

Tabellenheft CPU 945

Inhaltsverzeichnis

	Seite
Erläuterungen zur Operationsliste	1
Erläuterungen der Operanden	3
Grundoperationen	
Verknüpfungsoperationen	8
Speicheroperationen	8
Ladeoperationen	10
Transferoperationen	16
Zeitoperationen	22
Zähloperationen	22
Vergleichsoperationen	24
Arithmetische Operationen	28
Bausteinaufrufoperationen	30
Rücksprungoperationen	34
Null-Operationen	34
Stop-Operation	34
Bildaufbau-Operationen	34
Ergänzende Operationen	
Verknüpfungsoperationen	38
Bit-Testoperationen	40
Bit-Setzoperationen	42
Speicheroperationen	42
Zeit- und Zähloperationen	44
Lade- und Transferoperationen	46
Umwandlungsoperationen	46
Schiebe- und Rotieroperationen	48
Sprungoperationen	50
Alarmer sperren und freigeben	54
Dekrementieren/Inkrementieren	54
Sonstige Operationen	56
Systemoperationen	
Bit-Setzoperationen	58
Lade- und Transferoperationen	58
Arithmetische Operationen	64
Sonstige Operationen	66
Auflistung des Maschinencodes	68
Alphabetisches Verzeichnis der Operationen	80
Liste der Organisationsbausteine	84
Liste der Funktionsbausteine	92
Datenbaustein 1	96
Auswertung von ANZ 1 und ANZ 0	110

Erläuterungen zur Operationsliste

Abkürzungen	Erklärungen
AKKU 1	Akkumulator 1, 32 Bit breit (Beim Laden des AKKU 1 wird der ursprüngliche Inhalt in den AKKU 2 geschoben)*
AKKU 2	Akkumulator 2, 32 Bit breit
AKKU x-L	Low-Wort (Bits 0 - 15) des AKKU x
AKKU x-H	High-Wort (Bits 16 - 31) des AKKU x
ANZ 0/ANZ 1	Ergebnisanzeige 0/Ergebnisanzeige 1
AWL	STEP 5-Darstellungsart Anweisungsliste
Formal-operand	Ausdruck mit max. 4 Zeichen, wobei das erste Zeichen ein Buchstabe sein muß.
FUP	STEP 5-Darstellungsart Funktionsplan
KOP	STEP 5-Darstellungsart Kontaktplan
OS	Überlauf-Anzeige (Overflow) speichernd; wird gesetzt, wenn mindestens eine arithmetische Operation zu einem Überlauf führt (dient zur Erkennung von Arithmetik-Fehlern)
OV	Überlauf-Anzeige (Overflow). Diese Anzeige wird gesetzt, wenn z.B. bei arithmetischen Operationen der Zahlenbereich überschritten wird.
PAE	Prozeßabbild der Eingänge
PAA	Prozeßabbild der Ausgänge
VKE	Verknüpfungsergebnis
VKE abhängig	<p>J Die Anweisung wird nur ausgeführt, wenn das VKE = "1" ist.</p> <p>J ↑ / ↓ Die Anweisung wird nur ausgeführt, wenn positiver/negativer Flankenwechsel beim VKE vorliegt.</p> <p>N Die Anweisung wird immer ausgeführt.</p>
VKE beeinflussend	J/N Das VKE wird durch die Operation beeinflusst/nicht beeinflusst.

* nicht bei LIR, LDI, LIM, MSA, MBA

Erläuterungen zur Operationsliste

Abkürzungen	Erklärungen
VKE begren- zend	J Das VKE wird nicht verändert. Eine Weiterverknüpfung ist nicht mehr möglich. Bei der nächsten Binärverknüpfung (aber nicht Zuweisung) wird das VKE neu aufgebaut. N Abhängig davon, ob die Operation VKE beeinflussend ist oder nicht, wird das VKE entsprechend der Operation und des Status des abgefragten Bits weiter verknüpft oder unverändert belassen.

Erläuterung der Operanden

Abk.	Erläuterung	Zulässiger Wertebereich für Operanden	Größe in Bits
A	Ausgang	0.0 bis 127.7	1
AB	Ausgangsbyte	0 bis 127	8
AD	Ausgangsdoppelwort	0 bis 124	32
AW	Ausgangswort	0 bis 126	16
BF	Byte-Konstante (Festpunktzahl)	- 128 bis + 127	8
BS	Systemdatenbereich Systemdatenwort - bei Ladeoperationen und Transferoperationen - bei Bit-Test- und Setzoperationen	0 bis 255	16
		0.0 bis 255.15	1
BT	Erweiterter Systemdatenbereich Systemdatenwort - bei Ladeoperationen und Transferoperationen - bei Bit-Test- und Setzoperationen	0 bis 255	16
		0.0 bis 255.15	1
D	Datenbit	0.0 bis 255.15	1
DB	Datenbaustein	0 bis 255	-
DD	Datendoppelwort	0 bis 254	32
DH	Konstante (Doppelwort als Hexadezimalzahl)	0 bis FFFF FFFF	32
DL	Datenwort (linkes Byte)	0 bis 255	8
DR	Datenwort (rechtes Byte)	0 bis 255	8

Abk.	Erklärung	Zulässiger Wertebereich für Operanden	Größe in Bits
DW	Datenwort	0 bis 255	16
DX	Datenbaustein (Erweiterung)	0 bis 255	-
E	Eingang	0.0 bis 127.7	1
EB	Eingangsbyte	0 bis 127	8
ED	Eingangsdoppelwort	0 bis 124	32
EW	Eingangswort	0 bis 126	16
FB	Funktionsbaustein	0 bis 255	-
FX	Funktionsbaustein (Erweiterung)	0 bis 255	-
KB	Konstante (1 Byte)	0 bis 255	8
KC	Konstante (2 Character-Zeichen)	zwei beliebige alphanumerische Zeichen	16
KF	Konstante (Festpunktzahl)	- 32768 bis + 32767	16
KG	Konstante (Gleitpunktzahl)	$\pm 0,1469368 \times 10^{-38}$ bis $\pm 0,1701412 \times 10^{39}$ und $0,0 \times 10^0$	32
KH	Konstante (Hexa-Code)	0 bis FFFF	16
KM	Konstante (2 Byte Bitmuster)	beliebiges Bitmuster (16 Bit)	16

Abk.	Erklärung	Zulässiger Wertebereich für Operanden	Größe in Bits
KT	Konstante (Zeitwert)	0.0 bis 999.3	16
KY	Konstante (2 Byte)	0 bis 255 (je Byte)	16
KZ	Konstante (Zählwert)	0 bis 999	16
M	Merker	0.0 bis 255.7	1
MB	Merkerbyte	0 bis 255	8
MD	Merkerdoppelwort	0 bis 252	32
MW	Merkerwort	0 bis 254	16
OB ¹	Organisationsbaustein	0 bis 255	-
PB	Programmbaustein (bei Bausteinaufruf- und Rücksprungoperationen)	0 bis 255	-
PY	Peripheriebyte - Digital-Eingaben - Analog-Eingaben - Digital-Ausgaben - Analog-Ausgaben	0 bis 127 128 bis 255 0 bis 127 128 bis 255	8
PW	Peripheriewort - Digital-Eingaben - Analog-Eingaben - Digital-Ausgaben - Analog-Ausgaben	0 bis 126 128 bis 254 0 bis 126 128 bis 254	16
QB	Byte der erweiterten Peripherie	0 bis 255	8

1 Eine Übersicht über die Organisationsbausteine und ihre Funktion → Seite 84

Abk.	Erklärung	Zulässiger Wertebereich für Operanden	Größe in Bits
QW	Wort der erweiterten Peripherie	0 bis 254	16
S	Merker (Erweiterung)	0.0 bis 4095.7	1
SB	Schrittbaustein	0 bis 255	-
SD	Merkerdoppelwort (Erweiterung)	0 bis 4092	32
SW	Merkerwort (Erweiterung)	0 bis 4094	16
SY	Merkerbyte (Erweiterung)	0 bis 4095	8
T	Zeit - bei Bit-Test- und Setzoperationen (Systemoperationen)	0 bis 127 0.0 bis 127.15	-
Z	Zähler - bei Bit-Test- und Setzoperationen (Systemoperationen)	0 bis 127 0.0 bis 127.15	-

Grundoperationen

für Organisationsbausteine (OB)

für Funktionsbausteine (FB)

für Programmbausteine (PB)

für Schrittbauusteine (SB)

Operation (AWL)	Zulässige Operanden	1 VKE abhängig. 2 VKE beeinfl. 3 VKE begrenzt.			typische Ausführungszeit in μ s		Funktionsbeschreibung
		1	2	3	CPU 945		
Verknüpfungsoperationen							
U	E, A, M, T, Z, S, D	N	J	N	0,1 0,35		UND-Verknüpfung: Abfrage auf Signalzustand "1"
UN	E, A, M, T, Z, S, D	N	J	N	0,1 0,35		UND-Verknüpfung: Abfrage auf Signalzustand "0"
O	E, A, M, T, Z, S, D	N	J	N	0,1 0,35		ODER-Verknüpfung: Abfrage auf Signalzustand "1"
ON	E, A, M, T, Z, S, D	N	J	N	0,1 0,35		ODER-Verknüpfung: Abfrage auf Signalzustand "0"
O		N	J	J	0,1		ODER-Verknüpfung von UND-Funktionen
U(N	J	J	0,1		UND-Verknüpfung von Klammerausdrücken (8 Klammerebenen)
O(N	J	J	0,1		ODER-Verknüpfung von Klammerausdrücken (8 Klammerebenen)
)		N	J	N	0,1		Klammer zu (Abschluß eines Klammerausdrucks)
Speicheroperationen							
S	E, A, M, S, D	J	N	J	0,1 0,45		Den Operanden auf den Wert "1" setzen
R	E, A, M, S, D	J	N	J	0,1 0,45		Den Operanden auf den Wert "0" rücksetzen

Grundoperationen

für Organisationsbausteine (OB)

für Funktionsbausteine (FB)

für Programmbausteine (PB)

für Schrittbasteine (SB)

Operation (AWL)	Zulässige Operanden	1 VKE abhängig. 2 VKE beeinfl. 3 VKE begrenzt.			typische Ausführungszeit in μ s		Funktionsbeschreibung
		1	2	3	CPU 945		
Speicherooperationen (Fortsetzung)							
=	E, A, M, S, D	J	N	J	0,1 0,45		Dem Operanden wird der Wert des VKE zugewiesen
Ladeoperationen							
L	EB	N	N	N	0,1		Ein Eingangsbyte vom PAE in den AKKU 1 laden Byte n \rightarrow Bits 0 - 7
L	EW	N	N	N	0,1		Ein Eingangswort vom PAE in den AKKU 1 laden: Byte n \rightarrow Bits 8 - 15; Byte n + 1 \rightarrow Bits 0 - 7
L	ED	N	N	N	0,2		Ein Eingangsdoppelwort in den AKKU 1 laden: Byte n \rightarrow Bits 24 - 31; Byte n + 1 \rightarrow Bits 16 - 23; Byte n + 2 \rightarrow Bits 8 - 15; Byte n + 3 \rightarrow Bits 0 - 7;
L	AB	N	N	N	0,1		Ein Ausgangsbyte vom PAA in den AKKU 1 laden Byte n \rightarrow Bits 0 - 7
L	AW	N	N	N	0,1		Ein Ausgangswort vom PAA in den AKKU 1 laden: Byte n \rightarrow Bits 8 - 15; Byte n + 1 \rightarrow Bits 0 - 7
L	AD	N	N	N	0,2		Ein Ausgangsdoppelwort in den AKKU 1 laden: Byte n \rightarrow Bits 24 - 31; Byte n + 1 \rightarrow Bits 16 - 23; Byte n + 2 \rightarrow Bits 8 - 15; Byte n + 3 \rightarrow Bits 0 - 7;
L	PY	N	N	N	0,2*		Ein Peripheriebyte der Digital-/Analog-Eingaben in den AKKU 1 laden Byte n \rightarrow Bits 0 - 7

* + Readyverzugszeit der angesprochenen Peripheriebaugruppen

Grundoperationen

für Organisationsbausteine (OB)

für Funktionsbausteine (FB)

für Programmbausteine (PB)

für Schrittbausteine (SB)

Operation (AWL)	Zulässige Operanden	1 VKE abhängig. 2 VKE beeinfl. 3 VKE begrenzt.			typische Ausführungszeit in μ s		Funktionsbeschreibung
		1	2	3	CPU 945		
Ladeoperationen (Fortsetzung)							
L	PW	N	N	N	0,2**		Ein Peripheriewort der Digital-/Analog-Eingaben in den AKKU 1 laden: Byte n \rightarrow Bits 8 - 15; Byte n + 1 \rightarrow Bits 0 - 7
L	QB	N	N	N	0,2*		Ein Byte der erweiterten Peripherie in den AKKU 1 laden: Byte n \rightarrow Bits 0 - 7
L	QW	N	N	N	0,2**		Ein Wort der erweiterten Peripherie in den AKKU 1 laden: Byte n \rightarrow Bits 8 - 15; Byte n + 1 \rightarrow Bits 0 - 7
L	MB	N	N	N	0,1		Ein Merkerbyte in den AKKU 1 laden Byte n \rightarrow Bits 0 - 7
L	MW	N	N	N	0,1		Ein Merkerwort in den AKKU 1 laden: Byte n \rightarrow Bits 8 - 15; Byte n + 1 \rightarrow Bits 0 - 7
L	MD	N	N	N	0,2		Ein Merkerdoppelwort in den AKKU 1 laden: Byte n \rightarrow Bits 24 - 31; Byte n + 1 \rightarrow Bits 16 - 23; Byte n + 2 \rightarrow Bits 8 - 15; Byte n + 3 \rightarrow Bits 0 - 7
L	SY	N	N	N	0,35		Ein Merkerbyte (Erweiterung) in den AKKU 1 laden Byte n \rightarrow Bits 0 - 7
L	SW	N	N	N	0,35		Ein Merkerwort (Erweiterung) in den AKKU 1 laden: Byte n \rightarrow Bits 8 - 15; Byte n + 1 \rightarrow Bits 0 - 7
L	SD	N	N	N	0,45		Ein Merkerdoppelwort (Erweiterung) in den AKKU 1 laden: Byte n \rightarrow Bits 24 - 31; Byte n + 1 \rightarrow Bits 16 - 23; Byte n + 2 \rightarrow Bits 8 - 15; Byte n + 3 \rightarrow Bits 0 - 7

* + Readyverzugszeit der angesprochenen Peripheriebaugruppen

** + 2 \times Readyverzugszeit der angesprochenen Peripheriebaugruppen

Grundoperationen

für Organisationsbausteine (OB)

für Funktionsbausteine (FB)

für Programmbausteine (PB)

für Schrittbausteine (SB)

Operation (AWL)	Zulässige Operanden	1 VKE abhängig. 2 VKE beeinfl. 3 VKE begrenzt.			typische Ausführungszeit in μ s		Funktionsbeschreibung
		1	2	3	CPU 945		
Ladeoperationen (Fortsetzung)							
L	DL	N	N	N	0,2		Ein Datenwort (linkes Byte) des aktuellen Datenbausteins (DB/DX) in den AKKU 1 laden Byte n \rightarrow Bits 0 - 7
L	DR	N	N	N	0,2		Ein Datenwort (rechtes Byte) des aktuellen Datenbausteins (DB/DX) in den AKKU 1 laden Byte n \rightarrow Bits 0 - 7
L	DW	N	N	N	0,2		Ein Datenwort des aktuellen Datenbausteins (DB/DX) in den AKKU 1 laden: Byte n \rightarrow Bits 8 - 15; Byte n + 1 \rightarrow Bits 0 - 7
L	DD	N	N	N	0,3		Ein Datendoppelwort des aktuellen Datenbausteins in den AKKU 1 laden: Byte n \rightarrow Bits 24 - 31; Byte n + 1 \rightarrow Bits 16 - 23; Byte n + 2 \rightarrow Bits 8 - 15; Byte n + 3 \rightarrow Bits 0 - 7
L	KB	N	N	N	0,1		Ein Konstante (1-Byte-Zahl) in den AKKU 1 laden
L	KC	N	N	N	0,2		Ein Konstante (2 Character-Zeichen im ASCII-Format) in den AKKU 1 laden
L	KF	N	N	N	0,2		Eine Konstante (Festpunktzahl) in den AKKU 1 laden
L	KH	N	N	N	0,2		Eine Konstante (Hexa-Code) in den AKKU 1 laden
L	DH	N	N	N	0,3		Eine Konstante (Doppelwort im Hexa-Code) in den AKKU 1 laden
L	KG	N	N	N	0,3		Eine Konstante (Gleitpunktzahl) in den AKKU 1 laden
L	KM	N	N	N	0,2		Eine Konstante (Bitmuster) in den AKKU 1 laden

Grundoperationen

für Organisationsbausteine (OB)

für Funktionsbausteine (FB)

für Programmbausteine (PB)

für Schrittbauusteine (SB)

Operation (AWL)	Zulässige Operanden	1 VKE abhängig. 2 VKE beeinfl. 3 VKE begrenzt.			typische Ausführungszeit in μ s		Funktionsbeschreibung
		1	2	3	CPU 945		
Ladeoperationen (Fortsetzung)							
L	KY	N	N	N	0,2		Eine Konstante (2-Byte-Zahl) in den AKKU 1 laden
L	KT	N	N	N	0,2		Eine Konstante (Zeitwert) in den AKKU 1 laden (BCD-codiert)
L	KZ	N	N	N	0,2		Eine Konstante (Zählwert) in den AKKU 1 laden (BCD-codiert)
L	T, Z	N	N	N	0,1		Einen Zeit- oder Zählwert (dual-codiert) in den AKKU 1 laden
LC	T, Z	N	N	N	0,15		Zeit- oder Zählwerte (BCD-codiert) in den AKKU 1 laden
Transferoperationen							
T	EB	N	N	N	0,1		Den Inhalt des AKKU 1-L (Bits 0-7) zu einem Eingangsbyte transferieren (ins PAE)
T	EW	N	N	N	0,1		Den Inhalt des AKKU 1-L zu einem Eingangswort transferieren (ins PAE): Bits 8 - 15 \rightarrow Byte n; Bits 0 - 7 \rightarrow Byte n + 1
T	ED	N	N	N	0,2		Den Inhalt des AKKU 1 zu einem Eingangsdoppelwort transferieren (ins PAE): Bits 24 - 31 \rightarrow Byte n; Bits 16 - 23 \rightarrow Byte n + 1; Bits 8 - 15 \rightarrow Byte n + 2; Bits 0 - 7 \rightarrow Byte n + 3
T	AB	N	N	N	0,1		Den Inhalt des AKKU 1-L (Bits 0 - 7) zu einem Ausgangsbyte transferieren (ins PAA)
T	AW	N	N	N	0,1		Den Inhalt des AKKU 1-L zu einem Ausgangswort transferieren (ins PAA): Bits 8 - 15 \rightarrow Byte n; Bits 0 - 7 \rightarrow Byte n + 1

Grundoperationen

für Organisationsbausteine (OB)

für Funktionsbausteine (FB)

für Programmbausteine (PB)

für Schrittbauusteine (SB)

Operation (AWL)	Zulässige Operanden	1 VKE abhängig. 2 VKE beeinfl. 3 VKE begrenzt.			typische Ausführungszeit in μ s		Funktionsbeschreibung
		1	2	3	CPU 945		
Transferoperationen (Fortsetzung)							
T	AD	N	N	N	0,2		Den Inhalt des AKKU 1 zu einem Ausgangsdoppelwort transferieren (ins PAA): Bits 24 - 31 \rightarrow Byte n; Bits 16 - 23 \rightarrow Byte n + 1; Bits 8 - 15 \rightarrow Byte n + 2; Bits 0 - 7 \rightarrow Byte n + 3
T	PY	N	N	N	0,2*		Den Inhalt des AKKU 1-L (Bits 0 - 7) zu einem Peripheriebyte der Digital-Ausgaben mit Nachführen des PAA oder der Analog-Ausgaben transferieren.
T	PW	N	N	N	0,2**		Den Inhalt des AKKU 1-L zu einem Peripheriewort der Digital-Ausgaben mit Nachführen des PAA oder der Analog-Ausgaben transferieren: Bits 8 - 15 \rightarrow Byte n; Bits 0 - 7 \rightarrow Byte n + 1
T	QB	N	N	N	0,2*		Inhalt des AKKU 1-L (Bits 0 - 7) zu einem Byte der erweiterten Peripherie der Digital- oder Analog-Ausgaben transferieren
T	QW	N	N	N	0,2**		Inhalt des AKKU 1-L zu einem Wort der erweiterten Peripherie der Digital- oder Analog-Ausgaben transferieren: Bits 8-15 \rightarrow Byte n; Bits 0-7 \rightarrow Byte n + 1
T	MB	N	N	N	0,1		Den Inhalt des AKKU 1-L (Bits 0 ... 7) zu einem Merkerbyte transferieren
T	MW	N	N	N	0,1		Den Inhalt des AKKU 1-L zu einem Merkerwort transferieren: Bits 8 - 15 \rightarrow Byte n; Bits 0 - 7 \rightarrow Byte n + 1

* + Readyverzugszeit der angesprochenen Peripheriebaugruppen

** + 2 \times Readyverzugszeit der angesprochenen Peripheriebaugruppen

Grundoperationen

für Organisationsbausteine (OB)

für Funktionsbausteine (FB)

für Programmbausteine (PB)

für Schrittbauusteine (SB)

Operation (AWL)	Zulässige Operanden	1 VKE abhängig. 2 VKE beeinfl. 3 VKE begrenzt.			typische Ausführungszeit in μ s		Funktionsbeschreibung
		1	2	3	CPU 945		
Transferoperationen (Fortsetzung)							
T	MD	N	N	N	0,2		Den Inhalt des AKKU 1 zu einem Merkerdoppelwort transferieren: Bits 24 - 31 \rightarrow Byte n; Bits 16 - 23 \rightarrow Byte n + 1; Bits 8 - 15 \rightarrow Byte n + 2; Bits 0 - 7 \rightarrow Byte n + 3
T	SY	N	N	N	0,35		Den Inhalt des AKKU 1-L (Bits 0 - 7) zu einem Merkerbyte (Erweiterung) transferieren
T	SW	N	N	N	0,35		Den Inhalt des AKKU 1-L zu einem Merkerwort (Erweiterung) transferieren: Bits 8 - 15 \rightarrow Byte n; Bits 0 - 7 \rightarrow Byte n + 1
T	SD	N	N	N	0,45		Den Inhalt des AKKU 1 zu einem Merkerdoppelwort (Erweiterung) transferieren: Bits 24 - 31 \rightarrow Byte n; Bits 16 - 23 \rightarrow Byte n + 1; Bits 8 - 15 \rightarrow Byte n + 2; Bits 0 - 7 \rightarrow Byte n + 3
T	DL	N	N	N	0,2		Den Inhalt des AKKU 1-L (Bits 0 - 7) zu einem Datenwort (linkes Byte) des aktuellen Datenbausteins (DB/DX) transferieren
T	DR	N	N	N	0,2		Den Inhalt des AKKU 1-L (Bits 0 - 7) zu einem Datenwort (rechtes Byte) des aktuellen Datenbausteins (DB/DX) transferieren
T	DW	N	N	N	0,2		Den Inhalt des AKKU 1-L zu einem Datenwort des aktuellen Datenbausteins (DB/DX) transferieren
T	DD	N	N	N	0,3		Den Inhalt des AKKU 1 zu einem Datendoppelwort des aktuellen Datenbausteins (DB/DX) transferieren: Bits 24 - 31 \rightarrow Byte n; Bits 16 - 23 \rightarrow Byte n + 1; Bits 8 - 15 \rightarrow Byte n + 2; Bits 0 - 7 \rightarrow Byte n + 3

Grundoperationen

für Organisationsbausteine (OB)

für Funktionsbausteine (FB)

für Programmbausteine (PB)

für Schrittbausteine (SB)

Operation (AWL)	Zulässige Operanden	1 VKE abhängig. 2 VKE beeinfl. 3 VKE begrenzt.			typische Ausführungs- zeit in μ s		Funktionsbeschreibung
		1	2	3	CPU 945		
Zeitoperationen							
SI	T	J \uparrow	N	J	0,1		Eine Zeit (im AKKU 1-L hinterlegt) als Impuls starten (Signalbegrenzung)
SV	T	J \uparrow	N	J	0,1		Eine Zeit (im AKKU 1-L hinterlegt) als verlängerten Impuls starten (Signalbegrenzung und -verlängerung)
SE	T	J \uparrow	N	J	0,1		Eine Zeit (im AKKU 1-L hinterlegt) einschaltverzögernd starten
SS	T	J \uparrow	N	J	0,1		Eine Zeit (im AKKU 1-L hinterlegt) speichernd einschaltverzögernd starten
SA	T	J \downarrow	N	J	0,1		Eine Zeit (im AKKU 1-L hinterlegt) ausschaltverzögernd starten
R	T	J	N	J	0,1		Eine Zeit rücksetzen
Zähloperationen							
ZV	Z	J \uparrow	N	J	0,1		Zähler zählt um 1 vorwärts
ZR	Z	J \uparrow	N	J	0,1		Zähler zählt um 1 rückwärts
S	Z	J	N	J	0,1		Einen Zähler mit dem im AKKU 1-L hinterlegten Wert (BCD-Zahl von 0 bis 999) setzen
R	Z	J	N	J	0,1		Einen Zähler rücksetzen

Grundoperationen

für Organisationsbausteine (OB)

für Funktionsbausteine (FB)

für Programmbausteine (PB)

für Schrittbauusteine (SB)

Operation (AWL)	Zulässige Operanden	1 VKE abhängig. 2 VKE beeinfl. 3 VKE begrenzt.			typische Ausführungszeit in μ s		Funktionsbeschreibung
		1	2	3	CPU 945		
Vergleichsoperationen							
!=F		N	J	N	0,1		Vergleich zweier Festpunktzahlen auf gleich: Gilt AKKU 2-L=AKKU 1-L, dann wird das VKE="1". ANZ 1/ANZ 0 wird beeinflusst.
><F		N	J	N	0,1		Vergleich zweier Festpunktzahlen auf ungleich: Gilt AKKU 2-L \neq AKKU 1-L, dann wird das VKE="1". ANZ 1/ANZ 0 wird beeinflusst.
>F		N	J	N	0,1		Vergleich zweier Festpunktzahlen auf größer: Gilt AKKU 2-L > AKKU 1-L, dann wird das VKE="1". ANZ 1/ANZ 0 wird beeinflusst.
>=F		N	J	N	0,1		Vergleich zweier Festpunktzahlen auf größer oder gleich: Gilt AKKU 2-L \geq AKKU 1-L, dann wird das VKE="1". ANZ 1/ANZ 0 wird beeinflusst.
<F		N	J	N	0,1		Vergleich zweier Festpunktzahlen auf kleiner: Gilt AKKU 2-L < AKKU 1-L, dann wird das VKE="1". ANZ 1/ANZ 0 wird beeinflusst.
<=F		N	J	N	0,1		Vergleich zweier Festpunktzahlen auf kleiner oder gleich: Gilt AKKU 2-L \leq AKKU 1-L, dann wird das VKE="1". ANZ 1/ANZ 0 wird beeinflusst.
!=D		N	J	N	0,1		Vergleich zweier Festpunktdoppelwörter auf gleich: Gilt AKKU 2=AKKU 1, dann wird das VKE="1". ANZ 1/ANZ 0 wird beeinflusst.
><D		N	J	N	0,1		Vergleich zweier Festpunktdoppelwörter auf ungleich: Gilt AKKU 2 \neq AKKU 1, dann wird das VKE="1". ANZ 1/ANZ 0 wird beeinflusst.

Grundoperationen

für Organisationsbausteine (OB)

für Funktionsbausteine (FB)

für Programmbausteine (PB)

für Schrittbausteine (SB)

Operation (AWL)	Zulässige Operanden	1 VKE abhängig. 2 VKE beeinfl. 3 VKE begrenzt.			typische Ausführungszeit in μ s		Funktionsbeschreibung
		1	2	3	CPU 945		
Vergleichsoperationen (Fortsetzung)							
>D		N	J	N	0,1		Vergleich zweier Festpunktdoppelwörter auf größer: Gilt AKKU 2 > AKKU 1, dann wird das VKE = "1". ANZ 1/ANZ 0 wird beeinfl.ßt.
>=D		N	J	N	0,1		Vergleich zweier Festpunktdoppelwörter auf größer oder gleich: Gilt AKKU 2 \geq AKKU 1, dann wird das VKE = "1". ANZ 1/ANZ 0 wird beeinfl.ßt.
<D		N	J	N	0,1		Vergleich zweier Festpunktdoppelwörter auf kleiner: Gilt AKKU 2 < AKKU 1, dann wird das VKE = "1". ANZ 1/ANZ 0 wird beeinfl.ßt.
<=D		N	J	N	0,1		Vergleich zweier Festpunktdoppelwörter auf kleiner oder gleich: Gilt AKKU 2 \leq AKKU 1, dann wird das VKE = "1". ANZ 1/ANZ 0 wird beeinfl.ßt.
!=G		N	J	N	0,55		Vergleich zweier Gleitpunktzahlen auf gleich: Gilt AKKU 2 = AKKU 1, dann wird das VKE = "1". ANZ 1/ANZ 0 wird beeinfl.ßt.
><G		N	J	N	0,55		Vergleich zweier Gleitpunktzahlen auf ungleich: Gilt AKKU 2 \neq AKKU 1, dann wird das VKE = "1". ANZ 1/ANZ 0 wird beeinfl.ßt.
>G		N	J	N	0,55		Vergleich zweier Gleitpunktzahlen auf größer: Gilt AKKU 2 > AKKU 1, dann wird das VKE = "1". ANZ 1/ANZ 0 wird beeinfl.ßt.
>=G		N	J	N	0,55		Vergleich zweier Gleitpunktzahlen auf größer oder gleich: Gilt AKKU 2 \geq AKKU 1, dann wird das VKE = "1". ANZ 1/ANZ 0 wird beeinfl.ßt.

Grundoperationen

für Organisationsbausteine (OB)

für Funktionsbausteine (FB)

für Programmbausteine (PB)

für Schrittbausteine (SB)

Operation (AWL)	Zulässige Operanden	1 VKE abhängig. 2 VKE beeinfl. 3 VKE begrenzt.			typische Ausführungszeit in μ s		Funktionsbeschreibung
		1	2	3	CPU 945		
Vergleichsoperationen (Fortsetzung)							
<G		N	J	N	0,55		Vergleich zweier Gleitpunktzahlen auf kleiner: Gilt $AKKU\ 2 < AKKU\ 1$, dann wird das VKE = "1". ANZ 1/ANZ 0 wird beeinflusst.
<=G		N	J	N	0,55		Vergleich zweier Gleitpunktzahlen auf kleiner oder gleich: Gilt $AKKU\ 2 \leq AKKU\ 1$, dann wird das VKE = "1". ANZ 1/ANZ 0 wird beeinflusst.
Arithmetische Operationen							
+F		N	N	N	0,1		Zwei Festpunktzahlen addieren: $AKKU\ 1-L = (AKKU\ 2-L) + (AKKU\ 1-L)$. ANZ 1/ANZ 0/OV/OS auswertbar.
-F		N	N	N	0,1		Zwei Festpunktzahlen subtrahieren: $AKKU\ 1-L = (AKKU\ 2-L) - (AKKU\ 1-L)$. ANZ 1/ANZ 0/OV/OS auswertbar.
×F		N	N	N	0,35		Zwei Festpunktzahlen multiplizieren: $AKKU\ 1 = (AKKU\ 2-L) \times (AKKU\ 1-L)$. ANZ 1/ANZ 0/OV/OS auswertbar.
:F		N	N	N	0,4		Zwei Festpunktzahlen dividieren: $(AKKU\ 2-L) : (AKKU\ 1-L)$. In AKKU 1-L : Ergebnis; in AKKU 1-H : Rest. ANZ 1/ANZ 0/OV/OS auswertbar.
+D		N	N	N	0,1		Zwei Festpunktzahlen (32 Bit) addieren: $AKKU\ 1 = AKKU\ 2 + AKKU\ 1$. ANZ 1/ANZ 0/OV/OS auswertbar.

Grundoperationen

für Organisationsbausteine (OB)

für Funktionsbausteine (FB)

für Programmbausteine (PB)

für Schrittbausteine (SB)

Operation (AWL)	Zulässige Operanden	1 VKE abhängig. 2 VKE beeinfl. 3 VKE begrenzt.			typische Ausführungszeit in μ s		Funktionsbeschreibung
		1	2	3	CPU 945		
Arithmetische Operationen (Fortsetzung)							
- D		N	N	N	0,1		Zwei Festpunktzahlen (32 Bit) subtrahieren: AKKU 1 = AKKU 2 - AKKU 1. ANZ 1/ANZ 0/OV/OS auswertbar.
+ G		N	N	N	0,75		Zwei Gleitpunktzahlen addieren: AKKU 1 = AKKU 2 + AKKU 1. ANZ 1/ANZ 0/OV/OS auswertbar.
- G		N	N	N	0,75		Zwei Gleitpunktzahlen subtrahieren: AKKU 1 = AKKU 2 - AKKU 1. ANZ 1/ANZ 0/OV/OS auswertbar.
\times G		N	N	N	0,75		Zwei Gleitpunktzahlen multiplizieren: AKKU 1 = AKKU 2 \times AKKU 1. ANZ 1/ANZ 0/OV/OS auswertbar.
: G		N	N	N	1,35		Zwei Gleitpunktzahlen dividieren: AKKU 1 = AKKU 2 : AKKU 1. ANZ 1/ANZ 0/OV/OS auswertbar.
Bausteinaufrufoperationen							
SPA	OB	N	N	J	1		Absolut (unbedingt) zu einem Organisationsbaustein springen.
SPA	PB	N	N	J	1		Absolut (unbedingt) zu einem Programmbaustein springen.
SPA	FB	N	N	J	1,5		Absolut (unbedingt) zu einem Funktionsbaustein springen.
BA	FX	N	N	J	25,1		Absolut (unbedingt) zu einem Funktionsbaustein (FX) springen.

Grundoperationen

für Organisationsbausteine (OB)

für Funktionsbausteine (FB)

für Programmbausteine (PB)

für Schrittbausteine (SB)

Operation (AWL)	Zulässige Operanden	1 VKE abhängig. 2 VKE beeinfl. 3 VKE begrenzt.			typische Ausführungszeit in µs		Funktionsbeschreibung
		1	2	3	CPU 945		
Bausteinaufrufoperationen (Fortsetzung)							
SPA	SB	N	N	J	1		Absolut (unbedingt) zu einem Schrittbaustein springen.
SPB	OB	J	J ¹	J	1 0,1		Bedingt zu einem Organisationsbaustein springen. Zeit gilt für VKE = 1/VKE = 0
SPB	PB	J	J ¹	J	1 0,1		Bedingt zu einem Programmbaustein springen. Zeit gilt für VKE = 1/VKE = 0
SPB	FB	J	J ¹	J	1,5 0,35		Bedingt zu einem Funktionsbaustein springen. Zeit gilt für VKE = 1/VKE = 0
BAB	FX	J	J ¹	J	25,1 0,45		Bedingt zu einem Funktionsbaustein (FX) springen. Zeit gilt für VKE = 1/VKE = 0
SPB	SB	J	J ¹	J	1 0,1		Bedingt zu einem Schrittbaustein springen. Zeit gilt für VKE = 1/VKE = 0
A	DB	N	N	N	0,5		Einen Datenbaustein aufrufen
AX	DX	N	N	N	0,6		Einen Datenbaustein (Erweiterung) aufrufen
E	DB	N	N	N	45 - 60		Einen Datenbaustein erzeugen. Die Anzahl seiner Datenwörter muß im AKKU 1 hinterlegt sein.
EX	DX	N	N	N	45 - 60		Einen Datenbaustein (Erweiterung) erzeugen. Die Anzahl seiner Datenwörter muß im AKKU 1 hinterlegt sein.

¹ VKE wird auf 1 gesetzt

Grundoperationen

für Organisationsbausteine (OB)

für Funktionsbausteine (FB)

für Programmbausteine (PB)

für Schrittbausteine (SB)

Operation (AWL)	Zulässige Operanden	1 VKE abhängig. 2 VKE beeinfl. 3 VKE begrenzt.			typische Ausführungszeit in μ s		Funktionsbeschreibung
		1	2	3	CPU 945		
Rücksprungoperationen							
BE		N	N	J	0,8		Baustein beenden (Abschließen eines Bausteines)
BEB		J	J ¹	J	0,8 0,1		Baustein bedingt beenden Zeit gilt für VKE = 1/VKE = 0
BEA		N	N	J	0,8		Baustein absolut (unbedingt) beenden
Null-Operationen							
NOP 0		N	N	N	0,1		Nulloperation
NOP 1		N	N	N	0,1		Nulloperation
Stop-Operation							
STP		N	N	N	32		Stop am Ende der Programmbearbeitung; aktuelle Programmbearbeitung wird zu Ende gebracht; das PAA wird ausgegeben; danach geht AG in STOP.
Bildaufbau-Operationen							
BLD 0 ... 255		N	N	N	0,1		Wird von der CPU wie eine Nulloperation behandelt.
BLD 130		N	N	N	0,1		Bildaufbau-Befehl für das Programmiergerät: Erzeugen einer Leerzeile durch Carriage Return

¹ VKE wird auf 1 gesetzt

Grundoperationen

für Organisationsbausteine (OB)

für Funktionsbausteine (FB)

für Programmbausteine (PB)

für Schrittbauusteine (SB)

Operation (AWL)	Zulässige Operanden	1 VKE abhäng. 2 VKE beeinfl. 3 VKE begrenz.			typische Ausführungs- zeit in μ s		Funktionsbeschreibung
		1	2	3	CPU 945		
Bildaufbau-Operationen (Fortsetzung)							
BLD 131		N	N	N	0,1		Bildaufbau-Befehl für das Programmiergerät: Umschalten auf Anweisungsliste (AWL)
BLD 132		N	N	N	0,1		Bildaufbau-Befehl für das Programmiergerät: Umschalten auf Funktionsplan (FUP)
BLD 133		N	N	N	0,1		Bildaufbau-Befehl für das Programmiergerät: Umschalten auf Kontaktplan (KOP)
BLD 255		N	N	N	0,1		Bildaufbau-Befehl für das Programmiergerät: Segment beenden

Ergänzende Operationen

für Organisationsbausteine (OB)

für Funktionsbausteine (FB)

für Programmbausteine (PB)

für Schrittbauusteine (SB)

Operation (AWL)	Zulässige Operanden	1 VKE abhängig. 2 VKE beeinfl. 3 VKE begrenzt.			typische Ausführungszeit in µs		Funktionsbeschreibung
		1	2	3	CPU 945		
Verknüpfungsoperationen							
U=	Formaloperand E, A, M, T, Z	N	J	N	0,5*		UND-Verknüpfung: Formaloperanden auf den Signalzustand "1" abfragen. (Parametertyp: BI)
UN=	Formaloperand E, A, M, T, Z	N	J	N	0,5*		UND-Verknüpfung: Formaloperanden auf den Signalzustand "0" abfragen. (Parametertyp: BI)
O=	Formaloperand E, A, M, T, Z	N	J	N	0,5*		ODER-Verknüpfung: Formaloperanden auf den Signalzustand "1" abfragen. (Parametertyp: BI)
ON=	Formaloperand E, A, M, T, Z	N	J	N	0,5*		ODER-Verknüpfung: Formaloperanden auf den Signalzustand "0" abfragen. (Parametertyp: BI)
UW		N	N	N	0,1		UND-Verknüpfung (wortweise): AKKU 2-L mit AKKU 1-L; Ergebnis in AKKU 1-L. Ergebnis über ANZ 1/ANZ 0 auswertbar AKKU 1-H bleibt unverändert.
OW		N	N	N	0,1		ODER-Verknüpfung (wortweise): AKKU 2-L mit AKKU 1-L; Ergebnis in AKKU 1-L. Ergebnis über ANZ 1/ANZ 0 auswertbar AKKU 1-H bleibt unverändert.
XOW		N	N	N	0,1		Exklusiv-ODER-Verknüpfung (wortweise): AKKU 2-L mit AKKU 1-L; Ergebnis in AKKU 1-L. Ergebnis über ANZ 1/ANZ 0 auswertbar AKKU 1-H bleibt unverändert.

* plus Bearbeitungszeit des substituierten Befehls

Ergänzende Operationen

für Organisationsbausteine (OB)

für Funktionsbausteine (FB)

für Programmbausteine (PB)

für Schrittbauusteine (SB)

Operation (AWL)	Zulässige Operanden	1 VKE abhängig. 2 VKE beeinfl. 3 VKE begrenzt.			typische Ausführungs- zeit in μ s		Funktionsbeschreibung
		1	2	3	CPU 945		
Bit-Testoperationen							
P	E, A, M	N	J	N	0,2		Bit eines Eingangs-, Ausgangs- oder Merkerbytes auf Signalzustand "1" prüfen
P	T, Z	N	J	N	0,2		Bit eines Zeit- bzw. Zählwortes auf Signalzustand "1" prüfen
P	D	N	J	N	0,35		Bit eines Datenwortes (in einem DB/DX) auf Signalzustand "1" prüfen
P	BS	N	J	N	0,35		Bit eines Datenwortes im Bereich der Systemdaten auf Signalzustand "1" prüfen
P	BT	N	J	N	0,35		Bit eines Datenwortes im erweiterten Bereich der Systemdaten auf Signalzustand "1" prüfen
PN	E, A, M	N	J	N	0,2		Bit eines Eingangs-, Ausgangs- oder Merkerbytes auf Signalzustand "0" prüfen
PN	T, Z	N	J	N	0,2		Bit eines Zeit- bzw. Zählwortes auf Signalzustand "0" prüfen
PN	D	N	J	N	0,35		Bit eines Datenwortes (in einem DB/DX) auf Signalzustand "0" prüfen
PN	BS	N	J	N	0,35		Bit eines Datenwortes im Bereich der Systemdaten auf Signalzustand "0" prüfen
PN	BT	N	J	N	0,35		Bit eines Datenwortes im erweiterten Bereich der Systemdaten auf Signalzustand "0" prüfen

Ergänzende Operationen

für Organisationsbausteine (OB)

für Funktionsbausteine (FB)

für Programmbausteine (PB)

für Schrittbausteine (SB)

Operation (AWL)	Zulässige Operanden	1 VKE abhängig. 2 VKE beeinfl. 3 VKE begrenzt.			typische Ausführungszeit in μs		Funktionsbeschreibung
		1	2	3	CPU 945		
Bit-Setzoperationen							
SU	E, A, M	N	N	J	0,2		Ein Eingangs-Bit (im PAE); ein Ausgangs-Bit (im PAA) bzw. ein Merkerbit unbedingt setzen
SU	T, Z	N	N	J	0,2		Bit eines Zeit- bzw. Zählwortes unbedingt setzen
SU	D	N	N	J	0,45		Bit eines Datenwortes (in einem DB/DX) unbedingt setzen
RU	E, A, M	N	N	J	0,2		Ein Eingangs-Bit (im PAE); ein Ausgangs-Bit (im PAA) bzw. ein Merkerbit unbedingt rücksetzen
RU	T, Z	N	N	J	0,2		Bit eines Zeit- bzw. Zählwortes unbedingt rücksetzen
RU	D	N	N	J	0,45		Bit eines Datenwortes (in einem DB/DX) unbedingt rücksetzen
Speicheroperationen							
S=	Formaloperand E, A, M	J	N	J	0,5*		Einen Formaloperanden setzen, (bei VKE = 1) (Parametertyp: BI)
RB=	Formaloperand E, A, M	J	N	J	0,5*		Einen Formaloperanden rücksetzen, (bei VKE = 1) (Parametertyp: BI)
==	Formaloperand E, A, M	J	N	J	0,5*		Dem Status des Formaloperanden wird der Wert des VKE zugewiesen (Parametertyp: BI)

* plus Bearbeitungszeit des substituierten Befehls

Ergänzende Operationen

für Organisationsbausteine (OB)

für Funktionsbausteine (FB)

für Programmbausteine (PB)

für Schrittbausteine (SB)

Operation (AWL)	Zulässige Operanden	1 VKE abhängig. 2 VKE beeinfl. 3 VKE begrenzt.			typische Ausführungszeit in μ s		Funktionsbeschreibung
		1	2	3	CPU 945		
Zeit- und Zähloperationen							
FR	T, Z	J \uparrow	N	J	0,1		Zeit/Zähler für den Neustart freigeben. Wenn VKE = 1 anliegt, wird bei - 'FR T' die Zeit neu gestartet - 'FR Z' der Zähler gesetzt, vor- oder rückwärtsgezählt
FR=	Formaloperand T, Z	J \uparrow	N	J	0,5*		Formaloperand (Zeit/Zähler) für den Neustart freigeben. (Weitere Beschreibung s. Operation "FR")
RD=	Formaloperand T, Z	J	N	J	0,5*		Einen Formaloperanden rücksetzen (digital), (bei VKE = 1)
SI=	Formaloperand T	J \uparrow	N	J	0,5*		Eine Zeit (Formaloperand) als Impuls starten. Wert ist im AKKU 1 hinterlegt
SE=	Formaloperand T	J \uparrow	N	J	0,5*		Eine Zeit (Formaloperand) einschaltverzögernd starten. Wert ist im AKKU 1 hinterlegt
SVZ=	Formaloperand T, Z	J \uparrow	N	J	0,5*		Eine Zeit (Formaloperand) als verlängerten Impuls starten mit dem im AKKU 1 hinterlegten Wert bzw. einen Zähler (Formaloperand) setzen mit dem nachfolgenden angegebenen Zählwert
SSV=	Formaloperand T, Z	J \uparrow	N	J	0,5*		Eine Zeit (Formaloperand) als speichernde Einschaltverzögerung starten mit dem im AKKU 1 hinterlegten Wert bzw. Vorwärtszählen eines Zählers (Formaloperand)
SAR=	Formaloperand T, Z	J \downarrow J \uparrow	N	J	0,5*		Eine Zeit (Formaloperand) als Ausschaltverzögerung (\downarrow) starten mit dem im AKKU 1 hinterlegten Wert bzw. Rückwärtszählen (\uparrow) eines Zählers (Formaloperand)

* plus Bearbeitungszeit des substituierten Befehls

Ergänzende Operationen

für Organisationsbausteine (OB)

für Funktionsbausteine (FB)

für Programmbausteine (PB)

für Schrittbauusteine (SB)

Operation (AWL)	Zulässige Operanden	1 VKE abhängig. 2 VKE beeinfl. 3 VKE begrenzt.			typische Ausführungszeit in μ s		Funktionsbeschreibung
		1	2	3	CPU 945		
Lade- und Transferoperationen							
L=	Formaloperand E, A, M, T, Z	N	N	N	0,5*		Den Wert des Formaloperanden in den AKKU 1 laden (Parametertyp: BY, W, D; Aktualoperanden: MD, MB, MW, EB, EW, ED, AB, AW, AD, PY, PW, DL, DR, DW, DD, BS, BT, Z, T)
LC=	Formaloperand T, Z	N	N	N	0,5*		Den Wert des Formaloperanden im BCD-Code in den AKKU 1 laden
LW=	Formaloperand	N	N	N	0,45		Das Bitmuster eines Formaloperanden in den AKKU 1 laden (Parameterart: D; Parametertyp: KF, KH, KM; KY, KC, KT, KZ)
LD=	Formaloperand	N	N	N	0,55		Das Bitmuster (32 Bit) eines Formaloperanden in den AKKU 1 laden (Parameterart D; Parametertyp: KG)
T=	Formaloperand E, A, M	N	N	N	0,5*		Inhalt des AKKU 1 zum Formaloperanden transferieren (Parametertyp: BY, W, D; Aktualoperanden: MD, MB, MW, EB, EW, ED, AB, AW, AD, PY, PW, DL, DR, DW, DD, BS, BT)
Umwandlungsoperationen							
KEW		N	N	N	0,1		Das 1er-Komplement von AKKU 1-L (Bits 0 - 15) bilden
KZW		N	N	N	0,1		Das 2er-Komplement von AKKU 1-L (Bits 0 - 15) bilden. ANZ 1/ANZ 0 und OV werden beeinflusst
KZD		N	N	N	0,1		Das 2er-Komplement von AKKU 1 (Bits 0 - 31) bilden. ANZ1/ANZ0 und OV werden beeinflusst
DEF		N	N	N	0,55		16-Bit-Festpunktzahl von BCD in DUAL umwandeln

* plus Bearbeitungszeit des substituierten Befehls

Ergänzende Operationen

für Organisationsbausteine (OB)

für Funktionsbausteine (FB)

für Programmbausteine (PB)

für Schrittbauusteine (SB)

Operation (AWL)	Zulässige Operanden	1 VKE abhängig. 2 VKE beeinfl. 3 VKE begrenzt.			typische Ausführungszeit in μ s		Funktionsbeschreibung
		1	2	3	CPU 945		
Umwandlungsoperationen (Fortsetzung)							
DUF		N	N	N	0,65		16-Bit-Festpunktzahl von DUAL in BCD umwandeln
DED		N	N	N	0,55		32-Bit-Festpunktzahl von BCD in DUAL umwandeln
DUD		N	N	N	0,80		32-Bit-Festpunktzahl von DUAL in BCD umwandeln
FDG		N	N	N	0,45		Eine 32-Bit-Festpunktzahl in eine Gleitpunktzahl umwandeln
GFD		N	N	N	0,45		Eine Gleitpunktzahl in eine 32-Bit-Festpunktzahl umwandeln
Schiebe- und Rotieroperationen							
SLW	Parameter $n=0 \dots 15$	N	N	N	0,1		Inhalt von AKKU 1-L (Wort) nach links schieben um den im Parameter angegebenen Wert. Freierwerdende Stellen werden mit Nullen aufgefüllt. ANZ 1/ANZ 0 wird beeinflusst
SRW	Parameter $n=0 \dots 15$	N	N	N	0,1		Inhalt von AKKU 1-L (Wort) nach rechts schieben um den im Parameter angegebenen Wert. Freierwerdende Stellen werden mit Nullen aufgefüllt. ANZ 1/ANZ 0 wird beeinflusst
SLD	Parameter $n=0 \dots 32$	N	N	N	0,1		Inhalt von AKKU 1 (Doppelwort) nach links schieben um den im Parameter angegebenen Wert. Freierwerdende Stellen werden mit Nullen aufgefüllt. ANZ 1/ANZ 0 wird beeinflusst

Ergänzende Operationen

für Organisationsbausteine (OB)

für Funktionsbausteine (FB)

für Programmbausteine (PB)

für Schrittbasteine (SB)

Operation (AWL)	Zulässige Operanden	1 VKE abhängig. 2 VKE beeinfl. 3 VKE begrenz.			typische Ausführungszeit in μ s		Funktionsbeschreibung
		1	2	3	CPU 945		
Schiebe- und Rotieroperationen (Fortsetzung)							
SVW	Parameter n=0 ... 15	N	N	N	0,1		Inhalt von AKKU 1-L (Wort) mit Vorzeichen nach rechts schieben um den im Parameter angegebenen Wert. Freierdende Stellen werden mit dem Vorzeichen (Bit 15) aufgefüllt. ANZ 1/ANZ 0 wird beeinflusst
SVD	Parameter n=0 ... 32	N	N	N	0,1		Inhalt von AKKU 1 (Doppelwort) mit Vorzeichen nach rechts schieben um den im Parameter angegebenen Wert. Freierdende Stellen werden mit dem Vorzeichen (Bit 31) aufgefüllt. ANZ 1/ANZ 0 wird beeinflusst
RLD	Parameter n=0 ... 32	N	N	N	0,1		Inhalt von AKKU 1 (Doppelwort) nach links rotieren. Freierdende Stellen werden mit den hinausgeschobenen Bits aufgefüllt. ANZ1/ANZ0 wird beeinflusst
RRD	Parameter n=0 ... 32	N	N	N	0,1		Inhalt von AKKU 1 (Doppelwort) nach rechts rotieren. Freierdende Stellen werden mit den hinausgeschobenen Bits aufgefüllt. ANZ1/ANZ0 wird beeinflusst
Sprungoperationen							
SPA=	Symboladresse max. 4 Zeichen	N	N	N	0,25		Absolut (unbedingt) zur Symboladresse springen
SPB=	Symboladresse max. 4 Zeichen	J	J ¹	J	0,25 0,1		Bedingter Sprung zur Symboladresse; wird nur bei VKE = 1 ausgeführt. (Ist VKE = "0", wird das VKE auf "1" gesetzt) Zeit gilt für VKE = 1/VKE = 0

1 VKE wird auf "1" gesetzt

Ergänzende Operationen

für Organisationsbausteine (OB)

für Funktionsbausteine (FB)

für Programmbausteine (PB)

für Schrittbausteine (SB)

Operation (AWL)	Zulässige Operanden	1 VKE abhängig. 2 VKE beeinfl. 3 VKE begrenzt.			typische Ausführungszeit in μ s		Funktionsbeschreibung
		1	2	3	CPU 945		
Sprungoperationen (Fortsetzung)							
SPZ=	Symboladresse max. 4 Zeichen	N	N	N	0,25 0,1		Sprung bei Ergebnis 0: wird nur ausgeführt, wenn ANZ 1=0 und ANZ 0=0. Das VKE wird nicht verändert. Zeit gilt für VKE=1/VKE=0
SPN=	Symboladresse max. 4 Zeichen	N	N	N	0,25 0,1		Sprung bei Ergebnis $\neq 0$ wird nur ausgeführt, falls ANZ 1 \neq ANZ 0. Das VKE wird nicht verändert. Zeit gilt für VKE=1/VKE=0
SPP=	Symboladresse max. 4 Zeichen	N	N	N	0,25 0,1		Sprung bei Ergebnis > 0 : wird nur ausgeführt, falls ANZ 1=1 und ANZ 0=0. Das VKE wird nicht verändert. Zeit gilt für VKE=1/VKE=0
SPM=	Symboladresse max. 4 Zeichen	N	N	N	0,25 0,1		Sprung bei bei Ergebnis < 0 : wird nur ausgeführt, falls ANZ 1=0 und ANZ 0=1. Das VKE wird nicht verändert. Zeit gilt für VKE=1/VKE=0
SPO=	Symboladresse max. 4 Zeichen	N	N	N	0,25 0,1		Sprung bei "Überlauf": wird nur ausgeführt, wenn Anzeige OV=1 gesetzt ist. Das VKE wird nicht verändert. Zeit gilt für VKE=1/VKE=0
SPS=	Symboladresse max. 4 Zeichen	N	N	N	0,25 0,1		Sprung bei "Überlauf speichernd" wird nur ausgeführt, wenn Anzeige OS=1 gesetzt ist. Das VKE wird nicht verändert. Zeit gilt für VKE=1/VKE=0
SPR		N	N	N	0,25		Die lineare Programmbearbeitung wird unterbrochen und an der Stelle fortgesetzt, die durch die Sprungdistanz festgelegt ist. (Sprungdistanz in Worten: -32768 ... +32767)

Ergänzende Operationen

für Organisationsbausteine (OB)

für Funktionsbausteine (FB)

für Programmbausteine (PB)

für Schrittbasteine (SB)

Operation (AWL)	Zulässige Operanden	1 VKE abhäng. 2 VKE beeinfl. 3 VKE begrenz.			typische Ausführungs- zeit in μ s		Funktionsbeschreibung
		1	2	3	CPU 945		
Alarmer sperren und freigeben							
AS		N	N	N	28		Alarm sperren: Peripheriealarmer, Zeit-OB-Bearbeitung und Zeitalarm-OB-Bearbeitung werden gesperrt (gilt nicht für die OBs 26, 33, 35)
AF		N	N	N	30 - 35		Alarm freigeben: hebt die Wirkung der Operation AS wieder auf (gilt nicht für die OBs 26, 33, 35)
SIM		N	N	N	33 - 38		Interruptmaske setzen (32 Bit)
LIM		N	N	N	29		Interruptmaske lesen (32 Bit)
Dekrementieren / Inkrementieren							
D	Parameter n=0 ... 255	N	N	N	0,1		AKKU 1-LL wird um die im Parameter angegebene Zahl dekrementiert (kein Übertrag)
I	Parameter n=0 ... 255	N	N	N	0,1		AKKU 1-LL wird um die im Parameter angegebene Zahl inkrementiert (kein Übertrag)

Ergänzende Operationen

für Organisationsbausteine (OB)

für Funktionsbausteine (FB)

für Programmbausteine (PB)

für Schrittbausteine (SB)

Operation (AWL)	Zulässige Operanden	1 VKE abhäng. 2 VKE beeinfl. 3 VKE begrenzt.			typische Ausführungs- zeit in µs		Funktionsbeschreibung
		1	2	3	CPU 945		
Sonstige Operationen							
B=	Formaloperand DB, PB, FB, SB, OB	N	N	N	0,5*		Baustein bearbeiten. (Nur A DB, SPA PB, SPA FB, SPA SB und SPA OB können substituiert werden)
B	DW**	N	N	N	0,25		Datenwort bearbeiten: die nachfolgende Operation wird mit dem im Datenwort angegebenen Parameter kombiniert (ODER-Verknüpfung) und ausgeführt **
B	MW**	N	N	N	0,25		Merkerwort bearbeiten: die nachfolgende Operation wird mit dem im Merkerwort angegebenen Parameter kombi- niert (ODER-Verknüpfung) und ausgeführt **

* plus Bearbeitungszeit des substituierten Befehls

** Zulässige Operationen:

U, UN, O, ON;¹

S, R, =;²

FR T, R T, SA T, SE T, SI T, SS T, SV T;

FR Z, R Z, S Z, ZR Z, ZV Z;

L, T³

LC

SPA =, SPB =, SPZ =, SPN =, SPP =, SPM =, SPO =, SLW, SRW;

SLD, RLD, SVD, RRD,

D, I; ADD BF;

A DB; TNB, TNW, SPA, SPB

¹ nur für E, A, M, T, Z

² nur für E, A, M

³ nur für EB, AB, EW, AW, ED, AD, MB, MW, MD, DL, DR,
DW, DD, BS, BT, PY, PW, QB, QW

Systemoperationen

für Organisationsbausteine (OB)

für Funktionsbausteine (FB)

für Programmbausteine (PB)

für Schrittbauusteine (SB)

Operation (AWL)	Zulässige Operanden	1 VKE abhängig. 2 VKE beeinfl. 3 VKE begrenzt.			typische Ausführungszeit in μs		Funktionsbeschreibung
		1	2	3	CPU 945		
Bit-Setzoperationen							
SU	BS	N	N	J	0,45		Bit im Bereich der Systemdaten unbedingt setzen
SU	BT	N	N	J	0,45		Bit im erweiterten Bereich der Systemdaten unbedingt setzen
RU	BS	N	N	J	0,45		Bit im Bereich der Systemdaten unbedingt rücksetzen
RU	BT	N	N	J	0,45		Bit im erweiterten Bereich der Systemdaten unbedingt rücksetzen
Lade- und Transferoperationen							
LIR	0 (\rightarrow AKKU 1-L) 1 (\rightarrow AKKU 1-H) 2 (\rightarrow AKKU 2-L) 3 (\rightarrow AKKU 2-H) 8 (\rightarrow DBL)	N	N	N	0,25*		Das angegebene 16-Bit-Register wird mit dem Inhalt einer Speicherzelle (durch AKKU 1 adressiert) geladen
TIR	0 (\rightarrow AKKU 1-L) 1 (\rightarrow AKKU 1-H) 2 (\rightarrow AKKU 2-L) 3 (\rightarrow AKKU 2-H) 8 (\rightarrow DBL)	N	N	N	0,25*		Der Inhalt des angegebenen 16-Bit-Registers wird in eine Speicherzelle (durch AKKU 1 adressiert) transferiert
LDI	A1 (\rightarrow AKKU 1) A2 (\rightarrow AKKU 2) SA (\rightarrow SAZ) BR (\rightarrow BR-Register)	N	N	N	0,4**		Das angegebene 32-Bit-Register mit dem Inhalt einer Speicherzelle n (adressiert durch AKKU 1) und in die darauffolgenden n+1, n+2 und n+3 laden

* bei Zugriffen im Peripheriebereich + 2 \times Readyverzugszeit der angesprochenen Peripheriebaugruppe

** bei Zugriffen im Peripheriebereich + 4 \times Readyverzugszeit der angesprochenen Peripheriebaugruppe

Systemoperationen

- für Organisationsbausteine (OB)
 für Programmbausteine (PB)

- für Funktionsbausteine (FB)
 für Schrittbasteine (SB)

Operation (AWL)	Zulässige Operanden	1 VKE abhängig. 2 VKE beeinfl. 3 VKE begrenzt.			typische Ausführungszeit in μs		Funktionsbeschreibung
		1	2	3	CPU 945		
Lade- und Transferoperationen (Fortsetzung)							
TDI	A1 (\rightarrow AKKU 1) A2 (\rightarrow AKKU 2) SA (\rightarrow SAZ) BR (\rightarrow BR-Register)	N	N	N	0,3*		Den Inhalt des angegebenen 32-Bit-Registers in eine Speicherzelle n (adressiert durch AKKU 1) und in die darauffolgenden n + 1, n + 2 und n + 3 transferieren
TNW	Parameter n=0 ... 255	N	N	N	0,15 + $n \times 0,35^{**}$		Wortweiser Blocktransfer Endadresse Zielbereich: AKKU 1 Endadresse Quellbereich: AKKU 2 Es dürfen nur gerade Adressen verwendet werden. Die AKKUs sind nach der Befehlsbearbeitung um $2 \times$ die Anzahl der Worte dekrementiert
TNB	Parameter n=0 ... 255	N	N	N	0,15 + $n \times 0,35^{**}$		Byteweiser Blocktransfer Endadresse Zielbereich: AKKU 1 Endadresse Quellbereich: AKKU 2 Die AKKUs sind nach der Befehlsbearbeitung um die Anzahl der Bytes dekrementiert
L	BS	N	N	N	0,2		Ein Wort aus dem Bereich Systemdaten in den AKKU 1 laden
L	BT	N	N	N	0,2		Ein Wort aus dem erweiterten Bereich Systemdaten in den AKKU 1 laden
T	BS	N	N	N	0,2		Ein Wort in den Bereich Systemdaten transferieren
T	BT	N	N	N	0,2		Ein Wort in den erweiterten Bereich Systemdaten transferieren

* bei Zugriffen im Peripheriebereich + $4 \times$ Readyverzugszeit der angesprochenen Peripheriebaugruppe

** bei Zugriffen im Peripheriebereich + $1 \times$ Readyverzugszeit der angesprochenen Baugruppe pro Bytezugriff

Systemoperationen

für Organisationsbausteine (OB)

für Funktionsbausteine (FB)

für Programmbausteine (PB)

für Schrittbasteine (SB)

Operation (AWL)	Zulässige Operanden	1 VKE abhängig. 2 VKE beeinfl. 3 VKE begrenzt.			typische Ausführungszeit in μ s		Funktionsbeschreibung
		1	2	3	CPU 945		
Lade- und Transferoperationen (Fortsetzung)							
MBR	Parameter 0 ... FFFF _H	N	N	N	0,25		Das BR-Register mit einer 20-Bit-Konstanten laden
ABR	Offset - 32768 ... + 32767	N	N	N	0,25		Der Offset wird mit 2 multipliziert (Wort-Offset). Das Ergebnis zum Inhalt des BR-Registers addieren
LRW	Offset - 32768 ... + 32767	N	N	N	0,35		Der Offset wird mit 2 multipliziert (Wort-Offset). Das Ergebnis zum Inhalt des BR-Registers addieren und das so adressierte Wort in den AKKU 1-L laden
LRD	Offset - 32768 ... + 32767	N	N	N	0,45		Der Offset wird mit 2 multipliziert (Wort-Offset). Das Ergebnis zum Inhalt des BR-Registers addieren und das so adressierte Doppelwort in den AKKU 1 laden
TRW	Offset - 32768 ... + 32767	N	N	N	0,35		Der Offset wird mit 2 multipliziert (Wort-Offset). Das Ergebnis wird zum Inhalt des BR-Registers addiert und den Inhalt von AKKU 1-L in das so adressierte Wort transferieren
TRD	Offset - 32768 ... + 32767	N	N	N	0,45		Der Offset wird mit 2 multipliziert (Wort-Offset). Das Ergebnis wird zum Inhalt des BR-Registers addiert und den Inhalt von AKKU 1 in das so adressierte Doppelwort transferieren
MAS		N	N	N	0,25		Den Inhalt von AKKU 1 (Bits 31 ... 20 müssen Null sein) in den Stepadreßzähler (SAZ) transferieren
MAB		N	N	N	0,1		Den Inhalt von AKKU 1 (Bits 31 ... 20 bleiben unverändert) in das Basisadreßregister transferieren

Systemoperationen

für Organisationsbausteine (OB)

für Funktionsbausteine (FB)

für Programmbausteine (PB)

für Schrittbauusteine (SB)

Operation (AWL)	Zulässige Operanden	1 VKE abhängig. 2 VKE beeinfl. 3 VKE begrenzt.			typische Ausführungszeit in μ s		Funktionsbeschreibung
		1	2	3	CPU 945		
Lade- und Transferoperationen (Fortsetzung)							
MSA		N	N	N	0,1		Den Inhalt des Stepadreßzählers in den AKKU 1 (Bits 31 ... 20 werden zu Null gesetzt) transferieren (Adresse des nächsten Befehls)
MSB		N	N	N	0,1		Den Inhalt des Stepadreßzählers in das Basisadreßregister transferieren (Adresse des nächsten Befehls).
MBA		N	N	N	0,1		Den Inhalt des Basisadreßregisters in den AKKU 1 (Bits 31 ... 20 werden zu Null gesetzt) transferieren
MBS		N	N	N	0,25		Den Inhalt des Basisadreßregisters in den Stepadreßzähler transferieren
Arithmetische Operationen							
ADD	BF - 128 ... + 127	N	N	N	0,1		Byte-Konstante (Festpunkt) zum AKKU 1-L addieren
ADD	KF - 32768 ... + 32767	N	N	N	0,25		Festpunkt-Konstante (Wort) zum AKKU 1-L addieren
ADD	DH 0 ... FFFF FFFF	N	N	N	0,35		Konstante (Doppelwort als Hexadezimalzahl) zum AKKU 1 addieren

Systemoperationen

für Organisationsbausteine (OB)

für Funktionsbausteine (FB)

für Programmbausteine (PB)

für Schrittbauusteine (SB)

Operation (AWL)	Zulässige Operanden	1 VKE abhäng. 2 VKE beeinfl. 3 VKE begrenz.			typische Ausführungs- zeit in μ s		Funktionsbeschreibung
		1	2	3	CPU 945		
Sonstige Operationen							
STS		N	N	N	44		Stop-Befehl: unmittelbar nach dem Befehl wird die Programmbearbeitung abgebrochen
STP		N	N	N	32		Stop am Ende der Programmbearbeitung; aktuelle Pro- grammbearbeitung wird zu Ende gebracht; das PAA wird ausgegeben; danach geht AG in STOP
TAK		N	N	N	0,1		Den Inhalt von AKKU 1 und AKKU 2 tauschen

Auflistung des Maschinencodes

Erläuterungen zu den Indizes

- a + Byteadresse
- b + Bitadresse
- c + Parameteradresse
- d + Zeitgliednummer
- e + Konstante
- f + Bausteinnummer
- g + Wortadresse
- h + Schiebezahl
- i + relative Sprungadresse
- k + Registeradresse
- l + Blocklänge
- m + Sprungweite (16 Bit)
- n + Wert
- o + Zählernummer

Maschinen-Code								Opera- tion	Ope- rand
B0		B1		B2		B3			
L	R	L	R	L	R	L	R		
0	0	0	0					NOF 0	
0	1	0	0					KEW	
0	2	0 _d	0 _d					L	T
0	3	0 _l	0 _l					TNB	
0	4	0 _d	0 _d					FR	T
0	5	0	0					BEB	
0	6	0 _c	0 _c					FR=	
0	7	0 _c	0 _c					U=	
0	8	0	0					AS	
0	8	8	0					AF	
0	9	0	0					KZW	
0	A	0 _a	0 _a					L	MB
0	B	0 _a	0 _a					T	MB
0	C	0 _d	0 _d					LC	T
0	D	0 _i	0 _i					SPO=	

Maschinen-Code								Opera- tion	Ope- rand
B0		B1		B2		B3			
L	R	L	R	L	R	L	R		
0	E	0 _c	0 _c					LC=	
0	F	0 _c	0 _c					O=	
1	0	0 _e	0 _e					BLD	0-255
1	0	8	2					BLD	130
1	0	8	3					BLD	131
1	0	8	4					BLD	132
1	0	8	5					BLD	133
1	0	F	F					BLD	255
1	1	0 _n	0 _n					I	
1	2	0 _a	0 _a					L	MW
1	3	0 _a	0 _a					T	MW
1	4	0 _d	0 _d					SA	T
1	5	0 _i	0 _i					SPP=	
1	6	0 _c	0 _c					SAR=	
1	7	0 _c	0 _c					S=	
1	9	0 _n	0 _n					D	
1	A	0 _a	0 _a					L	MD
1	B	0 _a	0 _a					T	MD
1	C	0 _d	0 _d					SV	T
1	D	0 _f	0 _f					SPB	FB
1	E	0 _c	0 _c					SVZ=	
1	F	0 _c	0 _c					==	
2	0	0 _f	0 _f					A	DB
2	1	2	0					>F	
2	1	4	0					<F	

Maschinen-Code								Opera- tion	Ope- rand
B0		B1		B2		B3			
L	R	L	R	L	R	L	R		
2	1	6	0					><F	
2	1	8	0					!=F	
2	1	A	0					>=F	
2	1	C	0					<=F	
2	2	0 _g	0 _g					L	DL
2	3	0 _g	0 _g					T	DL
2	4	0 _d	0 _d					SE	T
2	5	0 _i	0 _i					SPM=	
2	6	0 _c	0 _c					SE=	
2	7	0 _c	0 _c					UN=	
2	8	0 _e	0 _e					L	KB
2	9	0 _h	0 _h					SLD	
2	A	0 _g	0 _g					L	DR
2	B	0 _g	0 _g					T	DR
2	C	0 _d	0 _d					SS	T
2	D	0 _i	0 _i					SPA=	
2	E	0 _c	0 _c					SSV=	
2	F	0 _c	0 _c					ON=	
3	0	0	1	0 _e	0 _e	0 _e	0 _e	L	KZ
3	0	0	2	0 _e	0 _e	0 _e	0 _e	L	KT
3	0	0	4	0 _e	0 _e	0 _e	0 _e	L	KF
3	0	1	0	0 _e	0 _e	0 _e	0 _e	L	KC
3	0	2	0	0 _e	0 _e	0 _e	0 _e	L	KY
3	0	4	0	0 _e	0 _e	0 _e	0 _e	L	KH
3	0	8	0	0 _e	0 _e	0 _e	0 _e	L	KM

Maschinen-Code								Opera- tion	Ope- rand
B0		B1		B2		B3			
L	R	L	R	L	R	L	R		
3	1	2	0					>G	
3	1	4	0					<G	
3	1	6	0					><G	
3	1	8	0					!=G	
3	1	A	0					>=G	
3	1	C	0					<=G	
3	2	0 _g	0 _g					L	DW
3	3	0 _g	0 _g					T	DW
3	4	0 _d	0 _d					SI	T
3	5	0 _i	0 _i					SPN=	
3	6	0 _c	0 _c					SI=	
3	7	0 _c	0 _c					RB=	
3	8	0	0	0 _e	0 _e	0 _e	0 _e	L	KG ¹
3	8	4	0	0 _e	0 _e	0 _e	0 _e	L	DH ¹
3	9	2	0					>D	
3	9	4	0					<D	
3	9	6	0					><D	
3	9	8	0					!=D	
3	9	A	0					>=D	
3	9	C	0					<=D	
3	A	0 _g	0 _g					L	DD
3	B	0 _g	0 _g					T	DD
3	C	0 _d	0 _d					R	T
3	D	0 _f	0 _f					SPA	FB
3	E	0 _c	0 _c					RD=	
3	F	0 _c	0 _c					LW=	

¹ 3-Wort-Befehl mit B4 und B5, belegt mit 0_e

Maschinen-Code								Opera- tion	Ope- rand
B0		B1		B2		B3			
L	R	L	R	L	R	L	R		
4	0	0	0 _k					LIR	
4	1	0	0					UW	
4	2	0 _o	0 _o					L	Z
4	3	0 _l	0 _l					TNW	
4	4	0 _o	0 _o					FR	Z
4	5	0 _i	0 _i					SPZ=	
4	6	0 _c	0 _c					L=	
4	8	0	0 _k					TIR	
4	9	0	0					OW	
4	A	0 _a	0 _a					L	EB
4	A	8 _a	0 _a					L	AB
4	B	0 _a	0 _a					T	EB
4	B	8 _a	0 _a					T	AB
4	C	0 _o	0 _o					LC	Z
4	D	0 _f	0 _f					SPB	OB
4	E	0 _g	0 _g					B	MW
4	F	0 _g	0 _g					L	BT
5	0	0 _e	0 _e					ADD	BF
5	1	0	0					XOW	
5	2	0 _a	0 _a					L	EW
5	2	8 _a	0 _a					L	AW
5	3	0 _a	0 _a					T	EW
5	3	8 _a	0 _a					T	AW
5	4	0 _o	0 _o					ZR	Z
5	5	0 _f	0 _f					SPB	PB

Maschinen-Code								Opera- tion	Ope- rand
B0		B1		B2		B3			
L	R	L	R	L	R	L	R		
5	6	0 _c	0 _c					LD=	
5	7	0 _a	0 _a					L	QW
5	8	0	0	0 _e	0 _e	0 _e	0 _e	ADD	KF
5	9	0	0					-F	
5	A	0 _a	0 _a					L	ED
5	A	8 _a	0 _a					L	AD
5	B	0 _a	0 _a					T	ED
5	B	8 _a	0 _a					T	AD
5	C	0 _o	0 _o					S	Z
5	D	0 _f	0 _f					SPB	SB
5	F	0 _a	0 _a					L	QB
6	0	0	0					:F	
6	0	0	3					:G	
6	0	0	4					xF	
6	0	0	5	0 _e	0 _e	0 _e	0 _e	ADD	DH ¹
6	0	0	7					xG	
6	0	0	9					-D	
6	0	0	B					-G	
6	0	0	C	0	0	0 _i	0 _i	SPS=	
6	0	0	D					+D	
6	0	0	F					+G	
6	1	0	0 _h					SLW	
6	2	0 _g	0 _g					L	BS
6	3	0 _g	0 _g					T	BS
6	4	0 _h	0 _h					RLD	

¹ 3-Wort-Befehl mit B4 und B5, belegt mit 0_e

Maschinen-Code								Opera- tion	Ope- rand
B0		B1		B2		B3			
L	R	L	R	L	R	L	R		
6	5	0	0					BE	
6	5	0	1					BEA	
6	6	0 _c	0 _c					T=	
6	8	0	0	0 _e	0 _e	0 _e	0 _e	LRW	
6	8	0 _h	1					SVW	
6	8	0	2					GFD	
6	8	0	3	0 _e	0 _e	0 _e	0 _e	TRW	
6	8	0	4	0 _e	0 _e	0 _e	0 _e	LRD	
6	8	0	5	0 _e	0 _e	0 _e	0 _e	TRD	
6	8	0	6					FDG	
6	8	0	7					KZD	
6	8	0	8					DUF	
6	8	0	A					DUD	
6	8	0	B					LDI	A1
6	8	0	C					DEF	
6	8	0	E					DED	
6	8	0	F					TDI	A1
6	8	1	9					MAS	
6	8	2	9					MAB	
6	8	2	B					LDI	A2
6	8	2	F					TDI	A2
6	8	4	9					MSA	
6	8	4	B					LDI	SA
6	8	4	F					TDI	SA
6	8	6	9					MSB	

Maschinen-Code								Opera- tion	Ope- rand
B0		B1		B2		B3			
L	R	L	R	L	R	L	R		
6	8	8	9					MBA	
6	8	9	9					MBS	
6	8	A	B					LDI	BR
6	8	A	F					TDI	BR
6	9	0	0 _h					SRW	
6	C	0 _o	0 _o					ZV	Z
6	D	0 _f	0 _f					SPA	OB
6	E	0 _g	0 _g					B	DW
6	F	0 _g	0 _g					T	BT
7	0	0	0					STS	
7	0	0	2					TAK	
7	0	0	3					STP	
7	0	0	B	0 _m	0 _m	0 _m	0 _m	SPR	
7	0	0	C					LIM	
7	0	0	D					SIM	
7	0	0	E	0	0 _b	0 _g	0 _g	RU	BT
7	0	0	E	4	0 _b	0 _g	0 _g	SU	BT
7	0	0	E	8	0 _b	0 _g	0 _g	PN	BT
7	0	0	E	C	0 _b	0 _g	0 _g	P	BT
7	0	1	5	0	0 _b	0 _o	0 _o	RU	Z
7	0	1	5	4	0 _b	0 _o	0 _o	SU	Z
7	0	1	5	8	0 _b	0 _o	0 _o	PN	Z
7	0	1	5	C	0 _b	0 _o	0 _o	P	Z
7	0	2	5	0	0 _b	0 _d	0 _d	RU	T
7	0	2	5	4	0 _b	0 _d	0 _d	SU	T

Maschinen-Code								Opera- tion	Ope- rand
B0		B1		B2		B3			
L	R	L	R	L	R	L	R		
7	0	2	5	8	0 _b	0 _d	0 _d	PN	T
7	0	2	5	C	0 _b	0 _d	0 _d	P	T
7	0	3	8	0	0 _b	0 _a	0 _a	RU	E
7	0	3	8	0	0 _b	8 _a	0 _a	RU	A
7	0	3	8	4	0 _b	0 _a	0 _a	SU	E
7	0	3	8	4	0 _b	8 _a	0 _a	SU	A
7	0	3	8	8	0 _b	0 _a	0 _a	PN	E
7	0	3	8	8	0 _b	8 _a	0 _a	PN	A
7	0	3	8	C	0 _b	0 _a	0 _a	P	E
7	0	3	8	C	0 _b	8 _a	0 _a	P	A
7	0	4	6	0	0 _b	0 _g	0 _g	RU	D
7	0	4	6	4	0 _b	0 _g	0 _g	SU	D
7	0	4	6	8	0 _b	0 _g	0 _g	PN	D
7	0	4	6	C	0 _b	0 _g	0 _g	P	D
7	0	4	9	0	0 _b	0 _a	0 _a	RU	M
7	0	4	9	4	0 _b	0 _a	0 _a	SU	M
7	0	4	9	8	0 _b	0 _a	0 _a	PN	M
7	0	4	9	C	0 _b	0 _a	0 _a	P	M
7	0	5	7	0	0 _b	0 _g	0 _g	RU	BS
7	0	5	7	4	0 _b	0 _g	0 _g	SU	BS
7	0	5	7	8	0 _b	0 _g	0 _g	PN	BS
7	0	5	7	C	0 _b	0 _g	0 _g	P	BS
7	1	0 _h	0 _h					SVD	
7	2	0 _a	0 _a					L	PY
7	3	0 _a	0 _a					T	PY

Maschinen-Code								Opera- tion	Ope- rand
B0		B1		B2		B3			
L	R	L	R	L	R	L	R		
7	4	0 _h	0 _h					RRD	
7	5	0 _f	0 _f					SPA	PB
7	6	0 _c	0 _c					B=	
7	7	0 _a	0 _a					T	QW
7	8	0	1	0	1	0 _f	0 _f	BA	FX
7	8	0	2	0	9	0 _f	0 _f	BAB	FX
7	8	0	3	1	1	0 _f	0 _f	AX	DX
7	8	0	4	0	0	0 _f	0 _f	EX	DX
7	8	0	5	0	0	0 _f	0 _f	E	DB
7	8	0 _e	9	0 _e	0 _e	0 _e	0 _e	MBR	
7	8	0	A	0 _e	0 _e	0 _e	0 _e	ABR	
7	8	0	B	0 _b	0 _a	0 _a	0 _a	U	S
7	8	1	B	0 _b	0 _a	0 _a	0 _a	O	S
7	8	2	B	0 _b	0 _a	0 _a	0 _a	S	S
7	8	3	B	0 _b	0 _a	0 _a	0 _a	=	S
7	8	3	F	0	0 _b	0 _g	0 _g	U	D
7	8	3	F	1	0 _b	0 _g	0 _g	O	D
7	8	3	F	2	0 _b	0 _g	0 _g	UN	D
7	8	3	F	3	0 _b	0 _g	0 _g	ON	D
7	8	3	F	4	0 _b	0 _g	0 _g	S	D
7	8	3	F	5	0 _b	0 _g	0 _g	R	D
7	8	3	F	6	0 _b	0 _g	0 _g	=	D
7	8	4	B	0 _b	0 _a	0 _a	0 _a	UN	S
7	8	5	B	0 _b	0 _a	0 _a	0 _a	ON	S
7	8	6	B	0 _b	0 _a	0 _a	0 _a	R	S

Maschinen-Code								Opera- tion	Ope- rand
B0		B1		B2		B3			
L	R	L	R	L	R	L	R		
7	8	A	B	0	0 _a	0 _a	0 _a	L	SY
7	8	B	B	0	0 _a	0 _a	0 _a	T	SY
7	8	C	B	0	0 _a	0 _a	0 _a	L	SW
7	8	D	B	0	0 _a	0 _a	0 _a	T	SW
7	8	E	B	0	0 _a	0 _a	0 _a	L	SD
7	8	F	B	0	0 _a	0 _a	0 _a	T	SD
7	9	0	0					+F	
7	A	0 _a	0 _a					L	PW
7	B	0 _a	0 _a					T	PW
7	C	0 _o	0 _o					R	Z
7	D	0 _f	0 _f					SPA	SB
7	F	0 _a	0 _a					T	QB
8	0 _b	0 _a	0 _a					U	M
8	8 _b	0 _a	0 _a					O	M
9	0 _b	0 _a	0 _a					S	M
9	8 _b	0 _a	0 _a					=	M
A	0 _b	0 _a	0 _a					UN	M
A	8 _b	0 _a	0 _a					ON	M
B	0 _b	0 _a	0 _a					R	M
B	8	0 _o	0 _o					U	Z
B	9	0 _o	0 _o					O	Z
B	A	0	0					U(
B	B	0	0					O(
B	C	0 _o	0 _o					UN	Z
B	D	0 _o	0 _o					ON	Z

Maschinen-Code								Opera- tion	Ope- rand
B0		B1		B2		B3			
L	R	L	R	L	R	L	R		
B	F	0	0)	
C	0 _b	0 _a	0 _a					U	E
C	0 _b	8 _a	0 _a					U	A
C	8 _b	0 _a	0 _a					O	E
C	8 _b	8 _a	0 _a					O	A
D	0 _b	0 _a	0 _a					S	E
D	0 _b	8 _a	0 _a					S	A
D	8 _b	0 _a	0 _a					=	E
D	8 _b	8 _a	0 _a					=	A
E	0 _b	0 _a	0 _a					UN	E
E	0 _b	8 _a	0 _a					UN	A
E	8 _b	0 _a	0 _a					ON	E
E	8 _b	8 _a	0 _a					ON	A
F	0 _b	0 _a	0 _a					R	E
F	0 _b	8 _a	0 _a					R	A
F	8	0 _d	0 _d					U	T
F	9	0 _d	0 _d					O	T
F	A	0 _i	0 _i					SPB=	
F	B	0	0					O	
F	C	0 _d	0 _d					UN	T
F	D	0 _d	0 _d					ON	T
F	F	F	F					NOP1	

Alphabetisches Verzeichnis der Operationen

Operation	Seite	Operation	Seite
! = D	24, 71	> < F	24, 70
! G	26, 71	> < G	26, 71
! = F	24, 70	> = F	24, 70
)	8, 79	> F	24, 69
+ D	28, 73	A	32, 69
+ G	30, 73	ABR	62, 77
+ F	28, 78	ADD	64, 72, 73
- D	30, 73	AF	54, 68
- F	28, 73	AS	54, 68
- G	30, 73	AX	32, 77
:F	28, 73	B	56, 72, 75
:G	30, 73	B =	56, 77
< = D	26, 71	BA	30, 77
< D	26, 71	BAB	32, 77
< = F	24, 70	BE	34, 74
< F	24, 69	BEA	34, 74
< = G	28, 71	BEB	34, 68
< G	28, 71	BLD 0...255	34, 69
=	10, 77-79	BLD 130	34, 69
==	42, 69	BLD 131	36, 69
> = D	26, 71	BLD 132	36, 69
> = G	26, 71	BLD 133	36, 69
> D	26, 71	BLD 255	36, 69
> G	26, 71	D	54, 69
> < D	24, 71	DED	48, 74

Operation	Seite	Operation	Seite
DEF	46, 74	MAB	62, 74
DUD	48, 74	MAS	62, 74
DUF	48, 74	MBA	64, 75
E	32, 77	MBR	62, 77
EX	32, 77	MBS	64, 75
FDG	48, 74	MSA	64, 74
FR	44, 68, 72	MSB	64, 74
FR =	44, 68	NOP 0	34, 68
GFD	48, 74	NOP 1	34, 79
I	54, 69	O	8, 77-79
KEW	46, 68	O{	8, 78
KZD	46, 74	O =	38, 69
KZW	46, 68	ON	8, 77-79
L	12-16, 60, 68-78	ON =	38, 70
L =	46, 72	OW	38, 72
LC	16, 68, 72	P	40, 75, 76
LC =	46, 69	PN	40, 75, 76
LD =	46, 73	R	8, 22, 71, 77-79
LDI	58, 74, 75	RB =	42, 71
LIM	54, 75	RD =	44, 71
LIR	58, 72	RLD	50, 73
LRD	62, 74	RRD	50, 77
LRW	62, 74	RU	42, 58, 75, 76
LW =	46, 71	S	8, 22, 73, 77-79

Operation	Seite	Operation	Seite
S=	42, 69	SS	22, 70
SA	22, 69	SSV=	44, 70
SAR=	44, 69	STP	34, 66, 75
SE	22, 70	STS	66, 75
SE=	44, 70	SU	42, 58, 75, 76
SI	22, 71	SV	22, 69
SI=	44, 71	SVD	50, 76
SIM	54, 75	SVW	50, 74
SLD	48, 70	SVZ=	44, 69
SLW	48, 73	T	16-20, 68-78
SPA	30, 32, 71, 75-78	T=	46, 74
SPA=	50, 70	TAK	66, 75
SPB	32, 69, 72, 73	TDI	60, 74, 75
SPB=	50, 79	TIR	58, 72
SPM=	52, 70	TNB	60, 68
SPN=	52, 71	TNW	60, 72
SPO=	52, 68	TRD	62, 74
SPP=	52, 69	TRW	62, 74
SPR	52, 75	U	8, 77-79
SPS =	52, 73	U{	8, 78
SPZ=	52, 72	U=	38, 68
SRW	48, 75	UN	8, 77-79

Operation	Seite	Operation	Seite
UN=	38, 70	XOW	38, 72
UW	38, 72	ZR	22, 72
× F	28, 73	ZV	22, 75
× G	30, 73		

Liste der Organisationsbausteine

OB muß vom Anwender programmiert werden und wird vom Betriebssystem aufgerufen

OB-Nr.	Funktion
OB1	zyklische Programmbearbeitung
alarm-, zeitalarm- und zeitgesteuerte Programmbearbeitung	
OB2	Alarm A: Alarmgenerierung durch die Digitaleingabebaugruppe -434-7 und durch die Digitalein-/ausgabebaugruppe -485-7 und intelligente Peripheriebaugruppe (IP)
OB3	Alarm B: Alarmgenerierung durch IP
OB4	Alarm C: Alarmgenerierung durch IP
OB5	Alarm D: Alarmgenerierung durch IP
OB6	Alarm ausgelöst durch internen Timer
OB10	zeitgesteuerte Programmbearbeitung (jeweils variabel: 1 ms ... 65535 ms)
OB11	
OB12	
OB13	

OB muß vom Anwender programmiert werden und wird vom Betriebssystem aufgerufen

OB-Nr.	Funktion
Behandlung von Anlaufverhalten	
OB21	bei manuellem Einschalten
OB22	bei Spannungswiederkehr
Behandlung von Systemfehlern	
OB26	Zykluszeitüberschreitung
OB33	Weckfehler
OB35	Peripheriefehler
Behandlung von Programmier- und Gerätefehlern	
OB19	bei Aufruf eines nicht geladenen Bausteins
OB23	Quittungsverzug bei Einzelzugriff auf den S5-Bus (z.B. L PW, L PY, T PW, T PY, LIR, TIR)
OB24	Quittungsverzug beim Aktualisieren des PA und der Koppelmerker
OB27	Substitutionsfehler
OB32	Transferfehler
OB34	Batterieausfall

OB ist bereits integriert und muß vom Anwender aufgerufen werden

OB-Nr.	Funktion	Bearbeitungszeit (in μ s)
OB31	Zykluszeittriggerung	9,1
OB125*	STEP5-Bausteine erzeugen*	41
OB160	programmierbare Zeitschleife	3 ... 65535
OB182*	Datenbereich kopieren	$35 + n \times 2,65$ (n: Anzahl der Datenwörter)
OB183*	DX duplizieren	$52 + n \times 2,35$ (n: Anzahl der Datenwörter im DX)
OB184*	DB duplizieren	$52 + n \times 2,35$ (n: Anzahl der Datenwörter im DB)
OB190*	Merker byteweise in DB übertragen	$13 + n \times 1,85$ (n: Anzahl Bytes)
OB191*	Daten aus DB byteweise in Merker übertragen	$13 + n \times 1,85$ (n: Anzahl Bytes)
OB192*	Merker wortweise in DB übertragen	$12 + n \times 1,3$ (n: Anzahl Bytes)
OB193*	Daten aus DB wortweise in Merker übertragen	$12 + n \times 1,3$ (n: Anzahl Bytes)
OB220*	Vorzeichenerweiterung	3
OB251*	PID-Regelalgorithmus	max. 110
OB254	Prozeßabbild einlesen	$65 + n \times (1,5 + \text{Readyverzugszeit der Baugruppe})$
OB255	Prozeßabbild ausgeben	$65 + n \times (1,5 + \text{Readyverzugszeit der Baugruppe})$

* ab Firmwarestand Z 02

OB 250, Betriebssystem-Dienste

Dienst-Nr.	Funktion	Bearbeitungszeit (typisch)
1	Aktivierung des OB6	21 μ s
2 ... 5	Neues Intervall für OB10 ... OB13	20 μ s
6	Verändern der Eintragungen in den BS128 ... 143	33 μ s
7	Reduzierung des PAA-Transfers	18 μ s
8	Aufbau der Liste aller ansprechbarer Peripheriebytes	1,7 ms ... 86 ms $[t = 1700 \mu\text{s} + n \times (8 \mu\text{s} + \text{Readyverzugszeit}) + m \times (20 \mu\text{s} + \text{Readyverzugszeit}) + (512 - m - n) \times 165 \mu\text{s}]$ m = vorhandene analoge Peripheriebytes n = vorhandene digitale Peripheriebytes
10/11	Erzeugen eines DB/DX ohne TRAF	30 μ s ... 45 μ s
12	Neuaufbau der Bausteinadreßliste	50 μ s je vorhandenem gültigen Codebaustein 43 μ s je vorhandenem gültigen DB/DX 26 μ s je vorhandenem ungültigen Codebaustein 25 μ s je vorhandenem ungültigen DB/DX
13	Adresse (Byte) lesen vom S5-Bus	33 μ s + Readyverzugszeit (bei QVZ: 193 μ s)
14	Adresse (Byte) schreiben auf S5-Bus	35 μ s + Readyverzugszeit (bei QVZ: 195 μ s)
15	Adresse (Wort) lesen vom S5-Bus	34 μ s + Readyverzugszeit (bei QVZ: 194 μ s)
16	Adresse (Wort) schreiben auf S5-Bus	36 μ s + 2 \times Readyverzugszeit (bei QVZ: 196 μ s)
17/18	Byte von Kachel lesen/ Byte von Kachel schreiben	43 μ s + Readyverzugszeit (bei QVZ: 200 μ s)
19	BASP setzen und rücksetzen	24 μ s

OB 250, Betriebssystem-Dienste (Fortsetzung)

Dienst-Nr.	Funktion	Bearbeitungszeit (typisch)
20	DBA-Register lesen	25 μ s
21	DBA-Register schreiben	26 μ s
22	DBL-Register lesen	25 μ s
23	DBL-Register schreiben	25 μ s
24	DX indiziert aufschlagen	29 μ s
25	FX indiziert aufrufen	30 μ s
26	Baustein aus Bausteinadreßliste austragen	28 μ s
27	Bausteinkennung ändern nach "gültig im EPROM"	4,8 ms + n \times 19 μ s (n: Anzahl der zu ändernden Bausteine)
28	Bausteinkennung ändern nach "gültig im RAM"	6 ms + n \times 19 μ s (n: Anzahl der zu ändernden Bausteine)

Liste der Funktionsbausteine

Integrierte Funktionsbausteine

FB-Nr.	Funktion	Bearbeitungszeit (typisch)	Bearbeitungszeit (Berechnungsgrundlage)
FB238	AG komprimieren	20 μ s*	
FB239	Baustein löschen	19 μ s	
FB241	Analogwert einlesen von Analog-eingabebaugruppe 463	60 μ s	$t \approx 45\mu\text{s} + 2 \times \text{Readyverzugszeit}$
FB242	Analogwert einlesen von Analog-eingabebaugruppe 464	60 ... 115 μ s	$t \approx 50\mu\text{s} + 2 \times \text{Readyverzugszeit}$
FB243	Analogwert einlesen von Analog-eingabebaugruppe 466	120 μ s	$t \approx 45\mu\text{s} + 2 \times \text{Readyverzugszeit}$
FB244	Daten senden	2,9ms**	<ul style="list-style-type: none"> Leerdurchlauf VKE=0: $t \approx 54\mu\text{s} + 4 \times \text{Readyverzugszeit}$ mit Datenübertragung: $t \approx 1070\mu\text{s} + 30 \times \text{Readyverzugszeit} + n \times (2,5\mu\text{s} + \text{Readyverzugszeit})$ mit $n = \text{Minimum von Telegrammlänge, Blockgröße (in Bytes)}$
FB245	Daten empfangen	4,4ms**	<ul style="list-style-type: none"> Leerdurchlauf VKE=0: $t \approx 54\mu\text{s} + 4 \times \text{Readyverzugszeit}$ mit Datenübernahme: $t \approx 1280\mu\text{s} + 20 \times \text{Readyverzugszeit} + n \times (5\mu\text{s} + \text{Readyverzugszeit})$; mit $n = \text{Minimum von Telegrammlänge, Blockgröße (in Bytes)}$
FB246	Daten holen	0,9ms**	<ul style="list-style-type: none"> Leerdurchlauf VKE=0: $t \approx 54\mu\text{s} + 4 \times \text{Readyverzugszeit}$ mit Parameterübergabe: $t \approx 820\mu\text{s} + 44 \times \text{QVZ}$
FB247	Auftragsbearbeitung überwachen	58 μ s***	$t \approx 54\mu\text{s} + 4 \times \text{Readyverzugszeit}$
FB248	Auftrag löschen	1,2ms	<ul style="list-style-type: none"> Leerdurchlauf: $t \approx 30\mu\text{s} + 3 \times \text{QVZ}$ aktiv: $t \approx 1200\mu\text{s} + 65 \times \text{QVZ}$

* nur Anstoß zum Komprimieren

** bei Readyverzugszeit = 1 μ s und Blockgröße 6 (512 Bytes)

*** bei Readyverzugszeit = 1 μ s

Integrierte Funktionsbausteine (Fortsetzung)

FB-Nr.	Funktion	Bearbeitungszeit (typisch)	Bearbeitungszeit Berechnungsgrundlage
FB249	Schnittstelle einrichten	< 10s	
FB250	Analogwert einlesen von Analog- eingabebaugruppe 460/465	40 ... 100 μ s	$t \approx 54\mu\text{s} + 2 \times \text{Readyverzugszeit}$ (bei zyklischer Abtastung) $t \approx 40 \dots 100\mu\text{s}$ (bei Einzelabtastung)
FB251	Analogwert ausgeben	60 μ s	$t \approx 45\mu\text{s} + 2 \times \text{Readyverzugszeit}$

Datenbaustein 1

Datenbaustein 1

Parameter	Argument	zulässiger Wertebereich	Bedeutung
Blockkennung: SL1:			SINEC L1
SLN	p	p=0, 1 ... 30, 255	<p>"SLave-Nummer" "0" = Masterfunktion bei Punkt-zu-Punkt-Kopplung (an 2. Schnittstelle); Diese Parameter gelten für die 2. Schnittstelle, wenn keine Rechnerkopplung oder kein ASCII-Treiber aktiviert ist.</p> <p>Lage des Sende-Fachs (Anfang des SF) Lage des Empfangs-Fachs (Anfang des EF) Lage des Koordinierungs-Bytes Empfangen Lage des Koordinierungs-Bytes Senden</p> <p>PG-Bus-Nummer</p> <p>Hinweis: KBS und KBE liegen in einem Merkerbyte oder im High-Byte des angegebenen Datenwortes (DL)!</p>
SF	} DBxDWy oder DXaDWy oder MBy SYz	x=0 ... 255	
EF		y=0 ... 255	
KBE		a=0 ... 255	
KBS		z=0 ... 4095	
PGN	p	p=1 ... 30	
Blockkennung: SDP:			System-Dependent Parameters
WD	p	p=0 ... 2550	<p>"Watch-Dog" (Zykluszeit-Überwachung) in Millisekunden, aber nur in 10 ms-Schritten einstellbar</p>
RDLY	r	r=0 ... 65535	<p>"Run DeLaY" Anlaufverzögerung nach NETZ EIN in Millisekunden</p>
RT	J/N	-	<p>"Resident Timers" Bei "J" sind alle Zeiten remanent, bei "N" nur T 0 ... 63*.</p>
RC	J/N	-	<p>"Resident Counters" Bei "J" sind alle Zähler remanent, bei "N" nur Z 0 ... 63*.</p>
RF	J/N	-	<p>"Resident Flags" Bei "J" sind alle Merker, S-Merker remanent, bei "N" nur die erste Hälfte*.</p>
PROT	J/N	-	<p>"PROTection" Softwareschutz aktivieren? (Ein- und Ausgabe des Programms nicht mehr möglich)</p>
PIO	J/N	-	<p>"Process Image Output" Prozeßabbild ausgeben/sperren?</p>

* zusätzlicher Schalter für Voreinstellung/Urlöschen am Bedienfeld der CPU auf "RE" stellen

Datenbaustein DB1 (Fortsetzung)

Parameter	Argument	zulässiger Wertebereich	Bedeutung
Blockkennung: SDP: (Fortsetzung)			System-Dependent-Parameters (Fortsetzung)
PII	J/N	-	"Process Image Input" Prozeßabbild einlesen sperren?
RPIC	s	s=0 ... 255	"Reduced Process Image Output Counter" Zähler für die Zyklen mit reduziertem PAA-Transfer
PPIT	J/N	-	"Parallel Process Image Transfer" Bei "N" ist der sequentielle PA-Transfer, bei "J" ist der parallele PA-Transfer eingestellt
Blockkennung: TFB:			Timer-Function Block
OB10 OB11 OB12 OB13	} p	p=0 ... 65535	Intervall, in dem der OB10 ... 13 aufgerufen und bearbeitet wird. Zeit in Millisekunden
Blockkennung: ASC:			ASCII-Treiber an 2. Schnittstelle
SF EF KBE KBS PAR MOD BDR	} DBxDW _y oder DXaDW _y oder MBy oder SYz n m	x=0 ... 255 y=0 ... 255 a=0 ... 255 z=0...4095 n=1 ... 8 m= 150, 200, 300, 600, 1200, 2400, 4800, 9600, 19200	Hinweis: Treibernummer "1" für die 2. Schnittstelle wird automatisch eingestellt. Lage des <i>Sende-Fachs</i> (Anfang des SF) Lage des <i>Empfangs-Fachs</i> (Anfang des EF) Lage des <i>Koordinierungs-Bytes Empfangen</i> Lage des <i>Koordinierungs-Bytes Senden</i> Lage des <i>PARAMetersatzes</i> " MODE number" Modusnummer des ASCII-Treibers " BauD Rate " Baudrate

Datenbaustein DB1 (Fortsetzung)

Parameter	Argument	zulässiger Wertebereich	Bedeutung
Blockkennung: ASC: (Fortsetzung)			ASCII-Treiber an 2. Schnittstelle (Fortsetzung)
PRTY	p	p= E O M S N	"PaRiTY" Parität Even = gerade Odd = ungerade Mark = Mark ("1") Space = Space ("0") None = keine Überprüfung
DF	q	q=0 ... 5, 7, 8	"Data Format" Datenformat
WCR	r	r=0 ... 2550	"Wait time after Carriage Return" Wartezeit nach CR in Millisekunden
WLF	r	r=0 ... 2550	"Wait time after Line Feed" Wartezeit nach LF in Millisekunden
WFF	r	r=0 ... 2550	"Wait time after Form Feed" Wartezeit nach FF in Millisekunden
DT	s	s= 10 ... 655350	"Delay Time" Zeichenverzugszeit in Millisekunden
ML	t	t=0 ... 1024	"Mail Length" Telegrammlänge in Bytes für Empfangstelegramm; für Modus 1, 8
ME	u v	u, v=0 ... 255	"Mail End" Endekennung; für Modus 2, 3, 4, 5, 6 RUB OUT = 7F _H = 127 _D XON = 11 _H = 17 _D XOFF = 13 _H = 19 _D CR = 0D _H = 13 _D LF = 0A _H = 10 _D FF = 0C _H = 12 _D EOT = 04 _H = 4 _D ETX = 03 _H = 3 _D
SLF	J/N	-	"Skip LF" LF unterdrücken
LPP	w	w= 1 ... 255	"Lines Per Page" Zeilen pro Seite
LM	w	w=0 ... 255	"Left Margin" Linker Rand
PN	o/u	-	"Page Number" Seitennummer; o=oben u=unten

Datenbaustein DB1 (Fortsetzung)

Parameter	Argument	zulässiger Wertebereich	Bedeutung
Blockkennung: ASC: (Fortsetzung)			ASCII-Treiber an 2. Schnittstelle (Fortsetzung)
HD1 HD2 FT1 FT2	"<string>" "<string>" "<string>" "<string>"	} max. 119 alpha- numerische Zeichen	" HeaDer 1 " Kopfzeile 1* "HeaDer 2" Kopfzeile 2* "FooTer 1" Fußzeile 1* "FooTer 2" Fußzeile 2* Druckersteuerzeichen werden als Hexadezimalzeichen wie folgt eingegeben: \$xx Beispiel: "TEXT \$1B\$38 WEITERER TEXT" (1B38=Breitschrift ein)
Blockkennung: RKT:			Rechnerkopplung an 2. Schnittstelle
SF EF KBS KBE PAR MOD BDR	} DBxDWy oder DXaDWy oder MBy oder SYz n m	x=0 ... 255 y=0 ... 255 a=0 ... 255 z=0 ... 4095 n= 1, 2 m= 150, 200, 300, 600, 1200, 2400, 4800, 9600, 19200	Hinweis: Treibernummer "2" für die 2. Schnittstelle wird automatisch eingestellt. Lage des <i>Sende-Fachs</i> (Anfang des SF) Lage des <i>Empfangs-Fachs</i> (Anfang des EF) Lage des <i>Koordinierungs-Bytes Empfangen</i> Lage des <i>Koordinierungs-Bytes Senden</i> Lage des <i>PARAMetersatzes</i> "MODE number" Modusnummer 1 = kein Block-Prüfzeichen 2 = Senden mit Block-Prüfzeichen "BauD Rate" Baudrate

- * - CR (Carriage Return) wird im Parameter-DB am Ende des ASCII-Strings automatisch generiert (Trennung der Zeilen).
 Wenn HD 1/2, FT 1/2 fehlt, dann wird nur CR abgelegt (→Kap. 12.5)
 - Für das Senden des \$-Zeichen muß \$\$ eingegeben werden.
 - Für das Senden des "-Zeichen muß \$" eingegeben werden.

Datenbaustein DB1 (Fortsetzung)

Parameter	Argument	zulässiger Wertebereich	Bedeutung
Blockkennung: RKT: (Fortsetzung)			Rechnerkopplung an 2. Schnittstelle (Fortsetzung)
PRTY	p	p= E O M S N	"PaRiTY" Parität Even = gerade Odd = ungerade Mark = Mark ("1") Space = Space ("0") None = keine Überprüfung
DF	q	q=0 ... 5, 7, 8	"Data Format" Datenformat
PRI	l/h		"PRIority" Priorität l = niedrig h = hoch
DT*	s	s = 10 ... 655350	"Delay Time" Zeichenverzugszeit in Millisekunden
TIO*	s	s = 10 ... 655350	"Time Out" Quittungsverzugszeit in Millisekunden
BWT*	s	s = 10 ... 655350	"Block Wait Time" Blockwartezeit in Millisekunden
TTE	r	r = 1 ... 255	"Tries To Erect" Aufbauversuche
TTS	r	r = 1 ... 255	"Tries To Send" Sendeversuche
Blockkennung: S2T:			Treiber Deaktivieren
DEAC	J/N	-	Deaktivieren von ASCII-Treiber oder Rechnerkopplung. Bei DEAC="J" wird Treibernummer "0" eingetragen (in BS46) und vorher definierte Parameter für RKT und ASC werden nicht übernommen.

* DT < TIO < BWT

Datenbaustein DB1 (Fortsetzung)

Parameter	Argument	zulässiger Wertebereich	Bedeutung
Blockkennung: CLP:			Clock-Parameter
CLK	DBxDWy oder DXxDWy oder MWb oder SWz	x=0 ... 255	" C lock Data" Beginn des Uhrendatenbereichs
STW		y=0 ... 255 b=0 ... 254 z=0 ... 4094	" S tatus W ord" Lage des Statuswortes
SET	wt tt.mm.jj hh:mm:ss AM/PM ¹	wt = 1 ... 7 tt = 01 ... 31 mm = 01 ... 12 jj = 00 ... 99 hh = 1 ... 12 00 ... 23 ss = 00 ... 59	Uhrzeit, Datum stellen Wochentag=So ... Sa Tag Monat Jahr AM/PM 24-Stunden-Modus Sekunden
TIS	wt tt.mm. hh:mm:ss AM/PM ¹		" T imer I nterrupt S et" Weckzeit stellen
OHS	hhhhhh:mm:ss ²	hhhhhh = 000000 ... 999999 mm = 00 ... 59 ss = 00 ... 59	" O peration H our counter S et" Betriebsstundenzähler stellen Stunden Minuten Sekunden

¹ Soll ein Argument (z.B. Wochentag) nicht übernommen werden: XX eingeben! - die Uhr läuft mit dem aktuellen Wert weiter. Geben Sie AM oder PM nach der Uhrzeit an, läuft die Uhr im jeweiligen 12 Stunden-Modus. Lassen Sie dieses Argument weg, läuft die Uhr im 24 Stunden-Modus.

² Soll ein Argument (z.B. Minuten) nicht übernommen werden: XX eingeben! - die Uhr läuft mit dem aktuellen Wert weiter.

Datenbaustein DB1 (Fortsetzung)

Parameter	Argument	zulässiger Wertebereich	Bedeutung
Blockkennung: CLP: (Fortsetzung)			Clock-Parameter (Fortsetzung)
OHE	J/N	-	" O peration H our counter E nable" Betriebsstundenzähler freigeben
STP	J/N	-	" SToP " Uhr im STOP-Zustand aktualisieren
SAV	J/N	-	" SAVe " Uhrzeit nach letztem RUN→STOP-Übergang bzw. NETZ AUS retten
CF	p	-400 ... +400	" C orrection F actor" Korrekturfaktor eingeben
Blockkennung: ERT:			Error Return
ERR	DBxDWy oder DXaDWy oder MBz oder SYp	x=0, 2 ... 255 y=0 ... 255 a=0 ... 255 z=0 ... 236 p=0 ... 4076	" ERR ors" Lage der Errorcodes

Auswertung von ANZ 1 und ANZ 0

ANZ 1	ANZ 0	Arith- meti- sche Opera- tionen	Digi- tale Ver- knüp- fungs- opera- tionen	Ver- gleichs- opera- tionen	Schie- beope- ratio- nen	Um- wand- lungs- opera- tio- nen	Ausge- führte Sprung- opera- tio- nen
0	0	Ergeb- nis =0	Ergeb- nis =0	AKKU 2 = AKKU 1	gescho- benes Bit =0	-	SPZ
0	1	Ergeb- nis <0	-	AKKU 2 < AKKU 1	-	Ergeb- nis <0	SPM SPN
1	0	Ergeb- nis >0	Ergeb- nis ≠0	AKKU 2 > AKKU 1	gescho- benes Bit =0	Ergeb- nis >0	SPP SPN

**An
Siemens AG
AUT 125 Doku
Postfach 1963
D-92209 Amberg**

Sollten Sie beim Lesen dieser Unterlage auf Druckfehler gestoßen sein, bitten wir Sie, uns diese mit diesem Vordruck mitzuteilen. Ebenso dankbar sind wir für Anregungen und Verbesserungsvorschläge.

Absender (bitte ausfüllen)

Name

Firma/Dienststelle

Anschrift

Telefon

Druckschrift:

Automatisierungsgerät

SIMATIC 55-115U (CPU 945)

Tabellenheft - Operationsliste

Bestell-Nr.: 6ES5 997-7LB11

Vorschläge und/oder Korrekturen: