serie CS1 CPU-Baugruppen werden ohne installierte Erhaltungsbatterie geliefert und die interne Uhr der CPU-Baugruppe startet mit der Zeit „00/01/01 00:00:00“ oder möglicherweise einer andere Zeit, wenn die Batterie angeschlossen wird. Um die Uhrfunktionen zu verwenden, schließen Sie die Batterie an, schalten Sie die Spannungsversorgung ein, und stellen Sie die Uhrzeit und das Datum mit einem Programmiergerät (Programmierkonsole oder CX-Programmer) oder dem FINS-Befehl (07 02, CLOCK WRITE). Die interne Uhr der CPU-Baugruppe startet nach dem Stellen.

Merker und Worte des Zusatz-Systembereichs

Bezeichnung	Adressen	Funktion
Uhrzeit und Datum	A35100 bis A35107	Sekunden: 00 bis 59 (BCD-Format)
	A35108 bis A35115	Minuten: 00 bis 59 (BCD-Format)
	A35200 bis A35207	Stunden: 00 bis 23 (BCD-Format)
	A35208 bis A35215	Tag: 00 bis 31 (BCD-Format)
	A35300 bis A35307	Monat: 00 bis 12 (BCD-Format)
	A35308 bis A35315	Jahr: 00 bis 99 (BCD-Format)
	A35400 bis A35407	Wochentag: 00: Sonntag, 01: Montag, 02: Dienstag, 03: Mittwoch, 04: Donnerstag, 05: Freitag, 06: Samstag
Startzeitpunkt	A510 und A511	Enthalten den Zeitpunkt, an dem die Spannungsversorgung eingeschaltet wurde.
Ausschaltzeitpunkt	A512 und A513	Enthalten den Zeitpunkt, an dem die Spannungsversorgung das letzte Mal unterbrochen wurde.
Gesamteinschaltdauer	A523	Dieses Wort enthält die Gesamteinschaltdauer des SPS-Systems (Binärformat, in 10-Stunden-Einheiten).

Zugehörige Befehle

Befehl	Bezeichnung	Funktion
SEC(065)	HOURS TO SECONDS	Wandelt Zeitdaten aus Stunden/Minuten/Sekunden in die entsprechende Zeit nur in Sekunden um.
HMS(066)	SECONDS TO HOURS	Wandelt Sekundendaten in die entsprechende Zeit in Stunden/Minuten/Sekunden um.
CADD(730)	CALENDAR ADD	Addiert Zeit zu den Kalenderdaten in den spezifizierten Worten.
CSUB(731)	CALENDAR SUBTRACT	Subtrahiert Zeit von den Kalenderdaten in den spezifizierten Worten.
DATE(735)	CLOCK ADJUSTMENT	Ändert die Einstellung der internen Uhr auf die Einstellung in den spezifizierten Worten.

6-6-7 Programmschutz

Das CS/CJ-Serie Anwenderprogramm kann schreibgeschützt oder lese- und schreibgeschützt werden.

Schreibschutz mit Hilfe des DIP-Schalters

Sie können das Anwenderprogramm gegen Überschreiben schützen, indem Sie Schalter 1 des DIP-Schalterblocks der CPU-Baugruppe auf EIN stellen. Wenn dieser Schalter auf EIN gestellt ist, kann das Anwenderprogramm nicht durch ein Programmiergerät (auch nicht durch Programmierkonsolen) geändert werden. Mit dieser Funktion können Sie verhindern, dass das Programm versehentlich überschrieben wird.

Das Programm kann jedoch immer noch angezeigt und gelesen werden.

- Hinweis** Bei CS/CJ-Serie CPU-Baugruppen ab Version 2.0 kann der oben genannte DIP-Schalter verwendet werden. Wenn Sie CX-Programmer ab Version 4.0 einsetzen, können Sie beim Einrichten eines Kennworts für das gesamte Programm oder für bestimmte Tasks den Lese-/Schreibschutz des Programms als Option festlegen. Weitere Informationen finden Sie im *CS-Serie SPS Bedienerhandbuch* und im *CJ-Serie SPS Bedienerhandbuch* unter 1-4-2 *Verbesserter Leseschutz durch Verwendung von Passwörtern* im Abschnitt *Aktivieren/Deaktivieren der Erstellung von Dateispeicher-Programmdateien*.

Lese-/Schreibschutz durch Verwendung von Passwörtern

In CX-Programmer können Sie den Lese- und Schreibzugriff auf den Anwenderprogrammbereich sperren. Dadurch schützen Sie Ihr Programm vor nicht autorisiertem Kopieren und verhindern den Verlust des geistigen Eigentums. Von einem Programmiergerät aus können Sie ein Passwort für den Programmschutz angeben und dadurch den Zugriff auf das gesamte Programm verhindern.

- Hinweis**
1. Wenn Sie das Passwort vergessen, kann das Programm nicht vom SPS-System auf einen Computer übertragen werden. Notieren Sie sich das Passwort, und bewahren Sie es an einem sicheren Ort auf.
 2. Wenn Sie das Passwort vergessen, können Sie keine Programme vom Computer auf das SPS-System übertragen. Programme können vom Computer auf das SPS-System übertragen werden, auch wenn der Passwortschutz nicht aufgehoben wurde.

Passwortschutz

- 1,2,3...**
1. Registrieren Sie ein Passwort im Online- oder Offlinemodus wie folgt:
 - a) Wählen Sie die SPS aus, und wählen Sie im Ansichtsmenü **Eigenschaften** aus.
 - b) Wählen Sie im Dialogfeld „SPS-Eigenschaften“ **Schutz** aus, und geben Sie das Passwort ein.
 2. Im Onlinemodus richten Sie den Passwortschutz wie folgt ein:
 - a) Wählen Sie **SPS, Schutz** und dann **Passwort eingeben** aus. Das Dialogfeld „Leseschutz einrichten“ wird angezeigt.
 - b) Klicken Sie auf **OK**.

- Hinweis** Bei CS/CJ-Serie CPU-Baugruppen ab Version 2.0 können Sie den Leseschutz nicht nur für das gesamte Programm, sondern auch für bestimmte Tasks setzen. Weitere Informationen finden Sie im *CS-Serie SPS Bedienerhandbuch* oder dem *CJ-Serie SPS Bedienerhandbuch* unter 1-4-2 *Verbesserter Leseschutz durch Verwendung von Passwörtern* im Abschnitt *Leseschutz für einzelne Tasks durch Verwendung von Passwörtern*.

Prüfen des Anwenderprogrammdatums

Bei einer CS1-H, CJ1-H, CJ1M oder CS1D CPU-Baugruppe können Sie anhand des Inhalts von A090 bis A097 das Datum überprüfen, an dem das Programm und die Parameter erstellt wurden.

Worte des Zusatz-Systembereichs

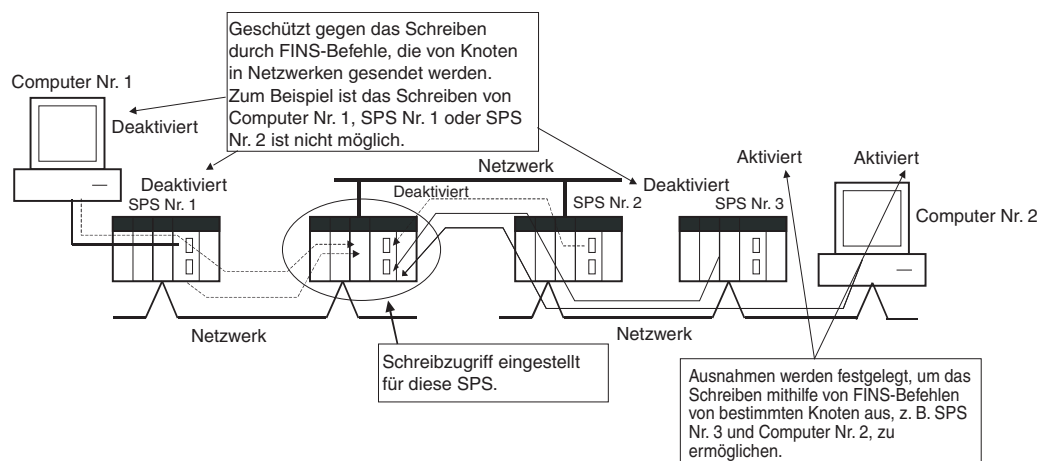
Bezeichnung	Adresse	Beschreibung
Anwenderprogrammdatum	A090 bis A093	Das Datum und die Uhrzeit des letzten Überschreibens des Anwenderprogramms im Speicher werden im BCD-Format angegeben.
		A09000 bis A09007 Sekunden (00 bis 59, BCD-Format)
		A09008 bis A09015 Minuten (00 bis 59, BCD-Format)
		A09100 bis A09107 Stunden (00 bis 23, BCD-Format)
		A09108 bis A09115 Tag (01 bis 31, BCD-Format)
		A09200 bis A09207 Monat (01 bis 12, BCD-Format)
		A09208 bis A09215 Jahr (00 bis 99, BCD-Format)
		A09300 bis A09307 Wochentag (00 bis 06, BCD-Format) Wochentag: 00: Sonntag, 01: Montag, 02: Dienstag, 03: Mittwoch, 04: Donnerstag, 05: Freitag, 06: Samstag
Parameterdatum	A094 bis A097	Das Datum und die Uhrzeit des letzten Überschreibens der Parameter im Speicher werden im BCD-Format angegeben. Das Format entspricht dem des Anwenderprogrammdatums, siehe oben.

6-6-8 Schreibschutz gegen FINS-Befehle, die über Netzwerke an die CPU-Baugruppen gesendet werden

Bei CS/CJ-Serie CPU-Baugruppen ab Version 2.0 können Sie die Schutzfunktion aktivieren und dadurch verhindern, dass über Netzwerke (d. h., andere Verbindungen als direkte serielle Verbindungen) mit FINS-Befehlen auf CPU-Baugruppen geschrieben werden kann oder sie in irgendeiner Weise aus der Ferne gesteuert werden können. Dazu zählt neben dem Schreiben aus FinsGateway-Anwendungen auch das aus CX-Programmer, CX-Protocol und CX-Process. Lesen ist in diesem Fall jedoch möglich.

Die folgenden Vorgänge sind gesperrt, wenn der Schreibschutz aktiviert ist: Herunterladen von Daten (Anwenderprogramm, SPS-Setup, E/A-Speicher oder weitere Daten), Ändern der Betriebsart, Online-Bearbeitung und jegliche weitere Schreib- oder Steuervorgänge.

Wenn der Schreibschutz aktiviert ist, können Sie bestimmte Knoten jedoch als Ausnahmen setzen, um bei diesen Schreib- und Steuervorgängen auszuführen.

**Hinweis**

Mit dieser Funktion werden nur Schreib-/Steuervorgänge durch FINS-Befehle verhindert. Sie verhindert keine Schreib-/Steuervorgänge, die mit anderen Methoden ausgeführt werden, beispielsweise über Data-Link.

Weitere Informationen finden Sie im *CS-Serie SPS Bedienerhandbuch* oder im *CJ-Serie SPS Bedienerhandbuch* unter 1-4-3 *Schreibschutz gegen FINS-Schreibbefehle, die über Netzwerke an CPU-Baugruppen gesendet werden*.

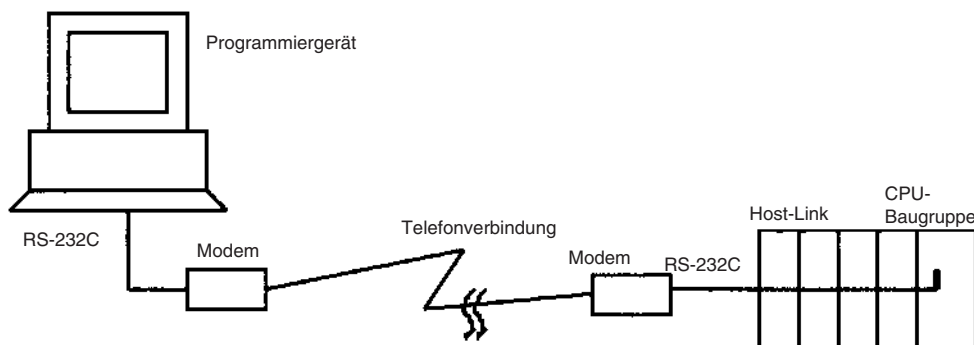
6-6-9 Dezentrale Programmierung und Überwachung

Sie können CS/CJ-Serie SPS-Systeme dezentral über ein Modem oder ein Controller-Link-Netzwerk programmieren und überwachen.

1,2,3...

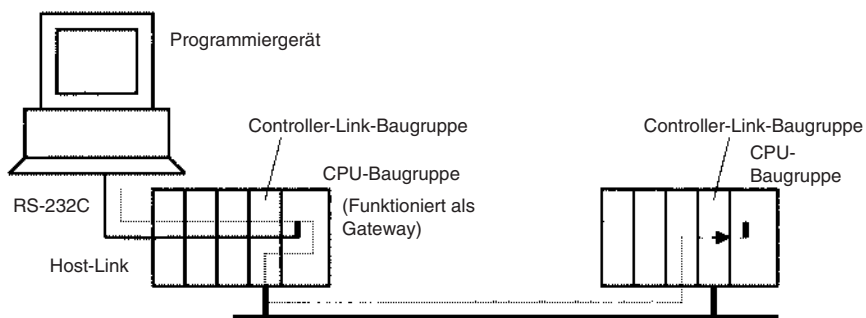
1. Verbindungen über Modem

Die Host-Link-Funktion kann über ein Modem ausgeführt werden. Dadurch werden der Betrieb eines dezentralen SPS-Systems, Datenübertragungen und sogar die Online-Bearbeitung des Programms eines dezentralen SPS-Systems per Telefonnetz ermöglicht. Von diesen Verbindungen werden alle Online-Vorgänge des Programmiergeräts unterstützt.



2. Verbindungen über Controller-Link-Netzwerk

SPS-Systeme in einem Controller-Link- oder Ethernet-Netzwerk können über Host-Link programmiert und überwacht werden. Von diesen Verbindungen werden alle Online-Vorgänge des Programmiergeräts unterstützt.



6-6-10 Baugruppen-Profile

Die folgenden Informationen zu CS/CJ-Serie Baugruppen können von CX-Programmer gelesen werden.

- Herstellungsinformationen (Lot-Nummer, Seriennummer usw.): Erleichtert beim Auftreten von Problemen mit Baugruppen das Bereitstellen von Informationen für OMRON.
- Baugruppeninformationen (Typ, Produktbezeichnung, genaue Rack-/Steckplatzposition): Bietet einen einfachen Zugriff auf Installationsinformationen.
- Anwenderdefinierten Text (max. 256 Zeichen): Gestattet die Aufzeichnung von zur Wartung erforderlichen Informationen (Historie der Baugruppeninspektionen, Herstellungsnummern und weiterer Anwendungsinformationen) auf Speicherkarten.

6-6-11 Flash-Speicher

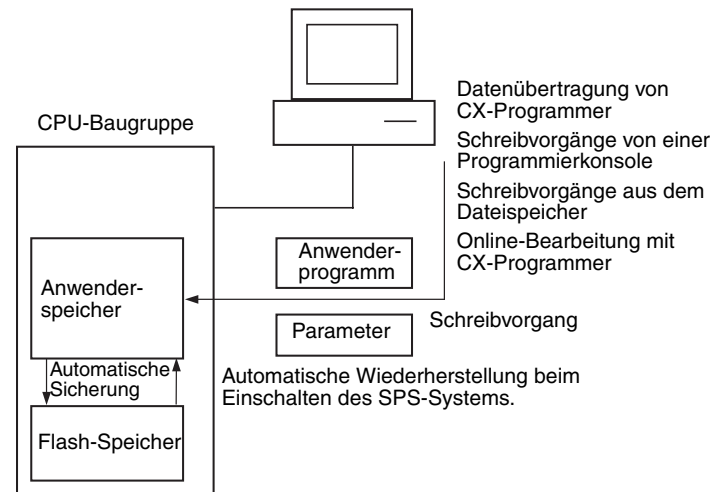
Diese Funktion wird nur von den CS1-H, CJ1-H, CJ1M und CS1D CPU-Baugruppen unterstützt.

Bei CS1-H, CJ1-H, CJ1M und CS1D CPU-Baugruppen werden das Anwenderprogramm und die Parameter automatisch im Flash-Speicher gesichert, wenn sie in der CPU-Baugruppe geschrieben oder geändert werden.

Die folgenden Daten werden automatisch gesichert: Anwenderprogramm und Parameter (einschließlich SPS-Setup, registrierte E/A-Tabellen, Routing-Tabellen und Daten der CPU-Bus-Baugruppe, z. B. die Data-Link-Tabellen).

Die Daten werden automatisch gesichert, wenn das Anwenderprogramm oder die Parameter in der CPU-Baugruppe geschrieben werden, unter anderem auch bei folgenden Vorgängen: Datenübertragungsvorgänge von CX-Programmer, Schreiben von Daten von einer Programmierkonsole, Online-Bearbeitung und Datenübertragungen von einer Speicherkarte oder aus dem EM-Dateispeicher.

Die in den Flash-Speicher geschriebenen Anwenderprogramm- und Parameterdaten werden beim Start automatisch in den Anwenderspeicher der CPU-Baugruppe übertragen.



Hinweis

1. Die BKUP-Anzeige an der Vorderseite der CPU-Baugruppe leuchtet, während Daten in den Flash-Speicher geschrieben werden. Schalten Sie nach der Datenübertragung von einem Programmiergerät bzw. aus dem Dateispeicher oder nach der Durchführung einer Online-Bearbeitung die Spannungsversorgung der CPU-Baugruppe nicht aus, bis der Sicherungsvorgang abgeschlossen ist (d. h., bis die BKUP-Anzeige erlischt).
2. Nur bei Online-Bearbeitung und wenn in der CPU-Baugruppe eine Batterie eingelegt ist gilt, dass die CPU-Baugruppe erneut mit den vorherigen Zuständen (z. B. mit leuchtender BKUP-Anzeige) starten kann, auch wenn die Spannungsversorgung ausgeschaltet wird, bevor der Sicherungsvorgang abgeschlossen ist. Dieser Vorgang kann dann allerdings bis zu einer Minute dauern. Wenn die CPU-Baugruppe für einen längeren Zeitraum ausgeschaltet bleibt, sollten Sie allerdings auch in diesem Fall (und auch bei vorhandener Batterie) immer sicherstellen, dass der Sicherungsvorgang abgeschlossen ist, bevor Sie die Spannungsversorgung ausschalten.

Der für die Datensicherung erforderliche Zeitraum (in dem die BKUP-Anzeige leuchtet) variiert je nach Größe des Anwenderprogramms, wie in der folgenden Tabelle dargestellt.

Größe des Anwenderprogramms	Dauer der Sicherung		
	MONITOR-Betriebsart		PROGRAM-Betriebsart
	Zykluszeit von 0,4 ms (Beispiel)	Zykluszeit von 10,0 ms (Beispiel)	
10 kSteps	2 s	8 s	1 s
60 kSteps	11 s	42 s	6 s
250 kSteps	42 s	170 s	22 s

- Hinweis**
1. Beim Einschalten der CPU-Baugruppe leuchtet die BKUP-Anzeige.
 2. Je nach Art der durchgeführten Online-Bearbeitung kann für die Datensicherung bis zu einer Minute erforderlich sein.

 **Achtung** Das Anwenderprogramm und die Parameterdaten werden von den CS1-H, CJ1-H, CJ1M und CS1D CPU-Baugruppen beim Schreiben in die CPU-Baugruppe automatisch im Flash-Speicher gesichert. Der E/A-Speicher (einschließlich der DM-, EM- und HR-Bereiche) wird jedoch nicht in den Flash-Speicher geschrieben. Die Inhalte der DM-, EM- und HR-Bereiche können während eines Ausfalls der Versorgungsspannung durch eine Batterie erhalten werden. Wenn die Batterie nicht ordnungsgemäß funktioniert, sind die Inhalte dieser Bereiche nach einem Ausfall der Versorgungsspannung evtl. nicht mehr korrekt. Wenn der Inhalt der DM-, EM- und HR-Bereiche zur Steuerung externer Ausgänge verwendet wird, müssen geeignete Maßnahmen zur Vermeidung von falschen Ausgaben ergriffen werden, wenn der Batterie-Fehlermerker (A40204) auf EIN gesetzt ist.

Hinweis Bei der Datensicherung anderer als normaler Datenübertragungen zeigt CX-Programmer in einem Speichersicherungs-Statusfenster einen Sicherungsstatus an (**SPS/Übertragung**). Um dieses Fenster anzuzeigen, müssen Sie in den SPS-Eigenschaften die Anzeige des Dialogfelds für den Sicherungsstatus aktivieren und das Fenster im Menu „Ansicht“ auswählen. Der Sicherungsstatus normaler Übertragungsvorgänge wird im Übertragungsfenster nach dem Übertragungsstatus für das Programm und andere Daten angezeigt.

Merker des Zusatz-Systembereichs

Bezeichnung	Adresse	Bedeutung
Flash-Speicher-Fehlermerker	A40310	EIN, wenn der Flashspeicher defekt ist.

6-6-12 Startbedingungen (Einstellung im SPS-Setup)

Diese Funktion wird nur von den CS1-H, CJ1-H, CJ1M und CS1D CPU-Baugruppen unterstützt.

Einige Baugruppen und interne Module benötigen nach dem Einschalten der Spannungsversorgung eine lange Zeit zum Starten und beeinflussen damit die Startzeit der CPU-Baugruppe. Sie können im SPS-Setup einstellen, dass die CPU-Baugruppe startet, ohne zu warten, bis diese Baugruppen initialisiert werden.

Diese Einstellung gilt für die ITNC-EIS01-CST und ITNC-EIX01-CST Open-Netzwerk-Controller CS1-Bus-Schnittstellen-Baugruppen. (Es werden derzeit keine anderen entsprechenden internen Module angeboten, als vom Oktober 2001.)

Diese Funktion können Sie mit den Einstellungen für die Startbedingungen und internen Module steuern, die in der folgenden Tabelle beschrieben werden.

Startbedingungen	SPS-Setup	
	Startbedingung (Programmierkonsolen- Adresse 83, Bit 15)	Einstellung für interne Module (Programmierkonsolen- Adresse 84, Bit 15)
Zum Start ohne Warten auf alle Baugruppen und Module	1: Betrieb ohne Warten aktiviert.	1: Nicht auf bestimmte interne Module warten.
Zum Start ohne Warten auf alle Baugruppen (Warten auf Module)	1: Betrieb ohne Warten aktiviert.	0: Vor dem Start auf die Module warten.
Vor dem Starten auf die Baugruppen und Module warten	0: Immer auf die Baugruppen/Module warten.	Beliebig

Hinweis CS1 CPU-Baugruppen starten erst, wenn alle Baugruppen und Module die Anfangs-Initialisierung abgeschlossen haben.

SPS-Setup

Adresse der Programmierkonsole		Bezeichnung	Einstellung	Standard-einstellung	CPU-Baugruppen-Aktualisierungszeitpunkt
Wort	Bit				
83	15	Startbedingung	0: Warten auf Baugruppen und Module. 1: Nicht warten.	0: Warten	Beim Einschalten
84	15	Einstellung für internes Modul	0: Auf alle Module warten. 1: Auf bestimmte Module nicht warten.	0: Warten	Beim Einschalten

Startbedingung

0: Wenn bestimmte Module bzw. Baugruppen die Initialisierung noch nicht abgeschlossen haben, wechselt die CPU-Baugruppe in der MONITOR- oder PROGRAM-Betriebsart in den Wartezustand (STANDBY) und wartet auf die entsprechenden Baugruppen und Module.

1: Auch wenn bestimmte Module bzw. Baugruppen die Initialisierung noch nicht abgeschlossen haben, setzt die CPU-Baugruppe ihre Initialisierung fort und startet in der MONITOR- oder PROGRAM-Betriebsart. Das Verfahren für interne Module ist jedoch auch von der folgenden Einstellung abhängig.

Einstellung für internes Modul

Diese Einstellung wird nur verwendet, wenn die Startbedingung auf 1 gesetzt ist und der Start somit ohne Warten auf bestimmte Baugruppen und Module erfolgt. Die Einstellung wird ignoriert, wenn die Startbedingung auf 0 gesetzt ist.

0: Wenn bestimmte Module die Initialisierung noch nicht abgeschlossen haben, wechselt die CPU-Baugruppe in der MONITOR- oder PROGRAM-Betriebsart in den Wartezustand (STANDBY) und wartet auf alle Module.

1: Auch wenn bestimmte Module die Initialisierung noch nicht abgeschlossen haben, setzt die CPU-Baugruppe ihre Initialisierung fort und startet in der MONITOR- oder PROGRAM-Betriebsart.

6-7 Diagnosefunktionen

In diesem Abschnitt wird eine kurze Übersicht über die folgenden Diagnose und Debugging-Funktionen gegeben.

- Fehlerprotokoll
- Ausgang-AUS-Funktion
- Fehleralarmfunktionen (FAL(006) und FALS(007))
- Fehlerpunkt-Erkennungsfunktion (FPD(269))

6-7-1 Fehlerprotokoll

Bei jedem Auftreten eines Fehlers in einer SPS der CS/CJ-Serie speichert die CPU-Baugruppe Fehlerinformationen im Fehlerprotokoll. Zu diesen Informationen gehören der Fehlercode (in A400 gespeichert), Detailinformationen zum Fehler und der Zeitpunkt des Auftretens. Das Fehlerprotokoll kann Informationen zu maximal 20 Fehlern aufnehmen.

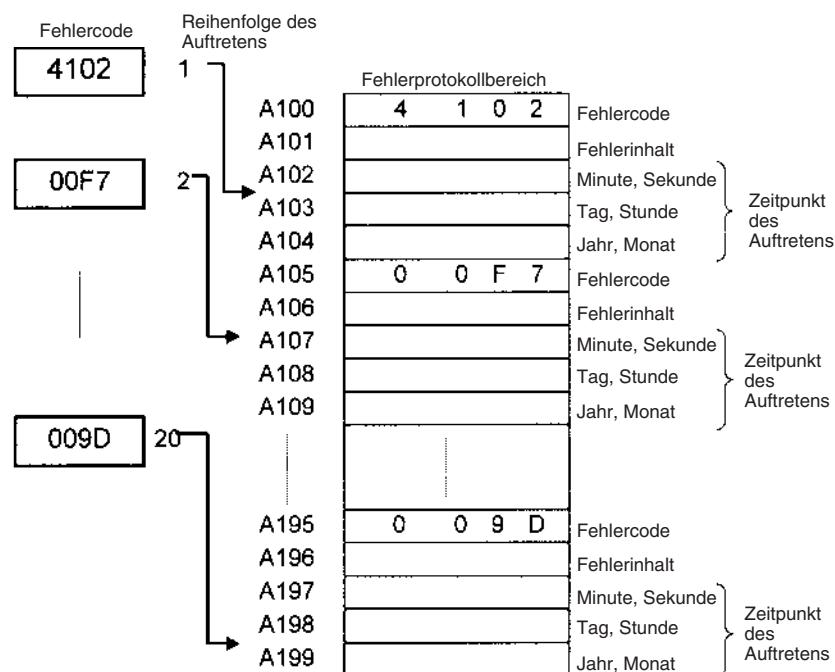
Neben tatsächlichen Systemfehlern zeichnet das SPS-System auch anwenderdefinierte FAL(006)- und FALS(007)-Fehler auf. Auf diese Weise kann der Systemstatus auf einfache Weise protokolliert werden.

Ausführliche Informationen hierzu finden Sie im *CS/CJ-Serie Bedienerhandbuch* im Abschnitt über Fehlerbehebung.

Hinweis

Anwenderdefinierte Fehler treten auf, wenn im Programm einer der Befehle FAL(006) oder FALS(007) ausgeführt wird. Die Ausführungsbedingungen dieser Befehle können somit als anwenderdefinierte Fehlerbedingungen betrachtet werden. FAL(006) erzeugt einen nicht schwerwiegenden Fehler, FALS(007) hingegen einen schwerwiegenden Fehler, der eine Unterbrechung der Programmausführung bewirkt.

Beim Auftreten von mehr als 20 Fehlern werden die ältesten Fehlerdaten (in A100 bis A104) gelöscht, die verbleibenden 19 Datensätze werden um einen Datensatz nach unten verschoben und der neueste Datensatz wird in A195 bis A199 gespeichert.



A300CH

↑ Fehlerprotokollzeiger

Die Anzahl der Datensätze wird im Binärformat im Fehlerprotokollzeiger (A300) gespeichert. Der Zeiger wird nicht inkrementiert, wenn mehr als 20 Fehler aufgetreten sind.

6-7-2 Ausgang-AUS-Funktion

Als Notfallmaßnahme können Sie beim Auftreten eines Fehlers alle Ausgänge von Ausgangs-Baugruppen ausschalten, indem Sie das Ausgang-AUS-Bit (A50015) auf EIN setzen. Die Betriebsart bleibt in der RUN- oder MONITOR-Betriebsart, jedoch werden alle Ausgänge auf AUS gesetzt.

Hinweis Im Normalfall (E/A-Speicher-Haftmerker = AUS) sind nach dem Wechsel der Betriebsart aus der RUN-/MONITOR-Betriebsart in die PROGRAM-Betriebsart alle Ausgänge der Ausgangs-Baugruppen ausgeschaltet. Mit dem Ausgang-AUS-Bit können Sie alle Ausgänge auf AUS setzen, ohne in die PROGRAM-Betriebsart zu wechseln und die Programmausführung zu unterbrechen.

Sicherheitshinweise zum Einsatz von DeviceNet

Wenn Sie bei der CS1W-DRM21 oder CJ1W-DRM21 die Master-Funktion verwenden, werden alle Slave-Ausgänge auf AUS gesetzt. Wenn Sie die Slave-Funktion verwenden, werden alle Eingaben zum Master auf AUS gesetzt. Wenn Sie die C200HW-DRM21-V1 verwenden, werden die Slave-Ausgänge jedoch nicht auf AUS gesetzt.

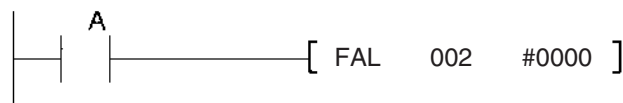
6-7-3 Fehleralarmfunktionen

Mit den Befehlen FAL(006) und FALS(007) können Sie anwenderdefinierte Fehler auslösen. FAL(006) erzeugt einen nicht schwerwiegenden Fehler, FALS(007) hingegen einen schwerwiegenden Fehler, der eine Unterbrechung der Programmausführung bewirkt.

Wenn die anwenderdefinierten Fehlerbedingungen (Ausführungsbedingungen für FAL(006) oder FAL(007)) erfüllt sind, wird der Fehleralarmbefehl ausgeführt und die folgende Verarbeitung durchgeführt.

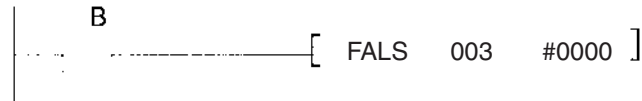
- 1,2,3...**
1. Der FAL-Fehlermerker (A40215) bzw. FALS-Fehlermerker (A40106) wird auf EIN gesetzt.
 2. Der entsprechende Fehlercode wird in A400 geschrieben.
 3. Der Fehlercode und der Zeitpunkt des Auftretens werden im Fehlerprotokoll gespeichert.
 4. Die Fehleranzeige an der Gerätefront blinkt oder leuchtet.
 5. Wenn FAL(006) ausgeführt wurde, führt die CPU-Baugruppe den Betrieb fort. Wenn FALS(007) ausgeführt wurde, unterbricht die CPU-Baugruppe den Betrieb. (Die Ausführung des Programms wird angehalten.)

Funktion von FAL(006)



Wenn die Ausführungsbedingung „A“ auf EIN gesetzt wird, wird ein Fehler mit der FAL-Nummer 2 ausgelöst und A40215 (FAL-Fehlermerker) sowie A36002 (FAL-Nummer-2-Merker) werden auf EIN gesetzt. Das Programm wird nicht angehalten.

Mit FAL(006) ausgelöste Fehler können Sie löschen, indem Sie FAL(006) mit FAL-Nummer 00 ausführen oder mit einem Programmiergerät (oder einer Programmierkonsole) den Lese-/Löschvorgang für Fehler durchführen.

Funktion von FALS(007)

Wenn die Ausführungsbedingung „B“ auf EIN gesetzt wird, wird ein Fehler mit der FALS-Nummer 3 ausgelöst und A40106 (FALS-Fehlermerker) auf EIN gesetzt. Die Programmausführung wird angehalten.

Mit FALS(007) ausgelöste Fehler können Sie löschen, indem Sie die Fehlerursache beheben und mit einem Programmiergerät (oder einer Programmierkonsole) den Lese-/Löschvorgang für Fehler durchführen.

6-7-4 Fehlerpunkt-Erkennung

Mit FPD(269) können Sie Zeitüberwachung und Logikdiagnose durchführen. Die Zeitüberwachungsfunktion löst einen nicht schwerwiegenden Fehler aus, wenn der Diagnoseausgang innerhalb der angegebenen Überwachungszeit nicht auf EIN gesetzt wird. Die Logikdiagnosefunktion gibt an, welche Eingabe verhindert, dass der Diagnoseausgang auf EIN gesetzt wird.

Zeitüberwachungsfunktion

FPD(269) beginnt bei seiner Ausführung eine Zeitmessung und setzt den Übertragsmerker (P_CY) auf EIN, wenn der Diagnoseausgang innerhalb der angegebenen Überwachungszeit nicht auf EIN gesetzt wird. Der Übertragsmerker kann als Ausführungsbedingung für einen Fehlerverarbeitungsblock dienen. Außerdem können Sie FPD(269) so programmieren, dass ein nicht schwerwiegender Fehler mit der gewünschten FAL-Nummer ausgelöst wird.

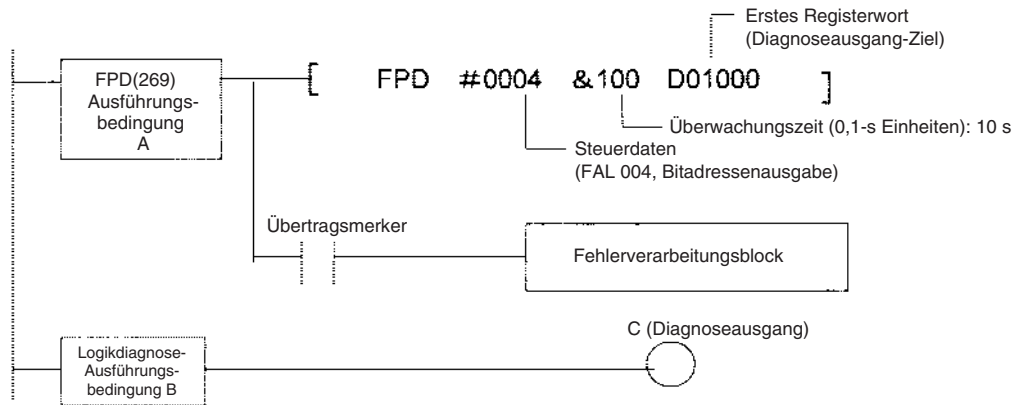
Beim Auslösen eines FAL-Fehlers wird eine vordefinierte Meldung eingetragen, die auf einem Programmiergerät angezeigt werden kann. Sie können mit FPD(269) direkt vor der Meldung das Ergebnis der Logikdiagnose (die Adresse des Bits, das verhindert, dass der Diagnoseausgang auf EIN gesetzt wird) ausgeben.

Mit der Teaching-Funktion können Sie automatisch die Zeit bestimmen, die erforderlich ist, um den Diagnoseausgang auf EIN zu setzen, und damit die Überwachungszeit setzen.

Logikdiagnosefunktion

Mit FPD(269) können Sie bestimmen, welches Eingangsbit verursacht, dass der Diagnoseausgang auf AUS gesetzt bleibt und die Adresse dieses Bits ausgeben. Die Ausgabe kann als Bitadresse (SPS-Speicheradresse) oder als Meldung (ASCII) erfolgen.

- Wenn Sie eine Bitadresse ausgeben lassen, können Sie die SPS-Speicheradresse des Bits in ein Indexregister übertragen und dieses zur indirekten Adressierung bei der weiteren Verarbeitung nutzen.
- Wenn Sie eine Meldung ausgeben lassen, wird die Bitadresse in einer ASCII-Meldung eingetragen, die auf einem Programmiergerät angezeigt werden kann.



Zeitüberwachung:

Es wird überwacht, ob Ausgang „C“ innerhalb von 10 Sekunden nach Eingang „A“ auf EIN gesetzt wird. Wenn „C“ nicht innerhalb von 10 Sekunden auf EIN gesetzt wird, wird ein Fehler erkannt und der Übertragsmerker auf EIN gesetzt. Der Übertragsmerker führt den Fehlerverarbeitungsblock aus. Außerdem wird ein (nicht schwerwiegender) FAL-Fehler mit der FAL-Nummer 004 ausgelöst.

Logikdiagnose:

Mit FPD(269) wird bestimmt, welches Eingangsbit in Block „B“ verhindert, dass der Ausgang „C“ auf EIN gesetzt wird. Diese Bitadresse wird in D01000 und D01001 ausgegeben.

Merker und Worte des Zusatz-Systembereichs

Bezeichnung	Adresse	Funktion
Fehlercode	A400	Beim Auftreten eines Fehlers wird der entsprechende Fehlercode in A400 gespeichert.
FAL-Fehlermerker	A40215	EIN, wenn FAL(006) ausgeführt wird.
FALS-Fehlermerker	A40106	EIN, wenn FALS(007) ausgeführt wird.
FAL-Nummer-Merker	A360 bis A391	Der entsprechende Merker wird auf EIN gesetzt, wenn ein FAL(006)- oder FALS(007)-Fehler auftritt.
Fehlerprotokollbereich	A100 bis A199	Im Fehlerprotokollbereich sind Informationen über die letzten 20 Fehler enthalten.
Fehlerprotokollzeiger	A300	Beim Auftreten eines Fehlers wird der Fehlerprotokollzeiger um 1 inkrementiert und zeigt dann (als Offset vom Anfang des Fehlerprotokollbereichs (A100)) auf die Position, an die der nächste Fehlerdatensatz geschrieben werden kann.
Fehlerprotokollzeiger-Rücksetzbit	A50014	Setzen Sie dieses Bit auf EIN, um den Fehlerprotokollzeiger (A300) auf 00 zurückzusetzen.
FPD-Teaching-Bit	A59800	Setzen Sie dieses Bit auf EIN, wenn die Überwachungszeit bei der Ausführung von FPD(269) automatisch gesetzt werden soll.

6-7-5 Simulieren von Systemfehlern

Diese Funktion wird nur von den CS1-H, CJ1-H, CJ1M und CS1D CPU-Baugruppen unterstützt.

Mit FAL(006) und FALS(007) können Sie absichtlich schwerwiegende und nicht schwerwiegende Fehler erzeugen. Das ist beim System-Debugging nützlich, um Anzeigemeldungen auf programmierbaren Bedienterminals oder anderen Bedieneroberflächen zu testen.

Gehen Sie dazu folgendermaßen vor.

1,2,3...

1. Setzen Sie in A529 die FAL- oder FALS-Nummer, die für die Simulation verwendet werden soll. (A529 wird bei der Fehlersimulation für FAL(006) und FALS(007) verwendet.)

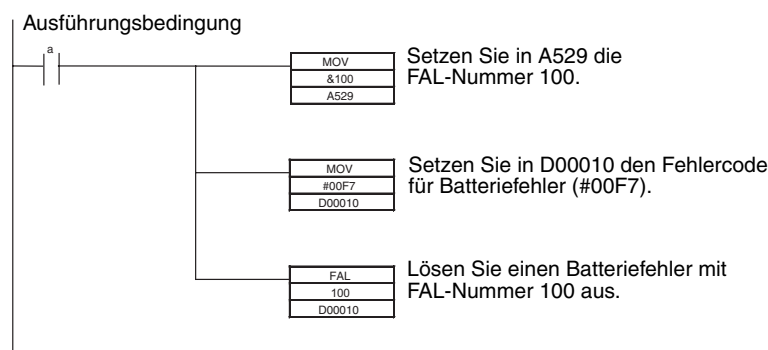
- Setzen Sie die für die Simulation zu verwendende FAL- oder FALS-Nummer als ersten Operand von FAL(006) oder FALS(007).
- Setzen Sie den Fehlercode und zu simulierenden Fehler als zweiten Operanden (zwei Worte) von FAL(006) oder FALS(007). Erzeugen Sie mit FAL(006) einen nicht schwerwiegenden Fehler und mit FALS(007) einen schwerwiegenden Fehler.

Sie können mehrere Systemfehler simulieren, indem Sie mehrere FAL(006)- oder FALS(007)-Befehle wie oben beschrieben verwenden.

Merker und Worte des Zusatz-Systembereichs

Bezeichnung	Adresse	Funktion
FAL-/FALS-Nummer für die Systemfehlersimulation	A529	Setzen Sie eine Dummy-FAL/FALS-Nummer, um den entsprechenden Systemfehler zu simulieren. 0001 bis 01FF Hex: FAL-/FALS-Nummer 1 bis 511 0000 oder 0200 bis FFFF Hex: Keine FAL-/FALS-Nummer für die Systemfehlersimulation

Beispiel für einen Batteriefehler



Hinweis

Sie können simulierte Systemfehler mit denselben Methoden löschen wie tatsächliche Systemfehler. Weitere Informationen hierzu finden Sie im *CS-Serie Bedienerhandbuch* oder im *CJ-Serie Bedienerhandbuch*. Alle mit FAL(006) und FALS(007) simulierten Systemfehler können gelöscht werden, indem Sie die Spannungsversorgung aus- und wieder einschalten.

6-7-6 Deaktivieren der Fehlerprotokollierung für anwenderdefinierte FAL-Fehler

Diese Funktion wird nur von CS1-H, CJ1-H, CJ1M und CS1D CPU-Baugruppen unterstützt.

Die SPS-Einstellungen enthalten eine Einstellung, mit der Sie verhindern können, dass anwenderdefinierte, durch FAL(006) hervorgerufene Fehler und Zeitüberschreitungen bei der Ausführung von FPD(269) im Fehlerprotokoll (A100 bis A199) aufgezeichnet werden. Wenn Sie diese Einstellung verwenden, wird der FAL-Fehler trotzdem ausgelöst und die folgenden Informationen werden ausgegeben: A40215 (FAL-Fehlermerker), A360 bis A391 (verwendete FAL-Nummern) und A400 (Fehlercode).

Diese Funktion können Sie verwenden, wenn im Fehlerprotokoll nur FAL-Systemfehler gespeichert werden müssen, z. B., wenn im Programm viele mit FAL(006) ausgelöste anwenderdefinierte Fehler vorkommen und diese das Fehlerprotokoll zu schnell füllen würden.

SPS-Setup

Adresse der Programmierkonsole		Bezeichnung	Einstellung	Standardeinstellung	CPU-Baugruppen-Aktualisierungszeitpunkt
Wort	Bit				
129	15	Anwender-FAL-Protokollierungseinstellung	0: Anwenderdefinierte FAL-Fehler im Fehlerprotokoll aufzeichnen. 1: Anwenderdefinierte FAL-Fehler nicht im Fehlerprotokoll aufzeichnen.	0: Aufzeichnen	Bei jeder Ausführung des Befehls FAL(006) (in jedem Zyklus)

Hinweis Im Fehlerprotokoll werden die folgenden Einträge gespeichert, auch wenn die Einstellung oben gesetzt ist, um die Aufzeichnung anwenderdefinierter FAL-Fehler zu verhindern.

- Anwenderdefinierte schwerwiegende Fehler (FALS(007))
- Nicht schwerwiegende Systemfehler
- Schwerwiegende Systemfehler
- Anwendersimulierte nicht schwerwiegende Systemfehler (FAL(006))
- Anwendersimulierte schwerwiegende Systemfehler (FALS(007))

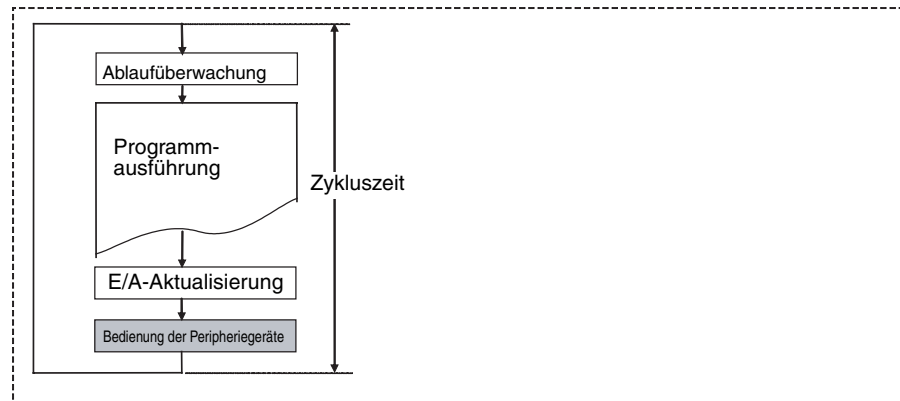
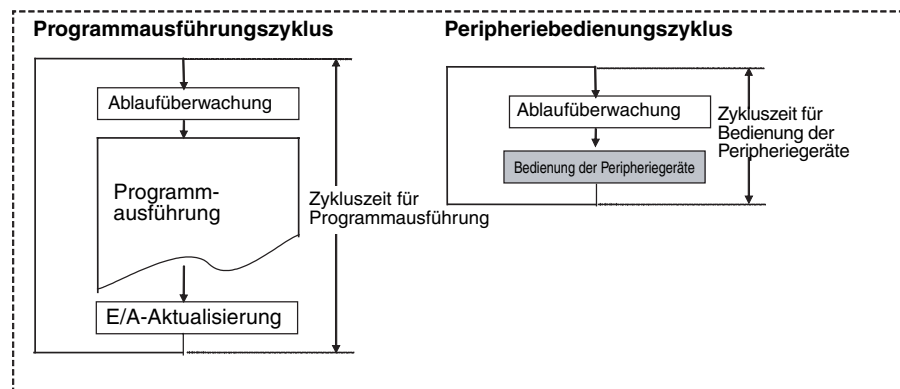
6-8 CPU-Verarbeitungsbetriebsarten

6-8-1 CPU-Verarbeitungsbetriebsarten

Normalerweise wird die Bedienung der Peripheriegeräte (siehe Hinweis) einmal am Ende jedes Zyklus (nach der E/A-Aktualisierung) für 4 % der Zykluszeit oder für eine vom Anwender definierte Zeit durchgeführt. Dadurch ist es unmöglich, Peripheriegeräte mit einer höheren Wiederholungsrate als die der Zykluszeit zu bedienen. Außerdem steigt die Zykluszeit um die für die Bedienung der Peripherie erforderliche Zeit an.

CS1D CPU-Baugruppen für Einzel-CPU-Systeme bzw. CS1-H oder CJ1-H CPU-Baugruppen unterstützen jedoch Parallelverarbeitungs-Betriebsarten, die die parallele Verarbeitung der Programmausführung und der Bedienung der Peripheriegeräte gestatten. Dadurch sind eine schnellere Bedienung der Peripheriegeräte und kürzere Zykluszeiten möglich, besonders wenn eine umfangreiche Bedienung der Peripheriegeräte erforderlich ist. (CS1D CPU-Baugruppen für Duplex-CPU-Systeme und CJ1M CPU-Baugruppen unterstützen keine Parallelverarbeitungs-Betriebsarten.)

Hinweis Zur Bedienung der Peripheriegeräte zählen auch für externe Geräte erforderliche, nicht zeitgesteuerte Dienste, wie das Bedienen von Ereignissen (z. B. bei der Kommunikation durch FINS-Befehle) für Spezial-E/A-Baugruppen, CPU-Bus-Baugruppen und interne Module (nur CS-Serie) sowie das Bedienen von Kommunikationsschnittstellen für die Peripherie- und RS-232C-Schnittstellen (jedoch ohne Data-Links und andere spezielle E/A-Aktualisierungen für CPU-Bus-Baugruppen).

Normale Betriebsart**Parallelverarbeitung****Parallelverarbeitungs-Betriebsarten**

Die Parallelverarbeitung kann auf zweierlei Weise erfolgen: Parallelverarbeitung mit synchronem und mit asynchronem Speicherzugriff.

■ **Parallelverarbeitungs-Betriebsart mit asynchronem Speicherzugriff**

In dieser Betriebsart wird der E/A-Speicherzugriff für die Bedienung der Peripheriegeräte nicht mit dem E/A-Speicherzugriff der Programmausführung synchronisiert. Mit anderen Worten wird die gesamte Bedienung der Peripheriegeräte parallel zur Programmausführung ausgeführt, einschließlich der Speicherzugriffe. Diese Betriebsart bietet bei hoher Belastung durch die Bedienung der Peripheriegeräte die schnellste Ausführung (im Vergleich zu den anderen Betriebsarten) für Programmausführung und Ereignisverarbeitung.

■ **Parallelverarbeitungs-Betriebsart mit synchronem Speicherzugriff**

In dieser Betriebsart werden die E/A-Speicherzugriffe der Bedienung der Peripheriegeräte nicht parallel mit denen der Programmausführung durchgeführt, sondern nach der Programmausführung, d. h., genauso wie in der normalen Ausführungsbetriebsart nach der E/A-Aktualisierung. Die gesamte sonstige Bedienung der Peripheriegeräte wird parallel zum Programm ausgeführt.

Diese Betriebsart bietet für Programmausführung und Ereignisverarbeitung eine schnellere Ausführung als die normale Verarbeitungsbetriebsart. Die Programmausführungs-Zykluszeit ist um die für die E/A-Aktualisierung der Bedienung der Peripheriegeräte erforderliche Zeit länger als die der Parallelverarbeitung mit asynchronem Speicherzugriff.

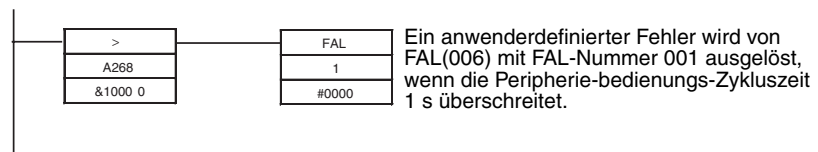
In der folgenden Tabelle werden die Zykluszeit und die Antwortzeit der Bedienung der Peripheriegeräte bei normaler Verarbeitung, Parallelverarbeitung mit asynchronem Speicherzugriff und Parallelverarbeitung mit synchronem Speicherzugriff aufgelistet. (Diese Werte gelten für ein Programm aus Grundbefehlen mit

einer Zykluszeit von 10 ms und bei Verwendung einer Ethernet-Baugruppe. Diese Werte dienen nur als Referenz und variieren je nach System.)

Beschreibung	Normale Betriebsart	Parallelverarbeitungsbetriebsart mit asynchronem Speicherzugriff	Parallelverarbeitungsbetriebsart mit synchronem Speicherzugriff
Zykluszeit	Als Vergleichswert auf 1 gesetzt	0,9	0,9
Bedienung der Peripheriegeräte	Als Vergleichswert auf 1 gesetzt	0,4	1,0

Hinweis

1. Zur Bedienung der Peripheriegeräte zählt auch das Bedienen von Ereignissen (z. B. bei der Kommunikation durch FINS-Befehle) für Spezial-E/A-Baugruppen, CPU-Bus-Baugruppen und interne Module (nur CS-Serie) sowie das Bedienen von Kommunikationsschnittstellen für die Peripherie- und RS-232C-Schnittstellen (jedoch ohne Data-Links und andere spezielle E/A-Aktualisierungen für CPU-Bus-Baugruppen).
2. CS1 CPU-Baugruppen ab Version 1 sowie CS1-H oder CJ1-H CPU-Baugruppen unterstützen auch einen Peripheriebedienungs-Prioritätsmodus, in dem die Bedienung der Peripheriegeräte in einem festen Zyklus während der Programmausführung durchgeführt wird. Dieser bietet eine schnellere Bedienung der Peripheriegeräte als die normale Verarbeitung, jedoch ist die Programmausführung langsamer. Die Reaktion auf Ereignisse erfolgt jedoch nicht so schnell wie in den Parallelverarbeitungsbetriebsarten. Somit sollte die Parallelverarbeitung mit asynchronem Speicherzugriff immer verwendet werden, wenn bei der Verarbeitung das Antworten auf Ereignisse eine hohe Priorität hat.
3. Bei der Parallelverarbeitung können in der CPU-Baugruppe die im Folgenden unter a) und b) beschriebenen Peripheriebedienungs-Zykluszeit-Überschreitungsfehler auftreten. Wenn dieser Fehler auftritt, zeigt das Programmiergerät an, dass die Zykluszeit zu lang ist, der Peripheriebedienungszykluszeit-Überschreitungsmerker (A40515) wird auf EIN gesetzt und der Vorgang wird angehalten (schwerwiegender Fehler).
 - a) Wenn die Peripheriebedienungs-Zykluszeit 2,0 s überschreitet, tritt ein Zykluszeit-Überschreitungsfehler auf. Sie können die Peripheriebedienungs-Zykluszeit in A268 überwachen, um mögliche Fehler vor dem Auftreten zu erkennen. Beispielsweise können Sie einen anwender-definierten Fehler mit der FAL-Nummer 001 auslösen, wenn die Peripheriebedienungs-Zykluszeit 1 s überschreitet (d. h., wenn der Inhalt von A268 2710 Hex (10000 dezimal) überschreitet).



- b) Ein Peripheriebedienungs-Zykluszeit-Überschreitungsfehler kann auch auftreten, wenn die Verarbeitungszeit für den Programmausführungszyklus (d. h., die Programmbefehle-Ausführungszeit) zu kurz ist. Diese Zeit wird bei normaler Ausführungsbetriebsart in A266 und A267 gespeichert. Gehen Sie als Richtlinie davon aus, dass bei einer Befehlsausführungszeit von 2 ms oder weniger ein Peripheriebedienungs-Zykluszeit-Überschreitungsfehler auftritt und die Parallelverarbeitungsbetriebsart nicht verwendet werden kann. Wenn Sie beim Debugging nur Abschnitte des Programms testen (dadurch kann es zu sehr kurzen Programmausführungszeiten kommen), verwenden Sie die normale Betriebsart, um diesen Fehler zu vermeiden.

Unterbrechen Sie die Verbindung zur Programmierkonsole, wenn Sie anwenderdefinierte Anwendungen in einer Parallelverarbeitungsbetriebsart ausführen. Der Programmierkonsole wird Bedienungszeit zugeordnet, um die Antwortgeschwindigkeit auf Tastatureingaben bei der Programmierkonsole zu

erhöhen. Dadurch wird die Peripheriebedienungszeit erhöht und die Effektivität der Parallelverarbeitung reduziert.

SPS-Einstellungen

Die Verarbeitungsbetriebsart wird in den SPS-Einstellungen spezifiziert.

Adresse der Programmierkonsole		Bezeichnung	Einstellung	Standardeinstellung	CPU-Baugruppen-Aktualisierungszeitpunkt
Wort	Bit				
219	08 bis 15	CPU-Verarbeitungsbetriebsart	00 Hex: Normale Betriebsart 01 Hex: Parallelverarbeitungsbetriebsart mit synchronem Speicherzugriff 02 Hex: Parallelverarbeitungsbetriebsart mit asynchronem Speicherzugriff 05 bis FF Hex: Zeitabschnitt-Programmausführungszeit für den Peripheriebedienungs-Prioritätsmodus (5 bis 255 ms in 1-ms-Schritten) Die Einstellungen 03 und 04 Hex sind nicht definiert (unzulässig) und verursachen einen (nicht schwerwiegenden) SPS-Einstellungs-Fehler.	00 Hex: Normale Betriebsart	Zu Beginn der Ausführung

Merker und Worte des Zusatz-Systembereichs

Bezeichnung	Adresse	Funktion
Peripheriebedienungs-Zykluszeit-überschreitung	A40515	Wird auf EIN gesetzt, wenn die Peripheriebedienungs-Zykluszeit länger als zwei Sekunden dauert. Der Betrieb wird angehalten.
Peripheriebedienungs-Zykluszeit	A268	Enthält die Peripheriebedienungs-Zykluszeit, wenn Sie eine der Parallelverarbeitungsbetriebsarten (mit synchronem oder asynchronem Speicherzugriff) verwenden und sich die SPS in der RUN- oder MONITOR-Betriebsart befindet. Die Zeit wird im Binärformat angegeben und liegt zwischen 0,0 und 2000,0 (in 0,1-ms-Schritten).
Ausführungszeit (Die Summe aller Zeitabschnitte für die Programmausführung und aller Zeitabschnitte für die Bedienung der Peripheriegeräte.)	A266 und A267	In der normalen Betriebsart ist nur die Programmbefehle-Ausführungszeit enthalten. Die Zeit wird als Wert im 32-Bit-Binärformat gespeichert. 00000000 bis FFFFFFFF Hex (in 0,1-ms-Schritten) (0 bis 429.496.729,5 ms) A266: Niederwertigstes Wort A267: Höchstwertigstes Wort

Parallelverarbeitungsbetriebsart mit asynchronem Speicherzugriff

Programmausführungen

Ablaufüberwachung		Überprüfung von E/A-Bus und sonstige Verarbeitung 0,3 ms
Programmbefehle-Ausführungszeit		Gesamtausführungsdauer aller Befehle
Berechnungen der minimalen Zykluszeit		Die Verarbeitungszeit für eine minimale Programmausführungs-Zykluszeit.
Zyklische E/A-Aktualisierung	E/A-Aktualisierung	E/A-Aktualisierungszeit für jede Baugruppe x Anzahl der Baugruppen
	Spezielle E/A-Aktualisierung bei CPU-Bus-Baugruppen	Spezielle E/A-Aktualisierungszeit für jede Baugruppe x Anzahl der Baugruppen
Bedienung der Peripheriegeräte	Dateizugriff	Im SPS-Setup eingestellte Peripheriegeräte-Bedienungszeit (Standardeinstellung: 4 % der Zykluszeit)

Bedienung der Peripheriegeräte

Ablaufüberwachung		Batterieprüfung, Anwenderprogrammspeicherprüfung usw. 0,2 ms
Bedienung der Peripheriegeräte	Ereignisbedienung bei Spezial-E/A-Baugruppen	Enthält die Ereignisbedienung für den Zugriff auf den E/A-Speicher (siehe Hinweis). Max. eine Sekunde pro Dienst.
	Ereignisbedienung bei CPU-Bus-Baugruppen	
	Bedienung der Peripherieschnittstelle	
	Bedienung der RS-232C-Schnittstelle	
	Ereignisbedienung für internes Modul (nur CS-Serie)	
	Ereignisbedienung für Kommunikationsschnittstellen (interne logische Schnittstellen), die verwendet werden (einschließlich Befehlsausführung im Hintergrund)	

Hinweis Zur Ereignisbedienung für den Zugriff auf den E/A-Speicher zählen: 1.) Die Bedienung aller empfangenen FINS-Befehle, die auf den E/A-Speicher zugreifen (E/A-Speicher-Lese-/Schreibbefehle mit allgemeinen Codes, die mit 01 Hex anfangen oder Befehle zum Veranlassen einer Zwangssetzung oder -rücksetzung mit allgemeinen Codes, die mit 23 Hex anfangen) und 2.) Die Bedienung aller empfangenen C-Mode-Befehle, die auf den E/A-Speicher zugreifen (ohne NT-Links, die die Peripherie- oder RS-232C-Schnittstelle verwenden).

Parallelverarbeitungs-Betriebsart mit synchronem Speicherzugriff**Programmausführungen**

Ablaufüberwachung		Überprüfung von E/A-Bus und sonstige Verarbeitung 0,3 ms
Programmbefehle-Ausführungszeit		Gesamtausführungsdauer aller Befehle
Berechnungen der minimalen Zykluszeit		Die Verarbeitungszeit für eine minimale Programmausführungs-Zykluszeit.
Zyklische E/A-Aktualisierung	E/A-Aktualisierung	E/A-Aktualisierungszeit für jede Baugruppe x Anzahl der Baugruppen
	Spezielle E/A-Aktualisierung bei CPU-Bus-Baugruppen	Spezielle E/A-Aktualisierungszeit für jede Baugruppe x Anzahl der Baugruppen
Bedienung der Peripheriegeräte	Dateizugriff	Im SPS-Setup eingestellte Peripheriegeräte-Bedienungszeit (Standardeinstellung: 4 % der Zykluszeit)
	Ereignisbedienung, die E/A-Speicherzugriff erfordert (siehe Hinweis).	

Bedienung der Peripheriegeräte

Ablaufüberwachung		Batterieprüfung, Anwenderprogrammspeicherprüfung usw. 0,2 ms
Bedienung der Peripheriegeräte	Ereignisbedienung bei Spezial-E/A-Baugruppen	Mit Ausnahme der Ereignisbedienung für den Zugriff auf den E/A-Speicher (siehe Hinweis). Max. eine Sekunde pro Dienst.
	Ereignisbedienung bei CPU-Bus-Baugruppen	
	Bedienung der Peripherieschnittstelle	
	Bedienung der RS-232C-Schnittstelle	
	Ereignisbedienung für internes Modul (nur CS-Serie)	
	Ereignisbedienung für Kommunikationsschnittstellen (interne logische Schnittstellen), die verwendet werden (einschließlich Befehlsausführung im Hintergrund)	

Hinweis Zur Ereignisbedienung für den Zugriff auf den E/A-Speicher zählen: 1.) Die Bedienung aller empfangenen FINS-Befehle, die auf den E/A-Speicher zugreifen (E/A-Speicher-Lese-/Schreibbefehle mit allgemeinen Codes, die mit 01 Hex anfangen oder Befehle zum Veranlassen einer Zwangssetzung oder -rücksetzung mit allgemeinen Codes, die mit 23 Hex anfangen) und 2.) Die Bedienung aller empfangenen C-Mode-Befehle, die auf den E/A-Speicher zugreifen (ohne NT-Links, die die Peripherie- oder RS-232C-Schnittstelle verwenden).

zung mit allgemeinen Codes, die mit 23 Hex anfangen) und 2.) Die Bedienung aller empfangenen C-Mode-Befehle, die auf den E/A-Speicher zugreifen (ohne NT-Links, die die Peripherie- oder RS-232C-Schnittstelle verwenden).

6-8-2 Parallelverarbeitungs-Betriebsart und minimale Zykluszeiten

Wenn bei der Verwendung einer Parallelverarbeitungs-Betriebsart eine minimale Zykluszeit spezifiziert ist, wird nach der Programmausführung eine Verzögerung eingefügt, bis die minimale Zykluszeit erreicht ist. Die Bedienung der Peripheriegeräte wird jedoch fortgesetzt.

6-8-3 Datenübereinstimmung bei Parallelverarbeitung mit asynchronem Speicherzugriff

Die Daten stimmen in den folgenden Fällen möglicherweise nicht überein, wenn Sie die Parallelverarbeitung mit asynchronem Speicherzugriff verwenden.

- Wenn mit einem Kommunikationsbefehl mehrere Worte aus dem E/A-Speicher gelesen werden, stimmen die Daten in den Worten möglicherweise nicht mehr mit den gelieferten Daten überein.
- Wenn ein Befehl mehrere Worte aus dem E/A-Speicher liest und während der Ausführung der Befehle eine Bedienung der Peripheriegeräte ausgeführt wird, stimmen die Daten in den Worten möglicherweise nicht mehr mit den gelieferten Daten überein.
- Wenn dasselbe Wort des E/A-Speichers mit mehreren Befehlen von verschiedenen Stellen des Programms gelesen wird und während der Ausführung der Befehle eine Bedienung der Peripheriegeräte ausgeführt wird, stimmen die Daten in den Worten möglicherweise nicht mehr mit den gelieferten Daten überein.

Mit den folgenden Schritten können Sie die Datenübereinstimmung gegebenenfalls sicherstellen.

1. Verwenden Sie die Parallelverarbeitungs-Betriebsart mit synchronem Speicherzugriff
2. Deaktivieren Sie an den erforderlichen Stellen des Programms die Bedienung der Peripheriegeräte mit IOSP(287) und aktivieren Sie diese später wieder mit IORS(288).

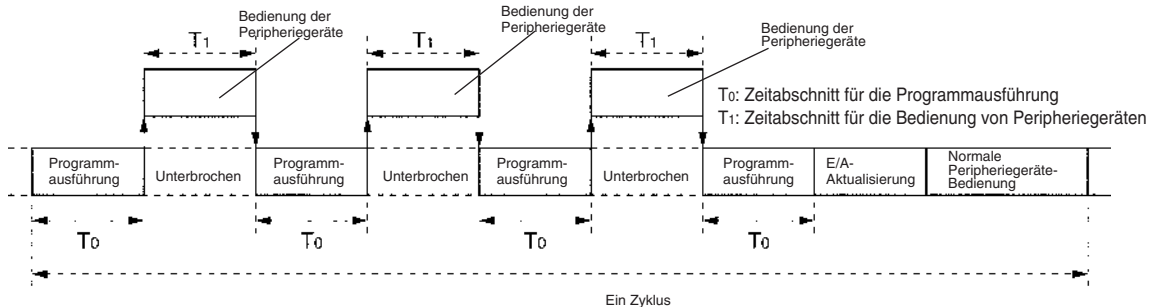
6-9 Peripheriebedienungs-Prioritätsmodus

Die Bedienung der Peripherie für die RS-232C-Schnittstelle, die Peripherieschnittstelle, das interne Modul (nur CS-Serie), CPU-Bus-Baugruppen und Spezial-E/A-Baugruppen wird normalerweise nur einmal am Ende des Zyklus nach der E/A-Aktualisierung ausgeführt. Dafür werden entweder 4 % der Zykluszeit zugewiesen, oder jedem Dienst wird eine vom Anwender definierte Zeit zugewiesen. Es steht jedoch ein Modus zur Verfügung, in dem eine periodische Bedienung innerhalb eines Zyklus möglich ist. Sie können diesen Modus, den Peripheriebedienungs-Prioritätsmodus, in den SPS-Einstellungen einstellen.

Hinweis Sie können den Peripheriebedienungs-Prioritätsmodus bei CPU-Baugruppen der CJ-Serie oder CS-Serie verwenden, jedoch muss die CS1 CPU-Baugruppe am oder nach dem 1. Dezember 2000 hergestellt worden sein (Lot-Nummer 001201□□□□ oder später). (Der Peripheriebedienungs-Prioritätsmodus wird von CS1D CPU-Baugruppen für Duplex-CPU-Systeme nicht unterstützt.)

6-9-1 Peripheriebedienungs-Prioritätsmodus

Wenn der Peripheriebedienungs-Prioritätsmodus gesetzt ist, wird die Programmausführung am spezifizierten Zeitpunkt für die Durchführung der spezifizierten Bedienung unterbrochen und anschließend wieder fortgesetzt. Dies findet während der gesamten Programmausführung wiederholt statt. Die normale Bedienung der Peripheriegeräte wird weiterhin nach der E/A-Aktualisierung durchgeführt.



Sie können den Peripheriebedienungs-Prioritätsmodus somit verwenden, um zusätzlich zur normalen Bedienung der Peripheriegeräte eine periodische Bedienung für ausgewählte Schnittstellen oder Baugruppen auszuführen. Dadurch können Sie Anwendungen programmieren, die eine Priorisierung der Bedienung der Peripheriegeräte gegenüber der Programmausführung erfordern, z. B. Prozesssteuerungsanwendungen, in denen für die Host-Überwachung schnelle Antworten erforderlich sind.

- Für die Peripheriebedienung mit Priorität können bis zu fünf Baugruppen oder Schnittstellen spezifiziert werden. Die CPU-Bus-Baugruppen und CS/CJ Spezial-E/A-Baugruppen werden anhand der Baugruppennummer spezifiziert.
- In jedem Zeitabschnitt für die Bedienung der Peripheriegeräte wird nur eine Baugruppe oder Schnittstelle bedient. Wenn die Bedienung abgeschlossen ist, bevor die spezifizierte Zeit abgelaufen ist, wird die Programmausführung sofort fortgesetzt und die nächste Baugruppe oder Schnittstelle wird erst im nächsten Zeitabschnitt für die Bedienung der Peripheriegeräte bedient. Jedoch ist es möglich, dass im selben Zyklus dieselbe Baugruppe oder Schnittstelle mehrfach bedient wird.
- Die Baugruppen oder Schnittstellen werden in der Reihenfolge bedient, in der sie von der CPU-Baugruppe erkannt werden.

Hinweis

1. Obwohl die folgenden Befehle die Kommunikationsschnittstellen verwenden, werden sie, auch im Peripheriebedienungs-Prioritätsmodus, nur einmal pro Ausführungszyklus ausgeführt:
RXD(235) (RECEIVE)
TXD(236) (TRANSMIT)
2. Wenn mit einem Kommunikationsbefehl mehrere Worte gelesen werden, kann die Übereinstimmung der gelesenen Daten im Peripheriebedienungs-Prioritätsmodus nicht gewährleistet werden.
3. Die CPU-Baugruppe überschreitet bei Verwendung des Peripheriebedienungs-Prioritätsmodus möglicherweise die maximale Zykluszeit. Sie können die maximale Zykluszeit im SPS-Setup als Einstellung der Watchdog-Zykluszeit einstellen. Wenn die Zykluszeit die Watchdog-Zykluszeit überschreitet, wird der Zykluszeit-zu-lang-Merker (A40108) auf EIN gesetzt und der Betrieb des SPS-Systems angehalten. Im Peripheriebedienungs-Prioritätsmodus sollten Sie die aktuelle Zykluszeit in A264 und A265 überwachen und die Watchdog-Zykluszeit (Adresse: +209) wie erforderlich anpassen. (Die Einstellung ist im Bereich von 10 bis 40.000 ms in 10-ms-Schritten einstellbar. Die Standardeinstellung ist 1 s.)

SPS-Einstellungen

Sie müssen im SPS-Setup die folgenden Einstellungen vornehmen, um den Peripheriebedienungs-Prioritätsmodus verwenden zu können.

- Zeitabschnitt für die Programmausführung (Befehlsausführungszeit): 5 bis 255 ms (in 1-ms-Schritten)
- Zeitabschnitt für die Bedienung von Peripheriegeräten (Ausführungszeit Peripherie-Service): 0,1 bis 25,5 ms (in 0,1-ms-Schritten)
- Baugruppen und/oder Schnittstellen für die Bedienung mit Priorität:
CPU-Bus-Baugruppen (nach Baugruppennr.)
CS/CJ Spezial-E/A-Baugruppen (nach Baugruppennr.)
Internes Modul (nur CS-Serie)
RS-232C-Schnittstelle
Peripherieschnittstelle

Adresse in der Programmierkonsole		Einstellungen	Standard-einstellung	Funktion	Inkraft-treten der neuen Einstellung
Wort	Bit(s)				
219	08 bis 15	00 05 bis FF (Hex)	00	00: Prioritätsbedienungsmodus deaktiviert 05 bis FF: Zeitabschnitt für die Ausführung von Befehlen (5 bis 255 ms in 1-ms-Schritten)	Wird beim Start des Betriebs wirksam. (Diese Einstellung kann während des laufenden Betriebs nicht geändert werden.)
	00 bis 07	00 bis FF (Hex)	00	00: Prioritätsbedienungsmodus deaktiviert 01 bis FF: Zeitabschnitt für die Bedienung von Peripheriegeräten (0,1 bis 25,5 ms in 0,1-ms-Schritten)	
220	08 bis 15	00 10 bis 1F 20 bis 2F E1 FC FD (Hex)	00	00: Prioritätsbedienungsmodus deaktiviert 10 bis 1F: Baugruppennummer der CPU-Bus-Baugruppe + 10 (Hex)	
	00 bis 07		00	20 bis 7F: Baugruppennummer der CS/CJ Spezial-E/A-Baugruppe + 20 (Hex)	
221	08 bis 15		00	E1: Internes Modul	
	00 bis 07		00	FC: RS-232C-Schnittstelle	
222	08 bis 15		00	FD: Peripherieschnittstelle	

- Betrieb und Fehler variieren je nach SPS-Einstellungen (siehe unten).
- Für CS1 oder CJ1 CPU-Baugruppen können Sie die Einstellungen nicht mit CX-Programmer vornehmen. Für CS1-H und CJ1-H CPU-Baugruppen können Sie die Einstellungen mit CX-Programmer ab Version 2.1 vornehmen.

Bedingungen			Betrieb der CPU-Baugruppe	SPS-Einstellungs-Fehler
Zeitabschnitt für die Bedienung von Peripheriegeräten	Zeitabschnitt für die Befehlsausführung	Spezifizierte Baugruppen und Schnittstellen		
01 bis FF: (0,1 bis 25,5 ms)	05 bis FF: (5 bis 255 ms)	Alle korrekten Einstellungen	Peripheriebedienungs-Prioritätsmodus	Keine
		00 und korrekte Einstellungen		
		Korrekte, jedoch redundante Einstellungen		
		Einige ungültige Einstellungen	Peripheriebedienungs-Prioritätsmodus für Elemente mit korrekten Einstellungen	Ausgelöst
		Alle Einstellungen 00	Normaler Betrieb	Ausgelöst
		00 und ungültige Einstellungen		
		Alle Einstellungen ungültig		
00	00	---	Normaler Betrieb	Keine
Alle sonstigen		---	Normaler Betrieb	Ausgelöst

Hinweis Wenn in den SPS-Einstellungen ein Fehler erkannt wurde, wird A40210 auf EIN gesetzt, und ein nicht schwerwiegender Fehler tritt auf.

Informationen im Zusatz-Systembereich Wenn die Zeiten der Zeitabschnitte für die Programmausführung und die Bedienung der Peripheriegeräte eingestellt sind, werden alle diesbezüglichen Abschnittszeiten in A266 und A267 gespeichert. Diese Informationen können zur entsprechenden Feinanpassung der Zeitabschnitte verwendet werden.

Wenn Sie nicht im Peripheriebedienungs-Prioritätsmodus arbeiten, wird die Programmausführungszeit gespeichert. Mit diesem Wert können die geeigneten Einstellungen für die Zeitabschnitte ermittelt werden.

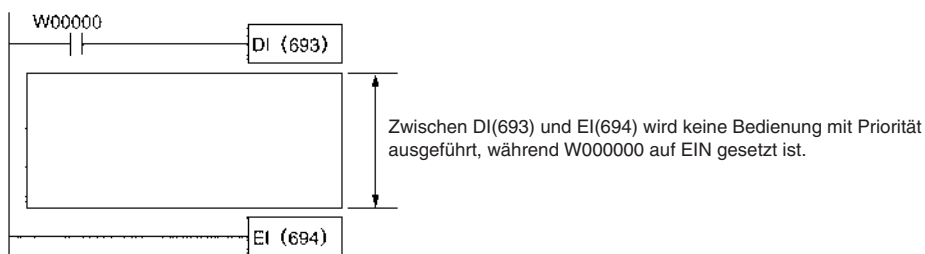
Worte	Inhalt	Bedeutung	Aktualisierung
A266 und A267	00000000 bis FFFFFFFF Hex (0 bis 4.294.967.295 dezimal)	Die Summe aller Zeitabschnitte für die Programmausführung und aller Zeitabschnitte für die Bedienung der Peripheriegeräte. 0,0 bis 429.496.729,5 ms (0,1-ms-Schritte) A267 (Höchstwertige Bytes) A266 (Niederwertigste Bytes) Der Wert wird als 32-Bit-Binärwert (8-stellig hexadezimal) gespeichert.	Der Inhalt wird in jedem Zyklus aktualisiert und zu Beginn des Betriebs gelöscht.

6-9-2 Vorübergehendes Deaktivieren der Peripheriebedienungs-Priorität

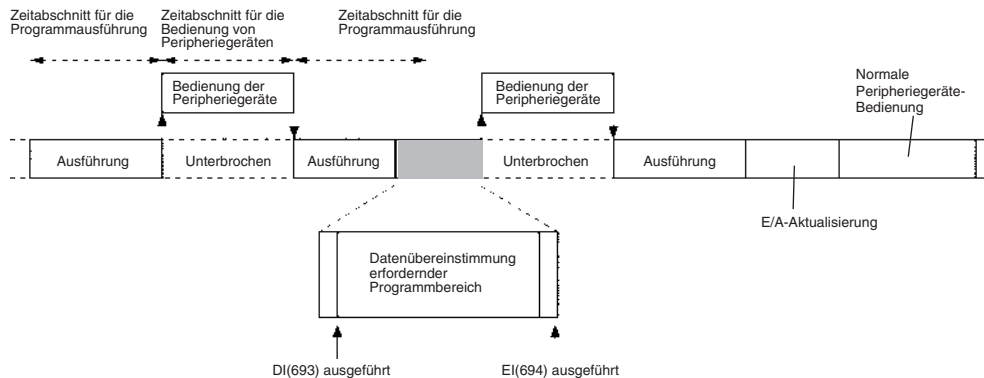
Die Datenübereinstimmung ist in folgenden Fällen nicht gewährleistet, wenn der Peripheriebedienungs-Prioritätsmodus verwendet wird.

- Wenn mit einem Kommunikationsbefehl mehr als ein Wort von einem Peripheriegerät gelesen wird. Die Daten werden möglicherweise in verschiedenen Peripheriegerätebedienungs-Zeitabschnitten ausgelesen und können dadurch möglicherweise nicht mehr übereinstimmen.
- Wenn Befehle mit langen Ausführungszeiten im Programm verwendet werden, z. B., wenn viele E/A-Speicherdaten übertragen werden. Die Übertragung wird möglicherweise für die Bedienung der Peripheriegeräte unterbrochen und dadurch können die Daten möglicherweise nicht mehr übereinstimmen. Dies kann dann auftreten, wenn Worte, die vom Programm geschrieben werden, von einem Peripheriegerät gelesen werden, bevor der Schreibvorgang abgeschlossen ist, oder wenn Worte, die vom Programm gelesen werden, von einem Peripheriegerät geschrieben werden, bevor der Lesevorgang abgeschlossen ist.
- Wenn zwei Befehle auf die gleichen Worte im Speicher zugreifen. Wenn diese Worte zwischen den Zeitabschnitten, in denen die beiden Befehle ausgeführt werden, von einem Peripheriegerät geschrieben werden, lesen die Befehle unterschiedliche Werte aus dem Speicher.

Wenn die Datenübereinstimmung gewährleistet sein muss, kann die Bedienung mit Priorität bei CS1 oder CJ1 CPU-Baugruppen mit den Befehlen DISABLE INTERRUPTS und ENABLE INTERRUPTS (DI(693) und EI(694)) in den erforderlichen Programmabschnitten verhindert werden, wie im folgenden Beispiel dargestellt. Bei CS1D CPU-Baugruppen für Einzel-CPU-Systeme und CS1-H, CJ1-H oder CJ1M CPU-Baugruppen können die Befehle DISABLE PERIPHERAL SERVICING und ENABLE PERIPHERAL SERVICING (IOSP(287) und IORS(288)) verwendet werden.



Funktionsweise



Hinweis

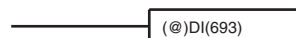
1. Mit den Befehlen DI(693) und IOSP(287) werden nicht nur die Interrupts für die Bedienung mit Priorität deaktiviert, sondern auch alle anderen Interrupts, einschließlich E/A-, zeitgesteuerte und externe Interrupts. Alle erzeugten Interrupts werden ausgeführt, nachdem die zyklische Task ausgeführt wurde (nach der Ausführung des END(001)-Befehls), es sei denn, es wird zuerst der Befehl CLI(691) zum Löschen der Interrupts ausgeführt.
2. Die Deaktivierung der Interrupts mit DI(693) oder IOSP(287) ist so lange gültig, bis die Befehle EI(694) bzw. IORS(288) oder END(001) ausgeführt werden oder der SPS-Betrieb angehalten wird. Es können daher keine Programmabschnitte erstellt werden, die über das Ende einer Task oder eines Zyklus hinausgehen. Verwenden Sie bei Bedarf in jeder zyklischen Task die Befehle DI(693) und EI(694) bzw. IOSP(287) und IORS(288), um die Interrupts in mehreren Zyklen oder Tasks zu deaktivieren.

CS1 und CJ1 CPU-Baugruppen

DI(693)

Bei der Ausführung von DI(693) werden alle Interrupts deaktiviert (außer den Interrupts für die Ausschalt-Interrupt-Task), einschließlich der Interrupts für Bedienung mit Priorität, der E/A-Interrupts, der zeitgesteuerten und externen Interrupts. Bereits deaktivierte Interrupts bleiben deaktiviert, wenn der Befehl DI(693) ausgeführt wird.

Symbol



Anwendbarkeit auf Programmbereiche

Bereich	Anwendbar
Blockprogrammbereiche	Ja
Step-Programmbereiche	Ja
Unterprogramme	Ja
Interrupt-Tasks	Nein

Bedingungsmerker

Merker	Kürzel	Funktion
Fehlermerker	P_ER	Wird auf EIN gesetzt, wenn DI(693) in einer Interrupt-Task ausgeführt wird, andernfalls AUS.

EI(694)

Bei der Ausführung von EI(694) werden alle Interrupts aktiviert (außer den Interrupts für die Ausschalt-Interrupt-Task), einschließlich der Interrupts für Bedienung mit Priorität, der E/A-Interrupts, der zeitgesteuerten und externen Interrupts. Bereits aktivierte Interrupts bleiben aktiviert, wenn der Befehl EI(694) ausgeführt wird.

Symbol



Anwendbarkeit auf Programmbereiche

Bereich	Anwendbar
Blockprogrammbereiche	Ja
Step-Programmbereiche	Ja
Unterprogramme	Ja
Interrupt-Tasks	Nein

Bedingungsmerker

Merker	Kürzel	Funktion
Fehlermerker	P_ER	Wird auf EIN gesetzt, wenn EI(694) in einer Interrupt-Task ausgeführt wird.

CS1D CPU-Baugruppen für Einzel-CPU-Systeme sowie CS1-H, CJ1-H und CJ1M CPU-Baugruppen

IOSP(287)

Der Befehl IOSP(287) deaktiviert die Bedienung der Peripheriegeräte. Die bereits deaktivierte Bedienung der Peripheriegeräte bleibt deaktiviert, wenn der Befehl IOSP(287) ausgeführt wird.

Symbol



Anwendbarkeit auf Programmbereiche

Bereich	Anwendbar
Blockprogrammbereiche	Ja
Step-Programmbereiche	Ja
Unterprogramme	Ja
Interrupt-Tasks	Nein

Bedingungsmerker

Merker	Kürzel	Funktion
Fehlermerker	P_ER	Wird auf EIN gesetzt, wenn IOSP(287) in einer Interrupt-Task ausgeführt wird, andernfalls AUS.

IORS(288)

Mit dem Befehl IORS(288) wird die Bedienung der Peripheriegeräte aktiviert, die mit IOSP(287) deaktiviert wurde. Die bereits aktivierte Bedienung der Peripheriegeräte bleibt aktiviert, wenn der Befehl IORS(288) ausgeführt wird.

Symbol



Anwendbarkeit auf Programmbereiche

Bereich	Anwendbar
Blockprogrammbereiche	Ja
Step-Programmbereiche	Ja
Unterprogramme	Ja
Interrupt-Tasks	Nein

Bedingungsmerker

Merker	Kürzel	Funktion
Fehlermerker	P_ER	Wird auf EIN gesetzt, wenn IORS(288) in einer Interrupt-Task ausgeführt wird.

6-10 Betrieb ohne Batterie

SPS-Systeme der CS-Serie und CJ-Serie können auch ohne Batterie (oder mit einer leeren Batterie) betrieben werden. Die Vorgehensweisen für den Betrieb ohne Batterie sind von den folgenden Bedingungen abhängig.

- CPU-Baugruppe
- Ob der Inhalt des E/A-Speichers (z. B. CIO-Bereich) erhalten bleiben soll.
- Ob die DM- und EM-Bereiche beim Startvorgang initialisiert werden sollen.
- Ob die DM- und EM-Bereiche vom Anwenderprogramm initialisiert werden sollen.

Die beschriebenen Unterschiede sind in der folgenden Tabelle zusammengefasst.

CPU-Bau- gruppe	Kein Erhalten des E/A-Speichers			Erhalten des E/A-Speichers
	Keine Initialisierung der DM- und EM- Bereiche beim Start	Initialisierung der DM- und EM-Bereiche beim Start		
		Vom Anwenderprogramm	Nicht vom Anwenderprogramm	
CS1-H, CJ1-H, CJ1M oder CS1D	Verwenden Sie den normalen Vorgang (mit Flash-Speicher) oder eine Speicherkarte.		Verwenden Sie die automatische Übertragung von einer Speicherkarte während des Startvorgangs. (Stellen Sie Schalter 2 des DIP-Schalterblocks auf EIN.)	Mit keiner Methode möglich. Es muss eine Batterie eingesetzt sein.
CS1 oder CJ1	Verwenden Sie die automatische Übertragung von einer Speicherkarte während des Startvorgangs. (Stellen Sie Schalter 2 des DIP-Schalterblocks auf EIN.)			

Hinweis

1. Wenn Sie den Betrieb ohne Batterie nutzen, sollten Sie in den SPS-Einstellungen die Erkennung des niedrigen Batterieladezustands deaktivieren, unabhängig davon, welche Methode Sie verwenden.
2. Wenn keine Batterie eingelegt ist oder die Batterie leer ist, gelten die folgenden Einschränkungen für den Betrieb der CPU-Baugruppe. Diese Einschränkungen gelten für alle CPU-Baugruppen.
 - Der Status des Ausgang-AUS-Bits (A50015) ist unzuverlässig. Wenn das Ausgang-AUS-Bit auf EIN gesetzt wird, werden die Ausgänge aller Ausgangs-Baugruppen auf AUS geschaltet. Fügen Sie im SPS-Programm folgende Befehle ein, um beim Einschalten des Gerätes zu verhindern, dass die Ausgänge aller Ausgangs-Baugruppen auf AUS geschaltet werden.

Erster-Zyklus-Merker
(A20011)



- Der Inhalt des E/A-Speichers (einschließlich der HR-, DM- und EM-Bereiche) kann nicht ordnungsgemäß erhalten werden. Stellen Sie das SPS-Setup so ein, dass der Status des E/A-Speicher-Haftmerkers (A50012) und Zwangssetzungs-Haftmerkers (A50013) nicht erhalten wird, wenn das Gerät eingeschaltet wird.
- Die Uhrfunktion kann nicht verwendet werden. Die Uhrdaten in A351 bis A354 und die Startzeit in A510 und A511 sind nicht zuverlässig. Die Datumsangaben der von der CPU-Baugruppe auf die Speicherkarte geschriebenen Dateien sind ebenfalls nicht zuverlässig.
- Die folgenden Daten sind beim Start auf null gesetzt: Einschaltdauer (A523), Ausschaltzeitpunkt (A512 und A513) und Anzahl der Ausschaltvorgänge (A514).
- Der Fehlerprotokollbereich in A100 bis A199 bleibt nicht erhalten.
- Die aktuelle EM-Bank ist beim Start immer 0 (Null).
- Im EM-Dateispeicher befinden sich beim Startvorgang keine Dateien, und die Dateispeicherfunktionen können nicht verwendet werden. Der EM-Dateispeicher muss im SPS-Setup zurückgesetzt und neu formatiert werden.

CS1-H, CJ1-H, CJ1M oder CS1D CPU-Baugruppen

Der normale Betrieb ist bei den CS1-H, CJ1-H, CJ1M oder CS1D CPU-Baugruppen ohne Batterie möglich. Das Anwenderprogramm und die Parameterdaten werden beim Startvorgang automatisch im Flash-Speicher der CPU-Baugruppe gesichert bzw. aus diesem wiederhergestellt. In diesem Fall bleibt der E/A-Speicher nicht erhalten, und die DM- und EM-Bereiche müssen durch das Anwenderprogramm initialisiert werden.

Der Betrieb ohne Batterie ist auch bei den CS1-H, CJ1-H, CJ1M oder CS1D CPU-Baugruppen möglich, wenn die Daten beim Startvorgang automatisch von einer Speicherkarte übertragen werden (wie bei den CS1 CPU-Baugruppen). (Bei Verwendung einer Speicherkarte können die DM- und EM-Bereichsdaten hinzugefügt werden.)

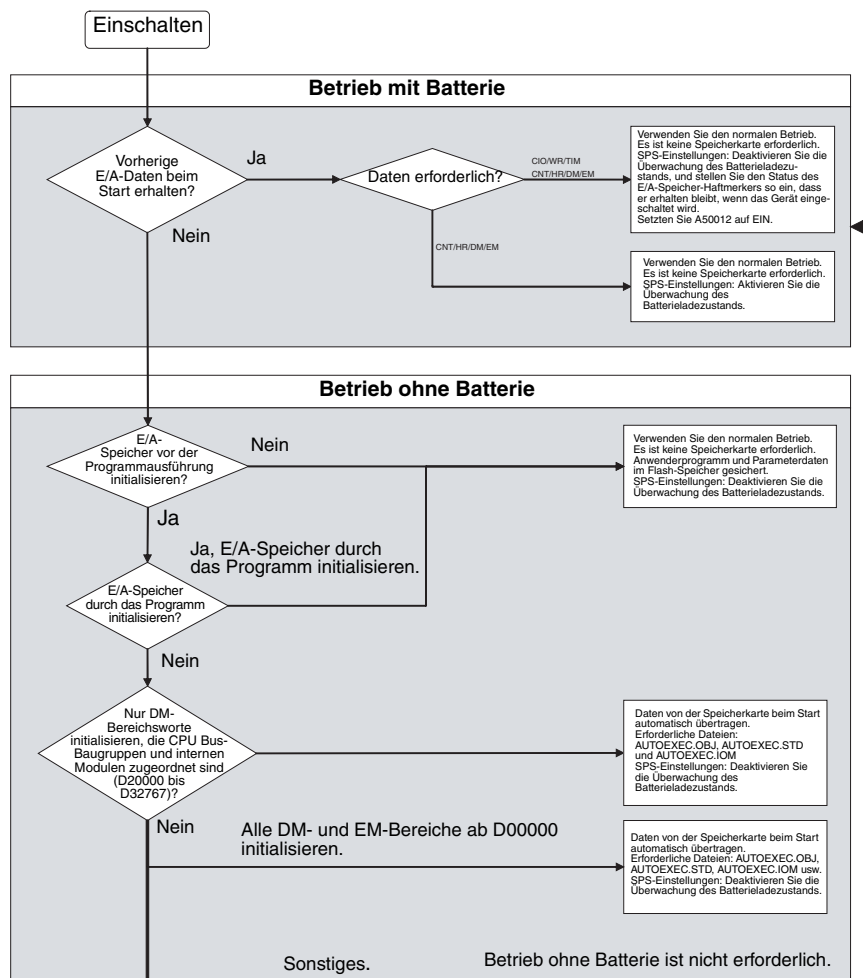
CS1 und CJ1 CPU-Baugruppen

Der Betrieb ohne Batterie ist bei den CS1 und CJ1 CPU-Baugruppen möglich, wenn die Daten beim Startvorgang automatisch von einer Speicherkarte übertragen werden. In diesem Fall bleibt der E/A-Speicher nicht erhalten. (Bei Verwendung einer Speicherkarte können die DM- und EM-Bereichsdaten hinzugefügt werden.)

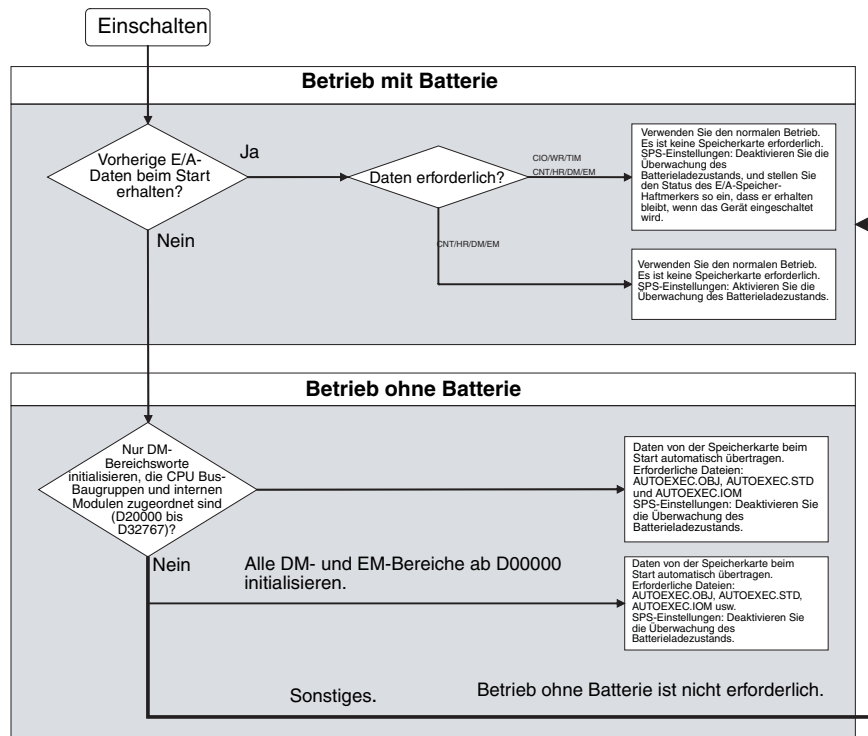
Vorgehensweise

Im folgenden Ablaufdiagramm werden die Vorgehensweisen für die beiden Arten von CPU-Baugruppen dargestellt.

CS1-H, CJ1-H, CJ1M oder CS1D CPU-Baugruppen



CS1 und CJ1 CPU-Baugruppen



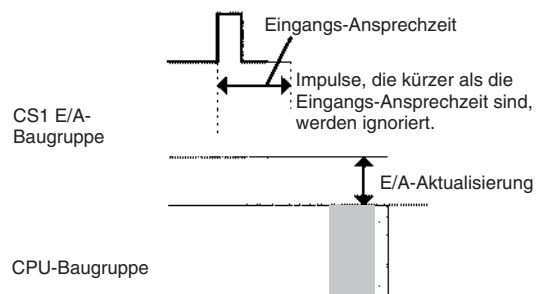
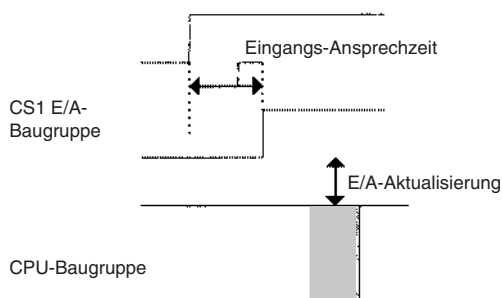
6-11 Weitere Funktionen

6-11-1 E/A-Ansprechzeit-Einstellungen

Die Eingangs-Ansprechzeiten können bei CS/CJ E/A-Baugruppen nach Rack- und Steckplatznummern eingestellt werden. Durch Verlängerung der Eingangs-Ansprechzeit können unerwünschte Effekte von Kontaktprellen und Störungen reduziert werden. Durch Verkürzung der Eingangs-Ansprechzeit (jedoch längerer Impulsdauer als die Zykluszeit) können kürzere Eingangsimpulse empfangen werden.

Hinweis

Bei CPU-Baugruppen der CS-Serie können kürzere Impulse als die Zykluszeit über schnelle Eingänge eingegeben werden, die in einigen C200H Multi-Point-E/A-Baugruppen oder in Schnelle-Zähler-Eingangsbaugruppen verfügbar sind. Weitere Informationen finden Sie unter 6-1-4 *Schnelle Eingänge*.



SPS-Einstellungen

Die Eingangs-Ansprechzeiten für die 80 Steckplätze in einem CS/CJ SPS-System (Rack 0, Steckplatz 0 bis Rack 7, Steckplatz 9) können in den 80 Byte der Adressen 10 bis 49 festgelegt werden.

Adresse der Programmierkonsole	Bezeichnung	Einstellung (Hex)	Standard-einstellung (Hex)
10 Bits 0 bis 7	CS/CJ E/A-Baugruppe Eingangs-Ansprechzeit für Rack 0, Steckplatz 0	00: 8 ms 10: 0 ms 11: 0,5 ms 12: 1 ms 13: 2 ms 14: 4 ms 15: 8 ms 16: 16 ms 17: 32 ms	00 (8 ms)
:	:	:	:
49 Bits 8 bis 15	CS/CJ E/A-Baugruppe Eingangs-Ansprechzeit für Rack 7, Steckplatz 9	Wie oben	00 (8 ms)

6-11-2 E/A-Bereichszuordnung

Mit der Programmiersoftware kann das erste Wort für die E/A-Zuordnung in Erweiterungs-Racks (CS/CJ-Erweiterungs-Racks und C200H-E/A-Erweiterungs-Racks) festgelegt werden. Durch diese Funktion können alle E/A-Zuordnungen der Racks innerhalb des Bereichs CIO 0000 bis CIO 0999 festgelegt werden. (Die ersten Worte werden durch die Racknummer zugeordnet.)

ABSCHNITT 7

Programmübertragung, Testbetrieb und Debugging

Dieser Abschnitt enthält eine Beschreibung der Vorgehensweisen für die Programmübertragung an die CPU-Baugruppe und der Funktionen zum Testen und zum Debugging des Programms.

7-1	Programmübertragung	348
7-2	Testbetrieb und Debugging	348
7-2-1	Zwangsweises Setzen/Rücksetzen	348
7-2-2	Flankenüberwachung	349
7-2-3	Online-Bearbeitung	350
7-2-4	Datenaufzeichnung	353

7-1 Programmübertragung

Um die Programme, die SPS-Einstellungen sowie die E/A-Speicherdaten und E/A-Kommentare auf die CPU-Baugruppe in der PROGRAM-Betriebsart zu übertragen, wird ein Programmiergerät verwendet.

Verfahren zur Programmübertragung mit CX-Programmer

- 1,2,3... 1. Wählen Sie **SPS, Übertragung**, und dann **Zur SPS**. Das Dialogfeld „Herunterladeoptionen“ wird angezeigt.
2. Legen Sie fest, welche der folgenden Elemente übertragen werden sollen: Programme, Einstellungen (SPS-Setup), E/A-Tabelle, Symbole, Kommentare und Programm-Index.
Hinweis Die E/A-Tabelle und die Kommentare können nur ausgewählt werden, wenn die Speicherkarte in der CPU-Baugruppe vorhanden sind.
3. Klicken Sie auf **OK**.

Das Programm kann unter Verwendung einer der folgenden Methoden übertragen werden.

- Automatische Übertragung beim Einschalten

Beim Einschalten wird die Datei AUTOEXEC.OBJ von der Speicherkarte in die CPU-Baugruppe eingelesen (Schalter 2 des DIP-Schalterblocks muss auf EIN stehen).

- Programmaustausch während des Betriebs

Während des Betriebs der CPU-Baugruppe kann die bestehende Programmdatei durch die Programmdatei, die im Zusatz-Systembereich spezifiziert ist, ausgetauscht werden. Dazu muss vom Programm aus das Austausch-Startbit im Zusatz-Systembereich (A65015) auf EIN gesetzt werden. Weitere Informationen finden Sie in **ABSCHNITT 5 Dateispeicherfunktionen**.

Hinweis Bei CX-Programmer-Version 4.0 oder höher in Verbindung mit CPU-Baugruppen der CS/CJ-Serie, Version 2.0 oder höher, können Task-Programme einzeln heruntergeladen werden. Weitere Informationen finden Sie im Abschnitt **1-4-1 Herunterladen und Hochladen von einzelnen Tasks** im **CS Serie SPS Bedienerhandbuch** oder im **CJ Serie SPS Bedienerhandbuch**.

7-2 Testbetrieb und Debugging

7-2-1 Zwangsweises Setzen/Rücksetzen

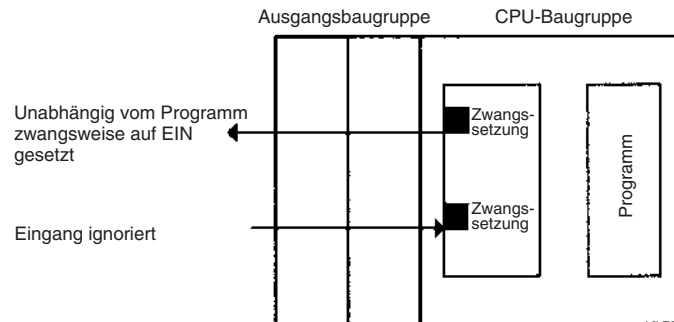
Mit Hilfe eines Programmiergeräts können spezifizierte Bits (CIO-Bereich, Zusatz-Systembereich, HR-Bereich oder Zeitgeber/Zähler-Ablaufmerker) zwangsweise gesetzt (EIN) oder zurückgesetzt (AUS) werden. Der erzwungene Status hat Priorität gegenüber dem, den das Programm oder die E/A-Aktualisierung ergibt. Dieser erzwungene Status kann nicht durch andere Befehle überschrieben werden. Er bleibt unabhängig vom Programmstatus und unabhängig von externen Eingaben so lange gespeichert, bis er über ein Programmiergerät wieder aufgehoben wird.

Zwangsweises Setzen/Rücksetzen wird eingesetzt, um während eines Testbetriebs Eingaben oder Ausgaben zu simulieren oder um beim Debugging bestimmte Bedingungen herbeizuführen.

Zwangsweises Setzen/Rücksetzen kann in der MONITOR- oder PROGRAM-Betriebsart durchgeführt werden, jedoch nicht in der RUN-Betriebsart.

Hinweis Damit der Bit-Status, der durch zwangsweises Setzen/Rücksetzen bewirkt wurde, bei Umschalten der Betriebsart beibehalten wird, setzen Sie den Zwangssetzungs-Haftmerker (A50013) und den E/A-Speicher-Haftmerker (A50012) gleichzeitig auf EIN.

Setzen Sie den Zwangssetzungs-Haftmerker (A50013) und den E/A-Speicher-Haftmerker (A50012) auf EIN. Der Zwangssetzungs-Haftmerker muss bei der Start-Einstellung in den SPS-Einstellungen eingestellt sein, dass sein Status und der der zwangsgesetzten/-zurückgesetzten Bits danach beibehalten wird, wenn die Spannungsversorgung zwischendurch mal ausgeschaltet wird.



In folgenden Bereichen kann zwangsweises Setzen/Rücksetzen vorgenommen werden.

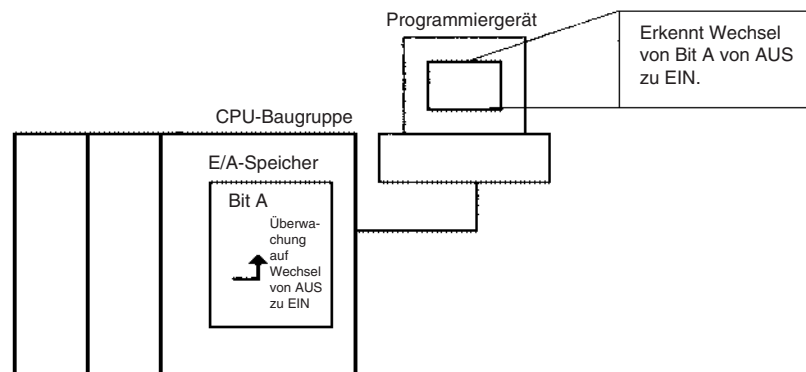
CIO (E/A-Bits, Data-Link-Bereich, CPU-Bus-Baugruppenbereiche, Spezial-E/A-Baugruppenbereich, Bits des internen Moduls, SYSMAC-BUS-Bits, Optische E/A-Baugruppen-Bits, Arbeitsbits), W-Bereich, Zeitgeberablaufmerker, H-Bereich, Zählerablaufmerker. (Bereich für internes Modul und SYSMAC BUS-Bereich werden nur von CPU-Baugruppen der CS-Serie unterstützt.)

Bedienverfahren zum Zwangssetzen mit Programmiersoftware

- Auswahl der Bits für zwangsweises Setzen/Rücksetzen.
- Auswahl von zwangsweisem Setzen oder Rücksetzen.
- Zwangssetzungsstatus aufheben (auch gleichzeitiges Aufheben aller erzwungenen Status).

7-2-2 Flankenüberwachung

Sobald von der CPU-Baugruppe registriert wird, dass ein mit der Programmiersoftware ausgewähltes Bit von AUS zu EIN bzw. von EIN zu AUS gesetzt hat, wird das durch einen Flankenüberwachungsmerker (A50809) signalisiert. Der Merker wird auf EIN gesetzt, wenn die festgelegten Bedingungen für die Flankenüberwachung erfüllt sind. Die Ergebnisse können auf dem Bildschirm des Programmiergeräts angezeigt und überwacht werden.



Bedienverfahren bei CX-Programmer

- 1,2,3...**
1. Klicken Sie mit der rechten Maustaste auf das Bit für Flankenüberwachung.
 2. Klicken Sie im SPS-Menü auf **Flankenüberwachung**. Das Dialogfeld „Flankenüberwachung“ wird angezeigt.
 3. Klicken Sie auf **steigend** oder **fallend**.
 4. Klicken Sie auf **Start**. Der Summer ertönt, sobald die spezifizierte Änderung vom Gerät erkannt wurde und der Zähler wird inkrementiert.
 5. Klicken Sie auf **Stopp**. Die Flankenüberwachung wird angehalten.

Zugehörige Bits/Worte im Zusatz-Systembereich

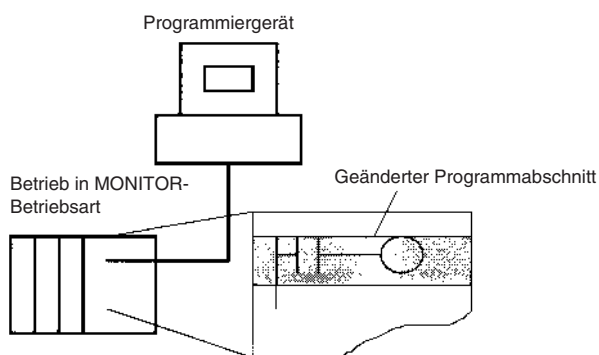
Bezeichnung	Adresse	Beschreibung
Flankenüberwachung-erfüllt-Merker	A50809	Wird auf EIN gesetzt, sobald die Bedingungen bei der Flankenüberwachung erfüllt sind. Hinweis: Der Merker wird gelöscht, sobald die Flankenüberwachung gestartet wird.

7-2-3 Online-Bearbeitung

Die Funktion zur Online-Bearbeitung wird eingesetzt, wenn vom Programmiergerät aus ein Programmteil in einer CPU-Baugruppe hinzugefügt oder geändert werden soll. Dazu muss sich die CPU-Baugruppe in der MONITOR- oder PROGRAM-Betriebsart befinden. Die Hinzufügungen oder Änderungen Befehl für Befehl mit der Programmierkonsole bzw. Programmabschnitt für Programmabschnitt mit dem CX-Programmer werden nacheinander abgearbeitet. Diese Funktion ist dazu konzipiert, kleinere Programmänderungen durchzuführen, ohne die CPU-Baugruppe anzuhalten.

Online-Bearbeitung kann von einer Programmierkonsole und von mehreren Computern, auf denen der CX-Programmer ausgeführt wird, gleichzeitig durchgeführt werden, sofern unterschiedliche Tasks bearbeitet werden.

Online-Bearbeitung



Wenn das Programm in der CPU-Baugruppe in der MONITOR-Betriebsart online bearbeitet wird, wird dadurch die Zykluszeit erhöht.

Die Zykluszeiten von CS1-H, CJ1-H, CJ1M und CS1D CPU-Baugruppen werden auch erhöht, wenn nach der Online-Bearbeitung eine Sicherung der Daten im Flash-Speicher erfolgt. Die BKUP-Anzeige leuchtet während dieses Vorgangs. Bei CX-Programmer wird der Fortschritt der Datensicherung angezeigt. Die Zeitzunahmen pro Zyklus werden in der folgenden Tabelle aufgelistet.

CPU-Baugruppe	Zunahme der Zykluszeit	
	Online-Bearbeitung	Sicherung im Flash-Speicher
CS1 CPU-Baugruppen vor EV1	max. 90 ms	Nicht unterstützt
CS1 CPU-Baugruppen ab EV1	max. 12 ms	
CS1-H CPU-Baugruppen		4 % oder Zykluszeit
CS1D CPU-Baugruppen		
CS1 CPU-Baugruppen		Nicht unterstützt
CJ1-H CPU-Baugruppen		4 % oder Zykluszeit
CJ1M CPU-Baugruppen		

Bei den CPU-Baugruppen des Typs CS1-H, CJ1-H, CJ1M oder CS1D können nur eine begrenzte Anzahl von Bearbeitungen aufeinander folgend durchgeführt werden. Die Anzahl ist abhängig von der Art der Bearbeitung. Es gelten jedoch folgende Richtwerte.

CJ1M-CPU□□:	40 Bearbeitungen
CS1G-CPU□□H/CJ1G-CPU□□H:	160 Bearbeitungen
CS1H-CPU□□H/CJ1H-CPU□□H/CS1D-CPU□□H/ CS1D-CPU□□S:	400 Bearbeitungen

Wenn die kritische Anzahl erreicht ist, zeigt der CX-Programmer oder die Programmierkonsole eine entsprechende Meldung an. Eine weitere Bearbeitung ist erst dann wieder möglich, nachdem die CPU-Baugruppe die Sicherung der Daten abgeschlossen hat.

Task-Größe und Zykluszeit-Verlängerung

Zwischen der Größe der bearbeiteten Task und der Zykluszeit-Verlängerung besteht folgendes Verhältnis:

Bei CS1, CS1-H, CS1D, CJ1 oder CJ1M CPU-Baugruppen der Version 1 oder höher hat die Größe der online bearbeiteten Tasks (Programme) kaum einen verlängernden Einfluss auf die Zykluszeit.

Bei einer CS1 CPU-Baugruppe vor EV1 bestimmt die Größe der gerade bearbeiteten Task die Zeit, für die das betreffende Programm für die Online-Bearbeitung angehalten wird. Durch Aufteilen des Programms in kleinere Tasks fällt die Verlängerung der Zykluszeit durch die Online-Bearbeitung geringer aus als bei früheren SPS-Modellen.

Vorsichtsmaßnahmen

Wenn bei Online-Bearbeitung in der MONITOR-Betriebsart ein Programm überschrieben wird, ist die Zykluszeit länger als normal. Achten Sie daher darauf, dass die verlängerte Zykluszeit nicht länger dauert als die Zyklus-Überwachungszeit, die im SPS-Setup eingestellt ist. Falls nämlich die Überwachungszeit überschritten wird, tritt ein Zykluszeit-Überschreitungsfehler auf, und die CPU-Baugruppe wird angehalten. In diesem Fall müssen Sie die CPU-Baugruppe neu starten. Wählen Sie dabei zunächst die PROGRAM-Betriebsart und wechseln Sie anschließend danach in die RUN- oder MONITOR-Betriebsart.

Hinweis

Falls die online bearbeitete Task ein Blockprogramm enthält, werden durch zuvor ausgeführte Befehle gesetzte Daten, wie Standby (WAIT)- oder Pause-Zustände, durch die Online-Bearbeitung aufgehoben, und die nächste Ausführung erfolgt von Anfang an.

Online-Bearbeitung mit CX-Programmer

1,2,3...

1. Zeigen Sie den zu bearbeitenden Programmabschnitt an.
2. Wählen Sie die zu bearbeitenden Befehle.
3. Wählen Sie **Programm, Online-Bearbeitung** und dann **Anfangen**.
4. Bearbeiten Sie die Befehle.
5. Wählen Sie **Programm, Online-Bearbeitung**, und dann **Änderungen senden**. Die Befehle werden überprüft, und wenn sie keine Fehler enthalten, werden sie zur CPU-Baugruppe übertragen. Die Befehle in der CPU-Baugruppe werden überschrieben. Dadurch wird die Zykluszeit verlängert.



Achtung

Fahren Sie erst mit der Online-Bearbeitung fort, nachdem Sie sich überzeugt haben, dass die verlängerte Zykluszeit den Betrieb nicht stört. Wenn die Zykluszeit zu lang ist, werden Eingangssignale möglicherweise nicht registriert.

Vorübergehende Deaktivierung der Online-Bearbeitung

Um für einen Zyklus die Reaktionseigenschaften für die Maschinensteuerung zu gewährleisten, kann für diesen Zyklus die Möglichkeit zur Online-Bearbeitung deaktiviert werden. Für das Programmiergerät wird die Online-Bearbeitung für diesen Zyklus gesperrt, und Anforderungen für Online-Bearbeitungen, die während dieses Zyklus eingehen, werden bis zum nächsten Zyklus geparkt, d. h. der Online-Bearbeitungsschritt wird in den Wartezustand versetzt.

Die Online-Bearbeitung wird gesperrt, indem das Online-Bearbeitungs-Deaktivierungsbit (A52709) auf EIN gesetzt wird und der Online-Bearbeitungs-Deaktivierungsbit-Validator (A52700 bis A52707) auf 5A. Sind diese Einstellungen in Kraft und geht dann die Anforderung für Online-Bearbeitung ein, wird diese in den Wartezustand gesetzt, und der Online-Bearbeitungs-Wartemerker (A20110) wird auf EIN gesetzt.

Online-Bearbeitung kann durchgeführt werden, wenn das Online-Bearbeitungs-Deaktivierungsbit (A52709) auf AUS gesetzt ist. Dann wird der Online-Bearbeitungs-Prozess-Merker (A20111) auf EIN gesetzt und der Online-Bearbeitungs-Wartemerker (A20110) auf AUS. Ist die Online-Bearbeitung abgeschlossen, wird der Online-Bearbeitungs-Prozess-Merker (A20111) auf AUS gesetzt.

Die Online-Bearbeitung kann auch während des Vorgangs der Online-Bearbeitung vorübergehend gesperrt werden, indem das Online-Bearbeitungs-Deaktivierungsbit (A52709) auf EIN gesetzt wird. Auch in diesem Fall wird der Online-Bearbeitungs-Wartemerker (A20110) auf EIN gesetzt.

Wenn eine zweite Anforderung für eine Online-Bearbeitung empfangen wird und die erste Anfrage auf Standby gesetzt ist, wird die zweite Anfrage nicht aufgezeichnet, und ein Fehler tritt auf.

Die Online-Bearbeitung kann auch deaktiviert werden, um eine versehentliche Online-Bearbeitung zu verhindern. Wie oben beschrieben wird die Online-Bearbeitung deaktiviert, indem das Online-Bearbeitungs-Deaktivierungsbit (A52709) auf EIN gesetzt wird und der Online-Bearbeitungs-Deaktivierungsbit-Validator (A52700 bis A52707) auf 5A.

Aktivieren der Online-Bearbeitung mit einem Programmiergerät

Die Online-Bearbeitung kann nicht nur vom Programm aus aktiviert werden, sondern vielmehr sie kann auch mit CX-Programmer aktiviert werden.

1,2,3...

1. Durchführen der Online-Bearbeitung mit einer Programmierkonsole

Wenn die Online-Bearbeitung mit einer Programmierkonsole ausgeführt wird und der Standby-Status der Online-Bearbeitung nicht gelöscht werden kann, wird die Programmierkonsole gesperrt, so dass die Arbeit mit der Programmierkonsole nicht fortgesetzt werden kann.

In diesem Fall schließen Sie CX-Programmer an eine andere serielle Schnittstelle an und setzen das Online-Bearbeitungs-Deaktivierungsbit (A52709) auf AUS. Dadurch wird die Online-Bearbeitung wieder ausgeführt, so dass die Online-Bearbeitung mit der Programmierkonsole wieder möglich ist.

2. Durchführen der Online-Bearbeitung mit CX-Programmer

CX-Programmer kann offline gehen, sofern der Betrieb fortgesetzt wird, obwohl ein Online-Bearbeitungsschritt im Wartezustand ist. In diesem Fall verbinden Sie den Computer erneut mit der SPS und setzen das Online-Bearbeitungs-Deaktivierungsbit (A52709) auf AUS.

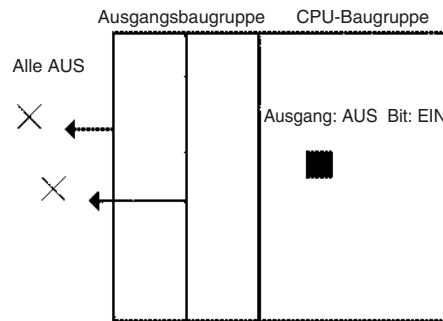
Zugehörige Bits/Worte im Zusatz-Systembereich

Bezeichnung	Adresse	Beschreibung
Online-Bearbeitung-Deaktivierungsbit-Validator	A52700 bis A52707	Validiert das Online-Bearbeitungs-Deaktivierungsbit (A52709). Nicht 5A: Online-Bearbeitungs-Deaktivierungsbit ungültig 5A: Online-Bearbeitungs-Deaktivierungsbit gültig
Online-Bearbeitung-Deaktivierungsbit	A52709	Um Online-Bearbeitung zu sperren, setzen Sie dieses Bit auf EIN und den Online-Bearbeitungs-Deaktivierungsbit-Validator (A52700 bis A52707) auf 5A.
Online-Bearbeitung-Wartemerker	A20110	Auf EIN gesetzt, wenn ein Online-Bearbeitungsschritt im Wartezustand ist, da die Online-Bearbeitung gesperrt ist.
Online-Bearbeitung-Prozess-Merker	A20111	EIN während der Ausführung eines Online-Bearbeitungsvorgangs.

Ausschalten der Ausgänge

Wenn das Ausgangs-AUS-Bit (A50015) durch einen OUT-Befehl oder durch ein Programmiergerät auf EIN gesetzt wird, werden alle Ausgänge von Ausgangs-Baugruppen ausgeschaltet (das gilt ebenso für die integrierten Ausgänge und die Impulsausgänge bei CJ1M CPU-Baugruppen), und die INH-Anzeige auf der Vorderseite der CPU-Baugruppe leuchtet.

Der Status des Ausgang-AUS-Bits bleibt selbst dann erhalten, wenn das Gerät aus- und wieder eingeschaltet wird.



7-2-4 Datenaufzeichnung

Die Datenaufzeichnungsfunktion erfasst spezifizierte E/A-Speicherdaten unter Verwendung einer der folgenden Methoden zur Bestimmung der Abtastung. Die Daten werden im Aufzeichnungsspeicher abgelegt, aus dem sie von einem Programmiergerät gelesen und überprüft werden können.

- Spezifizierter Aufzeichnungsintervall (10 bis 2.550 ms in Schritten von 10 ms)
- Eine Abtastung pro Zyklus
- Wenn der Befehl TRACE MEMORY SAMPLING (TRSM) ausgeführt wird

Für die Erfassung können bis zu 31 Bits und 6 Worte des E/A-Speichers spezifiziert werden. Der Aufzeichnungsspeicher hat eine Kapazität von 4000 Worten.

Grundverfahren

1,2,3...

1. Die Datenerfassung wird gestartet, sobald die Parameter mit CX-Programmer gesetzt wurden und der Befehl zum Start der Aufzeichnung ausgeführt wurde.
2. Die Aufzeichnung (nach Schritt 1 oben) erfolgt, sobald die entsprechende auslösende Bedingung (Trigger-Bedingung) erfüllt ist. Dann werden die Daten unmittelbar nach der Verzögerung (siehe Hinweis 1) in den Aufzeichnungsspeicher geschrieben.
3. Die Daten werden in den Aufzeichnungsspeicher geschrieben und die Aufzeichnung endet.

Hinweis

Verzögerungswert: Gibt an, um wie viele Erfassungsperioden die Aufzeichnung im Aufzeichnungsspeicher versetzt werden soll, wenn das Aufzeichnungsstartbit (A50814) auf EIN gesetzt wird. Die möglichen Einstellungen entnehmen Sie der folgenden Tabelle.

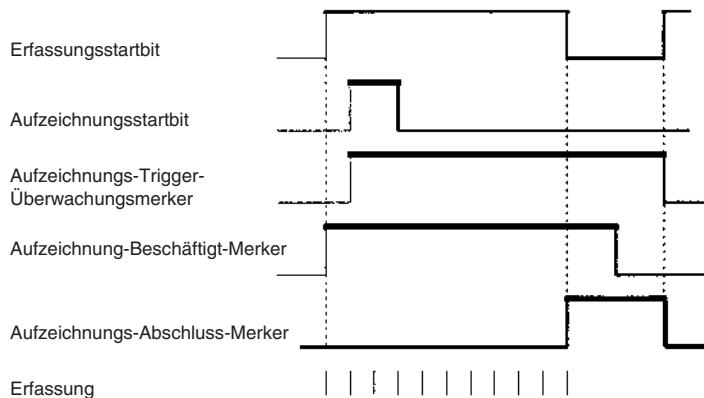
Anzahl der erfassten Worte	Einstellbereich
0	–1999 bis 2000
1	–1332 bis 1333
2	–999 bis 1000
3	–799 bis 800
4	–665 bis 666
5	–570 bis 571
6	–499 bis 500

Positive Verzögerung: Daten verzögert speichern nach festgelegter Verzögerung

Negative Verzögerung: Vorherige Daten speichern gemäß der festgelegten Verzögerung

Beispiel: Erfassung bei 10 ms mit einer Verzögerung von –30 macht $-30 \times 10 = 300$ ms, so dass die Daten aufgezeichnet werden, die 300 ms vor dem Auslöser anfallen.

Hinweis Das Erfassungsstartbit (A50815) darf nur mit einem Programmiergerät auf EIN gesetzt werden. Setzen Sie es niemals durch das Anwendungsprogramm auf EIN.



Folgende Aufzeichnungen können durchgeführt werden:

Zeitgesteuerte Datenaufzeichnung

Bei einer zeitgesteuerten Datenaufzeichnung findet die Aufzeichnung in festen Intervallen statt. Mögliche Intervalle sind 10 bis 2.550 ms in Schritten von 10 ms. Verwenden Sie nicht den TRSM-Befehl im Anwendungsprogramm, und achten Sie auch darauf, dass die Aufzeichnungsperiode größer als 0 ist.

1-Zyklus- Datenaufzeichnung

Bei einer 1-Zyklus-Datenaufzeichnung erfolgt eine Erfassung von aktualisierten E/A-Daten nach Beenden der Tasks im vollen Zyklus. Verwenden Sie nicht den TRSM-Befehl im Anwendungsprogramm, und achten Sie auch darauf, dass die Aufzeichnungsperiode größer als 0 ist.

Datenaufzeichnung durch TRSM

Es erfolgt eine einzelne Datenerfassung, wenn der Befehl TRACE MEMORY SAMPLING (TRSM) ausgeführt wird. Wird mehr als ein TRSM-Befehl im Programm ausgeführt, erfolgt jedes Mal bei Ausführung eines TRSM-Befehls eine Erfassung, sobald die auslösende Bedingung dafür erfüllt ist.

Verfahren zur Datenaufzeichnung

Verwenden Sie das folgende Verfahren zur Durchführung der Datenaufzeichnung.

- 1,2,3...**
 1. Verwenden Sie CX-Programmer, um die Aufzeichnungsparameter einzustellen (*SPS/ Datenaufzeichnung* und nehmen Sie die Einstellung unter *Operation/Konfigurieren* vor.): Adresse der erfassten Daten, Aufzeichnungsperiode, Verzögerungszeit und auslösende Bedingungen.
 2. Verwenden Sie CX-Programmer, um die Aufzeichnung zu starten, oder setzen Sie das Erfassungsstartbit (A50815) auf EIN.
 3. Setzen Sie die auslösende Bedingung (Trigger-Bedingung) in Kraft.
 4. Beenden Sie die Aufzeichnung.
 5. Verwenden Sie CX-Programmer, um die aufgezeichneten Daten zu lesen.
 - a) Wählen Sie im SPS-Menü die Option **Datenaufzeichnung**.
 - b) Wählen Sie im Menü „Operation“ die Option **Konfigurieren**.
 - c) Wählen Sie im Menü „Operation“ die Option **Ausführen**.
 - d) Wählen Sie im Menü „Operation“ die Option **Lesen**.

Zugehörige Bits/Worte im Zusatz-Systembereich

Bezeichnung	Adresse	Beschreibung
Erfassungsstartbit	A50815	Verwenden Sie ein Programmiergerät, um dieses Bit auf EIN zu setzen, damit die Aufzeichnung beginnt. Dieses Bit muss durch ein Programmiergerät auf EIN gesetzt werden. Setzen Sie dieses Bit nicht durch das Anwendungsprogramm auf EIN oder AUS. Hinweis: Dieses Bit wird zurückgesetzt, wenn die Datenaufzeichnung abgeschlossen ist.
Aufzeichnungsstartbit	A50814	Sobald dieses Bit auf EIN gesetzt ist, wird die auslösende Bedingung zum Starten der Aufzeichnung überwacht. Sobald diese erfüllt ist, werden die aufgezeichneten Daten in den Aufzeichnungsspeicher geschrieben. Folgende Aufzeichnungen werden durch dieses Bit freigegeben. 1) Zeitgesteuerte Datenaufzeichnung (Aufzeichnung in festen Intervallen von 10 bis 2250 ms) 2) Aufzeichnung durch TRSM-Befehl (durch Ausführung des Befehls TRSM) 3) 1-Zyklus-Datenaufzeichnung (Aufzeichnung, nachdem alle zyklischen Tasks ausgeführt wurden)
Aufzeichnungs-Trigger-Überwachungsmerker	A50811	Dieser Merker wird auf EIN gesetzt, nachdem das Aufzeichnungsstartbit auf EIN gesetzt wurde und sobald dann die auslösende Bedingung für die Aufzeichnung erfüllt ist. Dieser Merker wird auf AUS gesetzt, wenn die Aufzeichnung erneut gestartet wird, indem das Erfassungsstartbit auf EIN gesetzt wird.
Aufzeichnung-läuft-Merker	A50813	Dieser Merker wird auf EIN gesetzt, sobald durch ein Erfassungsstartbit die Datenaufzeichnung gestartet wurde. Nach Abschluss der Aufzeichnung wird der Merker auf AUS gesetzt.
Aufzeichnungs-Abschluss-Merker	A50812	Dieser Merker wird auf EIN gesetzt, wenn die auslösende Bedingung für die Aufzeichnung erfüllt ist und während einer Aufzeichnung der Aufzeichnungsspeicher voll wird. Der Merker wird auf AUS gesetzt, wenn die nächste Aufzeichnung gestartet wird.

Anhang A

SPS-Vergleichstabellen:

SPS der CJ-Serie, CS-Serie, C200HG/HE/HX, CQM1H, CVM1 und der CV-Serie

Funktionsvergleich

Kenndaten			CJ-Serie	CS-Serie	C200HX/HG/HE	CVM1/CV-Serie	CQM1H
Grundmerkmale	Kapazität	Max. Anzahl der E/A	2.560	5.120	1.184	6.144	512
		Programmkapazität	250 kSteps Ein Step entspricht im Wesentlichen einem Wort. Ausführliche Informationen hierzu finden Sie am Ende des Kapitels 10-5 <i>Ausführungszeit und Step-Anzahl der Befehle im Bedienerhandbuch.</i>	250 kSteps Ein Step entspricht im Wesentlichen einem Wort. Ausführliche Informationen hierzu finden Sie am Ende des Kapitels 10-5 <i>Ausführungszeit und Step-Anzahl der Befehle im Bedienerhandbuch.</i>	3,2 kWorte (63,2 kWorte bei -Z)	62 kWorte	15,2 kWorte
		Maximaler Datenspeicher	32 kWorte	32 kWorte	6 kWorte	24 kWorte	6 kWorte
		Lokale E/A	160 Worte (2.560 Bits)	320 Worte (5.120 Bits)	40 Worte (640 Bits)	128 Worte (2.048 Bits)	32 Worte (512 Bits)
		Arbeitsbits	2.644 Worte (42.304 Bits) + Arbeitsbereich: 512 Worte (8.192 Bits) = 3.156 Worte (50.496 Bits)	2.644 Worte (42.304 Bits) + Arbeitsbereich: 512 Worte (8.192 Bits) = 3.156 Worte (50.496 Bits)	408 Worte (6.528 Bits)	168 Worte (2.688 Bits) + 400 Worte (6.400 Bits)	158 Worte (2.528 Bits)
		Haftmerker-Bits	512 Worte (8.192 Bits)	512 Worte (8.192 Bits)	100 Worte (1.600 Bits)	300 Worte (4.800 Bits) Max.: 1.400 Worte (2.400 Bits)	100 Worte (1.600 Bits)
		Maximaler erweiterter Datenspeicher	32 kWorte x 13 Bänke	32 kWorte x 13 Bänke	6 kWorte x 3 Bänke (6 kWorte x 16 Bänke bei -Z)	32 kWorte x 8 Bänke (optional)	6 kWorte
		Max. Anzahl Zeitgeber/Zähler	jeweils 4.096	jeweils 4.096	Zeitgeber/Zähler zusammen: 512	jeweils 1.024	Zeitgeber/Zähler zusammen: 512
	Verarbeitungs-geschwindigkeit	Grundbefehle (LD)	CJ1: min. 0,08 µs CJ1-H: min. 0,02 µs CJ1M: min. 0,1 µs	CS1: min. 0,04 µs CS1-H: min. 0,02 µs	min. 0,104 µs	min. 0,125 µs	min. 0,375 µs
		Spezialbefehle (MOV)	CJ1: min. 0,25 µs CJ1-H: min. 0,18 µs CJ1M: min. 0,3 µs	CS1: min. 0,25 µs CS1-H: min. 0,18 µs	min. 0,417 µs	min. 4,3 µs	17,7 µs
		Organisationsablaufzeit des Systems	CJ1: min. 0,5 ms CJ1-H: min. 0,3 ms im Normalmodus, 0,2 ms in einem Parallelverarbeitungsmodus CJ1M: min. 0,5 ms	CS1: min. 0,5 ms CS1-H: min. 0,3 ms im Normalmodus, 0,2 ms in einem Parallelverarbeitungsmodus	0,7 ms	0,5 ms	0,7 ms
		Verzögerung bei der Online-Bearbeitung (Schreiben)	CJ1: ca. 12 ms CJ1-H: ca. 11 ms bei CPU4□ und 8 ms bei CPU6 CJ1M: ca. 14 ms	CS1: ca. 12 ms CS1-H: ca. 11 ms bei CPU4□ und 8 ms bei CPU6	80 ms (160 ms bei -Z)	500 ms	In der Regel 250 ms
Bauart	Schraubmontage		Nein	Ja	Ja	Ja	Nein
	Montage auf DIN-Schienen		Ja	Ja	Ja	Nein	Ja
	Baugruppenträger		Nein	Ja	Ja	Ja	Nein
	Größe (H x T, mm)		90 x 65	130 x 123	130 x 118	250 x 100	110 x 107

Kenndaten		CJ-Serie	CS-Serie	C200HX/HG/HE	CVM1/CV-Serie	CQM1H
Anzahl der Baugruppen/Racks	E/A-Baugruppen	40 Baugruppen	89 Baugruppen (einschl. Slave-Racks)	10 oder 16 Baugruppen	64 Baugruppen (8 Racks x 8 Baugruppen)	16 Baugruppen
	CPU-Bus-Baugruppen	16 Baugruppen	16 Baugruppen	Keine	16 Baugruppen	Keine
	Erweiterungs-Racks/Blöcke	3 Blöcke	7 Racks	3 Racks	7 Racks	1 Block
Task-Funktion		Ja	Ja	Nein	Nein	Nein
CPU-Verarbeitungsbetriebsart (Programmausführung und Bedienung der Peripheriegeräte)	Normale Betriebsart	Ja	Ja	---	---	---
	Peripheriebedienungs-Prioritätsmodus	Ja	Ja	---	---	---
	Parallelverarbeitungsbetriebsart mit synchronem Speicherzugriff	CJ1: Nein CJ1-H: Ja CJ1M: Nein	CS1: Nein CS1-H: Ja	Nein	Nein	Nein
	Parallelverarbeitungsbetriebsart mit asynchronem Speicherzugriff	CJ1: Nein CJ1-H: Ja CJ1M: Nein	CS1: Nein CS1-H: Ja	Nein	Nein	Nein
E/A-Aktualisierungsformat	Zyklische Aktualisierung	Ja	Ja	Ja	Ja	Ja
	Zeitgesteuerte Aktualisierung	Nein	Nein	Nein	Ja	Nein
	Nulldurchgang-Aktualisierung	Nein	Nein	Nein	Ja	Nein
	Direkte Aktualisierung	Ja	Ja	Nein	Ja	Nein
	Direkte Aktualisierung durch den Befehl IORF	Ja	Ja	Ja	Ja	Ja
Uhrfunktion		Ja	Ja	Ja	Ja	Ja (Speicherkassette erforderlich)
RUN-Ausgang		Ja (je nach Spannungsversorgungs-Baugruppe)	Ja (je nach Spannungsversorgungs-Baugruppe)	Ja (je nach Spannungsversorgungs-Baugruppe)	Ja	Nein
Start-Betriebsart (SPS-Setup-Standard-einstellung wenn keine Programmierkonsole angeschlossen ist)		RUN-Betriebsart	CS1: PROGRAM-Betriebsart CS1-H: RUN-Betriebsart	RUN-Betriebsart	RUN-Betriebsart	PROGRAM-Betriebsart
Deaktivieren der Ausschalt-Interrupt-Verarbeitung		CJ1: Nein CJ1-H: Ja CJ1M: Ja	CS1: Nein CS1-H: Ja	Nein	Nein	Nein
Nichtflüchtiger Speicher		CJ1: Speicherkarte CJ1-H: Speicherkarte oder Flash-Speicher CJ1M: Speicherkarte oder Flash-Speicher	CS1: Speicherkarte CS1-H: Speicherkarte oder Flash-Speicher	Speicherkarte	Speicherkarte	Speicherkassette
Automatische Sicherung im Flash-Speicher		CJ1: Nein CJ1-H: Ja CJ1M: Ja	CS1: Nein CS1-H: Ja	Nein	Nein	Nein
Neustart-Fortsetzung		Nein	Nein	Nein	Ja	Nein

Kenndaten		CJ-Serie	CS-Serie	C200HX/HG/HE	CVM1/CV-Serie	CQM1H
Externer Speicher	Medium	Speicherkarte (Flash-ROM)	Speicherkarte (Flash-ROM)	Speicherkassette (EEPROM, EPROM)	Speicherkarte (RAM, EEPROM, EPROM)	Speicherkassette (ROM, EEPROM, EPROM)
	Kapazität	48 MB	48 MB	4 bis 32 kWorte (4 bis 64 kWorte bei -Z)	32 bis 512 kWorte (RAM: 64 bis 512 KB, EEPROM: 64 bis 128 KB, EPROM: 0,5 bis 1 MB)	4 bis 16 kWorte
	Inhalt	Programme, E/A-Speicher, Parameter	Programme, E/A-Speicher, Parameter	Programme, E/A-Speicher, Parameter	Programme, E/A-Speicher, Parameter	Programme, nur lesbarer DM-Bereich, Parameter
	Lese-/Schreibmethode	Programmierkonsole, Anwenderprogramm (Dateispeicherbefehle) oder Host-Link	Programmierkonsole, Anwenderprogramm (Dateispeicherbefehle) oder Host-Link	SR-Bit auf EIN setzen	Programmierkonsole, Anwenderprogramm (Dateispeicherbefehle), Host-Link oder Speicherkarten-Schreibgerät	AR-Bit auf EIN setzen
	Dateiformat	Binär	Binär	Binär	Binär	Binär
	Erweiterter Datenspeicher als Dateispeicher	Ja (außer bei CJ1M CPU-Baugruppen)	Ja	Nein	Nein	Nein
	Automatische Programmübertragung beim Start	Ja	Ja	Ja	Ja	Ja
Internes Kommunikationsmodul		Nein	Seriell kommunikatonsmodul	Kommunikationsmodul	Nein	Kommunikationsmodul
Integrierte serielle Schnittstellen		Ja (1 x RS-232C)	Ja (1 x RS-232C)	Ja (1 x RS-232C)	Ja (1 x RS-232C oder RS-422)	Ja (1 x RS-232C)

Kenndaten			CJ-Serie	CS-Serie	C200HX/HG/HE	CVM1/CV-Serie	CQM1H
Serielle Kommunikation	Peripherie-schnittstelle	Peripheriebus	Ja	Ja	Ja	Ja	Ja
		Host-Link (SYSMAC WAY)	Ja	Ja	Ja	Nein (bei Anschluss an Peripherie-schnittstelle möglich)	Ja
		Seriell Gateway (Konvertierung in CompoWay/F)	Ja (CPU-Baugruppen ab Version 3.0)	Ja (CPU-Baugruppen ab Version 3.0)	Nein	Nein	Nein
		Protokollfrei	Nein	Nein	Ja	Nein	Ja
		NT-Link	Ja	Ja	Nein	Nein	Nein
	Integrierte RS-232C-Schnittstelle der CPU-Baugruppe	Peripheriebus	Ja	Ja	Ja	Nein	Nein
		Host-Link (SYSMAC WAY)	Ja	Ja	Ja	Ja	Ja
		Seriell Gateway (Konvertierung in CompoWay/F)	Ja (CPU-Baugruppen ab Version 3.0)	Ja (CPU-Baugruppen ab Version 3.0)	Nein	Nein	Nein
		Protokollfrei	Ja	Ja	Ja	Nein	Ja
		NT-Link	Ja (1:N)	Ja (1:N)	Ja	Nein	Ja (1:1)
		Serielle SPS-Links	Ja (nur CJ1M)	Nein	Nein	Nein	Nein
	RS-232C oder RS-422/RS-485 am Kommunikationsmodul	Peripheriebus	Nein	Nein	Ja	Nein	Nein
		Host-Link (SYSMAC WAY)	Nein	Ja Die WG-, MP- und CR-Befehle werden nicht unterstützt.	Ja Der CR-Befehl wird nicht unterstützt.	Ja Die WG- und MP-Befehle werden nicht unterstützt.	Ja Der CR-Befehl wird nicht unterstützt.
		Seriell Gateway (Konvertierung in CompoWay/F, Modbus-RTU, Modbus-ASCII oder Host Link FINS)	Nein	Ja (Serielle Kommunikationsmodul bzw. -baugruppe ab Baugruppenversion 1.2)	Nein	Nein	Nein
		Protokollfrei	Nein	Ja (Serielle Kommunikationsmodul bzw. -baugruppe ab Baugruppenversion 1.2)	Ja	Nein	Ja
		NT-Link	Nein	Ja	Ja	Nein	Ja (1:1 und 1:N)
		Protokoll-Makro	Nein	Ja	Ja	Nein	Ja
		CompoWay/F-Master	Nein	Ja (mittels Protokoll-Makro)	Ja (mittels Protokoll-Makro)	Nein	Ja (mittels Protokoll-Makro)

Kenndaten		CJ-Serie	CS-Serie	C200HX/HG/HE	CVM1/CV-Serie	CQM1H
Interrupts	E/A-Interrupts	Ja (maximal 2 Interrupt-Eingangs-Baugruppen: 32 Eingänge sowie 4 integrierte Eingänge an CJ1M CPU-Baugruppen) (CJ1 CPU-Baugruppen verfügen über keine integrierten Eingänge.)	Ja (maximal 2 oder 4 Interrupt-Eingangs-Baugruppen: 32 Eingänge)	Ja (maximal 2 Interrupt-Eingangs-Baugruppen: 16 Eingänge)	Ja (maximal 4 Interrupt-Eingangs-Baugruppen: 32 Eingänge)	Ja (4 in CPU-Bus-Baugruppe integriert)
	Zeitgesteuerte Interrupts	Ja	Ja	Ja	Ja	Ja
	Einzelimpuls-Zeitgeber-interrupts	Nein	Nein	Nein	Nein	Ja
	Eingangs-Interrupts im Zählermodus	Ja (nur CJ1M CPU-Baugruppen)	Nein	Nein	Nein	Ja
	Schnelle-Zähler-Interrupts	Ja (nur CJ1M CPU-Baugruppen)	Nein	Nein	Nein	Ja
	Externe Interrupts	Ja (CJ1 CPU-Baugruppen unterstützen keine externen Interrupts.)	Ja	Nein	Nein	Nein
	Vom Kommunikationsmodul	Nein	Ja	Ja	Nein	Nein
	Einschalt-Interrupt	Nein	Nein	Nein	Ja	Nein
	Ausschalt-Interrupt	Ja	Ja	Nein	Ja	Nein
	Interrupt-Ansprechzeit	0,17 ms Integrierte E/A bei CJ1M CPU-Baugruppen: 0,12 ms	C200H Spezial-E/A-Baugruppe: 1 ms E/A-Baugruppen der CJ-Serie: 0,1 ms	1 ms	---	ca. 0,1 ms
SPS-Setup-Bereich		Keine Anwenderadressen (Einstellung nur über Programmiersoftware und Programmierkonsole möglich)	Keine Anwenderadressen (Einstellung nur über Programmiersoftware und Programmierkonsole möglich)	Feste Zuordnung im DM-Bereich: DM 6600 bis DM 6655, DM 6550 bis DM 6559. Einstellung über Programmierkonsole möglich.	Keine Anwenderadressen (Einstellung nur über Programmiersoftware und teilweise über Programmierkonsole möglich)	Feste Zuordnung im DM-Bereich: DM 6600 bis DM 6655. Einstellung über Programmiersoftware und Programmierkonsole möglich.

Kenndaten			CJ-Serie	CS-Serie	C200HX/HG/HE	CVM1/CV-Serie	CQM1H
Konfiguration	E/A	Eingangs-Ansprechzeit bei E/A-Baugruppen	Einstellung im SPS-Setup	Einstellung im SPS-Setup	Nein	Nein	Einstellung im SPS-Setup
		Rack-Anfangsadressen	Einstellung in der E/A-Tabelle über Programmiersoftware (Reihenfolge der Rack-Nummern ist jedoch fest).	Einstellung in der E/A-Tabelle über Programmiersoftware (Reihenfolge der Rack-Nummern ist jedoch fest).	Nein	Einstellung im SPS-Setup (Reihenfolge der Rack-Nummern kann eingestellt werden.)	Nein
		Erste Adresse von SYS-MAC BUS optischen E/A-Baugruppen durch Master	Nein	Nein	Nein	Einstellung im SPS-Setup	Nein
		Betrieb bei E/A-Verifizierungsfehler	Nein	Nein	Nein	Einstellung im SPS-Setup	Nein
	Speicher	Programmspeicher-Schutz	Einstellung durch DIP-Schalter	Einstellung durch DIP-Schalter	Einstellung durch DIP-Schalter	Festlegung durch Schlüsselschaltereinstellung	Einstellung durch DIP-Schalter
		Haltebereiche	Nein	Nein	Nein	Einstellung im SPS-Setup	Nein
		Erhalten von E/A-Worten bei schwerwiegenden Fehlern (außer bei Spannungsausfall)	Nein	Nein	Nein	Einstellung im SPS-Setup	Nein
		Speicherinhalt mittels E/A-Speicher-Haftmarker beim Einschalten der SPS-Spannungsversorgung gespeichert	Einstellung im SPS-Setup	Einstellung im SPS-Setup	Einstellung im SPS-Setup	Einstellung im SPS-Setup	Einstellung im SPS-Setup
		Speicherinhalt mittels Zwangssetzungs-Haftmarker beim Einschalten der SPS-Spannungsversorgung gespeichert	Einstellung im SPS-Setup	Einstellung im SPS-Setup	Einstellung im SPS-Setup	Einstellung im SPS-Setup	Einstellung im SPS-Setup
		Statusüberwachung des DIP-Schalters	Ja	Ja	Ja	Nein	Ja
	Befehle	Einstellung von indirekten DM-Adressen auf BCD- oder Binärformat	Direkte Eingabe möglich	Direkte Eingabe möglich	Nein	Einstellung im SPS-Setup	Nein
		Mehrfachnutzung des JMP(0)-Befehls	Mehrfachnutzung möglich	Mehrfachnutzung möglich	Nein	Einstellung im SPS-Setup	Nein
		Betrieb bei Befehlsfehlern (Fortsetzung oder Stopp)	Einstellung im SPS-Setup	Einstellung im SPS-Setup	Nein	Nein	Nein
		Befehlsausführung im Hintergrund	CJ1: Nein CJ1-H: Ja CJ1M: Ja	CS1: Nein CS1-H: Ja	Nein	Nein	Nein
	Dateispeicher	Automatische Übertragung beim Start	Festlegung erfolgt durch DIP-Schaltereinstellung (wird automatisch von Speicherkarte gelesen)	Festlegung erfolgt durch DIP-Schaltereinstellung (wird automatisch von Speicherkarte gelesen)	Festlegung erfolgt durch DIP-Schaltereinstellung (wird automatisch von Speicherkassette gelesen)	Festlegung im SPS-Setup oder durch DIP-Schaltereinstellung (wird automatisch von Speicherkarte gelesen)	Festlegung erfolgt durch DIP-Schaltereinstellung (wird automatisch von Speicherkarte gelesen)
		Umwandlung in EM-Datei	Einstellung im SPS-Setup	Einstellung im SPS-Setup	Nein	Nein	Nein
	Interrupts	Interrupt-Ansprechzeit-Einstellung	Nein	Nein	Einstellung im SPS-Setup (C200H/Schnelles Ansprechen)	Nein	Nein
		Fehlererkennung	Einstellung im SPS-Setup	Einstellung im SPS-Setup	Einstellung im SPS-Setup	Nein	Nein
		Erhalten von E/A-Interrupts während der E/A-Interrupt-Programmausführung	Nein	Nein	Nein	Einstellung im SPS-Setup	Nein
		Ausschalt-Interrupt aktiviert/deaktiviert	Einstellung im SPS-Setup	Einstellung im SPS-Setup	Nein	Einstellung im SPS-Setup	Nein
		Intervalldauer-Einstellung für zeitgesteuerte Interrupts	Einstellung im SPS-Setup (10 ms, 1,0 ms) (außerdem 0,1 ms nur bei CJ1M CPU-Baugruppe)	Einstellung im SPS-Setup (10 ms, 1,0 ms)	Einstellung im SPS-Setup	Einstellung im SPS-Setup (10 ms, 1 ms, 0,5 ms)	Nein

Kenndaten			CJ-Serie	CS-Serie	C200HX/HG/HE	CVM1/CV-Serie	CQM1H
Konfiguration (Fortsetzung)	Spannungsversorgung	Neustart-Fortsetzungs-Haftmerker	Nein	Nein	Nein	Einstellung im SPS-Setup	Nein
		Betriebsart beim Einschalten	Einstellung im SPS-Setup	Einstellung im SPS-Setup	Einstellung im SPS-Setup	Einstellung im SPS-Setup	Einstellung im SPS-Setup
		Startbedingungen-Einstellungen	CJ1: Nein CJ1-H: Ja CJ1M: Ja	CS1: Nein CS1-H: Ja	Nein	Nein	Nein
		Aufzeichnung beim Start	Nein	Nein	Nein	Einstellung im SPS-Setup	Nein
		Überwachung der niedrigen Batteriespannung	Einstellung im SPS-Setup	Einstellung im SPS-Setup	Einstellung im SPS-Setup	Einstellung im SPS-Setup	Einstellung im SPS-Setup
		Kurzzeitige Spannungsausfallzeit	Nein	Nein	Nein	Einstellung im SPS-Setup	Nein
		Ausschalt-Erkennungsverzögerungszeit	Einstellung im SPS-Setup	Einstellung im SPS-Setup	Einstellung im SPS-Setup (Zeitdauer der Fortsetzung des Betriebs, nachdem das Ausschalten der Spannungsversorgung erkannt wurde)	Nein	Nein
		Kurzzeitiger Spannungsausfall als schwerwiegender/nicht schwerwiegender Fehler	Nein	Nein	Nein	Einstellung im SPS-Setup	Nein
	Zyklen	E/A-Aktualisierung	Nein	Nein	Einstellung im SPS-Setup (nur Spezial-E/A-Baugruppen)	Einstellung im SPS-Setup	Nein
		Konstante Zykluszeit	Einstellung im SPS-Setup (1 bis 32.000 ms)	Einstellung im SPS-Setup (1 bis 32.000 ms)	Einstellung im SPS-Setup (1 bis 9.999 ms)	Einstellung im SPS-Setup (1 bis 32.000 ms)	Einstellung im SPS-Setup (1 bis 9.999 ms)
		Zykluszeit-Überwachung	Einstellung im SPS-Setup (10 bis 40.000 ms) (Voreinstellung: 1.000 ms fest)	Einstellung im SPS-Setup (10 bis 40.000 ms) (Voreinstellung: 1.000 ms fest)	Einstellung im SPS-Setup (0 bis 99) Zeiteinheit: 1 s, 10 ms, 100 ms (Voreinstellung: 120 ms fest)	Einstellung im SPS-Setup (10 bis 40.000 ms) (Voreinstellung: 1.000 ms fest)	Einstellung im SPS-Setup (0 bis 99) Zeiteinheit: 1 s, 10 ms, 100 ms (Voreinstellung: 120 ms fest)
		Deaktivieren der Erkennung der Zykluszeit-Überschreitung	Nein	Nein	Einstellung im SPS-Setup	Nein	Einstellung im SPS-Setup
		Asynchrone Befehlsausführung und Bedienung von Peripheriegeräten	Nein	Nein	Nein	Einstellung im SPS-Setup	Nein
	Serielle Kommunikation	Kommunikationseinstellungen für RS-232C-Schnittstelle	DIP-Schaltereinstellung für automatische Erkennung oder SPS-Setup	DIP-Schaltereinstellung für automatische Erkennung oder SPS-Setup	DIP-Schaltereinstellung für Standardwerte oder SPS-Setup	DIP-Schaltereinstellung für Standardwerte oder SPS-Setup	DIP-Schaltereinstellung für Standardwerte oder SPS-Setup
		Kommunikationseinstellungen für Peripherieschnittstelle	Einstellung im SPS-Setup	Einstellung im SPS-Setup	SPS-Setup	Einstellung durch DIP-Schalter	Einstellung im SPS-Setup
		Kommunikationseinstellungen des Kommunikationsmoduls	Nein	Nein	SPS-Setup	Nein	SPS-Setup
	CPU-Verarbeitungsbedriebsart	Parallelverarbeitungs-Betriebsarten	CJ1: Nein CJ1-H: Ja CJ1M: Nein	CS1: Nein CS1-H: Ja	Nein	Nein	Nein
		Peripheriebedienungs-Prioritätsmodus	Ja	Ja	Nein	Nein	Nein
	Bedienung anderer Peripheriegeräte	Bedienungszeit	Einstellung im SPS-Setup (Feste Peripheriebedienungszeit)	Einstellung im SPS-Setup (Feste Peripheriebedienungszeit)	Einstellung im SPS-Setup (Integrierte RS-232C-Schnittstelle, Kommunikationsmodul, Peripherieschnittstelle)	Nein	Einstellung im SPS-Setup (Integrierte RS-232C-Schnittstelle, Kommunikationsmodul, Peripherieschnittstelle)
		Messen des Bedienungsintervalls der CPU-Bus-Baugruppe	Nein	Nein	Nein	Einstellung im SPS-Setup	Nein
		Stoppen der zyklischen Aktualisierung von Spezial-E/A-Baugruppen	Einstellung im SPS-Setup	Einstellung im SPS-Setup	Einstellung im SPS-Setup	Nein	Nein
		CPU-Bus-Link-Anwendung	Nein	Nein	Nein	Einstellung im SPS-Setup	Nein

Kenndaten			CJ-Serie	CS-Serie	C200HX/HG/HE	CVM1/CV-Serie	CQM1H
Konfiguration (Fortsetzung)	Progr.-m.-konsole	Sprache der Programmierkonsole	Einstellung durch DIP-Schalter	CS1: Einstellung durch DIP-Schalter CS1-H: Einstellung auf der Programmierkonsole	Einstellung durch DIP-Schalter	Nein	Einstellung durch DIP-Schalter
	Fehler	Fehlerprotokollbereich	Nein (fest)	Nein (fest)	Nein (fest: DM 6001 bis DM 6030)	Einstellung im SPS-Setup	Nein (fest: DM 6569 bis DM 6599)
		Kein Registrieren von anwenderdefinierten FAL-Fehlern im Fehlerprotokoll	CJ1: Nein CJ1-H: Ja CJ1M: Ja	CS1: Nein CS1-H: Ja	Nein	Nein	Nein
	Be-trieb	CPU-Standby	Nein	Nein	Nein	Einstellung im SPS-Setup	Nein
Zusatz-System-bereich	Bedin-gungs-merker	ER, CY, <, >, =, Immer-EIN/AUS-Merker usw.	Eingabe mittels Symbolen, z. B. P_ER.	Eingabe mittels Symbolen, z. B. P_ER.	Ja	Ja	Ja
		Taktimpulse	Eingabe mittels Symbolen, z. B. 0,1 s.	Eingabe mittels Symbolen, z. B. 0,1 s.	Ja	Ja	Ja
	Service	Deaktivierungsbit für CPU-Bus-Baugruppen-Service	Nein	Nein	Nein	Ja	Nein
		Codes für verbundene Geräte	Nein	Nein	Nein	Ja	Nein
		Verarbeitungszykluszeiten für Peripherieservice	Nein	Nein	Nein	Ja	Nein
		Service-Intervall der CPU-Bus-Baugruppe	Nein	Nein	Nein	Ja	Nein
		An die CPU angeschlossene Peripherie aktiviert/deaktiviert	Nein	Nein	Nein	Ja	Nein
		Deaktivierungsbit für den Host-Link/NT-Link-Service	Nein	Nein	Nein	Ja	Nein
		Deaktivierungsbit für den Peripherieservice	Nein	Nein	Nein	Ja	Nein
		Deaktivierungsbit für die normale zyklische Aktualisierung	Nein	Nein	Nein	Ja	Nein
		Allgemeiner Überwachungsbereich für internes Modul	Nein	Ja	Ja	Nein	Ja
		Zykluszeit-Überschreitung	Ja	Ja	Ja	Ja	Ja
	Tasks	Task-Erstausführungs-Merker	Ja	Ja	Nein (nur erster Programmzyklus-Merker)	Nein (nur erster Programmzyklus-Merker)	Nein (nur erster Programmzyklus-Merker)
	Debug-ging	Merker für deaktivierte Online-Bearbeitung	Ja	Ja	Ja (AR)	Nein	Nein
		Standby-Merker für die Online-Bearbeitung	Ja	Ja	Ja (AR)	Nein	Nein
		Ausgang-AUS-Bit	Ja	Ja	Ja	Ja	Ja
		Zwangssetzungs-Haftmerker	Ja	Ja	Ja	Ja	Ja
	Datei-spei-cher	Dateispeicher-Befehlsmerker	Ja	Ja	Nein	Ja	Nein
		EM-Dateispeicher-Formatierungsfehlermerker	Ja (außer bei CJ1M CPU-Baugruppen)	Ja	Nein	Nein	Nein
		EM-Datei-Format-Anfangsbank	Ja (außer bei CJ1M CPU-Baugruppen)	Ja	Nein	Nein	Nein
	Spei-cher	DIP-Schalter-Statusmerker	Ja (Schalter 6)	Ja (Schalter 6)	Ja (AR, nur Schalter 6)	Nein	Ja (AR, Schalter 6)
		E/A-Speicher-Haftmerker	Ja	Ja	Ja	Ja	Ja
	Inter-rupts	Max. Verarbeitungszeit für Unterprogramme/Interrupt-Task	Ja	Ja	Ja	Nein	Nein
		Interrupt-Task-Fehlermerker	Ja	Ja	Ja	Nein	Nein

Kenndaten			CJ-Serie	CS-Serie	C200HX/HG/HE	CVM1/CV-Serie	CQM1H
Zusatz-Systembereich, Fortsetzung	Fehler	Bereich/Zeiger für Fehlerprotokollspeicher	Ja	Ja	Nein	Ja	Nein
		Fehlercodes	Ja	Ja	Ja	Ja	Ja
	Grundeinstellungen	Initialisierung des SPS-Setups	Nein	Nein	Ja	Nein	Ja
	Kommunikation	SPS-Link Betriebsbeben-Merkern	Ja (SPS-Link Zusatz-Systembereichsbit)	Ja (SPS-Link Zusatz-Systembereichsbit)	Ja (AR)	Nein	Nein
	Spannungsversorgung	Spannungsausfall-Merker	Nein	Nein	Nein	Ja	Nein
		Ausschaltzeitpunkt	Nein	Nein	Nein	Ja	Nein
		Einschaltzeitpunkt der Versorgungsspannung	Ja	Ja	Nein	Ja	Nein
		Zeitpunkt des Spannungsausfalls (einschließlich Ausschalten)	Ja	Ja	Nein	Ja	Ja
		Anzahl der kurzzeitigen Spannungsausfälle	Ja (Anzahl der Spannungsausfälle)	Ja (Anzahl der Spannungsausfälle)	Ja (Anzahl der Spannungsausfälle)	Ja	Ja (Anzahl der Spannungsausfälle)
		Gesamteinschaltdauer	Ja	Ja	Nein	Nein	Nein
Speicheradressenzuordnung		Format	Die Zuordnung basiert auf der von den Baugruppen benötigten Wortanzahl entsprechend der Reihenfolge.	Die Zuordnung basiert auf der von den Baugruppen benötigten Wortanzahl. Freie Steckplätze werden übersprungen.	Festgelegte Zuordnung von Worten: Jeder Baugruppe wird automatisch ein Wort zugeordnet.	Die Zuordnung basiert auf der von den Baugruppen benötigten Wortanzahl. Freie Steckplätze werden übersprungen.	Die Zuordnung basiert auf der von den Baugruppen benötigten Wortanzahl entsprechend der Reihenfolge.
		Zuordnung bei Multi-point-E/A-Baugruppen der Gruppe 2	Keine	Wie bei E/A-Baugruppen	Gruppe-2-Zuordnungsbereich im IR-Bereich (Position wird durch Gerätefrontschalter bestimmt)	Keine	Keine
		Wortreservierungsmethode	Ändern der E/A-Tabelle mit CX-Programmer	Ändern der E/A-Tabelle mit CX-Programmer	Erstellen einer E/A-Tabelle mit leerem Steckplatz oder Ändern der E/A-Tabelle mit CX-Programmer	Dummy-E/A-Baugruppe oder Ändern der E/A-Tabelle mit CX-Programmer	Automatische Zuordnung beim Start
Zuordnung bei Spezial-E/A-Baugruppen	CIO-Bereich	Zuordnung im Bereich für Spezial-E/A-Baugruppen entsprechend der Baugruppennummer; 10 Worte pro Baugruppe bei insgesamt 96 Baugruppen.	Zuordnung im Bereich für Spezial-E/A-Baugruppen entsprechend der Baugruppennummer; 10 Worte pro Baugruppe bei insgesamt 96 Baugruppen.	Zuordnung im Bereich für Spezial-E/A-Baugruppen (im IR-Bereich) entsprechend der Baugruppennummer; 10 Worte pro Baugruppe bei insgesamt 16 Baugruppen.	Wie bei E/A-Baugruppen; 2 oder 4 Worte im E/A-Bereich zugeordnet (für jede Baugruppe unterschiedlich)	Wie bei E/A-Baugruppen; 1, 2 oder 4 Worte im E/A-Bereich zugeordnet (für jede Baugruppe unterschiedlich)	
	DM-Bereich	Zuordnung im Bereich zwischen D20000 und D29599 gemäß Baugruppennummer, 100 Worte pro Baugruppe bei insgesamt 96 Baugruppen.	Zuordnung im Bereich zwischen D20000 und D29599 gemäß Baugruppennummer, 100 Worte pro Baugruppe bei insgesamt 96 Baugruppen.	Zuordnung im Bereich zwischen DM 1000 und DM 1999 sowie DM 2000 bis DM 2599; 100 Worte pro Baugruppe bei insgesamt 16 Baugruppen.	Keine	Keine	
CPU-Bus-Baugruppe/CPU-Bus-Baugruppenzuordnung	CIO-Bereich	Zuordnung im CPU-Bus-Baugruppenbereich entsprechend der Baugruppennummer; 25 Worte pro Baugruppe bei insgesamt 16 Baugruppen.	Zuordnung im CPU-Bus-Baugruppenbereich entsprechend der Baugruppennummer; 25 Worte pro Baugruppe bei insgesamt 16 Baugruppen.	Keine	Zuordnung im CPU-Bus-Baugruppenbereich entsprechend der Baugruppennummer; 25 Worte pro Baugruppe bei insgesamt 16 Baugruppen.	Keine	
	DM-Bereich	Zuordnung im Bereich zwischen D30000 und D31599 gemäß Baugruppennummer, 100 Worte pro Baugruppe bei insgesamt 16 Baugruppen.	Zuordnung im Bereich zwischen D30000 und D31599 gemäß Baugruppennummer, 100 Worte pro Baugruppe bei insgesamt 16 Baugruppen.	Keine	Zuordnung im Bereich zwischen D02000 und D03599 gemäß Baugruppennummer, 100 Worte pro Baugruppe bei insgesamt 16 Baugruppen.	Keine	

Kenndaten		CJ-Serie	CS-Serie	C200HX/HG/HE	CVM1/CV-Serie	CQM1H
E/A-Speicher	CIO-Bereich	Ja	Ja	Ja	Ja	Ja
	WR-Bereich	Ja	Ja	Nein	Nein	Nein
	Temporärer Relaisbereich	Ja	Ja	Ja	Ja	Ja
	Zusatz-Systembereich	Ja	Ja	Ja	Ja	Ja
	SR-Bereich	Nein	Nein	Ja	Nein	Ja
	Link-Bereich (Schnittstellenmerker)	Ja (Data-Link-Bereich)	Ja (Data-Link-Bereich)	Ja (Data-Link-Bereich)	Nein	Ja
	Bereich für C200H Spezial-E/A-Baugruppen	Ja	Ja	Ja (CIO-Bereich)	Nein	Nein
	Bereich für die integrierten E/A	Ja (nur CJ1M CPU-Baugruppe mit integrierten E/A)	Nein	Nein	Nein	Nein
	Serieller-SPS-Link-Bereich	Ja (nur CJ1M CPU-Baugruppen)	Nein	Nein	Nein	Nein
	DM-Bereich	Ja	Ja	Ja	Ja	Ja
	Erweiterter Datenspeicherbereich (EM)	Ja (Adressen einschließlich Bank-Nr. können angegeben werden) (nicht von CJ1M CPU-Baugruppe unterstützt)	Ja (Adressen einschließlich Bank-Nr. können angegeben werden)	Ja (Adressen können für -Z angegeben werden; Bänke jedoch nicht)	Ja (Adresse einschließlich Bank können nicht angegeben werden; Bank muss geändert werden. EM-Datenspeicher-Erweiterung erforderlich.)	Ja (keine Bänke)
	Zeitgeber-/Zählerbereich	Ja	Ja	Ja	Ja	Ja
	Indexregister	Ja	Ja	Nein	Ja	Nein
	Datenregister	Ja	Ja	Nein	Ja	Nein
	Bereiche für das zwangsweise Setzen/Rücksetzen	CIO-Bereich	Ja	Ja	Ja	Ja
		WR-Bereich	Ja	Nein	Nein	Kein
		Haftmerkerbereich	Ja	Ja	Nein	Nein
		Zusatz-Systembereich	Nein	Ja	Nein	Ja
		SR-Bereich	Nein	Nein	Nein	Nein
		Link-Bereich	Nein	Ja	Nein	Nein
		Zeitgeber/Zähler	Ja (Merker)	Ja (Merker)	Ja (Merker)	Ja (Merker)
		DM-Bereich	Nein	Nein	Nein	Nein
		EM-Bereich	Nein	Nein	Nein	Nein
Befehlsvarianten/indirekte Adressen	Ausführung bei steigender Flanke (einmalig ausgeführt)		Ja (durch @ spezifiziert)	Ja (durch @ spezifiziert)	Ja (durch @ spezifiziert)	Ja (durch @ spezifiziert)
	Ausführung bei fallender Flanke (einmalig ausgeführt)		Ja (durch % spezifiziert)	Ja (durch % spezifiziert)	Nein (stattdessen DIFD-Befehl verwendet)	Ja (durch ↓ spezifiziert)
	Direkte Aktualisierung		Ja (durch ! spezifiziert)	Ja (durch ! spezifiziert)	Nein (stattdessen IORF-Befehl verwendet)	Ja (durch ! spezifiziert)
	Indirekte Adressierung für DM/EM	BCD-Modus	Ja (0000 bis 9999) Sternchen wird verwendet.	Ja (0000 bis 9999) Sternchen wird verwendet.	Ja (0 bis 9999)	Ja (0 bis 9999)
		Binärmodus	Ja (00000 bis 32767) @ wird verwendet. 0000 bis 7FFF Hex: 0000 bis 32767 8000 bis FFFF Hex: 00000 bis 32767 in nächster Bank	Ja (00000 bis 32767) @ wird verwendet. 0000 bis 7FFF Hex: 0000 bis 32767 8000 bis FFFF Hex: 00000 bis 32767 in nächster Bank	Nein	Ja, zur indirekten Adressierung aller SPS-Speicheradressen.
Einstellungen für Startadressen	Einstellung des ersten Worts eines Racks		Ja (bei allen CPU-Baugruppen)	Ja (bei allen CPU-Baugruppen)	Nein	Nein
	Einstellung des ersten Worts eines Steckplatzes		CPU-Baugruppen vor Version 2.0: Siehe Hinweis. CPU-Baugruppen Version 2.0 oder später: Ja	CPU-Baugruppen vor Version 2.0: Siehe Hinweis. CPU-Baugruppen Version 2.0 oder später: Ja	Nein	Nein

Kenndaten		CJ-Serie	CS-Serie	C200HX/HG/HE	CVM1/CV-Serie	CQM1H
Online-Verbindungen über Netzwerke ohne Erstellung von E/A-Tabellen		Mit automatischer E/A-Zuordnung beim Einschalten: Ja (bei allen CPU-Baugruppen) Anwenderdefinierte E/A-Zuordnung CPU-Baugruppen vor Version 2.0: Nein CPU-Baugruppen Version 2.0 oder später: Ja	CPU-Baugruppen vor Version 2.0: Nein CPU-Baugruppen Version 2.0 oder später: Ja	Nein	Nein	Ja, aber nur für Controller-Link
Arbeiten über mehrere Netzwerkebenen hinweg		CPU-Baugruppen vor Version 2.0: 3 Ebenen CPU-Baugruppen Version 2.0 oder später: 8 Ebenen	CPU-Baugruppen vor Version 2.0: 3 Ebenen CPU-Baugruppen Version 2.0 oder später: 8 Ebenen	Nein	Ja, 3 Ebenen	Nein
Online-Verbindungen zu SPS über NS-Terminals		CPU-Baugruppen vor Version 2.0: Nein CPU-Baugruppen Version 2.0 oder später: Ja	CPU-Baugruppen vor Version 2.0: Nein CPU-Baugruppen Version 2.0 oder später: Ja	Nein	Nein	Nein
Schutz von CPU-Baugruppen gegen FINS-Schreibbefehle, die über Netzwerke gesendet werden		CPU-Baugruppen vor Version 2.0: Nein CPU-Baugruppen Version 2.0 oder später: Ja	CPU-Baugruppen vor Version 2.0: Nein CPU-Baugruppen Version 2.0 oder später: Ja	Nein	Nein	Nein
Herunterladen einzelner Tasks		CPU-Baugruppen vor Version 2.0: Nein CPU-Baugruppen Version 2.0 oder später: Ja	CPU-Baugruppen vor Version 2.0: Nein CPU-Baugruppen Version 2.0 oder später: Ja	Nein	Nein	Nein
Leseschutz durch Passwörter	Gesamtes Anwenderprogramm	CPU-Baugruppen vor Version 2.0: Nein CPU-Baugruppen Version 2.0 oder später: Ja	CPU-Baugruppen vor Version 2.0: Nein CPU-Baugruppen Version 2.0 oder später: Ja	Nein	Nein	Nein
	Spezifizierte Tasks	CPU-Baugruppen vor Version 2.0: Nein CPU-Baugruppen Version 2.0 oder später: Ja	CPU-Baugruppen vor Version 2.0: Nein CPU-Baugruppen Version 2.0 oder später: Ja	Nein	Nein	Nein
	Aktivieren/Deaktivieren der Erstellung von Dateispeicherprogrammdateien	CPU-Baugruppen vor Version 2.0: Nein CPU-Baugruppen Version 2.0 oder später: Ja	CPU-Baugruppen vor Version 2.0: Nein CPU-Baugruppen Version 2.0 oder später: Ja	Nein	Nein	Nein
	Programmschreibschutz	CPU-Baugruppen vor Version 2.0: Nein CPU-Baugruppen Version 2.0 oder später: Ja	CPU-Baugruppen vor Version 2.0: Nein CPU-Baugruppen Version 2.0 oder später: Ja	Nein	Nein	Nein
Automatische Übertragung beim Einschalten ohne Parameterdatei (.STD)		CPU-Baugruppen vor Version 2.0: Nein CPU-Baugruppen Version 2.0 oder später: Ja	CPU-Baugruppen vor Version 2.0: Nein CPU-Baugruppen Version 2.0 oder später: Ja	Nein	Nein	Nein

Hinweis Bei CPU-Baugruppen, die ab dem 1. Juni 2002 (Lot-Nummer 020601□□□□ oder neuer) hergestellt wurden, können bis zu acht Steckplatzadressen festgelegt werden.

Befehlsvergleich

Gruppe/Befehl		AWL	CJ-Serie	CS-Serie	C200HX/HG/HE	CVM1/CV-Serie	CQM1H
Sequenzzei- gabebefehle	LOAD/AND/OR (Laden/ Und/Oder)	LD/AND/ OR	Ja	Ja	Ja	Ja	Ja
	AND LOAD/OR LOAD	AND LD/ OR LD	Ja	Ja	Ja	Ja	Ja
	NOT	NOT	Ja	Ja	Ja	Ja	Nein
	CONDITION ON (Steigende Flanke)	UP	Ja	Ja	Nein	Ja (*1)	Nein
	CONDITION OFF (Fallende Flanke)	DOWN	Ja	Ja	Nein	Ja (*1)	Nein
	BIT TEST	TST/ TSTN	Ja (Bitposition im Binärformat spezifiziert: 0000 bis 000F Hex)	Ja (Bitposition im Binärformat spezifiziert: 0000 bis 000F Hex)	Ja (Bitposition im BCD-Format spezifiziert) (*2)	Ja (Bitposition im BCD-Format spezifiziert) (*1)	Nein
Sequenzaus- gabebefehle	OUTPUT (Ausgang)	OUT	Ja	Ja	Ja	Ja	Ja
	TR	TR (Tem- porärer Merker)	Ja	Ja	Ja	Ja	Ja
	KEEP (Bistabiler Merker)	KEEP	Ja	Ja	Ja	Ja	Ja
	DIFFERENTIATE UP/ DOWN (Ausführung bei steigender/fallender Flanke)	DIFU/ DIFD	Ja (LD↑, AND↑, OR↑) (LD↓ AND↓, OR↓)	Ja (LD↑, AND↑, OR↑) (LD↓ AND↓, OR↓)	Ja (DIFU/DIFD)	Ja (LD↑, AND↑, OR↑) (LD↓, AND↓, OR↓)	Ja (DIFU/DIFD)
	SET und RESET (Set- zen und Rücksetzen)	SET/ RSET	Ja	Ja	Ja	Ja	Ja
	MULTIPLE BIT SET/ RESET (Mehrfaches Bit- Setzen/Rücksetzen)	SETA/ RSTA	Ja (Startbit und Anzahl der Bits im Binärformat spezi- fiziert.)	Ja (Startbit und Anzahl der Bits im Binärformat spezi- fiziert.)	Nein	(*1) (Startbit und Anzahl der Bits im BCD-Format spezifiziert.)	Nein
	SINGLE BIT SET/ RESET (Bit Setzen/ Rücksetzen)	SET/ RSTB	CJ1: Nein CJ1-H: Ja CJ1M: Ja	CS1: Nein CS1-H: Ja	Nein	Nein	Nein
	SINGLE BIT OUTPUT (Bit-Ausgabe)	OUTB	CJ1: Nein CJ1-H: Ja CJ1M: Ja	CS1: Nein CS1-H: Ja	Nein	Nein	Nein
Sequenz- steuer- befehle	END/NO OPERATION (Ende/ kein Vorgang)	END/ NOP	Ja	Ja	Ja	Ja	Ja
	INTERLOCK/INTER- LOCK CLEAR (Verriegelung und Verriegelung löschen)	IL/ILC	Ja	Ja	Ja	Ja	Ja
	Mehrfachverriegelung	MILH/ MILR/ MILC	CPU-Baugruppen vor Version 2.0: Nein CPU-Baugruppen Version 2.0 oder später: Ja	CPU-Baugruppen vor Version 2.0: Nein CPU-Baugruppen Version 2.0 oder später: Ja	Nein	Nein	Nein
	JUMP/JUMP END (Sprung/ Sprungende)	JMP/ JME	Ja (Jump-Nummer im BCD-Format spezifiziert: 0 bis 1023)	Ja (Jump-Nummer im BCD-Format spezifiziert: 0 bis 1023)	Ja (Jump-Nummer im BCD-Format spezifiziert: 0 bis 99)	Ja (Jump-Num- mer im BCD-For- mat spezifiziert: 0 bis 999)	Ja (Jump-Num- mer im BCD-For- mat spezifi- ziert: 0 bis 99)
	CONDITIONAL JUMP (Bedingter Sprung)	CJP/ CJPN	Ja (Jump-Nummer im BCD-Format spezifiziert: 0 bis 1023)	Ja (Jump-Nummer im BCD-Format spezifiziert: 0 bis 1023)	Nein	Ja (Jump-Num- mer im BCD-For- mat spezifiziert: 0 bis 999) (*1)	Nein
	MULTIPLE JUMP/ JUMP END (Mehrfach- Sprung/Mehrfach- Sprungende)	JMP0/ JME0	Ja	Ja	Nein	Nein (das SPS- Setup kann jedoch so einge- richtet werden, dass mehrere Jumps mit Jump- Nummer 0 mög- lich sind)	Nein
	FOR/NEXT LOOPS	FOR/ NEXT- Schleife	Ja	Ja	Nein	Nein	Nein
	BREAK LOOP (FOR/NEXT-Schleife vorzeitig verlassen)	BREAK	Ja	Ja	Nein	Nein	Nein

Gruppe/Befehl		AWL	CJ-Serie	CS-Serie	C200HX/HG/HE	CVM1/CV-Serie	CQM1H
Zeitgeber- und Zählerbefehle	TIMER	TIM (BCD)	Ja	Ja	Ja	Ja	Ja
		TIMX (Binär)	Ja (*4)	Ja (*4)	Nein	Nein	Nein
	HIGH-SPEED TIMER (Schneller Zeitgeber)	TIMH (BCD)	Ja	Ja	Ja	Ja	Ja
		TIMHX (Binär)	Ja (*4)	Ja (*4)	Nein	Nein	Nein
	ONE-MS TIMER (Ultraschneller Zeitgeber)	TMHH (BCD)	Ja	Ja	Nein	Nein	Nein
		TMHHX (Binär)	Ja (*4)	Ja (*4)	Nein	Nein	Nein
	ACCUMULATIVE TIMER (Summierender Zeitgeber)	TTIM (BCD)	Ja	Ja	Ja	Ja	Ja
		TTIMX (Binär)	Ja (*4)	Ja (*4)	Nein	Nein	Nein
	LONG TIMER (Langer Zeitgeber)	TIML (BCD)	Ja	Ja	Nein	Ja	Nein
		TIMLX (Binär)	Ja (*4)	Ja (*4)	Nein	Nein	Nein
	MULTI-OUTPUT TIMER (Multi-Ausgangs-Zeitgeber)	MTIM (BCD)	Ja	Ja	Nein	Ja	Nein
		MTIMX (Binär)	Ja (*4)	Ja (*4)	Nein	Nein	Nein
	COUNTER (Zähler)	CNT (BCD)	Ja	Ja	Ja	Ja	Ja
		CNTX (Binär)	Ja (*4)	Ja (*4)	Nein	Nein	Nein
	REVERSIBLE COUNTER (Umkehrbarer Zähler)	CNTR (BCD)	Ja	Ja	Ja	Ja	Ja
		CNTRX (Binär)	Ja (*4)	Ja (*4)	Nein	Nein	Nein
	RESET TIMER/COUNTER (Zeitgeber/Zähler rücksetzen)	CNR (BCD)	Ja (Setzt nur Zeitgeber oder Zähler zurück.)	Ja (Setzt nur Zeitgeber oder Zähler zurück.)	Nein	Ja (Setzt außerdem den im E/A-Bereich festgelegten Bereich auf Null.)	Nein
		CNRX (Binär)	Ja (*4)	Ja (*4)	Nein	Nein	Nein

Gruppe/Befehl		AWL	CJ-Serie	CS-Serie	C200HX/HG/HE	CVM1/CV-Serie	CQM1H
Vergleichs-befehle	Vergleichs-Zwischenbefehle	=, < usw.	Ja (Alle werden für LD, OR und AND unterstützt)	Ja (Alle werden für LD, OR und AND unterstützt)	Ja (*2) (nur für AND unterstützt)	Ja (*1) (nur für AND unterstützt)	Nein
	Kalender-Vergleichs-befehle	=DT, <DT usw.	CPU-Baugruppen vor Version 2.0: Nein CPU-Baugruppen Version 2.0 oder später: Ja	CPU-Baugruppen vor Version 2.0: Nein CPU-Baugruppen Version 2.0 oder später: Ja	Nein	Nein	Nein
	COMPARE/ DOUBLE COMPARE (Vergleich/ Doppelwortvergleich)	CMP/ CMPL	Ja	Ja	Ja	Ja (*3)	Ja
	SIGNED BINARY COMPARE (Vorzeichenbehafteter Binärvergleich) / DOUBLE SIGNED BINARY COMPARE (Vorzeichenbehafteter Doppelwortvergleich)	CPS/ CPSL	Ja	Ja	Ja	Ja (*1)	Ja
	BLOCK COMPARE (Blockvergleich)	BCMP	Ja	Ja	Ja	Ja	Ja
	EXTENDED BLOCK COMPARE (Erweiterter Blockvergleich)	BCMP2	Ja (nur CJ1M CPU-Baugruppen und CJ1-H CPU- Baugruppen Version 2.0)	Ja (nur CS1-H/ CS1D CPU-Bau- gruppen Version 2.0)	Nein	Nein	Nein
	TABLE COMPARE (Tabellen vergleichen)	TCMP	Ja	Ja	Ja	Ja	Ja
	MULTIPLE COMPARE (Mehrfa- ches Vergleichen)	MCMP	Ja	Ja	Ja	Ja	Ja
	EQUALS	EQU (Gleich)	Nein	Nein	Nein	Ja	Nein
	AREA RANGE COM- PARE (Vergleichen mit Bereich)	ZCP/ ZCPL	CJ1: Nein (wird durch Verwen- dung von Ver- gleichsbefehlen erreicht) CJ1-H: Ja CJ1M: Ja	CS1: Nein (wird durch Verwendung von Vergleichsbefeh- len erreicht) CS1-H: Ja	Ja	Nein	Nein (wird durch Verwendung von Vergleichsbefeh- len erreicht)

Gruppe/Befehl		AWL	CJ-Serie	CS-Serie	C200HX/HG/HE	CVM1/CV-Serie	CQM1H
Kopierbefehle	MOVE (Übertragen)	MOV	Ja	Ja	Ja	Ja	Ja
	DOUBLE MOVE (Doppelwort übertragen)	MOVL	Ja	Ja	Nein	Ja	Nein
	MOVE NOT (Übertragen invertiert)	MVN	Ja	Ja	Ja	Ja	Ja
	DOUBLE MOVE (Doppelwort übertragen invertiert)	MVNL	Ja	Ja	Nein	Ja	Nein
	DATA EXCHANGE (Datenaustausch)	XCHG	Ja	Ja	Ja	Ja	Ja
	DOUBLE DATA EXCHANGE (Doppelwort-Datenaustausch)	XCGL	Ja	Ja	Nein	Ja	Nein
	MOVE QUICK (Schnelles Übertragen)	MOVQ	Nein	Nein	Nein	Ja	Nein
	BLOCK TRANSFER (Blockübertragung)	XFER	Ja (Anzahl der zu übertragenden Worte im Binärformat spezifiziert: 0 bis 65535)	Ja (Anzahl der zu übertragenden Worte im Binärformat spezifiziert: 0 bis 65535)	Ja (Anzahl der zu übertragenden Worte in BCD spezifiziert: 0 bis 6144)	Ja (Anzahl der zu übertragenden Worte in BCD spezifiziert: 0 bis 9999)	Ja (Anzahl der zu übertragenden Worte in BCD spezifiziert: 0 bis 9999)
	BLOCK SET (Block setzen)	BSET	Ja	Ja	Ja	Ja	Ja
	MOVE BIT (Bit übertragen)	MOVB	Ja (Quellbit- und Zielbit-Position im Binärformat spezifiziert)	Ja (Quellbit- und Zielbit-Position im Binärformat spezifiziert)	Ja (Quellbit- und Zielbit-Position in BCD spezifiziert)	Ja (Quellbit- und Zielbit-Position in BCD spezifiziert)	Ja (Quellbit- und Zielbit-Position in BCD spezifiziert)
	MULTIPLE BIT TRANSFER (Mehrere Bits übertragen)	XFRB	Ja	Ja	Ja	Ja (*1)	Ja
	MOVE DIGIT (Stelle übertragen)	MOVD	Ja	Ja	Ja	Ja	Ja
	SINGLE WORD DISTRIBUTE (Wort mit Offset übertragen)	DIST	Ja (Stapelverarbeitungsfunktion ist mit einem anderen Befehl möglich. Offset-Wert im Binärformat spezifiziert: 0 bis 65535)	Ja (Stapelverarbeitungsfunktion ist mit einem anderen Befehl möglich. Offset-Wert im Binärformat spezifiziert: 0 bis 65535)	Ja (Stapelverarbeitungsfunktion ist möglich. Offset-Wert in BCD spezifiziert: 0 bis 8999)	Ja (Stapelverarbeitungsfunktion ist mit einem anderen Befehl möglich. Offset-Wert in BCD spezifiziert: 0 bis 9999)	Ja (Stapelverarbeitungsfunktion ist möglich. Offset-Wert in BCD spezifiziert: 0 bis 8999)
	DATA COLLECT (Daten mit Offset sammeln)	COLL	Ja (Stapelverarbeitungsfunktion ist mit einem anderen Befehl möglich. Offset-Wert im Binärformat spezifiziert: 0 bis 65535)	Ja (Stapelverarbeitungsfunktion ist mit einem anderen Befehl möglich. Offset-Wert im Binärformat spezifiziert: 0 bis 65535)	Ja (Stapelverarbeitungsfunktion ist möglich. Offset-Wert in BCD spezifiziert: 0 bis 7999)	Ja (Stapelverarbeitungsfunktion ist mit einem anderen Befehl möglich. Offset-Wert in BCD spezifiziert: 0 bis 9999)	Ja (Stapelverarbeitungsfunktion ist möglich. Offset-Wert in BCD spezifiziert: 0 bis 7999)
	EM BLOCK TRANSFER BETWEEN BANKS (Interbank Blockübertragung)	BXFR	Nein (Funktionell möglich für bis zu 65.535 Worte durch direktes Adressieren des EM-Bereichs mittels XFER)	Nein (Funktionell möglich für bis zu 65.535 Worte durch direktes Adressieren des EM-Bereichs mittels XFER)	Nein	Ja (*1)	Nein
	EM BLOCK TRANSFER (EM-Blockweise Übertragung)	XFR2	Nein	Nein	Ja	Nein	Nein
	EM BANK TRANSFER (Blockweises Übertragen zwischen den EM-Bänken)	BXF2	Nein	Nein	Ja	Nein	Nein
	MOVE TO REGISTER (Zum Register übertragen)	MOVR	Ja (Für indirektes DM/EM wird keine Adresse spezifiziert.)	Ja (Für indirektes DM/EM wird keine Adresse spezifiziert.)	Nein	Ja (Für indirektes EM/DM wird eine Adresse spezifiziert.)	Nein
	MOVE TIMER/COUNTER PV TO REGISTER (Zeitgeber/Zähler-Istwert zum Register übertragen)	MOVRW	Ja	Ja	Nein	Nein (nur für Abschluss-Merker unter Verwendung von MOVR möglich)	Nein

Gruppe/Befehl	AWL	CJ-Serie	CS-Serie	C200HX/HG/HE	CVM1/CV-Serie	CQM1H
Datenverschiebebefehle	SHIFT REGISTER (Schieberegister)	SFT	Ja	Ja	Ja	Ja
	REVERSIBLE SHIFT REGISTER (Umkehrbares Schieberegister)	SFTR	Ja	Ja	Ja	Ja
	ASYNCHRONOUS SHIFT REGISTER (Asynchrones Schieberegister)	ASFT	Ja	Ja	Ja	Ja
	WORD SHIFT (Wortverschiebung)	WSFT	Ja (Wie bei CV: 3 Operanden)	Ja (Wie bei CV: 3 Operanden)	Ja	Ja
	ARITHMETIC SHIFT LEFT/ ARITHMETIC SHIFT RIGHT (Arithmetisches Verschieben nach links/nach rechts)	ASL/ ASR	Ja	Ja	Ja	Ja
	ROTATE LEFT/ROTATE RIGHT (Rotation nach links/nach rechts)	ROL/ ROR	Ja	Ja	Ja	Ja
	ONE DIGIT SHIFT LEFT/ONE DIGIT SHIFT RIGHT (Eine Stelle nach links/nach rechts verschieben)	SLD/ SRD	Ja	Ja	Ja	Ja
	SHIFT N-BIT DATA LEFT/SHIFT N-BIT DATA RIGHT (N-Datenbits nach links/nach rechts verschieben)	NSFR/ NSFL	Ja (Schiebedaten und Startbit im Binärformat spezifiziert.)	Ja (Schiebedaten und Startbit im Binärformat spezifiziert.)	Nein	Ja (Schiebedaten und Startbit in BCD spezifiziert.) (*1)
	SHIFT N-BITS LEFT/SHIFT N-BITS RIGHT, DOUBLE SHIFT N-BITS LEFT/DOUBLE SHIFT N-BITS RIGHT (N-Datenbits nach links/nach rechts verschieben, N-Datenbits-Doppelwort nach links/nach rechts verschieben)	NASL/ NASR, NSLL/ NSRL	Ja (Anzahl der zu verschiebenden Bits im Binärformat spezifiziert.)	Ja (Anzahl der zu verschiebenden Bits im Binärformat spezifiziert.)	Nein	Ja (Anzahl der zu verschiebenden Bits in BCD spezifiziert.) (*1)
	DOUBLE SHIFT LEFT/DOUBLE SHIFT RIGHT (Doppelwort verschieben nach links/nach rechts)	ASLL/ ASRL	Ja	Ja	Nein	Ja
	DOUBLE ROTATE LEFT/DOUBLE ROTATE RIGHT (Doppelwort-Rotation nach links/nach rechts)	ROLL/ RORL	Ja	Ja	Nein	Ja
	ROTATE LEFT WITHOUT CARRY/ROTATE RIGHT WITHOUT CARRY/DOUBLE ROTATE LEFT WITHOUT CARRY/DOUBLE ROTATE RIGHT WITHOUT CARRY (Rotation nach links/nach rechts ohne Übertrag, Doppelwort-Rotation nach links/nach rechts ohne Übertrag)	RLNC/ RRNC, RLNL/ RRNL	Ja	Ja	Nein	Ja (*1)
	INCREMENT BCD/DECREMENT BCD (BCD-Wert inkrementieren/dekrementieren)	++B/--B (INC/DEC)	Ja (++B/--B)	Ja (++B/--B)	Ja (INC/DEC)	Ja (INC/DEC)
Inkrementier- und Dekrementierbefehle	DOUBLE INCREMENT BCD/DOUBLE DECREMENT BCD (BCD-Doppelwort inkrementieren/dekrementieren)	++BL/--BL (INCL/DECL)	Ja (++BL/--BL)	Ja (++BL/--BL)	Nein	Ja (INCL/DECL)
	INCREMENT BINARY/DECREMENT BINARY (Binärwert inkrementieren/dekrementieren)	++/- - (INCB/DECB)	Ja (CY wird bei einem Übertrag oder negativen Übertrag auf EIN gesetzt) (++/- -)	Ja (CY wird bei einem Übertrag oder negativen Übertrag auf EIN gesetzt) (++/- -)	Nein	Ja
	DOUBLE INCREMENT BINARY/DOUBLE DECREMENT BINARY (Binär-Doppelwort inkrementieren/dekrementieren)	++L/- -L (INBL/DCBL)	Ja (CY wird bei einem Übertrag oder negativen Übertrag auf EIN gesetzt) (++L/- -L)	Ja (CY wird bei einem Übertrag oder negativen Übertrag auf EIN gesetzt) (++L/- -L)	Nein	Ja
	Arithmetische Befehle		Ja	Ja	Ja	Ja

Gruppe/Befehl		AWL	CJ-Serie	CS-Serie	C200HX/HG/HE	CVM1/CV-Serie	CQM1H
Konvertierungs-befehle	BCD-TO-BINARY/ DOUBLE BCD-TO- DOUBLE BINARY (BCD zu Binär/ BCD- Doppelwort zu Binär)	BIN/BINL	Ja	Ja	Ja	Ja	Ja
	BINARY-TO-BCD/ DOUBLE BINARY-TO- DOUBLE BCD (Binär zu BCD / Binär-Doppel- wort zu BCD-Doppel- wort)	BCD/ BCDL	Ja	Ja	Ja	Ja	Ja
	2'S COMPLEMENT/ DOUBLE 2'S COM- PLEMENT (2er-Kom- plement/Doppelwort- 2er-Komplement)	NEG/ NEGL	Ja (Wie bei CV, jedoch wird der Unterlauf-Merker P_UF bei 8000 Hex an der Quelle nicht auf EIN gesetzt)	Ja (Wie bei CV, jedoch wird der Unterlauf-Merker P_UF bei 8000 Hex an der Quelle nicht auf EIN gesetzt)	Ja	Ja	Ja
	16-BIT TO 32-BIT SIGNED BINARY (16- zu 32-Bit vorzeichenbe- haftete Binärwerte)	SIGN	Ja	Ja	Nein	Ja	Nein
	DATA DECODER (Daten-Dekoder)	MLPX	Ja	Ja	Ja	Ja	Ja
	DATA ENCODER (Daten-Encoder)	DMPX	Ja (Wie bei CVM1-V2: Kann das rechte einge- schaltete Bit bei EIN spezifizieren.)	Ja (Wie bei CVM1-V2: Kann das rechte einge- schaltete Bit bei EIN spezifizieren.)	Ja (Nur das linke eingeschaltete Bit.)	Ja (CVM1-V2: Kann das rechte eingeschaltete Bit bei EIN spezi- fizieren.)	Ja (Nur das linke eingeschaltete Bit.)
	ASCII CONVERT (ASCII-Konvertierung)	ASC	Ja	Ja	Ja	Ja	Ja
	ASCII TO HEX (ASCII-zu-HEX)	HEX	Ja	Ja	Ja	Ja (*1)	Ja
	COLUMN TO LINE/ LINE TO COLUMN (Spalte zu Zeile/Zeile zu Spalte)	LINE/ COLM	Ja (Bitposition im Binärformat spezifiziert)	Ja (Bitposition im Binärformat spezifiziert)	Ja (Bitposition in BCD spezifiziert)	Ja (Bitposition in BCD spezifiziert)	Ja (Bitposition in BCD spezifiziert)
	SIGNED BCD-TO- BINARY/DOUBLE SIGNED BCD-TO- BINARY (Vorzeichen- behaftetes BCD zu Binär/ Vorzeichenbe- haftetes BCD-Doppel- wort zu Binär)	BINS/ BISL	Ja	Ja	Nein	Ja (*1)	Nein
	SIGNED BINARY-TO- BCD/DOUBLE SIGNED BINARY-TO- BCD (Vorzeichenbehaf- teter Binärwert zu BCD / Vorzeichenbehaftetes Binär -Doppelwort zu BCD)	BCDS/ BDSL	Ja	Ja	Nein	Ja (*1)	Nein
	GRAY CODE CONVERSION (Gray- Code konvertieren)	GRY	CPU-Baugruppen vor Version 2.0: Nein CPU-Baugruppen Version 2.0 oder später: Ja (Wird auch ab Lot- Nummer 030201 unterstützt.)	CPU-Baugruppen vor Version 2.0: Nein CPU-Baugruppen Version 2.0 oder später: Ja (Wird auch ab Lot- Nummer 030201 unterstützt.)	Nein	Nein	Nein
Boolesche Befehle (Logik- befehle)	LOGICAL AND/LOGI- CAL OR/EXCLUSIVE OR/EXCLUSIVE NOR (Logisches UND/ logi- sches ODER/exklusiv- es ODER/ exclusives Nicht-ODER)	ANDW, ORW, XORW, XNRW	Ja	Ja	Ja	Ja	Ja
	DOUBLE LOGICAL AND/DOUBLE LOGI- CAL OR/DOUBLE EXCLUSIVE OR/ DOUBLE EXCLUSIVE NOR (Doppelwort logi- sches UND/ logisches ODER/exklusives ODER/ exclusives Nicht-ODER)	ANDL, ORWL, XORL, XNRL	Ja	Ja	Nein	Ja	Nein
	COMPLEMENT/ DOUBLE COMPLE- MENT (Komplement/ Doppelwort-Komple- ment)	COM/ COML	Ja	Ja	Ja (nur COM)	Ja	Ja (nur COM)

Gruppe/Befehl		AWL	CJ-Serie	CS-Serie	C200HX/HG/HE	CVM1/CV-Serie	CQM1H
Spezielle arithmetische Befehle	BCD SQUARE ROOT (BCD-Quadratwurzel)	ROOT	Ja	Ja	Ja	Ja	Ja
	BINARY ROOT (Binärwurzel)	ROTB	Ja	Ja	Nein	Ja (*1)	Nein
	ARITHMETIC PROCESS (Arithmetische Verarbeitung)	APR	Ja	Ja	Ja	Ja	Ja
	FLOATING POINT DIVIDE (Fließkomma-Division)	FDIV	Ja	Ja	Ja	Ja	Nein
	BIT COUNTER (Bitzähler)	BCNT	Ja (Anzahl der zu zählenden Worte und Zählergebnisse im Binärformat: 0 bis FFFF Hex)	Ja (Anzahl der zu zählenden Worte und Zählergebnisse im Binärformat: 0 bis FFFF Hex)	Ja (Anzahl der zu zählenden Worte und Zählergebnisse in BCD: 1 bis 6656)	Ja (Anzahl der zu zählenden Worte und Zählergebnisse in BCD: 0 bis 9999, jedoch Fehler bei 0)	Ja (Anzahl der zu zählenden Worte und Zählergebnisse in BCD: 1 bis 6656)
Gleitkomma-Arithmetikbefehle	FLOATING TO 16-BIT/32-BIT BIN, 16-BIT/32-BIT BIN TO FLOATING (Fließkomma zu 16 Bits/ Fließkomma zu 32 Bits, 16 Bits zu Fließkomma/32 Bits zu Fließkomma)	FIX/ FIXL, FLT/FTL	Ja	Ja	Nein	Ja (*1)	Ja
	FLOATING-POINT ADD/FLOATING-POINT SUBTRACT/ FLOATING-POINT MULTIPLY/FLOATING-POINT DIVIDE (Fließkomma-Addition/ Fließkomma-Subtraktion/ Fließkomma-Multiplikation/ Fließkomma-Division)	+F, -F, *F, /F	Ja	Ja	Nein	Ja (*1)	Ja
	DEGREES TO RADIANS/RADIANS TO DEGREES (Grad in Radianten/Radianten in Grad)	RAD, DEG	Ja	Ja	Nein	Ja (*1)	Ja
	SINE/COSINE/TANGENT/ARC SINE/ARC TANGENT (Sinus/Kosinus/Tangens/Arkussinus/ Arkustangens)	SIN, COS, TAN, ASIN, ACOS, ATAN	Ja	Ja	Nein	Ja (*1)	Ja
	SQUARE ROOT (Quadratwurzel)	SQRT	Ja	Ja	Nein	Ja (*1)	Ja
	EXPONENT (Exponent)	EXP	Ja	Ja	Nein	Ja (*1)	Ja
	LOGARITHM (Logarithmus)	LOG	Ja	Ja	Nein	Ja (*1)	Ja
	EXPONENTIAL POWER (Potenzfunktion)	PWR	Ja	Ja	Nein	Nein	Nein
	Gleitkomma-Vergleich	Beispiele: =F, <>F	CJ1: Nein CJ1-H: Ja CJ1M: Ja	CS1: Nein CS1-H: Ja	Nein	Nein	Nein
	Gleitkomma zu Textzeichenkette	FSTR, FVAL	CJ1: Nein CJ1-H: Ja CJ1M: Ja	CS1: Nein CS1-H: Ja	Nein	Nein	Nein
Gleitkommaarithmetikbefehle mit doppelter Präzision zur Konvertierung und Berechnung	Wie bei Gleitkomma-Arithmetikbefehlen mit einfacher Präzision zur Konvertierung und Berechnung (siehe oben).	Beispiel: FIXD	CJ1: Nein CJ1-H: Ja CJ1M: Ja	CS1: Nein CS1-H: Ja	Nein	Nein	Nein

Gruppe/Befehl		AWL	CJ-Serie	CS-Serie	C200HX/HG/HE	CVM1/CV-Serie	CQM1H
Tabellenda- ten-Verarbei- tungsbefehle	SET STACK (Stack- Bereich spezifizieren)	SSET	Ja (4 Worte Sta- pelsteuerungs- informationen. Anzahl der Worte im Binärformat spezifiziert: 5 bis 65535)	Ja (4 Worte Sta- pelsteuerungs- informationen. Anzahl der Worte im Binärformat spezifiziert: 5 bis 65535)	Nein	Ja (4 Worte Stapelsteue- rungsinformatio- nen. Anzahl der Worte in BCD spezifiziert: 3 bis 9999)	Nein
	PUSH ONTO STACK (Auf Stapel schieben)	PUSH	Ja	Ja	Nein	Ja	Nein
	FIRST IN FIRST OUT (FIFO-Speicher)	FIFO	Ja	Ja	Nein	Ja	Nein
	LAST IN FIRST OUT (LIFO-Speicher)	LIFO	Ja	Ja	Nein	Ja	Nein
	FIND MAXIMUM/FIND MINIMUM (Finde Maxi- mum/ Finde Minimum)	MAX, MIN	Ja (2 Worte im Datensteuerungs- feld. Tabellen- länge im Binär- format spezifi- ziert: 1 bis FFFF)	Ja (2 Worte im Datensteuerungs- feld. Tabellen- länge im Binär- format spezifi- ziert: 1 bis FFFF)	Ja (Ein Wort im Datensteuerungs- feld. Tabellen- länge in BCD spezifiziert: 1 bis 999)	Ja (Ein Wort im Daten- steuerungsfeld. Tabellenlänge in BCD spezifiziert: 1 bis 999)	Ja (Ein Wort im Daten- steuerungsfeld. Tabellenlänge in BCD spezifiziert: 1 bis 999)
	DATA SEARCH (Datensuche)	SRCH	Ja (Tabellenlänge im Binärformat spezifiziert: 1 bis FFFF. SPS-Spei- cheradresse an IR0 ausgeben. Anzahl der Übere- instimmungen kann an DR0 aus- gegeben werden.)	Ja (Tabellenlänge im Binärformat spezifiziert: 1 bis FFFF. SPS-Spei- cheradresse an IR0 ausgeben. Anzahl der Übere- instimmungen kann an DR0 aus- gegeben werden.)	Ja (Tabellenlänge in BCD spezifi- ziert: 1 bis 6556. SPS-Speicher- adresse an C+1 ausgegeben. Anzahl der Übere- instimmungen kann nicht an DR0 ausgegeben wer- den.)	Ja (Tabellenlänge in BCD spezifi- ziert: 1 bis 9999. SPS-Speicher- adresse an IR0 ausgeben. Anzahl der Übere- instimmungen kann nicht an DR0 ausgege- ben werden.)	Ja (Tabellen- länge in BCD spezifiziert: 1 bis 6556. SPS-Spei- cheradresse an C+1 ausgeben. Anzahl der Übere- instimmungen kann nicht an DR0 ausgegeben werden.)
	FRAME CHECKSUM (Rahmenprüfsumme)	FCS	Ja	Ja	Ja	Nein	Ja
	SUM	SUM (Summe)	Ja (Wie bei C200HX/HG/HE: Summe sowohl bei Bytes als auch bei Worten möglich.)	Ja (Wie bei C200HX/HG/HE: Summe sowohl bei Bytes als auch bei Worten möglich.)	Ja (Summe sowohl bei Bytes als auch bei Worten möglich.)	Ja (Summe nur bei Worten möglich.)	Ja (Summe sowohl bei Bytes als auch bei Worten möglich.)
	SWAP BYTES (Bytes tauschen)	SWAP	Ja (Kann zur Datenübertra- gung und andere Anwendungen verwendet werden.)	Ja (Kann zur Datenübertra- gung und andere Anwendungen verwendet werden.)	Nein	Nein	Nein
	DIMENSION RECORD TABLE (Tabellendefini- tion)	DIM	Ja	Ja	Nein	Nein	Nein
	SET RECORD LOCATION (Datensatz- position setzen)	SETR	Ja	Ja	Nein	Nein	Nein
	GET RECORD LOCATION (Datensatz- position einlesen)	GETR	Ja	Ja	Nein	Nein	Nein

Gruppe/Befehl		AWL	CJ-Serie	CS-Serie	C200HX/HG/HE	CVM1/CV-Serie	CQM1H
Datensteuerungs-befehle	SCALING (Skalierung)	SCL	Ja	Ja	Ja	Nein	Ja
	SCALING 2 (Skalierung 2)	SCL2	Ja	Ja	Nein	Nein	Ja
	SCALING 3 (Skalierung 3)	SCL3	Ja	Ja	Nein	Nein	Ja
	PID CONTROL (PID-Regelung)	PID	Ja (Ausgang kann zwischen 0 % und 50 % geschaltet werden, wenn Istwert = Sollwert ist. PID und Erfassungsintervall im Binärformat spezifiziert.)	Ja (Ausgang kann zwischen 0 % und 50 % geschaltet werden, wenn Istwert = Sollwert ist. PID und Erfassungsintervall im Binärformat spezifiziert.)	Ja (PID und Erfassungsintervall in BCD spezifiziert)	Ja (PID und Erfassungsintervall in BCD spezifiziert) (*1)	Ja (PID und Erfassungsintervall in BCD spezifiziert)
	PID CONTROL WITH AUTOTUNING (PID-Regelung mit Autotuning)	PIDAT	CJ1: Nein CJ1-H: Ja CJ1M: Ja	CS1: Nein CS1-H: Ja	Nein	Nein	Nein
	LIMIT CONTROL (Begrenzungssteuerung)	LMT	Ja	Ja	Nein	Ja (*1)	Nein
	DEAD BAND CONTROL (Totband-Steuerung)	BAND	Ja	Ja	Nein	Ja (*1)	Nein
	DEAD ZONE CONTROL (Totzonen-Steuerung)	ZONE	Ja	Ja	Nein	Ja (*1)	Nein
	TIME-PROPORTIONAL OUTPUT (Zeitgeregelte Ausgabe)	TPO	CPU-Baugruppen vor Version 2.0: Nein CPU-Baugruppen Version 2.0 oder später: Ja	CPU-Baugruppen vor Version 2.0: Nein CPU-Baugruppen Version 2.0 oder später: Ja	Nein	Nein	Nein
	AVERAGE (Durchschnittswert)	AVG	Ja (Anzahl der Einzelwerte im Binärformat spezifiziert)	Ja (Anzahl der Einzelwerte im Binärformat spezifiziert)	Ja (Anzahl der Einzelwerte in BCD spezifiziert)	Nein	Ja (Anzahl der Einzelwerte in BCD spezifiziert)
Unterprogramm-befehle	SUBROUTINE CALL/ SUBROUTINE ENTRY/ SUBROUTINE RETURN (Unterprogramm-Aufruf/ Unterprogramm-Anfang/ Unterprogrammrückkehr)	SBS, SBN, RET	Ja (Unterprogramm-Nummer in BCD spezifiziert: 0 bis 1023)	Ja (Unterprogramm-Nummer in BCD spezifiziert: 0 bis 1023)	Ja (Unterprogramm-Nummer in BCD spezifiziert: 0 bis 255)	Ja (Unterprogramm-Nummer in BCD spezifiziert: 0 bis 999)	Ja (Unterprogramm-Nummer in BCD spezifiziert: 0 bis 255)
	MACRO (Makro)	MCRO	Ja (Unterprogramm-Nummer in BCD spezifiziert: 0 bis 1023)	Ja (Unterprogramm-Nummer in BCD spezifiziert: 0 bis 1023)	Ja (Unterprogramm-Nummer in BCD spezifiziert: 0 bis 255)	Ja (Unterprogramm-Nummer in BCD spezifiziert: 0 bis 999) (*1)	Ja (Unterprogramm-Nummer in BCD spezifiziert: 0 bis 255)
	Globale Unterprogrammbefehle	GSBS, GSBN, GRET	CJ1: Nein CJ1-H: Ja CJ1M: Ja	CS1: Nein CS1-H: Ja	Nein	Nein	Nein
Interrupt-Steuerungs-befehle	SET INTERRUPT MASK (Interrupt-Maske einstellen)	MSKS	Ja	Ja	Nein (Die ganze Interrupt-Verarbeitung wird mit INT ausgeführt)	Ja	Nein (Die ganze Interrupt-Verarbeitung wird mit INT ausgeführt)
	CLEAR INTERRUPT (Interrupt löschen)	CLI	Ja	Ja	Nein	Ja	Nein
	READ INTERRUPT MASK (Interrupt-Maske lesen)	MSKR	Ja	Ja	Nein	Ja	Nein
	DISABLE INTERRUPTS (Interrupts deaktivieren)	DI	Ja	Ja	Nein	Nein	Nein
	ENABLE INTERRUPTS (Interrupts aktivieren)	EI	Ja	Ja	Nein	Nein	Nein
	ENABLE TIMER (Intervall-Zeitgeber)	STIM	Nein	Nein	Nein	Nein	Ja

Gruppe/Befehl		AWL	CJ-Serie	CS-Serie	C200HX/HG/HE	CVM1/CV-Serie	CQM1H
Befehle für schnelle Zähler und Impulsausgaben	MODE CONTROL (Betriebsartensteuerung)	INI	Ja (*5)	Nein	Nein	Nein	Ja
	PRESENT VALUE READ (Istwert lesen)	PRV	Ja (*5)	Nein	Nein	Nein	Ja
	COUNTER FREQUENCY CONVERT (Impulsfrequenzkonvertierung)	PRV2	CJ1M CPU-Baugruppen Version 2.0 oder später: Ja (*5) CJ1-H (alle CPU-Baugruppen): Nein	Nein	Nein	Nein	Nein
	SET COMPARISON TABLE (Tabellen-Registervergleich)	CTBL	Ja (*5)	Nein	Nein	Nein	Ja
	SET PULSES (Impulsausgabe)	PULS	Ja (*5)	Nein	Nein	Nein	Ja
	SET FREQUENCY (Geschwindigkeitssteuerung)	SPED	Ja (*5)	Nein	Nein	Nein	Ja
	ACCELERATION CONTROL (Beschleunigungssteuerung)	ACC	Ja (*5)	Nein	Nein	Nein	Ja
	POSITION CONTROL (Positionierung)	PLS2	Ja (*5)	Nein	Nein	Nein	Ja
	ORIGIN SEARCH (Nullpunktsuche)	ORG	Ja (*5)	Nein	Nein	Nein	Nein
	PWM OUTPUT (Impulsausgabe mit variablem Tastverhältnis)	PWM	Ja (*5)	Nein	Nein	Nein	Ja
Schrittketten-Befehle	Schrittdefinition und Schrittanfang	STEP/SNXT	Ja	Ja	Ja	Ja	Ja
Befehle für E/A-Baugruppen	I/O REFRESH (E/A-Auffrischung)	IORF	Ja	Ja (Wird für C200H Multi-Point-E/A-Baugruppen der Gruppe 2 sowie für Spezial-E/A-Baugruppen verwendet. Enthält die Funktionalität von Multi-Point-E/A-Baugruppen der Gruppe 2 Auffrischungs-Befehl (MPRF))	Ja (Wird für C200H Multi-Point-E/A-Baugruppen der Gruppe 2 sowie für Spezial-E/A-Baugruppen verwendet.)	Ja	Ja
	7-SEGMENT DECODER	SDEC	Ja	Ja	Ja	Ja	Ja
	DIGITAL SWITCH INPUT (Digitalschaltereingabe)	DSW	CPU-Baugruppen vor Version 2.0: Nein CPU-Baugruppen Version 2.0 oder später: Ja	CPU-Baugruppen vor Version 2.0: Nein CPU-Baugruppen Version 2.0 oder später: Ja	Ja	Nein	Ja
	TEN KEY INPUT (Zehnertasteneingabe)	TKY	CPU-Baugruppen vor Version 2.0: Nein CPU-Baugruppen Version 2.0 oder später: Ja	CPU-Baugruppen vor Version 2.0: Nein CPU-Baugruppen Version 2.0 oder später: Ja	Ja	Nein	Ja
	HEXADECIMAL KEY INPUT (Hexadezimale Tastatureingabe)	HKY	CPU-Baugruppen vor Version 2.0: Nein CPU-Baugruppen Version 2.0 oder später: Ja	CPU-Baugruppen vor Version 2.0: Nein CPU-Baugruppen Version 2.0 oder später: Ja	Ja	Nein	Ja
	MATRIX INPUT (Matrixeingabe)	MTR	CPU-Baugruppen vor Version 2.0: Nein CPU-Baugruppen Version 2.0 oder später: Ja	CPU-Baugruppen vor Version 2.0: Nein CPU-Baugruppen Version 2.0 oder später: Ja	Ja	Nein	Nein
	7-SEGMENT DISPLAY OUTPUT (7-Segmentanzeigen Ausgabe)	7SEG	CPU-Baugruppen vor Version 2.0: Nein CPU-Baugruppen Version 2.0 oder später: Ja	CPU-Baugruppen vor Version 2.0: Nein CPU-Baugruppen Version 2.0 oder später: Ja	Ja	Nein	Ja
	GROUP-2 HIGH-DENSITY I/O-REFRESH (Multi-Point-E/A-Baugruppen der Gruppe 2 Auffrischung (MPRF))	MPRF	Nein	Nein	Ja	Nein	Nein
	TEN KEY INPUT	TKY	Nein	Nein	Ja	Nein	Ja
	HEXADECIMAL KEY INPUT	HKY	Nein	Nein	Ja	Nein	Ja
	DIGITAL SWITCH INPUT	DSW	Nein	Nein	Ja	Nein	Ja
	MATRIX INPUT	MTR	Nein	Nein	Ja	Nein	Nein
	7-SEGMENTS DISPLAY OUTPUT	7SEG	Nein	Nein	Ja	Nein	Ja

Gruppe/Befehl		AWL	CJ-Serie	CS-Serie	C200HX/HG/HE	CVM1/CV-Serie	CQM1H
Befehle für Spezial-E/A-Baugruppen	SPECIAL I/O UNIT READ und SPECIAL I/O UNIT WRITE (Spezielle E/A lesen/ Spezielle E/A schreiben) (I/O READ und I/O WRITE)	IORO/IOWR (READ/ WRIT)	IORO/IOWR (Bis zu 96 Baugruppen. Wird zum Senden von FINS-Befehlen nicht mehr verwendet.	IORO/IOWR (Bis zu 96 Baugruppen. Wird zum Senden von FINS-Befehlen nicht mehr verwendet.	IORO/IOWR	READ/WRIT	Nein
	I/O READ 2 und I/O WRITE 2 (Spezielle E/A lesen/ Spezielle E/A schreiben)	RD2/ WR2	Nein	Nein	Nein	Ja (*1)	Nein
Textzeichenketten-Verarbeitungsbefehle	MOV STRING (Zeichenkette übertragen)	MOV\$	Ja	Ja	Nein	Nein	Nein
	CONCATENATE STRING (Zeichenkettenverknüpfung)	+\$	Ja	Ja	Nein	Nein	Nein
	GET STRING LEFT (Zeichenkette von links extrahieren)	LEFT\$	Ja	Ja	Nein	Nein	Nein
	GET STRING RIGHT (Zeichenkette von rechts extrahieren)	RGHT\$	Ja	Ja	Nein	Nein	Nein
	GET STRING MIDDLE (Mittlere Zeichenkette extrahieren)	MID\$	Ja	Ja	Nein	Nein	Nein
	FIND IN STRING (In Zeichenkette finden)	FIND\$	Ja	Ja	Nein	Nein	Nein
	STRING LENGTH (Zeichenkettenlänge)	LEN\$	Ja	Ja	Nein	Nein	Nein
	REPLACE IN STRING (Zeichenkette ersetzen)	RPLC\$	Ja	Ja	Nein	Nein	Nein
	DELETE STRING (Zeichenkette löschen)	DEL\$	Ja	Ja	Nein	Nein	Nein
	EXCHANGE STRING (Zeichenkette austauschen)	XCHG\$	Ja	Ja	Nein	Nein	Nein
	CLEAR STRING (Zeichenkette auf ASCII Null setzen)	CLR\$	Ja	Ja	Nein	Nein	Nein
	INSERT INTO STRING (In Zeichenkette einfügen)	INS\$	Ja	Ja	Nein	Nein	Nein

Gruppe/Befehl		AWL	CJ-Serie	CS-Serie	C200HX/HG/HE	CVM1/CV-Serie	CQM1H
Befehle für serielle Kommunikation	RECEIVE (Empfangen)	RXD	Ja (Anzahl der gespeicherten Bytes im Binärformat spezifiziert) (Wird nur für RS-232C-Schnittstelle der CPU-Baugruppe verwendet. Kann nicht für die serielle Kommunikations-Baugruppe oder die Peripherieschnittstelle der CPU-Baugruppe verwendet werden.)	Ja (Anzahl der gespeicherten Bytes im Binärformat spezifiziert) (Wird nur für RS-232C-Schnittstelle der CPU-Baugruppe verwendet. Kann nicht für das interne Modul, die serielle Kommunikations-Baugruppe oder die Peripherieschnittstelle der CPU-Baugruppe verwendet werden.)	Ja (Anzahl der gespeicherten Bytes in BCD spezifiziert) (Wird für die Peripherieschnittstelle, RS-232C-Schnittstelle oder das Kommunikationsmodul der CPU-Baugruppe verwendet.)	Nein	Ja (Anzahl der gespeicherten Bytes in BCD spezifiziert) (Wird für die Peripherieschnittstelle, RS-232C-Schnittstelle oder das Kommunikationsmodul der CPU-Baugruppe verwendet.)
	Daten über eine CPU-Bus-Baugruppe empfangen	RXDU	Ja (Anzahl der gespeicherten Bytes im Binärformat spezifiziert) (Serielle Kommunikations-Baugruppen ab Baugruppenversion 1.2)	Ja (Anzahl der gespeicherten Bytes im Binärformat spezifiziert) (Serielle Kommunikations-Baugruppen ab Baugruppenversion 1.2)	Nein	Nein	Nein
	TRANSMIT (Senden)	TXD	Ja (Anzahl der gespeicherten Bytes im Binärformat spezifiziert) (Wird nur für RS-232C-Schnittstelle der CPU-Baugruppe verwendet. Kann nicht für die serielle Kommunikations-Baugruppe oder die Peripherieschnittstelle der CPU-Baugruppe verwendet werden.) (Unaufgeforderte Kommunikationen unter Verwendung des Host-Link-EX-Befehls nicht möglich.)	Ja (Anzahl der gespeicherten Bytes im Binärformat spezifiziert) (Wird nur für RS-232C-Schnittstelle der CPU-Baugruppe verwendet. Kann nicht für das interne Modul, die serielle Kommunikations-Baugruppe oder die Peripherieschnittstelle der CPU-Baugruppe verwendet werden.) (Unaufgeforderte Kommunikationen unter Verwendung des Host-Link-EX-Befehls nicht möglich)	Ja (Anzahl der gespeicherten Bytes in BCD spezifiziert) (Wird für die Peripherieschnittstelle, RS-232C-Schnittstelle oder das Kommunikationsmodul der CPU-Baugruppe verwendet.) (Unaufgeforderte Kommunikationen unter Verwendung des Host-Link-EX-Befehls möglich.)	Nein	Ja (Anzahl der gespeicherten Bytes in BCD spezifiziert) (Wird für die Peripherieschnittstelle, RS-232C-Schnittstelle oder das Kommunikationsmodul der CPU-Baugruppe verwendet.) (Unaufgeforderte Kommunikationen unter Verwendung des Host-Link-EX-Befehls möglich.)
	Daten über eine CPU-Bus-Baugruppe senden	TXDU	Ja (Anzahl der gespeicherten Bytes im Binärformat spezifiziert) (Serielle Kommunikations-Baugruppen ab Baugruppenversion 1.2)	Ja (Anzahl der gespeicherten Bytes im Binärformat spezifiziert) (Serielle Kommunikations-Baugruppen ab Baugruppenversion 1.2)	Nein	Nein	Nein
	CHANGE SERIAL PORT SETUP (RS232-Konfigurationseinstellung)	STUP	Ja (10 Worte zugewiesen) Kann für die serielle Kommunikations-Baugruppe verwendet werden.	Ja (10 Worte zugewiesen) Kann für die serielle Kommunikations-Baugruppe verwendet werden.	Ja (5 Worte zugewiesen)	Nein	Ja (5 Worte zugewiesen)
	PROTOCOL MACRO (Protokoll-Makro)	PMCR	Ja (Sequenznummer im Binärformat spezifiziert. Vier Operanden. Baugruppen-Zieladresse und Nummer der seriellen Schnittstelle können spezifiziert werden.)	Ja (Sequenznummer im Binärformat spezifiziert. Vier Operanden. Baugruppen-Zieladresse und Nummer der seriellen Schnittstelle können spezifiziert werden.)	Ja (Sequenznummer in BCD spezifiziert: Drei Operanden.)	Nein	Ja (Sequenznummer in BCD spezifiziert: Drei Operanden.)
	PCMCIA CARD MACRO (PCMCIA-Kartenmakro)	CMCR	Nein	Nein	Ja	Nein	Nein

Gruppe/Befehl		AWL	CJ-Serie	CS-Serie	C200HX/HG/HE	CVM1/CV-Serie	CQM1H
Netzwerk-befehle	NETWORK SEND/NET-WORK RECEIVE (Netzwerk Senden/Netzwerk Empfangen)	SEND/RECV	Ja (Kann für Host-Computer über Host-Link-Verbindungen verwendet werden. Kann nicht für serielle Kommunikations-Baugruppen oder die RS-232C-Schnittstelle der CPU-Baugruppe verwendet werden.)	Ja (Kann für Host-Computer über Host-Link-Verbindungen verwendet werden. Kann nicht für serielle Kommunikations-Baugruppen, die RS-232C-Schnittstelle der CPU-Baugruppe oder das interne Modul verwendet werden.)	Ja (Kann nicht für Host-Computer über Host-Link-Verbindungen verwendet werden.)	Ja (Kann für Host-Computer über Host-Link-Verbindungen verwendet werden.)	Ja (Kann nicht für Host-Computer über Host-Link-Verbindungen verwendet werden.)
	DELIVER COMMAND (FINS-Befehl zustellen)	CMND	Ja (Wird für Host-Computer über Host-Link-Verbindungen verwendet. Kann nicht für serielle Kommunikations-Baugruppen oder die RS-232C-Schnittstelle der CPU-Baugruppe verwendet werden.)	Ja (Wird für Host-Computer über Host-Link-Verbindungen verwendet. Kann nicht für serielle Kommunikations-Baugruppen, die RS-232C-Schnittstelle der CPU-Baugruppe oder das interne Modul verwendet werden.)	Nein	Ja (Kann für Host-Computer über Host-Link-Verbindungen verwendet werden.)	Ja (Kann nicht für Host-Computer über Host-Link-Verbindungen verwendet werden.)
	EXPLICIT MESSAGE SEND (Allgemeine Explicit Message-Kommunikation)	EXPLT	CPU-Baugruppen vor Version 2.0: Nein CPU-Baugruppen Version 2.0 oder später: Ja	CPU-Baugruppen vor Version 2.0: Nein CPU-Baugruppen Version 2.0 oder später: Ja	Nein	Nein	Nein
	EXPLICIT GET ATTRIBUTE (Explicit-Einfaches Attribut abrufen)	EGATR	CPU-Baugruppen vor Version 2.0: Nein CPU-Baugruppen Version 2.0 oder später: Ja	CPU-Baugruppen vor Version 2.0: Nein CPU-Baugruppen Version 2.0 oder später: Ja	Nein	Nein	Nein
	EXPLICIT SET ATTRIBUTE (Explicit-Einfaches Attribut setzen)	ESATR	CPU-Baugruppen vor Version 2.0: Nein CPU-Baugruppen Version 2.0 oder später: Ja	CPU-Baugruppen vor Version 2.0: Nein CPU-Baugruppen Version 2.0 oder später: Ja	Nein	Nein	Nein
	EXPLICIT WORD READ (Explicit-SPS-Wort lesen)	ECHRD	CPU-Baugruppen vor Version 2.0: Nein CPU-Baugruppen Version 2.0 oder später: Ja	CPU-Baugruppen vor Version 2.0: Nein CPU-Baugruppen Version 2.0 oder später: Ja	Nein	Nein	Nein
	EXPLICIT WORD WRITE (Explicit-SPS-Wort schreiben)	ECHWR	CPU-Baugruppen vor Version 2.0: Nein CPU-Baugruppen Version 2.0 oder später: Ja	CPU-Baugruppen vor Version 2.0: Nein CPU-Baugruppen Version 2.0 oder später: Ja	Nein	Nein	Nein
Dateispei-cherbefehle	READ DATA FILE/WRITE DATA FILE (Datei lesen/Datei schreiben)	FREAD/FWRIT	Ja	Ja	Nein	Ja (FILR/FILW)	Nein
	READ PROGRAM FILE (Externes Programm lesen)	FILP	Nein	Nein	Nein	Ja	Nein
	CHANGE STEP PROGRAM (Schrittprogramm wechseln)	FLSP	Nein	Nein	Nein	Ja	Nein
Anzeige-befehle	DISPLAY MESSAGE (Meldungsanzeige)	MSG	Ja (Meldungen durch NUL-Zeichen beendet)	Ja (Meldungen durch NUL-Zeichen beendet)	Ja (Meldungen durch CR beendet)	Ja (Meldungen durch CR beendet)	Ja (Meldungen durch CR beendet)
	DISPLAY LONG MESSAGE (Lange Meldung)	LMSG	Nein	Nein	Ja (Meldungen durch CR beendet)	Nein	Nein
	I/O DISPLAY (E/A-Anzeige)	IODP	Nein	Nein	Nein	Ja	Nein
	TERMINAL MODE (Terminal-Modus)	TERM	Nein	Nein	Ja	Nein	Nein
Datum- und Uhrzeit-befehle	CALENDAR ADD (Kalender-Addition)	CADD	Ja	Ja	Nein	Ja	Nein
	CALENDAR SUBTRACT (Kalender-Subtraktion)	CSUB	Ja	Ja	Nein	Ja	Nein
	HOURS TO SECONDS (Stunden zu Sekunden)	SEC	Ja	Ja	Ja	Ja	Ja
	SECONDS TO HOURS (Sekunden zu Stunden)	HMS	Ja	Ja	Ja	Ja	Ja
	CLOCK ADJUSTMENT (Uhreinstellung)	DATE	Ja	Ja	Nein	Ja (*1)	Nein
Debugging-Befehle	TRACE MEMORY SAMPLING (Aufzeichnungsspeicher-Stichprobe)	TRSM	Ja	Ja	Ja	Ja	Ja
	MARK TRACE (Trace kennzeichnen)	MARK	Nein	Nein	Nein	Ja (Markennummer in BCD spezifiziert)	Nein

Gruppe/Befehl		AWL	CJ-Serie	CS-Serie	C200HX/HG/HE	CVM1/CV-Serie	CQM1H
Fehlerdiagnosebefehle	FAILURE ALARM/ SEVERE FAILURE ALARM (Fehleralarm/ Schwerwiegender Fehleralarm)	FAL/FALS	Ja (Meldungen durch NUL-Zeichen beendet, Textzeilenketten in der Reihenfolge vom ganz linken zum ganz rechten Byte und dann vom ganz rechten zum ganz linken Wort gespeichert. FAL-Nummer im Binärformat spezifiziert.)	Ja (Meldungen durch NUL-Zeichen beendet, Textzeilenketten in der Reihenfolge vom ganz linken zum ganz rechten Byte und dann vom ganz rechten zum ganz linken Wort gespeichert. FAL-Nummer im Binärformat spezifiziert.)	Ja (Meldungen durch CR beendet, Textzeilenketten in der Reihenfolge vom ganz linken zum ganz rechten Byte und dann vom ganz linken Wort gespeichert. FAL-Nummer in BCD spezifiziert.)	Ja (Meldungen durch CR beendet, Textzeilenketten in der Reihenfolge vom ganz linken zum ganz rechten Byte und dann vom ganz linken Wort gespeichert. FAL-Nummer in BCD spezifiziert.)	Ja (Meldungen durch CR beendet, Textzeilenketten in der Reihenfolge vom ganz linken zum ganz rechten Byte und dann vom ganz rechten zum ganz linken Wort gespeichert. FAL-Nummer in BCD spezifiziert.)
	FAILURE POINT DETECTION (Fehlerpunkt-Erfassung)	FPD	Ja (Meldungen durch NUL-Zeichen beendet, Textzeilenketten in der Reihenfolge vom ganz linken zum ganz rechten Byte und dann vom ganz rechten zum ganz linken Wort gespeichert. FAL-Nummer im Binärformat spezifiziert.)	Ja (Meldungen durch NUL-Zeichen beendet, Textzeilenketten in der Reihenfolge vom ganz linken zum ganz rechten Byte und dann vom ganz rechten zum ganz linken Wort gespeichert. FAL-Nummer im Binärformat spezifiziert.)	Ja (Meldungen durch CR beendet, Textzeilenketten in der Reihenfolge vom ganz linken zum ganz rechten Byte und dann vom ganz rechten zum ganz linken Wort gespeichert. FAL-Nummer in BCD spezifiziert.) (*1)	Ja (Meldungen durch CR beendet, Textzeilenketten in der Reihenfolge vom ganz linken zum ganz rechten Byte und dann vom ganz rechten zum ganz linken Wort gespeichert. FAL-Nummer in BCD spezifiziert.)	Ja (Meldungen durch CR beendet, Textzeilenketten in der Reihenfolge vom ganz linken zum ganz rechten Byte und dann vom ganz rechten zum ganz linken Wort gespeichert. FAL-Nummer in BCD spezifiziert.)
Sonstige Befehle	SET CARRY/CLEAR CARRY (Übertragsmerker setzen/zurücksetzen)	STC/CLC	Ja	Ja	Ja	Ja	Ja
	LOAD FLAGS /SAVE FLAGS (Merker laden/ Merker speichern)	CCL, CCS	CJ1: Nein CJ1-H: Ja CJ1M: Ja	CS1: Nein CS1-H: Ja	Nein	Ja	Nein
	EXTEND MAXIMUM CYCLE TIME (Maximale Zykluszeitüberwachung erhöhen)	WDT	Ja	Ja	Ja	Ja (*1)	Ja
	CYCLE TIME (Zykluszeit)	SCAN	Nein	Nein	Ja	Nein	Nein
	LOAD REGISTER/SAVE REGISTER (Register laden/ Register speichern)	REGL, REGS	Nein	Nein	Nein	Ja	Nein
	SELECT EM BANK (EM Bank auswählen)	EMBC	Ja	Ja	Ja	Ja	Nein
	EXPANSION DM READ (Erweiterungsspeicher(DM) lesen)	XDMMR	Nein	Nein	Ja	Nein	Nein
	INDIRECT EM ADDRESSING (Indirekte EM- Nummer spezifizieren)	IEMS	Nein	Nein	Ja	Nein	Nein
	ENABLE ACCESS/ DISABLE ACCESS (Peri- pheriesservice deaktive- ren/aktivieren)	IOSP, IORS	Nein	CS1: Nein CS1-H: Ja	Nein	Ja	Nein
	Befehle für CV/CS- Adresskonvertierung	FRMCV TOCV	CJ1: Nein CJ1-H: Ja CJ1M: Ja	CS1: Nein CS1-H: Ja	Nein	Nein	Nein
Blockprogrammierungsbefehle		BPRG/ BEND, IF/ ELSE/ IEND, WAIT, EXIT, LOOP/ LEND, BPPS/ BPRS, TIMW, CNTW, TMHW	Ja	Ja	Nein	Ja (*1)	Nein
Task-Steuerbefehle	TASK ON/TASK OFF (Task EIN/Task AUS)	TKON/ TKOF	Ja	Ja	Nein	Nein	Nein

Hinweis *1: Wird nur von CVM1 (V2) unterstützt.

*2: Wird nur von CPU□□-Z-Modellen unterstützt.

*3: Fortsetzung im selben Programmdurchlauf von CV1M Version 2 unterstützt.

*4: Mit Ausnahme der CS1 und CJ1 CPU-Baugruppen.

*5: Nur CJ1M CPU-Baugruppen mit integrierten E/A. Einige Operanden unterscheiden sich von den bei CQM1H verwendeten.

Anhang B

Änderungen gegenüber älteren Host-Link-Systemen

Host-Link-Systeme, die mit seriellen Kommunikationsmodulen (nur CS-Serie) /-Baugruppen der CS/CJ-Serie aufgebaut wurden, unterscheiden sich von den Host-Link- und CPU-Baugruppen anderer SPS-Serien. Diese Unterschiede werden im Folgenden beschrieben.

RS-232C-Schnittstellen

Folgende Unterschiede sind bei der Umstellung von einem bestehenden Host-Link-System auf eines, das eine RS-232C-Schnittstelle an einer CPU-Baugruppe der CS/CJ-Serie, einem seriellen Kommunikationsmodul (nur CS-Reihe) oder einer seriellen Kommunikationsbaugruppe (CS1H/G-CPU□□ RS-232C-Schnittstelle, CS1W-SCU21-Schnittstellen, CS1W-SCB21-Schnittstellen, CS1W-SCB41-Schnittstelle 1 bzw. CJ1W-SCU41-Schnittstelle 2) nutzt, zu berücksichtigen.

Ältere Produkte	Produktbezeichnung	Erforderliche Änderungen beim Produkt der CS/CJ-Serie	
		Verdrahtung	Sonstiges
Host-Link-Baugruppen der C-Serie	3G2A5-LK201-E C500-LK203 3G2A6-LK201-E	Statt des 25-poligen Steckerbinders wird ein 9-poliger Steckverbinder verwendet. Die Produkte der CS/CJ-Serie unterstützen die Signale ST1, ST2 und RT nicht und deren Verdrahtung ist nicht erforderlich.	Folgende Änderungen sind bei Systemen notwendig, die mit ST1, ST2 und RT synchronisiert werden. Synchronisierte Übertragungen sind nicht mehr möglich. Vollduplex-Übertragungen sind mit Geräten der CS/CJ-Serie möglich. Allerdings muss das Kommunikationsprogramm des Host-Computers, die Hardware oder beides geändert werden. Folgende Änderungen sind bei Systemen notwendig, die nicht mit ST1, ST2 und RT synchronisiert werden. Möglicherweise können die Host-Computer-Programme ohne Änderung verwendet werden, solange die gleichen Kommunikationsparameter (z. B. Baudrate) verwendet werden. Allerdings kann es notwendig werden, die Programme so zu ändern, dass sie andere Textlängen in Frames bzw. andere CS/CJ-Befehlsspezifikationen zulassen. (Siehe Hinweis.)
	C200H-LK201	Statt des 25-poligen Steckerbinders wird ein 9-poliger Steckverbinder verwendet.	Möglicherweise können die Host-Computer-Programme ohne Änderung verwendet werden, solange die gleichen Kommunikationsparameter (z. B. Baudrate) verwendet werden. Allerdings kann es notwendig werden, die Programme so zu ändern, dass sie andere Textlängen in Frames bzw. andere CS/CJ-Befehlsspezifikationen zulassen. (Siehe Hinweis.)
CPU-Baugruppen der C-Serie	SRM1 CPM1 CPM1A CQM1-CPU□□-E C200HS-CPU□□-E C200HX/HG/HE-CPU□□-E C200HW-COM□□-E	Es wurden keine Änderungen in der Verdrahtung vorgenommen.	Möglicherweise können die Host-Computer-Programme ohne Änderung verwendet werden, solange die gleichen Kommunikationsparameter (z. B. Baudrate) verwendet werden. Allerdings kann es notwendig werden, die Programme so zu ändern, dass sie andere CS/CJ-Befehlsspezifikationen zulassen.

Ältere Produkte	Produktbezeichnung	Erforderliche Änderungen beim Produkt der CS/CJ-Serie	
		Verdrahtung	Sonstiges
CVM1 oder CV-Serie CPU-Baugruppen	CVM1/CV-CPU□□-E	Es wurden keine Änderungen in der Verdrahtung vorgenommen.	Möglicherweise können die Host-Computer-Programme ohne Änderung verwendet werden, solange die gleichen Kommunikationsparameter (z. B. Baudrate) verwendet werden. Allerdings kann es notwendig werden, die Programme so zu ändern, dass sie andere CS/CJ-Befehlsspezifikationen zulassen.
CVM1 oder CV-Serie Host-Link-Baugruppe	CV500-LK201	Schnittstelle 1: Statt des 25-poligen Steckerbinders wird ein 9-poliger Steckverbinder verwendet. Schnittstelle 2 auf RS-232C eingestellt: Die Signalmasse (SG) wurde von Pin 7 auf Pin 9 gelegt.	Folgende Änderungen sind für Halbduplex-Übertragungen notwendig, die CD nutzen. Überprüfen Sie das System auf Timing-Probleme bei der Verwendung der Befehle SEND, RECV bzw. CMND zur Einleitung der Kommunikation von der SPS oder beim Senden von Befehlen vom Host-Computer. Bei Bedarf ist auf Vollduplex-Übertragungen umzuschalten. Folgende Änderungen sind für Vollduplex-Übertragungen notwendig, die CD nicht nutzen. Möglicherweise können die Host-Computer-Programme ohne Änderung verwendet werden, solange die gleichen Kommunikationsparameter (z. B. Baudrate) verwendet werden. Allerdings kann es notwendig werden, die Programme so zu ändern, dass sie andere CS/CJ-Befehlsspezifikationen zulassen.

Hinweis Die Anzahl der Worte, die bei der Verwendung von C-Modus-Befehlen pro Frame gelesen und geschrieben werden kann (d. h. die Textlängen), unterscheidet sich bei Host-Link-Baugruppen der C-Serie und seriellen Kommunikationsmodulen/-Baugruppen der CS/CJ-Serie. Ein Host-Computer-Programm, das vorher für Host-Link-Baugruppen der C-Serie verwendet wurde, funktioniert möglicherweise nicht einwandfrei, wenn es für SPS-Systeme der CS/CJ-Serie eingesetzt wird. Überprüfen Sie das Host-Computer-Programm vor seiner Verwendung, und führen Sie alle Berichtigungen aus, die zur Bewältigung anderer Frametextlängen notwendig sind. Detaillierte Informationen hierzu finden Sie im *CS/CJ-Serie Kommunikationsbefehle Referenzhandbuch (W342)*.

RS-422A/485-Schnittstellen

Folgende Unterschiede sind bei der Umstellung von einem bestehenden zu einem Host-Link-System zu berücksichtigen, das eine RS-422A/485-Schnittstelle an einem seriellen Kommunikationsmodul der CS-Serie (CS1W-SCB41 Schnittstelle 2) oder an einer seriellen Kommunikationsbaugruppe der CJ-Serie (CJ1W-SCU41 Schnittstelle 1) nutzt.

Ältere Produkte	Produktbezeichnung	Erforderliche Änderungen beim Produkt der CS/CJ-Serie	
		Verdrahtung	Sonstiges
Host-Link-Baugruppen der C-Serie	3G2A5-LK201-E C200H-LK202 3G2A6-LK202-E	Die Belegung der Steckerpins wurde wie folgt geändert. SDA: von Pin 9 zu Pin 1 SDB: von Pin 5 zu Stift 2 RDA: von Pin 6 zu Pin 6 RDB: von Pin 1 zu Pin 8 SG: von Pin 3 zu nicht belegt FG: von Pin 7 zum Steckverbindergehäuse	Möglicherweise können die Host-Computer-Programme ohne Änderung verwendet werden, solange die gleichen Kommunikationsparameter (z. B. Baudrate) verwendet werden. Allerdings kann es notwendig werden, die Programme so zu ändern, dass sie andere Textlängen in Frames bzw. andere CS/CJ-Befehlsspezifikationen zulassen. (Siehe Hinweis.)
C200HX/HG/HE Kommunikationsmodul	C200HW-COM□□-E	Es wurden keine Änderungen in der Verdrahtung vorgenommen.	Möglicherweise können die Host-Computer-Programme ohne Änderung verwendet werden, solange die gleichen Kommunikationsparameter (z. B. Baudrate) verwendet werden. Allerdings kann es notwendig werden, die Programme so zu ändern, dass sie andere CS/CJ-Befehlsspezifikationen zulassen.

Ältere Produkte	Produktbezeichnung	Erforderliche Änderungen beim Produkt der CS/CJ-Serie	
		Verdrahtung	Sonstiges
CVM1 oder CV-Serie CPU-Baugruppen	CVM1/CV-CPU□□-E	Es wurden keine Änderungen in der Verdrahtung vorgenommen.	Möglicherweise können die Host-Computer-Programme ohne Änderung verwendet werden, solange die gleichen Kommunikationsparameter (z. B. Baudrate) verwendet werden. Allerdings kann es notwendig werden, die Programme so zu ändern, dass sie andere CS/CJ-Befehlsspezifikationen zulassen.
CVM1 oder CV-Serie Host-Link-Baugruppe	CV500-LK201		

Hinweis Die Anzahl der Worte, die bei der Verwendung von C-Modus-Befehlen pro Frame gelesen und geschrieben werden kann (d. h. die Textlängen), unterscheidet sich bei Host-Link-Baugruppen der C-Serie und seriellen Kommunikationsmodulen/-Baugruppen der CS/CJ-Serie. Ein Host-Computer-Programm, das vorher für Host-Link-Baugruppen der C-Serie verwendet wurde, funktioniert möglicherweise nicht einwandfrei, wenn es für SPS-Systeme der CS/CJ-Serie eingesetzt wird. Überprüfen Sie das Host-Computer-Programm vor seiner Verwendung, und führen Sie alle Berichtigungen aus, die zur Bewältigung anderer Frametextlängen notwendig sind. Detaillierte Informationen hierzu finden Sie im *CS/CJ-Serie Kommunikationsbefehle Referenzhandbuch (W342)*.

Index

A

Adressierung
 Indexregister, 274
 Indirekte Adressen, 26–27
 Operanden, 25
 Siehe auch Indexregister
 Speicheradressen, 24

Aktualisierung
 Direkte Aktualisierung, 34, 39, 260
 E/A-Aktualisierung, 39, 260
 IORF(097), 41, 187, 261
 Zyklische Aktualisierung, 39, 260

Aktualisierungsmodus, 305
 Zeitgeber und Zähler, 305

Alarmer
 Anwenderdefinierte Alarmer, 327

Ältere Produkte
 Vergleich, 384

Anwenderprogramm, 6–7
 Siehe auch Programmierung

Anwenderprogrammdatum, 321

Anwendungen
 Dateispeicher, 209
 Sicherheitshinweise, xxii

ASCII-Zeichen, 29

Ausführbarer Status
 Beschreibung, 16

Ausführungsbedingung
 Beschreibung, 21

Ausführungsbedingungen
 Tasks, 164
 Varianten, 34

Ausgang-AUS-Bit, 352

Ausgang-AUS-Funktion, 327

Ausgänge
 Ausschalten, 327

Ausgänge:
 Ausschalten, 352

Ausschalterkennungszeit, 317

Ausschalt-Interrupts
 Deaktivieren, 317
 Tasks, 162, 177, 181–183

Automatische Übertragung beim Start, 198, 228

AWL, 43
 Eingabe, 47

B

Batterie
 Batteriefach, 2
 Einsetzen, 2

Baugruppen
 Profile, 322

Baugruppen der C-Serie

 Änderungen der Kommunikationsspezifikationen, 384

Baugruppen der CV-Serie
 Änderungen der
 Kommunikationsspezifikationen, 384–385

BCD-Daten, 30

Bedienung der Peripheriegeräte
 Bedienungspriorität, 336

Bedingungsmerker, 55
 Funktion in Tasks, 170

Befehle
 Anzeigebefehle, 140
 Ausführungsbedingungen, 34
 Ausgabebefehle, 74
 Befehle für E/A-Baugruppen, 132
 Befehle mit Flankendifferenzierung, 37
 Befehle zur Fehlerbeseitigung, 141
 Blockprogramme, 63
 Blockprogrammierungsbefehle, 144
 Dateispeicher, 217
 Dateispeicherbefehle, 138
 Datensteuerungsbefehle, 122
 Datenverschiebepbefehle, 92
 Datum- und Uhrzeitbefehle, 140
 Dekrementierbefehle, 96
 Eingabe- und Ausgabebefehle, 21
 Eingabebefehle, 72
 Eingangs- und Ausgabebefehle, 23
 Eingangs-Flankenüberwachung, 35
 Einschränkungen in Tasks, 170
 Fehlerdiagnosebefehle, 142
 Gleitkomma-Arithmetikbefehle, 111
 Grundbefehle, 21
 Indexregister, 278
 Inkrementierbefehle, 96
 Interrupt-Steuerungsbefehle, 128
 Konvertierungsbefehle, 102
 Kopierbefehle, 89, 154–155
 Logikbefehle, 108
 Netzwerkbefehle, 136
 Operanden, 22
 Position im Kontaktplan, 23
 Schleifen, 22, 60
 Sequenzsteuerbefehle, 21, 77
 Serielle Kommunikationsbefehle, 135
 Spezielle arithmetische Befehle, 110
 Step-Befehle, 132
 Symbolarithmetische Befehle, 97
 Tabellendaten-Verarbeitungsbefehle, 115, 119
 Task-Steuerbefehle, 153
 Tasksteuerung, 167
 Textzeichenketten-Verarbeitungsbefehle, 150
 Unterprogrammbefehle, 127
 Varianten, 34
 Vergleichsbefehle, 85
 Zählerbefehle, 81
 Zeitgeberbefehle, 81

- Zeitverhalten, 37
- Befehle mit Flankendifferenzierung, 37
- Befehle, die an der fallenden Flanke ausgeführt werden, 35
- Befehle, die an der steigenden Flanke ausgeführt werden, 35
- Bereichsbefehle, 281
- Betrieb
 - CPU-Baugruppe, 1
 - Debugging, 348
 - Prinzipielle Funktion, 160
 - Testbetrieb, 348
- Betriebsart beim Einschalten, 316
- Betriebsarten
 - Beschreibung, 9
 - Betriebsart beim Einschalten, 11
- Betriebsumgebung
 - Sicherheitshinweise, xxii
- Binärdaten mit Vorzeichen, 30
- Binärdaten ohne Vorzeichen, 30
- Blockprogramm, 22
- Blockprogramme, 60, 63
 - Blockprogrammierungsbefehle, 144
 - Verhältnis zu Tasks, 176

C

- C200H Kommunikationsmodule, 384
- C200HX/HG/HE Kommunikationsmodul
 - Änderungen der Kommunikationsspezifikationen, 384
- C200HX/HG/HE SPS
 - Vergleich, 357
- CPU-Baugruppe
 - Betrieb, 1
 - Interne Struktur, 6
 - Kapazität, 42
 - Prinzipielle Funktion, 160
- CVM1-Baugruppen
 - Änderungen der Kommunikationsspezifikationen, 384–385
- CX-Programmer, 20
 - Dateispeicher, 213

D

- Dateinamen, 197
- Dateispeicher, 191
 - Anwendungen, 209, 247
 - Arbeiten mit Dateien, 211
 - Dateinamen und Dateitypen, 197
 - Dateispeicherbefehle, 138, 217
 - Funktionen, 191
 - Parameterdateien, 210
 - Programmdateien, 210
 - Zugriff auf Verzeichnisse, 202
- Daten aktualisieren, 300
- Datenaufzeichnung, 353

- Datenbereiche
 - Adressierung, 24
- Datendateien, 209
- Datenformate, 30
- Datensatztabellen-Befehle, 281
- Datentypen, 197
- Datum
 - Einstellen der Uhr, 5
 - Programm und Parameter, 320
- Debugging, 326, 348
 - Befehle zur Fehlerbeseitigung, 141
 - Fehlerdiagnosebefehle, 142
- DeviceNet
 - Sicherheitshinweise, 327
- Diagnose, 326
- Direkte Aktualisierung, 34, 39, 260

E

- E/A-Aktualisierung, 39
- E/A-Ansprechzeit
 - CS/CJ E/A-Baugruppen, 344
- E/A-Baugruppen
 - Befehle für E/A-Baugruppen, 132
 - Eingangs-Ansprechzeit, 344
- E/A-Interrupts
 - Tasks, 163, 177–180
- E/A-Speicher, 7
 - Adressierung, 24
 - Initialisierung, 10
 - Tasks, 169
- E/A-Speicher-Haftmarker, 316
- E/A-Zuordnungen
 - Einstellungen des ersten Wortes, 345
- Einstellungen
 - Siehe auch* Schaltereinstellungen
 - Starteinstellungen, 315
- EM-Dateispeicher, 193
 - Initialisierung, 247
 - Siehe auch* Dateispeicher
 - Vorgänge, 253
- EU-Richtlinien, xxvi
- Externe Interrupts
 - Tasks, 163, 178–180, 183

F

- Fehler
 - Anwenderdefinierte Fehler, 327
 - Anwenderprogrammspeicher-Überlauffehler, 66
 - Befehlsverarbeitungsfehler, 66
 - Fehlerprotokoll, 326
 - Fehlerpunkt-Erkennung, 328
 - Programmeingabe, 64
 - Programmierungsfehler, 68

- Schwerwiegend, 68
- Unzulässiger-Befehl-Fehler, 66
- Zugriffsfehler, 66
- Fehleralarme, 327
- Fehlerprotokoll, 326
- Fehlerpunkt-Erkennung, 328
- FINS-Befehle
 - Dateispeicher, 215
 - Liste, 289
- Flash-Speicher, 323
- FOR-NEXT-Schleife, 60

G

- Gesamter-Link-Verfahren, 300
- Gleich-Merker, 59
- Gleitkommandaten
 - Gleitkomma-Arithmetikbefehle, 111
 - Gleitkomma-Arithmetikbefehle mit doppelter Präzision, 115
- Gleitkommadezimalwert, 31
- Größer-als-Merker, 59

H

- Host-Link-Baugruppen
 - Änderungen der Kommunikationsspezifikationen, 384
- Host-Link-Baugruppen der C-Serie
 - Änderungen der Kommunikationsspezifikationen, 383
- Host-Link-Befehle, 287
- Host-Link-Kommunikation, 286

I

- Inbetriebnahme
 - Siehe auch* Installation
- Indexregister, 27, 274
- Initialisierung
 - E/A-Speicher, 10
 - EM-Dateispeicher, 247
 - Speicherkarten, 247
- Installation
 - Herstellen der Betriebsbereitschaft, 5
 - Inbetriebnahme- erste Schritte, 2
 - Sicherheitshinweise, xxii
- Interrupts, 259
 - Deaktivieren, 188
 - Priorität von Interrupt-Tasks, 184
 - Siehe auch* Externe Interrupts
- Interrupt-Tasks, 159, 162, 177–188
 - Priorität, 184
 - Sicherheitshinweise, 186
 - Zugehörige Merker und Worte, 185
- IORF(097)-Aktualisierung, 41, 261
 - Interrupt-Tasks, 187

K

- Kleiner-als-Merker, 59
- Kommunikation
 - Befehle für serielle Kommunikation, 135
 - Meldungen, 290
 - Protokollfrei, 291
 - Siehe auch* Serielle Kommunikation
- Konstanten
 - Operanden, 28

L

- Lese-/Schreibschutz, 320

M

- Master
 - Einstellung, 303
- Master-Link-Verfahren, 300
- Mathematik
 - Gleitkomma-Arithmetikbefehle, 111
 - Spezielle arithmetische Befehle, 110
 - Symbolarithmetische Befehle, 97
- Maximale Zykluszeit, 258
- Meldungen, 290
- Merker, 22
 - Bedingungsmerker, 55
- Minimale (feste) Zykluszeit, 257
- MONITOR-Betriebsart
 - Beschreibung, 9

N

- Negativ-Merker, 59
- Netzwerke
 - Netzwerkbefehle, 136

O

- Online-Bearbeitung, 350
- Operanden
 - Beschreibung, 22
 - Konstanten, 28
 - Spezifizieren, 25
 - Textzeichenketten, 28

P

- Parameterbereich, 7
 - Dateien, 210
- Parameterdatum, 321
- Peripheriebedienungs-Prioritätsmodus, 336
- PROGRAM-Betriebsart
 - Beschreibung, 9–10
- Programmdateien, 210

Programme
 Siehe auch Programmierung
Programmfehler, 68
Programmiergeräte
 Dateispeicher, 211
 Task-Programmierung, 189
Programmierkonsolen
 Dateispeicher, 213
Programmierung, 19
 Ausführungsbedingung, 21
 AWL, 43
 Befehlsposition, 23
 Beispiele, 50
 Beschränkungen, 45
 Blockprogramm, 22
 Blockprogramme, 60
 Beschränkungen, 63
 Dezentrale Programmierung, 322
 Erstellen von Tasks, 175
 Fehler, 64
 Grundbegriffe, 42
 Programme und Tasks, 12, 20
 Programmkapazität, 42
 Programmschutz, 319
 Programmstruktur, 12, 15, 43
 Schützen des Programms, 319
 Sicherheitshinweise, 55
 Siehe auch Blockprogramme
 Step-Programmierung, 60
 Beschränkungen, 62
 Tasks und Programme, 159
 Überprüfen von Programmen, 64
 Übertragen des Programms, 348
Programmkapazität, 42
Programmstruktur, 43
Programmübertragung, 348
Protokollfreie Kommunikation, 291

R

RS-232C-Schnittstellen
 Änderungen gegenüber älteren Produkten, 383
RS-422A/485-Schnittstellen
 Änderungen gegenüber älteren Produkten, 384
RUN-Ausgang, 317
RUN-Betriebsart
 Beschreibung, 10

S

Schleifen
 FOR-NEXT-Schleifen, 60
Schnelle Eingänge, 259
Schreibschutz, 319
Sequenzsteuerbefehle

 Beschreibung, 21
Serielle Kommunikation
 Funktionen, 284
Serielle SPS-Links, 299–300
 Entsprechende Merker, 304
 SPS-Einstellungen, 303
 Zugeordnete Worte, 302
Setup
 Siehe auch Installation
Sicherheitshinweise, xix–xx
 Allgemein, xx
 Anwendungen, xxii
 Betriebsumgebung, xxii
 E/A-Aktualisierung, 9
 Interrupt-Tasks, 186
 Programmierung, 55
 Sicherheit, xx
Sichern von Daten, 323
Slaves
 Einstellungen, 303
Speicher
 Blockdiagramm des CPU-Baugruppenspeichers, 7
 Löschen, 4
 Siehe auch Anwenderspeicher
 Siehe auch Dateispeicher
 Siehe auch E/A-Speicher
Speicher löschen, 4
Speicherkarten, 7, 193
 Initialisierung, 247
 Vorsichtsmaßnahmen, 193
Sperren, 22, 38, 60
Sprungbefehle, 38, 60
SPS
 Vergleich, 357
SPS der CV-Serie
 Vergleich, 357
SPS-Setup, 7
Standby-Status
 Beschreibung, 16
Stapelverarbeitung, 279
Start
 Warmstart und Warmstopp, 315
Step-Programmierung, 60
Systemstart
 Automatische Dateiübertragung, 198, 228

T

Tabellendaten
 Verarbeitung, 281
Task-Erstausführungs-Merker, 172
Task-Fehlermerker, 173
Task-Merker, 172
Tasks, 12, 157

- Ausführung, 168
- Ausführungsbedingungen, 164
- Ausführungszeit, 18
- Beispiele, 173
- Beschreibung, 14
- Einführung, 162
- Einschränkungen, 170
- Erstellen von Tasks, 189
- Erstellung, 175
- Funktion von Bedingungsmerkern, 170
- Interrupt-Tasks, 159, 162, 178
- Merker, 172
- Merkmale, 158
- Siehe auch* Interrupt-Tasks
- Siehe auch* Zyklische Tasks
- Status, 16
- Tasknummern, 169
- Task-Steuerbefehle, 153
- Verhältnis zu Blockprogrammen, 176
- Vorteile, 158
- Zeitgeber, 170
- Zyklische Tasks, 159, 162
- Testbetrieb, 348
- Textzeichenketten
 - Operanden, 28
 - Textzeichenketten-Verarbeitungsbefehle, 150

U

- Übertragsmerker, 59
- Überwachung
 - Dezentrale Überwachung, 322
 - Flankenüberwachung, 349
- Uhr, 318
 - Datum- und Uhrzeitbefehle, 140
 - Einstellen der Uhr, 5
- Unterprogramme, 60

V

- Vergleich
 - Ältere Produkte, 384
- Verzeichnisse, 202

W

- Warmstart, 315
- Warmstopp, 315

Z

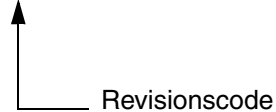
- Zähler
 - Aktualisierungsmodus, 305
- Zeit
 - Einstellen der Uhr, 5
- Zeitgeber, 305

- Erstellung mit zeitgesteuerten Interrupts, 313
- Zeitgesteuerte Interrupts
 - Tasks, 162, 177, 180–181
 - Verwendung als Zeitgeber, 313
- Zwangsweises Setzen von Bits
 - Debugging, 348
- Zyklische Aktualisierung, 39, 260
- Zyklische Tasks, 159, 162
 - Deaktivierter Status (INI), 165
 - READY-Status (ausführbereit), 165
 - RUN-Status, 165
 - Status, 165
 - WAIT-Status, 165
- Zykluszeit
 - Einstellung, 258
 - Minimale Zykluszeit, 257
 - Task-Ausführungszeit, 18
 - Überwachung, 258

Revisionshistorie

Der Revisionscode des Handbuchs bildet das Ende der Katalognummer, die auf der Vorderseite des Handbuchs angegeben ist.

Cat. No. W394-DE1-07



In der folgenden Tabelle sind die mit den einzelnen Überarbeitungen vorgenommenen Änderungen aufgeführt. Seitenzahlen beziehen sich stets auf die vorherige Version.

Revisionscode	Datum	Überarbeiteter Inhalt
01	April 2001	Erstveröffentlichung
02	Oktober 2001	Im gesamten Handbuch zusätzliche Informationen zu Hochgeschwindigkeits-CPU-Baugruppen der CS- und CJ-Serie (CS1G/H-CPU□□H und CJ1G/H-CPU□□H).
03	Juli 2002	Im gesamten Handbuch zusätzliche Informationen zu CJ1M CPU-Baugruppen. „PC“ für „Programmable Controller“ in „SPS“ (speicherprogrammierbare Steuerung) geändert. Weiterhin wurden folgende Änderungen vorgenommen: Seiten xvi und xviii: Zusätzlicher Warnhinweis. Seite xix: Punkt 2 unten auf der Seite geändert. Seite 28: Beschreibung für Textzeichenkette geändert. Seite 167: Programmierbeispiel geändert. Seiten 168, 169, 265 und 266: Zusätzliche Informationen zu DC-Spannungsversorgungen. Seite 179: Sicherheitshinweise für Speicherkarten hinzugefügt. Seite 229: Abbildung geändert. Seite 262: Zusätzliche Informationen zur Istwert-Auffrischungsmethode bei Zeitgebern/Zählern. Seite 273: Sicherheitshinweis für DeviceNet hinzugefügt. Seite 301: Baugruppen bei den Verarbeitungsgeschwindigkeiten korrigiert. Seite 304: Interrupt-Ansprechzeit korrigiert. Seite 320: CJ1-Unterstützung für IOSP/IORS geändert.
04	September 2002	Im gesamten Handbuch zusätzliche Informationen zu CJ1D CPU-Baugruppen. Weiterhin wurden folgende Änderungen vorgenommen: Seite xv: Warnhinweis zur Backup-Funktion hinzugefügt. Seite xvi: Erster Sicherheitshinweis wurde umgeschrieben und die Produktbezeichnungen der CPU-Baugruppen im Sicherheitshinweis zur Betriebsart beim Start korrigiert. Seite xviii: In der Mitte der Seite Warnhinweis für RS-232C-Schnittstelle hinzugefügt. Seite 6: Zusätzliche Informationen zu CX-Programmer-Versionen. Seite 184: Informationen zu Speicherkarten in der Tabelle korrigiert. Seite 274: Oben auf der Seite: Einstellbereich für die maximale Anzahl an Baugruppen geändert. Seite 294: Hinweis in Warnhinweis geändert und umgeschrieben. Seite 303: Hinweis 3 umgeschrieben. Seite 304: Mittlere Tabelle ergänzt.
05	April 2003	Seite 44: Erstes grundlegendes SPS-Programm-Konzept überarbeitet. Seite 45: Zweite Einschränkungsinformation geändert. Seite 46: Vierte Einschränkungsinformation geändert. Sechste Einschränkungsinformation entfernt. Seite 54: Informationen zu Netzwerken geändert, bei denen Vorsicht oder eine Überarbeitung erforderlich ist. Seiten 157 und 158: Änderungen an den Tabellen zu den Unterschieden zwischen zyklischen Tasks und zyklischen Zusatztasks. Seite 220: Hinweis 5 hinzugefügt. Seite 226: Informationen zu den unterstützten Baugruppen geändert. Seite 233: Informationen zu Spannungsausfall beim Zugriff auf den Dateispeicher geändert. Seite 264: Hinweis hinzugefügt. Seite 273: Hinweis hinzugefügt. Seite 276: Informationen in den Tabellen geändert. Hinweis unter der ersten Tabelle hinzugefügt, Hinweis unter der zweiten Tabelle geändert. Seite 310: Informationen zu CX-Programmer unter der ersten Tabelle geändert. Seite 314: Informationen im zweiten Hinweis ergänzt. Seite 326: Informationen zu Schritt 1 des Datenaufzeichnungsvorgangs ergänzt.

Versionshistorie

Revisionscode	Datum	Überarbeiteter Inhalt
06	Dezember 2003	Informationen zu neuen, von neuen CPU-Baugruppenversionen unterstützten Funktionen hinzugefügt (eine genaue Liste entfällt hier aufgrund der Vielzahl der Änderungen). Seiten xi bis xx: Sicherheitshinweise aktualisiert. Seite 72: Hinweise oberhalb der Tabelle sowie Befehle AND NOT und OR NOT hinzugefügt. Seiten 160, 201, 202, 228, 293 und 320: Hinweise hinzugefügt. Seite 189: Informationen zu CX-Programmer Version 1.0 oder höher entfernt. Seite 191: Informationen in und oberhalb der Tabelle sowie eine neue Tabelle hinzugefügt. Seiten 197, 199, 216 bis 218, 230 und 231: Informationen zum Austausch ohne Parameterbereichsdatei hinzugefügt. Seite 226: Tabelle erweitert. Seite 294: Abschnitt 6-6-8 hinzugefügt. Seiten 337 bis 339, 342, 344, 345 und 347: Zeilen zur Tabelle hinzugefügt. Seite 339: Informationen zu BCMP2 hinzugefügt.
07	Juli 2004	Änderungen im gesamten Handbuch, bedingt durch Informationen zu den neuen Funktionen, die der Übergang von Baugruppenversion 2.0 auf Baugruppenversion 3.0 für die CS/CJ-Serie CPU-Baugruppen brachte: Seite 8: Grafik geändert. Seite 32: Änderung der Werte in den Spalten „Dezimal“ und „Vorzeichenbehaftete Binärzahl in Hexadezimaldarstellung“. Seite 137: Informationen zur Erläuterung der Befehle TXD(236) und RXD(235) hinzugefügt. Informationen über neue Befehle in den neuen Abschnitten 3-23 und 3-24. Seite 190: Informationen in der Übersicht und der Grafik geändert und ergänzt. Information in der Tabelle geändert, neue Informationen hinzugefügt. Seiten 192 und 193: Informationen 5-1-2 reorganisiert und geändert. Seite 194: Tabelle erweitert. Seite 198: Tabelle erweitert, Hinweise gelöscht, neue Hinweise erstellt, Informationen zu CX-Programmer-Systemdateien hinzugefügt. Seite 202: Unter „Zellenformat“ „Zahl oder Text“ in „Text“ geändert. Seite 204: Tabelle geändert und Hinweis hinzugefügt. Seite 207: Information zu Symboltabellen- und Kommentardateien hinzugefügt. Seite 228: Neue Informationen wie Tabellen zum E/A-Zuordnungsmodus und zur Verifizierung hinzugefügt. Weiterhin Informationen zur Easy-Backup-Funktion hinzugefügt. Seite 229: Informationen zum Datenvergleich hinzugefügt Seiten 232 und 233: In den Tabellenüberschriften „E/A-Speicher“ durch „CPU-Baugruppe“ ersetzt. Seite 233: Neue Tabelle hinzugefügt. Seite 240: Informationen zum E/A-Zuordnungsmodus und zur Verifizierung hinzugefügt. Seite 277: Tabelle um Informationen zum seriellen Gateway erweitert. Seite 285: Neuer Abschnitt 6-3-4 zum seriellen Gateway hinzugefügt. Seiten 345 und 364: Informationen zur Tabelle hinzugefügt.