

ZUSE Z4 OPERATING MANUAL

Transcribed and translated by  
Fredrik Andersson  
nablaman (a) gmail.com

Lund, Sweden  
Summer 2022

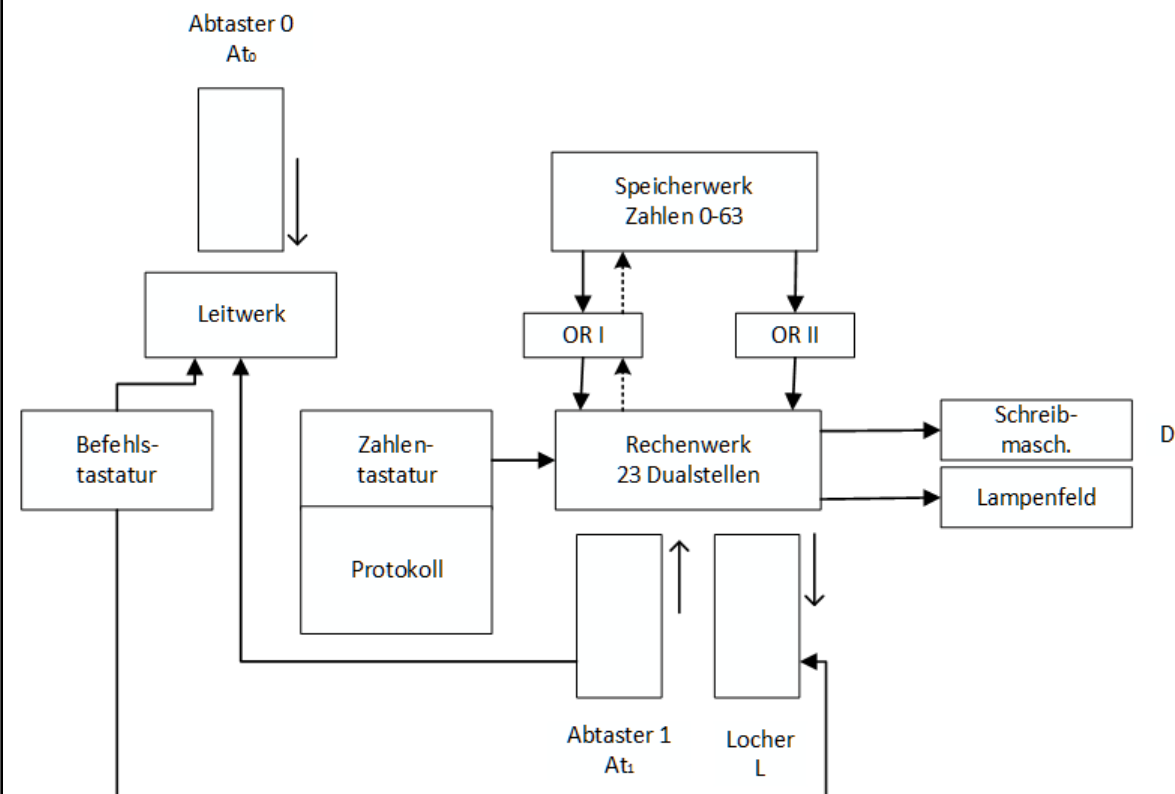
GEBRAUCHSANWEISUNG Z 4

Sommersemester 1952

OPERATING MANUAL Z 4

Summer term 1952

## 1. Kapitel: Organisation der Maschine



### §1. Rechenwerk

Halblogarithmisch dual;  $x = \pm 2^a b$   
wobei  $-64 \leq a \leq +63$ ; a hat 7 Dualstellen  
 $1 \leq b \leq 2$ ; b hat 24 Dualstellen  
 $b = L, 0LL\dots 0$ ; erste Stelle immer L.  
----- 24

(Daher im Speicher nur 23 Stellen nötig)

Vorzeichen = 1 weitere Dualstelle ( $=L \text{ --}0$ ). Zahlen wie 0,  $\infty$ , ? besitzen keine halblogarithmische Darstellung mit  $|b| \geq 1$ . Deshalb als Sonderwerte betrachtet. 2 Operandenregister OR I und OR II: dienen zur Bereitstellung der Operanden, bevor die Operation beginnt.

### §2. Speicherwerk

64 Zellen; Adressen 0-63. Jede Zelle enthält eine Zahl mit 32 Dualstellen, nämlich 7 Stellen für a, 23 für b, 1 Vorzeichen von b und 1 Sonderzeichen, wobei 0 normale Zahl und L Sonderwert bedeutet. Ablesen aus dem Speicher löscht der Speicher nicht. Speichern einer neuen Zahl löscht die bisherige Zahl in der betreffenden Zelle.

### §3. Gangarten der Maschine

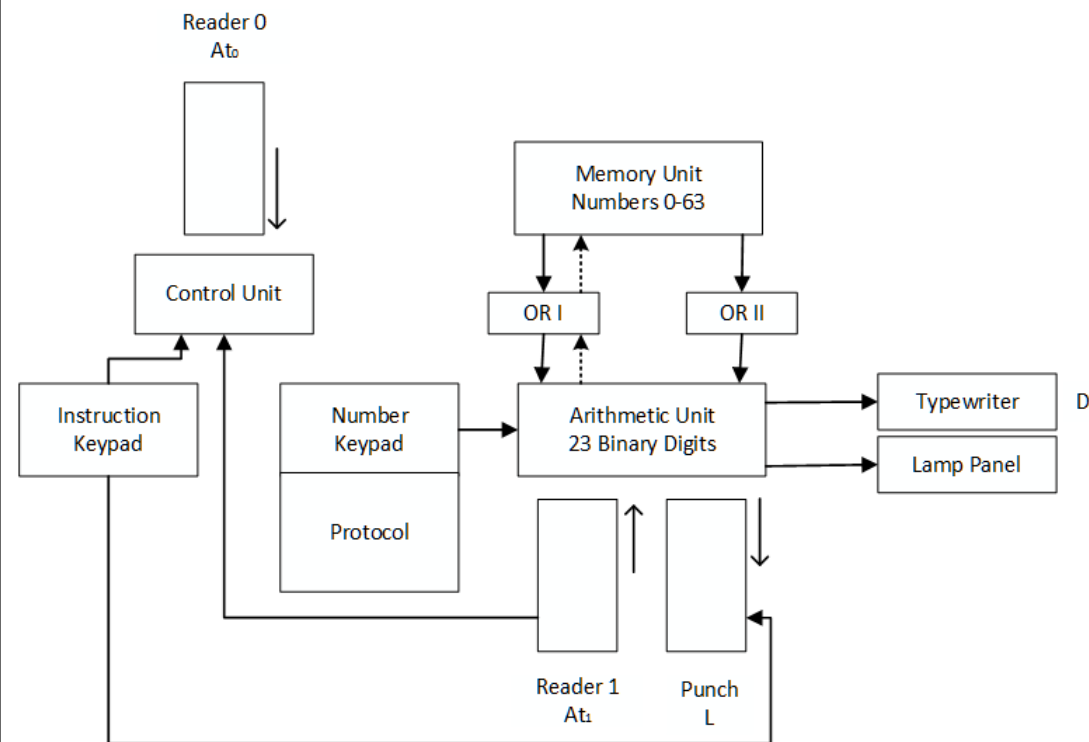
Die Maschine kann auf 4 verschiedene Arten benutzt werden.

**Gangart 1:** Normale Gangart zum Rechnen:  $At_0$  hat nur Befehle abzulesen.  $At_1$  kann nach Wahl Befehle oder Zahlen ablesen. Im ersteren Fall heisst der Streifen in  $At_0$  der "Hauptplan" und derjenige in  $At_1$  der "Underplan". Außerdem können Zahlen auf Film gelocht werden. Wenn in  $At_0$  kein Film eingelegt, führt die Maschine die auf der Befehlstastatur gegebenen Befehle sofort aus. (Fehlersuche, Kontrolle).

**Gangart 2 identisch mit Gangart 3:** Normale Gangart zum Rechenplananfertigen. Die auf der Befehlstastatur eingetasteten Befehle werden im Locher gelocht (aber nicht ausgeführt). Unabhängig von dieser Rechenplanfertigung kann die Maschine gleichzeitig programmgesteuert rechnen, falls dafür nur  $At_0$  benutzt wird und keine Zahlen gelocht werden.

**Gangart 4:** Fertigen eines Zahlenstreifens. Jede auf der Zahlen-

## Chapter 1: Organization of the Machine



### §1. Computation Unit

Semilogarithmic binary;  $x = \pm 2^a b$   
wherein  $-64 \leq a \leq +63$ ; a has 7 binary digits  
 $1 \leq b \leq 2$ ; b has 24 binary digits  
 $b = L, 0LL\dots 0$ ; first position always L.  
----- 24

(Hence only 23 digits are required in the memory)

Signs = 1 additional binary digit ( $=L \text{ --}0$ ). Numbers with 0,  $\infty$ , ? have no semilogarithmic representation with  $|b| \geq 1$ . Therefore regarded as special values. 2 operand registers, OR 1 and OR 2, serve as storage for operands before the instruction begins.

### §2. Memory Unit

64 cells, addressed as 0-63. Each cell contains one number with 32 binary digits, with 7 digits for a, 23 for b, 1 for the sign of b and 1 flag, where 0 means a normal number and L a special value. Reading from memory does not erase it. Storing a new value erases the previous value in the affected cell.

### §3. The Machine's Operating Modes

The machine can be used in 4 different operating modes.

**Mode 1:** Normal operating mode for computation.  $At_0$  will only read instructions.  $At_1$  can be configured to read either instructions or numbers. In the first case the punched tape in  $At_0$  is referred to as the "main program", and the one in  $At_1$  the "sub program". Also, numbers may be punched onto film. When no film is placed in  $At_0$ , the machine will immediately execute any instructions given via the instruction keyboard. (Fault finding, Checking)

**Mode 2, identical to mode 3:** Normal mode during programming. The instructions provided via the instruction keyboard will be punched by the punch (but not executed). The machine can execute programs simultaneously while being used for programming, provided only  $At_0$  is used and no numbers are to be punched.

**Mode 4:** Punching a number tape. All numbers that are entered on the Number Keyboard are translated to binary and then punched as binary numbers. No other operations can be executed simultaneously.

tastatur eingegebene Zahl wird ins Dualsystem übersetzt und als Dualzahl gelocht. Andere Operationen können nicht gleichzeitig durchgeführt werden.

Gangart 5: Ablesen (gewöhnlich zu Kontrollzwecken) eines Zahlenstreifens. Der Streifen muss in  $A_1$  eingelegt werden und die Zahlen erscheinen im Lampenfeld.

Außerdem besteht die spezielle Gangart 9, um in eine Befehlsreihe eine Zahl einzufügen, mit der gerechnet werden soll. ("Konstante auf Rechenplan").

#### §4. Befehlsabtastung und Leitwerk

Ein Befehl = Gruppe von Löchern auf der Film, die auf folgende 8 möglichen Stellen verteilt sind:

o o o o  
o o o o

wobei zwischen diesen beiden Zeilen Löcher vorhanden sind, die teils zum vorhergehenden Befehl gehören.

Einlegen des Films: Bild aufrecht, Tonstreifen links. (Schicht gegen den Beschauer). Abtastung der Befehle durch 8 Nadeln im Abtaster, die einen Kontakt schliessen, wenn sich an der betreffenden Stelle ein Loch befindet. Die entstehenden Impulse werden dem Leitwerk zugeführt, welches Befehle entschlüsselt und deren Ausführung veranlasst.

#### §5. Zahlenabtaster und Locher

Zur Entlastung des Speicherwerks können Zahlen auf Film gelocht werden. Eine Zahl (32 Dualstellen wie im Speicherwerk) hat auf dem Film dieselbe Anordnung wie 4 Befehle. Befindet sich noch keine Zahl auf dem Streifen, so locht die Maschine unmittelbar vor der ersten Zahl ein "Zahlenstartzeichen". Dieses dient auch zur Kennzeichnung des Streifens als Zahlenstreifen im Gegensatz zu einem Befehlsstreifen. Zahlenstreifen können nur in  $A_1$  abgetastet werden unter Gangart 1 oder 5. Zum Locher: Gangart 1 oder 4.

#### §6. Tastatur, Lampenfeld, Druckwerk, Protokoll

Die eingetastete Zahl erscheint zur Kontrolle im Lampenfeld. Wenn diese Kontrolle nicht stimmt: Taste "Irrtum" drücken und korrigieren. Stimmt die Kontrolle: Taste "Fertig" drücken; dies bewirkt Überführung der Zahl ins Rechenwerk und verunmöglicht jede Korrektur. Man kann genau 6 wesentliche Dezimalen eintasten, wobei man mit dem Komma beginnen (,370501) aber nicht aufhören darf. Eine ganze Zahl ist also ohne nachfolgendes Komma einzugeben. Dauert das Eintasten länger als 25 sc, so schaltet der Impulsgeber aus. Wiedereingangssetzung: Alle Filme abnehmen, Knopf "Start" betätigen und Rechnung von vorne beginnen. Die Aufforderung an die Bedienungsperson zum Eintasten (beim Rechnen mit eingelegtem Rechenplan) ist ein rotes Blinksignal, oder das Aufleuchten einer Protokoll-Lampe.

Errechnete Resultate können durch einen Befehl während des programmgesteuerten Rechnens (oder Gangart 5) ins Lampenfeld gegeben werden. Nach Kontrolle der erscheinenden muss durch Drücken der Taste "Lampenfeldlöschung" das Lampenfeld wieder freigegeben werden. Beim Ablesen des Lampenfelds ist auf die Faktoren  $10^{+6}$ ,  $10^{+12}$ ,  $10^{+18}$  zu achten. Beim Eintasten können diese Zehnerpotenzen auch eingegeben werden.

Das Protokollfeld dient zur Erleichterung des Eingebens vieler Zahlen in bestimmter Anordnung (Matrix). Es ist jeweils die Zahl des Protokolls einzugeben, unter der das Licht aufleuchtet. Dazu müssen die Zahlen auf einem Protokollformular notiert werden, das auf die Mattscheibe aufgelegt wird. Ein Protokollformular kann natürlich nur zusammen mit einem bestimmten Rechenplan verwendet werden.

Das Druckwerk besteht aus einer elektromagnetisch gesteuerten Schreibmaschine. Rechenresultate können entsprechend folgender 5 Schemata gedruckt werden. Jede Zahl benötigt der Platz von 12 Anschlägen der Schreibmaschine.

Mode 5: Reading (normally for verification purposes) a number tape. The tape must be positioned in  $A_1$  and the numbers will be displayed on the Lamp Panel.

In addition the special Mode 9 is used to insert a number needed for a computation into a series of instructions ("Constants in the Program")

#### §4. Command Entry and Control Unit

An instruction = A group of punched holes on the film, arranged in the following 8 possible locations:

o o o o  
o o o o

with possible holes belonging to the previous instruction in between these two rows.

Loading the films: Picture upright, soundstrip to the left. (Coating towards the operator). The instructions are scanned using 8 needles in the reader, which close a circuit when they find themselves at a hole. The resulting impulse is fed to the control unit, where the instruction is decoded and its execution induced.

#### §5. Number Reader and Punch

For offloading memory, numbers may be punched on film. One number (32 binary digits, like in memory) has the same arrangement on Film as 4 instructions. If no number is yet present on the tape, the machine will punch a "number start marker" immediately before the first number. These also serve to identify the tape as a number tape rather than an instruction tape. Number tapes can only be read in  $A_1$  in operation mode 1 or 5. To punch: Mode 1 or 4.

#### §6. Keyboard, Lamp Panel (Display), Printer, Log (Protocol)

The keyed-in number will be displayed on the display for verification. When the check fails: Press the "Error" key and correct number. If correct: Press the "Finished" key; This initiates transfer of the number to the computation unit and makes further corrections impossible. 6 significant decimals can be entered, starting (but not ending) with a comma (,370501). An integer is thus entered without a trailing comma. If the entry takes longer than 25 seconds, the impulse generator will switch off. Recovery: Remove all films, use the "Start" button and restart computation from the beginning. The alert signal to the operator who is entering (while computing with an entered program) will be a red blinking lamp, alternatively a log lamp.

Computed results can, through an instruction, be displayed on the display during programmed computation (or in Mode 5). After checking the displayed value, the display must be released again by pressing the key "Release Display". When reading the display, make note of the factors  $10^{+6}$ ,  $10^{+12}$ ,  $10^{+18}$ . These powers of ten can also be entered through the keyboard.

The log panel serves as an aid in entering many numbers in particular order (Matrix). Input should be given in accordance with which light is on in the log panel. In addition the numbers must be written down on a log form that is kept on the screen. A form can of course only be used together with one particular program.

The printer consists of an electromagnetically controlled typewriter. Results of computations can be printed using the following 5 schemas. Every number occupies the equivalent space of 12 typewriter strokes.

- 1) Druckschema D  
Drucken mit beweglichem Komma in der Form  $x = u \cdot 10^{6v}$  ( $-3 \leq v \leq +3$ ),  
u wird mit 6 wesentlichen Dezimalen normal gedruckt, während  
v verschlüsselt ist, nämlich:

/v für v = 1,2,3 keine Angabe für v = 0  
und /3-v für v = -1,-2,-3

Beispiel: 1,00 /2 =  $1 \cdot 10^{12}$   
1,00 /5 =  $1 \cdot 10^{-12}$

- 2) Druckschema D<sub>0</sub>  
Drucken des Resultats als ganze Zahl (Unterdrückung aller Stellen  
nach dem Komma).
- 3) Druckschema D<sub>1</sub>  
Drucken mit festem Komma: 2 Stellen nach dem Komma. In einer  
Kolonne gedruckter Zahlen stehen die Komma übereinander.
- 4) Druckschema D<sub>2</sub> Analog D<sub>1</sub> mit 4 Stellen nach Komma.
- 5) Druckschema D<sub>3</sub> Analog D<sub>1</sub> mit 6 Stellen nach Komma.

Wenn eine zu druckende Zahl bei Anwendung von D<sub>0</sub>, D<sub>1</sub>, D<sub>2</sub>, D<sub>3</sub>  
den durch das Schema gegebenen Platz überschreitet,  
schaltet die Maschine automatisch auf Schema D. Dies bewirkt, dass  
niemals Stellen vorne abgeschnitten werden.

Die Wagenbreite der Schreibmaschine fasst normal 6 Zahlen.  
Zwischen den einzelnen Zahlen in einer Zeile werden automatisch  
Zwischenräume freigelassen. Besonders grosse Zwischenräume, Ein-  
rücken einer Zeile, Wagenrücklauf mit Zeilenschaltung müssen beson-  
ders befohlen werden.

#### §7. Die Befehlstastatur

Die Befehlstastatur dient zur Rechenplanfertigung (Gangart 2)  
oder zum Steuern der einzelnen Rechenoperationen von Hand (Gangart 1  
jedoch nur, wenn kein Streifen in At<sub>0</sub>). Sie enthält die Knöpfe für  
alle Befehle und Knöpfe für die Adressen des Speicherwerks. (Für  
jeden Ables- und Speicherbefehl müssen daher 2 Knöpfe bedient wer-  
den). Außerdem sind noch Knöpfe vorhanden, die nur bei der Rechen-  
planfertigung benützt werden und keine Befehle im Sinne eines  
Rechenprogramms bedeuten, also auch nicht gelocht werden können  
(vgl. Anweisung zur Rechenplanfertigung)

#### §8. Ingangsetzen und Abstellen der Maschine

a) Normales Einschalten (Auszuführen, wenn die 4 Schalter am  
Umformerschaltbrett, obere Reihe, auf 0 stellen).

1. Die 4 genannten Schalter der Reihe nach von links nach rechts  
auf 1 stellen. Spannungsregler auf 4,5 stellen.
2. Am Pult der Z 4 den Schlüsselschalter einschalten.
3. Damit ist die Maschine unter Spannung, aber der Impulsgeber  
muss noch durch Betätigen des Knopfes "Impulsgeber start"  
in Gang gesetzt werden. Wenn eine der Lampen "Rechenplan im  
Locher" oder "Zahlenstreifen im Locher" aufleuchtet, ist der  
Knopf "Auslauf" der Befehlstastatur zu drücken.
4. Gewünschte Gangart einschalten  
Bei Gangart 1: Filme einlegen (zuerst in At<sub>1</sub>).  
Der Film kann irgendwo im leeren Raum vor dem Startzeichen  
eingelegt werden und läuft dann automatisch bis zu diesem.  
Die Art des in At<sub>1</sub> eingelegten Streifens (Befehle oder Zahlen)  
sollte durch Aufleuchten der betreffenden Lampe angezeigt  
werden. Erscheint die nicht, so nocheinmal einlegen.

- 1) Printer schema D  
Prints with a floating comma in the form  $x = u \cdot 10^{6v}$   
( $-3 \leq v \leq +3$ ), u will normally be printed with 6 significant  
digits, while v is encoded as:

/v for v = 1,2,3 undefined for v = 0  
and /3-v for v = -1,-2,-3

Example: 1,00 /2 =  $1 \cdot 10^{12}$   
1,00 /5 =  $1 \cdot 10^{-12}$

- 2) Printer schema D<sub>0</sub>  
Prints results using integers (suppresses all digits after the  
comma)
- 3) Printer schema D<sub>1</sub>  
Prints using fixed point: 2 digits after the comma. In a column  
of printed numbers, the commas will line up.
- 4) Printer schema D<sub>2</sub>  
Like D<sub>1</sub> with 4 digits after the comma.
- 5) Printer schema D<sub>3</sub>  
Like D<sub>1</sub> with 6 digits after the comma.

When a number to be printed using D<sub>0</sub>, D<sub>1</sub>, D<sub>2</sub>, D<sub>3</sub> exceeds the  
allotted schema, the machine switches automatically to schema D. This  
ensures that no digits will be cut off from the front.

The carriage width of the typewriter normally holds 6 numbers.  
Blank space will be inserted between the individual numbers on a  
line. Extra wide blank space, indenting a line, and carriage returns  
with line feeds must be specifically ordered.

#### §7. The Command Keyboard

The command keyboard serves in programming (Operating Mode 2) or for  
controlling individual manual computing operations (Operation Mode 1,  
but only when no tape is in At<sub>0</sub>). It contains the buttons for every  
instruction and buttons for memory addresses. (For all the read and  
memory instructions two buttons must be employed). In addition there  
are buttons available that are only used in programming and have no  
meaning as program instructions, and as such they also cannot be  
punched. (see Programming Manual)

#### §8. Starting and Stopping the Machine

a) Normal startup (To be performed when the 4 switches on the  
converter switchboard, upper row, are set to 0).

1. Set the 4 mentioned switches to 1, in sequence from left to  
right. Set voltage regulator to 4,5.
2. Engage the key switch on the Z 4's console.
3. With this, the machine is energized, but the impulse generator  
must also be started by using the button "Start Impulse  
Generator". If either of the lamps "Program in Punch" or  
"Number Tape in Punch" lights up, the button "Output" on the  
command keypad is to be pressed.
4. Setting desired Operation Mode  
For Mode 1: Load films (first in At<sub>1</sub>).  
The film can be positioned anywhere before the start marker  
and will automatically run up to this. The type of tape in At<sub>1</sub>  
(instructions or numbers) should be indicated by the  
respective lamp lighting up. If neither lights up, reload  
once more.

b) Die Z 4 schaltet den Umformer automatisch aus:

Wenn "Fin - Stop"-Schalter auf "ein" steht und 30 sc lang nicht mehr gerechnet wurde (z.B. nach Beendigung der Rechnung). Beim Eintreten in den Maschinenraum erkennt man die Tatsache, dass die Z 4 automatisch abgeschaltet hat, daran, dass die 4 Umformerschalter noch auf 1 stehen. Das Wiedereinschalten besteht in diesem Falle aus:

1. Alle Filme aus den Abtastern nehmen. Ebenso Papier aus der Schreibmaschine. Operationszähler notieren.
2. Schlüsselschalter aus-schalten. Erst nachher "Fin - Stop"-Schalter ausschalten. Dies bewirkt Anlaufen des Umformers. Dann ab Nr. 2 under a), wie beim normalen Einschalten.

c) Abstellen der Z 4. Filme abnehmen, Operationszählerstand notieren, Schlüsselschalter aus, Schreibmaschine ausschalten, die 4 Umformerschalter auf 0. Dadurch wird das Rechenwerk gelöscht, aber nicht das Speicherwerk.

d) Automatisches Ausschalten am Ende der Rechnung wird durch "Fin - Stop"-Schalter auf "ein" bewirkt. Dies ist dann durchzuführen, wenn der Maschinenraum auf längere Zeit verlassen wird.

**2. Kapitel: Die Befehle und ihre Wirkungen**

§1. Start und Fin

"St" Jeder Befehlstreifen muss mit einem Startzeichen "St" beginnen.

"Fin" Der Schluss ist durch das Zeichen "Fin" zu markieren (siehe auch bedingte Befehle). Dieses Zeichen beendet das Rechnen, löscht OR I und OR II, stellt den Filmtransport ab, lässt die rote Lampe "Schluss" aufleuchten und bewirkt endlich 25 sc später das Abschalten des Impulsgebers. (Bei eingeschaltetem "Fin - Stop"-Schalter auch Abschalten des Umformers). Für Beginn einer neuen Rechnung muss "Impulsgeber start" gedrückt werden. Zyklische Rechenpläne, die als endloses Band zusammen geklebt sind, dürfen natürlich kein Schlusszeichen enthalten, da sie ja mehrere Male durchlaufen werden sollen. (Ausnahme siehe "Up").

§2. Der Ablesebefehl

An (A = Befehl "Ablese aus Speicher", n = Adresse läuft von 0-63). Bewirkt Übertragung der Zahl in Zelle Nr. n nach OR I. Wenn jedoch OR I schon besetzt, so Übertragung nach OR II. Der Befehl ist sinnlos, wenn beide Operandenregister besetzt sind.

^1 befördert die Zahl 1 ins Rechenwerk und hat dieselbe Wirkung wie Ablese einer Zelle, welche die Zahl 1 enthält.

§3. Rechenbefehle mit 2 Operanden

Diese sind nur sinnvoll, wenn OR I und OR II besetzt sind. Das Resultat der Operation steht nachher in OR I, während OR II leer ist. Diese Befehle sind:

+ Addition der Zahlen in OR I und OR II

- OR I - OR II

y-x OR II - OR I

× OR I \* OR II

: OR I : OR II

b) The Z 4 switches the converter off automatically:

When the "Fin - Stop"-switch is on and no calculation has been performed in the last 30 seconds (e.g. at the end of a calculation). When entering the machine room, one can see that the Z 4 has automatically turned itself off when the 4 converter switches are still on 1. Switching on again in this case consists of:

1. Removing all films from the readers. Also the papers from the typewriter. Make a note of the operation counter.
2. Turn off the key switch. Only after this, proceed to turning off the "Fin - Stop"-switch. This causes the converters to start up. Then proceed from No. 2 under a), as in a normal startup.

c) Turning off the Z 4. Remove the films, make a note of the operation counter, turn the key switch off, turn the typewriter off, set the 4 converter switches to 0. Hereby the arithmetic unit will be cleared, but not the memory.

d) Automatic shutdown after computation is enabled by setting the "Fin - Stop"-switch to "on". This must be done if the machine room is unattended for any extended time.

**Chapter 2: The Instructions and their Effects**

§1. Start and Fin

"St" Every instruction tape must begin with a start marker "St".

"Fin" The end should be marked with a "Fin" marker (see also Conditional Instructions). This marker ends the computation, clears OR I and OR II, stops the film transport, leaves the red light "End" on and initiates shutdown of the impulse generator (and the converters, if the "Fin - Stop"-switch is also on) 25 seconds later. To start a new computation, "Impulse Generator Start" must be pressed. Cyclical programs, that are formed by endless tapes glued together, naturally must not contain any end markers as they're supposed to run through multiple times. (Exception see "Up").

§2. The Read Instructions

An (A = instruction "Read from memory", n = address running from 0-63). Causes transfer of the number in cell n into OR I. If however OR I is already occupied, transfer to OR II. Instruction has no effect if both operation registers are occupied.

^1 puts the number 1 into the arithmetic unit and has the same effect as reading a cell containing the number 1.

§3. Arithmetic Instructions with 2 Operands

These are only meaningful if both OR I and OR II are occupied. The result is then placed in OR I, while OR II will be empty. The instructions are:

+ Addition of the numbers in OR I and OR II

- OR I - OR II

y-x OR II - OR I

× OR I \* OR II

: OR I : OR II

Maj Beidet die grössere der beiden Zahlen in OR I, OR II.  
Beispiel: Maj von +7 und (-9) = +7

Min Beidet analog die kleinere Zahl  
Beispiel: Min von +7 und (-9) = -9

#### §4. Rechenbefehle mit 1 Operanden

Diese sind nur sinnvoll, wenn OR I besetzt und OR II leer ist.  
Resultat nach OR I.

$\times 1/2$ ,  $\times 2$ ,  $\times (-1)$ ,  $\times 10$ ,  $\times 3$ ,  $\times 1/3$ ,  $\times 1/5$ ,  $\times 1/7$ ,  $\times \pi$ ,  $\times 1/\pi$   
bewirkt Multiplikation der Zahl in OR I mit der betreffenden  
Konstanten. Speziell ergibt  $\times (-1)$  Änderung des Vorzeichens.

$x^2$  bildet das Quadrat der Zahl in OR I, analog:

$\sqrt{x}$ ,  $1/x$ ,  $|x|$ .

sgn x bildet das Vorzeichen einer Zahl, genauer  $\text{sgn } x = x/|x|$ .  
Für  $x = 0$  wird +1 oder ? gebildet, je nach Stellung des  
Schalters "sgn 0 = ".

F pos bildet Maj von 0 und x.

?  $\rightarrow$  1 ist wirkungslos, ausgenommen, wenn in OR I ein ? steht.  
Dann wird +1 gebildet.

#### §5. Der Speicherbefehl

S n Bewirkt Übertragung der Zahl in OR I nach der Zelle Nr. n  
des Speicherwerks unter Löschung von OR I. Der Befehl ist  
nur sinnvoll, wenn OR I besetzt und OR II leer ist.

Rh nur sinnvoll nach einem Rechenbefehl (mit 2 oder 1 Operanden)  
und vor einem Speicherbefehl S n oder Lochbefehl. Verhindert  
Löschung von OR I beim nachfolgenden Speichern.

Tr Das direkte Umspeichern, also eine Folge A 17, S 23 ist  
verboten. Tr ist ein Trennbefehl so dass die Folge Tr, A 17,  
S 23 erlaubt ist. Diese Trennung wirkt weiter, so dass auch  
längere Ketten von Ables- und Speicherbefehlen (z.B. Tr, A 17,  
S 23, A 18, S 24 u.s.w.) sinnvoll sind.  
Ferner kann eine gespeicherte Zahl nicht sofort wieder abgelesen  
werden, vielmehr müssen unterdessen 3 weitere Befehle einge-  
schaltet sein. Ist dies nicht der Fall, so ist unmittelbar  
nach dem Speicherbefehl "Tr" einzuschalten. (S 17, Tr, A 17  
oder S 17, Tr, A 16, A 17), siehe auch "verbotene Befehlsfolgen".

#### §6. Eingang und Ausgang

$\uparrow$  (Eingeben) bewirkt das Aufleuchten des roten Blinklichts als  
Aufforderung zum Eintasten einer Zahl bzw. das Aufleuchten  
eines Protokollfeldes, wenn ein Protokollbefehl vorausgeht.  
Nach Drücken der "Fertig"-Taste wird die Zahl ins Rechenwerk  
befördert, dort ins Dualsystem übersetzt und geht als Dualzahl  
nach OR I. Falls vorher eine andere Zahl in OR I stand, wird  
diese nach OR II befördert. Diese Vertauschung der Operanden  
wird bei unmittelbar nachfolgenden - oder y-x automatisch  
rückgängig gemacht, bei : jedoch nicht.  
Der Befehl ist sinnlos, wenn OR I und OR II besetzt.

$\downarrow$  (Rückübersetzen) nur sinnvoll, wenn OR I besetzt und OR II leer.  
Bewirkt Rückübersetzen der Zahl in OR I ins Dezimalsystem und  
Anzeigen im Lampenfeld. Folgt jedoch ein Protokollbefehl nach,  
so leuchtet gleichzeitig ein Protokollfeld auf. Folgt ein  
Druckbefehl, so erscheint die Zahl nicht im Lampenfeld.  
 $\downarrow$  löscht OR I.

$\downarrow h$  gleich rückübersetzen ohne Löschen von OR I.  
Die aufleuchtende Zahl muss nach Kontrolle oder Notierung

Maj The larger of the numbers in OR I, OR II.  
E.g. Maj of +7 and (-9) = +7

Min Similarly, the smaller of the numbers  
E.g. Min of +7 and (-9) = -9

#### §4. Arithmetic Instructions with 1 Operand

These are only meaningful if OR I is occupied and OR II is empty.  
Result in OR I.

$\times 1/2$ ,  $\times 2$ ,  $\times (-1)$ ,  $\times 10$ ,  $\times 3$ ,  $\times 1/3$ ,  $\times 1/5$ ,  $\times 1/7$ ,  $\times \pi$ ,  $\times 1/\pi$   
causes multiplication of the number in OR I with the correspon-  
ding constant. In particular  $\times (-1)$  flips the sign.

$x^2$  forms the square of the number in OR I, and similarly:

$\sqrt{x}$ ,  $1/x$ ,  $|x|$ .

sgn x forms the sign of a number, more accurately  $\text{sgn } x = x/|x|$ .  
For  $x = 0$  either +1 or ? is formed, depending on the  
position of the switch "sgn 0 = ".

F pos forms Maj of 0 and x.

?  $\rightarrow$  1 has no effect except when OR I contains a ?.  
Then it forms +1.

#### §5. The Memory Instructions

S n causes transfer of the number in OR I into memory cell n  
while also clearing OR I. The instruction is only meaning-  
ful when OR I is set and OR II is empty.

Rh only meaningful after an arithmetic instruction (with 2  
or 1 operands) and before a memory instruction S n or a  
punch instruction. Inhibits a subsequent memory  
instruction from clearing OR I.

Tr A direct load-store in memory, such as A 17, S 23, is not  
allowed. Tr is a separation instruction so that the sequence  
Tr, A 17, S 23 is allowed. The separation persists, so that  
longer read and store sequences (e.g. Tr, A 17, S 23, A 18,  
S 24 etc) are meaningful.  
In addition, a stored number cannot be immediately read back,  
but 3 additional instructions must first be entered. If this  
is not the case, the "Tr" is immediately entered after the  
memory instructions. (S 17, Tr, A 17 or S 17, Tr, A 16, A  
17), see also "Forbidden Instruction Sequences".

#### §6. Input and Output

$\uparrow$  (Input) turns on the blinking light to request the manual entry  
of a number, or lights up a log field if preceded by a  
log instruction. When the "Finished" button is pressed,  
the number is transferred to the arithmetic unit, where it is  
converted to binary and forwarded to OR I as a binary number.  
A number already in OR I will be transferred to OR II. This  
exchange of operands will be automatically undone by a  
immediately following - or y-x instruction, however not by :.  
The instruction is not meaningful when OR I and OR are both  
occupied.

$\downarrow$  (Translate back) only meaningful when OR I is set and OR II is  
empty. Causes translation back to decimal of the number in OR I  
and displays it on the display. If a log instruction follows  
next, a log field will light up. If a print instruction  
follows, the number will not be displayed on the display.  
 $\downarrow$  clears OR I.

$\downarrow h$  identical back translation without clearing OR I.

sofort gelöscht werden, da sonst die Maschine beim nächsten Rückübersetzen blockiert und 25 sc später abstellt.

#### §7. Protokollbefehle

Das Protokoll hat 10 Zeilen mit je 5 Feldern in rechteckiger Anordnung. Außerdem ist eine 0-te Zeile mit 5 Feldern vorhanden (ganz oben), die nicht ausgeleuchtet werden kann. Das Feld ganz links in der 0-ten Zeile ist die Ausgangsstellung. In diese wird automatisch zurückgeschaltet beim Abstellen der Maschine und durch die Befehle Fin, Fin', Spr.

→ Schaltet um ein Feld nach rechts.

↓ Schaltet in derselben Kolonne auf die nächste Zeile.

← Schaltet zurück auf Kolonne 1.

↑ Schaltet zurück auf Zeile 0.

Da die Ausgangslage (Zeile 0) unsichtbar ist, muss man zum Sichtbarmachen des Protokolllichts mindestens 1 Mal ↓ geben. Reihenfolge zum Eingeben: Zuerst Protokollschaltbefehl, dann ↑, " " Rückübersetzen: Zuerst ↓, dann Protokollschaltbefehl.

#### §8. Druckbefehle

D, D<sub>0</sub>, D<sub>1</sub>, D<sub>2</sub>, D<sub>3</sub> nur sinnvoll unmittelbar nach ↓ oder ↓h, bewirken Drucken der rückübersetzten Zahl entsprechend dem betreffenden Druckschema.

R Wagenrücklauf und Zeilenweilerschaltung. Ist wirkungslos, wenn Wagen ganz rechts steht. Sollte am Anfang jedes Rechenplans gegeben werden, der Druckbefehle enthält, damit die 1. Zahl links aussen geschrieben wird.

T Tabulator bewirkt Weiterlaufen des Wagens um 12 Anschläge (= Platz einer Zahl). Dient zur Einschaltung von Zwischenräumen über das normale Maß hinaus.

#### §9. Eingang und Ausgang mit Zahlenstreifen

L nur sinnvoll, wenn OR I besetzt und OR II leer (und kein Rechenplan im Locher). Nur bei Gangart 1 anwendbar. Locht die Zahl in OR I unter Löschung derselben. Dann ist alles bereit für die nächste Lochung. Zur Verhinderung der Löschung setze man Rh vor L. Nach Beendigung der Rechnung erst Film aus At<sub>0</sub> nehmen, dann Knopf "Auslauf" drucken und endlich den Zahlenstreifen abschneiden.

At<sub>1</sub> nur sinnvoll bei Gangart 1 und unter denselben Bedingungen wie A n. Verboten vor einem Ablesebefehl. Bewirkt Abtasten von einem Zahlenstreifen in At<sub>1</sub> und Beförderung in OR I bzw. OR II.

At<sub>0</sub> nur sinnvoll auf einem in At<sub>0</sub> eingelegten Befehlstreifen. Bewirkt Abtasten einer Konstanten, vom Rechenplan in At<sub>0</sub>, d.h. die auf den Befehl At<sub>0</sub> folgenden 5 Befehle bedeuten tatsächlich eine Zahl und einen Leerbefehl. Diese Zahl wird nach OR I befördert. (vgl. auch "Konstante und Rechenplan").

#### §10. Befehle zur Bildung von Ja - Nein - Werten (+1 oder -1)

Diese sind nur sinnvoll, wenn OR I besetzt und OR II leer ist. Im folgenden ist x die Zahl in OR I.

x = 0 Ist die im Befehl angedeutete Bedingung erfüllt, so  
x ≥ 0 wird in OR I der Wert +1 gebildet, andernfalls der  
|x| = ∞ Wert (-1).  
x = ?

The displayed number must, after being checked or noted, be immediately cleared, otherwise at the next back translation the machine will be blocked and 25 s later it will turn off.

#### §7. Log Instructions

The log has 10 rows of 5 fields each in an orthogonal arrangement. In addition a zeroth row with 5 fields available (at the top), which cannot be illuminated. The field at the very left of row 0 is the starting position. It will be reset when the machine is turned off and through the instructions Fin, Fin', Jump.

→ Shifts a field to the right.

↓ Shifts within the same column down to the next row.

← Shifts back to column 1.

↑ Shifts back to row 0.

Since the starting position (row 0) is not visible, at least one ↓ must be given to make the log lights visible.  
Sequence to input: Log shift instruction, then ↑  
Sequence to translate back: First ↓, then log shift instruction

#### §8. Print Instructions

D, D<sub>0</sub>, D<sub>1</sub>, D<sub>2</sub>, D<sub>3</sub> only meaningful immediately after ↓ or ↓h. Will print the back translated number according to the corresponding print schema.

R Carriage return and line feed. Has no effect if the carriage is at the far right. Should be given at the start of every program containing print instructions, so that the first number will be printed to the left.

T Tabulator will forward the carriage 12 strikes (= the space required for one number). Serves to produce a space greater than the normal one.

#### §9. Input and Output using Number Tapes

L only meaningful when OR I set and OR II empty (and no program in punch). Can only be used in Mode 1. Punches the number in OR I and then clears it. Everything will then be ready for the next punching. To prevent clear, place Rh before L. After computation finishes, first remove the film from At<sub>0</sub>, then press the "Discharge" button and, lastly, cut off the number tape.

At<sub>1</sub> only meaningful in Mode 1 and under the same conditions as A n. Not allowed before a read instruction. Reads from a number tape in At<sub>1</sub> and transfers it to OR I or OR II.

At<sub>0</sub> only meaningful on a program tape loaded into At<sub>0</sub>. Reads a constant, from the program in At<sub>0</sub>, i.e. the next 5 commands on the At<sub>0</sub> will be interpreted as a number and an empty feed instruction. This number will be transferred to OR I (see also "Constants in the Program")

#### §10. Instructions for forming Yes-No-Values (+1 or -1)

These are only meaningful when OR I is set and OR II is empty. In the following x is the number in OR I.

x = 0 If the condition indicated in the command is met, then  
x ≥ 0 the value +1 is formed in OR I, otherwise the  
|x| = ∞ value (-1).  
x = ?



$|x| \geq 1$

### §11. Unterpläne und bedingte Befehle

Up ist nur sinnvoll, wenn OR I und OR II leer, bei Gangart 1 und wenn Befehlstreifen (Unterplan) in  $At_1$ . Bewirkt Anhalten von  $At_0$  und Weiterarbeiten entsprechend den Befehlen des Unterplans. Das Schlusszeichen des Unterplans bewirkt zurückgehen auf den Hauptplan (=Plan in  $At_0$ ). Nach "Up" ist obligatorisch "A 63, x = ?, Fin'" zu geben. Dieselbe Folge ist als Vorsichtsmaßnahme nach dem Schlusszeichen des Unterplans zu geben. "Up" ist nur sinnvoll im Hauptplan.

Die folgenden bedingten Befehle sind nur sinnvoll nach einem Ja - Nein - Wert-Befehl; sie sind wirkungslos, wenn der Ja - Nein - Wert (-1) war und werden nur wie folgt ausgeführt, wenn der Ja - Nein - Wert (+1) war; Die bedingten Befehle löschen in beiden Fällen OR I.

Fin' bewirkt im Hauptplan Anhalten wie "Fin". Im Unterplan bewirkt es Anhalten im Gegensatz zum gewöhnlichen Schlusszeichen "Fin", welches das Zurückgehen auf den Hauptplan bedeutet. Will man vom Unterplan bedingt auf den Hauptplan zurück, so hat man die Befehlsreihe "Spr, Fin" im Unterplan anzuwenden (d.h. das unbedingte "Fin" wird bedingt übersprungen).

Spr bewirkt im Haupt- oder Unterplan die Nichtausführung der nachfolgenden Befehle bis zum nächsten Startzeichen, das diesen Sprungbefehl wieder aufhebt. Der Befehl "Spr" schaltet das Protokoll auf die Ausgangsstellung, wenn der Ja-Nein-Wert=+1.

Up' wirkt wie "Up" und unter denselben Bedingungen.

### §12. Befehle, die nicht zu benutzen sind.

Up",  $At_2$ ,  $At_3$ ,  $At_4$ ,  $At_5$

## 3. Kapitel: Die Rechenplanfertigung

### §1. Allgemeines

Zur Rechenplanfertigung ist die Maschine auf Gangart 2 zu schalten. Brennt die Lampe "Zahlenstreifen im Locher", so ist der Knopf "Auslauf" zu betätigen, worauf die Maschine bereit ist. Filmvorrat: Sinkt der Vorrat auf der Trommel (hinten, unterhalb der Mitte des Schaltpultes) unter 50 m, so leuchtet eine Warnlampe "Film < 50 m" auf. Dementsprechend muss bald eine neue Rolle eingelegt werden, wofür nur das Personal des Instituts zuständig ist.

### §2. Die normale Rechenplanfertigung

Ein zu lochender Befehl wird auf der Befehlstastatur eingetastet (Betätigung von 2 Knöpfen für A n und S n). Der eingetastete Befehl erscheint im Lampenfeld, wird aber noch nicht gelocht, sondern kann noch korrigiert werden, indem man die Irrtumtaste drückt und den verbesserten Befehl eintastet. Erkennt man einen Irrtum nach Drücken von "Ablesen" oder "Speichern", so muss zuerst eine fiktive Zahlennummer n gedrückt werden, bevor die Irrtumtaste betätigt wird. (Sonst würde auch der vorhergehende Befehl gelöscht).

Das Lochen eines im Lampenfeld stehenden Befehls erfolgt erst nach Eintasten des nächsten Befehls. Nach dem letzten Befehl ist der Knopf "Auslauf" zu drücken. Dies bewirkt Lochen des letzten Befehls und Auflaufen des Films.

Während der normalen Rechenplanfertigung muss  $At_1$  leer sein.

$|x| \geq 1$

### §11. Subprograms and Conditional Instructions

Up only meaningful when OR I and OR II are empty, in Mode 1 and if an instruction tape (subprogram) is in  $At_1$ . Stops  $At_0$  and continues working according to the instruction in the subprogram. The end marker in the subprogram will return to the main program (=program in  $At_0$ ). After "Up", it's mandatory to give "A 63, x = ?, Fin'". The same sequence is also given after the end marker, as a precaution. "Up" is only meaningful in the main program.

The following conditional instructions are only meaningful after a Yes-No-instruction; they have no effect if the Yes-No-value was (-1) and will only be executed when then Yes-No-value was (+1); The conditional instructions clears OR I in both cases.

Fin' stops like Fin in the main program. In the subprogram, on the other hand, stopping means going back to the main program. If one wants to return conditionally to the main program, the instruction sequence "Jump, Fin" is used in the subprogram (e.g. the unconditional "Fin" will be conditionally skipped).

Jump causes, in both main and sub programs, the instructions up to the next start marker to not be executed. The start marker will resume execution again. The "Jump" instruction shifts the log to the starting position when the Yes-No-value=+1.

U' works like and has the same conditions as "Up".

### §12. Instructions that are not used.

Up",  $At_2$ ,  $At_3$ ,  $At_4$ ,  $At_5$

## Chapter 3: Programming

### §1. General Notes

When programming, the machine should be switched to Mode 2. If the lamp "Number tape in punch" is lit, then "Discharge" should be pressed, whereafter the machine is ready. Film supply: If the supply on the drum (on the back, below the middle of the control panel) drops below 50 m, a warning lamp "Film < 50 m" is lit. Accordingly, a new roll must soon be installed, which is a task only the institute's personnel may perform.

### §2. The Normal Programming Flow

An instruction to be punched is entered on the instruction keypad (using the two keys for A n and S n). The entered instruction will be displayed on the display, however it will not yet be punched, to allow for corrections by pressing the "Correction" key and then entering the corrected instruction. If an error is spotted after pressing "Read" or "Store", a fictive digit number n must be pressed before pressing "Correction" (otherwise the preceding instruction would also be erased).

An instruction displayed on the display is punched only when the next instruction is entered. After the final instruction the "Discharge" key should be pressed. This punches the last instruction and then discharges the film.

$At_1$  must remain empty during normal programming flow.

Der Leerbefehl: Dies ist ein Befehl, der in einem Rechenplan nur die Wirkung hat, dass der Film weiterbefördert wird. Auf dem Rechenplan erscheint an der Stelle eine vollständig ungelochte Lochreihe. Dieser Befehl kann durch Betätigung der Taste "Leerschritt" auf den Rechenplan gegeben werden. Der beim Zusammenkleben eines Films entstehende blanke Teil besteht z.B. aus solchen Leerbefehlen.

### S3. Die genaue Bedeutung der Steuerknöpfe

Es ist zu unterscheiden zwischen den "Befehlsknöpfen" (von denen jeder einem zu lochenden Befehl gemäss 2. Kapitel entspricht), und den "Steuerknöpfen", die nicht einen zu lochenden Befehl bedeuten, sondern folgende Wirkungen haben:

**Kontr.** Wird zum Kontrollieren eines fertigen Plans benutzt, der in  $At_1$  einzulegen ist. Dies funktioniert nur, wenn der Film im blanken Teil vor dem Startzeichen eingelegt wird; in der Mitte eines Plans ist es nur möglich, falls sich dort ein Startzeichen befindet und dasselbe genau auf die Abtasterstifte gelegt wird entsprechend der folgenden Konfiguration:

```
          Startzeichen
    o o x o
      o o o o-Abtasterstifte
```

"Kontr." bewirkt Weiterschalten des Films auf dem nächsten Befehl und Anzeige desselben im Lampenfeld.

**Kontr.** = Dauerkontrolle. Bewirkt kontinuierliches Weiterlaufen des Films solange der Knopf gedrückt wird und Anzeige der dabei durchlaufenen Befehle. Dient weniger zur Kontrolle als zum Überspringen uninteressanter Stücke beim Kontrollieren.

**Lo** Wird benutzt, um einen einzelnen Befehl eines in  $At_1$  eingelegten Rechenplans auf einen neuen Plan im Locher zu übertragen. Bewirkt Lochen des im Lampenfeld aufleuchtenden Befehls, Weiterschalten des in  $At_1$  eingelegten Films zum nächsten Befehl und Aufleuchten desselben, ohne ihn zu lochen.

**Lo** Wird benutzt, um eine Folge von Befehlen eines in  $At_1$  eingelegten Rechenplans auf einen neuen Plan im Locher zu übertragen. Bewirkt kontinuierliches Weiterlaufen des Films in  $At_1$  solange der Knopf gedrückt wird und Aufleuchten und Lochen der dabei durchlaufenen Befehle. Beim Loslassen des Knopfes ist der aufleuchtende Befehl noch nicht gelocht.

Leerschritt ist kein Steuerknopf, sondern ein Befehlsknopf für den oben erwähnten Leerbefehl (der entsprechende Platz auf dem Film bleibt ungelocht, der Befehl ist wirkungslos). Wird hauptsächlich gebraucht, um einen im Lampenfeld stehenden Befehl noch zu lochen.

**Kop** Wenn ein Rechenplan genau ein einziges "Fin"-Zeichen und zwar am Schluss besitzt, so kann er durch diesen Steuerknopf kopiert werden, wobei der Kopiergang nach Drücken des Knopfes unbeeinflussbar, und ohne Anzeige im Lampenfeld bis zum Ende abläuft. Des "Fin"-Zeichen jedoch wird nicht kopiert. Ein "Fin"-Zeichen inmitten des Rechenplans unterbricht den Kopiervorgang. Das Fehlen eines "Fin"-Zeichens am Schluss bewirkt richtiges Kopieren, bringt aber nachher die Maschine in Unordnung. Schlüsselschalter aus- und wieder einschalten, Impulsgeber starten.

**Ersetzen:** Nur zu benutzen bei eingelegtem Plan in  $At_1$ . Dient zum Ersetzen eines auf dem ursprünglichen Plan falschen Befehls. Löscht den im Lampenfeld stehenden Befehl. Dann kann der verbesserte Befehl eingetastet werden. Dieser erscheint im

The empty feed instruction: This is an instruction that only has the role of feeding the film. It is represented as an completely unpunched row on the film. This instruction can be entered into the program by pressing the button "Empty feed". The blank part created by gluing several film together will for example come to be made up of only empty instructions.

### S3. The Exact Meaning of Control Keys

Distinguish between the "Instruction keys" (that are to be punched according to chapter S2) and the "Control keys", that don't have a corresponding punchable instruction, but rather have the following functions:

**Check.** Is used to check a completed program that is loaded in  $At_1$ . This will only work if the film is placed in the blank portion before the start marker; if in the middle of a program it can only be used if there is a following start marker and the reading pins are in this exact configuration:

```
          Start marker
    o o x o
      o o o o-Reading pins
```

CheckChe. works by forwarding the film to the next instruction and displaying that instruction on the display.

**Check.** = Continuous check. Works by continuously feeding the film and displaying the corresponding instructions for as long as the key is pressed. Less useful for actual checks than for skipping uninteresting parts when doing checks.

**Lo** Is used to transfer a single instruction from a program in  $At_1$  over to a new program in the punch. It works by punching the instruction displayed on the display, forwarding the film in  $At_1$  to the next instruction and then displaying that instruction, without punching it.

**Lo** Is used to transfer a sequence of instructions from a film in  $At_1$  over to a new film in the punch. It works by continuously feeding the film in  $At_1$  as long as the key is pressed, and displaying and punching the instructions that are read past. When releasing the button the displayed instruction will not yet be punched.

Empty feed is not a control key, but rather an instruction key for the above mentioned empty instruction (the corresponding location on the film will remain unpunched; the instruction is without effect). Will primarily be used to punch an instruction displayed on the display.

**Copy** If a program contains a single "Fin"-marker and is positioned at the end, it can be copied by pressing this key, whereafter the copying will run uninterruptibly and without display feedback until the end. The "Fin"-marker will however not be copied. A "Fin"-marker in the middle of the program will abort the copying. A missing "Fin"-marker at the end will still copy correctly, but will then bring the machine to an inconsistent state. Turn off and then on with the key breaker, and then start the impulse generator.

**Replace:** Only use when a program is in  $At_1$ . Replaces an incorrect instruction in the original program. Clears the instruction from the display. Then a corrected instruction can be entered. This will be displayed on the display and can

Lampenfeld und kann anschliessend durch Betätigen von "Lo" gelocht werden. Gleichzeitig erscheint der auf den falschen folgende Befehl des Mutterplans im Lampenfeld.

Zurückstellen: Nur zu benutzen bei eingelegtem Plan in At<sub>1</sub>. Dient zum Einfügen von neuen, auf dem ursprünglichen Plan noch nicht vorhandenen Befehlen. Bewirkt Löschung des Lampenfelds. Der dort vorhanden gewesene (noch ungelochte) Befehl wird aufbewahrt. Gleichzeitig wird die Maschine bereitgestellt zum Eintasten der einzuschaltenden Befehle in beliebiger Anzahl. Wie bei der normalen Rechenplanfertigung ist jeweils der im Lampenfeld stehende Befehl noch nicht gelocht und kann daher durch Betätigung der "Irrtumtaste" noch korrigiert werden. Nachdem der letzte einzuschaltende Befehl im Lampenfeld erschienen ist, ist er durch Betätigung von "Lo" zu lochen, worauf der aufbewahrte Befehl sofort im Lampenfeld erscheint.

Auslauf: Bewirkt Lochen des letzten noch im Lampenfeld stehenden Befehls und auslaufen des Films aus dem Locher. Der Film ist zum Abschneiden bereit. Löscht die Lampe "Rechenplan im Locher".

Schluss: Wie Auslauf, es wird aber außerdem nach dem letzten Befehl noch ein "Fin"-Zeichen gelocht.

#### §4. Abändern eines Rechenplans beim Kopieren desselben

Im folgenden wird angenommen, dass ein in At<sub>1</sub> eingelegter Plan mit Hilfe "Lo" oder "Lo" kopiert wird.

1. Weglassen eines Befehls. Man locht solange bis der wegzulassende Befehl im Lampenfeld erscheint, drückt dann die Taste "Kontr". Dies verhindert das Lochen des Befehls, der nächste Befehl erscheint im Lampenfeld.

2. Einfügen von neuen Befehlen. Sind zwischen Befehl x und y weitere Befehle einzuschalten, so locht man bis Befehl y erscheint, drückt "Zurückstellen", tastet die einzuschaltenden Befehle in normaler Rechenplanfertigung ein. Nach dem letzten Einschaltbefehl "Lo" drücken, worauf y erscheint.

3. Ersetzen eines Befehls. Lochen bis der zu ersetzende Befehl im Lampenfeld erscheint. Dann Taste "Ersetzen", Eintasten des neuen Befehls und Weiterlochen.

4. Ersetzen mehrerer Befehle erfolgt besser nicht mit der Taste "Ersetzen", sondern: Normal Lochen bis der erste zu ersetzende Befehl erscheint. Dann Taste "Kontr" bis der erste nicht zu ersetzende Befehl erscheint, Taste "Zurückstellen", Eintasten der neuen Befehle, dann "Lo".

#### §5. Allgemeines zur Herstellung von Befehlsstreifen

1. Bei langen Rechenplänen ist es zweckmäßig, gelegentlich wieder (beim Beginn einer Teilrechnung) ein Startzeichen einzuschalten und durch einen roten Strich auf dem Film wie folgt kenntlich zu machen.

-----o--o--x--o----- Strich  
o o o o

(Das Befehlszeichen besteht aus genau einem Loch in der gezeichneten Lage.) Dies bewirkt, dass der Film (beim Rechnen und beim Kopieren) an einer späteren Stelle eingelegt werden kann, also z.B. Teilrechnungen wiederholt werden können. Der Film läuft nach dem Einlegen in At<sub>0</sub> bis zum nächsten Zwischenstartzeichen; von dort an werden die Befehle ausgeführt.

2. Das "Fin"-Zeichen hat die Konfiguration:

o o x o

then be punched by pressing the "Lo" key. Simultaneously the next instruction after the erroneous one in the original program will be displayed on the display.

Reset: Only use when a program is in At<sub>1</sub>. Used to insert new instructions, not yet existing in the original program. Will clear the display. The current (yet unpunched) instruction will be stored. At the same time, the machine is made ready to accept any number of keyed-in instructions. Like in regular programming, the displayed instruction will not yet be punched and can still be corrected by using the "Correction" key. After the last keyed-in instruction is displayed, it can be punched by pressing the "Lo" key, whereafter the saved instruction is immediately displayed again.

Discharge: Punches the final instruction on the display and then discharges the film from the punch. The film is then ready to be cut off. Clears the light "Program in punch".

End: Like Discharge, but will punch a "Fin"-marker after the final instruction.

#### §4. Modifying a Program by Copying it

In the following it's assumed that a program in At<sub>1</sub> is to be copied using "Lo" or "Lo".

1. Removing an instruction. Punch until the instruction to be removed is displayed on the display. Then press the "Check" key. This stops the punching of the next instruction displayed on the display.

2. Inserting a new instruction. If between instruction x and y new instructions are to be inserted, then punch until instruction y is displayed in the lamp panel. Press "Reset", and then key the instructions to be inserted in the normal way. After the last new instruction has been entered, press "Lo", whereafter y is displayed.

3. Replacing an instruction. Punch up to the instruction to be replaced is displayed in the lamp panel. Then press "Replace", enter the new instruction and resume punching.

4. Replace multiple instructions is typically not done using the "Replace" key, rather: Punch normally until the first instruction to be replaced is displayed. Then press "Check" until the first instruction not to be replaced is displayed, press "Reset", enter the new instructions, then "Lo".

#### §5. General Notes on Program Tape Creation

1. For long programs is it sometimes useful (at the start of a subprogram) to re-punch a Start marker and then make a distinct red line on the film.

-----o--o--x--o----- Line  
o o o o

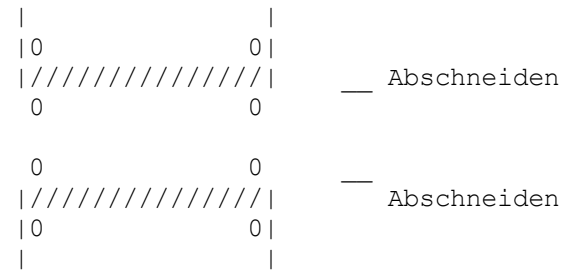
(The instruction symbol is recognized precisely by a hole in the indicated position.) This means the film (when computing and when copying) can be reinserted at a later point in time, for instance so that subprograms can be run multiple times. The film will, after being inserted into At<sub>0</sub>, run up until the next intermediary Start marker. All instructions up until that point will be executed.

2. The "Final"-marker has the following configuration:

o o x o

3. Kurze zyklische Rechenpläne unter 0,5 m Länge geben zu enge Schleifen. Man kopiert den Rechenplan mehrmals nacheinander auf demselben Streifen.

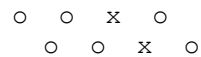
4. Kleben zyklischer Streifen:



Die schraffierten Stellen sind abzuschaben (Film auf Gelatine klebt nicht), mit Filmkitt aufeinanderzukleben und in der Filmpresse trocken zu lassen.

§6. Allgemeines zur Herstellung von Zahlenstreifen

1. Das Zahlenstartzeichen hat folgende Konfiguration:



2. Auch bei langen Zahlenstreifen sollen gelegentlich Zahlenstartzeichen eingefügt werden; dies muss im zugehörigen Befehlsstreifen berücksichtigt werden. Zu diesem Zweck ist in einer freien Zelle n des Speicherwerks die Zahl  $(-1 -8,1 \times 10^{-6})$  zu speichern.

- a) Manuelle Lochung (Gangart 4) Soll ein Zwischenzahlenstart gelocht werden, so Umschalten auf Gangart 1, A n, L; nach beendeter Lochung auf Gangart 4.
- b) Lochung während des programmgesteuerten Rechnens (Gangart 1). Im Rechenplan sind an der betreffenden Stelle die Befehle A n, L einzufügen.

3. Kopieren eines Zahlenstreifens:

Dafür bestehen 2 Möglichkeiten.

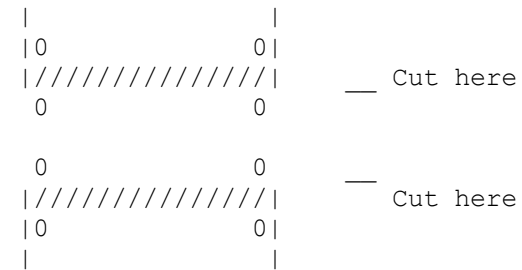
- a) Programmgesteuert mit Hilfe eines zyklischen Rechenplans St, Tr, At<sub>1</sub>, L, At<sub>1</sub>, L, u.s.w. der bewirkt, dass jeweiligen eine Zahl von dem in At<sub>1</sub> eingelegten Zahlenstreifen abgetastet und gelocht wird.
- b) Verwandlung des Zahlenstreifens in einen Befehlsstreifen und Kopieren desselben. Man locht manuell am Anfang vor dem Zahlenstartzeichen ein Befehlstartzeichen, legt den Streifen bei Gangart 2 in At<sub>1</sub> ein. Der Befehlstart erscheint im Lampenfeld, Taste "Kontr" einmal drücken, nachher Lo dauernd.

§7. Konstante auf Rechenplan

Gelegentlich ist es notwendig, Zahlenwerte (Problemkonstanten) auf dem Rechenplan zu haben, um die Verwendung eines besondern Zahlenstreifens zu vermeiden. Steht auf einem in At<sub>0</sub> eingelegten Rechenplan der Befehl At<sub>0</sub>, so bedeutet dies, dass die folgenden 5 Lochreihen nicht 5 Befehle, sondern eine Zahl und einen Leerbefehl bedeuten. Diese Zahl wird nach OR I bzw. OR II befördert und steht dort zur Weiterverwendung bereit. Beim Fertigen des Rechenplans ist dafür wie folgt vorzugehen: Gangart 2 (dedoch At<sub>0</sub> unbelegt), Taste At<sub>0</sub>, Taste "Leerschritt", auf Gangart 9 umschalten, Konstante eintasten, "Fertig" drücken,

3. Short repeated programs under 0,5 m in length will result in too stiff a film roll. In these cases the program is punched multiple times in sequence on the film strip.

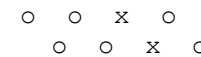
4. Glued repeated program



The shaded areas are scraped clean (the gelatine on the film cannot be glued), glued to each other with film glue and allowed to dry in the film press.

§6. General Notes on Number Tape Creation

1. The number tapes have the following configuration:



2. Also on long number tapes should occasional Start markers be inserted; this must then also be accounted for on the corresponding program tape. For this purpose the number  $(-1 -8,1 \times 10^{-6})$  must be stored in a free memory cell n.

- a) Manual punching (Mode 4) If an intermediary number start marker is to be punched, switch to Mode 1, A n, L; after punching is finished go back to Mode 4.
- b) Punching while a program is running (Mode 1) Insert the instructions A n, L at the appropriate location in the program.

3. Copying a Number Tape

There are two ways of doing this:

- a) In programmed mode using the repeated program St, Tr, At<sub>1</sub>, L, At<sub>1</sub>, etc, which will punch every number from the tape loaded into At<sub>1</sub>.
- b) By turning a number tape into a program tape and then copying that. First manually punch an instruction start marker before the number start marker, then place the tape in At<sub>1</sub> while in mode 2. The instruction start will light up in the lamp panel. Press "Check" once, and then Lo for the duration.

§7. Constants in Programs

It is occasionally necessary to have numerical values in programs, to avoid having to use separate number tapes. If the instruction At<sub>0</sub> is given on a program tape loaded into At<sub>0</sub>, this means that the following 5 lines are not 5 instructions, but rather a number and an empty instruction. This number will be transferred to OR I or OR II and will be available there for further use. When preparing the program, proceed as follows: Set operating mode 2 (but with At<sub>0</sub> unused), press At<sub>0</sub>, press "Empty feed", switch to mode 9, key in the constant, press "Finished", press "L", wait for the punching, go back to mode 2 and proceed with the next instruction.

Taste "L" drücken, Lochung abwarten, auf Gangart 2 zurückschalten und mit dem nächsten Befehl weiterfahren.

Kontrollieren eines solchen Rechenplats wie folgt: Erscheint  $A_{t_0}$  im Lampenfeld, so Taste "Kontr" drücken, Umschalten auf Gangart 9, Taste  $A_{t_1}$  und Taste  $\surd$  drücken. Konstante erscheint im Lampenfeld. Nach Kontrolle zurückschalten auf Gangart 2 und weiter die folgenden Befehle kontrollieren. (Der erste davon ist ein Leerbefehl.)

Eine Konstante kann auch nachträglich beim Kopieren eingefügt werden (vgl. § 4, Nr.2 "Einfügen von neuen Befehlen"). Das Umschalten auf Gangart 9 stört dabei das "Zurückstellen" nicht. Die Konstante kann auch während der Gangart 9 errechnet werden, z.B. Wurzel 2 bedingt: eintasten 2, Taste "Fertig",  $\surd$ , "L".

#### 4. Kapitel: Arithmetische Grundlagen

##### §1. Genauigkeit

Die Maschine rechnet mit 24 Dualstellen. Dem entspricht eine ungefähr 7-stellige dezimale Genauigkeit; trotzdem werden nur 6 Dezimalen gedruckt oder angezeigt. Da das Lochen von Zahlen im Dualsystem erfolgt mit allen 24 Stellen, ist diese Methode des Aufbewahrens von Zahler besser, wenn die Zahlen später wieder gebraucht werden. Infolge des Hin- und Rückübersetzens kann z.B. die Summation von runden Dezimalzahlen (z.B. fortlaufende Tabellierung des Arguments bei Berechnung einer Funktion) bald zu einer nicht mehr runden Dezimalzahl führen.

##### §2. Sonderwerte

Es sind dies folgende Werte, die nicht halblogarithmisch dargestellt werden können und deshalb von der Maschine in besonderer Weise behandelt werden.

Im Lampenfeld:  $0, \infty, +\infty, -\infty, \ll, ?$ .  
Beim Drucken entsprechend:  $0, 1/0, +1/0, -1/0, 0, 0/0$ .

Außer durch explizites Eintasten solcher Werte, können Sonderwerte während das automatischen Rechnens entstehen, nämlich:

0 durch Subtraktion von zwei Zahlen, die in allen Dualstellen übereinstimmen.

$\ll$  entsteht bei Division durch  $\infty$  oder wenn der Exponent einer errechneten Zahl kleiner (-64) wird. Beim Weiterrechnen wird dieser Sonderwert =0 gesetzt.

$\infty$  entsteht bei Division durch 0 oder wenn der Exponent einer errechneten Zahl größer als (+63) wird. Im letzteren Fall ist das Vorzeichen der Zahl bestimmt, je nachdem bildet die Maschine  $+\infty$  oder  $-\infty$ .

? entsteht bei folgenden mathematisch sinnlosen Rechnungen:  
 $0:0, 0 \times \infty, \infty - \infty$  u.s.w.  
Ausserdem, wenn 0 mit einer Zahl, deren Exponent 32 übersteigt, multipliziert wird oder wenn eine Zahl, deren Exponent kleiner als (-32) ist, durch 0 dividiert wird.  
Entsteht beim Wurzelziehen aus einer negativen Zahl ein imaginärer Wert, so wird dieser als ? behandelt.  
Jede Rechenoperation, bei der ein Operand =? ist, ergibt als Resultat wieder ?, ausgenommen bei den 2 Befehlen  $x=?$  und  $?-1$ .

Wie in Kapitel 1, §1 erwähnt, ist eine normale Zahl durch  $7 + 24 = 31$  Dualstellen dargestellt. Ein Sonderwert wird durch eine Eins in der 32-ten Dualstelle gekennzeichnet. (Sonderzeichen).

##### §3. Rechenzeiten

Check a thus entered number as follows: If  $A_{t_0}$  is lit in the display, then press "Check", switch to mode 9, press  $A_{t_1}$  and press  $\surd$ . The constant will be displayed on the display. After the check go back to mode 2 and continue checking the following instructions. (The first of which will be an empty feed instruction.)

A constant can also be inserted later through copying (see §4, no. 2, "Inserting a new instruction"). The switch to mode 9 will then not disturb the "Reset". Constants can also be computed in mode 9, for example the square root of 2 is obtained through: enter 2, press "Finished",  $\surd$ , "L".

#### Chapter 4. Computational Basics

##### §1. Accuracy

The machine operates with 24 binary digits. This corresponds to approximately 7 decimal digit accuracy; despite this only 6 digits will be punched or displayed. Since numbers are punched using the full 24 binary digits, they will be preserved for use in later computations. Otherwise the required translations back and forth would cause e.g. summation of integer numbers (used for instance in running tabulations in function computations) would soon drift out of integerness.

##### §2. Special Values

The following values, which cannot be represented in semi-logarithmic form, will be treated specially by the machine.

On the display:  $0, \infty, +\infty, -\infty, \ll, ?$ .  
When printed:  $0, 1/0, +1/0, -1/0, 0, 0/0$ .

Apart from when explicitly entered using the keyboard, these values can emerge during automatic program executing, as:

0 through subtraction of two numbers that are equal in all binary digits.

$\ll$  on division by  $\infty$  or if the exponent of a computed number is too small (-64). Upon further execution the special value =0 is substituted.

$\infty$  on division by 0 or if the exponent of a computed number is too big (+63). In the latter case the sign of the number is defined, and the machine substitutes  $+\infty$  or  $-\infty$  accordingly.

? is obtained when an undefined mathematical operation is executed:  $0:0, 0 \times \infty, \infty - \infty$  etc.  
Also if 0 is multiplied with a number whose exponent is greater than 32, or when a number whose exponent is smaller than -32 is divided by 0.  
? will also be used to represent an imaginary number being obtained through taking the square root of a negative number. Every computation instructing having =? has one of its operands will give ? as the result, except the two instructions  $x=?$  and  $?-1$ .

As mentioned in Chapter 1, §1, a normal number is formed using  $7 + 24 = 31$  binary digits. A special value is recognized by having a one in the 32th binary position (Special marker).

##### §3. Time Requirements for Computations

Die Rechenoperationen nehmen bei Stellung 4 des Drehzahlreglers folgende Zeiten (in Sekunden) in Anspruch:

Addition und Subtraktion	0,5 - 1,25
$\times 2, \times \frac{1}{2}, \times (-1)$	0,5
Multiplikation	3,5 (Mittelwert)
Division, Quadratwurzel	6,75
Speichern allein	0,5
Leerschritt (Überspringen blanker Teile)	0,25
$\nearrow$ und $\searrow$	3 (Mittelwert)

Speicherung und Ablesung werden in der Regel simultan mit den arithmetischen Operationen ausgeführt und benötigen daher keine zusätzliche Zeit.

Für Stellung 5 des Drehzahlreglers sind diese Zeiten mit 0,9 zu multiplizieren, für Stellung 0 mit 2.

## 5. Kapitel: Verbotene Befehlsfolgen

### §1. Verbotene Folgen und deren Abhilfe

Erfahrungsgemäß werden folgende Befehlsreihen nicht immer korrekt ausgeführt und sind daher durch die jeweiligen rechts angegebenen Befehle zu ersetzen (B bedeutet irgend einen sinnvollen Befehl):

<u>Verbotene Folge</u>	<u>Abhilfe</u>
1) S n, B, A n	S n, Tr, B, A n
2) B, A n, S m	B, Tr, A n, S m
3) Analog für $At_1$ statt A n oder/und L statt S m	
4) Rh, $\searrow$	$\searrow$ h
5) $At_1, A n, B$	A n, $At_1, B$
6) $\nearrow, \searrow$ h, D, L, S n	$\nearrow, L, S n, Tr, A n, \searrow, D$
7) Tr, A n, L, $At_1, S m$	Tr, A n, L, Tr, $At_1, S m$
8) A n, $x^2, S m, R, A p, \searrow$	R an Anfang oder Tr vor R.
9) analog wenn andere Rechenoperationen statt $x^2$	
10) analog wenn T statt R.	
11) $\nearrow$ 1, S n	Tr, $\nearrow$ 1, S n
12) A n, $\nearrow$ 1, -	$\nearrow$ 1, A n, y-x
13) analog bei Addition	
14) B, Tr, S n	B, S n, eventuell Tr,
15) A n, bedingter Befehl	A n, $x \geq 0$ , bedingter Befehl

### §2. Speziell erlaubte und nützliche Befehlsfolgen

- 1) Rh, L, Weiterrechnen (Verhindert Löschen von OR I beim Lochen).
- 2) Rechenoperation, S a, S b, S c u.s.w. (Speichern derselben Zahl in mehreren Zellen). Dabei darf dem ersten Speicherbefehl kein "Rh" vorausgehen.

The computation instructions will take the following times in Seconds (with the speed selector in position 4):

Addition and Subtraction	0,5 - 1,25
$\times 2, \times \frac{1}{2}, \times (-1)$	0,5
Multiplication	3,5 (average)
Division, Square root	6,75
Just storing to memory	0,5
Empty operation (skipping blank parts)	0,25
$\nearrow$ and $\searrow$	3 (average)

Storing to and reading from memory can normally be accomplished simultaneously with computations and require no extra time.

For position 5 on the speed selector these figures should be multiplied by 0,9, for position 0 with 2.

## Chapter 5. Forbidden Instruction Sequences

### §1. Forbidden Sequences and their Resolutions

From experience it is known that the following instruction sequences are not always correctly executed and should be substituted with the sequences on the right (B represents any defined instruction).

<u>Forbidden sequence</u>	<u>Resolution</u>
1) S n, B, A n	S n, Tr, B, A n
2) B, A n, S m	B, Tr, A n, S m
3) analogous for $At_1$ instead of A n and/or L instead of S m	
4) Rh, $\searrow$	$\searrow$ h
5) $At_1, A n, B$	A n, $At_1, B$
6) $\nearrow, \searrow$ h, D, L, S n	$\nearrow, L, S n, Tr, A n, \searrow, D$
7) Tr, A n, L, $At_1, S m$	Tr, A n, L, Tr, $At_1, S m$
8) A n, $x^2, S m, R, A p, \searrow$	R at beginning or Tr before R
9) analogous for other computation instructions instead of $x^2$	
10) analogous for T instead of R.	
11) $\nearrow$ 1, S n	Tr, $\nearrow$ 1, S n
12) A n, $\nearrow$ 1, -	$\nearrow$ 1, A n, y-x
13) analogous for addition	
14) B, Tr, S n	B, S n, optional Tr,
15) A n, conditional instruction A n, $x \geq 0$ , conditional instr.	

### §2. Specifically Allowed and Useful Instruction Sequences

- 1) Rh, L, further computation (Prevents clearing OR I when punching).
- 2) Computation, S a, S b, S c etc. (Stores the same number in multiple cells). Here the first storage instruction cannot be preceded by "Rh".

- 3) Analog: Rechenoperation L, S a, S b, ... u.s.w.  
(L muss vor den Speicherbefehlen und nur einmal stehen).
- 4) "Ja-Nein-Wert"-Befehl,  $\times(-1)$ , bedingter Befehl,  
(Notwendig, um das Kriterium in sein Gegenteil umzukehren.)
- 5) Tr, L, L, L, u.s.w. bewirkt Leertransport des Zahlenstreifens im Locher.
- 6)  $At_1, At_1, At_1, \dots, At_1, +, x=?, x=?, Fin'$ .

Bewirkt Leertransport des Zahlenstreifens im Abtaster, indem die abgetasteten Zahlen vernichtet werden.

### §3. Allgemeine Bemerkungen zum Trennbefehl

Tr nur sinnvoll, wenn OR I und OR II leer. Außerdem beachte man: Es bestehen 4 Varianten des "Umspeicherns", nämlich:

- a) A , S , A , S ...
- b) A , L , A , L ...
- c)  $At_1, S, At_1, S$  ...
- d)  $At_1, L, At_1, L$  ...

Der Trennbefehl (Kap. 2 § 5) erlaubt eine beliebig lange Serie von Umspeicherungen derselben Variante, muss aber eingeführt werden vor jedem Wechsel der Variante (vgl. verbotene Folge Nr. 7).

### 6. Kapitel: Das Rechnen mit Adressen

Häufig muss eine Folge von Befehlen mit vielen Zahlen ausgeübt werden, die in aufeinanderfolgenden Speicherzellen stehen. Die normale Rechenplanfertigung ist dann bei der Z 4 zeitraubend; statt dessen kann der Rechenplan von der Maschine selbst errechnet werden, was an folgendem Beispiel auseinandergesetzt werde.

In den Zellen 0 bis 49 stehen Zahlen  $x_0, x_1, \dots, x_{49}$ .

Auf einem Zahlenstreifen sind Werte  $y_0, y_1, \dots, y_{49}$  gelocht.

Man bilde das skalare Produkte dieser beiden Vektoren. Der normale Rechenplan würde lauten (Bildung der sukzessiven Teilsummen in Zelle 63):  
St,  $\uparrow 1, \uparrow 1, -, S 63$  (Um die Zahl 0 in Zelle 63 zu bilden).

Für  $i = 0$  bis 49:  
A  $i, At_1, \times, A 63, +, S 63$  (fortlaufender Plan).

Der Schluss lautet:  
Tr, A 63,  $\downarrow$ , Fin.

Um das langweilige Eintasten der sich immer wiederholenden Befehle von  $i = 0$  bis  $i = 49$  zu vermeiden, geht man folgendermaßen vor:

- Die Befehle des folgenden Plans sind einzuteilen in
- a) Befehle, die von  $i$  unabhängig sind. (konstante Befehle).
  - b) Befehle, die von  $i$  abhängen. (variable Befehle).

Die konstanten Befehle sind nach Möglichkeit in Vierergruppen (Typus I) zusammenzufassen. Wo dies nicht möglich ist, kann man oft durch Leerbefehle oder wirkungslose Befehle Vierergruppen erreichen. Die übrigen, einzeln anfallenden konstanten oder variablen Befehle ergänzt man (auf alle Fälle aber das erste St-Zeichen) wie folgt zu Vierergruppen (Typus II):

Es sei B der betreffende Befehl. (Handelt es sich um einen variabeln

- 3) Analogous: Computation L, S a, S b, ... etc.  
(L must come only once and before the storage instructions).
- 4) "Yes-No-Value"-instruction,  $\times(-1)$ , conditional instruction,  
(Required if the conditional criterion is to be inverted.)
- 5) Tr, L, L, L, etc will produce blank feeds on the number tape.
- 6)  $At_1, At_1, At_1, \dots, At_1, +, x=?, x=?, Fin'$ .  
  
Will produce blank feeds on the number page in the punch, and ignores the entered numbers.

### §3. General Notes on Separation Instructions

Tr is only meaningful when OR I and OR II are both empty. Also note: There are 4 types of load-store instruction sequences:

- a) A , S , A , S ...
- b) A , L , A , L ...
- c)  $At_1, S, At_1, S$  ...
- d)  $At_1, L, At_1, L$  ...

The separation instruction (chapter 2 § 5) allows sequences of load-store instructions of any length having the same type, but must be inserted before every change of sequence type (see forbidden sequence no. 7).

### Chapter 6. Programs with Addresses

One must often give a list of many numbers to be stored in sequence in memory. It would be time consuming to use the normal way of programming the Z4 for this; instead the program can be generated by the machine itself, as will be shown in the following example.

In cells 0 through 49 are the numbers  $x_0, x_1, \dots, x_{49}$ .

Punched on a number tape are the numbers  $y_0, y_1, \dots, y_{49}$ .

One wants to compute the scalar product between these two vectors. The normal program for this would read (i.e. forming the running sums in cell 63):  
St,  $\uparrow 1, \uparrow 1, -, S 63$  (To set cell 63 to the number 0).

For  $i = 0$  through 49:  
A  $i, At_1, \times, A 63, +, S 63$  (the inner program).

The end reads:  
Tr, A 63,  $\downarrow$ , Fin.

For the boring keying of the repetitive instructions for  $i = 0$  through 49, do as follows:

The instructions of the following program can be divided into:

- a) Instructions that are independent of  $i$ . (constant instructions).
- b) Instructions that depend on  $i$ . (variable instructions).

The constant instructions should whenever possible be grouped in groups of four (Type I). When not possible, it is often possible to add an empty or ineffective command to extend the group to four. The remaining, individually occurring constants or variables, are added (in all cases except for the first start marker) to groups of four like this (Type II):

Let B be the instruction in question. (If  $i$  concerns a variable read or store instruction, use the address of the first  $i$ -value); the groups of four looks like this:

Ablese- oder Speicherbefehl, so ist als Adresse diejenige für den ersten i-Wert einzusetzen); die Vierergruppe lautet dann:

:, S 0, Leerschritt, B

Im Beispiel ergibt sich so aus dem fortlaufenden Plan:

:, S 0, Leerschritt, A 0 (Typus II)

At<sub>1</sub>, ×, A 63, + (Typus I)

:, S 0, Leerschritt, S 63 oder S 63 Leerschritt, Leerschritt,  
(Typus II) Leerschritt (Typus I)

(Auf jeder Zelle steht eine Vierergruppe, die im folgenden als Zahl aufgefasst werden muss).

Dieser abgeänderte Plan wird als normaler Rechenplan mit einem unmittelbar vorhergehenden St-Zeichen gefertigt und das St-Zeichen von Hand in einen Zahlenstart abgeändert (S. 12), womit ein Zahlenstreifen entstanden ist, mit dem gerechnet werden kann.

Mit Hilfe eines ein für alle Mal vorhandenen Befehlstreifens "C" werden diese Zahlen ins Speicherwerk gegeben, wobei die erste Vierergruppe in die Zelle 0 und die weiteren fortlaufend in die nächsten Zellen eingehen.

Die Errechnung des gewünschten fortlaufenden Plans erfolgt mit Hilfe eines anzufertigenden "Superplans". Er lautet im Beispiel:

St, A 0, (-1), LB (Lochen des Befehls "A i")

A 1, L (Lochen der Befehle At<sub>1</sub>, ×, A 63, +)

A 2, (-1), LB bzw. A 2, L (Lochen des Befehls S 63)

^1, A 0, +, S 0 (Vergrößerung der Adresse um 1)

zykl.

L und LB sind Lochbefehle und zwar ist L anzuwenden bei Gruppen vom Typus I und LB bei Gruppen vom Typus II. In der Praxis muss der Superplan ergänzt werden durch ein bedingtes Schlusszeichen zum Aufhören nach i = 49 und durch die notwendigen Befehle zum Lochen des Vorspanns und Auslaufs des zu errechnenden Plans. (Gewöhnliche Methoden der Planfertigung).

Der bisher nicht erwähnte Befehl "LB" ("Lochen eines Befehls") ist nur sinnvoll, wenn OR I besetzt und OR II leer und der vorangehende Befehl kein Ablesebefehl war. Er bewirkt Lochen der 15.-22. (incl.) Dualstelle nach dem Komma der Mantisse b (Kal. 1) der in OR I stehenden Zahl. Diese Dualstellen bedeuten die Verzifferung eines Befehls.

:, S 0, Empty feed, B

The examples gives the following inner program steps:

:, S 0, Empty feed, A 0 (Type II)

At<sub>1</sub>, ×, A 63, + (Type I)

:, S 0, Empty feed, S 63 or S 63, Empty feed, Empty feed  
(Type II) Empty feed (Type I)

(There is a group of four in every cell, which in the following must be treated as a number).

This modified program will, if immediately preceded by a start marker, be ready and the start marker will turn it into a number tape (page 12), on which computation can take place.

Using a ready-made instruction tape "C", these numbers are entered into memory, with the first group of four into cell 0 and the others consecutively into following cells.

The desired computation is then accomplished with the help of a "super-program". In the example it would read:

St, A 0, (-1), LB (Punch instruction "A i")

A 1, L (Punch instruction At<sub>1</sub>, ×, A 63, +)

A 2, (-1), LB or A 2, L (Punch instructions S 63)

^1, A 0, +, S 0 (Increase address by 1)

and repeat.

L and LB are punch instructions, where L is used with groups of type I and LB for groups of type II. In practice the super program must be terminated with a conditional stop marker when i = 49 is reached, and through the necessary instructions to punch start and stop markers into the program (as always required in programs).

The previously unmentioned command "LB" ("punch an instruction") only makes sense if OR I is set and OR II is empty and the preceding instruction was not a load instruction. It works by punching the 15th-22nd (incl.) binary digits of the mantissa of b, after the comma (see chapter 1), of the number present in OR I. These binary digits represent and code for an instruction.