

# BELEGEXEMPLAR

Darf nicht entliehen werden!

Die ZUSE Z 23 V

(Ausgabe Januar 1964)



**ZUSE KG**  
**BAD HERSFELD**

**Elektronische Rechenanlagen**

Die ZUSE Z 23V ist durch einige weitere Befehlsmöglichkeiten gegenüber der Z 23 verbessert worden, die eine Beschleunigung der Programme bringen. Einige der neuen Möglichkeiten sind mehr interner Natur, jedoch sollen hier alle Kombinationen aufgeführt werden.

1. Multiplikation mit 10 für Strichzahlen

LØo bewirkt  $\langle a \rangle' \cdot 10 \rightarrow a$

In einer Wortzeit kann hier der Inhalt des Akkumulators mit 10 multipliziert werden (bisherige Möglichkeit 2 Befehle: LLA4

LA0 )

Dieser Befehl kann auch als Wiederholungsbefehl gegeben werden.

PQLØ1-n Wirkung  $\langle a \rangle \cdot 10^n \rightarrow a$

Für die Ausführung werden n+1 Wortzeiten benötigt.

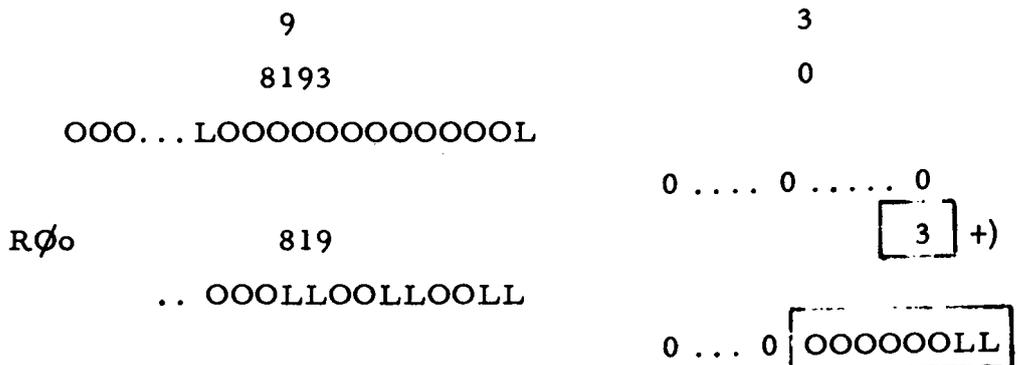
2. Division durch 10 für positive Strichzahlen

Speziell für die Ausgabe auf dem ANelex-Drucker wurde dieser Befehl geschaffen, jedoch kann er auch zu anderen Zwecken benutzt werden.

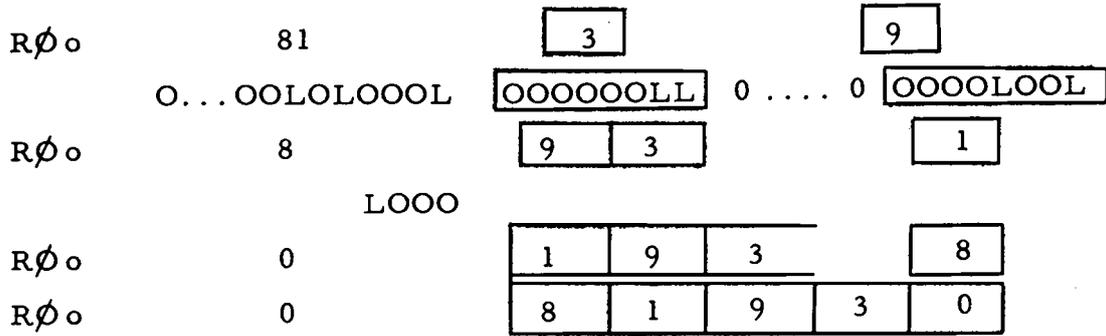
RØo bewirkt  $\langle a \rangle : 10 \rightarrow a$

3 um 8 Stellen zyklisch nach rechts verschieben und den Rest in die letzten 4 Stellen einsetzen.

Ein Beispiel veranschaulicht die Wirkung besser: Im Akkumulator stehen die Zahl 8193 als Binärzahl, eine 0 im Schnellspeicher 3



+) Ein Kästchen umfaßt stets 8 Binärstellen



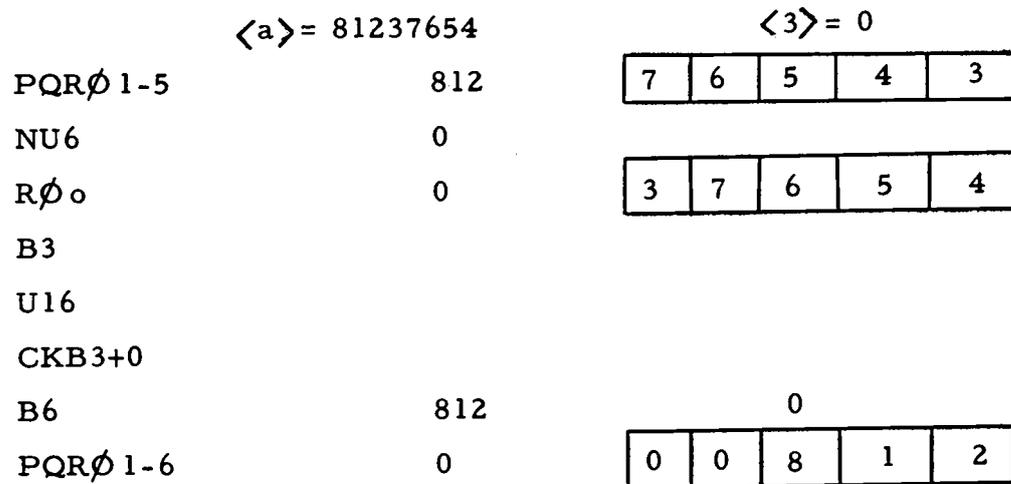
Diese hier aufgeführte Folge von 5 Divisionsbefehlen, die die Aufspaltung der Zahl in die Dezimalbestandteile bewirken, lassen sich auch zu einem Wiederholungsbefehl zusammenfassen:



Ausführungszeit: 6 Wortzeiten

Durch Verschieben um 8 Stellen nach rechts kann man die Ergebnisse auch in den letzten Stellen erhalten. Diese letzte zyklische Verschiebung läßt sich auch mit einem RØo-Befehl durchführen, wenn zu or eine Null in den Akkumulator gebracht wird; das ist besonders dann der Fall, wenn die zu zerlegende Zahl mehr als 5 Dezimalstellen umfaßt.

Beispiel: Aufspalten der als Binärzahl vorliegenden Zahl 81237654 in die Dezimalbestandteile



### 3. Schneller Befehl zum Zahlenlesen

Ein besonderer Befehl gestattet es in einer

#### 1. Wortzeit

ein Entschlüsseln der Fernschreibzeichen über eine im Schnellspeicher stehende Tabelle durchzuführen, sowie in einer

#### 2. Wortzeit

- a) falls das Fernschreibzeichen eine Ziffer (0...9) ist  
    <a> · 10 + eingelesene Ziffer zu bilden,
- b) falls das Fernschreibzeichen ein anderes Zeichen ist, die Ausführung des aus der Tabelle geholten Befehls.

Der Befehl lautet:

CGKLA15+t      bzw.      CGKLA10+t

t ist die Anfangsadresse der Entschlüsselungstabelle, die im Schnellspeicher steht und nach den Wertigkeiten der Fernschreibzeichen geordnet ist (vgl. Beispiel).

#### Beispiel:

Lesen von Festkommazahlen zum Endezeichen "/". Das Vorzeichen soll der Zahl angefügt sein (Zwischenraum, Wagenrücklauf, Pluszeichen, Apostroph oder Zeilenvorschub als Vorzeichen für positive Zahlen, "-" Zeichen für negative Zahlen). Auf ein Minuszeichen muß ein Trennzeichen (Wagenrücklauf oder Zeilenvorschub oder Zwischenraum) folgen. Für die Zeichen, welche im Beispiel nicht berücksichtigt werden sollen, wurde in der Tabelle E0 eingesetzt. Buchstabe bewirkt Löschen der Zahl, in welcher "Bu" vorkommt und weiteres Einlesen. Überlesen werden: Leer, Komma, Ziffer und Punkt. Die Zahlen sollen in die Schnellspeicherzellen 100 ff. eingelesen werden, die Tabelle steht in den Schnellspeicherzellen 48 ff.

Programm:

U2048U

LVB0

CKB19+99

HCKGLA15+48

⋮

Fortgang bei "/"

⋮

Tabelle:

SSpZelle	U48U	Fernschreibzeichen	Binär-Wert (Band-Wert) bei der Eingabe
48	A0	Leer	0
49	3'	3	1
50	YLVNGKUC19+1	Zeilentransport	2
51	NS4	-	3
52	YLVNGKUC19+1	Zwischenraum	4
53	YLVNGKUC19+1	'	5
54	8'	8	6
55	7'	7	7
56	YLVNGKUC19+1	Wagentransport	8
57	E0	+	9
58	4'	4	10
59	E0	;	11
60	A0	,	12
61	E0	[	13
62	E0	:	14
63	E0	(	15
64	5'	5	16
65	YLVNGKUC19+1	+	17
66	E0	)	18
67	2'	2	19
68	E0	10	20
69	6'	6	21
70	0'	0	22
71	1'	1	23
72	9'	9	24
73	E0	x	25
74	E0	]	26
75	A0	Ziffer	27
76	A0	.	28
77	E2051	/	29
78	E0	=	30
79	LVN80	Buchstabe	31
80	HCKGLA15+48		

#### 4. Tabellensuchbefehl

Um optimal aus geordneten oder ungeordneten Listen bestimmte Werte herauszusuchen, gibt es die Tabellensuchbefehle. Sie arbeiten ähnlich wie die Blocktransferbefehle, jedoch mit einer festbleibenden Schnellspeicheradresse ( $s < 16$ ) und sind mit einer Bedingung versehen. Sie laufen solange, bis der Zähler 13 auf Null gezählt hat oder bis die angegebene Bedingung nicht mehr erfüllt ist.

Die Arten der Tabellensuchbefehle.

##### 4.1 Suchen in einer möglicherweise ungeordneten Tabelle nach einem bestimmten Wert

CB1                       $\langle a \rangle \neq 0$ , damit Blocktransfer überhaupt beginnt

CKB13+n

HSV<sub>s+t</sub>                       $s < 16$  d. h. 2, 3, 5, 6 oder 11

Solange  $\langle s \rangle - \langle t+i \rangle \neq 0$  ist ( $i=0, 1, \dots$  bis maximal  $n$ ) wird der Blocktransfer weiter ausgeführt, sobald jedoch  $s \langle s \rangle - \langle t+i \rangle = 0$  ist, bricht der Blocktransfer ab. Aus der Anfangsadresse  $t$  und dem Inhalt des Zählers 13 kann die Adresse ausgerechnet werden, wo  $\langle s \rangle - \langle t+i \rangle = 0$  wurde.

##### 4.2 Besonderheit des Zählers 13

Der Zähler 13 war für die ZUSE Z 23 nur auf 8 Stellen ausgelegt, da für Blocktransfer zwischen Trommel und Schnellspeicher höchstens 256 Wörter übertragen werden konnten. Für die Tabellensuchbefehle können jedoch größere Listen abgesucht werden. Aus diesem Grunde ist Zähler 13 auf 13 Stellen ausgebaut worden.

Mit U13 bzw. CKB13+n kann Zähler 13 mit Zahlen bis 8191 gefüllt werden. Mit A13 bzw. S13 liefert Zähler 13 die letzten 8 Binärstellen zum Rechenwerk. Mit CKI13+8191 können sämtliche 13 Binärstellen vom Zähler 13 in den Akkumulator geholt werden.

##### 4.3 Suchen in einer geordneten Liste

Aufsteigende Werte in der Liste

CB1                       $\langle a \rangle \geq 0$ , damit Blocktransfer beginnt

CKB13+n

PPSV<sub>s+t</sub>                      ( $s < 16$ )

Solange  $\langle s \rangle - \langle t+i \rangle \geq 0$  ist, wird der Blocktransfer weiter ausgeführt bis maximal  $i=n$ . Sobald  $\langle s \rangle - \langle t+i \rangle < 0$  wird, bricht der Blocktransfer ab. Aus dem Inhalt von Zähler 13 und der Anfangsadresse  $t$  kann der Wert der Tabelle ermittelt werden, der erstmalig größer als  $\langle s \rangle$  ist.

Bei dieser und der folgenden Art des Tabellensuchens braucht der Wert, mit dem gesucht wird, d. h.  $\langle s \rangle$  nicht in der Liste vorzukommen.

Absteigende Folge der Werte in der Liste.

CNS1                     $\langle a \rangle < 0$  damit Blocktransfer beginnt

CKB13+n

QQSVs+t                ( $s < 16$ )

Solange  $\langle s \rangle - \langle t+i \rangle < 0$  ist ( $i=0, 1, 2$  bis maximal  $n$ ), wird der Transfer weiter fortgeführt. Sobald  $\langle s \rangle - \langle t+i \rangle \geq 0$  geworden ist, bricht der Blocktransfer ab. Aus dem Stand des Zählers 13 und der Anfangsadresse kann der Tabellenwert ermittelt werden, der erstmals kleiner oder gleich dem Wert im Schnellspeicher  $s$  ist.

4.4 In den oben angeführten Tabellensuchbefehlen wurde stets die Operation S verwendet, sie sind jedoch auch mit anderen Operationen möglich wie A, US, B, NS, I, IS. Stets wird das Ergebnis der Operation in den Akkumulator gebracht, wo es getestet wird. H hat im Falle des Blocktransfers die Bedeutung: es liegt erfüllte Bedingung vor, wenn  $\langle a \rangle \neq 0$  ist.

## 5. Interrupt

Die ZUSE Z 23V ist mit der Möglichkeit des Anschlusses eines Interruptkanals ausgerüstet.

Sobald von außen das Interruptsignal ansteht, wird der Befehlsablauf unterbrochen und ein Sonderprogramm angesprochen, das je nach Bedarf aufgestellt werden kann.

Das Interruptsignal bewirkt, daß nach Ausführung eines Arbeitsbefehls (ein nicht E-Befehl; Blocktransfer- und Wiederholungsbefehle werden erst fertig abgearbeitet) nicht der nächste Befehl geholt wird,

sondern ein Spezialbefehl

E255+8191 ins Befehlsregister gelangt,

während das Befehlszählregister erhalten bleibt. Dieser Befehl bringt

1. den Inhalt des Befehlszählregisters in den Schnellspeicher 255 als Rücksprungbefehl aus dem Interruptprogramm in das unterbrochene Programm
2. den Inhalt der Speicherzelle 8191 ins Befehlsregister, d.h. führt einen Sprung auf diese Zelle aus. In dieser Zelle muß der Sprung in das Interruptprogramm angegeben werden.

Durch die Wirkung des Befehls E255+8191 ist außer der Zelle 255 kein Speicher verändert worden. Im Interruptprogramm selbst können die sonstigen Register und Speicher, die das Programm benötigt, sichergestellt und am Ende wiederhergestellt werden.

Solange das Interruptprogramm läuft, ist keine neue Programmunterbrechung möglich. Die Interruptfähigkeit wird wiederhergestellt durch den Befehl E 255, der den Rücksprung in das unterbrochene Programm bewirkt.

Der Interrupt kann von einem Schalter am Bedienungspult ausgeschaltet werden.

Bad Hersfeld, 8.1.1964

Su/Re