

136.77
U.

**Ergänzung zum System-
und Programmierhandbuch
TA 10**

I N H A L T

8.	Ergänzung zum System- u. Programmierhandbuch TA 10	1
8.1	Das Modell TA 10/4	1
8.1.1	Schreibgeschwindigkeit	2
8.1.2	Tastatur u. Typenbelegung	2
8.1.3	Magnetkontokarten-Formate	3
8.2	Datentransfer	3
8.2.1	Position der PROM-Bausteine	4
8.3	Programmierung	5
8.3.1	Blockschaltbild TA 10/4	6
8.3.2	Lesebefehlsroutine	7
8.3.3	Aufzeichnungsbefehlsroutine	7
8.3.4	Einfahren der Kontokarte	8
8.3.5	Ausfahren der Kontokarte	9
8.3.6	Einfahren und Lesen	10
8.3.7	Schlußbefehl, Lesebefehlsroutine	11
8.3.8	Anfangsbefehl, Aufzeichnungsroutine	12
8.3.9	Ausfahren und Schreiben	13
8.3.10	Schreiben (bei Wagenrücklauf)	14
8.3.11	Zeilenschaltung	15
8.3.12	Saldenlisten	16
8.3.13	Output KA	17
8.3.14	Wait for Key 7	18
8.3.15	Jump letzte Zeile	19
8.3.16	Jump if Selector	21
8.3.17	Codierliste	24

- 8.1 MODELL TA 10/4 (Magnetkontencomputer) wird grundsätzlich nur mit Wagengröße IV (Spezialwagen mit 2 Papierführungs-wannen) und
KAN 1⁺⁾, ohne hinteren Kartenschacht
oder
EFKA⁺⁺⁾
geliefert.

Ausgeschlossen vom Modell TA 10/4 sind:

- der Lochstreifen-(Karten-) Stanzer
- die OCR-Tüchtigkeit
- sowie weitere Formulartechniken

Das Modell ist ausgestattet mit:

- 15 freien Datenspeichern
(je 15 Ziffern + Vorzeichen)
- max. 1024 Befehlsschritten
(2 kleine auswechselbare Kassetten mit jeweils 4 PROM's).
- 64 numerischen Stellen
(einschließlich Vorzeichen) Speicherkapazität auf den Magnetstreifen (2 kleine Streifen) je Kontenseite. Für die Vorzeichen ist jeweils eine numerische Stelle zu berücksichtigen.
- einem zusätzlichen RAM-Pufferspeicher
mit 64 numerischen Stellen einschließlich Vorzeichen für den Datentransfer.

¹⁾ KAN 1 = Vorsteckeinrichtung mit Magnetkonteneinzug

²⁾ EFKA = Vorsteck- und Endlosformulareinrichtung mit Magnetkonteneinzug

- Signallampen in der separaten Sondertastatur mit folgenden Bedeutungen:

- grüne Lampe = Anstehender Kontokarteneinfahrbefehl. Im Programmablauf ist ein Kontokarteneinfahr-befehl programmiert. Eine Kontokarte soll eingefahren werden.
- gelbe Lampe = Die Magnetbandstreifen sind leer, es sind keine Informationen auf den Magnetbandstreifen.
- rote Lampe = Eine Datenaufzeichnung auf die Magnetbandstreifen ist nicht möglich.
Die rote Lampe leuchtet, wenn die Kartentaschen von der Bedienungskraft zu weit eingestellt sind oder bei einigen Hardwarefehlern (z. B. Ausfall der Lichtschranke 3).
- blaue Lampe = Lesefehler
Die Informationen auf den Magnetbandstreifen sind fehlerhaft oder die Magnetbandstreifen sind zerstört.

8.1.1 Die Schreibgeschwindigkeit beträgt ohne Rotdruck:

- mit KAN 1 16 Z/s und
mit EFKA 14 Z/s.

Wird Rotdruck gewünscht, so muß dieser nachträglich vom Service-Techniker eingestellt werden. Zu beachten ist dabei jedoch, daß sich dann die Schreibgeschwindigkeit wie folgt reduziert:

- mit KAN 1 14 Z/s
mit EFKA 12 Z/s.

8.1.2 Tastatur- und Typenbelegungen wurden (bis auf die Signallampen in der separaten Sondertastatur) vom Modell TA 10/2 übernommen.

8.1.3 Magnetkontenkartenformate wurden auf nachstehende Standardkartenformate eingeschränkt:

Breite:	Höhe:
330 mm	297 mm
297 mm	297 mm
297 mm	210 mm
210 mm	297 mm
210 mm	210 mm

8.2 DATENTRANSFER

Wie aus dem Blockschaltbild ersichtlich, wird für die Magnetkontenverarbeitung die KAN-Elektronik zusätzlich mit einem RAM-Pufferspeicher ausgestattet. Dieser Speicher ermöglicht über die modifizierte LSKA-Schnittstelle den programmgesteuerten Datenverkehr mit der Grundmaschine, deren bisheriger Datenfluß nicht verändert ist.

Von dem oder auf den Magnetbandstreifen werden die Datenwörter, durch Vorzeichen getrennt, in einem Block in den oder aus dem Pufferspeicher gebracht.

In Verbindung mit den Datenspeichern, die je nach Programmierung beliebig ausgewählt werden können, erfolgt der Verkehr mit dem Pufferspeicher ebenfalls blockweise, das heißt, daß diese Transfers innerhalb eines abgeschlossenen Programmabschnittes ausgeführt werden müssen. Wortendekennzeichen dieser Übertragungen sind jeweils die Vorzeichen der einzelnen Datenwörter.

Beim Lesen der Magnetstreifen werden die Daten mittels einer hardwaremäßigen Routine (Modulo 4-Prüfung) gesichert.

Bit 7 b ermöglicht, daß die Sprungbefehle größer 512 Befehlsschritte adressiert werden können.

Zusätzliche Programmtestmöglichkeiten wurden für das Modell TA 10/4 nicht entwickelt. Programmtestmöglichkeiten sind für Programme:

- bis 512 Befehlsschritte:
an die TA 10/4 den Programm-Umlaufspeicher einer TA 10/3 anzuschließen,
- über 512 Befehlsschritte:
das sofortige Eintasten bzw. Einschreiben der Befehle in die PROM's über Programmeingabegerät PET 3

8.2.1 POSITION DER PROM-BAUSTEINE

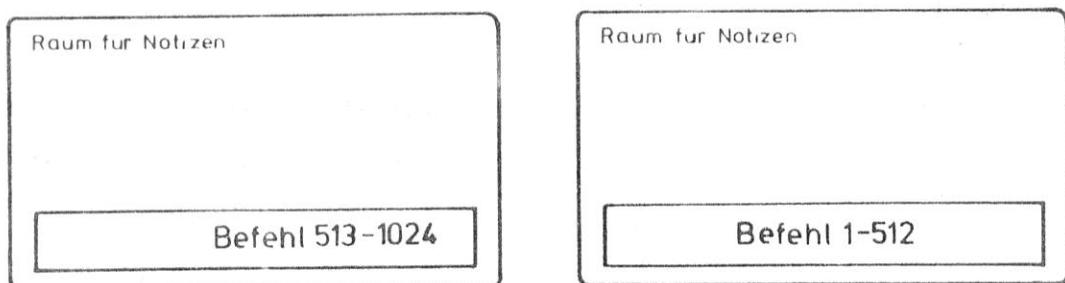


Abb. 1 Position der Kassetten mit Befehlsschrittangabe

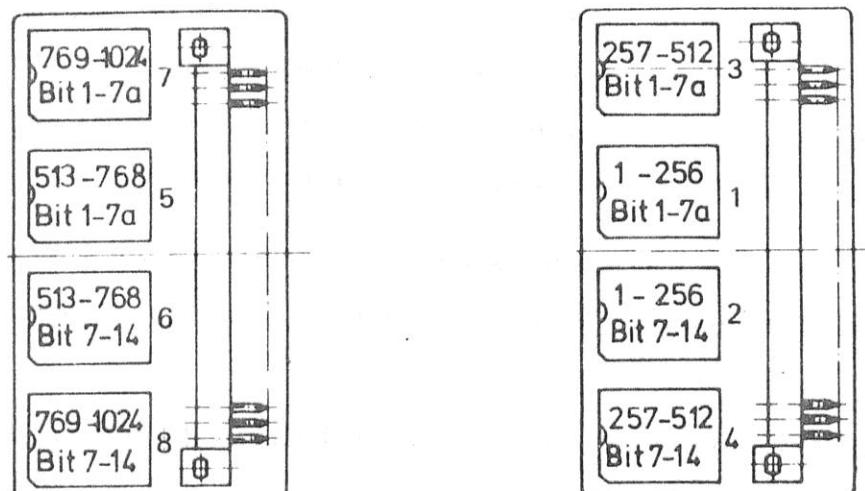
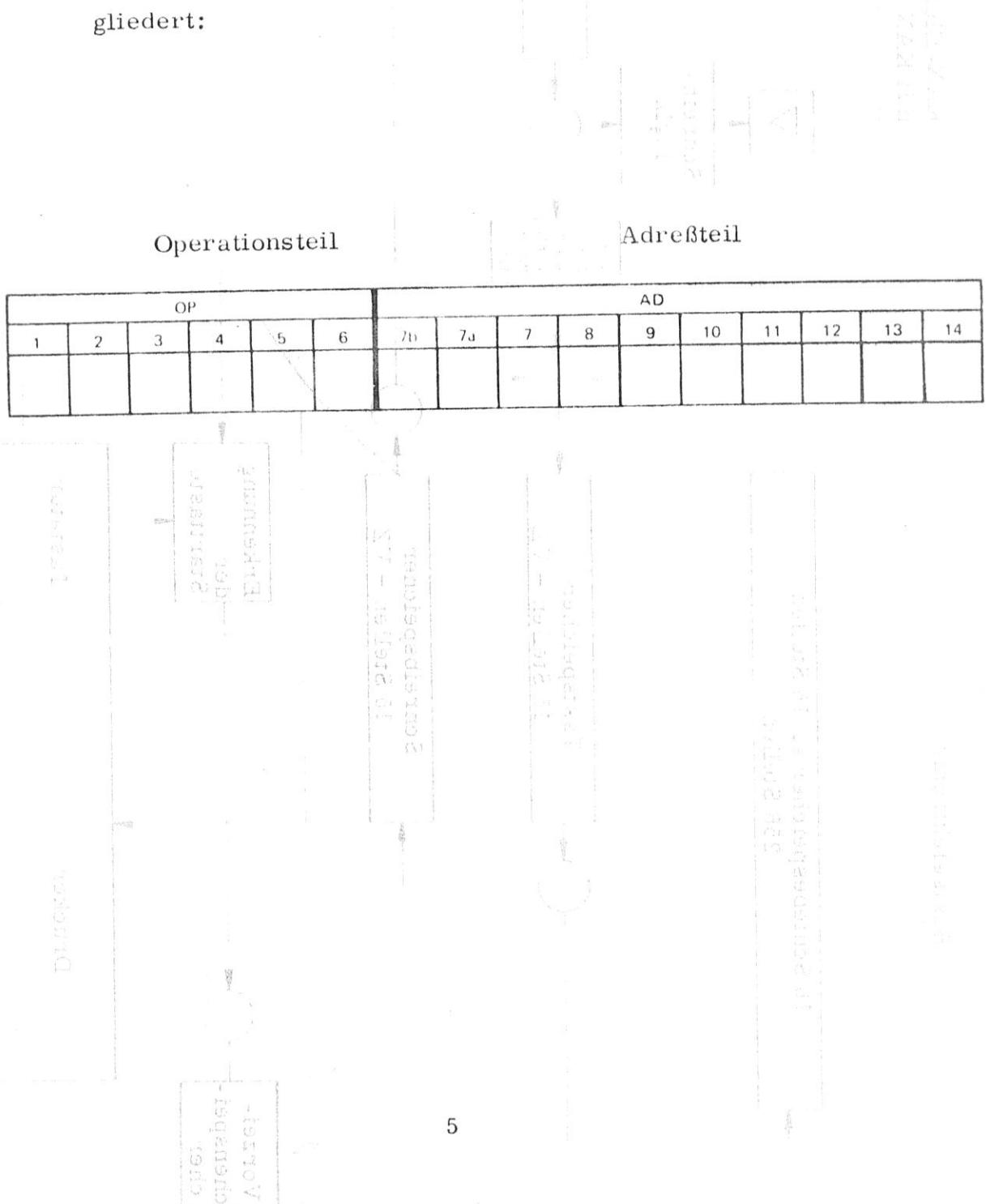


Abb. 2 Position der PROM-Bausteine mit Befehlsschrittangabe
Befehl 1 - 512 entspricht den Adressen 0 - 511
Befehl 513 - 1024 " " " 512 - 1023

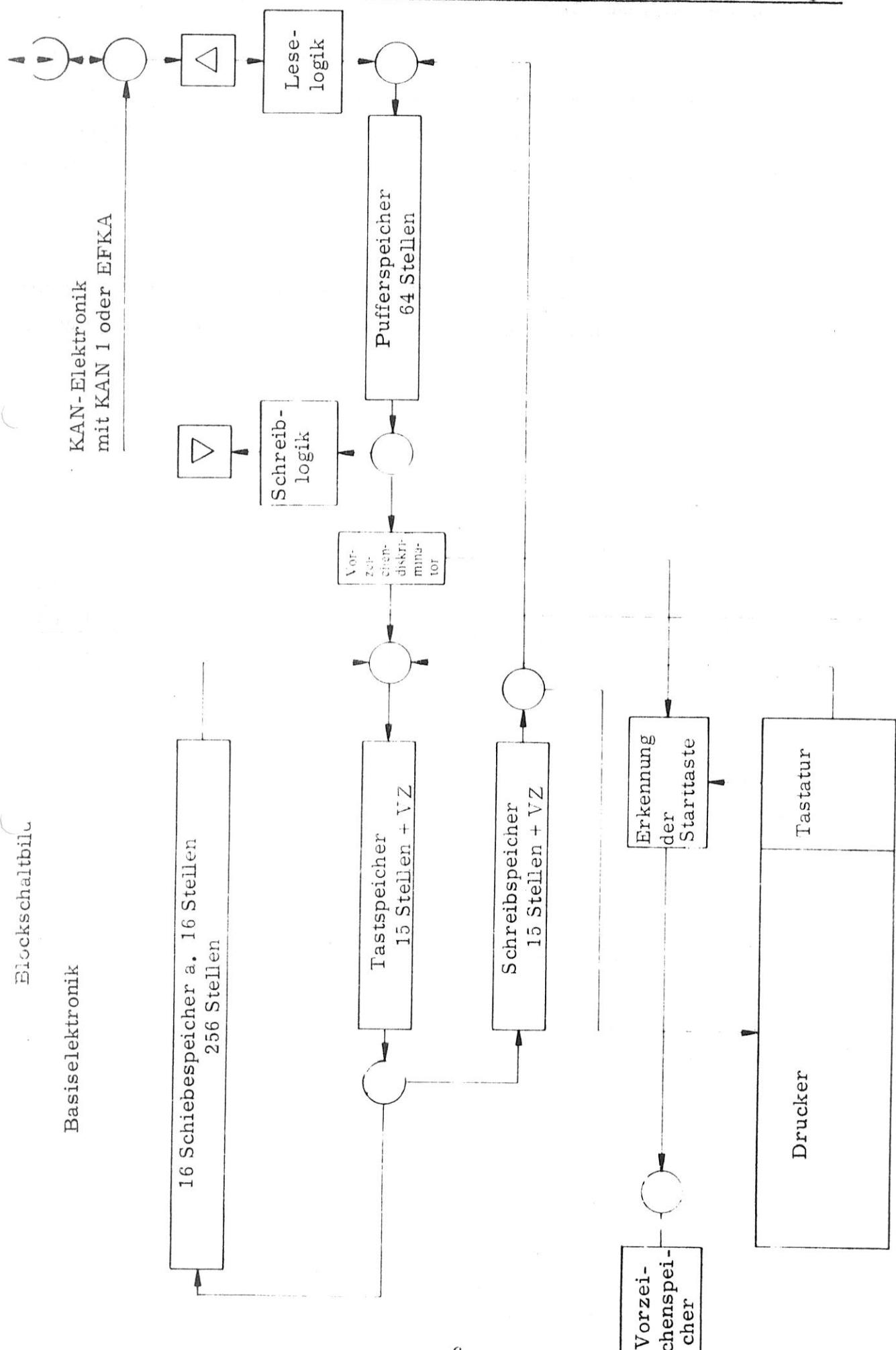
8.3 PROGRAMMIERUNG

Für die Programmierung des Modells TA 10/4 gelten, mit Ausnahme der Lochstreifen- (Karten-) Stanzerbefehle, die Befehle der TA 10/2. Die Lochstreifen- (Karten-) Stanzerbefehle wurden für den Betrieb mit Magnetkontokarten abgeändert und haben daher eine andere Bedeutung.

Außerdem wurde gegenüber den bisherigen TA 10-Modellen für alle TA 10/4-Befehle die Befehlswortlänge um 1 Bit (Bit 7 b im Adreßteil) auf 16 Binärstellen erweitert. Das Befehlswort ist nun wie folgt gegliedert:



8.3.1 Ausstattung des TA 10/4-Modells mit Magnetkonten-Einrichtungen



8.3.2 LESEBEFEHLSROUTINE

8.3.2.1 In einer Lesebefehlsroutine dürfen nur die Befehle SAL o., ELE, JSE, WAK/ni/7, ADD, TRV, BRV und ELE0 programmiert werden. Alle anderen Befehle sind unzulässig.

8.3.2.2 Eine Lesebefehlsroutine muß auf dem Befehl ELE o. SAL beginnen.

Nach ELE o. SAL muß der Befehl JSE folgen, damit bei Lesefehler oder leerer bzw. voll beschrifteter Kontokarte, oder wenn bereits eine Kontokarte im Kartenschacht ist, in eine Fehlerroutine verzweigt werden kann. Wird die Karte richtig eingefahren und gelesen, erfolgt der Datentransfer mit WAK/ni/7, TRV o. ADD usw.

Die Lesebefehlsroutine muß mit dem Befehl ELE 0 abgeschlossen werden.

Wird die Lesebefehlsroutine mit JSE verlassen, darf wieder frei programmiert werden.

8.3.2.3 Innerhalb einer Lesebefehlsroutine kann nach beliebig vielen Speichern transferiert oder addiert werden (siehe Beispiel bei ELE 0).

8.3.3 AUFZEICHNUNGSBEFEHLSROUTINE

8.3.3.1 In einer Aufzeichnungsbefehlsroutine dürfen nur die Befehle ASC 0, OKA, ADD, TRV, BRV und ASC oder SCH⁺⁾ programmiert werden. Alle anderen Befehle sind unzulässig.

8.3.3.2 Eine Aufzeichnungsbefehlsroutine muß mit dem Befehl ASC 0 beginnen. Nach ASC 0 erfolgt der Datentransfer mit BRV, OKA, BRV usw.

Die Aufzeichnungsbefehlsroutine muß mit dem Befehl ASC oder SCH⁺⁾ abgeschlossen werden.

Nach ASC 0, SCH⁺⁾ muß JSE folgen, um bei Nichterkennung des Kontokartenfensters (es wird nicht aufgezeichnet) in eine Fehlerroutine zu verzweigen.

8.3.3.3 Innerhalb einer Aufzeichnungsroutine kann nach beliebig vielen Speichern transferiert oder addiert werden (siehe Beispiel bei ASC)

⁺⁾ SCH/TAB \leqq Ist

8.3.4 Einfahren (der Kontokarte)

OP						AD									
1	2	3	4	5	6	7b	7a	7	8	9	10	11	12	13	14
☒					☒										

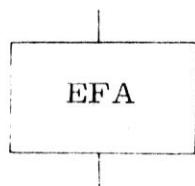
Symbolischer Befehlscode EFA/

Wirkung

Mit diesem Befehl wird die Kontokarte bis zur nächsten freien Zeile eingefahren. Die Karte stoppt zeilengerecht durch das rein mechanische Abfühlen der Schlitzungen auf der Kontokarte. Die Informationen auf den Magnetbandstreifen werden nicht gelesen.

Beispiel:

Symbol



1	2	3	4	5	6	7b	7a	7	8	9	10	11	12	13	14
☒					☒										

8.3.5 Ausfahren (der Kontokarte)

Symbolischer Befehlscode: AFA

Wirkung:

Mit diesem Befehl wird die Kontokarte aus dem Kartenschacht ausgefahren. Es wird dabei nichts auf die Magnetbandstreifen aufgezeichnet und es erfolgt keine Schlitzung der Kontokarte.

Symbol

1	2	3	4	5	6	7b	7a	7	8	9	10	11	12	13	14
AFA						X		X							

8.3.6 Einfahren und Lesen

OP						AD									
1	2	3	4	5	6	7b	7a	7	8	9	10	11	12	13	14
X	X				X		X	X							

Symbolischer Befehlscode: ELE

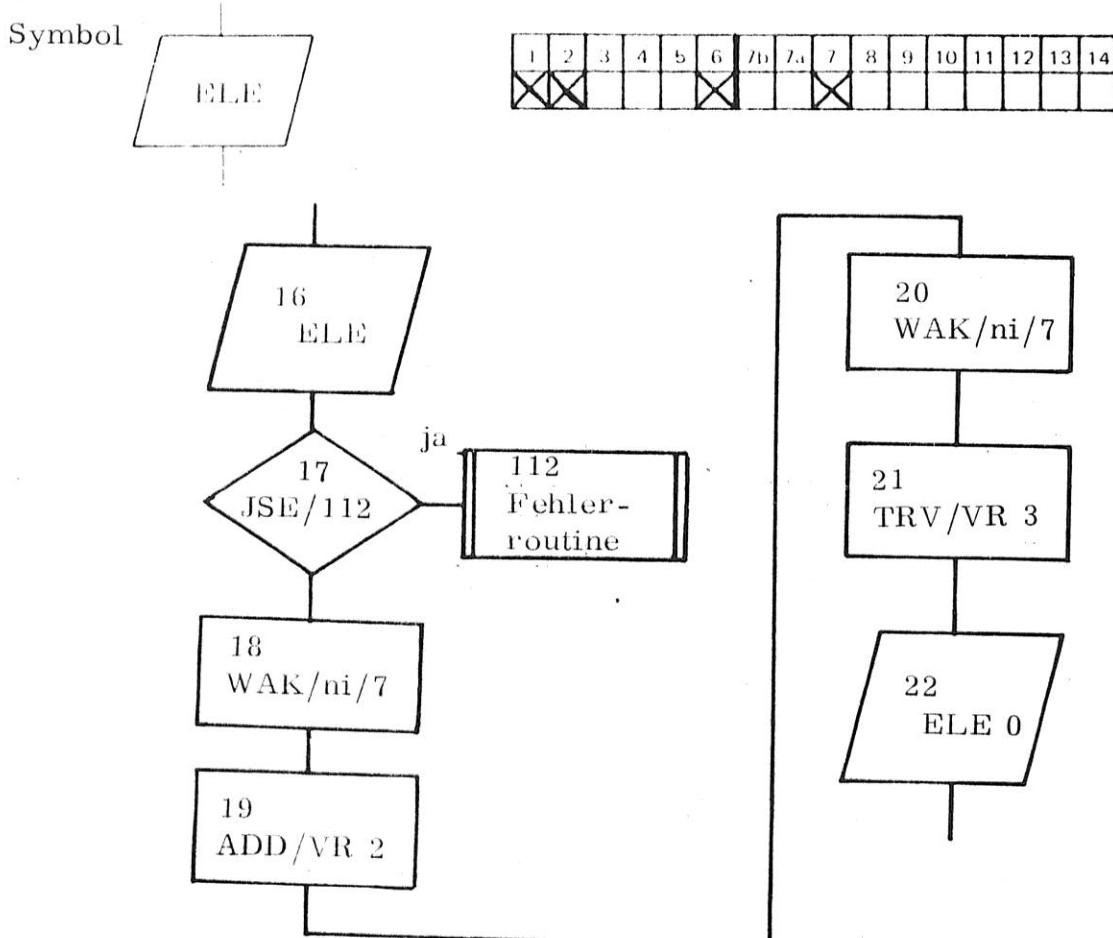
Wirkung:

Mit dem Befehl ELE wird eine Kontokarte in den Kartenschacht eingefahren. Die auf den Magnetbandstreifen stehenden Informationen werden gelesen und in dem Pufferspeicher (64-stellig) abgespeichert. Der Befehl ELE wird nur am Anfang einer Lesebefehlsroutine programmiert.

Die Kontokarte stoppt zeilengerecht durch das Abföhlen der Schlitzungen auf der Kontokarte.

Wird mit dem Befehl ELE eine leere bzw. voll beschriftete Kontokarte eingefahren oder es erfolgt beim Lesen ein Lesefehler, dann wird die Kontokarte wieder ausgefahren und die gelbe bzw. die blaue Lampe leuchtet. Nach Betätigung der Fehlerbehebungstaste (F7) wird in eine zu programmierende Fehlerroutine verzweigt. (Siehe JSE).

Ist eine Kontokarte bereits im Kartenschacht und ein ELE-Befehl ist programmiert, so wird der Programmablauf gestoppt und die Kontokarte wird ausgefahren. Die Maschine ist blockiert. Mit der Fehlerbehebungstaste (F7) wird die Maschinensperre wieder aufgehoben und in eine Fehlerroutine verzweigt.



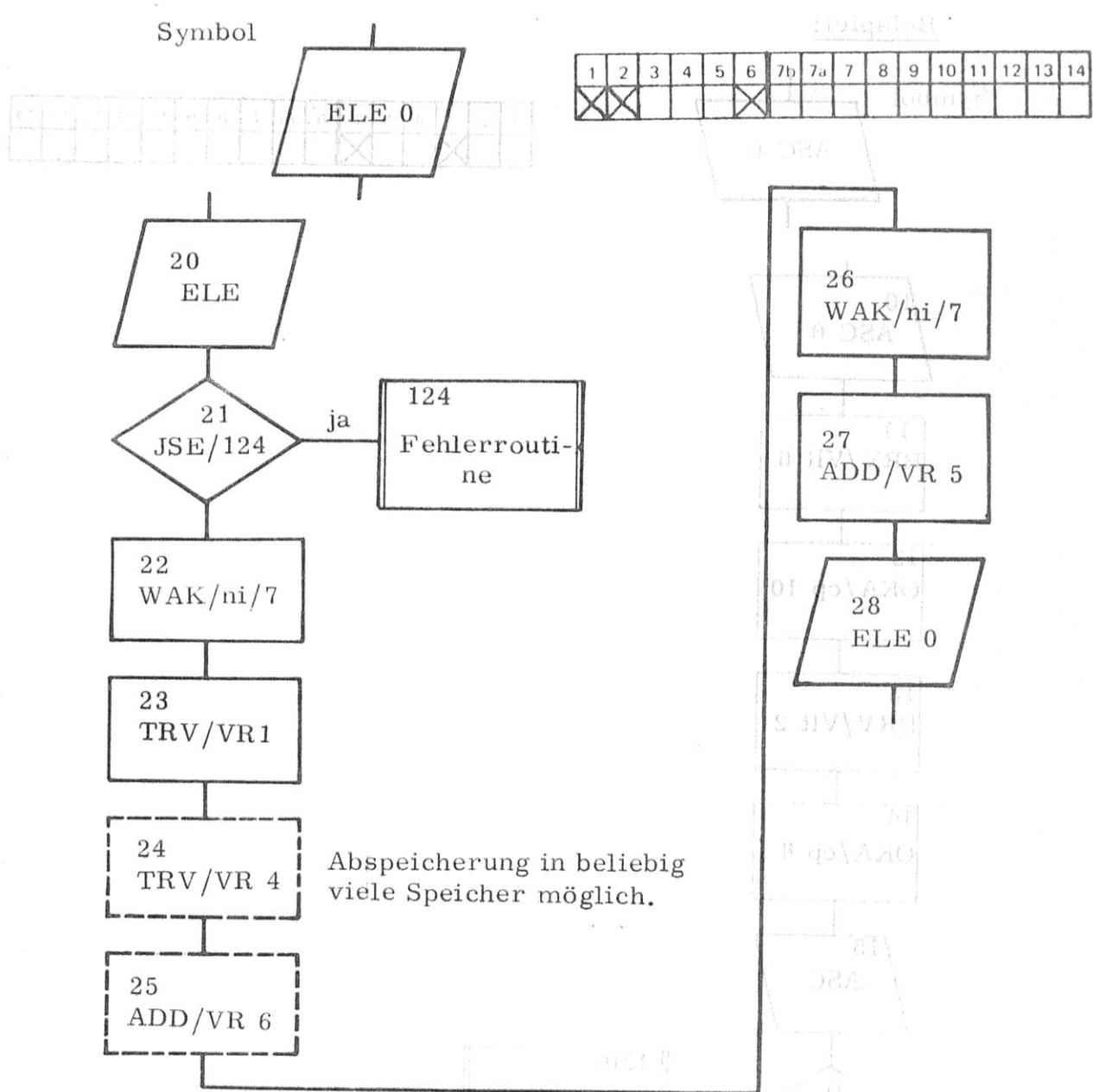
8.3.7 Schlußbefehl, Lesebefehlsroutine

OP						AD									
1	2	3	4	5	6	7b	7a	7	8	9	10	11	12	13	14
X	X				X										

Symbolischer Befehlscode: ELE 0

Wirkung:

Der Befehl ELE 0 ist ein Schlußbefehl mit dem die Übertragung aus dem Pufferspeicher in die Speicher der TA 10/4 beendet wird. Der Befehl ELE 0 muß deshalb am Ende jeder Lesebefehlsroutine programmiert werden.

Beispiel:

ASC 0

S.3.8 Anfangsbefehl Aufzeichnungsroutine

OP						AD									
1	2	3	4	5	6	7b	7a	7	8	9	10	11	12	13	14
		X				X									

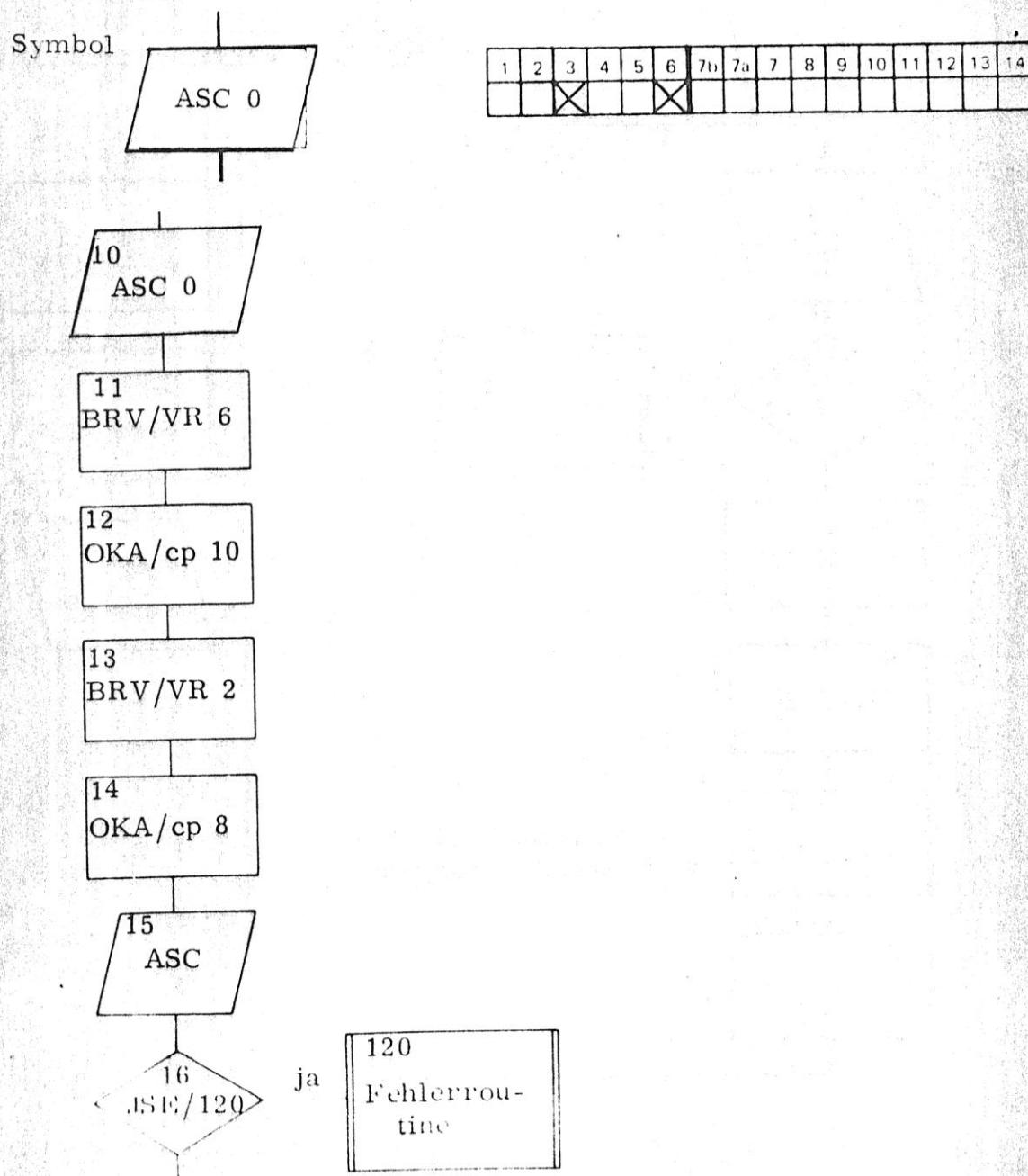
Symbolischer Befehlscode: ASC 0

Wirkung:

Der Befehl ASC 0 ist ein Anfangsbefehl mit dem die Übertragung in den Pufferspeicher vorbereitet wird.

Der Befehl ASC 0 muß daher am Anfang jeder Aufzeichnungsbefehlsroutine programmiert werden.

Beispiel:



8.3.9 Ausfahren und Schreiben

OP						AD									
1	2	3	4	5	6	7b	7a	7	8	9	10	11	12	13	14
		X			X			X							

Symbolischer Befehlscode: ASC

Wirkung:

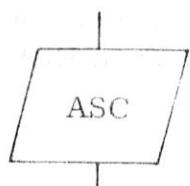
Mit dem Befehl ASC wird die Kontokarte aus dem Kartenschacht ausgefahren. Dabei werden die Informationen aus dem Pufferspeicher auf die Magnetbandstreifen aufgezeichnet. Eine Schlitzung der Kontokarte erfolgt nicht.

Wird bei diesem Befehl das Kontokartenfenster (LS3) nicht rechtzeitig erkannt, so leuchtet die rote Signallampe und es wird in die programmierte Fehlerroutine (JSE) verzweigt. Während des Ausfahrens der Karte werden in diesem Fall keine Informationen auf die Magnetstreifen aufgezeichnet.

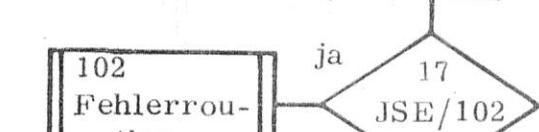
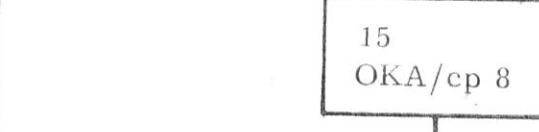
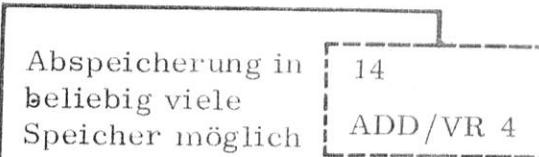
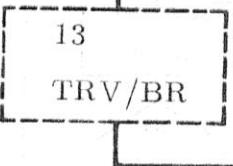
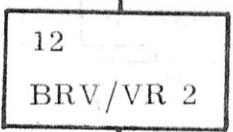
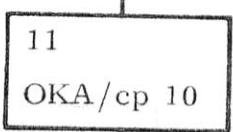
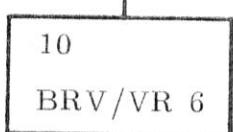
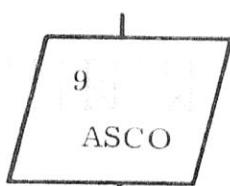
In Fehlerroutinen kann der ASC-Befehl ohne die Befehle ASC 0, BRV und OKA wiederholt werden, wenn inzwischen ELE, ELE 0, SCH, ASC oder ASC 0 nicht abgelaufen sind.

Beispiel:

Symbol



1	2	3	4	5	6	7b	7a	7	8	9	10	11	12	13	14
		X			X			X							



8.3.10 Schreiben (bei Wagenrücklauf)

OP						AD									
1	2	3	4	5	6	7b	7a	7	8	9	10	11	12	13	14
✗		✗			✗										

Symbolischer Befehlscode: SCH

Wirkung:

Mit dem Befehl SCH wird die Kontokarte beim nächsten Wagenrücklauf aus dem Kartenschacht ausgefahren.

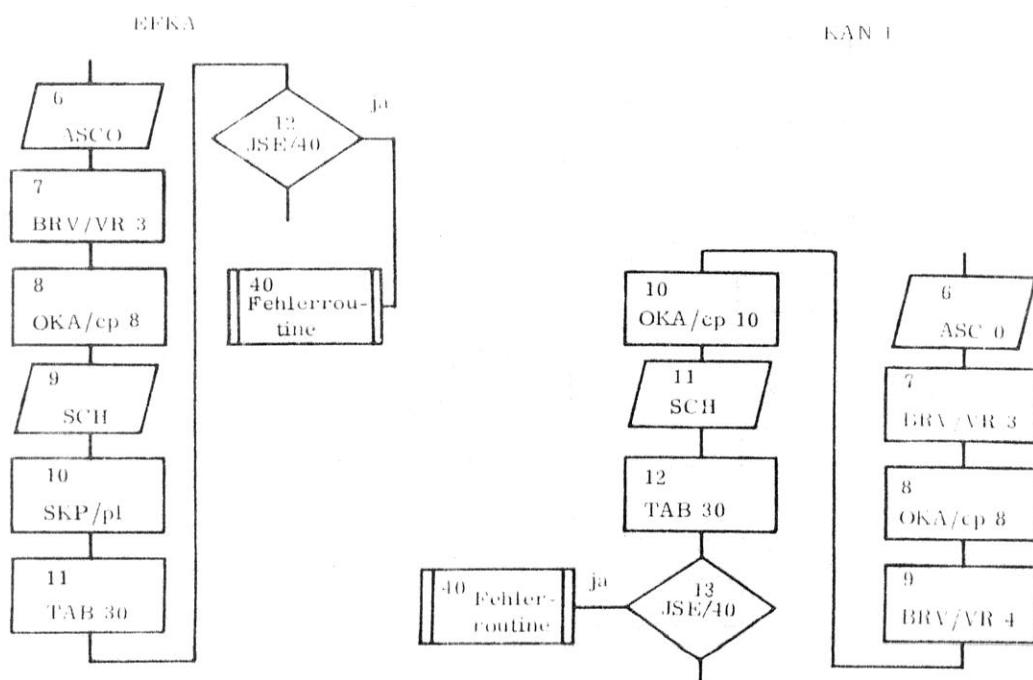
Dabei werden die Informationen aus dem Pufferspeicher auf die Magnetbandstreifen aufgezeichnet und es erfolgt eine Schlitzung der Kontokarte.

Der TAB-Befehl für den Wagenrücklauf muß unmittelbar nach dem SCH programmiert werden. Dazwischen darf nur ein SKP/pl programmiert werden, wenn EFKA angeschlossen ist.

Wird bei diesem Befehl das Kontokartenfenster (LS3) nicht rechtzeitig erkannt, so leuchtet die rote Signallampe und es wird in die programmierte Fehlerroutine verzweigt. Während des Ausfahrens der Karte werden in diesem Falle keine Informationen auf die Magnetstreifen aufgezeichnet.

In Fehlerroutinen kann der ASC-Befehl ohne die Befehle ASC 0, BRV und OKA wiederholt werden, wenn inzwischen ELE, ELE 0, SCH, ASC oder ASC 0 nicht abgelaufen sind.

Beispiel:



8.3.11 Zeilenschaltung

OP						AD									
1	2	3	4	5	6	7b	7a	7	8	9	10	11	12	13	14
	X	X				X									

Symbolischer Befehlscode: ZEI

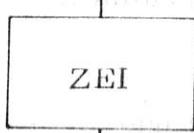
Wirkung:

Mit dem Befehl ZEI und nachfolgenden TAB-Befehl mit Position \leq Istteilung (also Wagenrücklauf) wird die Kontokarte um eine Zeile geschaltet, verbleibt aber im Kontenschacht. Es erfolgt eine Schlitzung der Kontokarte.

Nach ZEI muß unmittelbar der TAB-Befehl folgen. Dazwischen darf nur ein SKP/pl programmiert werden, wenn EFKA angeschlossen ist.

Beispiel:

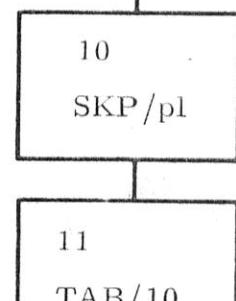
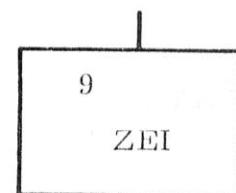
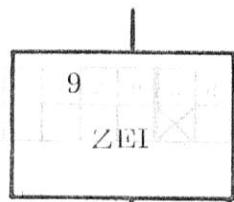
Symbol



1	2	3	4	5	6	7b	7a	7	8	9	10	11	12	13	14
	X	X				X									

KAN 1

EFKA



SAL

8.3.12 Saldenlisten

OP						AD									
1	2	3	4	5	6	7b	7a	7	8	9	10	11	12	13	14
X	X	X			X										

Symbolischer Befehlscode: SAL

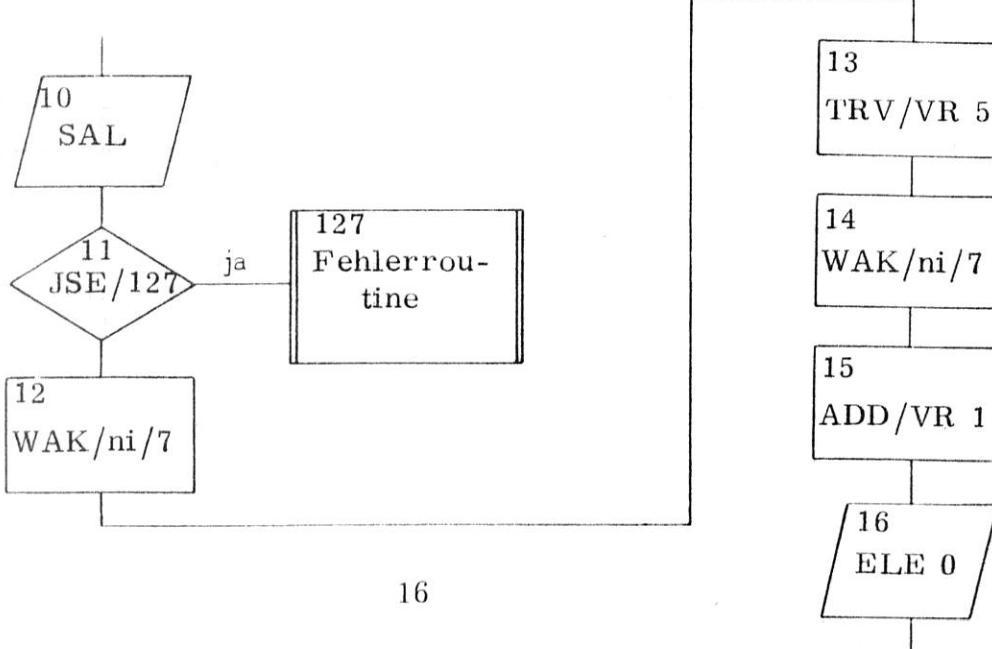
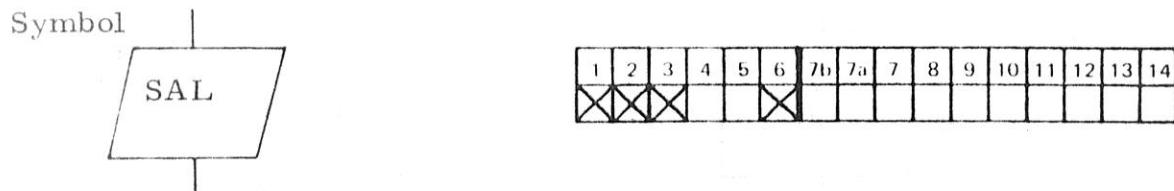
Wirkung:

Mit dem Befehl SAL wird eine Kontokarte in den Kartenschacht eingefahren. Die Informationen von den Magnetbandstreifen werden in den Pufferspeicher gelesen. Anschließend wird die Kontokarte wieder ausgefahren. Die Informationen auf dem Magnetbandstreifen werden dabei nicht verändert. Die Kontokarte wird nicht geschlitzt.

Wird mit dem Befehl SAL eine leere Kontokarte (noch nicht eröffnet oder beschriftet) eingefahren oder es erfolgt beim Lesen ein Lese Fehler, dann wird die Kontokarte im Kartenschacht festgehalten und die gelbe bzw. die blaue Lampe leuchtet. Nach Betätigung der Fehlerbehebungstaste (■) wird in eine zu programmierende Fehlerroutine verzweigt. (Siehe dazu JSE).

Ist eine Kontokarte bereits im Kartenschacht und der Befehl SAL ist programmiert, wird die Kontokarte wieder ausgefahren. Mit der Fehlerbehebungstaste (■) kann dann ebenfalls in eine zu programmierende Fehlerroutine verzweigt werden.

Beispiel:



8.3.13 OUT PUT KA

OP						AD									
1	2	3	4	5	6	7b	7a	7	8	9	10	11	12	13	14
XX			XX	XX								cp 8	cp 4	cp 2	cp 1

Symbolischer Befehlscode: OKA/cp 15-1

Wirkung:

Mit dem Befehl OKA werden die Daten aus dem Tastspeicher (AR) der TA 10/4 in den Pufferspeicher übertragen.

Die Kapazität (B 11 - B 14) wird ohne das Vorzeichen angegeben.
(KP muß größer 0 sein).

Der Befehl OKA darf nur in einer Aufzeichnungsbefehlsroutine programmiert werden.

In der Befehlsroutine kann die Summe der Kapazitäten (cp) der OKA-Befehle incl. Vorzeichen nicht größer als 64 programmiert werden.

Beispiel:

Symbol

OKA/cp 6

1	2	3	4	5	6	7b	7a	7	8	9	10	11	12	13	14
XX			XX	XX								XX	XX		

10
ASC 011
BRV/VR 112
OKA/cp 613
BRV/VR 214
OKA/cp 415
BRV/VR 316
OKA/cp 817
ASC18
JSE/98ja
98
Fehler-
routine

8.3.14 WAIT FOR KEY 7

OP						AD									
1	2	3	4	5	6	7b	7a	7	8	9	10	11	12	13	14
															

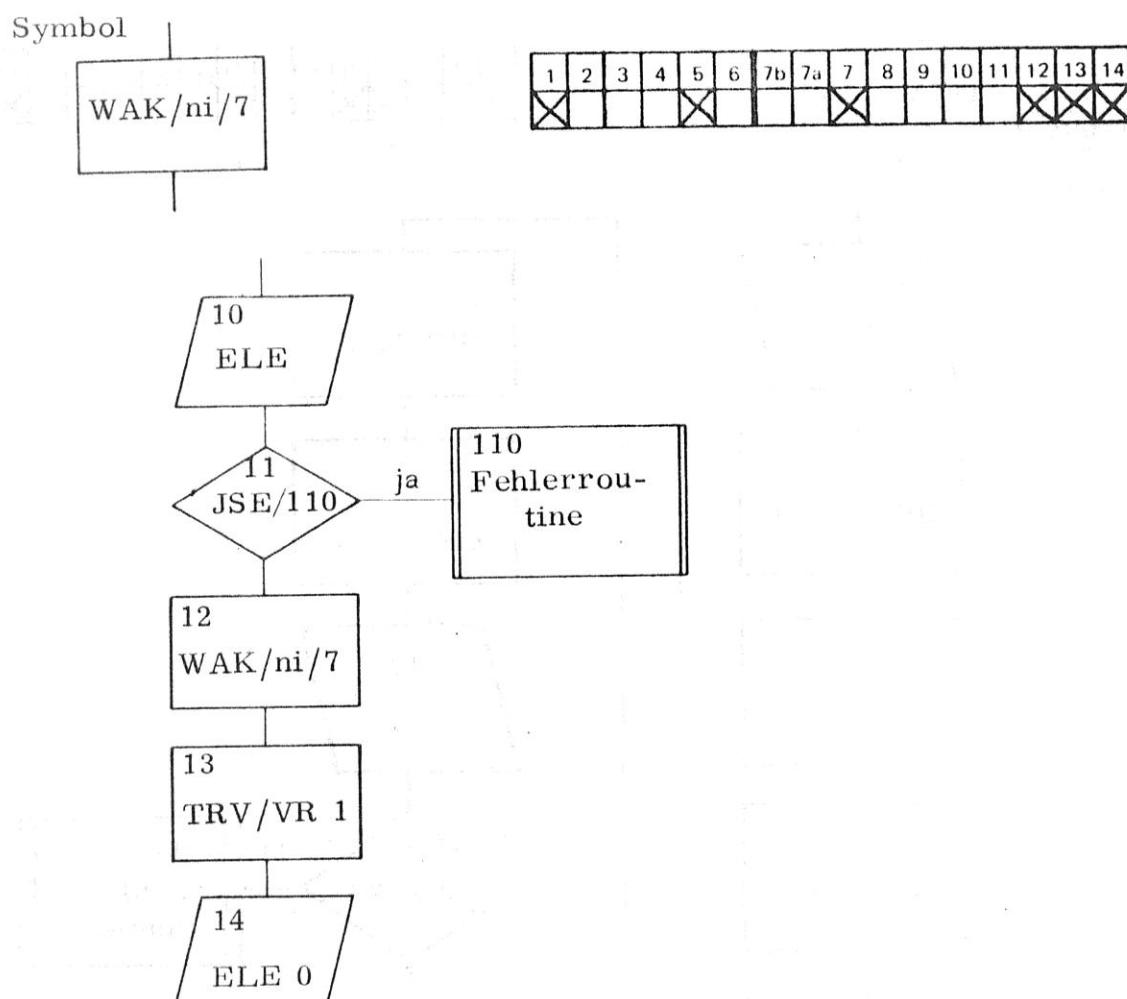
Symbolischer Befehlscode: WAK/ni/7

Wirkung:

Mit dem Befehl WAK/ni/7 werden die Daten aus dem Pufferspeicher in den Tastspeicher AR der TA 10/4 übertragen. Die nicht vorhandene Starttaste 7 wird dabei von der KAN bzw. EFKA-Elektronik simuliert, wenn der Vorzeichendiskriminator ein Vorzeichen (+ oder -) erkennt.

Der Befehl WAK/ni/7 darf nur in einer Lesebefehlsroutine programmiert werden.

Beispiel:



8.3.15 Jump letzte Zeile

OP						AD									
1	2	3	4	5	6	7b	7a	7	8	9	10	11	12	13	14
						ps 512	ps 256	ps 128	ps 64	ps 32	ps 16	ps 8	ps 4	ps 2	ps 1

Symbolischer Befehlscode: JLZ/ps 1023-0

Wirkung:

Dieser Befehl ist ein bedingter Sprungbefehl.

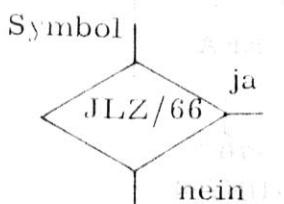
Ist eine Kontokarte im Kartenschacht bis zur letzten Zeile eingefahren, wird in der Maschine ein Merker gesetzt, der von diesem Befehl abgefragt und gelöscht wird.

Ist ein Merker gesetzt, so erfolgt ein Sprung nach dem im Adreßteil dieses Befehles binär verschlüsselt programmierten Befehlswortplatz (letzte Zeile Programmroutine).

Wird in der Programmroutine für letzte Zeile die Karte mit ASC oder SCH ausgefahren, wird die Information auf dem Magnetstreifen gelöscht. ~~Im Pufferspeicher bleibt der Wert erhalten.~~ Ist beim Ausfahren der Karte das Kontokartenfenster nicht erkannt worden, wird der Merker für JSE gesetzt und muß mit JSE abgefragt werden. Programmfolge also ASC o. SCH, (SCH/TAB \leqq Ist,) dann JSE. Wird für JSE keine eigene Routine verwendet, muß der Merker für JSE wie im Beispiel dargestellt, gelöscht werden.

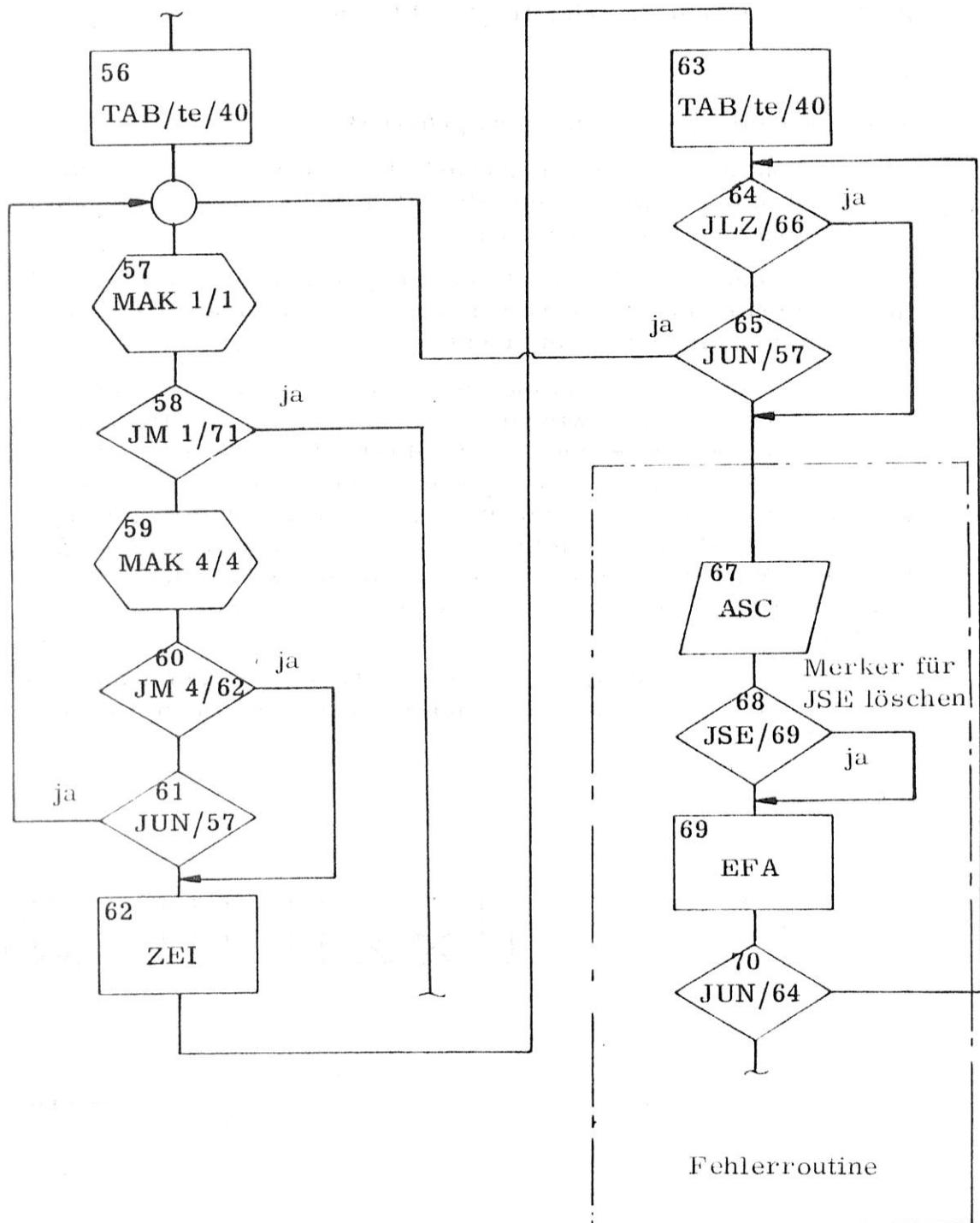
Ist kein Merker gesetzt, so wird der lineare Programmablauf fortgesetzt, d.h. es wird der nächstfolgend programmierte Befehl ausgeführt.

Beispiel:



1	2	3	4	5	6	7b	7a	7	8	9	10	11	12	13	14





8.3.16 Jump if Selector

OP						AD									
1	2	3	4	5	6	7b	7a	7	8	9	10	11	12	13	14
		X	X	X	X	ps 512	ps 256	ps 128	ps 64	ps 32	ps 16	ps 8	ps 4	ps 2	ps 1

Symbolischer Befehlscode: JSE/ps 1023-0

Der Befehl JSE hatte bisher nur Bedeutung bei angeschlossener Peripherie (LSKA).

Da die TA 10/4 keinen LSKA-Anschluß hat, wird der Befehl JSE in Verbindung mit der KAN 1 bzw. EFKA wirksam.

Wirkung:

Der Befehl JSE ist ein bedingter Sprungbefehl.

Abhängig von den Fehlern

- leere Kontokarte oder voll bebuchte Kontokarte
(keine Informationen auf den Magnetbandstreifen)
Die gelbe Lampe leuchtet.
- Lesefehler
(z. B. Magnetbandstreifen zerstört)
Die blaue Lampe leuchtet.
- Kontokarte bereits im Schacht.
Keine Fehleranzeige durch Lampen, aber die Maschine wird blockiert.
- Kontokarte liegt nicht richtig in der Spur.
Die rote Lampe leuchtet.

wird in der Maschine ein Merker (Selector) gesetzt, der von diesem Befehl abgefragt und gelöscht wird.

Ist ein Merker gesetzt, wird die Maschine blockiert. Nach Betätigung der Fehlerbehebungstaste () erfolgt ein Sprung nach dem im Adreßteil des JSE-Befehls binär verschlüsselt programmierten Befehlswortplatz.

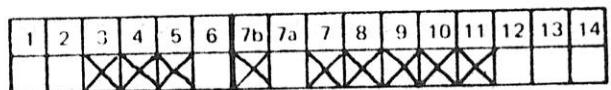
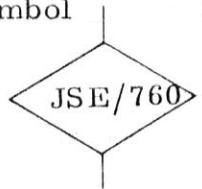
Ist kein Merker gesetzt, so wird der lineare Programmablauf fortgesetzt, d.h. es wird der nächstfolgend programmierte Befehl ausgeführt.

War vor dem Sprungbefehl JSE der Lesebefehl ELE programmiert und ein Merker wird in der Maschine gesetzt, wird die Kontokarte wieder ausgefahren und die Maschine blockiert. Nach Betätigung der Fehlerbehebungstaste wird in eine zu programmierende Fehlerroutine verzweigt.

War vor dem Sprungbefehl JSE der Lesebefehl SAL programmiert und ein Merker wird in der Maschine gesetzt, wird die Kontokarte im Kartenschacht festgehalten und die Maschine wird blockiert. Nach Betätigung der Fehlerbehebungstaste wird in eine zu programmierende Fehlerroutine verzweigt.

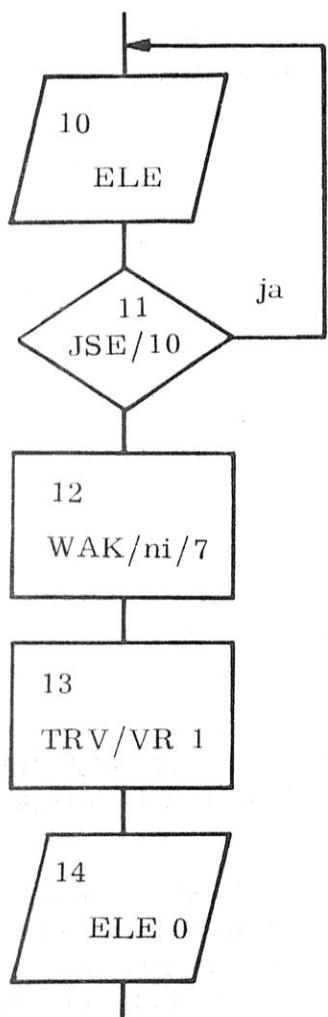
Beispiel:

Symbol

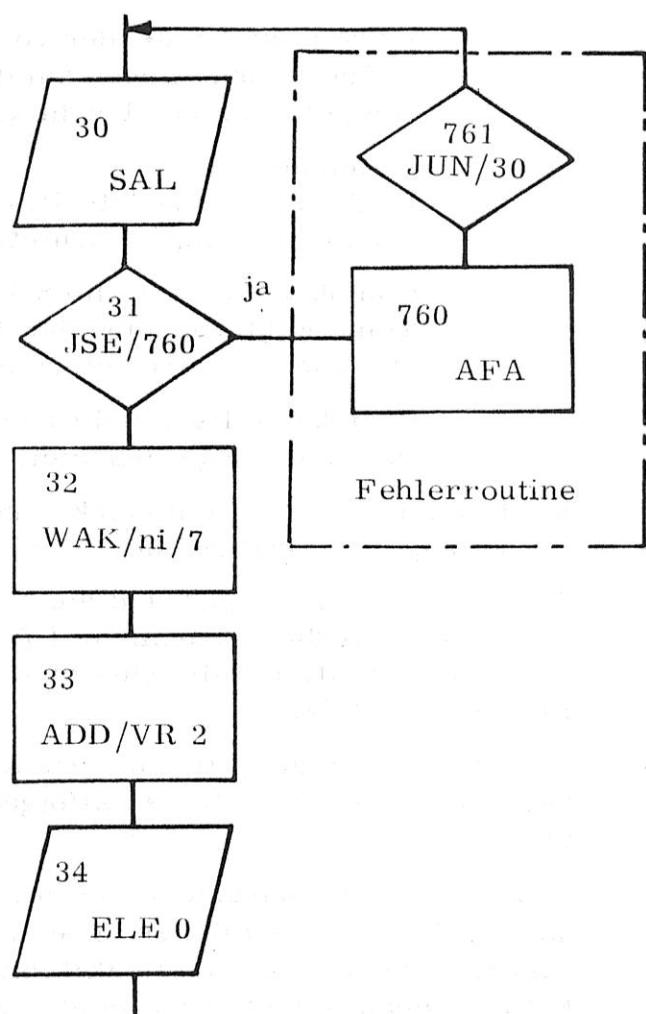


Beispiele für Fehlerroutinen:

bei ELE

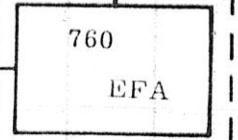
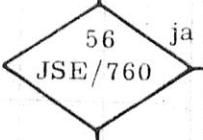
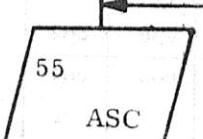
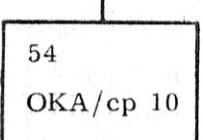
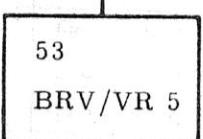
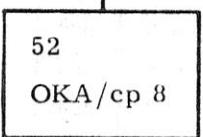
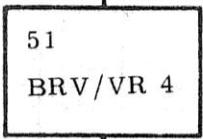
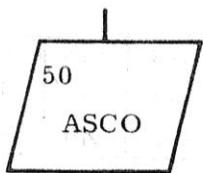


bei SAL



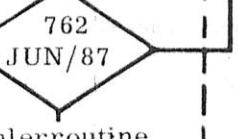
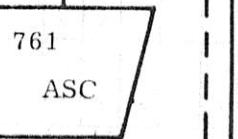
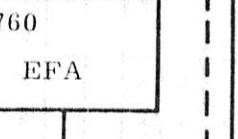
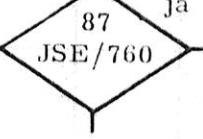
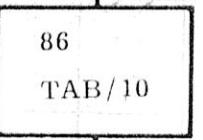
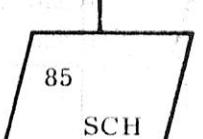
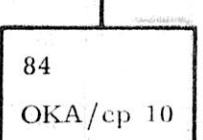
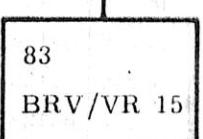
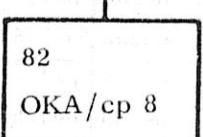
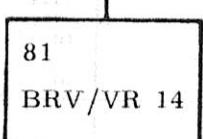
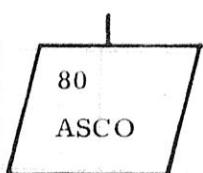
Beispiele für Fehlerroutinen:

bei ASC



Fehlerroutine

bei SCH



Fehlerroutine

8.3.17 Codierliste

No	Mne- moc- nics	Operating part						Address part										
		1	2	3	4	5	6	7b	7a	7	8	9	10	11	12	13	14	
1	TRV	x			x									vr8	vr4	vr2	vr1	
2	BRV	x												vr8	vr4	vr2	vr1	
3	ADD	x	x											vr8	vr4	vr2	vr1	
4	MUL	x		x						1.f.	ro			sr8	sr4	sr2	sr1	
5	WAK	x				x				ni				ky4	ky2	ky1		
6	MAK					x				ni		m4	m2	m1	ky4	ky2	ky1	
7	MAT	x	x	x										vr8	vr4	vr2	vr1	
8	JUN		x					ps512	ps256	ps128	ps64	ps32	ps16	ps8	ps4	ps2	ps1	
9	JM1		x		x			ps512	ps256	ps128	ps64	ps32	ps16	ps8	ps4	ps2	ps1	
10	JM2		x		x			ps512	ps256	ps128	ps64	ps32	ps16	ps8	ps4	ps2	ps1	
11	JM3		x		x	x		ps512	ps256	ps128	ps64	ps32	ps16	ps8	ps4	ps2	ps1	
12	JM4		x	x				ps512	ps256	ps128	ps64	ps32	ps16	ps8	ps4	ps2	ps1	
13	JNE		x	x		x		ps512	ps256	ps128	ps64	ps32	ps16	ps8	ps4	ps2	ps1	
14	JNZ		x	x	x			ps512	ps256	ps128	ps64	ps32	ps16	ps8	ps4	ps2	ps1	
15	JSE		x	x	x			ps512	ps256	ps128	ps64	ps32	ps16	ps8	ps4	ps2	ps1	
16	NOP																	
17	TAB		x							te	pc128	pc64	pc32	pc16	pc8	pc4	pc2	
18	TAK		x		x						pc128	pc64	pc32	pc16	pc8	pc4	pc2	
19	OUP		x	x						ac	np			sy8	sy4	sy2	sy1	
20	OUT			x						sp6	sp4	sp3	sp2	cp8	cp4	cp2	cp1	
21	OUK			x	x					sp6	sp4	sp3	sp2	cp8	cp4	cp2	cp1	
22	CON	x	x	x										co8	co4	co2	co1	
23	SKP				x	x									pl	cf2	cf1	
24	EFA	x				x												
25	AFA		x			x												
26	ELE	x	x			x					x							
27	ELE 0	x	x			x												
28	ASC 0			x		x												
29	ASC			x		x					x							
30	SCH	x	x			x												
31	ZEI		x	x		x												
32	SAL	x	x	x		x								cp8	cp4	cp2	cp1	
33	OKA	x			x	x												
34	WAK/ni/ 7	x			x			ps512	ps256	ps128	ps64	ps32	ps16	ps8	ps4	ps2	ps1	
35	JLZ	x		x	x	x									x	x	x	

Definition:

ac = alphanumeric characters

co = constant

cp = capacity

ky = key

m = mark

ni = numeric input

np = non printing

pc = positions counter

ps = program step

ro = no rounding

sp = space

sr = shift right

sy = symbol

te = text

vr = variable register

1.f = constant 1.factor

Command 19: sy 1-9 1-9

sy % 10

sy %o 11

sy0 12

sy * 13

sy◊ 14

syMW 15

Command 23: cf 1 ≤ skip line

cf 2 ≤ skip tabulat

cf 1+2 ≤ skip page

pl ≤ skip platen