

Operationsliste S7-400

CPU 412, 414, 416, 417

Diese Operationsliste hat die Bestellnummer:

6ES7498-8AA03-8AN0

Ausgabe 12/2002

A5E00069510-07

Copyright © Siemens AG 2002 All rights reserved

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts ist nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung

Siemens AG
Bereich Automation and Drives
Geschäftsgebiet Industrial Automation Systems
Postfach 4848, D- 90327 Nürnberg

Siemens Aktiengesellschaft

Haftungsausschluss

Wir haben den Inhalt der Druckschrift auf Übereinstimmung mit der beschriebenen Hard- und Software geprüft. Dennoch können Abweichungen nicht ausgeschlossen werden, so dass wir für die vollständige Übereinstimmung keine Gewähr übernehmen. Die Angaben in dieser Druckschrift werden regelmäßig überprüft, und notwendige Korrekturen sind in den nachfolgenden Auflagen enthalten. Für Verbesserungsvorschläge sind wir dankbar.

© Siemens AG 2002
Technische Änderungen vorbehalten.

6ES7498-8AA03-8AN0



Inhaltsverzeichnis

| | |
|--|-----------|
| Inhaltsverzeichnis | 1 |
| Gültigkeitsbereich | 5 |
| Operanden und Parameterbereiche | 6 |
| Konstanten und Wertebereiche | 9 |
| Abkürzungen | 10 |
| Register | 12 |
| Adressierungsbeispiele | 15 |
| Beispiel zur Pointerberechnung | 18 |
| Ausführungszeiten bei indirekter Adressierung | 19 |
| Berechnungsbeispiele | 21 |
| Operationsliste | 24 |
| Verknüpfungsoperationen mit Bitoperanden | 25 |
| Verknüpfungsoperationen von Klammerausdrücken | 28 |
| ODER-Verknüpfung von UND-Funktionen | 30 |
| Verknüpfungsoperationen mit Timern und Zählern | 31 |
| Verknüpfungsoperationen mit dem Inhalt von AKKU1 | 33 |

| | |
|--|----|
| Verknüpfungsoperationen mit Anzeigenbits | 35 |
| Flankenoperationen | 38 |
| Setzen/Rücksetzen von Bitoperanden | 39 |
| VKE direkt beeinflussende Operationen | 41 |
| Timeroperationen | 42 |
| Zähloperationen | 45 |
| Ladeoperationen | 47 |
| Ladeoperationen für Timer und Zähler | 53 |
| Transferoperationen | 54 |
| Lade- und Transferoperationen für Adreßregister | 57 |
| Lade- und Transferoperationen für das Statuswort | 59 |
| Ladeoperationen für DB-Nummer und DB-Länge | 60 |
| Festpunktarithmetik (16 Bit) | 61 |
| Festpunktarithmetik (32 Bit) | 63 |
| Gleitpunktarithmetik (32 Bit) | 65 |
| Quadratwurzel, Quadrat (32 Bit) | 67 |
| Logarithmusfunktionen (32 Bit) | 68 |

| | |
|--|----|
| Trigonometrische Funktionen (32 Bit) | 69 |
| Addition von Konstanten | 70 |
| Addition über Adreßregister | 71 |
| Vergleichsoperationen (16-Bit-Integerzahlen) | 72 |
| Vergleichsoperationen (32-Bit-Integerzahlen) | 73 |
| Vergleichsoperationen (32-Bit-Realzahlen) | 74 |
| Schiebeoperationen | 75 |
| Rotieroperationen | 77 |
| AKKU-Transferoperationen, Inkrementieren, Dekrementieren | 79 |
| Bildaufbauoperation, Nulloperation | 81 |
| Datentyp-Umwandlungsoperationen | 82 |
| Komplementbildung | 85 |
| Baustein-Aufrufoperationen | 86 |
| Baustein-Endeoperationen | 89 |
| Tausche Datenbausteine | 90 |
| Sprungoperationen | 91 |
| Operationen für das Master Control Relay (MCR) | 97 |

| | |
|---|------------|
| Organisationsbausteine (OB) | 99 |
| Funktionsbausteine (FB) | 104 |
| Funktionen (FC) und Datenbausteine | 105 |
| Systemfunktionen | 106 |
| Systemfunktionsbausteine | 138 |
| SZL-Teilliste | 147 |
| Alphabetisches Verzeichnis der Operationen | 153 |

Gültigkeitsbereich

Diese Operationsliste gilt für die nachfolgend aufgelisteten CPUs.

| Name | Bestellnummer | im folgenden bezeichnet als* |
|---------------|--------------------|------------------------------|
| CPU 412-1 | 6ES7412-1XF03-0AB0 | CPU 412 |
| CPU 412-2 | 6ES7412-2XG00-0AB0 | |
| CPU 412-2 PCI | 6ES7612-2QH00-0AB4 | |
| CPU 414-2 | 6ES7414-2XG03-0AB0 | CPU 414 |
| CPU 414-3 | 6ES7414-3XJ00-0AB0 | |
| CPU 414-4H | 6ES7414-4HJ00-0AB0 | |
| CPU 416-2 | 6ES7416-2XK02-0AB0 | CPU 416 |
| CPU 416-2 PCI | 6ES7616-2QL00-0AB4 | |
| CPU 416-3 | 6ES7416-3XL00-0AB0 | |
| CPU 417-4 | 6ES7417-4XL00-0AB0 | CPU 417 |
| CPU 417-4 H | 6ES7417-4HL01-0AB0 | |

* außer in den Tabellen, in denen eine detaillierte Unterscheidung notwendig ist

Operanden und Parameterbereiche

| Operand | Parameterbereich | | | | Beschreibung |
|---------|------------------|-----------------|-----------------|-----------------|--------------------------------|
| | CPU 412 | CPU 414 | CPU 416 | CPU 417 | |
| A** | 0.0 bis 127.7 | 0.0 bis 255.7 | 0.0 bis 511.7 | 0.0 bis 1023.7 | Ausgang (im PAA) |
| AB** | 0 bis 127 | 0 bis 255 | 0 bis 511 | 0 bis 1023 | Ausgangsbyte (im PAA) |
| AW** | 0 bis 126 | 0 bis 254 | 0 bis 510 | 0 bis 1022 | Ausgangswort (im PAA) |
| AD** | 0 bis 124 | 0 bis 252 | 0 bis 508 | 0 bis 1020 | Ausgangsdoppelwort (im PAA) |
| DBX | 0.0 bis 65533.7* | 0.0 bis 65533.7 | 0.0 bis 65533.7 | 0.0 bis 65533.7 | Datenbit im DB |
| DB | 1 bis 511 | 1 bis 4095 | 1 bis 4095 | 1 bis 8191 | Datenbaustein |
| DBB | 0 bis 65533* | 0 bis 65533 | 0 bis 65533 | 0 bis 65533 | Datenbyte im DB |
| DBW | 0 bis 65532* | 0 bis 65532 | 0 bis 65532 | 0 bis 65532 | Datenwort im DB |
| DBD | 0 bis 65530* | 0 bis 65530 | 0 bis 65530 | 0 bis 65530 | Datendoppelwort im DB |
| DIX | 0.0 bis 65533.7* | 0.0 bis 65533.7 | 0.0 bis 65533.7 | 0.0 bis 65533.7 | Datenbit im Instanz- DB |
| DI | 1 bis 511 | 1 bis 4095 | 1 bis 4095 | 1 bis 8191 | Instanz-Datenbaustein |
| DIB | 0 bis 65533* | 0 bis 65533 | 0 bis 65533 | 0 bis 65533 | Datenbyte im Instanz- DB |
| DIW | 0 bis 65532* | 0 bis 65532 | 0 bis 65532 | 0 bis 65532 | Datenwort im Instanz- DB |
| DID | 0 bis 65530* | 0 bis 65530 | 0 bis 65530 | 0 bis 65530 | Datendoppelwort im Instanz- DB |

* Zusätzlich begrenzt durch die Größe des Arbeitsspeichers

** Defaulteinstellung änderbar, siehe Techn. Daten

Operanden und Parameterbereiche, Fortsetzung

| Operand | Parameterbereich | | | | Beschreibung |
|---------|------------------|----------------|-----------------|-----------------|-----------------------------|
| | CPU 412 | CPU 414 | CPU 416 | CPU 417 | |
| E** | 0.0 bis 127.7 | 0.0 bis 255.7 | 0.0 bis 511.7 | 0.0 bis 1023.7 | Eingang (im PAE) |
| EB** | 0 bis 127 | 0 bis 255 | 0 bis 511 | 0 bis 1023 | Eingangsbyte (im PAE) |
| EW** | 0 bis 126 | 0 bis 254 | 0 bis 510 | 0 bis 1022 | Eingangswort (im PAE) |
| ED** | 0 bis 124 | 0 bis 252 | 0 bis 508 | 0 bis 1020 | Eingangsdoppelwort (im PAE) |
| L** | 0.0 bis 4095.7 | 0.0 bis 8191.7 | 0.0 bis 16383.7 | 0.0 bis 32767.0 | Lokaldaten |
| LB** | 0 bis 4095 | 0 bis 8191 | 0 bis 16383 | 0 bis 32767 | Lokaldatenbyte |
| LW** | 0 bis 4094 | 0 bis 8190 | 0 bis 16382 | 0 bis 32766 | Lokaldatenwort |
| LD** | 0 bis 4092 | 0 bis 8188 | 0 bis 16380 | 0 bis 32764 | Lokaldaten-Doppelwort |
| M | 0.0 bis 4095.7 | 0.0 bis 8191.7 | 0.0 bis 16383.7 | 0.0 bis 16383.7 | Merker |
| MB | 0 bis 4095 | 0 bis 8191 | 0 bis 16383 | 0 bis 16383 | Merkerbyte |
| MW | 0 bis 4094 | 0 bis 8190 | 0 bis 16382 | 0 bis 16382 | Merkerwort |
| MD | 0 bis 4092 | 0 bis 8188 | 0 bis 16380 | 0 bis 16380 | Merkerdoppelwort |

** Defaulteinstellung änderbar, siehe Techn. Daten

Operanden und Parameterbereiche, Fortsetzung

| Operand | Parameterbereich | | | | Beschreibung |
|---------|------------------|------------|-------------|-------------|--|
| | CPU 412 | CPU 414 | CPU 416 | CPU 417 | |
| PAB | 0 bis 4095 | 0 bis 8191 | 0 bis 16383 | 0 bis 16383 | Peripherieausgangsbyte (direkter Peripheriezugriff) |
| PAW | 0 bis 4094 | 0 bis 8190 | 0 bis 16382 | 0 bis 16382 | Peripherieausgangswort (direkter Peripheriezugriff) |
| PAD | 0 bis 4092 | 0 bis 8188 | 0 bis 16380 | 0 bis 16380 | Peripherieausgangsdoppelwort (direkter Peripheriezugriff) |
| PEB | 0 bis 4095 | 0 bis 8191 | 0 bis 16383 | 0 bis 16383 | Peripherieeingangsbyte (direkter Peripheriezugriff) |
| PEW | 0 bis 4094 | 0 bis 8190 | 0 bis 16382 | 0 bis 16382 | Peripherieeingangswort (direkter Peripheriezugriff) |
| PED | 0 bis 4092 | 0 bis 8188 | 0 bis 16380 | 0 bis 16380 | Peripherieeingangsdoppelwort (direkter Peripheriezugriff) |
| T | 0 bis 255 | 0 bis 255 | 0 bis 511 | 0 bis 511 | Timer (Zeiten) |
| Z | 0 bis 255 | 0 bis 255 | 0 bis 511 | 0 bis 511 | Zähler |

Konstanten und Wertebereiche

| Konstante | Wertebereich | Beschreibung |
|----------------------------|--------------|----------------------------------|
| B(b1,b2) B(b1,b2,b3,b4) | – | Konstante, 2 oder 4 Byte |
| D# Datum | – | IEC-Datumskonstante |
| L# Integer | – | 32-Bit-Integer-Konstante |
| P# Bitpointer | – | Pointerkonstante |
| S5T# Zeitwert | – | S7-Zeitkonstante * |
| T# Zeitwert | – | Zeitkonstante |
| TOD# Zeitwert | – | IEC-Zeitkonstante |
| C# Zählwert | – | Zählerkonstante (BCD-codiert) |
| 2#n | – | Binärkonstante |
| W#16# DW#16# | – | Hexadezimalkonstante |

* Dient zum Laden der S7-Timer

Abkürzungen

Folgende Abkürzungen werden in der Operationsliste verwendet:

| Abkürzung | ... steht für | Beispiel |
|-----------|---|------------|
| k8 | 8-Bit-Konstante 0 bis 255 | 32 |
| k16 | 16-Bit-Konstante 256 bis 32 767 | 28 131 |
| k32 | 32-Bit-Konstante 32 768 bis 999 999 999 | 127 624 |
| i8 | 8-Bit-Integer -128 bis +127 | -113 |
| i16 | 16-Bit-Integer -32768 bis +32767 | +6523 |
| i32 | 32-Bit-Integer -2 147 483 648 bis +2 147 483 647 | -2 222 222 |
| m | Pointer-Konstante | P#240.3 |
| n | Binärkonstante | 1001 1100 |
| p | Hexadezimalkonstante | EA12 |
| MARKE | symbolische Sprungadresse (max. 4 Buchstaben) | ZIEL |
| a | Byteadresse | |

Abkürzungen, Fortsetzung

| Abkürzung | ... steht für | Beispiel |
|-----------|--|---------------------------------------|
| b | Bitadresse | |
| c | Operandenbereich | E, A, M, L, DBX, DIX |
| d | Adresse steht im: MD, DBD, DID oder LD | |
| e | Nummer steht im: MW, DBW, DIW oder LW | |
| f | Timer-/Zähler-Nr. | |
| g | Operandenbereich | EB, AB, PEB, PAB, MB, LB, DBB, DIB |
| h | Operandenbereich | EW, AW, PEW, PAW, MW, LW, DBW, DIW |
| i | Operandenbereich | ED, AD, PED, PAD, MD, LD, DBD, DID |
| q | Baustein-Nr. | |

Register

AKKU1 bis AKKU4 (32 Bit)

Die AKKUs sind Register für die Verarbeitung von Bytes, Worten oder Doppelworten. Dazu werden die Operanden in die AKKUs geladen und dort verknüpft. Das Ergebnis der Operation steht immer im AKKU1 und kann von dort in eine Speicherzelle transferiert werden.

Die AKKUs sind 32 Bit breit.

Bezeichnungen:

| AKKU | Bit |
|---------------------|---------------|
| AKKUx (x = 1 bis 4) | Bit 0 bis 31 |
| AKKUx-L | Bit 0 bis 15 |
| AKKUx-H | Bit 16 bis 31 |
| AKKUx-LL | Bit 0 bis 7 |
| AKKUx-LH | Bit 8 bis 15 |
| AKKUx-HL | Bit 16 bis 23 |
| AKKUx-HH | Bit 24 bis 31 |

Adreßregister AR1 und AR2 (32 Bit)

Die Adreßregister enthalten die bereichsinternen oder bereichsübergreifenden Zeiger für die registerindirekt adressierenden Operationen. Die Adreßregister sind 32 Bit breit. Die bereichsinternen bzw. bereichsübergreifenden Zeiger haben folgenden Aufbau:

- bereichsinterner Zeiger: 00000000 00000bbb bbbbbbbb bbbbxxxx
- bereichsübergreifenderZeiger: **yyyyyyyy** 00000bbb bbbbbbbb bbbbxxxx

Legende: b Byteadresse
 x Bitnummer
 y Bereichskennung
 (siehe Kapitel Adressierungsbeispiele)

Statuswort (16 Bit)

Die Anzeigen werden durch die Operationen ausgewertet oder gesetzt.

Das Statuswort ist 16 Bit breit.

| Bit | Belegung | Bedeutung |
|----------|--------------|----------------------|
| 0 | /ER | Erstabfrage |
| 1 | VKE | Verknüpfungsergebnis |
| 2 | STA | Status |
| 3 | OR | Oder (und-vor-oder) |
| 4 | OS | Überlauf speichernd |
| 5 | OV | Überlauf |
| 6 | A0 | Ergebnisanzeige 0 |
| 7 | A1 | Ergebnisanzeige 1 |
| 8 | BIE | Binäresultat |
| 9 bis 15 | nicht belegt | – |

Adressierungsbeispiele

| Adressierungsbeispiele | Beschreibung |
|----------------------------------|--|
| Unmittelbare Adressierung | |
| L +27 | Lade 16-Bit-Integer-Konstante "27" in AKKU1 |
| L L#-1 | Lade 32-Bit-Integer-Konstante "-1" in AKKU1 |
| L 2#1010101010101010 | Lade Binärkonstante in AKKU1 |
| L DW#16#A0F0BCFD | Lade Hexadezimalkonstante in AKKU1 |
| L 'ENDE' | Lade ASCII-Zeichen in AKKU1 |
| L T#500 ms | Lade Zeitwert in AKKU1 |
| L C#100 | Lade Zählerwert in AKKU1 |
| L B#(100,12) | Lade Konstante als 2 Byte |
| L B#(100,12,50,8) | Lade Konstante als 4 Byte |
| L P#10.0 | Lade bereichsinternen Pointer in AKKU1 |
| L P#E20.6 | Lade bereichsübergreifenden Pointer in AKKU1 |
| L -2.5 | Lade Realzahl in AKKU1 |
| L D# 1995-01-20 | Lade Datum |
| L TOD 13:20:33.125 | Lade Uhrzeit |
| Direkte Adressierung | |
| U E 0.0 | UND-Verknüpfung des Eingangsbits 0.0 |
| L EB 1 | Lade Eingangsbyte 1 in AKKU1 |
| L EW 0 | Lade Eingangswort 0 in AKKU1 |
| L ED 0 | Lade Eingangsdoppelwort 0 in AKKU1 |

Adressierungsbeispiele, Fortsetzung

| Adressierungsbeispiele | Beschreibung |
|---|--|
| Indirekte Adressierung Timer/Zähler | |
| SI T [LW 8] | Starte Timer; die Timer-Nr. steht im Lokaldatenwort 8 |
| ZV Z [LW 10] | Zähle vorwärts; die Zähler-Nr. steht im Lokaldatenwort 10 |
| Speicherindirekte, bereichsinterne Adressierung | |
| U E [LD 12] Beispiel: L P#22.2 T LD 12 U E [LD 12] | UND-Operation; die Adresse des Eingangs steht als Pointer im Lokaldaten-Doppelwort 12 |
| U E [DBD 1] | UND-Operation; die Adresse des Eingangs steht als Pointer im Daten-Doppelwort 1 des aufgeschlagenen DB |
| U A [DID 12] | UND-Operation; die Adresse des Ausgangs steht als Pointer im Datendoppelwort 12 des aufgeschlagenen Instanz-DB |
| U A [MD 12] | UND-Operation; die Adresse des Ausgangs steht als Pointer im Merkerdoppelwort 12 |
| Registerindirekte, bereichsinterne Adressierung | |
| U E [AR1,P#12.2] | UND-Operation; die Adresse des Eingangs errechnet sich zu "Pointerwert im Adreßregister 1 + Pointer P#12.2" |

Adressierungsbeispiele, Fortsetzung

| Adressierungsbeispiele | Beschreibung | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|--|---|--------------------|-----------------------------|---------|---|-----------|----|-------------------|---|-----------|----|-----------------|---|-----------|----|-----------------|---|-----------|----|---------------|----|-----------|----|--------------|----|-----------|----|----------------------|---|-----------|----|-------------------|----|-----------|----|-----------------------------|--|--|--|--|---|
| Registerindirekte, bereichsübergreifende Adressierung | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Für die bereichsübergreifende, registerindirekte Adressierung muß die Adresse zusätzlich eine Bereichskennung enthalten. Die Adresse steht im Adreßregister. Es gibt folgende Bereichskennungen: | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| <table border="1"> <thead> <tr> <th data-bbox="312 379 461 434">Bereichs- kennung</th> <th data-bbox="501 379 612 434">Codierung binär</th> <th data-bbox="692 412 740 434">hex.</th> <th data-bbox="884 379 963 402">Bereich</th> </tr> </thead> <tbody> <tr> <td data-bbox="312 437 328 460">P</td> <td data-bbox="501 437 612 460">1000 0000</td> <td data-bbox="692 437 724 460">80</td> <td data-bbox="884 437 1051 460">Peripheriebereich</td> </tr> <tr> <td data-bbox="312 463 328 486">E</td> <td data-bbox="501 463 612 486">1000 0001</td> <td data-bbox="692 463 724 486">81</td> <td data-bbox="884 463 1043 486">Eingangsbereich</td> </tr> <tr> <td data-bbox="312 489 328 511">A</td> <td data-bbox="501 489 612 511">1000 0010</td> <td data-bbox="692 489 724 511">82</td> <td data-bbox="884 489 1051 511">Ausgangsbereich</td> </tr> <tr> <td data-bbox="312 515 328 537">M</td> <td data-bbox="501 515 612 537">1000 0011</td> <td data-bbox="692 515 724 537">83</td> <td data-bbox="884 515 1027 537">Merkerbereich</td> </tr> <tr> <td data-bbox="312 540 344 563">DB</td> <td data-bbox="501 540 612 563">1000 0100</td> <td data-bbox="692 540 724 563">84</td> <td data-bbox="884 540 1011 563">Datenbereich</td> </tr> <tr> <td data-bbox="312 566 336 589">DI</td> <td data-bbox="501 566 612 589">1000 0101</td> <td data-bbox="692 566 724 589">85</td> <td data-bbox="884 566 1091 589">Instanz-Datenbereich</td> </tr> <tr> <td data-bbox="312 592 328 614">L</td> <td data-bbox="501 592 612 614">1000 0110</td> <td data-bbox="692 592 724 614">86</td> <td data-bbox="884 592 1059 614">Lokaldatenbereich</td> </tr> <tr> <td data-bbox="312 618 344 640">VL</td> <td data-bbox="501 618 612 640">1000 0111</td> <td data-bbox="692 618 724 640">87</td> <td data-bbox="884 618 1155 640">Vorgänger-Lokaldatenbereich</td> </tr> <tr> <td colspan="4" data-bbox="884 644 1362 694">(Zugriff auf Lokaldaten des aufrufenden Bausteins)</td> </tr> </tbody> </table> | Bereichs- kennung | Codierung binär | hex. | Bereich | P | 1000 0000 | 80 | Peripheriebereich | E | 1000 0001 | 81 | Eingangsbereich | A | 1000 0010 | 82 | Ausgangsbereich | M | 1000 0011 | 83 | Merkerbereich | DB | 1000 0100 | 84 | Datenbereich | DI | 1000 0101 | 85 | Instanz-Datenbereich | L | 1000 0110 | 86 | Lokaldatenbereich | VL | 1000 0111 | 87 | Vorgänger-Lokaldatenbereich | (Zugriff auf Lokaldaten des aufrufenden Bausteins) | | | | L B [AR1,P#8.0] Lade Byte in AKKU1; die Adresse errechnet sich aus "Pointerwert im Adreßregister 1 + Pointer P#8.0" |
| Bereichs- kennung | Codierung binär | hex. | Bereich | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| P | 1000 0000 | 80 | Peripheriebereich | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| E | 1000 0001 | 81 | Eingangsbereich | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| A | 1000 0010 | 82 | Ausgangsbereich | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| M | 1000 0011 | 83 | Merkerbereich | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| DB | 1000 0100 | 84 | Datenbereich | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| DI | 1000 0101 | 85 | Instanz-Datenbereich | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| L | 1000 0110 | 86 | Lokaldatenbereich | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| VL | 1000 0111 | 87 | Vorgänger-Lokaldatenbereich | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| (Zugriff auf Lokaldaten des aufrufenden Bausteins) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| U [AR1,P#32.3] | UND-Operation; die Adresse des Operanden errechnet sich aus "Pointerwert im Adreßregister 1 + Pointer P#32.3" | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Adressierung über Parameter | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| U Parameter | Der Operand wird über den Parameter adressiert | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

Beispiel zur Pointerberechnung

- **Beispiel bei Summe der Bitadressen ≤ 7 :**

LAR1 P#8.2

U E [AR1,P#10.2]

Ergebnis: Adressiert wird Eingang 18.4 (durch jeweilige Addition der Byte- und Bitadressen)

- **Beispiel bei Summe der Bitadressen > 7 :**

L P#10.5

LAR1

U E [AR1,P#10.7]

Ergebnis: Adressiert wird Eingang 21.4 (durch Addition der Byte- und Bitadressen mit Übertrag)

Ausführungszeiten bei indirekter Adressierung

Eine Anweisung mit indirekt adressierten Operanden besteht aus 2 Teilen:

1. Teil: Laden der Adresse des Operanden

2. Teil: Ausführen der Operation

Das bedeutet, Sie müssen auch die Ausführungszeit einer Anweisung mit indirekt adressiertem Operanden aus diesen beiden Teilen berechnen.

Ausführungszeit berechnen

Für die Gesamt-Ausführungszeit gilt:

$$\begin{aligned} & \text{Ausführungszeit für das Laden der Adresse} \\ + & \text{ Ausführungszeit der Operation} \\ = & \underline{\underline{\text{Gesamt-Ausführungszeit der Operation}}} \end{aligned}$$

Die im Kapitel "Operationsliste" angegebenen Ausführungszeiten sind die Ausführungszeiten für den 2. Teil einer Anweisung, also für das eigentliche Ausführen einer Operation.

Zu dieser Zeit müssen Sie noch die Ausführungszeit für das Laden der Adresse des Operanden hinzufügen (siehe folgende Tabelle).

Ausführungszeiten bei indirekter Adressierung

Die folgende Tabelle gibt die Ausführungszeiten für das Laden der Adresse des Operanden an, abhängig von der Lage der Adresse.1

| Adresse liegt im ... | Ausführungszeit in µs | | | |
|---|-----------------------|---------|---------|---------|
| | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| Merkerbereich M | | | | |
| Wort | 0,2 | 0,2 | 0,16 | 0,2 |
| Doppelwort | 0,3 | 0,3 | 0,24 | 0,3 |
| Datenbaustein DB/DI | | | | |
| Wort | 0,2 | 0,2 | 0,16 | 0,2 |
| Doppelwort | 0,3 | 0,3 | 0,24 | 0,3 |
| Lokaldatenbereich L | | | | |
| Wort | 0,2 | 0,2 | 0,16 | 0,2 |
| Doppelwort | 0,3 | 0,3 | 0,24 | 0,3 |
| AR1/AR2 (bereichsintern) | 0,0 * | 0,0 * | 0,0 * | 0,0 * |
| AR1/AR2 (bereichsübergreifend) | 0,0 * | 0,0 * | 0,0 * | 0,0 * |
| Parameter (Wort) für: | | | | |
| • Zeiten | 0,2 | 0,2 | 0,16 | 0,2 |
| • Zähler | 0,2 | 0,2 | 0,16 | 0,2 |
| • Bausteinaufrufe | 0,2 | 0,2 | 0,16 | 0,2 |
| Parameter (Doppelwort) für Bit, Byte, Worte und Doppelworte | 0,3 | 0,3 | 0,24 | 0,3 |

* Die Adreßregister AR1/AR2 brauchen beim Adressieren nicht über separate Takte geladen zu werden

Auf den folgenden Seiten finden Sie Berechnungsbeispiele für die Operationslaufzeiten der verschiedenen indirekt adressierten Operanden.

Berechnungsbeispiele

Für die Ermittlung der Ausführungszeit finden Sie hier Berechnungsbeispiele für die verschiedenen indirekten Adressierungsarten.

- Ausführungszeit bei speicherindirekter, bereichsinterner Adressierung berechnen**

Beispiel: U E [DBD 12] mit CPU 414

1. Schritt: Laden des Inhalts von DBD 12 (Zeit steht in Tabelle auf Seite 19)

| Adresse liegt im ... | Ausführungszeit in μs |
|---|----------------------------------|
| Merkerbereich M Wort Doppelwort | 0,2 |
| | 0,3 |
| Datenbaustein DB/DI Wort Doppelwort | 0,2 |
| | 0,3 |

2. Schritt: UND-Verknüpfung des so adressierten Eingangs (Ausführungszeit siehe Tabelle des Kapitels "Operationsliste" auf Seite 21)

| Ausführungszeit in μs | |
|----------------------------------|---------------------------|
| direkte Adressierung | indirekte Adressierung |
| 0,2 : | Zeit für U E 0,1+ : |

Gesamt-Ausführungszeit

$$\begin{array}{r}
 0,3 \mu\text{s} \\
 + 0,1 \mu\text{s} \\
 \hline
 0,4 \mu\text{s}
 \end{array}$$

• **Ausführungszeit bei speicherindirekter, bereichsinterner Adressierung berechnen**

Beispiel: U [AR1, P#23.1] ... mit E 1.0 in AR1 mit CPU 416

1. Schritt: Laden des Inhalts von AR1 und erhöhen um den Offset 23.1 (Ausführungszeit siehe Tabelle auf Seite 19)

| Adresse liegt im ... | Ausführungszeit in μs |
|--------------------------------|----------------------------------|
| : | : |
| AR1/AR2 (bereichsübergreifend) | 0,00 |
| : | : |

2. Schritt: UND-Verknüpfung des so adressierten Eingangs (Ausführungszeit siehe Tabellen des Kapitels "Operationsliste")

| Ausführungszeit in μs | |
|----------------------------------|---------------------------|
| direkte Adressierung | indirekte Adressierung |
| 0,08 | Zeit für U E 0,08+ |
| : | : |

Gesamt-Ausführungszeit

$$\begin{array}{r}
 0,00 \mu\text{s} \\
 + 0,08 \mu\text{s} \\
 \hline
 \underline{\underline{0,08 \mu\text{s}}}
 \end{array}$$

• **Ausführungszeit bei speicherindirekter, bereichsinterner Adressierung berechnen**

Beispiel: U Parameter ... mit E 0.5 in der Baueinparameterliste bei CPU 414

1. Schritt: Laden des über den Parameter adressierten E 0.5 (Ausführungszeit siehe Tabelle auf Seite 19)

| Adresse liegt im ... | Ausführungszeit in μs |
|------------------------|----------------------------------|
| : | : |
| : | : |
| Parameter (Doppelwort) | 0,3 |

2. Schritt: UND-Verknüpfung des so adressierten Eingangs (Ausführungszeit siehe Tabelle des Kapitels "Operationsliste" auf Seite 25)

| Ausführungszeit in μs | |
|----------------------------------|------------------------|
| direkte Adressierung | indirekte Adressierung |
| 0,2 | Zeit für U E 0,1+ |
| : | : |

Gesamt-Ausführungszeit

$$\begin{array}{r}
 0,3 \mu\text{s} \\
 + 0,1 \mu\text{s} \\
 \hline
 0,4 \mu\text{s}
 \end{array}$$

Operationsliste

In diesem Kapitel sind die Operationen für die CPUs der S7-400 aufgelistet. Die Erläuterung der Operationen beschränkt sich auf eine knappe Form. Die genaue Funktionsbeschreibung finden Sie in den Referenzhandbüchern zu STEP 7.

Beachten Sie: Bei indirekter Adressierung (Beispiele siehe Seite 16) müssen Sie zu den Ausführungszeiten noch die Zeit für das Laden der Adresse des jeweiligen Operanden addieren (siehe Seite 19).

Verknüpfungsoperationen mit Bitoperanden

Alle Verknüpfungsoperationen (VKO) bilden ein Verknüpfungsergebnis (VKE-neu). Die erste VKO einer Verknüpfungskette bildet das VKE-neu aus dem abgefragten Signalzustand. Die nun folgenden VKOs bilden das VKE-neu aus dem abgefragten Signalzustand und dem VKE-alt. Die Verknüpfungskette wird durch eine Operation abgeschlossen, die das VKE begrenzt (z.B. Speicheroperation), d. h. das /ER-Bit auf Null setzt.

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in µs | | | | | | | | |
|-----------------------------|-----------|----------------|------------------------------------|-----------------------|---------|---------|---------|------|----|-----|-----|-----|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 | | | | | |
| U/UN | E/A | a.b | UND/UND-NICHT | 1*/2 | 0,2/0,3 | 0,1 | 0,08 | 0,1 | | | | |
| | M | a.b | Eingang/Ausgang | 1**/2 | 0,2/0,3 | 0,1 | 0,08 | 0,1 | | | | |
| | L | a.b | Merker | 2 | 0,3 | 0,1 | 0,08 | 0,1 | | | | |
| | DBX | a.b | Lokaldatenbit | 2 | 0,3 | 0,1 | 0,08 | 0,1 | | | | |
| | DIX | a.b | Datenbit | 2 | 0,3 | 0,1 | 0,08 | 0,1 | | | | |
| | | a.b | Instanz-Datenbit | 2 | 0,3 | 0,1 | 0,08 | 0,1 | | | | |
| | c [d] | | speicherindirekt, bereichsintern | 2 | 0,3+ | 0,1+ | 0,08+ | 0,1+ | | | | |
| | c [AR1,m] | | registerind., bereichsintern (AR1) | 2 | 0,3+ | 0,1+ | 0,08+ | 0,1+ | | | | |
| | c [AR2,m] | | registerind., bereichsintern (AR2) | 2 | 0,3+ | 0,1+ | 0,08+ | 0,1+ | | | | |
| | [AR1,m] | | bereichsübergreifend (AR1) | 2 | 0,3+ | 0,1+ | 0,08+ | 0,1+ | | | | |
| | [AR2,m] | | bereichsübergreifend (AR2) | 2 | 0,3+ | 0,1+ | 0,08+ | 0,1+ | | | | |
| Parameter | | über Parameter | 2 | 0,3+ | 0,1+ | 0,08+ | 0,1+ | | | | | |
| Statuswort für: U/UN | | | | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | | | | – | – | – | – | – | ja | – | ja | ja |
| Operation beeinflusst: | | | | – | – | – | – | – | ja | ja | ja | 1 |

+ plus Zeit für das Laden der Adresse des Operanden (siehe Seite19)

*) bei direkter Adressierung des Operanden; Adreßbereich 0 bis 127

***) bei direkter Adressierung des Operanden; Adreßbereich 0 bis 255

Verknüpfungsoperationen mit Bitoperanden, Fortsetzung

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in µs | | | |
|-----------|-----------|------------------------------------|-----------------|-----------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| O/ON | E/A a.b | Eingang/Ausgang | 1*/2 | 0,2/0,3 | 0,1 | 0,08 | 0,1 |
| | M a.b | Merker | 1**/2 | 0,2/0,3 | 0,1 | 0,08 | 0,1 |
| | L a.b | Lokaldatenbit | 2 | 0,3 | 0,1 | 0,08 | 0,1 |
| | DBX a.b | Datenbit | 2 | 0,3 | 0,1 | 0,08 | 0,1 |
| | DIX a.b | Instanz-Datenbit | 2 | 0,3 | 0,1 | 0,08 | 0,1 |
| | c [d] | speicherindirekt, bereichsintern | 2 | 0,3+ | 0,1+ | 0,08+ | 0,1+ |
| | c [AR1,m] | registerind., bereichsintern (AR1) | 2 | 0,3+ | 0,1+ | 0,08+ | 0,1+ |
| | c [AR2,m] | registerind., bereichsintern (AR2) | 2 | 0,3+ | 0,1+ | 0,08+ | 0,1+ |
| | [AR1,m] | bereichsübergreifend (AR1) | 2 | 0,3+ | 0,1+ | 0,08+ | 0,1+ |
| | [AR2,m] | bereichsübergreifend (AR2) | 2 | 0,3+ | 0,1+ | 0,08+ | 0,1+ |
| | Parameter | über Parameter | 2 | 0,3+ | 0,1+ | 0,08+ | 0,1+ |

| | | | | | | | | | |
|------------------------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: O, ON | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | – | – | – | – | – | – | – | ja | ja |
| Operation beeinflusst: | – | – | – | – | – | 0 | ja | ja | 1 |

+ plus Zeit für das Laden der Adresse des Operanden (siehe Seite 19)

*) bei direkter Adressierung des Operanden; Adreßbereich 0 bis 127

***) bei direkter Adressierung des Operanden; Adreßbereich 0 bis 255

Verknüpfungsoperationen mit Bitoperanden, Fortsetzung

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in µs | | | |
|-----------|-----------|---------------------------------------|-----------------|-----------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| X/XN | | EXKLUSIV-ODER/ EXKLUSIV-ODER-NICHT | | | | | |
| | E/A a.b | Eingang/Ausgang | 2 | 0,3 | 0,1 | 0,08 | 0,1 |
| | M a.b | Merker | 2 | 0,3 | 0,1 | 0,08 | 0,1 |
| | L a.b | Lokaldatenbit | 2 | 0,3 | 0,1 | 0,08 | 0,1 |
| | DBX a.b | Datenbit | 2 | 0,3 | 0,1 | 0,08 | 0,1 |
| | DIX a.b | Instanz-Datenbit | 2 | 0,3 | 0,1 | 0,08 | 0,1 |
| | c [d] | speicherindirekt, bereichsintern | 2 | 0,3+ | 0,1+ | 0,08+ | 0,1+ |
| | c [AR1,m] | registerind., bereichsintern (AR1) | 2 | 0,3+ | 0,1+ | 0,08+ | 0,1+ |
| | c [AR2,m] | registerind., bereichsintern (AR2) | 2 | 0,3+ | 0,1+ | 0,08+ | 0,1+ |
| | [AR1,m] | bereichsübergreifend (AR1) | 2 | 0,3+ | 0,1+ | 0,08+ | 0,1+ |
| | [AR2,m] | bereichsübergreifend (AR2) | 2 | 0,3+ | 0,1+ | 0,08+ | 0,1+ |
| | Parameter | über Parameter | 2 | 0,3+ | 0,1+ | 0,08+ | 0,1+ |

| | | | | | | | | | | |
|------------------------|--------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: | X, XN | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | | – | – | – | – | – | – | – | ja | ja |
| Operation beeinflusst: | | – | – | – | – | – | 0 | ja | ja | 1 |

+ plus Zeit für das Laden der Adresse des Operanden (siehe Seite 19)

Verknüpfungsoperationen von Klammersausdrücken

Retten der Bits VKE, OR und der entsprechenden Funktionskennung (U, UN, ...) auf den Klammerstack. 7 Klammerebenen sind pro Baustein möglich. Nach "Klammer zu" wird das gerettete VKE mit dem aktuellen VKE verknüpft, entsprechend der Funktionskennung; das aktuelle OR wird vom geretteten OR überschrieben.

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in µs | | | |
|-----------|---------|---------------------------------|-----------------|-----------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| U(| | UND-Klammer-Auf | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| UN(| | UND-NICHT-Klammer-Auf | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| O(| | ODER-Klammer-Auf | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| ON(| | ODER-NICHT-Klammer-Auf | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| X(| | EXKLUSIV-ODER-Klammer-Auf | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| XN(| | EXKLUSIV-ODER-NICHT-Klammer-Auf | 1 | 0,1 | 0,1 | 0,08 | 0,1 |

| | | | | | | | | | | |
|------------------------|----------------------------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: | U(, UN(, O(, ON(, X(, XN(| BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | | – | – | – | – | – | ja | – | ja | ja |
| Operation beeinflusst: | | – | – | – | – | – | 0 | 1 | – | 0 |

Verknüpfungsoperationen von Klammersausdrücken, Fortsetzung

| Opera- tion | Operand | Bedeutung | Län- ge in Wor- ten | Ausführungszeit in μ s | | | |
|----------------|---------|---|------------------------------|----------------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
|) | | Klammer zu, Entfernen eines Ein- trags vom Klammerstack. | 1 | 0,1 | 0,1 | 0,08 | 0,1 |

| | | | | | | | | | |
|------------------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für:) | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | – | – | – | – | – | – | – | ja | – |
| Operation beeinflusst: | – | – | – | – | – | ja | 1 | ja | 1 |

ODER-Verknüpfung von UND-Funktionen

Es erfolgt die ODER-Verknüpfung von UND-Funktionen nach der Regel: UND vor ODER

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μs | | | |
|-----------|---------|--|-----------------|----------------------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| O | | ODER-Verknüpfung von UND-Funktionen nach der Regel: UND-vor-ODER | 1 | 0,1 | 0,1 | 0,08 | 0,1 |

| | | | | | | | | | |
|--------------------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: O | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | - | - | - | - | - | - | - | ja | ja |
| Operation beeinflusst: | - | - | - | - | - | ja | 1 | - | ja |

Verknüpfungsoperationen mit Timern und Zählern

Abfrage des adressierten Timer/Zähler auf den Zustand. Das Ergebnis der Abfrage wird nach der entsprechenden Funktion mit dem VKE verknüpft.

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μ s | | | |
|-----------|---------------------------|--|-----------------|----------------------------|--------------|----------------|--------------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| U/UN | | UND/UND-NICHT | | | | | |
| | T f | Timer | 1*/2 | 0,3 | 0,1 | 0,08 | 0,1 |
| | T [e] | Timer, speicherindirekt adressiert | 2 | 0,3+ | 0,1+ | 0,08+ | 0,1+ |
| | Z f | Zähler | 1*/2 | 0,3 | 0,1 | 0,08 | 0,1 |
| | Z [e] | Zähler, speicherind. adressiert | 2 | 0,3+ | 0,1+ | 0,08+ | 0,1+ |
| | Timerpara. Zählerpara. | Timer/Zähler (über Parameter adressiert) | 2 | 0,3+ 0,3+ | 0,1+ 0,1+ | 0,08+ 0,08+ | 0,1+ 0,1+ |

| | | | | | | | | | |
|------------------------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: U, UN | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | – | – | – | – | – | ja | – | ja | ja |
| Operation beeinflusst: | – | – | – | – | – | ja | ja | ja | 1 |

+ plus Zeit für das Laden der Adresse des Operanden (siehe Seite19)

*) bei direkter Adressierung des Operanden; Adreßbereich 0 bis 255

Verknüpfungsoperationen mit Timern und Zählern, Fortsetzung

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in µs | | | |
|-----------|---------------------------|--|-----------------|-----------------------|--------------|----------------|--------------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| O/ON | T f | ODER/ODER-NICHT Timer | 1*/2 | 0,3 | 0,1 | 0,08 | 0,1 |
| | T [e] | Timer, speicherindirekt adressiert | 2 | 0,3+ | 0,1+ | 0,08+ | 0,1+ |
| | Z f | Zähler | 1*/2 | 0,3 | 0,1 | 0,08 | 0,1 |
| | Z [e] | Zähler, speicherind. adressiert | 2 | 0,3+ | 0,1+ | 0,08+ | 0,1+ |
| | Timerpara. Zählerpara. | Timer/Zähler (über Parameter adressiert) | 2 | 0,3+ 0,3+ | 0,1+ 0,1+ | 0,08+ 0,08+ | 0,1+ 0,1+ |
| X/XN | T f | EXKLUSIV-ODER/ EXKLUSIV-ODER-NICHT Timer | 2 | 0,3 | 0,1 | 0,08 | 0,1 |
| | T [e] | Timer, speicherindirekt adressiert | 2 | 0,3+ | 0,1+ | 0,08+ | 0,1+ |
| | Z f | Zähler | 2 | 0,3 | 0,1 | 0,08 | 0,1 |
| | Z [e] | Zähler, speicherind. adressiert | 2 | 0,3+ | 0,1+ | 0,08+ | 0,1+ |
| | Timerpara. Zählerpara. | EXKLUSIV-ODER Timer/Zähler (über Parameter adressiert) | 2 | 0,3+ 0,3+ | 0,1+ 0,1+ | 0,08+ 0,08+ | 0,1+ 0,1+ |

| | | | | | | | | | | |
|------------------------|---------------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: | O, ON, X, XN | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | | – | – | – | – | – | – | – | ja | ja |
| Operation beeinflusst: | | – | – | – | – | – | 0 | ja | ja | 1 |

+ plus Zeit für das Laden der Adresse des Operanden (siehe Seite 19)

*) bei direkter Adressierung des Operanden; Adreßbereich 0 bis 255

Verknüpfungsoperationen mit dem Inhalt von AKKU1

Verknüpfung des Inhalts von AKKU1 bzw. AKKU1-L mit einem Wort bzw. einem Doppelwort nach der entsprechenden Funktion. Das Wort bzw. Doppelwort wird entweder bei der Operation als Operand oder im AKKU2 mit angegeben. Das Ergebnis steht im AKKU1 bzw. AKKU1-L.

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μ s | | | |
|-----------|---------|--------------------------------|-----------------|----------------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| UW | | UND AKKU2-L | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| UW | W#16#p | UND 16-Bit-Konstante | 2 | 0,2 | 0,1 | 0,08 | 0,1 |
| OW | | ODER AKKU2-L | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| OW | W#16#p | ODER 16-Bit-Konstante | 2 | 0,2 | 0,1 | 0,08 | 0,1 |
| XOW | | EXKLUSIV-ODER AKKU2-L | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| XOW | W#16#p | EXKLUSIV-ODER 16-Bit-Konstante | 2 | 0,2 | 0,1 | 0,08 | 0,1 |

| | | | | | | | | | |
|------------------------------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: UW, OW, XOW | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | – | – | – | – | – | – | – | – | – |
| Operation beeinflusst: | – | ja | 0 | 0 | – | – | – | – | – |

Verknüpfungsoperationen mit dem Inhalt von AKKU1, Fortsetzung

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μ s | | | |
|-----------|---------|--------------------------------|-----------------|----------------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| UD | | UND AKKU2 | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| UD | DW#16#p | UND 32-Bit-Konstante | 3 | 0,3 | 0,15 | 0,12 | 0,15 |
| OD | | ODER AKKU2 | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| OD | DW#16#p | ODER 32-Bit-Konstante | 3 | 0,3 | 0,15 | 0,12 | 0,15 |
| XOD | | EXKLUSIV-ODER AKKU2 | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| XOD | DW#16#p | EXKLUSIV-ODER 32-Bit-Konstante | 3 | 0,3 | 0,15 | 0,12 | 0,15 |

| | | | | | | | | | |
|------------------------------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: UD, OD, XOD | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | – | – | – | – | – | – | – | – | – |
| Operation beeinflusst: | – | ja | 0 | 0 | – | – | – | – | – |

Verknüpfungsoperationen mit Anzeigenbits

Alle Verknüpfungsoperationen (VKO) bilden ein Verknüpfungsergebnis (VKE-neu). Die erste VKO einer Verknüpfungskette bildet das VKE-neu aus dem abgefragten Signalzustand. Die nun folgenden VKOs bilden das VKE-neu aus dem abgefragten Signalzustand und dem VKE-alt. Die Verknüpfungskette wird durch eine Operation abgeschlossen, die das VKE begrenzt (z.B. Speicheroperation), d. h. das /ER-Bit auf Null setzt.

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in µs | | | |
|----------------------|---------|--|-----------------|-----------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| U/UN O/ON X/XN | ==0 | UND/UND-NICHT ODER/ODER-NICHT EXKLUSIV-ODER/ EXKLUSIV-ODER-NICHT Ergebnis=0 (A1=0 und A0=0) | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| | >0 | Ergebnis>0 (A1=1 und A0=0) | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| | <0 | Ergebnis<0 (A1=0 und A0=1) | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| | <>0 | Ergebnis≠0 ((A1=0 und A0=1) oder (A1=1 und A0=0)) | 1 | 0,1 | 0,1 | 0,08 | 0,1 |

| | | | | | | | | | | |
|------------------------|-----------------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: | U/UN/O/ON/X/XN | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | | – | ja | ja | – | – | ja | – | ja | ja |
| Operation beeinflusst: | | – | – | – | – | – | ja | ja | ja | 1 |

Verknüpfungsoperationen mit Anzeigenbits, Fortsetzung

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μ s | | | |
|----------------------|---------|--|-----------------|----------------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| U/UN O/ON X/XN | <=0 | Ergebnis<=0 ((A1=0 und A0=1) oder (A1=0 und A0=0)) | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| | >=0 | Ergebnis>=0 ((A1=1 und A0=0) oder (A1=0 und A0=0)) | 1 | 0,1 | 0,1 | 0,08 | 0,1 |

| | | | | | | | | | |
|---------------------------------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: U/UN/O/ON/X/XN | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | – | ja | ja | – | – | ja | – | ja | ja |
| Operation beeinflusst: | – | – | – | – | – | ja | ja | ja | 1 |

Verknüpfungsoperationen mit Anzeigenbits, Fortsetzung

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μ s | | | |
|----------------------|---------|--|-----------------|----------------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| U/UN O/ON X/XN | UO | UND/UND-NICHT ODER/ODER-NICHT EXKLUSIV-ODER/ EXKLUSIV-ODER-NICHT unordered/unzulässige Arithmetikoperation (A1=1 und A0=1) | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| | OS | UND OS=1 | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| | BIE | UND BIE=1 | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| | OV | UND OV=1 | 1 | 0,1 | 0,1 | 0,08 | 0,1 |

| | | | | | | | | | |
|---------------------------------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: U/UN/O/ON/X/XN | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | ja | ja | ja | ja | ja | ja | – | ja | ja |
| Operation beeinflusst: | – | – | – | – | – | ja | ja | ja | 1 |

Flankenoperationen

Das aktuelle VKE wird verglichen mit dem Status des Operanden, dem "Flankenmerker". FP erkennt einen Flankenwechsel von "0" nach "1". FN erkennt einen Flankenwechsel von "1" nach "0".

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μ s | | | |
|-----------|-----------|--|-----------------|----------------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| FP/FN | E/A a.b | Anzeigen der steigenden/fallenden Flanke mit VKE=1. Flankenhilfsmerker ist das in der Operation adressierte Bit. | 2 | 0,4 | 0,2 | 0,16 | 0,2 |
| | M a.b | | 2 | 0,4 | 0,2 | 0,16 | 0,2 |
| | L a.b* | | 2 | 0,4 | 0,2 | 0,16 | 0,2 |
| | DBX a.b | | 2 | 0,4 | 0,2 | 0,16 | 0,2 |
| | DIX a.b | | 2 | 0,4 | 0,2 | 0,16 | 0,2 |
| | c [d] | | 2 | 0,4+ | 0,2+ | 0,16+ | 0,2+ |
| | c [AR1,m] | | 2 | 0,4+ | 0,2+ | 0,16+ | 0,2+ |
| | c [AR2,m] | | 2 | 0,4+ | 0,2+ | 0,16+ | 0,2+ |
| | [AR1,m] | | 2 | 0,4+ | 0,2+ | 0,16+ | 0,2+ |
| | [AR2,m] | | 2 | 0,4+ | 0,2+ | 0,16+ | 0,2+ |
| | Parameter | | 2 | 0,4+ | 0,2+ | 0,16+ | 0,2+ |

| Statuswort für: FP, FN | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
|-------------------------------|-----|----|----|----|----|----|-----|-----|-----|
| Operation wertet aus: | – | – | – | – | – | – | – | ja | – |
| Operation beeinflusst: | – | – | – | – | – | 0 | ja | ja | 1 |

+ plus Zeit für das Laden der Adresse des Operanden (siehe Seite 19)

*) nicht sinnvoll, falls zu überwachendes Bit im Prozeßabbild (Lokaldaten eines Bausteins sind nur zu dessen Laufzeit gültig.)

Setzen/Rücksetzen von Bitoperanden

Zuweisen des Wertes "1" bzw. "0" an den adressierten Operanden, wenn VKE = 1. MCR-Abhängigkeit beachten (siehe Seite 97).

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in µs | | | | |
|-----------|-----------|--------------------------------|------------------------------------|-----------------------|---------|---------|---------|------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 | |
| S R | | Setze adressiertes Bit auf "1" | | | | | | |
| | | Setze adressiertes Bit auf "0" | | | | | | |
| | E/A | a.b | Eingang/Ausgang | 1*/2 | 0,3/0,4 | 0,2 | 0,16 | 0,2 |
| | M | a.b | Merker | 1**/2 | 0,3/0,4 | 0,2 | 0,16 | 0,2 |
| | L | a.b | Lokaldatenbit | 2 | 0,4 | 0,2 | 0,16 | 0,2 |
| | DBX | a.b | Datenbit | 2 | 0,4 | 0,2 | 0,16 | 0,2 |
| | DIX | a.b | Instanz-Datenbit | 2 | 0,4 | 0,2 | 0,16 | 0,2 |
| | c [d] | | speicherindirekt, bereichsintern | 2 | 0,4+ | 0,2+ | 0,16+ | 0,2+ |
| | c [AR1,m] | | registerind., bereichsintern (AR1) | 2 | 0,4+ | 0,2+ | 0,16+ | 0,2+ |
| | c [AR2,m] | | registerind., bereichsintern (AR2) | 2 | 0,4+ | 0,2+ | 0,16+ | 0,2+ |
| | [AR1,m] | | bereichsübergreifend (AR1) | 2 | 0,4+ | 0,2+ | 0,16+ | 0,2+ |
| | [AR2,m] | | bereichsübergreifend (AR2) | 2 | 0,4+ | 0,2+ | 0,16+ | 0,2+ |
| | Parameter | | über Parameter | 2 | 0,4+ | 0,2+ | 0,16+ | 0,2+ |

| Statuswort für: | S, R | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
|------------------------|------|-----|----|----|----|----|----|-----|-----|-----|
| Operation wertet aus: | | – | – | – | – | – | – | – | ja | – |
| Operation beeinflusst: | | – | – | – | – | – | 0 | ja | – | 0 |

+ plus Zeit für das Laden der Adresse des Operanden (siehe Seite19)

*) bei direkter Adressierung des Operanden; Adreßbereich 0 bis 127

***) bei direkter Adressierung des Operanden; Adreßbereich 0 bis 255

Setzen/Rücksetzen von Bitoperanden, Fortsetzung

Der VKE-Wert wird in den adressierten Operanden geschrieben. MCR-Abhängigkeit beachten (siehe Seite 97).

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in µs | | | |
|-----------|----------------|-------------------------------------|-----------------|-----------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| = | E/A a.b | Zuweisen des VKE an Eingang/Ausgang | 1*/2 | 0,3/0,4 | 0,2 | 0,16 | 0,2 |
| | M a.b | an Merker | 1**/2 | 0,3/0,4 | 0,2 | 0,16 | 0,2 |
| | L a.b | an Lokaldatenbit | 2 | 0,4 | 0,2 | 0,16 | 0,2 |
| | DBX a.b | an Datenbit | 2 | 0,4 | 0,2 | 0,16 | 0,2 |
| | DIX a.b | an Instanz-Datenbit | 2 | 0,4 | 0,2 | 0,16 | 0,2 |
| | c [d] | speicherindirekt, bereichsintern | 2 | 0,4+ | 0,2+ | 0,16+ | 0,2+ |
| | c [AR1,m] | registerind., bereichsintern (AR1) | 2 | 0,4+ | 0,2+ | 0,16+ | 0,2+ |
| | c [AR2,m] | registerind., bereichsintern (AR2) | 2 | 0,4+ | 0,2+ | 0,16+ | 0,2+ |
| | [AR1,m] | bereichsübergreifend (AR1) | 2 | 0,4+ | 0,2+ | 0,16+ | 0,2+ |
| | [AR2,m] | bereichsübergreifend (AR2) | 2 | 0,4+ | 0,2+ | 0,16+ | 0,2+ |
| Parameter | über Parameter | 2 | 0,4+ | 0,2+ | 0,16+ | 0,2+ | |

| | | | | | | | | | |
|------------------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: = | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | - | - | - | - | - | - | - | ja | - |
| Operation beeinflusst: | - | - | - | - | - | 0 | ja | - | 0 |

+ plus Zeit für das Laden der Adresse des Operanden (siehe Seite 19)

*) bei direkter Adressierung des Operanden; Adreßbereich 0 bis 127

***) bei direkter Adressierung des Operanden; Adreßbereich 0 bis 255

VKE direkt beeinflussende Operationen

Die folgenden Operationen bearbeiten direkt das VKE.

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in µs | | | | | | |
|-----------------------------|---------|------------------------------|-----------------|-----------------------|----|---------|----|---------|---------|-----|
| | | | | CPU 412 | | CPU 414 | | CPU 416 | CPU 417 | |
| CLR | | Setze VKE auf "0" | 1 | 0,1 | | 0,1 | | 0,08 | 0,1 | |
| Statuswort für: CLR | | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | | - | - | - | - | - | - | - | - | - |
| Operation beeinflusst: | | - | - | - | - | - | 0 | 0 | 0 | 0 |
| SET | | Setze VKE auf "1" | 1 | 0,1 | | 0,1 | | 0,08 | 0,1 | |
| Statuswort für: SET | | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | | - | - | - | - | - | - | - | - | - |
| Operation beeinflusst: | | - | - | - | - | - | 0 | 1 | 1 | 0 |
| NOT | | Negiere das VKE | 1 | 0,1 | | 0,1 | | 0,08 | 0,1 | |
| Statuswort für: NOT | | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | | - | - | - | - | - | ja | - | ja | - |
| Operation beeinflusst: | | - | - | - | - | - | - | 1 | ja | - |
| SAVE | | Rette das VKE in das BIE-Bit | 1 | 0,1 | | 0,1 | | 0,08 | 0,1 | |
| Statuswort für: SAVE | | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | | - | - | - | - | - | - | - | ja | - |
| Operation beeinflusst: | | ja | - | - | - | - | - | - | - | - |

Timeroperationen

Starten bzw. Rücksetzen eines Timers. Die Zeitdauer muß im AKKU1-L stehen. Die Operationen werden durch einen Flankenwechsel am VKE ausgelöst. Das heißt, wenn das VKE zwischen zwei Aufrufen seinen Zustand geändert hat, wird die Operation ausgelöst.

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in µs | | | |
|-----------|--------------|---|--------------------|-----------------------|-------------|---------------|-------------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| SI | T f T [e] | Starte Timer als Impuls bei Flankenwechsel von "0" nach "1" | 1 ¹)/2 | 0,3/0,4 0,3+/0,4+ | 0,2 0,2+ | 0,16 0,16+ | 0,2 0,2+ |
| | Timerpara. | | 2 | 0,4+ | 0,2+ | 0,16+ | 0,2+ |
| SV | T f T [e] | Starte Timer als verlängerten Impuls bei Flankenwechsel von "0" nach "1" | 1 ¹)/2 | 0,3/0,4 0,4+ | 0,2 0,2+ | 0,16 0,16+ | 0,2 0,2+ |
| | Timerpara. | | 2 | 0,4+ | 0,2+ | 0,16+ | 0,2+ |
| SE | T f T [e] | Starte Timer als Einschaltverzögerung bei Flankenwechsel von "0" nach "1" | 1 ¹)/2 | 0,3/0,4 0,4+ | 0,2 0,2+ | 0,16 0,16+ | 0,2 0,2+ |
| | Timerpara. | | 2 | 0,4+ | 0,2+ | 0,16+ | 0,2+ |

| | | | | | | | | | | |
|------------------------|-------------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: | SI, SV, SE | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | | – | – | – | – | – | – | – | ja | – |
| Operation beeinflusst: | | – | – | – | – | – | 0 | – | – | 0 |

+ plus Zeit für das Laden der Adresse des Operanden (siehe Seite 19)

1) bei indirekter Adressierung des Operanden Timer-Nr.: 0 bis 255

Timeroperationen, Fortsetzung

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μs | | | |
|-----------|--------------|---|--------------------|----------------------------------|-------------|---------------|-------------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| SS | T f T [e] | Starte Timer als speichernde Einschaltverzögerung bei Flankenwechsel von "0" nach "1" | 1 ¹⁾ /2 | 0,3/0,4 0,4+ | 0,2 0,2+ | 0,16 0,16+ | 0,2 0,2+ |
| | Timerpara. | | 2 | 0,4+ | 0,2+ | 0,16+ | 0,2+ |
| SA | T f T [e] | Starte Timer als Ausschaltverzögerung bei Flankenwechsel von "1" nach "0" | 1 ¹⁾ /2 | 0,3/0,4 0,4+ | 0,2 0,2+ | 0,16 0,16+ | 0,2 0,2+ |
| | Timerpara. | | 2 | 0,4+ | 0,2+ | 0,16+ | 0,2+ |

| | | | | | | | | | |
|-------------------------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: SS, SA | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | - | - | - | - | - | - | - | ja | - |
| Operation beeinflusst: | - | - | - | - | - | 0 | - | - | 0 |

+ plus Zeit für das Laden der Adresse des Operanden (siehe Seite 19)

1) bei indirekter Adressierung des Operanden Timer-Nr.: 0 bis 255

Timeroperationen, Fortsetzung

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μs | | | |
|-----------|--------------|---|--------------------|----------------------------------|-------------|---------------|-------------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| FR | T f T [e] | Freigabe eines Timers für das erneute Starten bei Flankenwechsel von "0" nach "1" (Löschen des Flankenmerkers für das Starten der Zeit) | 1 ¹⁾ /2 | 0,3/0,4 0,4+ | 0,2 0,2+ | 0,16 0,16+ | 0,2 0,2+ |
| | Timerpara. | | 2 | 0,4+ | 0,2+ | 0,16+ | 0,2+ |
| R | T f T [e] | Rücksetzen einer Zeit | 1 ¹⁾ /2 | 0,3/0,4 0,4+ | 0,2 0,2+ | 0,16 0,16+ | 0,2 0,2+ |
| | Timerpara. | | 2 | 0,4+ | 0,2+ | 0,16+ | 0,2+ |

| | | | | | | | | | |
|------------------------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: FR, R | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | – | – | – | – | – | – | – | ja | – |
| Operation beeinflusst: | – | – | – | – | – | 0 | – | – | 0 |

+ plus Zeit für das Laden der Adresse des Operanden (siehe Seite 19)

1) bei indirekter Adressierung des Operanden Timer-Nr.: 0 bis 255

Zähloperationen

Der Zählwert muß im AKKU1-L als BCD-Zahl (0 - 999) vorliegen.

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in µs | | | |
|-----------|--------------|---|--------------------|-----------------------|-------------|---------------|-------------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| S | Z f Z [e] | Vorbelegen eines Zählers bei Flankenwechsel v. "0" nach "1" | 1 ¹⁾ /2 | 0,3/0,4 0,4+ | 0,2 0,2+ | 0,16 0,16+ | 0,2 0,2+ |
| | Zählerpara. | | 2 | 0,4+ | 0,2+ | 0,16+ | 0,2+ |
| R | Z f Z [e] | Rücksetzen des Zählers auf "0" bei VKE = "1" | 1 ¹⁾ /2 | 0,3/0,4 0,4+ | 0,2 0,2+ | 0,16 0,16+ | 0,2 0,2+ |
| | Zählerpara. | | 2 | 0,4+ | 0,2+ | 0,16+ | 0,2+ |
| ZV | Z f Z [e] | Zähle um 1 vorwärts bei Flankenwechsel von "0" nach "1" | 1 ¹⁾ /2 | 0,3/0,4 0,4+ | 0,2 0,2+ | 0,16 0,16+ | 0,2 0,2+ |
| | Zählerpara. | | 2 | 0,4+ | 0,2+ | 0,16+ | 0,2+ |

| | | | | | | | | | |
|---------------------------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: S, R, ZV | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | - | - | - | - | - | - | - | ja | - |
| Operation beeinflusst: | - | - | - | - | - | 0 | - | - | 0 |

+ plus Zeit für das Laden der Adresse des Operanden (siehe Seite 19)

1) bei indirekter Adressierung des Operanden Zähler-Nr.: 0 bis 255

Zähloperationen, Fortsetzung

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μ s | | | |
|-----------|--------------|---|--------------------|----------------------------|-------------|---------------|-------------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| ZR | Z f Z [e] | Zähle um 1 rückwärts bei Flankenwechsel von "0" nach "1" | 1 ¹)/2 | 0,3/0,4 0,4+ | 0,2 0,2+ | 0,16 0,16+ | 0,2 0,2+ |
| FR | Z f Z [e] | Freigabe eines Zählers bei Flankenwechsel von "0" nach "1" (Löschen des Flankenmerkers für Vorwärts-, Rückwärtszählen und Setzen eines Zählers) | 1 ¹)/2 | 0,3/0,4 0,4+ | 0,2 0,2+ | 0,16 0,16+ | 0,2 0,2+ |
| | Zählerpara. | | 2 | 0,4+ | 0,2+ | 0,16+ | 0,2+ |

| | | | | | | | | | |
|-------------------------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: ZR, FR | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | – | – | – | – | – | – | – | ja | – |
| Operation beeinflusst: | – | – | – | – | – | 0 | – | – | 0 |

+ plus Zeit für das Laden der Adresse des Operanden (siehe Seite 19)

1) bei indirekter Adressierung des Operanden Zähler-Nr.: 0 bis 255

Ladeoperationen

Laden der Operanden in AKKU1. Zuvor wird der Inhalt von AKKU1 in AKKU2 gerettet. Das Statuswort wird nicht beeinflusst.

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μs | | | |
|-----------|-----------|---------------------------------------|--------------------|----------------------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| L | EB a | Lade ... Eingangsbyte | 1 ¹)/2 | 0,2/03 | 0,1 | 0,08 | 0,1 |
| | AB a | Ausgangsbyte | 1 ¹)/2 | 0,2/0,3 | 0,1 | 0,08 | 0,1 |
| | PEB a | Peripherie-Eingangsbyte ²⁾ | 2 | 0,3 | 0,1 | 0,08 | 0,1 |
| | MB a | Merkerbyte | 1 ³)/2 | 0,2/0,3 | 0,1 | 0,08 | 0,1 |
| | LB a | Lokaldatenbyte | 2 | 0,3 | 0,1 | 0,08 | 0,1 |
| | DBB a | Datenbyte | 2 | 0,3 | 0,1 | 0,08 | 0,1 |
| | DIB a | Instanz-Datenbyte ... in AKKU1 | 2 | 0,3 | 0,1 | 0,08 | 0,1 |
| | g [d] | speicherindirekt, bereichsintern | 2 | 0,3+ | 0,1+ | 0,08+ | 0,1+ |
| | g [AR1,m] | registerind., bereichsintern (AR1) | 2 | 0,3+ | 0,1+ | 0,08+ | 0,1+ |
| | g [AR2,m] | registerind., bereichsintern (AR2) | 2 | 0,3+ | 0,1+ | 0,08+ | 0,1+ |
| | B[AR1,m] | bereichsübergreifend (AR1) | 2 | 0,3+ | 0,1+ | 0,08+ | 0,1+ |
| | B[AR2,m] | bereichsübergreifend (AR2) | 2 | 0,3+ | 0,1+ | 0,08+ | 0,1+ |
| | Parameter | über Parameter | 2 | 0,3+ | 0,1+ | 0,08+ | 0,1+ |

+ plus Zeit für das Laden der Adresse des Operanden (siehe Seite 19)

1) bei indirekter Adressierung des Operanden; Adreßbereich 0 bis 127

2) plus Reaktionszeit der E/A-Baugruppe ($> 1 \mu\text{s}$), bei den CPUs 414-4H und 417-4H: solo $34 \mu\text{s}$, redundant $64 \mu\text{s}$

3) bei direkter Adressierung des Operanden; Adreßbereich 0 bis 255

Ladeoperationen, Fortsetzung

Bei direkter Adressierung erfordert der Zugriff auf ungerade Wort-adressen bei allen Befehlen der CPU 416 einen Zuschlag von 0,08 µs und bei allen Befehlen der CPU 414 einen Zuschlag von 0,1 µs.

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in µs | | | |
|-----------|-----------|---------------------------------------|--------------------|-----------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| L | EW a | Lade ... Eingangswort | 1 ¹)/2 | 0,2/0,3 | 0,1 | 0,08 | 0,1 |
| | AW a | Ausgangswort | 1 ¹)/2 | 0,2/0,3 | 0,1 | 0,08 | 0,1 |
| | PEW a | Peripherie-Eingangswort ²⁾ | 2 | 0,3 | 0,1 | 0,08 | 0,1 |
| | MW a | Merkerwort | 1 ³)/2 | 0,2/0,3 | 0,1 | 0,08 | 0,1 |
| | LW a | Lokaldatenwort | 2 | 0,3 | 0,1 | 0,08 | 0,1 |
| | DBW a | Datenwort | 2 | 0,3 | 0,1 | 0,08 | 0,1 |
| | DIW a | Instanz-Datenwort ... in AKKU1-L | 2 | 0,3 | 0,1 | 0,08 | 0,1 |
| | h [d] | speicherindirekt, bereichsintern | 2 | 0,3+ | 0,1+ | 0,08+ | 0,1+ |
| | h [AR1,m] | registerind., bereichsintern (AR1) | 2 | 0,3+ | 0,1+ | 0,08+ | 0,1+ |
| | h [AR2,m] | registerind., bereichsintern (AR2) | 2 | 0,3+ | 0,1+ | 0,08+ | 0,1+ |
| | W[AR1,m] | bereichsübergreifend (AR1) | 2 | 0,3+ | 0,1+ | 0,08+ | 0,1+ |
| | W[AR2,m] | bereichsübergreifend (AR2) | 2 | 0,3+ | 0,1+ | 0,08+ | 0,1+ |
| | Parameter | über Parameter | 2 | 0,3+ | 0,1+ | 0,08+ | 0,1+ |

+ plus Zeit für das Laden der Adresse des Operanden (siehe Seite19)

1) bei indirekter Adressierung des Operanden; Adreßbereich 0 bis 127

2) plus Reaktionszeit der E/A-Baugruppe (> 1 µs), bei den CPUs 414-4H und 417-4H: solo 37µs, redundant 67µs

3) bei direkter Adressierung des Operanden; Adreßbereich 0 bis 255

Ladeoperationen, Fortsetzung

Die in der Tabelle angegebenen Ausführungszeiten bei direkter Adressierung erhöhen sich um 0,1 µs (CPU 412), 0,1 µs (CPU 414), 0,08 µs (CPU 416) bzw. 0,1 µs (CPU 417), wenn der Zugriff auf ungerade Adressen erfolgt.

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in µs | | | |
|-----------|-----------|---|--------------------|-----------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| L | ED a | Lade ... Eingangsdoppelwort | 1 ¹⁾ /2 | 0,3/0,4 | 0,2 | 0,16 | 0,2 |
| | AD a | Ausgangsdoppelwort | 1 ¹⁾ /2 | 0,3/0,4 | 0,2 | 0,16 | 0,2 |
| | PED a | Peripherie-Eingangsdoppelwort ²⁾ | 2 | 0,3/0,4 | 0,2 | 0,16 | 0,2 |
| | MD a | Merkerdoppelwort | 1 ³⁾ /2 | 0,3/0,4 | 0,2 | 0,16 | 0,2 |
| | LD a | Lokaldatendoppelwort | 2 | 0,4 | 0,2 | 0,16 | 0,2 |
| | DBD a | Datendoppelwort | 2 | 0,4 | 0,2 | 0,16 | 0,2 |
| | DID a | Instanz-Datendoppelwort ... in AKKU1 | 2 | 0,4 | 0,2 | 0,16 | 0,2 |
| | i [d] | speicherindirekt, bereichsintern | 2 | 0,4+ | 0,2+ | 0,16+ | 0,2+ |
| | i [AR1,m] | registerind., bereichsintern (AR1) | 2 | 0,4+ | 0,2+ | 0,16+ | 0,2+ |
| | i [AR2,m] | registerind., bereichsintern (AR2) | 2 | 0,4+ | 0,2+ | 0,16+ | 0,2+ |
| | D[AR1,m] | bereichsübergreifend (AR1) | 2 | 0,4+ | 0,2+ | 0,16+ | 0,2+ |
| | D[AR2,m] | bereichsübergreifend (AR2) | 2 | 0,4+ | 0,2+ | 0,16+ | 0,2+ |
| | Parameter | über Parameter | 2 | 0,4+ | 0,2+ | 0,16+ | 0,2+ |

+ plus Zeit für das Laden der Adresse des Operanden (siehe Seite 19)

1) bei indirekter Adressierung des Operanden; Adreßbereich 0 bis 127

2) plus Reaktionszeit der E/A-Baugruppe (> 1 µs), bei den CPUs 414-4H und 417-4H: solo 41µs, redundant 71µs

3) bei direkter Adressierung des Operanden; Adreßbereich 0 bis 255

Ladeoperationen, Fortsetzung

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μs | | | |
|-----------|-----------|---|-----------------|----------------------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| L | k8 | Lade ... 8-Bit-Konstante in AKKU1-LL | 2 | 0,2 | 0,1 | 0,08 | 0,1 |
| | k16 | 16-Bit-Konstante in AKKU1-L | 2 | 0,2 | 0,1 | 0,08 | 0,1 |
| | k32 | 32-Bit-Konstante in AKKU1 | 3 | 0,3 | 0,15 | 0,12 | 0,15 |
| | Parameter | Lade Konstante in AKKU1 (über Parameter adressiert) | 2 | 0,2/0,3+ | 0,1+ | 0,08+ | 0,1+ |
| L | 2#n | Lade 16-Bit-Binärkonstante in AKKU1-L | 2 | 0,2 | 0,1 | 0,08 | 0,1 |
| | | Lade 32-Bit-Binärkonstante in AKKU1 | 3 | 0,3 | 0,15 | 0,12 | 0,15 |
| | B#16#p | Lade 8-Bit-Hexadezimalkonstante in AKKU1-L | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| L | W#16#p | Lade 16-Bit-Hexadezimalkonstante in AKKU1-L | 2 | 0,2 | 0,1 | 0,08 | 0,1 |
| | DW#16#p | Lade 32-Bit-Hexadezimalkonstante in AKKU1 | 3 | 0,3 | 0,15 | 0,12 | 0,15 |

+ plus Zeit für das Laden der Adresse des Operanden (siehe Seite 19)

Ladeoperationen, Fortsetzung

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μs | | | |
|-----------|---------------------|--|-----------------|----------------------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| L | 'x' | Lade 1 Zeichen | 2 | 0,2 | 0,1 | 0,08 | 0,1 |
| | 'xx' | Lade 2 Zeichen | 2 | 0,2 | 0,1 | 0,08 | 0,1 |
| | 'xxx' | Lade 3 Zeichen | 3 | 0,3 | 0,15 | 0,12 | 0,15 |
| | 'xxxx' | Lade 4 Zeichen | 3 | 0,3 | 0,15 | 0,12 | 0,15 |
| L | D# Zeitwert | Lade IEC-Datumskonstante | 3 | 0,3 | 0,15 | 0,12 | 0,15 |
| L | S5T# Zeitwert | Lade S7-Zeitkonstante (16-Bit) | 2 | 0,2 | 0,1 | 0,08 | 0,1 |
| L | TOD# Zeitwert | Lade IEC-Zeitkonstante | 3 | 0,3 | 0,15 | 0,12 | 0,15 |
| L | T# Zeitwert | Lade 16-Bit-Zeitkonstante | 2 | 0,2 | 0,1 | 0,08 | 0,1 |
| | | Lade 32-Bit-Zeitkonstante | 3 | 0,3 | 0,15 | 0,12 | 0,15 |
| L | C# Zählwert | Lade Zählerkonstante (BCD-kodiert) | 2 | 0,2 | 0,1 | 0,08 | 0,1 |
| L | B# (b1, b2) | Lade Konstante als Byte (b1, b2) | 2 | 0,2 | 0,1 | 0,08 | 0,1 |
| | B# (b1, b2, b3, b4) | Lade Konstante als 4 Byte (b1, b2, b3, b4) | 3 | 0,3 | 0,15 | 0,12 | 0,15 |

Ladeoperationen, Fortsetzung

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μs | | | |
|-----------|----------------|------------------------------|-----------------|----------------------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| L | P# Bitpointer | Lade Bitpointer | 3 | 0,3 | 0,15 | 0,12 | 0,15 |
| L | L# Integerzahl | Lade 32-Bit-Integerkonstante | 3 | 0,3 | 0,15 | 0,12 | 0,15 |
| L | Realzahl | Lade Gleitpunktzahl | 3 | 0,3 | 0,15 | 0,12 | 0,15 |

Ladeoperationen für Timer und Zähler

Laden eines Zeitwertes oder Zählwertes in AKKU1. Zuvor wird der Inhalt von AKKU1 in AKKU2 gerettet. Das Statuswort wird nicht beeinflusst.

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μs | | | |
|-----------|--------------|---|-------------------------|----------------------------------|-------------|---------------|-------------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| L | T f T (e) | Lade Zeitwert | 1 ¹⁾ /2 2 | 0,2/0,3 0,3+ | 0,1 0,1+ | 0,08 0,08+ | 0,1 0,1+ |
| | Timerpara. | Lade Zeitwert (über Parameter adressiert) | 2 | 0,3+ | 0,1+ | 0,08+ | 0,1+ |
| L | Z f Z (e) | Lade Zählwert | 1 ¹⁾ /2 2 | 0,2/0,3 0,3+ | 0,1 0,1+ | 0,08 0,08+ | 0,1 0,1+ |
| | Zählerpara. | Lade Zählwert (über Parameter adressiert) | 2 | 0,3+ | 0,1+ | 0,08+ | 0,1+ |
| LC | T f T (e) | Lade Zeitwert BCD-codiert | 1 ¹⁾ /2 2 | 0,3 0,3+ | 0,3 0,3+ | 0,24 0,24+ | 0,3 0,3+ |
| | Timerpara. | Lade Zeitwert BCD-codiert (über Parameter adressiert) | 2 | 0,3+ | 0,3+ | 0,24+ | 0,3+ |
| LC | Z f Z (e) | Lade Zählwert BCD-codiert | 1 ¹⁾ /2 2 | 0,3 0,3+ | 0,3 0,3+ | 0,24 0,24+ | 0,3 0,3+ |
| | Zählerpara. | Lade Zählwert BCD-codiert (über Parameter adressiert) | 2 | 0,3+ | 0,3+ | 0,24+ | 0,3+ |

+ plus Zeit für das Laden der Adresse des Operanden (siehe Seite 19)

1) bei direkter Adressierung des Operanden; Timer-/Zähler-Nr.: 0 bis 255

Transferoperationen

Transferieren des Inhalts von AKKU1 in den adressierten Operanden. MCR-Abhängigkeit beachten (siehe Seite 97). Das Statuswort wird nicht beeinflusst.

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μs | | | |
|-----------|-----------|---|--------------------|----------------------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| T | EB a | Transferiere Inhalt von AKKU1-LL zum ... Eingangbyte | 1 ¹⁾ /2 | 0,2/0,3 | 0,1 | 0,08 | 0,1 |
| | AB a | Ausgangsbyte | 1 ¹⁾ /2 | 0,2/0,3 | 0,1 | 0,08 | 0,1 |
| | PAB a | Peripherie-Ausgangsbyte ²⁾ | 2 | 0,3 | 0,1 | 0,08 | 0,1 |
| | MB a | Merkerbyte | 1 ³⁾ /2 | 0,2/0,3 | 0,1 | 0,08 | 0,1 |
| | LB a | Lokaldatenbyte | 2 | 0,3 | 0,1 | 0,08 | 0,1 |
| | DBB a | Datenbyte | 2 | 0,3 | 0,1 | 0,08 | 0,1 |
| | DIB a | Instanz-Datenbyte | 2 | 0,3 | 0,1 | 0,08 | 0,1 |
| | g [d] | speicherindirekt, bereichsintern | 2 | 0,3+ | 0,1+ | 0,08+ | 0,1+ |
| | g [AR1,m] | registerind., bereichsintern (AR1) | 2 | 0,3+ | 0,1+ | 0,08+ | 0,1+ |
| | g [AR2,m] | registerind., bereichsintern (AR2) | 2 | 0,3+ | 0,1+ | 0,08+ | 0,1+ |
| | B[AR1,m] | bereichsübergreifend (AR1) | 2 | 0,3+ | 0,1+ | 0,08+ | 0,1+ |
| | B[AR2,m] | bereichsübergreifend (AR2) | 2 | 0,3+ | 0,1+ | 0,08+ | 0,1+ |
| | Parameter | über Parameter | 2 | 0,3+ | 0,1+ | 0,08+ | 0,1+ |

+ plus Zeit für das Laden der Adresse des Operanden (siehe Seite 19)

1) bei direkter Adressierung des Operanden; Adreßbereich 0 bis 127

2) Peripherie-Quittierungszeit muß berücksichtigt werden, bei den CPUs 414-4H und 417-4H: solo 29 μs , redundant 58 μs

3) bei direkter Adressierung des Operanden; Adreßbereich 0 bis 255

Transferoperationen, Fortsetzung

Bei direkter Adressierung erfordert der Zugriff auf ungerade Wortadressen bei allen Befehlen der CPU 417 einen Zuschlag von 0,1 µs bei allen Befehlen der CPU 416 einen Zuschlag von 0,08 µs, bei allen Befehlen der CPU 414 einen Zuschlag von 0,1 µs und bei allen Befehlen der CPU 412 einen Zuschlag von 0,1 µs.

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in µs | | | |
|-----------|----------------|---|--------------------|-----------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| T | | Transferiere Inhalt von AKKU1-L zum ... | | | | | |
| | EW a | Eingangswort | 1 ¹⁾ /2 | 0,2/0,3 | 0,1 | 0,08 | 0,1 |
| | AW a | Ausgangswort | 1 ¹⁾ /2 | 0,2/0,3 | 0,1 | 0,08 | 0,1 |
| | PAW a | Peripherie-Ausgangswort ²⁾ | 2 | 0,3 | 0,1 | 0,08 | 0,1 |
| | MW a | Merkerwort | 1 ³⁾ /2 | 0,2/0,3 | 0,1 | 0,08 | 0,1 |
| | LW a | Lokaldatenwort | 2 | 0,3 | 0,1 | 0,08 | 0,1 |
| | DBW a | Datenwort | 2 | 0,3 | 0,1 | 0,08 | 0,1 |
| | DIW a | Instanz-Datenwort | 2 | 0,3 | 0,1 | 0,08 | 0,1 |
| | h [d] | speicherindirekt, bereichsintern | 2 | 0,3+ | 0,1+ | 0,08+ | 0,1+ |
| | h [AR1,m] | registerind., bereichsintern (AR1) | 2 | 0,3+ | 0,1+ | 0,08+ | 0,1+ |
| | h [AR2,m] | registerind., bereichsintern (AR2) | 2 | 0,3+ | 0,1+ | 0,08+ | 0,1+ |
| | W[AR1,m] | bereichsübergreifend (AR1) | 2 | 0,3+ | 0,1+ | 0,08+ | 0,1+ |
| | W[AR2,m] | bereichsübergreifend (AR2) | 2 | 0,3+ | 0,1+ | 0,08+ | 0,1+ |
| Parameter | über Parameter | 2 | 0,3+ | 0,1+ | 0,08+ | 0,1+ | |

+ plus Zeit für das Laden der Adresse des Operanden (siehe Seite 19)

1) bei direkter Adressierung des Operanden; Adreßbereich 0 bis 127

2) Peripherie-Quittierungszeit muß berücksichtigt werden, bei den CPUs 414-4H und 417-4H: solo 32µs, redundant 61µs

3) bei direkter Adressierung des Operanden; Adreßbereich 0 bis 255

Transferoperationen, Fortsetzung

Die in der Tabelle angegebenen Ausführungszeiten bei direkter Adressierung erhöhen sich um 0,1 µs (CPU 412), 0,1 µs (CPU 414), 0,08 µs (CPU 416) bzw. 0,1 µs (CPU 417), wenn der Zugriff auf ungerade Adressen erfolgt. MCR-Abhängigkeit beachten (siehe Seite 97).

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in µs | | | |
|-----------|-----------|---|--------------------|-----------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| T | | Transferiere Inhalt von AKKU1 zum ... | | | | | |
| | ED a | Eingangsdoppelwort | 1 ¹⁾ /2 | 0,3/0,4 | 0,2 | 0,16 | 0,2 |
| | AD a | Ausgangsdoppelwort | 1 ¹⁾ /2 | 0,3/0,4 | 0,2 | 0,16 | 0,2 |
| | PAD a | Peripherie-Ausgangsdoppelwort ²⁾ | 2 | 0,4 | 0,2 | 0,16 | 0,2 |
| | MD a | Merkerdoppelwort | 1 ³⁾ /2 | 0,3/0,4 | 0,2 | 0,16 | 0,2 |
| | LD a | Lokaldatendoppelwort | 2 | 0,4 | 0,2 | 0,16 | 0,2 |
| | DBD a | Datendoppelwort | 2 | 0,4 | 0,2 | 0,16 | 0,2 |
| | DID a | Instanz-Datendoppelwort | 2 | 0,4 | 0,2 | 0,16 | 0,2 |
| T | i [d] | speicherindirekt, bereichsintern | 2 | 0,4+ | 0,2+ | 0,16+ | 0,2+ |
| | i [AR1,m] | registerind., bereichsintern (AR1) | 2 | 0,4+ | 0,2+ | 0,16+ | 0,2+ |
| | i [AR2,m] | registerind., bereichsintern (AR2) | 2 | 0,4+ | 0,2+ | 0,16+ | 0,2+ |
| | D[AR1,m] | bereichsübergreifend (AR1) | 2 | 0,4+ | 0,2+ | 0,16+ | 0,2+ |
| | D[AR2,m] | bereichsübergreifend (AR2) | 2 | 0,4+ | 0,2+ | 0,16+ | 0,2+ |
| | Parameter | über Parameter | 2 | 0,4+ | 0,2+ | 0,16+ | 0,2+ |

+ plus Zeit für das Laden der Adresse des Operanden (siehe Seite 19)

1) bei direkter Adressierung des Operanden; Adreßbereich 0 bis 127

2) Peripherie-Quittierungszeit muß berücksichtigt werden, bei den CPUs 414-4H und 417-4H: solo 36µs, redundant 65µs

3) bei direkter Adressierung des Operanden; Adreßbereich 0 bis 255

Lade- und Transferoperationen für Adreßregister

Laden eines Doppelwortes aus einem Speicher oder einem Register in Adreßregister 1 (AR1) oder Adreßregister 2 (AR2). Das Statuswort wird nicht beeinflusst.

| Opera- tion | Operand | Bedeutung | Län- ge in Wor- ten | Ausführungszeit in μ s | | | |
|----------------|---------|--------------------------------|------------------------------|----------------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| LAR1 | – | Lade Inhalt aus ... AKKU1 | 1 | 0,2 | 0,2 | 0,16 | 0,2 |
| | AR2 | Adreßregister 2 | 1 | 0,2 | 0,2 | 0,16 | 0,2 |
| | DBD a | Datendoppelwort | 2 | 0,4 | 0,3 | 0,24 | 0,3 |
| | DID a | Instanz-Datendoppelwort | 2 | 0,4 | 0,3 | 0,24 | 0,3 |
| | m | 32-Bit-Konstante als Pointer | 3 | 0,3 | 0,2 | 0,16 | 0,2 |
| | LD a | Lokaldatendoppelwort | 2 | 0,4 | 0,3 | 0,24 | 0,3 |
| | MD a | Merkerdoppelwort ... in AR1 | 2 | 0,4 | 0,3 | 0,24 | 0,3 |
| LAR2 | – | Lade Inhalt aus ... AKKU1 | 1 | 0,2 | 0,2 | 0,16 | 0,2 |
| | DBD a | Datendoppelwort | 2 | 0,4 | 0,3 | 0,24 | 0,3 |
| | DID a | Instanz-Datendoppelwort | 2 | 0,4 | 0,3 | 0,24 | 0,3 |
| | m | 32-Bit-Konstante als Pointer | 3 | 0,3 | 0,2 | 0,16 | 0,2 |
| | LD a | Lokaldatendoppelwort | 2 | 0,4 | 0,3 | 0,24 | 0,3 |
| | MD a | Merkerdoppelwort ... in AR2 | 2 | 0,4 | 0,3 | 0,24 | 0,3 |

Lade- und Transferoperationen für Adreßregister, Fortsetzung

Transferieren eines Doppelwortes aus Adreßregister 1 (AR1) oder Adreßregister 2 (AR2) in einen Speicher oder ein Register. Zuvor wird der Inhalt von AKKU1 in AKKU2 gerettet. Das Statuswort wird nicht beeinflußt.

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μs | | | |
|-----------|---------|---|-----------------|----------------------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| TAR1 | – | Transferiere Inhalt aus AR1 in ... AKKU1 | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| | AR2 | Adreßregister 2 | 1 | 0,2 | 0,2 | 0,16 | 0,2 |
| | DBD a | Datendoppelwort | 2 | 0,4 | 0,2 | 0,16 | 0,2 |
| | DID a | Instanz-Datendoppelwort | 2 | 0,4 | 0,2 | 0,16 | 0,2 |
| | LD a | Lokaldatendoppelwort | 2 | 0,4 | 0,2 | 0,16 | 0,2 |
| | MD a | Merkerdoppelwort | 2 | 0,4 | 0,2 | 0,16 | 0,2 |
| TAR2 | – | Transferiere Inhalt aus AR2 in ... AKKU1 | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| | DBD a | Datendoppelwort | 2 | 0,4 | 0,2 | 0,16 | 0,2 |
| | DID a | Instanz-Datendoppelwort | 2 | 0,4 | 0,2 | 0,16 | 0,2 |
| | LD a | Lokaldatendoppelwort | 2 | 0,4 | 0,2 | 0,16 | 0,2 |
| | MD a | Merkerdoppelwort | 2 | 0,4 | 0,2 | 0,16 | 0,2 |
| TAR | | Tausche die Inhalte von AR1 und AR2 | 1 | 0,4 | 0,4 | 0,32 | 0,4 |

Lade- und Transferoperationen für das Statuswort

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μs | | | |
|-----------|---------|--------------------------|-----------------|----------------------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| L | STW | Lade Statuswort in AKKU1 | 1 | 0,1 | 0,1 | 0,08 | 0,1 |

| | | | | | | | | | | |
|------------------------|--------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: | L STW | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | | ja | ja | ja | ja | ja | ja | ja | ja | ja |
| Operation beeinflusst: | | – | – | – | – | – | – | – | – | – |

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μs | | | |
|-----------|---------|---|-----------------|----------------------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| T | STW | Transferiere AKKU1 (Bits 0 bis 8) in das Statuswort | 1 | 0,1 | 0,1 | 0,08 | 0,1 |

| | | | | | | | | | | |
|------------------------|--------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: | T STW | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | | – | – | – | – | – | – | – | – | – |
| Operation beeinflusst: | | ja | ja | ja | ja | ja | ja | ja | ja | ja |

Ladeoperationen für DB-Nummer und DB-Länge

Laden der Nummer/Länge eines Datenbausteins in AKKU1. Der alte Inhalt von AKKU1 wird in AKKU2 gerettet. Das Statuswort wird nicht beeinflusst.

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μs | | | |
|-----------|---------|---|-----------------|----------------------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| L | DBNO | Lade Nummer des Datenbausteins | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| L | DINO | Lade Nummer des Instanz-Datenbausteins | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| L | DBLG | Lade Länge des Datenbausteins in Byte | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| L | DILG | Lade Länge des Instanz-Datenbausteins in Byte | 1 | 0,1 | 0,1 | 0,08 | 0,1 |

Festpunktarithmetik (16 Bit)

Arithmetische Operationen zweier 16-Bit-Zahlen. Das Ergebnis wird in AKKU1 bzw. AKKU1-L geschrieben. Danach werden AKKU3 und AKKU4 nach AKKU2 und AKKU3 übertragen.

| Opera- tion | Operand | Bedeutung | Län- ge in Wor- ten | Ausführungszeit in μ s | | | |
|----------------|---------|--|------------------------------|----------------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| +I | | Addiere 2 Integerzahlen (16-Bit) (AKKU1-L)=(AKKU1-L)+ (AKKU2-L) | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| -I | | Subtrahiere 2 Integerzahlen (16-Bit) (AKKU1-L)=(AKKU2-L)- (AKKU1-L) | 1 | 0,1 | 0,1 | 0,08 | 0,1 |

| | | | | | | | | | |
|-------------------------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: +I, -I | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | - | - | - | - | - | - | - | - | - |
| Operation beeinflusst: | - | ja | ja | ja | ja | - | - | - | - |

Festpunktarithmetik (16 Bit), Fortsetzung

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μ s | | | |
|-----------|---------|--|-----------------|----------------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| *I | | Multipliziere 2 Integerzahlen (16-Bit) (AKKU1)=(AKKU2-L)* (AKKU1-L) | 1 | 0,8 | 0,8 | 0,64 | 0,8 |
| /I | | Dividiere 2 Integerzahlen (16-Bit) (AKKU1-L)=(AKKU2-L): (AKKU1-L) Im AKKU1-H steht der Rest der Division. | 1 | 0,8 | 0,8 | 0,64 | 0,8 |

| | | | | | | | | | |
|-------------------------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: *I, /I | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | – | – | – | – | – | – | – | – | – |
| Operation beeinflusst: | – | ja | ja | ja | ja | – | – | – | – |

Festpunktarithmetik (32 Bit)

Arithmetische Operationen zweier 32-Bit-Zahlen. Das Ergebnis wird in AKKU1 geschrieben. Danach werden AKKU3 und AKKU4 nach AKKU2 und AKKU3 übertragen.

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μs | | | |
|-----------|---------|---|-----------------|----------------------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| +D | | Addiere 2 Integerzahlen (32-Bit) (AKKU1)=(AKKU2)+(AKKU1) | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| -D | | Subtrahiere 2 Integerzahlen (32-Bit) (AKKU1)=(AKKU2)-(AKKU1) | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| *D | | Multipliziere 2 Integerzahlen (32-Bit) (AKKU1)=(AKKU2)*(AKKU1) | 1 | 1,3 | 1,3 | 1,04 | 1,3 |

| | | | | | | | | | |
|----------------------------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: +D, -D,*D | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | - | - | - | - | - | - | - | - | - |
| Operation beeinflusst: | - | ja | ja | ja | ja | - | - | - | - |

Festpunktarithmetik (32 Bit), Fortsetzung

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μs | | | |
|-----------|---------|--|-----------------|----------------------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| /D | | Dividiere 2 Integerzahlen (32-Bit) (AKKU1)=(AKKU2):(AKKU1) | 1 | 1,3 | 1,3 | 1,04 | 1,3 |
| MOD | | Dividiere 2 Integerzahlen (32-Bit) und lade den Rest der Division in AKKU1: (AKKU1)=Rest von [(AKKU2):(AKKU1)] | 1 | 1,3 | 1,3 | 1,04 | 1,3 |

| | | | | | | | | | |
|--------------------------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: /D, MOD | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | – | – | – | – | – | – | – | – | – |
| Operation beeinflusst: | – | ja | ja | ja | ja | – | – | – | – |

Gleitpunktarithmetik (32 Bit)

Das Ergebnis der arithmetischen Operationen steht im AKKU1. Danach werden AKKU 3 und AKKU 4 nach AKKU 2 und AKKU 3 übertragen.

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μs | | | |
|-----------|---------|--|-----------------|----------------------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| +R | | Addiere 2 Realzahlen (32-Bit) (AKKU1)=(AKKU2)+(AKKU1) | 1 | 0,6 | 0,6 | 0,48 | 0,6 |
| -R | | Subtrahiere 2 Realzahlen (32-Bit) (AKKU1)=(AKKU2)-(AKKU1) | 1 | 0,6 | 0,6 | 0,48 | 0,6 |
| *R | | Multipliziere 2 Realzahlen (32-Bit) (AKKU1)=(AKKU2)*(AKKU1) | 1 | 1,4 | 1,4 | 1,12 | 1,4 |
| /R | | Dividiere 2 Realzahlen (32-Bit) (AKKU1)=(AKKU2):(AKKU1) | 1 | 2,1 | 2,1 | 1,68 | 2,1 |

| | | | | | | | | | |
|---------------------------------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: +R, -R, *R, /R | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | - | - | - | - | - | - | - | - | - |
| Operation beeinflusst: | - | ja | ja | ja | ja | - | - | - | - |

Gleitpunktarithmetik (32 Bit), Fortsetzung

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μ s | | | |
|-----------|---------|------------------------------------|-----------------|----------------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| NEGR | | Negiere Realzahl im AKKU1 | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| ABS | | Bilde Betrag der Realzahl im AKKU1 | 1 | 0,1 | 0,1 | 0,08 | 0,1 |

| | | | | | | | | | |
|----------------------------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: NEGR, ABS | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | – | – | – | – | – | – | – | – | – |
| Operation beeinflusst: | – | – | – | – | – | – | – | – | – |

Quadratwurzel, Quadrat (32 Bit)

Das Ergebnis der Operation steht im AKKU1. Die Operation SQRT ist durch Alarmer unterbrechbar.

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μs | | | |
|-----------|---------|--|-----------------|----------------------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| SQRT | | Berechne die Quadratwurzel einer Realzahl in AKKU1 | 1 | 72 | 40 | 37 - 39 | 40 |
| SQR | | Quadriere die Realzahl in AKKU1 | 1 | 1,4 | 1,4 | 1,12 | 1,4 |

| | | | | | | | | | |
|----------------------------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: SQRT, SQR | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | – | – | – | – | – | – | – | – | – |
| Operation beeinflusst: | – | ja | ja | ja | ja | – | – | – | – |

Logarithmusfunktionen (32 Bit)

Das Ergebnis der Logarithmusfunktion steht im AKKU1. Die Operationen sind durch Alarmer unterbrechbar.

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μs | | | |
|-----------|---------|--|-----------------|----------------------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| LN | | Bilde den natürlichen Logarithmus einer Realzahl in AKKU1 | 1 | 63 | 35 | 33 | 35 |
| EXP | | Berechne den Exponentialwert einer Realzahl in AKKU1 zur Basis e (= 2,71828) | 1 | 63 | 35 | 32 - 34 | 35 |

| | | | | | | | | | |
|--------------------------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: LN, EXP | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | - | - | - | - | - | - | - | - | - |
| Operation beeinflusst: | - | ja | ja | ja | ja | - | - | - | - |

Trigonometrische Funktionen (32 Bit)

Das Ergebnis der Operation steht im AKKU1. Die Operationen sind durch Alarmer unterbrechbar.

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μs | | | |
|-----------|---------|--|-----------------|----------------------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| SIN | | Berechne den Sinus einer Realzahl | 1 | 56 | 31 | 30 | 31 |
| ASIN | | Berechne den Arcussinus einer Realzahl | 1 | 117 - 133 | 65 - 74 | 62 - 70 | 65 - 74 |
| COS | | Berechne den Cosinus einer Realzahl | 1 | 58 | 32 | 30 | 32 |
| ACOS | | Berechne den Arcuscosinus einer Realzahl | 1 | 122 - 139 | 68 - 77 | 65 - 72 | 68 - 77 |
| TAN | | Berechne den Tangens einer Realzahl | 1 | 58 - 63 | 32 - 35 | 30 - 33 | 32 - 35 |
| ATAN | | Berechne den Arcustangens einer Realzahl | 1 | 43 - 58 | 24 - 32 | 23 - 30 | 24 - 32 |

| | | | | | | | | | | |
|------------------------|--|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: | SIN, ASIN, COS, ACOS, TAN, ATAN | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | | - | - | - | - | - | - | - | - | - |
| Operation beeinflusst: | | - | ja | ja | ja | ja | - | - | - | - |

Addition von Konstanten

Addition von Integer-Konstanten zum AKKU1. Das Statuswort wird nicht beeinflusst.

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μs | | | |
|-----------|---------|---------------------------------------|-----------------|----------------------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| + | i8 | Addiere eine 8-Bit-Integer-Konstante | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| + | i16 | Addiere eine 16-Bit-Integer-Konstante | 2 | 0,2 | 0,1 | 0,08 | 0,1 |
| + | i32 | Addiere eine 32-Bit-Integer-Konstante | 3 | 0,3 | 0,15 | 0,12 | 0,15 |

Addition über Adreßregister

Addition einer 16-Bit-Integerzahl zum Inhalt des Adreßregisters. Der Wert wird entweder als Operand bei der Operation angegeben oder steht im AKKU1-L. Das Statuswort wird nicht beeinflußt.

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μs | | | |
|-----------|----------------|------------------------------------|-----------------|----------------------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| +AR1 | | Addiere Inhalt von AKKU1-L zum AR1 | 1 | 0,2 | 0,2 | 0,16 | 0,2 |
| +AR1 | m (0 bis 4095) | Addiere Pointer-Konstante zum AR1 | 2 | 0,2 | 0,2 | 0,16 | 0,2 |
| +AR2 | | Addiere Inhalt von AKKU1-L zum AR2 | 1 | 0,2 | 0,2 | 0,16 | 0,2 |
| +AR2 | m (0 bis 4095) | Addiere Pointer-Konstante zum AR2 | 2 | 0,2 | 0,2 | 0,16 | 0,2 |

Vergleichsoperationen (16-Bit-Integerzahlen)

Vergleich der 16-Bit-Integerzahlen in AKKU1-L und AKKU2-L. VKE=1, wenn Bedingung erfüllt.

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μ s | | | |
|-----------|---------|------------------------|-----------------|----------------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| = | | AKKU2-L=AKKU1-L | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| <> | | AKKU2-L \neq AKKU1-L | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| < | | AKKU2-L<AKKU1-L | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| <= | | AKKU2-L<=AKKU1-L | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| > | | AKKU2-L>AKKU1-L | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| >= | | AKKU2-L>=AKKU1-L | 1 | 0,1 | 0,1 | 0,08 | 0,1 |

| | | | | | | | | | |
|--|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: = , <> , < , <= , > , >= | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | - | - | - | - | - | - | - | - | - |
| Operation beeinflusst: | - | ja | ja | 0 | - | 0 | ja | ja | 1 |

Vergleichsoperationen (32-Bit-Integerzahlen)

Vergleich der 32-Bit-Integerzahlen in AKKU1 und AKKU2. VKE=1, wenn Bedingung erfüllt.

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μ s | | | |
|-----------|---------|--------------------|-----------------|----------------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| ==D | | AKKU2=AKKU1 | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| <>D | | AKKU2 \neq AKKU1 | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| <D | | AKKU2<AKKU1 | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| <=D | | AKKU2<=AKKU1 | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| >D | | AKKU2>AKKU1 | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| >=D | | AKKU2>=AKKU1 | 1 | 0,1 | 0,1 | 0,08 | 0,1 |

| | | | | | | | | | |
|--|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: ==D,< >D, <D, <=D, >D, >=D | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | – | – | – | – | – | – | – | – | – |
| Operation beeinflusst: | – | ja | ja | 0 | – | 0 | ja | ja | 1 |

Vergleichsoperationen (32-Bit-Realzahlen)

Vergleich der 32-Bit-Realzahlen in AKKU1 und AKKU2.
VKE=1, wenn Bedingung erfüllt.

| Opera- tion | Operand | Bedeutung | Län- ge in Wor- ten | Ausführungszeit in μ s | | | |
|----------------|---------|--------------------|------------------------------|----------------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| ==R | | AKKU2=AKKU1 | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| <>R | | AKKU2 \neq AKKU1 | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| <R | | AKKU2<AKKU1 | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| <=R | | AKKU2<=AKKU1 | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| >R | | AKKU2>AKKU1 | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| >=R | | AKKU2>=AKKU1 | 1 | 0,1 | 0,1 | 0,08 | 0,1 |

| | | | | | | | | | |
|--|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: ==R, <>R, <R, <=R, >R, >=R | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | – | – | – | – | – | – | – | – | – |
| Operation beeinflusst: | – | ja | ja | ja | ja | 0 | ja | ja | 1 |

Schiebeoperationen

Schiebe Inhalt von AKKU1 oder AKKU1-L um die angegebene Anzahl von Stellen nach links/rechts. Ist kein Operand angegeben, wird als Anzahl der Inhalt von AKKU2-LL genommen. Das zuletzt geschobene Bit wird ins Anzeigenbit A1 geladen.

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μ s | | | |
|-------------------|----------|--|-----------------|----------------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| SLW ¹⁾ | 0 ... 15 | Schiebe Inhalt von AKKU1-L nach links. Freiwerdende Stellen werden mit Nullen aufgefüllt. | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| SLW | | | | | | | |
| SLD | 0 ... 32 | Schiebe Inhalt von AKKU1 nach links. Freiwerdende Stellen werden mit Nullen aufgefüllt. | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| SLD | | | | | | | |
| SRW ¹⁾ | 0 ... 15 | Schiebe Inhalt von AKKU1-L nach rechts. Freiwerdende Stellen werden mit Nullen aufgefüllt. | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| SRW | | | | | | | |

| | | | | | | | | | | |
|------------------------|-----------------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: | SLW, SLD, SRW, | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | | – | – | – | – | – | – | – | – | – |
| Operation beeinflusst: | | – | ja | 0 | 0 | – | – | – | – | – |

1) Anzahl der geschobenen Stellen: 0 bis 16

Schiebeoperationen, Fortsetzung

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μ s | | | |
|-------------------|----------|---|-----------------|----------------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| SRD | | Schiebe Inhalt von AKKU1 nach rechts. Freierdende Stellen werden mit Nullen aufgefüllt. | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| SRD | 0 ... 32 | | | | | | |
| SSI ¹⁾ | | Schiebe Inhalt von AKKU1-L mit Vorzeichen nach rechts. Freierdende Stellen werden mit dem Vorzeichen (Bit 15) aufgefüllt. | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| SSI | 0 ... 15 | | | | | | |
| SSD | | Schiebe Inhalt von AKKU1 mit Vorzeichen nach rechts. Freierdende Stellen werden mit dem Vorzeichen (Bit 31) aufgefüllt | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| SSD | 0 ... 32 | | | | | | |

| | | | | | | | | | |
|--------------------------------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: SRD, SSI, SSD | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | – | – | – | – | – | – | – | – | – |
| Operation beeinflusst: | – | ja | 0 | 0 | – | – | – | – | – |

1) Anzahl der geschobenen Stellen: 0 bis 16

Rotieroperationen

Rotiere Inhalt von AKKU1 um die angegebene Anzahl von Stellen nach links/rechts. Ist kein Operand angegeben, wird als Anzahl der Inhalt von AKKU2-LL genommen. Das zuletzt geschobene Bit wird ins Anzeigenbit A1 geladen.

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μ s | | | |
|-----------|----------|--------------------------------------|-----------------|----------------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| RLD | | Rotiere Inhalt von AKKU1 nach links | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| RLD | 0 ... 32 | | | | | | |
| RRD | | Rotiere Inhalt von AKKU1 nach rechts | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| RRD | 0 ... 32 | | | | | | |

| | | | | | | | | | | |
|------------------------|-----------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: | RLD, RRD | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | | – | – | – | – | – | – | – | – | – |
| Operation beeinflusst: | | – | ja | ja | ja | – | – | – | – | – |

Rotieroperationen, Fortsetzung

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μs | | | |
|-----------|---------|--|-----------------|----------------------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| RLDA | | Rotiere Inhalt von AKKU1 um eine Bitposition nach links über Anzeigenbit A1 | | 0,1 | 0,1 | 0,08 | 0,1 |
| RRDA | | Rotiere Inhalt von AKKU1 um eine Bitposition nach rechts über Anzeigenbit A1 | | 0,1 | 0,1 | 0,08 | 0,1 |

| | | | | | | | | | | |
|------------------------|-------------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: | RLDA, RRDA | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | | – | – | – | – | – | – | – | – | – |
| Operation beeinflusst: | | – | ja | 0 | 0 | – | – | – | – | – |

AKKU-Transferoperationen, Inkrementieren, Dekrementieren

Das Statuswort wird nicht beeinflusst.

| Opera- tion | Operand | Bedeutung | Län- ge in Wor- ten | Ausführungszeit in μs | | | |
|----------------|---------|---|------------------------------|----------------------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| TAW | | Umkehr der Reihenfolge der Bytes im AKKU1-L. | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| TAD | | Umkehr der Reihenfolge der Bytes in AKKU1. | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| TAK | | Tausche Inhalte von AKKU1 und AKKU2 | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| ENT | | Inhalt von AKKU2 und AKKU3 wird nach AKKU3 und AKKU4 übertragen. | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| LEAVE | | Inhalt von AKKU3 und AKKU4 wird nach AKKU2 und AKKU3 übertragen. | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| PUSH | | Inhalt von AKKU1, AKKU2 und AKKU3 wird nach AKKU2, AKKU3 und AKKU4 übertragen | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| POP | | Inhalt von AKKU2, AKKU3 und AKKU4 wird nach AKKU1, AKKU2 und AKKU3 übertragen | 1 | 0,1 | 0,1 | 0,08 | 0,1 |

AKKU-Transferoperationen, Inkrementieren, Dekrementieren, Fortsetzung

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μs | | | |
|-----------|---------|------------------------|-----------------|----------------------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| INC | k8 | Inkrementiere AKKU1-LL | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| DEC | k8 | Dekrementiere AKKU1-LL | 1 | 0,1 | 0,1 | 0,08 | 0,1 |

Bildaufbauoperation, Nulloperation

Das Statuswort wird nicht beeinflusst.

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μs | | | |
|-----------|---------|--|-----------------|----------------------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| BLD | k8 | Bildaufbauoperation; wird von der CPU wie eine Nulloperation behandelt. | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| NOP | 0 1 | Nulloperation | 1 | 0,1 | 0,1 | 0,08 | 0,1 |

Datentyp-Umwandlungsoperationen

Die Ergebnisse der Wandlung stehen im AKKU1.

| Opera- tion | Ope- rand | Bedeutung | Län- ge in Wor- ten | Ausführungszeit in μ s | | | |
|----------------|--------------|--|------------------------------|----------------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| BTI | | Konvertiere AKKU1-L von BCD (0 bis +/- 999) in Integerzahl (16 Bit) (BCD To Int) | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| BTD | | Konvertiere AKKU1 von BCD (0 bis +/- 9 999 999) in Double-Integerzahl (32 Bit) (BCD To Doubleint) | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| DTR | | Konvertiere AKKU1 von Double-Integerzahl (32 Bit) in Realzahl (32 Bit) (Doubleint To Real) | 1 | 0,3 | 0,3 | 0,24 | 0,3 |
| ITD | | Konvertiere AKKU1 von Integerzahl (16 Bit) in Double-Integerzahl (32 Bit) (Int To Doubleint) | 1 | 0,1 | 0,1 | 0,08 | 0,1 |

| | | | | | | | | | | |
|------------------------|---------------------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: | BTI, BTD, DTR, ITD | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | | – | – | – | – | – | – | – | – | – |
| Operation beeinflusst: | | – | – | – | – | – | – | – | – | – |

Datentyp-Umwandlungsoperationen, Fortsetzung

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μs | | | |
|-----------|---------|--|-----------------|----------------------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| ITB | | Konvertiere AKKU1-L von Integerzahl (16 Bit) nach BCD 0 bis +/- 999 (Int To BCD) | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| DTB | | Konvertiere AKKU1 von Double-Integerzahl (32 Bit) nach BCD 0 bis +/- 9 999 999 (Doubleint To BCD) | 1 | 0,2 | 0,2 | 0,16 | 0,2 |

| | | | | | | | | | |
|---------------------------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: ITB, DTB | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | - | - | - | - | - | - | - | - | - |
| Operation beeinflusst: | - | - | - | ja | ja | - | - | - | - |

Datentyp-Umwandlungsoperationen, Fortsetzung

Die zu wandelnde Realzahl steht in AKKU1.

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μ s | | | |
|-----------|---------|---|-----------------|----------------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| RND | | Wandle Realzahl in 32-Bit-Integerzahl um. | 1 | 0,4 | 0,4 | 0,32 | 0,4 |
| RND- | | Wandle Realzahl in 32-Bit-Integerzahl um. Es wird abgerundet zur nächsten ganzen Zahl. | 1 | 0,4 | 0,4 | 0,32 | 0,4 |
| RND+ | | Wandle Realzahl in 32-Bit-Integerzahl um. Es wird aufgerundet zur nächsten ganzen Zahl. | 1 | 0,4 | 0,4 | 0,32 | 0,4 |
| TRUNC | | Wandle Realzahl in 32-Bit-Integerzahl um. Es werden die Nachkommastellen abgeschnitten. | 1 | 0,4 | 0,4 | 0,32 | 0,4 |

| | | | | | | | | | | |
|------------------------|-------------------------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: | RND, RND-, RND+, TRUNC | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | | – | – | – | – | – | – | – | – | – |
| Operation beeinflusst: | | – | – | – | ja | ja | – | – | – | – |

Komplementbildung

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μs | | | |
|-----------|---------|----------------------------------|-----------------|----------------------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| INVI | | Bilde 1er-Komplement von AKKU1-L | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| INVD | | Bilde 1er-Komplement von AKKU1 | 1 | 0,1 | 0,1 | 0,08 | 0,1 |

| | | | | | | | | | |
|-----------------------------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: INVI, INVD | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | – | – | – | – | – | – | – | – | – |
| Operation beeinflusst: | – | – | – | – | – | – | – | – | – |

| | | | | | | | |
|------|--|---|---|-----|-----|------|-----|
| NEGI | | Bilde 2er-Komplement von AKKU1-L (Integerzahl) | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
| NEGD | | Bilde 2er-Komplement von AKKU1 (Double-Integerzahl) | 1 | 0,1 | 0,1 | 0,08 | 0,1 |

| | | | | | | | | | |
|-----------------------------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: NEGI, NEGD | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | – | – | – | – | – | – | – | – | – |
| Operation beeinflusst: | – | ja | ja | ja | ja | – | – | – | – |

Baustein-Aufrufoperationen

Die Laufzeiten der SFCs sind im Kapitel "Systemfunktionen" ab Seite 106 angegeben. Die Aussagen zum Statuswort beziehen sich nur auf den Bausteinaufruf selbst und nicht auf die in diesem Baustein ausgeführten Befehle.

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μs | | | |
|-----------|-------------|---|--------------------|----------------------------------|-------------------|--------------------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| CALL | FB q, DB q | Unbedingter Aufruf eines FB mit Parameterübergabe | 1 ¹⁾ /2 | 8,2 ³⁾ | 3,2 ³⁾ | 2,56 ³⁾ | XX |
| CALL | SFB q, DB q | Unbedingter Aufruf eines SFB, mit Parameterübergabe | 2 | 8,2 ³⁾ | 3,2 ³⁾ | 2,56 ³⁾ | XX |
| CALL | FC q | Unbedingter Aufruf einer Funktion mit Parameterübergabe | 1 ¹⁾ /2 | 4,6 ³⁾ | 1,8 ³⁾ | 1,44 ³⁾ | XX |
| CALL | SFC q | Unbedingter Aufruf einer SFC, mit Parameterübergabe | 2 | 4,6 ³⁾ | 1,8 ³⁾ | 1,44 ³⁾ | XX |

| | | | | | | | | | |
|-----------------------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: CALL | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | – | – | – | – | – | – | – | – | – |
| Operation beeinflusst: | – | – | – | – | 0 | 0 | 1 | – | 0 |

1) bei direkter Adressierung des Operanden Baustein-Nr. 0 bis 255

3) plus Zeit für Parameterversorgung

Baustein-Aufrufoperationen, Fortsetzung

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in µs | | | |
|-----------|-----------|--|--------------------|---------------------------|------------------------|--------------------------|------------------------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| UC | FB q | Unbedingter Aufruf von Bausteinen ohne Parameterübergabe | 1 ¹⁾ /2 | 2,1/2,2 | 1,4 | 1,12 | 1,4 |
| | FC q | speicherindirekter FB-Aufruf | 2 | 2,1/2,2 | 1,4 | 1,12 | 1,4 |
| | FB [e] | speicherindirekter FC-Aufruf | 2 | 2,2+ | 1,4+ | 1,12+ | 1,4+ |
| | FC [e] | FB/FC-Aufruf über Parameter | 2 | 2,2+ | 1,4+ | 1,12+ | 1,4+ |
| | Parameter | | 2 | 2,2+ | 1,4+ | 1,12+ | 1,4+ |
| CC | FB q | Bedingter Aufruf von Bausteinen ohne Parameterübergabe | 1 ¹⁾ /2 | 2,3/2,4/0,4 ⁴⁾ | 1,4/0,4 ⁴⁾ | 1,12/0,32 ⁴⁾ | 1,4/0,4 ⁴⁾ |
| | FC q | speicherindirekter FB-Aufruf | 2 | 2,3/2,4/0,4 ⁴⁾ | 1,4/0,4 ⁴⁾ | 1,12/0,32 ⁴⁾ | 1,4/0,4 ⁴⁾ |
| | FB [e] | speicherindirekter FC-Aufruf | 2 | 2,4+/0,4 ⁴⁾ | 1,4+/0,4 ⁴⁾ | 1,12+/0,32 ⁴⁾ | 1,4+/0,4 ⁴⁾ |
| | FC [e] | FB/FC-Aufruf über Parameter | 2 | 2,4+/0,4 ⁴⁾ | 1,4+/0,4 ⁴⁾ | 1,12+/0,32 ⁴⁾ | 1,4+/0,4 ⁴⁾ |
| | Parameter | | 2 | 2,4+/0,4 ⁴⁾ | 1,4+/0,4 ⁴⁾ | 1,12+/0,32 ⁴⁾ | 1,4+/0,4 ⁴⁾ |

| | | | | | | | | | |
|---|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: UC, CC ²⁾ | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | – | – | – | – | – | – | – | – | – |
| Operation beeinflusst: | – | – | – | – | 0 | 0 | 1 | – | 0 |

- + plus Zeit zum Laden der Adresse des Operanden (siehe S. 19)
- 1) bei direkter Adressierung des Operanden Baustein-Nr. 0 bis 255
- 2) Abhängig von VKE, setzt VKE = 1
- 4) Wenn Aufruf nicht ausgeführt wird

Baustein-Aufrufoperationen, Fortsetzung

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μ s direkte Adressierung | | | |
|-----------|-----------|----------------------------------|--------------------|---|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| AUF | DB q | Aufschlagen eines Datenbausteins | 1 ¹⁾ /2 | 0,6/0,7 | 0,3 | 0,24 | 0,3 |
| | DI q | Instanz-Datenbausteins | | 0,7 | 0,3 | 0,24 | 0,3 |
| | DB [e] | Datenbausteins, speicherindirekt | | 0,7+ | 0,3+ | 0,24+ | 0,3+ |
| | DI [e] | Instanz-DB, speicherindirekt | | 0,7+ | 0,3+ | 0,24+ | 0,3+ |
| | Parameter | Datenbausteins über Parameter | | 0,7+ | 0,3+ | 0,24+ | 0,3+ |

| | | | | | | | | | |
|----------------------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: AUF | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | - | - | - | - | - | - | - | - | - |
| Operation beeinflusst: | - | - | - | - | - | - | - | - | - |

+ plus Zeit zum Laden der Adresse des Operanden (siehe S. 19)

1) bei direkter Adressierung des Operanden DB Baustein-Nr. 0 bis 255

Baustein-Endeoperationen

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μs | | | |
|-----------|---------|-------------------------|-----------------|----------------------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| BE | | Beende Baustein | 1 | 2,8 | 2,0 | 1,60 | 2,0 |
| BEA | | Beende Baustein absolut | 1 | 2,8 | 2,0 | 1,60 | 2,0 |

| | | | | | | | | | |
|--------------------------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: BE, BEA | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | – | – | – | – | – | – | – | – | – |
| Operation beeinflusst: | – | – | – | – | 0 | 0 | 1 | – | 0 |

| | | | | | | | |
|-----|--|-------------------------------------|--|--------------------------|--------------------------|----------------------------|--------------------------|
| BEB | | Beende Baustein bedingt bei VKE="1" | | 3,0 0,4 ¹⁾ | 2,2 0,4 ¹⁾ | 1,76 0,32 ¹⁾ | 2,2 0,4 ¹⁾ |
|-----|--|-------------------------------------|--|--------------------------|--------------------------|----------------------------|--------------------------|

| | | | | | | | | | |
|----------------------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: BEB | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | – | – | – | – | – | – | – | ja | – |
| Operation beeinflusst: | – | – | – | – | ja | 0 | 1 | 1 | 0 |

1) Wenn Sprung nicht ausgeführt wird.

Tausche Datenbausteine

Tauschen der beiden aktuellen Datenbausteine. Der aktuelle Datenbaustein wird zum aktuellen Instanz-Datenbaustein und umgekehrt. Das Statuswort wird nicht beeinflusst.

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μs | | | |
|-----------|---------|------------------------|-----------------|----------------------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| TDB | | Tausche Datenbausteine | 1 | 0,2 | 0,2 | 0,16 | |

Sprungoperationen

Sprung, abhängig von der Bedingung.

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μs | | | |
|-----------|---------|-------------------|--------------------|----------------------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| SPA | MARKE | Springe unbedingt | 1 ¹⁾ /2 | 0,5/0,6 | 0,5 | 0,4 | 0,5 |

| | | | | | | | | | |
|----------------------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: SPA | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | – | – | – | – | – | – | – | – | – |
| Operation beeinflusst: | – | – | – | – | – | – | – | – | – |

| | | | | | | | |
|------|-------|---------------------|--------------------|-----------------------|-----------------------|------------------------|-----------------------|
| SPB | MARKE | Springe bei VKE="1" | 1 ¹⁾ /2 | 0,5/0,6 ²⁾ | 0,5/0,2 ²⁾ | 0,4/0,16 ²⁾ | 0,5/0,2 ²⁾ |
| SPBN | MARKE | Springe bei VKE="0" | 2 | 0,6/0,2 ²⁾ | 0,5/0,2 ²⁾ | 0,4/0,16 ²⁾ | 0,5/0,2 ²⁾ |

| | | | | | | | | | |
|----------------------------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: SPB, SPBN | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | – | – | – | – | – | – | – | ja | – |
| Operation beeinflusst: | – | – | – | – | – | 0 | 1 | 1 | 0 |

1) 1 Wort lang bei Sprungweiten von -128 ... +127

2) Wenn Sprung nicht ausgeführt wird

Sprungoperationen, Fortsetzung

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μs | | | |
|-----------|---------|--|-----------------|----------------------------------|-----------------------|------------------------|-----------------------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| SPBB | MARKE | Springe bei VKE="1" Retten des VKE in das BIE-Bit | 2 | 0,6/0,2 ²⁾ | 0,5/0,2 ²⁾ | 0,4/0,16 ²⁾ | 0,5/0,2 ²⁾ |
| SPBNB | MARKE | Springe bei VKE="0" Retten des VKE in das BIE-Bit | 2 | 0,6/0,2 ²⁾ | 0,5/0,2 ²⁾ | 0,4/0,16 ²⁾ | 0,5/0,2 ²⁾ |

| | | | | | | | | | |
|------------------------------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: SPBB, SPBNB | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | – | – | – | – | – | – | – | ja | – |
| Operation beeinflusst: | ja | – | – | – | – | 0 | 1 | 1 | 0 |

| | | | | | | | |
|-------|-------|---------------------|---|-----------------------|-----------------------|------------------------|-----------------------|
| SPBI | MARKE | Springe bei BIE="1" | 2 | 0,6/0,2 ²⁾ | 0,5/0,2 ²⁾ | 0,4/0,16 ²⁾ | 0,5/0,2 ²⁾ |
| SPBIN | MARKE | Springe bei BIE="0" | 2 | 0,6/0,2 ²⁾ | 0,5/0,2 ²⁾ | 0,4/0,16 ²⁾ | 0,5/0,2 ²⁾ |

| | | | | | | | | | |
|------------------------------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: SPBI, SPBIN | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | ja | – | – | – | – | – | – | – | – |
| Operation beeinflusst: | – | – | – | – | – | 0 | 1 | – | 0 |

2) Wenn Sprung nicht ausgeführt wird

Sprungoperationen, Fortsetzung

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μ s | | | |
|-----------|---------|--|--------------------|----------------------------|-----------------------|------------------------|-----------------------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| SPO | MARKE | Springe bei Überlauf speichernd (OV="1") | 1 ¹⁾ /2 | 0,5/0,6/0,2 ²⁾ | 0,5/0,2 ²⁾ | 0,4/0,16 ²⁾ | 0,5/0,2 ²⁾ |

| | | | | | | | | | |
|----------------------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: SPO | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | – | – | – | ja | – | – | – | – | – |
| Operation beeinflusst: | – | – | – | – | – | – | – | – | – |

| | | | | | | | |
|-----|-------|--|---|-----------------------|-----------------------|------------------------|-----------------------|
| SPS | MARKE | Springe bei Überlauf speichernd (OS="1") | 2 | 0,6/0,2 ²⁾ | 0,5/0,2 ²⁾ | 0,4/0,16 ²⁾ | 0,5/0,2 ²⁾ |
|-----|-------|--|---|-----------------------|-----------------------|------------------------|-----------------------|

| | | | | | | | | | |
|----------------------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: SPS | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | – | – | – | – | ja | – | – | – | – |
| Operation beeinflusst: | – | – | – | – | 0 | – | – | – | – |

1) 1 Wort lang bei Sprungweiten von -128 ... +127

2) Wenn Sprung nicht ausgeführt wird

Sprungoperationen, Fortsetzung

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μs | | | |
|-----------|---------|--|--------------------|----------------------------------|-----------------------|------------------------|-----------------------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| SPU | MARKE | Sprünge bei "Unzulässiger Arithmetikoperation" (A1=1 und A0=1) | 2 | 0,6/0,2 ²⁾ | 0,5/0,2 ²⁾ | 0,4/0,16 ²⁾ | 0,5/0,2 ²⁾ |
| SPZ | MARKE | Sprünge bei Ergebnis=0 (A1=0 und A0=0) | 1 ¹⁾ /2 | 0,5/0,6/0,2 ²⁾ | 0,5/0,2 ²⁾ | 0,4/0,16 ²⁾ | 0,5/0,2 ²⁾ |
| SPP | MARKE | Sprünge bei Ergebnis>0 (A1=1 und A0=0) | 1 ¹⁾ /2 | 0,5/0,6/0,2 ²⁾ | 0,5/0,2 ²⁾ | 0,4/0,16 ²⁾ | 0,5/0,2 ²⁾ |
| SPM | MARKE | Sprünge bei Ergebnis<0 (A1=0 und A0=1) | 1 ¹⁾ /2 | 0,5/0,6/0,2 ²⁾ | 0,5/0,2 ²⁾ | 0,4/0,16 ²⁾ | 0,5/0,2 ²⁾ |
| SPN | MARKE | Sprünge bei Ergebnis \neq 0 (A1=1 und A0=0) oder (A1=0 und A0=1) | 1 ¹⁾ /2 | 0,5/0,6/0,2 ²⁾ | 0,5/0,2 ²⁾ | 0,4/0,16 ²⁾ | 0,5/0,2 ²⁾ |

| | | | | | | | | | |
|--|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: SPU, SPZ, SPP, SPM, SPN | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | – | ja | ja | – | – | – | – | – | – |
| Operation beeinflusst: | – | – | – | – | – | – | – | – | – |

1) 1 Wort lang bei Sprungweiten von -128 ... +127

2) Wenn Sprung nicht ausgeführt wird

Sprungoperationen, Fortsetzung

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μs | | | |
|-----------|---------|--|-----------------|----------------------------------|-----------------------|------------------------|-----------------------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| SPMZ | MARKE | Springe bei Ergebnis ≤ 0 (A1=0 und A0=1) oder (A1=0 und A0=0) | 2 | 0,6/0,2 ²⁾ | 0,5/0,2 ²⁾ | 0,4/0,16 ²⁾ | 0,5/0,2 ²⁾ |
| SPPZ | MARKE | Springe bei Ergebnis ≥ 0 (A1=1 und A0=0) oder (A1=0 und A0=0) | 2 | 0,6/0,2 ²⁾ | 0,5/0,2 ²⁾ | 0,4/0,16 ²⁾ | 0,5/0,2 ²⁾ |

| | | | | | | | | | |
|-----------------------------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: SPMZ, SPPZ | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | – | ja | ja | – | – | – | – | – | – |
| Operation beeinflusst: | – | – | – | – | – | – | – | – | – |

2) Wenn Sprung nicht ausgeführt wird

Sprungoperationen, Fortsetzung

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μ s | | | |
|-----------|---------|---|-----------------|----------------------------|-----------------------|------------------------|-----------------------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| SPL | MARKE | Sprungverteiler Der Operation folgt eine Liste von Sprungoperationen. Der Operand ist eine Sprungmarke auf die der Liste folgenden Operation. AKKU1-LL enthält die Nr. der Sprungoperation (max. 254), die ausgeführt werden soll, wobei die erste Sprungoperationsnummer 0 ist. | 2 | 0,8 | 0,7 | 0,56 | 0,7 |
| LOOP | MARKE | Dekrementiere AKKU1-L und springe bei $AKKU1-L \neq 0$ (Schleifenprogrammierung) | 2 | 0,6/0,2 ¹⁾ | 0,5/0,2 ¹⁾ | 0,4/0,08 ¹⁾ | 0,5/0,2 ¹⁾ |

| | | | | | | | | | |
|----------------------------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: SPL, LOOP | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | – | – | – | – | – | – | – | – | – |
| Operation beeinflusst: | – | – | – | – | – | – | – | – | – |

1) Wenn Befehl nicht ausgeführt wird

Operationen für das Master Control Relay (MCR)

MCR=1→MCR ist deaktiviert

MCR=0→MCR ist aktiviert; "T"- und "="-Operationen schreiben bei

VKE = "0" Nullen auf die entsprechenden Operanden; "S"- und "R"-Operationen lassen den Speicherinhalt unverändert.

| Opera-tion | Operand | Bedeutung | Län-ge in Wor-ten | Ausführungszeit in µs | | | |
|------------|---------|---|-------------------|-----------------------|---------|---------|---------|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 |
| MCR(| | Öffnen einer MCR-Zone. Retten des VKE auf den MCR-Stack. | 1 | 0,1 | 0,1 | 0,08 | 0,1 |

| | | | | | | | | | |
|-----------------------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für: MCR(| BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | - | - | - | - | - | - | - | ja | - |
| Operation beeinflusst: | - | - | - | - | - | 0 | 1 | - | 0 |

| | | | | | | | |
|------|--|--|---|-----|-----|------|-----|
|)MCR | | Schließen einer MCR-Zone. Entfernen eines Eintrags vom MCR-Stack. | 1 | 0,1 | 0,1 | 0,08 | 0,1 |
|------|--|--|---|-----|-----|------|-----|

| | | | | | | | | | |
|-----------------------------|-----|----|----|----|----|----|-----|-----|-----|
| Statuswort für:)MCR | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | - | - | - | - | - | - | - | - | - |
| Operation beeinflusst: | - | - | - | - | - | 0 | 1 | - | 0 |

Operationen für das Master Control Relay (MCR), Fortsetzung

| Operation | Operand | Bedeutung | Länge in Worten | Ausführungszeit in μs | | | | | | | |
|-----------------------------------|---------|-----------------|-----------------|----------------------------------|---------|---------|---------|----|-----|-----|-----|
| | | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 | | | | |
| MCRA | | Aktiviere MCR | 1 | 0,1 | 0,1 | 0,08 | 0,1 | | | | |
| MCRD | | Deaktiviere MCR | 1 | 0,1 | 0,1 | 0,08 | 0,1 | | | | |
| Statuswort für: MCRA, MCRD | | | BIE | A1 | A0 | OV | OS | OR | STA | VKE | /ER |
| Operation wertet aus: | | | – | – | – | – | – | – | – | – | – |
| Operation beeinflusst: | | | – | – | – | – | – | – | – | – | – |

Organisationsbausteine (OB)

Ein Anwenderprogramm für eine S7-400 besteht aus Bausteinen, die die Anweisungen, Parameter und Daten für die jeweilige CPU enthalten. Die einzelnen CPUs der S7-400 unterscheiden sich in der Menge der Bausteine, die Sie für die jeweilige CPU anlegen können

bzw. die vom Betriebssystem der CPU bereitgestellt werden. Eine ausführliche Beschreibung der OBs und deren Anwendung finden Sie im *STEP 7 Programmierhandbuch* bzw. im Handbuch *Programmieren mit STEP 7 V 5.0*.

| Organisationsbausteine | CPU 412 | CPU 414 | CPU 414-4H | CPU 416 | CPU 417 | CPU 417-4H | Startereignisse (Hexadezimalwert) |
|------------------------|---------|---------|------------|---------|---------|------------|-----------------------------------|
| Freier Zyklus: | | | | | | | |
| OB 1 | x | x | x | x | x | x | 1101, 1102, 1103, 1104, 1105 |
| Uhrzeitalarme: | | | | | | | |
| OB 10 | x | x | x | x | x | x | 1111 |
| OB 11 | x | x | x | x | x | x | 1112 |
| OB 12 | | x | x | x | x | x | 1113 |
| OB 13 | | x | x | x | x | x | 1114 |
| OB 14 | | | | x | x | x | 1115 |
| OB 15 | | | | x | x | x | 1116 |
| OB 16 | | | | x | x | x | 1117 |
| OB 17 | | | | x | x | x | 1118 |

Organisationsbausteine (OB), Fortsetzung

| Organisationsbausteine | CPU 412 | CPU 414 | CPU 414-4H | CPU 416 | CPU 417 | CPU 417-4H | Startereignisse (Hexadezimalwert) |
|------------------------|---------|---------|------------|---------|---------|------------|-----------------------------------|
| Verzögerungsalarme: | | | | | | | |
| OB 20 | x | x | x | x | x | x | 1121 |
| OB 21 | x | x | x | x | x | x | 1122 |
| OB 22 | | x | x | x | x | x | 1123 |
| OB 23 | | x | x | x | x | x | 1124 |
| Weckalarme: | | | | | | | |
| OB 30 | | | | x | x | x | 1131 |
| OB 31 | | | | x | x | x | 1132 |
| OB 32 | x | x | x | x | x | x | 1133 |
| OB 33 | | x | x | x | x | x | 1134 |
| OB 34 | | x | x | x | x | x | 1135 |
| OB 35 | x | x | x | x | x | x | 1136 |
| OB 36 | | | | x | x | x | 1137 |
| OB 37 | | | | x | x | x | 1138 |
| OB 38 | | | | x | x | x | 1139 |

Organisationsbausteine (OB), Fortsetzung

| Organisationsbausteine | CPU 412 | CPU 414 | CPU 414-4H | CPU 416 | CPU 417 | CPU 417-4H | Startereignisse (Hexadezimalwert) |
|------------------------|---------|---------|------------|---------|---------|------------|--------------------------------------|
| Prozeßalarme: | | | | | | | |
| OB 40 | x | x | x | x | x | x | 1141, 1142, 1143, 1144, 1145 |
| OB 41 | x | x | x | x | x | x | 1141, 1142, 1143, 1144, 1145 |
| OB 42 | | x | x | x | x | x | 1141, 1142, 1143, 1144, 1145 |
| OB 43 | | x | x | x | x | x | 1141, 1142, 1143, 1144, 1145 |
| OB 44 | | | | x | x | x | 1141, 1142, 1143, 1144, 1145 |
| OB 45 | | | | x | x | x | 1141, 1142, 1143, 1144, 1145 |
| OB 46 | | | | x | x | x | 1141, 1142, 1143, 1144, 1145 |
| OB 47 | | | | x | x | x | 1141, 1142, 1143, 1144, 1145 |
| Alarm-OBs für DPV1: | | | | | | | |
| OB 55 | x | x | x | x | x | x | 1155 |
| OB 56 | x | x | x | x | x | x | 1156 |
| OB 57 | x | x | x | x | x | x | 1157 |

Organisationsbausteine (OB), Fortsetzung

| Organisationsbausteine | CPU 412 | CPU 414 | CPU 414-4H | CPU 416 | CPU 417 | CPU 417-4H | Startereignisse (Hexadezimalwert) |
|--------------------------|---------|---------|------------|---------|---------|------------|--|
| Multicomputingalarm: | | | | | | | |
| OB 60 | x | x | | x | x | | 1161, 1162 |
| Taktsynchronalarm: | | | | | | | |
| OB 61 | x | x | x | x | x | x | 1164 |
| OB 62 | x | x | x | x | x | x | 1165 |
| OB 63 | x | x | x | x | x | x | 1166 |
| OB 64 | x | x | x | x | x | x | 1167 |
| Redundanzfehleralarme: | | | | | | | |
| OB 70 | | | x | | | x | 73A2, 73A3, 72A3 |
| OB 72 | | | x | | | x | 7301, 7302, 7303, 7320, 7321, 7322, 7323, 7331, 7333, 7334, 7335, 7340, 7341, 7342, 7343, 7344, 7950, 7951, 7952, 7852, 7953, 7954, 7955, 7855, 7956, 73C1, 73C2 |
| Asynchrone Fehleralarme: | | | | | | | |
| OB 80 | x | x | x | x | x | x | 3501, 3502, 3505, 3506, 3507, 3508, 3509, 350A |
| OB 81 | x | x | x | x | x | x | 3821, 3822, 3823, 3825, 3826, 3827, 3831, 3832, 3833, 3921, 3922, 3923, 3925, 3926, 3927, 3931, 3932, 3933 |
| OB 82 | x | x | x | x | x | x | 3842, 3942 |
| OB 83 | x | x | x | x | x | x | 3267, 3367, 3861, 3863, 3864, 3865, 3961, 3968 |

1) nicht bei CPU 412-1

| Organisationsbausteine | CPU 412 | CPU 414 | CPU 414-4H | CPU 416 | CPU 417 | CPU 417-4H | Startereignisse (Hexadezimalwert) |
|-------------------------|---------|---------|------------|---------|---------|------------|---|
| OB 85 | x | x | x | x | x | x | 35A1, 35A2, 35A3, 38B3, 38B4, 39B1, 39B2, 39B3, 39B4 |
| OB 86 | x | x | x | x | x | x | 38C1, 38C2, 39C1, 38C6, 38C7, 38C8 38C4 ¹ , 38C5 ¹ , 39C3 ¹ , 39C4 ¹ , 39C5 ¹) |
| OB 87 | x | x | x | x | x | x | 35D2, 35D3, 35D4, 35D5, 35E1, 35E2, 35E3, 35E4, 35E5, 35E6 |
| OB 88 | x | x | x | x | x | x | 3571, 3572, 3573, 3574, 3575, 3576, 3578, 357A |
| Hintergrund: | | | | | | | |
| OB 90 | x | x | | x | x | | 1191, 1192, 1193, 1195 |
| Neustart (Warmstart): | | | | | | | |
| OB 100 | x | x | x | x | x | x | 1381, 1382, 138A, 138B |
| Wiederanlauf: | | | | | | | |
| OB 101 | x | x | | x | x | | 1383, 1384 |
| Kaltstart: | | | | | | | |
| OB 102 | x | x | x | x | x | x | 1385, 1386, 1387, 1388 |
| Synchrone Fehleralarme: | | | | | | | |
| OB 121 | x | x | x | x | x | x | 2521, 2522, 2523, 2524, 2525, 2526, 2527, 2528, 2529, 2530, 2531, 2532, 2533, 2534, 2535, 253A, 253C, 253D, 253E, 253F |
| OB 122 | x | x | x | x | x | x | 2942, 2943, 2944, 2945 |

Funktionsbausteine (FB)

Die nachfolgende Tabelle listet Anzahl, Nummer und maximale Größe der Funktionsbausteine auf, die Sie in den einzelnen CPUs der S7-400 anlegen können.

| Funktions- bausteine | CPU 412-1 | CPU 412-2 | CPU 414 | CPU 416 | CPU 417 |
|---|------------------|------------------|----------------|----------------|----------------|
| Anzahl | 256 | 256 | 2048 | 2048 | 6144 |
| Zulässige Nummer | 0 bis 255 | 0 bis 255 | 0 bis 2047 | 0 bis 2047 | 0 bis 6143 |
| Maximale Größe (ab- laufrelevanter Code) | 48 KByte | 64 kByte | 64 KByte | 64 KByte | 64 KByte |

Funktionen (FC) und Datenbausteine

Die nachfolgende Tabellen listen Anzahl, Nummer und maximale Größe der Funktionen und Datenbausteine auf, die Sie in den einzelnen CPUs der S7-400 anlegen können.

| Funktionen | CPU 412-1 | CPU 412-2 | CPU 414 | CPU 416 | CPU 417 |
|--|------------------|------------------|----------------|----------------|----------------|
| Anzahl | 256 | 256 | 2048 | 2048 | 6144 |
| Zulässige Nummer | 0 bis 255 | 0 bis 255 | 0 bis 2047 | 0 bis 2047 | 0 bis 6143 |
| Maximale Größe (ablaufrelevanter Code) | 48 KByte | 64 kByte | 64 KByte | 64 KByte | 64 KByte |

| Datenbausteine | CPU 412-1 | CPU 412-2 | CPU 414 | CPU 416 | CPU 417 |
|------------------------------------|------------------|------------------|----------------|----------------|----------------|
| Anzahl | 511 | 511 | 4095 | 4095 | 8191 |
| Zulässige Nummer | 1 bis 511 | 1 bis 511 | 1 bis 4095 | 1 bis 4095 | 1 bis 8191 |
| Maximale Größe (Anzahl Datenbytes) | 48 KByte | 64 kByte | 64 KByte | 64 KByte | 64 KByte |

Systemfunktionen

Nachfolgende Tabellen zeigen die Systemfunktionen, die vom Betriebssystem der S7-400 CPUs bereitgestellt werden, und die Ausführungszeiten auf der jeweiligen CPU. (X: Funktion vorhanden, Ausführungszeiten lagen bei Drucklegung noch nicht vor).

| SFC-Nr. | SFC-Name | Bedeutung | Ausführungszeit in μs | | | | | |
|---------|----------|---|----------------------------------|---------|---------|---------|---------------------------|--------------------------------|
| | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 | CPUs 414-4H 417-4H (solo) | CPUs 414-4H 417-4H (redundant) |
| 0 | SET_CLK | Uhrzeit stellen | 340 | 249 | 215 | 249 | 289 | 288 |
| 1 | READ_CLK | Uhrzeit lesen | 40 | 29 | 23 | 29 | 29 | 54 |
| 2 | SET_RTM | Betriebsstundenzähler setzen | 35 | 26 | 20 | 26 | 25 | 26 |
| 3 | CTRL_RTM | Betriebsstundenzähler starten/stoppen | 30 | 23 | 18 | 23 | 22 | 22 |
| 4 | READ_RTM | Betriebsstundenzähler auslesen | 41 | 30 | 23 | 30 | 29 | 58 |
| 5 | GADR_LGC | Logische Adresse eines Kanals ermitteln Rack-0 | 55 | 39 | 31 | 39 | 38 | 38 |
| | | interne DP | 66 | 46 | 36 | 46 | 46 | 46 |
| 6 | RD_SINFO | Startinformation des akt. OB auslesen | 54 | 38 | 30 | 38 | 39 | 39 |
| 7 | DP_PRAL | Einen Prozeßalarm beim DP-Master auslösen Erstaufruf | 294 | 208 | 166 | 208 | -- | -- |
| | | Zwischenaufruf | 43 | 30 | 24 | 30 | -- | -- |
| | | Letztaufruf | 46 | 32 | 25 | 32 | -- | -- |

| SFC-Nr. | SFC-Name | Bedeutung | Ausführungszeit in μs | | | | | |
|---------|----------|--|----------------------------------|---------|---------|---------|---------------------------|--------------------------------|
| | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 | CPUs 414-4H 417-4H (solo) | CPUs 414-4H 417-4H (redundant) |
| 9 | EN_MSG | Bausteinbezogene, symbolbezogene und Leittechniksammelmeldungen freigeben Erstaufruf, REQ = 1 | 176 | 122 | 97 | 122 | 128 | 232 |
| | | Letztaufruf | 61 | 44 | 34 | 44 | 39 | 62 |
| 10 | DIS_MSG | Bausteinbezogene, symbolbezogene und Leittechniksammelmeldungen sperren Erstaufruf, REQ = 1 | 176 | 122 | 97 | 122 | 128 | 232 |
| | | Letztaufruf | 61 | 44 | 34 | 44 | 39 | 63 |

| SFC-Nr. | SFC-Name | Bedeutung | Ausführungszeit in μs | | | | | |
|---------|----------|---|----------------------------------|--------------|--------------|--------------|---------------------------|--------------------------------|
| | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 | CPUs 414-4H 417-4H (solo) | CPUs 414-4H 417-4H (redundant) |
| 11 | DPSYC_FR | Gruppen von DP-Slaves synchronisieren Erstaufruf, integrierte DP-Schnittstelle, REQ = 1 | 170 | 110 | 90 | 110 | -- | -- |
| | | Zwischenaufruf, integrierte DP-Schnittstelle, BUSY = 1 ¹⁾ | $51 + n * 4$ | $36 + n * 3$ | $28 + n * 2$ | $36 + n * 3$ | -- | -- |
| | | Letztaufruf, integrierte DP-Schnittstelle, BUSY = 0 ¹⁾ | $51 + n * 4$ | $36 + n * 3$ | $28 + n * 2$ | $36 + n * 3$ | -- | -- |
| 11 | DPSYC_FR | Erstaufruf, externe DP-Schnittstelle, REQ = 1 | 94 | 71 | 60 | 71 | -- | -- |
| | | Zwischenaufruf, externe DP-Schnittstelle, BUSY = 1 ¹⁾ | $64 + n * 4$ | $50 + n * 3$ | $39 + n * 2$ | $50 + n * 3$ | -- | -- |
| | | Letztaufruf, externe DP-Schnittstelle, BUSY = 0 ¹⁾ | $64 + n * 4$ | $50 + n * 3$ | $39 + n * 2$ | $50 + n * 3$ | -- | -- |

¹⁾ n = Zahl der aktiven Aufträge mit gleicher logischer Adresse

| SFC-Nr. | SFC-Name | Bedeutung | Ausführungszeit in μ s | | | | | |
|---------|----------|---|----------------------------|---------|---------|---------|---------------------------|--------------------------------|
| | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 | CPUs 414-4H 417-4H (solo) | CPUs 414-4H 417-4H (redundant) |
| 12 | D_ACT_DP | Deaktivieren und Aktivieren von DP-Slaves über integrierte DP-Schnittstelle, MODE = 0 | 117 | 76 | 61 | 76 | -- | -- |
| 12 | D_ACT_DP | Deaktivieren und Aktivieren von DP-Slaves über integrierte DP-Schnittstelle, MODE = 1 Erstaufruf | 269 | 179 | 142 | 179 | -- | -- |
| | | Zwischenaufruf | 114 | 73 | 59 | 73 | -- | -- |
| | | Letztaufruf | 231 | 167 | 121 | 167 | -- | -- |
| 12 | D_ACT_DP | Deaktivieren und Aktivieren von DP-Slaves über integrierte DP-Schnittstelle, MODE = 2 Erstaufruf | 378 | 268 | 202 | 268 | -- | -- |
| | | Zwischenaufruf | 113 | 72 | 58 | 72 | -- | -- |
| | | Letztaufruf | 119 | 76 | 62 | 76 | -- | -- |
| 12 | D_ACT_DP | Deaktivieren und Aktivieren von DP-Slaves über externe DP-Schnittstelle, MODE = 0 | X | X | X | X | -- | -- |

| SFC-Nr. | SFC-Name | Bedeutung | Ausführungszeit in μs | | | | | |
|---------|----------|---|----------------------------------|---------|---------|---------|---------------------------------|--------------------------------------|
| | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 | CPUs 414-4H 417-4H (solo) | CPUs 414-4H 417-4H (redundant) |
| 12 | D_ACT_DP | Deaktivieren und Aktivieren von DP-Slaves über externe DP-Schnittstelle, MODE = 1 Erstaufruf | X | X | X | X | -- | -- |
| | | Zwischenaufruf | X | X | X | X | -- | -- |
| | | Letztaufruf | X | X | X | X | -- | -- |
| 12 | D_ACT_DP | Deaktivieren und Aktivieren von DP-Slaves über externe DP-Schnittstelle, MODE = 2 Erstaufruf | X | X | X | X | -- | -- |
| | | Zwischenaufruf | X | X | X | X | -- | -- |
| | | Letztaufruf | X | X | X | X | -- | -- |
| 13 | DPNRM_DG | Slavediagnosedaten lesen Erstaufruf | 300 | 200 | 165 | 200 | 210 | 290 |
| | | Zwischenaufruf | -- | -- | -- | -- | 79 | 79 |
| | | Letztaufruf (28 Byte) | 180 | 125 | 100 | 125 | 101 | 101 |

| SFC-Nr. | SFC-Name | Bedeutung | Ausführungszeit in μ s | | | | | |
|---------|----------|---|--------------------------------------|--|--|--|--|--|
| | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 | CPUs 414-4H 417-4H (solo) | CPUs 414-4H 417-4H (redundant) |
| 14 | DPRD_DAT | Konsistente Nutzdaten lesen über integrierte DP-Schnittstelle 3 byte | 83 | 56 | 45 | 56 | 70 | 96 |
| | | über integrierte DP-Schnittstelle 32 byte | 94 | 67 | 54 | 67 | 88 | 122 |
| | | über externe DP-Schnittstelle 3 byte | 86 | 62 | 50 | 62 | 76 | 99 |
| | | über externe DP-Schnittstelle 32 byte | 181 | 156 | 137 | 156 | 152 | 209 |
| 15 | DPWR_DAT | Konsistente Nutzdaten schreiben über integrierte DP-Schnittstelle 3 byte | 84 ¹⁾ /91 ²⁾ | 57 ¹⁾ / 61 ²⁾ | 45 ¹⁾ / 49 ²⁾ | 57 ¹⁾ / 61 ²⁾ | 72 ¹⁾ / 76 ²⁾ | 94 ¹⁾ / 98 ²⁾ |
| | | über integrierte DP-Schnittstelle 32 byte | 96 ¹⁾ /127 ²⁾ | 67 ¹⁾ / 97 ²⁾ | 53 ¹⁾ / 78 ²⁾ | 67 ¹⁾ / 97 ²⁾ | 88 ¹⁾ / 119 ²⁾ | 110 ¹⁾ / 142 ²⁾ |
| | | über externe DP-Schnittstelle 3 byte | 88 ¹⁾ /94 ²⁾ | 62 ¹⁾ / 67 ²⁾ | 50 ¹⁾ / 54 ²⁾ | 62 ¹⁾ / 67 ²⁾ | 77 ¹⁾ /83 ²⁾ | 100 ¹⁾ / 105 ²⁾ |
| | | über externe DP-Schnittstelle 32 byte | 178 ¹⁾ /209 ²⁾ | 150 ¹⁾ / 181 ²⁾ | 130 ¹⁾ / 154 ²⁾ | 150 ¹⁾ / 181 ²⁾ | 171 ¹⁾ / 201 ²⁾ | 193 ¹⁾ / 224 ²⁾ |
| 17 | ALARM_SQ | Quittierbare bausteinbezogene Meldungen erzeugen. Erstaufruf, SIG = 0 → 1 | 440 | 305 | 240 | 305 | 266 | 358 |
| | | Leeraufruf | 130 | 90 | 72 | 90 | 90 | 156 |
| 18 | ALARM_S | nicht quittierbare bausteinbezogene Meldungen erzeugen. Erstaufruf, SIG = 0 → 1 | 460 | 310 | 250 | 310 | 275 | 365 |
| | | Leeraufwurf | 140 | 90 | 75 | 90 | 97 | 163 |

1) ohne Datenübertragung ins Prozeßabbild

2) mit Datenübertragung ins Prozeßabbild

| SFC-Nr. | SFC-Name | Bedeutung | Ausführungszeit in μs | | | | | |
|---------|----------|---|----------------------------------|-------------------|-------------------|-------------------|---------------------------|--------------------------------|
| | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 | CPUs 414-4H 417-4H (solo) | CPUs 414-4H 417-4H (redundant) |
| 19 | ALARM_SC | Quittierzustand der letzten ALARM_SQ-gekommen-Meldung | 85 | 60 | 46 | 60 | 56 | 82 |
| 20 | BLKMOV | Variable kopieren innerhalb des Arbeitsspeichers (n = Anzahl der zu kopierenden Byte) | 60 + n * 0,3 | 41 + n * 0,13 | 32 + n * 0,23 | 41 + n * 0,13 | 42 + n * 0,17 | 42 + n * 0,17 |
| | | Quelle = Ladespeicher | 1400 + n * 1,0 | 1160 + n * 0,7 | 1100 + n * 0,7 | 1160 + n * 0,7 | 1124 + n * 1,0 | 2065 + n * 1,98 |
| 21 | FILL | Feld vorbesetzen innerhalb des Arbeitsspeichers (n = Länge der Zielvariablen in Byte) | 60 + n * 0,15 | 44 + n * 0,13 | 34 + n * 0,1 | 44 + n * 0,13 | 45 + n * 0,12 | 45 + n * 0,12 |
| 22 | CREAT_DB | Datenbaustein erzeugen n = DB-Längen [Byte] | 142 | 94 | 72 | 94 | 155 + n * 0,1 | 424 + n * 0,1 |
| | | letzte freie DB-Nr. aus Feld von 100 DBs belegen | 606 | 400 | 320 | 400 | 2877 | 13601 |
| 23 | DEL_DB | Datenbaustein löschen | 122 | 81 | 64 | 81 | 179 | 625 |
| 24 | TEST_DB | Datenbaustein testen | 47 | 32 | 25 | 32 | 68 | 248 |
| 25 | COMPRESS | Anwenderspeicher komprimieren Erstaufruf (Anstoß) | 112 | 78 | 63 | 78 | 93 | 173 |
| | | Folgeaufruf | 32 | 23 | 18 | 23 | 22 | 22 |

| SFC-Nr. | SFC-Name | Bedeutung | Ausführungszeit in μ s | | | | | |
|---------|----------|--|----------------------------|---------|---------|---------|---------------------------------|--------------------------------------|
| | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 | CPUs 414-4H 417-4H (solo) | CPUs 414-4H 417-4H (redundant) |
| 26 | UPDAT_PI | Prozeßabbild der Eingänge aktualisieren (Laufzeitangabe für 1 DI 32 im ZG) | 45 | 35 | 29 | 35 | 67 | 103 |
| | | AI 8*13Bit | 70 | 59 | 51 | 59 | 155 | 192 |
| 27 | UPDAT_PO | Ausgänge aktualisieren (Laufzeitangabe für 1 DO 32 im ZG) | 45 | 35 | 29 | 35 | 54 | 81 |
| | | AO 8 * 13 Bit | 66 | 55 | 48 | 55 | 122 | 149 |
| 28 | SET_TINT | Uhrzeitalarm stellen | 108 | 75 | 60 | 75 | 74 | 98 |
| 29 | CAN_TINT | Uhrzeitalarm stornieren | 40 | 29 | 22 | 29 | 34 | 34 |
| 30 | ACT_TINT | Uhrzeitalarm aktivieren | 73 | 53 | 41 | 53 | 51 | 75 |
| 31 | QRY_TINT | Uhrzeitalarm abfragen | 44 | 34 | 27 | 34 | 33 | 34 |
| 32 | SRT_DINT | Verzögerungsalarm starten | 65 | 46 | 36 | 46 | 44 | 44 |
| 33 | CAN_DINT | Verzögerungsalarm stornieren | 41 | 30 | 23 | 30 | 36 | 36 |
| 34 | QRY_DINT | Verzögerungsalarm abfragen | 43 | 33 | 26 | 33 | 32 | 32 |
| 35 | MP_ALM | Multicomputingalarm auslösen | 240 | 171 | 138 | 171 | -- | -- |
| 36 | MSK_FLT | Synchronfehlerereignisse maskieren | 30 | 22 | 17 | 22 | 21 | 21 |
| 37 | DMSK_FLT | Synchronfehlerereignisse demaskieren | 31 | 23 | 18 | 23 | 22 | 23 |
| 38 | READ_ERR | Ereignisstatusregister lesen | 32 | 23 | 18 | 23 | 23 | 23 |

| SFC-Nr. | SFC-Name | Bedeutung | Ausführungszeit in μ s | | | | | |
|---------|----------|--|----------------------------|----------|----------|----------|---------------------------|--------------------------------|
| | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 | CPUs 414-4H 417-4H (solo) | CPUs 414-4H 417-4H (redundant) |
| 39 | DIS_IRT | Verwerfen neuer Ereignisse | 555 | 535 | 580 | 535 | 731 | 732 |
| | | Sperrern aller Ereignisse (MODE = 0) | | | | | | |
| | | Sperrern aller Ereignisse einer Alarmklasse (MODE = 1) | 70 - 190 | 50 - 145 | 40 - 160 | 50-145 | 42-194 | 42-194 |
| | | Sperrern eines Ereignisses (MODE = 2) | 40 - 50 | 30 - 37 | 24 - 28 | 30 - 37 | 31 - 36 | 31 - 37 |
| 40 | EN_IRT | Verwerfen von Ereignissen aufheben | 555 | 535 | 580 | 535 | 736 | 737 |
| | | Freigeben aller Ereignisse (MODE = 0) | | | | | | |
| | | Freigeben aller Ereignisse einer Alarmklasse (MODE = 1) | 70 - 190 | 50 - 145 | 40 - 160 | 50 - 145 | 42 - 197 | 42 - 197 |
| | | Freigeben eines Ereignisses (MODE = 2) | 40 - 50 | 30 - 37 | 24 - 28 | 30 - 37 | 31 - 37 | 31 - 37 |
| 41 | DIS_AIRT | Verzögern von Alarmereignissen beim erstmaligen Aktivieren der Verzögerung ¹⁾ | 248 | 166 | 132 | 166 | 165 | 165 |
| | | wenn die Verzögerung schon aktiviert ist | 26 | 19 | 14 | 19 | 18 | 18 |

¹⁾ Die Laufzeit der SFC 41 beim erstmaligen Aktivieren der Verzögerung ist abhängig von der Prioritätsklasse, innerhalb der die SFC 41 aufgerufen wird. Die angegebene Laufzeit bezieht sich auf den Aufruf in OB 1. Sie nimmt mit steigender Prioritätsklassen-Nr. ab.

| SFC-Nr. | SFC-Name | Bedeutung | Ausführungszeit in μ s | | | | | |
|---------|----------|---|----------------------------|---------|---------|---------|---------------------------|--------------------------------|
| | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 | CPUs 414-4H 417-4H (solo) | CPUs 414-4H 417-4H (redundant) |
| 42 | EN_AIRT | Verzögern von Alarmereignissen aufheben beim Aufheben der letzten Verzögerung ²⁾ | 26 | 19 | 14 | 19 | 18 | 18 |
| | | wenn noch weitere Verzögerungen vorhanden sind | 435 | 320 | 272 | 320 | 343 | 343 |
| 43 | RE_TRIGR | Zykluszeitüberwachung nachtriggern | 155 | 104 | 84 | 104 | 118 | 307 |
| 44 | REPL_VAL | Ersatzwert in AKKU 1 übertragen | 30 | 21 | 16 | 21 | 20 | 20 |
| 46 | STP | CPU in STOP überführen nicht zu messen | -- | -- | -- | -- | -- | -- |
| 47 | WAIT | Programmbearbeitung verzögern zusätzlich zur Wartezeit | 13 - 18 | 7 - 15 | 4 - 11 | 7 - 15 | 6 - 13 | 6 - 13 |
| 48 | SNC_RTCB | Slave-Uhren synchronisieren | 25 | 19 | 14 | 19 | 18 | 41 |
| 49 | LGC_GADR | Den zu einer logischen Adresse gehörigen Steckplatz ermitteln | 55 | 40 | 31 | 40 | 41 | 41 |
| 50 | RD_LGADR | Sämtliche logischen Adressen einer Baugruppe ermitteln (Laufzeitangabe für 1 DI 32 im ZG) | 146 | 101 | 80 | 101 | 104 | 104 |

²⁾ Die Laufzeit der SFC 42 beim Aufheben der letzten Verzögerung ist abhängig von der Prioritätsklasse, innerhalb der die SFC 42 aufgerufen wird. Die angegebene Laufzeit bezieht sich auf den Aufruf in OB 1. Sie nimmt mit steigender Prioritätsklassen-Nr. ab.

| SFC-Nr. | SFC-Name | Bedeutung | Ausführungszeit in μ s | | | | | |
|---------|----------|---|----------------------------|--------------|--------------|--------------|---------------------------------|--------------------------------------|
| | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 | CPUs 414-4H 417-4H (solo) | CPUs 414-4H 417-4H (redundant) |
| 51 | RDSYSST | Liste aller SZL-Ausk. (0000) | 618 | 493 | 395 | 493 | 477 | 477 |
| | | Liste aller SZL-Ausk. (0F00) | 140 | 97 | 77 | 97 | 97 | 97 |
| 51 | RDSYSST | Teilliste "Baugr.-Identifikation" Auslesen aller Datensätze (0011) | 224 | 170 | 135 | 170 | 169 | 168 |
| | | Auslesen eines Datensatzes (0111) | 175 | 125 | 100 | 125 | 123 | 122 |
| | | Auslesen der Kopfinfo (0F11) | 145 | 100 | 80 | 100 | 99 | 99 |
| 51 | RDSYSST | Teilliste "CPU-Merkmale" Auslesen aller Datensätze (0012) | 317 | 235 | 187 | 235 | 233 | 232 |
| | | Auslesen eines Datensatzes (0112) | 190 - 215 | 135 - 155 | 108 - 123 | 135 - 155 | 135 - 155 | 135 - 154 |
| | | Auslesen der Kopfinfo (0F12) | 145 | 100 | 80 | 100 | 99 | 98 |
| 51 | RDSYSST | Teilliste "Speichern" Auslesen aller Datensätze (0013) | 185 | 134 | 105 | 134 | 134 | 133 |
| | | Auslesen eines Datensatzes (0113) | 185 | 134 | 105 | 134 | 134 | 133 |
| | | Auslesen der Kopfinfo (0F13) | 145 | 100 | 80 | 100 | 100 | 99 |

| SFC-Nr. | SFC-Name | Bedeutung | Ausführungszeit in μ s | | | | | |
|---------|----------|---|----------------------------|-----------|-----------|-----------|---------------------------|--------------------------------|
| | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 | CPUs 414-4H 417-4H (solo) | CPUs 414-4H 417-4H (redundant) |
| 51 | RDSYSST | Teilliste "Systembereiche" | 220 | 145 | 120 | 145 | 145 | 144 |
| | | Auslesen aller Datensätze (0014) | | | | | | |
| | | Auslesen eines Datensatzes (0114) | 170 | 117 | 93 | 117 | 117 | 117 |
| | | Auslesen der Kopfinfo (0F14) | 745 | 480 | 480 | 99 | 99 | 99 |
| 51 | RDSYSST | Teilliste "Bausteintypen" | 196 | 425 | 425 | 145 | 145 | 144 |
| | | Auslesen aller Datensätze (0015) | | | | | | |
| | | Auslesen eines Datensatzes (0115) | 165 - 185 | 118 - 128 | 94 - 102 | 118 - 128 | 119 - 128 | 118 - 127 |
| | | Auslesen der Kopfinfo (0F15) | 142 | 100 | 78 | 100 | 98 | 98 |
| 51 | RDSYSST | Teilliste "Prioritätsklassen" | 858 | 740 | 765 | 740 | 947 | 947 |
| | | Auslesen aller Datensätze (0016) | | | | | | |
| | | Auslesen eines Datensatzes (0116) | 196 - 347 | 110 - 250 | 110 - 135 | 110 - 250 | 137 - 291 | 137 - 290 |
| | | Auslesen der Kopfinfo (0F16) | 153 | 106 | 85 | 106 | 107 | 106 |
| 51 | RDSYSST | Teilliste "Zustand der Baugruppen-LEDs" | 322 | 216 | 175 | 216 | 225 | -- |
| | | Auslesen des Zustands aller LEDs (0019) | | | | | | |
| | | Auslesen des Zustands einer LED (0119) | 225 | 150 | 120 | 150 | 151 | -- |
| | | Auslesen der Kopfinfo (0F19) | 206 | 136 | 110 | 136 | 136 | -- |

| SFC-Nr. | SFC-Name | Bedeutung | Ausführungszeit in μ s | | | | | |
|---------|----------|--|----------------------------|------------------------|------------------------|------------------------|---------------------------|--------------------------------|
| | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 | CPUs 414-4H 417-4H (solo) | CPUs 414-4H 417-4H (redundant) |
| 51 | RDSYSST | Teilliste "Alarm-/Fehlerzuordnung" Auslesen aller Datensätze (0021) | 1225 | 1010 | 1055 | 1010 | 1298 | 1297 |
| | | Auslesen aller Datensätze einer Alarm- klasse (0121) | 210 - 590 | 145 - 410 | 115 - 330 | 145 - 410 | 145 - 365 | 144 - 364 |
| | | Auslesen eines Datensatzes (0221) | 195 - 215 | 135 - 150 | 110 - 120 | 135 - 150 | 135 - 152 | 135 - 151 |
| | | Auslesen aller belegten Alarme einer Klasse (0921) | 225 - 640 | 155 - 440 | 125 - 390 | 155 - 440 | 155 - 485 | 155 - 485 |
| | | Alternativ: n= Zahl der gel. OBs (0921) | (225/ 375)+ n*34 | (155/ 260)+ n*23 | (125/ 245)+ n*18 | (155/ 260)+ n*23 | (155/ 305)+ n*23 | (155/ 305)+ n*23 |
| 51 | RDSYSST | Teilliste "Alarm-/Fehlerzuordnung" Auslesen aller belegten Alarme (0A21) | 930 - 1510 | 795 - 1285 | 835 - 1390 | 795 - 1285 | 1037 - 1697 | 1037 - 1697 |
| | | Auslesen der Kopfinfo (0F21) | 155 | 107 | 85 | 107 | 108 | 107 |
| 51 | RDSYSST | Teilliste "Alarmstatus" Auslesen aller Datensätze einer Alarm- klasse (0122) | 225 - 660 | 160 - 490 | 125 - 390 | 160 - 490 | 157 - 432 | 160 - 450 |
| | | Auslesen eines Datensatzes (0222) | 210 - 225 | 148 - 158 | 118 - 128 | 148 - 158 | 148 - 155 | 148 - 158 |
| | | Auslesen aller belegten Alarme einer Klasse (0822) | 235 - 720 | 165 - 515 | 130 - 470 | 165 - 515 | 165 - 560 | 165 - 560 |

| SFC-Nr. | SFC-Name | Bedeutung | Ausführungszeit in μ s | | | | | |
|---------|----------|---|----------------------------|--------------------|--------------------|--------------------|---------------------------|--------------------------------|
| | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 | CPUs 414-4H 417-4H (solo) | CPUs 414-4H 417-4H (redundant) |
| | | Alternativ: n= Zahl der gel. OBs | (235/375) + n * 45 | (165/260) + n * 35 | (130/245) + n * 25 | (165/260) + n * 35 | (165/305) + n * 35 | (165/305) + n * 35 |
| | | Auslesen der Kopfinfo (0F22) | 158 | 110 | 87 | 110 | 44 | 108 |
| 51 | RDSYSST | Teilliste "Status der Prioritätsklassen" Auslesen eines Datensatzes (0123) | 210 | 147 | 117 | 147 | 147 | 147 |
| | | Alle Prioritätsklassen in Bearbeitung (0223) (n= Zahl der Prioritätsklassen) | 535 + n * 52 | 450 + n * 35 | 443 + n * 28 | 450 + n * 35 | 540 + n * 36 | 540 + n * 36 |
| | | Auslesen der Kopfinfo (0F23) | 145 | 100 | 80 | 100 | 101 | 100 |
| 51 | RDSYSST | Teilliste "Betriebszustände" Auslesen des letzten Betriebszustands-Übergangs (0124) | 200 | 140 | 111 | 140 | 139 | 138 |
| | | Auslesen des aktuellen Betriebszustands | 175 | 125 | 100 | 125 | 125 | 125 |
| 51 | RDSYSST | Teilliste "Zustandsinfo. Kommunikation" Auslesen Zustandsinfo einer Kommunikationseinheit (0132) INDEX=5 | 205 | 150 | 120 | 150 | 157 | 181 |

| SFC-Nr. | SFC-Name | Bedeutung | Ausführungszeit in μ s | | | | | |
|---------|----------|---|----------------------------|--------------|--------------|--------------|---------------------------------|--------------------------------------|
| | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 | CPUs 414-4H 417-4H (solo) | CPUs 414-4H 417-4H (redundant) |
| 51 | RDSYSST | Teilliste "Zustandsinfo. Kommunikation" Auslesen Zustandsinfo einer Kommunikationseinheit (0232) INDEX=4 | - | - | - | - | 235 | 425 |
| 51 | RDSYSST | Teilliste "Startinfofoliste" Alle Sync-Fehler-Startinfo einer Prioritätsklasse (0281) | 190 - 225 | 128 - 155 | 102 - 135 | 128 - 155 | 127 - 168 | 127 - 167 |
| | | Alle Startinfo einer Prioritätsklasse (0381) | 210 - 395 | 128 - 305 | 102 - 255 | 128 - 305 | 128 - 318 | 127 - 317 |
| | | Alle Sync-Fehler-Startinfo einer Prioritätsklasse vor Bearbeitung (0681) | 190 - 225 | 128 - 155 | 102 - 135 | 128 - 155 | 127 - 168 | 127 - 167 |
| | | Alle Startinfo einer Prioritätsklasse vor Bearb. (0781) | 190 - 390 | 145 - 295 | 115 - 235 | 145 - 295 | 142 - 293 | 141 - 293 |
| | | Alle Sync-Fehler-Startinfo einer Prioritätsklasse in Bearbeitung (0A81) | 190 - 225 | 130 - 160 | 102 - 135 | 130 - 160 | 129 - 170 | 128 - 169 |
| | | Alle Startinfo einer Prioritätsklasse in Bearb. (0B81) | 190 - 240 | 130 - 170 | 102 - 145 | 130 - 170 | 129 - 179 | 129 - 179 |
| | | Auslesen einer Kopfinfo (0F81) | 160 | 112 | 90 | 112 | 112 | 111 |

| SFC-Nr. | SFC-Name | Bedeutung | Ausführungszeit in μ s | | | | | |
|---------|----------|--|----------------------------|-----------|-----------|-----------|---------------------------|--------------------------------|
| | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 | CPUs 414-4H 417-4H (solo) | CPUs 414-4H 417-4H (redundant) |
| 51 | RDSYSST | Teilliste "Startinfoliste" Alle Sync-Fehler-Startereignis einer Prioritätsklasse (0282) | 190 - 220 | 128 - 150 | 102 - 125 | 128 - 150 | 128 - 156 | 128 - 155 |
| | | Alle Startereignis einer PK (0382) | 210 - 305 | 128 - 225 | 102 - 185 | 128 - 225 | 129 - 230 | 128 - 229 |
| | | Alle Sync-Fehler-Startereignis einer Prioritätsklasse vor Bearbeitung (0682) | 190 - 220 | 128 - 150 | 102 - 125 | 128 - 150 | 128 - 156 | 128 - 155 |
| | | Alle Startereignis einer Prioritätsklasse vor Bearb. (0782) | 210 - 310 | 145 - 225 | 115 - 180 | 145 - 225 | 143 - 225 | 142 - 223 |
| | | Alle Sync-Fehler-Startereignis einer Prioritätsklasse in Bearbeitung (0A82) | 190 - 220 | 130 - 150 | 102 - 125 | 130 - 150 | 130 - 158 | 129 - 161 |
| | | Alle Startereignis einer Prioritätsklasse in Bearb. (0B82) | 190 - 225 | 130 - 155 | 102 - 130 | 130 - 155 | 130 - 162 | 130 - 161 |
| | | Auslesen einer Kopfinfo (0F82) | 160 | 112 | 90 | 112 | 113 | 112 |

| SFC-Nr. | SFC-Name | Bedeutung | Ausführungszeit in μ s | | | | | |
|---------|----------|--|----------------------------|-----------------|-----------------|-----------------|---------------------------|--------------------------------|
| | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 | CPUs 414-4H 417-4H (solo) | CPUs 414-4H 417-4H (redundant) |
| 51 | RDSYSST | Teilliste "Baugruppenzustandsinfo" Auslesen der Zustandsinfo aller gesteckten Baugruppen (n=Zahl der DS) (0091) | 660 n * 22 | 508 + n * 19 | 408 + n * 16 | 508 + n * 19 | -- | -- |
| | | Auslesen der Zustandsinfo aller Baugruppen/Baugruppenträger mit falscher Typkennung (0191) | 570 + n * 70 | 427 n * 60 | 365 + n * 40 | 405+ n * 24 | -- | -- |
| 51 | RDSYSST | aller gestörten Baugruppen (0291) | 580 + n * 138 | 428 + n * 22 | 344 + n * 18 | 428 + n * 22 | -- | -- |
| | | aller nicht verfügbaren Baugruppen (0391) | 585 + n * 72 | 430 + n * 60 | 370 + n * 40 | 430 + n * 60 | -- | -- |
| | | Auslesen der Zustandsinfo aller Submodule der Host-Baugruppe im angegebenen Baugruppenträger (0991) | 354 + n * 30 | 250 + n * 26 | 200 + n * 21 | 250 + n * 26 | -- | -- |
| | | zentral, einer Baugruppe mit logischer Basisadresse (0C91) | 200 - 315 | 180 | 145 | 180 | 177 | 242 |
| | | dezentral, einer Baugruppe mit logischer Basisadresse (0C91) | 315 | 225 | 180 | 225 | 224 | 289 |

| SFC-Nr. | SFC-Name | Bedeutung | Ausführungszeit in μ s | | | | | |
|---------|----------|--|----------------------------|-----------|-----------|-----------|---------------------------|--------------------------------|
| | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 | CPUs 414-4H 417-4H (solo) | CPUs 414-4H 417-4H (redundant) |
| 51 | RDSYSST | Teilliste "Baugruppenzustandsinfo" einer Baugruppe (dezentral) mit logischer Basisadresse (4C91) Erstaufruf | 200 - 315 | 145 - 240 | 130 - 190 | 145 - 240 | 242 | 305 |
| | | Teilliste "Baugruppenzustandsinfo" einer Baugruppe (dezentral) mit logischer Basisadresse (4C91) Zwischenaufruf | -- | -- | -- | -- | 148 | 148 |
| | | Teilliste "Baugruppenzustandsinfo" einer Baugruppe (dezentral) mit logischer Basisadresse (4C91) Letztaufruf | -- | -- | -- | -- | 167 | 167 |

| SFC-Nr. | SFC-Name | Bedeutung | Ausführungszeit in μs | | | | | |
|---------|----------|---|----------------------------------|----------------|----------------|----------------|---------------------------|--------------------------------|
| | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 | CPUs 414-4H 417-4H (solo) | CPUs 414-4H 417-4H (redundant) |
| 51 | RDSYSST | zentral aller Baugruppen im angegebenen Baugruppenträger (n=Zahl der DS) (0D91) | 377 + n* 13 | 275 + n* 16 | 240 + n* 10 | 275 + n* 16 | 260 + n* 20 | 405 + n* 23 |
| | | dezentral aller Baugruppen in der angegebenen DP-Station (0D91) | 330 - 390 | 250 - 300 | 200 - 240 | 250 - 300 | 305 | 408 - 420 |
| | | Auslesen einer Kopfinfo (0F91) | 560 | 435 | 350 | 435 | -- | -- |
| 51 | RDSYSST | Teilliste "Baugruppenträger-/Stationszustandsinformation" zentral Auslesen des Sollzustands von Baugruppenträger 0 (0092) | 180 | 127 | 100 | 127 | 130 | 154 |
| | | dezentral Auslesen des Sollzustands von DP-System 1 (0092) | 900 | 725 | 585 | 725 | 712 | 743 |

| SFC-Nr. | SFC-Name | Bedeutung | Ausführungszeit in μ s | | | | | |
|---------|----------|--|----------------------------|-------------|--------------|-------------|---------------------------|--------------------------------|
| | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 | CPUs 414-4H 417-4H (solo) | CPUs 414-4H 417-4H (redundant) |
| 51 | RDSYSST | zentral Auslesen des Istzustands von Baugruppenträger 0 (0292) | 180 | 127 | 103 | 127 | 131 | 155 |
| | | dezentral Auslesen Istzustands von DP-System 1 (0292) | 940 | 745 | 600 | 745 | 725 | 757 |
| | | Auslesen der Kopfinfo (0F92) | 160 | 113 | 90 | 113 | 113 | 113 |
| 51 | RDSYSST | Teilliste "Diagnosepuffer" Auslesen aller im aktuellen Betriebszustand lieferbaren Ereignisinfos (max. 23) (00A0) | 195 - 525 | 138 - 410 | 110 - 330 | 138 - 410 | 140 - 412 | 140 - 412 |
| | | Auslesen der n neuesten Einträge (n = 1-23) (01A0) | 195 + n* 14,5 | 138 + n* 12 | 110 + n* 9,5 | 138 + n* 12 | 140 + n* 12 | 140 + n* 12 |
| | | Auslesen der Standard-OB-Startinformationen (04A0) Max-Wert von 04A0 ist berechnet | 195 - 1270 | 138 - 1530 | 110 - 1095 | 138 - 1530 | 140 - 1540 | 140 - 1540 |

| SFC-Nr. | SFC-Name | Bedeutung | Ausführungszeit in μ s | | | | | |
|---------|----------|--|----------------------------|---------------|---------------|---------------|---------------------------|--------------------------------|
| | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 | CPUs 414-4H 417-4H (solo) | CPUs 414-4H 417-4H (redundant) |
| 51 | RDSYSST | Auslesen aller Kommunikations-Informationen (05A0) Auslesen aller OVS-Informationen (06A0) Auslesen aller TIS-Informationen (07A0) Auslesen aller BZ-Informationen (08A0) Auslesen aller Asynchron-Fehler-Startinformationen (09A0) Auslesen aller Synchronfehler-Startinformationen (0AA0) Auslesen aller STOPP-/Abbruch-/BZ-Übergangs-Informationen (0BA0) Auslesen aller HF-Informationen (0CA0) Auslesen aller DiagnoseInformationen (0DA0) Auslesen aller Anwenderinformationen (0EA0) | 195 - 1270 | 138 - 1530 | 110 - 1095 | 138 - 1530 | 140 - 1540 | 140 - 1540 |
| | | Auslesen der Kopfinfo (0FA0) | 167 | -- | 90 | -- | 114 | 114 |
| 51 | RDSYSST | Teilliste "Diagnosedaten DS 0" Auslesen über log. Basisadresse (00B1) zentral | 406 | 286 | 233 | 286 | 300 | 360 |

| SFC-Nr. | SFC-Name | Bedeutung | Ausführungszeit in μ s | | | | | |
|---------|----------|---|----------------------------|---------|---------|---------|---------------------------|--------------------------------|
| | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 | CPUs 414-4H 417-4H (solo) | CPUs 414-4H 417-4H (redundant) |
| 51 | RDSYSST | dezentral Erstaufruf | 392 | 270 | 217 | 270 | 278 | 356 |
| | | dezentral Zwischenaufruf, REQ = 0 | 215 | 150 | 120 | 150 | 153 | 152 |
| | | dezentral Letztaufruf | 405 | 165 | 132 | 165 | 170 | 169 |
| 51 | RDSYSST | Teilliste "Diagnosedaten DS 1" Auslesen über physikal. Adresse (00B2) Auslesen eines 16 Byte langen DS 1 | 408 | 300 | 250 | 300 | 313 | 375 |
| 51 | RDSYSST | Teilliste "Diagnosedaten DS 1" Auslesen über log. Basisadresse (00B3) Auslesen eines 16 Byte langen DS 1 zentral | 447 | 324 | 268 | 324 | 340 | 402 |
| | | dezentral Erstaufruf | 395 | 270 | 218 | 270 | 272 | 356 |
| | | dezentral Zwischenaufruf | 218 | 150 | 120 | 150 | 153 | 153 |
| | | dezentral Letztaufruf | 257 | 178 | 142 | 178 | 182 | 182 |

DS = Datensatz

| SFC-Nr. | SFC-Name | Bedeutung | Ausführungszeit in μ s | | | | | |
|---------|----------|--|----------------------------|---------|---------|---------|---------------------------|--------------------------------|
| | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 | CPUs 414-4H 417-4H (solo) | CPUs 414-4H 417-4H (redundant) |
| 51 | RDSYSST | Teilliste "Diagnosedaten DP-Slave" Auslesen über projektierte Diagnose- adresse (00B4) Erstaufruf | 385 | 266 | 213 | 266 | 272 | 351 |
| | | Zwischenaufruf, REQ = 0 | -- | -- | 115 | -- | 149 | 148 |
| | | Letztaufruf (6 - 240 Byte) | 246 | 170 | 135 | 170 | 174 | 173 |
| 52 | WR_USMSG | Anwendereintrag in Diagnosepuffer schreiben mit Meldung | 186 | 128 | 102 | 128 | 75 | 100 |
| | | ohne Meldung | 107 | 75 | 60 | 75 | 74 | 98 |
| 54 | RD_DPARM | Dynamische Parameter lesen zentral AI 8*13 Bit | 180 | 125 | 95 | 125 | 126 | 153 |
| | | dezentral AI 8*12 Bit (DS1 = 14 Byte) | 200 | 135 | 105 | 135 | 121 | 121 |
| 55 | WR_PARM | Dynamische Parameter schreiben zentral AI 8*13 Bit | 485 | 345 | 280 | 345 | 360 | 418 |
| | | dezentral Erstaufruf AI 8*12 Bit (14 - 240 Byte) | 370 | 260 | 210 | 260 | 268 | 347 |
| | | dezentral Folge-/Letztaufruf, REQ = 0 | 175 | 115 | 90 | 115 | 122 | 122 |

| SFC-Nr. | SFC-Name | Bedeutung | Ausführungszeit in μ s | | | | | |
|---------|-----------|--|----------------------------|---------|---------|---------|---------------------------------|--------------------------------------|
| | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 | CPUs 414-4H 417-4H (solo) | CPUs 414-4H 417-4H (redundant) |
| 56 | WR_DPARAM | Vordefinierte dynamische Parameter schreiben AI 8*13 Bit zentral | 445 | 336 | 280 | 336 | 353 | 411 |
| | | dezentral Erstaufruf AI 8*12 Bit (2 - 240 Byte) | 300 | 205 | 165 | 205 | 217 | 296 |
| | | Folge-/Letztaufruf | 145 | 100 | 80 | 100 | 106 | 106 |
| 57 | PARAM_MOD | Baugruppe parametrieren zentral BG/DS-Anzahl/DS-Längen in Byte AI 8*13 Bit | 770 | 580 | 490 | 580 | 609 | 695 |
| | | dezentral AO 8*12 Bit Erstaufruf (16 - 240 Byte) | 300 | 205 | 165 | 205 | 215 | 295 |
| | | dezentral Folge-/Letztaufruf | 145 | 100 | 80 | 100 | 104 | 104 |

| SFC-Nr. | SFC-Name | Bedeutung | Ausführungszeit in μ s | | | | | |
|---------|----------|--|----------------------------|-------------------|-------------------|-------------------|---------------------------|--------------------------------|
| | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 | CPUs 414-4H 417-4H (solo) | CPUs 414-4H 417-4H (redundant) |
| 58 | WR_REC | Parameterdatensatz schreiben zentral (n = Anzahl Byte) | 390 + n * 2,87 | 267 + n * 2,71 | 217 + n * 2,52 | 267 + n * 2,71 | 282 + n * 2,68 | 311 + n * 2,71 |
| | | Erstaufruf, integrierte DP-Anschaltung (n = Anzahl Byte) | 334 + n * 0,42 | 228 + n * 0,35 | 182 + n * 0,30 | 228 + n * 0,35 | 222 + n * 0,39 | 276 + n * 0,32 |
| | | Zwischenaufruf, REQ = 0, integrierte DP-Anschaltung | 138 | 90 | 70 | 90 | 94 | 94 |
| | | Letztaufruf, integrierte DP-Anschaltung | 138 | 90 | 70 | 90 | 95 | 94 |
| | | Erstaufruf, externe DP-Anschaltung (n = Anzahl Byte) | 322 + n * 0,32 | 215 + n * 0,26 | 171 + n * 0,23 | 215 + n * 0,26 | 208 + n * 0,26 | 208 + n * 0,29 |
| | | Zwischenaufruf, REQ = 0, externe DP-Anschaltung | 139 | 90 | 72 | 90 | 95 | 94 |
| | | Letztaufruf, externe DP-Anschaltung | 140 | 91 | 72 | 91 | 95 | 95 |

| SFC-Nr. | SFC-Name | Bedeutung | Ausführungszeit in μs | | | | | |
|---------|----------|---|----------------------------------|----------|----------|----------|---------------------------|--------------------------------|
| | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 | CPUs 414-4H 417-4H (solo) | CPUs 414-4H 417-4H (redundant) |
| 59 | RD_REC | Datensatz lesen | 390 + | 267 + | 218 + | 267 + | 282 + | 342 + |
| | | Erstaufruf, zentral (n = Anzahl Byte) | n * 3,13 | n * 2,90 | n * 2,71 | n * 2,90 | n * 2,97 | n * 3,13 |
| | | Erstaufruf, integrierte DP-Anschaltung | 322 | 217 | 172 | 217 | 212 | 264 |
| | | Zwischenaufruf, REQ = 0, integrierte DP-Anschaltung | 138 | 90 | 70 | 90 | 95 | 94 |
| | | Letztaufruf, integrierte DP-Anschaltung (n = Anzahl Byte) | 198 + | 132 + | 106 + | 132 + | 138 + | 138 + |
| | | Erstaufruf, externe DP-Anschaltung | n * 0,35 | n * 0,33 | n * 0,27 | n * 0,33 | n * 0,33 | n * 0,33 |
| | | Zwischenaufruf, REQ = 0, externe DP-Anschaltung | 304 | 204 | 163 | 204 | 198 | 197 |
| 60 | GD_SND | Zwischenaufruf, REQ = 0, externe DP-Anschaltung | 139 | 91 | 72 | 91 | 95 | 94 |
| | | Letztaufruf, externe DP-Anschaltung (n = Anzahl Byte) | 200 + | 132 + | 105 + | 132 + | 136 + | 136 + |
| | | n * 0,33 | n * 0,2 | n * 0,2 | n * 0,2 | n * 0,33 | n * 0,27 | |
| 60 | GD_SND | GD-Paket senden 1 Byte | 295 | 215 | 175 | 215 | -- | -- |
| | | 32 Byte | 910 | 640 | 515 | 640 | -- | -- |
| 61 | GD_RCV | GD-Paket übernehmen (1 - 32 Byte) | 145 | 105 | 85 | 105 | -- | -- |

| SFC-Nr. | SFC-Name | Bedeutung | Ausführungszeit in μ s | | | | | |
|---------|----------|--|----------------------------|--------------|--------------|--------------|---------------------------|--------------------------------|
| | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 | CPUs 414-4H 417-4H (solo) | CPUs 414-4H 417-4H (redundant) |
| 62 | CONTROL | den Zustand der Verbindung, die zu einer lokalen Kommunikations-SFB-Instanz gehört, abfragen | 116 | 87 | 69 | 87 | 107 | 136 |
| 64 | TIME_TCK | Millisekundentimer auslesen | 24 | 19 | 15 | 19 | 19 | 47 |
| 65 | X_SEND | Daten an externen Partner senden Erstaufruf, Verbindung aufbauen (1-76 Byte) REQ = 1 | 860 - 910 | 710 - 740 | 765 - 795 | 710 - 740 | -- | -- |
| | | Erstaufruf Verbindung vorhanden (1-76 Byte) | 590 - 635 | 400 - 430 | 320 - 345 | 400 - 430 | -- | -- |
| | | Zwischenaufruf (1-76 Byte) | 180 | 130 | 100 | 130 | -- | -- |
| | | Letztaufruf, BUSY = 0 | 285 | 195 | 155 | 195 | -- | -- |
| 66 | X_RCV | Daten von externem Partner empfangen Empfang prüfen (1-76) Byte | 92 | 65 | 55 | 65 | -- | -- |
| | | Daten lesen (1-76 Byte) | 275 - 315 | 190 - 220 | 150 - 175 | 190 - 220 | -- | -- |

| SFC-Nr. | SFC-Name | Bedeutung | Ausführungszeit in μ s | | | | | |
|---------|----------|---|----------------------------|--------------|--------------|--------------|---------------------------|--------------------------------|
| | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 | CPUs 414-4H 417-4H (solo) | CPUs 414-4H 417-4H (redundant) |
| 67 | X_GET | Daten aus externem Partner lesen Erstaufruf, Verbindung aufbauen (1-76 Byte) REQ = 1 | 760 | 645 | 715 | 645 | -- | -- |
| | | Erstaufruf Verbindung vorhanden (1-76 Byte) | 490 | 335 | 265 | 335 | -- | -- |
| | | Zwischenaufruf (1-76 Byte) | 195 | 135 | 110 | 135 | -- | -- |
| | | Letztaufruf, BUSY = 0 | 450 - 490 | 310 - 340 | 245 - 270 | 310 - 340 | -- | -- |
| 68 | X_PUT | Daten in externen Partner schreiben Erstaufruf, Verbindung aufbauen (1-76 Byte) REQ = 1 | 880 - 925 | 725 - 755 | 780 - 810 | 725 - 755 | -- | -- |
| | | Erstaufruf Verbindung vorhanden (1-76 Byte) | 610 - 655 | 415 - 445 | 330 - 360 | 415 - 445 | -- | -- |
| | | Zwischenaufruf (1-76 Byte) | 195 | 135 | 110 | 135 | -- | -- |
| | | Letztaufruf, BUSY = 0 | 300 | 205 | 162 | 205 | -- | -- |

| SFC-Nr. | SFC-Name | Bedeutung | Ausführungszeit in μ s | | | | | |
|---------|----------|---|----------------------------|-----------|-----------|-----------|---------------------------|--------------------------------|
| | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 | CPUs 414-4H 417-4H (solo) | CPUs 414-4H 417-4H (redundant) |
| 69 | X_ABORT | Verbindung zu externem Partner abbrechen Erstaufruf, REQ = 1 | 220 | 160 | 125 | 160 | -- | -- |
| | | Zwischenaufruf | 125 | 90 | 70 | 90 | -- | -- |
| | | Letztaufruf, BUSY = 0 | 365 | 375 | 75 - 500 | 375 | -- | -- |
| 72 | I_GET | Daten aus internem Partner lesen Erstaufruf, Verbindung aufbauen (1-76 Byte) REQ = 1 | 815 | 680 | 745 | 680 | -- | -- |
| | | Erstaufruf Verbindung vorhanden (1-76 Byte) | 505 | 345 | 275 | 345 | -- | -- |
| | | Zwischenaufruf (1-76 Byte) | 205 | 145 | 115 | 145 | -- | -- |
| | | Letztaufruf, BUSY = 0 | 460 - 505 | 315 - 345 | 250 - 275 | 315 - 345 | -- | -- |
| 73 | I_PUT | Daten in internen Partner schreiben Erstaufruf, Verbindung aufbauen (1-76 Byte) REQ = 1 | 690 - 980 | 430 - 800 | 340 - 840 | 430 - 800 | -- | -- |
| | | Erstaufruf Verbindung vorhanden (1-76 Byte) | 625 - 665 | 425 - 455 | 340 - 365 | 425 - 455 | -- | -- |
| | | Zwischenaufruf (1-76 Byte) | 205 | 145 | 115 | 145 | -- | -- |
| | | Letztaufruf, BUSY = 0 | 310 | 215 | 170 | 215 | -- | -- |

| SFC-Nr. | SFC-Name | Bedeutung | Ausführungszeit in μs | | | | | |
|---------|--------------------|--|----------------------------------|------------------|------------------|------------------|---------------------------------|--------------------------------------|
| | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 | CPUs 414-4H 417-4H (solo) | CPUs 414-4H 417-4H (redundant) |
| 74 | I_ABORT | Verbindung zu internem Partner abbrechen Erstaufruf, REQ = 1 | 225 | 160 | 125 | 160 | -- | -- |
| | | Zwischenaufruf | 125 | 90 | 75 | 90 | -- | -- |
| | | Letztaufruf, ohne / mit Verbindung, BUSY = 0 | 365 | 380 | 70 / 503 | 380 | -- | -- |
| 79 | SET ¹⁾ | Bitfeld im Peripheriebereich setzen n = Anzahl der auf 1 zu setzenden Bits | 43 + n * 0,39 | 28 + n * 0,32 | 23 + n * 0,26 | 28 + n * 0,32 | 53 + n * 1,35 | 80 + n * 1,32 |
| 80 | RSET ¹⁾ | Bitfeld im Peripheriebereich löschen n = Anzahl der auf 0 zu setzenden Bits | 43 + n * 0,39 | 28 + n * 0,32 | 23 + n * 0,26 | 28 + n * 0,32 | 53 + n * 1,35 | 80 + n * 1,32 |
| 81 | UBLKMOV | Variable ununterbrechbar kopieren n = Anzahl der zu kopierenden Bytes | 62 + n * 0,30 | 44 + n * 0,17 | 33 + n * 0,17 | 44 + n * 0,17 | 43 + n * 0,17 | 42 + n * 0,17 |
| 87 | C_DIAG | Aktuellen Verbindungszustand ermitteln MODE = 0 | 42 | 28 | 22 | 27 | 36 | 36 |
| | | Mode = 1, 2, 3 | 187 | 189 | 276 | 346 | 249 | 249 |
| 90 | H_CTRL | Abläufe bei H-Systemen beeinflussen | -- | -- | -- | -- | 19 - 21 | 19 - 21 |

¹⁾ Gemessen mit Peripheriebaugruppen vom Typ "Simulator Binär C79459-A1002-A1, Ausgabestand 1" im Zentralgerät

| SFC-Nr. | SFC-Name | Bedeutung | Ausführungszeit in μs | | | | | |
|---------|----------|--|----------------------------------|---------|---------|---------|---------------------------|--------------------------------|
| | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 | CPUs 414-4H 417-4H (solo) | CPUs 414-4H 417-4H (redundant) |
| 100 | SET_CLKS | Uhrzeit stellen und Uhrzeitstatus setzen MODE = 1 | 370 | 263 | 227 | 263 | 439 | 1169 |
| | | MODE = 2 | 125 | 84 | 67 | 84 | 192 | 403 |
| | | MODE = 3 | 375 | 266 | 232 | 266 | 442 | 1167 |
| 103 | DP_TOPOL | Ermitteln der Bustopologie in einem DP-Mastersystem Erstauf Ruf, REQ = 1 | 325 | 213 | 170 | 213 | 227 | 345 |
| | | Zwischenauf Ruf | 63 | 45 | 36 | 45 | 51 | 51 |
| | | Letztauf Ruf BUSY = 0 | 66 | 47 | 38 | 47 | 53 | 56 |
| 104 | CIR | Steuern des CiR-Vorgangs MODE = 0, Auskunft | 23 | 18 | 14 | 18 | 19 | – |
| | | MODE = 1, CiR-Vorgang freigeben | 24 | 18 | 15 | 18 | 19 | – |
| | | MODE = 2, CiR-Vorgang gänzlich sperren | 24 | 18 | 15 | 18 | 19 | – |
| | | MODE = 3, CiR-Vorgang bedingt sperren | 26 | 19 | 16 | 19 | 20 | – |

| SFC-Nr. | SFC-Name | Bedeutung | Ausführungszeit in μ s | | | | | |
|---------|----------|---|----------------------------|--------------------------|--------------------------|--------------------------|---------------------------|--------------------------------|
| | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 | CPUs 414-4H 417-4H (solo) | CPUs 414-4H 417-4H (redundant) |
| 105 | READ_SI | Auslesen dynamisch belegter Systemressourcen MODE = 0 | 176 - 1807 ⁰⁾ | 117 - 3574 ⁰⁾ | 94 - 2859 ⁰⁾ | 117 - 3574 ⁰⁾ | 117 - 3205 ⁰⁾ | 117 - 3206 ⁰⁾ |
| | | MODE = 1 | 204 - 2098 ¹⁾ | 138 - 4128 ¹⁾ | 110 - 3302 ¹⁾ | 138 - 4128 ¹⁾ | 136 - 3802 ¹⁾ | 303 - 3971 ¹⁾ |
| | | MODE = 2 | 205 - 1478 ¹⁾ | 140 - 2868 ¹⁾ | 111 - 2294 ¹⁾ | 140 - 2868 ¹⁾ | 137 - 2901 ¹⁾ | 304 - 3069 ¹⁾ |
| | | MODE = 3 | 206 - 2152 ²⁾ | 140 - 4129 ²⁾ | 111 - 3303 ²⁾ | 140 - 4129 ²⁾ | 137 - 3802 ²⁾ | 305 - 3970 ²⁾ |

0) Abhängig von der Größe des Zielbereichs SYS_INST und der Anzahl der noch auszulesenden Systemressourcen

1) Abhängig von der Anzahl aktiver Meldungen (belegte Systemressourcen)

2) Abhängig von der Anzahl aktiver Meldungen (belegte Systemressourcen) und der Anzahl belegter Instanzen mit der gesuchten CMP_ID.

| SFC-Nr. | SFC-Name | Bedeutung | Ausführungszeit in μ s | | | | | |
|---------|----------|---|----------------------------|--------------------------|-------------------------|--------------------------|---------------------------|--------------------------------|
| | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 | CPUs 414-4H 417-4H (solo) | CPUs 414-4H 417-4H (redundant) |
| 106 | DEL_SI | Freigeben dynamisch belegter Systemressourcen MODE = 1 | 176 - 1289 ¹⁾ | 125 - 2666 ¹⁾ | 97 - 2131 ¹⁾ | 125 - 2666 ¹⁾ | 145 - 6954 ¹⁾ | 507 - 23875 ¹⁾ |
| | | MODE = 2 | 180 - 1272 ¹⁾ | 127 - 2580 ¹⁾ | 99 - 2061 ¹⁾ | 127 - 2580 ¹⁾ | 147 - 2668 ¹⁾ | 510 - 3033 ¹⁾ |
| | | MODE = 3 | 177 - 1350 ²⁾ | 125 - 2705 ²⁾ | 98 - 2162 ²⁾ | 125 - 2705 ²⁾ | 145 - 6974 ²⁾ | 507 - 23906 ²⁾ |
| 107 | ALARM_DQ | Quittierbare bausteinbezogene Meldungen erzeugen Erstaufruf, SIG = 0 -> 1 | 497 | 336 | 267 | 336 | 349 | 566 |
| | | Leeraufruf | 145 | 98 | 78 | 98 | 101 | 157 |

1) Abhängig von der Anzahl aktiver Meldungen (belegte Systemressourcen)

2) Abhängig von der Anzahl aktiver Meldungen (belegte Systemressourcen) und der Anzahl belegter Instanzen mit der gesuchten CMP_ID.

| SFC-Nr. | SFC-Name | Bedeutung | Ausführungszeit in μ s | | | | | |
|---------|----------|---|----------------------------|---------|---------|---------|---------------------------|--------------------------------|
| | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 | CPUs 414-4H 417-4H (solo) | CPUs 414-4H 417-4H (redundant) |
| 108 | ALARM_D | Nicht quittierbare bausteinbezogene Meldungen erzeugen Erstaufruf, SIG = 0 -> 1 | 499 | 337 | 266 | 337 | 350 | 548 |
| | | Leeraufruf | 146 | 98 | 78 | 98 | 101 | 156 |
| 126 | SYNC_PI | Teilprozessabbild der Eingänge taktsynchron aktualisieren | | | | | | |
| 127 | SYNC_PO | Teilprozessabbild der Ausgänge taktsynchron aktualisieren | | | | | | |

Systemfunktionsbausteine

Die nachfolgende Tabelle listet die Systemfunktionsbausteine auf, die vom Betriebssystem der S7-400 CPUs bereitgestellt werden, und die Ausführungszeiten auf der jeweiligen CPU (X: Funktion vorhanden, Ausführungszeiten lagen bei Drucklegung noch nicht vor).

| SFB-Nr. | SFB-Name | Bedeutung | Ausführungszeit in μ s | | | | | |
|---------|----------|---|----------------------------|--------------|--------------|--------------|---------------------------------|--------------------------------------|
| | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 | CPUs 414-4H 417-4H (solo) | CPUs 414-4H 417-4H (redundant) |
| 0 | CTU | Vorwärtszählen | 26 | 16 | 13 | 16 | 17 | 16 |
| 1 | CTD | Rückwärtszählen | 25 | 17 | 13 | 17 | 17 | 17 |
| 2 | CTUD | Vorwärts- und Rückwärtszählen | 29 | 19 | 15 | 19 | 19 | 19 |
| 3 | TP | Impuls erzeugen | 34 | 23 | 18 | 23 | 24 | 52 |
| 4 | TON | Einschaltverzögerung erzeugen | 34 | 23 | 18 | 23 | 24 | 52 |
| 5 | TOF | Ausschaltverzögerung erzeugen | 36 | 24 | 19 | 24 | 20 | 53 |
| 8 | USEND | Daten unkoordiniert senden (ein Sendeparameter versorgt) Auftragsaktivierung (1 - 440 Byte) | 473 - 737 | 318 - 509 | 253 - 407 | 317 - 509 | 330 - 436 | 425 - 542 |
| | | Auftragsüberprüfung | 159 | 107 | 86 | 108 | 115 | 145 |
| | | Auftragsende, DONE = 1 | 152 | 103 | 82 | 104 | 107 | 137 |

| SFB-Nr. | SFB-Name | Bedeutung | Ausführungszeit in μ s | | | | | |
|---------|----------|--|----------------------------|--------------|--------------|--------------|---------------------------|--------------------------------|
| | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 | CPUs 414-4H 417-4H (solo) | CPUs 414-4H 417-4H (redundant) |
| 9 | URCV | Daten unkoordiniert empfangen (ein Empfangsparameter versorgt) Auftragsaktivierung | 137 | 93 | 74 | 94 | 100 | 130 |
| | | Auftragsüberprüfung | 137 | 93 | 74 | 94 | 100 | 130 |
| | | Auftragsende (NDR = 1; 1 - 440 Byte) | 345 - 610 | 232 - 421 | 186 - 337 | 233 - 421 | 243 - 363 | 314 - 435 |
| 12 | BSEND | Daten blockorientiert senden Auftragsaktivierung (1 - 3000 Byte) | 386 | 258 | 207 | 258 | 264 | 323 |
| | | Auftragsüberprüfung | 171 | 115 | 92 | 116 | 122 | 152 |
| | | Auftragsende, DONE = 1 | 165 | 110 | 88 | 111 | 115 | 145 |
| 13 | BRCV | Daten blockorientiert empfangen Auftragsaktivierung (1 - 3000 Byte) | 203 | 138 | 110 | 139 | 145 | 175 |
| | | Auftragsüberprüfung | 161 | 110 | 88 | 111 | 117 | 147 |
| | | Auftragsende | 162 | 109 | 87 | 110 | 113 | 143 |

| SFB-Nr. | SFB-Name | Bedeutung | Ausführungszeit in μ s | | | | | |
|---------|----------|--|----------------------------|--------------|--------------|--------------|---------------------------|--------------------------------|
| | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 | CPUs 414-4H 417-4H (solo) | CPUs 414-4H 417-4H (redundant) |
| 14 | GET | Daten aus remoter CPU lesen (ein Bereich angegeben) Auftragsaktivierung | 336 | 227 | 183 | 228 | 227 | 297 |
| | | Auftragsüberprüfung | 161 | 109 | 87 | 110 | 116 | 146 |
| | | Auftragsende, NDR = 1 (1 - 450 Byte) | 344 - 626 | 231 - 431 | 185 - 345 | 232 - 432 | 243 - 369 | 314 - 441 |
| 15 | PUT | Daten in remote CPU schreiben (ein Bereich angegeben) Auftragsaktivierung (1 - 404 Byte) | 498 - 748 | 337 - 513 | 269 - 410 | 337 - 515 | 349 - 458 | 443 - 552 |
| | | Auftragsüberprüfung | 161 | 108 | 87 | 109 | 116 | 146 |
| | | Auftragsende, DONE = 1 | 154 | 104 | 83 | 105 | 108 | 138 |
| 16 | PRINT | Daten an einen Drucker senden Auftragsaktivierung, REQ = 1 | 513 - 757 | 338 - 516 | 271 - 414 | 339 - 518 | 354 - 462 | 449 - 545 |
| | | Auftragsüberprüfung | 160 | 107 | 86 | 108 | 115 | 145 |
| | | Auftragsende, DONE = 1 | 153 | 103 | 82 | 104 | 107 | 137 |

| SFB-Nr. | SFB-Name | Bedeutung | Ausführungszeit in μ s | | | | | |
|---------|----------|--|----------------------------|---------|---------|---------|---------------------------------|--------------------------------------|
| | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 | CPUs 414-4H 417-4H (solo) | CPUs 414-4H 417-4H (redundant) |
| 19 | START | Neustart (Warmstart) oder Kaltstart in remotem Gerät durchführen Auftragsaktivierung, REQ = 1 | 497 | 333 | 265 | 333 | 339 | 408 |
| | | Auftragsüberprüfung | 169 | 114 | 91 | 115 | 121 | 151 |
| | | Auftragsende, DONE = 1 | 164 | 110 | 88 | 111 | 115 | 146 |
| 20 | STOP | Remotes Gerät in STOP versetzen Auftragsaktivierung, REQ = 1 | 472 | 314 | 251 | 314 | 322 | 384 |
| | | Auftragsüberprüfung | 169 | 114 | 91 | 115 | 121 | 151 |
| | | Auftragsende, DONE = 1 | 164 | 110 | 88 | 111 | 115 | 146 |
| 21 | RESUME | Wiederanlauf in remotem Gerät durchführen Auftragsaktivierung, REQ = 1 | 496 | 334 | 265 | 332 | 339 | 399 |
| | | Auftragsüberprüfung | 169 | 114 | 91 | 115 | 121 | 151 |
| | | Auftragsende, DONE = 1 | 164 | 110 | 88 | 111 | 115 | 145 |
| 22 | STATUS | Gerätestatus eines remoten Partners abfragen Auftragsaktivierung, REQ = 1 | 268 | 183 | 146 | 184 | 188 | 258 |
| | | Auftragsüberprüfung | 161 | 108 | 87 | 109 | 116 | 146 |
| | | Auftragsende, NDR = 1 | 604 | 404 | 323 | 404 | 415 | 486 |

| SFB-Nr. | SFB-Name | Bedeutung | Ausführungszeit in μ s | | | | | |
|---------|-----------|---|----------------------------|--------------|--------------|--------------|---------------------------------|--------------------------------------|
| | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 | CPUs 414-4H 417-4H (solo) | CPUs 414-4H 417-4H (redundant) |
| 23 | USTATUS | Remoten Gerätestatus unkoordiniert empfangen Auftragsaktivierung, NDR = 1 | 137 | 93 | 74 | 94 | 100 | 131 |
| | | Auftragsüberprüfung | 137 | 93 | 74 | 94 | 100 | 130 |
| | | Auftragsende | 604 | 404 | 323 | 404 | 415 | 486 |
| 31 | NOTIFY_8P | Bausteinbezogene Meldung ohne Quittierungsanzeige erzeugen Auftragsaktivierung, SIG = 0→ 1 (1 - 420 Byte) | 561 - 823 | 373 - 580 | 301 - 462 | 373 - 580 | 384 - 510 | 519 - 644 |
| | | Auftragsüberprüfung | 186 | 125 | 100 | 125 | 133 | 163 |
| | | Auftragsende, DONE = 1 | 191 | 128 | 102 | 128 | 130 | 160 |
| 32 | DRUM | Schrittschaltwerk realisieren | 52 | 33 | 26 | 33 | 35 | 62 |
| 33 | ALARM | Bausteinbezogene Meldung mit Quittierungsanzeige erzeugen Auftragsaktivierung, SIG = 0→ 1 (1 - 420 Byte) | 581 - 843 | 386 - 587 | 307 - 470 | 385 - 589 | 392 - 518 | 527 - 652 |
| | | Auftragsüberprüfung | 205 | 136 | 109 | 137 | 141 | 171 |
| | | Auftragsende, DONE = 1 | 207 | 137 | 110 | 138 | 136 | 166 |

| SFB-Nr. | SFB-Name | Bedeutung | Ausführungszeit in μ s | | | | | |
|---------|----------|---|----------------------------|-----------|-----------|-----------|---------------------------------|--------------------------------------|
| | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 | CPUs 414-4H 417-4H (solo) | CPUs 414-4H 417-4H (redundant) |
| 34 | ALARM_8 | Bausteinbezogene Meldung ohne Begleitwerte für 8 Signale erzeugen Auftragsaktivierung, SIG = 0→1 (1 - 420 Byte) | 416 | 278 | 222 | 279 | 278 | 372 |
| | | Auftragsüberprüfung | 203 | 135 | 108 | 136 | 140 | 170 |
| | | Auftragsende, DONE = 1 | 206 | 137 | 109 | 138 | 135 | 166 |
| 35 | ALARM_8P | Bausteinbezogene Meldung mit Begleitwerten für 8 Signale erzeugen Auftragsaktivierung, SIG = 0→1 (1 - 420 Byte) | 620 - 883 | 407 - 599 | 326 - 479 | 385 - 597 | 410 - 525 | 519 - 634 |
| | | Auftragsüberprüfung | 233 | 154 | 123 | 137 | 152 | 182 |
| | | Auftragsende, DONE = 1 | 236 | 155 | 125 | 138 | 148 | 180 |
| 36 | NOTIFY | Bausteinbezogene Meldung ohne Quittierungsanzeige erzeugen Auftragsaktivierung, SIG = 0→1 (1 - 420 Byte) | 561 - 823 | 373 - 580 | 301 - 462 | 379 - 578 | 384 - 510 | 519 - 644 |
| | | Auftragsüberprüfung | 186 | 125 | 100 | 126 | 133 | 163 |
| | | Auftragsende, DONE = 1 | 191 | 128 | 102 | 129 | 130 | 160 |

| SFB-Nr. | SFB-Name | Bedeutung | Ausführungszeit in μ s | | | | | |
|---------|----------|---|----------------------------|---------|---------|---------|---------------------------|--------------------------------|
| | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 | CPUs 414-4H 417-4H (solo) | CPUs 414-4H 417-4H (redundant) |
| 37 | AR_SEND | Archivdaten senden Auftragsaktivierung, REQ = 1 (1 - 3000 Byte) | 388 | 258 | 208 | 258 | 265 | 328 |
| | | Auftragsüberprüfung | 173 | 116 | 92 | 116 | 123 | 155 |
| | | Auftragsende, DONE = 1 | 167 | 111 | 88 | 112 | 115 | 147 |
| 52 | RDREC | Datensatz aus einem DP-Slave lesen integrierte DP-Schnittstelle, Erstaufruf (2-16 Byte) | 341 | 221 | 177 | 221 | 228 | 269 |
| | | Zwischenaufruf | 173 | 111 | 89 | 111 | 117 | 114 |
| | | Letztaufruf | 236 | 157 | 127 | 157 | 164 | 161 |
| 52 | RDREC | Datensatz aus einem DP-Slave lesen externe DP-Schnittstelle, Erstaufruf (4-16 Byte) | 323 | 211 | 170 | 211 | 213 | 210 |
| | | Zwischenaufruf | 174 | 112 | 90 | 112 | 117 | 114 |
| | | Letztaufruf | 238 | 154 | 124 | 154 | 161 | 158 |

| SFB-Nr. | SFB-Name | Bedeutung | Ausführungszeit in μ s | | | | | |
|---------|----------|--|----------------------------|---------|---------|---------|---------------------------|--------------------------------|
| | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 | CPUs 414-4H 417-4H (solo) | CPUs 414-4H 417-4H (redundant) |
| 53 | WRREC | Datensatz in einen DP-Slave schreiben integrierte DP-Schnittstelle, Erstaufruf (1-10 Byte) | 354 | 234 | 187 | 234 | 241 | 281 |
| | | Zwischenaufruf | 170 | 110 | 88 | 110 | 116 | 112 |
| | | Letztaufruf | 171 | 110 | 89 | 110 | 116 | 113 |
| 53 | WRREC | Datensatz in einen DP-Slave schreiben externe DP-Schnittstelle, Erstaufruf (2-14 Byte) | 339 | 224 | 180 | 224 | 226 | 223 |
| | | Zwischenaufruf | 170 | 110 | 89 | 110 | 116 | 113 |
| | | Letztaufruf | 172 | 111 | 89 | 111 | 117 | 113 |
| 54 | RALRM | Alarm von einem DP-Slave empfangen Laufzeitmessung bei nicht peripheriegebundenen OBs, MODE = 1, OB 1 | 133 | 81 | 70 | 81 | 83 | 83 |
| 54 | RALRM | Alarm von einem DP-Slave empfangen Laufzeitmessung an integrierter DP-Schnittstelle, MODE = 1, OB 40, OB 83, OB 86 | 250 | 164 | 135 | 164 | 245 | 245 |
| | | OB 55 bis OB 57, OB 82 | 257 | 171 | 140 | 171 | 251 | 251 |
| | | OB 70 | -- | -- | -- | -- | 242 | 242 |

| SFB-Nr. | SFB-Name | Bedeutung | Ausführungszeit in μ s | | | | | |
|---------|----------|---|----------------------------|---------|---------|---------|---------------------------------|--------------------------------------|
| | | | CPU 412 | CPU 414 | CPU 416 | CPU 417 | CPUs 414-4H 417-4H (solo) | CPUs 414-4H 417-4H (redundant) |
| 54 | RALRM | Alarm von einem DP-Slave empfangen Laufzeitmessung an externer DP-Schnittstelle, MODE = 1, OB 40, OB 83, OB 86 | 429 | 290 | 234 | 290 | 458 | 458 |
| | | OB 55 bis OB 57, OB 82 | 704 | 499 | 413 | 499 | 747 | 747 |
| | | OB 70 | -- | -- | -- | -- | 460 | 460 |
| 54 | RALRM | Alarm von einem DP-Slave empfangen Laufzeitmessung an zentraler Peripherie, MODE = 1, OB 40, OB 82, OB 83, OB 86 | 215 | 138 | 111 | 138 | 143 | 143 |
| | | OB 55 bis OB 57 | 619 | 472 | 414 | 472 | 567 | 567 |

SZL-Teilliste

| SZL-ID | Auskunffunktionen |
|--------|---|
| | Baugruppen-Identifikation |
| 0111 | Ein Identifikationsdatensatz |
| | CPU-Merkmale |
| 0012 | Alle Merkmale |
| 0112 | Merkmale einer Gruppe |
| | MC7-Bearbeitungseinheit |
| | Zeitsystem |
| | Systemverhalten |
| | MC7-Sprachbeschreibung |
| | Verfügbarkeit von SFCs |
| 0F12 | Nur SZL-Teillistenkopfinformation |
| | Anwenderspeicherbereiche |
| 0113 | Ein Datensatz für den angegebenen Speicherbereich |
| | Arbeitsspeicher |
| 0F13 | Nur SZL-Teillistenkopfinformation |

SZL-Teilliste, Fortsetzung

| SZL-ID | Auskunftfunktionen |
|--------|--|
| | Systembereiche |
| 0014 | Datensätze aller Systembereiche |
| 0F14 | Nur SZL-Teillistenkopfinformation |
| | Bausteintypen |
| 0015 | Datensätze aller Bausteintypen |
| | Zustand der Baugruppen-LEDs |
| 0019 | Zustand aller LEDs |
| 0F19 | Nur SZL-Teillistenkopfinformation |
| | Identifikation einer Komponente |
| 001C | Identifikation aller Komponenten |
| 011C | Identifikation einer Komponente |
| 0F1C | Nur SZL-Teillistenkopfinformation |
| | Alarmstatus |
| 0222 | Datensatz zum angegebenen Alarm |
| | freier Zyklus |
| | Uhrzeitalarm |

SZL-Teilliste, Fortsetzung

| SZL-ID | Auskunftsfunktionen |
|--------|--|
| | Verzögerungsalarm |
| | Weckalarm |
| | Prozessalarm |
| | DP-Alarm |
| | Multicomputing- oder Taktsynchronalarm |
| | Redundanzalarm |
| | Asynchroner Fehleralarm |
| | Hintergrund |
| | Anlauf |
| | Synchroner Fehleralarm |
| | Zuordnung zwischen Teilprozeßabbildern und OBs |
| 0025 | Zuordnung zwischen allen Teilprozeßabbildern und OBs innerhalb der CPU |
| 0125 | Zuordnung zwischen einem Teilprozeßabbild und dem zugehörigen OB |
| 0225 | Zuordnung zwischen einem OB und den zugehörigen Teilprozeßabbildern |
| 0F25 | Nur SZL–Teillistenkopfinfo |

SZL-Teilliste, Fortsetzung

| SZL-ID | Auskunftfunktionen |
|--------|---|
| | Kommunikationszustandsdaten |
| 0132 | Zustandsdaten zu einem Kommunikationsteil |
| | Diagnose |
| | Zeitsystem |
| 0232 | Zustandsdaten zu einem Kommunikationsteil |
| | CPU-Schutzstufe und Bedienschalterstellungen |
| | H-CPU-Sammelinformation |
| 0071 | Informationen über den aktuellen Zustand des H-Systems |
| 0F71 | Nur SZL-Teillistenkopfinformation |
| | Zustand der Baugruppen-LEDs |
| 0174 | Zustand einer-LED |
| | Geschaltete DP-Slaves im H-System |
| 0C75 | Kommunikationszustand zwischen dem H-System und einem geschalteten DP-Slave |

SZL-Teilliste, Fortsetzung

| SZL-ID | Auskunfftfunktionen |
|--------|--|
| | DP-Mastersystem-Information |
| 0090 | Informationen über alle der CPU bekannten DP-Mastersysteme |
| 0190 | Informationen über ein DP-Mastersystem |
| 0F90 | Nur SZL-Teillistenkopfinformation |
| | Baugruppenzustandsinformation (Es werden maximal 27 Datensätze geliefert) |
| 0091 | Baugruppenzustandsinformation aller gesteckten Baugruppen/Submodule |
| 0191 | Zustandsinformation aller Baugruppen/Baugruppenträger mit falscher Typkennung |
| 0291 | Baugruppenzustandsinformation aller gestörten Baugruppen |
| 0391 | Baugruppenzustandsinformation aller nicht verfügbaren Baugruppen |
| 0591 | Baugruppenzustandsinformation aller Submodule der Hostbaugruppe |
| 0991 | Baugruppenzustandsinformation aller Submodule der Hostbaugruppe im angegebenen Baugruppenträger |
| 0C91 | Baugruppenzustandsinformation einer Baugruppe im zentralen Aufbau oder an einer integrierten DP-Anschaltung |
| 4C91 | Baugruppenzustandsinformation einer Baugruppe an einer externen DP-Anschaltung |
| 0D91 | Baugruppenzustandsinformation aller Baugruppen im angegebenen Baugruppenträger/in der angegebenen DP-Station |
| 0E91 | Baugruppenzustandsinformation aller zugeordneten Baugruppen |

SZL-Teilliste, Fortsetzung

| SZL-ID | Auskunffunktionen |
|--------|---|
| | Baugruppenträger-/Stationszustandsinformation |
| 0092 | Sollzustand der Baugruppenträger im zentralen Aufbau/der Stationen eines DP-Mastersystems, das über eine integrierte DP-Anschaltung angeschlossen ist. |
| 4092 | Sollzustand der Stationen eines DP-Mastersystems, das über eine externe DP-Anschaltung angeschlossen ist |
| 0192 | Aktivierungsstatus der Stationen eines DP-Mastersystems, das über eine integrierte DP-Anschaltung angeschlossen ist |
| 0292 | Istzustand der Baugruppenträger im zentralen Aufbau/der Stationen eines DP-Mastersystems, das über eine integrierte DP-Anschaltung angeschlossen ist. |
| 0392 | Zustand der Batteriepufferung eines Racks/Baugruppenträgers einer CPU, wenn mindestens eine Batterie ausgefallen ist |
| 0492 | Zustand der gesamten Batteriepufferung aller Racks/Baugruppenträger einer CPU |
| 0592 | Istzustand der Baugruppenträger im zentralen Aufbau/der Stationen eines DP-Mastersystems, das über eine integrierte DP-Anschaltung angeschlossen ist. |
| 4292 | Istzustand der Stationen eines DP-Mastersystems, das über eine externe DP-Anschaltung angeschlossen ist |
| 0692 | OK-Zustand der Erweiterungsgeräte im zentralen Aufbau/der Stationen eines DP-Mastersystems, das über eine integrierte DP-Anschaltung angeschlossen ist. |
| 4692 | OK-Zustand der Stationen eines DP-Mastersystems, das über eine externe DP-Anschaltung angeschlossen ist. |
| 0492 | Zustand der Gesamtpufferung der Baugruppenträger im zentralen Aufbau |

SZL-Teilliste, Fortsetzung

| SZL-ID | Auskunffunktionen |
|--------|---|
| | Erweiterte DP-Mastersystem-Information |
| 0195 | Erweiterte Informationen über ein DP-Mastersystem |
| 0F95 | Nur SZL-Teillistenkopfinformation |
| | Diagnosepuffer (Es werden maximal 21 Datensätze geliefert) |
| 00A0 | Alle im aktuellen Betriebszustand lieferbaren Einträge |
| 01A0 | Die neuesten Einträge, die Anzahl wird über dem Index angegeben |
| 0FA0 | Nur SZL-Teillistenkopfinformation |
| | Diagnosedaten auf Baugruppen |
| 00B1 | Die ersten vier Diagnosebytes einer Baugruppe (DS0) |
| 00B2 | Alle Diagnosedaten einer Baugruppe (≤ 220 byte, DS1) (keine DP-Baugruppe) |
| 00B3 | Alle Diagnosedaten einer Baugruppe (≤ 220 byte, DS1) |
| 00B4 | Diagnosedaten eines DP-Slaves |

Alphabetisches Verzeichnis der Operationen

| <i>Operation</i> | <i>Seite</i> | <i>Operation</i> | <i>Seite</i> |
|------------------|--------------|------------------|--------------|
|) | 29 | ==I | 72 |
|)MCR | 97 | ==R | 74 |
| + | 70 | <=D | 73 |
| +AR1 | 71 | <=I | 72 |
| +AR2 | 71 | <=R | 74 |
| +D | 63 | <D | 73 |
| +I | 61 | <I | 72 |
| +R | 65 | <R | 74 |
| -D | 63 | > | 73 |
| -I | 61 | > | 72 |
| -R | 65 | >=D | 73 |
| *D | 63 | >=I | 72 |
| *I | 62 | >=R | 74 |
| *R | 65 | >D | 73 |
| /D | 64 | >I | 72 |
| /I | 62 | >R | 74 |
| /R | 65 | ABS | 66 |
| = | 40 | ACOS | 69 |
| ==D | 73 | ASIN | 69 |

| <i>Operation</i> | <i>Seite</i> | <i>Operation</i> | <i>Seite</i> |
|------------------|--------------|------------------|----------------------------|
| ATAN | 69 | INVD | 85 |
| AUF | 88 | INVI | 85 |
| BE | 89 | ITB | 83 |
| BEA | 89 | ITD | 82 |
| BEB | 89 | L | 47 48 49 50 51 53 59 60 |
| BLD | 81 | LAR1 | 57 |
| BTD | 82 | LAR2 | 57 |
| BTI | 82 | LC | 53 |
| CALL | 86 | LEAVE | 79 |
| CC | 87 | LN | 68 |
| CLR | 41 | LOOP | 96 |
| COS | 69 | MCR(| 97 |
| DEC | 80 | MCRA | 98 |
| DTB | 83 | MCRD | 98 |
| DTR | 82 | MOD | 64 |
| ENT | 79 | NEGD | 85 |
| EXP | 68 | NEGI | 85 |
| FN | 38 | NEGR | 66 |
| FP | 38 | NOP | 81 |
| FR | 44 46 | | |
| INC | 80 | | |

| <i>Operation</i> | <i>Seite</i> | <i>Operation</i> | <i>Seite</i> |
|------------------|-------------------|------------------|--------------|
| NOT | 41 | SET | 41 |
| O | 26 30 32 35 36 37 | SI | 42 |
| O(| 28 | SIN | 69 |
| OD | 34 | SLD | 75 |
| ON | 26 32 35 36 37 | SLW | 75 |
| ON(| 28 | SPA | 91 |
| OW | 33 | SPB | 91 |
| POP | 79 | SPBB | 92 |
| PUSH | 79 | SPBI | 92 |
| R | 39 44 45 | SPBIN | 92 |
| RLD | 77 | SPBN | 91 |
| RLDA | 78 | SPBNB | 92 |
| RND | 84 | SPL | 96 |
| RND+ | 84 | SPM | 94 |
| RND- | 84 | SPMZ | 95 |
| RRD | 77 | SPN | 94 |
| RRDA | 78 | SPO | 93 |
| S | 39 45 | SPP | 94 |
| SA | 43 | SPPZ | 95 |
| SAVE | 41 | SPS | 93 |
| SE | 42 | | |

| <i>Operation</i> | <i>Seite</i> | <i>Operation</i> | <i>Seite</i> |
|------------------|--------------------|------------------|-----------------------|
| SPU | 94 | TAW | 79 |
| SPZ | 94 | TDB | 90 |
| SQR | 67 | TRUNC | 84 |
| SQRT | 67 | U | 25 31 35 36 37 |
| SRD | 76 | U(| 28 |
| SRW | 75 | UC | 87 |
| SS | 43 | UD | 34 |
| SSD | 76 | UN | 25 31 35 36 37 |
| SSI | 76 | UN(| 28 |
| SV | 42 | UW | 33 |
| T | 54 55 56 59 | X | 27 32 35 36 37 |
| TAD | 79 | X(| 28 |
| TAK | 79 | XN | 27 32 35 36 37 |
| TAN | 69 | XN(| 28 |
| TAR | 58 | XOD | 34 |
| TAR1 | 58 | XOW | 33 |
| TAR2 | 58 | ZR | 46 |
| | | ZV | 45 |

