

ZUSE KG · BAD HERSFELD



Elektronische Rechenanlagen

PROGRAMMIERUNGS - KURZFASSUNG FÜR

PERIPHERE GERÄTE DER ZUSE Z 23 V

INTERNE PROGRAMMIERUNG DER ZUSE Z 23 V
FÜR AUSGABE AUF DEM ANELEX-DRUCKER

1. Zum ANelex-Code

Jedem ANelex-Zeichen ist eine Doppeltetrade im Dreixceß-Code zugeordnet. Näheres siehe in den bekannten Zuordnungstabellen.

Ein ZUSE Z 23-Wort faßt daher 5 derart vercodete ANelex-Zeichen; für den Druck einer ANelex-Zeile von 120 Zeichen werden maximal 24 Z 23-Worte notwendig, ergänzt um ein sog. "Formatwort".

2. Vorbereitungen für den Druck einer Zeile

Vor dem Druck ist der ANelex-Puffer in nachstehender Reihenfolge zu füllen:

- a) mit dem Formatwort
- b) mit den auszudruckenden Informationen im ANelex-Code wobei die am weitesten rechts im Druckbild erscheinenden Zeichen zuerst in den Puffer kommen müssen.

Aufbau des Formatworts:

- | | | |
|--|---|---------------------------------------|
| 5 Tetraden mit dem Inhalt Null | } | alle Angaben
im Dreixceß-
Code. |
| 3 Tetraden zur Angabe der Anzahl Zeichen,
die vom linken Rand gerechnet zum
Abdruck kommen sollen. | | |
| 2 Tetraden zur Angabe über die Art des
Zeilenvorschubs | | |

Zeilenvorschub:

Tetraden	Bedeutung
00	kein Zeilenvorschub
01	einfacher Zeilenvorschub
02	doppelter Zeilenvorschub
03	Zeilenvorschub, bis Lochung auf Kanal 1 des Formatlochstr. erscheint
04	" " " " 2 " " "
05	" " " " 3 " " "
06	" " " " 4 " " "
07	" " " " 5 " " "
08	" " " " 6 " " "
09	" " " " 7 " " "

3. Interne Befehlsgebung

Bezeichnung:

- t : Trommelspeicheradresse
s : Schnellspeicheradresse
k : Kernspeicheradresse
a : Akkumulator

Befehl	Wirkung
YHU9	$\langle a \rangle \rightarrow$ Puffer
CU9+o BV0+t	} $\langle t+i \rangle \rightarrow$ Puffer, $i=0, 1, \dots, \langle 13 \rangle$
CU9+o BKVs+o	
CU9+o BKVo+k	} $\langle k+i \rangle \rightarrow$ Puffer, $i=0, 1, \dots, \langle 13 \rangle$
CU9+o OVs+t	
CU9+o OKVs+k	} für $s < 16$: $\langle s \rangle$ op $\langle k+i \rangle$ Puffer $i=0, 1, \dots, \langle 13 \rangle$
YHO9	
YHI9	Puffer löschen
YHB9	Alarmabfrage $o \rightarrow a$, wenn alles richtig -1' $\rightarrow a$, bei Alarm. Ursachen: Auftreten von Pseudotetraden; weniger Zeichen in den Puffer geliefert als im Formatwort angegeben.

Arbeit mit dem ANelex-Druckprogramm

Das Programm übernimmt

- die Übersetzung von Strich- und Gleitkommazahlen in den ANelex-Code
- das Füllen des ANelex-Puffers für den Druck
- das Abdrucken des Pufferinhaltes bei fehlerfreier Übertragung.

Die Ausgabeliste

Bevor das Druckprogramm aufgerufen wird, müssen die in einer Zeile abdruckenden Informationen in einer Ausgabeliste zusammengestellt werden, in dem in aufeinanderfolgenden Speicherzellen (der Trommel oder der Kernspeichererweiterung)

Zahlen in der internen Form abgelegt werden, jede Zahl gefolgt von ihrer Druckanordnung in der nächsten Zelle.

bei Text dessen Anfangsadresse (mit $V=2^{21}$, wenn eine Adresse des Zusatzkernspeichers gemeint ist), ergänzt um OL in den Kernspeicherstellen notiert wird, wiederum gefolgt von einer Druckanordnung.

Text wird vom Druckprogramm nicht in den ANelex-Code umgewandelt, muß also bereits umgeformt vorliegen.

Die letzte Zelle der Ausgabeliste muß eine Information über die Art des Zeilenvorschubs enthalten. Kennzeichen dieser Information ist LO in den Bits 1 und 2. Zahlen in den Trommeladrestellen regeln den Zeilenvorschub, und zwar stimmt die Zuordnung der Zahlen 0 bis 9 zu den Arten des Zeilenvorschubs mit der Zuordnung beim Formatwort überein. Die Zahlen 0 bis 9 werden jedoch im gewöhnlichen Binärsystem in die Trommeladrestellen gesetzt.

Einteilung der ANelex-Zeile

Die 120 Druckstellen einer Zeile werden in 24 Gruppen, genannt "Positionen", zu je 5 Zeichen unterteilt. Die Numerierung der Positionen läuft von links nach rechts.

Druckanordnung

für Zahlen:

- bits 1-20: Nullen
- bits 21-25: Nummer p derjenigen Position, von der ab die Zahl rechtsbündig stehen soll.
- bits 26-30: Anzahl z der Zwischenräume, die zwischen Zahl und rechter Positionsgrenze gelassen werden sollen.
- bits 31-35: Stellen v vor dem Komma
- bits 36-40: Stellen n nach dem Komma

für Text:

- bits 1-20: Nullen
- bits 21-25: Nummer p derjenigen Position, von der ab der Text linksbündig stehen soll.
- bits 36-40: Anzahl der Positionen (=Anzahl der Z 23-Worte) die der zu druckende Text belegt. Zeilenvorschub darf nicht im Text enthalten sein.

Die Besetzung der bits 26-35 hat keinen Einfluß auf die Druckanordnung.

Ausgabe von Gleitkommazahlen

Gleitkommazahlen werden nur in Normalform ausgegeben. Dabei können v und n folgendermaßen gewählt werden:

$$1 \leq v \leq 11, \quad 0 \leq n \leq 9, \quad \text{jedoch } v + n \leq 11$$

Ausgabe von Strichzahlen

Mit $n=0$ werden Strichzahlen als ganze Zahl gedruckt. Hingegen wird bei $n \neq 0$ an entsprechender Stelle ein Komma gesetzt. Für v und n gilt

$$1 \leq v \leq 12, \quad 0 \leq n \leq 11, \quad \text{jedoch } v + n \leq 12$$

Aufruf des Unterprogramms

Im Akkumulator ist die Anfangsadresse der Ausgabeliste anzugeben. Steht die Ausgabeliste in der Kernspeichererweiterung, so ist zusätzlich das V-bit in L zu setzen. Das relativ adressierte Programm kann dann durch Ft aufgerufen werden

INTERNE BEFEHLSGEBUNG DER ZUSE Z 23 V FÜR
BULL-LOCHKARTENDOPPLER

Speicherung der Karteninformationen auf der Trommel

Gearbeitet wird mit 80-spaltigen und 12-zeiligen Lochkarten. Nach Eingabe und vor Ausgabe einer Karteninformation liegt diese folgendermaßen auf der Trommel:

Jede Spalte (12bits) liegt in einer Trommelspeicherzelle (40bits). Eine Lochkarte belegt also 80 hintereinanderliegende Trommelspeicherzellen.

Die Kartenspalte wird unverschlüsselt in die Stelle 25 bis 36 der Trommelspeicherzelle übertragen. Die Stellen 1 bis 24 und 37 bis 40 werden mit Null besetzt.

Bezeichnung: t sei eine Trommeladresse,
k eine Kernspeicheradresse

Lesen einer Lochkarte:

UK 8: bewirkt das Einziehen einer Lochkarte auf die Lesebahn und die Übertragung der Lochkarteninformation in den Puffer. Im Puffer vorhandene Information wird überspeichert.

CKB 13+79 }
UV 8+t } <Puffer> → t + i, i = 0, 1, ..., 79, Puffer löschen.

Stanzen einer Lochkarte:

BK 8: bewirkt das Einziehen einer Lochkarte auf die Stanzbahn und das Stanzen des Pufferinhalts in die Lochkarte.

CKB 13+79 }
BV 8+t } <t + i> → Puffer, i = 0, 1, ..., 79

Bei Kernspeichererweiterung kann die Lochkarteninformation auch in den Zusatzkernspeichern untergebracht werden. Die Blocktransferbefehle UV 8+t, bzw. BV 8+t sind dann zu ersetzen durch

UKV 8 + k, bzw. BKV 8 + k

Beispiele

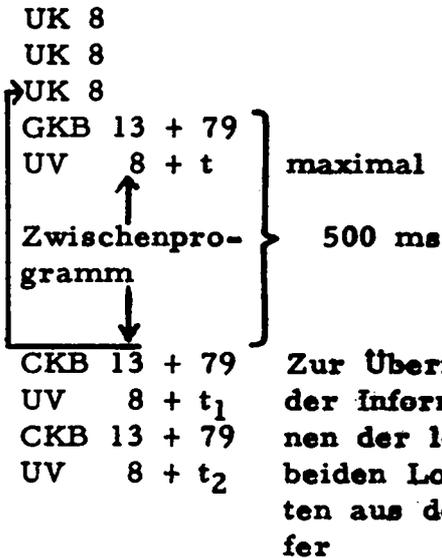
Beim Lesen von Lochkarten kann zwischen schnellem und langsamem Ablauf gewählt werden. Für den schnellen Ablauf sind drei UK 8-Befehle dem Programm voranzustellen, von denen der erste der erstmaligen Füllung des Puffers dient, die beiden weiteren aber vorsorglich die nächsten Lochkarten in die Lesebahn bringen. Soll keine Information verlorengehen, so darf hierbei jedoch die Zeit zwischen zwei Pufferleerungen (einschließlich der ersten Leerung) nicht mehr als 500 ms betragen.

Beim langsamen Lesen wird das Programm nur durch zwei UK 8-Befehle eingeleitet, von denen der erste wiederum zur erstmaligen Füllung des Puffers dient. Jetzt stehen zwischen zwei

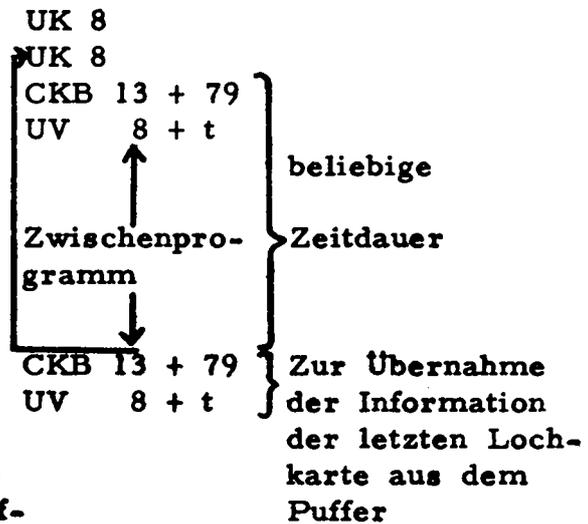
Pufferleerungen beliebige Zeit zur Verfügung.

Unter der Voraussetzung, daß nur ein einziger Kartenstapel zum Lesen in das Magazin des Bull-Dopplers eingelegt wird, programmiert man:

bei schnellem Lesen



bei langsamem Lesen



Muß der Kartenstapel in mehreren Kartenstößen in das Magazin gelegt werden, so lassen sich diese Programme ebenfalls verwenden, jedoch arbeiten sie dann nicht optimal, Näheres möge ausführlichen Beschreibungen entnommen werden.

BEFEHLSCODE DER KERNSPEICHERERWEITERUNG
 DER ZUSE Z 23 V

Bezeichnungen

s	: bisherige <u>Schnellspeicher</u> adresse	$0 \leq s \leq 255$
t	: <u>Trommel</u> speicheradresse	$0 \leq t \leq 8191$
k	: neue <u>Kernspeicher</u> adresse	$1 \leq k \leq 8191$
a	: Akkumulator	
b	: Befehlsregister	

Befehlsstruktur

Bezieht sich ein Befehl auf eine Kernspeicheradresse k, so erscheint diese in den Trommeladressstellen. Ein Zusatz K oder V zum Befehl bewirkt, daß die dort stehende Zahl als Kernspeicheradresse erkannt wird.

Ist k (bzw. t) kleiner als 256, so müssen Einadressbefehle in der Form $0+k$ (bzw. $0+t$) geschrieben werden.

1. Zusatz K: findet immer dann Anwendung, wenn der dadurch entstehende Befehl nicht schon ohne Kernspeichererweiterung mit einer Bedeutung belegt ist.

Beispiele:

Bt : $\langle t \rangle \rightarrow a$, aber BkK : $\langle k \rangle \rightarrow a$
 Bs+t : $\langle t \rangle \rightarrow s$, aber BKs+k : $\langle k \rangle \rightarrow s$

2. Zusatz V:

EVk	:	$\langle k \rangle \rightarrow b$
PEVs+k:		wenn $\langle s \rangle \geq 0$, dann $\langle k \rangle \rightarrow b$
QEVs+k:		wenn $\langle s \rangle < 0$, dann $\langle k \rangle \rightarrow b$
G \bar{O} Vs+k:		1) $(\langle s \rangle + k)_{28\dots 40} \Rightarrow \bar{k}$ $\langle s \rangle \rightarrow s$ G, V löschen, K setzen (bei $\bar{O}=E$ bleibt V und es wird kein K gesetzt)
		2) $\bar{O}K\bar{k}$ ausführen (bei $\bar{O}=E$ wird EV \bar{k} ausgeführt)
CG \bar{O} Vs+k:		1) $(\langle s \rangle + k)_{28\dots 40} \Rightarrow \bar{k}$ $\langle s \rangle + k \rightarrow s$ C, G, V löschen, K setzen (bei $\bar{O}=E$ bleibt V und es wird kein K gesetzt)
		2) $\bar{O}K\bar{k}$ ausführen (bei $\bar{O}=E$ wird EV \bar{k} ausgeführt)

Blocktransfer zwischen Trommel und Kernspeicher

CKB 13+n	}	$\langle t + i \rangle \rightarrow k + i$	für $i = 0, 1, 2, \dots, n$
CUK k			
BVo+t			

$$\left. \begin{array}{l} \text{CKB13+n} \\ \text{CUKk} \\ \text{UVo+t} \end{array} \right\} \langle k + i \rangle \rightarrow t + i \quad \text{für } i = 0, 1, 2, \dots, n$$

Für $O = A, S, US, \emptyset, J, JS$ kann programmiert werden

$$\left. \begin{array}{l} \text{CKB13+n} \\ \text{CUKk} \\ \text{OVs+t} \end{array} \right\} \left. \begin{array}{l} \text{für } s < 16 : \langle s \rangle \text{ op } \langle t+i \rangle \rightarrow k+i \\ \text{für } s \geq 16 : \langle s+i \rangle \text{ op } \langle t+i \rangle \rightarrow k+i \end{array} \right\} \text{für } i = 0, 1, 2, \dots, n$$

Beispiel

Unterprogramm zur Berechnung des Ausdrucks

$$y = (\dots((a_n \cdot x + a_{n-1}) \cdot x + a_{n-2}) \cdot x \dots + a_1) \cdot x + a_0,$$

d.h. also abwechselnde Multiplikation und Addition.

(Berechnung von Polynomwerten nach Horner)

Die Zahlen $a_n, a_{n-1}, \dots, a_1, a_0$ (Koeffizienten) sollen in der angegebenen Reihenfolge in aufeinanderfolgenden Kernspeicherzellen liegen, deren Anfangsadresse $\{a_n\}$ vom Hauptprogramm in Schnellspeicherzelle 101 gespeichert sein soll. Ferner soll das Hauptprogramm -n' (den negativen genommenen "Grad" des Polynoms) in Schnellspeicherzelle 100 und die Zahl x in Schnellspeicherzelle 102 bereitstellen. Der Rücksprung ins Hauptprogramm erfolge nach Speicherzelle $m+1$, wenn von m aus ins Unterprogramm gesprungen wurde!

Das Unterprogramm stehe ab Kernspeicherzelle 150.

Es lautet dann:

o + 150	B5	}	Umspeichern des Rücksprungbefehls
1	UKo+160		
2	GBV101+0		
3	U6	}	$a_n \rightarrow 6$
4	B102		
5	x		$(\dots) \cdot x$
6	CGBV101+1		$a_i \rightarrow a$ für $i=n-1, n-2, \dots, 1, 0,$
7	+		$[(\dots) \cdot x] + a_i$
8	CKA100+1		Zählspeicher um 1 erhöhen
9	QEV100+154		Sprung nach 154, solange $\langle 100 \rangle < 0$.
o + 160	Eo		

INTERNE MAGNETBAND-BEFEHLSGEBUNG FÜR FESTE
BLOCKLÄNGE

Bezeichnungen:

t, t₁, t₂ : Trommeladressen
 b : Blockadresse (b = 1, 2, ..., 8191)
 s : Nummer des Laufwerks (s = 0, 1, 2, 3)
 b_s : Blockadresse b des Bandes im Laufwerk s
 a : Akkumulator

Befehlsgebung:a) Transfer Puffer ↔ Magnetband

Wird ausgelöst durch U7
 und zwar bewirkt dieser Befehl

beim Lesen

d.h., wenn vorher keine Übertragung von Informationen aus der Maschine in den Magnetbandpuffer stattgefunden hat:

Anwahl von b_s und Übertragung der Blockinformation in den Puffer.

beim Schreiben

d.h., wenn vorher eine Übertragung von Informationen aus der Maschine in den Magnetbandpuffer stattgefunden hat:

Anwahl von b_s und Beschreiben des Blocks mit der Pufferinformation.

Um die Anwahl von b_s durch U7 zu ermöglichen, muß zuvor im Akku in den Trommeladreßstellen die Blockadresse b, in den Schnellspeicheradreßstellen die Nummer s des Laufwerks angegeben werden.

b) Transfer Trommel ↔ Puffer

B7 + t } <t+i> → Puffer, (i=0, 1, ..., 127), wenn Puffer
 oder A7 + t } vorher gelöscht wurde

U7 + t <Puffer> → t+i, (i=0, 1, ..., 127), wenn Puffer
 vorher gefüllt wurde.

c) Befehlsfolgen für Transfer Trommel ↔ MagnetbandLesen

CBs+b
 U7
 U7+t

Schreiben

B7+t
 CBs+b
 U7

d) Benutzen des Puffers als Zwischenspeicher

B 7 + t₁ <t₁ + i> → Puffer i=0, 1, 2, ..., 127
 U 7 + t₂ <Puffer> → t₂ + i i=0, 1, 2, ..., 127

e) Schneller Rücklauf

für CDC- und Ampex-Bänder (nur bei 1/2-Zollbändern)	}	Band im Laufwerk s läuft schnell zum Anfang zurück	für Bell-Bänder
CB (16+s) +o U7			CB (32+s) +o U7
			} Band im Laufwerk s läuft schnell zum Anfang zurück

f) Alarm- und Freigabeabfrage

B 7 bringt	}	0 → a, wenn kein Alarm
		-1' → a, wenn Quersummenkontrolle Lese- oder Schreibfehler ergeben hat.
		Bo+o → a, wenn Übertragung noch nicht beendet.
		2' → a, wenn Blockadresse auf Band nicht gefunden
		<u>Nur für Bell-Magnetbandsystem</u>
		3' → a, bei Gerätealarm
		4' → a, wenn wegen eingeschalteter Schreibsperre Band nicht beschrieben werden kann.

g) Kernspeichererweiterung

Ist Kernspeichererweiterung vorhanden, so können die Informationen auch aus Kernspeicherzellen in den Puffer übertragen, bzw. vom Puffer in Kernspeicherzellen gebracht werden.

Die Transferbefehle B7 + t (bzw. A7 + t) und U7 + t sind dann zu ersetzen durch BK7 + k (bzw. AK7 + k) und UK7 + k, wobei k eine Kernspeicheradresse bezeichnet.

Beispiel
=====

Kopieren von 1/2-zölligen Ampex-Bändern, wobei sich das Original im Laufwerk s₁, die Kopie im Laufwerk s₂ befinden soll. Alarmabfragen sind durchzuführen. Wurde die Blockadresse nicht gefunden oder ein Lese- oder Schreibfehler festgestellt, so sind insgesamt 3 Lese- oder Schreibversuche vorzunehmen. Bleibt auch dann der Erfolg aus, so soll gestoppt werden. Nach Beendigung des Kopiervorganges sollen beide Bänder zurückgespult werden.

Das Kopierprogramm stehe ab S.Sp. 100, das Unterprogramm zur Alarmabfrage ab S.Sp. 150.

100 CKNS	2+4	}	Programm auf Anfangsstand setzen.
1 CKØ	112+8191		
2 CKS	112+8190		
3 CKØ	117+8191		
4 LLVCKS	117+8190	}	Zur Gewährleistung, daß der Magnetbandpuffer wirklich gelöscht ist
5 Bo			
6 CKB	13+127		
7 UV	o+t ₁		
8 B	7+t ₁		
9 U	7+t ₁		

ZUSÄTZLICHE BEFEHLSMÖGLICHKEITEN DER
ZUSE Z 23 V

1. Multiplikation von Strichzahlen mit Potenzen von 10

Befehl	Wirkung	
LØo	$\langle a \rangle \cdot 10 \longrightarrow a$	1 Wortzeit
PQLØ1-n	$\langle a \rangle \cdot 10^n \longrightarrow a$	n+1 Wortzeiten
RØo	$\langle a \rangle \cdot 10^{-1} \longrightarrow a$ $\langle 3 \rangle$ um 8 Stellen zyklisch nach rechts verschieben und den Divisionsrest in die untersten vier Stellen von 3 setzen. Bei $\langle a \rangle = 0$ findet nur die zyklische Verschiebung statt, <u>nicht</u> aber das Einsetzen des Restes Null!	Nur gültig für $\langle a \rangle \neq 0!$ 1 Wortzeit
PQRØ1-n	$\langle a \rangle \cdot 10^{-n} \longrightarrow a$; n-malige Behandlung von 3 in der zuvor beschriebenen Weise.	Nur gültig für $\langle a \rangle \neq 0!$ n+1 Wortzeiten (Wenn $\langle a \rangle \neq 0$, meist nur sinnvoll für $n \leq 5$)

2. Schneller Befehl zum Zahlenlesen

Ab Schnellspeicherzelle z soll eine geeignete Entschlüsselungstabelle für Fernschreibzeichen liegen.

Der Befehl CGKLas+z (für s=15, 1. Leser, bzw. s=10, 2. Leser) bewirkt dann

in der 1. Wortzeit $\langle s \rangle + z \longrightarrow \tilde{s} ; \langle b \rangle = \text{CGKLas+z} \longrightarrow c$
 in der 2. Wortzeit
 falls $\langle \tilde{s} \rangle$ ein Befehl ist $\langle \tilde{a} \rangle \longrightarrow b$. Danach dann wieder
 $\langle c \rangle = \text{CGKLas+z} \longrightarrow b$, sofern $\langle \tilde{s} \rangle$
 kein E-Befehl war.
 falls $\langle \tilde{s} \rangle$ eine Zahl ist, $\langle a \rangle \cdot 10 + \langle \tilde{s} \rangle \longrightarrow a$;
 $\langle c \rangle \longrightarrow b$

3. Tabellensuchbefehle (bedingter Blocktransfer)

Für $\Theta = A, S, US, I, IS$ und $s = 2, 3, 5, 6, 11$ kann programmiert werden

H Θ Vs + t
 PP Θ Vs + t
 QQ Θ Vs + t

Wirkung: Der Blocktransfer $\langle s \rangle$ op $\langle t+i \rangle \longrightarrow a$ wird entweder ausgeführt für alle $i = 0, 1, \dots, \langle 13 \rangle$ oder aber schon dann abgebrochen wenn für ein gewisses i

bei Zusatz H $\langle s \rangle \text{ op } \langle t+i \rangle = a = o$
 bei Zusatz PP $\langle s \rangle \text{ op } \langle t+i \rangle = a \leftarrow o$
 bei Zusatz QQ $\langle s \rangle \text{ op } \langle t+i \rangle = a \geq o$

4. Interrupt

Ein Interruptsignal bewirkt

E 255 + 8191 \rightarrow b, $\langle c \rangle \rightarrow c$

Der Befehl E 255 + 8191 hat die Wirkung

$\langle c \rangle \rightarrow 255$, $\langle 8191 \rangle \rightarrow b$

8191 muß den Sprungbefehl auf das Interruptprogramm enthalten; dieses muß durch E 255 (Rücksprung in das unterbrochene Programm) abgeschlossen werden.

Interruptfähigkeit nur bei eingeschalteter Interrupttaste am Bedienungspult. Die Maschine ist nicht interruptfähig, wenn im Befehlsregister ein E-Befehl steht oder ein Befehl, dessen Ausführung mehrere Wortzeiten beansprucht.

Beispiele

- Umwandeln der im Akku als Binärzahl vorliegenden Zahl 1033264 in die binärdezimale Doppeltetraden-Darstellung.

Befehl	$\langle a \rangle$	$\langle 3 \rangle$	$\langle 6 \rangle$	$\langle 16 \rangle$
	1033264	o		
PQRØ1-5	10	3-2-6-4-3		
NU6	o		10	
RØo	o	3-3-2-6-4		
B3	3-3-2-6-4			
NU16	o			3-3-2-6-4 =====
U3		o		
B6	10			
PQRØ1-6	o	o-o-o-1-o =====		

- Zu vorgegebenem Argument x ist ein Funktionswert f(x) aus einer folgendermaßen gespeicherten Tabelle zu suchen:
 ab Trommelspeicherzelle t₁ stehen n+1 Argumente x_j, ab Trommelspeicherzelle t₂ in entsprechender Reihenfolge die zugeordneten Funktionswerte f(x_j).
 Es werde vorausgesetzt, daß x unter den x_j wirklich vorkommt.
 x selbst stehe in Schnellspeicherzelle 3.

CB1 $\langle a \rangle \neq o$, damit Blocktransfer überhaupt beginnt.
 CKB 13+n
 HSV 3+t₁ x = $\langle t+i \rangle \rightarrow a$, bis für ein gewisses i x - $\langle t+i \rangle = o$.
 CKUS 13+(n-1) i - 1 \rightarrow 13
 B 13
 GB 4+t₂ f(x) \rightarrow a

3. Lesen von Strichzahlen

Vereinbarungen: Vorzeichen sollen den Zahlen nachgestellt sein; das positive Vorzeichen soll ausgelassen werden können. Als Trennungszeichen zwischen den einzelnen Zahlen sollen Zwischenraum, Wagenrücklauf, Zeilenvorschub, Pluszeichen oder Apostroph fungieren.

Mehrere Trennungszeichen sollen nicht zu mehrfachen Wegspeichern der aufgebauten Zahl führen. Das Ende der einzulesenden Zahlenfolge werde durch "/" markiert. Überlesen werden sollen "Leer", "Komma", "Ziffer", "Punkt". Als Löschzeichen soll "Buchstabe" wirken.

Die Zahlen sollen in die Schnellspeicherzellen 100 ff eingelesen werden.

Diese Vereinbarungen erfordern die im folgenden grob skizzierte Zuordnungstabelle, die noch nach den Binär-Werten der Fernschreibzeichen geordnet werden muß. Zeichen, die im Beispiel keine Rolle spielen sollen, kann Eo zugeordnet werden. Die Tabelle sei ab Schnellspeicherzelle 48 gespeichert. Sie werden in Schnellspeicherzelle 80 abgeschlossen durch den Befehl HCKGLA 15 + 48.

<u>Fernschreibzeichen</u>	<u>Zugeordnete Information</u>
Zeilentransport, Zwischenraum, Wagenrücklauf, Apostroph	YLVNGKUC 19 + 1
Leer, Ziffer, Punkt, Komma	Ao
Minuszeichen	NS4
"/"	E2051
Buchstabe	LVN80
0, 1, 2, 3, 4, 5, 6, 7, 8, 9	0, 1, 2, 3, 4, 5, 6, 7, 8, 9

Die folgenden drei ab Trommelspeicherzelle 2048 gespeicherten Befehle besorgen das Einlesen:

2048	LVB0
2049	CKB 19 + 99
2050	HCKGLA 15 + 48
2051	Fortsetzung nach "/"