

Interne Mitteilung für GR/P1 Nr.24/67

Typ: Vorläufige Beschreibung

Thema: Hardware-seitige Eigenschaften des TR440

Datum: 20.8.1967 (Neuauflage)

Autor(en): Dr. Danke / Rübin

Erläuterungen: Wird ergänzt und fortgesetzt.  
Ersetzt die Interne Mitteilung für ER/E44  
Nr.14/67 sowie die Ergänzung Nr. 1 zur  
Internen Mitteilung Nr.14/67.

Verteiler:  
siehe S.86

Inhalt:

Vorwort

1. Einige technische Angaben zum zentralen Rechner
2. Wortstruktur des TR440
  - 2.1. Wortarten
    - 2.1.1. Duale Gleitkommazahl
    - 2.1.2. Darstellbarer Zahlenbereich
  - 2.2. Duale Festkommazahl
    - 2.2.2. Darstellbarer Zahlenbereich
  - 2.3. Befehlsganzwort
  - 2.4. -Wort
3. Das Rechenwerk
  - 3.1. Die Register des Rechenwerkes
4. Das Befehlswerk
  - 4.1. Die Register des Befehlswerks
  - 4.2. Blockschaltbild des Befehlswerks
  - 4.3. Arbeitsweise der Register des Befehlswerks
    - 4.3.1. Adressenregister BA und Coderegister BC mit ihren Zusatzregistern BZ und BD
    - 4.3.2. Bereitadressenregister BB und Adressenhilfsregister BH
    - 4.3.3. Befehls-Folgeadressen-Register BF (Befehlszähler)
      - 4.3.3.1. Schleifenzähler BTZ1 und Taktzähler BTZ2
    - 4.3.4. Leitadressen-Register BL und Zusatzleitadressen-Register BLZ1 und BLZ2
    - 4.3.5. Die vier Seitenadreßregister BP1 bis BP4
    - 4.3.6. Die Indexregister BI1 bis BI4
    - 4.3.7. Indexbasisadressenregister BXB und Indexbasis-adressenzusatzregister BXBZ
    - 4.3.8. Das Merklichter- (und Wahlschalter) register BK
    - 4.3.9. Unterprogrammregister BU (Unterprogrammordnungs-zähler) und Unterprogramm-Hilfsregister BV
    - 4.3.10. Uhrregister BG
    - 4.3.11. Weckerregister BW
    - 4.3.12. Prüfreger BT und Vergleichsregister BTV

- 5. Die Adressierungsmodi und die Adreßrechnungsmodi
  - 5.1. Absoluter Adressierungsmodus, kurz Systemmodus
  - 5.2. Spezieller Adressierungsmodus, kurz Spezialmodus
  - 5.3. Relativer Adressierungsmodus, kurz Normalmodus
  - 5.4. Abwicklermodus
  - 5.5. Adreßrechnungsmodus
  - 5.6. Wartungsmodus
  
- 6. Speicherorganisation und Speicheransteuerung
  - 6.1. Das Speicherwerk
  - 6.2. Speicherverschränkung, Kacheln
  - 6.3. Arbeitsdefinitionen
    - 6.3.1. Seiten und Kacheln
  - 6.4. Hardwareseitige Belegung der Speicherkachel 0
    - 6.4.1. Speicherkachel 0
    - 6.4.2. Bedeutung und Stellung der Steuerbits
  - 6.5. Speicherzuteilung
    - 6.5.1. Aufteilung und Belegung eines Leitblocks
    - 6.5.2. Adressierung des Leitblocks
  - 6.6. Speicheransteuerung
  - 6.7. Die Seitenadreßregister
  - 6.8. Indexspeicheransteuerung
  - 6.9. Ansteuerung der Indexzellen und Indexregister
  
- 7. Wirkung von Unterbrechungen
  - 7.1. Eingriffe
    - 7.1.1. Definition
    - 7.1.2. Eingriffsarten
    - 7.1.3. Das Eingriffswort
    - 7.1.4. Hardwareseitige Vorgänge bei Eintreffen eines Eingriffs
    - 7.1.5. Unter Eingriffssperre eintreffende Eingriffe
    - 7.1.6. Die Eingriffssperre

- 7.2. Alarme
  - 7.2.1. Definition
  - 7.2.2. Alarmgruppen und Alarmarten
  - 7.2.3. Alarmsperren
  - 7.2.4. Hardware-seitige Abspeicherungen, Modusumschaltungen und Sprungziele bei Alarmen
- 7.3. Unterbrechbarkeit der Befehle
  - 7.3.1. Vereinfachter Ablauf der Abrufphase
  - 7.3.2. Unterbrechungsstellen im Befehlsablauf
  - 7.3.3. Unterbrechungsmöglichkeiten bei komplexen Befehlen
- 8. Hardware-seitige Bedingungen bei Makros und SSR-Befehlen
  - 8.1. Leercode (Makros)
  - 8.2. An den Interncode geknüpfte Bedingungen
  - 8.3. Liste der Leercodes
  - 8.4. Hardware-seitige Vorgänge bei Erkennen eines Leercodes (Makro)
  - 8.5. Hardware-seitige Vorgänge bei Erkennen eines SSR-Befehls

## Vorwort

In der vorliegenden Beschreibung werden die Eigenschaften des Rechners TR440 beschrieben, die durch seine Konstruktion und Verdrahtung unveränderlich vorgegeben sind. Es wird lediglich der grobe organisatorische Aufbau aus den verschiedenen Werken beschrieben, ihre gegenseitige Abhängigkeit und ihr Zusammenspiel ohne Rücksicht auf einen speziellen Ausbau von Speicher oder Peripherie. Eigenschaften, die der Rechner erst durch ein spezielles Betriebs-System erhält, sind unberücksichtigt. Manche hardwaremäßigen Eigenschaften sind jedoch im Hinblick auf spezielle softwaremäßige Vorstellungen entstanden und im Zusammenhang mit diesen zu sehen.

### 1. Einige technische Angaben zum zentralen Rechner

Der TR440 verarbeitet die Information wortweise parallel bei einer Wortlänge von 52 Binärstellen, von denen vier Stellen als Prüfstellen für Dreierprobenprüfung und Typenkenennung zur Kennzeichnung der Wortart dienen. Er ist eine Ein-Adress-Anlage mit etwa 233 fest verdrahteten Befehlen. Es werden verschiedene Befehlsformate innerhalb einer Gesamtlänge von 24 Bit verwendet. Zur Adressierung sind vier Adressierungsmodi möglich.

Die Rechengenauigkeit umfaßt bei Rechnung mit dualen Festkommazahlen etwa 13 Dezimalen, bei dualer Gleitkommarechnung etwa 11 Dezimalen.

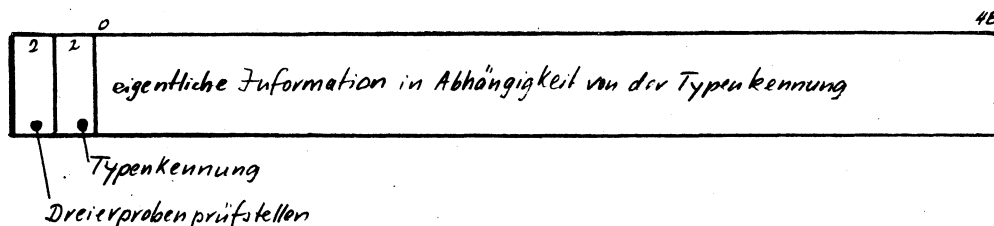
Der Ferritkernspeicher ist ausbaufähig von  $65\,536 = 64K$  Ganzwort-Zellen in drei Stufen bis zu  $262\,144 = 256K$  Zellen, die halbzellenweise adressierbar sind.

Die Anlage kann mit ein bis vier Rechnerkernen ausgestattet werden, die unabhängig voneinander und gleichzeitig mit dem Kernspeicher und mit den übrigen Werken arbeiten können.

Es sind bis zu 16 EA-Kanäle einrichtbar mit maximal 4 Peripheriegeräten je Normalkanal (800 000 Bytes/sek) oder sinnvollerweise nur einem Peripheriegerät je Schnellkanal (5 000 000 Bytes/sek).

## 2. Wortstruktur des TR440

Die wesentliche Einheit der Informationsverarbeitung ist das Ganzwort. Das Ganzwort ist 52 Bit lang, von denen die ersten beiden als Dreierprobenprüfstellen (Quersummenprüfung modulo 3), die beiden folgenden zur Kennzeichnung des Worttyps verwendet werden (Typenkennung). Es bleiben also 48 Bit für die eigentliche Information.



Die durch die Typenkennung 0 und 1 gekennzeichneten Worttypen werden gemeinsam als Zahlwörter interpretiert. Jedes Zahlwort im Speicher erhält eine Binärstelle für die Marke, die durch das Programm eingesetzt und abgefragt werden kann. Wird ein markiertes Wort in ein Register übernommen, so wird die Marke abgespalten und in das Markenregister RM übernommen. Die Markenstelle im Register wird vorzeichengleich gesetzt und dient als Überlauf-Auffangstelle.

Die im Speicher, im Rechenwerk und im Bereitadressen-Register auftretenden negativen Zahlwörter werden im  $(B-1)$ -Komplement dargestellt: in der Vorzeichen- und in den Ziffernstellen ist die binäre Null (0) mit der binären Eins (1) vertauscht und umgekehrt. Die Null besitzt daher die Darstellungen +0 d. i. 0000...00 und -0 d. i. 1111...11. Da dadurch eine negative Zahl in ihrer Komplementdarstellung z. B. als Ganzwort  $< 1$  um  $(1 - 2^{-46})$  zu groß ist, wird eine Summe aus zwei negativen Zahlen oder eine positive Differenz durch Einerrücklauf von der vordersten (linken) Stelle zur Einerstelle um  $(1 - 2^{-46})$  verringert. Der Einerrücklauf bewirkt also, daß bei L-Überlauf in der höchsten Vorzeichenstelle ein 1 zur Einerstelle addiert wird.

## 2.1 Wortarten

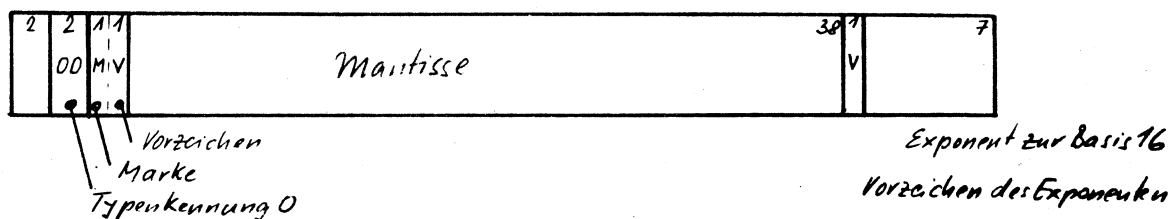
### 2.1.1 Duale Gleitkommazahl

Diese Zahlendarstellung eignet sich vornehmlich für technisch wissenschaftliche Berechnungen, da sie es ermöglicht, mit sehr verschieden großen Zahlen zu rechnen, wobei die Ergebnisse unmittelbar normalisiert werden (s. Abs. 2.1.2). Durch die Normalisierung werden die Resultate so verschoben, daß Überläufe beseitigt und die vorhandenen Binärstellen möglichst vollständig ausgenutzt werden.

Die Darstellung einer Gleitkommazahl  $x$  erfolgt in der Form

$$x = a \cdot 16^b$$

mit  $a \leq 0$  als 38-stelliger Mantisse und  $b \leq 0$  als 7-stelligem Exponenten.



### 2.1.2 Darstellbarer Zahlenbereich

Die 38 Binärstellen der Mantisse haben von links nach rechts die Wertigkeiten  $2^{-1}$ ,  $2^{-2}$ , ...,  $2^{-38}$ . Es sind also Vielfache von  $|2^{-38}|$  bis zum Wert  $|1 - 2^{-38}|$  darstellbar. Der kleinste Betrag von  $a$  ist bei normalisierten Gleitkommazahlen  $1/16 = 2^{-4}$ , da bei diesen mindestens eine der vier vordersten Binärstellen der Mantisse von ihren Vorzeichenstellen verschieden sein muß.

Die Binärstellen des Exponenten haben von links nach rechts die Wertigkeiten  $2^6$ ,  $2^5$ , ...,  $2^0$ . Der Exponent überstreicht also den Bereich von  $-127 \leq b \leq +127$ .



Damit liegen normalisierte Gleitkommazahlen  $x$  stets im Bereich

$$2^{-512} \leq |x| \leq 2^{508} \cdot 2^{470}$$

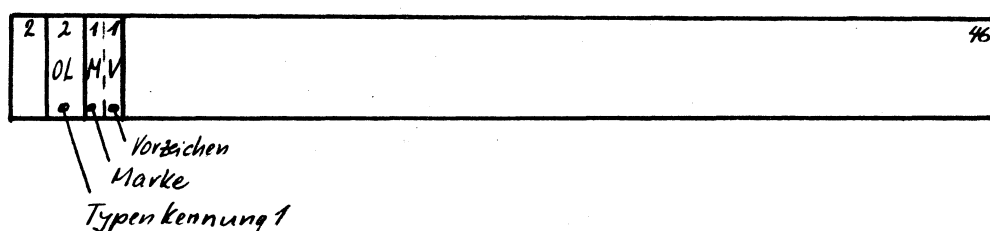
entsprechend

$$7,458\,341 \cdot 10^{-155} \leq |x| \leq 8,379\,880 \cdot 10^{152} \quad (\text{gerundet})$$

Die Gleitkomma-Arithmetik setzt normalisierte Operanden voraus und liefert i. a. auch wieder normalisierte, in Abhängigkeit von der höchsten unwesentlichen Stelle gerundete Ergebnisse. Entsteht bei der Rechnung ein Exponent  $b > 127$ , so wird die richtige Mantisse, aber ein Exponent  $b-255$  gebildet und Arithmetischer Alarm gemeldet. Eine Überschreitung des Exponentenbereiches nach unten wird nicht vollzogen.

### 2.2.1 Duale Festkommazahl

Diese Zahlendarstellung eignet sich besonders für die Datenverarbeitung, da hier die Größenordnungen der Eingangsdaten, Zwischenergebnisse und Resultate leicht zu übersehen sind. Es steht im Belieben des Programmierers, die Lage des Kommas festzusetzen. Bei Vergleichen ist jedoch auf gleiche Komma-stellung in den zu vergleichenden Größen zu achten. Gegebenen-falls ist vorher durch Schiften die Stellungsgleichheit her-beizuführen.



### 2.2.2 Darstellbarer Zahlenbereich

- a) Das Komma steht linksbündig vor der ersten (linken) Binär-stelle. Dann sind Zahlen  $z$  im Bereich

$$-(1 - 2^{-46}) \leq z \leq (1 - 2^{-46})$$

entsprechend

$$-(1 - 1,421086 \cdot 10^{-14}) \leq z \leq 1 - 1,421086 \cdot 10^{-14}$$

darstellbar.

- b) Steht das Komma rechtsbündig hinter der letzten Binärstelle, dann wird ein Bereich von

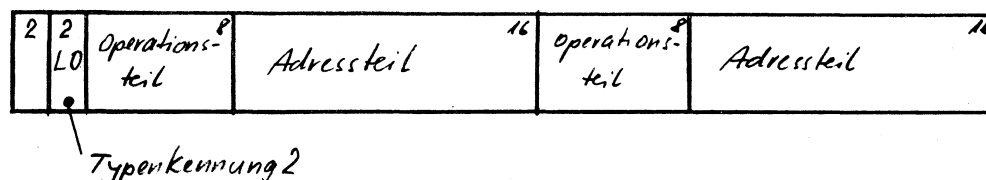
$$-(2^{46} - 1) \leq z \leq (2^{46} - 1)$$

überstrichen, entsprechend

$$-70\,368\,744\,177\,663 \leq z \leq 70\,368\,744\,177\,663 \quad .$$

### 2.3 Befehlsganzwort

Das Befehlsganzwort enthält zwei Maschinenbefehle mit je einem 8 Binärstellen umfassenden Operationsteil (Code) und einem 16 Binärstellen langen Adreßteil.

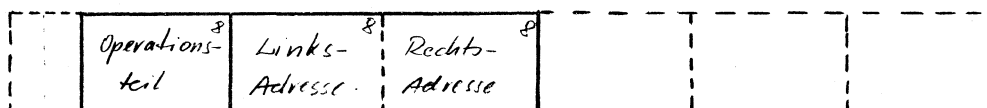


Da nicht alle der möglichen 256 Befehls-codes durch Maschinen-befehle belegt sind, verbleiben einige Codes für softwaremäßig festzulegende Befehle (Makros, s. Abs. 8.1).

Der 16-stellige Adressteil ermöglicht die unmittelbare Adressierung von 65 536 Halbwortzellen entsprechend einer Großseite. Zur Adressierung des Speichers und zu den verschiedenen Adressierungsmodi vgl. Abs. 5.

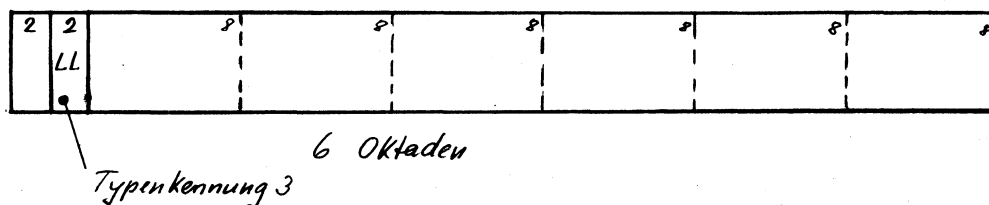
Aus dem Speicher wird stets ein Befehls-Ganzwort ausgelesen und zunächst der Befehl mit der geraden Adresse (linkes Halbwort) entschlüsselt und ausgeführt, dann der mit der ungeraden Adresse. Bei Sprung auf ein Befehlswort mit ungerader Adresse wird nach dem Auslesen des Ganzwortes unmittelbar der Befehl mit der ungeraden Adresse entschlüsselt.

Der Adreßteil eines Befehls kann (z. B. bei der Adressierung von Indexzellen, zur Angabe von Parametern und bei Doppelcode-befehlen) in eine Linksadresse und eine Rechtsadresse von je 8 Binärstellen unterteilt werden.



## 2.4 α-Wort

Ein α-Wort enthält alphanumerische Zeichen und Ziffern in einer Folge von 6 Oktaden.



### 3. Das Rechenwerk

Das Rechenwerk dient der Ausführung der geforderten Rechenoperationen. Es arbeitet bei reinen Rechenoperationen unabhängig von den übrigen Werken (Befehlswerk, EA-Werk, Speicherwerk). Über das Rechenwerks-Leitwerk verkehrt es mit dem Sammelregister, das auch dem Befehlswerk angehört.

#### 3.1 Die Register des Rechenwerkes

Bezeichnung		Anzahl Binärst.	Name des Registers
	Kurzbez.		
R	A	52	Akkumulator
R	B	52	Akkumulator-Hilfsregister
R	C	52	Sicherstellungsregister für RB
R	Q	52	(Multiplikator-)Quotienten-Register
R	P	52	(Multiplikator-)Quotienten-Hilfsregister
R	U	52	Übertragsregister
R	V	48	Übertrags-Hilfsregister
R	D	52	Multiplikanden-Register
R	H	52	Hilfsregister
R	Y	8	Schifftzähler
R	Z	8	Hilfsregister für RY
R	X	8	Exponentenzähler
R	W	8	Hilfsregister für RX
R	M	1	Markenregister
R	S	52	Sammelregister
R	T	52	Sicherstellungsregister für RS

  = dem Programmierer durch Maschinenbefehle zugänglich.

Die adressierbaren Register nehmen mit Ausnahme des Schifftzählers und des Markenregisters ein Ganzwort mit den Prüfstellen für Typenkennung und Dreierprobe auf, umfassen also 52 Binärstellen.

Die Kurzbezeichnungen der Register werden nur dort verwendet, wo eine Verwechslung mit Registern anderer Werke ausgeschlossen ist.

Der Akkumulator RA mit Akkumulator-Hilfsregister RB und Sicherstellungsregister RC

Der Akkumulator ist das Register, in dem die meisten Rechenoperationen ausgeführt werden. Es enthält vor Ausführung eines Rechenwerksbefehls in der Regel den Operanden, nach Ausführung das Resultat. Mit Hilfe seines Hilfsregisters kann es vielfältige Transporte ausführen. Für interne Vorgänge benötigt das Akkumulator-Hilfsregister noch das Sicherstellungsregister.

(Multiplikator-)Quotienten-Register RQ und sein Hilfsregister RP

Dieses Register enthält bei Multiplikationen und Divisionen das Resultat oder einen Teil davon. Es kann mit dem Akkumulator zu einem doppelt langen Register verknüpft werden. Sein Hilfsregister RP ist für interne Abläufe erforderlich.

Multiplikanden-Register RD

Dieses Register nimmt in der Regel den zweiten Operanden nach der Ausführung einer Rechenwerksoperation auf.

Übertragsregister RU und sein Hilfsregister RV

Diese Register nehmen Überträge für Operationen im RA, RB und RD auf. Für interne Vorgänge ist wieder ein Hilfsregister RV erforderlich.

#### Hilfsregister RH

Dieses Register dient als zugriffszeitfreie Speicherzelle. Es nimmt Vergleichsgrößen für Vergleiche mit dem Inhalt des Akkumulators auf und ermöglicht das Aufsummieren der Ergebnisse mehrerer Multiplikationsoperationen (akkumulierende Multiplikation).

#### Schifftzähler RY und Hilfsregister RZ

Dieses 8 Binärstellen umfassende Register zählt die Schifftschritte beim Schifften, bei Gleitkommaoperationen und Normalisierungsbefehlen. Für rechnerinterne Vorgänge wird ein Hilfsregister RZ benötigt.

#### Exponentenzähler RX und Hilfsregister RW

Die Errechnung des Exponenten (zur Basis 16) bei Gleitkommaoperationen wird im 8 Binärstellen langen Exponentenzähler vorgenommen. Auch hier ist ein Hilfsregister erforderlich.

#### Markenregister RM

Wird ein Zahlwort aus dem Speicher in ein Register übernommen, so wird, falls das Zahlwort markiert war, die Markenstelle abgespalten und die Marke in das Markenregister (1 Binärstelle) geschrieben, da die Markenstelle des Zahlwortes im Register als Überlauf-Auffangstelle verwendet wird. Sie wird deshalb beim Einspeichern vorzeichengleich gesetzt.

#### Sammelregister RS und sein Sicherstellungsregister RT

Dieses 52 Binärstellen umfassende Register ist sowohl vom Leitwerk des Befehlswerks aus zugänglich, als auch vom Leitwerk des Rechenwerkes.

Jeder Transport vom oder in den Speicher erfolgt über das  
Sammelregister. Der Operand eines Rechenwerksbefehls wird  
bereits während der Entschlüsselung des Befehls im Befehls-  
werk im Sammelregister bereitgestellt.



#### 4. Das Befehlswerk

Das Befehlswerk arbeitet mit dem ihm zugeordneten Leitwerk unabhängig von den übrigen Werken (Rechenwerk, Speicherwerk, Ein- und Ausgabewerk). Innerhalb des Befehlswerkes wird mit relativen Adressen gearbeitet. Erst bei einer Speicheransteuerung werden relative Adressen über die Seitenadressregister durch absolute Adressen ersetzt, wenn nicht Systemmodus eingeschaltet ist oder ein Sprung im oder in den Spezialmodus erfolgt.

Transporte in das Befehlswerk gehen über das Sammelregister, welches sowohl dem Befehls- als auch dem Rechenwerk zugeordnet ist.

#### 4.1 Die Register des Befehlswerks

Bezeichnung	Anzahl		Name des Registers
	Kurzbez.	Binärst.	
B	A	24	Adressenregister
B	Z	24	Zusatzadressenregister
B	C	8	Coderegister
B	D	8	Zusatzcoderegister
B	B	24	Bereitadressenregister
B	H	24	Adressenhilfsregister für BB
B	J	24	Sicherstellungsregister für BH
B	F	24	Befehls-Folge-Register (Befehlszähler)
B	L	16	Leitadressenregister
B	LZ1	5	Zusatzleitadressenregister 1 } $\Delta$ -Register
B	LZ2	5	
B	P1	24	Seitenadressenregister 1
B	P2	24	Seitenadressenregister 2
B	P3	24	Seitenadressenregister 3
B	P4	24	Seitenadressenregister 4
B	XB	22	Indexbasisadressenregister
B	XBZ	11	Indexbasisadressenzusatzregister
B	I1	34	Indexregister 1
B	I2	34	Indexregister 2
B	I3	34	Indexregister 3
B	I4	34	Indexregister 4
B	K +)	8	Merklichterregister
B	K +)	8	Wahlschalterregister
B	U	8	Unterprogrammregister (Unterprogr.-Ordnungsz.)
B	V	8	Hilfsregister für Unterprogrammregister
B	W	16	Wecker
B	G	24	Uhr
B	T	24	Prüfregister
B	TZ1	6	Schleifenzähler
B	TZ2	6	Taktzähler
B	TV	12	Vergleichsregister
R	S	52	Sammelregister(Befehls- als auch Rechenwerk zugeordnet)

  = für Programmierer zugänglich

  = nur für Systemprogrammierer zugänglich

<sup>+</sup>) Anmerkung: Das Register BK enthält in 16 Binärstellen die Merklichter und Wahlschalter.

#### 4.2 Blockschaltbild des Befehlswerks



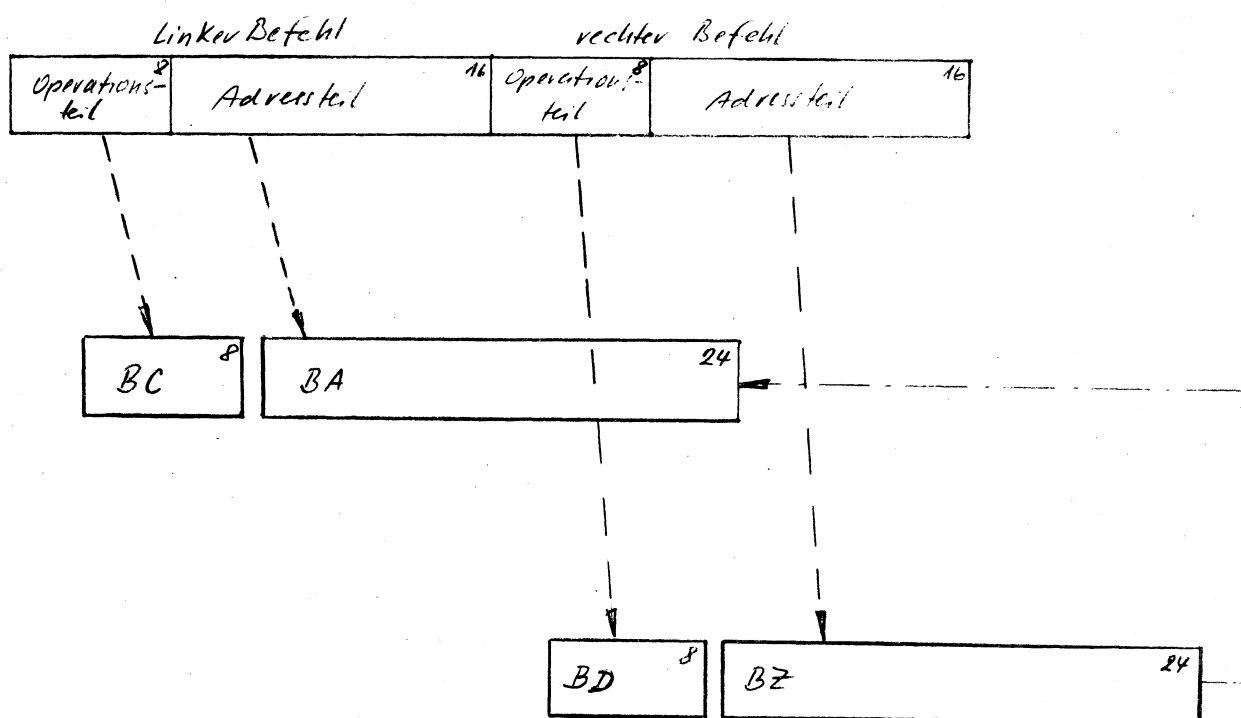
BA	Adressenregister
BZ	Zusatzadressenregister
BC	Coderegister
BD	Zusatzcoderegister
BB	Bereitadressenregister
BH	Adressenhilfsregister
BF	Befehlszähler
BL	Leitadressenregister
BLZ 1	Zusatzleitadressen reg. 1
BLZ 2	Zusatz leit adressen reg. 2
BP1-BP4	Seitenadressregister
BXB	Indexbasisregister
BXEZ	Indexbasiszusatzregister
BI1-BI4	Indexregister
BK	Merklichterregister
BU	Unterprogrammregister
BV	Unterprogrammhilfs reg.
BLV	Wecker
BG	Uhr
BTZ 1	Schleifenzähler
BTZ 2	Taktzähler

### 4.3 Arbeitsweise der Register des Befehlswerks

#### 4.3.1 Adressenregister BA und Coderegister BC mit ihren Zusatzregistern BZ und BD

Code- und Adressenregister dienen zur Übernahme von Operations- und Adreßteil eines Befehls, die entsprechenden Zusatzregister übernehmen Operations- und Adreßteil des Halbwortes mit der ungeraden Adresse, da stets ein Ganzwort aus dem Speicher ins Befehlswerk ausgelesen wird. Die Adressen werden dabei auf 24-Bit-Adressen verlängert durch die Größen mod1 und mod2, siehe "Wirkung der Befehle".

Zur Ausführung des rechten Befehls des Ganzwortes (ungerade Adresse) werden die Inhalte von Adressen- und Coderegister durch die der entsprechenden Zusatzregister überschrieben.



#### 4.3.2 Bereitadressenregister BB und Adressenhilfsregister BH

Das Bereitadressenregister umfaßt 24 Binärstellen, davon eine Vorzeichenstelle. Es nimmt z. B. die Indexgrößen auf, mit denen im Adressenregister BA Adreßrechnungen ausgeführt werden. Für die Adressenrechnung können alle 24 Stellen einschließlich der Vorzeichenstelle in Anspruch genommen werden. Der Einer-rücklauf erfolgt im mod24 von der 1. zur 24. Stelle, im mod16 von der 9. zur 24. Stelle. Zur Speicheransteuerung werden lediglich die letzten 22 Binärstellen genommen. Das Adressen-hilfsregister BH dient dem internen Ablauf.

#### 4.3.3 Befehls-Folge-Register BF (Befehlszähler)

Das Befehls-Folge-Register umfaßt 24 Binärstellen und enthält nach Ablauf der Abrufphase eines Befehls die Adresse des nächsten abzuhandelnden Befehlswortes. Beim Durchlaufen der ersten Abrufphase eines Befehls wird das Befehls-Folge-Register um 1 erhöht, auch bei Sprungbefehlen. Nach Ausführung des Befehls mit der ungeraden Adresse wird das nächste Befehlsganzwort in die Register BC, BA, BD und BZ übernommen.

Um innerhalb einer Großseite ohne Modifizierung des Befehls springen zu können, werden nur die rechten 16 Bit des Befehls-Folge-Registers verändert, die vorderen Binärstellen des Befehls-Folge-Registers adressieren die Großseite und bleiben erhalten. Nur durch die Befehle SE, SUE und VMO bei entsprechender Spezifikation sowie bei MABI und MU, wenn der Zweitcode ein Sprungbefehl ist, werden auch die linken 8 Binärstellen aus dem Adressenregister in das Befehls-Folge-Register übernommen und gestatten damit Sprünge in andere Großseiten.

Da ein zusammenhängendes Programm ohne die Daten nur selten den Adressenraum einer Großseite (32 K) überschreiten wird, werden in der Regel auch nur die rechten 16 Binärstellen des Befehls-Folge-Registers verändert.

Nach der Ausführungsphase eines Sprungbefehls enthält das Befehls-Folge-Register die Adresse, an der fortgefahren werden soll. Bei Sprung in ein Unterprogramm mit dem Befehl SU wird die dem Sprungbefehl folgende Adresse in einer Indexzelle festgehalten, beim Befehl SFB im Bereitadressenregister. Ein Sprung auf ein Wort mit  $TK \neq 2$  wird nicht ausgeführt, das Befehls-Folge-Register wird also nicht erhöht.

#### 4.3.3.1 Schleifenzähler BTZ1 und Taktzähler BTZ2

Diese beiden je 6 Binärstellen umfassenden Register sind dem Befehls-Folge-Register zugeordnet und übernehmen gewisse Aufgaben bei Prüf- und Wartungsläufen.



#### 4.3.4 Leitadressen-Register BL und Zusatz- leitadressen-Register BLZ1 und BLZ2

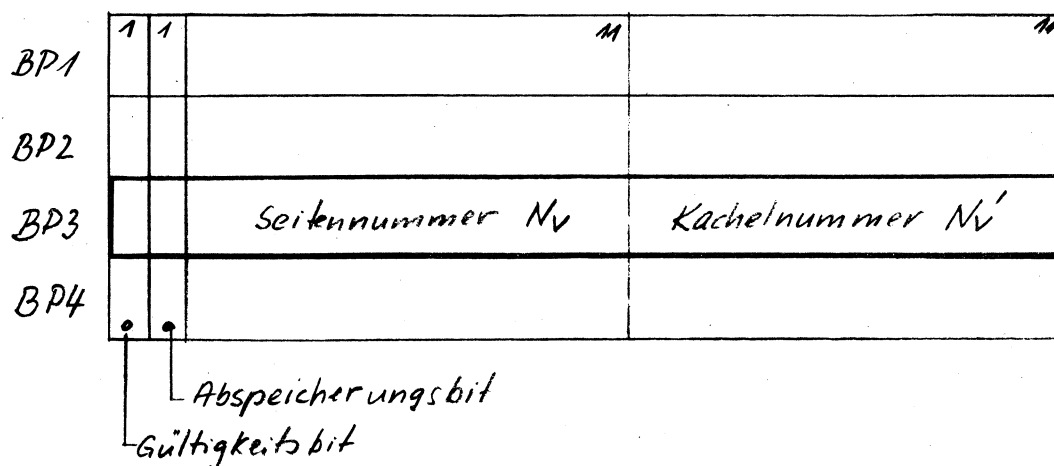
Das 16 Binärstellen umfassende Leitadressen-Register dient mit den beiden ihm zugeordneten Zusatzleitadressen-Registern von je 5 Binärstellen Länge der Adressierung der Zellen des Leitblocks und der Speicheransteuerung im Normal-, Abwickler- und Spezialmodus.

Ihre Funktion ist im Abschnitt Speicheransteuerung ausführlich erläutert.

#### 4.3.5 Die vier Seitenadreßregister BP1 bis BP4

Die vier Seitenadreßregister sind je 24 Binärstellen lang. Sie enthalten in den Stellen 3 - 13 die linken 11 Bit von relativen ~~Anfangs~~<sup>Seiten</sup>adressen ~~von Ganzseiten~~ (Seitennummern  $N_v$ ) eines nicht im Systemmodus arbeitenden Programmes, in den Stellen 14 - 24 die linken 11 Bit der Adresse der Kachel (Kachelnummer  $N'_v$ ), die der links stehenden Seite zugewiesen ist. Die erste Stelle im Register zeigt an, ob der Inhalt der Zelle gültig ist (Gültigkeitsbit), die zweite zeigt an, ob die Information in der Kachel verändert wurde (Abspeicherungsbit).

Beim Übertragen der Kachelnummer aus dem Viertelwort des Leitblocks wird das Abspeicherbit mit übernommen. Beim ersten Einschreiben der Kachelnummer in den Leitblock ist es in 0 gesetzt. Beim ersten Schreiben-Zugriff auf einen Platz in der Kachel wird das Gültigkeitsbit gelöscht und Speicherschutzalarm ausgelöst. Ist die Seite zum Schreiben freigegeben, wird bei der nächsten Speicheransteuerung die Kachelnummer mit dem inzwischen auf 1 gesetzten Abspeicherbit übernommen. Bei Neueinstellung der Bereichsgrenzen  $\Delta_1$  und  $\Delta_2$  und bei Neueinstellen der Leitadresse werden die Gültigkeitsbits gelöscht.



Die Arbeitsweise der Seitenadreßregister ist im Abschnitt Speicheransteuerung näher erläutert.

Den Seitenadreßregistern ist ein Alterungsmechanismus zugeordnet, der aus 6 Flipflop gebildet wird. Er ordnet die Register in der zeitlichen Reihenfolge, in der sie zuletzt benutzt wurden und bezeichnet damit das zuletzt benutzte Register als das jüngste, das am längsten nicht angesprochene als das älteste.

Bei Ungültigsetzen eines Seitenadreßregisters wird es zum ältesten.

#### 4.3.6 Die Indexregister BI1 bis BI4

Die vier Indexregister mit je 34 Binärstellen bilden einen primären Indexspeicher. In den Binärstellen 3 - 10 steht jeweils eine der 256 Indexadressen und in den folgenden 24 Stellen der Inhalt dieser Indexzellen, die Indexgröße. Die erste Stelle in den Registern zeigt an, ob der Inhalt des Indexregisters gültig ist (Gültigkeitsbit). Die zweite Stelle enthält das Rückspeicherungsbit. Es wird gesetzt, wenn die Indexgröße verändert wird und zeigt damit an, daß beim Aufgeben dieses Indexregisters oder vor Überschreiben die Indexgröße in die zugehörige Speicherzelle zurückzuspeichern ist.

BI1	1	1	8	24
BI2				
BI3			Indexadresse	Indexgröße
BI4				

— Rückspeicherungsbit  
 — Gültigkeitsbit

Den vier Indexregistern ist ein Alterungsmechanismus wie bei den Seitenadreßregistern zugeordnet. Bei Ungültigsetzen eines Indexregisters wird es zum ältesten. \*)

Das Arbeiten mit den Indexregistern und das Zusammenspiel mit dem Indexbasisregister ist in Abschnitt "Indexspeicheransteuerung" beschrieben.

\*) Bei Deklarieren einer neuen Indexbasis werden die gültigen Indexregister zurückgespeichert.

#### 4.3.7 Indexbasisregister BXB und Indexbasiszusatzregister BXBZ

Durch die Befehle BCI  $n$  und ZI  $n_2$  wird an beliebiger Stelle  $n$  im Hauptspeicher ein Indexzellenbereich von 256 Halbworten deklariert. Die absolute Anfangsadresse dieses Indexzellenbereichs, die Indexbasisadresse, wird im Indexbasisregister festgehalten, das 22 Binärstellen umfaßt. Die 256 Indexzellen (Halbwortzellen) werden relativ zur Indexbasisadresse gezählt.

Bei Überschreiten einer Kachelgrenze durch den Indexzellenbereich wird die Adresse der Kachel, die die Fortsetzung des Indexzellenbereichs enthält, im Indexbasiszusatzregister festgehalten.

Das Zusammenspiel von Indexbasisregister und Indexregistern ist im Abschnitt "Indexspeicheransteuerung" beschrieben.

#### 4.3.8 Das Merklichter- (und Wahlschalter)-register BK

Das 8 Binärstellen umfassende Merklichterregister dient der Aufnahme von Steuerzeichen (Merklichtern), die während eines Programmlaufes erzeugt und für spätere Abfragen aufbewahrt werden.

Das ebenfalls 8 Binärstellen lange Wahlschalterregister dient nur Wartungs- und Testläufen sowie für TR4-Programme. Es befindet sich am Bedienpult und nimmt von außen manuell eingestellte Kriterien auf.

#### 4.3.9 Unterprogrammregister BU (Unterprogrammordnungszähler) und Unterprogramm-Hilfsregister BV

Das 8 Binärstellen umfassende Unterprogrammregister nimmt bei einem Sprung mit dem Befehl SU in ein Unterprogramm die Adresse derjenigen Indexzelle auf (Adresse relativ zur Indexbasisadresse), in die die auf den Sprungbefehl folgende Adresse für den Rücksprung abgelegt wird.

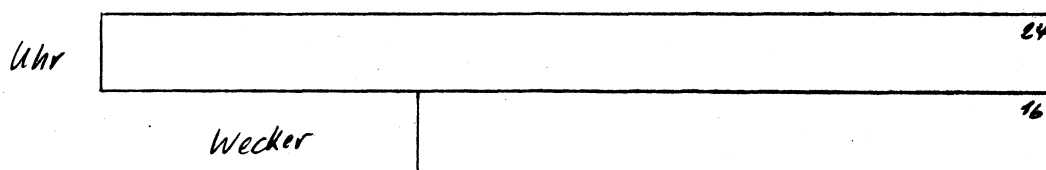
Für rechnerinterne Vorgänge ist dem Unterprogrammregister ein Hilfsregister gleicher Stellenzahl zugeordnet.

#### 4.3.10 Uhrregister BG

Das Uhrregister ist ein 24-stelliges Zählregister für die Zählung relativer Zeitabschnitte. Es zählt aufwärts in Zähleinheiten von  $10^{-5}$  us und überzählt folglich alle  $2^{24} \cdot 10^{-5}$  sec, also alle 2,7962 min.

#### 4.3.11 Weckerregister BW

Das 16 Binärstellen umfassende Weckerregister gestattet die Voreinstellung eines bestimmten Zeitpunktes, zu dem ein Weckeralarm gemeldet werden soll. Der Alarm tritt ein, wenn die rechten 16 Stellen des Uhrregisters mit dem Inhalt des Weckerregisters übereinstimmen. Der Weckeralarm tritt mindestens alle 0,65536 sec auf.



#### 4.3.12 Prüfregister BT und Vergleichsregister BTV

Das Prüfregister (24 Binärstellen) und das Vergleichsregister (12 Binärstellen) dienen zur Durchführung von Prüf- und Wartungsläufen in der Wartungsvariante. Es können Befehlsfolgeregister BF, Schleifenzähler BTZ1 und Taktzähler BTZ2 mit dem Inhalt dieser beiden Register verglichen werden (siehe Abs. 5.6).



## 5. Die Adressierungsmodi und die Adreßrechnungsmodi

Um verschiedene Stufen der Programmierung zu ermöglichen, werden mehrere Adressierungsmodi verwendet, zwischen denen durch bestimmte Befehle und bei Unterbrechung durch Eingriffe oder Alarme umgeschaltet wird. Es sind dies der Systemmodus, der Spezialmodus und der Normalmodus. Sie werden durch die Zustände der Flipflop BEBY und BEBN dargestellt. Durch den Zustand eines zusätzlichen Flipflop BEBO, des Abwicklerflipflop, können die Adressierungsmodi erweitert werden.

### 5.1 Absoluter Adressierungsmodus, kurz Systemmodus

Im Systemmodus werden die Zellen des Speichers durch absolute 22-Bit-Adressen adressiert. Damit ist maximal ein Speicher-  
raum von 2048 K adressierbar. Alle den Speicher ansprechenden Befehle haben absolute Adressen, und das Befehlsfolgeregister schaltet in absoluten Adressen fort. Bei Unterbrechung im Systemmodus werden die Register und die Steuerflipflop in der Speicher-  
kachel 0 abgelegt.

Der Systemmodus wird durch die Flipflopzustände BEBY,  $\overline{\text{BEBN}}$  angezeigt. Die Stellung des Abwickler-Flipflop hat nur beim Befehl VPU Bedeutung.

### 5.2 Spezieller Adressierungsmodus, kurz Spezialmodus

Im Spezialmodus adressieren alle den Speicher ansprechenden Befehle relativ zu einer Anfangsadresse 0. Die Seitennummer wird über einen Leitblock ersetzt. Das Befehlsfolgeregister schaltet jedoch in absoluten Adressen fort. Bei Unterbrechung im Spezialmodus werden die Befehlswerk-Register und die Steuer-Flipflop im Leitblock abgelegt.

Der Modus wird durch die Flipflopzustände BEBY, BEBN dargestellt. Die Stellung des Abwicklerflipflop wirkt sich nur auf den Befehl VPU aus.

### 5.3 Relativer Adressierungsmodus, kurz Normalmodus

Im Normalmodus adressieren alle den Hauptspeicher ansprechenden Befehle und alle Sprungbefehle relativ zu einer Anfangsadresse 0. Das Befehlsfolgeregister schaltet in relativen Adressen fort. Die Seitennummern werden über den Leitblock ersetzt. Der im Normalmodus zulässige Adressenbereich wird durch die zum Leitblock-Anfang relative Anfangsadresse  $\Delta_1^0$  eines Ausschnittes der Tabelle der Kachelnummern im Leitblock und durch die Länge  $\Delta_2^0$  dieses Ausschnittes angegeben. Die Grenzübergaben  $\Delta_1^0$  und  $\Delta_2^0$  stehen im Leitblock in der Halbzelle mit der relativen Adresse 64. Bei Unterbrechung werden die Register im Leitblock abgelegt.

Der Normalmodus wird durch die Flipflopzustände  $\overline{\text{BEBY}}$ , BEBN,  $\overline{\text{BEBO}}$  dargestellt.

#### 5.4 Abwicklermodus

Der Abwicklermodus ist ein im zulässigen Adressenbereich erweiterter relativer Adressierungsmodus. Er unterscheidet sich vom Normalmodus durch andere Grenzangaben  $\Delta_1^A$  und  $\Delta_2^A$ , die im Leitblock in der Halbzelle mit der relativen Adresse 65 stehen.

Der Abwicklermodus wird durch die Flipflopzustände  $\overline{\text{BEBY}}$ ,  $\text{BEBN}$ ,  $\text{BEB0}$  dargestellt.

#### 5.5 Adreßrechnungsmodus

Adreßrechnungen können in zwei verschiedenen Adreßrechnungsmodi ausgeführt werden unabhängig vom eingestellten Adressierungsmodus. Im Modus 16 läuft die Adreßrechnung über die rechten 16 Bit der Register, und der Einerrücklauf erfolgt von der 9. zur 24. Stelle. Die vorderen 8 Bit werden gelöscht. Im Modus 24 erfaßt die Adreßrechnung alle 24 Bit der Register, der Einerrücklauf erfolgt von der 1. zur 24. Stelle der Adresse. Alle Befehle, die Adreßrechnungen bewirken, arbeiten nach Modus 16 anders als nach Modus 24. Die Interpretation des Interncodes der Befehle ist bei einer Reihe von Befehlen vom Adreßrechnungsmodus abhängig, ~~s. Abs. 8.3.~~

Der Adreßrechnungsmodus wird durch das Flipflop  $\text{BEBA}$  eingestellt:

Modus 16:  $\text{BEBA}$   
Modus 24:  $\overline{\text{BEBA}}$

## 5.6 Wartungsmodus

Für Prüf- und Wartungsläufe besteht ein eigener Modus. Ihm sind einige der nicht durch TAS-Befehle belegte Befehlscodes zulässig. Einige der in den anderen Modi nicht adressierbaren Register können im Wartungsmodus angesprochen werden. Der Wartungsmodus wird durch ein mit dem Befehl VPU einstellbares Steuerbit eingeschaltet (vgl. Abs. 6.4.2)

## 6. Speicherorganisation und Speicheransteuerung

### 6.1 Das Speicherwerk

Das Speicherwerk besteht aus dem Hauptspeicher, dem Speicherwerks-Leitwerk und den Speicherablauf-Steuerungen des Hauptspeichers. Für den Hauptspeicher sind drei verschiedene Ausbaustufen vorgesehen:

Grundausbau mit 64 K Ganzwort-Zellen (1 K = 1024)

1. Ausbaustufe mit 128 K Ganzwort-Zellen
2. Ausbaustufe mit 256 K Ganzwort-Zellen (Vollausbau)

Bei Ausstattung mit einem Massenkernspeicher ist der Hauptspeicher auf maximal 2048 K ausbaufähig.

### 6.2 Speicherverschränkung, Kacheln

Der Speicher ist aus Speichereinheiten (Moduln) von 16 K Ganzwort-Zellen aufgebaut. Bei Grundausbau (4 Moduln) liegen je vier adressenmäßig benachbarte Ganzwortzellen in einem der vier Moduln, in den weiteren Ausbaustufen des Speichers verteilen sich 8 bzw. 16 konsekutiv adressierbare Ganzwortzellen über 8 bzw. 16 Moduln. 1024 konsekutiv adressierbare Ganzwortzellen bilden eine Speicherkachel, und zwar die Ganzwortzellen 0 - 1023 die Kachel 0, die Ganzwortzellen 1024 - 2047 die Kachel 1 usw.. Jede Kachel hat folglich teil an allen Moduln. Die Zuordnung zwischen Ganzwortadressen und Plätzen in den Speichermoduln erfolgt je nach Ausbaustufe über die Bits 22 und 23, 21 - 23 bzw. 20 - 23 einer Ganzwortadresse, so daß benachbarte Ganzwortadressen zyklisch in benachbarten Speichermoduln liegen.

Jeder Speichermodul besitzt eine eigene Speicherablaufsteuerung. Ist sie durch Zugriff auf Information im zugehörigen Speichermodul belegt, so ist gleichzeitig kein weiterer Zugriff auf diesen Modul möglich, wohl aber auf einen anderen Modul. Rechenwerk und EA-Werk können folglich echt parallel arbeiten, solange sie auf verschiedene Moduln zugreifen. Die Zykluszeit ist deshalb vom Speicherausbau abhängig und nur als Mittelwert angebbbar. Sie beträgt für einen Modul 0,9  $\mu$ sec. Die Zugriffszeit zu einem Modul beträgt im Mittel 0,5  $\mu$ sec. Durch die Verschränkung des Speichers ergibt sich in Abhängigkeit vom Speicherausbau ein mittlerer Abstand zwischen zwei Speicheraufrufen von etwa 0,1  $\mu$ sec.

### 6.3 Arbeitsdefinitionen

Programmteile, die im Normal- oder Abwicklermodus arbeiten, sollen als Prozesse bezeichnet werden. Der Begriff Prozeß sei hier als Arbeitsdefinition verstanden und decke sich nicht notwendig mit dem gleichnamigen Begriff der TR440-Software. Ein Prozeß besitzt einen sogenannten Leitblock, der nicht seinem Adressenraum angehört. Der Leitblock beginnt stets mit einer Achtelkachel. Die Kacheladresse der ersten Zelle des Leitblocks ist die Leitadresse.

Der Leitblock wird durch Bereichsangaben  $\Delta_1$  und  $\Delta_2$  unterteilt. Die Bereichsangaben bestimmen den erreichbaren Adressenraum im Normal- bzw. Abwicklermodus. Die Bereichsangaben können vom Betriebssystem verändert werden. Für Normal- und Abwicklermodus gelten in der Regel verschiedene Bereichsangaben.

Die Programme, die im System- oder Spezialmodus arbeiten, seien als Betriebssystem bezeichnet (auch dieser Begriff muß sich hier nicht notwendig mit dem gleichnamigen der TR440-Software decken). ~~Der Systemmodus unterliegt keiner Einschränkung bezüglich des in diesem Modus erreichbaren Adressenraumes. Sie unterliegen keiner Einschränkung bezüglich des in diesen Modi erreichbaren Adressenraumes.~~

Um eine klare Trennung zwischen den vom Programmierer angegebenen, in der Regel relativen Adressen, Seitennummern und Seitenadressen einerseits und dem Speicher mit seinen Speicherzellen (Plätzen) und Kacheln andererseits zu erreichen, wird im folgenden unterschieden zwischen programmrelativen Adressen und Seiten einerseits und den Kacheladressen und Kacheln im Kernspeicher andererseits. Die Bits 3 - 13 der Seitenadresse werden als Seitennummer, die Bits 3 - 13 der Kacheladresse als Kachelnummer bezeichnet.

Obgleich aus dem Kernspeicher stets nur Ganzworte (48 Bit  
eigentliche Information, dazu die Prüfstellen) ausgelesen  
werden, und zwar zunächst in das Sammelregister RS, wird auch  
von Halb- und Viertelworten gesprochen, da je nach Wirkung  
des Befehls aus dem Sammelregister auch Teile eines Ganzwortes  
geholt werden.



### 6.3.1 Seiten und Kacheln

Je 2048 hintereinanderliegende Halbwortadressen entsprechend 1024 konsekutiv adressierten Ganzworten bilden eine Seite. Die Halbwortadressen 0 bis 2047 bilden die erste Seite (Seite 0), die Halbwortadressen 2048 bis 4095 die zweite Seite (Seite 1) usw.. Je 32 hintereinander liegende Seiten bilden eine Großseite, d. h. in Großseite 0 liegen die Seiten 0 bis 31 mit den Halbwortadressen 0 bis 65 535, in Großseite 1 die Seiten 32 bis 63 usw..

Einer Seite eines Prozesses kann im Kernspeicher ~~genau~~ eine Kachel des Speichers zugewiesen werden. Aufeinanderfolgenden Seiten brauchen jedoch nicht aufeinanderfolgende Kacheln zugewiesen zu werden. Die Seiten einer Großseite z. B. eines Prozesses werden im allgemeinen also weder in konsekutiv nummerierten Kacheln noch in einer Großkachel liegen.

Die Zuordnung von Seitennummern zu Kachelnummern erfolgt durch einen Zuordnungsvorgang. Sie geht aus einer Tabelle im Leitblock hervor. Die Kachel 0 ist dabei teilweise reserviert für Abspeichervorgänge bei Unterbrechungen durch Eingriffe, Alarme und SSR- und Makro-Sprünge für die vier möglichen Rechnerkerne. Die Kachel 1 ist dem EA-Verkehr vorbehalten. Erst die folgenden Kacheln können zugeteilt werden.

#### 6.4 Hardwareseitige Belegung der Speicherkachel 0

Bei Unterbrechung durch Eingriff und Alarme oder durch SSR-Sprünge und Sprünge in ein Makro werden die Inhalte der Befehlswerksregister und eine Reihe von Bits abgespeichert, die den Zustand einzelner Steuerflipflops bei der Unterbrechung kennzeichnen.<sup>+) Bei Eingriff wird das Eingriffswort abgelegt. Die Ansprungsadressen für die Weiterbehandlung der verschiedenen Fälle und die einzustellende Indexbasis sind vorzugeben. In Abhängigkeit vom eingestellten Adressierungsmodus erfolgt die Abspeicherung in die Speicherkachel 0 oder in den Leitblock, siehe Abs. "Unterbrechungen". Im grundsätzlichen Aufbau stimmen die ersten 24 Ganzwortzellen der Speicherkachel 0 mit den ersten 24 Ganzwortzellen eines Leitblocks überein.</sup>

<sup>+) Anmerkung:</sup> Das 2. und 3. Ganzwort eines solchen Blockes aus 4 Ganzworten erhält beim Einschreiben TK3.



## 6.4.2 Bedeutung und Stellung der Steuerbits

Bei Abspeicherung der Befehlswerks-Register (bei Eingriff, Alarmen, SSR und MAKRO) stehen im rechten Halbwort des 3. Ganzwortes folgende Steuerbits:

Bit	Name	Bedeutung, siehe auch Anmerkung S. 43b
-----	------	----------------------------------------

### Betriebsmodi:

48	BEBY	Systemmodus	} vgl. Absatz 5
47	BEBN	Normalmodus	
46	BEBO	Abwicklermodus	

### Alarmbits:

45	BEED	Dreierproben-Alarm	
44	BEEW	Wecker-Alarm	
43	(BEFE	Eingriffssperre) <sup>+</sup> )	
42	BEEF	Stop-Alarm, vgl. Abs. 7.3	
41	BEEK	Befehls-Alarm	
40	BEEU	Alarm: Überlauf des Unterprogramm-Registers	
39	BEEC	Speicherschutz-Alarm	
38	REBUE	Arithmetischer Alarm	} vgl. "Wirkung der TR440-Befehle"
37	REAL	Typenkennungs-Alarm	

### Betriebsmodi:

36	BEBF	Steuerbit: Stop nach Abrufphase, vgl. Abs. 7.3
35	BEBT	Steuerbit: Wartungsmodus
34	BEBA	Steuerbit: Modus 16
33	BEBE	Steuerbit: Stop vor Abrufphase, vgl. Abs. 7.3

### Zustandsbits:

32	BEMP	Der anstehende Befehl ist ein Sprungbefehl
31	BEML	Der Befehl LEI geht vorher

3+) wird beim Abspeichern nicht gelöscht und bei VPU nicht gesetzt

Bit	Name	Bedeutung
-----	------	-----------

Zustandsbits:

30	BEMB	Der Befehl MABI geht vorher
29	BEMU	Der Befehl MU geht vorher
28	BEMM	Der Befehl MM geht vorher (im Modus 16)
27	BEMN	Der vorhergehende Befehl definiert mod2
26	BEMO	Der Befehl MFU oder MCFU geht vorher
25	BEMA	Der Befehl MF, MCF oder MD geht vorher

} mod1 wird  
} definiert

Bei Abspeicherung der Befehlswerksregister bei Eingriff und Alarmen stehen im linken Halbwort des 4. Ganzwortes folgende Steuerbits:

Bit	Name	Bedeutung
-----	------	-----------

Alarmbits:

24	BEFT	Technischer Fehler
23	BEER1	Rechneralarm vom 1. Rechnerkern
22	BEER2	Rechneralarm vom 2. Rechnerkern
21	BEER3	Rechneralarm vom 3. Rechnerkern
20	BEER4	Rechneralarm vom 4. Rechnerkern
19	BEEH	Hauptalarm (Stromausfall bzw. -Abschaltung)

Zustandsbits:

18	BEMQ	Der Befehl ist nicht zu Ende, d. h. er befindet sich noch in der Ausführungsphase.
17	BEIC	Der Befehl wurde bei Bildung der absoluten Indexbasis (durch Speicherschutz-Alarm) unterbrochen.
16	BEAB	Der Befehl wurde in der Abrufphase bei Auslesen eines Operanden (durch Speicherschutz-Alarm) unterbrochen.

Bit	Name	Bedeutung
-----	------	-----------

Zustandsbits:

15	BEAA	Der Befehl wurde am Anfang der Abrufphase unterbrochen.
14	BEAC	Der Befehl wurde im Abspeicher-Nanoprogramm unterbrochen.
13	BE10	Kennzeichnung verschiedener Ansprungsstellen aus dem Mikroprogramm der Ausführungsphase.
12	BE20	
11	BE30	
10	BESP	Die Dreierprobe darf nicht ersetzt werden.
9	BEVA	In der Ausführungsphase wird ein Hauptspeicheroperand benötigt.

Die Bits 1 - 8 geben den Code des unterbrochenen Befehls wieder.

Anmerkung: Die angeführte Bedeutung der Steuerbits ist die Hauptbedeutung. Infolge mehrfacher Ausnutzung der Steuerbits durch die Mikroprogramme ergibt sich bei manchen Befehlen eine von der angegebenen abweichende Bedeutung.

## 6.5 Speicherzuteilung

Einem Prozess oder Teilen eines Prozesses ist ebenso wie dem Betriebssystem bei der Übernahme in den Kernspeicher des TR440 ein Speicherraum zuzuweisen: der vom Prozess adressierte Adressenraum oder Teile davon werden auf einen Raum im Speicher in Form einzelner Kacheln abgebildet. Dadurch entspricht eine adressenmäßige Seite Information des Prozesses im Kernspeicher dem Inhalt einer bestimmten Kachel.

Die den Prozessen oder dem Betriebssystem oder Teilen davon zugewiesenen Kacheln können beliebig im Kernspeicher verteilt liegen. Sie sind durch ihre Kachelnummer bestimmt.

Die einem Prozess zugewiesenen Kacheln werden in einer speziell für den Prozess anzulegenden Tabelle durch die Kachelnummern festgehalten. Für diese Tabelle und eine Anzahl Hilfszellen wird einem Prozess ein Speicherraum von maximal 1024 Ganzwortzellen zugeordnet, der Leitblock des Prozesses. Die Kacheladresse der ersten Speicherzelle des Leitblocks ist die Leitadresse. Sie weist auf den Anfang einer Achtelkachel und ist unabhängig von der Länge des Leitblocks.

Die linken 16 Bit der aktuellen Leitadresse stehen im Leitadressenregister. Es wird bei einer Umschaltung in den Systemmodus oder Spezialmodus nicht umgesetzt. Da der Leitblock stets mit einer Achtelkacheladresse beginnt, sind die rechten 8 Bit der Leitadresse immer Null. Daher ist das Leitadressenregister nur 16 Binärstellen lang.

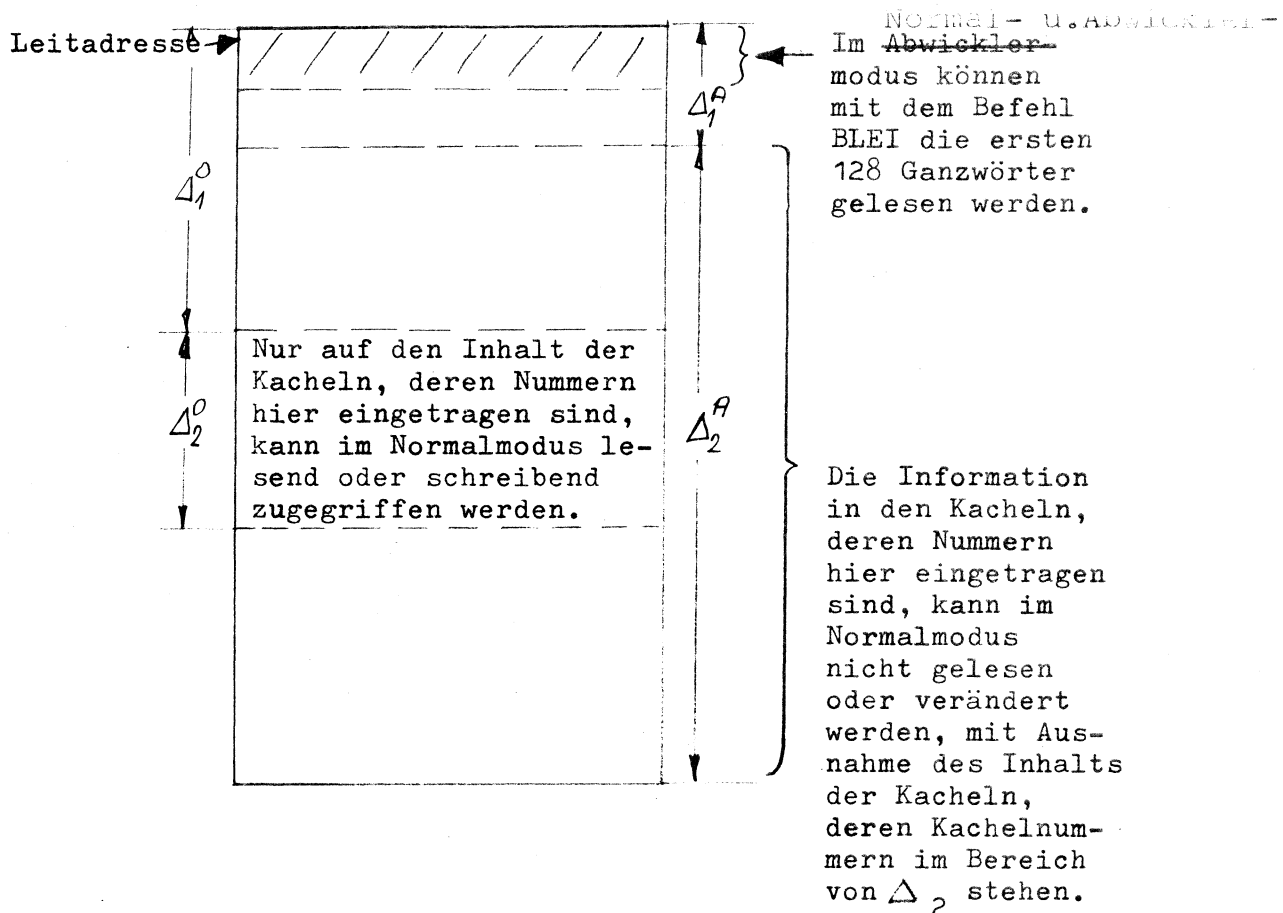
#### 6.5.1 Aufteilung und Belegung eines Leitblocks

Der Leitblock enthält in den vordersten Ganzwortzellen eine Reihe von Hilfszellen, die z. T. für hardwareseitige Vorgänge benötigt werden. Im übrigen Teil enthält er Tabellen von Kachelnummern, die verschiedenen Wertepaaren der Bereichsgrenzen zugeordnet sind. Die Kachelnummern stehen in Viertelwörtern, die auch z. T. nicht besetzt sein können.

Im Leitblock kann nur im System- oder Spezialmodus geändert (geschrieben) werden.



# Aufteilung des Leitblocks



$\Delta_1^0, \Delta_1^0$  = Bereichsangaben für den Normalmodus

$\Delta_1^A, \Delta_2^A$  = Bereichsangaben für den Abwicklermodus

# Hardwareseitig zugeordnete Zellen im Leitblock

Leitadresse

+ 0	-	-	
2	-	-	
4	Indexbasis *)		-
6	Sprungziel bei SSR		-
8	-	-	
10	BB	BA	Abspeicherung bei SSR-Sprung
12	BF	Steuerbits	
14	-	-	
16	-	-	
18	BB	BA	Abspeicherung bei Makro-Sprung
20	BF	Steuerbits	
22	-	-	
24	RS		
26	BB	BA	Abspeicherung bei Eingriff
28	BF	Steuerbits	
30	Code	Steuerbits	
32	RS		
34	BB	BA	Abspeicherung bei Alarm, wenn Alarmsperre 1 nicht und nicht Systemmodus gesetzt war
36	BF	Steuerbits	
38	Code	Steuerbits	
40	RS		
42	BB	BA	Abspeicherung bei Alarm, wenn Alarmsperre 1 und nicht Systemmodus gesetzt war
44	BF	Steuerbits	
46	Code	Steuerbits	
48			*)  Die Typenkennung von Zelle an, mit welchen Bereichsgr aus der angegebenen relati dexbasisadresse die absolu dexbasisadresse zu errechn  $\langle 4 \rangle_t = 3$ (oder 1): Bereichsgr aus $\langle 64 \rangle_2$  $\langle 4 \rangle_t = 2$ (oder 0): Bereichsgr aus $\langle 65 \rangle_2$
50			
52			
54			
56			
58			
60			
62	3 7 8 12	27 31 32 36	
64	B422 B421 -	B422 B421 -	
	für Normalmodus für Abwicklermodus		

Die Halbwort-Kacheladressen 4 und 6 sowie 64 und 65 müssen vorbesetzt werden. Die nicht besetzten Halbworte werden von der Hardware nicht verändert. Zu den Steuerbits vergl. Abs. 6.4.2.

Bei Einspeicherung durch die Hardware wird das 2.u.3.G. eines Viererblockes mit TK3 abgelegt.

### 6.5.2 Adressierung des Leitblocks

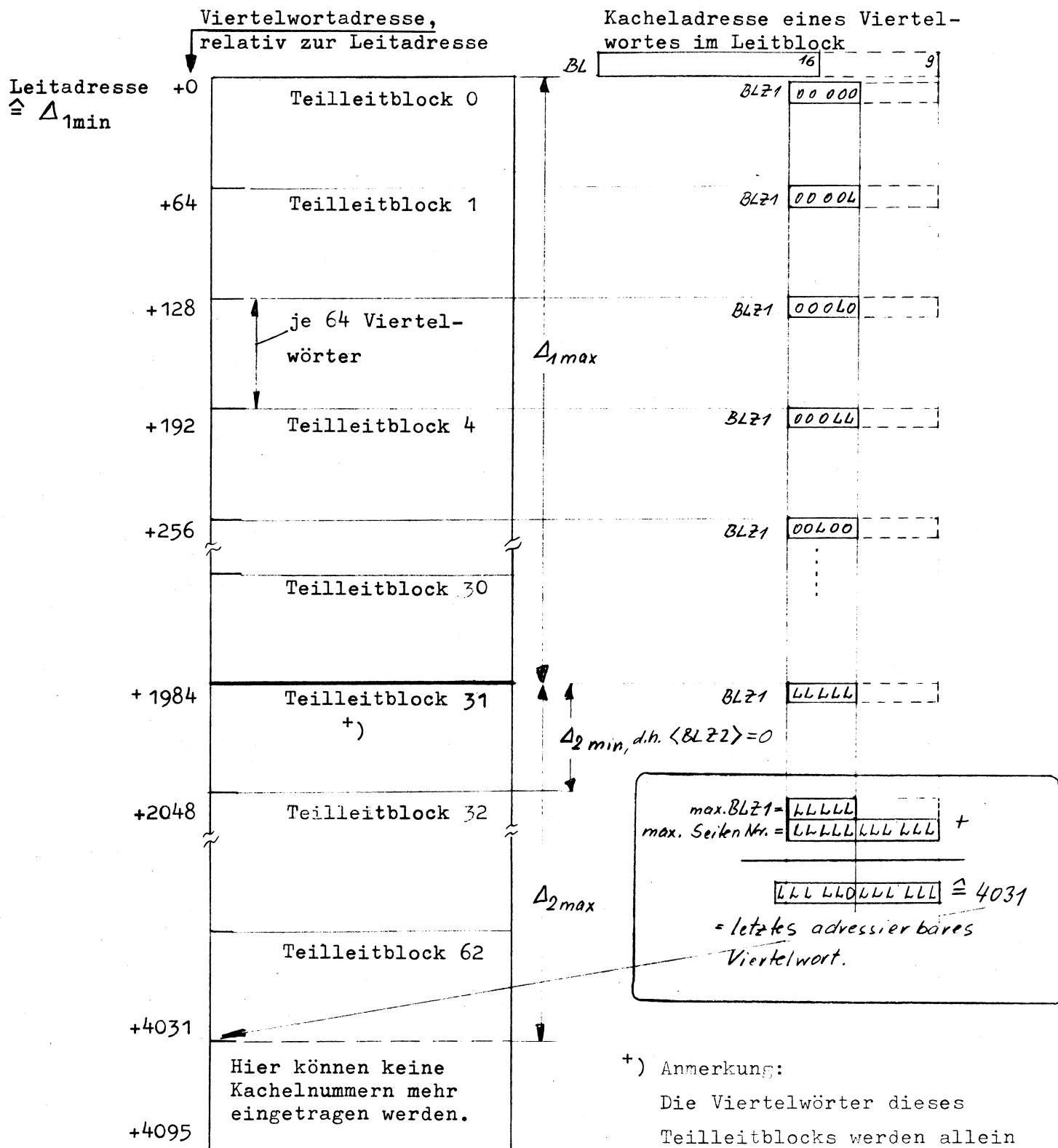
Zur Adressierung der Kacheltabellen im Leitblock stellt man sich den Leitblock zweckmäßig als in Viertelworte unterteilt vor (obgleich im Speicher nur ~~Ganzworte~~ <sup>Halbworte</sup> adressiert werden können). Dazu sind die Bereichsangaben (je 5 Bit) nach rechts um 6 mit Nullen besetzten Stellen verlängert zu denken. Sie unterteilen den Leitblock in Abschnitte (Teilleitblöcke) zu je  $2^6$  Viertelworten  $\hat{=}$  16 Ganzworten mit den zur Leitadresse relativen Ganzwortkacheladressen 0, 16, 32 bis höchstens 992. Mit der Größe  $\Delta_1$  werden also maximal 31 Teilleitblöcke erfaßt. Die Größe  $\Delta_2$  erfaßt mit Hilfe der rechten 6 Bit ~~die~~ der Seitennummer  $N_V$  maximal die folgenden 32 Teilleitblöcke.

Es ist zwischen den Bereichsangaben  $\Delta_1^O$  und  $\Delta_2^O$ , die im Normalmodus gelten, zu unterscheiden und den Bereichsangaben  $\Delta_1^A$  und  $\Delta_2^A$ , die im Abwicklermodus gelten.

Normalmodus:  $\langle \text{BLZ1} \rangle := \langle \langle \text{BL} \rangle \cdot 2^8 + 64 \rangle_{8,12} = \Delta_1^O$   
 $\langle \text{BLZ2} \rangle := \langle \langle \text{BL} \rangle \cdot 2^8 + 64 \rangle_{3,7} = \Delta_2^O$

Die durch die Angabe  $\Delta_1^O$  überstrichenen Teilleitblöcke enthalten Hilfszellen und z. T. auch Kachelnummern. Die ersten 24 Ganzworte und die Plätze mit den zur Leitadresse relativen Halbwortkacheladressen 64 und 65 sind für hardwareseitige Abspeicherungen und Einstellungen in der Anordnung entsprechend der in Speicherkachel 0 vorbehalten. Die dort nicht belegten Plätze, das erste Ganzwort und die Halbwortzellen mit den zur Leitadresse relativen Kacheladressen 3 und 5 werden hardwareseitig nicht verändert. Die Halbzellen 64 und 65 enthalten die Bereichsgrenzen für Normal- und Abwicklermodus.

# Aufteilung des Leitblocks in Viertelwörter



Der Inhalt der ersten 128 Ganzworte des Leitblocks kann mit dem Befehl BLEI im Normalmodus gelesen werden.

Die durch  $\Delta_2^0$  überstrichenen Teilleitblöcke enthalten in Viertelwörtern die Nummern der Kacheln, die dem im Normalmodus arbeitenden Prozessteil zugeteilt sind und gegebenenfalls weitere, noch durch keine Kachelnummer besetzte Viertelworte. Zur Adressierung dieser Viertelworte vgl. Abs. 6.6 .

Bei der Speicheransteuerung kann durch die rechten 16 Bit der relativen Seitenadresse der letzte durch  $\Delta_2^0$  adressierbare Teilleitblock noch ganz überstrichen werden. Die Angabe  $\Delta_2^0$  (und ebenso  $\Delta_2^A$ ) gestattet es also, einem Prozess im Speicher maximal  $32 \cdot 16 \cdot 4$  Kachelnummern  $\hat{=}$  2048 K zuzuordnen. Der kleinste, einem Prozess im Leitblock zuteilbare Speicherraum umfaßt 16 Ganzworte = 64 Kachelnr. = 2 Großseiten. Da nicht alle Viertelwörter der Kacheltabelle besetzt zu sein brauchen, benötigt der Prozess im Kernspeicher nur den Raum, dessen Kachelnummern im Leitblock in der Kacheltabelle vermerkt ist, sowie den Raum, den das Betriebssystem für diesen Prozess speziell anlegt. Es ist jedoch auch möglich, daß Teile eines Programms sich bereits im Kernspeicher befinden, ohne daß die Nummern der mit diesen Teilen belegten Kacheln bereits in der Kacheltabelle vermerkt sind. Ebenso kann dieselbe Kachelnummer auch mehrfach im Leitblock verzeichnet sein.

$$\begin{aligned} \text{Abwicklermodus: } \langle \text{BLZ1} \rangle &:= \langle \langle \text{BL} \rangle \cdot 2^8 + 65 \rangle_{8,12} = \Delta_1^A \\ \langle \text{BLZ2} \rangle &:= \langle \langle \text{BL} \rangle \cdot 2^8 + 65 \rangle_{3,7} = \Delta_2^A \end{aligned}$$

Die Bereichsangaben in den Registern BLZ1 und BLZ2 teilen den Leitblock ganz entsprechend wie im Normalmodus in Teilleitblöcke von 16 Ganzworten ein. Im Abwicklermodus kann der Inhalt der ersten 128 Ganzworte des Leitblocks mit dem Befehl BLEI gelesen werden.

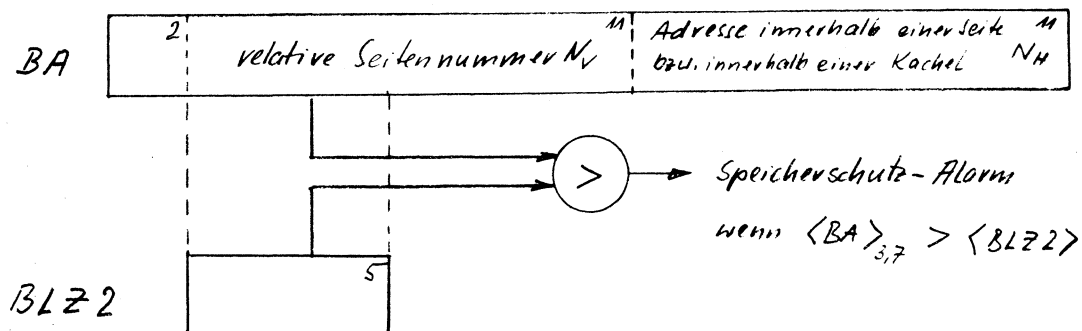
Der durch die Bereichsangaben  $\Delta_2^A$  adressierbare Teil des Leitblocks enthält in Viertelworten die Nummern der Kacheln, die dem in Abwicklermodus geschriebenen Prozess zugeteilt sind bzw. Platz für die Eintragung der Kachelnummern zuteilbarer Kacheln.

Die in den Kacheln tabellen rechtsbündig in die Viertelworte eingetragenen Kachelnummern bestehen aus 11 Bit. Das links stehende 1. Bit dient zur Kennzeichnung, ob die Information in dieser Kachel verändert wurde (Abspeicherbit). Bei einem Schreibzugriff auf einen Platz in dieser Kachel wird das Abspeicherbit im Seitenadressregister geprüft. Ist es in 0 gesetzt, wird Speicherschutzalarm gemeldet (siehe Abs. 7.2.2).

Die Bereichsangaben gestatten es, die Zuteilung von Speicherraum an einzelne Prozesse und Prozessteile so zu organisieren, daß eine gegenseitige Beeinflussung dieser Prozesse oder Prozessteile ausgeschlossen ist.

## 6.6 Speicheransteuerung

Von der durch die Größen  $mod1$  und  $mod2$  (s. "Wirkung der Befehle") aus der 16-Bit-Adresse entstandenen programmrelativen 24-Bit-Adresse adressieren die rechten 11 Bit innerhalb einer Seite relativ zum Anfang der Seite (sie werden im folgenden kurz  $N_H$  genannt). Die linken 13 Bit, von denen die beiden vordersten für die Speicheransteuerung ohne Bedeutung sind, im folgenden kurz  $N_V$  genannt, definieren die programmrelative Seitennummer. Diese wird vor Ansteuerung des Speicherplatzes durch eine Kachelnummer  $N_V^i$  aus einem der Viertelwörter der Kacheltabelle im Leitblock ersetzt. Vor dieser Ersetzung wird geprüft, ob die angegebene programmrelative Seitennummer die Bereichsgrenze  $\langle BLZ2 \rangle$  nicht überschreitet. Andernfalls wird Speicherschutzalarm gemeldet.



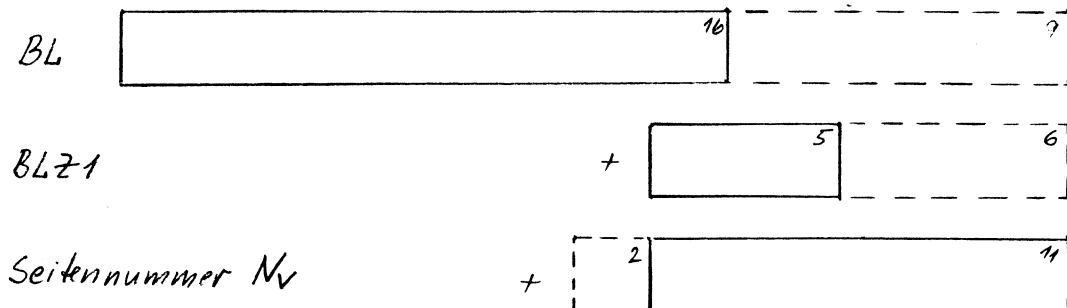
Die Art der Speicheraansteuerung ist vom Adressierungsmodus abhängig. Im Systemmodus wird der Speicherplatz, die Kacheladresse, mit der durch die Größen mod1 und mod2 aus der 16-Bit-Adresse entstandenen 24-Bit-Adresse angesteuert. Bei Sprungbefehlen im System- und Spezialmodus werden jedoch bei Ermittlung des Sprungziels die linken 8 Bit des Befehlszählers übernommen, sodaß nur Sprünge innerhalb einer Großseite möglich sind. Ausnahme: bei SE, SUE und VMO mit entsprechender Spezifikation und Sprüngen, die mit MU oder MABI modifiziert werden, werden alle 24 Bit des Befehlszählers ersetzt, so daß Sprünge in andere Großseiten möglich sind. In allen anderen Fällen muß die Kacheladresse aus der programmrelativen Adresse über die aktuelle Leitadresse unter Beachtung der eingestellten Bereichsgrenzen errechnet werden.

Die linken 13 Bit der programmrelativen 24-Bit-Adresse (von der wiederum die beiden links stehenden Bits ohne Bedeutung für die Speicheraansteuerung sind), bilden die Seitennummer  $N_V$  und werden zur Auswahl eines Viertelwortes im Leitblock benötigt. Sie werden, falls sie nicht in einem gültigen Seitenadreßregister stehen (s. Abs. 6.7), mit dem Inhalt des Leitadressenregisters  $\langle BL \rangle$  und der gültigen Bereichsgrenze  $\langle BLZ1 \rangle$  zur Viertelwort-Kacheladresse eines Viertelwortes im Leitblock in der Kacheltabelle modifiziert. Die Seitennummer  $N_V$  wird durch den Inhalt des so errechneten Viertelwortes aus dem Leitblock, durch die Kachelnummer  $N'_V$ , ersetzt. Mit  $N'_V$  und  $N_H$ , den rechten 11 Bit der Seitenadresse, ergibt sich die Kacheladresse, mit der der Speicher angesteuert wird.



Kacheladresse eines Viertelwortes im Leitblock

Ein bei der Addition sich ergebender Überlauf bleibt unberücksichtigt!



Formal ergibt sich folgende Beziehung:

1. die rechten 11 Binärstellen der im BA stehenden Adresse bleiben unverändert. Sie adressieren innerhalb einer Seite ebenso wie innerhalb einer Kachel relativ zum Anfang der Seite wie der Kachel.

2. die linken 13 Binärstellen des BA ergeben sich aus

$$\begin{aligned} \langle BA \rangle_{1,2} &:= \text{ohne Bedeutung für die Speichervansteuerung} \\ \langle BA \rangle_{3,13} &:= \langle \langle \langle BL \rangle \cdot 2^8 + \langle BLZ1 \rangle \cdot 2^5 + \langle BA \rangle_{3,12} \rangle_x \end{aligned}$$

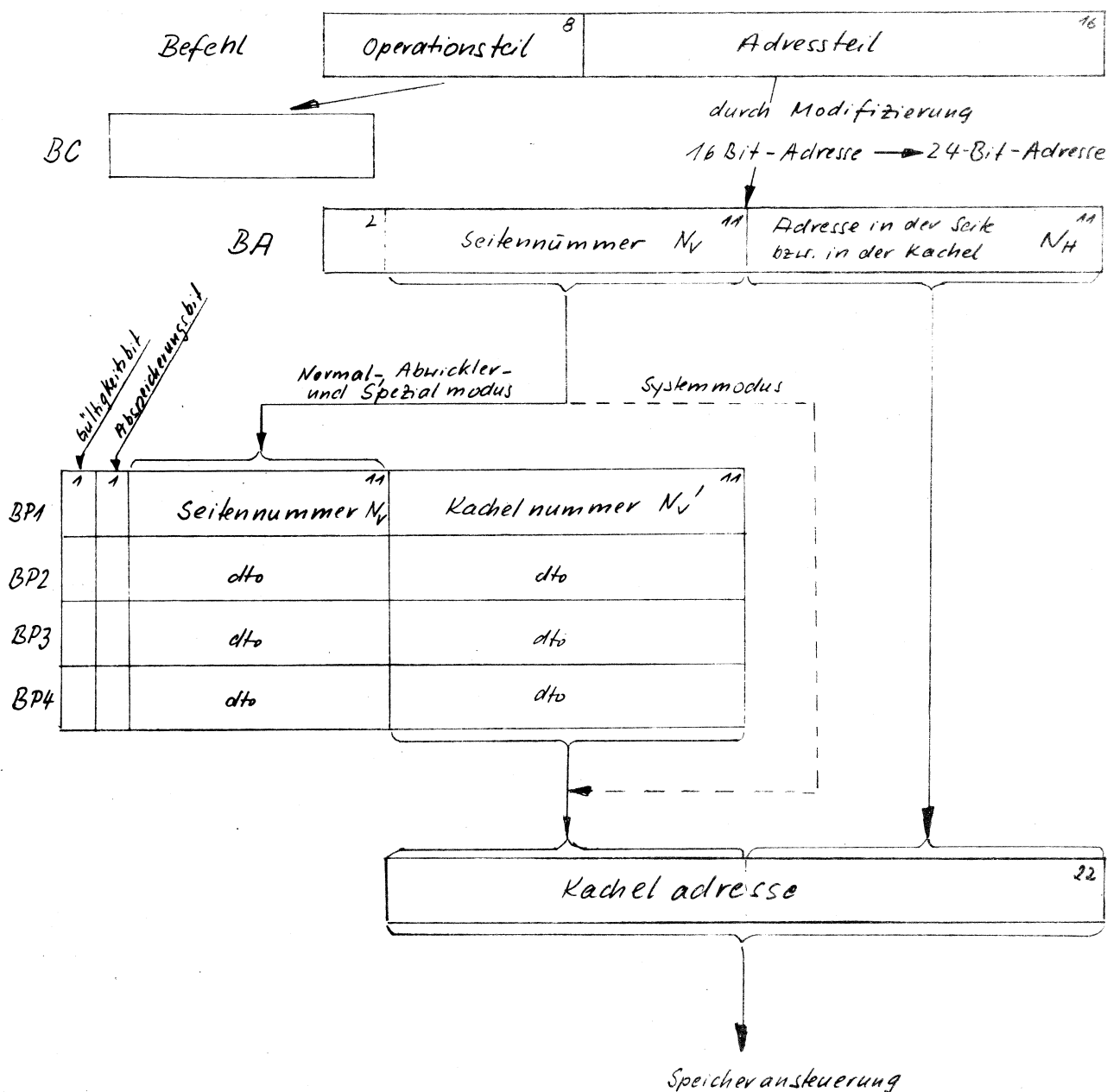
$$\begin{aligned} \text{mit } x = 2,12 \text{ , wenn } \langle BA \rangle_{13} &= 0 \\ x = 14,24 \text{ , wenn } \langle BA \rangle_{13} &= 1 \end{aligned}$$

### 6.7 Die Seitenadreßregister

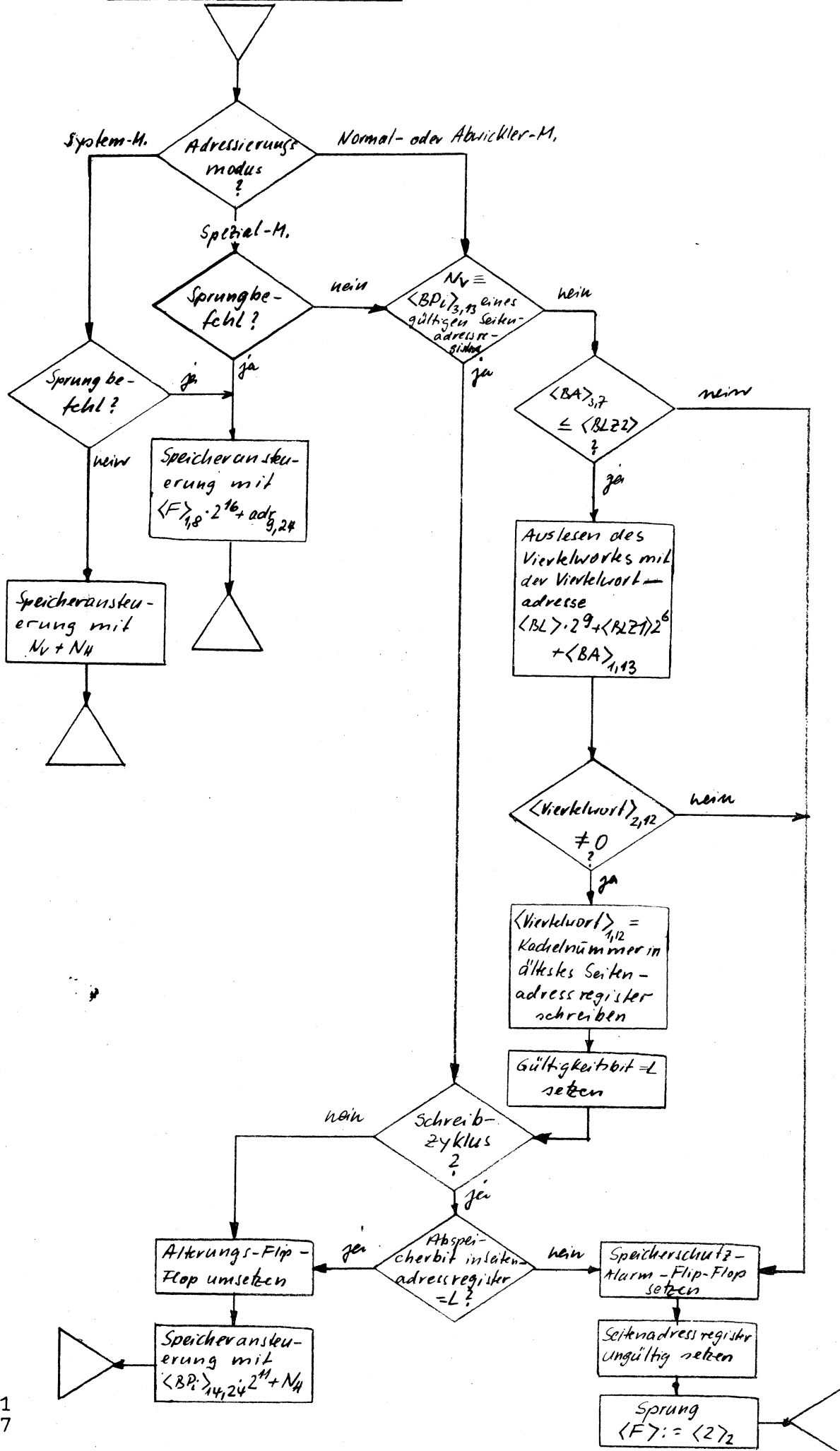
Um nicht bei jeder Speicheransteuerung erneut die Kachelnummer aus dem Leitblock ermitteln zu müssen, werden die vier zuletzt angesprochenen Seitennummern  $N_V$  zusammen mit den zugehörigen Kachelnummern  $N'_V$  in vier Assoziativregistern, den Seitenadreßregistern, gespeichert. Bei einer Speicheransteuerung wird zuerst geprüft, ob die Seitennummer  $N_V$  der programmrelativen Seitenadresse bereits in einem gültigen der vier Seitenadreßregister steht. Ist das der Fall, wird sie direkt durch die Kachelnummer  $N'_V$  ersetzt und damit die Speicheransteuerung vorgenommen.

Steht in keinem der gültigen Seitenadreßregister die zu ersetzende Seitennummer  $N_V$ , dann wird die Kachelnummer wie oben beschrieben über Leitadresse und gültige Bereichsgrenze aus dem Leitblock ermittelt und in das älteste Seitenadreßregister geschrieben. Enthält das gefundene Viertelwort im Leitblock keine Kachelnummer, so wird Speicherschutzalarm gemeldet.

Ersetzung der programmrelativen 16-Bit-Adresse durch die Kacheladresse



Zu 6.6 Hardwareseitige Vorgänge bei der Speicheransteuerung.  
Vereinfachte Darstellung



## 6.8 Indexspeicheransteuerung

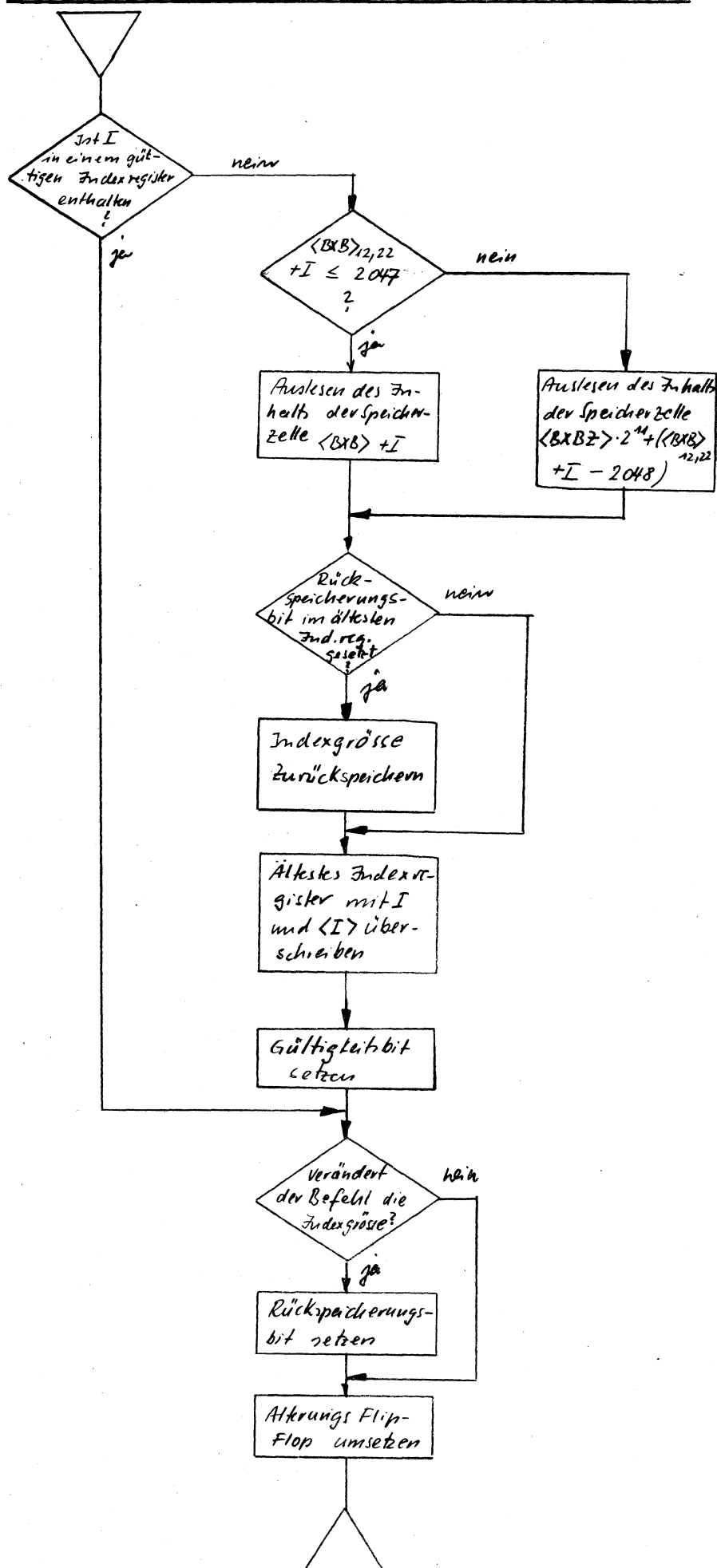
Durch die Befehle BCI  $n$  und ZI  $n_2$  wird ein Bereich des Hauptspeichers von 256 Halbwortzellen, die auch in zwei verschiedenen Kacheln liegen können, als sekundärer Indexspeicher deklariert. Die Kacheladresse des Anfangs dieses Bereichs wird im Indexbasisregister abgelegt. Es wird geprüft, ob Anfangs- und Endadresse des Indexbereichs in zugewiesenen und nicht schreibgeschützten Kacheln liegen. Andernfalls wird Speicherschutzalarm gemeldet.

Bei Adressierung einer Indexzelle wird die Indexadresse (8 Bit) zur Indexbasisadresse addiert und damit der Speicher angesteuert. Zur Beschleunigung des Verkehrs mit den Indexzellen werden in den vier assoziativen Indexregistern (primärer Indexspeicher) die vier zuletzt benutzten Indexadressen und die zugehörige Indexgröße gehalten. Vor der Speicheransteuerung wird zunächst geprüft, ob die angesprochene Indexadresse in einem der gültigen Indexregister steht. Ist das der Fall, so wird die Indexgröße (24 Bit) direkt ausgelesen. Andernfalls wird der Speicher wie oben beschrieben mit Indexbasisadresse und Indexadresse angesteuert, und Indexadresse und Indexgröße werden in ein ungültiges oder das älteste Indexregister eingetragen. Gegebenenfalls wird die zu überschreibende Indexgröße vor dem Überschreiben in den Speicher zurückgebracht. Beim Ändern einer Indexgröße wird das Rückspeicherungsbit in L gesetzt.

Überschreitet der Indexzellenbereich eine Kachelgrenze, so liegen die folgenden Indexzellen in der nächsten zugewiesenen Kachel, deren Nummer im Indexbasisadressen-Zusatzregister festgehalten wird.



## 6.9 Ansteuerung der Indexzellen und Indexregister



## 7. Wirkung von Unterbrechungen

Die Rechnerkerne können durch Eingriffe und Alarme in den augenblicklich in ihnen ablaufenden Vorgängen unterbrochen werden. Mit der Unterbrechung wird Information über die Ursache der Unterbrechung mitgeliefert. Diese Information wird zur Untersuchung und Behandlung der Unterbrechung aufbereitet. Ebenso werden einige Befehlswerks-Register vom Zustand der Unterbrechung und der Zustand einiger Steuerflipflops sichergestellt. Danach wird eine bestimmte vorgebbare Adresse angesprungen.



## 7.1 Eingriffe

### 7.1.1 Definition

Eingriffe sind Unterbrechungen des Rechners, die von außen, also von der Peripherie der Anlage her, den augenblicklich in Befehls- und Rechenwerk ablaufenden Vorgang unterbrechen. Ein Eingriff kann nur nach Beendigung einer Abrufphase eines einzelnen Befehls gemeldet werden. Siehe Abs. 7.3 .

Bei der Unterbrechung durch einen Eingriff werden drei verschiedene Vorgänge angestoßen:

- a) Der Zustand des den Rechnerkern belegenden Prozesses oder Teils des Betriebssystems, der mit dem Eingriff i. A. in keinerlei Zusammenhang steht, ist zu erhalten, um den unterbrochenen zu gegebenem Zeitpunkt an der Unterbrechungsstelle fortzusetzen. Der Unterbrochene wird die Unterbrechung i. A. nicht selbst bemerken.
- b) Die mit dem Eingriff mitgeschickte Information, das Eingriffswort, muß zur Analyse in geeigneter Weise abgelegt werden.
- c) Die mit dem Eingriff mitgeschickte Information muß analysiert und demjenigen mitgeteilt werden, den sie betrifft. Beides kann bis zu einem späteren Zeitpunkt verschoben werden.

Das Mikroprogramm "Eingriff" wird angesprungen, wenn folgende Bedingungen erfüllt sind:

Eingriff := BEEE · BEFE · BEFA · "Alarm" · "ES-Code"  
                  • et  
                  + vel

BEEE = Eingriffswunsch vom EA-Werk liegt vor

BEFE = Eingriffssperre

BEFA = Alarmsperre 1

ES-Code = ein den Eingriff sperrender Code steht an.

### 7.1.2 Eingriffsarten

**Abschnittseingriffe:** Ein Abschnitt ist die durch einen EA-Werksbefehl transportierte Informationsmenge. Er umfaßt maximal eine Seite. Abschnittseingriffe werden vom Kanal bei Wechsel des EA-Werksbefehls gegeben, wenn im folgenden EA-Werksbefehl das Bit "Abschnittseingriff" (Bit 24) gesetzt ist, also nach Abschluß des Speichertransportes, der mit dem vorhergehenden EA-Werksbefehls gesteuert wurde.

**Blockeingriffe:** Sind in einem EA-Werksbefehl die Bits für "Durchstart" oder "Neustart" gesetzt, so wird vor Ausführung des durch diesen EA-Werksbefehl definierten Transportes ein Blockeingriff erzeugt. Diese Blockeingriffe können allerdings durch eine Spezifikation in der Startinformation bis zur nächsten Startinformation unterdrückt werden. Das Ende der Übertragung wird durch einen Fehlereingriff (Stopfehler) angezeigt.

**Fehlereingriff:** Die Geräte an einem Kanal übergeben mit der Blockendemeldung gegebenenfalls ein Fehlerzeichen, das im Eingriffswort vermerkt wird. Andererseits kann auch der Kanal selbst gewisse Fehler erkennen und einen Fehlereingriff erzeugen.

Anrufseingriff:

Peripheriegeräte melden durch einen Anrufseingriff ihre Bereitschaft, Information zu übernehmen oder zu übergeben. Bei Koppelung mehrerer Rechner meldet ein anderer Rechner seine Bereitschaft zur Informationsübergabe oder -übernahme ebenfalls durch einen Anruf.

Während Abschnitts-, Block- und Fehlereingriffe nur in der Folge eines vorhergegangenen Startbefehls eintreffen können, also einen laufenden EA-Auftrag betreffen, kann ein Anrufseingriff jederzeit eintreffen, soweit nicht die Eingriffssperre verhängt ist.

### 7.1.3 Das Eingriffswort

Bei einem Eingriff wird vom Eingriffssteuerwerk ein Eingriffswort gebildet, das in Zelle 0 der Speicherkachel 0 abgelegt wird. An einzelnen Bits wird erkannt, um welchen Eingriff es sich handelt und von welchem Kanal und welchem Gerät an diesem Kanal der Eingriff herrührt. Bei Fehlereingriffen enthält es außerdem Fehlerzeichen zur Analyse des Fehlers.

1	9	16 17	24 25	40	43	46	48
0 L		Fehler z vom Gerät	Geräte - Nr. 0-7	Fehler FF		Kanal - Nr	

TK

- Bit 1 = L : Anrufseingriff
- Bit 2 = L : Fehlereingriff
- Bit 47 = L : Abschnittseingriff
- Bit 48 = L : Blockeingriff

#### 7.1.4 Hardwareseitige Vorgänge bei Eintreffen eines Eingriffs

Es wird das Flipflop "Eingriffssperre" gesetzt (BEFE = L). Weitere Eingriffe werden zunächst nicht abgenommen, bleiben aber im Kanalsteuerwerk vermerkt. Einige Registerinhalte und die Stellungen gewisser Steuerflipflops werden abgespeichert. Ist Systemmodus eingestellt, so erfolgen die Abspeicherungen in die Speicherkachel 0, in allen anderen Fällen in den Leitblock, dessen Leitadresse im Leitadressenregister vermerkt steht. Die unten angegebenen Adressen gelten dann relativ zur Leitadresse.

<0> := Eingriffswort (das gilt unabhängig vom eingestellten Adressierungsmodus)  
<24> := <Verteilerregister RS>  
<26<sub>2</sub>>:= <Bereitadressenregister BB>  
<27<sub>2</sub>>:= <Adressenregister BA>  
<28<sub>2</sub>>:= <Befehls-Folge-Register BF>  
<29<sub>2</sub>>:= Steuerbits, siehe Abs. 6.4.2  
<30<sub>2</sub>>:= Code, Steuerbits, siehe Abs. 6.4.2  
<31<sub>2</sub>>:= <Adressen-Hilfsregister BH>

Es erfolgt ein Sprung auf die in Zelle 5 in Kachel 0 angegebene Adresse.

<F> := <5<sub>2</sub>>

# Hardwareseitige Vorgänge bei Eintreffen eines Eingriffs

## Generelle Einstellungen und Abspeicherungen

BEFE = L (Der Eingriff wird gesperrt)  
 < O > := Eingriffswort  
 < BB ><sub>1,16</sub> := < BL > , war Systemmodus eingeschaltet,  
 gilt: < BB > := 0  
 BEBA (Modus 24); BEBY, BEBN (Systemmodus)  
 < F > := < 5<sub>2</sub> >  
 < BLZ1 > , < BLZ2 > bleiben unverändert

Alter Modus	Ablage der BWR	<Indexzellen>:= soweit Rückspei- cherbit gesetzt	< BXB >:=
Normalmodus	<< BL > + 24 > ff	< BI <sub>1...4</sub> >	< 4 <sub>2</sub> >
Abwicklermodus	<< BL > + 24 > ff	< BI <sub>1...4</sub> >	< 4 <sub>2</sub> >
Spezialmodus	<< BL > + 24 > ff	< BI <sub>1...4</sub> >	< 4 <sub>2</sub> >
Systemmodus	< 24 > ff	- (d.h. bleibt unverändert)	-

#### 7.1.5 Unter Eingriffssperre eintreffende Eingriffe

Die beim Kanal gebildeten Eingriffswünsche entstehen unabhängig von einer eventuell gesetzten Eingriffssperre. Die Eingriffe werden jedoch zunächst nicht gemeldet, sondern bleiben beim Kanalsteuerwerk stehen. Dabei werden Anrufe vom selben Gerät überspeichert. Ein auf einen Blockeingriff auf-  
laufender Blockeingriff führt zu einem Fehlereingriff (Eingriffsfehler). Die Übertragung wird abgebrochen.



### 7.1.6 Die Eingriffssperre

Der Eingriff kann auf dreierlei verschiedene Weise gesperrt sein bzw. gesperrt werden.

- a) Durch setzen des Flipflops "Eingriffssperre" (BEFE) durch den Befehl VMO E. Bei Unterbrechung durch einen Eingriff, Sprung ins System (SSR) oder in ein Makro wird dieses Flipflop durch die Hardware gesetzt.
- b) Das Mikroprogramm eines Befehls ist nur am Ende der Abrufphase durch einen Eingriff unterbrechbar. In der übrigen Zeit verhält sich der Rechner gegenüber Eingriffen wie bei gesetzter Eingriffssperre. Diese Sperrzeiten liegen größenordnungsmäßig zwischen Bruchteilen von Mikrosekunden bis zu einigen Mikrosekunden.
- c) Bei der Installation von weniger als vier Rechnerkernen lassen sich die Eingriffe auf einen nicht vorhandenen Rechnerkern leiten (über die kanalspezifische Kanalzuordnungszelle, siehe Absatz     ). Das kann selektiv für einzelne Kanäle und auch für einzelne Geräte an einem Kanal durchgeführt werden. Für das betreffende Gerät oder den Kanal wirkt das so, als ob der Eingriff gesperrt sei.

## 7.2 Alarme

### 7.2.1 Definition

Alarme sind Unterbrechungen, die durch Vorgänge in den Befehls- und/oder den Rechenwerksregistern, in Mikroprogrammen oder durch Ausfall der Stromversorgung hervorgerufen werden. Die Alarmunterbrechung stößt die folgenden Vorgänge an:

- a) Der Zustand der wesentlichen Register und Steuerflipflops wird abgespeichert, um einerseits die Alarmursache analysieren zu können, andererseits um später eine Fortsetzung des unterbrochenen Prozesses oder Teils des Betriebssystems, das den Rechnerkern belegt hatte, zu ermöglichen. Der unterbrochene wird die Unterbrechung nicht in jedem Fall bemerken.
- b) Es wird eine Alarmsperre verhängt.
- c) Es erfolgt ein Sprung auf eine an bestimmter Stelle angegebene Adresse.

Das Mikroprogramm "Alarm" wird angesprungen, wenn folgende Bedingungen erfüllt sind:

$$\begin{aligned} \text{Alarm} := & (\text{BEEW} + \text{BEER1} + \text{BEER2} + \text{BEER3} + \text{BEER4} + \text{BEEH}) \cdot \overline{\text{BEFA}} \\ & + \text{BEED} + \text{BEEC} + \text{BEFT} \\ & + \text{BEES} + \text{BEEU} + \text{BEEK} \\ & + (\text{REAL} + \text{REBUE}) \cdot \text{Rechenwerksbefehl} \cdot \overline{\text{SAA}} \cdot \overline{\text{SAT}} \\ & \cdot \text{et} \\ & + \text{vel} \end{aligned}$$

BEEW: Weckeralarm  
BEER1...4:Rechneralarm 1...4  
BEEH: Hauptalarm  
BEFA: Alarmsperre 1  
BEED: Dreierprobenalarm  
BEEC: Speicherschutzalarm  
BEFT: Mikroprogramm läuft in Schleife  
  
BEES: Stop bei Ende des Befehls  
BEEU: U-Überlauf  
BEEK: Befehlsalarm  
REAL: TK-Alarm  
REBUE: A-Alarm

### 7.2.2 Alarmgruppen und Alarmarten

Es werden zwei Alarmgruppen unterschieden, die durch ihre Behandlung z. T. voneinander abweichen.

#### Alarme der Gruppe I

Bezeichnung	Flip-flop	Stellg. bei Abspeich. <sup>+</sup>	Beschreibung
TK-Alarm	REAL	37	Tritt u. a. bei fehlerhafter Versorgung zahlreicher Befehle auf, z. B. Division zweier Zahlen ungleicher TK, s. "Wirkung der Befehle".
Arithmetischer Alarm	REBUE	38	Tritt u. a. bei fehlerhafter Versorgung zahlreicher Befehle auf, z. B. Division durch 0, s. "Wirkung der Befehle".
Stop-Alarm	BEEF	42	<u>Fall A: BEBE = L</u> Programmierte Unterbrechung am Anfang der Abrufphase eines Befehls, s. auch Abs. 7.3 .2. <u>Fall B: BEBF = L</u> Programmierte Unterbrechung am Ende der Abrufphase eines Befehls, s. auch Abs. 7.3 .2.

<sup>+</sup>) Bezieht sich auf die Stellung im 3. (Bits 25 - 48) und 4. Ganzwort (Bits 9 - 24) bei Abspeicherung der Befehlswerksregister, s. Abs 6.4.2 .

Alarmer der Gruppe I (Fortsetzung)

Bezeichnung	Flip-flop	Stellg. bei Ab-speich.	Beschreibung
BU-Überlauf	BEEU	40	Das Unterprogrammregister hat 255 überzählt.
Befehlsalarm	BEEK	41	Der Befehlszähler führt auf ein Wort, dessen TK ungleich 2 ist. Der abgespeicherte Stand des Befehls-Folge-Zählers ist undefiniert.
Dreierprobenalarm	BEEU	43	Ein Transport eines Wortes vom oder zum Speicher wurde fehlerhaft ausgeführt, so daß die angegebene DP nicht mit der errechneten übereinstimmt.
Speicherschutzalarm	BEEC	39	Unzulässiger Zugriff auf eine schreibgeschützte oder eine (evtl. noch) nicht zugeteilte oder nicht zuteilbare Kachel des Speichers.
Mikroprogramm-Ausführungsalarm	BEFT	24	Technischer Fehler im Mikroprogramm. Der Alarm führt zum Rechner-Halt.

### Alarmer der Gruppe II

Die Alarmer der Gruppe II können eine gewisse Zeit verzögert werden, beim Hauptalarm sind jedoch wegen des Spannungsabfalls Grenzen gesetzt.

Bezeichnung	Flip-flop	Stellg. bei Ab-speich.	Beschreibung
Weckeralarm	BEEW	44	Der Inhalt des Wecker-registers stimmt mit den Bits 9-24 des Uhr-registers überein. Der Alarm führt nur zum eigenen Rechner-kern.
Rechneralarm	BEER1	23	Unterbricht einen oder mehrere Rechnerkerne, je nach Spezifikation im Y-Befehl.
	BEER2	22	
	BEER3	21	
	BEER4	20	
Hauptalarm	BEEH	19	Netzausfall oder -Aus-schaltung, es wird noch kurzzeitig von einem durch ein Schwungrad angetriebenen Genera-tor Spannung geliefert, die zur Ausführung einer gewissen Zahl von Re-chenoperationen ausreicht, jedoch nicht mehr zur Betätigung von EA-Ge-räten.

### 7.2.3 Alarmsperren

Es gibt zwei Alarmsperren, die durch Flipflopzustände dargestellt werden.

BEFA: Alarmsperre 1

BEFB: Alarmsperre 2

Sie werden unabhängig von Alarmgruppe und Alarmart beim Eintritt eines ersten und eines eventuell darauffolgenden zweiten Alarms gesetzt.

Tritt ein Alarm der Gruppe I auf, während Alarmsperre 2 (und damit auch Alarmsperre 1) gesetzt ist, so führt das zum Rechnerhalt. Ein Alarm der Gruppe II wird verzögert. Bei einer anstehenden Alarmsperre (1 oder 1 und 2) kann kein Eingriff gemeldet werden.

#### 7.2.4 Hardwareseitige Abspeicherungen, Modusumschaltungen und Sprungziele bei Alarmen

Bei Eintreffen eines Alarmes werden in Abhängigkeit von der Stellung des Flipflops BEBN, von Alarmart und einer eventuell bereits anstehenden Alarmsperre die in der folgenden Tabelle aufgeführten Abspeicherungen und Neueinstellungen vorgenommen.



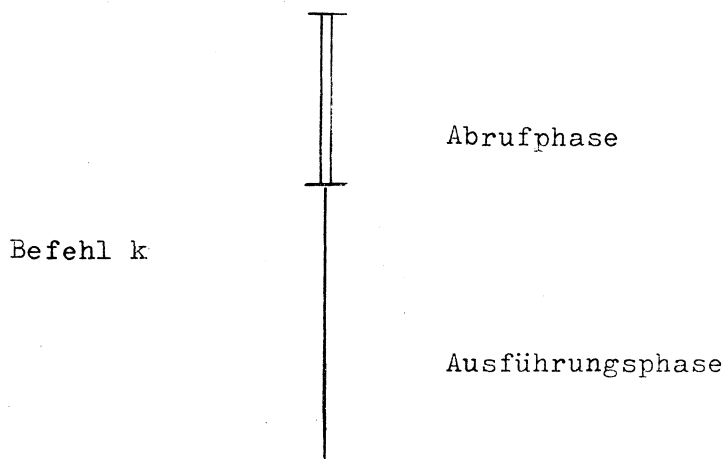
Generell: (die Reihenfolge entspricht nicht dem zeitlichen Ablauf im Mikroprogramm)

1.  $\langle BB \rangle_{1,16} := \langle BL \rangle$  bzw.  $\langle BB \rangle := 0$ , wenn Systemmodus eingestellt war
2. mod 24 wird eingestellt
3. neuer Modus = Systemmodus
4.  $\langle BLZ1, BLZ2 \rangle$  bleibt unverändert
5. - bedeutet: bleibt unverändert

Einstellung bei Eintreffen eines Alarms			Einstellung nach Eintreffen eines Alarms				
Modus bei Alarmfall	Alarm-sperre	Alarm-Gruppe	Alarm-sperre	Ablage d. Befehls-Reg.	$\langle \text{Index-zellen} \rangle :=$	$\langle \text{BxB} \rangle :=$	$\langle F \rangle :=$
Normal- oder Abwickler- oder Spezialmodus	keine	I, II	1	$\langle \langle BL \rangle + 32 \rangle_{ff}$	$\langle BI_{1...4} \rangle$	$\langle 4_2 \rangle$	$\langle 2_2 \rangle$
	1	I	2	$\langle \langle BL \rangle + 40 \rangle_{ff}$	$\langle BI_{1...4} \rangle$	$\langle 4_2 \rangle$	$\langle 3_2 \rangle$
Systemmodus	keine	I, II	1	$\langle 32 \rangle_{ff}$	-	-	$\langle 2_2 \rangle$
	1	I	2	$\langle 40 \rangle_{ff}$	-	-	$\langle 3_2 \rangle$
beliebiger Modus	1 od. 1u.2 2	II I					
der Alarm wird verzögert, bei Alarmsperre 1 aufgehoben ist Rechner-Halt							

### 7.3 Unterbrechung der Befehle

Der Ablauf eines TR440-Befehls setzt sich aus zwei aufeinanderfolgenden Phasen zusammen, der Abrufphase und der Ausführungsphase. Manche Befehle durchlaufen die Ausführungsphase in Form einer Schleife mehrfach oder benutzen in der Ausführungsphase einen Teil der Abrufphase wie ein Unterprogramm.



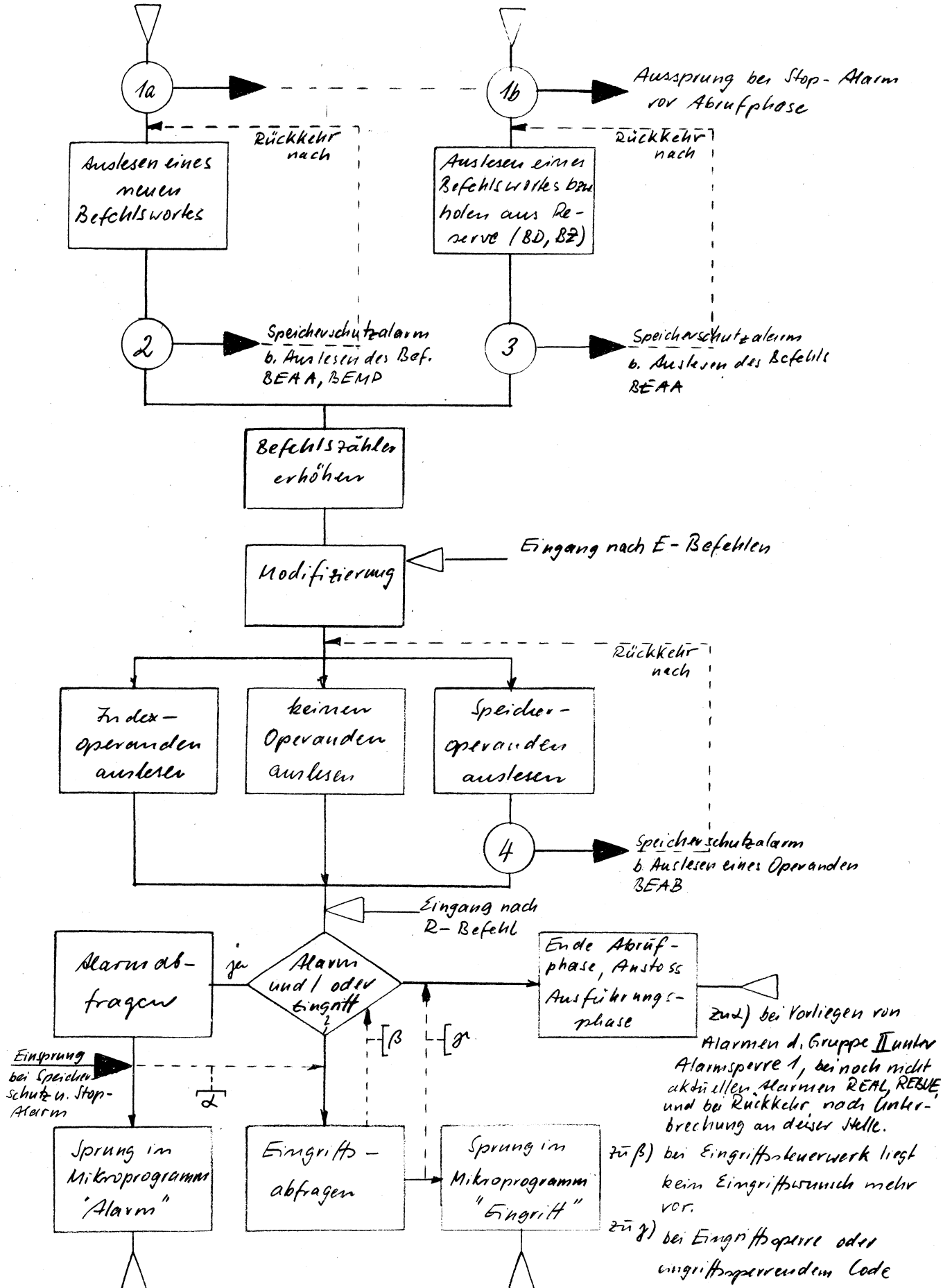
Die Abrufphase ist ein von der Wirkung des einzelnen Befehls weitgehend unabhängiges Mikroprogramm, das nur das Befehlswerk belegt und sich für alle Befehle unter Verwendung verschiedener Eingänge gemeinsam darstellen läßt. Die Ausführungsphase ist durch die Wirkung des einzelnen Befehls bedingt und sie läßt sich nur in sehr grober Form allgemein wiedergeben.

Der zeitliche Ablauf einer Kette von Befehlen hängt von der Belegung von Befehls- und Rechenwerk ab. Belegt der Befehl k in der Ausführungsphase nur das Rechenwerk, so läuft zeitlich parallel dazu die Abrufphase des folgenden Befehls ab und anschließend auch die Ausführungsphase, falls diese nur das Befehlswerk belegt. Benötigt die Ausführungsphase jedoch das Rechenwerk, so kann sie erst gestartet werden, wenn die Ausführungsphase des vorhergehenden, das Rechenwerk belegenden Befehls abgeschlossen ist.

### 7.3.1 Stark vereinfachter Ablauf der Abrufphase

Eingang, wenn BEMP gesetzt ist (Sprungbefehle u. a.)

Normaleingang



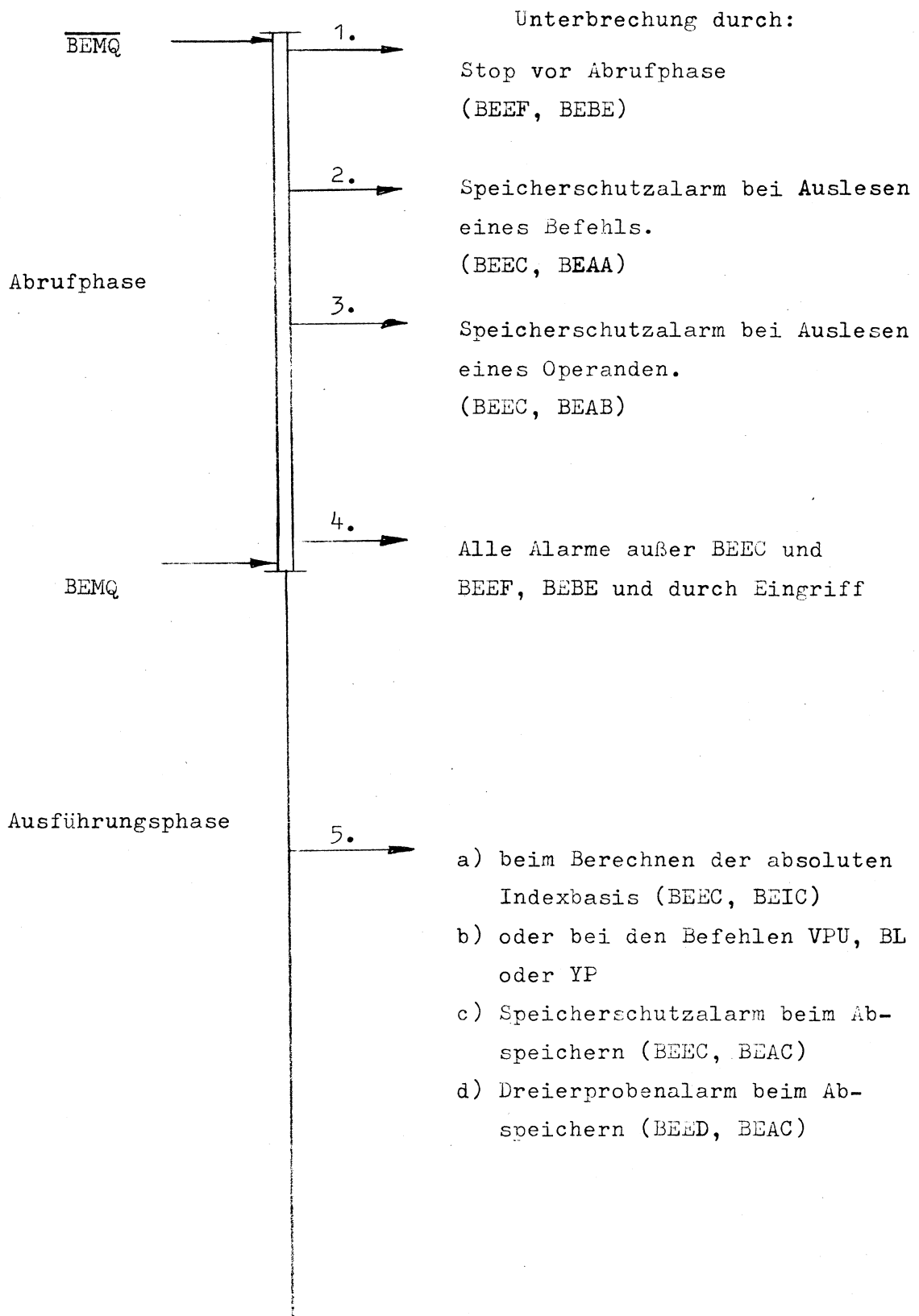
### 7.3.2 Unterbrechungsstellen im Befehlsablauf (Abruf- und Ausführungsphase)

Die Unterbrechungswünsche (Eingriffe und Alarme) werden von den Mikroprogrammen zunächst in Flipflopzuständen festgehalten. Diese werden nur am Ende der Abrufphase abgefragt. Nur bei Auftreten von Speicherschutzalarm (BEEC), Dreierprobenalarm im Abspeichernanoprogramm (BEED, BEAC) und bei dem Stop-Alarm BEEF, der durch den Betriebsmodus BEBE hervorgerufen wurde, wird unmittelbar der Alarmabfragezweig angesprungen. Die Aussprungstelle wird zur eindeutigen Rückkehr in Steuerflipflops vermerkt (BEAA, BEAB, BEAC, BEMP).

1. Steht bei gesetzter Alarmsperre 1 ein Alarm der Gruppe II an, so wird die Ausführungsphase angestoßen. Es wird fortgefahren und der Alarm erst gemeldet, wenn die Alarmsperre 1 aufgehoben ist. Alle anderen Alarme führen zur Unterbrechung, wobei die Alarmsperre 2 gesetzt wird.  
Beachte jedoch:
2. Liegt nur Typenkennungs- und/oder arithmetischer Alarm an, so wird unabhängig von einer bereits anstehenden Alarmsperre nur dann in das Mikroprogramm "Alarm" gesprungen, wenn der anstehende Befehl diesen Alarm nicht umspringt (mit SAT bzw. SAA und Löschen des Alarms) oder wenn er das Rechenwerk belegt. Anderenfalls werden diese Alarme zurückgestellt, und es wird nach anstehenden Eingriffen abgefragt.
3. Stehen keine Alarme an, die zu einer Unterbrechung führen, so wird nach einem anstehenden Eingriffswunsch abgefragt. Der Eingriffswunsch führt nur dann nicht in das Mikroprogramm "Eingriff", wenn die Alarmsperre 1 und/oder die Eingriffssperre gesetzt ist oder ein den Eingriff sperrender Code (nur BC und SSR) ansteht.

4. Im Alarmfall können mehrere Alarme gleichzeitig anstehen. Bei Speicherschutzalarm oder Stopalarm am Beginn der Ab-rufphase werden die zu diesem Zeitpunkt in den Flipflops bereits angezeigten Alarme mit gemeldet. Diese Alarme treten dann früher auf, als sie ohne Entstehen eines Speicherschutzalarms oder Stopalarms gemeldet worden wären.

# Unterbrechungsstellen in einem Befehl

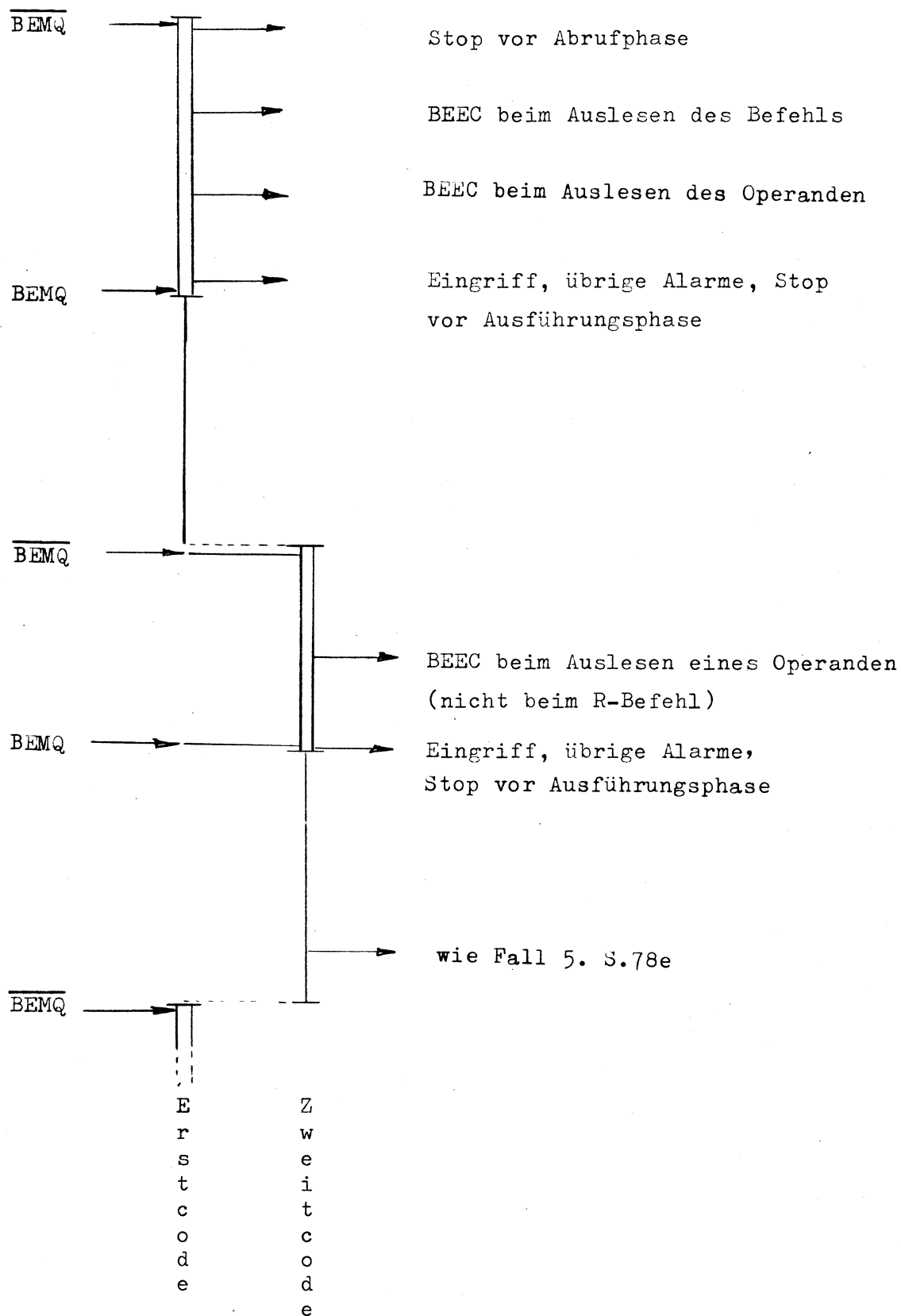


In den Fällen 1, 2, 3, 5 werden die Alarme, die in den Flipflops bereits vermerkt sind, ebenfalls angezeigt.

### 7.3.3 Unterbrechungsmöglichkeiten bei komplexen Befehlen

-Ersetze-Befehle : (E, EMB, EZ, EMU, LEI, MAB, MABI, MU, R, RLR, T) rufen in der Ausführungsphase des Erstcodes den Zweitcode in der Art eines Unterprogramms auf. Dabei entfällt in der Abrufphase des Zweitcodes der Teil des "Mikroprogramms Abrufphase", der den Befehl ausliest und das Befehls-Folge-Register erhöht. Diese Befehle sind zwischen Abruf- und Ausführungsphase des Erstcodes und zwischen Abruf- und Ausführungsphase des Zweitcodes durch Alarme und Eingriffe unterbrechbar. In der Abrufphase des Erstcodes kann Speicherschutzalarm beim Auslesen des Befehls auftreten. In der Abrufphase des Zweitcodes kann mit Ausnahme des R-Befehls je nach Art des Zweitcodes Speicherschutzalarm beim Auslesen eines Operanden entstehen. Schließlich kann in der Ausführungsphase des Zweitcodes noch Speicherschutzalarm gemeldet werden. Der Alarm "Stop vor Abrufphase" unterbricht nicht am Anfang der Abrufphase des Zweitcodes.

# Ersetze-Befehle





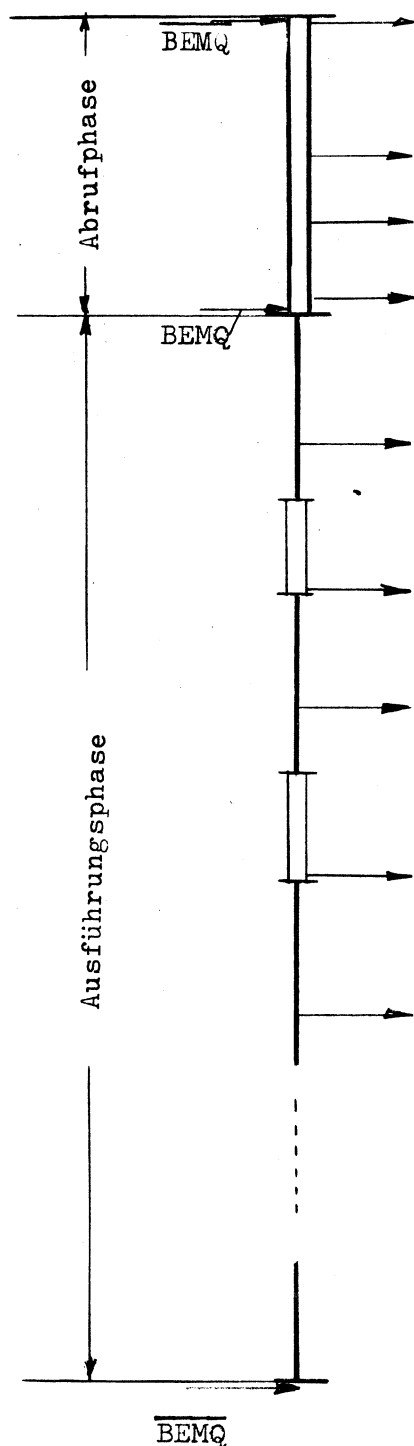
Befehle, die mehrfach Teile der Abruf- oder Ausführungsphase aufrufen

(BZ, BZ2, BZN, CZ, CNZ, MCE, MD, QBR, QCR, TLD, TLI, TLOG, TMAX, TMIN, US, WTR, WTV)

Diese Befehle können in der Ausführungsphase mehrfach einen Teil des "Mikroprogramms Abrufphase" in der Art eines Unterprogramms aufrufen. Vor jedem Verlassen dieses Mikroprogramms wird auf anstehende Alarmer und Eingriffe abgefragt. Je nach Art des Befehls können die verschiedenen Möglichkeiten des Speicherschutzalarms eintreten. Der Stop-Alarm vor Abrufphase kann nur beim ersten Einsprung in die Abrufphase auftreten, der Stop-Alarm vor Ausführungsphase nur nach erstmaligem Durchlaufen der Abrufphase.

Mehrfach durchlaufene Befehle

Unterbrechung durch:



Stop vor Abrufphase

BEEC beim Auslesen des Befehls

BEEC beim Auslesen eines Operanden

Eingriff, übrige Alarme, Stop  
nach Abrufphase

wie Fall 5. S.78e

Eingriff, übrige Alarme, nicht  
Stop nach Abrufphase

wie Fall 5. S.78e

Eingriff, übrige Alarme, nicht  
Stop nach Abrufphase

wie Fall 5. S.78e

⋮

## 8. Hardwareseitige Bedingungen bei Makros und SSR-Befehlen

### 8.1 Leercodes (Makros)

Von den möglichen 256 Interncodes sind nicht alle durch fest verdrahtete Maschinenbefehle belegt. Die freien Codes (Leercodes) können zur Programmierung von Makros verwendet werden. Soweit zum derzeitigen Stand Externcodes von Makros bekannt sind, sind sie in die folgende Aufstellung aufgenommen.

Einige Eigenschaften der zu schaffenden Makros sind durch Bedingungen, die an den Interncode geknüpft sind, hardwaremäßig vorgegeben.

## 8.2 An den Interncode geknüpfte Bedingungen

Bei der Befehlsvorentschlüsselung eines Makros werden grundsätzlich Befehls- und Rechenwerk belegt.

Weitere Bedingungen sind an die folgenden Spezifikationen geknüpft:

E = L bedeutet: auf die Adresse des Makro wird nicht mod2 addiert.

E = 0 bedeutet: die Adresse des Makro wird durch mod2 verändert .

A = L bedeutet: entsprechend der angegebenen Adresse wird in der Befehlsvorentschlüsselung ein Speicheroperand aus dem Arbeitsspeicher in das Sammelregister ausgelesen. Dieser Speicheroperand ist zwar für die beabsichtigte Wirkung des Makro ohne Bedeutung, kann jedoch zu Speicherschutzalarm führen.

A = 0 bedeutet: es wird kein Arbeitsspeicheroperand ausgelesen.

X = L wie A = L, jedoch für Indexspeicheroperanden.

X = 0 bedeutet: es wird kein Indexspeicheroperand ausgelesen.

### 8.3 Liste der Leercodes

- bedeutet:Leercode

/ bedeutet:belegt durch Maschinenbefehl, s. "Wirkung der Befehle"

Interncode	Externcode		Befehlsvorentschlüsselung	
	Mod16	Mod24	E	AX
21	WTZ	/	L	00
22	WTK	/	L	0L
23	WTD	/	L	0L
3B	/	-	O	00
3C	-	/	L	LO
3D	-	/	L	LO
50	FW	-	O	LO
51	GW	-	O	LO
67	-	/	O	LO
9E	VRL	-	O	00
AA	SKB	/	O	00
AB	SGB	/	O	00
BO	TAD	/	O	00
B1	TDA	/	O	00
B2	-	/	L	00
B3	-	/	L	00
B4	-	/	L	00
B5	-	/	L	00
B6	-	/	L	LO
B7	-	/	<del>L</del>	<del>LO</del>
BC	-	/	L	LO
BD	-	/	L	LO
BE	-	/	O	00
CO	DML	-	O	00
C1	DMN	-	O	00
C2	DDV	-	O	00

Interncode	Externcode		Befehlsvorentschlüsselung	
	Mod16	Mod24	E	AX
C3	DDD	-	O	OO
C4	DAC	-	O	OO
C5	DSC	-	O	OO
C6	DA	-	O	OO
C7	DSB	-	O	OO
CD	HN	-	L	OO
CE	-	/	L	OO
CF	-	/	L	OO
DO	-	/	O	OO
D1	-	/	O	LO
D2	-	/	O	OO
D3	-	/	O	LO
D4	-	/	O	OO
D5	-	/	O	OO
D6	-	-	O	OO
D7	-	-	O	OO
DE	DR	-	O	OO
EO	-	/	L	OO
E1	VRK	/	L	OO
E2	LK1	-	L	OO
E3	LK2	/	L	OL
E4	FPOL	-	L	OO
E5	GPOL	-	L	OO
E6	FMSK	/	L	OL
E7	GMSK	/	L	OL
E8	-	-	L	LO
E9	-	-	L	LO
ED	TLB	/	L	LO
FO	HA	DA	L	LO
F1	HSB	DSB	L	LO
F2	HAU	DML	L	LO
F3	HSU	MLD	L	LO
F4	-	/	O	LO


ER/E441  
I/24/67  
160567

Interncode	Externcode		Befehlsvorentschlüsselung	
	Mod16	Mod24	E	AX
F5	-	/	O	LO
F6	-	/	O	LO
F7	-	/	O	LO
F8	HNM	WB	L	OO
F9	HWZ	-	L	OO
FA	HML	-	L	OO
FB	HWH	-	L	OO
FC	HB	-	L	OO
FD	HC	-	L	OO

#### 8.4 Hardwareseitige Vorgänge bei Erkennen eines Leercodes (MAKRO)

Die Abspeicherungen und Neueinstellungen bei Erkennen eines Leercodes sind von der Stellung des Flipflops BEBY abhängig.

- Generell:
1.  $\langle BB \rangle_{17,24} : o = \langle BC \rangle$
  2. modus 24
  3. BEFE = L (Der Eingriff wird gesperrt)
  4.  $\langle BLZ1, BLZ2 \rangle$  bleiben unverändert

Bei Erkennen des Leercodes ist ein- gestellt:	Neuer Modus	Befehlswerk- register	$\langle F \rangle :=$	$\langle \text{Indexzelle} \rangle :=$	$\langle \text{BXB} \rangle :=$
Normal- oder Abwicklermodus	Spezialmodus	$\langle BL \rangle + 16_{ff}$	$\langle 7 \rangle_2$		$\langle 4 \rangle_2$
Spezial- oder Systemmodus	Systemmodus	$\langle 16 \rangle_{ff}$	$\langle 7 \rangle_2$	$\langle BI_{1, \dots, 4} \rangle$	$\langle 4 \rangle_2$



# 8.5. Hardware-seitige Vorgänge bei Erkennen eines SSR-Befehls

- Generell: 1.  $\langle BB \rangle := \langle BA \rangle$   
 2. Mod 24 wird angesetzt  
 - : keine Änderung

Bei Erkennen des SSR ist angestellt	neuer Modus	Eingangs- sperrung	Befehls- werksre- gister	<Index- zeile>:=	<BxB>:=	<BLZ1,2>:=	<F>:=
Normalmodus	Abwickler- modus	-	$\langle \langle BL \rangle + 16 \rangle_{ff}$	-	-	$\langle \langle BL \rangle + 65 \rangle_{3,12}$	$\langle \langle BL \rangle + 6 \rangle_2$
Abwicklermodus	Spezial- modus	+	$\langle \langle BL \rangle + 8 \rangle_{ff}$	-	-	-	$\langle 6_2 \rangle$
Spezial- und Systemmodus	System- modus	-	$\langle 8 \rangle_{ff}$	$\langle BI_{1-4} \rangle$	$\langle 4_2 \rangle$	-	$\langle 6_2 \rangle$

#### 8.6. Grundzustand im TR440

Durch Eingreifen am Bedienfeld des TR440 kann ein ausgewählter Rechnerkern aus einem beliebigen Zustand dieses Rechnerkerns in einen definierten Ausgangszustand versetzt werden. Dieser Zustand heißt "Grundzustand".

1. Im Grundzustand ist der zentrale Takt abgeschaltet und ein Anzeigetakt eingeschaltet.
2. Alle signifikanten Steuerflipflop im ausgewählten Rechnerkern, im EA-Befehlswerk und in den Kanalwerken werden gelöscht. Die Flipflop  
BEEH (Hauptalarm)  
BEFT (techn.Ausführungsfehler)  
BEBY (Systemmodus)  
werden gesetzt.
3. Bei eingeschalteter Anzeige erscheint der Zustand der Register und der anzeigbaren Steuerflipflop des ausgewählten Rechnerkerns auf dem Anzeigefeld.
4. Der Grundzustand wird nur erreicht nach Auswahl eines Rechnerkerns mit Hilfe der Schalter RK1 bis RK4 bzw. RS für alle Rechnerkerne und Drücken der Taste "Grundzustand".
5. Der Eintritt in den Grundzustand kann durch keine der vorhandenen Hardware-Sperren verhindert oder verzögert werden.
6. Der Grundzustand wird durch Lösen der Taste "Grundzustand" verlassen. Die Anzeige wird abgeschaltet, sofern sie angeschaltet war. Der zentrale Takt wird wieder eingeschaltet.

7. Über einen Zweig der Abrufphase wird das Alarm-Mikroprogramm angesprungen. Die Register werden gemäß dem eingestellten Systemmodus in Kachel 0 abgelegt, s.S.78. Nach Setzen der Alarmsperre 1 wird die Zelle 2 in Kachel 0 angesprungen.

Anmerkung: Bei den z. Zt. in Prüfung befindlichen TR440-Anlagen erfolgt die Auswahl des Rechnerkerns nur durch den Schalter RK 1. Schalter RS darf noch nicht betätigt werden. Bei Anlagen mit mehreren Rechnerkernen können die Flipflops des EA-Werkes und des Eingriffs-Werkes nur bei eingelegter Taste RS gelöscht werden.

#### 8.6.1. Alarm "Technischer Fehler"

Bei Auftreten eines Fehlers in einem Mikroprogramm laufen folgende Vorgänge ab:

1. Alle signifikanten Steuerflipflop des betreffenden Rechnerkerns werden gelöscht.

2. Es werden die Flipflop

BEFT (techn.Ausführungsfehler)

BEFA (Alarmsperre 1)

BEFB (Alarmsperre 2)

gesetzt.

3. Das Alarm-Mikroprogramm wird angesprungen und läuft infolge der beiden gesetzten Alarmsperren auf HALT.

4. Der Rechnerhalt kann aufgehoben werden

a) durch Einlegen der Taste "Grundzustand" s.S.86

b) durch Einlegen der Taste "Prüfkanal" s.u.

#### 8.6.2. Normierung

Beim Einlegen der Taste "Prüfkanal" werden die signifikanten Steuerflipflop im zentralen Rechner gelöscht. Der zentrale Takt wird abgeschaltet und der Anzeigetakt eingeschaltet. \*

Nach Lösen der Taste "Prüfkanal" wird die Anzeige abgeschaltet und der zentrale Takt eingeschaltet. Das Mikroprogramm "Primitivkanal" wird angesprungen. Der Primitivkanal befindet sich im Wartezustand zur Eingabe von Zeichen über Schreibmaschine oder Lochstreifen.

\* Zur Zeit werden dabei auch die signifikanten Steuer-Flip-Flop im EA-Werk und im Eingriffswerk gelöscht.