

Die Operationen, welche die Register benutzen (s.3.3.), erfolgen immer in fixer Länge, d.h. je nach gewählter Kapazität auf 5 oder 10 Bytes. Darum wird, wenn eine Zahl nicht 9 oder 19 Stellen erreicht, ihr Vorzeichen so oft wiederholt, wie die Kapazität des Registers es erlaubt.

Außer ihrer Verwendung bei der Verarbeitung von Werten, dienen die Register 0 - 9 auch zum Speichern von speziellen Adressen (s.2.1.4.). Diese Adressen, aus 4 Dezimalziffern bestehend, sind in gepackter Form auf den zwei rechten Bytes des Registers gespeichert. Die restlichen drei Bytes können als Normalspeicher benutzt werden (s. 2.1.3.4.). In Abschnitt 3.6.5 sind Beispiele dieser Benutzung der numerischen Register gegeben.

Anmerkung:

Ein Programm benutzt nicht immer alle 100 Register. Die nicht benutzten Register können als Normalspeicher Verwendung finden.

2.1.3.4. Normalspeicher

Von Stelle 0596 an oder bereits vorher, wenn die numerischen Register nicht benutzt wurden, ist der Speicher nicht spezialisiert. Die Einteilung dieses Speicherbereichs wird vom Programmierer bestimmt, und zwar nach Methoden, die in Abschnitt 2.1.4. beschrieben werden.

Im allgemeinen wird der Speicher enthalten:

- das durchzuführende Programm
- die zu verarbeitenden Daten und die auszugebenden Ergebnisse
- in gewissen Fällen Zusatzinformationen (Konstanten, Tabellen, Zwischenergebnisse etc.).

Alle in 2.1.2.2 angegebenen Arten der Einspeicherung sind im Normalspeicher möglich. Die Informationen werden in wirklicher Länge durch Befehle, die auf Zeichenbasis arbeiten, verarbeitet (s.3.4. - 3.7.). Hauptsächlich werden in diesem Speicherbereich die Ein- und Ausgaben der Informationen, ihre Umschlüsselung, logische Operationen und die Bereitstellung der Daten durchgeführt.

2.1.4. Die Adressierung im Zentralspeicher

2.1.4.1. Prinzipien

Um eine zusammenhängende Gruppe von Bytes anzusprechen, z.B. durch einen Programmbefehl, muß die Zentraleinheit ihren genauen Platz im Kernspeicher kennen.

Der Anfang einer Gruppe wird im allgemeinen durch die Adresse des ersten Bytes von rechts oder links, je nach Ablafrichtung des Befehls, angegeben.

Um das Ende einer Gruppe anzuzeigen, gibt es zwei Möglichkeiten:

1. Die Maschine kennt die Zahl der zu bearbeitenden Bytes.

Es handelt sich hier um spezialisierte Zonen von fester Länge, die durch bestimmte Befehle angesprochen werden (numerische Register beispielsweise). Durch die Analyse des Operationstyp des Befehls erkennt die Zentraleinheit welcher Art die Zone ist und kann daraus die Länge ableiten.

Die Anfangsadresse der Bytegruppe wird durch den Programmierer festgelegt und reicht aus, die Einspeicherung exakt zu beenden.

2. Die Maschine kennt nicht die Zahl der zu bearbeitenden Bytes.

Hierbei handelt es sich um Gruppen von beliebiger Anzahl Bytes, die Normalspeicherzonen bilden.

Außer der Anfangsadresse hat der Programmierer auch noch die Zahl der zu verarbeitenden Bytes zu bestimmen. Zwei Methoden ergeben das gewünschte Resultat:

- a) Die erste besteht darin, die Anzahl der Bytes im Befehl selbst festzulegen. Dieser kann dadurch die bereits bearbeiteten Bytes abzählen bis Gleichheit mit der angegebenen Zahl erreicht wird.

Diese Art wird bei den Befehlen angewandt die auf Zeichenbasis arbeiten. Allgemein werden nur kleinere Gruppen davon betroffen (logische Operationen, Speicherüberträge etc.)

- b) Die zweite Methode besteht darin, die Gruppe von Bytes durch ein Spezialzeichen zu begrenzen, der sogenannten "Trennmarke". Diese wird während des Ablaufs des Befehls festgestellt und beendet ihn.

Diese Art wird bei den Befehlen angewandt, die komplette Zonen bearbeiten. Diese Zonen sind im allgemeinen umfangreicher (Ein/Ausgabeoperationen, Umschlüsselungen etc.).

2.1.4.2. Die Adressierung der Speicherzonen

Im Abschnitt 2.1.1.1 haben wir gesehen, daß jede Stelle des Zentral-speichers einzeln adressierbar ist. Sie wird darum von 0 bis 2499, 4999 oder 9999 numeriert. Die Nummer eines Bytes ist gleichzeitig seine Adresse.

Im Speicher stehen alle Adressen in fester Länge auf 2 Byte gepackt (vier Dezimalziffern).

Beispiel: Adresse 89 : 00 89

Die Halbbytes einer Stelle sind nicht getrennt adressierbar.

Zur Organisation der Arbeit teilt der Programmierer den Zentralspeicher in Zonen ein, deren Zahl, Bedeutung und Belegung von der jeweiligen Arbeit abhängen. Im allgemeinen müssen, außer den reservierten Zonen am Anfang des Speichers (s.2.1.3.1.) und der für das Programm bestimmten Zone, folgende Zonen vorgesehen werden:

- eine Registerzone für die Rechenoperationen
- eine oder mehrere Ein/Ausgabezonen für den Austausch mit den Rand-einheiten.
- eine oder mehrere Arbeitsfelder zum Zwischenspeichern von vorläufigen Ergebnissen.
- eine oder mehrere Speicherzonen, um permanente Daten am Anfang der Arbeit einzuspeichern (Konstanten).

Um diesen Erfordernissen Rechnung zu tragen, wird der Zentralspeicher des GE-55 in mehrere Zonen mit einer Länge von 1 - 999 Stellen eingeteilt. Unter diesen Zonen hat man nun die spezialisierten Registerzonen und die Normalspeicherzonen zu unterscheiden.

Die Registerzone (s.2.1.3.)

Die Registerzone enthält:

- an den Stellen 85 - 95 Spezialregister
- an den Stellen 96 - 595 numerische Register

Die Registerzone kann verkleinert werden, wenn das Programm nicht alle 100 numerischen Register benötigt.

Adressierung der Spezialregister:

Man greift auf diese Register in der gleichen Weise zu wie auf eine Normalspeicherzone.

Adressierung der numerischen Register:

Die numerischen Register sind Zonen von fixer Länge (standardmäßig 5 Bytes), die ganz bestimmte Plätze im Zentralspeicher belegen.

Um auf eins von ihnen zuzugreifen, gibt der Programmierer lediglich die Nummer des Registers an (00 - 99). Die Zentraleinheit rechnet nun

automatisch die Adresse A der rechten Stelle nach folgender Formel aus:

$$A = 5 R + 100$$

Beispiel: Das Register 00 belegt die Stellen 96 - 100.
Seine reelle Adresse ist: 0100.

Werden Befehle verwendet, die Doppelregister ansprechen, wird die Nummer des rechten Registers angegeben.

Beispiel: Das Doppelregister 17 besteht aus den beiden Einfachregistern 16 und 17 und belegt die Stellen 176 bis 185. Seine reelle Adresse ist: 0185.

Solange man die Register in der eben besprochenen Weise anwendet, ist es nicht möglich auf eine einzelne Stelle eines numerischen Registers zuzugreifen. Das ist erst möglich, wenn man die Registerzone als Normalspeicherbereich ansieht.

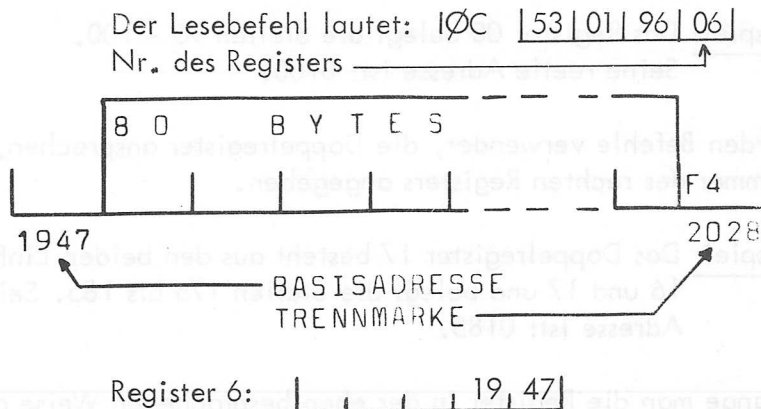
Die Normalspeicherzonen (s.2.1.3.4.)

Die Einteilung, Belegung und Anwendung der Normalspeicherzone liegt im Belieben des Programmierers.

Die Zentraleinheit lokalisiert eine Normalspeicherzone, indem sie ein numerisches Register untersucht, in dem die Anfangsadresse dieser Zone gespeichert ist. Diese Adresse, auch Basisadresse genannt, ist genau genommen die Adresse des Bytes, das links vor der Zone liegt.

Der Programmierer belegt für jede angelegte Zone ein numerisches Register und benutzt dessen Nummer in den Befehlen, die diese Zone ansprechen. Die Basisadressen werden also in einem Programm nicht erwähnt. Während des Programmladens werden sie in den Kernspeicher wie Konstanten eingegeben. Bei Ausführung des Programms werden die Basisadressen dann aufgrund der in den Befehlen erwähnten Registernummern eingesetzt.

Beispiel: Die Zone für den Kartenleser ist 80 Stellen groß und belegt z. B. die Stellen 1948 bis 2027; die Basisadresse (1947) wird in dem Register 6 gespeichert.



Wenn eine Zone von einem Befehl angesprochen wird, der komplette Zonen bearbeitet (Ein/Ausgabebefehl, Umschlüsselungsbefehle; s. 3.6. und 3.7.) muß sie durch eine Trennmarke F4 rechts von dem letzten benutzten Byte (s. obiges Beispiel) begrenzt werden. Diese Befehle können als Basisregister ein beliebiges Register von 00 - 99 bearbeiten.

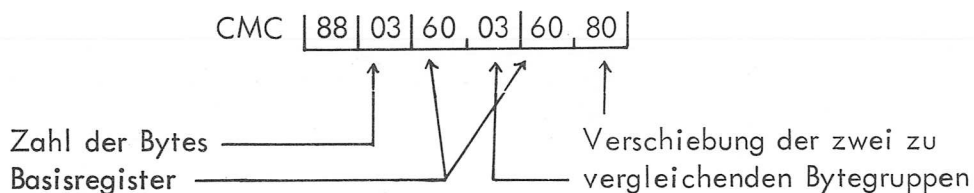
Die anderen Befehle (Bearbeitungen und Überträge innerhalb des Speichers), die auf Zeichenbasis arbeiten, können dagegen nur die Register 0 - 9 verwenden (s.3.4.). Infolgedessen kann auch eine Ein/Ausgabezone, deren Inhalt später weiterverarbeitet werden soll, nur durch die Register 0 - 9 definiert werden.

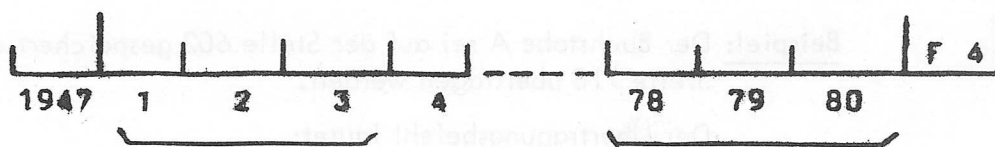
Um auf ein Byte einer Normalspeicherzone zuzugreifen, gibt der Programmierer das Basisregister dieser Zone und die "Verschiebung" an, d.h. die Stelle in Bezug auf den Anfang der Zone. Zu diesem Zweck werden die Bytes einer Zone von 1 bis n von links an numeriert. Um auf mehrere zusammenhängende Bytes der gleichen Zone zuzugreifen, wird die Verschiebung (=Stelle) des rechten Bytes und die Zahl der Bytes der betr. Gruppe angegeben.

Um die reelle Adresse zu erhalten, addiert die Zentraleinheit die Basisadresse und die Verschiebung.

Beispiel: Die ersten 3 Bytes der Lesezone sollen mit denen der letzten Bytes der gleichen Zone verglichen werden.

Der Vergleichsbefehl lautet:





Die spezialisierten Zonen am Anfang des Speichers.

Der Programmierer kann die ersten 999 Bytes des Zentralspeichers vergleichen mit einer Normalspeicherzone, deren Basisadresse "Null" ist. Die Verschiebungen entsprechen in diesem Fall den reellen Adressen. Das dafür benutzte Basisregister ist das Register 00 (s. Beispiel in 2.1.4.3.).

Auf diese Weise hat man mit Befehlen, die auf Zeichenbasis arbeiten, Zugriff auf die spezialisierten Register sowie auf die Zahlen, die sich in den numerischen Registern befinden, in reeller Länge (natürlich nur in vollen Bytes).

2.1.4.3. Die Adressierung der Informationen

Der vorherige Abschnitt hat die verschiedenen adressierbaren Einheiten des Zentralspeichers dargestellt, d.h. quasi die "Behälter" der Informationen: Byte, Register, Zone.

Es ist jetzt zu zeigen, in welcher Weise man auf Informationen zugreift, d.h. auf den "Inhalt" dieser Speichereinheiten.

Vorab ist es jedoch nützlich, sich nochmals die Struktur der Information im Speicher und die der verschiedenen Einheiten mit denen man arbeitet, ins Gedächtnis zurückzurufen. Obwohl verschlüsselt, behält die vom Rechner verarbeitete Information doch ihre ursprüngliche Struktur. Sie setzt sich, genau wie ein Schriftstück, aus "Wörtern" zusammen, die wiederum aus einem oder mehreren "Zeichen" bestehen, die dem Zeichenvorrat des Druckers entnommen sind.

Adressierung der Zeichen

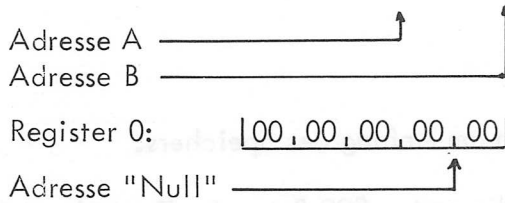
Im Abschnitt 2.1.2 haben wir gesehen, daß ein Byte entweder 1 Zeichen im ISO-Code oder 2 Zeichen (meistens Ziffern) gepackt im internen Code, enthalten kann.

Belegt ein Zeichen nur ein Byte, ist seine Adresse gleich dem Byte, das dieses Zeichen enthält.

Beispiel: Der Buchstabe A sei auf der Stelle 602 gespeichert und soll nach Stelle 718 übertragen werden.

Der Übertragungsbefehl lautet:

MVC |80,01|06,02|07,18|



Die Halbbytes einer Stelle sind nicht adressierbar. Belegen zwei Zeichen dasselbe Byte, kann man nicht direkt auf eines von ihnen zugreifen. Man muß vorher mit dem entsprechenden Befehl diese zwei Zeichen auf zwei Bytes entpackt übertragen.

Diese Arbeitsweise und die Einschränkungen des Abschnitt 2.1.4.2 gelten für alle Bytes, die Zeichen enthalten.

Adressierung der Worte

Definitionsgemäß ist ein " Wort" eine Reihe von Zeichen. Die Zahl ist variabel, aber bekannt. Auf ein Wort von gegebener Länge kann man zugreifen, wenn es sich in einer Normalspeicherzone oder einer vergleichbaren Zone (Registerzone als Normalspeicher) befindet. Der Zugriff erfolgt in der gleichen Weise, wie es im vorigen Abschnitt für eine Gruppe von Bytes beschrieben wurde; eine Gruppe von Bytes enthält das in Betracht kommende Wort.

Man kann dagegen nicht in reeller Länge auf ein Wort zugreifen, das sich in einem numerischen Register befindet, indem man einen Befehl verwendet, der eine Registeroperation beinhaltet (s.3.3). Diese Befehle arbeiten in fixer Länge, d.h. mit dem gesamten Inhalt des Registers.

2.1.5. Der Informationsfluß

2.1.5.1. Die verschiedenen Arten des Informationsflusses

Der Informationsfluß im GE-55 kann verschiedener Art sein, je nach Natur der abgebenden oder empfangenden Organe. Dieser Fluß erfolgt durch eine große Anzahl von Übertragungsbefehlen, wovon gewisse, stark symbolisiert, die Programmierung vereinfachen und die Leistung dieser Organe optimieren.

Um einen Überblick über die verschiedenen Arten des Informationsflusses zu geben, soll ein bestimmtes Problem untersucht werden, z.B. ein Problem, das den Kartenleser, den Kartenstanzer und den Drucker benutzt.

Um dieses Problem zu lösen und unter Berücksichtigung dessen, was in den vorhergehenden Abschnitten gesagt wurde, belegt man 5 Zonen im Zentral-
speicher:

- eine Registerzone für das Rechnen und die Adressierung der Zonen und
- vier Zonen im Normalspeicherbereich, nämlich eine Zone für den Kartenleser (Eingabezone) zwei Zonen für den Kartenstanzer und den Drucker (Ausgabezone)
eine Zone zur Zwischenspeicherung und zur Bearbeitung.

Der Ablauf kann folgendermaßen schematisiert werden:

1. Den Inhalt einer Karte, die vom Kartenleser gelesen wurde, in die Eingabezone des Kernspeichers übertragen. Je nach Art werden die eingelesenen Angaben einer oder mehreren Verarbeitungen unterworfen:
 - . logische Analyse in der Eingabezone selbst (Klartext, Kennzeichen, Angaben zur Berechnung etc.)
 - . Bereitstellung in einer oder beiden Ausgabezonen (Klartext, Kennzeichen, Rechenresultate etc.)
2. Zwischenspeicherung in einer Bearbeitungszone für nachträgliche oder mehrmalige Benutzung (Klartexte, die auf mehrere Blätter übertragen werden sollen etc.)
3. Berechnungen in den numerischen Registern (numerische Daten).
4. Das Resultat der Berechnungen, das sich in den Registern befindet, wird dann in die eine oder andere oder beide Ausgabezonen übertragen. In gewissen Fällen kann auch hier ein Übertrag in eine Bearbeitungszone zur weiteren Verwendung vorgenommen werden.
5. Wenn eine Anzahl Informationen, die einer Karte oder einer Druckzeile entsprechen, sich in den Ausgabezonen befinden, werden sie an den Kartenstanzer oder den Drucker übertragen.

Die Informationen müssen also folgendermaßen übertragen werden:

- von einer Randeinheit (Leser) zu einer Eingabezone;
- von einer Eingabezone nach ein oder zwei Ausgabezonen;
- von einer Eingabezone nach einer Bearbeitungszone;
- von einer Eingabezone nach einem Register;
- von einem numerischen Register nach ein oder zwei Ausgabezonen;

- von einer Bearbeitungszone nach einer Ausgabezone;
- von einer Ausgabezone nach einer Randeinheit (Stanzer, Drucker).

Ergänzend zu diesem Informationsfluß erfolgen noch zusätzliche interne Überträge, entweder von den Bearbeitungszone oder den Registern her.

Alle Informationsbewegungen erfolgen Zeichen für Zeichen. Sie unterscheiden sich in ihrem Ablauf und durch ihre jeweiligen zusätzlichen Funktionen je nach der empfangenden Zone dieser Information. Bei der Durchführung der Überträge müssen in erster Linie die Überträge von der Zentraleinheit zu den Randeinheiten von dem internen Austausch im Kernspeicher unterschieden werden.

2.1.5.2. Die Ein- und Ausgabe in die bzw. aus der Zentraleinheit

Die Informationen von und nach einer Randeinheit (Kartenleser, Stanzer, Drucker) werden in der gleichen Reihenfolge gespeichert, wie sie sich auch auf der Randeinheit befinden. Übertragen wird von links nach rechts in aufsteigender Reihenfolge der Adressen der Zone, die der Randeinheit zugeteilt ist. Das Ende des Übertrags wird durch eine Trennmarke angezeigt.

Der Programmierer definiert die anzusprechende Randeinheit durch einen Spezialcode und das Register, in dem sich die Basisadresse der Zone befindet. Die Länge der Überträge braucht nicht angegeben zu werden. Diese Überträge werden durch den Ein-/Ausgabebefehl kontrolliert (s.3.6.).

2.1.5.3. Die Überträge im Innern des Zentralspeichers

Die Informationsworte können, wie bereits gesagt wurde, einzeln übertragen werden. Der Übertrag eines Wortes oder einer Wortgruppe geschieht in absteigender Reihenfolge der Adressen, d.h., von rechts nach links. Die Anzahl der zu übertragenden Zeichen wird im Befehl angegeben.

Für die Adressierung wendet der Programmierer die in Abschnitt 2.1.4.3 gegebenen Regeln an. Es soll aber nochmals betont werden, wie einfach die Ein- und Ausgabezonen adressiert werden können: denn die Informationen in diesen Zonen befinden sich in der gleichen Reihenfolge im Speicher, wie sie sich auch an den Randeinheiten befunden haben. Um ein Wort in der dem Kartenleser zugeteilten Zone zu adressieren, braucht der Programmierer nur das Kartenbild der eingelesenen Karte nachzusehen und anzugeben:

- als Verschiebung die Nummer der rechten Spalte des Wortes in der Karte;
- als Länge die Anzahl der von diesem Wort belegten Spalten.

Dasselbe gilt auch für die Ausgabezonen, indem der Kartenvordruck für den Kartenstanzer oder das zu beschriftende Formular für den Drucker zu Rate gezogen wird.

Zusatzfunktionen

Bestimmte Übertragungsbefehle sind mit einem Formatwechsel der Informationen verbunden. Das gilt aber ausschließlich für numerische Informationen.

Übertrag einer Zone aus dem Normalspeicherbereich in ein numerisches Register.

In einer Zone des Normalspeicherbereichs, z.B. Eingabezone, steht ein Rechenfaktor in seinem absoluten Wert. Er ist in reeller Länge und in ungepackter Form eingespeichert. Wird dieser Operand in ein numerisches Register übertragen, erfolgt die Einspeicherung dort in fixer Länge und in gepackter, algebraischer Form (s. 2.1.3.3.). Diese Umwandlung erfolgt durch die Maschine unter Beobachtung des Vorzeichenregisters. Der Programmierer muß also vor dem Übertrag, je nach dem Vorzeichen des Wertes, dieses Register entsprechend belegen (s. 2.1.3.2.).

Übertrag eines numerischen Registers in eine Normalspeicherzone.

Im Gegensatz zu dem vorher Gesagten muß ein Wert, der in algebraischer, ungepackter Form in einem numerischen Register steht, beim Übertrag in eine Normalspeicherzone in absoluter Größe umgeformt werden. Die Maschine gibt das Vorzeichen im Vorzeichenregister an. Nur der benutzbare Wert wird übertragen und in dieser Zone in ungepackter Form gespeichert. Nullen werden links durch Leerzeichen ersetzt.

Einfügen von Zeichen

Es besteht die Möglichkeit, in einen nach der Ausgabezone (Druckzone) übertragenen Wert ein Zeichen einzufügen (Komma, Punkt etc.). Dieser Wert kann sowohl aus dem Normalspeicherbereich wie auch aus einem numerischen Register kommen.

Die Befehle

Es gibt drei Arten von Übertragungsbefehlen für Überträge innerhalb des Zentralspeichers: Mehrfachübertragungsbefehle, Übertragungsbefehle im Normalspeicherbereich und Registerüberträge.

Mehrfachübertragungsbefehle (s.3.5.)

Diese Befehle übertragen Informationen von Zone nach Zone, d.h.

- von einer Zone im Normalspeicherbereich (z.B. Eingabezone) nach einer anderen Zone im Normalspeicherbereich (z.B. Ausgabezone);
oder
- von einer Zone im Normalspeicherbereich nach der Registerzone und umgekehrt.

Die Mehrfachübertragungsbefehle können, wie ihr Name sagt, in demselben Befehl einen Wert (im Normalspeicherbereich) oder ein Resultat (in der Registerzone) in eine oder mehrere empfangende Zonen übertragen. Diese Befehle haben folgende Zusatzfunktionen:

Zur Adressierung der Zonen im Normalspeicherbereich benutzen sie für die abgebenden Zonen die numerischen Register 6 und 7 und für die empfangenden Zonen die Register 8 und 9.

Diese Zuteilung erlaubt dem Programmierer nur die Verschiebungen der Worte in den Zonen anzugeben, die Basisregister sind bereits automatisch durch den Befehl ausgewählt.

Die Übertragungsbefehle im Normalspeicherbereich (s. 3.4.3.)

Sie erlauben eine Information innerhalb einer oder zwischen zwei Zonen zu übertragen, wenn der Mehrfachübertragungsbefehl nicht benutzt werden kann. Während dieses Übertrags wird die Information nicht umgeformt. Deshalb ist eine besondere Anwendung dieser Befehle der Übertrag einer Zahl in algebraischer, gepackter Form in eine Zone des Normalspeicherbereichs. Die Registerzone wird als Normalspeicherzone betrachtet (s.2.1.4.).

Um dieselbe Zahl vom Normalspeicherbereich nach einem Register zu übertragen, wird ein Spezialbefehl gebraucht. Mit 2 Spezialbefehlen kann man packen oder entpacken, ohne den Wert der Zahl zu verändern.

Die Registerüberträge (s.3.3.)

Mit ihnen erfolgt der Übertrag einer Zahl von einem Register in ein anderes, von einem Einfachregister in ein Doppelregister und der Austausch des Inhalts zweier Register.

2.1.6. Der Programmablauf

Der Programmablauf wird durch das Programmadressregister (PAR) bestimmt. Nach dem Ende eines jeden Befehls untersucht die Zentraleinheit dieses Register, um die Adresse des nächsten Befehls zu übernehmen. Am Anfang der Arbeit befindet sich die Anfangsadresse des Programms in diesem Speicher. Diese Adresse wird am Ende des Programmladens in das Register übernommen.

Die Zentraleinheit läßt die Befehle in der Reihenfolge ihrer Ein-
speicherung im Zentralspeicher ablaufen, d.h. in aufsteigender Reihen-
folge ihrer Adressen. Jedesmal, wenn ein Befehl übernommen wurde,
wird seine Länge dem Inhalt des PAR zuaddiert und ergibt so die Adresse
des nächsten Befehls.

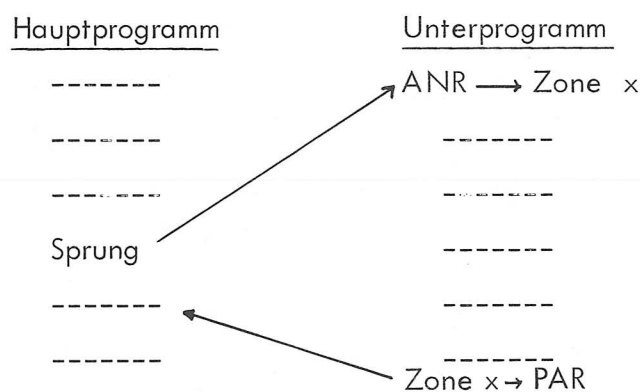
Dieser Ablauf kann in verschiedenen Fällen unterbrochen werden (Sprung
an den Anfang einer Programmsequenz, von einer Serie zu einer anderen
etc. oder auch während der Normalarbeit - siehe weiter unten: Pro-
grammunterbrechung). Im ersten Fall wird die Adresse des Befehls
auf den man springen will, in das PAR übernommen. Durch einen Sprung-
befehl wird diese Sprungadresse automatisch übernommen. Sie kann auch
durch einen Übertragungsbefehl gegeben werden, wenn sie vorher als
normale Angabe gespeichert wurde. Ist diese Adresse übernommen worden,
so springt das Programm auf diese angegebene Adresse und der normale
Programmablauf setzt wieder ein.

Einfügen einer Serie in eine andere Serie

Man kann durch einen Sprungbefehl beispielsweise von einer Serie S1 in
eine andere Serie S2 springen. Dieser Sprungbefehl bewirkt den Übertrag
der Adresse des nächstfolgenden Befehls (Rücksprungadresse) von dem
PAR in das Rücksprungadressregister (ANR). Wenn man am Ende der
Serie S2 wieder in die Serie S1 zurück will, und zwar an den Punkt, von
dem aus man gesprungen ist, genügt es, wenn man einen Übertrag des
ANR zum PAR macht. Da das ANR bei jedem Sprung auf dem neuesten
Stand gehalten wird, kann man durch diese Methode eine gleiche Serie
(oder Unterprogramm) aus einem Hauptprogramm von verschiedenen
LEVEL her ansprechen.

Anmerkung:

Wenn das Unterprogramm selbst auch Sprungbefehle benutzt, muß der
Programmierer am Anfang des Unterprogramms den Inhalt des ANR in eine
spezielle Zone übertragen, um damit die Rücksprungadresse des Haupt-
programms zu erhalten, denn alle Sprungbefehle ändern den Inhalt dieses
Registers. Folgende schematisch dargestellte Organisation könnte syste-
matisch angewendet werden:



Aufbau der Sprungbefehle (s. 3.2.2.)

Die Sprungbefehle ermöglichen es, den Sprung von einer Serie zur anderen von der An- oder Abwesenheit einer Bedingung abhängig zu machen.

Diese Abhängigkeit könnte unter der Form eines Zeichens der Inhalt eines Spezialregisters sein (Vergleichsauswertung, Resultate aus Rechenoperationen etc.) oder auch eine andere beliebige Stelle des Normalspeicherbereichs (Schlüssel, Kartenart, Resultat einer logischen Operation etc.).

Programmunterbrechung

Das Hauptprogramm kann unterbrochen werden, um ein Testprogramm ablaufen zu lassen. Diese Unterbrechung geschieht wie ein Programmsprung:

- der Inhalt des PAR (= Rücksprungadresse des unterbrochenen Programms) wird in das ANR übertragen.
- Übertrag des AFP in das PAR. Dieses Register enthält die Anfangsadresse des Testprogramms (Unterbrechungsprogramm). Die Unterbrechung kann durch die Bedienungskraft mit Hilfe einer Spezialtaste angerufen werden. Sie ermöglicht es, ein Programm anzurufen, das eine Operation noch einmal vorne beginnen läßt. Wenn z.B. das Stanzen einer Karte unterbrochen wurde, kann dieses Programm das erneute Stanzen einer Karte versuchen, solange sich die Daten noch in der Stanzzone befinden.

Die Programmunterbrechung kann auch automatisch erfolgen als Folge von bestimmten schwerwiegenden Fehlern, die das Fortsetzen der Arbeit verbieten (s.4.3.2.2.).

2.1.7. Bedienung der Maschine

Die Tasten und Lampen, die den Betriebszustand aller Einheiten des GE-55 kontrollieren, sind in einer Reihe nebeneinander angeordnet und befinden sich am oberen Teil der Zentraleinheit, direkt im Blickfeld der Maschinenbedienung. Auf den Randeinheiten befinden sich nur die Tasten, die zu einem speziellen Eingriff an diesen Randeinheiten dienen.

An der Zentraleinheit befindet sich:

- in der Mitte eine Sichtanzeige, an der die Eingabe über die numerische Tastatur durch Klarschrift kontrolliert werden kann oder an der Informationen aus dem Zentralspeicher in Klarschrift sichtbar gemacht werden können. Die Arbeitsweise dieser Anzeigevorrichtung wird in Abschnitt 2.2 erklärt.
- rechts von der Sichtanzeige die Tasten zur Bedienung der Maschine. Sie sind mit Lampen versehen.
- links die Lampen der Störungsanzeige.

2.1.7.1. Die Drucktasten

ON

(power on) schaltet die Stromzufuhr zur Maschine ein. Die Maschine ist nach Betätigen dieser Taste nicht sofort betriebsbereit, sondern es muß abgewartet werden, bis sie eine bestimmte Temperatur erreicht hat. Diese wird durch Aufleuchten der Lampen LOAD, PRS, RUN und PRR angezeigt. Erst wenn diese vier Lampen brennen, kann mit dem Laden des Programms begonnen werden. Diese Anlaufzeit kann max. 5 Minuten dauern.

OFF

(power off) schaltet die Stromzufuhr zur Maschine ab. Diese Taste dient zum Abschalten der gesamten Stromzufuhr zur Maschine. Sie soll während der Arbeit so wenig wie möglich benutzt werden, da mit ihr auch die interne Klimatisierung abgeschaltet wird. Die bestmögliche Ausnutzung der elektronischen Stromkreise ist aber nur bei möglichst geringen Temperaturschwankungen gegeben. Bei schwerwiegenden Störungen an den Randeinheiten soll sie aber benutzt werden, um diese restlos abzuschalten. Das gilt besonders bei manuellen Eingriffen (z.B. Entfernen eines Kartenstaus am Stanzer).

LOAD

(program loading)
Mit dieser Taste wird ein Programm in Maschinensprache in den Kernspeicher geladen. Die Programmkarten werden in den Kartenleser eingelegt und mit dieser Taste vom Kartenleser gelesen und in den Kernspeicher übertragen. Nach erfolgter Eingabe wird das Programm gestartet.

Die Adresse des Befehls, der als erster ablaufen soll, wird durch die Programmstartkarte gegeben.

Diese Taste hat die gleiche Wirkung wie der Befehl LOAD, der in Abschnitt 3.8.2. beschrieben wird. Im gleichen Abschnitt wird auch der Aufbau der Programmkarten erklärt.

Die Taste LOAD annulliert die Paarigkeitskontrolle. Man kann ein durch PAR angehaltenes Programm damit erneut laden.

SSS

(single step stop) Doppelfunktionstaste.

Wenn der Schalter eingelegt ist (Lampe brennt), erfolgt ein schrittweiser Programmablauf. Das Programm wird vor jedem Befehl angehalten. Jeder folgende Befehl läuft erst nach Druck auf die Taste START ab. Das Programm läuft erst wieder automatisch weiter, wenn die Taste SSS ausgerastet wird (Lampe erlischt) und ein letztmaliger Druck auf die Taste START erfolgt ist.

Diese Taste dient in erster Linie den Maschinenteknikern zur Kontrolle der Stromkreise der Maschine. Im normalen Arbeitsablauf wird sie nur gebraucht, um das Programm direkt anzuhalten, wenn eine besondere Manipulation gemacht werden soll, z.B. Kernspeicherausdruck.

KHLT

(key halt) Doppelfunktionstaste

Ist dieser Schalter eingelegt (Lampe brennt), hält das Programm an bestimmten Stellen an, die durch den Befehl KHLT bezeichnet sind (s. 3.8.4.).

Jedesmal, wenn das Programm auf diesen Befehl trifft, hält es (bei eingelegter Drucktaste) an. Durch die Taste START erfolgt automatischer Weiterlauf bis zum nächsten Befehl KHLT. Das automatische Weiterlaufen erfolgt erst wieder, wenn die Taste KHLT ausgerastet (Lampe aus) wurde und ein letztmaliger Druck auf START erfolgte.

INH

(inhibit) Doppelfunktionstaste

Bei eingelegter Taste (Lampe an) wird die gesamte Arbeit der Zentraleinheit (und des Stundenzählers) vor Übernahme des nächsten Befehls gestoppt. Keine Funktion der Zentraleinheit kann angesprochen werden, solange diese Taste eingelegt ist. Um ein Programm wieder anlaufen zu lassen, muß die Taste ausgerastet (Lampe aus) und Taste START gedrückt werden.

Diese Taste soll der Taste OFF vorgezogen werden, um eine Arbeitsunterbrechung zu erreichen. Die Klimaanlage der Maschine wird durch sie nicht abgeschaltet.

STOP

(stop timing)

Mit dieser Taste kann die Arbeit der Zentraleinheit unmittelbar, d.h. sogar mitten in einem Befehlsablauf, unterbrochen werden. Das Programm kann wieder gestartet werden

- entweder durch Druck auf die Taste START, wenn man den unterbrochenen Befehl zu Ende führen will;
- oder durch Druck der Taste RUN am Anfang des nächsten Befehls, wenn man den unterbrochenen Befehl nicht fortsetzen will und es sich nicht um einen Ein/Ausgabebefehl handelt.

Die Taste STOP kann eine Einheit, die nicht auf der Stelle halten kann (Kartenleser, Trommel) im ersten Fall nicht sofort unterbrechen. Diese Randeinheit führt die angefangene Operation zu Ende (Lesen der Karte oder der Trommelbahn beispielsweise), obwohl die Verbindung zur Zentraleinheit unterbrochen ist. Eine Anzahl von Zeichen geht dadurch verloren. Erfolgt aber kein simultanes Arbeiten dieser Einheit mit solchen, die auf der Stelle halten können (Drucker und Stanzer), können diese letzteren auf der Stelle anhalten. Sie beginnen ihre Arbeit wieder mit der Zentraleinheit.

START

(start timing) startet die Arbeit der Zentraleinheit und des Programms an der Stelle, wo eine Unterbrechung erfolgte, d.h. entweder vor oder während der Durchführung eines Befehls.

RUN

(start run) startet den Programmablauf am Anfang des Befehls, dessen Adresse sich im PAR befindet. Unter Umständen wird die Wirkung der Parigkeitskontrolle vorher gelöscht.

Die Taste RUN kann dazu benutzt werden, um ein Programm mit dem nächsten Befehl zu starten, wenn es während des Ablaufs eines Befehls mit der Taste STOP oder infolge einer Störung (vorbehaltlich der Folgen eines Verzichts auf diesen Befehl: Verlust von Informationen etc.) angehalten wurde. Diese Taste dient nicht dazu, an das Ende eines Ein/Ausgabebefehls zu springen. In diesem Fall wird zwar der folgende Befehl richtig durchgeführt, die Zentraleinheit nimmt jedoch danach die unterbrochene Ein/Ausgabeoperation wieder auf und beendet sie.

Mit der Taste RUN kann nur nach Betätigen der Tasten SSS, KHLT oder INH ein Programm gestartet werden.

PRS

(print store)

Die Taste PRS ruft ein fest eingespeichertes Unterprogramm an. Das Unterprogramm ermöglicht einen teilweisen Kernspeicherausdruck. Die Anfangsadresse dieses Ausdruckes wird über die numerische Tastatur eingegeben. Derselbe Effekt wird erreicht, wenn im Programm der Befehl PRSTO aufgefunden wird. (Siehe Abschnitt 3.8.1.)

In beiden Fällen werden automatisch die ersten 200 Stellen des Kernspeichers ausgedruckt. Zum Ausdruck des Kernspeichers werden nacheinander die Tasten SSS - KHT - PRS - RUN betätigt.

Erfolgt der Kernspeicherausdruck, um eine Programmblockierung oder eine Störung im Kernspeicher (Paarigkeitskontrolle) festzustellen, sind folgende Regeln anzuwenden:

Im ersten Fall muß das Programm durch die Taste SSS blockiert werden, ehe der Ausdruck beginnt.

Im zweiten Fall genügt es, nur einen Kernspeicherausdruck zu befehlen. Handelt es sich um einen Schlüsselfehler, hält der Drucker an dieser Stelle an, da er es nicht drucken kann.

Wird ein Kernspeicherausdruck während des Programmablaufs gewünscht (Probeausdruck beim Testen), so muß, damit die Arbeit nicht beeinträchtigt wird:

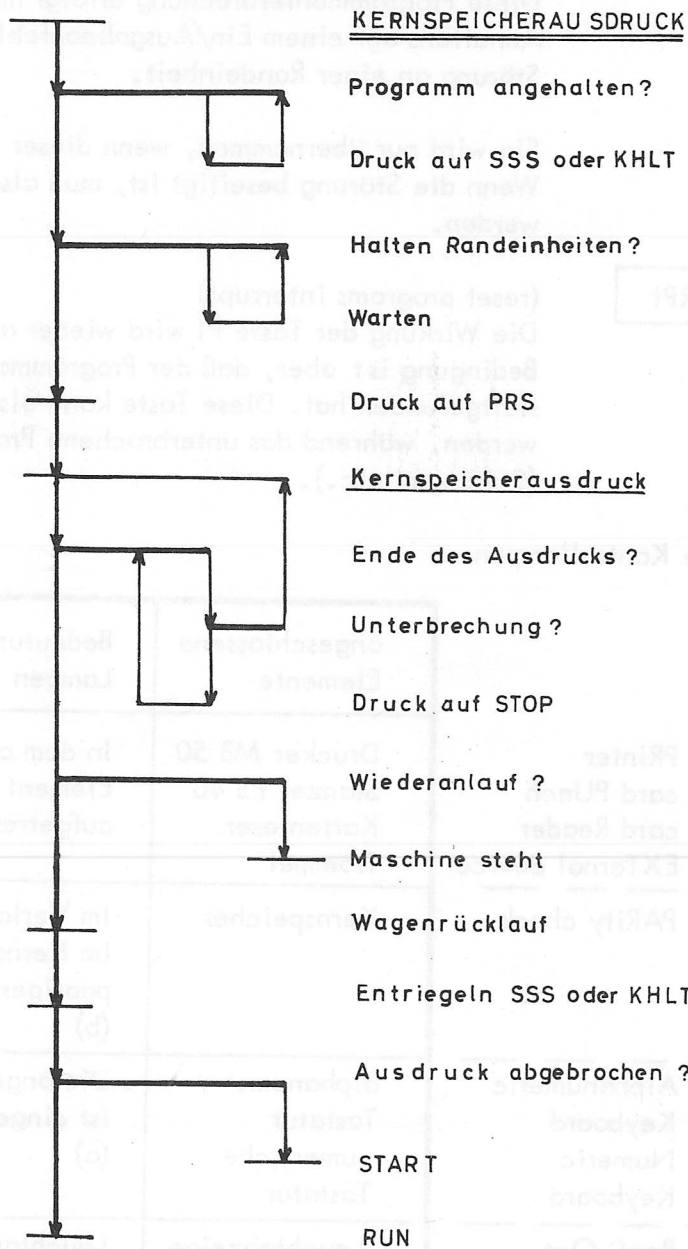
- das Programm am Anfang eines Befehls durch SSS oder an bestimmten Stellen durch den Befehl KHLT und die entsprechende Drucktaste angehalten werden.
- geprüft werden, ob die Operationen der Randeinheiten fertig sind (Einheiten angehalten und zugehörige Lampen aus; s. 2.1.7.2.)

Wird durch das normale Programm das Drucken einer Liste vorgesehen, so muß am Ende des Kernspeicherausdrucks der Schreibkopf auf Stelle 1 und das Formular gewechselt werden, bevor das Programm wieder gestartet wird.

Auf alle Fälle kann der Ausdruck durch die Taste STOP angehalten werden, wenn genügend Informationen ausgedruckt sind.

→ haus.
5.2.3
(11.11.68)

Die hier beschriebenen Verfahren sind im folgenden Organigramm schematisch wiedergegeben:



PRR

Diese Taste ist ohne Funktion

PI

(programm interrupt)

Diese Taste hält das gerade ablaufende Programm vor dem nächsten Befehl an, blockiert dieses Programm und startet zur gleichen Zeit ein Programm, das sich an der im Register AFP angegebenen Stelle befindet (siehe Programmunterbrechung in Abschnitt 2.1.6.).

Diese Programmunterbrechung erfolgt meistens während des Anhaltens auf einem Ein/Ausgabebefehl durch eine Störung an einer Randeinheit.

Sie wird nur übernommen, wenn dieser Befehl beendet ist. Wenn die Störung beseitigt ist, muß also neu gestartet werden.

RPI

(reset programs interrupt)

Die Wirkung der Taste PI wird wieder aufgehoben. Bedingung ist aber, daß der Programmaustausch noch nicht stattgefunden hat. Diese Taste kann also nur benutzt werden, während das unterbrochene Programm anhält (Störungen etc.).

2.1.7.2. Die Kontrolllampen

- PR = PRinter
- PU = card PUnch
- R = card Reader
- EXT = EXTERNAL device
- PAR = PARity check

- A = Alphanumeric Keyboard
- N = Numeric Keyboard
- RDO = ReaD Out

angeschlossene Elemente	Bedeutung der brennenden Lampen
Drucker MB 50 Stanzer PS 40 Kartenleser Trommel	In dem angeschlossenen Element ist ein Fehler aufgetreten (a)
Kernspeicher	Im Verlauf einer Operation im Kernspeicher ist ein paariges Byte aufgetreten (b)
alphanum. Tastatur numerische Tastatur	Die angeschlossene Tastatur ist eingeschaltet (a)
Leuchtanzeige	Leuchtanzeige wahrnehmen (a)

(a) Siehe Kapitel über die Randeinheiten.
 (b) Paarigkeitskontrolle

Ein aufgetretenes paariges Byte während einer Operation im Kernspeicher hält unmittelbar die Funktionsweise der Zentraleinheit und den ablaufenden Befehl an. Es kennzeichnet einen Lesefehler der Bits. Der Platz des fehlerhaften Zeichens und infolgedessen auch die Information, in der es aufgetreten ist, ist unbekannt.

Darum muß man:

- einen Kernspeicherausdruck machen, um die Adresse des fehlerhaften Zeichens zu finden,
- den aufgetretenen Fehler einem Techniker melden,
- die Arbeit fortsetzen oder den fehlerhaften Teil der Arbeit neu durchführen.

Zwei Möglichkeiten des Wiederanlaufs gibt es:

- die erste besteht darin, das Programm mit Taste LOAD neu zu laden und die Arbeit von vorne zu beginnen.
- die zweite besteht darin, das Programm von einem Wiederanlaufpunkt, der vom Programmierer festgesetzt wird, neu zu starten.

Dieser teilweise Wiederanlauf kann auch durch das Unterbrechungsprogramm erfolgen, das durch Programmunterbrechung angerufen wird.

Es ist aber auch durchaus möglich, auf den fehlerhaften Teil der Arbeit zu verzichten, indem man mit Taste RUN fortfährt. Dabei müssen aber alle Konsequenzen, die sich aus einer fehlerhaften Information ergeben, in Betracht gezogen werden. Diese Entscheidung kann nur der für die Arbeit Verantwortliche treffen.