

PROGRAMMGESTEUERTE
ELEKTRONISCHE RECHENANLAGE
Z U S E Z 2 3
Programmierungsanleitung

Ausgabe Februar 1961

Z U S E K G • Bad Hersfeld
Western Germany

**Institut
für angewandte Mathematik
Technische Hochschule Karlsruhe**

V o r w o r t

Die ZUSE Z 23 stellt eine Weiterentwicklung der elektronischen Rechenanlage ZUSE Z 22 dar. Während die ZUSE Z 22 noch auf Röhrenbasis aufbaut, ist die ZUSE Z 23 volltransistorisiert. Die ZUSE Z 23 ist eine im reinen Dualsystem arbeitende Serienmaschine mit einer Wortlänge von 40 Binärstellen (Bits).

Der interne analytische Code der ZUSE Z 22 wurde weitgehend beibehalten, nur an einigen Stellen wurden besondere Kombinationen geschaffen, um den Rechenablauf schneller zu gestalten. Das Prinzip, nur wenige logisch starke Einzelbefehle zu verdrahten, die jedoch in reicher Mannigfaltigkeit kombiniert werden können, wurde von der ZUSE Z 22 übernommen. Daher sind für die Eingabe und Ausgabe und die arithmetischen Operationen Programme in der Maschine gespeichert, die sogenannten Grundprogramme.

Grundsätzlich könnte jeder Benutzer sich einen eigenen Code wählen, wenn er nur ein entsprechendes Grundprogramm aufstellt. Um jedoch den Benutzern diese Arbeit abzunehmen, wird mit der Maschine ein Grundprogramm mitgeliefert, das den mit der ZUSE Z 22 bewährten Freiburger Code benutzt.

In diesem Freiburger Code wird mit Zahlen im gleitenden Komma gearbeitet und zwar mit einer Genauigkeit von 9 Dezimalstellen und einem Zahlenbereich von 10^{-39} bis 10^{+38} . Der Freiburger Code benutzt einige leicht einprägsame Abkürzungen, so sind die Aufrufbefehle für die arithmetischen Programme durch die in der Mathematik gebräuchlichen Zeichen verschlüsselt (+, -, X, :). Der Freiburger Code gestattet außerdem die Ein- und Ausgabe von Klartext, d.h. alphanumerischem Text für Überschriften und Erklärungen usw.

Diese Programmieranleitung behandelt den externen Freiburger Code und geht nur soweit auf den "Intern Code" ein, wie es der Benutzer zum Umrechnen und Aufbauen von Adressen und Zählungen benötigt. Alle Einzelheiten des "Intern Code" werden in einer besonderen Ausarbeitung behandelt.

I n h a l t

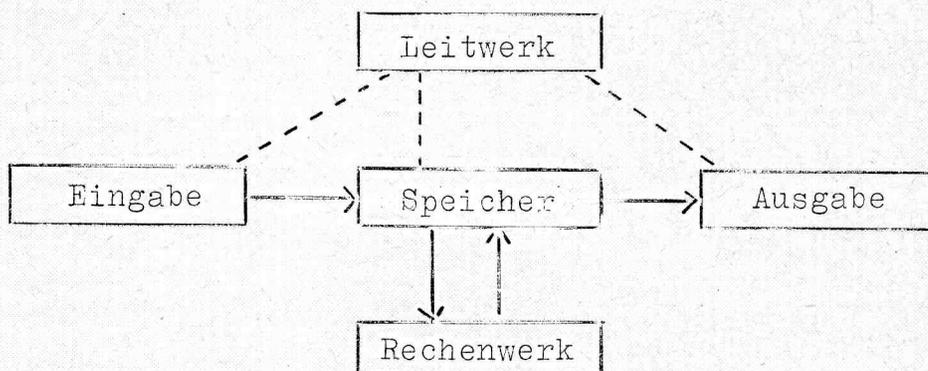
Vorwort	I
1. Einführung	
1.1 Überblick	1
1.2 Ein- und Ausgabe	1
1.3 Leitwerk	2
1.4 Speicher	2
1.5 Rechenwerk	3
2. Freiburger Code	
2.1 Vorbemerkung	4
2.2.1 Befehlsstruktur	4
2 Darstellung von Zahlen	6
3 Strichzahlen	7
4 Klartext	8
2.3 Symbolik	8
2.4 Erklärung der Befehle	10
1 Sprung- und Rufbefehle	10
2 Lese- und Speicherbefehle	14
3 Blocktransfer	17
4 Arithmetische Befehle	18
2.5 Druckbefehl	22
1 Ausgabeform der Zahlen	23
2 Druckanordnung	25
3 Ausgabeform von Befehlen	27
4 Ausgabe von Klartext	28
5 Tabellierbefehle	29
2.6 Bedingungszeichen	31
2.7 Stopbefehl	33
2.8 Banäbefehle	34
3. Beispiel	
3.1 Vorbereitung	36
3.2 Flußdiagramm	37
3.3 Speicherliste	41
3.4 Lochstreifenherstellung (Programmierungstisch)	45
3.5 Eingabe des Lochstreifens (Leseprogramm)	50

3.6 Beschreibung des Bedienungspultes	53
4. Adressenrechnen und Zählen	
4.1 Rechnen mit Strichzahlen	58
4.2 Konstantenzuführung aus dem Befehlsregister	62
4.3 Logische Operationen	65
4.4 Adressensubstitution	
1 Einfache Adressensubstitution	68
2 Adressenmodifikation mit Indexregister, das sich nicht verändert	69
3 Adressenmodifikation mit verändertem Indexregisterinhalt	70
5. Relatives und adressenloses Programmieren	
5.1 Relativadressen	77
5.2 Adressenloses Programmieren (Symbolische Adressen)	82
6. Tabellen	
6.1 Befehlsliste	
6.2 Schnellspeicher	
6.3 Fernschreib-Code	
6.4 Oktalliste	
6.5 Blockschemata der ZUSE Z 23	
6.6 Bedienungspult	

1. Einführung

1.1 Überblick

Die ZUSE Z 23 besitzt im wesentlichen 5 Hauptteile: Die Eingabe, das Leitwerk, das Rechenwerk, den Speicher und die Ausgabe.



Die Ein- und Ausgabe bilden die Vermittlung zwischen den Bedienenden und der eigentlichen Rechananlage.

Alle Programme, d.h. die einzelnen Rechanweisungen für die Maschine und alle Eingangszahlenwerte werden mittels Lochstreifen in die Maschine eingegeben. Die Lochstreifen werden auf einem gesonderten Programmierungstisch hergestellt, der aus einem Fernschreiber mit angebautem Locher besteht.

1.2 Ein- und Ausgabe

Die Eingabe besteht aus einem photoelektrischen Abtaster, der bis zu 300 Zeichen/sec verarbeiten kann. Die eingegebenen Werte werden durch das Leseprogramm in die interne Sprache der Maschine übersetzt und im Speicher aufbewahrt.

Durch das Leitwerk werden die einzelnen Anweisungen, die bei der Eingabe in die Rechananlage zunächst nur gespeichert wurden, vom Speicher geholt und ausgeführt, d.h. es werden Befehle zur Ausführung ins Leitwerk, Zahlen zum Rechnen in

das Rechenwerk transportiert und Ergebnisse wiederum gespeichert oder zur Ausgabe gegeben, wo sie in Klarschrift bzw. auf Lochstreifen festgehalten werden.

Die Ausgabe besteht in der einfachsten Ausführung aus einem Fernschreiber, der mit 10 Zeichen/sec arbeitet, in dem ein Streifenlocher eingebaut ist. Sollen größere Mengen von Zahlen ausgegeben werden, so besteht die Möglichkeit einen Streifenlocher mit 50 Zeichen/sec anzuschließen. Die erhaltenen Streifen können unabhängig von der Maschine an lochstreifengesteuerten Schreibmaschinen ausgedruckt oder aber als Eingabestreifen für den Rechner benutzt werden.

1.3 Das Leitwerk

Das Leitwerk besteht aus 3 Registern, die für die Aufnahme eines Befehls (Befehlsregister), die Entschlüsselung des Befehls (Steuerregister) und den Aufruf des nächsten Befehls (Befehlszählregister) sorgen. Vom Leitwerk werden durch die Adressen die aufgerufenen Speicherzellen angewählt und durch den Operationsteil entsprechende Wege in der Rechanlage geöffnet bzw. geschlossen.

1.4 Der Speicher

Der Speicher besteht aus einer Magnettrommel und einem Schnellspeicher. Die Magnettrommel hat eine Speicherkapazität von 8 192 Worten; jede dieser 8 192 Speicherzellen ist direkt adressierbar und kann eine Zahl, einen Befehl oder einen 7stelligen α -numerischen Text, sogenannten Klartext aufnehmen. Die Zugriffszeit zu einer beliebigen Trommelspeicherzelle beträgt im Mittel 5 msec.

Die 240 Kernspeicherzellen, aus denen der Schnellspeicher besteht, sind jederzeit ohne Zugriffszeit aufrufbar und können ebenso wie die Trommelspeicherzellen je ein Wort aufnehmen. Alle Kernspeicherzellen können auch als Akkumulator, Index- oder Zählregister verwendet werden.

Zwischen Magnettrommel und Schnellspeicher ist ein direkter optimaler Blocktransfer in beiden Richtungen möglich, so daß z.B. ganze Programmteile oder auch Datengruppen in den Schnellspeicher gebracht werden können, um im Schnellspeicher ohne Zugriffszeiten arbeiten zu können. Die Zeit des Blocktransfers beträgt für je 32 Worte 10 msec.

1.5 Das Rechenwerk

Das Rechenwerk besteht aus dem Operationswerk, in dem ganze Zahlen (Strichzahlen) addiert und subtrahiert sowie logische Operationen (Konjunktion, Disjunktion) ausgeführt werden können.

Dieses Operationswerk ist im allgemeinen dem Hauptakkumulator (Schnellspeicherzelle 4) vorgeschaltet, in dem sich vor der Operation der erste Operand und nach der Operation das Ergebnis befindet. Das Operationswerk kann jedoch unter Beibehaltung des Inhalts des Hauptakkumulators jeder anderen Kernspeicherzelle vorgeschaltet werden, wobei diese den ersten Operanden liefert und das Resultat wieder aufnimmt. Durch ein Befehlszeichen (N) kann der erste Operand auf Null gesetzt werden. Speziell die Kernspeicherzelle 4 kann vor Zuführung zum Operationswerk links (\bar{L}) oder rechts (R) verschoben werden.

2. Der Freiburger Code

2.1 Vorbemerkung

Da in der ZUSE Z 23 nur Weniges fest verdrahtet ist, wird das Meiste von Programmen ausgeführt.

Als sogenannte Grundprogramme wird mit der Maschine mitgeliefert:

Das Leseprogramm, das die vom Lochstreifen abgetasteten Buchstaben und Ziffern in die interne Sprache der Maschine übersetzt,

die arithmetischen Programme für die Ausführung von Addition, Subtraktion, Multiplikation, Division und Wurzelziehen im gleitenden Komma,

das Druckprogramm, das die Ergebnisse ins Dezimalsystem rückübersetzt und auf die Ausgabeeinheit gibt.

Diese Grundprogramme sind in den Trommelzellen 0 bis 1024 gespeichert und durch Sperrschalter vor versehentlichem Überschreiben geschützt.

2.2.1 Die Befehlsstruktur

In der ZUSE Z 23 besteht jeder Befehl aus zwei Bestandteilen.

1. Der Operationsteil, dargestellt durch einen oder mehrere Buchstaben, die angeben, was ausgeführt werden soll, bzw. auch unter welchen Bedingungen der Befehl ausgeführt werden soll.
2. Aus einem Adressenteil, der durch eine oder mehrere Ziffern dargestellt wird und angibt, welche Speicherzelle bei der im Operationsteil angegebenen Operation aufgerufen werden soll. Im Adressenteil können zwei Adressen auftreten, als erste eine Kernspeicheradresse und als zweite eine Trommeladresse; diese Adressen werden durch ein +Zeichen voneinander getrennt. Alle

Einzeladressen unter 256 werden automatisch vom Leseprogramm als Kernspeicheradressen im Befehl eingesetzt. Meint man dennoch eine Trommelspeicheradresse unter 256, so wird durch Einfügen von einer Ziffer "Null" und einem +Zeichen zwischen dem Operationsteil und der Trommelspeicheradresse eindeutig eine solche auch unter 256 erkannt. (Z.B. Sprung auf die Trommelzelle 6 Eo+6).

Ausnahmen von diesem Aufbau eines Befehls aus Operations- und Adressenteil bilden die vercodeten Rufbefehle für die arithmetischen Operationen (+, -, X, :, usw). Auch diese Operationen können unter Bedingung gestellt werden, jedoch muß hier unbedingt die Reihenfolge

Bedingungszeichen - übriger Befehl

eingehalten werden, die auch bei den übrigen Befehlen wegen der Übersichtlichkeit eingehalten werden sollte.

Zur internen Unterscheidung in der Maschine werden die ersten beiden Stellen benutzt

- 00 Kennzeichen für positive Zahlen
- 01 Kennzeichen für Klartext und Stop
- 10 Kennzeichen für Befehle
- 11 Kennzeichen für negative Zahlen

Die Aufteilung eines Befehles im Inneren der Maschine hat folgende Gestalt:

2 Bit	5 Bit	12 Bit	8 Bit	13 Bit
-------	-------	--------	-------	--------

Kenn- Bedingungs- Operations- Schnellspeicher - Trommel-
zeichen zeichen zeichen Adresse

2.2.2 Die Darstellung von Zahlen

Im allgemeinen werden im Freiburger Code die Zahlen in gleitendem Komma verarbeitet. Hier wird die halblogarithmische Darstellung verwendet und als Trennung zwischen der Mantisse und dem folgenden Dezimalexponent ein Schrägstrich eingeführt.

Z.B. $3,1496/+3$ als Darstellung für die Zahl 3149,6

Die Mantisse einer Zahl ist in der Maschine mit 9 Dezimalstellen angegeben und der Exponent mit 2 Dezimalstellen mit einem Bereich von 10^{-39} bis 10^{+38} .

Man hat jedoch eine weitgehende Freiheit in der Form, wie man die Zahlen in die Maschine eingibt. Es ist durchaus nicht notwendig, daß man die Zahlen in der halblogarithmischen Form schreibt und man braucht auch nicht immer die volle Stellenzahl einzugeben,

z.B. die Zahl $+7683,850$ kann in dieser Form eingegeben werden, als $7683,85$ oder $7,68385/3$ oder $,768385/4$

Positive Vorzeichen der Zahl und des Exponenten können weggelassen werden, negative müssen angegeben werden. Ist die Zahl bzw. die Mantisse kleiner als 1, so braucht man die Null vor dem Komma nicht miteinzugeben.

Die zu einer Zahl gehörigen Zeichen müssen zur Eingabe direkt hintereinander ohne Abstand gegeben werden, da ein Zwischenraum oder Wagenrücklauf oder Zeilentransport als Schlußzeichen für eine Zahl (für ein Wort überhaupt) aufgefaßt wird. Ein Zwischenraum in einer Zahl würde also die getrennten Teile für sich als Zahl kennzeichnen. Als größte Zahl kann $,1/39$ eingegeben werden (mit 9 Ziffern Mantisse größte Zahl $,134217727/39$).

Werden größere Zahlen eingegeben, so schreibt die Maschine aus BUEB Eo+285 wobei BUEB = Bereichsüberschreitung bedeutet und Eo+285 die Stelle des Leseprogramms, wo die Bereichsüberschreitung festgestellt wurde.

Die als kleinste Zahl einzugebende Zahl ist 2^{-38} (mit 9 Dezimalstellen Mantisse $,214748364/38$) Kleinere Zahlen werden beim Einlesen als 0 in die Maschine aufgenommen.

Eine Zahl x wird in der Maschine in ein Produkt aus Mantisse \hat{x} , deren Betrag zwischen 0,5 und 1 liegt ($0,5 \leq |\hat{x}| < 1$), und einer Dualpotenz mit dem Exponenten \bar{x} zerlegt derart, daß

$$\pm x = \pm \hat{x} \cdot 2^{\bar{x}} \text{ ist.}$$

Die Aufteilung auf die Dualstellen ist folgende:

- 2 Bits als Vorzeichenstellen (+ oder -)
- 30 Bits für die Mantisse \hat{x}
- 8 Bits für den um 128 erhöhten Exponenten $\bar{x} + 128$

Die fiktive Erhöhung des Dualexponenten um 128 wurde zur Vermeidung einer gesonderten Vorzeichenkennzeichnung vorgenommen. Mit 8 Bits ist im Dualsystem für $\bar{x} + 128$ ein Bereich von 0 bis 255, d.h. für \bar{x} von -128 bis 127 möglich. Dem Zahlenbereich von 2^{-128} bis 2^{+127} entspricht im Dezimalsystem der obengenannte Bereich von 10^{-39} bis 10^{+38} .

Negative Zahlen werden durch das Komplement der Mantisse dargestellt, wobei auch die beiden Kennzeichenstellen mit einbezogen sind.

Zu der Form, in der die Zahlen aus der Maschine ausgegeben werden, vergleiche man Kapitel 2.5.

2.2.3 Strichzahlen

Neben diesen Gleitkomma-Zahlen kann man für Zählzwecke und zum Adressenrechnen noch eine andere Art von Zahlen eingeben und in der Maschine verarbeiten. Diese Zahlen werden bei der Eingabe und der Ausgabe mit einem Apostroph nach der letzten Ziffer gekennzeichnet und deshalb als "Strichzahlen" bezeichnet. Die Strichzahlen können nur ganze Zahlen sein und stehen in der Maschine in den niedrigsten

Stellen (in denen bei Befehlen die Trommeladressen stehen).
Über das Rechnen mit diesen Zahlen vergl. Kapitel 4.

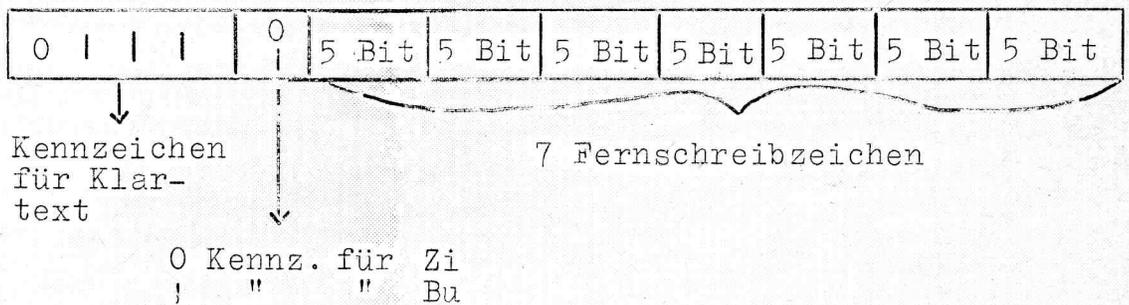
2.2.4 Klartext

Als weitere Wortart existiert in der Maschine der sogenannte Klartext. Hier werden die einzelnen Fernschreibzeichen ohne irgend eine Umsetzung hintereinander, jeweils 7 Zeichen in einem Wort gespeichert.

Der Klartext dient dazu, Überschriften und Erläuterungen zu den Ausgabewerten zu setzen. Sie werden vom Programm an den entsprechenden Stellen ausgegeben.

Zur Eingabe wird der Text am Anfang und am Ende mit einem \mathcal{R} -Zeichen gekennzeichnet. Auch bei der Ausgabe wird der Text zusammenhängend geschrieben und mit der Klartextkennzeichnung \mathcal{R} am Anfang und \mathcal{R} am Ende versehen.

Die interne Aufteilung eines Klartextwortes hat folgende Gestalt:



2.3 Symbolik

Um mit einer kurzen und prägnanten Formulierung die Wirkung der einzelnen Befehle darstellen zu können, soll eine Symbolik eingeführt werden.

Symbol	Erläuterung
a	der Akkumulator
b	das Befehlsregister, das die einzelnen Befehle der Reihe nach aufnimmt und parallel an das Steuerregister weitergibt.
r	Steuerregister, von wo aus die einzelnen Stromtore in der Maschine geöffnet bzw. geschlossen werden, damit die den Befehlen entsprechenden Wege in der Maschine freigegeben werden.
c	Befehlszählregister, das zur Aufnahme der Adresse des Folgebefehls bestimmt ist.
m, n	allgemeine Adressen (Schnellspeicher- oder Trommeladresse)
s	eine Schnellspeicheradresse
t	eine Trommeladresse
<m>	Das Wort das in der Speicherzelle steht, die mit der Adresse m aufgerufen werden kann; kurz: Inhalt von m
{x}	Adresse der Speicherzelle, in der der Wert x steht
<m> _{p...q}	Inhalt der Dualstellen p bis q der Speicherzelle m; d.h. es ist hier nicht der gesamte Speicherinhalt von m gemeint, sondern nur der Teil, der in den bezeichneten Dualstellen steht. Hierbei ist eine Numerierung der Speicherzelle von der Vorzeichenstelle anfangen vorausgesetzt.

1	2	3	4	...	19	20	27	28	40
---	---	---	---	-----	----	----	----	----	----

Kenn- zeichen	Bedingungs- zeichen und Operations- teil	Schnell- speicher- Adressen	Trommel- speicher- Adressen
------------------	---	-----------------------------------	-----------------------------------

Symbol	Erläuterung
	Speziell sei eingeführt
$\langle m \rangle_t$	Inhalt der Trommeladressenstellen der Speicherzelle m, also die Stellen 28 bis 40
$\langle m \rangle_s$	Inhalt der Schnellspeicheradress-Stellen der Speicherzelle m, also die Stellen 20 bis 27
$\dots \rightarrow \dots$	Transportsymbol ("geht nach"), das bedeutet, daß der links angegebene Wert an rechts angegebene Stellen transportiert wird, z.B.
$\langle m \rangle \rightarrow a$	der Inhalt der Speicherzelle m wird in den Akkumulator gebracht kurz: Inhalt von m geht nach Akku
$\dots \Rightarrow \dots$	"ergibt" Zeichen. Hier wird links vom Doppelpfeil angegeben, nach welchen Vorschriften aus bekannten Größen der rechts stehende Wert gebildet werden soll, z.B.
$k+1 \Rightarrow k$	Der um 1 erhöhte bisherige Index k "ergibt" für den nächsten Durchlauf den neuen Index k.
$x + \Delta x \Rightarrow x$	Das bisherige Argument x um die Schrittweite Δx verändert "ergibt" das neue Argument.
k'	Eine ganze Zahl aufbauend auf der untersten Dualstelle (Stelle 40)

2.4 Erklärung der Befehle

2.4.1 Sprung- und Rufbefehle

Der Sprungbefehl wird durch das Operationszeichen E dargestellt.

E_m $\langle m \rangle \rightarrow b$ $E(m+1); \rightarrow c$

4. Der Sprungbefehl hat zwei Wirkungen

- 1) wird durch den Sprungbefehl der in der angegebenen Zelle m stehende Befehl zur Ausführung ins Befehlsregister geholt
- 2) wird der Abrufbefehl E_{m+1} im Befehlszählregister bereit gestellt.

Durch diesen im Befehlszählregister stehenden Abrufbefehl wird dann während bzw. nach der Ausführung des aus der Speicherzelle m geholten Befehls der folgende Befehl aus der nächsten Speicherzelle ins Befehlsregister geholt. Da immer eine Erhöhung der Adresse um 1 erfolgt, ist gewährleistet, daß die Befehle in der Reihenfolge der Speicherzellen ausgeführt werden. Diese Folge wird unterbrochen, sobald aus einer Speicherzelle ein Sprungbefehl auf eine andere Stelle ins Befehlsregister geholt wird, der dann einen entsprechenden Ablauf von der anderen Stelle an auslöst.

Falls die Adresse des F -Befehls eine Trommeladresse ist, so wird in den Trommeladressen eine Eins (also $1'$) zuaddiert, falls es eine Schnellspeicheradresse ist, so wird beim Übergang vom Befehlsregister zum Befehlszählregister in den Schnellspeicheradressenstellen eine Eins (als $8192' = 2^{13}$) zuaddiert. Durch diese Organisation sind Programmabläufe auf der Trommel oder im Kernspeicher möglich.

Befehl	Wirkung	
allgemein		
E_m	$\langle m \rangle \rightarrow b$	$E_{m+1} \rightarrow c$
speziell		
E_t	$\langle t \rangle \rightarrow b$	$E(t+1') \rightarrow c$
E_s	$\langle s \rangle \rightarrow b$	$E(s+1) \rightarrow c$

B. Der Rufbefehl, dargestellt durch das Operationszeichen F, ist ebenfalls ein Sprungbefehl, jedoch mit dem Zusatz, daß der E-Befehl, der den F-Befehl ins Befehlsregister gerufen hat und mit um 1 erhöhter Adresse im Befehlszählregister steht, in Schnellspeicherzelle 5 notiert wird; d.h. es wird der Abrufbefehl für die nächste Speicherzelle nach dem F-Befehl aufbewahrt in Schnellspeicherzelle 5. Diese F-Befehle dienen zur Einschaltung von Unterprogrammen an beliebiger Stelle des Programms. Der Aufrufbefehl bringt mit dem Sprung auf ein Unterprogramm gleichzeitig den Rücksprungbefehl zum übergeordneten Programm in die Speicherzelle 5.

$F_m \quad \langle c \rangle \rightarrow 5 \quad \langle m \rangle \rightarrow b \quad E(m+1) \rightarrow c$

Da die Rücksprungbefehlsnotierung nur einmal erfolgen soll, wird vor der Übernahme des Befehls F_m vom Befehlsregister ins Befehlszählregister der F-Befehl in einen E-Befehl verwandelt.

Bei mehrfacher Ineinanderschachtelung von Unterprogrammen muß man den Rücksprungbefehl vor Aufruf eines neuen Unterprogramms aus Speicherzelle 5 an eine andere Stelle umspeichern.

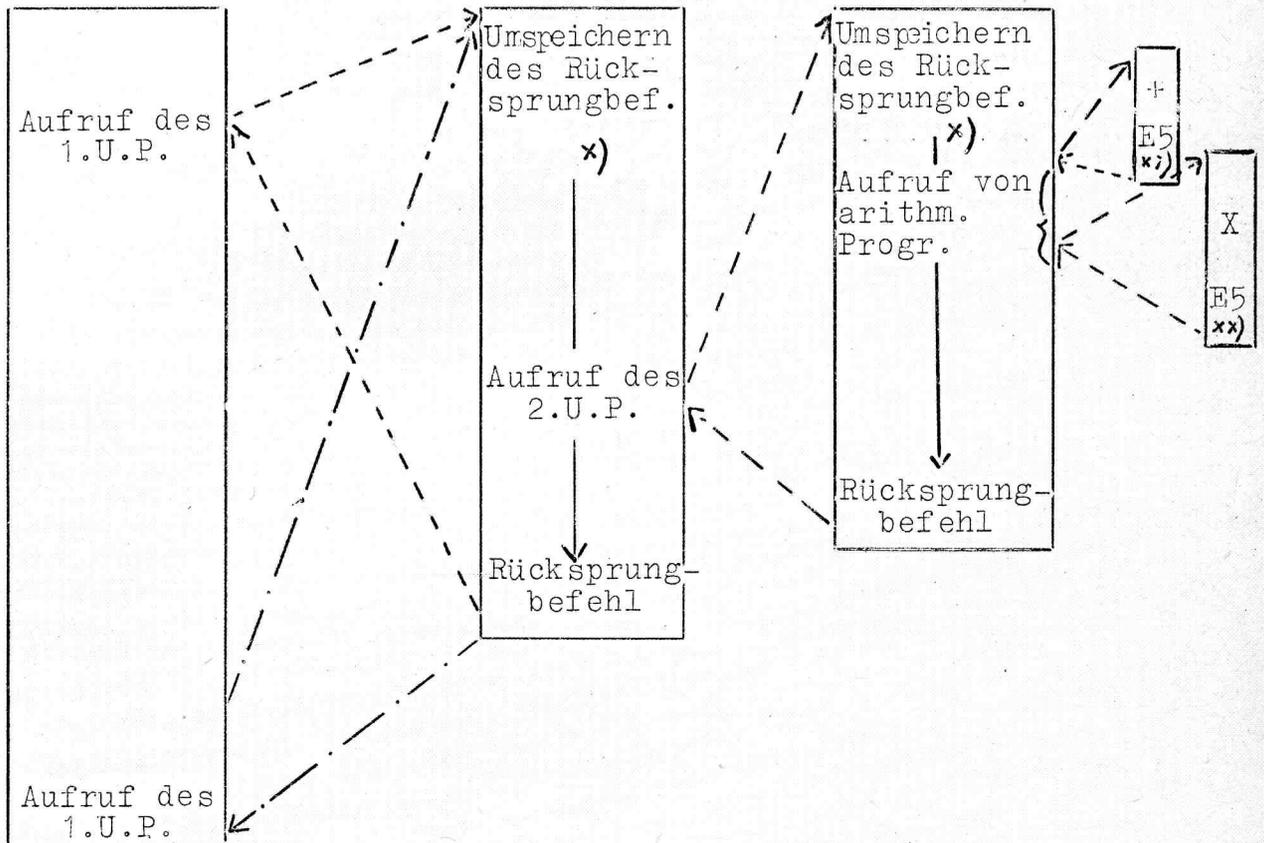
Hierbei ist noch mit zu berücksichtigen, daß die Befehle zur Ausführung der Arithmetik +, -, X, :, M, W, innerhalb der Maschine F-Befehle sind, also auch hier im Unterprogramm vor Ausführung solcher Befehle der Rücksprungbefehl aus Speicherzelle 5 umgespeichert werden muß. Über die Art, wie eine solche Umspeicherung erfolgt, vergleiche man den nächsten Abschnitt.

Im allgemeinen wird der Zusammenhang zwischen Haupt- und Unterprogramm im Groben die nachstehend aufgezeichnete Struktur haben.

Hauptprogramm

1. Unterprogramm
(1.U.P.)

2. Unterprogramm
(2.U.P.)



x) Meist wird der Sprungbefehl mit der Rückkehradresse aus Schnellspeicherzelle 5 in die letzte frei gelassene Speicherzelle des Unterprogrammes gespeichert.

xy) Sofern im Unterprogramm kein weiteres Unterprogramm aufgerufen wird, kann der Rücksprungbefehl aus Schnellspeicherzelle 5 durch den Rufbefehl E5 direkt in das Befehlsregister gebracht werden.

2.4.2 Lese- und Speicherbefehle

Diese Befehle dienen dazu, Zahlen oder andere Worte aus dem Speicher herauszulesen bzw. in gewisse Speicherzellen zur Aufbewahrung zu schreiben.

Lesebefehl

Bm <m> --> a

Bringe Inhalt von m nach a:

Der Inhalt der angewählten Speicherzelle m wird in den Akkumulator gebracht, eine vorher im Akkumulator stehende Zahl wird vor der Übertragung gelöscht. Die in der Speicherzelle m stehende Zahl bleibt dort auch nach dem Ablesen erhalten. Gleichzeitig wird der nächste Befehl ins Befehlsregister geholt.

Speicherbefehl

Um <a> --> m <a> -> a

Umspeichern des Inhalts von a nach m

Der gerade im Akkumulator stehende Wert wird in den angegebenen Speicher m umgespeichert unter Erhaltung des Wertes im Akkumulator.

Gleichzeitig wird der nächste Befehl aus dem Programm in das Befehlsregister gebracht.

Ein kleines Beispiel soll eine Anwendung des Befehls zeigen.

Es sollen die in den Speicherzellen 2000 bis 2005 stehenden Worte in die Speicherzellen 7053 bis 7058 umgespeichert werden. (Es werden etwa die Anfangswerte für ein Programm in die dazu vorgesehenen Speicher gebracht).

Der Transport direkt von einer Trommelspeicherzelle in eine andere ist nicht möglich, deshalb muß die Umspeicherung in zwei Schritten erfolgen.

1. Bringe die Werte in den Akkumulator
2. Speichere die Werte aus dem Akkumulator in den Speicher

Das Programm selbst, das die Umspeicherung vornimmt, soll etwa in den Speicherzellen 2050 ff stehen. Wir stellen hier eine Speicherliste auf, in der in den einzelnen Spalten die Speicherzellennummer, der Inhalt der Speicherzellen und Erläuterungen enthalten sind.

Zellen-Nr.	Inhalt	Erläuterungen
2050	B2000	$\langle 2000 \rangle \rightarrow a$ } $\langle 2000 \rangle \rightarrow 7053$
2051	U7053	
2052	B2001	$\langle 2001 \rangle \rightarrow a$ } $\langle 2001 \rangle \rightarrow 7054$
2053	U7054	
2054	B2002	$\}$ $\langle 2002 \rangle \rightarrow 7055$
2055	U7055	
2056	B2003	$\}$ $\langle 2003 \rangle \rightarrow 7056$
2057	U7056	
2058	B2004	$\}$ $\langle 2004 \rangle \rightarrow 7057$
2059	U7057	
2060	B2005	$\}$ $\langle 2005 \rangle \rightarrow 7058$
2061	U7058	

Die Befehle Bm und Um haben nur eine Adresse und wirken dann stets auf den Akkumulator a; m kann hier eine Schnellspeicheradresse ($m < 256$) oder eine Trommeladresse sein.

Die Befehlszeichen B und U können auch mit zwei Adressen auftreten, wobei die erste eine Schnellspeicher- und die zweite eine Trommeladresse ist. Mit der ersten Adresse wird die Schnellspeicherzelle bezeichnet, die in diesem Fall als Akkumulator dienen soll.

Bs+t $\langle t \rangle \rightarrow s$ $\langle a \rangle \rightarrow a$

Es wird der Inhalt der Trommelzelle t in die angegebene Schnellspeicherzelle s gebracht, ein vorher in s stehender Wert wird zerstört. Der Inhalt des Akkumulators a bleibt in diesem Fall unberührt.

Us+t <s> --> t <s> --> s <a> --> a

Umspeichern aus der Schnellspeicherzelle s in die Trommelzelle t. Der Inhalt von s bleibt erhalten. Auch hier bleibt der Inhalt des Akkumulators a unverändert erhalten.

Wird dem Befehl U noch ein N zugesetzt, so wird nach der Umspeicherung aus der Schnellspeicherzelle diese auf Null gesetzt.

NUs+t <s> --> t 0 --> s <a> --> a

Als Beispiel sei wiederum ein Umspeichern von Zellen der Trommel 2000 ff in die Schnellspeicherzellen 40 bis 45 durchgeführt.

Speicherliste

Zellen-Nr.	Inhalt	Erläuterung
6000	B40+2000	<2000> --> 40
1	B41+2001	<2001> --> 41
2	B42+2002	<2002> --> 42
3	B43+2003	
4	B44+2004	
5	B45+2005	

Die Ergebnisse eines Programms stehen etwa in den Schnellspeicherzellen 60, 62, 64, 66 und sollen auf die Trommel in die Zellen 2100 ff gebracht werden.

Speicherliste

Zellen-Nr.	Inhalt	Erläuterung
2918	U60+2100	<60> --> 2100
2919	U62+2101	<62> --> 2101
2920	U64+2102	<64> --> 2102
2921	U66+2103	<66> --> 2103

2.4.3 Blocktransfer

Als weitere Variante dieser beiden Befehlszeichen B und U gibt es die Befehle BVs+t und UVs+t bzw. NUVs+t für den Blocktransfer zwischen Trommel und Kernspeicher. Hier werden nicht nur einzelne Worte übertragen, sondern durch einen solchen Befehl ausgelöst eine ganze Reihe von Worten. Die Anzahl der zu übertragenden Worte muß vor dem Geben des Blocktransfer-Befehls als Strichzahl im Zähler 13 (unter Schnellspeicheradresse 13 aufrufbar) eingegeben werden, und zwar muß man zur Übertragung von n+1 Worten n' nach 13 bringen. Diese Angabe in 13 geschieht am einfachsten mit einem Befehl, der erst später erklärt wird.

$$\begin{array}{l} \text{CKB13+n} \\ \text{BV40+4070} \end{array} \left. \vphantom{\begin{array}{l} \text{CKB13+n} \\ \text{BV40+4070} \end{array}} \right\} \begin{array}{l} \langle 4070 \rangle \rightarrow 40, \quad \langle 4071 \rangle \rightarrow 41 \dots \\ \dots \dots \langle 4070+n \rangle \rightarrow (40+n) \end{array}$$

Diese beiden Befehle haben die Wirkung, daß die n+1 Worte aus den Trommelzellen 4070 bis 4070+n in die Schnellspeicherzellen 40 bis 40+n gebracht werden. Entsprechend ist auch ein Blocktransfer vom Kernspeicher zur Trommel möglich.

$$\begin{array}{l} \text{CKB13+n} \\ \text{UV70+6030} \end{array} \left. \vphantom{\begin{array}{l} \text{CKB13+n} \\ \text{UV70+6030} \end{array}} \right\} \begin{array}{l} \langle 70 \rangle \rightarrow 6030 \quad \langle 71 \rangle \rightarrow 6031 \dots \\ \quad \quad \quad \langle 71+n \rangle \rightarrow (6031+n) \end{array}$$

Die Inhalte der Schnellspeicherzellen 70 ff bleiben erhalten. Wenn statt UV70+6030 der Befehl NUV70+6030 gegeben wird, werden die Schnellspeicherzellen nach Speichern ihres ursprünglichen Inhalts auf der Trommel gelöscht.

Als Beispiel soll die bei den einfachen B- und U-Befehlen aufgeführte Umspeicherung der Zelleninhalte 2000 bis 2005 in die Speicherzelle 7053 bis 7058 mit den Blocktransferbefehlen durchgeführt werden.

Das Programm stehe etwa wieder in den Zellen 2050 ff.

Zellen-Nr.	Inhalt	
2050	CKB13+5	5' --> 13, es sollen 6 Worte umgespeichert werden
2051	BV25+2000	Übertragung der Zellen 2000 bis 2005 auf die Kernspeicher 25 bis 30
2052	CKB13+5	5' --> 13, es sollen 6 Worte umgespeichert werden
2053	UV25+7053	Übertrag der Inhalte der Kernspeicherzellen 25 ff auf die Trommelzellen 7053 ff

Man sieht, daß das Programm mittels dieser Befehle wesentlich kürzer ist, d.h., daß es weniger Befehle benötigt. Außerdem läuft das hier angegebene Programm auch noch wesentlich schneller ab, da der Blocktransfer optimal d.h. ohne Wartezeiten auf Trommelzellen abläuft.

2.4.4 Arithmetische Befehle

Für den Aufruf der Operationen im gleitenden Komma werden Zeichen verwendet, die den allgemein üblichen entsprechen. Für die Operationen, die in der Maschine durch Unterprogramme ausgeführt werden, wird vorausgesetzt, daß

- der 1. Operand in der Schnellspeicherzelle 6
- der 2. Operand im Akkumulator

steht, ehe die Operation aufgerufen wird.

Bei den Unterprogrammen werden die Schnellspeicherzellen 2 bis 18 (7-10 und 12-15 sind Sonderadressen) als Zwischenspeicher gebraucht; d.h. Werte, die in diesen Speicherzellen stehen, werden beim Aufruf der arithmetischen Operationen zerstört.

Das Ergebnis der Operationen steht stets in den beiden Operandenregistern, also im Akkumulator und auch in der Schnellspeicherzelle 6 zur Verfügung.

+ = Addition im gleitenden Komma
 $\langle b \rangle + \langle a \rangle \rightarrow a, 6$

Der Befehl + bewirkt, daß zu der Zahl in der Schnellspeicherzelle 6 die Zahl im Akkumulator addiert wird und das Ergebnis in Akkumulator und Schnellspeicherzelle 6 zur Verfügung steht.

- = Subtraktion im gleitenden Komma
 $\langle b \rangle - \langle a \rangle \rightarrow a, 6$

Der Befehl - bewirkt die Subtraktion des Akkumulatorinhalts von der Zahl in der Schnellspeicherzelle 6 und bringt das Ergebnis nach a und 6.

X = Multiplikation im gleitenden Komma
 $\langle b \rangle \cdot \langle a \rangle \rightarrow a, 6$

Der Befehl X (der Buchstabe X) bewirkt, daß das Produkt der Zahlen in Schnellspeicherzelle 6 und im Akkumulator in die beiden Operandenregister a und 6 gebracht wird.

: = Division im gleitenden Komma
 $\langle b \rangle : \langle a \rangle \rightarrow a, 6$

Der Befehl : bewirkt, daß die Zahl in Schnellspeicherzelle 6 durch die Zahl im Akkumulator dividiert wird und das Ergebnis im Akkumulator und in Schnellspeicherzelle 6 bereitgestellt wird.

M = Multiplikation mit -1 im gleitenden Komma
 $\langle a \rangle \cdot (-1) \rightarrow a, 6$

Der Befehl M bewirkt, daß die im Akkumulator stehende Zahl mit umgekehrten Vorzeichen in die Schnellspeicherzelle 6 und in den Akkumulator gebracht wird.

Bei dieser Operation mit nur einem Operanden muß dieser im Akkumulator stehen, das Ergebnis geht jedoch analog zu den anderen Operationen nach a und 6. Das gleiche gilt für die beiden folgenden Wurzeloperationen.

W = Quadratwurzelziehen im gleitenden Komma

$$\sqrt[2]{\langle a \rangle} \rightarrow a,6$$

Der Befehl W bewirkt, daß aus der Zahl im Akkumulator die Quadratwurzel gezogen wird und das Ergebnis in den Akkumulator und die Schnellspeicherzelle 6 gebracht wird; das Ergebnis ist stets positiv.

Falls die Zahl im Akkumulator, aus der die Wurzel gezogen werden soll, negativ ist, also die Wurzel im Reellen nicht gezogen werden kann, schreibt die Maschine auf einer neuen Zeile aus

IM Em

und stoppt mit einem Sprungbefehl auf Zelle 1039 im Befehlsregister.

IM ist eine Abkürzung für "imaginär" und Em ist der Aufruf-Befehl für den nächsten Befehl nach dem Befehl W, bezeichnet also die Programmstelle, an der das Ergebnis aus dem Bereich der reellen Zahlen herausführen würde.

Entsprechend schreibt die Maschine

BUEB Em

aus und stoppt mit einem Sprungbefehl auf Zelle 1039 im Befehlsregister, falls die Zahlen die bei den arithmetischen

Operationen entstehen, zu groß werden, insbesondere wenn z.B. durch Null dividiert werden soll.

BUEB ist eine Abkürzung für "Bereichsüberschreitung" und Em bezeichnet die Stelle des Programms, an der die Bereichsüberschreitung auftrat.

Ein weiterer Wurzelbefehl, der allerdings erst nach Einlesen eines Zusatzprogramms zum Grundprogramm möglich wird, ist der Befehl

W2 = Bildung von $\sqrt{1-x^2}$ im gleitenden Komma

$$\sqrt[2]{1-\langle a \rangle^2} \rightarrow a,6$$

Bei diesem Befehl wird die Quadratbildung und die Subtraktion in doppelter Stellenzahl ausgeführt und das Wurzelziehen erfolgt aus dieser Zahl doppelter Länge, so daß kein Genauigkeitsverlust durch Abschneiden entsteht. Andernfalls würde der bei der Quadrierung einer 9stelligen Zahl und der Subtraktion entstehende 18stellige Radikand auf 9 Stellen abgerundet werden. Ein relativer Fehler von 10^{-10} würde aber nach dem Wurzelziehen einen solchen von 10^{-5} ergeben.

Auch hier wird IM und die Stelle des Programms ausgeschrieben, wenn die Zahl im Akkumulator dem Betrag nach größer als 1 ist.

An einem kleinen Beispiel soll die Anwendung der einzelnen Befehle gezeigt werden.

Es soll der Ausdruck

$$\sqrt[2]{\frac{p+q}{r-s}}$$

gebildet werden, wobei p, q, r, s gegebene Zahlen sind. Diese Zahlen seien in den Trommelspeicherzellen 3500 ff gespeichert. Das Programm selbst stehe in den Zellen 3600 ff.

Zellen-Nr.	Inhalt	Bemerkung
3500	78,653	p
3501	-79,831	q
3502	0,0364	r
3503	18,3869	s
3504	zur Aufnahme von p+q bestimmt
3600	B6+3500	p → 6
3601	B3501	q → a
3602	+	p+q → a,6
3603	U3504	p+q → 3504
3604	B6+3502	r → 6
3605	B3503	s → a
3606	-	r-s → a,6
3607	B6+3504	p+q → 6 a = r-s → a
3608	:	$\frac{p+q}{r-s} \rightarrow a,6$
3609	W	$\sqrt{\frac{p+q}{r-s}} \rightarrow a,6$
.....	

2.5 Druckbefehl

Die Ausgabe der errechneten Werte oder sonstiger Werte erfolgt durch das Druckprogramm, das die interne duale Form ins Dezimalsystem übersetzt und die entsprechenden Fernschreibzeichen an die Fernschreibmaschine bzw. auf den Locher gibt. Auf dem Locher ausgestanzte Ergebnisse können später unabhängig von der Maschine auf einem gesonderten Fernschreiber (Programmierungstisch) niedergeschrieben werden.

Das Druckprogramm, bei dem wiederum die Schnellspeicherzellen 2 bis 18 benutzt werden, wird durch den Befehl D aufgerufen.

D = Druckbefehl

<a> → Fernschreibmaschine

<a> → a,6

Das gerade im Akkumulator stehende Wort (gleichgültig ob Zahl oder Befehl, Klartext siehe unten) wird übersetzt und auf dem Fernschreiber niedergeschrieben. Das ausgeschriebene Wort steht nach der Ausgabe im Akkumulator und Schnell-speicherzelle 6 wieder zur Verfügung.

D2 = Druckbefehl

wenn $\langle 1029 \rangle \geq 0$

$\langle a \rangle \rightarrow$ Fernschreibmaschine $\langle a \rangle \rightarrow a,6$

wenn $\langle 1029 \rangle < 0$

$\langle a \rangle \rightarrow$ Locher $\langle a \rangle \rightarrow a,6$

Der Befehl D2 bewirkt ebenfalls eine Übersetzung des im Akkumulator stehenden Wertes, jedoch erfolgt die Ausgabe in Abhängigkeit vom Inhalt der Zelle 1029 entweder auf den Fernschreiber oder auf den Locher.

Diese Möglichkeit, den Befehl D2 von einem Speicherinhalt abhängig zu machen, ist geschaffen worden, damit man beim Ausprüfen von Programmen zunächst alle Ergebnisse auf dem Fernschreiber sofort sichtbar erhält, ohne die verschiedenen D2 Befehle, die bei der eigentlichen Rechnung die Ergebnisse auf den schnelleren Locher ausgeben, abändern zu müssen.

Die in der Maschine in der Gleitkommadarstellung im Dualsystem vorhandenen Zahlen können im Dezimalsystem in zwei verschiedenen Darstellungen ausgegeben werden.

2.5.1 Ausgabeform der Zahlen

A) Ausgabe im gleitenden Komma

Die Zahlen erscheinen mit einer Vorzeichenstelle (bei positiven Zahlen Zwischenraum, bei negativen Zahlen - und dem Komma, danach die angegebene Anzahl von Mantissenstellen. Durch Schrägstrich und Vorzeichen getrennt, folgt dann der stets zweistellig angegebene Dezimalexponent, z.B. bei 8stelligen Mantissen

-,31416384/+01

,12000361/-13

0

Ist die Zahl gleich Null, so erscheint nur an der ersten Mantissenstelle eine 0 und entsprechend viele Zwischenräume danach, daß die Schreibmaschine am Anfang der nächsten Spalte steht.

B) Ausgabe in Normalform

Die Zahlen werden mit der richtigen Stellenanzahl vor und hinter dem Komma in der normal gebräuchlichen Schreibweise ausgegeben. Die letzte Stelle nach dem Komma wird durch Addition von 0,5 Einheiten der letzten Dezimalstelle vor der Ausgabe normal gerundet. Das positive Vorzeichen wird durch einen Zwischenraum, das negative durch - vor der ersten gültigen Ziffer angegeben; führende Nullen vor dem Komma werden durch Zwischenraum berücksichtigt, so daß die Kommata der einzelnen Zahlen richtig untereinander stehen.

Z.B. bei 4 Stellen vor dem Komma und 3 Stellen nach dem Komma:

-1234,567
31,819
-0,007
0,000
0

Für die exakte Null erscheint nur eine 0 an der Stelle vor dem Komma und entsprechend viele Zwischenräume. Ist die Zahl kleiner als die letzte Einheit nach dem Komma aber nicht Null, so erscheint mit Vorzeichen eine Null vor dem Komma und entsprechend viele Nullen nach dem Komma.

C) Ausgabe der Strichzahlen

Die Strichzahlen haben wiederum eine Vorzeichenstelle, die bei positiven Zahlen mit einem Zwischenraum direkt vor der ersten gültigen Ziffer, bei negativen Zahlen durch - an dieser Stelle ausgegeben wird. Nicht besetzte führende Stellen werden durch Zwischenraum ersetzt.

Unmittelbar nach der letzten Ziffer wird ein Apostroph zur Kennzeichnung ausgegeben, z.B.:

n = 4'
-1234'
6'
142'
1007'

Nach dem Apostroph werden die vorgesehene Anzahl von Zwischenräumen bzw. Wagen- und Zeilentransport am Zeilenende an den Fernschreiber gegeben.

Bei einer Null kann in der Maschine nicht entschieden werden, ob es sich um eine Gleitkommazahl oder eine Strichzahl handelt. Deshalb wird Null stets mit der Stellenzahl einer Gleitkommazahl ausgegeben (siehe unten).

2.5.2 Druckanordnung

Zur Aufnahme der Angaben über die Form, in der die ausgedruckten Werte angeordnet und mit wieviel Stellen die Zahlen niedergeschrieben werden sollen, sind noch einige weitere Trommelspeicher vorgesehen und zwar die Zellen 1030 bis 1036. Sämtliche Angaben werden durch Strichzahlen eingegeben.

1030 1' Hier wird die Stellenanzahl der Mantisse angegeben, mit der eine Zahl im gleitenden Komma ausgeschrieben werden soll (bei $\langle 1036 \rangle \geq 0$).
vergl. A

1031 α' In diesen beiden Zellen wird angegeben, mit wie-
1032 β' viel Stellen vor dem Komma (1031) und wieviel
Stellen nach dem Komma (1032) eine Zahl aus-
geschrieben werden soll (bei $\langle 1036 \rangle < 0$).vergl. B

Die Angaben 1030 bis 1032 beziehen sich auf die in der Maschine grundsätzlich in Gleitkommadarstellung vorhandenen Zahlen, die bei der Ausgabe entweder in Gleitkommadarstellung oder durch Kommaverschiebung in Normalform niedergeschrieben werden.

- 1033 k' Hier wird angegeben wieviel Spalten (je Spalte eine Zahl) die ausgeschriebene Tabelle haben soll.
- 1034 z' Die Anzahl der Zwischenräume, die zwischen den einzelnen Spalten zwecks besserer Übersicht gelassen werden sollen. Die Zwischenräume werden jeweils nach der Zahl ausgegeben, nach der Zahl in der letzten Spalte erfolgt automatisch Wagen- und Zeilentransport.
- 1035 n' Falls in der Tabelle Strichzahlen ausgegeben werden, so ist hier die Anzahl der Stellen, mit denen die Strichzahlen herausgeschrieben werden sollen, anzugeben. (Siehe C)
- 1036 +1' bzw. -1' Durch die Angabe einer positiven bzw. negativen Zahl wird entschieden, ob die in Gleitkommaform vorfindbaren Zahlen in Gleitkommaform ($\langle 1036 \rangle$ positiv) oder aber in Normalform ($\langle 1036 \rangle$ negativ) ausgegeben werden sollen.

Die Angaben sind teilweise gewissen Beschränkungen unterworfen. Die Stellenzahl (1030) bei Gleitkommaausgabe kann zwischen 1 und 9 liegen, da mehr als 9 Stellen doch nicht vorliegen.

$$1 \leq l \leq 9$$

Entsprechend sollen die Angaben α und β (1031, 1032), d.h. die Stellen vor und die Stellen nach dem Komma bei Normalformausgabe, zusammen nicht mehr als 9 betragen.

$$\alpha + \beta \leq 9$$

Hier kann β auch Null sein, d.h. es werden keine Stellen nach dem Komma ausgeschrieben.

Die Normaldarstellung der Zahlen sollte nur gewählt werden, wenn im Voraus bekannt ist, wie groß die größte auszugebende Zahl sein wird.

Hat eine Zahl mehr als die angegebenen α Stellen vor dem Komma, so gibt die Maschine die Zahlen automatisch im Gleitkommä mit der in 1030 notierten Stellenzahl. Aus diesem Grunde sollten stets alle Angaben für die Druckanordnung in den Zellen 1030 bis 1036 gemacht werden.

Die Stellenzahl für Strichzahlen ist auf maximal 12 beschränkt, da die Kapazität nur 12 Stellen beträgt.

$$1 \leq n \leq 12$$

Falls die auszugebenden Strichzahlen mehr als n Stellen haben, so werden sie mit der maximal möglichen Stellenzahl, nämlich 12stellig herausgeschrieben.

Die Anzahl der Spalten und Zwischenräume ist beliebig, nur ist zu beachten, daß auf einer Zeile des Fernschreibers maximal 104 Anschläge gemacht werden können und die Zahlen in Gleitkommadarstellung mit 1+6 Anschlägen (siehe oben) die Zahlen in Normalform mit $\alpha+\beta+2$ Anschlägen (siehe oben) und die Strichzahlen mit $n+2$ Anschlägen auf dem Papier erscheinen.

Es sei noch erwähnt, daß die Anzahl der Ziffern und der Zwischenräume von Spalte zu Spalte variieren kann. Man hat dazu im Programm vor den Druckbefehlen die Angaben in den Zellen 1030 - 1036 entsprechend zu ändern. (Vergl. hierzu das Kap. 4 Adressenrechnung und die Programmbeispiele)

2.5.3 Ausgabe von Befehlen

Befehle werden auch nebeneinander mit entsprechend vielen Zwischenräumen geschrieben, jedoch ist hier keine genaue Einhaltung der Spalten möglich, da die Befehle unterschiedlich lang sind. Der Operationsteil der Befehle wird in der internen Form ausgegeben, der Adressenteil erscheint stets in zwei Teilen, die durch ein +Zeichen getrennt sind, wobei der erste Teil die Schnellspeicheradresse, der zweite die Trommeladresse bedeuten. Falls eine der beiden Adressen Null ist, wird an der entsprechenden Stelle nur 0 ausgegeben.

Die vercodeten Rufbefehle +, -, X usw. werden in der internen Form als F-Befehle ausgegeben. Bei den Befehlen, die sich nur auf Schnellspeicher beziehen, kommt ein vom Leseprogramm zugesetztes K mit heraus (K = Kennzeichen, daß nur die Schnellspeicheradresse wirksam sein soll).

Im einzelnen erscheint bei der Ausgabe:

Eingabeform	Ausgeschriebene Form
B14o8	NAo+14o8
B24	NAK24+o
+	Fo+454
-	Fo+451
X	Fo+362
:	Fo+425
M	Fo+419
D	Fo+512
D2	Fo+514
W	Fo+9o8
W2	Fo+91o

und für die hier noch nicht behandelten internen Zeichen

L...	LLR...
O...	AS...
I...	UA...

2.5.4 Ausgabe von Klartext

Da Überschriften und sonstige Angaben meist mehr als 7 Zeichen umfassen, also in mehrere aufeinanderfolgende Zellen eingelesen wurden, ist bei der Ausgabe von Klartext vorgesehen, daß man nur die erste Zelle des Klartextes in den Akkumulator zu bringen braucht. Durch das Druckprogramm werden dann laufend die weiteren Zellen geholt und ausgegeben. Dieses geschieht solange, bis aus einer Zelle etwas geholt wird, was kein Klartext ist. Der Klartext wird bei der Ausgabe am Anfang und am Ende mit dem Kennzeichen  (kein Anschlag auf dem Fernschreiber aber eine Lochung) versehen.

Eine Spaltenzählung erfolgt beim Klartext nicht, es werden auch keine Zwischenräume, außer den im Text selbst geschriebenen, ausgegeben. Die Ausgabe des Klartextes ist allerdings nur möglich, wenn der Klartext und der Druckbefehl auf der Trommel steht, also nicht vom Programm im Kernspeicher.

2.5.5 Tabellierbefehle

Zur Anordnung der Tabellen sind noch einige spezielle Unterprogramme vorgesehen.

F1000 Wagen- und Zeilentransport auf Fernschreiber, d.h. der Wagen der Fernschreibmaschine wird an den Anfang der nächsten Zeile gerückt.

$\langle a \rangle \rightarrow a,6$

F1002 wenn $\langle 1029 \rangle \geq 0$
Wagen- und Zeilentransport auf den Fernschreiber

$\langle a \rangle \rightarrow a,6$

wenn $\langle 1029 \rangle < 0$

Wagen- und Zeilentransport auf den Locher,

$\langle a \rangle \rightarrow a,6$

Bei diesem Befehl ist nach Ausgabe der entsprechenden Fernschreibzeichen der ursprünglich im Akkumulator stehende Wert noch erhalten und steht auch in Schnellspeicherzelle 6 zur Verfügung. Der Befehl braucht nur gegeben zu werden, wenn eine Zeile nicht voll beschrieben werden soll oder wenn ein doppelter Zeilenabstand vorgesehen ist.

Soll eine Spalte in einer Tabelle übersprungen werden, so kann dieses mit dem Befehl

F800 = Spaltensprung auf Fernschreiber, d.h. weiterrücken um so viele Zeichenbreiten, daß der Wagen der Fernschreibmaschine am Anfang der nächsten Spalte steht.

$\langle a \rangle \rightarrow a,6$

F802 = Spaltensprung, $\langle a \rangle \rightarrow a,6$
bei $\langle 1029 \rangle \geq 0$ auf Fernschreiber
bei $\langle 1029 \rangle < 0$ auf Locher

Der im Akkumulator stehende Wert bleibt im Akkumulator und Schnellspeicherzelle 6 erhalten.

Die Anzahl der ausgegebenen Zwischenräume hängt davon ab, mit wieviel Stellen die Gleitkommazahlen herausgeschrieben werden sollen; bei $\langle 1036 \rangle \geq 0$ von der in 1030 angegebenen Stellenzahl, bei $\langle 1036 \rangle < 0$ von den in 1031 und 1032 stehenden Zahlen. Die Ausführung des Befehls F800 wird bei der Spaltenzählung mit berücksichtigt.

Mit den beiden folgenden Befehlen kann man noch um beliebig viele Zeichenbreiten einrücken bzw. beliebig oft Zeilenvorschub an den Fernschreiber bzw. Locher geben, z.B. zur Einteilung der ausgeschriebenen Ergebnisse auf einzelnen Seiten DIN A 4 mit entsprechenden freigelassenen oberen und unteren Rändern.

F840 Einrücken um die im Akkumulator durch eine negative Strichzahl angegebene Anzahl von Zeichenbreiten.

F842 Entsprechend werden die Fernschreibzeichen für Zwischenraum
bei $\langle 1029 \rangle \geq 0$ an die Fernschreibmaschine
bei $\langle 1029 \rangle < 0$ an den Locher
gegeben.

F850 Es wird das Papier im Fernschreiber um sovielen Zeilen weitertransportiert, wie die negative Strichzahl im Akkumulator angibt.

F852 wirkt entsprechend
bei $\langle 1029 \rangle \geq 0$ auf den Fernschreiber
bei $\langle 1029 \rangle < 0$ auf den Locher

2. 6 Bedingungszeichen

Alle Befehle der ZUSE Z 23 können mit Bedingungszeichen versehen werden, d.h. die Befehle sollen nur dann ausgeführt werden, wenn die abgefragte Bedingung erfüllt ist. Ist die Bedingung nicht erfüllt, so wird ohne Ausführung des bedingten Befehls auf die folgende Speicherzelle übergegangen.

Als Bedingungen können gesetzt werden:

- PP . . . führe den Befehl . . . nur aus, wenn der Inhalt des Akkumulators positiv ist, wobei die Null mitgerechnet wird, $\langle a \rangle \geq 0$
- QQ . . . führe den Befehl . . . nur aus, wenn der Inhalt des Akkumulators negativ ist (Null ausgeschlossen) $\langle a \rangle < 0$

und die Kombination der beiden Zeichen als Nullabfrage.

- PPQQ . . . führe den Befehl . . . nur aus, wenn im Akkumulator eine Null steht, $\langle a \rangle = 0$

Als weitere Bedingungszeichen, die allerdings hauptsächlich in den Grundprogrammen benutzt werden, gibt es (da die getesteten Schnellspeicherzellen beim Grundprogramm belegt werden)

- P . . . führe den Befehl . . . nur aus, wenn der Inhalt der Schnellspeicherzelle 2 positiv ist $\langle 2 \rangle \geq 0$
- Q . . . führe den Befehl . . . nur aus, wenn der Inhalt der Schnellspeicherzelle 2 negativ ist $\langle 2 \rangle < 0$
- Y . . . führe den Befehl . . . nur aus, wenn die letzte Stelle von Schnellspeicherzelle 3 mit einer 1 besetzt ist. $\langle 3 \rangle_{40} = 1$

und eine Kombination:

PQQQ ... führe den Befehl . . . nur aus, wenn die ersten beiden Stellen des Akkumulators nicht übereinstimmen, d.h. wenn im Akkumulator ein Befehl oder ein Klartext steht.

$$\langle a \rangle_1 \neq \langle a \rangle_2$$

Test beliebiger Schnellspeicherzellen

Für Sprungbefehle auf der Trommel lassen sich sämtliche Schnellspeicherzellen (außer 3,4 und 5) auf das Vorzeichen abfragen.

Bei diesen Sprungbefehlen ist das Bedingungszeichen P oder Q zu geben und in den Schnellspeicheradressenstellen die Schnellspeicherzelle, die getestet werden soll, während die Trommeladresse die Stelle angibt, an die bei erfüllter Bedingung gesprungen werden soll.

PEs+t führe den Sprung an die Trommelzelle t nur aus, wenn in der Schnellspeicherzelle s etwas Positives (einschließlich Null) steht $\langle s \rangle \geq 0$

QEs+t führe den Sprung auf die Trommelzelle t nur aus, wenn der Inhalt der Schnellspeicherzelle s negativ ist $\langle s \rangle < 0$

Bedingungsschalter

Am Bedienungspult der ZUSE Z 23 sind 5 Tasten angebracht, deren Stellung man unter der Schnellspeicheradresse 14 abfragen kann. Die einzelnen Tasten bewirken, daß man 1', 2', 4', 8' oder 16' erhält, also durch additive Kombination der Tasten 32 verschiedene Möglichkeiten nämlich die Zahlen 0 - 31 einstellen kann, um damit den Rechenablauf zu beeinflussen, ohne irgendwie das Programm selbst abzuändern.

Akkumulatortastatur

Daneben besteht noch die Möglichkeit über die Akkumulatortastatur ein ganzes Wort in den Akkumulator zu bringen. Während der Rechnung kann etwas Neues eingetastet werden, das vom Programm durch den folgenden Befehl übernommen wird.

U4 Übernahme des in der Akkumulatortastatur eingetasteten Wortes in den Akkumulator.

2.7 Stopbefehl

Als Stopzeichen wirkt das Befehlszeichen Z, das jedem Befehl beigefügt werden kann.

Befehle, die mit dem Stopzeichen versehen sind, bewirken ein Stop der Maschine und werden erst nach Betätigung der Starttaste ausgeführt.

Im allgemeinen wird am Ende eines Programms der Stopbefehl

Zo (der Buchstabe Z und die Ziffer Null)
 Stop der Maschine

stehen.

Dieser Befehl kann auch unter Bedingungen gestellt werden; dann tritt erst Stop ein, wenn die angegebene Bedingung erfüllt ist, z.B.

PPZo wenn der Akkumulatorinhalt positiv ist ($\langle a \rangle \geq 0$), soll ein Stop eintreten. Im anderen Fall wird auf die nächste Speicherzelle übergegangen.

Bedingter Stop

Eine Taste am Bedienungspult, die mit "Bed. Stop" (bedingter Stop) bezeichnet ist, bewirkt, daß die Maschine stehen bleibt (Taste eingedrückt) bzw. die Stelle übergeht (Taste auf aus), wenn eine Zahl ins Befehlsregister kommt.

Man kann dies benutzen, um an verschiedenen Stellen einen Stop in das Programm einzubauen. Um die einzelnen Stellen aber zu unterscheiden, bringt man die Adresse der nächsten Speicherzelle als Strichzahlen in Zellen, die im Programmablauf wie Befehle ins Befehlsregister gebracht werden (z.B. in die Zelle 4181 bringt man 4182'). An diesen Stellen kann man dann entweder mit "Weiter" und "Start" fortfahren oder durch eine Eintastung zu einem anderen Programmteil übergehen. Wenn der Schalter "Bed. Stop" ausgestellt wird, werden diese so markierten Stellen überlaufen.

Falls man diese Möglichkeit nicht ausnutzen will, aber den Schalter einstellt, so bleibt die Maschine stehen, sofern das Programm falsch läuft und Zahlen anstelle von Befehlen ins Befehlsregister bringt. Das kann beim Programmprüfen eine wichtige Hilfe sein.

2.8 Bandbefehle

Bisher wurden nur einzelne Befehle oder kleine Programmteile besprochen, die aber aus einem größeren Rahmen herausgerissen waren und stets wurde vorausgesetzt, daß die Befehle und Zahlenwerte schon in den Speicherzellen standen.

Hier folgt nun die Behandlung der sogenannten Bandbefehle, mit denen der Maschine gewisse Angaben gemacht werden, erstens wohin die einzelnen Befehle und Zahlen, die auf dem Lochstreifen stehen, gespeichert werden sollen und zweitens an welcher Stelle das Programm die Rechnung beginnen soll, d.h. wo der erste auszuführende Befehl steht, mit dem das Programm gestartet wird.

Diese Bandbefehle stehen nur auf dem Lochband und enthalten Anweisungen für das Leseprogramm, ohne aber selbst in Speicherzellen gebracht zu werden.

UmU Dieser Bandbefehl bewirkt, daß die folgenden Worte auf dem Lochstreifen in die Speicherzellen m , $m+1$, $m+2$, . . . usw. eingespeichert werden.
Z.B. Es soll in die Speicherzellen 1030 ff eine Druckanordnung eingegeben werden

6 Stellen bei Gleitkommazahlen
3 Stellen vor dem Komma } bei Normalform
4 Stellen nach dem Komma }
3 Spalten
2 Zwischenräume zwischen den Spalten
4 Stellen bei Strichzahlen

Ausgabe in der Normalform

so müßten auf dem Lochstreifen folgende Angaben gemacht werden

U1o3oU

6'

3'

4'

3'

2'

4'

-1'

(vergl. Lochstreifenherstellung 3.4)

Falls darauf wieder ein UmU folgt, so wird von der nun neu bestimmten Stelle m an das Weitere gespeichert.

EmE

Dieser Bandbefehl, der am Ende des Lochstreifens gegeben wird, wenn das ganze Programm und alle Daten eingelesen sind, bewirkt, daß die Maschine mit dem Befehl in der Speicherzelle m anfängt zu rechnen.

Häufig ist es zweckmäßig den Startbefehl noch mit einem Stopzeichen Z zu versehen; man schreibt also

EZmE

Dann stoppt die Maschine mit dem Sprungbefehl auf die Zelle m im Befehlsregister. Man kann dann nochmals überprüfen, ob alle Lochstreifen eingegeben, die Schalter alle richtig gestellt sind; und danach mit der Starttaste das Programm starten.

Eine weitere Bandbefehlsart wird im Kapitel 5 Relativadressen besprochen werden.

3. Beispiel

3.1 Vorbereitung

Um ein Problem für eine Rechenanlage vorzubereiten, bedarf es gewisser Vorarbeiten. Die erste Aufgabe, die vielfach die schwierigste und umfangreichste ist, besteht darin, das Problem überhaupt erst einmal in Formeln, in mathematische Ausdrücke zu fassen.

Hat man eine mathematische Formulierung gefunden, ist es vielfach notwendig, daß man das Näherungsverfahren auswählt, mit dem man das Problem behandeln will; wenn z.B. eine Differentialgleichung zu lösen ist, welches Verfahren soll man verwenden, Runge-Kutta, Adam'sches Interpolationsverfahren oder ein anderes? Je nach Art des Problems wird man ein geeignetes Näherungsverfahren wählen.

Als dritter Punkt käme dann das Aufstellen eines Flußdiagramms, das bildlich den Ablauf des Rechengangs, den Zusammenhang der einzelnen Abschnitte der Rechnung darstellt. (Vergl. nächsten Abschnitt)

Das Letzte ist dann die Vercodung des im Flußdiagramm dargestellten Programmablaufs, also die Übersetzung in eine der Rechenanlage verständlichen Sprache.

Von diesen 4 für die Vorbereitung eines Problems notwendigen Abschnitten ist nur der letzte maschinengebunden, da nur in diesem die maschineneigene Programmiersprache verwendet wird.

Als Beispiel soll hier bewußt eine nicht zu umfangreiche Aufgabe programmiert werden, wobei die ersten beiden Vorbereitungsschritte schon ausgeführt sein sollen. Es würde über den Rahmen der Programmieranleitung hinausgehen, auch dieses hier behandeln zu wollen.

Aufgabe:

Es soll eine Funktion $F(x)$ tabelliert werden, die durch den Quotienten zweier Polynome 4. Grades dargestellt ist.

$$F(x) = \frac{a_4x^4 + a_3x^3 + a_2x^2 + a_1x + a_0}{b_4x^4 + b_3x^3 + b_2x^2 + b_1x + b_0}$$

Die Tabellierung soll für das Intervall von $x_0=0$ bis $x_n=3$ mit einer Schrittweite $\Delta x = 0,125$ durchgeführt werden.

Niedergeschrieben werden soll jeweils das Argument und der Funktionswert nebeneinander mit einer Überschrift in Klartext

X	F(X)
---	------

3.2 Das Flußdiagramm

In dem Flußdiagramm werden in einzelnen Kästchen die Operationen angegeben, die ausgeführt werden sollen. Der Ablauf des Programms wird durch Verbindung der einzelnen Kästchen angezeigt. Um die Stellen besonders hervorzuheben, an denen Entscheidungen getroffen werden und damit Verzweigungen im Programmablauf eintreten, werden diese durch Ovale gekennzeichnet, in denen die gestellte Bedingung, mit einem Fragezeichen versehen, eingetragen wird.

Für die Darstellung des Flußdiagramms sollen folgende Konventionen gelten:

1. Jedes Kästchen bzw. Oval hat nur einen Eingang
2. Jedes Kästchen (unbedingte Operation) hat nur einen Ausgang.
3. Bei bedingten Operationen (in Ovalen) wird die abgefragte Bedingung mit einem Fragezeichen versehen. Ovale haben stets zwei Ausgänge, die mit "ja" und "nein" bezeichnet werden und damit die Ausgänge bei erfüllter bzw. nicht erfüllter Bedingung bezeichnen.

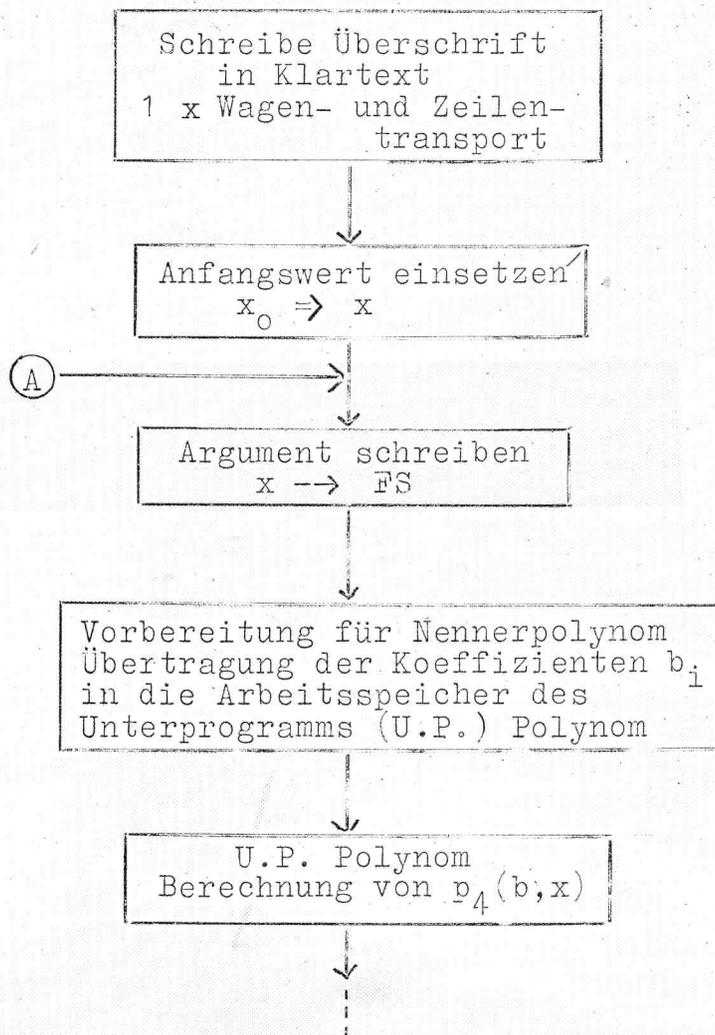
Um einen Überblick über das gesamte Programm zu bekommen, wird man in Kästchen ganze Programmteile oder Unterprogramme darstellen, die dann in gesonderten Flußdiagrammen ausführlich behandelt werden.

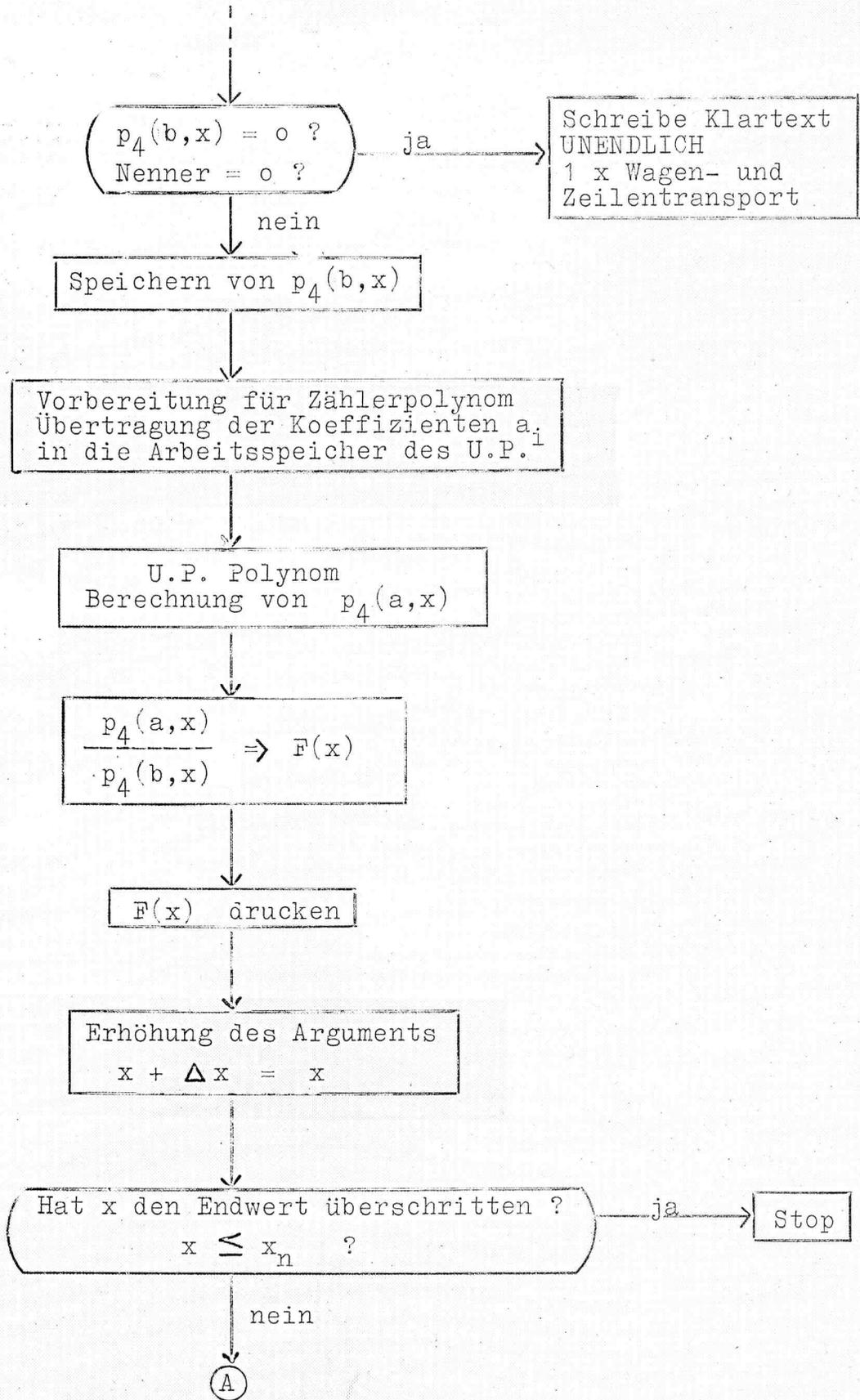
Wie sieht nun das Flußdiagramm zu dem gegebenen Beispiel aus:

$$F(x) = \frac{a_4x^4 + a_3x^3 + a_2x^2 + a_1x + a_0}{b_4x^4 + b_3x^3 + b_2x^2 + b_1x + b_0} = \frac{p_4(a,x)}{p_4(b,x)}$$

Es tritt hier im Zähler wie im Nenner ein Polynom 4. Grades auf, also ein gleicher Ablauf, nur daß andere Koeffizienten vorliegen. Man wird daher hierfür ein Unterprogramm aufstellen, für das nur vom Hauptprogramm die Parameter (hier also die Koeffizienten) an bestimmter Stelle bereitgestellt werden.

Flußdiagramm:





Wenn nicht eine direkte Verbindung zwischen zwei Punkten, die nacheinander ablaufen sollen, möglich ist, so bezeichnet man die Stellen mit kleineren Kreisen mit einem Buchstaben oder einer Zahl, wie hier z.B. (A).

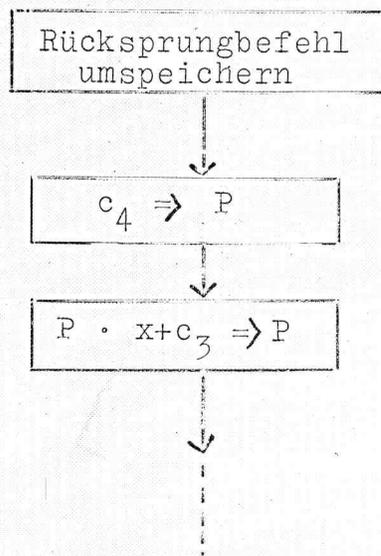
Hier mußte noch eine spezielle Abfrage erfolgen, ob der Nenner = 0 ist, da in diesem Fall die Division ja Bereichsüberschreitung bringen und die Maschine stoppen würde. In diesem Fall soll ein Klartext ausgeschrieben werden. Nach dem Ausschreiben von Klartext wird stets einmal Wagen- und Zeilentransport gegeben, da ja Klartext bei der Spaltenzählung nicht mitgezählt wird und somit nicht automatisch auf die nächste Zeile übergegangen wird.

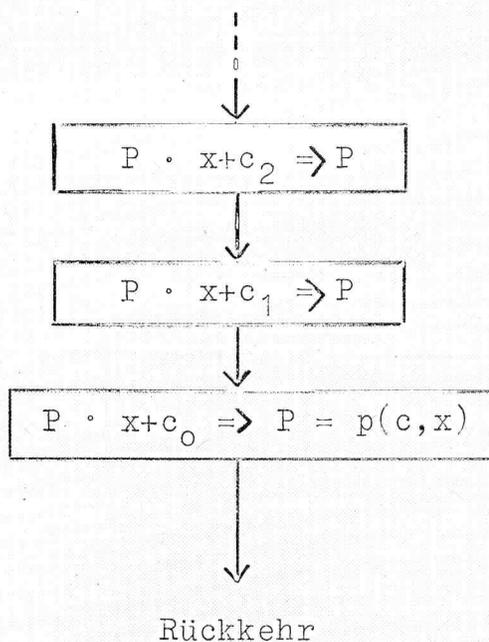
Ein besonderes Flußdiagramm wird für das Unterprogramm Polynom aufgestellt. Das Polynom wird nach dem Horner'schen Schema berechnet, um mit möglichst wenig Operationen auszukommen.

$$\begin{aligned} p_4(c,x) &= c_4x^4 + c_3x^3 + c_2x^2 + c_1x + c_0 \\ &= (((c_4x + c_3)x + c_2)x + c_1)x + c_0 \end{aligned}$$

Hier sind mit c_i die Koeffizienten gemeint, die gerade in dem Arbeitsspeicher zur Verfügung stehen (P bezeichnet den jeweils aufgebauten Teil des Polynoms).

Flußdiagramm





3.3 Speicherliste

In der Speicherliste sind in einzelnen Spalten die Zellennummern, ihr Inhalt und gewisse Bemerkungen und Erläuterungen angegeben.

Als erstes soll die Speicherliste für das Unterprogramm aufgestellt werden, und zwar soll dieses Programm im Kernspeicher stehen in den Zellen 200 ff, in den Kernspeicherzellen 190 soll das Argument, in den Zellen 191 bis 195 sollen die Koeffizienten c_4, c_3, \dots, c_0 stehen.

Zellen-Nr.	Inhalt	Erläuterungen
190	x	Das jeweilige Argument x, Zahlenwert wird vom Hauptprogramm eingesetzt.
191	c_4	Die Koeffizienten a_i bzw. b_i , die Zahlenwerte werden vom Hauptprogramm eingesetzt.
192	c_3	
193	c_2	
194	c_1	
195	c_0	
200 1	B5 U220	Rücksprungbefehl aus Speicherzelle 5 umspeichern.

202	B191	}	$c_4 \Rightarrow P \rightarrow 6$
3	U6		
4	B190	}	$P \cdot x + c_3 \Rightarrow P \rightarrow a,6$
5	x		
6	B192		
7	+	}	
8	B190		
9	x	}	$P \cdot x + c_2 \Rightarrow P \rightarrow a,6$
210	B193		
1	+	}	
2	B190		
3	x	}	$P \cdot x + c_1 \Rightarrow P \rightarrow a,6$
4	B194		
5	+		
6	B190	}	
7	x		
8	B195	}	$P \cdot x + c_0 \Rightarrow P = p_4(c, x) \rightarrow a,6$
9	+		
220	Eo		Hier wird vom Programm selbst der richtige Rücksprungbefehl eingesetzt.

Das Hauptprogramm soll auf der Trommel stehen in den Speicherzellen ab 1800, während die Koeffizienten a_i ab 1790, die Koeffizienten b_i ab 1780 eingespeichert seien und die Überschrift ab 1750, die Erklärung UNENDLICH ab 1770. Der Anfangswert x_0 , die Schrittweite Δx und der Endwert x_n stehe in Zelle 1795 ff.

1750 } \mathbb{R} X F(X) \mathbb{R} Wieviel Zwischenräume im Klartext
 175... } o gegeben werden müssen, muß man an Hand der Form, in der ausgeschrieben werden soll, auszählen, damit die Bezeichnung über den entsprechenden Spalten steht. Der Text wird mehrere Worte umfassen; als Kennzeichen für die Maschine

177o 1	$\left. \begin{array}{l} \text{UNENDLICH} \\ 0 \end{array} \right\}$	<p>wie weit der Text geht, wird in die nächste Zelle etwas eingespeichert, das kein Klartext ist, z.B. eine o.</p> <p>Der Text umfaßt zwei Worte (7 Zeichen = 1 Wort) Es wird wiederum in die nachfolgende Zelle eine o eingespeichert.</p> <p>Man braucht bei dem Klartext nicht genau abzuzählen, wieviele Speicherzellen tatsächlich belegt sind, nur muß man genügend Plätze vorsehen, damit kein Überschreiben von verschiedenen Angaben eintritt.</p>
178o 1 2 3 4	1 5,5 3,25 -6,375 -3,375	b_4 Die Koeffizienten für das b_3 Nennerpolynom b_2 b_1 b_0
179o 1 2 3 4	1,65 3,135 -10,824 -17,127 -2,97	a_4 Die Koeffizienten für das a_3 Zählerpolynom a_2 a_1 a_0
1795 6 7	o o,125 3,01	x_0 Δx $x_n + \epsilon$ Zur Abfrage, ob das Ende erreicht ist, verwendet man einen Wert der etwas größer ist, als das letzte x, für das die Berechnung noch vorgenommen werden soll.

1800	B1750	}	Ausschreiben der Überschrift, die ab Zelle 1750 gespeichert ist
1	D		1 x Wagen- und Zeilentransport nach der Überschrift
2	F1000		$x_0 \rightarrow$ 190 Einsetzen des Anfangswertes
3	B190+1795		
4	B190	}	x ausschreiben
5	D		Vorbereitungsbefehl zum Blocktransfer von 5 Zahlen
6	CKB13+4		Blocktransfer $b_i \rightarrow$ 191 ff damit ist die Vorbereitung für das U.P. getroffen
7	BV191+1780		Aufruf des Unterprogramms Polynom, nach Rückkehr steht $p_4(b,x)$ in a und 6
8	F200		Falls $p_4(b,x) = 0$ Sprung nach 1825
9	PPQQE1825		$p_4(b,x) \rightarrow$ 189
1810	U189	}	Blocktransfer der Koeffizienten a_i nach 191 ff Vorbereitung für das U.P.
1	CKB13+4		U.P. Polynom, $p_4(a,x) \rightarrow$ a,6
2	BV191+1790		$p_4(b,x) \rightarrow$ a
3	F200		$\frac{p_4(a,x)}{p_4(b,x)} \Rightarrow F(x) \rightarrow$ a,6
4	B189		Ausschreiben von F(x)
5	:		$\Delta x \rightarrow 6$
6	D		$x \rightarrow a$
7	B6+1796		} $x + \Delta x \Rightarrow x$
8	B190		
9	+		
1820	U190		$x_n + \epsilon \rightarrow a$
1	B1797		$x - (x_n + \epsilon) \rightarrow$ a,6
2	-		wenn $x < x_n + \epsilon$ erneute Durchrechnung ab 1804
3	QQE1804		sonst Stop
4	Zo		
5	B1770	}	Ausschreiben von UNENDLICH
6	D		1 x Wagen- und Zeilentransport
7	F1000		

1828	E1817	Sprung an die Stelle, an der das Argument erhöht wird.
------	-------	--

Weiter muß der Maschine noch eingegeben werden, in welcher Form die Ergebnisse niedergeschrieben werden sollen; diese Angaben gehören in die Speicherzellen 1030 ff. Es soll in Normalform herausgeschrieben werden mit 2 Stellen vor dem Komma und 6 Stellen nach dem Komma. Es sind zwei Spalten vorgesehen für das Argument x und den Funktionswert $f(x)$, die durch 3 Zwischenräume voneinander getrennt sein sollen.

1030	7'	Falls die Zahlen größer als 100 werden, sollte in Gleitkommadarstellung ausgeschrieben werden mit 7 Stellen Mantisse
1	2'	} Stellenzahl bei Normalform vor und nach dem Komma
2	6'	
3	2'	2 Spalten
4	3'	3 Zwischenräume
5	4'	für Strichzahlen hier unnötig
6	-1'	Kennzeichen für Ausgabe in Normalform

Damit ist das Programm auf dem Papier aufgestellt und muß nun in einen Lochstreifen gestanzt werden, um es in die Maschine einlesen zu können.

3.4 Lochstreifenherstellung

Die Lochstreifenherstellung erfolgt auf dem Programmierungstisch, der aus einem Schaltkästchen, einem Fernschreiber mit angebautem Locher, einem Abtaster und einem Schrittstopgerät besteht.

Mit dem Schaltkästchen wird die ganze Anlage ein- (linker Knopf) bzw. ausgeschaltet (rechter Knopf).

Auf dem Fernschreiber werden die Programme und Daten geschrieben; es entsteht dabei ein Protokoll und ein Lochstreifen.

Zum Kopieren oder Ausbessern von Programmen benutzt man den Abtaster und das Schrittstopgerät, das erlaubt laufend oder zeichenweise zu kopieren.

Die Tastatur des Fernschreibers umfaßt:

1. Das Alphabet

Diese Tasten können nur betätigt werden, wenn vorher einmal die Buchstabentaste (Bezeichnung A...) gegeben wurde.

2. Ziffern und Zeichen

Diese Tasten können nur nach Betätigung der Zifferntaste (Bezeichnung 1...) bedient werden.

Es sind die 10 Ziffern 0 bis 9 und die Zeichen + - : / , . ? ' () =   vorhanden.

3. Die Betriebszeichen für die Fernschreibmaschine

Buchstabentaste (kurz Bu)

Zifferntaste (kurz Zi)

Wagenrücklauf auf der Taste bezeichnet mit < (kurz Wr)

Zeilentransport auf der Taste bezeichnet mit ≡ (kurz Zl)

Zwischenraum unbezeichnete große Taste unten in der Mitte (kurz Zw)

Leer auf der Taste bezeichnet mit 
locht auf dem Lochstreifen nur das Transportloch

Die Betriebszeichen können stets betätigt werden, gleichgültig, ob die Fernschreibmaschine gerade auf Buchstaben oder Ziffern gestellt ist.

Die Betriebszeichen haben für das Leseprogramm der ZUSE Z 23 besondere Bedeutung, die im folgenden aufgeführt sei:

1. Innerhalb eines Operationsteiles

- a) leer, Wr, Zw, Zl sind wirkungslos
(nach X, M, W, D ist Wr, Zw oder Zl Schlußzeichen)
- b) Zi Schlußzeichen des Operationsteils, Übergang zum Einlesen des Adressenteils.
- c) Bu Irrungszeichen, der bisher eingelesene Operationsteil des angefangenen Wortes wird gelöscht.
- d) Die nicht benutzten Buchstaben J und T führen aus dem Leseprogramm heraus.
J auf die Zelle 1040
T auf die Zelle 1041

2. Im Adressenteil eines Befehls

- a) Zi, leer wirkungslos
- b) Wr, Zw oder Zl sind Schlußzeichen
- c) Bu führt auf den Teil des Leseprogramms, in dem Bandbefehle eingelesen werden.
(ein zweites Bu bedeutet Irrung, d.h. der bereits eingelesene Teil des angefangenen Wortes (Operationsteil und Adressenteil wird gelöscht).
- d) Die nicht benutzten Zeichen führen aus dem Leseprogramm

⊕	auf die Zelle	1042
(" "	1043
)	" "	1044
?	" "	1045
.	" "	1046
=	" "	1047

3. Bei Zahlen

- a) Zi, leer wirkungslos
- b) Wr, Zw oder Zl Schlußzeichen
- c) ein- oder mehrfach Bu Irrungszeichen, alle Zeichen der angefangenen Zahl werden gelöscht.

4. Bei Strichzahlen

Wie bei Zahlen, nur gilt hier das Apostroph als Schlußzeichen.

Zwischen zwei Worte können beliebig viele Betriebszeichen der Fernschreibmaschine gegeben werden.

Beim Lochen des Beispiels gibt man nun vor jedem Programmabschnitt bzw. jeder Zahlengruppe einen UmU-Bandbefehl, wohin die folgenden Worte gespeichert werden sollen. In Speicherzellen, die vom Programm selbst belegt werden, braucht vorher nichts eingespeichert werden (hier z.B. die Schnellspeicherzellen 190 bis 195).

Der Programmtext wird hintereinander geschrieben (es kann auch jeder Befehl untereinander geschrieben werden).

Zwischen den einzelnen Abschnitten und am Anfang und Ende wird mehrfach "Leer" gegeben, um Stellen zu erhalten, auf denen Beschriftungen angebracht werden können. Am Anfang ist ca. 15 cm Leerstreifen notwendig, damit man den Streifen in den Abtaster einlegen kann.

Im Protokoll erscheint dann folgendes Schriftbild
( ist mit der Hand hinzugesetzt, da die Taste wohl eine Lochung aber keinen Anschlag liefert):

U200U

B5 U220 B191 U6 B190 X B192 + B190 X B193 + B190 X B194 + B190 X B195 + Eo

U1750U

∫ X F(X) ∫ o

U1770U

∫ UNENDLICH ∫ o

U1780U

1 5,5 3,25 -6,375 -3,375

U1790U

1,65 3,135 -10,824 -17,127 -2,97

U1795U

o 0,125 3,01

U1800U

B1750 D F1000 B190+1795

B190 D CKB13+4 BV191+1780 F200 PPQQE1825 U189 CKB13+4 BV191+1790 F200 B189 : D

B6+1796 B190 + U190 B1797 - QQE1804 Zo

B1770 D F1000 E1817

U1030U

7' 2' 6' 2' 3' 4' -1'

E1800E

Speziell bei größeren Programmen oder solchen, bei denen mit verschiedenen Eingabewerten gerechnet werden soll, erstellt man zweckmäßigerweise zwei getrennte Lochstreifen. Der erste Lochstreifen wird das Programm, die benötigten Unterprogramme und die unveränderlichen Daten (Konstanten, Klartext usw.) und die Angabe zur Druckanordnung enthalten. Auf den zweiten Lochstreifen wird man die veränderlichen Eingabewerte lochen und den Startbefehl.

Für unser Beispiel wäre demnach auf den ersten Streifen zusammenzufassen:

U200U	Unterprogramm
U1750U	Überschrift
U1770U	Klartext
U1800U	Programm
U1030U	Druckanordnung

und auf den zweiten Streifen:

U1780U	Koeffizienten des Nennerpolynoms
U1790U	Koeffizienten des Zählerpolynoms
U1795U	Anfangswert, Schrittwerte, usw.
E1800E	Startbefehl

3.5 Eingabe des Lochstreifens

Der so hergestellte Lochstreifen kann nun in die Maschine eingegeben werden.

Dazu muß man den Lochstreifen in den photoelektrischen Abtaster einlegen und das Leseprogramm starten.

Das Leseprogramm kann in zwei Arten gestartet werden:

1. Start ab der Trommelzelle 1

Eo+1 Hierzu ist am Bedienungspult (vergl. 3.6) der Befehl Eo+1 in der Tastenreihe des Befehlsregisters einzustellen, d.h. die Befehlstaste und die Trommeladresse 1,

(das ist die erste und die letzte Taste) einzudrücken und alle anderen Tasten herauszunehmen.

Durch Betätigung der Taste Befehlsübernahme (Bef.Üb.) wird der eingestellte Befehl ins Befehlsregister übernommen und die entsprechenden Stellen leuchten nun auf. Allerdings leuchtet nicht die erste Stelle auf, sondern die zweite Stelle "Stop", da die Maschine ja in Stopzustand ist.

Mit der Taste "Start" läuft nun das Leseprogramm an und beginnt den Lochstreifen einzulesen.

Das Leseprogramm mit Eo+1 gestartet, überträgt die Übersetzungstabellen Fernschreibcode in Maschinencode in die Kernspeicher, so daß damit die Schnellspeicherzellen bis 99 während des Einlesens belegt sind. Falls man Programmteile in den Kernspeicher legen will, so kann man erst ab Kernspeicherzelle 100 die Speicher belegen.

Man kann jedoch immer so vorgehen: Alle Programmteile, sowohl die Trommel- als auch die Schnellspeicherprogrammteile werden während des Einlesens auf Trommelzellen gebracht. Nach dem Einlesen werden die Schnellspeicherprogrammteile während des Programmablaufs durch Geben der Transfer-Befehle von der Trommel in den Schnellspeicher übertragen und anschließend durch einen Sprungbefehl auf eine Schnellspeicherzelle in Tätigkeit gesetzt. Dabei können natürlich alle Schnellspeicher auch die unterhalb 99 benutzt werden.

Das Leseprogramm mit Eo+1 gestartet, braucht etwa 3 msec zum Entschlüsseln eines Zeichens. Für die Schlußverarbeitung wird noch einige Zeit benötigt.

2. Start ab Trommelzelle 6

Eo+6 Hierzu ist am Bedienungspult der Befehl Eo+6 in der Tastenreihe des Befehlsregisters einzustellen, d.h., die Befehlstaste und die Trommeladressen 4 und 2.

Durch "Befehlsübernahme" gelangt der Befehl ins Befehlsregister und "Start" läßt das Leseprogramm anlaufen.

Wenn das Leseprogramm mit Eo+6 gestartet wird, so werden nur die Schnellspeicherzellen bis 32 belegt und die Entschlüsselung der Fernschreibzeichen auf der Trommel durchgeführt.

Die Entschlüsselungszeit für ein Fernschreibzeichen beträgt hierbei 20 msec.

Diese Art des Einlesens dauert also länger, hat aber den Vorteil, daß nur weniger Schnellspeicherzellen belegt werden. Das ist besonders dann günstig, wenn man während der Rechnung einzelne Zahlen einlesen will.

Man hat hier die Möglichkeit, mit dem Befehl F522 ein Wort in den Akkumulator einzulesen.

Hierbei werden die Schnellspeicherzellen bis 32 benutzt, d.h. die gleichen Schnellspeicher, die nach einer Konvention auch bei den allgemeinen Unterprogrammen z.B. trigonometrische Funktionen, Logarithmus usw. belegt werden.

F522 1 Wort vom Abtaster in den Akkumulator lesen.

Das Ergebnis, das die Maschine bei unserem Beispiel ausschreibt hat dann folgende Form:

X	F(X)
0	0,880000
0,125000	1,283050
0,250000	1,683744
0,375000	2,143883
0,500000	2,748900
0,625000	3,675136
0,750000	5,434000
0,875000	10,562842
1,000000	UNENDLICH
1,125000	-9,612453
1,250000	-4,501304
1,375000	-2,773282
1,500000	-1,893375
1,625000	-1,354222
1,750000	-0,986480
1,875000	-0,717424
2,000000	-0,510593
2,125000	-0,345659
2,250000	-0,210373
2,375000	-0,096904
2,500000	-0,000000
2,625000	0,083995
2,750000	0,157711
2,875000	0,223090
3,000000	0,281600

Als weitere Besonderheiten des Leseprogramms seien erwähnt:

Bei den UmU-Bandbefehlen wird der Befehl Um in die Schnellspeicherzelle 19 gebracht; in dieser Zelle wird die Adresse jeweils nach Speichern eines eingelesenen Wortes um 1 erhöht.

Das Bandbefehlszeichen E kann auch an andere als Em-Befehle angehängt werden. Ein Bandbefehlszeichen E bewirkt, daß der Befehl nicht gespeichert, sondern sofort ausgeführt wird. Die Schnellspeicherzelle 26 dient hierbei als Hilfsakkumulator, da der Akkumulator selbst beim Lesen ja ständig benötigt wird. Die E-Bandbefehle wirken vom Lochstreifen direkt auf den Inhalt von Zelle 26 ebenso wie die gewöhnlichen Befehle im Programmablauf auf den Akkumulator wirken.

Z.B.

	Bandbefehle		gewöhnliche Befehle
B19E	<19> -> 26	B19	<19> -> a
U218E	<26> -> 218	U218	<a> -> 218
	<26> -> 26		<a> -> a

3.6 Das Bedienungspult

Am Bedienungspult sind, wie die Abbildung zeigt, vier Tastenreihen vorhanden. (vergl. 6.6)

Befehlsregister In der 2. Reihe von oben ist eine Tastatur für das Befehlsregister. Hier ist für jede einzelne Binärstelle eines Befehlswortes eine Taste vorgesehen. Einmal kann man über diese Tastatur einzelne Befehle eingeben, zum anderen zeigen die Lampen in den Tasten den jeweiligen Stand des Befehlsregisters an. Die ersten beiden mit "Start" und "Stop" bezeichneten Tasten sind die Kennzeichenstellen. Bei nicht laufender Maschine leuchtet stets nur die zweite Lampe auf.

Die nächsten 5 Stellen bezeichnen die Bedingungszeichen PP, P, QQ, Q, Y.

Danach kommen die weiteren internen Befehlszeichen von C bis V. Eine Stelle E ist nicht gesondert vorhanden, da die Kombination, die weder U noch A noch S enthält mit dem Buchstaben E bezeichnet ist.

Die nächsten 8 Stellen sind die Stellen der Schnellspeicheradressen, die sich aus den einzelnen Zweierpotenzen 1, 2, 4, 8, 16, 32, 64, 128 zusammensetzen.

Und entsprechend sind die letzten 13 Stellen die Trommeladressen.

- | | |
|-----------------------|--|
| Befehls-
übernahme | Ein in die Tastatur eingetasteter Befehl wird nach Betätigung der Taste "Befehlsübernahme" in der untersten Reihe in das Befehlsregister übernommen und leuchtet dann mit den Lampen auf. |
| Weiter | Durch "Weiter" in der untersten Zeile wird der gerade im Befehlsregister stehende Befehl ausgeführt, danach stoppt die Maschine. |
| Start | Mit "Start" in der untersten Reihe kann die Rechnung begonnen werden. |
| Stop | Mit "Stop" in der untersten Reihe kann die laufende Maschine angehalten werden. (Die 3 Tasten "Weiter", "Start" und "Stop" sind je zweimal in der untersten Reihe vorhanden). |
| Akkumu-
lator | In der 3. Reihe von oben ist für den Akkumulator bis auf die ersten beiden Stellen die gleiche Bezeichnung angebracht wie beim Befehlsregister. Die Lampen zeigen den jeweiligen Inhalt des Akkumulators an. |

- Akku-
übernahme Über die Tastatur kann 1 Wort eingestellt und durch die Taste "Akkuübernahme" in den Akkumulator übernommen werden. Die Einstellung der Akkumulatortastatur kann auch während des Programms durch den Befehl U4 in den Akkumulator übertragen werden.
- Laut-
sprecher In der unteren Reihe sind noch Schalter, mit denen Lautsprecher ein- und ausgeschaltet werden können.
- Wort-
transport Beim Rechnen wird zur Überwachung der Maschine bei jedem Worttransport in der Maschine ein Signal auf einen Lautsprecher gegeben, so daß man den Lauf der Maschine akustisch verfolgen kann. Mit einer Taste kann dieser Lautsprecher ausgeschaltet, mit einer zweiten können 2 Lautstärken eingestellt werden.
- Alarm Daneben ist noch ein Schalter, mit dem der "Alarm" Lautsprecher ausgeschaltet werden kann. Dieser Lautsprecher gibt ein Signal (Dauerton), wenn bei der Ablesung einer Speicherzelle die Quersummenkontrolle nicht stimmt; d.h. beim Beschreiben einer Zelle wird neben den 40 Bits der Information noch eine 41. Stelle mitgeschrieben, die die Zahl der Einsen in dem Wort auf eine gerade Anzahl ergänzt. Beim Ablesen wird dann wieder kontrolliert, ob in diesen 41 Stellen eine gerade Anzahl von Einsen vorliegt. Wenn das nicht der Fall ist, so stoppt die Maschine und gibt über den "Alarm" Lautsprecher ein akustisches Signal. Daneben leuchtet auch eine der mit "Alarm" bezeichneten Lampen auf, die anzeigt, ob die Fehlablesung von der Trommel oder vom Schnell-speicher erfolgt. Eine dritte Lampe zeigt an, wenn eine Netzstörung vorlag. Für Prüfzwecke lassen sich die "Alarme", die ja einen Stop der Maschine hervorrufen, ausschalten.

Nacht Die erste Taste links oben bewirkt, wenn sie eingeschaltet ist, daß sich die Maschine abstellt, sobald ein Stop eintritt. Man kann damit die Maschine abends weiterrechnen lassen und sie schaltet sich selbst ab, wenn die Rechnung beendet ist.

Trommel In der oberen Zeile sind zunächst Tasten zum Ein-
Maschine und Ausschalten der Trommel, der Maschine sowie
FS der Ein- und Ausgabegeräte angebracht.
Abtaster

FS Der Fernschreiber kann auch ohne die Maschine eingeschaltet werden, so daß man hier auch Programme und Zahlenwerte ablocken kann. Wird bei eingeschalteter Maschine die Taste "FS" ausgestellt, so schaltet sich der Fernschreiber selbsttätig ab, wenn er länger als 30 sec nicht angesprochen wurde. Bei "FS" ein, wird die automatische Abschaltung unterbunden.

Abtaster Der Abtaster kann auf einen möglicherweise vorhandenen Drucker umgeschaltet werden (Abtaster auf Drucker).

<2> 1 Die mit <2>₁ bzw. <3>₄₀ bezeichneten Lampen zeigen
<3> 40 an, was sich in den entsprechend bezeichneten Stellen der Schnellspeicherzelle 2 bzw. 3 gerade befindet, Aufleuchten bedeutet eine 1, Erlöschen eine 0.

Schreib- Durch Drücken dieser Tasten kann man die jeweils
sperre angegebenen Gruppen von je 1 o24 Speicherzellen der Trommel sperren, d.h. man kann diese Zellen wohl ablesen, aber man kann nichts in diese Zellen schreiben. Auf diese Weise kann man verhindern, daß Programmteile oder Zahlenwerte versehentlich überschrieben werden.

Löschung RO +1 Addw. Diese Tasten sind speziell für die Prüfung der Maschine vorgesehen.

Bedingungs- schalter Hier kann man zur Beeinflussung eines Programmablaufs 32 verschiedene Kombinationen der 5 Tasten einstellen, die vom Programm unter der Adresse 14 als Strichzahlen von 0 - 31' abgefragt werden können.

Adressen- stop Die restlichen Tasten der oberen Reihe sind zur Einstellung einer Adresse vorgesehen, an der die Maschine stoppen soll. Sobald die eingetastete Adresse irgendwie angesprochen wird, tritt ein Stop der Maschine ein. Diese Stopeinrichtung, die besonders beim Programmtest wertvolle Dienste leisten kann, wird erst nach Einschalten der 1. Taste "Ein" wirksam.

Beim Adressenstop, Befehlsregister und Akkumulator sind die Adressen durch farbige Gravierung in 3er Gruppen eingeteilt. Durch diese Einteilung soll die Eintastung von beliebigen Adressen erleichtert werden. Der Programmierungsanleitung liegt eine besondere Liste "Oktaltabelle" bei.

In dieser Tabelle ist jede 16. Adresse aufgenommen und neben diesen Strichzahlen stehen 5stellige Zahlen, die nur die Ziffern 0 bis 7 aufweisen (Oktalzahl).

Diese Ziffern geben an, welche Zahl jeweils in den zugeordneten 3er Gruppen als Binärzahl eingetastet werden soll. Z.B. für die Adresse 2512' findet man 04720.

Es ist dann die im unteren Schema gekennzeichnete Stelle einzutasten.

Adressenstellen im Befehlsregister

4096	2048	1024	512	256	128	64	32	16	8	4	2	1
	x			x	x	x		x				
0		4			7			2				0

Die einzutastende Oktalzahl für Adressen die zwischen den angegebenen liegen, kann man durch Ergänzung in den letzten 4 Stellen leicht selbst aufbauen.

4. Adressenrechnen und Zählen

4.1 Rechnen mit Strichzahlen

Da die Strichzahlen einen ganz anderen internen Aufbau als die Gleitkommazahlen haben, können sie nicht mit den Befehlen für die arithmetischen Operationen im gleitenden Komma verknüpft werden.

Die Addition und Subtraktion dieser Strichzahlen werden nicht durch Unterprogramme, sondern direkt durch in der Maschine verdrahtete Elementaroperationen ausgeführt.

Addition von Strichzahlen

Am $\langle a \rangle' + \langle m \rangle' \rightarrow a$

Zum Akkumulatorinhalt wird der Inhalt der Speicherzelle m addiert und das Ergebnis wird im Akkumulator notiert.

Das Apostroph an den spitzen Klammern soll besonders daraufhinweisen, daß die Speicherinhalte als Strichzahlen aufgefaßt werden. Es kann sich hierbei z.B. auch um einen Befehl handeln, dessen Adresse durch Addition einer Strichzahl verändert wird.

Subtraktion von Strichzahlen

Sm $\langle a \rangle' - \langle m \rangle' \rightarrow a$

Hier wird der Inhalt der aufgerufenen Speicherzelle m vom Akkumulatorinhalt abgezogen.

Umgekehrte Subtraktion

USm $\langle m \rangle' - \langle a \rangle' \rightarrow a$

Es wird wiederum eine Subtraktion von Strichzahlen ausgeführt, jedoch gegenüber dem S-Befehl mit vertauschten Operanden, d.h. der Inhalt des Akkumulators wird von dem Speicherinhalt abgezogen. Das Ergebnis geht wieder in den Akkumulator.

Das zusätzliche Befehlszeichen N bedeutet Nullsetzen.
Wenn man es zu A oder S setzt, wird vor der Addition bzw.
Subtraktion der Akkumulator auf Null gesetzt.
NA wird im allgemeinen mit B abgekürzt.

NA_m ≡ B_m <m> → a

Der aufgerufene Speicherinhalt wird in den Akkumulator gebracht.

NS_m -<m>' → a

Die in Speicherzelle m stehende Zahl wird - als Strichzahl aufgefaßt - negativ in den Akkumulator gebracht.

(NUS_m hat die gleiche Wirkung wie B_m)

<m> → a

Diese Befehle lassen sich auch mit zwei Adressen versehen, dann kann man den Inhalt der aufgerufenen Trommelzelle direkt in die angegebene Schnellspeicherzelle hineinaddieren ohne den Hauptakkumulator (a) zu zerstören.

AS+t <s>' + <t>' → s <a> → a

Der Inhalt der aufgerufenen Trommelzelle t wird zum Inhalt der Schnellspeicherzelle s addiert und das Ergebnis geht in die Schnellspeicherzelle s. Der Akkumulatorinhalt <a> bleibt erhalten.

SS+t <s>' - <t>' → s <a> → a

Vom Inhalt der Schnellspeicherzelle s wird der Inhalt der Trommelzelle t subtrahiert.

USs+t <t>' - <s>' → s <a> → a

Wiederum Subtraktion mit vertauschten Operanden.

Auch den Doppeladressbefehlen kann ein N = Nullsetzen hinzugesetzt werden, das jetzt auf die aufgerufene Schnellspeicherzelle s wirkt.

So entsteht der schon früher behandelte Übertragungsbefehl von der Trommel zum Schnellspeicher.

$Bs+t = NAs+t$ $\langle t \rangle \rightarrow s$ $\langle a \rangle \rightarrow a$

und entsprechend

$NSs+t$ $-\langle t \rangle' \rightarrow s$ $\langle a \rangle \rightarrow a$

Hier wird der Inhalt der Trommelzelle t als Strichzahl aufgefaßt mit umgekehrten Vorzeichen in die Schnellspeicherzelle s gebracht.

$(NUSs+t \equiv Bs+t$ $\langle t \rangle \rightarrow s$ $\langle a \rangle \rightarrow a$)

Da beim Aufbau von Adressen u. ä. auch häufig einmal eine Multiplikation auftreten kann, sind in das Grundprogramm auch Unterprogramme mitaufgenommen worden, die es erlauben Strichzahlen miteinander zu multiplizieren oder durcheinander zu dividieren.

Hierbei wurde die gleiche Konvention wie bei Gleitkommaoperationen verwendet, daß der 1. Operand in der Schnellspeicherzelle 6 und der 2. Operand im Akkumulator steht, das Ergebnis in den Akkumulator und die Schnellspeicherzelle 6 gebracht wird.

Die Unterprogramme für Strichzahl-Multiplikation und -Division benötigen als Zwischenspeicher die Schnellspeicherzellen 2 - 6.

Die Operationen mit Strichzahlen werden durch vercodete Rufbefehle aufgerufen, die wie die der Gleitkommaoperation den allgemein gebräuchlichen Zeichen entsprechen. Für die Strichzahloperation werden diese Zeichen zusätzlich mit einem Apostroph versehen.

X' Multiplikation von Strichzahlen

$\langle 6 \rangle' \cdot \langle a \rangle' \rightarrow a,6$

: ' Division von Strichzahlen

$\langle 6 \rangle' : \langle a \rangle' \rightarrow a,6$

Falls bei der Multiplikation größere Zahlen als 2^{38} entstehen oder wenn durch 0 dividiert werden soll, schreibt die Maschine zur Kenntlichmachung der Bereichsüberschreitung

BUEB Em

und stoppt. m ist die Stelle des Programms, wo das Programm nach Ausführung der Operation fortfahren sollte.

Beispiel

Es liegt eine Matrix vor

$$A = \begin{pmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1m} \\ a_{21} & a_{22} & a_{23} & \dots & a_{2m} \\ \dots & \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & a_{n3} & \dots & a_{nm} \end{pmatrix}$$

mit n Zeilen und m Spalten.

Innerhalb der Maschine sei ab Trommelzelle 2000 fortlaufend gespeichert

2000		n'	die Zeilenzahl
2001		m'	die Spaltenzahl
2002		a ₁₁	} die Koeffizienten zeilenweise hintereinander
		a ₁₂	
		⋮	
		⋮	
		a _{1m}	
		a ₂₁	
		a ₂₂	
		⋮	
		⋮	
		a _{nm}	

Beim Rechnen mit dieser Matrix soll nun bestimmt werden, in welcher Speicherzelle das letzte Element der Matrix steht, d.h. es muß $n' \cdot m'$ gebildet werden und dieses zur Adresse 2001 hinzuaddiert werden, um die Adresse von a_{nm} zu bekommen.

Der Teil des Programms, der diese Umrechnung vornimmt, würde lauten, wenn die Strichzahl 2001' in Schnellspeicherzelle 40 steht

B6+2000	$n' \rightarrow 6$
B2001	$m' \rightarrow a$
X'	$\langle 6 \rangle' \cdot \langle a \rangle' = n' \cdot m' \rightarrow a,6$
A40	$\langle a \rangle' + \langle 40 \rangle' = n' \cdot m' + 2001 \rightarrow a$
.	
.	
.	

Eine weitere Ausführung von Strichzahloperationen wird in späteren Beispielen behandelt werden.

4.2 Konstantenzuführung aus dem Befehlsregister

Beim Aufbau von Adressen oder bei Zählungen braucht man häufig Strichzahlen, die addiert oder subtrahiert werden sollen. Man könnte die benötigten Strichzahlen in irgendwelche Speicherzellen bringen und sie aus diesen mit Am- oder Sm-Befehlen holen.

Durch die Möglichkeit der Konstantenzuführung aus dem Befehlsregister erübrigt sich jedoch eine extra Speicherung der Strichzahlen.

Das Befehlszeichen C bewirkt, daß die in den Adressenstellen des Befehl stehende Zahl nicht zur Anwahl einer Speicherzelle benutzt, sondern dem Addierwerk als Strichzahl zugeführt werden. Welche Operation mit der ankommenden Strichzahl ausgeführt wird, geben die übrigen Operationszeichen A, S, US an.

CAn $\langle a \rangle' + n' \rightarrow a$

Es wird die im Befehl selbst angegebene Zahl n' zum Inhalt des Akkumulators addiert.

Man beachte genau den Unterschied der Befehle

An : $\langle a \rangle' + \langle n \rangle' \rightarrow a$

CAn : $\langle a \rangle' + n' \rightarrow a$

CSn $\langle a \rangle' - n' \rightarrow a$

CUSn $n' - \langle a \rangle' \rightarrow a$

CBn $n' \rightarrow a$

CNSn $-n' \rightarrow a$

Mit den C-Befehlen lassen sich Strichzahlen bis

$$n' = 2^{22} - 1 = 4194303'$$

dem Hauptakkumulator zuführen.

Die Zahl n' kann wie oben in dem Befehl enthalten sein; man kann sie jedoch auch in ganze Vielfache von 8192 und Rest zerlegen, also $n' = 8192 \cdot s' + t'$ und die Befehle in der Form CA_{s+t} , CS_{s+t} usw. schreiben.

Diese Form eignet sich dann wenn man z.B. die Schnell-speicheradressenstellen als Strichzahl aufgefaßt zu einem Wort addieren will; es erübrigt sich dann das Ausrechnen der Konstanten.

Da die letzte Stelle des Operationsteiles in diesem Fall noch mit zur Konstante gehört, kann hier s maximal 511 werden (im Gegensatz zur Schnellspeicheranwahl, die nur bis 255 geht). t kann höchstens 8191 sein.

Beispiel:

Mit einem kleinen Programm sollen die Speicherzellen 2001 bis 2056 ausgeschrieben werden.

Das Programm soll in Zelle 6000 anfangen.

6000	CNS56	-56' → a	
1	U35	<a> → 35	
2	B2001	<2001> → a	
3	D	<a> → FS	
4	B35	} <35>' + 1' → a	
5	CA1		
6	PPZo		<a> = 0 ? Schon 56 Worte ausgedruckt? Wenn ja, Stop.
7	U35	<a> → 35	
8	B6002	<6002> → a	} Mit diesen 3 Befehlen wird die Adresse des in Speicherzelle 6002 stehenden Befehls um 1 erhöht.
9	CA1	<a>' + 1' → a	
10	U6002	<a> → 6002	
11	E6002		Beim 1. Rücksprung auf die Speicherzelle 6002 findet die Maschine dort den Befehl B2002 beim 2. Rücksprung B2003 beim 3. Rücksprung B2004 usw. vor.

Dieses kleine Beispiel zeigt, wie ein Programm sich selbst abändern kann.

Bei den Befehlen in Zelle 6008 bis 6010 wird der Befehlstext, der aus der Zelle 6002 geholt wird, als Strichzahl aufgefaßt. Der Befehlstext wird erst dann zu einem auszuführenden Befehl, wenn er im Befehlsregister steht.

Für Zählungen ist es häufig notwendig, Inhalte von Schnellspeicherzellen zu vergrößern oder zu verkleinern.

Das kann man natürlich tun, indem man den Inhalt der Schnellspeicherzelle in den Akkumulator bringt, mit CA- bzw. CS-Befehlen die Veränderung vornimmt und wieder wegspeichert, wie es im obigen Beispiel in Zelle 6004 - 6007 durchgeführt wird.

Dabei wird allerdings der ursprüngliche Inhalt des Akkumulators zerstört.

Mit einer Kombination der C-Befehle läßt sich eine solche Änderung in einen beliebigen Schnellspeicher ausführen, ohne den Inhalt des Hauptakkumulators zu berühren. Diese Kombination ist CK.

Hierbei gibt die Schnellspeicheradresse den Speicher an, in dem gezählt werden soll und die Trommeladresse die Änderung. Der übrige Operationsteil (A, S, US, B, NS) gibt an, ob addiert, subtrahiert usw. werden soll.

CKAs+t	$\langle s \rangle' + t' \rightarrow s$	$\langle a \rangle \rightarrow a$
CKSs+t	$\langle s \rangle' - t' \rightarrow s$	$\langle a \rangle \rightarrow a$
CKUSs+t	$t' - \langle s \rangle' \rightarrow s$	$\langle a \rangle \rightarrow a$
CKBs+t	$t' \rightarrow s$	$\langle a \rangle \rightarrow a$
CKNSs+t	$-t' \rightarrow s$	$\langle a \rangle \rightarrow a$

Man beachte dabei, daß hier infolge des zusätzlichen Kennzeichens K nicht s und t als Konstante wirken, sondern s den Schnellspeicher s anwählt und nur t als Konstante wirkt. t' kann deshalb nur maximal den Wert 8191' annehmen.

Mit Hilfe dieser Befehle würde das obige Beispiel zum Ausdrucken der Zelleninhalte 2001 ff lauten:

6000	CKNS35+56	-56 → 35
1	B2001	} Zelleninhalte drucken
2	D	
3	CKA35+1	$\langle 35 \rangle' + 1' \rightarrow 35$
4	PZ35+0	wenn $\langle 35 \rangle \geq 0$ Stop
5	B6001	} Verändern der Adresse des Befehls in Zelle 6001
6	CA1	
7	U6001	
8	E6001	Rücksprung auf 6001

4.3 Logische Operationen

Neben den arithmetischen Operationen kann die Maschine auch logisch operieren. Sie kann "Und" (Konjunktion) und "Oder" (Disjunktion) Operationen ausführen.

Die "Und" - Operation wird in der Maschine mit Intersektion bezeichnet und wird binärstellenweise mit den beiden aufgerufenen Speicherinhalten durchgeführt. Als Ergebnis, das stets im Akkumulator steht, erscheinen nur in den Stellen eine Eins, in denen in beiden eingehenden Zahlen eine Eins gestanden hat.

Die Intersektion wird zum Herausschneiden bestimmter Teile benutzt, z.B. zum Heraustrennen der Trommeladressen oder des Operationsteils.

Im $\langle a \rangle \wedge \langle m \rangle \rightarrow a$

Bei Intersektionsbefehlen mit zwei Adressen werden die beiden Speicherinhalte zur Konjunktion gebracht.

Is+t $\langle s \rangle \wedge \langle t \rangle \rightarrow a$ $\langle s \rangle \rightarrow s$

Hier ist besonders zu beachten, daß das Ergebnis in den Akkumulator geht und der aufgerufene Schnellspeicherinhalt erhalten bleibt.

Die Intersektion kann auch mit dem negativ gemachten Inhalt der aufgerufenen Speicherzelle durchgeführt werden.

ISm $\langle a \rangle \wedge (-\langle m \rangle) \rightarrow a$

ISs+t $\langle s \rangle \wedge (-\langle t \rangle) \rightarrow a$ $\langle s \rangle \rightarrow s$

Meist wird die Intersektion mit Konstanten durchgeführt, die durch C-Befehle aus dem Befehlsregister zugeführt werden.

CIn $\langle a \rangle \wedge n' \rightarrow a$

CISn $\langle a \rangle \wedge (-n') \rightarrow a$

Es seien hier noch einige spezielle Befehle angegeben:

CI8191 : $\langle a \rangle \wedge 8191' \rightarrow a$

d.h. nach Ausführung des Befehls steht im Akkumulator nur noch die Trommeladresse des ursprünglich im Akkumulator stehenden Befehls.

CI255+o

nach Ausführung des Befehls steht nur noch die Schnell-
speicheradresse des ursprünglich im Akkumulator stehenden
Befehls

CIS1+o was auch geschrieben werden kann CIS8192

Der Operationsteil und die Schnellspeicheradresse des im
Akkumulator stehenden Befehls bleibt im Akkumulator, während
die Trommeladresse gelöscht wird.

CIS256+o

Der Operationsteil wird herausgetrennt, d.h. Schnellspeicher-
adresse und Trommeladresse werden auf Null gesetzt.

Die Intersektion kann auch mit CK-Befehlen durchgeführt
werden.

CKIs+t $\langle s \rangle \wedge t' \rightarrow a$ $\langle s \rangle \rightarrow s$
CKISs+t $\langle s \rangle \wedge (-t') \rightarrow a$ $\langle s \rangle \rightarrow s$

Als Beispiel soll die Trennung einer Gleitkommazahl in
Mantisse und Exponent vorgenommen werden (interne Darstel-
lung einer Gleitkommazahl vergl. 2.2.2).

Die Zahl stehe in Schnellspeicherzelle 6, der Exponent soll
in die Zelle 16, die Mantisse in die Zelle 17 gebracht
werden, wobei die Zahl selbst erhalten bleiben soll.

CKI6+255 $\langle 6 \rangle \wedge 255' \Rightarrow$ Exponent $\rightarrow a$ $\langle 6 \rangle \rightarrow 6$
U16 Exponent $\rightarrow 16$
CKIS6+256 $\langle 6 \rangle \wedge (-256') \Rightarrow$ Mantisse mit Vorzeichen-
 stelle $\rightarrow a$ $\langle 6 \rangle \rightarrow 6$
U17 Mantisse $\rightarrow 17$

Die "Oder"-Operation bringt eine disjunktive Überlagerung
der beiden aufgerufenen Worte, d.h. es erscheinen im
Ergebnis nur in den Stellen Nullen, die in beiden ein-
gehenden Worten Null waren.

Die "Oder"-Operation wird mit dem Befehlszeichen \bigcirc aufgerufen, das hier zur besseren Unterscheidung gegenüber der Null mit \bigcirc bezeichnet werden soll. Am Fernschreiber erscheint für die Buchstaben \bigcirc ein großer Buchstabe, während für die Ziffer Null ein kleines o geschrieben wird.

$\bigcirc m$ $\langle a \rangle \vee \langle m \rangle \rightarrow a$
 $\bigcirc s+t$ $\langle s \rangle \vee \langle t \rangle \rightarrow s$ $\langle a \rangle \rightarrow a$

Bei Doppeladressbefehlen wird das Ergebnis der Disjunktion in die aufgerufene Schnellspeicherzelle geliefert, während der Inhalt des Hauptakkumulators erhalten bleibt.

$\bigcirc n$ $\langle a \rangle \vee n' \rightarrow a$
 $\bigcirc K \bigcirc s+t$ $\langle s \rangle \vee t' \rightarrow s$ $\langle a \rangle \rightarrow a$

4.4 Adressensubstitution

Die ZUSE Z 23 hat die Möglichkeit der automatischen Adressenmodifikation. Diese Befehle, die das Befehlszeichen G enthalten, wirken in 2 Schritten.

Im 1. Schritt wird, durch das G bzw. die Kombinationen CG, GK oder CGK ausgelöst, eine Adresse aufgebaut, mit der im 2. Schritt, die durch die übrigen Befehlszeichen angegebene Operation ausgeführt wird.

Es können bei diesem Adressenaufbau sowohl Trommeladressen (G-Kombination ohne K) als auch Schnellspeicheradressen (G-Kombination mit K) aufgebaut werden.

In den folgenden Befehlsbeschreibungen wird der im 1. Schritt der Adressenumrechnung nicht wirksame Operationsteil mit \ominus bezeichnet;

\ominus kann für B, U, NU, A, S, US, E usw. stehen.

4.4.1 Einfache Adressensubstitution

Hier wird aus dem angegebenen Speicher erst die richtige Trommeladresse geholt, mit der der Befehl ausgeführt werden

Es soll also die Funktion

$$F(x) = \frac{P_6(x)}{P_8(x)}$$

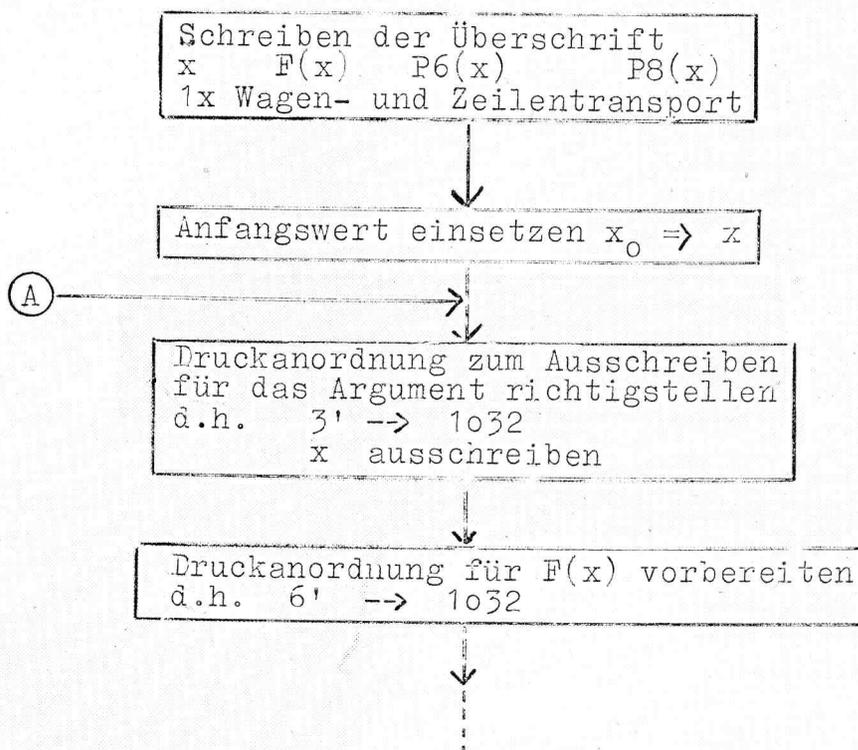
tabelliert werden, von $x_0 = 0$ bis $x_n = 3$ mit einer Schrittweite $\Delta x = 0,125$.

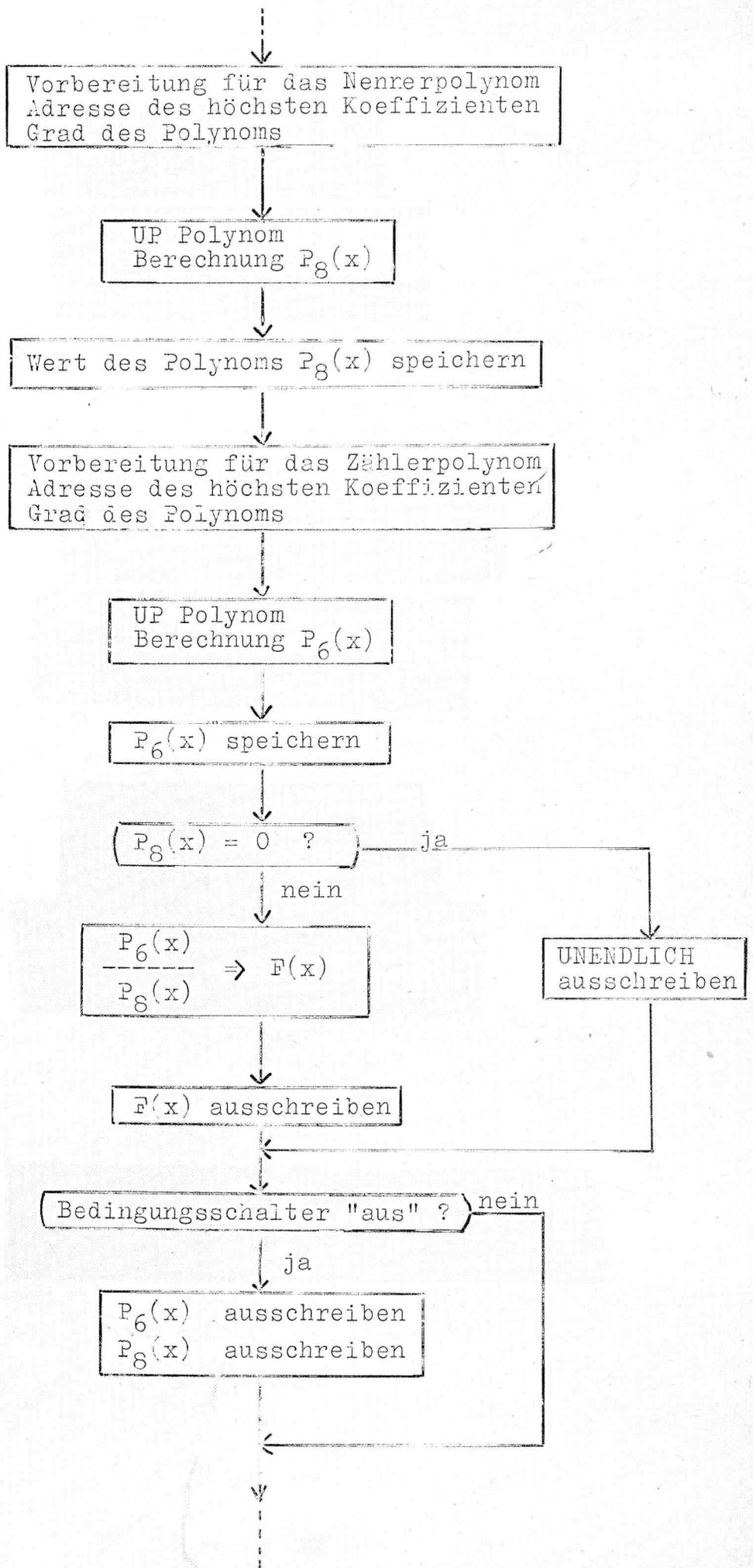
Da das Argument stets nur 3 Stellen nach dem Komma hat, sollen hier auch nur diese Stellen herausgeschrieben werden, während das Ergebnis mit 6 Stellen nach dem Komma erscheinen soll. Hierin ist ein Beispiel zur Änderung der Druckanordnung gegeben.

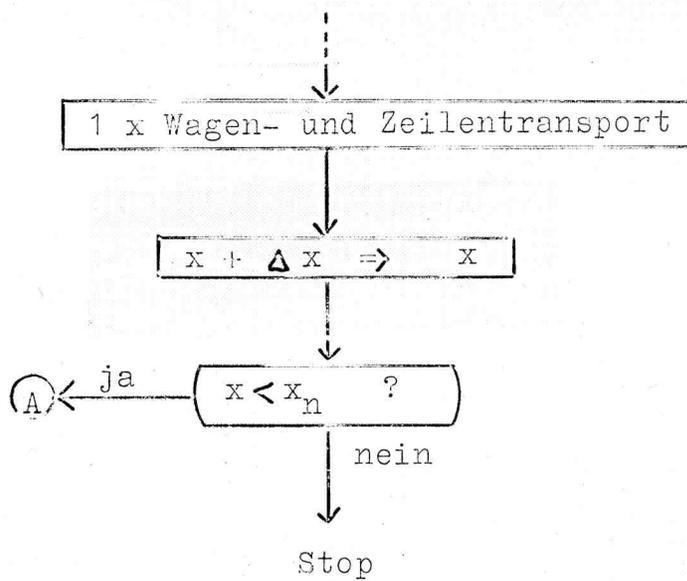
Außerdem sollen als Anwendung des Bedingungsschalters - die Werte des Zähler- und Nennerpolynoms nach dem Funktionswert zusätzlich ausgeschrieben werden, wenn der Bedingungsschalter "aus"-geschaltet ist. Wenn dieser Schalter eingetastet ist, soll nur Argument und Funktionswert ausgeschrieben werden.

Das Unterprogramm Polynom soll so gestaltet sein, daß es für beliebigen Grad des Polynoms arbeitet.

Das Flußdiagramm hat dann folgendes Aussehen:



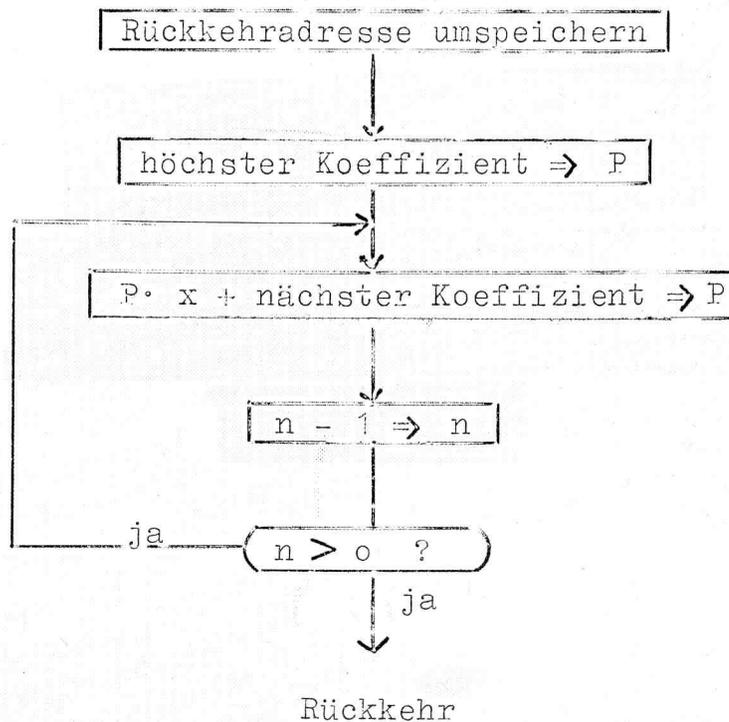




Beim Sprung in das Unterprogramm sollen folgende Angaben gemacht sein:

1. Das Argument in Schnellspeicher 30
2. Die Adresse des höchsten Koeffizienten in 31 (es sei vorausgesetzt, daß die Koeffizienten hintereinander $a_n, a_{n-1}, a_{n-2} \dots$ in Trommelzellen gespeichert sind).
3. Der Grad des Polynoms als negative Strichzahl in 32.

Das Flußdiagramm hat dann die Gestalt:



Das Unterprogramm für die Polynomberechnung soll wieder im Kernspeicher stehen in den Zellen 200 ff.

Speicherliste Polynom

Zellen-Nr.	Inhalt	Erläuterungen
200	B5	} Rücksprungbefehl notieren
1	U212	
2	GB31+0	} höchster Koeffizient $\rightarrow 6 \Rightarrow P$
3	U6	
4	B30	} P. x + nächster Koeffizient $\rightarrow 6 \Rightarrow P$
5	X	
6	CGB31+1	} $n-1 \Rightarrow n$ bzw. $-n+1 \Rightarrow -n$
7	+	
8	CKA32+1	} $-n < 0$ dann Sprung auf 204
9	B32	
10	QQE204	} hole Wert des Polynoms nach a Rücksprungbefehl wird vom Programm selbst eingesetzt.
11	B6	
12	E0	

Speicherverteilung für das Hauptprogramm
und sonstige Daten

6000 ff	Programm
6080 ff	Anfangs- und Endwert, Schrittweite
6100 ff	Überschrift in Klartext: x F(x) P6(x) P8(x)
6120 ff	Klartext: UNENDLICH
6150 ff	Koeffizienten des Zählerpolynoms $a_6, a_5 \dots a_0$
6160 ff	Koeffizienten des Nennerpolynoms $b_8, b_7 \dots b_0$

Speicherliste

6000	B6100	Überschrift schreiben
1	D	
2	F1000	Wagen- und Zeilentransport

3	B30+6080	$x_0 \Rightarrow x$
4	CB3	} 3 Stellen nach dem Komma
5	U1032	
6	B30	
7	D	} Argument schreiben
8	CB6	} 6 Stellen nach dem Komma für Druck von F(x)
9	U1032	
10	CKB31+6160	
11	CKNS32+8	Adresse von $b_8 \rightarrow 31$ -8' $\rightarrow 32$
12	F200	U.P. Polynom $P_8(x) \rightarrow a,6$
13	U35	$P_8(x) \rightarrow 35$
14	CKB31+6150	Adresse von $a_6 \rightarrow 31$
15	CKNS32+6	-6' $\rightarrow 32$
16	F200	U.P. Polynom $P_6(x) \rightarrow a,6$
17	U36	$P_6(x) \rightarrow 36$
18	B35	} wenn $P_8(x) = 0$ Sprung nach 6037
19	PPQQE6037	
20	:	} $P_6(x)$ $\frac{P_6(x)}{P_8(x)} \Rightarrow F(x)$ schreiben
21	D	
22	NS14	} wenn Bedingungsschalter "ein" Sprung nach 6028
23	QQE6028	
24	B36	} $P_6(x)$ $P_8(x)$ } schreiben
25	D	
26	B35	
27	D	
28	F1000	Wagen- und Zeilentransport
29	B6+6082	$\Delta x \rightarrow 6$
30	B30	} $x + \Delta x \Rightarrow x$
31	+	
32	U30	
33	B6081	} $x - x_n$ wenn $x - x_n < 0$ Sprung nach 6004 sonst Stop
34	-	
35	QQE6004	
36	Z0	
37	B6120	Ausschreiben des Textes
38	D	UNEDNLICH
39	E6022	Sprung nach 6022

6080		x_0
1		$x_n + \epsilon$
2		Δx
6100 ff	$\uparrow \dots X \dots F(x) \dots$	$P6(x) \dots P8(x) \uparrow \circ \equiv$ Klartext für die Überschrift
6120 ff	\uparrow UNENDLICH $\uparrow \circ \equiv$)	Klartext als Erläuterung
6150		a_6
1		a_5
2		a_4
.		.
.		.
.		.
6156		a_0
6160		b_8
1		b_7
.		.
.		.
.		.
6168		b_0

In 6080 ff, 6150 ff und 6160 ff müssen die entsprechenden Werte eingesetzt werden.

\equiv) Da die Maschine den Klartext fortlaufend ab Zelle 6100 so lange druckt, solange in den Zellen Klartext vorhanden ist, muß als Kennzeichnung am Ende des Klartextes etwas stehen, was vom Klartext verschieden ist, z.B. \circ .

5. Relatives und adressenloses Programmieren

5.1 Relativadressen

Beim Aufstellen von Programmen will man sich häufig nicht auf bestimmte Adressen festlegen. Das trifft besonders für Unterprogramme zu, die bei den verschiedensten Programmen verwendet werden sollen.

Man möchte die Programme möglichst so aufstellen, daß man sie an eine beliebige Stelle der Trommel speichern kann. Diese Möglichkeit ist durch die Relativadressen gegeben. Man stellt hier die Programme zunächst so auf, als ob sie ab Trommelzelle 0 stehen würden und bezeichnet alle Adressen, die sich auf eine solche Trommelzelle relativ zu Anfang beziehen (Relativadresse) mit einem Bandbefehlszeichen (Am). Beim Einlesen wird dann zu der schon angegebenen Adresse die im Speicher m stehende tatsächliche Anfangsadresse (Absolutadresse) addiert und der Befehl mit der fertigen Adresse weggespeichert. Im Programmablauf haben Bandbefehle also keinerlei Bedeutung mehr, nur beim Einlesen bewirken sie die Adressierung der Befehle in Abhängigkeit von der jeweiligen Stelle, an der das Programm eingespeichert wird.

Das Bandbefehlszeichen für die Relativadresse wird mit ...Am direkt hinter der Adresse angegeben und hat die Wirkung, daß vor Wegspeichern des Befehls ... der Inhalt der Speicherzelle m addiert wird.

...Am <m> zum eingelesenen Befehl ... addieren

Als Speicher m, aus dem die Absolutadresse hinzuaddiert wird, kann Schnellspeicherzelle 26 oder jede Schnellspeicherzelle 100 bis 255 oder eine beliebige Trommelzelle verwendet werden. In diese Speicherzelle muß vor Einlesen des Programms die richtige Anfangsadresse eingegeben werden. Sehr häufig wird die Schnellspeicherzelle 26 für diese Änderungen benutzt, da diese Zelle sich leicht mit der entsprechenden Anfangsadresse füllen läßt. Das ist auf zwei Arten möglich:

- 1) Dem Programm, das in Relativadressen geschrieben ist, wird ein gesonderter Vorstreifen vorangeschickt

UmU
Zo+1E

UmU bestimmt die Stelle m, ab wo das Programm eingelesen werden soll (das U-Bandbefehlszeichen bewirkt, daß der Befehl Um nach Schnellspeicherzelle 19 gebracht wird).
vergl. 3.5.

Zo+1E bewirkt nach Einlesen des UmU-Befehls einen Stop der Maschine.

Das eigentliche Programm beginnt mit einem Bandbefehl

CKI19+8191E

.....

Dieser Bandbefehl bewirkt, daß der Trommeladressenteil des Inhalts von 19, also die Anfangsadresse m', in die Schnellspeicherzelle 26 gebracht wird (vergl. 3.5).

Auf den Inhalt von 26 als Anfangsadresse kann dann beim Einlesen durch die Bandbefehle ...A26 Bezug genommen werden.

Falls auf eine andere Zelle als 26 bezogen werden soll, so müßte nach dem CKI19+8191E - Befehl noch ein entsprechender Umspeicherbandbefehl folgen.

UnE <26> -> n

- 2) Dem Programm wird ein Vorstreifen vorausgeschickt

CBmE
Zo+1E

CBmE bewirkt, daß m' in die Schnellspeicherzelle 26 gebracht wird.

Zo+1E bewirkt Stop vom Lochstreifen her.

Das eigentliche Programm beginnt dann mit einem Bandbefehl

UoA26U

.....

Hier sind zwei Bandbefehlszeichen in einem Befehl angegeben:

...A26 bewirkt, daß die in 26 angegebene Adresse m zu Uo addiert wird, es entsteht der Befehl Um, der infolge des U-Bandbefehlszeichen nach Schnellspeicherzelle 19 gebracht wird.

Wenn eine andere Zelle als Schnellspeicher 26 für den Bandbefehl ...Am benutzt werden soll, kann diese auch durch Bandbefehle gefüllt werden:

CBmE m' → 26
UtE <26> → t
Zo+1E

oder auch direkt in eine Schnellspeicherzelle s

CKBs+mE m' → s
Zo+1E

Beispiel

Es soll ein Unterprogramm für das Skalarprodukt zweier n-dimensionalen Vektoren aufgestellt werden.

$$\vec{x} \cdot \vec{y} = \sum_{i=1}^n x_i y_i$$

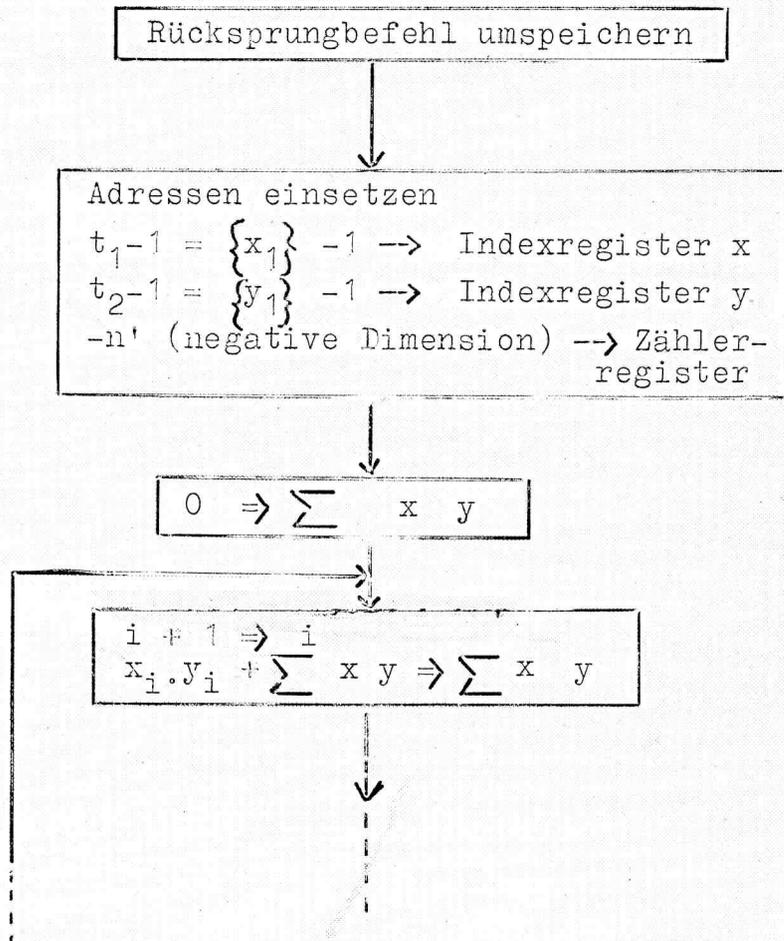
Hierbei soll gleichzeitig eine Möglichkeit der Übergabe der Parameter an ein Unterprogramm gezeigt werden. Bei den bisherigen Beispielen wurden die Parameter (Argument, Koeffizienten, Adressen, Grad usw.) in bestimmte feste Speicherzellen gebracht. Hier sollen die Parameterangaben hinter dem

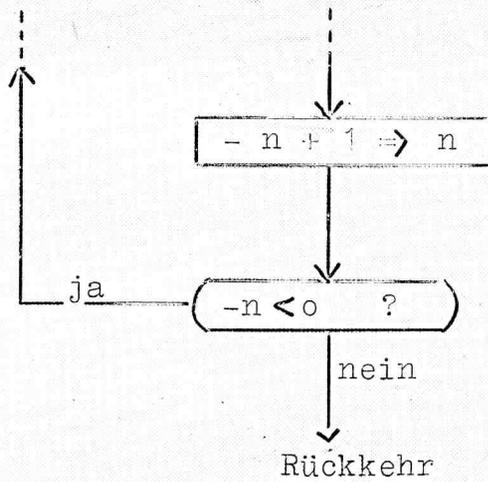
Aufrufbefehl des Unterprogramms Skalarprodukt stehen; der Rücksprung muß dann in die Zelle nach den Parameterangaben erfolgen. Das Unterprogramm holt sich die Angaben dann selbst. Es ist hier vorausgesetzt, daß der Aufrufbefehl des Unterprogramms und die Komponente der Vektoren auf der Trommel stehen.

Im Hauptprogramm würde für den Aufruf stehen:

t	Em	Aufruf des Unterprogramms
t+1	t ₁	{x ₁ } Adresse der ersten Komponente des Vektors \vec{x}
t+2	t ₂	{y ₁ } Adresse der ersten Komponente des Vektors \vec{y}
t+3	n'	Dimension der Vektoren
t+4	auf diese Stelle erfolgt der Rücksprung aus dem Unterprogramm

Flußdiagramm für das Unterprogramm Skalarprodukt





Relative Zellen-Nr.	Inhalt	Bemerkung
-	UoA26U	Bandbefehl, steht nur auf dem Streifen (verlangt Vorstreifen CBmE)
0	B5	} Rücksprungbefehl mit um 3 erhöhter Adresse umspeichern in Relativzelle 21
1	CA3	
2	U21A26	
3	GB5+0	
4	CS1	$t_1 \rightarrow a$, da in 5 die Rückadresse auf die Stelle nach dem F-Befehl steht
5	U20	$t_{1-1}' = \{x_1\} \quad -1 \rightarrow 20$
6	GB5+1	} $t_{2-1}' = \{y_1\} \quad -1 \rightarrow 21$
7	CS1	
8	U21	
9	GNS5+2	} $-n' \rightarrow 22$
10	U22	
11	CKB23+0	$0 \Rightarrow \sum x y \rightarrow 23$
12	CGB20+1	$\{x_i\} + 1 \Rightarrow \{x_i\} \rightarrow 20, x_i \rightarrow a$
13	U6	} $\{y_i\} + 1 \Rightarrow \{y_i\} \rightarrow 21, y_i \rightarrow a$
14	CGB21+1	
15	X	$x_i \cdot y_i \rightarrow a, 6$
16	B23	$\sum x y$
17	+	} $x_i y_i + \sum x y \Rightarrow \sum x y \rightarrow 23$
18	U23	
19	CKA22+1	$-n+1' \Rightarrow -n \rightarrow 22$
20	QE22+12A26	wenn $-n < 0$ Sprung auf Relativzelle 12
21	Fo	Hier wird vom Programm selbst der Rücksprungbefehl eingesetzt.

5.2 Adressenloses Programmieren (Symbolische Adressen)

Mit Hilfe eines Zusatzprogramms zum Leseprogramm ist es möglich, daß die ZUSE Z 23 dem Programmierer eine große Arbeit abnimmt, nämlich das Einsetzen der endgültigen Adressen.

Beim Aufstellen von Programmen kann man vielfach noch keine Adressen einsetzen, da sich die Adressen durch Veränderungen und Verbesserungen ständig noch verschieben. Man läßt deshalb die Adressenstellen noch offen und macht sich dabei Notizen, auf welche Stelle man sich beziehen will, z.B.

B	β
F	Sinus
U	$\sin \beta$
.	
.	
.	

Erst wenn das Programm fertig aufgestellt ist, bestimmt man für die bezeichneten Werte Adressen und muß nun das Programm nochmals durchgehen, um die Adressen einzusetzen. Falls sich nachträglich noch etwas ändert, müssen auch alle Adressen wieder überprüft werden, wobei leicht eine Veränderung übersehen werden könnte.

Die Arbeit des Adresseneinsetzens, ja auch die fortlaufende Zellenummerierung, entfällt beim adressenlosen Programmieren. Anstelle der Trommeladressen werden Namen geschrieben und die Stellen, auf die man sich bezieht, werden mit dem gleichen Namen bezeichnet. Eine Zuordnung von Namen zur endgültigen Adresse wird dann von der Maschine selbst bei der Eingabe vorgenommen. Es ist dann auch gleichgültig, ob man an irgendwelchen Stellen noch nachträglich etwas einfügt oder wegläßt, da in diesem Fall beim Einlesen in die Maschine eine andere Zuordnung Name - Adresse vorgenommen wird.

Außerdem wird der Speicherraum der Maschine besser ausgenutzt, weil die einzelnen Teile direkt hintereinander gepackt werden. Bei festen Adressen will man meist Adressen von Unterprogrammen auf volle Hunderter legen, um sich diese Adressen leichter merken zu können, wodurch möglicherweise größere Lücken zwischen den Programmteilen entstehen. Beim adressenlosen Programmieren kann das Unterprogramm mit einem Namen aufgerufen werden, der der Wirkung entspricht, so daß auf eine glatte Adresse nicht geachtet zu werden braucht.

Die Namen werden zur Eingabe in runde Klammern eingeschossen und können aus einem bis fünf beliebigen Fernschreibzeichen bestehen. Als Adresse hinter einem Operationsteil müssen sie ohne Zwischenraum angeschlossen werden; der gleiche Name kann als Adresse beliebig oft gegeben werden. Alleinstehend als Bezeichnung einer Stelle darf jeder Name nur einmal erscheinen.

Für das oben angeführte Beispiel würde man vielleicht sofort fertig hinschreiben:

B(BETA)
F(SIN)
U(SBETA)

Im Programm müßten dann unter anderem mit (BETA), (SBETA) und (SIN) = Anfang des Sinusunterprogramms bezeichnete Stellen auftreten. Es können auch Befehle von der Form B6(A) gegeben werden; hier wird die 6. Zelle nach der mit (A) bezeichneten Stelle angesprochen. Beim Einlesen führt das Zeichen "(" aus dem Leseprogramm auf das Adressierprogramm, um den Namen bis zum Schlußzeichen ")" einzulesen. Steht der Name allein zur Bezeichnung einer Stelle, so wird der Name mit der Adresse, wohin das nächste Wort gespeichert werden soll ($\langle 19 \rangle_t$), in eine "Adreßbuch" genannte Speicherliste eingetragen. Wenn der Name nach einem Operationsteil steht, so wird im "Adreßbuch" gesucht, ob dieser Name schon alleinstehend vorkam und somit die zugeordnete Adresse vorhanden ist. Ist dies der Fall, so wird diese Adresse dem zu speichernden Befehl hinzugefügt. Andernfalls wird zunächst

der unvollständig adressierte Befehl gespeichert und in ein "Vormerkbuch" (Speicherliste) wird der Name sowie die Adresse der Zelle, in der der noch unvollständige Befehl steht, notiert. Das "Vormerkbuch" wird jeweils beim Einlesen eines alleinstehenden Namens durchsucht, ob Befehle mit diesen Namen schon vorher eingelesen und als unvollständig im "Vormerkbuch" notiert wurden. Bei diesen Befehlen werden die den Namen zugeordneten Adressen eingesetzt und die entsprechenden Notierungen im "Vormerkbuch" gestrichen.

Da man, speziell beim Ausprüfen eines Programmes, wissen möchte, welche Adressen den einzelnen Namen zugeordnet wurden, kann man sich das "Adreßbuch" und "Vormerkbuch" ausdrucken lassen. Es erscheint dann jeweils ein Name und die Adresse nebeneinander.

Wenn am Bedingungsschalter mindestens eine Taste "ein" geschaltet ist, wird beim Einlesen jeder ins Adreßbuch eingetragene Name ausgeschrieben. Mit dem alleinstehenden Zeichen ")" wird anschließend das "Vormerkbuch" ausgeschrieben, d.h. nicht erledigte Vormerkungen. Es erscheinen dann Namen und Zellennummern, in denen zu den Befehlen noch nicht die Adressen des Namens eingesetzt wurden. Dies kann nur dann zutreffen, wenn zu einem Namen in Verbindung mit einem Befehl der gleiche alleinstehende Name nicht angegeben wurde oder beide Namen versehentlich verschieden geschrieben wurden.

Wenn alle Bedingungstasten "aus" geschaltet sind, so passiert beim Einlesen nichts. Mit ")" wird dann das "Adreßbuch" und das "Vormerkbuch" ausgeschrieben.

) bei Bedingungsschalter ein "Vormerkbuch" ausschreiben
" " " aus "Adreßbuch und Vormerkbuch"
" " " ausschreiben

Eine Löschung oder Teillöschung des "Adreßbuches" kann man mit dem speziellen Befehl V(NAM) erreichen.

V(NAM) lösche alle Namen im "Adreßbuch" von dem hier angegebenen Namen NAM an. Die so gelöschten Namen können mit anderer Bedeutung wieder verwendet werden.

Wird der angegebene Name im "Adreßbuch" nicht gefunden, so stoppt die Maschine und löscht nach Start das ganze "Adreßbuch" und "Vormerkbuch". Diese Gesamtlöschung macht man am besten mit

V() : Maschine stoppt. Nach Start Löschen von "Adreß- und "Vormerkbuch".

Über Eingabe des Adressierprogramms und Länge des "Bücher" vergl. Programmbeschreibung Adressierprogramm.

Beispiel

Es seien 3 Listen von Werten gegeben

$$\begin{array}{ccccccc} l_1, & l_2, & l_3 & \cdot & \cdot & \cdot & l_n \\ r_1, & r_2, & r_3 & \cdot & \cdot & \cdot & r_n \\ \mathcal{L}_1, & \mathcal{L}_2, & \mathcal{L}_3 & \cdot & \cdot & \cdot & \mathcal{L}_n \end{array}$$

Aus jeweils einem Tripel dieser Zahlen soll der Ausdruck

$$\frac{l_i}{r_i} \cdot \sin \mathcal{L}_i = p_i$$

gebildet und in eine 4. Liste eingetragen werden.

Die Größen l_i seien von einer mit (L) bezeichneten Stelle an, die r_i ab (R), die \mathcal{L}_i ab (ALF) gespeichert. Unter (N) sei die Anzahl n der Wertetripel gespeichert. Das Unterprogramm Sinus sei schon auf der Trommel gespeichert und mit dem Namen (SIN) bezeichnet.

(L) l_1
 l_2

l_3
 \cdot
 \cdot
 \cdot
 l_n
(R) r_1
 r_2
 \cdot
 \cdot
 \cdot
 r_n
(ALF) \mathcal{L}_1
 \mathcal{L}_2
 \cdot
 \cdot
 \cdot
 \mathcal{L}_n
(N) $n!$
(ANF) $CKB35+0$ $0 \Rightarrow i$
 $GB35+(ALF)$
 $F(SIN)$
 $GB35+(L)$
 X
 $GB35+(R)$
 $:$
 $GJ35+(P)$
 $CKA35+1$
 $B35$
 $S(N)$
 $QQF1(ANF)$
 Z_0
(P) $o_N(ANF)E$
 $ZE(ANF)E$

Erläuterungen erübrigen sich fast, da diese ja schon durch die Namen gegeben werden.

Dem Programm ist nur noch ein Vorstreifen vorauszuschicken

UmU

Zo+1E

der bestimmt, an welche Stelle der Trommel das Programm eingespeichert werden soll.

Befehl	Wirkung	
	1) Sprungbefehle	
Et	$\langle t \rangle \rightarrow b$	$Et+1! \rightarrow c$
Es	$\langle s \rangle \rightarrow b$	$E(s+1) \rightarrow c$
PEs+t	wenn $\langle s \rangle \geq 0$ wenn $\langle s \rangle < 0$	$\langle t \rangle \rightarrow b$ $Et+1! \rightarrow c$ nächster Befehl wird geholt
QEs+t	wenn $\langle s \rangle < 0$ wenn $\langle s \rangle \geq 0$	$\langle t \rangle \rightarrow b$ $Et+1! \rightarrow c$ nächster Befehl wird geholt
	$s \neq 3, 4, 5$	
Fm	$\langle m \rangle \rightarrow b$	$Em+1 \rightarrow c$ $\langle c \rangle \rightarrow 5$

2) Lese- und Speicherbefehle

Bm	$\langle m \rangle \rightarrow a$		
Bs+t	$\langle t \rangle \rightarrow s$	$\langle a \rangle \rightarrow a$	
Um	$\langle a \rangle \rightarrow m$	$\langle a \rangle \rightarrow a$	
Us+t	$\langle s \rangle \rightarrow t$	$\langle s \rangle \rightarrow s$	$\langle a \rangle \rightarrow a$
NUM	$\langle a \rangle \rightarrow m$	$o \rightarrow a$	
NUs+t	$\langle s \rangle \rightarrow t$	$o \rightarrow s$	$\langle a \rangle \rightarrow a$

3) Blocktransfer Trommel \leftrightarrow Schnellspeicher

Es werden n Zelleninhalte übertragen, wenn (n-1) nach Schnellspeicher 13 gebracht wird und der Befehl gegeben wird

BVs+t	$\langle t \rangle \rightarrow s$	$\langle t+1 \rangle \rightarrow s+1$	\dots	$\langle t+n-1 \rangle \rightarrow s+n-1$
UVs+t	$\langle s \rangle \rightarrow t$	$\langle s+1 \rangle \rightarrow t+1$	\dots	$\langle s+n-1 \rangle \rightarrow t+n-1$
	$\langle s \rangle \rightarrow s$	$\langle s+1 \rangle \rightarrow s+1$	\dots	$\langle s+n-1 \rangle \rightarrow s+n-1$
NUVs+t	$\langle s \rangle \rightarrow t$	$\langle s+1 \rangle \rightarrow t+1$	\dots	$\langle s+n-1 \rangle \rightarrow t+n-1$
	$o \rightarrow s$	$o \rightarrow s+1$	\dots	$o \rightarrow s+n-1$

Befehl	Wirkung	Benutzte Schnellspeicher
4) Arithmetische Befehle "Gleitendes Komma"		
+	$\langle 6 \rangle + \langle a \rangle \rightarrow a,6$	2-6, 11-13, 16, 17
-	$\langle 6 \rangle - \langle a \rangle \rightarrow a,6$	
X	$\langle 6 \rangle \cdot \langle a \rangle \rightarrow a,6$	
:	$\langle 6 \rangle : \langle a \rangle \rightarrow a,6$	
W	$\sqrt[2]{\langle a \rangle} \rightarrow a,6$	2-6, 11-13, 16-18
W2	$\sqrt[2]{1 - \langle a \rangle^2} \rightarrow a,6$	
M	$\langle a \rangle \cdot (-1) \rightarrow a,6$	4-6

5) Bedingungszeichen

PP...	} Befehl ... nur ausführen, wenn:	$\langle a \rangle \geq 0$	
QQ..		$\langle a \rangle < 0$	
PPQQ...		$\langle a \rangle = 0$	
P...		$\langle 2 \rangle \geq 0$	
Q...		$\langle 2 \rangle < 0$	
Y...		$\langle 3 \rangle_{40} = 1$	
PQQQ...		$\langle a \rangle_1 \neq \langle a \rangle_2$	

6) Ausgabe und Tabellierungsbefehle

D	$\langle a \rangle \rightarrow$ Fernschreiber	$\langle a \rangle \rightarrow a,6$	2-6, 11-13, 16-18
D2	$\langle a \rangle \rightarrow$ Fernschreiber		
	wenn $\langle 1029 \rangle \geq 0$	} $\langle a \rangle \rightarrow a,6$	
	$\langle a \rangle \rightarrow$ Locher		
	wenn $\langle 1029 \rangle < 0$		
F1000	Wagen- und Zeilentransport auf FS		2 - 6
F1002	Wagen- und Zeilentransport		
	auf FS	bei $\langle 1029 \rangle \geq 0$	
	auf Locher	bei $\langle 1029 \rangle < 0$	
	$\langle a \rangle \rightarrow a,6$		

Befehl	Wirkung	Benutzte Schnellspeicher	
F800	Spaltensprung <a> → a,6	auf FS auf FS bei <1029> ≥ 0 auf Locher bei <1029> < 0	2 - 6, 11
F802			
F840	n Zeichenbreite einrücken, wenn -n' im Akku o → a	auf FS auf FS bei <1029> ≥ 0 auf Locher bei <1029> < 0	2 - 5, 11
F842			
F850	n mal Zeilen- transport wenn -n' im Akku o → a	auf FS auf FS bei <1029> ≥ 0 auf Locher bei <1029> < 0	2 - 5, 11
F852			

7) Angaben zur Tabellierung

U1029U	
+1bzw-1	Kennzeichen ob D2 auf FS(+1) oder Locher (-1) wirken soll
l'	Stellenzahl bei Gleitkomma- Ausgabe
α'	Stellen vor } dem Komma bei " nach } Ausgabe in Normalform
β'	
k'	Spaltenzahl
z'	Zwischenräume zwischen Spalten
n'	Stellenzahl für Strichzahlen
+1bzw-1	Kennzeichen ob Gleitkomma(+1) oder Normalform (-1) Ausgabe

8) Stopbefehl

Zo	Stop vom Programm her
----	-----------------------

Befehl	Wirkung	Benutzte Schnellspeicher
	9) Bandbefehle	
UmU	Das Folgende in Zelle m, m+1, m+2 usw. speichern	
...E	Befehl ... sofort ausführen	
...Am	<m> zum Befehl ... addieren	

10) Aufruf des Leseprogramms

Eo+i	Leseprogramm mit Übertragung der Entschlüsselungstabelle in die Kernspeicher (Entschlüsselungszeit für ein Zeichen 5 msec)	2 - 99
Eo+6	Leseprogramm ohne Übertragung der Entschlüsselungstabelle in die Kernspeicher (Entschlüsselungszeit für ein Zeichen 20 msec)	2 - 32
F522	1 Wort vom Lochstreifen ablesen und in den Akkumulator bringen	2 - 32

11) Befehle für Adressenrechnungen und Zählen

Am	<a>'+<m>' → a	
As+t	<s>'+<t>' → s	<a> → a
Sm	<a>'-<m>' → a	
Ss+t	<s>'-<t>' → s	<a> → a
NSm	-<m>' → a	
NSs+t	-<t>' → s	<a> → a
USm	<m>'<a>' → a	
USs+t	<t>'<s>' → s	<a> → a
X'	<6>'<a>' → a,6	2 - 6
:'	<6>'<a>' → a,6	2 - 6
F1020	<a>' → a,6 als Gleitkommazahl	2 - 6,12,13,16

12) Konstantenzuführung

CAn	$\langle a \rangle^{!+n!} \rightarrow a$	
CBn=CNAn	$n! \rightarrow a$	
CSn	$\langle a \rangle - n! \rightarrow a$	
CNSn	$-n! \rightarrow a$	
CUSn	$n! - \langle a \rangle! \rightarrow a$	
CKAs+t	$\langle s \rangle^{!+t!} \rightarrow s$	$\langle a \rangle \rightarrow a$
CKSs+t	$\langle s \rangle^{!-t!} \rightarrow s$	$\langle a \rangle \rightarrow a$
CKSU _s +t	$t! - \langle s \rangle! \rightarrow s$	$\langle a \rangle \rightarrow a$
CKBs+t	$t! \rightarrow s$	$\langle a \rangle \rightarrow a$
CKNS _s +t	$-t! \rightarrow s$	$\langle a \rangle \rightarrow a$

13) Logische Befehle

$\exists m$	$\langle a \rangle \vee \langle m \rangle \rightarrow a$	
$\exists s+t$	$\langle s \rangle \vee \langle t \rangle \rightarrow s$	$\langle a \rangle \rightarrow a$
CK \exists _s +t	$\langle s \rangle \vee t! \rightarrow s$	$\langle a \rangle \rightarrow a$
$\exists m$	$\langle a \rangle \wedge \langle m \rangle \rightarrow a$	
ISm	$\langle a \rangle \wedge (-\langle m \rangle) \rightarrow a$	
Is+t	$\langle s \rangle \wedge \langle t \rangle \rightarrow a$!! $\langle s \rangle \rightarrow s$
ISs+t	$\langle s \rangle \wedge (-\langle t \rangle) \rightarrow a$!! $\langle s \rangle \rightarrow s$
CIn	$\langle a \rangle \wedge n! \rightarrow a$	
CISn	$\langle a \rangle \wedge (-n!) \rightarrow a$	
CKIs+t	$\langle s \rangle \wedge t! \rightarrow a$!! $\langle s \rangle \rightarrow s$
CKIS _s +t	$\langle s \rangle \wedge (-t!) \rightarrow a$!! $\langle s \rangle \rightarrow s$

14) Adressensubstitution

G...t	1. Schritt $\langle t \rangle_t \Rightarrow \tilde{t}$ G löschen 2. Schritt ... \tilde{t} ausführen
G...s+t	1. Schritt $(\langle s \rangle + t)_t \Rightarrow \tilde{t}$ $\langle s \rangle \rightarrow s$ G, s löschen 2. Schritt ... \tilde{t} ausführen

Befehl	Wirkung	
CG...s+t	1.Schritt	$(\langle s \rangle + t)_t \Rightarrow \tilde{t} \quad \langle s \rangle + t \rightarrow s$ CGs löschen
	2.Schritt	... \tilde{t} ausführen
GK...s+t	1.Schritt	$(\langle s \rangle + t)_{33...40} \Rightarrow \tilde{s} \quad \langle s \rangle \rightarrow s$ G, s löschen
	2.Schritt	... $K\tilde{s}$ ausführen
CGK...s+t	1.Schritt	$(\langle s \rangle + t)_{33...40} \Rightarrow \tilde{s} \quad \langle s \rangle + t \rightarrow s$ CGs löschen
	2.Schritt	... $K\tilde{s}$ ausführen

Liste der Schnellspeicher		
Adresse	Wirkung beim Lesen	Wirkung beim Schreiben
0	liefert 0	Mauldrucker
1	liefert 2^{39} Vorzeichenstelle	Creed-Locher
2	Test P, Q bei beliebigen Befehlen	
3	Verlängerung des Akkus, Test Y auf letzte Stelle	
4	Akkumulator Test PP, QQ, PPQQ, PQQQ	Übernahme der Handeinstellung
5	Rückkehradressenspeicher	
6	Ohne Besonderheit	
7	} für Sonderzwecke vorgesehen	
8		
9		
10		
11	Ohne Besonderheit	
12 \equiv 5		
13	Zähler für Serienschiebung und Blocktransfer	
14	Bedingungsschalter	
15	Ferranti-Abtaster	Fernschreiber
16-	} allgemein ohne Besonderheit	
⋮		
⋮		
255		

Für Trommelsprungbefehle sind testbar (PEs+t, QEs+t)
2, 6, 11, 16 bis 255.

Erläuterungen zu den Ein- und Ausgabepuffern

Zu Ferranti A15: Das abgetastete Fernschreibzeichen kommt in die untersten 5 Stellen (Stellen 36 - 40) (maximal 300 Zeichen/sec).

Zu Fernschreiber U15 bzw. schneller Locher U1 (Zusatz): Die Stellen 2 bis 6 gehen als Fernschreibzeichen auf den Fernschreiber bzw. Locher. (Puffer mit 1 Zeichen, Fernschreiber 10 Zeichen/sec., Locher 50 Zeichen/sec).

Zu Mauldrucker (Zusatzgerät) Uo: Es gehen die Stellen 35 bis 40 (6 Kanal) auf den Drucker (Puffer mit 80 Zeichen = 1 Zeile, der beliebig schnell gefüllt und einmal pro sec geleert werden kann).

Aufteilung eines Befehlswortes

Numerierung der Stellen

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
Start	Stop	PP	P	QQ	Q	Y	C	N	LL	R	U	A	S	F	K	H	G	V

20 . . . 27	28 . . . 40
Schnell-Speicher	Trommel-Speicher
Adresse	

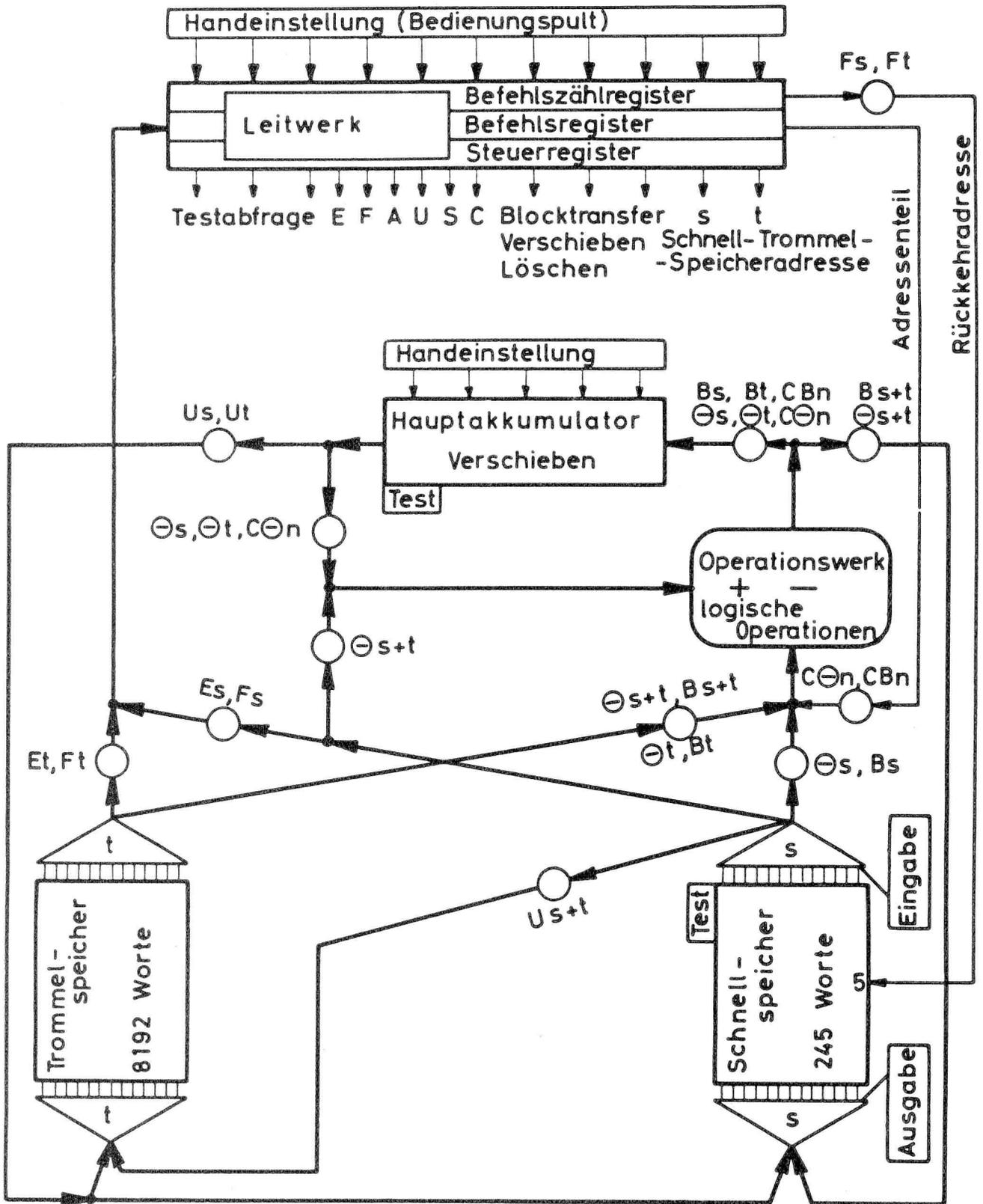
OKTAL - TABELLE

0	0000	800'	0144o	1600'	0310o	2400'	0454o	3200'	0620o
16'	0002o	816'	0146o	1616'	0312o	2416'	0456o	3216'	0622o
32'	0004o	832'	0150o	1632'	0314o	2432'	0460o	3232'	0624o
48'	0006o	848'	0152o	1648'	0316o	2448'	0462o	3248'	0626o
64'	0010o	864'	0154o	1664'	0320o	2464'	0464o	3264'	0630o
80'	0012o	880'	0156o	1680'	0322o	2480'	0466o	3280'	0632o
96'	0014o	896'	0160o	1696'	0324o	2496'	0470o	3296'	0634o
112'	0016o	912'	0162o	1712'	0326o	2512'	0472o	3312'	0636o
128'	0020o	928'	0164o	1728'	0330o	2528'	0474o	3328'	0640o
144'	0022o	944'	0166o	1744'	0332o	2544'	0476o	3344'	0642o
160'	0024o	960'	0170o	1760'	0334o	2560'	0500o	3360'	0644o
176'	0026o	976'	0172o	1776'	0336o	2576'	0502o	3376'	0646o
192'	0030o	992'	0174o	1792'	0340o	2592'	0504o	3392'	0650o
208'	0032o	1008'	0176o	1808'	0342o	2608'	0506o	3408'	0652o
224'	0034o	1024'	0200o	1824'	0344o	2624'	0510o	3424'	0654o
240'	0036o	1040'	0202o	1840'	0346o	2640'	0512o	3440'	0656o
256'	0040o	1056'	0204o	1856'	0350o	2656'	0514o	3456'	0660o
272'	0042o	1072'	0206o	1872'	0352o	2672'	0516o	3472'	0662o
288'	0044o	1088'	0210o	1888'	0354o	2688'	0520o	3488'	0664o
304'	0046o	1104'	0212o	1904'	0356o	2704'	0522o	3504'	0666o
320'	0050o	1120'	0214o	1920'	0360o	2720'	0524o	3520'	0670o
336'	0052o	1136'	0216o	1936'	0362o	2736'	0526o	3536'	0672o
352'	0054o	1152'	0220o	1952'	0364o	2752'	0530o	3552'	0674o
368'	0056o	1168'	0222o	1968'	0366o	2768'	0532o	3568'	0676o
384'	0060o	1184'	0224o	1984'	0370o	2784'	0534o	3584'	0700o
400'	0062o	1200'	0226o	2000'	0372o	2800'	0536o	3600'	0702o
416'	0064o	1216'	0230o	2016'	0374o	2816'	0540o	3616'	0704o
432'	0066o	1232'	0232o	2032'	0376o	2832'	0542o	3632'	0706o
448'	0070o	1248'	0234o	2048'	0400o	2848'	0544o	3648'	0710o
464'	0072o	1264'	0236o	2064'	0402o	2864'	0546o	3664'	0712o
480'	0074o	1280'	0240o	2080'	0404o	2880'	0550o	3680'	0714o
496'	0076o	1296'	0242o	2096'	0406o	2896'	0552o	3696'	0716o
512'	0100o	1312'	0244o	2112'	0410o	2912'	0554o	3712'	0720o
528'	0102o	1328'	0246o	2128'	0412o	2928'	0556o	3728'	0722o
544'	0104o	1344'	0250o	2144'	0414o	2944'	0560o	3744'	0724o
560'	0106o	1360'	0252o	2160'	0416o	2960'	0562o	3760'	0726o
576'	0110o	1376'	0254o	2176'	0420o	2976'	0564o	3776'	0730o
592'	0112o	1392'	0256o	2192'	0422o	2992'	0566o	3792'	0732o
608'	0114o	1408'	0260o	2208'	0424o	3008'	0570o	3808'	0734o
624'	0116o	1424'	0262o	2224'	0426o	3024'	0572o	3824'	0736o
640'	0120o	1440'	0264o	2240'	0430o	3040'	0574o	3840'	0740o
656'	0122o	1456'	0266o	2256'	0432o	3056'	0576o	3856'	0742o
672'	0124o	1472'	0270o	2272'	0434o	3072'	0600o	3872'	0744o
688'	0126o	1488'	0272o	2288'	0436o	3088'	0602o	3888'	0746o
704'	0130o	1504'	0274o	2304'	0440o	3104'	0604o	3904'	0750o
720'	0132o	1520'	0276o	2320'	0442o	3120'	0606o	3920'	0752o
736'	0134o	1536'	0300o	2336'	0444o	3136'	0610o	3936'	0754o
752'	0136o	1552'	0302o	2352'	0446o	3152'	0612o	3952'	0756o
768'	0140o	1568'	0304o	2368'	0450o	3168'	0614o	3968'	0760o
784'	0142o	1584'	0306o	2384'	0452o	3184'	0616o	3984'	0762o

4000'	0764o	4800'	11300	5600'	1274o	6400'	14400	7200'	1604o
4016'	0766o	4816'	1132o	5616'	1276o	6416'	1442o	7216'	1606o
4032'	0770o	4832'	1134o	5632'	1300o	6432'	1444o	7232'	1610o
4048'	0772o	4848'	1136o	5648'	1302o	6448'	1446o	7248'	1612o
4064'	0774o	4864'	1140o	5664'	1304o	6464'	1450o	7264'	1614o
4080'	0776o	4880'	1142o	5680'	1306o	6480'	1452o	7280'	1616o
4096'	1000o	4896'	1144o	5696'	1310o	6496'	1454o	7296'	1620o
4112'	1002o	4912'	1146o	5712'	1312o	6512'	1456o	7312'	1622o
4128'	1004o	4928'	1150o	5728'	1314o	6528'	1460o	7328'	1624o
4144'	1006o	4944'	1152o	5744'	1316o	6544'	1462o	7344'	1626o
4160'	1010o	4960'	1154o	5760'	1320o	6560'	1464o	7360'	1630o
4176'	1012o	4976'	1156o	5776'	1322o	6576'	1466o	7376'	1632o
4192'	1014o	4992'	1160o	5792'	1324o	6592'	1470o	7392'	1634o
4208'	1016o	5008'	1162o	5808'	1326o	6608'	1472o	7408'	1636o
4224'	1020o	5024'	1164o	5824'	1330o	6624'	1474o	7424'	1640o
4240'	1022o	5040'	1166o	5840'	1332o	6640'	1476o	7440'	1642o
4256'	1024o	5056'	1170o	5856'	1334o	6656'	1500o	7456'	1644o
4272'	1026o	5072'	1172o	5872'	1336o	6672'	1502o	7472'	1646o
4288'	1030o	5088'	1174o	5888'	1340o	6688'	1504o	7488'	1650o
4304'	1032o	5104'	1176o	5904'	1342o	6704'	1506o	7504'	1652o
4320'	1034o	5120'	1200o	5920'	1344o	6720'	1510o	7520'	1654o
4336'	1036o	5136'	1202o	5936'	1346o	6736'	1512o	7536'	1656o
4352'	1040o	5152'	1204o	5952'	1350o	6752'	1514o	7552'	1660o
4368'	1042o	5168'	1206o	5968'	1352o	6768'	1516o	7568'	1662o
4384'	1044o	5184'	1210o	5984'	1354o	6784'	1520o	7584'	1664o
4400'	1046o	5200'	1212o	6000'	1356o	6800'	1522o	7600'	1666o
4416'	1050o	5216'	1214o	6016'	1360o	6816'	1524o	7616'	1670o
4432'	1052o	5232'	1216o	6032'	1362o	6832'	1526o	7632'	1672o
4448'	1054o	5248'	1220o	6048'	1364o	6848'	1530o	7648'	1674o
4464'	1056o	5264'	1222o	6064'	1366o	6864'	1532o	7664'	1676o
4480'	1060o	5280'	1224o	6080'	1370o	6880'	1534o	7680'	1700o
4496'	1062o	5296'	1226o	6096'	1372o	6896'	1536o	7696'	1702o
4512'	1064o	5312'	1230o	6112'	1374o	6912'	1540o	7712'	1704o
4528'	1066o	5328'	1232o	6128'	1376o	6928'	1542o	7728'	1706o
4544'	1070o	5344'	1234o	6144'	1400o	6944'	1544o	7744'	1710o
4560'	1072o	5360'	1236o	6160'	1402o	6960'	1546o	7760'	1712o
4576'	1074o	5376'	1240o	6176'	1404o	6976'	1550o	7776'	1714o
4592'	1076o	5392'	1242o	6192'	1406o	6992'	1552o	7792'	1716o
4608'	1100o	5408'	1244o	6208'	1410o	7008'	1554o	7808'	1720o
4624'	1102o	5424'	1246o	6224'	1412o	7024'	1556o	7824'	1722o
4640'	1104o	5440'	1250o	6240'	1414o	7040'	1560o	7840'	1724o
4656'	1106o	5456'	1252o	6256'	1416o	7056'	1562o	7856'	1726o
4672'	1110o	5472'	1254o	6272'	1420o	7072'	1564o	7872'	1730o
4688'	1112o	5488'	1256o	6288'	1422o	7088'	1566o	7888'	1732o
4704'	1114o	5504'	1260o	6304'	1424o	7104'	1570o	7904'	1734o
4720'	1116o	5520'	1262o	6320'	1426o	7120'	1572o	7920'	1736o
4736'	1120o	5536'	1264o	6336'	1430o	7136'	1574o	7936'	1740o
4752'	1122o	5552'	1266o	6352'	1432o	7152'	1576o	7952'	1742o
4768'	1124o	5568'	1270o	6368'	1434o	7168'	1600o	7968'	1744o
4784'	1126o	5584'	1272o	6384'	1436o	7184'	1602o	7984'	1746o
8000'	1750o	8016'	1752o	8032'	1754o	8048'	1756o	8064'	1760o
8080'	1762o	8096'	1764o	8112'	1766o	8128'	1770o	8144'	1772o
8160'	1774o	8176'	1776o						

Blockschema ZUSE Z 23

(vereinfacht auf Externcode ohne Adressensubstitution, Logische Operationen u.ä.)



Zeichenerklärung:

- $\bullet \text{---} \bigcirc \text{---}$ Das Tor ist bei einem s Schnellspeicheradresse
- $\ominus s$ Befehl der Art $\ominus s$ geöffnet t Trommelspeicheradresse
- \ominus Abkürzung für A, S oder US n Gesamter Adressenteil als Konstante