

## Verkürzte Befehlsliste der S 2002

### Inhaltsverzeichnis:

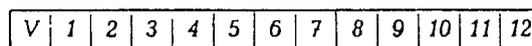
1.	Zahlen- und Befehlsdarstellung	2
2.	Speicher	3
3.	Register	3
4.	Adressenmodifikation	4
5.	Befehle	6
5.1	Allgemeine Steuerwerkbefehle	8
5.2	Indexregisterbefehle	8
5.3	Sprungbefehle	10
5.4	Allgemeine Rechenwerkbefehle	11
5.5	Verschiebefehle	11
5.6	Transferbefehle	12
5.7	Arithmetische Befehle	12
	5.71 Festkommabefehle	12
	5.72 Gleitkommabefehle	13
5.8	Trommelbefehle	14
5.9	Ein- und Ausgabebefehle	15
6.	Runden	17
7.	Bedienungstisch	17

# 1 Zahlen- und Befehlsdarstellung

Die Siemens-Datenverarbeitungsanlage 2002 ist eine dezimal arbeitende Maschine. Die Dezimalziffern werden auf vier parallelen Kanälen durch vierstellige Binärzahlen (Tetraden) im 3-Excess-Code (Stibitz-Code) wie folgt verschlüsselt:

Dezimal	3-Excess-Code
0	0 0 1 1
1	0 1 0 0
2	0 1 0 1
3	0 1 1 0
4	0 1 1 1
5	1 0 0 0
6	1 0 0 1
7	1 0 1 0
8	1 0 1 1
9	1 1 0 0

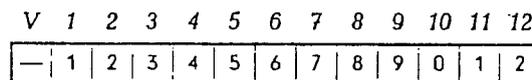
Die einzelnen Dezimalziffern werden in Serie verarbeitet. Je 12 Dezimalziffern bilden zusammen mit einem Vorzeichen ein Wort. Die einzelnen Stellen eines Wortes werden nach folgendem Schema mit V (Vorzeichen), 1, 2, ..., 12 (Stelle 1, ..., Stelle 12) bezeichnet:



Ein Wort kann von der Maschine auf vier verschiedene Arten interpretiert werden:

1.1 als **zwölfstellige Festkommazahl**. Hierbei wird das Komma unmittelbar vor der höchstwertigen Stelle dieser zwölfstelligen Zahl angenommen, d. h. unmittelbar vor Stelle 1.

Beispiel: Die Zahl -0,123456789012 wird in der Maschine wie folgt dargestellt:



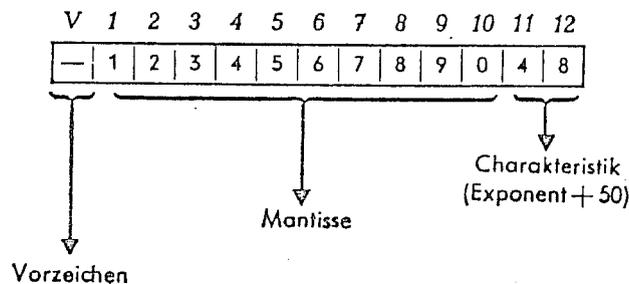
1.2 als **Gleitkommazahl**. Gleitkommazahlen werden in der Maschine, ausgehend von der halblogarithmischen Darstellung

$$x = \pm a \cdot 10^b,$$

in der Form



wiedergegeben. Die *Mantisse* a ist zehnstellig, die *Charakteristik* b+50 zweistellig. Das Komma der Mantisse steht unmittelbar vor der höchstwertigen Stelle der Mantisse. So ist z. B. das Maschinenwort

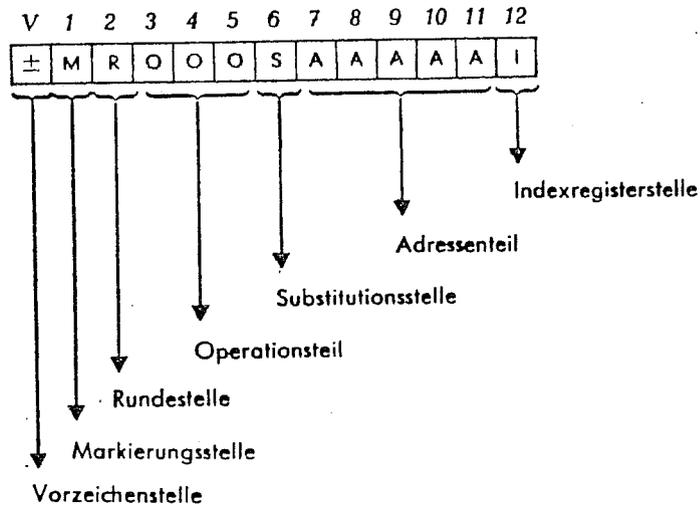


bei einer Interpretation als Gleitkommazahl zu lesen als

$$= - 0,1234567890 \cdot 10^{-2}$$

$$= - 0,001234567890$$

1.3 als **Befehl**, d. h. als Zusammenfassung aller Daten, die zum Kennzeichnen und Ausführen einer Operation erforderlich sind.



Die Datenverarbeitungsanlage 2002 ist eine Ein-Adreß-Maschine, d. h. ein Befehl enthält nur *einen* Adressenteil. Im allgemeinen bezeichnet der Adressenteil diejenige Speicherzelle, mit deren Inhalt der betreffende Befehl auszuführen ist.

Drei Dezimalstellen charakterisieren die auszuführende Operation (z. B. Festkomma-Addition: 055). Die Substitutions- und Indexregisterstelle werden zur Modifikation der Adresse verwendet (siehe Abschnitt 4). Die Rundstelle gibt im Fall einer arithmetischen Operation an, ob gerundet werden soll oder nicht.

Die Vorzeichen- und Markierungsstelle stehen für zusätzliche Kennzeichnungen von Befehlen zur Verfügung.

1.4 als **alpha-numerischer Ausdruck**. Dabei werden jeweils zwei Dezimalziffern zur Kennzeichnung eines alphanumerischen Zeichens verwendet, so daß ein Wort 6 solcher Zeichen enthalten kann.

## 2 Speicher

Die Datenverarbeitungsanlage 2002 enthält *Magnetkernspeicher* als Arbeitsspeicher und einen *Magnettrommelspeicher* als Zubringerspeicher. Der Magnettrommelspeicher gehört jedoch nicht zur Standardausführung; er kann fortgelassen werden. Beide Speicher sind in Speicherzellen aufgeteilt; in jeder Zelle kann ein Wort (12 Dezimalstellen und ein Vorzeichen) gespeichert werden. Vom Magnetkernspeicher stehen Einheiten mit Kapazitäten von 1000, 5000 und 10 000 Worten zur Verfügung; maximal können 10 Einheiten angeschlossen werden. Der Magnettrommelspeicher hat eine Kapazität von 10 000 Worten. Die Speicherzellen sind zu ihrer Identifizierung numeriert; die jeder Speicherzelle zugeordnete Nummer wird als ihre *Adresse* bezeichnet. Verallgemeinert verwendet man diesen Begriff auch zur Kennzeichnung des Inhalts der bezeichneten Speicherzelle (Adresse eines Wortes).

## 3 Register

Das Rechen- und Steuerwerk der Datenverarbeitungsanlage 2002 enthält eine Reihe von Speicherzellen, *Register* genannt, die z. T. ein volles Wort (Vorzeichen und 12 Dezimalstellen), z. T. nur den Adressenteil eines Wortes (5 Dezimalstellen) aufnehmen können.

Von den Registern des Rechenwerks interessieren in diesem Zusammenhang drei, deren Bedeutung im Abschnitt 5 im einzelnen erläutert wird.

<i>Akkumulatives Register AR</i>	1 Vorzeichenstelle 12 Dezimalstellen
<i>Quotienten-Multiplikator-Register QR</i>	1 Vorzeichenstelle 12 Dezimalstellen
<i>Divisor-Register DR</i>	1 Vorzeichenstelle 12 Dezimalstellen

Zum Steuerwerk gehören die folgenden Register:

<i>Befehlsregister BR</i>	1 Vorzeichenstelle
	12 Dezimalstellen
<i>Speicherregister SR</i>	1 Vorzeichenstelle
	12 Dezimalstellen
<i>Indexregister IR 1, IR 2, IR 3</i>	je 5 Dezimalstellen
<i>Befehlszählerregister BZR (IR 4)</i>	5 Dezimalstellen
<i>Adressenregister ADR</i>	5 Dezimalstellen

Zum Trommelspeicher gehören die folgenden Register:

<i>Adressenregister des Trommelspeichers</i>	4 Dezimalstellen
<i>Blocklängeregister des Trommelspeichers</i>	4 Dezimalstellen

Es ist nützlich, folgende Abkürzungen einzuführen: Der Inhalt der Speicherzelle mit der Adresse  $x$  wird durch

$$C(x)$$

bezeichnet.  $C(x)$  ist ein Wort mit den Stellen  $V, 1, \dots, 12$ , die durch

$$C(x)_V, C(x)_1, \dots, C(x)_{12}$$

bezeichnet werden. Entsprechend sollen z. B. durch

$$C(x)_{6 \dots 12}$$

die Stellen 6 bis 12 des Inhalts der Speicherzelle  $x$  bezeichnet werden. Insbesondere wird durch die Stellen 7 bis 11 eines Befehlswortes der Adressenteil dieses Befehls gegeben. Der Einfachheit halber verwendet man häufig auch für

$$C(x)_{7 \dots 11}$$

die Abkürzung

$$C(x)_A$$

Analog zur Abkürzung  $C(x)$  für den Inhalt der Speicherzelle  $x$  werden verwendet:

$$C(AR)$$

für den Inhalt des akkumulativen Registers,

$$C(BZR)$$

für den Inhalt des Befehlszählerregisters usw. für die übrigen Register des Rechen- und Steuerwerks.

## 4 Adressenmodifikationen

Die Datenverarbeitungsanlage 2002 gestattet es, automatisch ohne Inanspruchnahme des Rechenwerks, den Adressenteil des auszuführenden Befehls zu modifizieren, und zwar in Abhängigkeit von der *Substitutionsstelle S (S-Modifikation oder Substitution)* und/oder von der *Indexregisterstelle I (I-Modifikation)* des Befehls. Zur Beschreibung dieser Modifikationen werden die Befehle in zwei Gruppen aufgeteilt, in die *indizierbaren* und *nicht indizierbaren* Befehle.

Die *nicht indizierbaren* Befehle gestatten *keine* I-Modifikationen und sind durch die Endziffer 1 der numerischen Verschlüsselung des Operationsteils gekennzeichnet (siehe Abschnitt 5). Es handelt sich dabei um die Indexregisterbefehle, die den Inhalt eines Indexregisters in ein anderes Register oder in den Speicher geben, den Inhalt eines Indexregisters verändern oder Entscheidungen in Abhängigkeit vom Inhalt der Indexregister fällen. Alle anderen Befehle sind indizierbar.

Unter den *indizierbaren* Befehlen gibt es eine Reihe von Befehlen, bei denen der Adressenteil bedeutungslos ist, wie z. B. beim Stop-Befehl STP. Demzufolge ist eine Adressenmodifikation dieser Befehle sinnlos; das Steuerwerk nimmt bei diesen Befehlen keine Adressenmodifikationen vor. Diese Befehle sind gekennzeichnet durch die Endziffern 0, ferner 03 und 8 der numerischen Verschlüsselung des Operationsteils.

Der auszuführende Befehl (Inhalt der durch den Befehlszählerstand gekennzeichneten Speicherzelle) wird ins Befehlsregister  $BR_1$  gegeben.

Der ins Befehlsregister  $BR$  gelesene Befehl hat also, abgesehen vom Vorzeichen, von der Markierungs- und Rundestelle, die Form

Operationsteil	Substitutionsstelle	Adresse	Indexregisterstelle
$Op$	$s$	$x$	$i$

#### 4.1 Adressenmodifikation für indizierbare Befehle

##### I-Modifikation

Die in der Indexregisterstelle  $I$  stehende Ziffer  $i$  wird abgefragt. Ist  $i=1, 2, 3$  oder  $4$ , so wird zur Adresse  $x$  der Inhalt des Indexregisters  $I, 2, 3$  oder der Inhalt des Befehlszählerregisters  $IR 4$  addiert. Ergibt sich als Summe eine größere Zahl als  $99999$ , so werden von der Summe nur die fünf niedrigwertigen Stellen berücksichtigt. Ist  $i=0$ , so bleibt der Adressenteil unverändert. Die sich nach dieser I-Modifikation aus der Adresse  $x$  ergebende neue Adresse wird mit

$x_i$

bezeichnet.

##### S-Modifikation

An die I-Modifikation kann sich die S-Modifikation in Abhängigkeit von der Ziffer  $s$  in der Substitutionsstelle  $S$  des im  $BR$  stehenden Befehls anschließen.

Ist  $s=0$ , so wird der Befehl mit der Adresse  $x_i$  ausgeführt,

Ist  $s=1$ , so wird der Inhalt der Kernspeicherzelle  $x_i$  gelesen.<sup>1)</sup> Aus dem Inhalt dieser Speicherzelle werden die Stellen 6 bis 12, also die Substitutionsstelle, der Adressenteil und die Indexregisterstelle, ausgeblendet und in die Stellen 6 bis 12 des Befehlsregisters  $BR$  gegeben. Im Befehlsregister bleiben also Vorzeichen-, Markierungs- und Rundestelle sowie der Operationsteil unverändert, während die Substitutionsstelle, der Adressenteil und die Indexregisterstelle durch die entsprechenden Stellen des Inhalts der Speicherzelle  $x_i$  ersetzt werden. Die neu ins Befehlsregister  $BR$  gelesenen Ziffern werden der Einfachheit halber mit  $s_1, x_1$  und  $i_1$  bezeichnet, so daß der jetzige Inhalt des Befehlsregister  $BR$  folgendermaßen dargestellt werden kann:

Operationsteil	Substitutionsstelle	Adresse	Indexregisterstelle
$Op$	$s_1$	$x_1$	$i_1$

Nun wiederholt sich der Zyklus.

Gemäß der jetzt im  $BR$  stehenden Kennziffer  $i_1$  wird mit der Adresse  $x_1$  die I-Modifikation ausgeführt,  $x_{i_1}$  ins Adressenregister gegeben und  $s_1$  abgefragt.

Ist  $s_1=0$ , so wird der Befehl mit der Adresse  $x_{i_1}$  ausgeführt.<sup>2)</sup>

Ist  $s_1=1$ , so wird  $x_{i_1}$  an den Speicher gegeben und der Inhalt der Speicherzelle  $x_{i_1}$  gelesen. Die Stellen 6...12 des Inhalts dieser Speicherzelle werden in die Stellen 6 bis 12 des Befehlsregisters  $BR$  gegeben. Bezeichnet man die jetzt im  $BR$  stehenden Ziffern konsequent mit  $s_2, x_2, i_2$ , so hat  $C(BR)$  die Form:

Operationsteil	Substitutionsstelle	Adresse	Indexregisterstelle
$Op$	$s_2$	$x_2$	$i_2$

Dieser Zyklus wird so oft durchlaufen, bis zum ersten Mal die in der Substitutionsstelle  $S$  des Befehlsregisters  $BR$  stehende Ziffer  $s_n$  gleich Null ist. Dann wird der Befehl mit der Adresse  $x_{i_n}$  ausgeführt.

<sup>1)</sup>  $s = 5, 6, 7$  oder  $8$  wird wie  $s = 0$  interpretiert.

<sup>2)</sup>  $s = 2, 3, 4$  oder  $9$  wird wie  $s = 1$  interpretiert.

#### 4.2 Adressenmodifikation für nicht indizierbare Befehle

Sie unterscheidet sich von der beschriebenen Modifikation für indizierbare Befehle nur durch den Fortfall der I-Modifikationen.

Die durch den Befehlszählerstand gekennzeichnete Speicherzelle des Kernspeichers wird angesteuert und deren Inhalt (der auszuführende Befehl) ins Befehlsregister  $BR$  gelesen. Es wird nun ohne vorangehende I-Modifikation die in der Substitutionsstelle des  $BR$  stehende Ziffer  $s$  abgefragt.

Ist  $s=0$ , so wird der im Befehlsregister stehende Befehl ausgeführt.<sup>1)</sup>

Ist  $s=1$ , so wird die Adresse  $x$  (nicht  $x_i$ ), also die Stellen 7 bis 11 des Befehlsregisters, in das Adressenregister gegeben.<sup>2)</sup> Der Inhalt der Speicherzelle  $x$  wird gelesen, aus dem Inhalt die Stellen 6 bis 11 (nicht 6 bis 12) ausgeblendet und in die Stellen 6 bis 11 des Befehlsregisters gegeben. Bezeichnet man die neu ins Befehlsregister  $BR$  gelesenen Ziffern wie oben mit  $s_i$  und  $x_i$ , so hat der Inhalt des Befehlsregisters  $BR$  die Form:

Operationsteil	Substitutionsstelle	Adresse	Indexregisterstelle
$Op$	$s_i$	$x_i$	$i$

Im  $BR$  bleiben also das Vorzeichen, die Markierungs- und Rundestelle, der Operationsteil und die Indexregisterstelle unverändert.

Dieser Zyklus wiederholt sich so oft, bis zum ersten Mal die in der Substitutionsstelle stehende Ziffer  $s_n$  gleich Null ist. Dann wird der im  $BR$  stehende Befehl mit der Adresse  $x_n$  ausgeführt.

Der allgemeine Ablauf der Adressenmodifikation wird in den beiden folgenden Schemata zusammengefaßt. Es sei angenommen, daß der auszuführende Befehl bereits im Befehlsregister  $BR$  steht.

#### Adressenmodifikation für indizierbare Befehle

1. I-Modifikation:  $x_i$  wird ins Adressenregister gegeben.
2. Ist  $s=0$ , so wird der im Befehlsregister stehende Befehl mit der Adresse  $x_i$  ausgeführt.<sup>3)</sup>  
Ist  $s=1$ , so wird Schritt 3 ausgeführt.<sup>4)</sup>
3. S-Modifikation: Die Stellen 6 bis 12, d. h. die Substitutionsstelle, der Adressenteil und die Indexregisterstelle des Inhalts der Zelle  $x_i$  ersetzen die Stellen 6 bis 12 des Inhalts des Befehlsregisters.
4. Rückkehr zu Schritt 1.

#### Adressenmodifikation für nicht indizierbare Befehle

1. Ist  $s=0$ , so wird der im Befehlsregister stehende Befehl ausgeführt.<sup>3)</sup>  
Ist  $s=1$ , so wird Schritt 2 ausgeführt.<sup>4)</sup>
2. S-Modifikation: Die Stellen 6 bis 11, d. h. die Substitutionsstelle und der Adressenteil des Inhalts der Speicherzelle  $x$  ersetzen die Stellen 6 bis 11 des Inhalts des Befehlsregisters.
3. Rückkehr zu Schritt 1.

## 5 Befehle

Vor der Ausführung eines Befehls wird, wie in Abschnitt 4 dargestellt, der Adressenteil des im Befehlsregister  $BR$  stehenden Befehls in Abhängigkeit von den Indexregister- und Substitutionsstellen so oft modifiziert, bis

$$C(BR)_6 = s = 0$$

wird. Der dann im Befehlsregister  $BR$  gespeicherte Befehl hat also, abgesehen vom Vorzeichen, von der Markierungs- und Rundestelle, die Form

$$Op \quad 0 \quad x \quad i$$

1)  $s = 5, 6, 7$  oder  $8$  wird wie  $s = 0$  interpretiert.

2)  $s = 2, 3, 4$  oder  $9$  wird wie  $s = 1$  interpretiert.

3)  $s = 5, 6, 7$  oder  $8$  wird wie  $s = 0$  interpretiert.

4)  $s = 2, 3, 4$  oder  $9$  wird wie  $s = 1$  interpretiert.

Es genügt demnach, bei der Beschreibung der Befehle der Datenverarbeitungsanlage 2002 die Befehle in der abgekürzten Form

Op x i

zu behandeln.

Die Befehle werden in Abhängigkeit vom Operationsablauf zweckmäßig in Gruppen zusammengefaßt. Alle Befehle einer Gruppe stimmen in der letzten Ziffer der numerischen Verschlüsselung des Operationsteils (Stelle 5 des Befehlswortes) überein.

Die Endziffern der numerischen Verschlüsselung des Operationsteils charakterisieren die einzelnen Befehlsgruppen wie folgt:

- ..0 Steuerwerkbefehle ohne Ansteuern des Speichers  
Adressenmodifikationen werden nicht ausgeführt.
- ..1 Indexregisterbefehle  
Nicht indizierbar
  - .01 S-Modifikation wird nicht ausgeführt
  - .11 S-Modifikation möglich
- ..2 Sprungbefehle  
S-Modifikation möglich  
Indizierbar
- .03 Rechenwerkbefehle ohne Ansteuern des Kernspeichers  
Adressenmodifikationen werden nicht ausgeführt.
- .33 Verschiebepbefehle  
S-Modifikation möglich  
Indizierbar
- ..4 Rechenwerkbefehle mit Schreiben in den Kernspeicher  
S-Modifikation möglich  
Indizierbar
- ..5 Rechenwerkbefehle mit Lesen aus dem Kernspeicher  
S-Modifikation möglich  
Indizierbar
- ..6 Trommelbefehle  
S-Modifikation möglich  
Indizierbar
- ..7 Befehle für Lochkarten- und Großspeichersteuereinheit  
S-Modifikation möglich  
Indizierbar
- ..8 Lochstreifen- und Anlogsichtgeräatbefehle  
Adressenmodifikationen werden nicht ausgeführt.

Bei jeder Befehlsgruppe sind für die nachfolgenden Befehle die in der Indexregisterstelle zugelassenen Ziffern angegeben.

Aus mnemotechnischen Gründen wird für jeden Befehl eine dreistellige Buchstabenbezeichnung (Buchstaben-codes) eingeführt, die mit Vorteil beim Programmieren verwendet werden kann. Mit Hilfe eines geeigneten Programms können bei der Eingabe die Buchstabenbezeichnungen in die zugehörigen Ziffernkombinationen übersetzt werden. Bei Verwendung der Programmiersprache PROSA wird die Übersetzung ebenfalls durch die Datenverarbeitungsanlage vorgenommen.

In der Beschreibung wird für jeden Befehl folgendes angegeben:

Befehlsbezeichnung  
 Buchstabencode  
 Zifferncode  
 Befehlsdefinition

Bei der Beschreibung der Befehle wird angenommen, daß der jeweils behandelte Befehl in der Speicherzelle  $\alpha$  des Kernspeichers gespeichert ist.

5.1 Allgemeine Steuerwerkzeuge  $i = 0, 1, \dots, 9$

**NULLOPERATION**

**NOP**

000 x i      Der Befehlszählerstand wird um 1 erhöht.

**STOP**

**STP**

200 x i      Die Anlage bleibt mit dem Befehlszählerstand  $\alpha + 1$  stehen (in  $\alpha$  steht der STOP-Befehl); im Befehlsregister ist  $C(\alpha)$  gespeichert. Nach Drücken der Taste START wird das Programm mit dem in  $\alpha + 1$  gespeicherten Befehl fortgesetzt.

**HALT**

**HLT**

100 x i      a) Ist die Taste HALT am Bedienungstisch gedrückt, so wird HLT wie STP interpretiert.  
 b) Ist die Taste HALT am Bedienungstisch nicht gedrückt, so wird HLT wie NOP interpretiert.

5.2 Indexregisterbefehle  $i = 1, 2, 3, 4$

**ADDIERE INDEXREGISTER**

**ADI**

(nicht indizierbar)

011 x i       $x$  wird zum Inhalt des Indexregisters  $i$  addiert. Die fünf niedrigstwertigen Dezimalstellen der Summe werden im Indexregister  $i$  gespeichert. Nach Ausführen von  $ADI\ x\ i$  unterbleibt die Erhöhung des Befehlszählerstandes um 1.

**SUBTRAHIERE INDEXREGISTER**

**SBI**

(nicht indizierbar)

111 x i       $x$  wird vom Inhalt des Indexregisters  $i$  subtrahiert. Die fünf niedrigstwertigen Dezimalstellen der Differenz werden im Indexregister  $i$  gespeichert. Nach Ausführen von  $SBI\ x\ i$  unterbleibt die Erhöhung des Befehlszählerstandes um 1.

Bemerkung:

Falls  $C(IR_i) - x$  negativ ist, wird die Differenz als sog. 10-Komplement ins Indexregister  $i$  gegeben.

Beispiel:

$C(IR_i)$	:	00001
$x$	:	00025
$C(IR_i) - x$	:	99976

99976 wird im Indexregister  $i$  gespeichert.

Die in einem Indexregister stehende Größe ist stets positiv.

## ENTSCHEIDE, OB INDEX GLEICH ADRESSE

ICL

(nicht indizierbar)

211 x i

Ist  $C(IRi) = x$ , so wird als nächster Befehl der in  $\alpha + 1$  gespeicherte Befehl ausgeführt.

Ist  $C(IRi) \neq x$ , so wird als nächster Befehl der in  $\alpha + 2$  gespeicherte Befehl ausgeführt.

Der Inhalt des Indexregisters  $i$  bleibt für  $i = 1, 2, 3$  unverändert; für  $i = 4$  ist nach Ausführen des Befehls  $C(IR4) = C(BZR) = \alpha + 1$  bzw.  $\alpha + 2$ .

## ENTSCHEIDE, OB INDEX GRÖßER ODER GLEICH ADRESSE

IGR

(nicht indizierbar)

311 x i

Ist  $C(IRi) < x$ , so wird als nächster Befehl der in  $\alpha + 1$  gespeicherte Befehl ausgeführt.

Ist  $C(IRi) \geq x$ , so wird als nächster Befehl der in  $\alpha + 2$  gespeicherte Befehl ausgeführt.

Der Inhalt des Indexregisters  $i$  bleibt für  $i = 1, 2, 3$  unverändert; für  $i = 4$  ist nach Ausführen des Befehls  $C(IR4) = C(BZR) = \alpha + 1$  bzw.  $\alpha + 2$ .

Bemerkung:

Ist bei einem SBI-Befehl auf Grund von  $C(IRi) - x < 0$  im Indexregister  $i$  als Ergebnis das 10-Komplement gespeichert worden, so wird bei der Ausführung eines nachfolgenden IGR-Befehls das Ergebnis als positive Zahl interpretiert.

Beispiel:

$$\begin{array}{r} C(IRi) : 99993 \text{ äquivalent } -7 \\ x : 00005 \\ \hline C(IRi) > x \end{array}$$

Als nächster Befehl wird der in  $\alpha + 2$  gespeicherte Befehl ausgeführt.

## SPEICHERE INDEXREGISTER

SIR

(nicht indizierbar)

411 x i

$C(IRi)$  wird in der Speicherzelle  $x$  des Kernspeichers an den Stellen 7 bis 11 [Adressenteil von  $C(x)$ ] gespeichert. Die übrigen Stellen von  $C(x)$  werden gleich Null, das Vorzeichen plus gesetzt. Der Inhalt des Indexregisters  $i$  bleibt für  $i = 1, 2, 3$  unverändert; für  $i = 4$  ist nach Ausführen des Befehls  $C(IR4) = C(BZR) = \alpha + 1$ .

## LADE INDEXREGISTER MIT ADRESSE

LIA

(nicht indizierbar)

511 x i

$x$  wird ins Indexregister  $i$  gegeben. Nach Ausführen von LIA  $x4$  unterbleibt die Erhöhung des Befehlszählerstandes um 1.

## LADE INDEXREGISTER VOM BEFEHLSZÄHLER

LID

(nicht indizierbar)

601 x i

$C(BZR)$  wird ins Indexregister  $i$  gegeben. Für  $i = 1, 2, 3$  ist nach Ausführen des Befehls  $C(IRi) = \alpha$ ; für  $i = 4$  ist  $C(IR4) = C(BZR) = \alpha + 1$ .

### TRANSFER EHT VOM INDEXREGISTER

TEI

(nicht indizierbar)

701 x i

$C(IRi)$  wird in die Stellen 7 bis 11 (Adressenteil) des akkumulativen Registers gegeben. Die restlichen Stellen des AR werden gleich Null, das Vorzeichen plus gesetzt. Der Inhalt des Indexregisters  $i$  bleibt für  $i = 1, 2, 3$  unverändert; für  $i = 4$  ist nach Ausführen des Befehls  $C(IR4) = C(BZR) = \alpha + 1$ .

### TRANSFER AUS INS INDEXREGISTER

TAI

(nicht indizierbar)

801 x i

$C(AR)_A$  (die Stellen 7 bis 11 des AR) wird ins Indexregister  $i$  gegeben. Der Inhalt des akkumulativen Registers bleibt unverändert. Nach Ausführen von TAI x 4 unterbleibt die Erhöhung des Befehlszählerstandes.

### 5.3 Sprungbefehle $i = 0, 1, 2, 3, 4$

#### SPRINGE

SPR

122 x i

Als nächster Befehl wird der in der Speicherzelle  $x_i$  gespeicherte Befehl ausgeführt.

#### UNTERPROGRAMMSPRUNG

UNT

222 x i

$\alpha + 1 = C(BZR) + 1$  wird in die Stellen 7 bis 11 der Kernspeicherzelle  $x_i$  gegeben. Die restlichen Stellen der Speicherzelle werden mit Nullen gefüllt, das Vorzeichen wird plus gesetzt. Als nächster Befehl wird der in der Zelle  $x_{i+1}$  gespeicherte Befehl ausgeführt.

#### SPRINGE, FALLS AR GLEICH NULL

SGN

322 x i

Ist  $C(AR) = \pm 000\,000\,000\,000$ , so wird als nächster Befehl der in  $x_i$  gespeicherte Befehl ausgeführt, andernfalls wird der nächste Befehl aus  $\alpha + 1$  entnommen.

#### SPRINGE, FALLS AR UNGLEICH NULL

SUN

622 x i

Ist  $C(AR) \neq \pm 000\,000\,000\,000$ , so wird als nächster Befehl der in  $x_i$  gespeicherte Befehl ausgeführt, andernfalls wird der nächste Befehl aus  $\alpha + 1$  entnommen.

#### SPRINGE, FALLS AR PLUS

SAP

722 x i

Ist  $C(AR)_V = +$ , so wird als nächster Befehl der in  $x_i$  gespeicherte Befehl ausgeführt, andernfalls wird der nächste Befehl aus  $\alpha + 1$  entnommen.

#### SPRINGE, FALLS AR MINUS

SAM

822 x i

Ist  $C(AR)_V = -$ , so wird als nächster Befehl der in  $x_i$  gespeicherte Befehl ausgeführt, andernfalls wird der nächste Befehl aus  $\alpha + 1$  entnommen.

#### SPRINGE BEI ÜBERLAUF

SUL

922 x i

Ist die ADD-Überlaufanzeige gesetzt (Lampe ADD-ÜL leuchtet), so wird als nächster Befehl der in  $x_i$  gespeicherte Befehl ausgeführt und die ADD-Überlaufanzeige abgeschaltet, andernfalls wird der nächste Befehl aus  $\alpha + 1$  entnommen. Die ADD-Überlaufanzeige wird automatisch gesetzt, wenn bei folgenden Operationen ein Überlauf des AR eintritt: ADD, SUB, ADU, SBU, RND; DVL mit Runden, DVK mit Runden.

4 Allgemeine Rechenwertbefehle  $i = 0, 1, 2, 3, 4$  bei UND, ADM  
 $i = 0, 1, \dots, 9$  bei KPL, ABP, ABM, RND

### KOMPLEMENTIERE

KPL

103 x i C(AR) wird durch  $-C(AR)$  ersetzt.

### ABSOLUTBETRAG PLUS

ADP

103 x i C(AR) wird durch  $+|C(AR)|$  ersetzt.

### ABSOLUTBETRAG MINUS

ABM

103 x i C(AR) wird durch  $-|C(AR)|$  ersetzt.

### RUNDE

RND

103 x i Ist  $C(QR)_i \geq 5$ , so wird zum Betrag des AR in der niedrigstwertigen Stelle eine 1 addiert, falls  $C(AR)_i = C(QR)_i$ , und vom Betrag des AR in der niedrigstwertigen Stelle eine 1 subtrahiert, falls  $C(AR)_i \neq C(QR)_i$ . Ist  $C(QR)_i < 5$ , so bleibt  $C(AR)$  unverändert.  $C(QR)$  bleibt erhalten. Bei AR-Überlauf wird die ADD-Überlaufanzeige gesetzt, und die Anlage stoppt, wenn nicht die Taste ÜBERL am Bedienungstisch gedrückt ist.

### UND BEFEHL

155 x „Logisches Und“.  $C(AR)_k$  bleibt erhalten, falls  $C(x_i)_k \neq 0$  ist, sonst wird  $C(AR)_k = 0$  gesetzt,  $k = 1, \dots, 12$ . Das Vorzeichen des AR bleibt unverändert.

### ADRESSENMODIFIKATION

ADM

155 x i  $C(AR)_7 \dots 12$  (Adressenteil) bleibt erhalten. Die übrigen Stellen des AR, d.h.  $C(AR)_i, 1 \dots 6, 12$  werden durch  $C(x_i)_i, 1 \dots 6, 12$  ersetzt.

### Verschiebefehle $i = 0, 1, 2, 3, 4$

Die Verschiebefehle werden die Inhalte des AR bzw. AR und QR um  $n$  Stellen nach rechts oder links verschoben, entsprechend einer Multiplikation mit  $10^{-n}$  oder  $10^n$ . Die frei werdenden Stellen werden durch Nullen besetzt. Die Anzahl  $n$  der Verschiebestellen wird durch die beiden niedrigstwertigen Ziffern der modifizierten Adresse  $x_i$  ( $n = x_i \text{ modulo } 100$ ) gegeben. Die restlichen drei Adressenstellen werden beliebig besetzt werden.

### VERSCHIEBE AR NACH RECHTS

VAR

133 x i Die Stellen 1 bis 12 des AR werden um  $x_i$  (modulo 100) Stellen nach rechts verschoben. Frei werdende Stellen des AR werden durch Nullen besetzt. Die über Stelle 12 des AR hinausgeschobenen Ziffern gehen verloren. Das Vorzeichen des AR bleibt erhalten (auch bei VAR 12).

### VERSCHIEBE AR NACH LINKS

133 x i Die Stellen 1 bis 12 des AR werden um  $x_i$  (modulo 100) Stellen nach links verschoben. Frei werdende Stellen des AR werden durch Nullen besetzt. Die über Stelle 1 des AR hinausgeschobenen Ziffern gehen verloren. Das Vorzeichen des AR bleibt erhalten (auch bei VAL 12).

## VERSCHIEBE AR UND QR NACH RECHTS

333 x i

Die Stellen 1 bis 12 des AR und QR werden, als 24stelliges Register

$$AR_1 \dots 12 QR_1 \dots 12$$

betrachtet, um  $x_i$  (modulo 100) Stellen nach rechts verschoben. Die über Stelle 12 des AR hinausgeschobenen Ziffern werden ins QR (Stelle 1 und folgende) gegeben. Frei werdende Stellen des AR und QR werden durch Nullen besetzt. Die über Stelle 12 des QR hinausgeschobenen Ziffern gehen verloren. Das Vorzeichen des QR wird durch das Vorzeichen des AR ersetzt (auch bei VQR 0).

VQR

## VERSCHIEBE AR UND QR NACH LINKS

433 x i

Die Stellen 1 bis 12 des AR und QR werden, als 24stelliges Register

$$AR_1 \dots 12 QR_1 \dots 12$$

betrachtet, um  $x_i$  (modulo 100) Stellen nach links verschoben. Die über Stelle 1 des QR hinausgeschobenen Ziffern werden ins AR (Stelle 12 und vorangehende) gegeben. Frei werdende Stellen des AR und QR werden durch Nullen besetzt. Die über Stelle 1 des AR hinausgeschobenen Ziffern gehen verloren. Das Vorzeichen des AR wird durch das Vorzeichen des QR ersetzt (auch bei VQL 0).

VQL

### 5.6 Transferbefehle $i = 0, 1, 2, 3, 4$

#### TRANSFER EIN PLUS

533 x i

$C(AR)$  wird durch  $C(x_i)$  ersetzt.

TEP

#### TRANSFER EIN MINUS

633 x i

$C(AR)$  wird durch  $-C(x_i)$  ersetzt.

TEM

#### LADE QR VOM SPEICHER

833 x i

$C(QR)$  wird durch  $C(x_i)$  ersetzt.

LQS

#### TRANSFER AUS ZUM SPEICHER

444 x i

$C(AR)$  wird in der Speicherzelle  $x_i$  gespeichert.  $C(AR)$  bleibt unverändert.

TAS

#### SPEICHERE QUOTIENTENREGISTER

844 x i

$C(QR)$  wird in der Speicherzelle  $x_i$  gespeichert.  $C(QR)$  bleibt unverändert.

SQR

### 5.7 Arithmetische Befehle $i = 0, 1, 2, 3, 4$

#### 5.71 Festkommabefehle

#### FESTKOMMA-ADDITION

055 x i

$C(x_i)$  wird zu  $C(AR)$  addiert und die Summe ins AR gegeben. Bei AR-Überlauf wird die ADD-Überlaufanzeige gesetzt (Lampe ADD-OL leuchtet), und die Anlage stoppt mit dem Befehlszählerstand  $a+1$ , wenn nicht die Taste ÜBERL am Bedienungstisch gedrückt ist.

ADD

#### FESTKOMMA-ADDITION OHNE ÜBERLAUF-STOP

085 x i

Wie ADD, jedoch stoppt die Anlage bei Überlauf nicht.

ADU

### FESTKOMMA-SUBTRAKTION

SUB

133 x i      C(x<sub>i</sub>) wird von C(AR) subtrahiert und die Differenz ins AR gegeben. Sonst wie ADD.

### FESTKOMMA-SUBTRAKTION OHNE ÜBERLAUFSTOP

SBU

183 x i      Wie SUB, jedoch stoppt die Anlage bei Überlauf nicht.

### FESTKOMMA-MULTIPLIKATION

MLT

233 x i      C(x<sub>i</sub>) wird mit C(AR) multipliziert. Die 12 höchstwertigen Stellen des 24stelligen Produkts stehen im AR, die 12 niedrigstwertigen Stellen im QR. Das Vorzeichen des QR ist gleich dem Vorzeichen des AR.

### FESTKOMMA-DIVISION KURZ

DVK

333 x i      C(AR) wird durch C(x<sub>i</sub>) dividiert. Ist

$$|C(x_i)| \leq |C(AR)|,$$

so bleibt die Anlage vor Beginn der Division stehen (am Bedienungstisch leuchtet die Lampe DIV-ÜL). Der Dividend C(AR) bleibt unverändert. Die Division wird ausgeführt, falls

$$|C(x_i)| > |C(AR)|.$$

Der 12stellige Quotient (Komma vor der höchstwertigen Stelle) mit Vorzeichen steht im AR, der Rest im QR. Das Vorzeichen des QR ist gleich dem Vorzeichen des Dividenten.

### FESTKOMMA-DIVISION LANG

DVL

433 x i      C(AR) und C(QR)<sub>1...12</sub> werden als 24stelliger Dividend behandelt, C(x<sub>i</sub>) als 12stelliger Divisor. Sonst wie DVK.

#### 5.72 Gleitkommabefehle

Die Darstellung der durch die nachstehend aufgeführten Gleitkommaoperationen verarbeiteten Gleitkommazahlen in der Form

$$x = \pm a \cdot 10^b$$

wird für von Null verschiedene x erst durch eine zusätzliche Forderung an a eindeutig, z. B. durch die Bedingung

$$0,1 \leq |a| < 1$$

(d. h., daß die höchstwertige Mantissenstelle  $\neq 0$  sein soll). Für die Gleitkomma-Null muß man eine weitere Verabredung treffen, z. B. die, daß für  $x=0$  (und somit  $a=0$ ) der Exponent b gleich dem kleinsten von der Maschine noch verarbeiteten Exponenten ist, im Falle der Datenverarbeitungsanlage 2002 also gleich -50 (Charakteristik = 00). Gleitkommazahlen, die diesen zusätzlichen Bedingungen genügen, heißen normalisierte Gleitkommazahlen, und entsprechend Gleitkommaoperationen, deren Resultate in normalisierter Form anfallen, normalisierte Gleitkommaoperationen.

Eine nicht normalisierte Gleitkommazahl kann durch den Befehl GAN x<sub>i</sub> mit C(x<sub>i</sub>) = 0 normalisiert werden.

### GLEITKOMMA-ADDITION, NORMALISIERT

GAN

063 x i      C(x<sub>i</sub>) wird als Gleitkommazahl zu C(AR) addiert. Die normalisierte Summe C(AR) + C(x<sub>i</sub>) wird ins AR und QR gegeben.

### GLEITKOMMA-ADDITION, UNNORMALISIERT

GAU

075 x i      C(x<sub>i</sub>) wird als Gleitkommazahl zu C(AR) addiert. Die (unnormalisierte) Summe C(AR) + C(x<sub>i</sub>) wird ins AR und QR gegeben.

### GLEITKOMMA-SUBTRAKTION, NORMALISIERT

GSN

165 x i      C(x<sub>i</sub>) wird als Gleitkommazahl von C(AR) subtrahiert. Die (normalisierte) Differenz C(AR) - C(x<sub>i</sub>) wird ins AR und QR gegeben.

**GLEITKOMMA-SUBTRAKTION, UNNORMALISIERT**

**GSU**

175  $x_i$   $C(x_i)$  wird als Gleitkommazahl von  $C(AR)$  subtrahiert. Die (unnormalisierte) Differenz  $C(AR) - C(x_i)$  wird ins  $AR$  und  $QR$  gegeben.

**GLEITKOMMA-MULTIPLIKATION, NORMALISIERT**

**GMN**

265  $x_i$   $C(x_i)$  wird als Gleitkommazahl mit  $C(AR)$  multipliziert. Das normalisierte Produkt  $C(AR) \cdot C(x_i)$  wird ins  $AR$  und  $QR$  gegeben.

**GLEITKOMMA-MULTIPLIKATION, UNNORMALISIERT**

**GMU**

275  $x_i$   $C(x_i)$  wird als Gleitkommazahl mit  $C(AR)$  multipliziert. Das (unnormalisierte) Produkt  $C(AR) \cdot C(x_i)$  wird ins  $AR$  und  $QR$  gegeben.

**GLEITKOMMA-DIVISION, NORMALISIERT**

**GDN**

365  $x_i$   $C(AR)$  wird als Gleitkommazahl durch  $C(x_i)$  dividiert. Der Quotient wird ins  $AR$ , der Rest ins  $QR$  gegeben.

**GLEITKOMMA-DIVISION, UNNORMALISIERT**

**GDU**

375  $x_i$   $C(AR)$  wird als Gleitkommazahl durch  $C(x_i)$  dividiert. Der Quotient wird ins  $AR$ , der Rest ins  $QR$  gegeben.

**5.8 Trommelbefehle  $i = 0, 1, 2, 3, 4$**

**TROMMELADRESSE**

**TRA**

066  $x_i$   $x_i$  wird in das Adressenregister des Trommelspeichers gegeben.

**BLOCKLÄNGE**

**BLL**

166  $x_i$   $x_i$  wird in das Blocklängeregister des Trommelspeichers gegeben.

**SCHREIBE AUF DIE TROMMEL**

**STR**

466  $x_i$  Beginnend mit der Speicherzelle  $x_i$  des Kernspeichers wird ein Block, dessen Länge durch den Inhalt des Blocklängeregisters des Trommelspeichers gegeben ist, in den Trommelspeicher übertragen. Der Anfang des Blocks auf der Trommel ist durch die im Adressenregister des Trommelspeichers stehende Adresse gekennzeichnet.

Ist  $a$  die im Adressenregister des Trommelspeichers gespeicherte Trommeladresse,  $b$  die im Blocklängeregister des Trommelspeichers gespeicherte Blocklänge, so wird durch

STR 466  $x_i$

der Inhalt der Speicherzellen

$x_i$   
 $x_i + 1$   
 $\cdot$   
 $\cdot$   
 $\cdot$   
 $x_i + (b - 1)$

des Kernspeichers gespeichert in den Speicherzellen

$a$   
 $a + 1$   
 $\cdot$   
 $\cdot$   
 $\cdot$   
 $a + (b - 1)$

des Trommelspeichers.

Während der Ausführung des Befehls STR wird der Inhalt des Adressenregisters und des Blocklängeregisters gelöscht.

# LIES VON DER TROMMEL

LTR

366 x i

Ist  $a$  die im Adressenregister des Trommelspeichers gespeicherte Trommeladresse,  $b$  die im Blocklängeregister des Trommelspeichers gespeicherte Blocklänge, so wird durch

LTR 366 x i

der Inhalt der Speicherzellen

$a$   
 $a+1$   
.  
.  
 $a+(b-1)$

des Trommelspeichers gespeichert in den Speicherzellen

$x_i$   
 $x_i+1$   
.  
.  
 $x_i+(b-1)$

des Kernspeichers.

Während der Ausführung des Befehls LTR wird der Inhalt des Adressenregisters und des Blocklängeregisters gelöscht.

### Bemerkung:

Die Befehle TRA und BLL sind innerhalb des Programms nicht an eine bestimmte Reihenfolge gebunden. Beide Befehle müssen aber gegeben worden sein, bevor durch LTR oder STR ein Blocktransfer ausgelöst werden soll. Die Befehle der Folgen TRA, BLL, LTR bzw. STR oder BLL, TRA, LTR bzw. STR müssen nicht unmittelbar nacheinander gegeben werden, sondern können durch beliebig viele andere Befehle getrennt werden. Bei BLL 0 wird der zugehörige LTR- oder STR-Befehl wie NOP ausgeführt.

## 5.9 Ein- und Ausgabebefehle

### 5.91 Lochstreifenbefehle $i = 0, 1, \dots, 9$

Auf der Eingabeseite steht ein Lochstreifenabtaster zur Verfügung, der im Zügig-Betrieb 200 Zeichen pro Sekunde, im Start-Stop-Betrieb maximal 200 Zeichen pro Sekunde abtasten kann. Die Resultate werden von einem Streifenlocher mit einer Geschwindigkeit von 60 Zeichen in der Sekunde gelocht oder auf einem Blattschreiber mit einer Geschwindigkeit von 10 Zeichen pro Sekunde gedruckt.

Für die Daten-Eingabe sind die beiden folgenden Möglichkeiten zu unterscheiden:

1. Eingabe ZÜGIG                      Eine festverdrahtete Eingabe-Steuerung übernimmt das Einspeichern der in normierter Weise auf dem Lochstreifen gelochten Daten in den Kernspeicher
2. Eingabe START-STOP                Die einzelnen Zeichen des Lochstreifens werden durch ein schon im Speicher stehendes Programm nacheinander gelesen. Zwischen dem Abruf der einzelnen Zeichen kann gerechnet werden.

Bei der Daten-Ausgabe kann durch Betätigen der Taste LOCHER oder BLATTSCHREIBER an der Ausgabeinheit festgelegt werden, ob an den Streifenlocher oder an den Blattschreiber oder an beide ausgegeben werden soll. Die Taste PROGRAMM an der Ausgabeinheit gestattet die programmierte Auswahl des Lochers oder des Blattschreibers als Ausgabegerät mit den beiden Umschaltbefehlen DLO und DBL.

## LIES ALPHANUMERISCH

258 x i Die 5er-Kombination unter dem Abtastkopf wird abgetastet und durch zwei Dezimalstellen ( $Z_a, Z_n$ ) verschlüsselt;  $Z_a$  wird an die Stelle 11,  $Z_n$  an die Stelle 12 des AR gegeben.  $C(AR)_7, 1 \dots 10$  bleibt erhalten. Anschließend wird der Lochstreifen um ein Zeichen weitertransportiert.

LAN

## LIES NUMERISCH

158 x i Wie LAN mit dem Unterschied, daß  $Z_a$  unterdrückt wird.  $C(AR)_7, 1 \dots 11$  bleibt erhalten.

LNМ

## DRUCKE ALPHANUMERISCH

248 x i A. Locher  
Die durch  $C(AR)_{11, 12}$  gegebene zweistellige Dezimalzahl ( $Z_a, Z_n$ ) wird in eine 5er-Kombination umgesetzt und diese gelocht. Anschließend wird der Lochstreifen um ein Zeichen weitertransportiert.

DAN

B. Blattschreiber  
Die durch  $C(AR)_{11, 12}$  gegebene zweistellige Dezimalzahl ( $Z_a, Z_n$ ) wird entschlüsselt und das entsprechende Zeichen in Abhängigkeit vom Zustand (Bu, Zi) des Blattschreibers gedruckt. Anschließend wird der Typenhebelwagen des Blattschreibers um ein Zeichen vorgeschoben. Falls es sich bei dem entschlüsselten Zeichen um Buchstaben-, Ziffernumschaltung, Wagenrücklauf oder Zeilenvorschub handelt, wird der Blattschreiber in den neuen Zustand gebracht.

Der Inhalt des AR bleibt erhalten.

## DRUCKE NUMERISCH

148 x i Wie DAN mit dem Unterschied, daß bei der Ausgabeverschlüsselung nur  $Z_n$  berücksichtigt wird.

DNM

## DRUCKE PLUS — MINUS

048 x i A. Locher  
Die dem Vorzeichen des AR entsprechende 5er-Kombination (+ = 10001; — = 11000) wird gelocht und der Lochstreifen um ein Zeichen weitertransportiert.

DPM

B. Blattschreiber  
Das Vorzeichen des AR wird gedruckt und der Typenhebelwagen des Blattschreibers um ein Zeichen vorgeschoben. (Der Blattschreiber muß vorher in Ziffernstellung gebracht worden sein, sonst wird „Z“ bzw. „A“ gedruckt.)

## DRUCKE WAGENRÜCKLAUF

348 x i A. Locher  
Die dem Wagenrücklauf entsprechende 5er-Kombination (00010) wird gelocht und der Lochstreifen um ein Zeichen weitertransportiert.

WRL

B. Blattschreiber  
Der Typenhebelwagen des Blattschreibers wird an den Zeilenanfang zurückgesetzt.

## DRUCKE ZEILENVORSCHUB

ZLV

448 x i

### A. Locher

Die dem Zeilenvorschub entsprechende 5er-Kombination (01000) wird gelocht und der Lochstreifen um ein Zeichen weitertransportiert.

### B. Blattschreiber

Das Papier des Blattschreibers wird um eine Zeile vorgeschoben.

## DRUCKE ZWISCHENRAUM

ZWR

548 x i

### A. Locher

Die dem Zwischenraum entsprechende 5er-Kombination (00100) wird gelocht und der Lochstreifen um ein Zeichen weitertransportiert.

### B. Blattschreiber

Der Typenhebelwagen des Blattschreibers wird um ein Zeichen vorgeschoben.

## DRUCKE AKKUMULATIVES REGISTER

DAR

648 x i

### A. Locher

Der Inhalt des  $AR$  wird in der Reihenfolge  $C(AR)_y, C(AR)_1, \dots, C(AR)_{12}$  gelocht und der Lochstreifen um ein Zeichen weitertransportiert.

### B. Blattschreiber

Der Inhalt des  $AR$  wird in der Reihenfolge  $C(AR)_y, C(AR)_1, \dots, C(AR)_{12}$  gedruckt und der Typenhebelwagen des Blattschreibers um ein Zeichen vorgeschoben. (Der Blattschreiber muß vorher in Ziffernstellung gebracht worden sein.)

Nach Ausführen des Befehls ist

$C(AR) = +000\ 000\ 000\ 000.$

## 6 Runden

Das Runden nach arithmetischen oder Verschiebeoperationen geschieht grundsätzlich in Abhängigkeit von der Besetzung der Rundestelle (Stelle 2 des Befehlswortes) und nach vollständiger Ausführung der betreffenden Operation.

Ist Stelle 2 des Befehlswortes gleich 0 bzw. 5, 6, 7 oder 8, so wird gerundet, falls die betreffende Operation eine Gleitkommaoperation ist, und nicht gerundet im Fall einer Festkomma- oder Verschiebeoperation.

Ist Stelle 2 des Befehlswortes gleich 1 bzw. 2, 3, 4 oder 9, so wird nicht gerundet, falls die betreffende Operation eine Gleitkommaoperation ist, und gerundet im Falle einer Festkomma- oder Verschiebeoperation. Diese Festlegung entspricht der Tatsache, daß man bei Gleitkommaoperationen meistens, bei Festkomma- und Verschiebeoperationen weniger häufig runden will.

## 7 Bedienungstisch

Am Bedienungstisch sind alle Anzeigen, Schalter und Tasten zusammengefaßt, die für den Betrieb und die Überwachung der Anlage notwendig sind.

Durch die Taste EIN werden alle Betriebsspannungen der Anlage eingeschaltet; die Lampe AK leuchtet, wenn die Anlage betriebsbereit ist. Durch die Taste AUS werden alle Betriebsspannungen der Anlage abgeschaltet.

In einem 13stelligen Anzeigefeld wird fortlaufend je nach Tastenstellung der Inhalt des  $AR$ ,  $QR$ ,  $BR$  oder  $SR$  im 3-Exzess-Code durch Glühlampen angezeigt; in einem fünfstelligen Anzeigefeld je nach Tastenstellung der Inhalt des  $BZR$ ,  $IR1$ ,  $IR2$  oder  $IR3$ . Beiden Anzeigefeldern sind dezimale Anzeigefelder zugeordnet, die nach Drücken der Taste DEZ ANZ den Inhalt der durch die gedrückten Tasten gekennzeichneten Register in dezimaler Form wiedergeben.

Die Funktionen der übrigen Tasten des Bedienungstisches hängen von den sich gegenseitig ausschließenden Tasten **PROGR** und **MANUELL** ab. Bei gedrückter Taste **PROGR** wird die Anlage vom gespeicherten Programm gesteuert. Ein Betätigen der für manuelle Operationen vorgesehenen Tasten **EING SP**, **AUSG SP**, **EING AR**, **EING BZR**, **LÖSCH I**, **LÖSCH II**, **LÖSCH III** und **FA LÖSCH** ist dann wirkungslos. Wird die Taste **MANUELL** gedrückt, so wird die Programmsteuerung nach Ausführung des gerade bearbeiteten Befehls unterbrochen, und die Anlage führt nach Drücken der entsprechenden Tasten folgende Einzeloperationen aus:

<b>EING SP</b>	In die durch das fünfstellige Drehschalterfeld gekennzeichnete Speicherzelle wird das in dem 13stelligen Drehschalterfeld eingestellte Wort eingeschrieben und im 13stelligen Anzeigefeld angezeigt. Die Auslösung erfolgt durch die Taste <b>START</b> .
<b>AUSG SP</b>	Der Inhalt der durch das fünfstellige Drehschalterfeld gekennzeichneten Speicherzelle wird ins <b>SR</b> gelesen und in dem 13stelligen Anzeigefeld angezeigt. Die Auslösung erfolgt durch die Taste <b>START</b> .
<b>EING AR</b>	Das in dem 13stelligen Drehschalterfeld eingestellte Wort wird ins <b>AR</b> gegeben. Die Auslösung erfolgt durch die Taste <b>START</b> .
<b>EING BZR</b>	Die in dem fünfstelligen Drehschalterfeld eingestellte Adresse wird ins <b>BZR</b> gegeben. Die Auslösung erfolgt durch die Taste <b>START</b> .
<b>LÖSCH I</b>	Alle Register und Anzeigen der Anlage außer dem <b>BZR</b> werden gelöscht. <b>LÖSCH I</b> ist eine Zusammenfassung der Tasten <b>LÖSCH II</b> , <b>LÖSCH III</b> und <b>FA LÖSCH</b> .
<b>LÖSCH II</b>	Alle Register des Steuerwerks außer <b>IR 1...4</b> , die Register der Lochkarten- und Großspeichersteuereinheit sowie alle Anzeigen außer <b>E</b> , <b>F</b> , <b>K</b> , <b>L</b> , <b>DAN</b> und <b>ADD-UL</b> werden gelöscht.
<b>LÖSCH III</b>	Alle Register des Rechenwerks, <b>IR 1...3</b> und die Anzeige <b>ADD-UL</b> werden gelöscht.
<b>FA LÖSCH</b>	Alle Anzeigen werden gelöscht, die Anzeige <b>ADD-UL</b> jedoch nur dann, wenn die Taste <b>ÜBERL</b> nicht gedrückt ist.

Bei gedrückter Taste **PROGR** kann die Programmsteuerung durch die folgenden Tasten des Bedienungstisches beeinflusst werden:

<b>START</b>	Die Anlage beginnt das Programm mit dem durch das <b>BZR</b> gekennzeichneten Befehl.
<b>STOP</b>	Nach Ausführung des gerade bearbeiteten Befehls und nach Eingeben des neuen Befehlszählerstandes ins <b>BZR</b> wird der Programmablauf unterbrochen. Das Programm kann durch erneutes Betätigen der Taste <b>START</b> fortgesetzt werden.
<b>EINZEL</b>	Ist die Taste <b>EINZEL</b> gedrückt, so wird nach Betätigen der Taste <b>START</b> der Befehl ausgeführt, der durch den Befehlszählerstand gekennzeichnet ist. Anschließend wird ins <b>BZR</b> die Speichernummer des nächsten auszuführenden Befehls gegeben und der Programmablauf unterbrochen.
<b>DEZ ANZ</b>	Die Taste <b>DEZ ANZ</b> wirkt (abgesehen davon, daß sie die dezimale Anzeige einschaltet) wie die Tasten <b>EINZEL</b> und <b>STOP</b> .
<b>MARK SPRUNG</b>	Unabhängig von der Taste <b>MARK SPRUNG</b> (Markierungssprung) werden positive wie negative Befehle in gleicher Weise ausgeführt und nach Ausführung des Befehls das <b>BZR</b> in üblicher Weise auf den Stand eingestellt, der die Lage des nächsten Befehls angibt. Ist die Taste <b>MARK SPRUNG</b> gedrückt, so wird vor der Ausführung dieses nächsten Befehls geprüft, ob der vorige Befehl ein negatives Vorzeichen hatte. Ist dies der Fall, so wird <b>C(BZR)</b> in Zelle 00000 gespeichert und der nächste Befehl der Zelle 00001 entnommen. Ist gleichzeitig mit <b>MARK SPRUNG</b> die Taste <b>EINZEL</b> gedrückt, so bleibt die Anlage nach Ausführung eines negativen Befehls mit neuem <b>BZR</b> -Stand stehen. Anschließendes Drücken der Taste <b>START</b> löst den Markierungssprung aus, d. h. <b>C(BZR)</b> wird in Zelle 00000 gespeichert und das <b>BZR</b> auf 00001 gestellt. Der Befehl in Zelle 00001 wird nach obermaligem Betätigen der Taste <b>START</b> ausgeführt.

- UBERL: Ist die Taste UBERL (Oberlauf) gedrückt, so hält die Anlage bei einem Oberlauf des AR nicht an. Ist die Taste UBERL nicht gedrückt, so stoppt die Anlage bei AR-Oberläufen (Ausnahmen: ADU 093, SBU 183). In allen Fällen wird ein AR-Oberlauf durch die Anzeige ADD-UL angezeigt.
- HALT: Ist die Taste HALT gedrückt, so wird bei der Operation HLT nach Erhöhen des Befehlszählerstandes um 1 gestoppt; ist die Taste HLT nicht gedrückt, läuft das Programm weiter. Durch erneutes Betätigen der Taste START kann das Programm fortgesetzt werden.
- BZR STOP: Ist die Taste BZR STOP gedrückt, so wird der Befehl, der in der eingestellten Speicherzelle steht, vollständig ausgeführt und anschließend das BZR auf den Stand gebracht, der die Lage des nächsten Befehls angibt. Dann stoppt die Anlage mit der Anzeige BZR STOP. BZR STOP geht dem Markierungssprung vor, d. h. der Programmablauf wird vor Ausführung des Markierungssprunges unterbrochen, wenn die Speicherzelle, in der ein negativer Befehl steht, im Drehschalterfeld eingestellt ist.

Zur Kennzeichnung des Programmablaufs sind folgende Anzeigelampen vorhanden:

- EING: Die Lampe leuchtet während der Ausführung eines Eingabebefehls.
- AUSG: Die Lampe leuchtet während der Ausführung eines Ausgabebefehls.
- STOP: Die Lampe leuchtet nach Unterbrechung des Programmablaufs durch den Befehl STP. Durch erneutes Betätigen der Taste START wird die Anzeige gelöscht.
- HALT: Die Lampe leuchtet nach Unterbrechung des Programmablaufs durch den Befehl HLT (wenn die Taste HALT gedrückt ist). Durch erneutes Betätigen der Taste START wird die Anzeige gelöscht.
- EINZEL: Die Lampe leuchtet, wenn eine der Tasten EINZEL oder DEZ ANZ gedrückt ist.
- BZR STOP: Die Lampe leuchtet, wenn die Taste BZR STOP gedrückt und der im fünfstelligen Drehschalterfeld eingestellte Befehlszählerstand erreicht ist. Durch erneutes Betätigen der Taste START wird die Anzeige gelöscht.
- LS: Die Lampe leuchtet während der Ausführung eines Lochstreifenbefehls.
- LK: Die Lampe leuchtet während der Ausführung eines Lochkartenbefehls.
- MB: Die Lampe leuchtet während der Ausführung eines Magnetbandbefehls.
  
- OPERATION: Der Operationsteil eines Befehls ist nicht interpretierbar.
- KSP-ADR: Die Kernspeicheradresse liegt außerhalb des zulässigen Bereichs.
- INDEX: Die in der Index-Stelle des Befehls angegebene Zahl ist unzulässig.
- BLOCK-L: Die an den Trommelspeicher auszuliefernde Blocklänge liegt außerhalb des zulässigen Bereichs.
- TSP-ADR: Die Trommelspeicheradresse liegt außerhalb des zulässigen Bereichs.
- DAN: Die zweistellige Dezimalzahl  $C(AR)_{11, 12}$  ergibt bei der Ausgabeverschlüsselung kein zugelassenes Zeichen.
- ADD-UL: Bei der Ausführung der Befehle ADD, SUB, ADU, SBU, RND ist ein Überlauf des AR eingetreten. DVL, DVK
- CHAR-UL: Die Charakteristik des Resultats einer Gleitkomma-Operation ist  $> 99$ .
- DIV-UL: Der Quotient einer Festkomma-Division ist  $\geq 1$ , oder der Divisor einer Gleitkomma-Division ist  $= 0$ .

Nr.	Buchstaben- reihe	Ziffernreihe	Lochstreifen- Code					Verschlüsselung in der Datenverarbei- tungsanlage 2002	
			1	2	3	4	5	Z <sub>s</sub>	Z <sub>n</sub>
1	A	—	1	1	0	0	0	1	1
2	B	x (mal)	1	0	0	1	1	1	2
3	C	:	0	1	1	1	0	1	3
4	D	Wer da	1	0	0	1	0	1	4
5	E	3	1	0	0	0	0	0	3
6	F	[	1	0	1	1	0	1	5
7	G	]	0	1	0	1	1	1	6
8	H	10 (Basis 10)	0	0	1	0	1	1	7
9	I	8	0	1	1	0	0	0	8
10	J	;	1	1	0	1	0	1	8
11	K	(	1	1	1	1	0	1	9
12	L	)	0	1	0	0	1	2	0
13	M	. (Punkt)	0	0	1	1	1	2	3
14	N	, (Komma)	0	0	1	1	0	2	4
15	O	9	0	0	0	1	1	0	9
16	P	0	0	1	1	0	1	0	0
17	Q	1	1	1	1	0	1	0	1
18	R	4	0	1	0	1	0	0	4
19	S	' (Apostroph)	1	0	1	0	0	2	1
20	T	5	0	0	0	0	1	0	5
21	U	7	1	1	1	0	0	0	7
22	V	=	0	1	1	1	1	2	2
23	W	2	1	1	0	0	1	0	2
24	X	/	1	0	1	1	1	2	5
25	Y	6	1	0	1	0	1	0	6
26	Z	+	1	0	0	0	1	1	0
27	WR (Wagenrücklauf)		0	0	0	1	0	2	6
28	ZL (Zeilenvorschub)		0	1	0	0	0	2	7
29	Bu (Buchstaben)		1	1	1	1	1	2	8
30	Zi (Ziffern)		1	1	0	1	1	2	9
31	ZWR (Zwischenraum)		0	0	1	0	0	3	0
32 <sup>1)</sup>	#		0	0	0	0	0	3	1

<sup>1)</sup> Dieses Zeichen kann nur auf bestimmten Blattschreibermodellen ausgegeben werden.

### Verschlüsselung der Lochstreifenzeichen