

COMPUTER
EIN/AUSGABE-GERÄTE

COMPUTER EIN/AUSGABE-GERÄTE

I N D E X

1. Einleitung

- 1.1 Bedeutung
- 1.2 Verständnis
- 1.3 Wichtigkeit
 - 1.3.1 Zeitliche Änderung
 - 1.3.2 System-Größe
- 1.4 E/A-Vorgänge
- 1.5 Aufgaben der Vorlesung

2. Grundproblem der E/A-Geräte

- 2.1 E/A-Geschwindigkeit
- 2.2 Lösungen zum Grundproblem
 - 2.2.1 Konvoi-Verfahren
 - 2.2.2 Pufferung
 - 2.2.3 Beendigungswarten
 - 2.2.3.1 Vergleich mit vorigen Verfahren
 - 2.2.4 Parallelarbeit mit Kanälen
 - 2.2.5 Unterbrecherwerk
 - 2.2.6 Zyklenstehlen
 - 2.2.6.1 Eingriff-Aktivierung
 - 2.2.6.2 Ausschluß des Rechners
 - 2.2.6.3 Zusammenfassen der Daten
 - 2.2.6.4 System-Änderung
 - 2.2.7 Speicher-Schachtelung
 - 2.2.7.1 Adressenzählung
 - 2.2.7.2 Effektive Zykluszeit
 - 2.2.7.3 Vergleich mit den bisherigen Methoden
 - 2.2.7.4 Verwirklichung

3. Geschichte der E/A-Geräte

- 3.1 Mechanische Maschinen
- 3.2 Elektro-mechanische Maschinen
- 3.3 Elektronische Maschinen
- 3.4 E/A-Geräte
 - 3.4.1 Datenstationen
- 3.5 Zwischenspeicher-Geräte

4. Charakteristiken der Daten

- 4.1 Daten und Information
- 4.2 Zweck der Daten
- 4.3 Daten-Zuordnung
- 4.4 Daten Verschachtelung
 - 4.4.1 Stufe der Daten
 - 4.4.2 Reihenfolge der Daten
- 4.5 Daten-Quanten
 - 4.5.1 Benutzerquanten
 - 4.5.2 Computerquanten
 - 4.5.3 E/A-Quanten
 - 4.5.4 Abbildung und Transformation
 - 4.5.5 Bestimmung der Quanten
 - 4.5.6 Identifizierung
 - 4.5.6.1 Stufen der Identifizierung
 - 4.5.6.2 Definitionen
 - 4.5.6.3 Organisation der Felder
 - 4.5.6.4 Vor- und Nachteile
- 4.6 Arbeiten mit Dateien
 - 4.6.1 Rechnerquanten
 - 4.6.2 E/A-Quanten
 - 4.6.3 Quantenzusammenhänge

5. Daten-Medien

- 5.1 Mittel zum Festhalten von Daten

5.2 Komponenten der Geräte zur Verarbeitung des Mediums

5.2.1 Anordnung

5.2.2 Funktionen

5.2.3 Arbeitsweise

5.3 Medien

5.3.1 Lochkarten

5.3.2 Lochstreifen

5.3.3 Magnetband

5.3.3.1 Schreibdichte

5.3.3.2 Bits/Bytes pro Zoll

5.3.3.3 Arten

5.3.3.4 Übertragungsgeschwindigkeit

5.3.3.5 Nachteile

5.3.3.6 Bandabnutzung

5.3.3.7 Parität

5.3.3.8 Schreibdichten

5.3.3.9 Vergleich zu Karte und Streifen

5.3.3.10 Registrierung

5.3.3.11 Lücken

5.3.3.12 Brauchbare Länge

5.3.3.13 Datenkapazität

5.3.3.14 Blocken

5.3.3.15 Dichte und Kapazität

5.3.4 Direktzugriffseinheiten

5.3.4.1 Bewegungsunterschiede

5.3.4.2 Vorteile

5.3.4.3 Arten

5.3.4.4 Oberflächen

5.3.4.5 Lesen und Schreiben

5.3.4.6 Registrierflächen

5.3.5 Registrierung auf magnetischem Material

5.3.5.1 RZ-Verfahren

5.3.5.1.1 Schreiben

5.3.5.1.2 Lesen

5.3.5.2 RB-Verfahren

- 5.3.5.3 Nachteile
- 5.3.5.4 NRZ-Verfahren
- 5.3.5.5 NRZ1-Verfahren
- 5.3.5.6 Probleme bei den Verfahren
- 5.3.5.7 Phasenverschlüsselungsverfahren
 - 5.3.5.7.1 Zwei-Phasenverfahren
 - 5.3.5.7.2 Manchester-Verfahren

6. Ein-Ausgabegeräte

- 6.1 Lese-Schreibverfahren
 - 6.1.1 Mechanische Verfahren
 - 6.1.2 Elektrische Verfahren
 - 6.1.3 Elektro-Magnetische Verfahren
 - 6.1.4 Optische Verfahren
 - 6.1.5 Elektronische Verfahren
- 6.2 Vor- und Nachteile der Verfahren
- 6.3 Lochstreifen-Geräte
 - 6.3.1 Charakteristiken
 - 6.3.1.1 Lochstreifenleser
 - 6.3.1.2 Lochstreifenstanzer
 - 6.3.1.3 Unabhängige Hersteller
 - 6.3.1.4 Indirekte Lochstreifengeräte
 - 6.3.2 Vor- und Nachteile
 - 6.3.3 Programmierung
 - 6.3.4 Lesevorgang
 - 6.3.5 Stanzvorgang
 - 6.3.6 Allgemeiner Vorgang
 - 6.3.7 Bauformen
- 6.4 Lochkarten-Geräte
 - 6.4.1 Charakteristiken
 - 6.4.2 Lese- und Stanzvorgänge
 - 6.4.3 Programmierung des Lesers
 - 6.4.4 Stanzer
 - 6.4.5 Indirekte Kartengeräte
- 6.5 Schnelldrucker
 - 6.5.1 Charakteristiken
 - 6.5.2 Druckvorgang

- 6.5.3 Papiervorschub
- 6.5.4 Programmierung
- 6.5.5 Druckverfahren
- 6.6 Magnetbandlaufwerke
 - 6.6.1 Charakteristiken
 - 6.6.2 Aufbau der Laufwerke
 - 6.6.3 Lese-/Schreibvorgänge
 - 6.6.4 Zeitfolgen
 - 6.6.5 Entschiefen
 - 6.6.6 Programmierung der Magnetbandlaufwerke
 - 6.6.7 Fehlererkennung und Korrektur
- 6.7 Random- und Dichtzugriffseinheiten
 - 6.7.1 Terminologie
 - 6.7.2 Platte
 - 6.7.2.1 Charakteristiken
 - 6.7.2.2 Organisation und Arbeitsweise
 - 6.7.3 Platte
 - 6.7.3.1 Aufbau und Wirkungsweise
 - 6.7.3.2 Charakteristiken
 - 6.7.3.3 Positionierungsverfahren
 - 6.7.4 Datenzelle
 - 6.7.4.1 Charakteristiken
 - 6.7.4.2 Aufbau der Datenzelle
 - 6.7.4.3 Arbeitsweise
- 6.8 Konsolschreibmaschinen
 - 6.8.1 Prinzipieller Aufbau
 - 6.8.2 Arbeitsweise
 - 6.8.3 Kontrolleinheit
 - 6.8.4 Konsoleingabe
 - 6.8.5 Konsolausgabe
 - 6.8.6 Datensichtgerät
- 6.9 Datensichtgerät
 - 6.9.1 Aufbau
 - 6.9.2 Bilderzeugung
 - 6.9.2.1 Punktverfahren
 - 6.9.2.2 Vektorverfahren
 - 6.9.2.3 Zeilenverfahren
 - 6.9.2.4 Zeichengeneratoren

6.9.3 Programmierung der Bilderzeugung

6.9.3.1 Zeichengenerator

6.9.3.2 Punktverfahren

6.9.3.3 Lichtgriffel

7. Satellitenrechner

8. Plotter

8.1 Analog-Plotter

8.2 Digital-Plotter

9. Kanalwerke

9.1 Grundproblem

9.2 Standardkanäle/Schnellkanäle

9.3 Aufbau der Kanäle

9.3.1 Allgemeiner Aufbau

9.3.2 Funktioneller Aufbau

9.4 Arbeitsweise

9.4.1 E/A-Befehle

9.4.2 Abspeichern und Holen der Daten

9.4.3 Schnittstellen

1. Einleitung

Bei der Behandlung dieses Themas stehen mehrere Wege offen, einmal ein theoretischer mit großer Allgemeingültigkeit, doch mit wenig Wirklichkeitsbezug, und zum anderen ein technischer mit einem Wust von speziellen Einzelheiten. Ein Mittelweg unter Berücksichtigung technischer Einzelheiten, soweit sie dem Verständnis dienen, scheint die beste Lösung zu sein.

1.1 Bedeutung

Die Bedeutung peripherer Geräte wird i.a. weit unterschätzt. Ist doch kein sinnvolles Rechnersystem ohne wenigstens minimale Vorkehrungen für Ein- und Ausgabe denkbar. Die Unterschätzung bezieht sich auf studien-, systemorganisatorische und auf sachliche Betrachtung.

a) Studien-organisatorisch

Ein Informatikstudent hört über viele Jahre hinweg Vorlesungen über Mathematik, Theorie, Software usw., doch später im Berufsleben als Anwendungs- oder Systemprogrammierer kann gelegentlich das bestentworfene und -entwickelte Programm nicht ablaufen, weil das Lesegerät (Karten-, Lochstreifen- oder Magnetbandleser) nicht funktioniert und der Anwender sich nicht zu helfen weiß.

b) System-organisatorisch

Häufig stehen - selbst noch in neuester Zeit - Rechnersysteme mit einem Anschaffungswert von mehreren Millionen nutzlos herum, weil das hauptsächliche Ein/Ausgabegerät defekt ist. Außerdem wird geschätzt, daß in neuerer Zeit von den Gesamtkosten für ein Rechnersystem etwa die Hälfte auf die E/A-Geräte und deren Zusatzkomponenten entfällt.

c) Sachlich

Der E/A-Anteil in Programmen erscheint unauffällig und klein gegenüber der Gesamtzahl der Befehle. Dies rührt daher, daß meist standardisierte E/A-Unterprogramme verwendet werden und so die eigentlichen Verhältnisse verschleiert werden.

Um die E/A-Vorgänge in der richtigen Perspektive sehen zu können, wäre es zweckvoll, die Zeit zu betrachten, während der die E/A-Geräte im System arbeiten und mit der entsprechenden Zeit des Zentralrechners zu vergleichen.

1.2 Verständnis

Die E/A-Vorgänge werden meist nicht nur unterschätzt, sondern ebenfalls unvollständig oder gar falsch dargestellt. So findet man häufig folgende schematische Darstellung der Systemkomponenten:

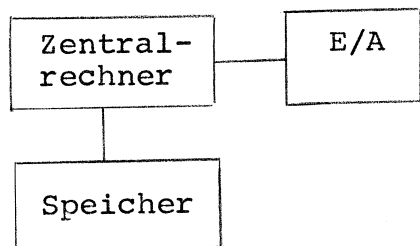


Abb. 1

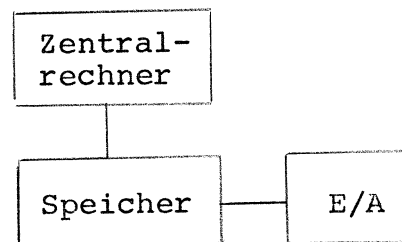


Abb. 2

Beide Bilder sind zumindest unvollständig.

Oft findet man ein mangelndes Verständnis des Zusammenwirkens der E/A-Geräte mit den übrigen Systemkomponenten. Z.B. existiert in einem Benutzer-Manual eines großen Rechnerherstellers das folgende Bild (Abb. 3):

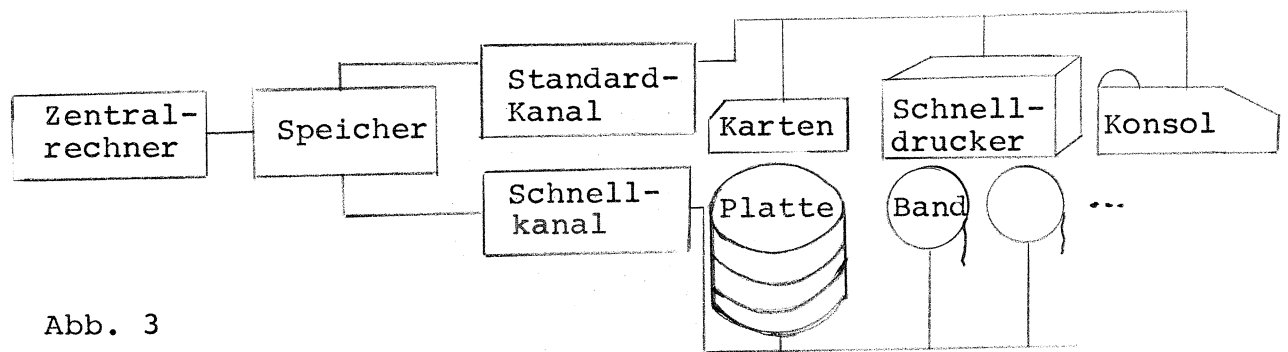


Abb. 3

Dabei fiel es nicht schwer, den Gesamtdurchlauf durch dieses System mit einer geringen Änderung der Konfiguration i.a. Fall zu erhöhen (z.B. eine oder zwei Band-einheiten am Standard-, statt am Schnellkanal für kontinuierliches Arbeiten).

1.3 Wichtigkeit

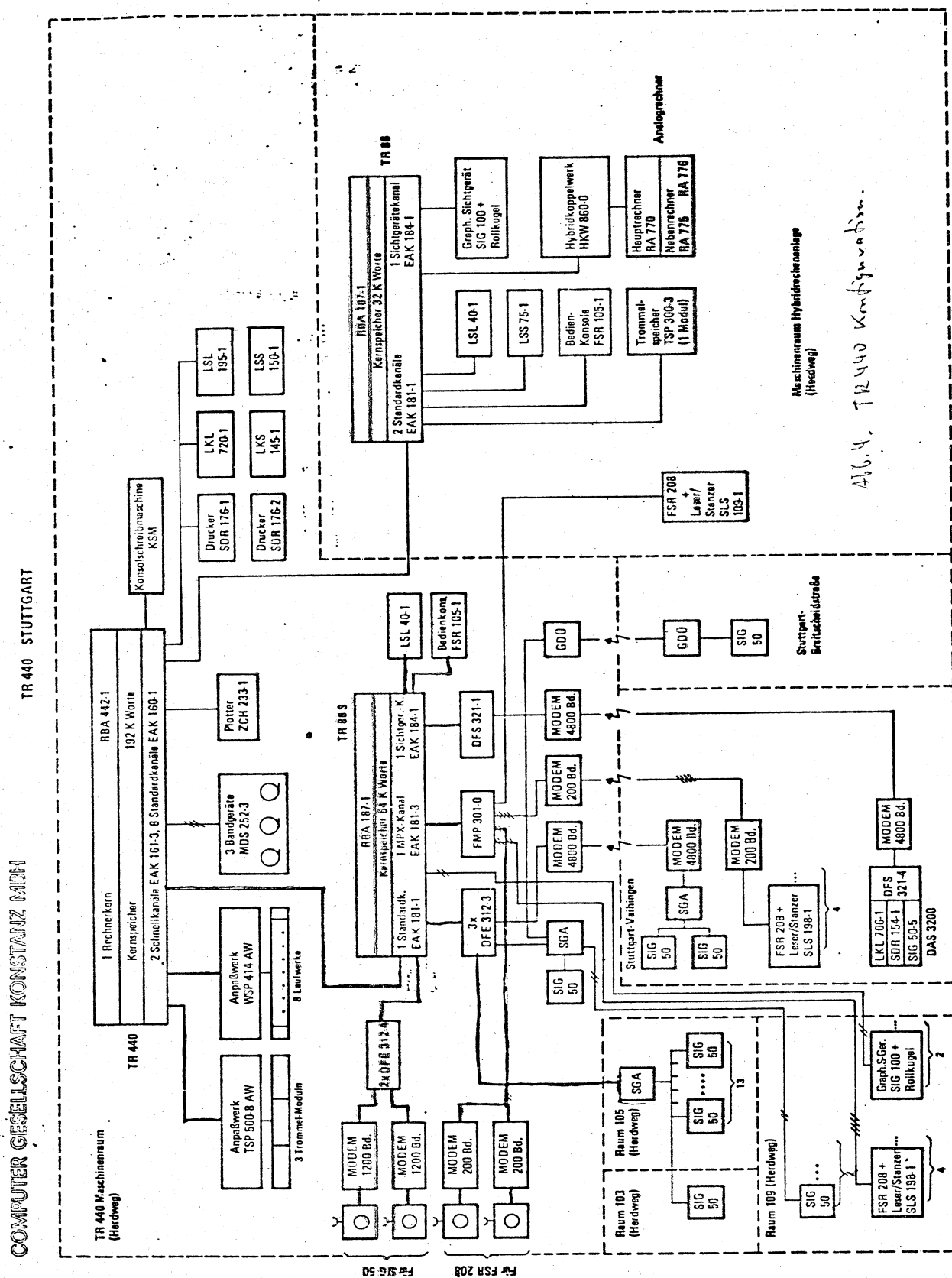
1.3.1 Zeitliche Änderung

Die Bedeutung der E/A-Geräte in finanzieller Hinsicht als Systemkomponenten wurde bereits erwähnt. Diese Wichtigkeit der peripheren Einheiten verlief und verläuft jedoch nicht statisch, sondern dynamisch im Verlauf der Zeit. So wurden früher nur etwa 20 % der Gesamtkosten eines Systems für E/A-Geräte veranschlagt, gegenüber den etwa 50 % heute. Dies rührt daher, daß heutzutage verhältnismäßig mehr E/A-Geräte in ein System genommen werden als früher, und daß die Preise für Zentralrechner und Speicher kleiner geworden oder weniger stark gestiegen sind.

1.3.2 System-Größe

Die System-Größe betreffend findet sich eine ähnlich dynamische Erscheinung. Ein mittleres System enthielt früher nur wenige E/A-Geräte, z.B. eine Karteneinheit, einen oder zwei Schnelldrucker und zwei bis acht Band-

Abb. 4 TR 44o Konfiguration



einheiten. Heute enthalten mittlere Systeme meist zusätzlich noch Trommel-, oder auch Plattenspeicher, sowie oft Satelliten- und Datenstationen, sind also mehr kommunikativ orientiert.

1.4 E/A-Vorgänge

In größeren modernen Systemen, für Beispiele siehe Abb. 4 und 5, spielen sich sehr komplexe E/A-Vorgänge ab.

Z.B. in Abb. 5 zwischen:

- a) Datenstationen und E/A-Geräten einmal und Satellitenrechnern (-)
- b) Satellitenrechnern und Vorprozessor (---)
- c) Vorprozessor und Hauptprozessoren (....)
- d) Hauptprozessoren und Speichereinheiten (.-.-.-)
- e) andere (..-..-..-)

Aus dem Bild kann erkannt werden, daß zwei Typen von E/A-Geräten zu unterscheiden sind:

1. Direktgeräte (Karten, Lochstreifen, Datenstation, Drucker, Plotter, Wandler)
2. Zwischenspeichergeräte (Massenkernspeicher, Bänder, Trommeln, Platten, Magnetkarten, usw.)

Beide Typen werden generell als E/A-Geräte bezeichnet. Die Direktgeräte liefern E/A zwischen Benutzer und System, die Zwischenspeichergeräte zum und vom Rechner und Arbeitsspeicher.

1.5 Aufgaben der Vorlesung

Die Aufgaben dieser Vorlesung sind folgende:

1. Kennenlernen der E/A-Geräte
2. Verständnis der E/A-Geräte

3. Beherrschung der E/A-Vorgänge (Programmierung der E/A-Geräte, Kanal- und Unterbrecherwerke)

2. Grundproblem der E/A-Geräte

Betrachtet man die E/A-Geräte gegenüber den anderen Komponenten in einem System (z.B. Rechner und Speicher), so fällt sofort die Erscheinung auf, daß die E/A-Geräte wesentlich langsamer arbeiten, wie jene Komponenten. Und zwar beläuft sich der Unterschied darin meist auf Größenordnungen. Nehmen wir z.B. einen Rechner mit 100 nsec. Zykluszeit, auch Schnellzugriffsspeicher und integrierte Speicher mit ähnlichen Geschwindigkeiten sind erhältlich, und vergleichen damit die Stanzzeit für einen Lochstreifen, so finden wir einen Unterschied von 3 Größenordnungen. Andere E/A-Geräte arbeiten wieder mit anderen Geschwindigkeiten und zwar finden wir Unterschiede zwischen den Klassen dieser Geräte und auch zwischen den einzelnen Geräten einer Klasse.

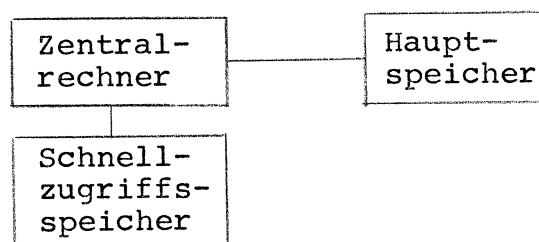


Abb. 6 Schnellzugriffsspeicherkonfiguration
(erstmal bei der Honeywell 800 vor ca. 12 Jahren verwendet, neuerdings bei 360/85, 360/195 und 370).

2.1 E/A-Geschwindigkeiten

Betrachtet man die Geschwindigkeiten der Klassen von E/A-Geräten, so lassen sich diese in eine Reihenfolge

absteigender Arbeitszeit bringen:

1. Datenstationen (z.B. Bildschirme, Fernschreiber, Plotter, langsame Drucker) für direkten Zugang der Benutzer zum System. Arbeitszeiten liegen in der Größenordnung von .1 sec. pro Zeichen.
2. Lochkarten, Lochstreifen, Schnelldrucker für indirekten Zugang der Benutzer zum System mit Arbeitszeiten in der Größenordnung 1 - 100 msec. pro Zeichen.
3. Bewegliche Hilfs- (Zwischen-)speicher (Magnetband, Trommel, Platte, Magnetkarten und -streifen) für indirekten Zugriff (mit Wartezeiten für mechanische Bewegungsvorgänge) des Rechners. Arbeitszeiten in der Größenordnung von 1 μ sec. bis 1 msec. pro Zeichen (in Gruppen oder Blocks).
4. Feststehende Hilfsspeicher (Massen- oder Großkernspeicher) für direkten Zugriff des Rechners (ohne Wartezeiten auf mechanische Bewegungsvorgänge) mit Arbeitszeiten von 100 nsec. bis 1 μ sec. pro Zeichen in Agglomeraten.

2.2 Lösungen zum Grundproblem

Zur Lösung des Grundproblems sind viele Möglichkeiten vorhanden. Im folgenden werden als die wichtigsten herausgegriffen: Konvoi-Verfahren, Pufferung, Beendigungswarten, Parallelarbeit mit Kanälen, und Verschachtelung.

2.2.1 Konvoi-Verfahren

Das Prinzip hier verlangt als einfachste Lösung eine Verlangsamung aller Systemkomponenten auf die Geschwindigkeit der langsamsten, also eine generelle Synchronisation aller Komponenten. Wegen der immensen Verluste an Effizienz wurde dies für Systeme, selbst in den An-

fängen der Computertechnik, nicht verwendet. Jedoch findet man das Prinzip noch in einfachen Zentralrechnern verwirklicht.

Da generelles Arbeiten in seriell, synchronisiertem Modus höchst unwirtschaftlich ist, müssen Wege gefunden werden, um Parallelismus verschiedener Art zu verwenden. D.h., die E/A-Funktionen müssen asynchron mit den Rechnervorgängen verlaufen.

2.2.2 Pufferung

Schon die ENIAC-Maschine arbeitete auf der Basis, daß jeweils eine Lochkarte vorausgegeben wird, parallel zum Arbeiten des Rechners. Die Methode arbeitet i.a. relativ zufriedenstellend bei kleinen Datenmengen; bei größeren wird möglicherweise der Rechner im ungünstigsten Fall auf die Geschwindigkeit des betreffenden E/A-Gerätes verlangsamt. Eine solche Erscheinung findet sich häufig beim Erschöpfen der Information in Schnellzugriffsspeichern, wo auf die Geschwindigkeit des Hauptspeichers verlangsamt wird.

Dieses Prinzip bildet übrigens die Basis des Spulens bei modernen Betriebssystemen und kann dort zu großer Unwirtschaftlichkeit führen, wenn der Zentralrechner anderweitig völlig ausgelastet ist.

Zu erwähnen ist noch ein erhöhter Aufwand beim Betriebssystem für die Pufferung, das Vorauslesen/Nachschieben durch die E/A-Geräte.

2.2.3 Beendigungswarten

Auf derselben Stufe der Wirtschaftlichkeit wie das Vorauslesen ist das Warten auf die Beendigung eines

Ein- oder Ausgabevorganges, wobei im übrigen der Rechner unabhängig arbeitet. Nehmen wir z.B. einen 8-Kanal Lochstreifen, etwas versetzt, von der Mitte haben wir Transportlöcher, dann einmal 3 Positionen auf der einen Seite und 5 Positionen auf der anderen Seite. Die Anordnung von gestanzten und nicht gestanzten Löchern wird als Sprosse, ein Zeichen repräsentierend, bezeichnet.

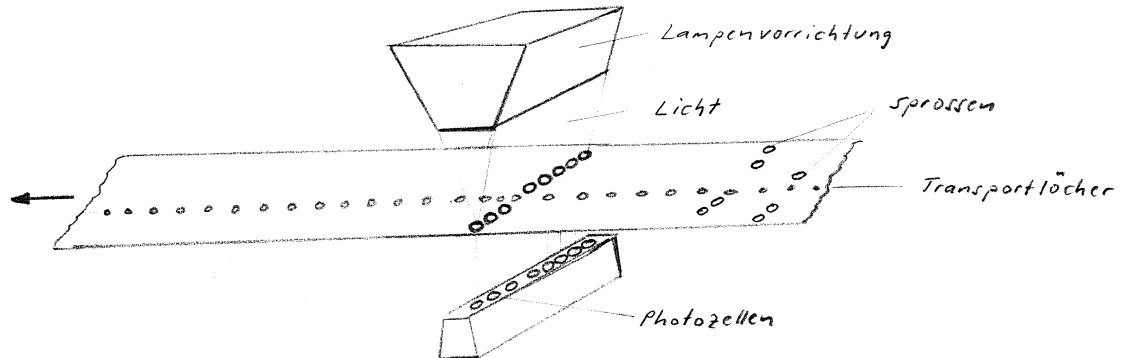


Abb. 7: Lochstreifen lesen

Außerdem haben wir folgendes: der Lochstreifen kann auch Binärwörter enthalten. Die Anordnung ist dabei je nach Wortgröße des Rechners, an den das Gerät angeschlossen ist, daß 3 bis 4 dieser Sprossen zu einem Binärwort zusammengefaßt werden. Dies sind Einzelheiten des Lochstreifens, wir werden für weitere Einzelheiten sowie die Bearbeitung innerhalb des Streifenlesers betreffend, wieder später darauf zurückkommen.

Doch möchte ich noch die Anordnung in Bild 7 erklären: Wir haben einmal eine Lampe oder eine Lampenvorrichtung, von wo ein Lichtbündel herausfällt, darunter geht der Lochstreifen vorbei und unter dem Lochstreifen haben wir nun eine Anordnung von Photozellen. Diese Photozellen erzeugen einen Strom im Falle, daß Licht der Lampe durch die Löcher des Lochstreifens darauffällt. Dieser Strom wird dann abgetastet, und die Stanzungen des Lochstreifens

fens so als Strom/Spannungsimpulse weitergegeben.

Mit der Programmierung des Lesevorgangs ist folgendes einfaches Programm gegeben:

```
rsa    read select alphanumeric
rsf    reader flag skip
jmp.-1 jump back for wait
rrb    read reader buffer
{      } store character and check for end of reading
```

Die rsa, read select alphanumeric-Instruktion verlangt das Ansteuern des Lochstreifenlesers für alphanumerisches Lesen, d.h., wir wollen ein Zeichen oder eine Sprosse lesen durch den Lochstreifenleser.

Die nächste Instruktion ist ein rsf, reader skip on flag, und zwar ist dieser Lochstreifen ein mechanisches Gebilde und muß bewegt werden, es sind Motoren und Rollen nötig, die den Lochstreifen bewegen. Und diese Gebilde müssen angetrieben werden. Dies braucht Zeit und während dieser Zeit fließt noch kein Strom in den Photozellen und erst dann, wenn der Lochstreifen sich bewegt, kann ich lesen. Diese Zeit ist die erwähnte Latenzzeit, die bei dem Zusammenwirken eines E/A-Gerätes mit dem Rechner zu beachten ist.

Der Rechner ist wesentlich schneller, er braucht keine mechanische Bewegung, er kann in dieser Wartezeit nichts tun, er richtet sich nach dem Lochstreifenleser, wenn er von dort etwas lesen will. Hinter den Photozellen kommen ein paar Speicherzellen, ein Puffer. Dieser Puffer wird gefüllt durch die Stromstöße von den Photozellen. Sobald der Puffer voll ist, wird ein Zeichen gesetzt, das flag. Dies kann vom Zentralrechner erkannt werden und dann wird die nächste Instruktion übersprungen. Das ist die Idee dieser Instruktionsfolge. Die nächste Instruktion ist ein Sprungbefehl zurück auf die vorherige Instruktion zum Warten auf das flag, d.h., ich warte bis eben das Zeichen gelesen ist und der Wert dafür im Puffer-

speicher vorhanden ist. Bei diesem Vorgang gehen etwa 80 bis 100 msec. verloren. Während dieser Zeit könnte der Rechner etwas anderes tun, was auch immer. Dies wäre eine zweite Möglichkeit um die Zeitdifferenz zu überbrücken.

Wenn ich dann das flag habe, wird der Sprungbefehl übersprungen und dann kann ich nun ablesen, was im Puffer steht, z.B. durch die rrb-Instruktion, d.h., hier wird der Datenwert des Zeichens zum Speichern in den Akkumulator gebracht. Dann muß es vom Akkumulator weggespeichert werden in den Hauptspeicher. Nachfolgend müssen Programminstruktionen existieren, die dann eigentlich in das Betriebssystem gehören, die feststellen müssen, ob alle Zeichen eingelesen sind, wo die Zeichen weggespeichert werden müssen, usw. Dies wird uns nicht so sehr interessieren, auf jeden Fall haben wir hier eine Sequenz, die die Zeitdifferenz zwischen dem Ein- und Ausgabegerät, hier dem Lochstreifenleser und dem Zentralrechner und dem Hauptspeicher überbrückt.

Eine ähnliche Folge haben wir ebenfalls für die Zeichenausgabe zu erwarten, wenn ein Zeichen oder ein Wort gestanzt werden soll.

2.2.3.1 Vergleich mit dem vorigen Verfahren

Wir haben folgenden Effekt, daß das System nicht über das Ganze gesehen verlangsamt wird, wie auf Stufe 1, sondern nur während der Zeit, in der etwas gelesen oder geschrieben wird, also nur während des eigentlichen Lese- und Schreibvorgangs.

Die Frage taucht auf, ob das nicht eine überholte Methode ist. Sie ist es nicht, sie wird immer noch

bei Minicomputern verwendet und zwar bei solchen Computern, die ohnehin einen billigen Rechner haben. Es ist eine außerordentlich einfache Methode, um Ein- und Auszulesen und deshalb für diese Computer wie geschaffen. Außerdem wird sie noch verwendet bei den Anfangsladern, selbst von größeren und sehr großen Systemen, den Bootstrap-Loadern, dem Ur-lader.

Wenn man anfängt, das System zu starten, muß es erst einmal einen Lader bekommen, und dieser Lader muß irgendwie in das System hereingebracht werden, in den Speicher, und dafür werden solche Sequenzen zum Teil verwendet.

Warum wird eine solch einfache Methode selbst heutzutage dort noch verwendet? Die Antwort ist, daß nichts anderes ohnehin im Moment zu tun ist, denn es ist ja noch nichts im Speicher. Daher können wir uns erlauben, zu warten, bis eben die Daten eingelesen sind.

2.2.4 Parallelarbeit mit Kanälen

Die nächste Stufe der Lösung des Problems ist die Parallelarbeit von E/A-Vorgängen mit dem Zentralrechner und dem Speicher durch Verwendung von Kanälen. Parallelarbeit muß natürlich hier asynchron vor sich gehen, und zwar meist mit Unterbrecherwerk und Kanalwerk. Diese Idee wurde herausentwickelt durch die Nachteile, die bei den vorher erwähnten Methoden auftreten. Diese Nachteile sind, daß eben doch noch während langer Zeitspannen der Rechner unbenutzt dasteht. D.h., wenn der Rechner nichts tut, und der Speicher nichts tut, dann liegt der Großteil sehr wertvoller Kapazitäten im System brach, was man sich eigentlich in einem teureren System nicht leisten kann.

Sie werden im weiteren Verlauf sehen, völlig wegkommen kann man nicht von dem Problem. Es wird nur umgangen, auf mehr oder weniger gute Weise. Hier ist die Idee folgende: Da ist ein Kanalwerk, ein sehr einfacher Rechner, der übernimmt die Datensammlung und die Überwachung der E/A-Vorgänge. Das ist die Grundidee für diese Lösung durch Parallelarbeit.

Sobald ein kleiner, sehr einfacher und dadurch sehr billiger Rechner die Datensammlung und Überwachung der E/A-Geräte übernimmt, dann ist der Zentralrechner oder Hauptrechner frei, um möglicherweise weiterrechnen zu können, siehe Abb. 8:

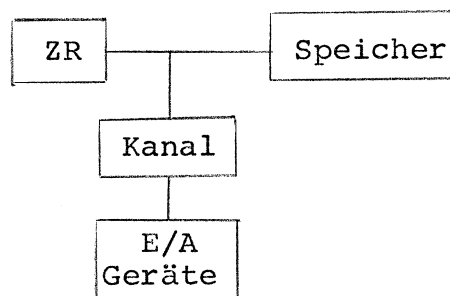


Abb. 8: Kanalordnung

Wenn wir jedoch näher zusehen, so hat diese Anordnung auch seine Haken, und zwar folgende: Wir haben hier einmal einen Zentralrechner und dann den Hauptspeicher. Wenn auf derselben Stufe des Zentralrechners ein Kanalwerk sitzt, dann entsteht ein Konflikt zwischen diesen zwei Rechnern. Natürlich existiert eine Differenz in der Leistung. Der Zentralrechner ist normalerweise wesentlich größer, komplizierter und schneller wie das Kanalwerk. Doch haben beide eine ähnliche Funktion und die Frage ist, wie die beiden zusammenarbeiten. Beide können an den Speicher-Bus angeschlossen werden, und die Sache ist gelöst im Bilde. Aber die Frage ist, wie arbeiten sie zusammen. Wir haben nun bei der Datenübertragung folgendes zu beachten:

Während keine Daten übertragen werden, kann der Zentralrechner den Speicher bedienen oder vom Speicher Instruktionen und Daten aufarbeiten. Das Kanalwerk hat nun die E/A-Geräte daran anhängen und arbeitet mit den E/A-Geräten. Dem Kanalwerk sind kleine Pufferspeicher zugeordnet und in Wirklichkeit haben wir sozusagen zwei getrennte Systeme in diesem Moment. Der Zentralrechner arbeitet mit dem Speicher und das Kanalwerk mit dem Puffern arbeitet mit dem E/A-Geräten. Dabei entsteht kein Konflikt. Nur in den Zeiten, wenn die Daten von den Puffern in den Speicher oder vom Speicher in die Puffer übertragen werden müssen, was ja sein muß, denn sonst bekomme ich keine Information in den Rechner herein oder keine Resultate heraus. Dieses Problem läßt sich nicht übergehen, wenn das System sinnvoll arbeiten soll. Deshalb müssen wir die Datenübertragung betrachten und da existiert das alte Problem immer noch.

Die Datenübertragung kann in diesem Fall einmal durch Benachrichtigung über sog. Semaphore erfolgen als die einfachste Methode. Das Kanalwerk benachrichtigt irgendwie den Zentralrechner, daß Daten vorhanden sind. So wurde es anfangs versucht, aber es hat sich nicht bewährt, und zwar deshalb, weil es keine elegante Methode gibt wie das Kanalwerk den Zentralrechner benachrichtigen soll.

Das Kanalwerk muß in das normale Zusammenspiel von Speicher und Zentralrechner irgendwie eingreifen. Es könnte auf einfache Weise eingreifen, z. B. den Zentralrechner kurz vom Speicher trennen und dann in dem Speicher eine Nachricht hinterlassen, hier habe ich Daten für dich bereit, hol' die 'mal ab. So wurde es gedacht, aber es gibt keine Ersparnis von dem was tatsächlich gemacht wird.

Was tatsächlich gemacht wird, ist, daß das Unterbrecherwerk verwendet wird; darauf kommen wir noch in Einzelheiten zu sprechen.

2.2.5 Unterbrecherwerk

Ich greife hier vor, damit Sie hier einen Überblick bekommen über das Zusammenwirken des Systems mit den E/A-Geräten. Das Unterbrecherwerk übernimmt die Datenübertragung durch einen Speichereingriff. So kann das Unterbrecherwerk den Wert des Puffers in den Akkumulator hereinbringen, z.B. mit dem rrb. Aber warum sollte es, wenn man schon ein Unterbrecherwerk hat, so kann es direkt in den Speicher eintreten. Dies klingt sehr elegant, aber der Speichereingriff geht in Wirklichkeit durch Zyklenstehlen vonstatten.

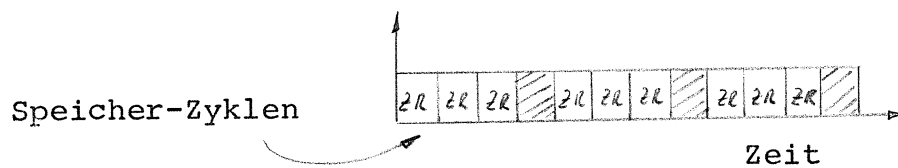
2.2.6 Zyklenstehlen

Bei dem Zyklenstehlen wird der Zentralrechner nicht für die Datenübertragung in Anspruch genommen. Die andere Möglichkeit wäre, daß der Zentralrechner für die eigentliche Abspeicherung verwandt wird. Beide Methoden sind äquivalent. Es ist kein Gewinn mit der Benutzung des Zentralrechners zu erhalten. Im Gegenteil, bei der Zentralrechnermethode muß noch erst der Akkumulator weggespeichert und wieder geladen werden, oder einer dafür reserviert werden, wenn wir mehrere Akkumulatoren haben. Deshalb arbeiten die meisten Rechner auf der Methode des Zyklenstehls.

Im Einzelnen geht es nun so vor sich:

Wenn ein Datenwert von den E/A-Geräten her im Puffer gesammelt wird, dann tritt das Flag auf, das Flag ist nun wie wir später sehen werden, mit dem Unterbrecherwerk verbunden. Das Unterbrecherwerk setzt nun den Zentralrechner außer Gang vom Zugriff zum Hauptspeicher.

Wir haben nun folgendes Zeitschaltbild (Abb. 9):



ZR = vom Zentralrechner beansprucht
 = vom Kanal entwendete Zyklen

Abb. 9: Zeitverteilung beim Zyklenstehlen

wir haben hier die Speicherzyklen über die Zeit verteilt. Diese werden nun normalerweise vom Zentralrechner in Anspruch genommen und pro Instruktion i.a. zwei Speicherzyklen benötigt, für Sprungbefehle meist nur ein Zyklus, auf jeden Fall eine vorher festgesetzte Zahl von Zyklen. Die ersten 3 Speicherzyklen in Bild 9 werden für unseren Zentralrechner verwendet, sagen wir von einer Additions-Instruktion und von einem Sprungbefehl, und nun kommt das Unterbrecherwerk in Gang und zwar deshalb, weil das Flag heraufkam.

2.2.6.1 Eingriff-Aktivierung

Während die Instruktion durchgeführt wird, d.h. während der Durchführung einer Instruktion, tritt keine Unterbrechung auf, die Unterbrechung tritt erst am Ende auf. Dann wird nun, falls verlangt, der Zyklus vom Speicher nicht mehr dem Zentralrechner gegeben, sondern vom Kanal abgenommen. Zur Abspeicherung des Puffers gibt jetzt der Kanal den Inhalt des Puffers an den Speicher ab, und dann tritt wieder der Zentralrechner für die nächsten Speicherzyklen in Aktion. Im Betriebssystem müssen noch ein paar Zyklen verwendet werden, um den E/A-Ablauf unter Kontrolle zu halten.

Dies ist die Methode mit der alle mittleren Rechner,

zum Teil auch relativ große Rechner arbeiten. Sie sehen, das Problem liegt weder im Kanal noch im Zentralrechner, sondern das Problem liegt im Speicher, und zwar an der Arbeitsweise des Speichers. Der Speicher arbeitet im Takt und der Kanal muß in den Takt des Speichers eintreten. Dies finden Sie kaum irgendwo beschrieben. Die Rechnerhersteller wollen die Kunden nicht erschrecken, indem sie sagen, daß von der Arbeitszeit der Rechner mit z.B. 6 Millionen Instruktionen pro Sekunde und so, diese Zeiten für Datenübertragungen weggehen. Während dieser Zeit sitzt der Rechner da und tut nichts, er wartet. Vorallem Anfänger finden dies etwas überraschend. Selbst in kleineren Installationen finden Sie neuerdings Testgeräte um den Ablauf des Systems festzuhalten, dabei kann man oft feststellen, daß der Rechner, sagen wir 30 %, 40 % der Zeit untätig sitzt, und man meinte, das System lief bei voller Last. Das rührt eben davon her, daß zu gewissen Zeiten der Rechner einfach nichts tun kann, weil die Datenübertragungen stattfinden müssen. Und zwar ist es so, daß alle Kanäle Vorrecht haben vor dem Rechner. Dies muß sein, denn sonst kann der Rechner, wenn keine neuen Daten eingegeben und Resultate abgegeben werden, auch wieder nichts tun. Das ist der Grund, weshalb die Kanäle Vorrecht vor dem Zentralrechner haben.

Man könnte sagen, das ist eine sehr ungünstige Methode, wenn ich ein sehr teures System habe und sehr hohe Übertragungsraten für Daten habe. Es könnten sehr wenige Zugriffe zum Speicher für den Rechner übrig bleiben und das ist tatsächlich auch oft der Fall, wie im folgenden Beispiel gezeigt.

2.2.6.2 Ausschluß des Rechners

Bei schneller Übertragung haben wir den Effekt, daß die

Übertragung mit hoher Übertragungsrate vor sich geht. Z.B. nehmen wir schnelle Bänder mit 320 K Bytes/sec. Übertragungsrate, wie die IBM 2420, Univac 20 Geräte usw., dann bleibt sehr wenig Zeit übrig, d.h., es stünden bei Zeichenübertragung nur etwa 3 μ sec Zeit zwischen dem Ankommen der Zeichen zum Abspeichern zur Verfügung. Oder bei dem Burroughs 9495 Gerät mit 400 K Bytes/sec. haben wir 2,5 μ sec. zur Verfügung. Nun müssen wir noch angeben, wie schnell die Rechner sind, an die die Bänder angeschlossen werden sollen, so z.B. eine IBM 360/50 oder der Hauptspeicher der IBM 370/155 mit 2 μ sec. Speicherzyklus, oder die TR 440 mit 0,9 μ sec. Speicherzyklus für die Datenübertragung.

Doch ist es aber nicht ganz so einfach, es gibt Methoden, die scheinen etwas zu verbessern. Im Grunde genommen aber, ist die Sache sehr traurig was im System vor sich geht. Allerdings ist zu beachten, daß der angegebene Speicherzyklus jeweils pro Wort genommen ist. Sie werden sagen, na ja, also bei der TR 440 mit 48 Bits oder 6 Zeichen zu 8 Bits pro Wort entsteht sodann kein Problem, wenigstens wäre die Datenübertragung möglich. Doch haben wir z.B. auch schnelle Trommeln.

Es gibt die IBM 2305/1 Trommeln mit 3 Millionen Zeichen pro Sekunde Übertragungsrate. Da wäre es wirklich sinnlos die Zeichen einzeln zu übertragen, sagen wir, die Trommel könnte sonst an das 360/50 System nicht angeschlossen werden, da nur etwa 0,3 μ sec. zwischen den Zeichen zur Verfügung steht. Zur Lösung könnten wir einen Pufferspeicher im Zentralrechner vorsehen, ich könnte z. B. sagen: ich lese immer 6 oder 7 Akkumulatoren auf und speichere die dann ab.

2.2.6.3 Zusammenfassung der Daten

Jedenfalls das Problem existiert einmal dadurch, daß manche E/A-Geräte an manche Rechner gar nicht ange-

geschlossen werden können. Z.B., wenn man jetzt, was tatsächlich gemacht wird, die Zeichen zusammenfaßt, z.B. zu Wörtern, Doppelwörtern, usw., dann könnte eine Lösung gefunden werden. So kann ich bei der IBM 360/85 Quadruple-Wörter übertragen. Wenn ich ein volles Wort übertragen will, dann habe ich 12 msec. Zeit, zur Verfügung für die Abspeicherung, doch das reicht aber bei der 360/50 nicht, und zwar deshalb nicht, weil ich mit einem Betriebssystem arbeiten muß, das auch noch ein paar Instruktionen braucht.

Wenn wir das System analysieren, so sehen wir, daß selbst bei größeren Aggregaten (z.B. Quadruple-Wörtern) der Vorgang einfach zu schnell, viel zu schnell, verläuft, z.B. für das Modell 50. Wenn ich ein so schnelles E/A-Gerät verwenden will, müßte ich dann eben sehen, daß ich ein schnelleres System habe, z.B. eine 360/85 (wegen des Schnellzugriffsspeichers bleibt für den Zentralrechner noch etwas zu tun übrig).

Zusammenfassend, es tritt möglicherweise dieselbe Erscheinung wie beim Vorlesen mit entsprechenden Datenmengen auf oder fast wie beim Warten, es verlangsamt sich das System und wir haben genau denselben Effekt wie vorher.

2.2.6.4 System-Änderung

Ein Vorschlag wäre, es mit einem Doppelbus zu versuchen. Das wäre die nächste Stufe. Doch wird dies wesentlich teurer, weil eine entsprechende Kontrolleinheit dazu kommen muß. Sie ist sehr kompliziert. Auf was dieser Vorschlag hinausgeht ist folgendes: wenn die Übertragung der Daten im System so schlecht ist, warum werden noch solche Methoden verwendet, und zwar heute noch, in der heutigen Technologie. Die Rechner, die auf dem Markt sind, verwenden diese Me-

thoden des Zyklenstehleins bis zu großen Systemen,
bis in die unteren Grenzen der Großmaschinen.

2.2.7 Speicher-Schachtelung

Was die Großmaschinen verwenden ist die nächste Frage. Wie läßt sich dieses Problem etwas verbessern oder eine Lösung, eine bessere Lösung finden für dieses Problem was besonders bei Großmaschinen verlangt werden muß (große Systeme in der Art der CDC 7600, Cyber 170, 360 - 370/195, die größten Systeme, die existieren). Dort tritt folgendes auf: diese haben eine andere Speicherstruktur als bisher angenommen. Wir haben hier eine Speicherschachtelung.

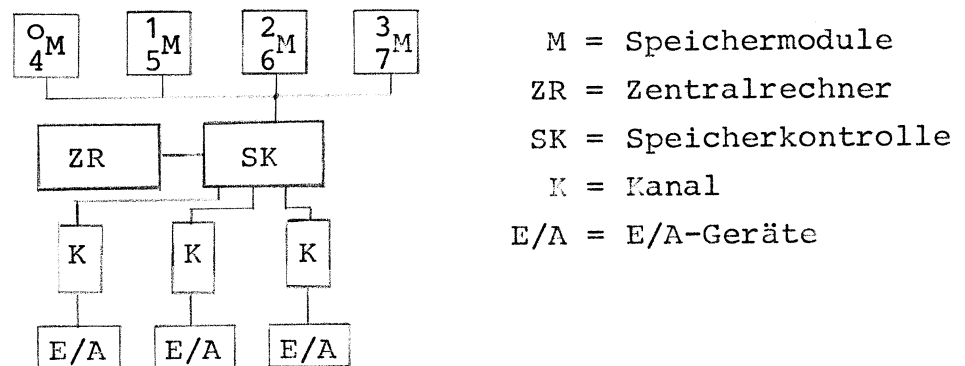


Abb. 10: Speicherschachtelung

Das sieht folgendermaßen aus: wir haben hier einen Zentralrechner (ZR), der an eine Kontrolleinheit geht, das ist die Speicherkontrolle (SK). Die Speicherkontrolle arbeitet ebenfalls mit der Speicheranordnung. Und zwar existiert dieser Speicher nicht in einem monolithischen Block, sondern in Modulen, d.h., 4 Speichermodulen im Bild, jeweils werden 2,4,8,16, usw. Module verwendet. An die Speicherkontrolle wird auch der Kanal oder die Kanäle angeschlossen, wenn wir mit mehreren Kanälen arbeiten, möglicherweise über Anpasser. Daran hängen nun

E/A-Geräte.

2.2.7.1 Adressenzählung

Die Idee bei der Speicherschachtelung ist folgende: Die Adressen zählen nicht mehr in jedem Modul fortlaufen durch, sondern springen von einem Modul zum anderen, z.B. die Adresse 0 gehört zum Modul 1 und die Adresse 1 zum Modul 2, 2 zum Modul 3, 3 zum Modul 4 und 4 wieder zum Modul 1 usw.

In einem normalen Programm werden die Instruktionen und auch viele Daten zu einem großen Teil fortlaufend aufgerufen in fortlaufender Reihe. Darauf beruht nun die Wirkungsweise des verschachtelten Speichers. Hat nun der Speicher, sagen wir, 400 nsec. Zykluszeit, dann kann ich in einem Zyklus von jedem dieser Module Ein Datenelement ein- oder auslesen. Wenn nun der Zeitgeber für die Module nicht gleichzeitig abläuft, sondern zeitlich für jeden etwas versetzt, dann habe ich effektiv wohl die Erscheinung, daß jeder Modul nach 100 nsec. ansprechen könnte. Wenn eine Reihe durchlaufen ist, dann sind 400 nsec. verbraucht, die Daten oder Instruktionen sind nun wieder bereit für den ersten Modul und ich kann ihn wieder in Betrieb setzen und das nächste Datenelement verlangen. Damit haben wir nun die Zykluseinheit durch die kleinere effektive Zykluszeit zu ersetzen. Diese ist die normale Zykluszeit dividiert durch die Zahl der Moduln. Leider werden in normalen Programmen Daten und Instruktionen nicht nur fortlaufend aufgerufen. Und zwar haben wir Sprungbefehle, wobei die Sequenz durchbrochen wird, d.h., ich habe einige Instruktionen in der fortlaufenden Reihenfolge und das Programm springt nun heraus, dann sind die drei nächsten, schon aufgerufenen Adressen in unserem Beispiel gewöhnlich unbrauchbar.

2.2.7.2 Effektive Zykluszeit

Die effektive Zykluszeit (Z_e) ist kleiner, höchstens gleich der normalen Zykluszeit (Z_n) geteilt durch die Zahl der Moduln (N).

$$Z_e \leq \frac{Z_n}{N}$$

Es ist also das Ideal, das wir anstreben. Wenn wir Daten oder Instruktionen haben, die wieder mit einem Vielfachen der Modulzahl fortschreibt, dann bricht ebenfalls der Vorteil der Speicherschachtelung zusammen.

Das ist also die Methode mit der bei großen Systemen gearbeitet wird. Wie arbeiten nun hier die Kanäle mit der Speicherkontrolle des Systems?

Die Speicherkontrolle arbeitet mit einer Adressenschlange, in die die aufgerufenen Adressen vom Zentralrechner, und die von den Kanälen eingehen, und zwar haben die Adressen von den Kanälen Vorrang vor den Adressen für den Zentralrechner.

2.2.7.3 Vergleich mit den bisherigen Methoden

Im Grunde genommen ist nur das alte Problem auf eine andere Ebene geschoben. Wir haben angefangen das ganze System zu verlangsamen, dann haben wir das System verlangsamt nur im Falle mit großen Datenmengen bei der Vorauslesung, oder dann das System verlangsamt nur während der E/A-Vorgänge, oder bei der Parallelarbeit durch Zyklenstehlen, wenn tatsächlich Daten für den Speicher bereitliegen. Hier haben wir nun einen Eingriff in die Adressenschlange, d.h., die Adressen des Zentralrechners werden verdrängt von einer Anforderung eines Kanals. Der Kanal hat mit der Datenübertra-

gung auf die Speicherzyklen denselben Effekt wie ein Sprungbefehl in einem Programm, und zwar deshalb, weil der Zentralrechner nicht mit jeder beliebigen Adresse arbeiten kann. Er muß ein spezielles Programm bearbeiten und die nächste Instruktion im Programm ist vorgeschrieben oder gewisse Daten sind verlangt.

Auch dieses Problem ist nicht neu, es wurde frühzeitig erkannt, und welche Lösungen vorgesehen werden, durch die Kanäle, war schon angedeutet worden. Sie haben gesehen, wir haben versucht hier nicht nur Zeichen abzuspeichern, sondern versucht die Zeichen zu Konglomeraten zusammenzufassen. Genau dieses Prinzip wird nun hier bei den großen Systemen versucht.

2.2.7.4 Verwirklichung

In den alten, mittleren und Kleinsystemen haben die Kanäle einen Pufferspeicher von Zeichen- oder Wortgröße. Die Kanäle haben hier Zusatzspeicher, d.h., bei der CDC bis zu 15 periphere Prozessoren mit der Möglichkeit bis zu 65.000 Wörter zu speichern. Dies bedeutet, daß wir vorher Datenübertragung vom und zum Puffer in den Hauptspeicher hatten. Nun haben wir einen Nebenspeicher im peripheren Prozessor, so daß er nicht mehr die einzelnen Daten im Hauptspeicher speichern muß. So hat CDC einen Sonderspeicher entwickelt, bei dem 1.000 Wörter in 100 nsec. übertragen werden können. Ich weiß nicht, ob er hier bei diesem Problem verwendet wird, ich bezweifle es, und zwar deshalb, weil sehr exakte und besondere Anforderungen an die Arbeitsweise des Speichers gestellt werden. Immerhin, Sie sehen, ein Ausweg wird auf diese Weise gesucht.

Einzelheiten über die angedeuteten Gebiete, die Auslastung der Systeme, der E/A-Geräte, die Entwurfskri-

terien der Systeme, darauf wird später mehr eingegangen, besonders auf die Unterbrecher- und Kanalwerke, die natürlich eine zentrale Stellung in einem System einnehmen. Die Ausführlichkeit hier dient dazu, einen Überblick zu bekommen.

3. Geschichte der E/A-Geräte

Die Geschichte der E/A-Geräte ist nicht besonders interessant, durch Hinweise auf die Zusammenhänge dürfte es ein wenig interessanter werden.

Es sind Beiträge in einem Buch ¹⁾ gesammelt und Einleitungen dazu geschrieben worden.

Uns interessiert besonders die Geschichte der E/A-Geräte im Zusammenhang mit den Rechnern. Rechengeräte lassen sich in drei Gruppen einteilen: mechanische, elektro-mechanische und elektronische Rechner.

3.1 Mechanische Maschinen

Betrachten wir zuerst die mechanischen Maschinen und davon die Ziffernrechner. Diese wurden von Schickhardt, Pascal und Leibnitz vorgeschlagen. In neuester Zeit sind elektronische Versionen solcher Ziffernrechner als Taschenrechner auf dem Markt. Bei diesen Geräten ist die Ein- und Ausgabe: mit Handeingabe und optischer Anzeige.

Am Anfang des Letzten Jahrhunderts hat Babbage's Sohn ebenfalls Handeingabe und Ausdrucken bei einem Rechner benutzt, der nach den Ideen seines Vaters gebaut wurde. Ausdrucken wird auch heute noch als Ausgabe verwendet, wenn auch nicht mehr in der alten Form. Interessanter sind die programm-kontrollierten Rechner in der ersten Gruppe; etwa 1937 - 1944

1) "The Origins of Digital Computers"
B.Randel (Ed.) Springer 1973

die MARK I an der Harvard University unter Aiken mit Unterstützung der IBM. Dabei wurden Lochkarten zur Ein- und Ausgabe verwendet. Die Lochkarten selber waren keine Erfindung der IBM. Sie gehen zurück bis in die Mitte des letzten Jahrhunderts, wo ein Herr Jaquard Webstühle damit zur Steuerung der Webmuster auf den Webstühlen betrieben hat. Dann hat Hollerith die Idee der Lochkarten wieder aufgegriffen und damit Volkszählungen in den Vereinigten Staaten Ende des 19. Jahrhunderts durchgeführt. Die Firma Hollerith, von ihm gegründet, wurde von Herrn Watson übernommen und entwickelte sich in die heutige IBM. Deshalb wohl schlagen wir uns immer noch mit den Lochkarten herum.

3.2 Elektro-mechanische Maschinen

Die nächste Gruppe von Rechnern, die elektro-mechanischen Maschinen (nur die programm-kontrollierten interessieren uns hier) gehen auf die Mitte der 30-er Jahre zurück. So hat Stibitz in den Bell-Laboratories einen Relais-Rechner für komplexe Zahlen gebaut. Damals waren als E/A-Geräte Fernschreibmaschinen verwendet worden, welche gerade für den Fernschreibverkehr entwickelt worden waren.

Eine ähnliche Maschine wurde von Zuse 1941 in seiner Waschküche entwickelt, die Z-3, mit Lochstreifen-(Fernschreiber) Eingabe und optischer Anzeige als Ausgabe. Leider ist nichts mehr davon da, keine Unterlagen mehr vorhanden, Herr Zuse hat nichts darüber veröffentlicht.

Rückblickend bietet sich eine interessante Erscheinung und zwar ist es folgendes: Die Rechner mit denen wir heute arbeiten, basieren auf einer speziellen Technologie, ebenso wie die elektromechanischen Maschinen, die auf der Technologie der Relais basieren. Letztere waren zu Beginn der 1930-er Jahre für die Telefonschaltungen verwendet, und einigermaßen zur Vollkommenheit entwickelt worden und zuverlässig arbeitend.

Zur selben Zeit arbeitet die IBM an einem "Sequence Relay Calculator", dieser wurde 1944 fertiggestellt. Etwas später, und zwar 1946, hat die IBM eine 600-er Serie hergestellt, ebenfalls mit Lochkarten als E/A. Die ausgegebenen Lochkarten wurden tabelliert, also ausgedruckt. In der Folge dieser Maschinen entstanden insbesondere die 603 und die 604. Die 604 hat dazu als eine der ersten Maschinen eine neue Technologie verwandt, das war die Elektronik (auch hier beschäftigen wir uns nur mit programm-kontrollierten Maschinen).

3.3 Elektronische Maschinen

Ähnlich wie bei den elektromechanischen Maschinen existiert hier eine spezielle Technologie in der Elektronik, aus der Radar- und Impulstechnik der Kriegsjahre herausgewachsen sind. Der erste elektronische Rechner war die ENIAC (Electronic Integrator And Computer), unter der Aufsicht von Mauchly und Eckert, 1946 an der University of Pennsylvania, entwickelt. Die Maschine arbeitete mit Stecktafel-Programmen, wie bei Analog-Rechnern. Bei dieser Maschine wurden ebenfalls Lochkarten als Ein- und Ausgabe verwendet. Die Programmierung ist sehr umständlich und nicht besonders gut und deshalb haben die Leute, die die Maschine gebaut haben, nunmehr bessere Programmiermöglichkeiten gesucht. So kamen sie auf die Idee, das Programm ebenfalls in den Speicher hineinzustecken, also eine speicherprogrammierte Maschine. Die Idee wird von Neumann zugeschrieben, d.h., er hat sie zuerst veröffentlicht.

Die Nachfolger der ENIAC Maschine war die EDVAC Maschine mit dieser Möglichkeit. Von Neumann hat sich um die Theorie der elektronischen Rechenmaschinen sehr verdient gemacht und zwar in solchem Maße, daß die heutigen elektronischen Digitalrechenmaschinen noch als "von-Neumann-Typ" bezeichnet werden. Es wäre noch die EDSAC Maschine von Wilkes zu erwähnen. Wilkes kam nach Philadelphia, zur University of Pennsylvania, zu Studienbesuch und hat in der Forschungsgruppe

der ENIAC mitgearbeitet. Die DSAC arbeitet zur Ein- und Ausgabe mit Lochstreifen und der Fernschreibmaschine, eine Technologie, die ihm in England zur Verfügung stand.

Die Reihenfolge der Rechner führt uns weiter von der ENIAC über die EDVAC zur BINAC, für deren Entwicklung und Herstellung Eckert und Mauchly eine kleine Gesellschaft gegründet hatten, wie in den USA üblich. Die BINAC war 1950 fertiggestellt, und daraus wuchs die nächste Maschine, die UNIVAC I, das war 1951, ebenfalls von Eckert und Mauchly. Das war die erste kommerziell erhältliche Maschine gewesen. Und zwar wurde die Gesellschaft später, die UNIVAC-Gesellschaft, von der Sperry Rand Corp. übernommen.

Die letzte Maschine zeichnete sich durch einen Trommelspeicher als Hauptspeicher aus und verwandte Lochkarten (die 90-Spalten-Version) zur Ein- und Ausgabe; und zwar deshalb, weil die Sperry Rand Corp. seinerzeit auch die Remington Rand Corp. übernommen hatte, und die Lochkarten in der Firma vorlagen.

Kurze Zeit später wandte sich IBM, nicht müßig, ebenfalls der Rechner-Weiterentwicklung zu, nach dem Erfolg der 600-Serie, von der über 10.000 Maschinen insgesamt verkauft wurden. Es wurden die 701, 702 usw. Maschinen nach den Vorschlägen von Neumanns entwickelt in den Jahren 1952 bis 1953. Als Konkurrenzmaschine zur UNIVAC I wurde 1956 von IBM die 650 entwickelt, und zwar mit Magnetband E/A, neben Lochkarten, vor allem als Zwischenspeicher, wobei ebenfalls Trommelspeicher als Hauptspeicher vorlag.

3.4 E/A-Geräte

Nun, Sie sehen, es zeigt sich, daß als E/A-Geräte von Anfang an für unsere Rechner, Lochkarten, Fernschreiber mit Lochstreifen und Ausdrucker vorlagen, und zwar als marktübliche Geräte. Nachfolgendes Bild Abb. 11 zeigt den

Stammbaum der Rechner und Hauptrechner mit der ENIAC als Wurzel, die Hauptwurzel der Elektronenrechner.

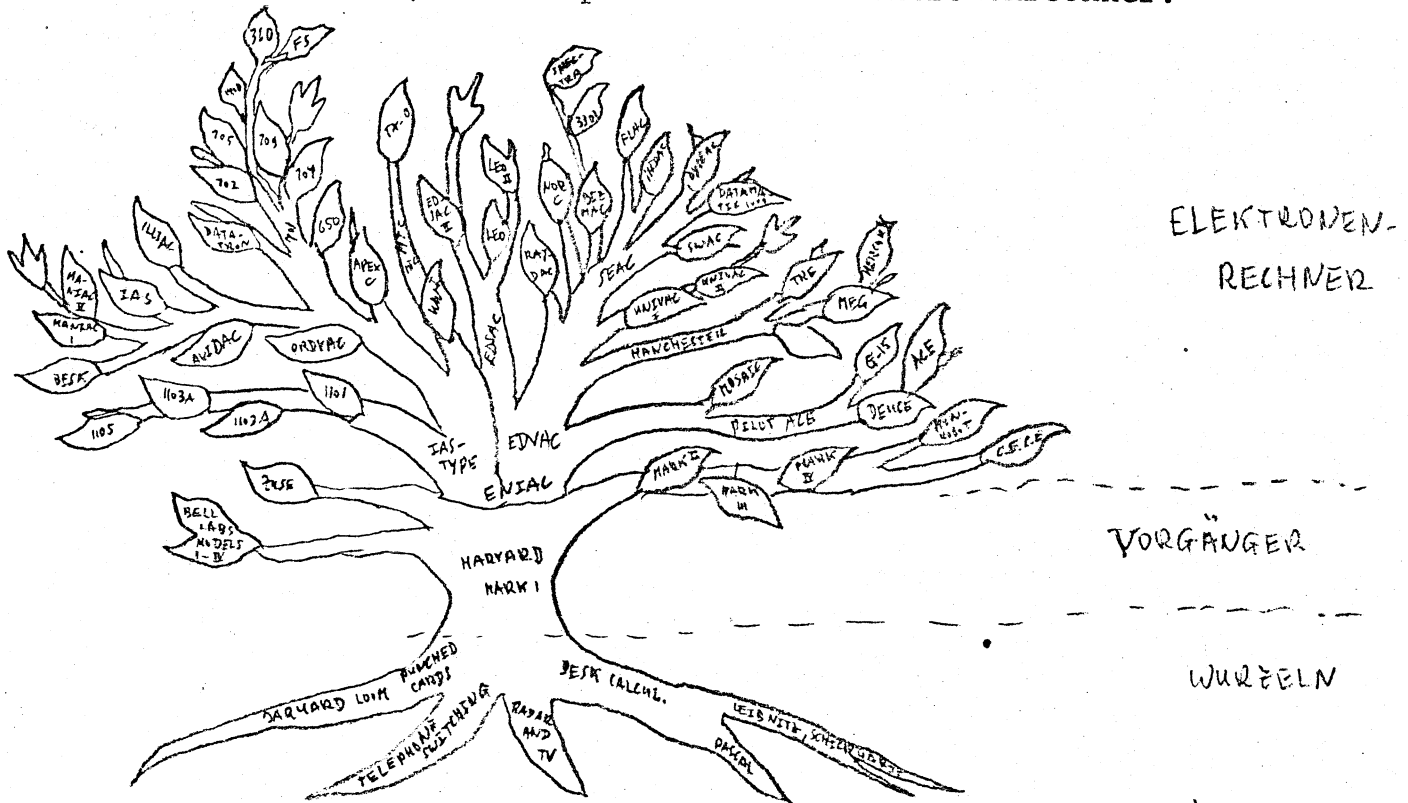


Abb. 11: Stammbaum der Elektronenrechner

(nach NSF)

Alle Firmen auf dem Gebiet der Elektronik, die etwas von sich hielten, haben angefangen, anfangs der 50-er Jahre Rechner zu bauen, meist mit geringem Erfolg. Es gibt nur noch wenige Firmen, die Großmaschinen herstellen; für die kleinen Maschinen dagegen existiert noch eine größere Zahl noch zur Zeit.

3.4.1 Datenstationen

In neuerer Zeit haben wir an E/A-Geräten die Erscheinung der Datenstationen, wozu in erster Linie Fernschreiber verwendet wurden (eine alt bewährte Technik). Dazu neuerdings Datensichtgeräte, seit etwa 1965 bis 1966, und Bildschirme. Wir haben die Plotter seit etwa derselben Zeit. Dazu kamen dann die Satellitenstationen als konglomerate von E/A-Geräten, wobei Lochkarten und Schnell-

drucker hauptsächlich als E/A-Geräte in Frage kommen. Die Idee des Schnelldruckers kam von der Tabelliermaschine her, wobei erst die Ergebniskarten ausgestanzt und dann tabelliert wurden. Von hier war es nicht weit, die Idee zu haben, warum man nicht die Tabelliermaschine direkt an den Rechner anschließen kann, dann braucht man die Lochkarte als Zwischenstufe beim Ausdrucken nicht zu durchlaufen. Natürlich waren von den Tabellierern her die Maschinen anfangs sehr langsam. Die nächste Stufe für die Schnelldrucker war, diese zu beschleunigen.

Die neueste Entwicklung in den letzten drei Jahren sind intelligente Terminals, das sind solche Terminals, bei denen die Daten in der Datenstation selber vor- und nachbearbeitet werden; nicht nur Sammeln in Puffern, sondern z.B. Übersetzen in einen anderen Kode, oder eine andere Zeichendarstellung, oder sonst irgendwie anders aufgearbeitet werden.

Die Entwicklung integrierter elektronischer Bauteile, auf dem Markt erhältlich, hat dazu beigetragen, die Kapazität und die Arbeitsweise der Terminals etwas zu verbessern.

Als neuere Idee wurden gewöhnliche Fernsehschirme als Ausgabe-Geräte vorgeschlagen, doch dürfte es noch einige Zeit dauern, bis diese anschlußfähig sind. Dies sind die eigentlichen Ausgabegeräte.

3.5 Zwischenspeicher-Geräte

Ich habe auch schon die Zwischenspeichergeräte erwähnt. Wir hatten schon die Lochkarten, die nachher tabelliert wurden. Dies ist nichts anderes als ein Zwischenspeichergerät, allerdings nicht Zwischenspeicher für den Rechner, sondern ein Zwischenspeicher für den Benutzer. Dagegen

ist das Magnetband, das um 1950 auftrat, ein echtes Zwischenspeichermedium. Später kamen dazu neue Zwischenspeichertypen im Plattenspeicher und im Großkernspeicher (in etwa der Mitte der 60-er Jahre).

Als neue Entwicklung sehen wir Kassettenbandspeicher, speziell bei kleineren Maschinen und auch für Satellitenstationen. Diese arbeiten ähnlich wie bei den Tonbandgeräten, doch natürlich mit Hochleistungskassetten.

Es sind auch andere Medien versucht worden, z. B. Drahtspeicher mit laufendem Draht, doch haben diese sich bisher nicht bewährt.

Das war eine kleine Abhandlung über die Geschichte der E/A-Geräte, insbesondere im Zusammenhang mit den Rechnern, die wir besonders festhalten wollten.

Siehe auch über die Entwicklung der Rechner, Technologie und Rechner Saul Rosen, in Proc. SJCC 1964, und Computing Surveys.

4. Charakteristiken der Daten

Vieles, was wir in diesem Kapitel behandeln, wird Ihnen von anderen Vorlesungen bekannt sein. Wir haben schon den Begriff "Daten" verwendet, ohne ihn zu definieren, dazu gibt es noch den Begriff "Information".

4.1 Daten und Information

Daten und Information sind im allgemeinen ein und dasselbe im allgemeinen Sprachgebrauch. Man spricht von Datenverarbeitungsmaschinen und Informationsverarbeitungssystemen und meint dasselbe. Manchmal, das hängt dann vom Zusammenhang ab, bedeuten aufbereitete Daten Information.

Dies ließe sich so definieren, daß Information die brauchbaren oder nützlichen Daten sind; in diesem Falle wären die Daten eine echte Übermenge der Information:
Daten \supseteq Information.

Leider gibt es nichts, das den Zusammenhang zwischen Daten und Information besser bestimmt. Dies ist ein sehr wichtiges Gebiet, die genaue Unterscheidung zwischen Daten und Information wäre sehr erwünscht und auch Genaueres zu wissen über, z. B. die Bestimmung des Anteils an nützlichen Daten in einer allgemeinen Datei. Dies ist ein bedeutendes Forschungsgebiet, woran besonders die Informationswissenschaftler interessiert sind. Für ein Weiterkommen sind Anzeichen vorhanden, z. B. mit einem Kontext oder mit dem Zusammenhang in dem man evtl. die Daten erscheinen lassen kann. Es ist also sehr schwierig, hier etwas zu entwickeln. Jemand müßte eine sehr gute mathematische und formelle Ausbildung besitzen, um es generell definieren zu können. Dies dürfte wenigstens im großen Rahmen möglich sein, um Einzelheiten geht es vorerst nicht. Jeder Fortschritt wäre mehr, als unsere einfache Definition liefern kann.

Sie haben den Ausdruck "GIGO" gehört, "Garbage-In-Garbage-Out", d.h., Daten werden in die Maschine eingegeben und andere Daten kommen heraus, ohne daß sie recht verstanden werden - genau das ist das Problem. Bevor ich die Daten definiere, will ich den Zweck der Daten angeben.

4.2 Zweck der Daten

Der Zweck der Daten ist das Festhalten von Größen und Werten, d.h., eigentlich sind es die Datenplätze zum Festhalten von Größen und Werten, z.B. der Kontenstand eines Kunden bei der Bank, oder Temperaturen, Luftdruck beim Wetteramt, usw. die aufgenommen werden. Dies sind die Daten, die

jeweils dort anfallen. Dabei tritt das Problem der Feststellung und Messung der Daten auf, oder die Frage, wie werden die Daten festgestellt, wie werden die gemessen, wie werden die Daten verifiziert?

Letztere Gebiete sind außerhalb unseres Bereichs, es sind eigene Forschungsgebiete, schon recht weit entwickelt. Die Physiker haben dasselbe Problem des Messens und der Verifizierung der Daten, darauf können wir hier leider nicht eingehen.

4.3 Daten-Zuordnung

Die Größen und Werte hängen nicht in der Luft, sie sind zugeordnet zu Individuen und/oder Kategorien. Diese Individuen oder Kategorien möchte ich als Subjekte bezeichnen. Die Größe und Werte werden selber oft als Attribute bezeichnet. Unter diesen Größen und Werten haben wir zwei Klassen zu unterscheiden. Wir kommen wieder zurück auf die Daten und die aufbereiteten Daten; wir haben unter den Attributen die bedeutsamen und unbedeutsamen. Z.B. in der Personaldatei eines Betriebs erscheinen Namen, Alter, Kinderzahl, gearbeitete Stunden, usw. eines Arbeitnehmers, dagegen nicht dessen Pulszahl und Körpertemperatur als Größen und Werte festgehalten. Letztere sind bedeutsam in der Patientendatei eines Krankenhauses, dagegen spielen hier die gearbeiteten Stunden keine bedeutende Rolle. Es existiert eine Dichotomie: die wichtigen und die unwichtigen Daten.

4.4 Daten-Verschachtelung

Die Subjekte haben Objekte zugeordnet für verschiedene Attribute.

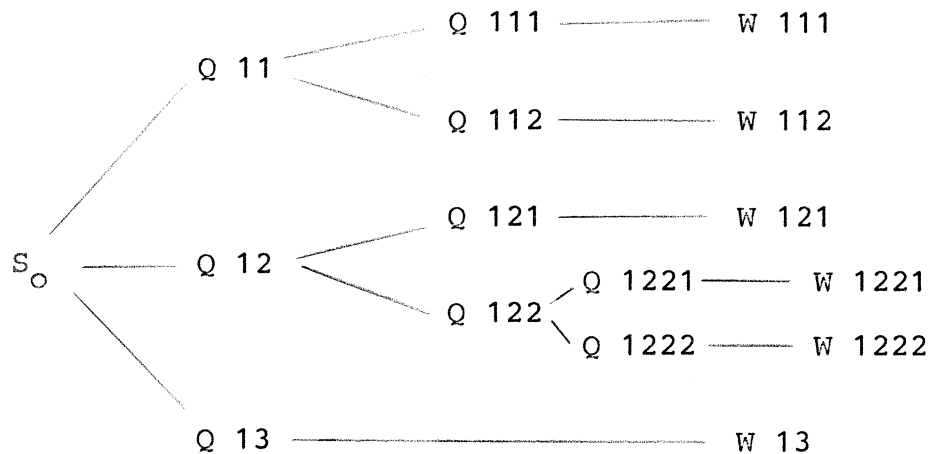


Abb. 12: Datenverschachtelung

z.B. nehmen wir das Subjekt S_0 . Dies hat die Objekte $Q 11$, $Q 12$ und $Q 13$ direkt zugeordnet. Diese Objekte haben z.T. wiederum Objekte zugeordnet. Erst auf der letzten Stufe erscheinen die Werte W .

4.4.1 Stufe der Daten

Sie sehen eindeutig, daß die Werte der letzten Stufe der Objekte in der Verschachtelung zugehören, z. B. Subjekte sind die Bankkunden einer Bank, wobei man Kontonummer, Adresse, Kontostand, Zinssätze usw. als Objekte hat. Sie sehen, die Werte erscheinen in dieser Verschachtelung, in dieser Hierarchie, nur auf der Endstufe. Es ist nichts anderes als eine Baumstruktur. Die Verschachtelung richtet sich natürlich nach der Art und dem Typ der Datei, wobei sehr viele Variationen davon möglich sind.

4.4.2 Reihenfolge der Daten

Wenn die Organisation (d.h., die Organisation ist die Art und der Typ der Verschachtelung) festgelegt ist, dann ist es nur nötig, die Werte in einer gewissen Rei-

henfolge festzuhalten. Was wir dabei haben, ist eine implizierte Struktur der Datei. Z.B. sagen wir: 12, 8; 753; 13, 3; 755; 13, 5; 756 usw. Das könnten dies dann die Temperaturen und die Barometerstände in einer Datei von Meßwerten für ein Wetteramt sein. Wir haben die Temperatur und den Barometerstand um 11 Uhr, ebenso dann um 12 Uhr, usw. Sie sehen, die Struktur tritt nicht mehr expliziert in der Datei auf. Daraus wird klar, daß hier die Daten nicht allein verwendet werden sollen, sondern immer ein Datenpaar zusammen. Dies ist eine allgemeine Erscheinung von Dateien, daß die Daten meist nicht einzeln auftreten und alleine sinnvoll sind, sondern sozusagen nur als Pakete und Quanten. Dies sind Agglomerationen von Daten, die irgendwie zusammengehören, aus der Datei selbst nicht erkenntlich. In unserem Beispiel könnte man z. B. immer drei Werte zusammenfassen, aus den Zahlenwerten weiß man es nicht. Vielleicht könnte man in unserem speziellen Beispiel aus den Größenordnungen der Zahlen schließen, daß es zwei Reihen von Meßwerten sind. Dies ist ein Sonderfall, manchmal möglich zur Analyse.

4.5 Datenquanten

Wir haben hier die Daten zu Quanten zusammengefaßt oder agglomeriert. Es ist nun folgendes: ein Unterschied existiert nun in der Datenteilung in Quanten, je verschiedenen für Benutzer und Komponenten des Rechnersystems. Jeder teilt die Daten verschieden ein.

4.5.1 Benutzerquanten

Der Benutzer verwendet die Daten in Benutzerquanten. Diese basieren auf der Dateienorganisation und der Art der durch ihn leichtesten Verarbeitung (im Beispiel würde wohl jeder die Werte für einen Zeitpunkt zusammenfassen, und nicht vier Werte für zwei Zeitpunkte).

4.5.2 Computer-Quanten

Dann haben wir in dem Rechnersystem den Rechner, der Computer-Quanten benutzt. Diese Computer-Quanten basieren auf der Verarbeitbarkeit im System. Z.B. Speichereinteilung in Wörter oder Zeichen, oder die Rechereinteilung in Bits oder Bytes, usw.

4.5.3 E/A-Quanten

Dann treten die E/A-Geräte an, die uns im speziellen interessieren. Sie haben ihre eigenen E/A-Quanten, basierend auf der Abspeicherungsmöglichkeit der Daten vor und nach der Bearbeitung durch den Rechner.

4.5.4 Abbildung und Transformation

Ein Großteil der Programmierung, die für die Lösung eines Problems stattfindet, ist nichts anderes als Abbildung und Transformation zwischen diesen Arten der Daten-Quantisierung dieser verschiedenen Arten der Einteilung der Daten. So existiert in Cobol, wie Sie wissen, eine Daten-Division, wo die Organisation der Daten definiert ist, was genau nichts anderes ist als die Abbildung der Daten auf das Programm. Die Transformation für die Daten ist definiert in Cobol in der Prozedur-Division. Dies ist für uns besonders interessant, weil eben in Cobol Benutzer- und Rechnerquanten getrennt sind, im Gegensatz zu anderen Programmiersprachen.

4.5.5 Bestimmung der Quanten

Sehen wir uns einmal die Benutzerquanten genauer an. Hier wird natürlich die Verschachtelung vom Benutzer aufgestellt, und der Benutzer definiert die Quanten in solcher Weise, wie sie ihm organisatorisch am be-

sten erscheinen. Diese Arbeitsweise ist gut, und zwar deshalb, weil der Benutzer offensichtlich am meisten über die Datei und deren Einteilung und über die Subjekte weiß, oder wissen sollte. Oft ist zur sinnvollen Einteilung einer Datei die Mitarbeit eines System-Ingenieurs nötig, der diese Informationen vom Benutzer erhalten muß (es muß bestimmt werden, was und wie es gegeben ist, ebenso was und wie etwas gesucht werden soll).

4.5.6 Identifizierung

Die Subjekte und Objekte werden zur Identifizierung mit Namen bezeichnet. So z.B.:

<u>Attribute</u>	<u>zugeordneter Wert</u>
Arbeitnehmer	Karl Schmidt
Wohnort	7 Stuttgart 1
Straße	Hauptstraße
Nr.	10
Namen	Werte

Die erste Spalte enthält die Namen, wobei der "Arbeitnehmer" das Subjekt dieser Datei ist, gefolgt von den Objekten. Entsprechend sind dann Werte zu diesen Subjekten in der zweiten Spalte zu finden. Dies ist nicht anders als das uns schon längst bekannte System der Adressen und der Werte, dazu im Speicher des Rechners. Wir haben Namen, das sind die Adressen, und dann Werte dazu.

4.5.6.1 Stufen der Identifizierung

Was sind die Bezeichnung und Identifizierung dieser Objekte und Subjekte in dieser Organisation? Wir müssen dazu folgende Stufen betrachten:

Stufe	Benennung
Gesamtheit	Datei
Subjekt	Daten-Satz
Objekt	Name des Objekts
Attribute	Felder/Unterfelder

Jedem Eintrag in eine Stufe ist eine Benennung zugeordnet. Der Gesamtheit aller Individuen entspricht die Datei. Die Objekte und Attribute können vielfältig auftreten und ergeben damit die Verschachtelung einer Datei. Die Zusammenfassung aller Subjekte zu einer Gesamtheit mit demselben Hauptmerkmal ergibt die Datei (z.B. Arbeitnehmer einer Firma, Bankkunden einer Bank, usw.)

4.5.6.2 Definition

Eine Datei ist die Gesamtheit der bedeutsamen Daten aller Subjekte S, mit derselben Charakteristik.

Ein Datensatz ist die Sammlung der bedeutsamen Daten über ein Subjekt S.

Ein Datenfeld enthält den augenblicklichen Wert des einzelnen Objekts (Attributs), wobei der Name des Feldes der Name des Attributs ist.

Alle Felder sind nicht gleich in ihrer Bedeutung. Eines ragt heraus, das Schlüsselfeld. Dies ist das Feld, das das Subjekt eindeutig identifiziert (z.B. Artikelnummer in der Datei eines Katalogs, oder die Nummer eines Reisepasses, usw.)

4.5.6.3 Organisation der Felder

Nun kommen wir wieder zurück auf das was vorher schon erwähnt wurde: Die Organisation der Felder, d.h., die Art der Organisation für die Objekte. Davon sind zwei Arten zu unterscheiden: explizit und implizit. Explizite Organisation verwendet die Namen der Attribute mit den Werten, z.B.

Name	=	Karl Schmidt
Wohnort	=	Stuttgart 1
usw.		

Die Art ist die implizite Organisation. Hier ist die Reihenfolge und Verschachtelung der Worte in Feldern durch Konvention festgelegt wie seither angenommen; z. B. das 1. Feld enthält jeweils den Vor- und Familiennamen, das 2. Feld enthält den Wohnort, usw. In diesem Falle wissen wir z.B., daß das 5. Feld eines jeden Datensatzes die Hausnummer eines Teilnehmers in einer Fernsprech-Datei enthält. Diese Arten in der Unterscheidung werden auch bei der Darstellung von Makros in Assemblersprachen benützt (Schlüsselwort-Makros und Reihenfolge-Makros).

4.5.6.4 Vor- und Nachteile

Der Vorteil der explizierten Darstellung (zu Lasten des Platzes nötig) ist, daß wir nicht mehr an eine feste Reihenfolge gebunden sind, sondern beliebig vermischt werden können. Der Nachteil ist der, daß für jedes Feld die Schlüsselwörter mitgeschleppt werden müssen.

4.6 Arbeiten mit Dateien

Die nächste Frage betrifft die Arbeiten, der eine Datei unterworfen wird oder werden kann. Diese Arbeiten sind:

Aufrufen (d.h., Abfragen zur Feststellung von augenblicklichen Werten),
Werte-Änderung (die Augenblickswerte der Attribute werden geändert und neue Werte eingetragen),
Datei-Änderung (Subjekte werden zugefügt oder entfernt in der Datei; Änderung der Gesamtheit)
Reorganisation der Datei (die Dateien-Struktur wird geändert, z.B. invertiert). Invertieren einer Datei beinhaltet, daß ein anderes Schlüsselfeld herausgezogen wird.

Im allgemeinen finden wir, daß eine aufsteigende Kompliziertheit der Vorgänge in dieser Reihenfolge auftritt, jeweils für diesselbe Datei betrachtet. In manchen Dateien ist schon das Abfragen von Werten schwierig, umso schwieriger sind dann die Arbeiten der anderen Stufen. Das ist jedoch nicht immer der Fall und zwar deshalb, weil es möglichkeiten gibt, die Dateien entsprechend aufzusetzen, daß eine bestimmte Arbeitsweise bevorzugt wird auf Kosten der anderen, z.B. Dateieinänderung läßt sich leicht vereinfachen auf Kosten anderer Arbeiten, z. B. hier des Aufrufens oder umgekehrt.

Diese Dinge sind wichtig, weil diese die erste Stufe für die E/A für den Rechner darstellen, und die E/A-Organisation beeinflussen, und die somit auch verarbeitet werden müssen.

4.6.1 Rechnerquanten

Innerhalb der Maschine haben wir die Rechner- und Computer-Quanten. Das sind die Arbeitseinheiten des Rechners für die Daten. Sie sind im allgemeinen von denen des Benutzers verschieden. Von den Elementen des Rechners, die die Quanten bestimmen, ist besonders wichtig der (oder die) Speicher. Dabei haben wir wieder eine Tabelle:

Stufe	Größe und Bedeutung
Gesamtspeicher (einschließlich des Schnellzugriffsspeichers)	Größtmögliche im System direkt bearbeitbare Datenmenge
Block/Seite	Datensicherungs- und Übertragungseinheit (bes. bei Paging)
Wörtergruppe	Daten werden zusammen übertragen (z.B. bei IBM 360/85 = 4 Wörter)
Wort } Zelle Zeichen	größte durch den Rechner als eine Einheit bearbeitende Datenmenge, z.B. repräsentiert eine Zahl (binär); repräsentiert ein Zeichen, Ziffer, oder 2 Ziffern (z.B. Dezimalarithmetik der 360/370-Maschinen)
Bit	kleinste bearbeitbare Dateneinheit

Dazu kommen noch als bedeutende Rechnerelemente: die Register, meist in Wortgröße, z.B. Akkumulator, Multiplikator-Quotientenregister. Die Indexregister sind dagegen oft nur von Adressenbreite. Die maschineninternen Bearbeitungsfunktionen sind meist von Wort-, Zeichen-, oder Bit-Größe, je nach Bus-Breite des Rechners.

4.6.2 E/A-Quanten

Die E/A-Quanten sind die Arbeitseinheiten der E/A-Geräte und generell von den vorhergehend erwähnten Quanten verschieden:

Stufe	Bedeutung
Volumen	Gesamtheit der Daten, die an einem E/A-Gerät zur Verfügung stehen ohne Neuladen z.B. bei Bandeinheiten: ein Band (reel); bei Platteneinheiten: ein Plattenstapel oder -turm (discpack);
Block	größte Datenmenge, die zusammen ein-/ausgegeben werden kann in einer Operation. Diese kann eine feste Zahl sein = z.B. für Karteneinheit keine andere Wahl (80 Zeichen) oder variabel, z.B. bei Platte, Trommel und Band;
Wort Zeichen }	E/A-Quanten für Lochstreifen und Datenstationen (arbeiten jedoch auch oft mit Zeilen)

4.6.3 Quanten-Zusammenhänge

Die Quanten der verschiedenen Einteilungen sind nicht unabhängig voneinander, sondern wirken zusammen, z.B. in der folgenden Tabelle ist das Zusammenwirken von Benutzer- und E/A-Quanten festgehalten:

Element	Benutzer-Quanten	E/A-Quanten
Dateien	1 Datei mehrere Dateien (Bibliothek) 1 Datei	1 Volumen: normalerweise eindeutige Zuordnung zu 1 Volumen: multi-Dateien Volumen Mehrere Volumen: multi-Volumen-Datei
Sätze	1 Block 1 Block	1 Satz = ungeblockte Sätze mehrere Sätze = geblockte Sätze Blockfaktor = Zahl der Sätze im Block

Element	Benutzer-Quanten	E/A-Quanten
Felder	1 Wort	1 Feld = Einzelwort-Feld
	mehrere Wörter	1 Feld = Mehrwort-Feld
Unter-felder	1 Zeichen	1 Unterfeld = Einzelzeichen-feld
	mehrere Zeichen	1 Unterfeld = Mehrzeichenfeld

5. Daten-Medien

5.1 Mittel zum Festhalten von Daten

Die Daten-Medien dienen zum Festhalten der Daten, zur Speicherung oder Verarbeitung. Das Medium hat seine Oberfläche verändert oder zerstört durch das Schreiben der Daten. Bei bloßer Veränderung ergibt sich zerstörungsfreies Schreiben, z.B. beim Magnetband. D.h., das Kriterium ist, je nachdem, ob das Medium ohne weiteres wieder verwendet werden kann (z.B. nicht das Überkleben von Stanzungen bei Karten oder Lochstreifen). Zerstörendes Schreiben erfolgt bei Lochkarten und Lochstreifen.

In der folgenden Tabelle sind die Zusammenhänge dargestellt:

Art des Mediums	Schreiben durch	nicht schreiben	Funktionen möglich	Art des Vorgangs
Lochkarten	Lochung	keine Lochung		mechanisch
Lochstreifen	Lochung	keine Lochung	Lesen/ Schreiben	
Schriftlesen	Schreibung	nicht schreiben	lesen	optisch

Art des Mediums	Schreiben durch	nicht schreiben	Funktionen möglich	Art des Vorgangs
Drucker	drucken	kein Drucken	schreiben	i.a.mechanisch
Magnetband, Platte, Trommel Datenzeile	Änderung der Magnetisierung	Magnetisierung im Ausgangszustand	lesen schreiben	elektromagnetisch

5.2 Komponenten der Geräte zur Verarbeitung des Mediums

5.2.1 Anordnung

An den Geräten sind generell folgende Komponenten unterscheidbar: Kontrolleinheit, Pufferspeicher, Bewegungseinheit, und der Geber/Aufnehmer. Die Komponenten sind folgendermaßen angeordnet, siehe Bild 13:

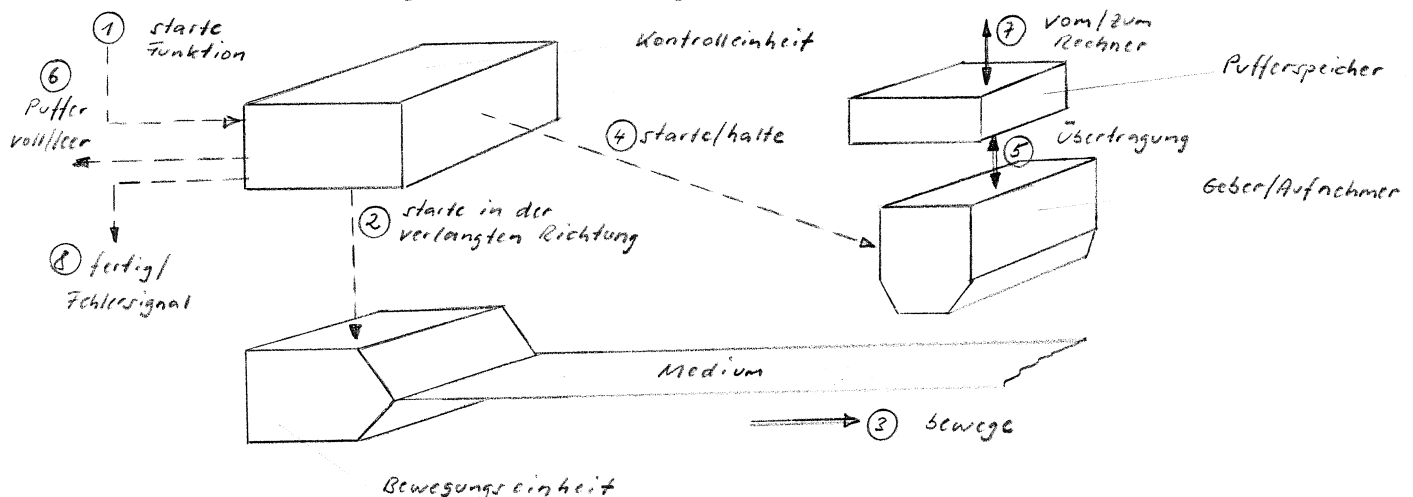


Abb. 13: Anordnung der E/A-Funktionen

Die Kontrollsignalwege sind gestrichelt, die physischen und Datenwege ausgezogen.

5.2.2 Funktionen

Folgende Funktionen sind für die Komponenten gegeben:
Die Kontrolleinheit empfängt Kontrollsignale vom Rechner und vom Benutzer. Sie überträgt die Signale an die Bewegungseinheit und den Geber/Aufnehmer und meldet den erfolgreichen Abschluß der Arbeit zurück, ebenso irgendwelche Wartezustände (Puffer voll/leer) und alle Fehlerzustände.

Die Bewegungseinheit bewegt das Medium in der verlangten Richtung, so daß die richtige Stelle am Geber/Aufnehmer liegt oder dorthin bewegt wird. Sie kann je nach Einheit, einwegig oder doppelwegig arbeiten.

Der Geber/Aufnehmer dient zur Übergabe der Daten vom/zum Medium durch Transformation elektronischer Impulse. So erkennt der Lochstreifenleser die Löcher im Band und erzeugt dafür Impulse. Der Lochkartenstanzer übersetzt die empfangenen elektrischen Impulse in Löcher in den Lochkarten. Der Pufferspeicher hat die Aufgabe des Sammelns von Datenteilen (z.B. Bits zu Zeichen oder Wörtern, Doppelwörtern, usw.)

5.2.3 Arbeitsweise

Die Arbeitsweise erfolgt in der durch die Ziffern gegebenen Reihenfolge. So wird die Kontrolleinheit durch einen Starter-Funktion-Befehl aktiviert (Schritt 1). Dieser wiederum läßt die Bewegungseinheit in der verlangten Richtung starten (Schritt 2), damit bewegt sich das Medium, wie verlangt (Schritt 3). Dann wird der Geber/Aufnehmer aktiviert (oder am Ende angehalten, (Schritt 4), und dieser nimmt die Information vom Medium oder gibt sie dort ab (Schritt 5). Die Daten werden zum/vom Pufferspeicher und von da zum/vom Rechner übertragen (Schritte 6 und 7). Schließlich meldet die Kontrolleinheit besondere Wartezustände (Puffer voll/leer, Schritt 8), oder

auch die Beendigung der Arbeit oder Fehlerzustände zurück.

5.3 Medien

Die Daten-Medien selbst bestehen aus flachen Gebilden aus Papier, Karton, Kunststoff, oder Metall verschiedener Art. Am gebräuchlichsten sind Lochkarte, Lochstreifen sowie magnetische Oberflächen.

5.3.1 Lochkarten

Die bekannten Lochkarten sind Kartenstücke genormter Abmessungen und Qualität. Ebenso genormt ist die Datenverteilung auf diesem Medium (80 Spalten in 12 Zeilen) mit maximal 80 Zeichen pro Karte, sowie die Datenverschlüsselung in der Representierung der Zeichen durch die Lochungen, siehe Bild 14:

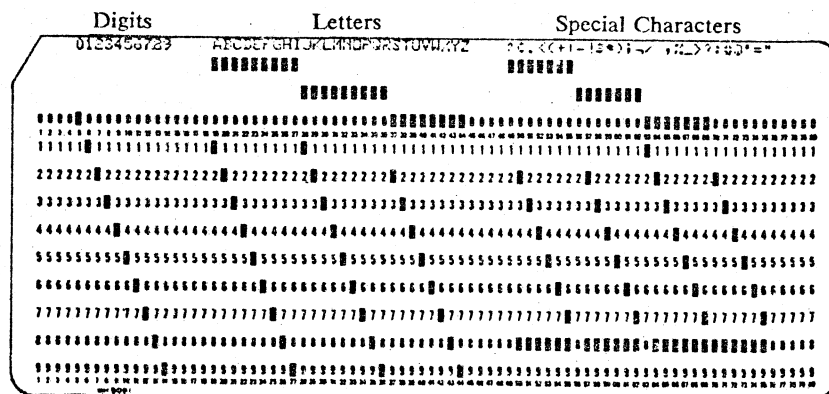
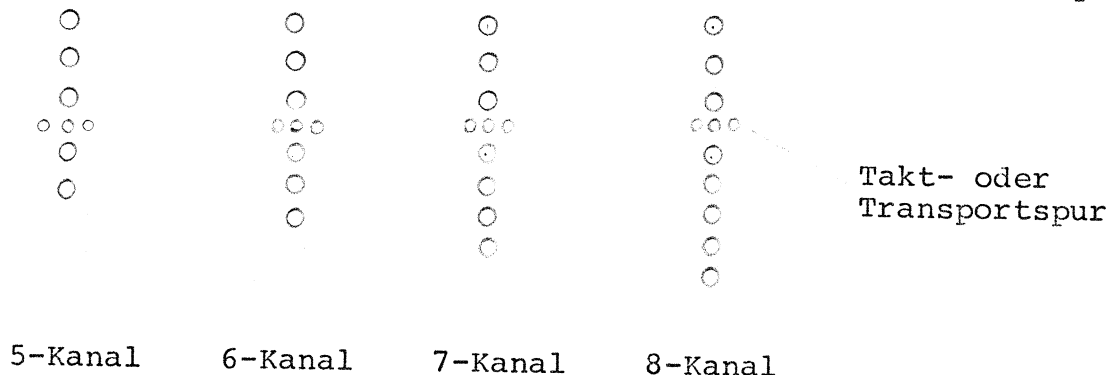


Abb. 14: Karten-Kode

Lochkarten können ebenfalls Daten in Binär-Verschlüsselung aufnehmen, bis zu 960 Bits pro Karte. Dabei kann die Reihenfolge der Bits entweder spalten- oder zeilenweise gezählt werden.

5.3.2 Lochstreifen

Lochstreifen sind Papierbänder, zur Aufnahme von Lochungen geeignet. Es sind zu unterscheiden Streifen mit fünf, sechs, sieben oder acht Kanälen oder Informationsspuren:



Am meisten in Gebrauch sind fünf oder acht-Kanal Streifen. Eine volle Rolle eines acht-Kanal Streifens ist etwa 280 m lang und enthält, wenn voll, bis zu etwa 110.000 Zeichen. Die Kodierung der Information ist meist abhängig vom Hersteller. Am gebräuchlichsten sind der Baudot-Kode und der ASCII-Standard-Kode. Der erstere trifft für die 5-Kanal Streifen zu, siehe Tabelle, und verwendet Umschaltzeichen zwischen Buchstaben und Ziffern oder Zeichen. Der ASCII-Kode findet für die 8-Kanal Streifen Verwendung (siehe Tabelle).

Baudot-Code-Tabelle:

TELETYPE CODE

	High order bits			
	00	01	10	11
Low order bits				
000		line feed	E 3	A -
001	T 5	L)	Z "	W 2
010	car ret	R 4	D ø	J '
011	O 9	G &	B ?	figures
100	space	I 8	S bell	U 7
101	H #	P O	Y 6	Q 1
110	N ,	C :	F !	K (
111	M .	V ;	X /	letters
=====				
letters	37	figures		33
A	3b	o		15
B	23	1		35
C	16	2		31
D	22	3		2o
E	2o	4		12
F	26	5		o1
G	13	6		25
H	o5	7		34
I	14	8		14
J	32	9		o3
K	36	(36
L	11)		11
M	o7	.		o7
N	o6	,		o6
O	o3	-		3o
P	15	?		23
Q	35	:		16
R	12	ø		22
S	24	bell		24
T	o1	&		13
U	34	≠		o5
V	17	'		32
W	31	;		17
X	27	/		27
Y	25	!		26
Z	21	"		21
space	o4	carriage return		o2
line feed	1o			

ASCII-Code-Tabelle:

Character	8-Bit-Code (in Octal)	Character	8-Bit-Code (in Octal)
=====			
A	301	!	241
B	302	"	242
C	303	#	243
D	304	\$	244
E	305	%	245
F	306	&	246
G	307	'	247
H	310	(250
I	311)	251
J	312	*	252
K	313	+	253
L	314	,	254
M	315	-	255
N	316	.	256
O	317	/	257
P	320	:	272
Q	321	;	273
R	322	<	274
S	323	=	275
T	324	>	276
U	325	?	277
V	326	@	300
W	327	[333
X	330	/	334
Y	331]	335
Z	332	↑	336
O	260	→	337
1	261	Leader/Trailer	200*
2	262	Line-Feed	212*
3	263	Carriage-Return	215
4	264	Space	240
5	265	Rub-Out	377*
6	266	Blank	000*
7	267		
8	270		
9	271		

* Ignored by the operating system

5.3.3 Magnetband

Beim Magnetband handelt es sich um einen magnetisierbaren Streifen, bestehend aus einer Plastikunterschicht als Träger mit einer magnetisierbaren Oberfläche, ähnlich der Tonbänder. Standardisiert hat sich ein Halbzoll Magnetband mit etwa 800 m Länge, d.h., genau 2400 ft., falls nicht abgeschnippelt. Daneben gibt es noch ein 1 Zoll breites Magnetband, wenn Sie ein Digital-Equipment-Gerät haben. Es unterscheidet sich von dem Standardmagnetband auch in anderen Dingen, z.B. es gibt dort keine Zwischenräume, Zwischenlücken, Datenlücken, wie wir es sofort sehen werden.

Beim Standardmagnetband existiert ein Schreibschutzring, der das unbeabsichtige Überschreiben von Daten verhindern soll. Meistens erfüllt er seine Funktion. Im Gegensatz zur Lochkarte und zum Lochstreifen wird beim Schreiben der Daten auf dem Magnetband das Magnetband nicht permanent verändert, nicht zerstört, d.h., es ist wieder verwendbar. Die Registrierung erfolgt durch Ummagnetisierung und löscht dabei meist die alten Aufzeichnungen aus.

5.3.3.1 Schreibdichte

Verglichen mit dem Lochstreifen und mit der Lochkarte haben wir beim Magnetband eine wesentlich größere Informationsdichte und zwar bis zu 1.600 Bits pro Zoll. Das wird als Schreibdichte bezeichnet. Es ist nichts anderes als die Sprossen, die wir beim Lochstreifen auch betrachtet haben. Wenn Sie vergleichen, beim Lochstreifen haben wir 5 Sprossen pro Zoll, hier haben wir 1.600 Sprossen pro Zoll, das ist ein wesentlicher Unterschied. Wichtig neben 1.600-er Bits pro Zoll Schreibdichte sind weiterhin noch die 200-er, 556-er und die 800-er. Die 200-er ist die älteste davon, die 556-er ist z.T. heute noch in manchen IBM-Geräten in Gebrauch (728/I oder II) wohingegen die

1.600 und 800 sind die z.Zt. am meist gebrauchten.

5.3.3.2 Bits/Bytes pro Zoll

Die Zählung der Bits erfolgt in der Längsrichtung, das entspricht in der Querrichtung jedem Bit 1 Byte oder Zeichenposition. Darum habe ich es auch mit den Sprossen vom Lochstreifen verglichen. Die Bits werden in der Längsrichtung gezählt.

5.3.3.3 Arten

Wir müssen im wesentlichen unterscheiden zwischen Siebenkanal und Siebenspurbändern und 9-Kanalbändern. Der Unterschied ist, daß der 7-Kanalstreifen 6 Bits für die Registrierung eines Zeichens, plus ein Paritätsbit hat, dagegen das 9-Kanalband $8 + 1$. Es rührt davon her, daß die Registrierung einmal ein 6-Bit-Code ist (z.B. BCD von IBM, daher kommt der Standard überhaupt) und zum anderen ist es ASCII oder EBCDIC als Repräsentationscode der Zeichen.

5.3.3.4 Übertragungsgeschwindigkeit

Herrührend aus der großen Informationsdichte resultiert auch eine größere Übertragungsgeschwindigkeit der Zeichen gegenüber den Karten und den Lochstreifen.

5.3.3.5 Nachteile

Ein großer Nachteil des Magnetbandes besteht darin, daß die Zeichen nicht mehr visuell geprüft werden können, wie man es bei Lochstreifen oder Lochkarte tut, das ist ein Nachteil. Es gibt Geräte, die dies bewerkstelligen können, doch sind sie sehr teuer und sind sehr wenig in Gebrauch, weil es sich nicht

lohnt. Man kann das Band über das System ausschreiben und dann visuell nachprüfen.

Außerdem bedingt durch den Mechanismus der Schreibung und der Lesung, sind Bitgewinne und Bitverluste beim Arbeiten mit Magnetband möglich. Sie sind etwa grob geschätzt in einer Fehlerrate von 1 Bit pro 10^8 (also eins in 100 Millionen Bits). Wenn man ein schlechtes, abgewetztes Band hatte, kann es sein, daß die Fehlerrate 1.000-mal höher liegt als die hier angegebene.

5.3.3.6 Bandabnutzung

Ein schlechtes Band ist eines, das sehr abgeschürft ist, es gibt Stellen darauf, wo keine Registrierung mehr möglich ist, wo der Träger zu wenig magnetsisierbare Schicht enthält. Aus diesem Grunde ist die Paritätsprüfung nötig.

5.3.3.7 Parität

Dafür haben wir nun einmal dieses Paritätsbild (Abb. 15), mit den transversalen Paritätsbits.

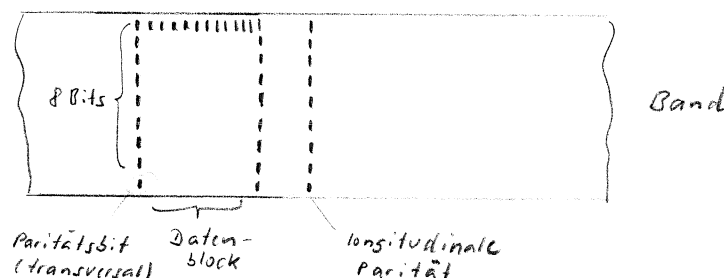


Abb. 15: Parität beim Magnetband

Zu dem transversalen Paritätsbit kommt noch eine longitudinale Paritätsprüfung dazu, d.h., jedesmal wenn ich eine Gruppe von Daten geschrieben habe, wird daraus eine Parität gebildet, die ebenfalls zu Prüfzwecken und zu Regenerationszwecken auf das

Band geschrieben wird. Wir sehen dann später, daß eigentlich oft bei neueren Bändern sogar zwei dieser Paritätsgruppen registriert werden. Bei der Parität gibt es auch wieder verschiedene Möglichkeiten. Wir haben einmal die ungerade Parität, und wir haben die gerade Parität.

5.3.3.8 Schreibdichten

Dann müssen Sie beachten, daß wir die Schreibdichte variieren können, 200, 556, 800 oder 1.600 Bits pro Zoll und dazu die Parität, es gibt also eine Vielfalt von Möglichkeiten, die die Registrierung des Bandes beeinflussen können. Wenn wir ein Band bekommen, müssen diese Daten angegeben sein. Wir müssen lesen können, was die Schreibdichte war oder was die Schreibdichte ist auf dem Band, das gelesen werden soll und auch, was die Parität sein soll. Wenn das nicht der Fall ist, muß es heraus gefunden werden, was möglich aber zeitraubend ist. Es ist deshalb möglich, weil diese Dinge am Laufwerk eingestellt werden können, auf dem Magnetbandlaufwerk, kann die Dichte eingestellt werden und auch manchmal die Parität, nicht bei allen, aber bei den meisten.

5.3.3.9 Vergleich zu Karten und Streifen

Wir haben noch einen anderen Gegensatz zu den Lochkarten und Lochstreifen, und der ist, daß Daten-E/A nur vom und zum laufenden Band möglich ist. Während wir eine Karte von einem Kartenleser lesen können, während die Karte ruht, sie braucht sich nicht zu bewegen, wir können dann allerdings nur ein Zeichen lesen, oder ebenso vom Lochstreifen. Vom Magnetband können wir überhaupt nichts lesen, wenn das Magnetband ruht.

5.3.3.1o Registrierung

Das Magnetband muß bewegt werden, um überhaupt etwas zu zeigen und zwar zeigt Abb. 16 den Registrier- und Lesevorgang.

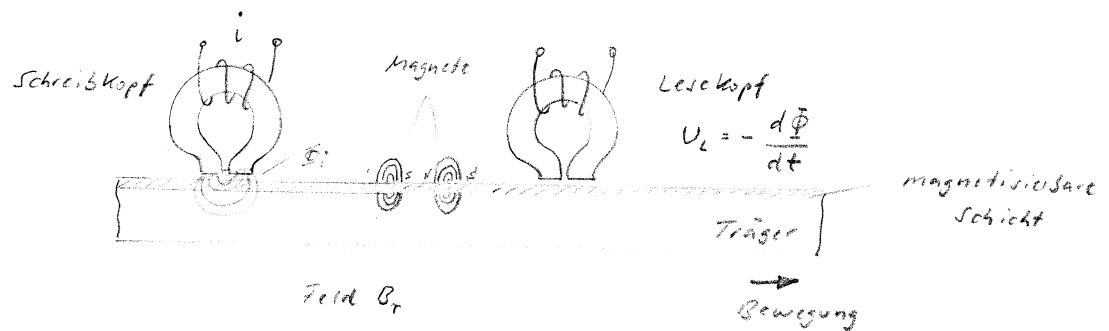


Abb. 16: Registrier- und Lesevorgang

Hier habe ich den Träger aus einem Plastikmaterial, oben habe ich die magnetisierbare Schicht die an einem Lese- und einem Schreibkopf vorbeikommt. Der Schreibkopf besteht aus einem Eisenkern mit einer Spule. Ebenso haben wir einen Lesekopf. Wenn ich einen Strom durch den Schreibkopf schicke, bildet sich ein magnetisches Feld aus; weil nun ein Spalt ist, wird das Magnetfeld hier um den Spalt herum sich ausbilden. Wenn ich hier eine permanent magnetisierbare Schicht vorbei bringe, wird sich in der Schicht ein Feld bilden. D.h., ich erzeuge hier ein paar kleine Magnete in der magnetisierbaren Schicht. Und zwar ist die Feldstärke, sagen wir Φ (i), von dem Schreibstrom abhängig. Dieses Magnetband bewegt sich, und nun, wenn diese kleinen Magnete an dem Lesekopf vorbeikommen, dann erzeugen die eine Induktion in dem Lesekopf, d.h., eine kleine Spannung wird abfühlbar sein an den Drähten, die ich verstärken und dann anzeigen kann, nach der Formel

$$U_L = - \frac{d\Phi}{dt}$$

Je schneller das Band sich bewegt und je stärker das Magnetfeld ist, desto größer ist die Lesespannung. Deshalb ist kein kontinuierliches Lesen und Schreiben möglich. Sie verstehen so, weshalb kein Lesen vom Band im Stillstand möglich ist. Außerdem existiert noch folgendes: Es ist auch kein kontinuierliches Schreiben vom Stillstand möglich.

5.3.3.11 Lücken

Dies führt zu einer weiteren Erscheinung, die den Magnetbändern eigentümlich ist, die bereits erwähnten Schreiblücken, oder Blocklücken. Ich habe hier bereits angenommen, daß hier die Daten zu Blöcken zusammengefaßt sind. Diese Blocklücke wird zum Beschleunigen und zum Verlangsamen des Bandes benötigt. Das Band muß auf die Richtgeschwindigkeit beschleunigt werden, um Lesen oder um Schreiben zu können, und wenn wir mit dem Lesen und Schreiben eines Blockes fertig sind, dann müssen wir das Band wieder anhalten. Dazu braucht es Zeit. Während dieser Zeit läuft das Band natürlich noch ein Stückchen weiter. Das ist die Ursache, dieser Blocklücken, d.h., wir haben hier eine Art Überlauf. Die Blocklücken sind in der Größenanordnung von 1/4 bis 3/4 Zoll. Am häufigsten finden wir 1,75 oder 0,6 Zoll.

5.3.3.12 Brauchbare Länge

Von den bis 2400 ft. Länge des Bandes können wir nicht das ganze Band für Datenspeicherung benützen, d.h., zuerst brauchen wir ein Stück zum Aus- oder Einfädeln des Bandes immer am Anfang oder am Ende, etwa 4 m, durch reflektierende Aufkleber aus Aluminiumfolie gekennzeichnet. Letztere werden mit optoelektrischen Methoden abgetastet. Sobald der Fleck ankommt, weiß die Magneinheit, daß das Band zu Ende ist und stoppt an dieser Stelle.

5.3.3.13 Datenkapazität

An den Enden kann ich keine Daten speichern auf dem Band, wieviel Daten kann ich nun auf dem Rest des Bandes speichern, d.h., wir fragen nach der Datenkapazität eines Bandes. Diese Datenkapazität des Bandes, hängt von der Schreibdicke ab. Je dichter ich schreibe, desto mehr Daten bekomme ich natürlich auf das Band.

Nehmen wir ein Beispiel: Falls wir Karten auf Band lesen, und schreiben nun diese Daten, Zeichen für Zeichen, auf das Band, eine Karte nach der anderen. Das sind dann 80 Zeichen pro Karte, und zwar ungeblickt geschrieben. Dies erzeugt uns mehr leeres Band als Daten. Unser Magnetband wird mehr Zwischenraum enthalten als Daten. Z.B. bei 1.600 Bits pro Zoll und 80 Zeichen in den Karten, dann belegen 80 Zeichen 0,05 Zoll auf dem Band. Die Lücke zwischen den 80 Zeichen beträgt aber, und zwar nehmen wir an, 0,6 Zoll, d.h., umgerechnet haben wir die 12-fache Länge des Zwischenraumes zwischen den Daten. Nehmen wir als Beispiel an: Hier habe ich das Band und hier habe ich die Karten registriert, hier habe ich eine Karte, dann Zwischenraum, dann wieder eine Karte, usw.,

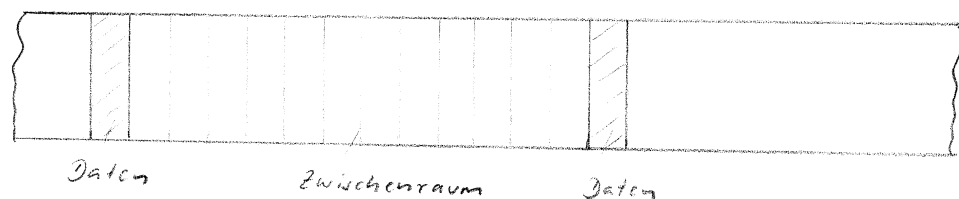


Abb. 17: Längenverhältnisse bei ungeblickten Karten auf Band

dann sieht es etwa so aus, maßstäblich, gesehen, wie in Abb. 17, wo kurze Datenblöcke mit großen Lücken abwechseln. D.h., umgerechnet, wenn Sie es genau betrachten, ich habe etwa $8 \frac{1}{3} \%$ der möglichen Dateninformationsdichte auf dem Band erreicht. Der

größte Teil des Bandes wird verschwendet, um eben nur das Band beschleunigen zu können beim Start und zu verzögern beim Anhalten.

Dies ist ein sehr uninteressanter Fall natürlich, denn wer wird denn ein Band nehmen und es nur mit 8 % mit Daten füllen, d.h., wir müssen den anderen Fall betrachten, wieviel könnte im günstigsten Fall auf ein Band geschrieben werden. Bei der Bandkapazität bei 800 Bits pro Zoll (wenn Sie 1.600 haben, verdoppelt es sich eben), habe ich etwa 23 Millionen Zeichen ohne Lücken. Falls es mir möglich wäre, ein Band ohne Lücken zu beschreiben, könnte ich darauf 23 Millionen Zeichen bei der Dichte 800 daraufbringen, bei 1.600 würde ich natürlich 46 Millionen Zeichen draufbringen.

Mit Karten ungeblockt, habe ich für etwa 4 Millionen Zeichen Platz bei der Schreibdichte 800 BpZ und ebenfalls bei der Schreibdichte 160 BpZ.

5.3.3.14 Blocken

Andererseits ist es so, daß wir Blöcke schreiben wollen, d.h., wir wollen die Karten nicht einzeln schreiben, sondern die kurzen Datenbereiche zu größerer Länge zusammenfassen.

Die Idee ist folgende, durch Zusammenfassen der Daten könnte ich vielleicht 3,4 oder vielleicht noch mehr Karten pro Lücke schreiben, siehe Abb. 18.

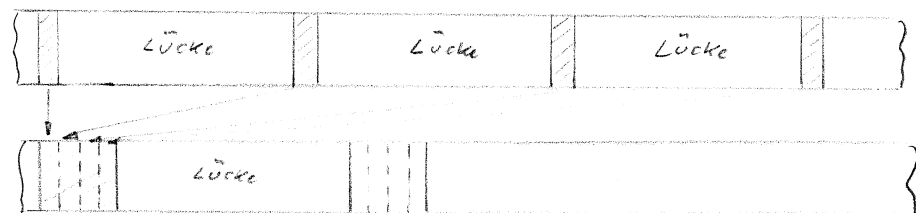


Abb. 18: Zusammenfassung der Daten: Blocken

Die Zahl der Karten, die zusammengefaßt werden, wird als Blockfaktor bezeichnet. Z.B., wenn ich 50 Karten zusammenfasse, habe ich einen Blockfaktor von 50. Das sind dann 4.000 Zeichen pro Block, dies wird nun als Block bezeichnet und dann habe ich eine Kapazität von etwa 20 Millionen Zeichen bei der Schreibdichte 800 BpZ. Sie sehen, das schlechte Verhältnis bessert sich wesentlich. Durch dieses Blocken kann ich erreichen, daß ich mehr Daten auf das Band bekomme. Bei der größeren Dichte, ebenfalls mit einem Blockfaktor 50, bekomme ich etwa 37 Millionen Zeichen Kapazität. Was ich zeigen will, ist, Information wirtschaftlich auf einem Band speichern zu können. Dazu müssen wir die Daten zusammenfassen, die normalerweise auf Karten ankommen, damit es sinnvoll wird. Es wird ja nicht nur die Kapazität des Bandes verschwendet, sondern ebenfalls die Lese- und Schreibzeit. Die Zeit ist als untere Grenze etwa proportional der Länge des Bandes das gelesen wird, meistens noch länger, und zwar deshalb, weil wir das Band beschleunigen und weil es verzögert werden muß.

5.3.3.15 Dichte und Kapazität

Um eine bessere Ausnutzung der Rechnerinstallation zu erreichen, möchte ich ebenfalls die Informationsdichte erhöhen. Zum zweiten hängt, wie erwähnt, die Datenkapazität auf dem Band, d.h., der Gewinn durch die Blockung, von der Schreibdichte ab. Und zwar ist es so, daß ich bei größerer Schreibdichte höhere Blockierungsfaktoren erwarten sollte oder verlangen muß. Sie sehen, der Gewinn bei der größten Schreibdichte im Beispiel, ist wesentlich, prozentual natürlich der gleiche, abder der absolute Wert ist hier viel größer. Bei derselben Datenkapazität auf dem Band, d.h., um denselben Verlust von 3 Millionen zu erreichen, muß ich hier größere Blockfaktoren annehmen, siehe

Abb. 19.

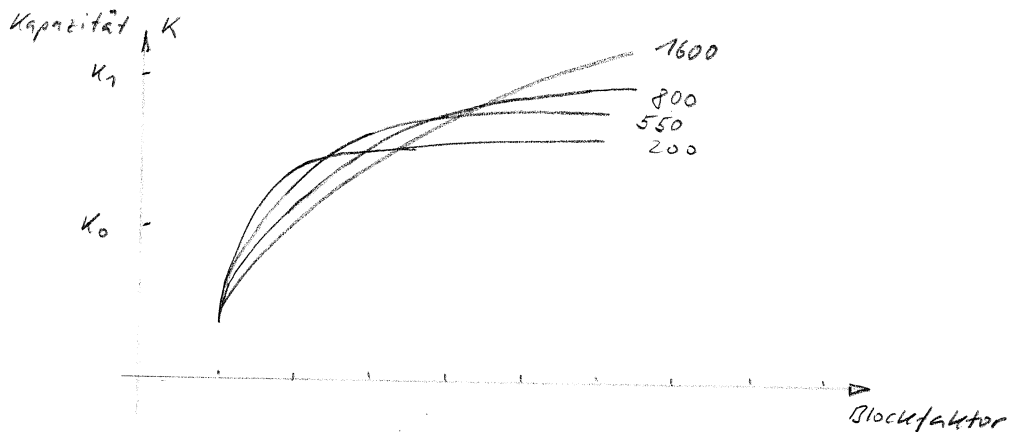


Abb. 19: Bandkapazität und Blockfaktor

Wenn ich eine sehr hohe Dichte, sagen wir 1.600 bits habe, dann steigt die Kapazität etwa in der Funktion der unteren Kurve im Bild 19. Wenn ich eine kleinere Schreibdichte habe, steigt die Kapazität rascher an, bei 800 und bei 556 noch mehr und bei 200 ist es dann der rascheste Anstieg. Die Endkapazität liegt bei größerer Dichte natürlich höher, d.h., die Kurven schneiden sich. Um diesselbe Kapazität, d.h., z.B., wenn ich Kapazität K_1 bekommen möchte, brauche ich bei der Dichte 200 BpZ einen gewissen Blockfaktor, bei der Dichte 556 BpZ muß ich einen höheren Blockfaktor verlangen, bei noch größerer Dichte 800 BpZ brauche ich einen noch höheren, bei 1.600 BpZ brauche ich den größten Blockfaktor. Das ist das Verhältnis, das ich hier zeigen möchte.

$$K = \frac{K' \cdot L}{b_n} \cdot \frac{1}{\frac{B}{F \cdot A} + 1}$$

K' = Schreibdichte [Bit/Zoll]

L = Bandlänge des Bandes [Zoll]

b_n = Sporenanzahl / Zoll

B = Länge der Blockstücke [Zoll]

F = Blockungsfaktor

A = Länge eines Sektors auf Band
(z.B. Lochkarte) [Zoll]

5.3.4 Direktzugriffseinheiten

Bei Einheiten mit Direktzugriff z.B. bei der Trommel, Platte oder bei der Datenzelle, verwenden wir ebenfalls Magnetisierungsmethoden. Direktzugriff heißt, ich kann einen Datenblock sofort lesen oder schreiben ohne vorher andere Daten verarbeiten oder darüber hinweglesen zu müssen, d.h., ein Datenblock ist direkt ohne Verzögerung ansprechbar, ohne Vor- und Rückspulen, ohne anderes zu lesen, was man auch bei Karten machen kann, z.B. 20 Karten durchlesen, und dann wieder die Karte lesen. Das ist nicht nötig bei der Trommel oder Platte. Zu beachten ist, ohne zu große Verzögerung, denn Sie wissen, daß die Platte oder die Trommel umläuft und deshalb haben wir auch die Verzögerung, gemeint ist hier: nicht das Vor- und Rückspulen eines Bandes.

5.3.4.1 Bewegungsunterschiede

Während die Karten oder die Lochstreifen i.a. ruhen (zum schreiben müssen sie ruhen, sonst kann man kein Loch einstanzen, in eine sich bewegende Karte oder einen sich bewegenden Lochstreifen).

Das Magnetband kann dagegen nur bei Bewegung gelesen oder geschrieben werden, d.h., die Lochstreifen und die Karten ruhen im allgemeinen, das Magnetband wird auch meist dasitzen bis Daten gelesen oder geschrieben werden. Dagegen sind Trommel und Platte dauernd im Umlauf.

5.3.4.2 Vorteile

Diese Bewegungsart hat Vorteile und zwar folgende: Dieselben Daten erscheinen bei jedem Umlauf wieder. Bei Trommel, Platte und Datenzelle wird ebenfalls auf magnetisierbare Oberflächen geschrieben.

5.3.4.3 Arten

Es gibt zwei Gruppen, einmal haben wir die Geräte mit schaltbaren Köpfen; falls dies der Fall ist, wissen wir, da müssen mehrere pro Einheit vorhanden sein. D.h., wenn ich elektronisch schalten kann, dann müssen mehrere da sein, denn sonst kann ich nicht alle Datenbahnen bearbeiten. Wenn die Lese- und Schreibköpfe beweglich sind, dann ist es z.B. möglich, daß nur einer existiert.

5.3.4.4 Oberflächen

Die beschreibbaren Oberflächen selber waren auf dem Band 2.400 ft. lang und 1/2" breit. Hier hat die Oberfläche eine andere Form. Bei der Platte haben wir eine runde Oberfläche, die Mitte ist unbenützt, das innere Drittel etwa der Platte wird nicht benützt für die Datenspuren. Im Gegensatz zu den bekannten Schallplatten sind es hier Kreise verschiedener Länge, keine Spiralen, die verwendet werden. Dann haben wir die Trommel, wo die Datenspuren ebenfalls Kreise sind, doch alle von gleicher Länge. Bei der Datenzelle haben wir Magnetkarten, die ausgelesen, ausgesucht, auf eine Trommel gespannt und dann gelesen werden mit gleich langen Datenstrecken.

Die Oberflächen, die wir hier haben, können wir uns alle durch Abbildung als Ebene denken. Die Datenspuren sind bei der Platte und Trommel geschlossene Bahnen, jedoch nicht bei der Datenzelle. Diese Zurückbiegung ist nicht bei der Datenzelle vorhanden.

5.3.4.5 Lesen und Schreiben

Nun, wenn ich nur einen Lese- und Schreibkopf habe,

haben Sie schon gesehen, mußte dieser zwischen den Datenspuren beweglich sein. D.h., wir haben noch eine weitere Folgerung daraus. Wenn wir nur einen Lese- und Schreibkopf haben, dann muß dieser mit bit-serienmäßigem Lesen oder Schreiben arbeiten, d.h., er muß jeweils ein bit lesen oder schreiben. Nun, das ist wichtig und zwar deshalb, weil wir manchmal einen einzigen Kopf haben, der nicht nur eine Datenbahn bearbeiten kann, sondern mehrere Datenbahnen bearbeiten kann. Dann liest und schreibt er Zeichen, so wie wir es beim Magnetband haben. Es gibt noch folgende Möglichkeit: daß ich nicht nur einen Kopf für mehrere Datenspuren habe, sondern mehrere Köpfe für mehrere Datenspuren, dann kann ich nicht nur Zeichen seriell schreiben, sondern sogar parallel lesen oder Schreiben.

5.3.4.6 Registrierflächen

Außerdem müssen wir die Flächenzahl betrachten. Wir haben bei der Trommel eine Fläche, bei der Platte haben wir mehrere Flächen, z.B. einen Plattenstapel, auch Plattenturm genannt, und bei den Magnetkarten in der Datenzelle existieren sehr viele Flächen. Entsprechend ist auch die Kapazität dieser Dateneinheiten, bei einer Fläche haben sie natürlich eine relativ kleine Zahl, bei Datenkapazität bei mehreren Flächen eine größere, bei sehr viel haben sie ebenfalls eine sehr große Zahl als Datenkapazität.

5.3.5 Registrierung auf magnetischem Material

Die nächste Frage, die uns beschäftigt, ist die Signalfesthaltung auf magnetisierbarem Material. Dies bezieht sich nun auf die Magnetbänder und die Direktzugriffsspeichereinheiten, d.h., die Lese- und Schreibverfahren.

Zum Lesen und Schreiben auf magnetisierbaren Oberflächen existieren verschiedene Methoden, oder haben sich eingebürgert. Und zwar sind es die folgenden: einmal die RZ-Methode (Return to Zero) zur Kodierung, dann die RB oder Return to Bias-Kodierung, dann haben wir die NRZ- (Non return to Zero) Kodierung und die NRZ1-(Non return to Zero for ones) Kodierung. Es sind vier Verfahren, die sich etwas unterscheiden. Außerdem werden noch die Phasenkodierungs- oder Verschlüsselungsverfahren, das Zweiphasenverfahren und das Manchesterphasen-Verfahren verwendet.

5.3.5.1 RZ-Verfahren

Nun, das RZ-Verfahren geht davon aus, daß das Band als Ausgangszustand ummagnetisiert ist. Wenn dies nicht der Fall ist, wird es durch eine entsprechende Vorrichtung entmagnetsisiert bevor es beschrieben wird. Die nächste Frage ist, was auf Band, Platte oder Trommel geschrieben werden soll. Wir wollen die Bits schreiben, die die Zeichen repräsentieren. Wir haben in Abb. 16 gesehen, daß durch eine Magnetisierung kleine Magnete auf der magnetisierbaren Oberfläche erzeugt werden. D.h., wir wollen Signale auf Band oder auf einer Ebene produzieren für Ein-Bits und für Null-Bits, auf welche Art und Weise dies geschieht, das ist hier unser Problem.

Signalfesthaltung auf Magnetmaterial (Schreib-Lese-Verfahren)

Verschiedene Methoden:

- RZ (Return to Zero) - Verschlüsselung
- RB (Return to Bias) - Verschlüsselung
- NRZ (Non return to zero) - Verschlüsselung
- NRZ I (Non return to zero for One's)-Verschlüsselung

Phase encoding
(Phasenverschlüsselung)

Di-phase encoding: Zweiphasensystemverschlüsselung

Manchester phase encoding: Manchesterphasenverschlüsselung

RZ-Verfahren

nimmt an, daß das Band nicht beschrieben (wird demagnetisiert vor der Schreibung)

Signale auf Band produziert für 1 und 0 Bits.

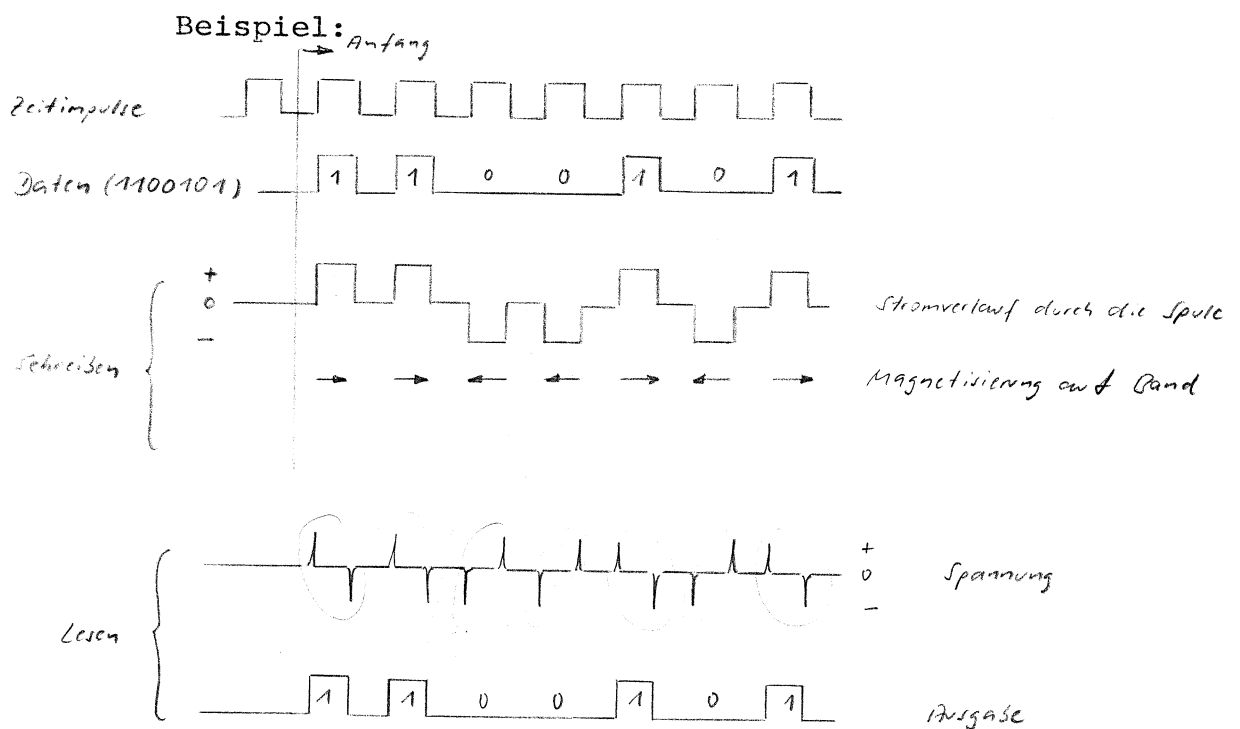


Abb. 20: Schreiben und Lesen mit dem RZ-Verfahren

Beim Lesen entspricht jeweils ein +-Spannungsimpuls, gefolgt von einem negativen, einen 1 -Bit; ein negativer, gefolgt von einem positiven, einem 0-Bit.

RB-Verfahren

ebenso: Annahme, daß Band unmagnetisiert, dann nur 1-Bits geschrieben, d.h., alle Magnetisierungen gehen in dieselbe Richtung:

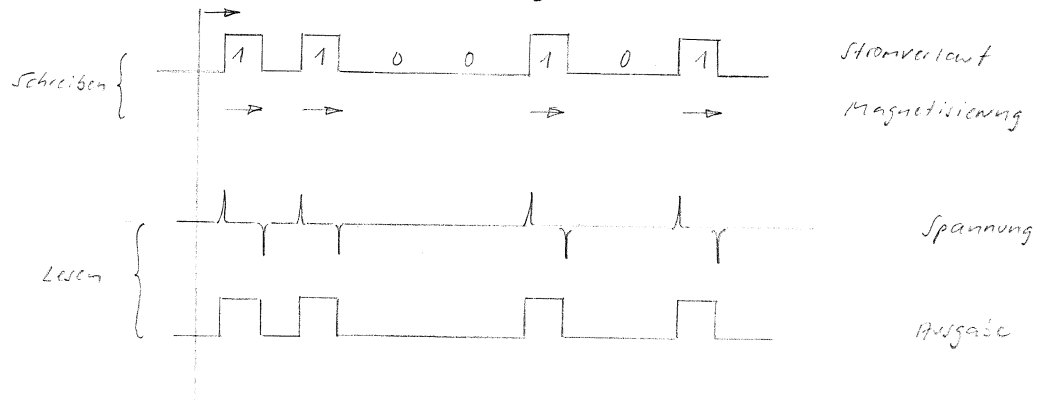


Abb. 21: Schreiben und Lesen mit dem RB-Verfahren

Nehmen wir an, ich habe Zeitimpulse (wir sprechen über das RZ-Verfahren), z.B. in Abb. 20, dann Daten, z.B. vom Speicher kommend ebenfalls als Impulse. Nun nehmen wir an, wir haben dafür folgende Impulsfolge 1100101.

Wir müssen natürlich mit den Daten anfangen (zweite Zeile in Abb. 20), wir haben anfangs einen Datenimpuls, das wäre eine 1, dann ebenfalls zum nächsten Zeitpunkt eine 1, gefolgt von einer Null, sie wird nicht als Impuls erscheinen, ebenfalls wieder eine Null, dann folgt eine 1, zu diesem Zeitpunkt, dann haben wir wieder eine Null, da ist kein Impuls da, dann haben wir wieder eine 1 und dann haben wir eine Null und hier haben wir wieder eine 1. Dies würde die

Impulsfolge für unsere Daten sein, nach rechts laufend.

5.3.5.2.1 Schreiben

Beim Schreiben kann ich nun folgenden Stromverlauf in der Schreibspule erzeugen (dritte Zeile in Abb. 2o): ich habe eine positive und eine negative Richtung im Strom. Solange kein Bit erscheint, habe ich Null, habe ich keinen Strom fließen. Sobald das erste Bit erscheint, wird nun ein Strom in positiver Richtung angeschaltet. Ebenfalls bei dem nächsten Bit und bei dem Nullbit. Hier kann der Strom in die negative Richtung umgepolt werden, ebenfalls bei dem nächsten Nullbit, usw. Der Stromverlauf in der Spule entspricht den Daten und erzeugt auf dem darunterliegenden Band oder auf der Fläche folgende Magnetformen (siehe Zeile vier in Abb. 2o): Ich habe hier einmal einen Magnet nach rechts, und wieder einen so. Anschließend haben wir einen Magnet in der Gegenrichtung usw., genau den Ein- und Nullbits entsprechend. Dies ist sehr einfach und erfüllt genau den Zweck, den wir verlangen, die Magnetisierung auf Band, oder auf der Trommel oder auf der Platte oder generell auf jedem magnetisierbaren Medium. Dies war beim Schreiben.

5.3.5.2.2 Die nächste Frage ist, was passiert beim Lesen. Beim Lesen unserer Daten erhalten wir den Spannungsverlauf von Zeile fünf in Abb. 2o.

Sie müssen die Zeitachse auf allen diesen Zeilen in Abb. 2o nach rechts sich denken oder nach links je nach dem wie Sie es sich ansehen wollen. Beim Lesen erscheint jeweils eine positive Spannung, gefolgt von einer negativen Spannung nach der Zeitverzögerung, der Zeitimpuls entspricht dann einem Einbit. Für ein Nullbit ist die Folge umgekehrt. Wir können daraus die ursprüngliche Datenfolge sofort regenerie-

ren (siehe Zeile sechs in Abb. 20), in der folgenden Weise: für einen positiven gefolgt von einem negativen Puls haben wir ein Einbit, hier dasselbe wieder, in umgekehrter Weise für einen negativen. Das Verfahren heißt Return-to-Zero, also zum Nullpunkt zurückkehrend, der Stromverlauf in der Magnetisierungsspule kehrt zum Nullpunkt zurück.

5.3.5.2 RB-Verfahren

Das RB- (Return to Bias) Verfahren ist ein klein wenig verschieden, nicht allzu sehr, und ich möchte dazu nur den Stromverlauf und den Spannungsverlauf aufzeichnen. Wir nehmen für die Zeit- und Datenimpulse dasselbe Beispiel. Nun, der Stromverlauf ist bei diesem Verfahren folgender: Bei dem RB-Verfahren, beim RZ-Verfahren haben wir einen Strom in positiver Richtung für ein Einbit und in negativer Richtung für ein Nullbit. Davon unterscheidet sich nun das Return-to-Bias-Kodierungsverfahren und zwar dadurch, daß durch einen Strom in positiver Richtung für ein Einbit geschrieben wird und wenn ein Nullbit erscheint, wird nicht geschrieben. Das Nullbit wird ignoriert. Das heißt, alle Magnetisierungen gehen in eine Richtung, während vorher ein Wechsel dieser Magnete erfolgte. Den Stromverlauf in der Spule ist in Abb. 21 dargestellt. Die Stromspannungsimpulse erscheinen dann, wenn ein Einbit erscheint. Daraus kann man die Einbit generieren; die Nullbits müssen aus der Zeitspur erzeugt werden, d.h., man braucht die Zeitspur, um die Nullbits generieren zu können. Falls keine Zeitspur existiert, es könnte ja sein, daß sich E/A verlangsamt oder verschnellert hat, dann würde man bei Verlangsamung mehr Nullbits finden als tatsächlich da sein müßten; oder falls es sich beschleunigt hat, findet man mehr Bits als ursprünglich vorhanden sind. Dies ist ein Problem, das bei dem RB-Verfahren auftritt. Es ist eben

dadurch nötig, daß eine genaue Zeitfolge vorliegt, d.h., man kann oder muß die Zeitfolge von Zeit zu Zeit anpassen, falls sie sich etwas verschiebt. Dies geschieht mit Hilfe der Folge der Spannungsimpulse, die durch die Magnete erscheint. Man kann abtasten, wie der Zeitverlauf der Spannungsimpulse ist, die Frequenz und dann entsprechend einen Zeitgeber etwas verstellen. Das Problem tritt deshalb auf, weil die Daten ja nicht gelesen werden, wann sie geschrieben werden, sondern sie können nicht nur zu einem anderen Zeitpunkt an derselben Anlage geschrieben und gelesen werden, sondern auch an einer ganz verschiedenen Anlage zu verschiedenem Zeitpunkt. Daher wird das Problem dieser Zeitschwierigkeit, Synchronisation, wichtig. Dies ist ein Nachteil des RB-Verfahrens, falls Einzelbits geschrieben werden, bei bitserieller Schreibung oder Lesung.

Beim bitparallelem Schreiben, d.h., zeichenseriell oder zeichenparallel Schreiben, wo die Bits sowieso parallel erscheinen, muß nicht unbedingt eine Schwierigkeit auftreten und zwar ist es so, daß bei ungerader Parität auf jeden Fall ein Bit erscheint. Nehmen wir an, wir haben ein Zeichen mit dem überhaupt keine Bits geschrieben werden, dann habe ich aber auf jeden Fall das Paritätsbit vorhanden, wenn ich Bit-parallel schreibe. Daraus kann die Synchronisation rekonstruiert oder wiederhergestellt werden. Der Nachteil des RB-Verfahrens ist, daß so viele Magnete auf dem Band erscheinen. Die vielen Magnete auf dem Band interferieren natürlich, besonders bei großer Schreibdichte und wenn sie längere Zeit gelagert werden. Bei weniger ergibt es weniger Schwierigkeiten.

5.3.5.3 Nachteile

Die Nachteile beider Verfahren sind folgende:

Für jedes Bit werden 2 Impulse erzeugt, an sich zur Kontrolle sehr gut so, aber andererseits z.B. beim RZ haben wir zwei Pulse für Einer- und Nullbit, beim RB-Verfahren haben wir für die Einerbits zwei Impulse. Also, Sie sehen, es ist besser bei dem RB-Verfahren, da habe ich nur 2 Impulse für die Einerbits, aber wenn ich eine Pulsfolge von vielen einzelnen Bits habe, dann muß ich eben auch eine Riesenzahl von Impulsen erzeugen und dieses Problem der dichten Pulsfolge, und demzufolge Magnete in der Oberfläche, verhindert eine starke Schreib- und Packungsdichte. Z.B., wenn Sie ein Band haben, das z.B. mit 200 BpZ geschrieben ist, dann können Sie annehmen, es ist wahrscheinlich im RZ- oder RB-Verfahren. Wenn Sie ein Band haben, das mit einer Schreibdichte 1.600 Bit per Zoll geschrieben ist, wissen Sie, daß das nicht das RZ- und nicht im RB-Verfahren registriert worden sein kann. Größere Schreibdichte beinhaltet eine schnellere Zeitfolge, d.h., die Impulse rücken zusammen, die Daten rücken zusammen, und der Stromverlauf rückt zusammen und die Magnete rücken zusammen und überlappen sich dann. Dann wird es schwierig, diese dann wieder beim Lesen voneinander zu trennen. Wegen dieser Nachteile gibt es andere Verfahren, und zwar die NRZ und NRZ1- sowie die Phasenkodierungsverfahren.

Nachteil der RB gegenüber RZ: Zeitimpulse (für 0-Bit) fehlen in der Ausgabe zur Kontrolle falls Einzelbits geschrieben werden.

Anwendung: Beim Parallel-Schreiben von Zeichen mit ungerader Parität (dann erscheint auf jeden Fall in jeder Zeichenposition ein Bit).

Nachteil beider Verfahren: für jedes Bit werden 2 Impulse (bei RZ für 1- und 0-Bits, bei RB für 1-Bits) Problem zu dichter Pulsfolge: verhindert starke Packungsdichte wegen möglicher Überlappungen, deshalb andere Verfahren

5.3.5.4 NRZ-Verfahren

Wie der Name sagt, geht der Strom nicht mehr auf Null zurück, nach jedem oder auf jedes Bit, sondern wir haben folgendes: Wir haben einen Wechsel der Stromrichtung nur beim Wechsel in der Signalfolge. Nehmen wir einmal die schlimmste Signalfolge an. In diesem Fall wäre es eine 10101 usw.-Folge. Hier haben wir nur einen Impuls pro Bit, wogegen bei den anderen Verfahren, bei RZ-Verfahren wäre es am schlimmsten, da würden auf jeden Fall 2 Impulse pro Bit auftreten. In dem RB-Verfahren habe ich für gewisse Zeichenfolgen (große Folgen von Einsen), 2 Impulse pro Bit fortwährend zu schreiben.

Der Stromverlauf im Bild ändert sich hier und nachher auch die Magnetisierung wie in Abb. 22 dargestellt.

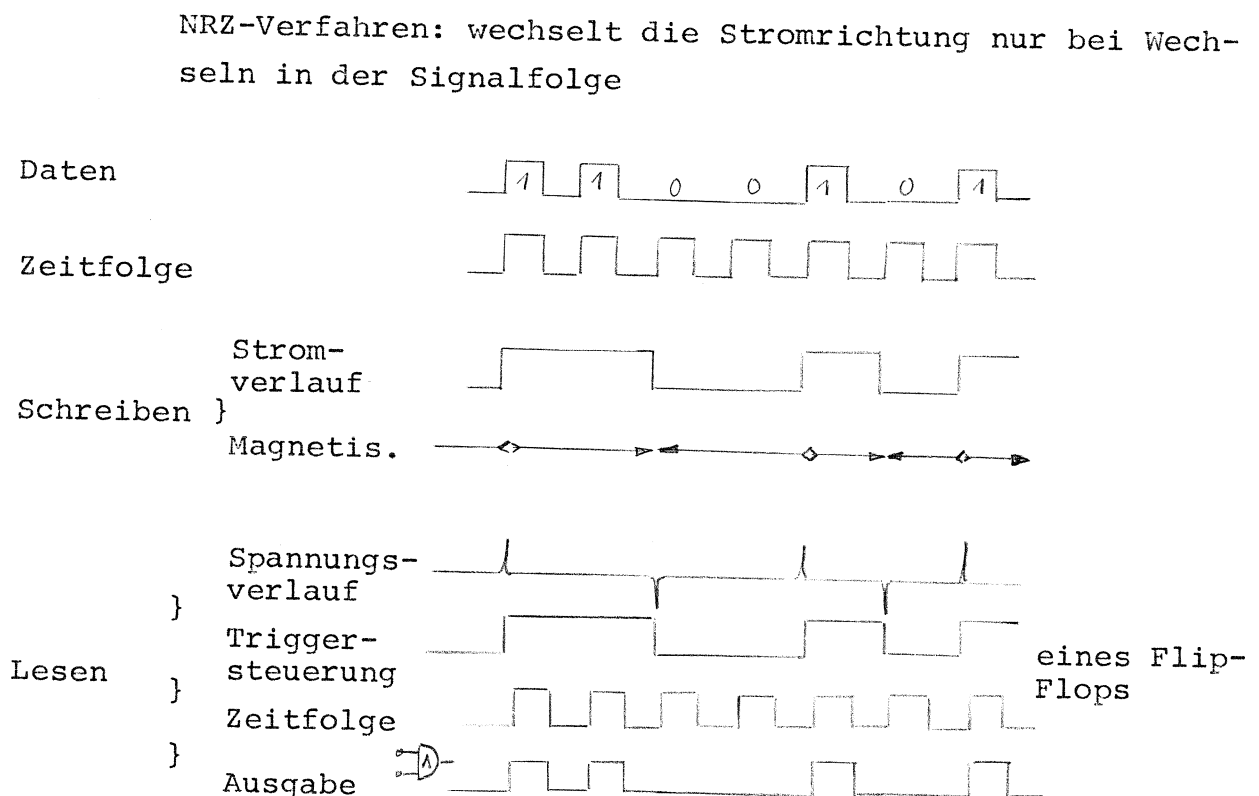


Abb. 22: Pulsfolgen beim NRZ-Verfahren

Nachteil: ebenfalls keine Selbstzeitgebung (wie bei RB, geeignet für ungerade Parität)

NRZ1-Verfahren: Modifikation des NRZ-Verfahrens:
jedes 1-Bit wechselt die Polarität
jedes 0-Bit wechselt die Polarität nicht.

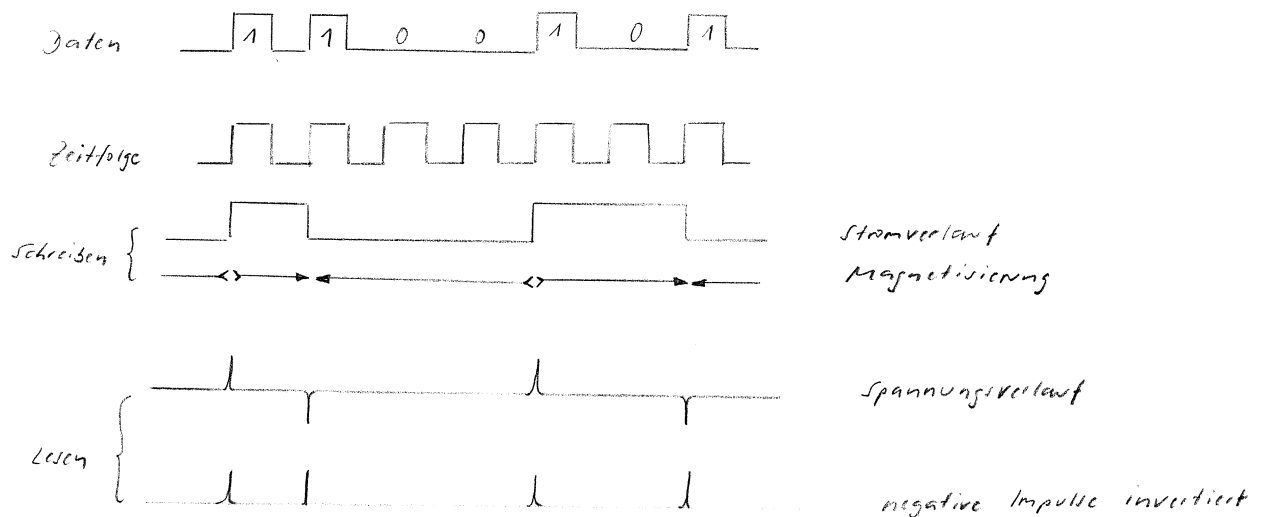


Abb. 23: Pulsfolgen beim NRZ1-Verfahren

Solange keine Bits erscheinen, haben wir einen Nullstrom. Sobald das erste Einerbit in den Daten erscheint, schalten wir den Strom ein. Dann erscheint wieder ein Einerbit, d.h., der Strom wird beibehalten bis das Nullbit erscheint. Dann haben wir einen Wechsel in der Datenfolge, d.h., ich muß nun den Strom wechseln, abschalten oder kann ihn umpolen, was generell hier ziemlich ähnlich ist. Hier habe ich dann Null für den Schreibstrom bis sich die Datenfolge ändert, usw. So entsteht eine wesentlich geringere Zahl von Wechseln in dem Stromverlauf als ich vorher hatte, und genauso erfolgt dann ein geringerer Wechsel in den Magneten, die auf der magnetisierten Oberfläche erzeugt werden. Mit dem Spannungsverlauf aus

den Impulsen allein kann man die ursprünglichen Daten nicht regenerieren. Dazu braucht man etwas mehr. Ausgehend von der Annahme, daß ein Wechsel in der Spannung ein Flip-Flop mit einer Triggersteuerung umpolt und aus der Spannungsfolge selbst kann man die Daten wieder erzeugen. Mit den Zeitimpulsen kann man natürlich auch die Nullbits leicht einsetzen. Der Nachteil ist, daß ebenfalls keine Selbst-Zeitgebung vorliegt. Falls Zeittriggerung gesteuert werden soll, muß ungerade Parität gewählt werden, beim Bit-parallel-schreiben mit dieser Methode. Andernfalls können Schwierigkeiten auftreten beim Bitschreiben. Z.B., wenn ich auf Platte in einem fort mit dieser Methode schreibe, und ich habe irgendwelche Unregelmäßigkeiten im Bewegungsverlauf, dann kann es sein, ich verliere Bits oder gebe Bits dazu, was dann durch die Parität geprüft wird, aber trotzdem Schwierigkeiten erzeugen kann. Wenn es zu viele sind, hilft die Parität leider auch nicht mehr viel, sie kann feststellen, daß etwas fehlt, doch wenn viel fehlt, kann man es nicht mehr rekonstruieren. D.h., das Verfahren ist geeignet für Bit-paralleles Arbeiten in ungerader Parität.

5.3.5.5 NRZ1-Verfahren

Eine Modifizierung dieses Verfahrens ist das NRZ1-Verfahren (non-return-to-zero-for-one's-Verfahren), eine Modifikation des NRZ-Verfahrens. Und zwar auf die folgende Weise: Jedes Einerbit wechselt die Polarität des Schreibstroms und jedes Nullbit wechselt die Polarität des Schreibstroms nicht. Dies ist eine zusätzliche Einschränkung, siehe Abb. 23.

Ich muß den Stromverlauf etwas ändern, und natürlich werden die Magnete geändert, und der Spannungsverlauf ebenfalls, daß nur jedes Einerbit der Daten die Polarität wechselt. Die Magnete sehen, entsprechend dem

Schreibstrom, folgendermaßen aus: Wir haben hier den Magneten, der nach links herausläuft, dann fängt einer nach rechts an, er wird etwas kleiner als vorher, dafür verlängert sich der zweite Magnet, usw. Der Spannungsverlauf im Lesekopf erscheint in Zeile 5 von Abb. 23: erst mit einem positiven Impuls und dann mit einem negativen Impuls, usw.

Die Pulse entsprechen genau den Einerbits, nur müssen wir sie leider z.T. invertieren, wenn wir die Ausgabe erzeugen wollen. Dann müssen wir wieder mit der Zeitfolge aussteuern und erhalten dann die Ausgabe, wo jeweils das eine Bit erzeugt wird, wird der Puls ausgegeben. Die Nullbits erscheinen ebenfalls nicht in der Ausgabe. Wir haben dieselben Nachteile des NRZ1-Verfahrens, wie bei dem NRZ-Verfahren.

5.3.5.6 Probleme bei den Verfahren

Wie angedeutet, existieren Schwierigkeiten damit. Deshalb braucht man die Phasen-Kodierungsverfahren. Das sind die Methoden, die bei den modernen Geräten verwendet werden. Zusammenfassend, die Probleme bei den bisherigen Verfahren sind, daß die Folgen der Impulse zu lang werden, d.h., die Zeit dazwischen wird zu lang. Aber im ungünstigsten Fall, ist die Folge auch sehr kurz. Wenn ich Verstärker bauen muß, und die sind nötig, so müssen diese eine sehr große Bandbreite haben, was technische Schwierigkeiten bereitet.

Der zweite Nachteil war, daß beim Bit-Schreiben und -lesen kein Zeitverhalten festgestellt werden kann. Beim Zeichenschreiben und -lesen kann das Zeitverhalten aus den Bahnen, vor allem bei ungerader Parität, abgelesen werden, weil da immer ein Bit existieren muß wegen der Paritätsforderung. Bitlesen kommt sehr häufig bei Trommeln und Plattenspeichern vor.

Neben der Schwierigkeit mit der Bandbreite für den Verstärker haben wir noch die Schwierigkeit mit dem Zeitverhalten. Diese Frage ist sehr wichtig und zwar deshalb, weil wir unterscheiden müssen, was ein Einbit ist und wir können die Bits nicht erzeugen, wenn wir das Zeitverhalten nicht kennen. Aus diesen Gründen wurden die Phasenverschlüsselungsverfahren entwickelt.

5.3.5.7 Phasenverschlüsselungsverfahren

Es sind eine sehr große Anzahl von Phasenverschlüsselungsverfahren, zwei davon sind die wichtigsten. Die Charakteristik der Phasenverschlüsselungsverfahren ist die, daß die Zahl der Wechsel in der Polarisation des Schreibstroms von der Art der Bits abhängt. Dies trat auch bei den früheren Verfahren auf, wir arbeiten nicht nur allein mit den Bits, sondern auch mit der Zeitfolge. Bei den anderen Verfahren war der Schreibstrom, einzig und allein bestimmt durch die Art des Bits, das geschrieben werden muß oder gelesen wird. Hier kommt noch eine Zeitfolgefunktion dazu. Die Polarität des Schreibstroms ist hier eine Funktion, einmal von der Art des Bits und dann von der Zeitfolge, z.B. ein Wechsel für ein Nullbit, zwei Wechsel für ein Einbit und zwar jeweils dann mit der Zeitfolge. Dies Beispiel wird in Wirklichkeit nicht benutzt und zwar deshalb, weil es genauso wie die Anfangsverfahren an der großen Zahl der Magnete und der großen Zahl der Wechsel krankt und eine größere Schreibdichte verhindern würde. Immerhin, der Vorteil hier wäre, daß nur 2 Längen von magnetisierten Gebilden auf der magnetisierbaren Oberfläche vorkommen können (einmal von der Länge einer halben Signalzeit und von der Länge einer vollen Signalzeit). Die einzigen Frequenzen die so vorkommen könnten, sind halbe Signalzeiten.

Für diesen Fall ist es sehr einfach, Verstärker zu bauen. Aber es sind noch Umschaltungen des Schreibstroms da und damit eine zu dichte Belegung der magnetisierten Oberfläche mit Magneten, die sich gegenseitig beeinflussen, und damit die Schreibdichte erheblich herabsetzen.

5.3.5.7.1 Zwei-Phasenverfahren

Das Zweiphasenverfahren ist ein gebräuchliches Verfahren und zwar wird es von IBM bei der Trommel, bei der Platte und bei der Datenzelle verwendet, also wirklich aktuell. Es wird auch zuweilen als Doppelfrequenz-NRZ1-Verfahren bezeichnet. Es unterscheidet sich von den anderen Verfahren dadurch, daß der Schreibstrom die Polarität am Anfang eines jeden Zeitsignals wechselt, unabhängig von der augenblicklichen Richtung. Bei dem anderen Verfahren, das ebenso wichtig ist, ist dies nicht der Fall. Dort ändert der Schreibstrom die Richtung in der Mitte jedes Zeitsignals für ein Einerbit, siehe Abb. 25. Jeder Puls im Spannungsverlauf entspricht einem Einerbit, deshalb: nur Invertierung der negativen Pulse nötig (und Verzögerung) für die Ausgabe

Probleme:

1) Schwierigkeiten existieren mit NRZ und NRZ1-Verfahren für lange Folgen von 1 oder 0. Erzeugt Schreibstrom ohne Richtungswechsel (erzeugt lange Felder der Magnetisierung: lange Magnete). Technisches Problem: Verstärker mit gutem Niederfrequenz-Ansprechen nötig, aber auch gutes Hochfrequenz-Ansprechen für 10101 oder 1111-Folgen.

2. Beim Zeichenschreiben und -lesen kann Zeitverhalten bestimmt werden (von den anderen Bahnen), dagegen nicht beim Bit-Arbeiten (üblich bei Trommel und

Platte).

Deshalb: Phasenverschlüsselung

Zahl der Wechsel in der Polarität des Schreibstroms hängt von der Art des Bits ab, z.B. 1 Wechsel für 0-Bits, 2 Wechsel für 1-Bit; damit kommen dann nur 2 Längen der magnetisierten Strecke vor: 1/2 Signal-Zeit, volle Signal-Zeit.

Zwei-Phasen-System (IBM bei Trommel, Platte, Datenzelle), auch als Doppelfrequenz NRZ1 bezeichnet. Schreibstrom wechselt Richtung am Anfang jedes Zeitsignals (unabhängig von der augenblicklichen Richtung) Schreibstrom wechselt Richtung in der Mitte jedes Zeitsignals nur für Einerbit.

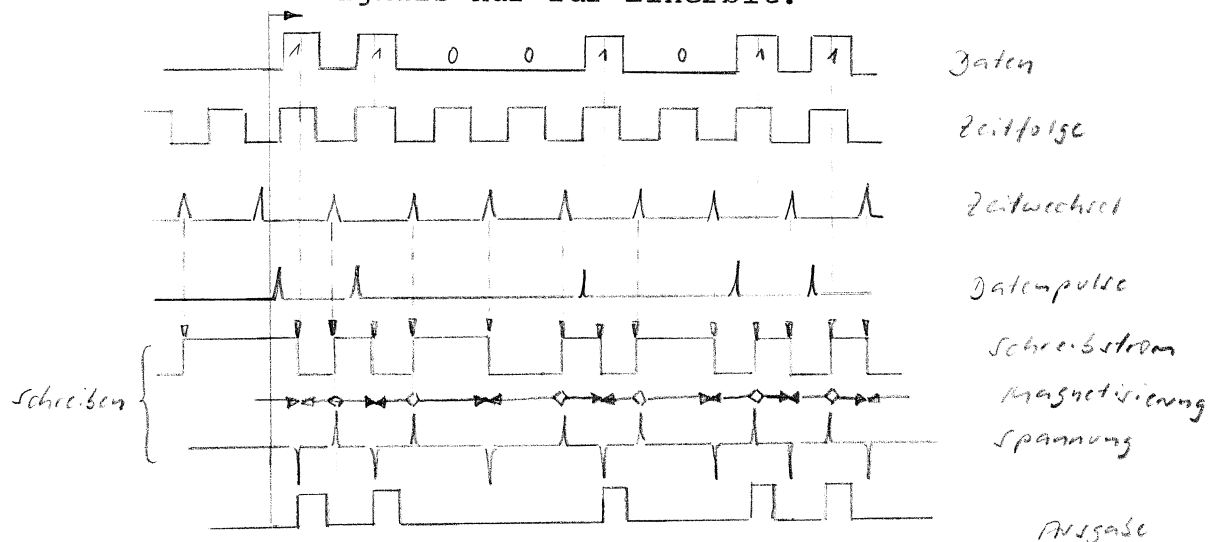


Abb. 25: Pulsfolge bei Zwei-Phasenverfahren

Wir haben neben der Datenfolge noch den Zeitwechsel und einen Frequenzerzeuger, in der Mitte jedes Zeitsignals. Dies läßt sich relativ leicht erzeugen. Beim Anschalten der Zeitsignale wird mit einer kleinen Verzögerung ein Impuls erzeugt. Aus diesen Impulsen des Zeitwechsels und den Datenpulsen kann nun der Schreibstrom angegeben werden. Der Nullpunkt ist willkürlich gewählt, aber es könnte auch etwas

höher liegen, ein Bias kann durchaus drin sein, wodurch sich der Schreibstrom im Niveau verschieben würde. Doch ist dies nicht wesentlich. Wir haben dieselbe Erscheinung, daß eine gewisse Anzahl Frequenzen hervorgezogen werden, die für die Verstärkung der induzierten Spannungen sehr günstig liegen. Dies ist beim Schreiben; beim Lesen erhält man Spannungsimpulse folgender Art, wie in Abb. 25 angegeben. Daraus muß nun die Ausgabe synthetisiert werden, und zwar aus der Zeitfolge und dem Spannungspuls. Wenn ein Zeitfolgepuls und ein Spannungspuls auftritt, dann wird der Puls ausgewählt und erscheint wie die letzte Zeile in Abb. 25. Es existiert eine Phasenverschiebung um eine Zeiteinheit, doch die Pulse sind erzeugt und natürlich kann ich die Bits nun durch Aussteuern einer Rechteckfrequenz in der ursprünglichen Art erzeugen. Dies ist das Zweiphasenverfahren in der Verschlüsselung bei dem Schreiben auf magnetsierter Oberfläche.

5.3.5.7.2 Manchester-Verfahren

Das Manchesterverfahren wird ebenfalls von IBM verwendet und zwar bei den Magnetbandeinheiten. Die Methode ist hier, daß der Schreibstrom seine Richtung immer zwischen den Zeitpulsen ändert. Dazu kommt noch die Polung des Schreibstroms. Sie hängt davon ab, ob ein Einer- oder Nullbit als Datenpuls vorliegt. Ein Problem existiert, wenn ich unter Umständen gerade angesteuert habe und wenn ein Einerbit vorliegt, dann kann ich nicht mehr weiter aussteuern, d.h., es muß eine Zusatzbestimmung eingeführt werden. Dies ist folgende: Die Richtung des Schreibstroms wechselt zu Beginn des Zeitsignals, wenn benötigt, um die richtige Richtung für die Zeit zwischen den Zeitpulsen zu erhalten, siehe Abb. 26. Manchester-Phasen-Kodierung (in Magnetbandeinheiten der IBM); Methode: Schreibstrom ändert Richtung immer zwischen den Zeitpulsen; Polung des Schreibstroms hängt davon ab, ob eine 1 oder 0 als Datenpuls vorliegt; Richtung des Schreibstroms wechselt zu Beginn des Zeitsignals, wenn nötig, um die richtige Richtung für Zeit zwischen den Zeitpulsen zu bekommen.

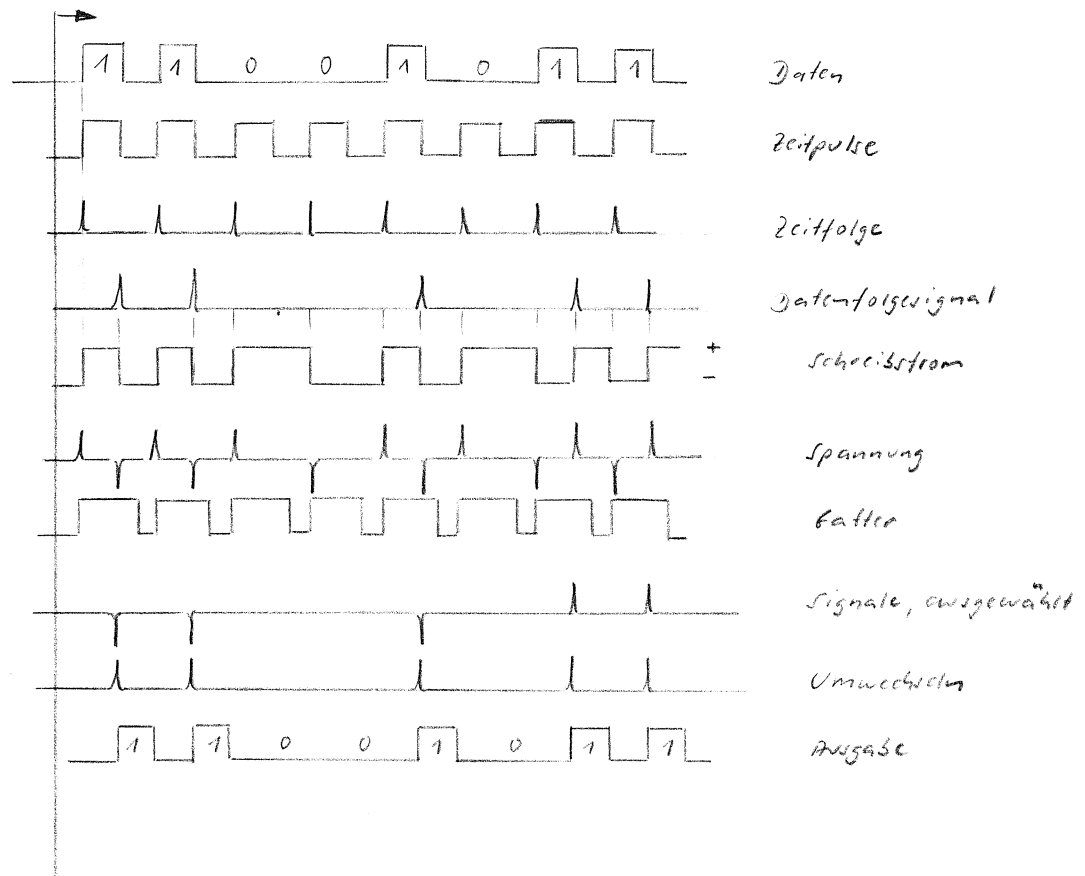


Abb. 26: Pulsverlauf beim Manchester-Verfahren

Hier muß eine scharfe Synchronisation zwischen dem Lesegerät und dem Schreibgerät gefordert werden. Dabei kommt es sehr häufig vor, daß irgendeine Datei auf einem Gerät geschrieben und auf dem anderen Gerät gelesen wird, und wenn die Synchronisation nicht stimmt, dann treten Fehler auf. Auch hier treten nunwiederum zwei spezielle Frequenzen auf, die im Schreibstrom und natürlich dann in der Magnetisierung auf der Oberfläche und in der Folge der induzierten Spannungspulse wieder auftreten. Die Frequenzen sind die volle und die halbe Pulsfrequenz. Dazu muß noch eine Gatterfolge genommen werden, und zwar wie in Zeile 8 in Abb. 26.

Die Gatterschaltung geht mit jedem Zeitpuls herauf und fällt kurz vor dem nächsten zurück. Mit dieser Gatterfolge werden die Spannungspulse ausgewählt und zwar mit einer einfachen Und-Schaltung. So trat im Beispiel erst einmal ein negativer Puls auf, dann wieder ein negativer Puls, der nächste negative Puls fällt aus, usw. Ein Teil der Spannungspulse wird durch die Gatterschaltung ausgewählt und zwar braucht man die Spannungspulse, die in der Mitte der Zeitwechsel erscheinen. Wenn nötig müssen negative Pulse invertiert werden. Die erhaltenen Pulse steuern eine Rechteckschaltung aus für die Ausgabe. Wiederum erfolgt natürlich eine Verzögerung um einen halben Zeitpuls eigentlich zwischen den Daten, die geschrieben werden und den Daten, die gelesen werden. Da aber die Schreibung und Lesung von einander verschieden sind, ist das unwesentlich.

6. E/A-Geräte

Bei den E/A-Geräten gibt es zwei Klassen, die Einweg-Geräte und die Doppelweg-Geräte. Unter die Einweg-Geräte zählen die Karteneinheit, die Lochstreifeneinheit, Schnelldrucker, diese alle sind seriell vorwärts (man kann nicht mit dem Kartenstanzer 10 Karten zurückschalten, und eine Karte stanzen, dann wieder zurückspringen). Bei den Doppelweggeräten existieren Magnetband, Trommel, Platte, Datenzelle und Großkernspeicher: seriell vor- und rückwärts arbeitend.

Dann gibt es zwei Gruppen, die sich mit den genannten Klassen überschneiden, mit serielltem Zugriff mit mit direktem Zugriff. Zum seriellen Zugriff gehören alle die oben stehenden Ein- und Doppelweg-Geräte und beim direkten Zugriff haben wir die Platte, die Trommel, die Datenzelle und den Großkernspeicher.

Hinzu kommt noch eine neue Einteilung, die Wirkungsweise mit Direktgeräten, an den Rechner angeschlossen und indirekten Geräten, die am Rechner nicht angeschlossen sind. Die Einteilung in die Gruppe betrifft den Zugriff, in die Klasse betrifft die Richtung, in die Kategorie betrifft die Wirkungsweise.

6.1 Lese-Schreibverfahren

Wir können die verschiedenen Lese- und Schreibverfahren unterscheiden, je nach Art der Registrierung.

Tabelle 6.1: Lese- und Schreibverfahren

Art	Aufgabe	Vorgang
mechanisch	<u>Lesen von Lochkarten</u>	Fühlstifte
	<u>Stanzen von Lochkarten</u>	Stanzstempel
	<u>Lesen vom Lochstreifen</u>	wie oben
	<u>Stanzen vom Lochstreifen</u>	
	<u>Schreiben des Schnelldruckers</u>	Kette oder Walze
elektrisch	<u>Lesen von Lochkarten</u> <u>Lesen von Lochstreifen</u>	Öffnen und Schließen von Kontakten durch Bürsten
elektromagnetisch	<u>Lesen und Schreiben von Magnetband, Trommel, Platte, Datenzelle, Magnetkarte</u>	Elektromagnetische Induktion
optisch	<u>Lesen von Lochkarten</u>	Photozellen
	<u>Lesen von Lochstreifen</u>	
	<u>Lesen und Schreiben von Laserbändern</u>	Laseroptik
elektronisch	<u>Lesen von Lochkarten und Lochstreifen</u>	Änderung eines Schwingkreises

In dieser Tabelle sind die Standardverfahren unterstrichen. Wir haben mechanische Lese- und Schreibverfahren, dann elektrische, elektromagnetische, optische und elektronische.

6.1.1 Mechanische Verfahren

Mechanische Leseverfahren benutzt man zum Lesen von Lochkarten durch Fühlstifte. Falls ein Loch vorhanden ist, fällt ein Fühlstift durch und schließt einen Kontakt. Dies ist veraltet, doch das Schreiben der Lochkarten erfolgt noch mit Stanzstempeln. Lochkarten kann man bisher noch nicht anders stanzen, vielleicht in Zukunft mit Laserstrahlen. Lesen und Schreiben von Lochstreifen geschieht ebenso wie bei den Lochkarten. Es gibt noch einzelne Lochstreifenleser, die mit Fühlstiften arbeiten. Außerdem ist das Schreiben des Schnelldruckers, z.Zt. ebenfalls noch als Standardverfahren, ein mechanisches Verfahren. Es gibt andere Verfahren für das Drucken, z.Zt. in der Entwicklung, die wahrscheinlich in ein paar Jahren marktreif sein werden und dann möglicherweise das mechanische Verfahren verdrängen werden.

6.1.2 Elektrische Verfahren

Elektrische Verfahren haben wir beim Lesen von Lochkarten durch Öffnen und Schließen von Kontaktbürsten, und sie existieren auch noch für Lochstreifen.

6.1.3 Elektro-magnetische Verfahren

Dann haben wir die elektro-magnetischen Verfahren beim Lesen und Schreiben. Das sind Standardverfahren für Magnetband, Trommel, Platte und Datenzelle durch elektro-magnetische Induktionen.

6.1.4 Optische Verfahren

Als optische Verfahren verwenden wir das Lesen von Lochkarten und das Lesen von Lochstreifen. Lochstreifen werden im wesentlichen schon seit einiger Zeit optisch gelesen, mit Photozellen oder Fotodioden

6.1.5 Elektronische Verfahren

Die elektrischen Verfahren arbeiten durch Änderung von Schwingkreisen zum Lesen von Lochkarten und Lochstreifen. Durchgesetzt haben sich diese Verfahren bisher nicht, vielleicht in der Zukunft. Lesen von Lochstreifen optisch ist Standard und auch das Lesen von Lochkarten auf diese Weise hat sich in den letzten Jahren eingebürgert. Vor allem bei Hochleistungskartenlesern wird optisch gelesen, weil es mechanisch zu langsam ist.

6.2 Vor- und Nachteile der Verfahren

Art	Vorteile	Nachteile	Geschwindigkeit
mechanisch	relativ unempfindlich gegen Störungen	langsam, große Abnutzung der Medien, intermittierender Vorschub	bis 20 zeichen/sec.
elektrisch	einfacher Aufbau, wenig empfindlich	etwas Abnutzung	bis 200
elektronisch	keine Beeinflussung der Medien relativ schnell	schwierige Schaltungen gleichmäßige Medienqualität	bis 2.000 etwa Grenze des Papier-vorschubs
optisch	siehe elektronisch	staubempfindlich	bis 2.000 wie elektr.
elektromagnetisch	schnell	evtl. Signalverlust und Fremdeinflüsse	bis 400.000

Tabelle 6.2: Vor- und Nachteile der Verfahren

Dies ist ebenfalls eine Tabelle. Bei den verschiedenen Arten der Methoden haben wir Vorteile, Nachteile und dann noch die erzielbaren Geschwindigkeiten angegeben. Die Vorteile der mechanischen Verfahren sind, daß sie relativ unempfindlich gegen Störungen sind. Der Nachteil dabei ist, daß wir meist intermittierenden Vorschub haben. Und damit eine große Abnutzung der Datenträger, Medien, worin die Daten eingetragen sind. Geschwindigkeiten liegen bis etwa 20 Zeichen pro Sekunde. Das ist die obere Grenze mit den mechanischen Methoden. Die elektrischen Methoden haben den Vorteil einfachen Aufbaus gegenüber den mechanischen und sind dabei selbst wenig empfindlich. Etwas empfindlicher als die mechanischen Verfahren, aber doch noch nicht so wie bei denen der nächsten Zeile. Wir haben hier etwas Abnutzung, nicht ganz so viel wie bei den mechanischen Verfahren, doch merklich. Z.B. Lochstreifen, die relativ schnell gelesen werden, zeigen nach einiger Zeit eine raue Oberfläche und damit Abnutzung. Sobald die Oberfläche rauh ist, dann geht die Abnutzung, auch Ausreißen der Löcher, sehr rasch von statten. Hier ist die Geschwindigkeit bis etwa 200 Zeichen/sec. also 1 Größenordnung schneller als die mechanischen Verfahren.

Bei den optischen Verfahren haben wir als Vorteil keine direkte Beeinflussung der Datenträger. Und zwar deshalb, weil die Lese- und Abgriffmethode hier überhaupt nicht mehr die Oberfläche berührt. Allerdings ist der Nachteil, daß es staubempfindlich ist. Staub ist ein großer Gegner der optischen Methode. Hier geht die Geschwindigkeit bis etwa 2.000 Zeichen pro Sekunde, wiederum eine Größenordnung mehr. Die elektronischen Verfahren haben den Vorteil, daß sie unempfindlich gegen Staub sind. Allerdings ist der Nachteil hier, daß hier komplexe Schaltungen nötig sind und auch eine gleichmäßige Medienqualität verwendet werden muß. Hier ebenfalls existiert eine

obere Grenze von etwa 2.000 Zeichen pro Sekunde. Nun, die Frage ist, warum bei einer Methode, die völlig von der vorhergehenden verschieden ist, dieselbe Obergrenze auftritt. Das rührt daher, daß der Vorschub bei beiden mechanisch erfolgt und dies etwa die Grenze des mechanisch möglichen Direktvorschubs für das Lesen von Daten ist.

Bei den elektromagnetischen Verfahren ist der Vorteil, daß sie sehr schnell sind, und zwar bedeutend schneller als die anderen. Die Nachteile sind, daß ein eventueller Signalverlust existiert und eventuelle Fremdeinflüsse schaden, und zwar z.B. bei der Parität. Wenn dies nicht der Fall wäre, bräuchte man keine solch großen Aufwendungen für die Paritätsprüfungen zu treffen. Hier haben wir eine Geschwindigkeit von bis zu etwa 400.000 Zeichen pro Sekunde. Diese große Geschwindigkeit rührt einmal daher, daß die Schreibdichten bei dem elektromagnetischen Verfahren, z.B. bei dem Magnetband angewandt werden, wesentlich größer sind als bei den anderen z.B. Karte 80 Zeichen auf ein Stück, das etwa 15 cm lang ist; auf dieser Fläche könnten wir bei der Datenzelle ein paar zigtausend Zeichen speichern, nicht nur 80. Zum anderen durch die größere Vorschubgeschwindigkeit. Das sind die Ursachen, weshalb die elektromagnetischen Verfahren so große Geschwindigkeiten haben.

6.3 Lochstreifen-Geräte

Lochstreifen-Geräte waren schon an verschiedenen Beispielen eingeführt worden, sodaß keine Zeit verloren zu werden braucht, um sie generell zu betrachten.

6.3.1 Charakteristiken

Die Charakteristiken, eine Auswahl für einen Überblick

sind in Tabelle 6.3.1 aufgeführt. Diese Daten sind aus-
 gesucht aus einer Veröffentlichung Computer Review, von
 der GML-Gesellschaft jährlich herausgegeben, früher vier-
 teljährlich. Der Grund ist, daß sich die Marktlage nicht
 mehr so rasch ändert. In diesem Buch findet man die
 Charakteristiken aller derzeit auf dem Markt befindli-
 chen Rechner, der Zentralrechner und der peripheren Ein-
 heiten. Dies ist sehr wichtig, bei der Auswahl der An-
 lage, welche Leistungen zu welchem Preis erhältlich sind.

In Tabelle 6.3.1 sind die Charakteristiken aufgeführt:
 einmal für den Lochstreifenleser, dann für den Lochstrei-
 fenstanzer.

	Leser				Stanzer			
Geschwind. (Zch/sec.)	30	60	1000	<u>1500</u>	15	60	150	<u>240</u>
Hersteller	Tally, IBM, IBM, NCR Digitronics				IBM, Tally, Tally, Univac			
Miete	<u>20</u>	100	<u>900</u> <u>110</u>	700	330	<u>100</u>	120	650

Tabelle 6.3.1: Charakteristiken der Lochstreifen-Geräte

Die Zeilen sind die Geschwindigkeit in Zeichen pro Se-
 kunde, dann der Hersteller, der Preis in DM/Monat Miete,
 umgerechnet zum Kurs von 1 Dollar zu DM 2,75.

6.3.1.1 Lochstreifenleser

Die Geschwindigkeiten für die Leser sind 30, 60, 1000,
 1500 Zeichen/Sec. Das sind die wesentlichen Geschwindig-
 keiten, die existieren und zwar das einfachste Gerät
 von einer Fa. Tally hergestellt, der 60-Zeichenleser
 existiert von IBM, ebenfalls stellt IBM das 1000-Zeichen
 pro Sekunde Gerät. Dann der Leser mit 1500 Zeichen ist
 von NCR. Die Mieten dabei sind: Beim ersten Gerät DM
 20,-- Miete im Monat, die von IBM ist 100,-- im Monat,
 dann von IBM 900,-- im Monat für den 1000 Zeichenleser.
 NCR verlangt etwa DM 750,-- für den 1500 Zeichenleser.

6.3.1.2 Die Lochstreifenstanzer

Die Lochstreifenstanzer sind nicht ganz so schnell, weil diese eben mechanisch gestanzt werden müssen. Die Stanzgeschwindigkeit beträgt einmal 15, dann 60, 150, und dann 240 Zeichen pro Sekunde, das ist die obere Leistungsgrenze. Der 15 Zeichenstanzer wird von IBM hergestellt, der 60 Zeichenstanzer von Tally, der 150 Kartenlesestanzer ebenfalls von Tally, der 240 von Univac und die Kosten sind DM 330,--, DM 100,--, DM 120,-- und DM 650,-- für die Univac-Stanzer. Die Maxima der erhältlichen Leistungen sind unterstrichen. 1500 Zeichen/sec. ist die maximale Geschwindigkeit für einen Lochstreifenleser; ebenfalls der Stanzer mit 240 Zeichen/sec. ist das Maximum, das am Markt existiert.

6.3.1.3 Unabhängige Hersteller

Eine besondere Erscheinung tritt auf, daß wenn man Einheiten für ein System anschaffen will, dann sind die unabhängigen Hersteller wie Tally, usw. wesentlich billiger als die Rechnerhersteller z.B. IBM, NCR usw. Z.B. verlangt IBM 900,-- für den 1000,-- Zeichen/Sec. Lochstreifenleser, wohingegen Digitronics für dieselbe Leistung um nur DM 110,-- vermietet. Genau dasselbe ist bei Tally mit den Lochstreifenstanzern, verglichen mit dem IBM-Gerät. Dies ist bei allen Systemkomponenten der Fall, nicht nur bei peripheren Geräten, sondern auch bei Zusatzkernspeichern. Ein möglicher Grund mag in der Qualität der Geräte liegen, so dürften die billigen keine Hochleistungsgeräte sein, d.h., diese können nicht dauernd z.B. 1000 Zeichen/sec. lesen, der Leser von Digitronics ist nicht dazu ausgelegt, daß er nun 24 Stunden am Tag 1000 Zeichen/sec. liest, er bricht in kürzester Zeit zusammen. Dagegen ist der 60-

Zeichenleser von IBM auf sehr starke Beanspruchung ausgelegt. Und genauso hier der Stanzer mit 150 Zeichen/sec. von Tally. Der billige Stanzer ist eben dafür gedacht, daß er gelegentlich 150 Zeichen/sec. stanzt, sagen wir 20 % der Zeit, aber nicht durchgehend, wohingegen der 15 Zeichenstanzer von IBM es durchgehend machen könnte.

6.3.1.4 Indirekte Lochstreifengeräte

Die indirekten Lochstreifengeräte sind der Flexowriter eine Art Schreibmaschine mit Lochstreifenzufuhr, um Lochstreifen, unabhängig von einem Rechner herstellen zu können, oder dann die Fernschreibmaschine für das 5-Kanalband oder neuerdings das 8-Spurenband.

6.3.2 Vor- und Nachteile

Die Frage, weshalb heutzutage noch Lochstreifen verwendet werden, läßt sich folgendermaßen beantworten:

Es sind gewisse Vorteile der Lochstreifen vorhanden, diese sind gegenüber Karten:

1. schnelleres Lesen als bei der Karte,
2. keine Schwierigkeiten mit der Reihenfolge der Zeichen und zwar deshalb, weil sozusagen die Zeichen auf dem Band eines an das andere angeheftet sind. Die Karten können aus der Reihe kommen (meistens fallen sie dem Operateur hinunter, wenn man eine wichtige Rechnung durchzuführen hat, und dann kommen sie zurück ohne Ergebnis, dann sieht man, daß die Karten nicht mehr in der Reihenfolge sind). Das kann bei den Lochstreifen nicht passieren, weil sie zusammenhängen.
3. keine Schwierigkeiten mit Verbiegen und abgegriffenen Karten.
4. sind auch noch Temperatureinflüsse zu beachten, praktisch vernachlässigbar für Lochstreifen, wohingegen

die Karten hier sehr empfindlich sind. Wenn ein Stapel für einige Monate in einem sehr trockenen Zimmer gelagert wurde, dann ergeben sich sehr große Schwierigkeiten, sie in irgendeinen Leser wieder einführen zu können.

5. Lochstreifen sind billiger pro Zeichen als Karten, einmal vom Medium her und zum anderen noch bedeutender von den Geräten her.
6. Größere Datendichte als bei Karten.
7. Leichter Transport großer Dateien. Hier belegt eine große Datei nur eine große Papierrolle oder zwei.

Die Nachteile sind folgende, die zu einem gewissen Teil die Vorteile aufheben.

1. sind Lochstreifen sehr schlecht abzuändern, d.h., ein Lochstreifen von 2000 Zeichen und 1 Zeichen ist zu ändern, dann kann praktisch die ganze Datei weggeworfen werden, wohingegen bei den Karten eben nur eine Karte ausgetauscht wird.
2. Haben wir keine Möglichkeit des Umordnens, d.h., sortieren bei den Lochstreifen, weil sie ja zusammenhängen, ist nicht möglich.
3. Eine Datenschlange. D.h., wenn eine Datei aus Lochstreifensegmenten besteht, dann kann sie ungeordnet werden, aber dann gibt es Schwierigkeiten mit dem Einfädeln in den Leser.

Nun, betrachtet man die Nachteile mit den Vorteilen und wägt sie gegeneinander ab, so springt ein wichtiges Anwendungsgebiet sofort heraus, das ist die Dateispeicherung, wenn wenig oder keine Änderungen mehr nötig sind. Die Speicherung erfolgt hier außerhalb des Systems, d.h., in einem System, das mit Time-Sharing arbeitet, sind gelegentlich die Platten voll. Dann müssen die Dateien ir-

gendwie herausgespeichert werden. Für die Dateien, die sich sehr wenig ändern, kann man Lochstreifen nehmen, das ist besser als Karten und zwar deshalb, weil eben die Vorteile hier da sind und dann kommt auch noch dazu, daß wieder schnell eingelesen werden kann, falls die Datei wieder gebraucht wird (Dead storage von Dateien). Außerdem sind die Lochstreifen die billigsten E/A-Geräte. Solche niedrige Ziffern für die Mietkosten sind schwer zu finden, wenn wir zu Karten und anderen Geräten übergehen.

6.3.3 Programmierung

Die nächste Frage betrifft die Programmierung. Einmal das Lesen. Dies ist oben bereits behandelt. Dann kommt das Stanzen von Lochstreifen. Dies kann z. B. durchgeführt werden auf folgende Weise:

LAC	x	Load ac
PLS		punch load select; füllt Stanzpuffer, löscht Flag und startet Stanzen
PSF		punch skip on flag; überspringe nächsten Befehl bei Erscheinen des Flags
JMP	.-1	jump back; warte auf das Flag

{ }

Prüfung für Ende des Stanzens, ggf. Wiederladen des Akkus und Wiederholen.

Das kleine Programm fängt mit einer LAC x-Instruktion an, die den zu stanzenden Wert vom Speicherplatz x in den Akku bringt. Dann folgt eine Instruktion PLS, punch-load-select, und zwar hat sie folgende Aufgabe: sie füllt den Stanzpuffer vom Akku aus mit dem Zeichen das rechts im Akku steht. Die Zahl der Bits richtet sich natürlich danach, welche Breite der Lochstreifen hat, wenn ich 8 Bits habe, dann werden 8 Bits aus dem Akku herausgenommen, usw. Das PLS füllt den Stanzpuffer vom Akku aus, löscht das Stanzflag und steuert das Stanzen. Die nächste In-

struktion bewirkt ein Abfragen des Punchflags durch PSF, d.h., punch-skip-on-flag. Sobald das Flag erscheint, wird die nächste Instruktion übersprungen. Und nun müssen wir wieder warten, bis die mechanischen Vorgänge ablaufen. Wir müssen auf den PSF-Befehl zurückspringen, bis das Zeichen ausgestanzt ist, was relativ langsam geht. Sobald das Flag erscheint, wird der Sprungbefehl übersprungen. Dann folgen noch Befehle zur Prüfung, ob alle Zeichen gestanzt sind, die gestanzt werden sollen.

Bei Ausgabe von vielen Zeichen, soll kontinuierliches Stanzen angestrebt werden, damit sie so schnell wie möglich herauskommen. Nachdem ein Zeichen ausgestanzt ist, muß das Band wieder vorwärts bewegt werden, damit stehen z.B. bei 63,3 Zeichen pro Sekunde Stanzgeschwindigkeit, nun 11,3 bis 15,8 Millisekunden zwischen aufeinanderfolgenden PLS-Instruktionen zur Verfügung. Falls etwas anderes zu tun ist, könnte dies getan werden. Diese Zeit ist nötig für den Vorschub des Lochstreifens und zwar deshalb, weil er für einen Augenblick während des Stanzvorgangs feststehen muß. Dann muß er wieder weiterbewegt werden.

6.3.4 Lesevorgang

Beim Lesevorgang haben wir, daß Licht oder ein Stempel durch die Stanzungen des Lochstreifens auf Fotozellen fällt oder Kontakte werden geschlossen. Vorher wird noch das Flag gelöscht beim Ausführen der Leseinstruktion. Der Fotostrom, oder der Strom von den Kontakten, wird abgegriffen, verstärkt und auf einen Puffer geleitet. Als nächstes wird der Lochstreifen auf die nächste Sprosse weitergeführt, und sobald dies der Fall ist, wird ein Flag gesetzt, d.h., jetzt ist der Lochstreifen wieder lesebereit. Beim Lesen mit optischen und elektrischen Methoden besteht normalerweise konti-

nuierlicher Betrieb. Dagegen nicht bei mechanischen, wo wir genauso wie beim Stanzen während dem Abfühlen den Lochstreifen nicht vorwärts bewegen können, sonst bleiben die Stempel stecken mit Vorschubhemmung, und das Gerät ist nicht mehr betriebsfähig.

Sobald das Flag erscheint, beendet dies nun den Lesevorgang. Dann wird der Sprungbefehl übersprungen und ich kann die Daten aus dem Puffer herausnehmen in den Akkumulator oder in den Speicher. Die Frage, wann eine solche Anordnung gewählt und nicht erst vorgeschoben und dann gelesen, sondern erst gelesen und dann auf das nächste Zeichen vorgeschoben wird, hat folgende Antwort. Es besteht hier ein kleiner Vorteil. Wenn mit dem Unterbrecher gelesen wird, dann kann die Zeit für etwas anderes verwendet werden, und man braucht nicht so lange warten.

6.3.5 Stanzvorgang

Der Stanzvorgang verläuft analog. Der PLS-Befehl leitet den Vorgang ein. Jeder Lochstreifenstanzer hat einen ähnlichen Befehl um den Stanzvorgang zu initiieren. Der Stanzvorgang löscht das Stanzflag, verstärkt die Signale vom Stanzpuffer, und gibt sie auf die Stanzstempel und stanzt dann den Lochstreifen. Dann wird der Lochstreifen um eine Sprosse weitergeschoben und das Flag gesetzt. Dies ist natürlich intermittierender Betrieb, weil man die Stanzstempel durch den Papiervorschub nicht verklemmen möchte.

6.3.6 Allgemeiner Vorgang

Generell ist es derselbe Vorgang für alle E/A-Geräte. Wir haben einmal

- a) das Ansteuern, des E/A-Gerätes z.B. hier durch das PLS, dann haben wir
- b) einen Datentransfer (Datenübertragung vom Puffer in

- den Akku bei der Eingabe oder umgekehrt vom Akku zum Puffer bei der Ausgabe). Dann haben wir als nächstes
- c) das Abfragen des Flags, und dann haben wir
 - d) Löschen des Flags oder Springen auf das Flag. Z.B. Springen auf das Flag kann durch das Unterbrecherwerk bewerkstelligt werden.

Es wurden bereits Puffer verwendet, einmal beim Lesen von Lochstreifen und ebenfalls wieder beim Stanzen des Lochstreifens. Was ist der Zweck der E/A-Puffer? Dies ist das Sammeln von Bits und Bytes zu größeren Einheiten, z.B. zu Zeichen, Wörtern oder Doppelwörtern, usw. So gibt es z.B. Hochleistungs-Lochstreifenleser, die bis zu 256 Zeichen in einem Puffer sammeln, wie z.B. bei dem erwähnten 1000-Zeichenleser. Man möchte den Rechner deshalb nicht zu häufig unterbrechen, deshalb wird hier ein großer Puffer verwendet, der dann in großen Zügen gespeichert werden kann. Nun noch eine Frage über die Flags, woher kommen sie? Die Flags werden von den betroffenen E/A-Geräten erzeugt. Die Flags, von den E/A-Geräten erzeugt, dienen zum Anzeigen der Beendigung einer Aufgabe; d.h., die Freigabe des Rechners für andere Aufgaben bei langsamen E/A-Geräten zu veranlassen. Langsame E/A-Geräte sind solche, wo z.B. die Zeiten zwischen den Zeichen oder Wörtern mehr als 50 Mikrosekunden betragen. Wo diese Zeit kleiner ist, haben wir schnelle E/A-Geräte. Außerdem, wo liegen die Puffer, sie sind in den E/A-Geräten, nicht im Rechner.

6.3.7 Bauformen

Kurz noch zu den Bauformen der Lochstreifengeräte. Sie betreffen im wesentlichen die Zu- und Abführung des Lochstreifens im allgemeinen. Dabei gibt es sehr wenig Variationen, abgesehen von den Methoden des Lesens und Stanzens hauptsächlich das Auf- und Abwickeln der Lochstreifen betreffend.

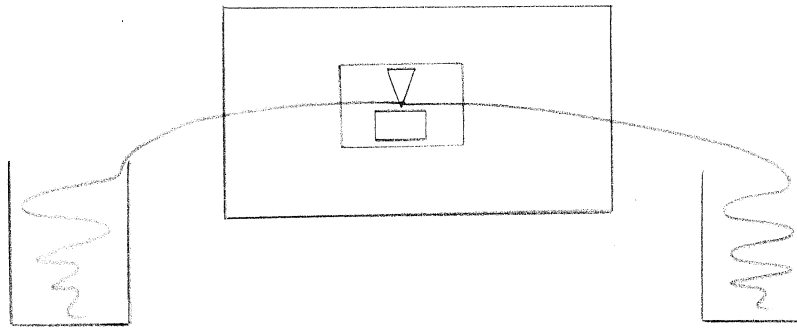


Abb. 27: Zwei Körbe, geeignet für kurze Bänder

In Abb. 27 sind 2 Körbe dargestellt, für kurze Bänder geeignet und zwar läuft das Band von einem in den anderen Korb herein. Für längere Bänder ist dieses Verfahren nicht sehr vorteilhaft, und zwar deshalb, weil sich das Band leicht verwickelt und dann reißt.

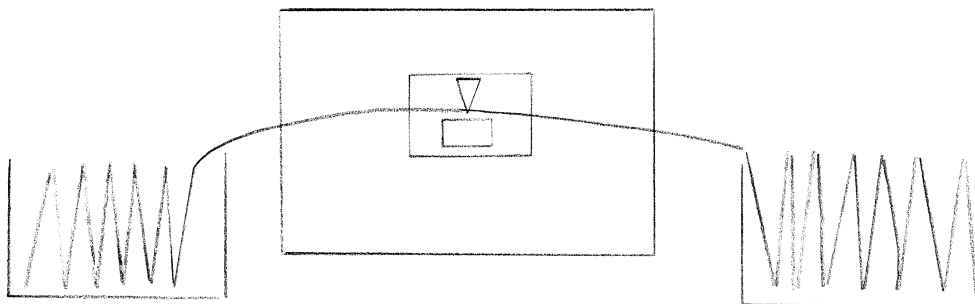


Abb. 28: Bänder gefaltet in Taschen

Das zweite Verfahren hat die Lochstreifen in Taschen gefaltet, wie der Balg bei einer Ziehharmonika. Das ist sehr praktisch und zwar deshalb, weil mit diesem Falten nach dem Lesen oder Stanzen der Lochstreifen nicht umgespult zu werden braucht, was bei dem nächsten Verfahren nötig ist, das im allgemeinen durchgeführt wird mit Auf- und Abwickeln.

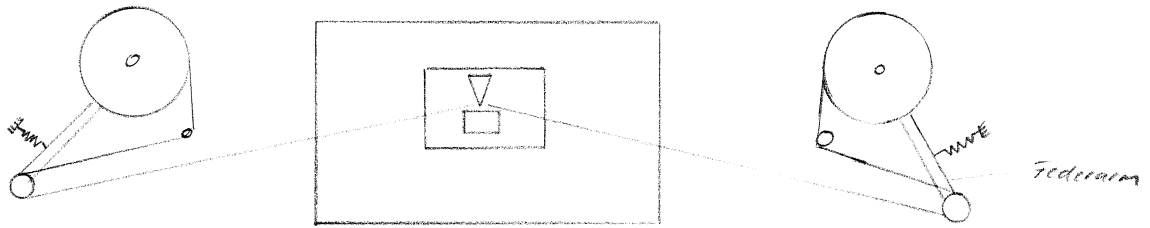


Abb. 29: Auf- und Abwickeln der Bänder

Hier werden die Lochstreifen von einer Spule abgewickelt, meist mit Federarmen, um die Lochstreifen zu spannen. Der Nachteil dabei ist, daß der Anfang der Bänder nach dem Durchlauf innen in der Spule sitzt, d.h., es muß dann umgespult werden, wie es bei Filmen nötig ist, oder eine andere Methode gefunden werden, mit der das Band dann von innen heraus gelesen werden kann. Es gibt solche Leser.

6.4 Lochkarten-Geräte

Lochkartengeräte sind aus zwei Gründen wichtig: Einmal als Direktgeräte, angeschlossen an das System zur Kommunikation von Daten und Programmen mit dem Rechner, und zum anderen als indirekte Geräte, unabhängig vom System zur Vorbereitung und Präparierung von Daten und Programmen. Außerdem zum Auslochen, Verifizieren, Sortieren und Mischen von Kartenstapeln.

Direktgeräte dienen zur Eingabe von Lochkarten und zur Ausgabe von Lochkarten durch Stanzen. Uns interessieren speziell die Direktgeräte und zwar weil die Geräte vom System her zu betrachten sind. In den letzten Jahren ist der Schwerpunkt von den Lochkartengeräten etwas abgegangen und zwar deshalb, weil Datenstationen mit direkter Eingabe

be in das System auf Dateien aufgekommen sind, sodaß man den Umweg über Lochkarten vermeiden kann.

6.4.1 Charakteristiken

Die Einteilung ist in Lochkartenleser, Lochkartenstanzer und dann Kombinationen von Lesern und Stanzern. Die wichtigste Datengröße ist auch hier die Geschwindigkeit, gemessen in Karten pro Minute. Bei den Lesern haben wir 100, 300, mit 100 K/m als untere Grenze (es existiert ein Kartenleser mit 50, doch ist dieser nicht allzu verbreitet). Für 400 bis 600 K/M existiert eine große Anzahl von Lesern mit verschiedenen Geschwindigkeiten. Bei den Stanzern ist das Minimum 91 K/m, dann 100, 250, 300, 400 und einen mit 500 K/m. Die Stanzerkombinationen müssen für Lesen und Stanzen spezifiziert werden. Es gibt einen mit 200 K/m Lesen und 200 K/m Stanzen, (300 und 270), dann 300 Lesen und 50 Stanzen, 400 und 400, 800 und 250, und 500 und 500, dann schließlich 1000 und 1300.

Tabelle 6.4.1.A: Charakteristiken der Lochkartengeräte,
Leser und Stanzer

Leser:

Geschwindigkeit	Hersteller	Miete (DM/Monat)
100	NCR	95
300	Uptime	140
400	IBM, H	700, 510
600	U, IBM, Uptime	750, 550, 275
1000	U, H	790, 900
1500	Uptime, RCA	830, 1780
2000	H, NCR	1500, 2000

Stanzer:

Geschwindigkeit	Hersteller	Miete (DM/Monat)
91	IBM	720
100	NCR	345
250	NCR, U	1100, 1240
300	X, B, H	2000, 1800, 2300
400	H	80
500	IBM	1760

Tabelle 6.4.1.B: Charakteristiken der Lochkartengeräte,
Kombinationen

Geschwindigkeit (K/m)								
	Lesen	300	300	300	400	800	500	1000
	Stanzen	200	50	270	400	250	500	300
Hersteller		U 603	IBM	H NCR	H	H	IBM	IBM
Miete/Monat		570	750	950	920	1820	1980	1850

In der Übersicht Tabelle 6.4.1 sieht man, daß die Werte sehr streuen, außerdem erkennt man sofort wieder die alte Erscheinung, ebenfalls wie bei den Lochstreifengeräten, daß die unabhängigen Hersteller, wie Uptime, wesentlich billiger sind als die Rechnerhersteller wie Univac und IBM. Zum Teil sind es nicht nur Preis-, sondern auch Qualitäts- und Anforderungsunterschiede an die Geräte. Die IBM-Geräte, z.B. sind im wesentlichen Hochleistungsgeräte, also für sehr starke Beanspruchung geeignet und die unabhängigen Herstellergeräte sind im allgemeinen nicht so sehr beanspruchbar. Bei den Stanzen erkennt man sehr starke Preisunterschiede, die daher rühren, daß verschiedene Zusatzmöglichkeiten vorhanden sind. Bei dem Xerox-Gerät gibt es z.B. Echo-Prüfung mit Lochzählung. Bei dem Burroughs existiert eine Paritätsprüfung; bei dem

Honeywell-Gerät ein Lesen nach dem Stanzen mit Paritätsprüfung usw. Aus diesen Unterschieden rührt natürlich auch ein großer Teil des Preisunterschieds. Außerdem gibt es Kombinationsgeräte, die die gleiche Anzahl von Karten lesen wie auch stanzen können, z.B. 200 - 200, 400 - 400 und 500 - 500. Diese Geräte sind zum Teil relativ älter und zwar deshalb, weil sich herausgestellt hat, daß im Normalbetrieb wesentlich mehr Karten gelesen als gestanzt werden. Sehr wenige Installationen stanzen dieselbe Zahl Karten, die auch gelesen werden. Deshalb ist es nicht nötig, eine Kombinationseinheit zu haben, die genauso leistungsfähig zum Stanzen wie zum Lesen ist. Daher rührt auch die andere Auslegung für die neuere Einheit, die IBM 2540, die 1000 Karten lesen und 300 Karten stanzen kann. dies ist gleichzeitig das Maximum der Leseleistung dieser Kombinationseinheiten. Bei den Kartenlesern ist das Minimum 100 K/m und das Maximum bei 2000 K/m, also ein Bereich der Leistung von einem Faktor 20 wird durch die Geräte überspannt.

6.4.2 Lese- und Stanzvorgänge

In Abb. 30 ist eine Kombinationseinheit skizziert.

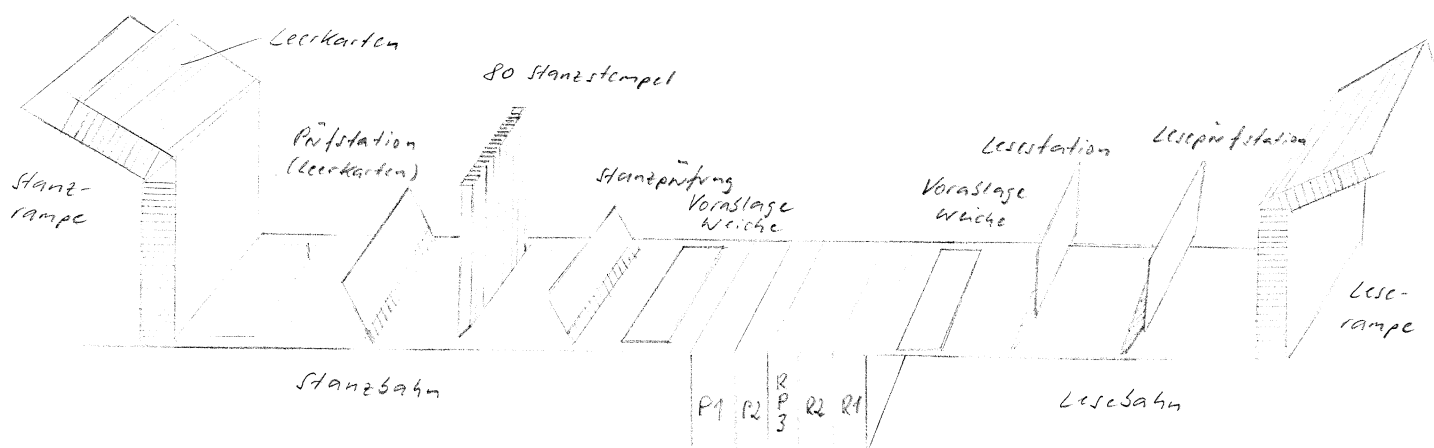


Abb. 30: Kombinationseinheit

Von den Charakteristiken muß man wissen, welche Bereiche möglich sind. Z.B. existiert kein Kartenleser, der 10.000 Karten pro Minute, auch keiner der 20 K/m liest. Genauso beim Stanzer; ein Stanzer der 1000 Karten stanzt, ist jenseits des derzeit technisch Möglichen. Wenn man ein neues System aufbauen will, oder neue Einheiten kaufen will, dann sind die Daten in der Tabelle wichtig. Diese Charakteristiken stammen aus dem Computer Review der GML Computer Corporation. Es gibt noch eine andere Zusammenstellung von Auerbach, Philadelphia, USA, die alle Rechner und Einheiten, die am Markt sind, untersucht hat. Und zwar sehr ausführlich, wesentlich ausführlicher als in dem GML Büchlein. Die angegebenen Zahlenwerte sind die neuesten Werte; bei den Preisen ist ein Dollar umgerechnet worden auf DM 2,75. Wenn man heute ein Gerät kaufen will, so muß man auf die Hersteller zurückgehen, aber die Preise werden wahrscheinlich nicht mehr als 10 % hin- und herschwanken. Die meisten Geräte werden nicht vom Ladentisch weggekauft, durch Verhandlungen, auch wegen dem Instandhaltungsvertrag, usw., sind zuweilen Konzessionen möglich. Außerdem, die angegebenen Preise sind für Einzelgeräte, wie die übrigen Charakteristiken. Beim Plattengerät und auch beim Magnetbandgerät ist folgende Erscheinung, daß es steckkompatible Geräte gibt. Dies beinhaltet, daß ein Gerät, sagen wir ein IBM-Gerät mit dem von einem unabhängigen Hersteller ohne weiteres ausgetauscht werden kann. Solche Geräte sind normalerweise wesentlich billiger, z.B. für 40 bis 70 % des Preises von IBM. Steckkompatibilität ist nicht immer der Fall, vor allem bei älteren Geräten, auch älteren Systemen. Wenn irgendetwas am System verändert wird, dann muß die innere Verdrahtung des Systems geändert werden. Denn es ist auf die peripheren Einheiten des Systems abgestimmt, wie auch die peripheren Einheiten auf das System abgestimmt sind. Bei den modernen Geräten ist es nicht mehr so und deshalb war es möglich,

die steckkompatiblen Geräte der unabhängigen Hersteller zu fabrizieren. IBM hatte das Standard-Interface mit der 360-Serie 1964 eingeführt. Vorher war es so, daß jedes E/A-Gerät extra angeschlossen werden mußte. Diese Erscheinung trifft nicht nur bei den E/A-Geräten zu, es ist auch bei den Speichern, Kernspeichern. Es gibt Systeme, die Komponenten von etwa 10 - 15 verschiedener Hersteller enthalten. Vor 10 Jahren wäre dies undenkbar gewesen.

Bei der Kombi-Einheit in Abb. 30 sieht man rechts die Lesebahn mit der Leserampe. Die Leserampe faßt normalerweise 2000 Karten, d.h., einen ganzen Karton Karten; es gibt eine Vorrichtung, die dann die Karten in den Lese-stapel eingibt. Beim Lesen einer Karte läuft diese durch die Leseprüfstationen, das sind Bürsten, die die Karten abfühlen, wenn sie herauskommen. Dann kommt die Lesestation selber, dann haben wir hier die Vorablage. Diese Vorablage ist eine Art Weiche, da wird bestimmt, wo die Karte von hier aus hingeht. Wozu wird man wohl die Leseprüfstation brauchen? Man kann ja die eine Karte eigentlich nur einmal lesen, das hat ja keinen Sinn da die Parität zu prüfen und zwar deshalb, weil praktisch alle Kombinationen in der Karte, sagen wir mal, Sie haben numerische Werte und Sie haben alphanumerische Zeichen, sagen wir mal, es bleibt die Zone weg, dann bleibt immer noch die Ziffer übrig. Der Leser kann dann nicht feststellen, daß hier etwas falsch ist. Die Paritätsprüfung hier am Kartenleser ist also sinnlos. Sonderfälle existieren, z.B., wenn Sie Sonderzeichen haben, dann kann es sein, daß wenn eine Lochung fehlt, die übrige Kombination nicht mehr zuverlässig ist und dies kann dann festgestellt werden. Aber im allgemeinen ist es nicht sehr sinnvoll, hier Paritätsprüfungen durchzuführen. Wozu wird dann die Lesestation gebraucht? Zum Vorlesen und dann Nachprüfen. Es wird nachgeprüft, ob irgendwie etwas kaputt gegangen ist, sagen wir, eine der Bürsten hebt sich etwas ab und hat keinen Kontakt mehr, dann wird im-

mer eine Null gelesen; dies kann dann festgestellt werden. Wenn die Karte durchkommt, wird in der Vorablage nun bestimmt, wo sie hinläuft. In den meisten neueren Karteneinheiten haben wir 4 Fächer für die Ablage der Karten.

Erstmal R 1 (Leserablage 1), R 2 (Leserablage 2), R 3 und P 3 (Lese- und Stanzablage 3). Auf der linken Seite sind entsprechend P 1 und P 2, die Stanzablagen. Die Stanzbahn verläuft etwas ähnlich der Lesebahn. Die Stanzrampe ist ebenfalls ausgerichtet, daß etwa 2000 Karten Platz haben. Von der Stanzbahn laufen die Leerkarten heraus unter die erste Station Bürsten, die nun diese Karten abfühlen. Wozu werden nun diese Karten abgefühlt? Der Stanzer hat ja noch nichts gestanzt. Die werden geprüft, ob sie nicht schon vorher Löcher enthalten. So ein Fall wird festgestellt und entsprechend bei der Ablage berücksichtigt. Die Stanzstempel gehen quer über die ganze Kartenreihe, dann folgt die Stanzprüfung, hier ebenfalls eine Bürstenstanzprüfung. Ebenfalls existiert eine Prüfstation und eine Vorablage. Die Karten werden "reihenweise" gelesen und gestanzt. D.h., in diesem Fall sind 80 Lesestationen und 80 Stanzstempel vorhanden für die 80 Spalten der Karte, ebenso 80 Bürsten für jede Lesestation. Der Grund dafür ist, daß die Lese- und Stanzvorgänge sehr langsam sind, so daß beim Bearbeiten der Breitseite nach, ein beträchtlicher Zeitgewinn erzielt werden kann. Intern werden dann die gelesenen Werte umgeschaltet auf Spalten für Leser und Stanzer, und zwar deshalb, weil normalerweise keine Karte verarbeitet wird, sondern die Zeichen in der Karte. Beim binären Arbeiten kann man die 80 Bits zeilenweise verwenden, fortlaufend in der Reihe (zeilenweise binär). In diesem Fall braucht man keine Umschaltung für die Zeichen. Neuerdings gibt es auch optische Leser, bei den Hochleistungsgeräten, dabei laufen die Karten nicht mehr zeilenweise, sondern spaltenweise. Der Vorgang

läuft relativ schnell sowieso, deshalb wurde hier dann ein Kompromiß geschlossen und zwar weil das Gerät dann im Aufbau wesentlich billiger kommt und die Karten entsprechend rasch abgefühlt werden können. Statt 12 Zeilen sind 80 Spalten dann in diesem Fall abzulesen. Beim Zeilenlesen braucht man einen Puffer von mindestens 80 Bitplätzen. Meist hat es wenig Sinn, aber manchmal wird so gearbeitet und dann intern im Programm oder im Betriebssystem wird dann umgesetzt. Oft ist es so, daß ein Puffer existiert für 960 Bits oder auch für 640 Bits, wenn wir einen 8-bit-Code haben pro Zeichen.

Wenn Sie ein etwas billigeres Gerät haben das spaltenweise arbeitet, dann kann man annehmen, daß es vielleicht nur die Möglichkeit hat, Zeichen zu verarbeiten bei denen der Puffer kleiner wird, so würden dann die geringeren Kosten herauskommen. Der Puffer muß mindestens 6 Bitplätze haben und zwar für verschlüsselte Zeichen in einem 6 Bit Code. So können verschlüsselte Zeichen im BCD-Code durch die Einheit abgelesen und sofort übersetzt werden; der Puffer enthält dann gleich das verschlüsselte Zeichen. Wenn man binär lesen kann, dann müssen eben 12 Plätze vorhanden sein. Der eigentliche mechanische Vorgang ist wenig interessant, vor allem relativ langsam.

6.4.3 Programmierung des Lesers

Die Programmierung der Lochkartengeräte soll auch hier wieder ohne Kanal- und Unterbrecherwerk erfolgen. Da die Arbeitsweise bei den Lochkartengeräten entweder alphanumerisch oder binär sein kann, also Zeichenlesen oder Bitlesen, falls es das Gerät erlaubt, müssen entsprechende Instruktionen dafür da sein. Z.B. lesen wir eine Karte spaltenweise alphanumerisch. Davon kann man annehmen, wenn es ein mechanisches Gerät ist, daß es ein-

fach ist. Wenn es zeilenweise liest, muß es ein etwas komplizierteres Gerät sein. Das Beispiel wurde aus gutem Grund gewählt, und zwar deshalb, weil man einen sehr komplizierten Kartenleser z.B. mit Zeilenlesung sehr wahrscheinlich nicht ohne Kanal- und Unterbrecherwerk bearbeiten wird.

Programm:

```

KRTLS :  CRSA          Card reader select alphanumeric,
                        starte lesen einer Karte
                        { } evtl. Instruktionen zur Initiali-
                        sierung;

KRTSP:   CRSF          CR skip on flag; überspringe näch-
                        ste Instruktion bei flag
                        JMP KRTSP   warte auf's flag
                        CRRB        CR read buffer
                        { }        Abspeichern des Zeichens oder Sammeln
                        in Wörter, Test für Lesende (80 Spal-
                        ten)
                        JMP KRTSP

```

Das Programm heißt KRTLS, Kartenlesen, dann tritt eine CRSA-Instruktion (card reader select alphanumeric) auf. Man müßte dann vermuten, daß es auch card reader select binary gibt, was bei diesem Gerät tatsächlich der Fall ist. Durch den Befehl wird eine Karte aus dem Stapel herausgezogen und in die Lesebahn hereingeschoben. Das dauert einige Zeit, während dessen könnten wir Instruktionen anbringen zur Initialisierung dieses Programms, z.B. zum Zähler und Adressen setzen; z.B. 80 Zeichen oder nur 60 Zeichen. Dies, sowie die Initialisierung der Adressen zur Speicherung der Zeichen kann in dieser Zeit aufgesetzt werden. Dann erscheint das Label KRTSP, Kartenspalte, mit der Instruktion CRSF, wieder die alte Instruktion: warten auf das flag, in diesem Fall heißt

es card reader skip on flag, d.h., falls das flag da ist, dann überspringe die nächste Instruktion. Die nächste Instruktion ist, wie zu erwarten, wieder ein Sprungbefehl zurück auf KRTSP. Es muß hier gewartet werden, bis eben die Karte unter den Leserbürsten erscheint, dann wird das Flag gesetzt und man kann lesen. Sobald das flag gesetzt ist, so heißt dies, daß der Puffer voll ist mit dem Wert des ersten Zeichens, der ersten Spalte. Dies muß nun vom Puffer durch das CRRB abgenommen werden, card reader read buffer. Der Inhalt des Puffers kommt in den Akkumulator. Rechts im Akku erscheint dann das Zeichen. Dann wird das Flag gelöscht. Nun kommt es darauf an, was für eine Art und Type von Leser ich habe, manchmal wird die exakte Bitrepräsentation des Zeichens gebracht, manche Leser übersetzen das Zeichen in die oktale Repräsentation des Codes für die Recheneinheit. Das sind jedoch Spezialitäten. Diese müssen im Einzelfall aus den Spezifikationen entnommen werden. Nachdem nun das Zeichen gelesen ist, von Spalte 1 muß es nun aus dem Akku abgespeichert werden oder wir können mehrere Zeichen im Akku durch verschieben ansammeln bis der Akkumulator voll ist, bevor wir abspeichern. Dann folgt noch der Test für das Leseende, d.h., ob wir unsere 80 Zeichen gelesen haben. Wenn nicht, dann muß zurückgesprungen werden nach KRTSP, d.h., wir warten nun auf die nächste Spalte, bis 80 Spalten abgelesen sind.

Hier tritt etwas anderes auf als bisher, daß die CRSA-Instruktion das Lesen einer ganzen Karte initialisiert, d.h., wenn die Karte kommt, müssen die Werte abgelesen werden, ob wir es wollen oder nicht. Wenn sie nicht abgenommen werden, sind die Werte eben verloren. Deshalb ist das Ankommen eines Zeichens durch das Flag gekennzeichnet. Ein solcher Flag bleibt in diesem Fall für 300 Mikrosekunden bestehen. Zwischen dem ersten Anruf einer Karte und dem Ablesen des ersten Zeichens sind in

einem Fall z.B. für einen Kartenleser, den ich hier als Beispiel genommen habe, 108 Millisekunden verfügbar. Dann bleiben zwischen dem ersten Flag und dem nächsten Flag, also zwischen den Spalten nur noch 2,3 Millisekunden übrig. Dies rührt aus der Mechanik und der Geometrie des Kartenlesers. Eine Karte muß erst einmal vorgeschoben werden, dann fallen diese Zeichen, diese Spalten, eine nach der anderen an.

Im Falle binären Lesens, z.B. bei diesem Kartenleser, sind 6 Pufferstationen vorhanden, d.h., wenn Sie binär lesen wollen. Doch eine Spalte in der Karte hat 12 Bit-Positionen, eine Schwierigkeit mit der Puffergröße von 6 Bits. Dies wurde hier intern verdrahtet. Es sind dann 2 CRSB erforderlich. Ich muß zuerst ein CRSB geben für binäres Lesen. Dann sind 2 CRSB erforderlich, d.h., erst wird die eine Hälfte der Spalte in den Puffer gebracht und dann wird diese weggespeichert und das Wegspeichern löscht das Flag, dann kommt sofort die zweite Hälfte herein und muß dann spätestens nach den nächsten 3 Mikrosekunden ebenfalls weggespeichert werden.

Wenn durch eine CRSA oder CRSB Instruktion die Lesebahn eingeschaltet wird, so läuft diese eine gewisse Zeit. Falls während dieser Zeit keine neue CRSA oder CRSB Instruktion erfolgt, schaltet das Gerät die Rollen automatisch selber aus. Es ist ein Verzögerungsschalter vorhanden, der durch die CRSA und CRSB Instruktionen gesteuert wird.

Wenn beim Lesen ein Fehler entdeckt wird, z.B. durch die zweiten Lesestationen, dann wird nun die Karte nicht mehr in dem normalen Ablagefach abgelegt, sondern in einem anderen, der nächsten R2 oder, je nach dem, der R3, das kann vorprogrammiert oder am Kartengerät eingestellt werden. Normalerweise schaltet die Einheit ab, wenn ein Fehler existiert, zum Eingriff des Operators,

um festzustellen, was falsch war und um die Sache entsprechend zu bereinigen.

6.4.4 Stanzer

Der Kartenstanzer arbeitet in ähnlicher Weise, allerdings mit dem Unterschied, daß die Bewegung in der Stanzbahn nicht mehr kontinuierlich verläuft wie in die Lesebahn, sondern intermittierend, und zwar deshalb, weil die Stanzstempel Zeit brauchen, um die Karte stanzen zu können. Es kann nicht gestanzt werden, während die Karte durchläuft, dagegen kann bei bewegter Karte gelesen werden. Das ist auch der Grund, weshalb die Stanzgeschwindigkeiten, in den Charakteristiken wesentlich kleiner sind als die Lesegeschwindigkeiten. Es gibt keinen Kartenstanzer mit 1000 Karten pro Minute, weil die Technologie derzeit nicht in der Lage ist, dies zu bewerkstelligen. Dagegen gibt es Leser mit 2000 K/m, das ist nicht so schwer.

6.4.5 Indirekte Kartengeräte

Die indirekten Kartengeräte sind bereits erwähnt und aufgezählt worden, auf Einzelheiten soll hier verzichtet werden, weil sie durch die Direkteingabe auf Magnetband und Direkteingabe in das System etwas überholt sind. Literatur und weitere Angabe dieser Geräte, falls man auf sie zurückkommen muß, da gibt es Manuals der IBM.

6.5 Schnelldrucker

Der Schnelldrucker ist wohl vom System her betrachtet, das wichtigste Ausgabegerät, weil normalerweise jedes System eine Riesenzahl von Ausgabedaten erzeugt; Stapel von Ausgabeseiten, die meist zum großen Teil vollgedruckt sind, mit bis zu 60 x 80, oder 60 x 160 Zeichen pro Seite (als

obere Grenze). Es gibt Sondergrößen, wo dann die Zahlen größer werden, z.B. 180, 200 oder so. Oft ist das Format nur 60 x 120 oder 50 x 120 Zeichen, je nach Standard der Installation.

Aus diesem Grunde existiert kaum ein System das nicht wenigstens einen Schnelldrucker hat. Meist haben kommerzielle Systeme nicht nur einen, sondern mehrere angeschlossen, vor allem mittlere und größere Systeme, um die große Anzahl von Ausgabedaten bewältigen zu können. Manchmal findet man bei kleinen wissenschaftlichen Installationen keinen Schnelldrucker, besonders bei älteren Systemen, wo eben relativ geringe Zahlen von Daten anfallen. Unter den Schnelldruckern müssen wir 2 Klassen unterscheiden, einmal Drucker mit fliegendem Druck und zweitens mit intermittierendem Druck. D.h., der fliegende Druck arbeitet derart, daß hier abgedruckt wird, ohne daß das Papier stillsteht. Während bei intermittierendem Druck bei Papierstillstand gedruckt wird; d.h., eigentlich ist nicht das Drucken das intermittiert oder kontinuierlich ist, sondern der Papiervorschub.

6.5.1 Charakteristiken

Die Charakteristiken der Schnelldrucker sind in Tabelle 6.5.1 aufgeführt (s. Seite 108). Wir haben drei Gruppen mit kleiner Leistung, mittlerer Leistung und dann mit hoher Leistung. Die Geschwindigkeit wird in Zeilen pro Minute angegeben, d.h., bis zu 160 x N Zeichen werden pro Minute angegeben oder gedruckt. Die Geschwindigkeiten sind 50 bis 60, dann 60 bis 150 Zeilen pro Minute (was die Bereiche bedeuten, werden Sie sofort sehen, wenn wir die Tabelle diskutieren), 300 bis 375 Zeichen pro Minute für kleine Leistung, die Hersteller dieser Geräte sind IBM, dann Datamark Corp. und eines von Mohawk Data Systems. Letzterer ist

	kleine Leistung				
Geschwindigkeit (Zeilen/min)	50 - 60	60 - 150	300 - 375		
Hersteller	IBM	Datamark	Mohawk		
Miete (DM/Monat)	450,--	550,--	700,--		
Kontrolleinheit Miete (DM/Monat)	0	0	450,--		
	mittlere Leistung				
Geschwindigkeit (Zeilen/min)	300-700	340-1000	800	1000-1200	1100
Hersteller	IBM	IBM	NCR	Mohawk	B
Miete (DM/Monat)	1450,--	1100,--	1800,--	1450,--	2000,--
Kontrolleinheit Miete (DM/Monat)	0	*	0	600,--	200,--
	hohe Leistung				
Geschwindigkeit (Zeilen/min)	1500 (3000)	1600	1900	2000-2500	
Hersteller	NCR	U	H	IBM	
Miete (DM/Monat)	2500,--	3400,--	4800,--	4800,--	
Kontrolleinheit Miete (DM/Monat)	0	2400,--		2000,--	

Tabelle 6.5.1: Charakteristiken der Schnelldrucker

z.B. einer dieser Hersteller, die steckkompatible Geräte herstellen zur Ablösung von IBM Geräten.

Die Mietkosten sind in DM pro Monat angegeben, umgerechnet zum Kurs von DM 2.75. Dazu kommt noch die Miete der Kontrolleinheit bei verschiedenen Geräten, auch in DM pro Monat. Wo kein Kontrollgerät nötig ist, wird das Gerät direkt vom Rechner kontrolliert. An eine Kontrolleinheit können normalerweise mehrere Drucker angeschlossen werden, die dann gleichzeitig arbeiten können. Die Geräte mittlerer Leistung gehen von etwa 300 bis 1100 Zeilen pro Minute. Der Mohawk Data Systems Schnelldrucker avisiert auf das IBM 2203 mit etwas größerer Leistung, die aber nicht immer benötigt wird und ist billiger. Eine mittlere Geschwindigkeit von 340 bis 1000 Z/min. liegt bei dem IBM Gerät 1403 vor, sehr erprobt und kostet DM 1100,-- Miete, die Zusatzkosten für die Kontrolleinheit sind nicht angegeben, d.h., das hängt vom Rechner ab, den man hat. In der Gruppe der Hochleistungsgeräte ragt das Gerät mit 2000 bis 2500 Zeilen/min. von IBM 3211 heraus mit DM 4800,-- Miete und DM 2000,-- für die Kontrolleinheit. Die Hochleistungsgeräte sind nicht gerade billig, deshalb setzen sich die Kosten eines Systems etwa zur Hälfte aus der Peripherie zusammen.

Die Bereiche, die in der Tabelle angegeben sind, besonders die Zahl 3000 bei dem NCR Gerät, bedeuten folgendes: falls numerisch geschrieben wird, gilt die obere Grenze, falls alphanumerisch geschrieben wird, gilt die untere Grenze. Wenn nur Ziffern geschrieben werden, dann kann man mit doppelter Geschwindigkeit ausgeben als wenn ich alphanumerisch schreiben muß. Dies kommt manchmal vor, generell hat es nicht sehr viel Bedeutung (es kommt vor beim Formulare, Kataloge drucken, usw., wenn nur die Preise eingedruckt werden müssen). Auch bei den Schnell-

druckern gibt es Variationen, besonders im Zeichensatz, d.h., in der Auswahl der verschiedenen Zeichen. Außerdem in der Länge der Druckzeile und auch in der Geschwindigkeit für Mehrzeilen-Vorschub. Das letztere bedeutet, daß manche Drucker bei Leerzeilen eine größere Geschwindigkeit haben.

6.5.2 Druckvorgang

Der Druckvorgang sieht folgendermaßen aus, s.Abb. 31.

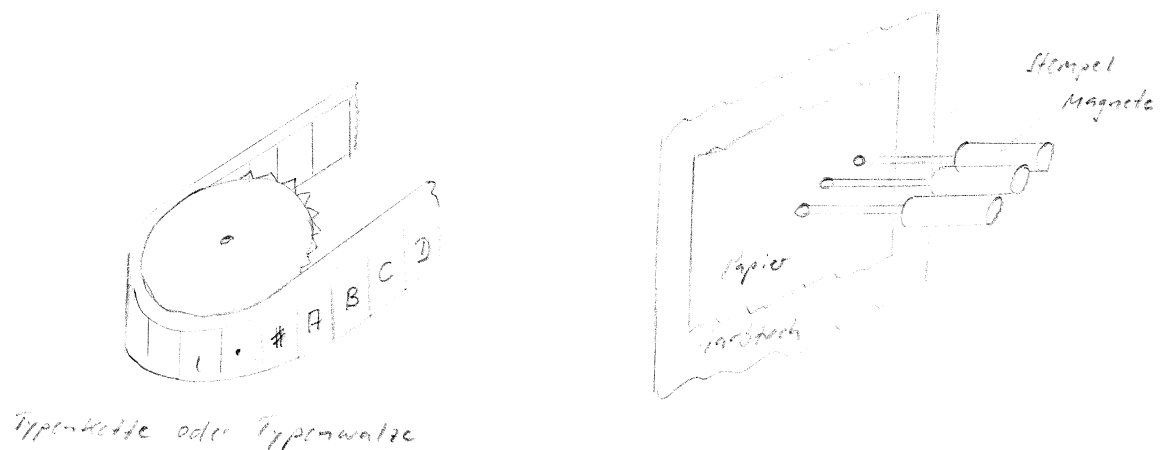


Abb. 31: Druckvorgang

Es existiert eine Reihe von Magnetspulen. In diese Magnetspulen führen die Druckstempel ein. Dann kommt hier das Papier, das vorgeschoben wird. Die gefalteten Papierstapel liegen meist unten. Hinter dem Papier kommt das Farbtuch, das von einer Rolle abläuft und wieder aufgerollt wird. Dann folgt in der modernsten Ausführung eine Drucktypenkette, die über ein Zahnrad läuft. Die Typenkette besteht, wie der Name sagt, aus den Typen, aufgereiht, eine an der anderen, z.B. A, B, C, usw. Diese Typen bilden die Kette, die dauernd umläuft. Es existiert nun für jede Druckposition auf dem Papier eine Stempereinheit, mit einer Spule und einem Druckstempel. Z.B., wenn ich 160 habe, sind 160 Spulen und

160 Stempel und nur eine Kette vorhanden. Deshalb muß die Kette umlaufen, damit sie alle Positionen drucken kann. Wenn nun die gesuchte Type erscheint, z.B. eine A soll an dieser Stelle gedruckt werden, dann wird hier die entsprechende Spule aktiviert und schlägt nun das Papier im richtigen Augenblick gegen das Farbtuch, und das Farbtuch gegen die Type und die Type erscheint auf der Rückseite (so in Abb. 31 gesehen) des Papiers. Das ist der Druckvorgang generell bei einem Gerät mit Typenkette.

Es gibt noch eine andere Möglichkeit, daß eine Walze existiert, eine Typenwalze, siehe Abb. 32, wo die Zeichen drauf sind; z.B. A, B, C, usw.

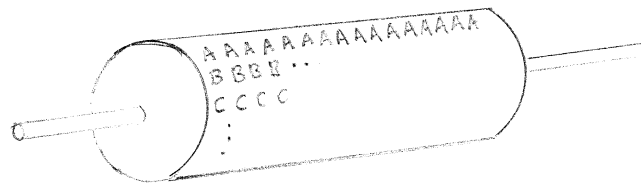


Abb. 32: Typenwalze

Jedes Zeichen muß hier für alle Druckpositionen existieren. Daraus wird die Vereinfachung, die die Kette mit sich bringt, ersichtlich. Man braucht jede Type nicht mehr für jede Druckposition, wie auf der Walze, sondern nur einmal. Das ist der Grund, weshalb die meisten Schnelldrucker, vor allem die mit höherer Leistung, alles Kettendrucker sind. Falls mit einem beschränkten Typensatz gedruckt wird, z.B. nur numerisch, dann kann die Typenkette ausgewechselt werden, mit einer Typenkette, die den beschränkten Typensatz enthält. Z.B.

kann ich einen Typensatz haben, der nur die Ziffern enthält, d.h., ich habe zweimal die Typen auf der Kette, und so die Druckgeschwindigkeit auf das Doppelte erhöht.

Gedruckt wird der Inhalt des Druckpuffers. Dieser enthält nun ebenfalls so viele Zeichen, wie der Drucker Druckpositionen hat, d.h., entweder 120, 135, 150 oder 160 Zeichen. Die Frage ist nun, wie wird nun gedruckt.

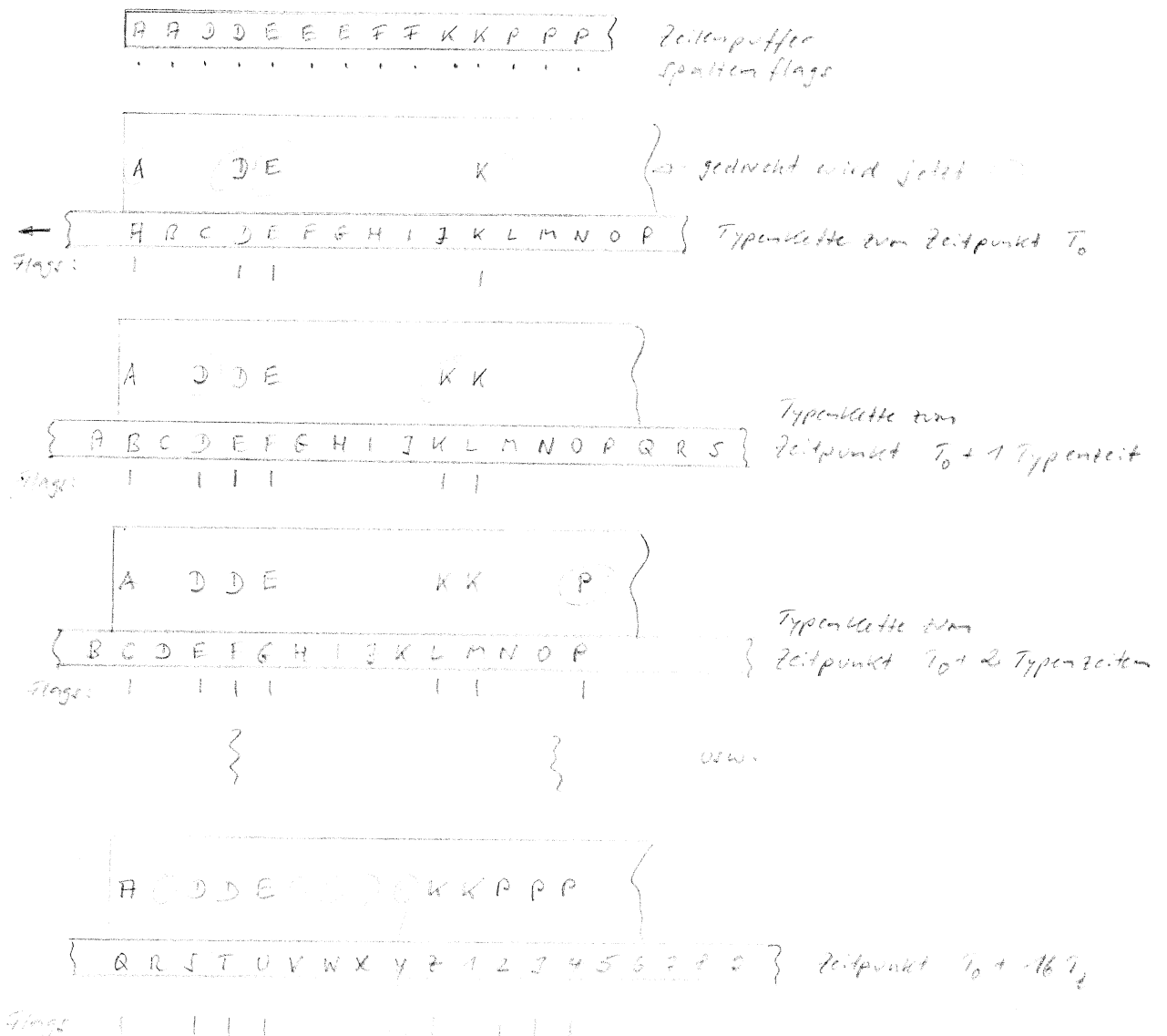


Abb. 33: Druckvorgang

Im ersten Augenblick habe ich in Abb. 33 die Typenkette mit den Zeichen A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P, usw., zum Zeitpunkt T_0 . Dahinter liegt das Papier, das von unten kommt und nach hinten übergeworfen wird, und nun werden in diesem Augenblick T_0 die Typen gedruckt, die da sind und vom Puffer verlangt werden. Das Zeichen A kann ich drucken, da die Type vorhanden ist, dann das Zeichen D, das erste E, und dann das letzte K. Das ist alles, was in diesem Augenblick gedruckt werden kann, von dem was im Puffer ist. Unter der Typenkette ist eine Reihe Flags gezeichnet, die gesetzt werden, wenn das Zeichen in der Zeile gedruckt wird. Zu diesem Zeitpunkt T_0 sind nur 4 Flags gesetzt, d.h., 4 Zeichen gedruckt worden. Die Kette läuft kontinuierlich um. Im nächsten Augenblick ist die Kette um eine Typenzeit weitergelaufen, und zwar das Zeichen A der Kette ist hier um eine Position vorgerückt. Die Typenzeit T_z , die ist definiert durch die Umlaufzeit dividiert durch N_z , die Typenzahl: $T_z = U_z / N_z$. Zum Zeitpunkt T_0 plus einmal T_z , in Abb. 33, ist die Typenkette eine Position weitergerückt. In diesem Augenblick kann nun das vordere D gedruckt werden. Ebenso das vordere K, das ist nun alles. Ich muß nun die Pufferflags ergänzen. Zu diesem Zeitpunkt habe ich nun 2 Zeichen gedruckt. Die Kette läuft nun um und es wird nun ein Zeichen nach dem anderen gedruckt, wie es vom Puffer verlangt wird. Z.B. der nächste Zeitpunkt bei $T_0 + 2 T_z$ in Abb. 33. Man muß meist zum Drucken einer Zeile einen vollen Umlauf warten. Es ist nicht etwa so, daß das Papier in diesem Fall kontinuierlich vorgeschoben wird, das ist nicht möglich, denn die Zeichen werden zu verschiedenen Zeiten gedruckt. Bei einem Drucker, der etwas ausgeleiert ist, sieht man, daß die Zeichen rauf und runter rücken, das zeigt an, daß das Papier ein bißchen weitergegangen ist und während des Druckvorgangs nicht stehengeblieben ist.

Wenn eine Zeile gedruckt ist, wird das Papier intermittierend eine weitere Zeile vorgeschoben. Wie wird das festgestellt, daß alle Zeichen vom Puffer gedruckt sind? Die Flags zeigen diesen Zustand an. Wenn alle Flags gesetzt sind, dann weiß die Einheit, daß alle Zeichen, die im Puffer vorkommen, gedruckt worden sind und dann wird nun das Zeilenflag gesetzt. Das Zeilenflag ist abfragbar durch ein Programm, oder ist mit dem Unterbrecherwerk verbunden.

Wenn ein Zeichen im Puffer erscheint, das nicht auf der Typenkette vorkommt, muß man noch eine Vorrichtung haben, die nach einem Umlauf anzeigt, daß noch ungesetzte Flags in dem Flagregister da sind und eine Fehlanzeige erfolgt. Das Gerät wird in diesem Fall normalerweise stehenbleiben. Es kann ein Schalter eingestellt werden, daß die Zeichenpositionen, die nicht gedruckt werden frei bleiben können.

6.5.3 Papiervorschub

Der Papiervorschub erfolgt unter Programmkontrolle.

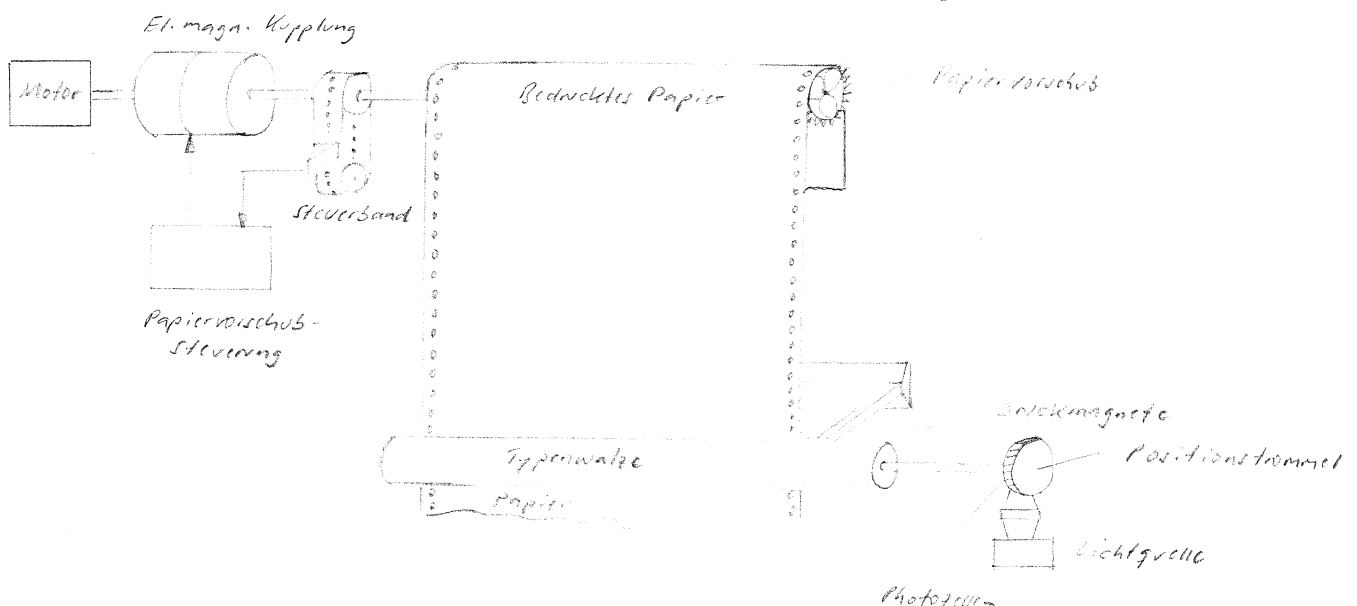


Abb. 34: Papiervorschub

Das Papier kommt von unten herauf, in Abb. 34, in einem Drucker mit einer Reihe von Fotozellen. Rechts über der Welle der Typenwalze ist die Positionstrommel. Darüber ist eine Lichtquelle. Wir müssen dauernd die Position der Trommeln oder entsprechend der Typenkette, kennen, damit dann die entsprechenden Druckmagnete betätigt werden können. In diesem Fall haben wir eine Rotation der Trommel. Eine Kette, die nur numerische Zeichen hat, hat dann eben einfach mehrere Anzeiger, und jedesmal, wenn einer davon an der Lichtquelle vorbeikommt, weiß das Gerät, daß hier die Typenreihe wieder anfängt. Hinter der Typenwalze, dem Papier und dem Farbtuch, sind die Druckmagnete. Dann existiert auf der oberen Welle ein Zahnrad für den Papiervorschub. Das Wesentliche ist die Steuereinheit, auf der linken Seite in Abb. 34, mit dem Kontrollband. Die Welle kommt über eine elektromagnetische Kupplung und dann auf einen Motor. In der Steuereinheit ist ein Abfühler für das Kontrollband. Letzteres ist nichts anderes als ein Lochstreifen. Dieser Lochstreifen, siehe Abb. 35, hat Vorschublöcher und enthält dann dazu die entsprechenden Lochungen.

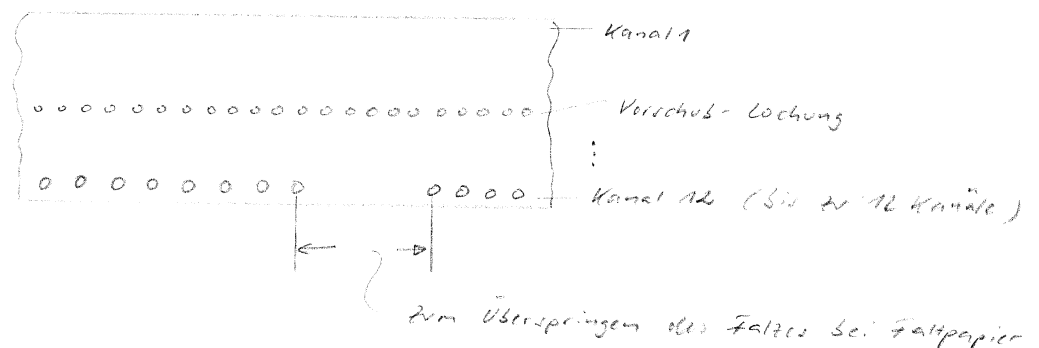


Abb. 35: Kontrollband

Diese werden abgefühlt mit dem Abfühler, meist optisch abgetastet, und betätigen über die Steuereinheit die Kupplung. In dem Kontrollband, dem Lochstreifen, sind bis zu 12 Spuren vorhanden und der Abstand zwischen zwei Löchern in Längsrichtung entspricht dem Zeilenabstand. Wenn beim Abfühlen ein Loch kommt, wird bis zur nächsten Zeile vorgeschoben. Falls der Falz zwischen zwei Seiten übersprungen werden soll, dann sind auf dem entsprechenden Stück des Kontrollbandes keine Löcher und es wird bis zum nächsten Loch abgetastet, d.h., die Kupplung rastet ein, und schiebt das Papier durch, bis eben wieder ein neues Loch erscheint. Von den bis zu 12 Kanälen hat jeder eine Bedeutung, z.B. ein Kanal ist für den Seitenvorschub, wenn eine ganze Seite vorgeschrieben werden soll, dann ist in diesem Kanal nur ein Loch auf dem Band; in der nächsten Spulenspur sind zwei Löcher um auf den Anfang der Seite und die Mitte der Seite vorspringen zu können, usw. Das Vorspringen auf die nächste Seite, wird oft vom Programmierer verlangt; z.B. in FORTRAN das Format 1H1 verlangt, daß nun der Drucker auf die nächste Seite vorspringt. In diesem Fall wird hier nun die entsprechende Spur abgefühlt. Das Kontrollband in der Steuereinheit ist austauschbar für besondere Formulare und Vordrucke.

6.5.4 Programmierung

Die Programmierung des Schnelldruckers soll wieder ohne Kanal und Unterbrecherwerk erfolgen, was in Wirklichkeit nicht sehr häufig der Fall ist. Es kommt selten vor, daß wir einen Schnelldrucker an ein System anschließen, das ohne Kanal und Unterbrecherwerk arbeitet. Die Programmierung zeigt hier jedoch am klarsten, auf was es hier ankommt.

```

DRUCKE: LPSF                line printer skip on flag, Schnell-
                             drucker springe auf's Flag

    JMP DRUCKE

    LSLX  X                  line spacing load spacing value,
                             lade Vorschubwert und starte
                             Initialisierung des Druckvorganges
                             Laden des Druckerpuffers

    LSSF                    line spacing skip on flag, Vorschub
                             springe auf's Flag

    JMP  .-1

    LPSE                    line printer select, drucke Inhalt
                             des Drucker Puffers
                             Prüfung, ob alle Zeilen gedruckt, wenn ja →
                             Ausgang, sonst JMP DRUCKE

```

Nehmen wir ein kleines Programm, DRUCKE genannt, dann ist die erste Instruktion LPSF, das ist Line-Printer-Skip-on-flag, d.h., warte auf die Beendigung der vorhergehenden Zeile. Hier nehmen wir an, daß der Drucker derzeit im Betrieb ist. In diesem Fall wartet das Programm auf das Ende des Druckens der augenblicklichen Zeile durch zurückspringen auf DRUCKE (Warteschleife für das Ausdrucken der vorhergehenden Zeile), sobald wir damit fertig sind, können wir mit dem nächsten Druckvorgang anfangen. Dieser wird gestartet mit dem LSLX, line-spacing load spacing value, d.h., z.B. das X repräsentiert den Wert des Papier-Vorschubs. Vorschub von einer Zeile würde den Wert 0 ver-

langen. In FORTRAN das Format IHI braucht den entsprechenden Wert für das X, das dann auf die neue Seite vorgeschoben wird. Es folgen in dem Programm Instruktionen für die Initialisierung des Druckvorganges, d.h., für die neue Zeile, die gedruckt werden soll und zwar, müssen wir die Adressen setzen, von woher die Puffer geladen werden sollen. Nun kommt das Laden des Druckerpuffers selber, entweder vom Akkumulator her oder über den Kanal vom Speicher. Nachdem das getan ist, kommen wir nun zum eigentlichen Drucken. Wir haben nun erst zu prüfen, ob der vorhin begonnene Vorschubsvorgang beendet ist; durch die Instruktion LSSF, line-spacing-skip-on-flag, d.h., wir warten hier auf die Beendigung des Vorschubs. Sobald das flag erscheint, wird der Sprungbefehl `jmp .-1` übersprungen und nun können wir das Drucken der Zeile verlangen, dabei wird der Inhalt des Puffers gedruckt, durch Übertragung des Inhalts in die Druckstempel. Das sind interne Vorgänge in dem Druckerlaufwerk. Hier haben wir dazu die LPSE-Instruktion, line-printer-select.

Normalerweise schließt sich dann eine Prüfung an, ob alle Zeilen ausgedruckt sind, wenn ja, dann springen wir heraus, wir sind fertig mit dem Vorgang, andernfalls springen wir zurück nach DRUCKE, zum Anfang dieses Programms.

Hier sehen wir eine gänzlich andere Anordnung der Befehle gegenüber früher, und zwar fangen wir hier nicht mit dem Drucken an oder mit dem Laden der Puffer und dann folgt das Drucken, sondern mit dem Warten auf das flag. Der Grund dafür ist folgender:

Wir haben hier eine Ausgabe-Operation vorliegen, die relativ wenig mit dem Rechner zu tun hat. Sobald diese gestartet ist, braucht sich der Rechner nicht mehr damit aufhalten. Dies bewirkt eine Art interne Pufferung, falls nur einzelne Zeilen ausgegeben werden müssen. Bei mehreren

Zeilen hintereinander muß dann eben auch gewartet werden wie früher. Bei der Eingabe ist der Vorgang so, daß der Rechner normalerweise die Daten zum Weiterarbeiten sofort braucht, deshalb müssen wir da unbedingt darauf warten. Bei der Ausgabe ist der Rechner normalerweise mit den Daten fertig, so daß wir auf die Beendigung des Ausgabevorgangs nicht zu warten brauchen. Allerdings, bevor wir einen neuen Ausgabevorgang starten wollen an dem Gerät, müssen wir prüfen, ob der vorhergehende Vorgang beendet ist. Später soll darauf eingegangen werden, wie die Vorgänge mit Kanal und Unterbrecherwerk bearbeitet werden. Sie sind dann wesentlich einfacher. Ein Zweck der Kanal- und Unterbrecherwerke ist ja, die Ein- und Ausgabe zu vereinfachen.

Bei den nichtmechanischen Druckverfahren haben wir generell

1. fotografische Verfahren,
2. elektrochemische,
3. thermische,
4. elektrografische,
5. xerografische und
6. magnetische.

Die magnetischen Verfahren sind etwas abwegig für Schnelldrucker. Unter den Verfahren müssen wir zwei Klassen unterscheiden. Einmal die Verfahren mit Nachbehandlung, d.h., mit äußerer Nachbehandlung des Mediums. Die Ausgabenblätter werden nachbehandelt, z.B. beim fotografischen, die Bilder entwickeln und fixieren, oder bei den anderen entsprechend. Zum anderen existieren die Verfahren ohne Nachbehandlung.

Nachbehandlungen verlangen normalerweise die Verfahren Nr. 1, 2, 4 und 6. Ohne Nachbehandlung dann die Verfahren Nr. 3 und 5.

Bevor wir die einzelnen Verfahren behandeln, soll kurz

auf deren Bedeutung eingegangen werden. Bedeutung haben zusätzlich die fotografischen und die xerografischen Verfahren erlangt, siehe folgende Tabelle für die Eigenschaften.

Nichtmechanische Druckverfahren:

- 1 Fotografische
- 2 Elektrochemische
- 3 Thermische
- 4 Elektrografische
- 5 Xerografische
- 6 Magnetische

2 Klassen: $\left\{ \begin{array}{l} \text{Mit} \\ \text{Ohne} \end{array} \right\}$ äußere Nachbehandlung $\left\{ \begin{array}{l} 1, 2, 4, 6 \\ 3, 5 \end{array} \right\}$

Erfolgversprechend:

- 1 COM am Markt (Computer output to Microfilm)
- 5 noch nicht am Markt: statt Zeile ganze Seite drucken (beliebig viele Kopien)

Tabelle 6.5.5 siehe Seite -121-.

Am meisten erfolgversprechend sind Verfahren der ersten Art und der fünften Art. Die Verfahren der ersten Art sind schon am Markt, die COM-Geräte, Computer Output to Microfilm, wo eine Art Mikrofilm beschrieben wird und damit die Daten gespeichert werden.

Verfahren fünf wurde von der Xerox Schnelldrucker der Presse vorgestellt, wobei die ganze

COM	Art	Erzeugung	Aufzeichnung	Nachbehand- lung	Geschw. Z/sec.
1	von Leuchtschirmen, Blenden, neuerdings Laser-Holographie	auf foto.Papier und Emulsion	entwickeln	~ 1.000	
2	Mosaikdrucke d. Stromfluß	durch Oxydation od. Farbe in Oberfläche	evtl. Trocknen	2-300	
3	Kontakt mit kleinen Heizelementen	Wärmemuster (Verfärbung)	-	50	
4	Hochspannungsimpul- se, Elektronenstrahl	Kunstharzschicht	entwickeln einschmelzen	1.000	
5	Elektr. (statische Ladungen) d.Belich- tung entfernt	Selenschicht oder Zinkoxydpapier	einschmelzen	1.000	

Tabelle 6.5.5: Nicht-mechanische Druckverfahren

Seite auf einmal gedruckt werden kann. Dies ist eine Dimension mehr als bisher.

Die Schnelldrucker herkömmlicher Bauart arbeiten mit einer Zeile pro Druckvorgang. Hier, mit dem xerografischen Verfahren haben wir die Möglichkeit eine ganze Seite pro Druckvorgang herauszubringen. Die schwächste Stelle in einem Ausgabesystem ist eben der mechanische Vorgang bei der Ausgabe. Das haben Sie bei den Eingabegeräten gesehen, die mechanischen Beschränkungen der Vorgänge sind die kritischen Parameter für die Geschwindigkeit. Genauso hier bei den Druckverfahren. Bei nur einer Zeile ausgeben, haben wir die mechanischen Vorgänge einmal, jedoch je mehr Daten wir ausgeben können pro mechanischem Vorgang, desto größer wird unsere Ausgabe-geschwindigkeit und damit können wir 9.000 Zeichen pro Ausgabevorgang auf normaler Ausgabeseite erreichen, wenn wir hier nun ganze Seiten ausgeben können. Wir haben bis zu 60 Zeilen á 150 Zeichen pro Seite, also maximal bis zu 9.000 Zeichen pro Seite. Eine Sekunde brauchen wir etwa schätzungsweise für eine Seite mit dem Xerox-Verfahren bei sehr großen Ausgabedichten.

6.6 Die Magnetbandlaufwerke

6.6.1 Charakteristiken

Magnetbänder sind eines der wichtigsten Zwischenspeichermedien für durchgehendes Arbeiten, eigentlich unersetzlich und zwar deshalb, weil wir ein sehr schnelles E/A-Gerät für direkten Betrieb haben. Andererseits können wir indirekt Daten und Programme vorbereiten, auf Magnetband bringen, und dann diese hier indirekt wieder einlesen. Es liegt Flexibilität in zwei Richtungen vor, einmal als

Direktgerät (angeschlossen zur Zwischenspeicherung) und dann als eigentliches E/A-Gerät für indirekten Betrieb mit dem Rechnersystem.

Daher rührt es, daß kaum ein System heute ohne Magnetbandlaufwerke existiert. In neuerer Zeit hat sich eingebürgert, daß kleine Systeme oft nur mit Platteneinheiten laufen. Z.B. wäre eine IBM 370/125 sinnvoll in der Auslegung der Konfiguration, wenn Sie einen Rechner haben mit Speicher und einer Platteneinheit, vielleicht einen kleinen Schnelldrucker und eine Karteneinheit, keine Bänder mehr. In den letzten Jahren hat sich dies eingebürgert, und selbst für die Datenspeicherung auf auswechselbaren Platten, die aber auch heute noch mit den Magnetbändern am billigsten ist. Wir können je nach Schreibdicke etwa 20 bis 40 Millionen Zeichen auf einem Magnetband speichern. Es gibt nur ein Medium das zur Speicherung von Informationen billiger arbeiten kann, das ist das Buch. Letzteres ist preislich unübertroffen. Das rührt daher, daß wir eine große Auflage haben, in der mechanischen Vervielfältigung, daher der niedrige Preis.

Bei Dateien, die nur einmal oder in nur wenigen Exemplaren existieren, ist das Magnetband immer noch das Billigste. In neuerer Zeit haben wir die Plattentürme oder Plattenstapel, die nun auch zur Speicherung verwendet werden. Auswechselbare Platteneinheiten, die dann ebenfalls entsprechend in Plattenlaufwerke eingesetzt und herausgenommen werden können. Nun, wenn wir kurz über die Charakteristiken bei der Magnetbandlaufeinheit sprechen, so sehen wir eine starke Streuung der hauptsächlichen Charakteristik und das ist die Durchlaufgeschwindigkeit des Bandes und damit zusammenhängend die Übertragungsgeschwindigkeit der Daten. D.h., die Übertragungsrate ist gleich dem Produkt aus der Durchlaufgeschwindigkeit multipliziert mit der Informationsdicke (oder

Schreibdichte). Die Schreibdichte, das sind so und soviel Bits pro Zoll, Zeichen pro Zoll, und dann die Geschwindigkeit, das sind dann so und soviel Zoll pro Sekunde, damit haben Sie die Übertragungsrate in Zeichen pro Sekunde.

Tabelle siehe Seite -125-

Die erste Zeile der Tabelle gibt die Durchlaufgeschwindigkeiten in Zoll pro Sekunde an, die nächste die Herstellerfirma, dann die Übertragungsrate, die Informationsdichte und den Mietpreis in DM pro Monat. Schließlich noch eine Angabe über die Kontrolle des Laufwerks.

Es sind drei Gruppen von Spalten gezeigt, einmal ein kleines Sondergerät, dann ein Normal- und ein Hochleistungsgerät.

Das Sondergerät hat eine Geschwindigkeit von 7,5 Zoll/sec. es ist sehr langsam; von Collins Radio hergestellt. Die Übertragungsrate ist 15 bits/sec.; die Informationsdichte ist 2000 Bits pro Zoll ...

Die Normalgeräte liegen in der Geschwindigkeit zwischen 20 und 150 Zoll/sec. Die gebräuchlichsten Schreibdichten sind für die schnellen Geräte dieser Klasse 800 Bpt. Die Normalgeräte werden von allen Rechnerherstellern und unabhängigen Herstellern angeboten. Die monatlichen Mietpreise zwischen 800 bis 1500 DM/Monat richten sich nach den Übertragungsgeschwindigkeiten und Zeichendichten der Geräte. Weiterhin werden einige Hochleistungsgeräte mit 1600 Bpt von den großen Rechnerherstellern angeboten, die etwa auch im Preis doppelt so hoch liegen wie die Normalgeräte. Zusätzlich zu den Magnetbandlaufwerken werden Kontrolleinheiten für bis zu 16 Bändeinheiten benötigt (außer bei zentralrechnerkontrollierten Geräten), die zwischen 500 und 3.300 DM/Monat Mietpreis liegen.

Geschw. Zoll/sec.	Sondergerät 7,5	Normalgeräte 20 - 150	Hochleistungsgeräte					
			200	200	200	200	200	200
Hersteller	Collins	alle Rechnerhersteller und unabhängige Hersteller	RCA	IBM	H	HP	HP	B
Übertragungsrate Z/sec.	15 bits/sec.	3 - 30 K Zeichen/sec.	320	160- 320	37- 71	20- 40	15- 60	17- 72
Informationsdichte	2000 Bits/ inch	200-556-800	1600	800/ 1600	800/ 1600	200/ 800	200/ 800	200/ 1600
Preis (Miete) DM/Monat	nicht angegeben Contr.d.ZR	800-1500 DM/M Plug-Compatible	2400	1600	1000	600	1300	3000
								Contr.d.ZR

Tabelle 6.6.1: Charakteristiken der Magnetbandgeräte

Das sind die wesentlichen Charakteristiken.

6.6.2 Aufbau der Laufwerke

Das Magnetbandlaufwerk hat etwa folgenden Aufbau:

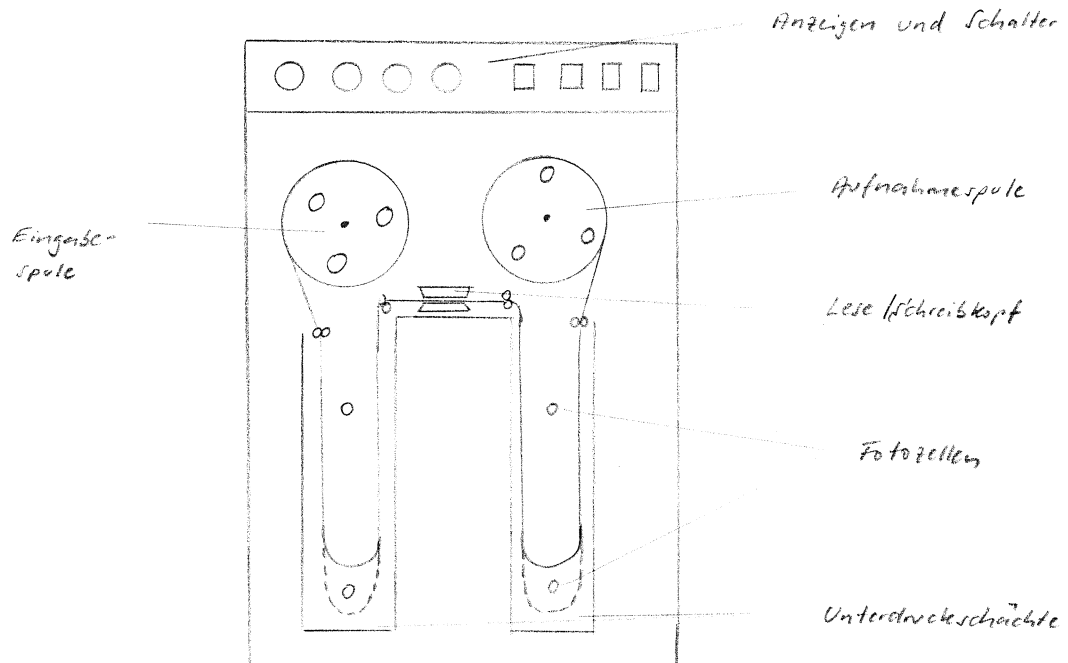


Abb. 36: Aufbau eines Magnetbandlaufwerks

Wir haben einen Kasten, wo oben Anzeigen und Schalter angebracht sind, Anzeigen über den derzeitigen Zustand des Laufwerks und Schalter zur Bedienung und Änderung der Zustände des Laufwerks. Wir müssen das Fach oben unterteilen. Dann haben wir die Magnetbandspulen. Es sind zwei vorhanden, das eine ist die Eingabespule, das andere ist die Aufnahmespule. Die Lade- oder Eingabespule ist das Magnetband, von dem wir lesen wollen oder auf das wir schreiben wollen. Die Aufnahmespule ist, wenn wir anfangen wollen, eine leere Spule. In den meist gebräuchlichen Laufwerken habe ich dann zwei Schächte, links und rechts je einen Schacht und zwar

dient jeder Schacht zur Aufnahme von Magnetbandschleifen. Hier sind zwei Rollen und nun geht das Magnetband in diesen Schacht herein, läuft aus dem Schacht heraus wieder über zwei Rollen und dann unter die Lese- und Schreibköpfe und wieder über zwei Rollen und in den anderen Schacht hinein, aus dem anderen Schacht heraus und wieder durch zwei Rollen und in die Aufnahmespule. Das ist schon der Weg des Bandes, wie ich ihn hier gezeichnet habe. Diese Schächte haben, entweder wie im Bild, Löcher eingezeichnet, das wäre dann mit Vakuumbetrieb, oder sie sind zum selben Zweck mit Fotozellen ausgestattet. Also entweder gibt es zwei Löcher je oben und unten in der Mitte des Schachtes, wo Vakuum abgesaugt wird und je nachdem, ob an beide Löcher Luft kommt oder nicht, weiß nun das Laufwerk wie lange der Vorschub des Bandes ist, der sich in dem Kasten befindet. Entsprechend wird dann von den Spulen nachgeliefert oder abgezogen. Durch Fotozellen läßt sich dasselbe erreichen, z.B. solange die untere Fotozelle Licht empfängt, d.h., das Magnetband ist dann oberhalb dieser Station. Wenn ich vorwärts arbeite, kann ich noch das Band verschieben bis der Kontakt unterbrochen wird. Nun kommt die zweite Station der Fotozellen oben. Diese zweite Station läuft in Tandem mit der ersten Station in einer Art Wechselschaltung. Sobald hier unterbrochen wird, wird festgestellt, daß das Band läuft. Wenn die Photozellen der oberen Station ebenfalls Licht empfangen, dann wird das Magnetband von der Spule nachgeschoben, bis der untere Kontakt unterbrochen wird. Die untere Station dient zum Abschalten des Laufwerks der Ladespule und die obere dient zum Anschalten der Ladespule, d.h., der Kupplung des Motors.

Entsprechend dann umgekehrt für die Aufnahmespule beim vorwärts arbeiten; beim rückwärts arbeiten sind dann die Seiten vertauscht. Genau denselben Zweck erfüllen die Vakuumlöcher; hier wird dann eben nicht mit elektrischen

Methoden gearbeitet, sondern mit pneumatischen Methoden; da wird eben dann ein entsprechender Schalter eingebaut, aber generell ist es dasselbe.

Nun, worüber sprechen wir hier, wir sprechen hier über das Problem der Schnellzuführung durch mechanische Gebilde. Wir haben gesehen, daß die Lochstreifen von einer Spule abgeführt werden und dann auf eine andere Spule aufgeführt oder von einer Art Balg abgeführt und von einem anderen Balg aufgeführt werden. Wenn ich kurze Datenmengen zu lesen oder zu schreiben habe, muß ich jeweils das ganze Gebilde beschleunigen oder verlangsamen. Sie wissen, bei der großen Datendichte, über die wir gesprochen haben, würde hier bei den Magnetbändern eine sehr große Schwierigkeit auftreten und zwar deshalb, weil bei jeder Beschleunigung und Verlangsamung eine entsprechend große Lücke auf dem Band auftritt. D.h., wir wollen diese Lese- und Schreibbeschleunigungen so klein wie möglich halten, bzw. die Zeiten so kurz wie möglich oder den Vorschub des Bandes so einfach und leicht wie möglich gestalten. Das ist der Grund, weshalb hier eigentlich zwei Arten von Vorschub gegeben werden. Einmal haben wir den groben Vorschub durch die Spulen, die dann entweder durch die Vakuumanlage oder die Fotozellen gesteuert werden und dann haben wir den feinen Vorschub, der durch die Rollen gesteuert wird. Sie sehen, der Feinvorschub, der hat nur ein kleines Stückchen Band zu bewältigen, das daran liegt und zwischen den Spulen sich herbewegt.

Die Sache ist, daß wir hier eine zweistufige Bewegungsvorrichtung haben, einmal Vor- und Nachschub von und zur Spule, zum zweiten den Vor- und Nachschub von und zu den Lese-/Schreibköpfen. Beides ist hier getrennt und deshalb kommen wir auf große Übertragungsgeschwindigkeiten beim Magnetband. Das könnte auch sinngemäß auf Lochstreifen übertragen werden. Es kommt jedoch nicht viel dabei heraus und zwar deshalb, weil die Informationsdich-

te auf dem Lochstreifen zu gering ist. Grund ist die Informationsbreite, es sind nur etwa 5 Zeichen pro Zoll, da lohnt es sich nicht; beim Magnetband haben wir 1600 Zeichen pro Zoll, da ist es wert, daß man eine solche Ergänzung der Bewegungseinrichtung vornimmt.

Eingebürgert haben sich diese Unterdruckschächte mit den Vakuumlöchern oder den Fotozellen.

Andere Methoden arbeiten mit Bandschächten/Taschen, mit Schwinghebel-Umlenkrollen und mit Unterdruckkammern.

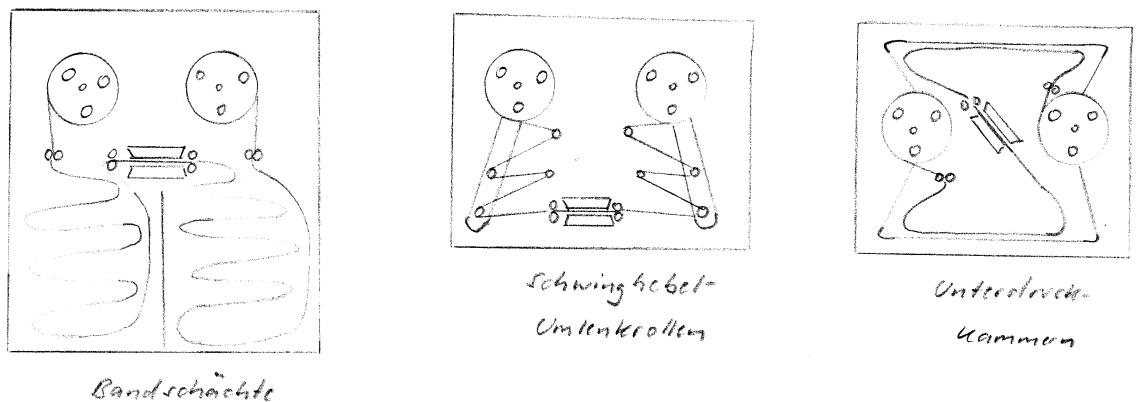


Bild 37: Verschiedene Laufwerktypen

Bei der nächsten Methode (b) haben wir wieder einen Lese-/Schreibkopf und nun haben wir hier Bandschächte oder Bandtaschen, d.h., das Band, das läuft herunter in einer großen mehrmals umgewickelten Schleife, geht dann unter den Lese-/Schreibkopf und läuft in die Aufnahmespule. Der Vorgang, ebenfalls zweistufig, sorgt dafür, daß immer genügend Vorschub da ist. Eine dritte Methode ist c) mit Schwinghebel-Umlenkrollen. Diese Schwinghebel sind an der Welle angebracht, auf der die Spule läuft, mit Rollen darauf, sowie dann feste Rollen und den Lese-/Schreib-

kopf. Nun geht das Band von der Spule einmal über eine feste Rolle, dann über eine bewegliche Rolle, über die nächste feste Rolle, bewegliche Rolle usw; dann unter dem Lese-/Schreibkopf vorbei. Auf der Aufnahmeseite haben wir entsprechend eine Vorrichtung derselben Art. Diese Schwinghebel stehen unter Federdruck, so daß immer genügend Vorschub vorhanden ist. Sie sehen, der Nachteil dieser Anordnung ist, daß es hier nur eine einstufige Bewegungsvorrichtung ist. Das ist genau dasselbe, was wir bei den Lochstreifen haben, manche Lochstreifengeräte haben eine ähnliche Einrichtung. Die letzte Methode d) arbeitet mit Unterdruckkammern, vor allem gebräuchlich bei kleineren Geräten. Wenn Sie eine sehr kleine Einheit haben, können Sie dieses Verfahren nützlich finden und zwar ist es so, daß hier die Schächte etwas eckiger als vorher ausgebaut sind. Die Lese-/Schreibköpfe sind etwas schräg angebracht. Sie sehen, das ist eine raumsparende Anordnung der Variation a. Hier sind dann Rollen, das Band geht am Lese-/Schreibkopf vorbei. Hier sind wieder ein paar Rollen angebracht und nun geht das Band in einen Schacht rein, hier ist eine Vakuumöffnung und das Band geht entweder darüber hinweg oder nicht. Die Schächte sind vorne und hinten mit Glasplatten geschlossen; sie haben genau die Breite des Bandes, sind also einen halben Zoll breit, und wenn nun das Band über die Löcher hinaus geht, dann kann Luft durchtreten, wenn nicht, dann wird die Luft durch das Band abgeschlossen. Bei den Unterdruckkammern wird das Band ebenfalls in diese Ecken angesaugt, und dann sind zwei Rollen, die das Band von der Ladespule herunterbringen. Auf dieser Seite geht es zurück, ebenfalls durch zwei Rollen auf die Aufnahmespule. Das ist das Verfahren mit den Unterdruckkammern.

Nun, was sind dabei die Unterschiede? Die Unterschiede sind folgende: Sie sehen, daß man eine relativ große Länge von Band zum Lesen oder Schreiben bereit halten

muß, das auf verschiedene Weise geliefert wird. Der Nachteil bei den Bandtaschen ist, daß das Band sehr stark abgerieben wird. Das Magnetband wird sehr stark beansprucht durch die Beweg-Berührung, es schleift aufeinander, genauso bei Methode b), das Band wird hier sehr stark beansprucht durch die Hebel (durch die Federhebewegung), auch durch die Umlenkung des Bandes. Hier, sehen Sie, wird das Band um 180° umgewendet, nicht nur einmal, sondern 2-, 3-, 4-mal. Genau derselbe Nachteil existiert bei der Unterdruckkammer, das Band wird sehr oft sehr stark umgewendet, was sich eben auf die Lebensdauer der magnetisierbaren Schicht nachteilig auswirkt. Dies ist nicht so sehr der Fall bei den Unterdruckschächten oder Fotozellenschächten, hier ist die größte Umwendung nur 90° , deshalb hat sich diese Anordnung am besten bewährt, insbesondere bei den schnellen Einheiten. Die Schnelleinheiten arbeiten nach diesem Prinzip, langsamere Einheiten können auch auf dem anderen Prinzip arbeiten mit einfachem Vorschub.

Das wichtigste Element am Bandlaufwerk ist natürlich der Lese-/Schreibkopf, denn dort werden die Daten vom Band auf die Einheit übertragen. Das ist das wichtigste und auch das kritischste Element.

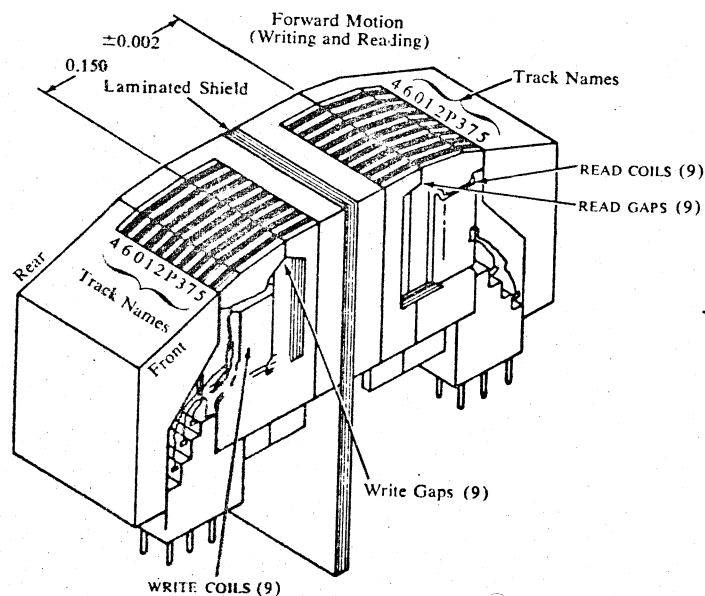


Abb. 38: Aufbau der Lese- und Schreibköpfe

Hier ist der Schreibkopf, hier ist der Lesekopf. Oben habe ich Schreibschlitze, d.h., Schreibspuren, das sind Magnete und bei den modernen Einheiten haben wir 9 Schreibspuren. Die Schreibströme erzeugen Felder in den Magneten und erzeugen ebenfalls Felder um die Schreibschlitze. Dazwischen haben wir eine Abschirmung, die Abschirmung des Lesekopfes vom Schreibkopf, denn wir wollen keine magnetischen Felder vom Lese- zum Schreibkopf induzieren. Hier ist ein Schlitz, die Magnete haben den Schlitz und hier an dem Schlitz wird geschrieben, das eigentliche Schreiben erfolgt an dem Schlitz. Genauso erfolgt das Lesen an einem entsprechenden Schlitz, hier haben wir die Lesespuren, ebenfalls 9, an dem Schlitz werden durch Felder in dem Band Spannungen induziert. Dann habe wir noch als Anordnung den Schreibschutzring zu betrachten.

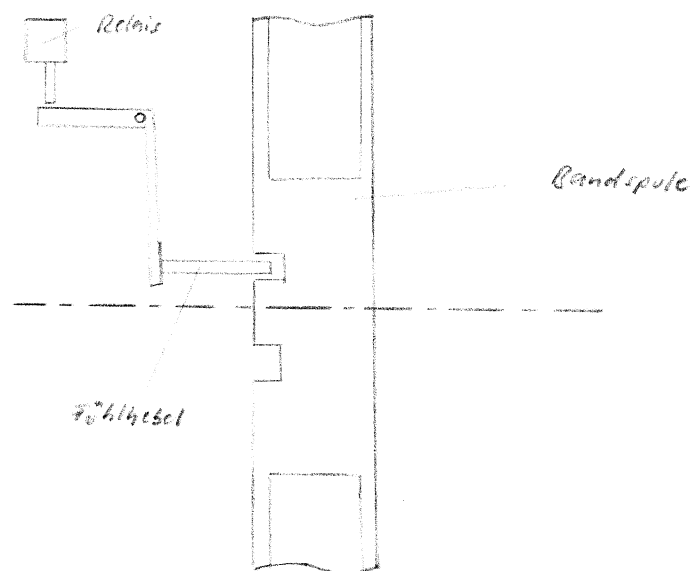


Bild 39: Schreibschutzring

Wir haben hier in der Mitte der Spule eine Öffnung, mit der das Band auf die Banderinheit gebracht und aufgesetzt wird, und in der Spule ist hinten ein runder Schlitz, auf

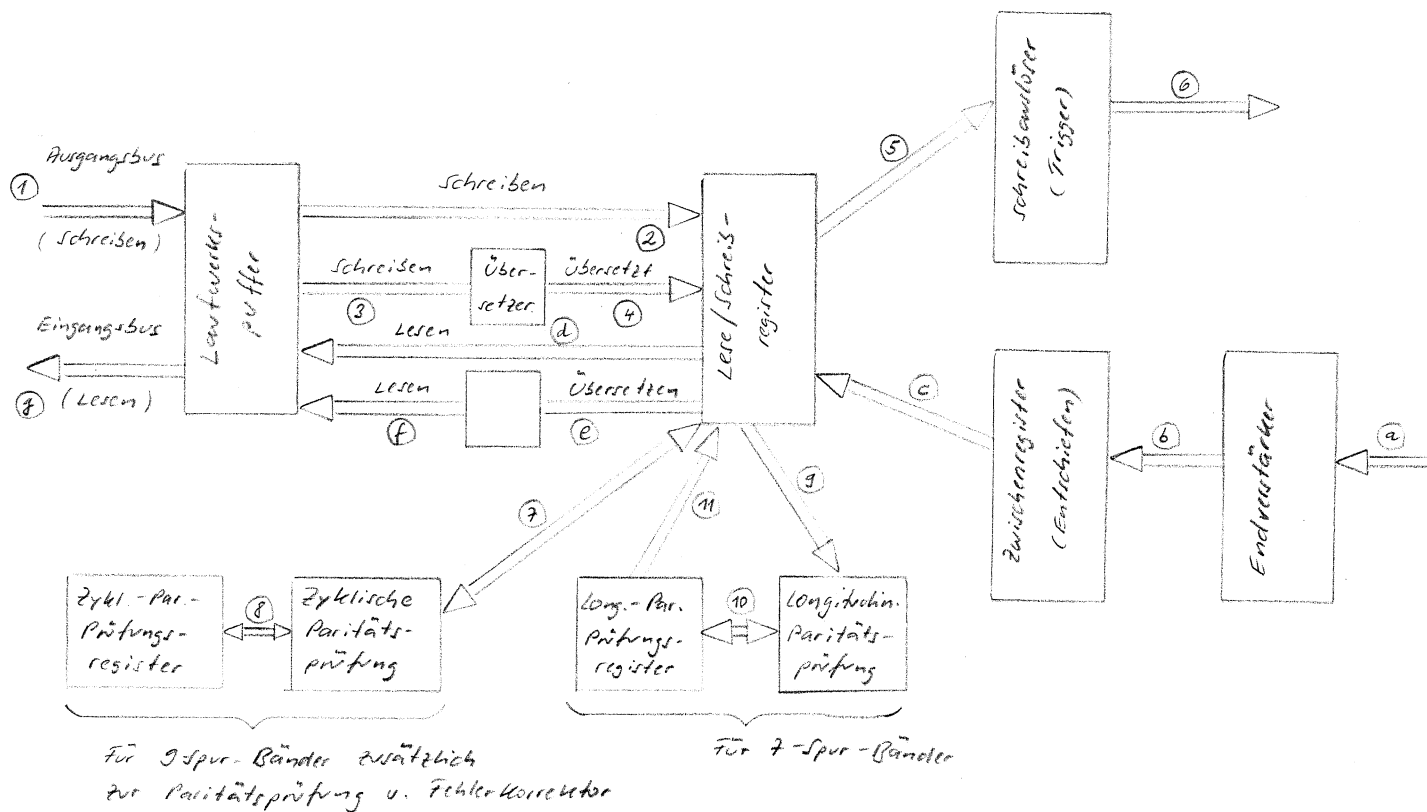
dem der Schreibschutzring liegt, der durch einen Taststift von dem Laufwerk abgefühlt wird. Wenn der Schreibschutzring nicht da ist, dann kann das Band nicht beschrieben werden, wenn der Schreibschutzring da ist, dann kann geschrieben werden.

6.6.3 Lese-/Schreibvorgänge

Das Interessantere sind die Vorgänge, die sich an dem Magnetband abspielen.

Abb. 4o siehe Seite -134-

Hier haben wir einen Laufwerkspuffer, dann haben wir hier einmal den Ausgangsbus zum Schreiben. Hier kommen wir vom Rechner über den Ausgangsbus in den Laufwerkspuffer zum Schreiben auf Magnetband und dann natürlich entsprechend einen Eingangsbus (das ist zum Lesen vom Magnetband). Von hier haben wir nun der Information zu übergeben oder von einem Lese-/Schreibregister zu bekommen. Die Daten können direkt geschrieben werden, oder es kann mit Übersetzung geschrieben werden. Nun, weshalb brauchen wir das? Ich bin darüber bisher hinweggegangen und zwar ist es folgendes: Sie finden zwei Typen von Laufwerken auf dem Markt und zwar sind es einmal 7-Kanal oder 7-Spur-Laufwerke und dann 9-Spur-Laufwerke, die älteren sind 7-Spur-Laufwerke. Wenn ich ein 9-Spur-Laufwerk habe, das haben wir vorhin bei dem Lese-/Schreibkopf gesehen, ist das einzige, was verschieden ist, eigentlich nur der Schreibkopf. Das Band ist genau dasselbe, zum Unterschied zu den Lochstreifen. Sie wissen, bei den Lochstreifen habe ich die Spurenzahl (die Verschiedenheit der Spurenzahl), im Medium festgehalten. Ich habe dort 5-Kanal-Streifen, 7-Kanal- oder 8-Kanalstreifen; hier haben wir die Zahl der Spuren in dem Schreibvorgang festgehalten, es ist dasselbe Halbzoll-Magnetband. Dies ist der Unterschied dieser Spuren zu den Spuren bei Lochstreifen.



- Schreiben:**
- 1 Daten zur Ausgabe gehen in den Laufwerkspuffer und werden
 - 2 von dort ins Lese/Schreibregister übertragen
 - 3 + 4 Bei Schreiben auf 7-Spur-Band übersetzen 9 Bit-Code nach 7 Bit-Code
 - 5 Übergabe der Daten in den Schreibausschreiber
 - 6 Schreiben auf Band
 - 7 + 8 Bei 9-Spur-Bändern: Durchgang der Daten zur zyklischen Paritätsprüfung, evtl. Fehlerkorrekturen
 - 9 + 10 + 11 Bei allen Bändern: Übergabe der Daten zur longitudinalen Paritätsprüfung. Das Prüfzeichen wird am Ende des Blocks geschrieben (nach dem zykl. Paritäts-Zeichen)
- Lesen:**
- a Bits von der Bandeinheit gehen durch den Endverstärker
 - b Entschiefen
 - c Übergabe zum Lese/Schreibregister, und von da zu den Paritätsprüfungen
 - d Übergabe direkt in den Laufwerkspuffer
 - e + f Übergabe übersetzt in den Laufwerkspuffer
 - g Daten werden auf den Eingangsbus gegeben

Abb. 40: Lese/Schreibvorgänge

Nun, um auch ältere Bänder auf den modernen Geräten verwenden zu können, besteht die Möglichkeit, 7-Spurbänder zu bearbeiten mit dem 9-Kanal-Laufwerk, und zwar brauchen wir in diesem Fall den Übersetzer. Falls ich eine Codierung habe, die 6 Bits pro Zeichen verwendet, muß ich dann hier übersetzen, falls ich 9 Bits schreiben will oder umgekehrt, falls ich 6 Bits (oder 7 mit Parität) pro Sprosse schreiben will, mit einem Rechner, der einen 8-Bit-Code verwendet, muß ich eben den 8-Bit-Code in einen 6-Bit-Code übersetzen, oder dann eben 7-spurig schreiben, dazu dient dieser Übersetzer. Er wird wieder beim Lesevorgang erscheinen.

Hier haben wir nun das Lesen, es geht in der umgekehrten Richtung vonstatten und dann haben wir ebenfalls das Lesen mit Übersetzen.

Beim Schreiben gehe ich nun von dem Lese-/Schreibregister in den Schreibauslöser und dann wird von dem Schreibauslöser auf Band geschrieben, über die Schreibköpfe.

Der Lesevorgang ist folgender: Wir haben hier die Endverstärker, wo wir vom Band mit den Signalen hereinkommen; dann kommen wir in ein Zwischenregister und dann vom Zwischenregister in das Lese-/Schreibregister. Das ist der generelle Aufbau der Lese- und Schreibvorgänge beim Magnetbandlaufwerk.

Das ist nicht alles, wir haben noch eine Paritätsprüfung, wie Sie wissen. Einmal habe ich hier eine zyklische Paritätsprüfung, dazu habe ich ein Paritätsprüfungsregister für die zyklische Paritätsprüfung, das ist die longitudinale Paritätsprüfung, mit ebenfalls einem Paritätsprüfungsregister. Diese wirkt zurück auf diese Einheit und diese ebenfalls.

Wozu dient das Zwischenregister? Das Zwischenregister dient zum Entschiefen. Falls wir 7-Spur-Bänder bearbeiten, dann haben wir nur die longitudinale Paritätsprüfung, falls wir 9-Spur-Bänder bearbeiten, haben wir beide, die zyklische und die longitudinale Paritätsprüfung. Sie sehen, bei 9-Spur-Bändern haben wir zwei Paritätsarten und zwar deshalb, weil die zyklische Paritätsprüfung benutzt wird, um gelegentliche Fehler auszubessern, sozusagen für die Fehlerkorrektur benutzt wird.

Nun welche Schritte haben wir:

1. Die Daten kommen vom Rechner über den Ausgangsbuss in den Laufwerkspuffer. Von da gehen sie in das
2. Lese-/Schreibregister direkt, oder mit dem
3. Übersetzer in das Lese-/Schreibregister, falls auf 7-Spur-Band geschrieben werden soll von einem 8-Bit-Codierungsrechner und werden dann hier übersetzt in einen 7-Bit-Code. Dann haben wir
4. die Übergabe der Daten. Als
5. werden die Daten in den Schreibausslöser gegeben, d.h., hier wird nun der Schreibvorgang getriggert, und als
6. wird auf Band als der eigentlichen Schreibvorgang geschrieben. Nun, bei 9-Spur-Bändern haben wir folgendes:
Wir haben
7. den Durchgang der Daten vom Lese-/Schreibregister. Sie sehen, das Lese-/Schreibregister ist eigentlich eine Weiche, um die Daten zum Band zu geben und dann gleichzeitig abzuzweigen, die zyklische und die longitudinale Paritätsprüfung durchführen zu können. Bei 9-Spur-Bändern kommen erst die Daten in die zyklische Paritätsprüfung und dann
8. ins zyklische Paritätsprüfungsregister. Bei den 7- und 9-Spur-Bändern haben wir die Daten durchgehend in die longitudinale Paritätsprüfung,
9. in das longitudinale Paritätsprüfungsregister.

10. Werden diese beiden am Ende des Blocks geschrieben.

Fall ich ein 7-Spur-Band habe, werden nur die longitudinalen Paritätsbits geschrieben, andernfalls, bei 9-Spurbändern, werden erst die zyklischen Paritätsbits geschrieben, dann die longitudinalen Paritätsbits. Das ist also der Schreibvorgang.

Beim Lesevorgang ist es gerade umgekehrt, wir haben die Signale von der Bandeinheit.

Die Bits kommen von der Bandeinheit in die Endverstärker, werden hier nochmals verstärkt, damit sie bearbeitet werden können. Dann kommen sie in das Zwischenregister oder in die Entschiefe-Vorrichtung und dann zusammen zum Lese- und Schreibregister. Von da ebenfalls zu den Paritätsprüfungen, einmal zur zyklischen, falls wir ein 9-Spur-Band haben, oder falls wir ein 7-Spur-Band haben allein in das longitudinale, oder auch bei 9-Spur-Bändern in beide Paritätsvorrichtungen und direkt oder übersetzt beim Lesen in den Laufwerkspuffer. Falls ich ein 7-Spur-Band lese, in einem 8-Bit-Maschinencode muß ich mit dem Laufwerkspuffer übersetzen. Dann geht es weiter in den Eingangsbus, in den Rechner des Systems. Das ist der generelle Vorgang beim Lesen und beim Schreiben.

Bei allen Bändern sind zusätzlich zu den Paritätsprüfungen in der longitudinalen Richtung (bei 9-Spur-Bändern longitudinale und zyklische) noch die transversalen Paritätsprüfungen für jedes einzelne Zeichen zu berücksichtigen.

6.6.4 Zeitfolgen

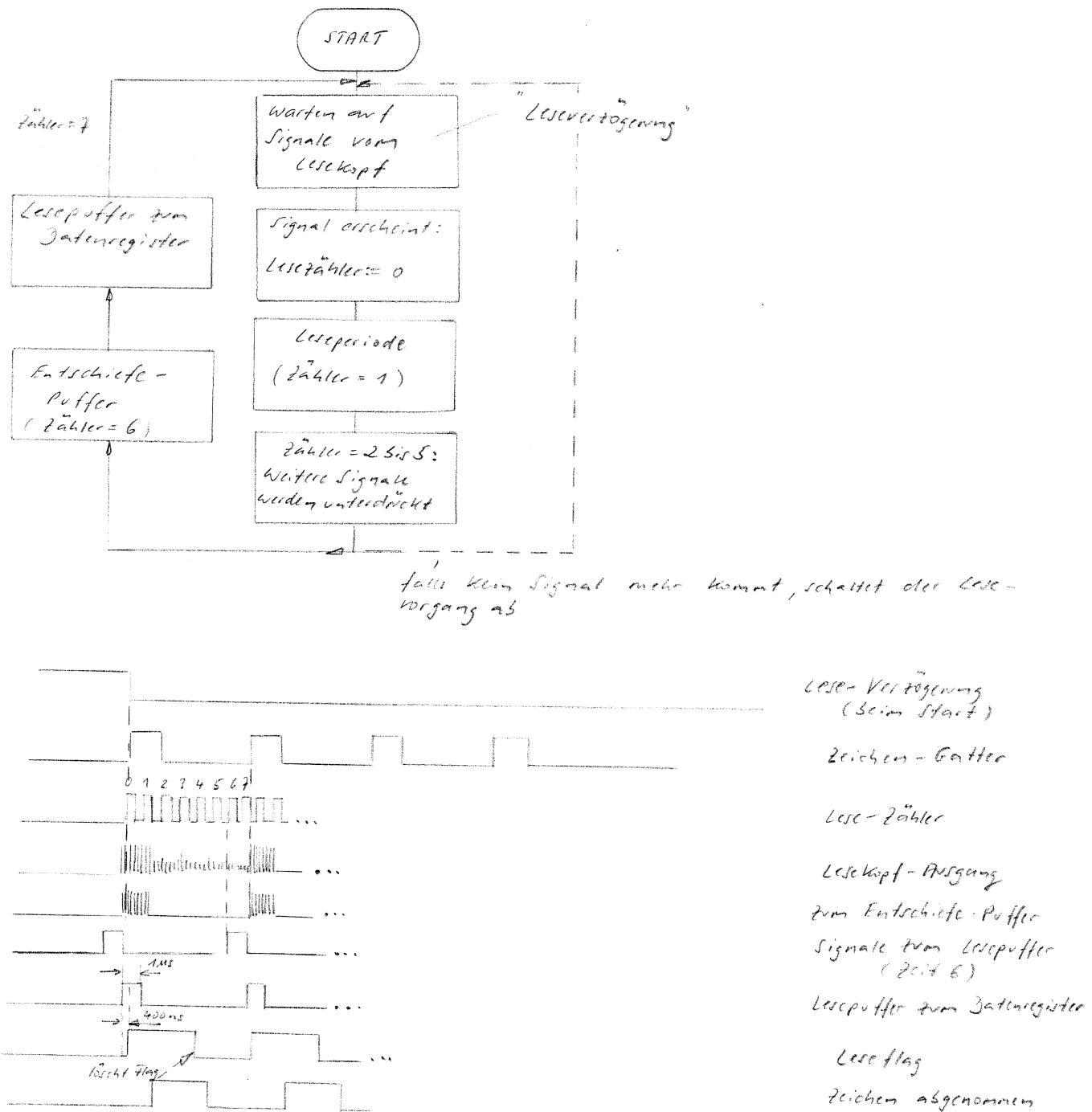


Abb. 41: Zeitfolgen beim Lesevorgang

Wenn wir starten (START), dann müssen wir erst warten auf das Signal vom Lesekopf. Zunächst betrachten wir das Lesen vom Magnetband. Wir starten das Magnetband und müssen dann warten, bis die Signale erscheinen, und zwar existiert eine Leseverzögerung, die darauf beruht, daß das Band beschleunigt bewegt werden muß. Beim ersten Anfang muß das Band beschleunigt werden, dann später während des Lesens von einem Zeichen zum nächsten, muß das Band bewegt werden, deshalb die Leseverzögerung, und zwar zwei Schritte auf einmal, eine sehr lange Stufe beim Anfang und zum anderen eine kurze beim Verschieben zwischen zwei Zeichen.

Als nächstes haben wir die Leseperiode. Diese Leseperiode startet, sobald die ersten Bits erscheinen, d.h., sobald das Signal erscheint. An diesem Punkt wird ein Lesezähler initiiert, d.h., wir setzen ihn auf Null. Während der Leseperiode geht dieser Lesezähler auf 1, dann geht dieser Lesezähler (er ist nichts anderes als ein Zeitgeber) auf 2. Sobald der Lesezähler auf 2 geht, wird die Leseperiode beendet, d.h., alles was jetzt kommt, wird nicht mehr übertragen und nicht mehr gelesen, d.h., weitere Signale werden unterdrückt. Dann geht der Lesezähler auf 3, 4, 5 und nun gehen die erhaltenen Signale in den Entschiefepuffer. Der Entschiefepuffer ist deshalb nötig, weil das Band unter Umständen schief über die Lese- und Schreibköpfe läuft. Deshalb kommen die Bits für ein Zeichen nicht alle gleichzeitig an, sondern sind zum Teil etwas verschoben, und deshalb brauchen wir diesen Entschiefepuffer (Zeit 6). Zur Zeit 7 werden die Daten im Lesepuffer zum Datenregister weitergegeben. Dann fängt der Vorgang wieder von vorne an: Falls wir noch mehr Zeichen lesen wollen, müssen wir warten, bis das Band weitergeschoben ist, dann kommen die nächsten Signale usw. Am Ende eines Blocks, wenn kein Signal mehr erscheint, d.h., wir haben den Zwischenraum zur zyklischen Paritätsprüfung oder beim 7-Spurband die longitudinale Paritätsprüfung, wird dieser Zwischenraum abgefühlt

und die Zeit wird in der Leseverzögerungseinheit fortgesetzt. Sobald diese erscheint, weiß die Einheit, der Block ist zu Ende. Nun kommt die zyklische Paritätsprüfung bei der 9-Spureinheit oder die longitudinale Paritätsprüfung bei 7-Spureinheiten. Betrachten wir nun den Zeitverlauf dieser Vorgänge. Wir haben die Leseverzögerung beim Start. Dies ist ein Rechteckpuls, der heruntergeht, sobald die ersten Zeichen erscheinen. Nun haben wir Zeichengatterpulse, die nun anfangen und darum mitgesteuert werden. Der Lesezähler wird initiiert, sobald die ersten Signale erscheinen, läuft dann bis auf 7 und geht wieder auf Null zurück. Dann kommt wieder die nächste Periode dieses Lesezählers. Sobald der Zwischenraum kommt, schaltet die Leseverzögerung aus, andernfalls geht es dann weiter mit dem nächsten Zeichen. Dann haben wir folgendes: Zur Zeit 0 und 1 wird gelesen, zur Zeit 2 wird angesteuert. Die Zeichengatterpulse steuern nun die Signale aus, die vom Lesekopf kommen, zum Entschiefepuffer gehen nur die Signale, die angesteuert wurden. Nun haben wir eine Pulsfolge, die die Signale zum Lesepuffer gibt, das ist dann zur Zeit 6. Signale werden zum Lesepuffer in den Zeiten 0 bis 5 gegeben, nun zur Zeit 6 wird angesteuert. Zu diesem Zeitpunkt werden die Signale, die vom Entschiefepuffer kommen in einen Zwischenlesepuffer eingegeben, d.h., zu diesem Zeitpunkt werden die Signale vom Entschiefepuffer herausgelesen und weitergegeben. Dann haben wir noch die Signale vom Lesepuffer zum Datenregister. Das Leseflag überlappt nun für 400 Nanosekunden die Signalfolge der Lesepufferübertragung und bleibt dann stehen. Die Ansteigszeit beträgt 600 Nanosekunden. Die Zeichenabnahme aus dem Puffer löscht nun das Flag. Soweit die Signalfolge beim Lesen vom Magnetband. Wie gesagt, dies ist gekoppelt mit der Feststellung des Blockendes: Sobald keine Zeichen mehr erscheinen, wird die Leseverzögerung zurückgestellt und die Magnetbandeinheit weiß, der Block ist jetzt zu Ende. Die nächsten Pulse, die dann kommen, gehören zur zyklischen

Parität oder zur longitudinalen Parität.

6.6.5 Entschiefen

Das Entschiefen der Zeichen ist sehr wichtig bei Schreib- und Lesevorgängen, besonders bei großer Schreibdichte. Denn je größer die Schreibdichte ist, desto näher liegen die Zeichen - d.h., die magnetisierten Flächen auf dem Band - beieinander und desto rascher folgen die Signale aufeinander beim Lesen. Diese Dichte auf dem Band erzeugt Probleme, falls das Band nicht genau waagrecht läuft, sondern ein klein wenig schief. Man kann sich bei 1600 BpZ vorstellen, daß geringe Verschiebungen ausreichen, um Bits nicht mehr geradlinig unter dem Lesekopf erscheinen zu lassen, d.h., wir müssen die Zeichen ausrichten. Es gibt an sich zwei Methoden, um dieses Problem zu umgehen: a) das Band sehr genau parallel zu führen oder b) entschiefen. Wie wir sehen werden, ist das Entschiefen einfach zu bewerkstelligen, siehe Bild 42.

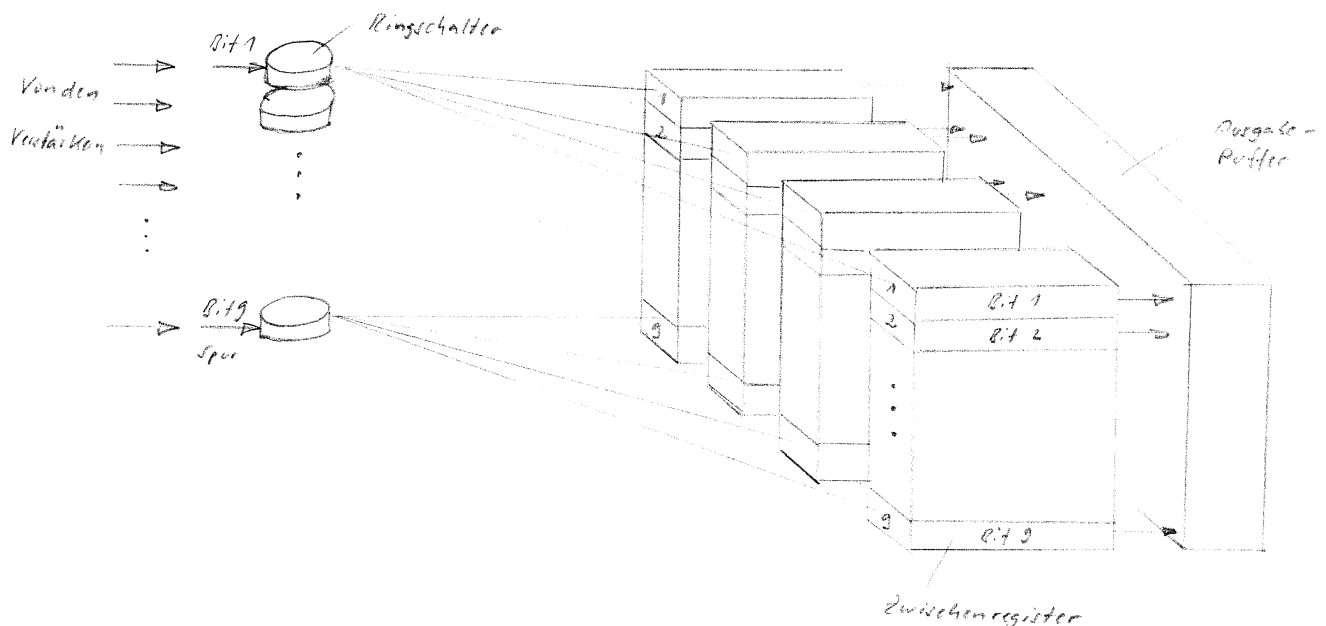


Abb. 42: Entschiefen

Wir haben für jede Spur einen Ringzähler. Hier haben wir den Eingang von den Leseköpfen über die Verstärker.

Falls ich nun lese, werden Bits auf einer Seite zuerst erscheinen, und was ich möchte, ist eine Art stroboskopischer Effekt, ich möchte die zeitliche Folge der Bits in eine räumliche Differenz ausbilden. Dazu dienen diese Zwischenpuffer, insgesamt vier bei dem IBM-Band-Magnetleser. Jeder dieser Puffer hat Platz für 9 Bits. Oben habe ich dann Bit 1, 2, 3, 4 bis 9 und im nächsten ebenfalls, usw. Es existiert eine vierfache Verschachtelung oder Auffächerung dieser Bitfolgen. Diese Ringschalter sind von jeder Spur mit jedem Puffer verbunden. Der erste Ringzähler mit dem ersten Bit, der zweite Ringzähler mit dem zweiten Bit, der letzte Ringzähler mit dem 9. Bit. Dahinter habe ich einen Ausgabepuffer, der die Ausgänge dieser Entschiefepuffer aufheben kann. Wie ist nun die Arbeitsweise dieser Entschiefeanordnung? Die ankommenden Bits werden nun nacheinander durch die Ringzähler in die 4 Zwischenregister eingelesen. Sobald nun eines dieser Zwischenregister voll ist, d.h., alle Bitpositionen belegt sind, wird dieses Zeichen in den Ausgabepuffer ausgelesen, d.h., wir haben nun ein Zeichen erhalten und das benachbarte Register wird dann angesprochen usw. Warum sind gerade 4 Zwischenregister vorhanden, nun diese 4 Zwischenregister reichen eben aus, um eine Schreibdichte von 1600 Bits bearbeiten zu können. Hätten Sie eine Schreibdichte von 3200 Bits pro Zoll, dann müssten wahrscheinlich die Zahl dieser Zwischenregister vergrößern, aber prinzipiell wäre kein Unterschied vorhanden.

6.6.6 Programmierung der Magnetbandlaufwerke

Diese Programmierung ist nun wesentlich komplizierter als bisher und zwar aus folgenden Gründen: Einmal haben wir hier mehr Operationen und zweitens haben wir möglicherweise mehrere Laufwerke an einem System angeschlossen. Welche Operationen kommen in Frage? Wir haben einmal das Lesen vom Magnetband, wir haben das Schreiben auf Magnetband, dabei haben wir einmal Lesen vorwärts und beides auch rück-

wärts, Lesen und Schreiben vorwärts und Lesen und Schreiben rückwärts vom Magnetband. Dann haben wir Rückspulen, dann haben wir Vorwärtsbewegen um einen Block, Vorbewegen um eine Datei, dann haben wir Rückbewegen um einen Block und ebenfalls Rückbewegen um eine Datei und schließlich haben wir das Schreiben eines Datenzeichens. Sie sehen, anstatt der einfachen Arbeitsweise wie wir sie z.B. beim Lochstreifenleser hatten, da kann man eben eine Sprosse lesen oder notfalls ein Wort lesen, haben wir hier 2 Leseoperationen, 2 Schreiboperationen, Rückspulen, 2 Vorbewegoperationen, 2 Rückbewegoperationen und das Schreiben des Magnetzeichens, die hier alle programmierbar sein müssen. Dann zum zweiten mehrere Laufwerke, es ist sehr häufig, daß in einer Installation C 15 meistens eine gerade Zahl, 8, 10, 12, 16, Magnetbandlaufwerke angeschlossen sind, d.h., wir haben folgende Anordnung: Rechner, Speicher, und dann noch eine Kontrolleinheit und daran haben wir eine bestimmte Zahl von Bändern angeschlossen. Und nun ist eben zu bestimmen, welches Band ich bearbeiten möchte, welches Band nun an den Rechner angeschlossen werden soll.

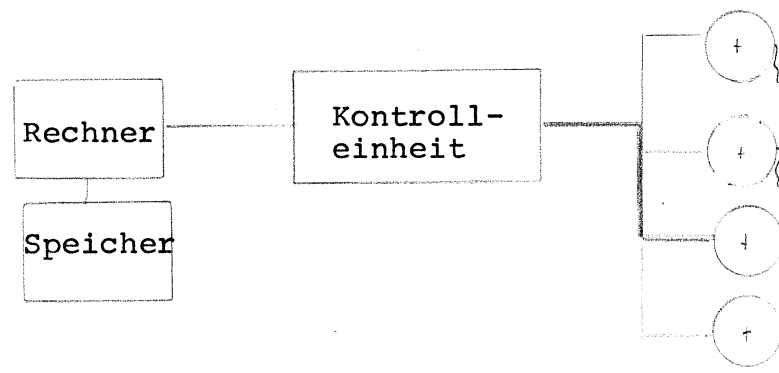


Abb. 43: Anschluß von Magnetbändern

Dafür sind noch zusätzlich zu den bereits genannten Operationen Kontrolloperationen nötig. Solche Kontrolloperationen können z.B. sein MTSE, d.h., "magnetic tape select", "wähle das Magnetband aus", es wird ein bestimm-

tes Magnetband, ein bestimmtes Laufwerk, mit der Kontrolleinheit und mit dem Rechner verbunden. In diesem Fall ohne Kanal- und Unterbrecherwerk z.B., würde ich die Adresse dieses Laufwerks im Akkumulator haben, wenn diese Instruktion gegeben wird. Eine andere Instruktion könnte MTLC sein, d.h., "magnetic tape load control", d.h., hier wird die Arbeitsweise zur gesuchten Operationsrichtung und Operationsart bestimmt. Zu diesen Operationen muß noch das Laden und Lesen des Puffers aufgenommen werden, und zwar des Puffers des angeschlossenen Laufwerks. Mit allen diesen Instruktionen sind Flags verbunden, die auch durch diese Operationen gesetzt und gelöscht werden und es existieren ebenfalls Instruktionen, um diese Flags direkt zu löschen oder zu setzen.

Wie werden diese Instruktionen zusammengesetzt, denn eine Instruktion allein ist und wenig wert, d.h., wir müssen die Reihenfolge der Instruktionen wissen. Das erste was wir tun müssen, ist das Laden des Kontrollworts. Nehmen wir an, daß wir das Kontrollwort aus dem Akkumulator herausnehmen bei der Operation MTSE. Vorher muß das Kontrollwort natürlich in den Akkumulator gebracht werden. Dann kommt als nächstes die Auswahl des Laufwerks mit der entsprechenden Adresse. Wir haben die MTSE-Instruktion, und ein eventuelles Neuladen des Kontrollworts für Operation. An sich können wir das mit dem ersten Schritt schon tun, wir können beides, denn alle Magnetbandlaufwerke haben den Adressenteil und den Operationsteil nebeneinander im Kontrollwort, sodaß eine Ladeeinheit an sich genügt.

Wir müssen das tun, falls wir in der Operationsrichtung weiterarbeiten wollen, dann brauchen wir nicht mehr zurückzugehen auf den ersten Schritt. Als nächstes haben wir die Arbeitskontrolle, mit z.B. einem Befehl MTLC für die Operationsrichtung und Operationsart. Dies sind natürlich Codewörter für diese verschiedenen Möglichkeiten, z.B. 60 für Vorwärtslesen. Nun haben wir wieder diesselbe Erscheinung wie früher, es müssen mechanische

Gebilde bewegt werden, d.h., es braucht Zeit, und wir müssen wieder warten, bis die Operation durchgeführt worden ist, d.h., wir warten auf bestimmte flags, z.B. "Überspringe nächste Instruktion", falls das Flag nicht gesetzt ist oder je nach Fabrikat, falls das flag gesetzt ist.

Als nächstes müssen wir warten, bis die Operation zu Ende ist und Daten zur Verfügung stehen. Beim Lesen vom Magnetband müssen die Daten vor den Puffer geholt werden oder beim Schreiben bereits vom Speicher geholt werden und der nächste Schritt ist dann beim Lesen, daß die Daten nun abgespeichert werden müssen. Beim Schreiben haben wir, daß die Daten in den Puffer gebracht werden. Als letzte Stufe haben wir die Prüfung, ob der Vorgang dann zu Ende ist oder nicht, andernfalls müssen wir zurückspringen für die nächsten Daten, oder möglicherweise für die nächste Operation in derselben Richtung, oder für eine neue Operation beim Laden eines neuen Kontrollwortes. Es existieren Laufwerke mit Varianten dieser Anordnung, z.B. das Holen von Daten vom Puffer kann wie ein select wirken für konstante Arbeit, d.h., wenn wir weiter lesen wollen, dann wird es automatisch das Magnetband weiterbewegen und wir können dann die Daten abnehmen, ohne daß irgendwelche verloren gehen. Für kontinuierliches Arbeiten ist dies eine sehr nette Vereinfachung dieser Instruktionsfolge.

Als nächste Variante kann das "select" nur eine gewisse Zeit, z.B. 30 Mikrosekunden wirken, und während eines "select" kann ich dann mehrere Sprossen lesen. Wenn Sie Daten vom Puffer holen, wirkt es wie ein neues Select und dieses Select kann noch eine gewisse Zeit weiter wirken, während dieser Zeit können Sie noch mehr Daten holen. Oft ist es so, daß Sie das tun müssen, wenn Sie es nicht tun, läuft das Band über den Block hinaus, die Daten gehen verloren. Sie müssen in diesem Fall mehrere dieser Leserpufferoperationen hin-

tereinander schalten. Es gibt eine Magnetbandeinheit, die das hat, da müssen Sie zwei Leseoperationen hintereinander schalten für kontinuierliches Lesen, um innerhalb der Zeit zu bleiben. Bevor Sie anfangen zu lesen, stellen Sie fest, ob Sie eine gerade Zahl oder eine ungerade Zahl von Daten lesen, und je nach dem, fangen Sie mit der zweiten dieser Leseoperationen an oder mit der ersten.

Es gibt noch die Schwierigkeit, daß Sie eine große Zahl von Instruktionen für das Abspeichern der Daten haben. Sie müssen die Folge so kurz wie möglich machen, möglichst nur eine Instruktion bis zwei, für die Abspeicherung der Daten, denn sonst gehen Daten verloren, weil Magnetbandeinheiten unter Umständen sehr schnell arbeiten können.

Es ist noch folgendes zu erwähnen. Die Flags, die hier mit den Instruktionen und den Operationen verbunden sind, sind an das Unterbrecherwerk angeschlossen.

Wenn Sie nun ein Magnetbandlaufwerk haben, das sehr schnell arbeitet, dann kann es sein, daß nicht genügend Zeit zur Verfügung steht, um die Daten über das Unterbrecherwerk im Rechner verarbeiten zu können. Denn wenn eine Unterbrechung kommt, muß diese Unterbrechung untersucht werden. Diese Zeit reicht unter Umständen nicht aus, d.h., wenn Sie eine sehr schnelle Magnetbandeinheit haben und einen relativ langsamen Rechner, kann es sein, daß Sie das Unterbrecherwerk ausschalten müssen, wenn sie mit Magnetband arbeiten.

6.6.7 Fehlererkennung und Korrektur

Das bisher angegebene Programm nimmt an, daß alles ohne Schwierigkeiten verläuft. Dies ist häufig jedoch nicht der Fall. Beispielsweise können folgende Fehlerzustände auftreten: Paritätsfehler, d.h., ein Bit in der Sprosse ist verloren gegangen und die Parität stimmt nicht mehr. Dann kann das Band zu Ende sein, d.h., wir lau-

fen über das Band hinaus, wir wollen vom Band lesen und jetzt stellt die Bandeinheit fest, daß das Band hier zu Ende ist. Weiter haben wir z. B. Fehler, daß das Laufwerk abgeschaltet ist, und dann zuletzt haben wir noch Zeitfehler, d.h., der Vorschub des Bandes ist nicht mehr synchronisiert. Wie können diese Fehlerzustände bearbeitet werden?

Wir haben Zeit zur Verfügung in den Instruktionen; während wir auf das Ankommen der Daten warten, können wir nun diese Fehler abfragen. Diese Fehler sind ebenfalls mit Flags verbunden und deshalb können wir zur Zeit 6 die Flags für diese Zustände abfragen.

Falls ein Fehlerzustand eintritt, müssen entsprechende Aktionen unternommen werden. Bei älteren Magnetbandlaufwerken ist dies meist mit dem Betriebssystem zu tun, d. h., sobald ein Fehler auftritt, springt hier nun das Programm zurück auf das Betriebssystem. Das Betriebssystem untersucht, was für ein Fehler vorlag und geht zurück, um die Sache besser zu machen, falls dies möglich ist. Bei neueren Geräten haben wir bereits eine teilweise automatische Fehlerkorrektur im Laufwerk, z. B. Paritätsfehler. Wenn einmal Paritätsfehler auftreten, können diese schon im Laufwerk bearbeitet werden, z. B. für den Fall, daß Fehler nur in einzelnen Spuren als Paritätsfehler auftreten, können diese automatisch im Laufwerk behandelt werden. Dies ist die Paritätsfehlerbearbeitung in modernen Laufwerken.

Dazu folgende Abbildung:

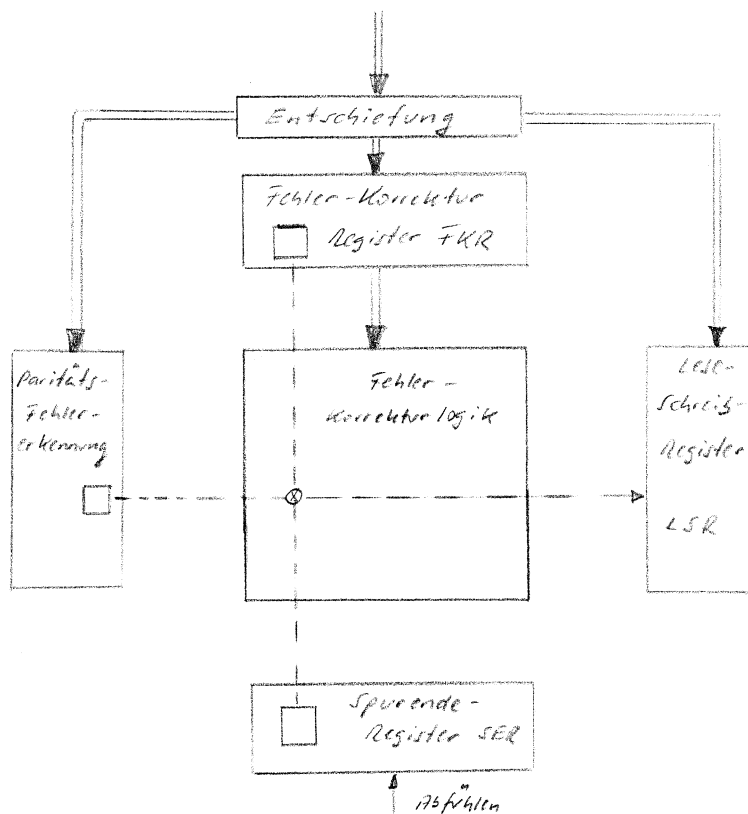


Abb. 44: Paritätsfehler-Korrektur

Wir kommen mit den Daten in den Entschiefepuffer und in die Entschiefereinheit. Vor der Entschiefung kann festgestellt werden, ob alle Bits da sind, d.h., falls ein Paritätsfehler auftritt, wird ein Zeichen nicht vollständig fertig in der Entschiefungseinheit. Im Zwischenregister werden nun plötzlich zwei Zeichen auftreten. Wir gehen von hier aus ins Fehlerkorrekturregister (FKR), hier wird nun festgestellt, welches Bit fehlt. Sagen wir, im Korrekturregister fehlt ein Bit, dann gehen wir von der Entschiefung nun in die Paritätsfehlererkennung und hier wird ebenfalls festgestellt, daß ein Bit fehlt. Von hier gehen wir nun in die Fehlerkorrekturlogik, das ist relativ einfach. Hier wird nun festgestellt, auf welchem Platz dieses Bit fehlte, das ist nun wieder der entsprechende Bit-Platz. Das Spurenderegister wird abgefühlt für die zyklische Paritätsprüfung. Dann gehen wir zum Leseschreibregister. Falls ein Fehler vorlag, gehen wir von der Entschiefung durch die Fehlerkorrekturlogik, falls kein Fehler vorlag, können

wir sofort in das Leseschreibregister übergehen, von da aus weiterarbeiten. Interessant ist nun diese Fehlerkorrekturlogik.

Der Vorgang bewirkt folgendes: Die Bitposition eines schlechten Bits in einer Spur tritt in der entsprechenden Position im Spurenderegister auf. Was versteht man unter einem schlechten Bit? Sehen wir uns dazu folgendes Diagramm an:

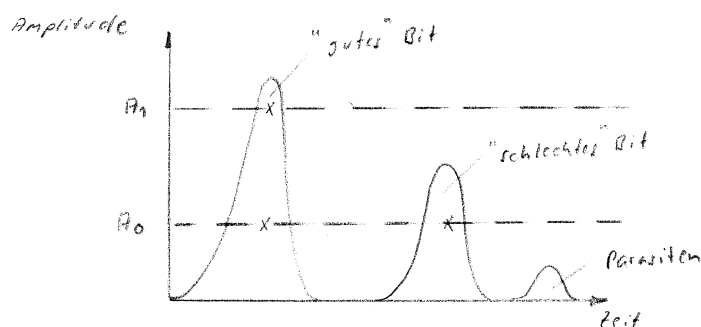


Abb. 45: Bewertung von Bit-Amplituden

Für ein gutes Bit habe ich eine sehr starke Amplitude, für ein schlechtes Bit habe ich eine kleinere Amplitude. Unter einem guten Bit versteht man nun folgendes: Wir haben zwei Positionen in der Amplitude, z. B. A_0 und A_1 . Wenn beide Positionen A_0 und A_1 durch das Signal angesteuert werden, haben wir ein schlechtes Bit. Auf dem Magnetband ist das so, daß die Magnete sehr selten vollständig verschwinden, im allgemeinen wird dieses Signal immer schwächer werden, und zwar deshalb, weil die Bereiche der Magnetisierung immer kleiner werden. Natürlich hängt die induzierte Spannung von der Stärke dieser Magnete ab. Wenn einzelne Magnetfeldchen, die Elementarbereiche dieser Magnete, umklappen, dann wird die Amplitude schwächer. Deshalb kann dann die obere Signalgrenze

herunterfallen und wir haben ein schlechtes Bit.

Für die Korrektur in dem Vorgang wird das schlechte Bit in der Spur unterdrückt durch das Spurenderegister (SER), d.h., es kommt eine Null in die Position des schlechten Bits. Sobald ein Zeichen mit einem schlechten Bit im Fehlerkorrekturregister (FKR) erscheint, wird die Parität des Zeichens geprüft. Wenn nun die Parität nicht stimmt, dann wird ein 1-Bit in die Position des schlechten Bits gesetzt. Dann tritt das Zeichen in das Lese- und Schreibregister ein. Falls die Parität stimmt, dann wird kein 1-Bit in die Position des schlechten Bits gesetzt. Die Fehlerbearbeitung geht während des Bandlesens vor sich. Es tritt nun folgender Fall auf, daß Sie ein Band haben, das nun Fehler zeigt und unter Umständen irgendwelche Bits völlig verschwinden und die Parität nicht stimmt. Falls nun diese Bits völlig verschwinden, dann müssen wir noch die longitudinale und die transversale Paritätsprüfung dazunehmen. Dann können wir feststellen, wo ein Bit völlig verschwunden ist.

Ein Band, das sehr alt und abgearbeitet ist, hat sehr häufig die Eigenschaft, daß Bits verloren gehen. Da hilft nur eins, das Band sofort kopieren und aufarbeiten, damit Sie die Information die in diesem Band festgehalten ist, nicht verlieren können. Wenn Sie die Information kopiert haben, wird das Band gereinigt und geprüft. Nun gibt es eine Grenze der Bearbeitbarkeit. Falls Fehler in mehreren Spuren auftreten, ist es leider nicht mehr möglich, diese automatisch zu beheben. In diesem Falle muß dann das Band aufgearbeitet werden. Falls nur in einem Block der Fehler auftritt, kann er automatisch behoben werden.

Wenn ein Paritätsfehler auftritt, der nicht behoben werden kann, dann wird zurückgesprungen in das Betriebssystem. Das Betriebssystem spult den Block zurück und versucht erneut zu lesen und hofft, daß es nur ein Stückchen Staub war, das nun weggefallen ist, und das Bit nun erscheinen läßt. Bei Mißerfolg tritt derselbe Vorgang erneut auf, das Band alterniert dann um diesen Block. Sobald auf dem Band ein Fehler auftritt, sollten die Daten sofort auf ein neues Band kopiert werden, das alte muß dann geprüft, getestet und aufgearbeitet werden.

6.7 Random- und Direktzugriffseinheiten

6.7.1 Terminologie und Aufbau der Platte

Das Hauptanwendungsgebiet dieser Random- und Direktzugriffseinheiten ist die Zwischenspeicherung großer Datenmengen. Sehr wichtig dabei ist, daß wir relativ geringe Kosten für große Datenmengen bei entsprechender schneller Zugriffsmöglichkeit haben. Erst einmal ein wenig zur Terminologie. Ein Modul ist die größte Untereinheit, d. h., ein Plattenlaufwerk oder ein Trommellaufwerk oder eine Datenzelleneinheit usw. Auf dem Modul sitzt eine Speichereinheit, diese Speichereinheit enthält Flächen. In diesen Flächen sind Datenspuren oder Spuren zur Registrierung von Daten. Zu den Spuren gehört noch ein Indizierungspunkt, der Indizierungspunkt der Spur ist der Anfang und das Ende der Spur, und jede Spur enthält Datenblocks. Wie sehen die Spuren aus? Die Spuren sind konzentrische Kreise, es sind Kreise auf dem Plattenlaufwerk, auf der Datenspur sind die Daten Linienstücke. Wir haben noch den Begriff eines Zylinders bei einer Platteneinheit (siehe Bild 46).

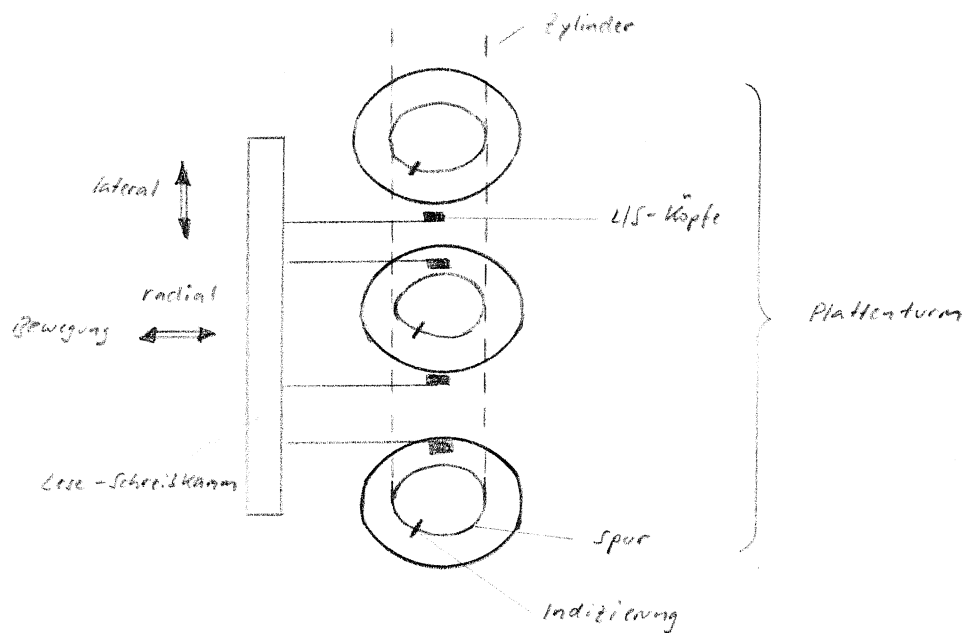


Abb. 46: Aufbau der Platte

Nehmen wir diese 3 Platten, dann haben wir den Lese- und Schreibkamm, d.h., das ist die Einheit woran sich die Lese- und Schreibköpfe befinden. Der Zylinder ist definiert durch alle die Spuren, die ohne eine Positionsänderung des Lese- und Schreibkamms zu erreichen sind. Dieser Kamm kann sich lateral also auf und abbewegen, d.h., im allgemeinen sind die Schreibköpfe, die bewegt werden, lateral angeordnet und werden radial ein und auswärts bewegt. In diesen Spuren haben wir einen Indizierungspunkt, d.h., das ist dort, wo die Spur anfängt oder aufhört. Die Zählung der Spuren geht von außen nach innen, z.B. 0 bis 202, z. B. bei einer IBM-Platte, 200 sind benutzt, und 2 sind Reservespuren. Falls eine Spur durch einen Fehler ausfällt, kann auf eine Reservespur umgeschaltet werden und die defekte Spur dann einfach ignoriert werden. Es geht einige Zeit bis alle 3 Spuren aufgebraucht sind und wenn dann wieder eine Spur ausfällt, dann muß die ganze Platte aufgearbeitet werden.

In der Spur sind die Zeichen registriert, normalerweise zeichenseriell, d.h. 8 oder 9 Bits. Wie sehen nun diese Datenblöcke in der Spur aus?

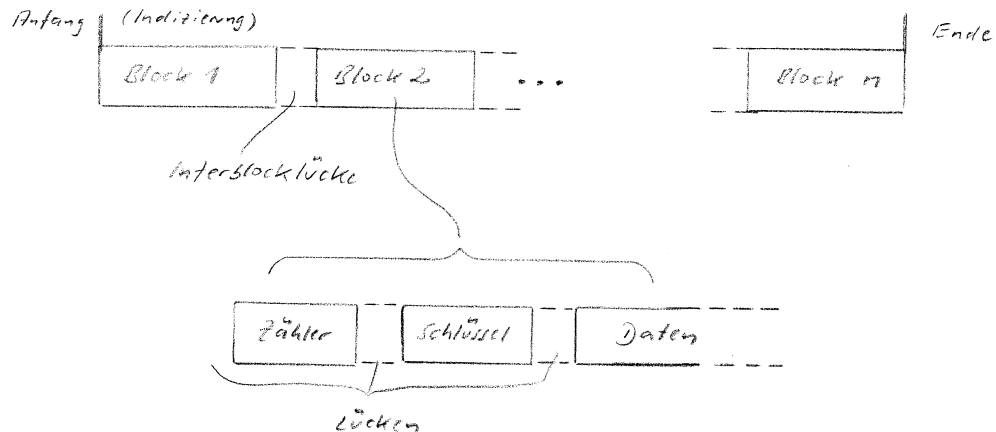


Abb. 47: Aufbau der Datenblöcke

Wir haben hier den Anfang, das ist der Indizierungs-
punkt, dann haben wir einen Datenblock, dann eine In-
terblocklücke, ähnlich wie bei den Magnetbändern, dann
tritt der nächste Datenblock auf, usw. Dann folgt wie-
der die Indizierung, womit die Spur wieder zum Anfang
zurückgeht. Jeder dieser Blöcke hat nun mehrere Unter-
blöcke, ein Zählerblock, dann ein Schlüsselblock und
dann der Datenblock. Manchmal fehlt der Schlüsselblock,
aber im allgemeinen ist er da, wir haben auf jeden
Fall den Zähler- und den Datenblock. Wozu dient der
Zähler? Der Zählerunterblock hat folgende Aufgabe: Er
enthält folgende Informationen: die Identifizierung
und zwar Zylinder-Nummer, Kopfnummer und Blocknummer,
dann die Längenangabe und zwar wie lang der Block ins-
gesamt ist, wie lang der Zählerblock und der Schlüssel-
block ist, wie lang der Datenblock ist und dann noch
Indikatoren. Der Schlüsselunterblock dient zur Daten-

identifizierung, z. B. welchen Code die Daten haben, oder wozu die Daten gebraucht werden usw. Dazwischen drin haben wir Interblocklücken.

6.7.2 Trommelspeicher

Unter den Trommelspeichern unterscheiden wir zwei Klassen, einmal mit feststehenden Lese-/Schreibköpfen, d.h., für jede Lese- und Schreibspur muß dann ein Kopf da sein, oder dann mit fliegenden Köpfen. Dabei kann es sein, daß nur einer für die ganze Trommel da ist. Es gibt Trommeln, wo mehrere Köpfe pro Spur existieren, um die Latenzzeit zu verringern. Hier wird dann der Kopf elektronisch ausgewählt, der am nächsten zu dem gesuchten Bereich der Spur gelegen ist. Am Rande der Trommel befinden sich Zeitgebermarkierungen für die Feststellung der jeweiligen Trommelposition.

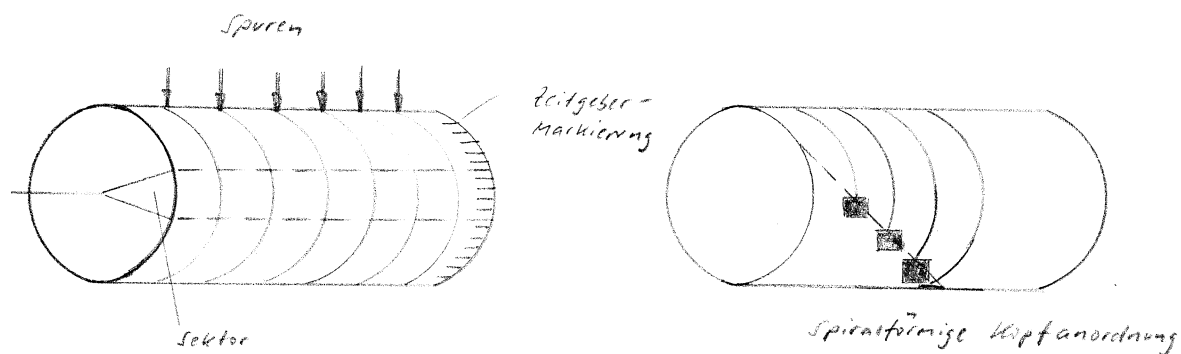


Abb. 48: Aufbau des Trommelspeichers

Nun ist es so, daß die Köpfe bei den existierenden Trommeln normalerweise nebeneinander keinen Platz haben, da die Spuren sehr dicht beieinander sind. Des-

halb müssen die Köpfe spiralg versetzt werden.

6.7.2.1 Charakteristiken

Nun, die Charakteristiken der Trommeln sind folgende: wir unterscheiden einmal kleine Trommeln, dann mittlere und dann große. Die Unterscheidung ist gerichtet nach der Datenkapazität und nach dem Preis, es besteht eine ziemlich eindeutige Korrelation zwischen der Kapazität und dem Preis, je kleiner die Trommel, desto billiger; je größer die Trommel, desto teurer. Einmal haben wir nun hier die Kapazität, die uns interessiert und zwar sind es Millionen Zeichen (Mega), die Zahl der Spuren in einer Trommel, dann die mittlere Zugriffszeit, wobei diese etwa 1/2 Umlaufszeit beträgt. Dies ist eine vorgegebene Konstruktionsgröße, die Umlaufgeschwindigkeit wird vom Konstrukteur bestimmt. Dies wird auch zuweilen als Rotationsverzögerung bezeichnet. Dann haben wir die Übertragungsrate. Sie setzt sich ihrerseits wieder zusammen aus eben dieser Umlaufgeschwindigkeit und aus der Schreibdichte. Dann haben wir den Hersteller und die Miete, monatlich in DM für ein Gerät.

<u>Charakteristiken</u>	<u>kleine</u>		
Kapazität (mio. Zeichen)	.4 - 1,5	<u>.03</u> - .06	1,5 - 6
Anz. der Spuren	32 - 128	32 - 64	256 - 1024
mittl. Zugriffszeit (msec.)	17	17	17
Übertragungsrate (K Zeichen/sec.)	338	32	<u>.19</u>
Hersteller	HP	CDC	Singer/Gen. Prec.
Miete (DM/Monat)	<u>1150</u> -2000	2200-2900	1350-2900

	<u>mittlere</u>				
Kapazität (mio. Zeichen)	2,6 - 7,8	4	4	0,8 - 5,2	4,7
der Spuren	512	200	800	256/128	400
mittl. Zugriffs- zeit (msec.)	<u>26,5</u>	8,75		<u>4,25-17</u>	17
Übertragungs- rate (K Zeichen/ sec.)	111	1200	303,8	75,360, <u>1440</u>	370
Hersteller	H	IBM	U		H
Miete (DM/Monat)	2800-6700	6100-7000	5700- 10000		9500

	<u>große</u>	
Kapazität (Mio. Zeichen)	12,6	6,6
der Spuren	256	1600
mittl. Zugriffs- zeit (msec.)	17	8,6
Übertragungsrate (K Zeichen/sec.)	1440	333
Hersteller	U	RCA
Miete (DM/Monat)	14000	18500

Zugriffszeit = 1/2 Durchlaufszeit (auch als Rotations-Verzögerungszeit bezeichnet)

Tabelle 6.7.2.1: Trommelcharakteristiken

Mit der kleinen Trommel fangen wir an, da ist eine mit 0,03 bis 0,06 Mega Zeichen, die Zahl der Spuren ist 32, oder 64 bei der großen. Die mittlere Zugriffszeit ist 17 Millisekunden, und Sie werden sehen, 17 Millisekunden hat sich sehr eingebürgert bei den umlaufenden Datenspeichergeräten. Es ist fast ein Standardwert, d.h., die Umlaufzeit der Trommel beträgt 34 Millisekunden, es wird auch bei den Platten verwandt. Die Übertragungsrate dieser Trommel ist 32 Kilozeichen pro Sekunde. Hersteller ist die Control Data Corporation, und die Miete beträgt DM 2.200 bis 2.900 pro Monat. Sie sehen, die Trommel ist nicht billig.

Es existiert eine zweite kleine, die ich ausgewählt habe mit ,4 bis 1,5 M Zeichen, jeweils in diskreten Schritten mit 32 bis 128 Spuren. Die mittlere Zugriffszeit beträgt 17 Millisekunden, die Übertragungsrate ist hier 338 K Zeichen pro Sekunde, der Hersteller ist Hewlett Packard. Die Miete beträgt 1.150 bis 2.000 DM pro Monat. Dies waren zwei relativ kleine Trommeln. Sie werden sehen, es existiert kein großer Vorteil, eine kleine Trommel zu wählen, da die Preise für die mittelgroßen Trommeln in der gleichen Größenordnung liegen. Deshalb wäre es nicht gut, wenn jemand eine kleine Trommel wegen des Preises wählen würde, denn bei der mittleren Größe haben wir eine größere Kapazität für etwa denselben Preis. Da habe ich 1,5 bis 6 Mega Zeichen, wiederum in diskreten Schritten mit 256 bis 1.024 Spuren, wiederum 17 Millisekunden Zugriffszeit. Die Übertragungsrate ist hier mit .19 KZ/sec. sehr klein. Der Hersteller ist Singer/General Precision. Der Mietpreis ist 1350 bis 2900 DM pro Monat. Sie sehen die Preise sind hier nicht wesentlich von den Kleintrommeln verschieden. Eines muß man herausstellen, die Übertragungsrate ist sehr gering hier bei dieser Trommel, daher der billige Preis. Die nächsten Größen

sind 2,6 bis 7,8 M Zeichen. Wir haben 512 Spuren, und zwar für beide. Die Zahl der Spuren ist diesselbe, d.h., der Unterschied in der Kapazität muß durch die Schreibdichte hervorgerufen werden. Die mittlere Zugriffszeit ist 26,5 Millisekunden. Hier haben wir 111 Kilo Zeichen, der Hersteller ist Honeywell, der Preis DM 2.800 bis 6.700 pro Monat. Sie sehen, wir haben hier für DM 2.800 eine Trommel, die das 40-fache der Trommel von Control Data als Kapazität hat und dabei noch etwas billiger ist. Das sind Unterschiede, die man beachten muß.

Nun zu den 4 Mio Zeichen-Trommeln. Wir haben 200 bis 800 Spuren, 8,79 sec. mittlere Zugriffszeit und eine Übertragungsrate von 1.200 oder 303,8 Kilo Zeichen pro Sekunde, der Hersteller ist IBM und die Type ist 2303. Diese existiert auch als vereinfachte Modifikation, als 2301 mit derselben Kapazität, aber mit kleinerer Anzahl von Spuren und dann eben einer grösseren Übertragungsrate, da die Dichte bedeutend grösser ist. Der Preis liegt bei DM 7.000 pro Monat.

Dann haben wir eine mit 0,8 bis 5,2 M Zeichen; die hat 256 plus 128 Spuren und hat eine Zugriffszeit von 4,25 bis 17 msec. Die Übertragungsrate variiert hier mit der Größe der Trommel: 75, 360 oder 1400 K Zeichen/sec. Der Hersteller ist Univac, es gibt die Flying Head Drum 330 oder die Flying Head 880, mit der großen Kapazität, und der Preis hier ist 5,700 bis 10.000 DM pro Monat. Dann haben wir noch zwei Trommeln mit großer Kapazität: 12,6 Millionen Zeichen und 256 Spuren, 17 msec. Zugriffszeit, 1440 K Zeichen/sec. als Übertragungsrate von Univac, das ist die Fixed-Head-Drum FH 1782, mit DM 14.000 DM im Monat Miete. Schließlich haben wir noch eine mit 6,6 Mill. Zeichen, 1600 Spuren, 8,6 msec., und Übertragungsrate 333 K Zeichen/sec., die RCA/Univac 8560. Der Preis liegt bei 18.500 DM pro Monat.

Alle Trommeln sind teuer. Sie sehen, die billigste, die wir hier haben, hat schon über Tausend Mark Monatsmiete.

6.7.2.2 Organisation und Arbeitsweise

Ich möchte hier auf die die IBM 230B eingehen und zwar deshalb, weil sie sehr weit verbreitet ist, wohl die meistgebrauchte Trommel. Diese Trommel hat 800 Spuren plus 80 Hilfsspuren mit je 4.892 Zeichen. Der Sinn der Hilfsspuren ist der, daß gelegentlich eine der Spuren schlecht wird, d.h., die magnetisierbare Oberfläche verliert diese Eigenschaft, und dann würde die Trommel an ihrer Kapazität verlieren. Wenn nun Hilfsspuren da sind, können diese anstelle der Originalspur als Ersatz verwendet werden. Normalerweise werden sie nicht verwendet. Die nächste Frage ist die Adressierung. Die Trommelfläche ist in 4 Sek-

toren geteilt, siehe Abbildung 49.

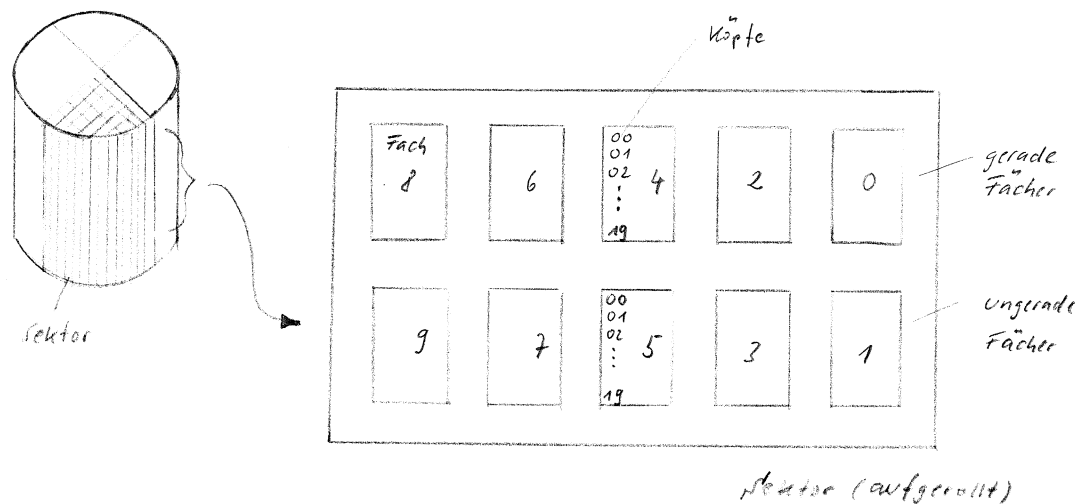


Abb. 49: Organisation der Trommel IBM 2303

Jeder Sektor hat 10 Fächer (racks), also insgesamt 40 Fächer. Die Zählung ist so, daß oben die Fächer mit gerader Nummer, unten die mit ungerader Nummer sind. Für jedes Fach existiert ein Satz Köpfe 00, 01, 02 ... 19 (feste Köpfe) plus zwei Hilfsköpfen. Die Hilfsköpfe dienen demselben Zweck wie die Hilfspuren. Falls einer der Köpfe ausfällt, kann einer der Hilfsköpfe verwendet werden und natürlich entsprechend die Spur. D.h., falls eine Schwierigkeit auftritt, falls eine Spur schlecht wird, kann elektronisch umgeschaltet werden. Erstmal kann elektronisch umgeschaltet werden, und dann kann bei der nächsten Wartung nachgeprüft werden, was mit der alten Spur los ist. Das Interessante ist nun der Kopf. Die Adressierung bezieht sich auf die Köpfe. Die Kopfadresse ist bestimmt durch: a) die Nummer des Sektors, b) die Nummer des Fachs und dann die Stellung des Kopfes im Fach, d) die Nummer des Kopfes selber.

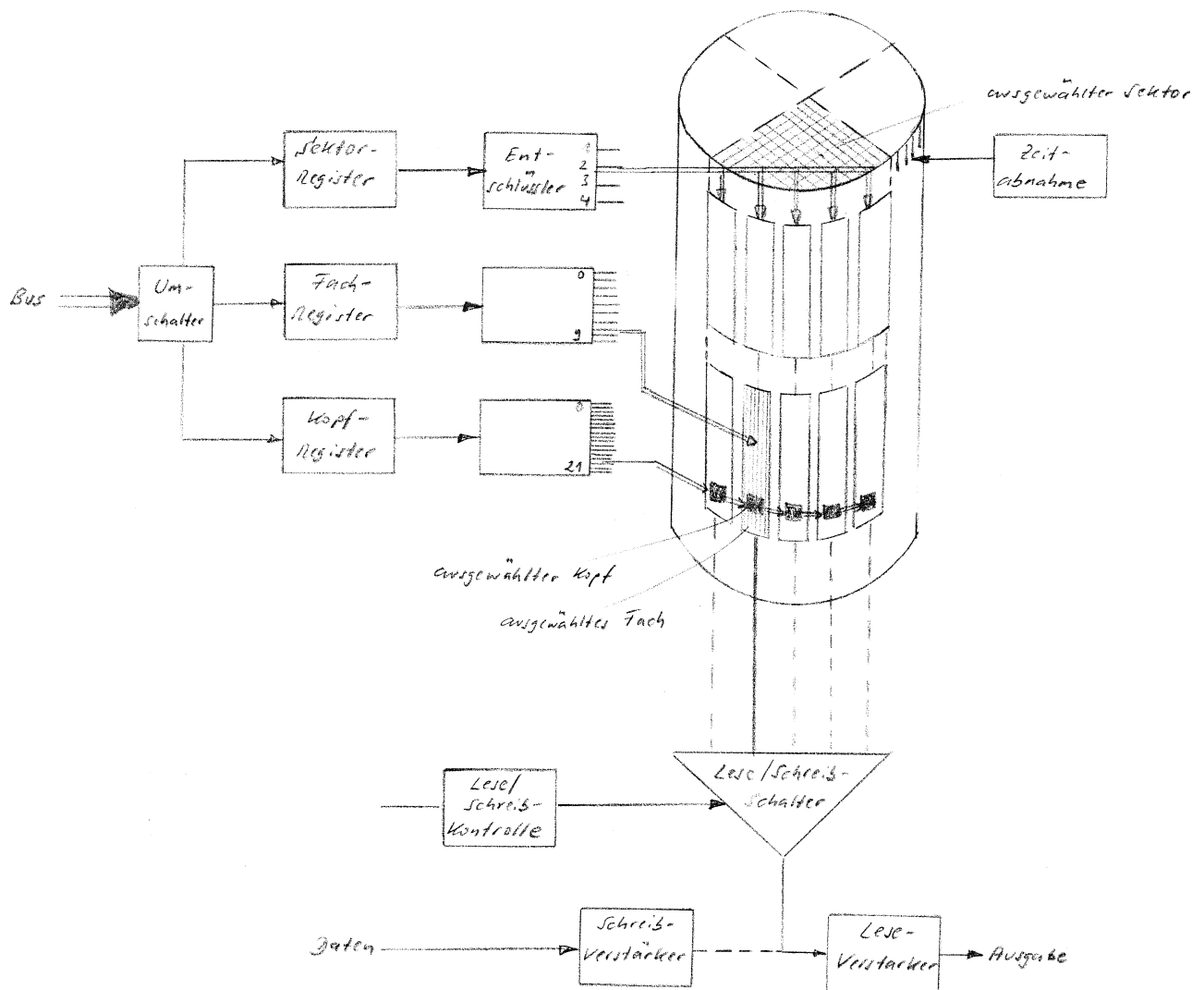


Abb. 50: Kopfadressierung

Wenn ich mit dieser Trommel arbeiten möchte, muß ich mit dem Datenbus hereinkommen. Der Datenbus geht in einen Schalter, einen Umschalter, und zwar wird hier nach den entsprechenden Registern umgeschaltet, einmal für das Sektorregister, dann für das Fachregister und schließlich für das Kopfregister. Diese Register halten die Adressen dieser Einheiten. Dann kommt eine Entschlüsselung der Adressen, dann haben wir hier 4 Ausgänge für die 4 Sektoren. Und nun haben wir das Fachregister, das hat 10 Ausgänge, eines davon zeigt nun auf das Fach, das uns gerade interessiert. Dann haben wir das Kopfregister, dafür haben wir 22 Ausgän-

ge, 20 reguläre und 2 zum Ersatz. Die Adresse des Kopfregisters gibt nun an, welche Spur ausgewählt wird. Die Verbindung zum Kopf geht in den Lese- und Schreibschalter ein (er wird von der Lese-/Schreibkontrolle betätigt, je nach dem ob gelesen oder geschrieben werden soll). Und dahinein habe ich den Ausgang zum Leseverstärker und die Ausgabe zum Rechner, Ausgabe der Daten wieder zum Datenbus. Andererseits habe ich den Eingang zum Lese-/Schreibschalter vom Schreibverstärker. Hier kommen die Daten herein, die geschrieben werden sollen. Das ist die generelle Anordnung, dieser 2303 IBM Trommel.

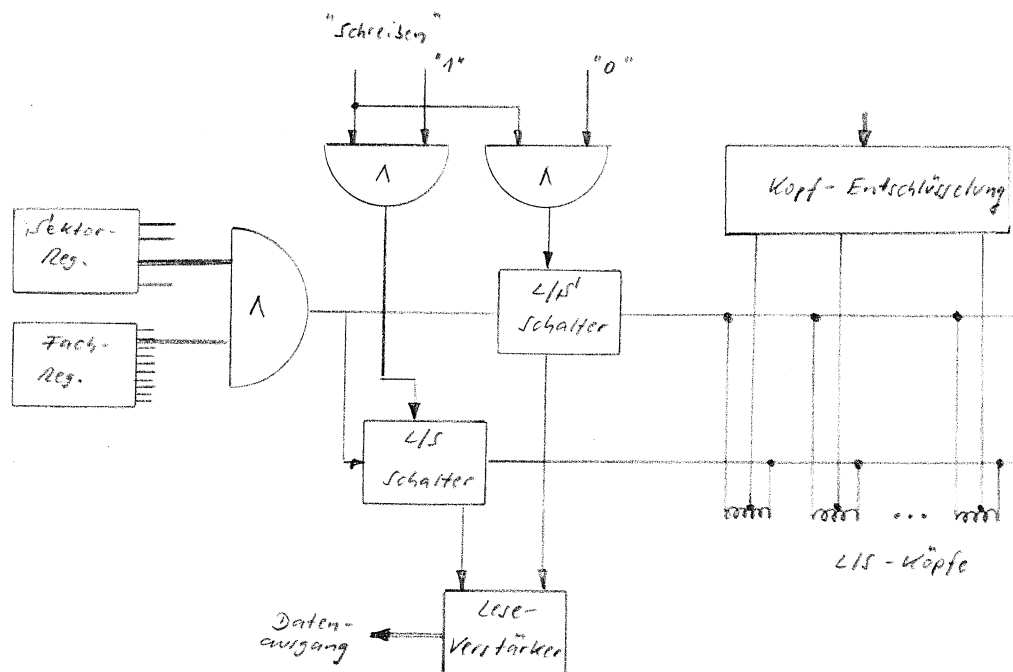


Abb 51: Lese-/Schreibschaltung

Noch ein wenig zu dem Lese-/Schreibschalter. Wir haben hier das Sektorregister. Und das Fachregister. Sie sehen, wir sind verbunden mit dem Lese-/Schreibschalter, was in der vorhergehenden Zeichnung nicht berücksichtigt wurde. Dann sind 4 Ausgänge da, davon wird

einer ausgewählt und diese gehen in eine Und-Schaltung. Von der Und-Schaltung habe ich einen Ausgang zu einem Flip-Flop, dem Lese-/Schreibschalter. Oben haben wir eine andere Und-Schaltung und zwar mit der Indizierung für Schreiben. Wenn ich eine 1 schreiben will, haben wir den Eingang für die "1" und wenn ich eine Null schreiben will, haben wir den Eingang für eine "0". Von hier komme ich herein in den zweiten Lese-/Schreibschalter. Dann haben wir den Ausgang in die Lese-/Schreibverstärker. Wie geht es nun weiter mit den Lese-/Schreibköpfen? Wir haben die Kopfschlüsselung, mit 22 Ausgängen. Davon ist einer ausgewählt. Die Verbindung geht über die Lese-/Schreibspule und zurück zum Lese-/Schreibschalter. Wir haben eine Schaltung zum Schreiben und zum Lesen und zwar über die Umkehrung der Polung, wie in dem früheren Bilde dargestellt. Wie geht nun die Registrierung auf der Trommel vor sich? Mit einem der früher betrachteten Verfahren wird die Information auf der Trommel festgehalten. Die nächste Frage ist, wie sind die Daten auf der Trommel organisiert? Wir haben dazu Datenspuren und Zeitspuren auf der Trommel: Einmal eine Spur, der Hauptzeitimpuls. Mit diesen Zeitspuren wirken nun Pulsgeber, für den Hauptzeitpuls, auch Indexpuls genannt. Dann habe ich die Wörterzeitpulse mit dem 0.Puls, dem 1. Puls, dem 2. Puls usw., dann kommen noch die Bitzeitpulse, und zwar sind diese etwas versetzt zu den Wörterzeitpulsen. Weiter haben wir entsprechend die Datenbits in den Datenspuren. Betrachten wir z.B. die Bahn mit den Bits 10111011 usw., falls ich bitseriell schreiben will. Wenn ich bitparallel oder zeichenseriell schreibe, dann betrachten wir die Spuren x_0 , x_1 , usw.

Abb. 52 siehe Seite 164.

Das sind die verschiedenen Möglichkeiten. Nehmen wir nun an, wir arbeiten bitseriell. Falls wir zeichenseriell arbeiten, kompliziert sich die Sache nicht wesentlich.

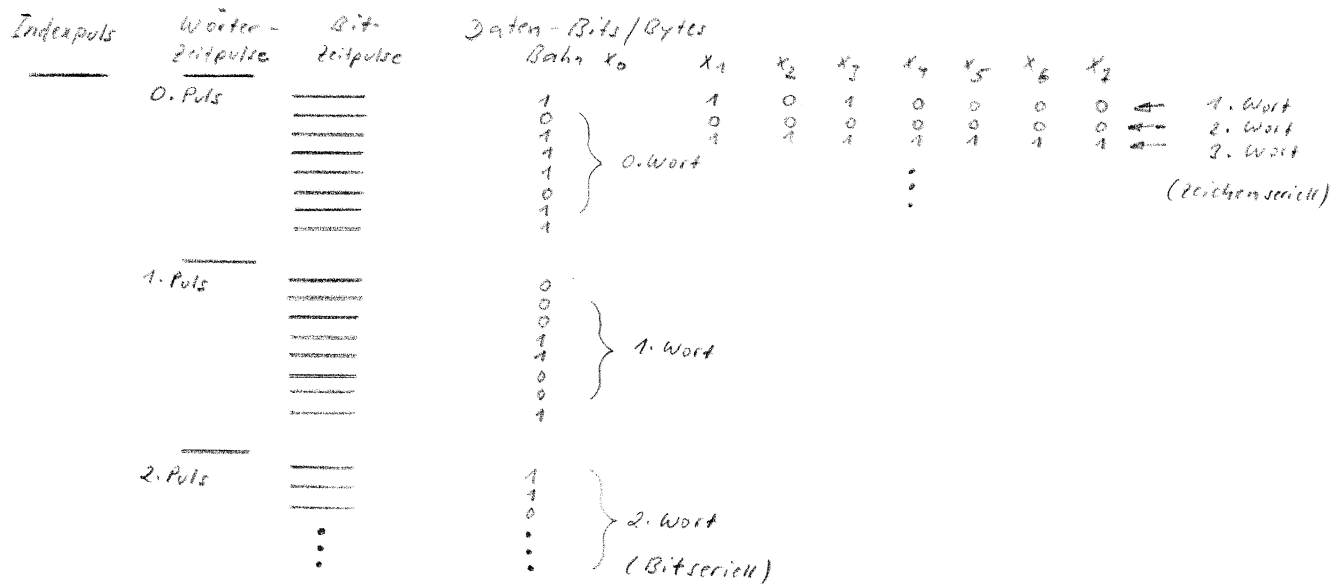


Abb. 52: Daten- und Zeitspuren auf der Trommel

In der nächsten Schaltung sehen wir, wie der Lesevorgang vor sich geht.

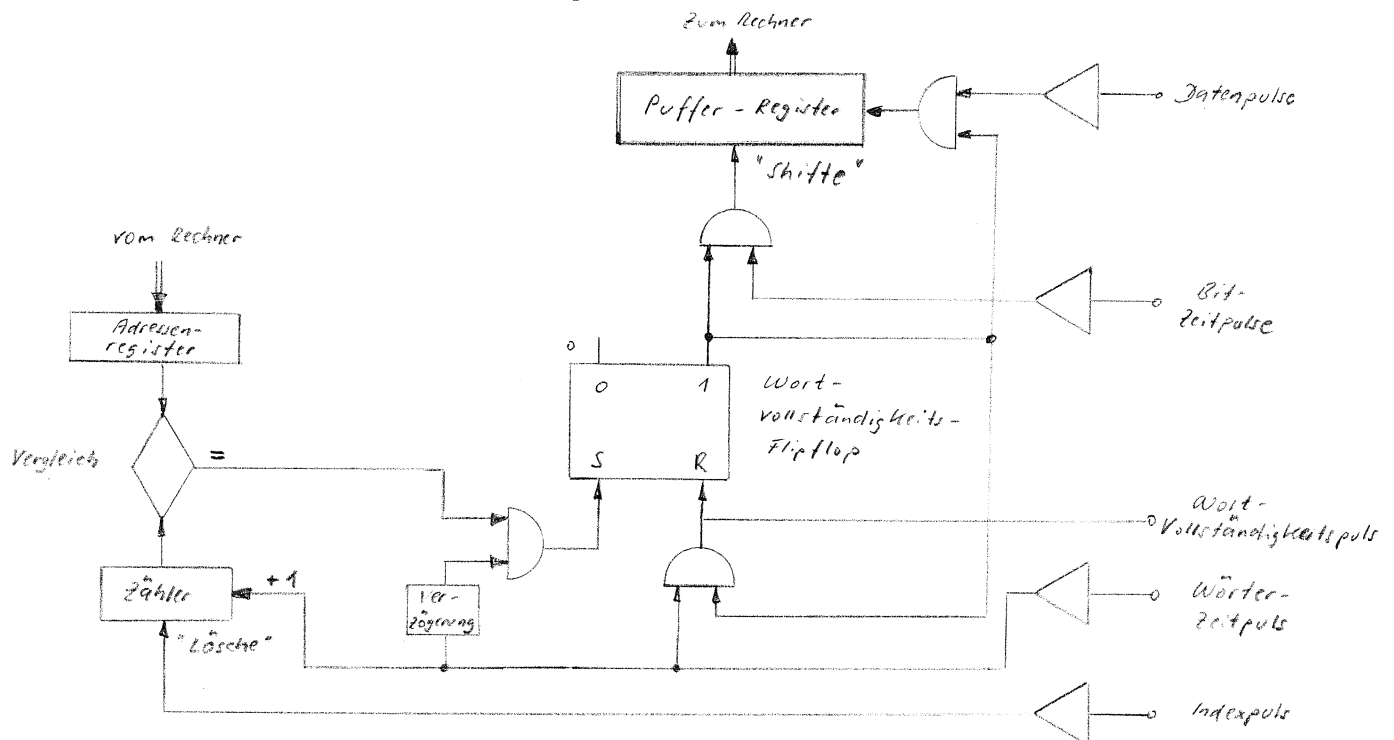


Abb. 53: Lesevorgang

Die linke Seite zeigt die Eingänge der Pulse, z.B. die Datenpulse. Die Datenpulse gehen in eine Und-Schaltung mit einem zusätzlichen Eingang herein. Von da aus gehen die Daten in ein Pufferegister, d.h., ein Datenpuls der hier ankommt, kommt in den Pufferspeicher und muß dann verschoben werden mit dem nächsten. Die Bitzeitpulse gehen ebenfalls in eine Und-Schaltung, damit verschieben wir den Inhalt des Pufferspeicherregisters um ein Bit nach links. Der nächste Eingang ist nun der Wort-Vollständigkeitspuls, ein Vollständigkeitspuls, der entweder vorgegeben ist, oder aus den Bitzeitpulsen mit einem einfachen Ringzähler abgeleitet werden kann (sobald der Zähler überläuft, schaltet er den Wortvollständigkeitspuls). Der Wortzeitpuls schaltet ein Flip-Flop. Wenn der Wortvollständigkeitspuls kommt, wird das FF zurückgesetzt oder in anderen Worten, von diesem Moment an wird nichts mehr weitergeschoben. Nun kann ich die Daten zum Rechner geben. Der Wörterzeitpuls schaltet ebenfalls dieses Flip-Flop. Weiterhin geht er heraus in einen Zähler und addiert + 1 pro Wörterzeitpuls. Der letzte Eingang ist der Hauptzeitpuls oder Indexpuls. Dieser Puls hat die Aufgabe, den Zähler zu löschen. Wozu brauchen wir die Anordnung auf der linken Seite? Diese Anordnung dient dazu, das richtige Wort auf der Trommel auszuwählen. Hier habe ich nun einen Vergleicher und dieser Vergleicher benutzt einmal den Zähler und dann benützt er auch das Wort-Adressenregister. Die Wortadresse wird von der Kontrolleinheit der Trommel geladen. Und nun müssen wir versuchen, die richtige Adresse auf der Trommel zu erwischen. Falls nun der Inhalt des Wortadressregisters und des Zählers derselbe ist, dann wird hier über eine Und-Schaltung das Flip-Flop gesetzt. Die Verzögerung wird zum Ausgleich der Zeitdauer des Vergleichs benötigt. Der gesamte Vorgang ist folgender: Zuerst wird das Wortadressregister geladen, mit der Adres-

se des Worts, das gelesen werden soll. Die Trommel dreht sich nun weiter. Sobald der Hauptzeitpuls kommt, wird der Zähler gelöscht, und nun vergleicht der Vergleicher den Inhalt des Zählers mit dem Wort-Adressregister. Falls der Inhalt übereinstimmt, wird das Flip-Flop gesetzt, d.h., das Pufferregister fängt an zu verschieben, und die Daten werden darin aufgenommen, sobald sie vom Lesekopf her erscheinen.

Falls dies Wort noch nicht das Richtige war, wird mit jedem Wortzeitpuls eine 1 zu dem Zähler hinzugezählt, bis eben das richtige Wort unter dem Lesekopf erscheint und dann wird gelesen. Wenn ein Wort gelesen ist, durch den Vollständigkeitspuls angezeigt, wird das Flip-Flop ausgeschaltet und die ganze Operation ist beendet, d.h., der Inhalt des Pufferspeichers kann abgespeichert werden. Es ist noch folgendes zu beachten: Diese elektronischen Vorgänge, die hier ablaufen, sind natürlich wesentlich schneller als die Umdrehungsgeschwindigkeit der Trommel, deshalb entsteht kein Problem mit dem Vergleicher. Für die Elektronik werden ein paar Mikrosekunden benötigt, wogegen die Umlaufzeit eben 17 Millisekunden ist.

6.7.3 Platte

6.7.3.1 Aufbau und Wirkungsweise

Auch hier müssen wir wieder zwischen 2 Typen unterscheiden, und zwar festeingebaute Platten und auswechselbare oder austauschbare Platten, sog. Plattentürme, oder Plattenstapel. Z.B. Plattentürme IBM 2311. Feste Platten z.B. hat die Honeywell 262. Normalerweise haben wir folgende Eigenschaften, daß die Platteneinheiten mit festen Platten sehr große Kapazitäten haben. Die auswechselbaren Platteneinheiten haben

nur kleinere Kapazitäten und zwar deshalb, weil eben die Masse der zu bewegenden Platten bei großen Kapazitäten sehr groß würde. Der Vorteil der auswechselbaren Platten ist natürlich, daß sie herausgenommen werden können. Sie können gelagert werden, falls nicht benötigt, d.h., wir haben hier eine Austauschbarkeit und Lagerung in den Archiven, analog zu den Magnetbändern mit dem einen Unterschied, daß die Platten im Preis höher sind, und zwar wesentlich höher, etwa 10 bis 20 mal teurer als ein Magnetband. Neuerdings ist die mittlere Zugriffszeit zu den Daten auf der Platte wesentlich geringer, so daß sich heutzutage sehr stark die Platteneinheiten eingebürgert haben und anfangen, die Magnetbandeinheiten zu verdrängen, wie bereits bei den Magnetbandeinheiten erwähnt. Ein Plattenturm hat z.B. 6 Platten mit 10 brauchbaren Oberflächen, d.h., die oberste und die unterste Fläche des Plattenturmes werden generell nicht benutzt. Die oberste Fläche wird nicht benutzt, weil sich da meist Staub ansammelt und der Staub ein tödlicher Feind der Platteneinheiten und der Arbeitsweise ist. Und die untere Platte wird für andere Zwecke noch gebraucht. Der generelle Aufbau ist im Bild 54 dargestellt.

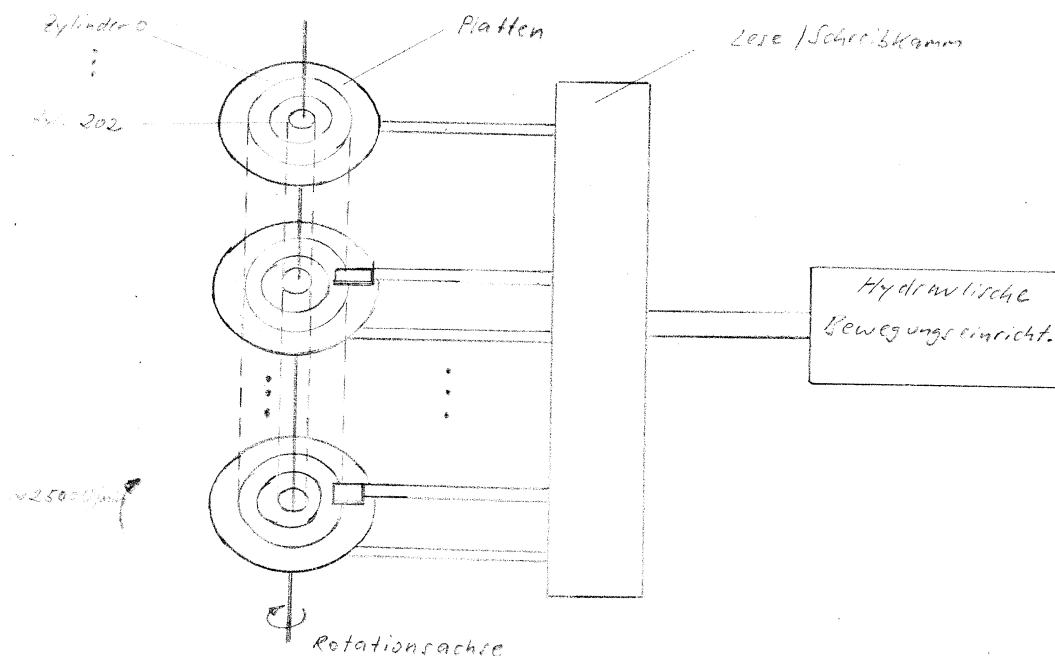


Abb. 54: Aufbau einer Platteneinheit

Wir haben die Rotationsachse, um die sich dann dieser Aufbau dreht. Dann haben wir die Datenspuren. Dies sind konzentrische Kreise auf der Platte, und zwar bei der 2301 Platte ist die äußere Datenspur bzw. der Zylinder 0 und die innerste ist Zylinder 202. Ein Zylinder ist die Menge aller Datenspuren, die ohne Vor- und Rückwärtsbewegung des Lese-Schreibkamms erreicht oder abgegriffen werden können. Lese-Schreibköpfe sind jeweils paarweise für die Unterseite der oberen Platte und für die Oberseite der nächsten Platte angeordnet. Alle zusammen gehen dann hier in den Lese-Schreibkamm, der vor- und rückwärts bewegt werden kann, um die anderen Spuren erreichen zu können. Der Zylinder hat die Bedeutung, daß auf einer Spur sofort gelesen werden kann, wenn die gesuchten Daten in dieser Spur oder in diesem Zylinder sind, ohne daß die ganze Kammvorrichtung vorwärts oder zurück bewegt werden muß. D.h., alles was getan werden muß, ist, den Lese-Schreibkopf zu aktivieren, was durch die Pfeile gekennzeichnet ist. Wir vergleichen nun diese Anordnung mit der Anordnung der Trommel, wir haben nicht nur die zeitliche Latenzzeit, sondern wir haben zusätzlich im allgemeinen Fall die Zugriffszeit durch die Bewegung des Schreibkamms dieser Kopfvorrichtung. Diese Bewegung geschieht besonders bei der IBM 2311, durch eine hydraulische Vorrichtung und ist natürlich relativ langsam, d.h., braucht etwas Zeit, bis die Drücke sich ausbilden. Deshalb kommt dem Begriff des Zylinders gewisse Bedeutung zu. Nach dem Laden eines neuen Plattenturms, z.B. IBM 2311, werden die Lese-Schreibköpfe erst dann eingefahren, wenn die Platten auf der vorgeschriebenen Geschwindigkeit sind, und zwar deshalb, weil sich dann ein Luftfilm auf der Plattenfläche ausbildet und der Kopf dann direkt auf dem Luftfilm aufsitzt. Der einzige Zwischenraum zwischen der Platte und dem Kopf ist eine Luftfilm von etwa 4 Mikrometer Dicke. Sie sehen also, die Köpfe sitzen

dicht auf der Platte auf, und falls die Bewegung sich verlangsamt, verschwindet dieser Luftfilm, der eben durch die Bewegung aufrecht erhalten wird (Bernoulli-Prinzip). Sobald die Bewegung nachläßt, verschwindet dieser Luftfilm, d.h. dann, daß der Kopf, der Lese-Schreibkopf auf der Platte aufsitzt ("abstürzt") und dann die Platte zerkratzt. Sobald die Rotationsgeschwindigkeit auf eine gewisse Grenzggeschwindigkeit absinkt, dann muß der Lese-Schreibkamm sofort entfernt werden, um die Zerstörung des Plattenturms zu vermeiden. Dies geht sehr einfach, durch gewisse Kontrolleinheiten. Doch ist es sehr wichtig, daß die exakte Geschwindigkeit des Turms festgestellt wird. Dies geschieht dadurch, daß auf der untersten Seite eine gekerbte Platte angebracht ist. Diese Platte hat die sog. Sektorenschlitze, zusätzlich noch einen Indexschlitz. Dieser Indexschlitz zeigt den Beginn der Spuren an. Diese Schlitze werden mittels Induktion abgefühlt und verstärkt. Wir haben die Pulsfolge erzeugt durch die Sektorenschlitze und dann haben wir zusätzlich einen Impuls vom Indexschlitz.

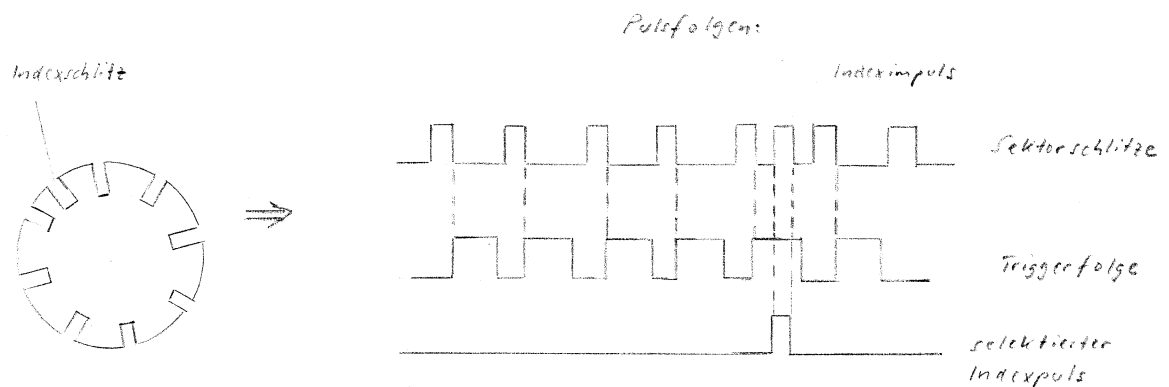


Abb. 55: Sektorenschlitz-Platte

Dies ist die Pulsfolge durch die Sektoren- und Indexschlitze. Um den Anfang der Spuren des Plattenturms festzustellen, brauche ich nur eine Trigger-Pulsfolge

überlagern, gesteuert wird jeweils durch den Abfall der Pulse von den Sektorenschlitten. Wenn ich nun diese beiden überlagere, bleibt nur der Puls als Indexpuls übrig.

6.7.3.2 Charakteristiken

Tabelle 6.7.3.2 siehe Seite 170 a.

Wie vorher hatten wir auch hier kleine, mittlere und große Einheiten. Wiederum haben wir hier die Kapazität als wichtige Größe in Millionen Zeichen (Mega-Zeichen). Dann haben wir die Gesamtzahl der Spuren, dann die Zugriffszeit in Millisekunden und die Rotationszeit, ebenfalls in Millisekunden. Die Rotationszeit ist die doppelte mittlere Latenzzeit für das Aufgreifen der Daten in der Spur. Die mittlere Zugriffszeit ist die Zeit, die nötig ist, um eine Spur zu erreichen, bei der der Lese-/Schreibkamm bewegt werden muß. Dann haben wir wiederum die Übertragungsrate, in Kilozeichen pro Sekunde. Schließlich noch den Hersteller der Platteneinheiten und die Miete in DM pro Monat.

Sie sehen, gegenüber der Trommel kommt hier noch die Zugriffszeit als neue Zahl dazu. Bei den kleinen Platteneinheiten habe ich eine kleine Auswahl, es gibt eine große Zahl dieser Platteneinheiten. Vor allem im mittleren Kapazitätsbereich ist eine sehr sehr große Zahl von Herstellern tätig. Deshalb sind hier nur einige wenige ausgewählt. Da habe ich eine Platte mit 2,4 bis 7,4 MZeichen. Die Zahl der Spuren ist 100 bis 200, Zugriffszeit ist 39 msec. bis 700 msec. 0,7 sek. sind nötig, im ungünstigsten Fall, um den Kamm von einer Spur zur nächsten zu bewegen. Dann haben wir die Rotationszeit 20 msec., die Übertragungsrate 100 KZeichen pro sec. Das ist die IBM-Platte 5444-1 und kostet DM

Charakteristiken

	kleine (Auswahl) Datenzellen-Finheiten			mittlere (Auswahl) Datenzellen-Einheiten		
Kapazität mio Zch	2.4-7.4	3.6	7.25	11.2	20	83
Spuren insgesamt	100-200	300	2000	2000	1200	5529
Zugriffszeit (ms)	39-395 -700	20-280	25-135	25-135	?	110
Rotationszeit (ms)	20	17.5	25	12.5	20	21.5
Übertragungsrate K Zch sec	199	148	156	240	230	50
Hersteller	IBM 5444/1	H155	IBM 2311/12	Stand. Comp	B	NCR 353
Miete (DM/Mo)	425	925	975	2150	1950	3700
						10500 +1650
						14700
						8 Plattenein- heiten, auch 6 lieferbar

Tabelle 6.7.3.2: Charakteristiken

große (Auswahl) Datenzellen-Einheiten		andere Datenzellen-Einheiten	
341	300	629	838
131K	32K	76.8K	65K
145-170	15-120	40-180	21-140
30	2.5.07	25	35
80	190	168	416
H388	H262	Byrant 4000	CDC821
11300	21000 +1650	?	?
		U/RCA RACE	IBM 2321
		561	400
		2.048K	2000K
		135-235	350
		60	25
		70	55
		9100	10200

425,-- Monatsmiete. Die nächste kleine Platte mit 3,6 MZeichen, mit 800 Spuren, 20 - 200 msec Zugriffszeit, Rotationszeit ist 17,5, die Übertragungsrate 148 KZeichen, Hersteller: Honewell, Type 455, mit DM 925,-- Monatsmiete.

Eine mit 7,25 MZeichen als die letzte der kleinen Platten mit 2000 Spuren, 25 - 135 msec Zugriffszeit, 25 msec. Rotationszeit, Übertragungsrate 156 KZeichen, die IBM 2311-12, mit DM 975,-- Monatsmiete. Sie sehen, die Platten hier mit etwa derselben Kapazität kosten einiges weniger als die Trommeln.

Dann haben wir mittlere Platten, einmal mit 11,2 MZeichen, ebenfalls mit 2000 Spuren, 25 - 135 msec Zugriffszeit, 12,5 msec Rotationszeit, 240 KZeichen Übertragungsrate, von der Standard Computer Corporation, mit einer Monatsmiete von DM 2.150,--.

Dann haben wir hier eine mit 20 Mio. Zeichen, 1200 Spuren, die Zugriffszeit ist nicht angegeben, 20 msec Rotationszeit, 230 KZeichen/sec, die Burroughs Platte für DM 1.950,-- pro Monat.

Dann haben wir eine von NCR mit 83 Mio Zeichen, 5529 Spuren, etwa 110 msec und 21,5 msec Rotationszeit, 50 KZeichen Übertragungsrate, NCR 353, mit 3.700,-- DM Monatsmiete. Auch hier sehr beachtliche Kosten. Allerdings vergessen Sie nicht, hier haben wir 83 Mio Zeichen, also etwa eine Größenordnung mehr Zeichen für denselben Preis wie bei den Trommeln.

Dann gebe ich hier noch eine größere an: 173,6 Megazeichen, die hat 32.000 Spuren, 25 - 130 msec Zugriffszeit, 25 msec Rotationszeit, 312 KZeichen Übertragungsrate, die IBM 2314-A2, und die kostet 14.700,-- DM Monatsmiete. Sie sehen, hier haben wir 173 Mio. Zeichen gespeichert und das ist noch nicht einmal eine große Platte.

Jetzt kommen wir zu den großen Platteneinheiten. Da-

von sind 4 hier aufgeführt.

341 Mio Zeichen, 131 K Spuren, 145-170 Zugriffszeit, 30 msec Rotationszeit, Übertragungsrate 80 K Zeichen, die Honeywell 388 mit DM 11.300,-- Monatsmiete.

Einige Werte der Magnetkarteneinheiten: hier haben wir 561 Mio Zeichen einmal mit 2048 Karten und hier haben wir eine MZ von 135 - 235 msec, 60 msec UZ, 70 K Zeichen Übertragungsrate, die Univac/RCA RACE-Einheit mit DM 9.100,-- Monatsmiete. Dann noch die IBM 2321 Datenzelle mit ähnlichen Charakteristiken.

Sie sehen im Vergleich, daß die Magnetkarteneinheiten billiger sind als die großen Platten entsprechender Kapazität. Hier haben wir Honeywell 388, die ist schon teurer als die Datenzelle IBM 2321 oder höchstens gleich im Preis, aber mit größerer Kapazität. Allerdings ist die Zugriffszeit etwas größer, doch ist die Rotationszeit ein wenig kleiner. Das Besondere bei den Platteneinheiten ist, daß eine große Anzahl von Herstellern steckkompatible Einheiten insbesondere für die IBM 2311-12 und die IBM 2314 produzieren, d.h., in allen äußerlichen Charakteristiken identisch. Nur in den Geräten selber sind Verschiedenheiten und im Aufbau, und zwar im allgemeinen aus patentrechtlichen Überlegungen heraus.

6.7.3.3 Positionierungsverfahren

Bei den Platteneinheiten hat es sich gezeigt, daß eine große Zahl von Herstellern steckkompatible Platten herstellt und produziert, insbesondere für die IBM 2311, d.h., daß diese Einheiten sich gegenüber der Kontrolleinheit genauso verhalten müssen wie die ursprünglichen Geräte, und zwar nicht nur im Aussehen, in der Einteilung und Organisation der Daten, sondern auch bezüglich der Kontrollsignale. Diese müssen identisch sein, denn sonst wäre ein steckcompatibles Ver-

halten nicht möglich. Bei den Kontrollsignalen haben wir solche, die von der Einheit gesendet werden und dann Kontrollsignale, die von der Einheit empfangen werden. Unterschiede in der Bauweise eines unabhängigen Herstellers haben sich auf andere Charakteristiken zu beziehen als diese. Andererseits besteht die Schwierigkeit, daß ein Gerät von dem ursprünglichen Hersteller patentiert ist und ein unabhängiger Hersteller dann die Patente irgendwie umgehen muß. Abweichungen in der Konstruktion und in der Organisation einer Platteneinheit oder jeder anderen Einheit sind nur außerhalb dieses Bereichs möglich. Unterschiede sind in folgenden Merkmalen gebräuchlich:

- a) Art und Weise der Bewegung des Lese-/Schreibkamms
- b) Feststellung der augenblicklichen Kammposition, d.h., auf welchem Zylinder der Kamm gerade steht, sowie der Grobeinstellung
- c) Feineinstellung des Lese-/Schreibkamms
- d) Sicherheitskammentzug

Statt der hydraulischen Bewegung bei der IBM 2311 Platteneinheit haben wir z. B. häufig elektrische Linearmotoren und zwar in folgender

a) Art und Weise:

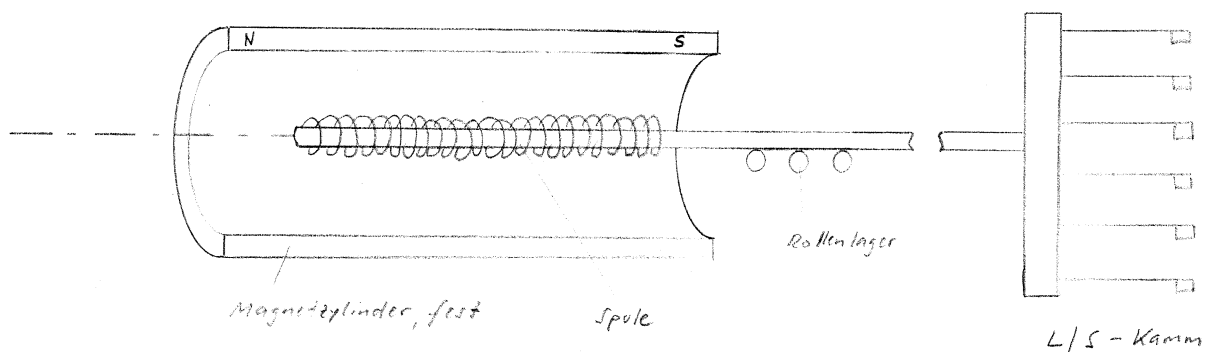


Abb. 56:

Wir haben einen Magnetzylinder, in dem sich ein Stab mit einer Spule befindet. Dieser Magnetzylinder ist im Gerät festmontiert und der Stab ist am Kamm angeschlossen. Auf dem Stab ist die Spule befestigt. Nun haben wir noch ein Lager auf dem dieser Lese-/Schreibkamm vor- und rückwärts bewegt werden kann. Ein Gleichstrom durch die Spule zieht nun diese Kammvorrichtung vor und zurück, je nach der Polung. Dies ist eine Variation der Bewegung der Kammvorrichtung anstelle der hydraulischen Bewegung.

b) Kammpositionsprüfung

Die nächste Variation ist die Kammpositionsprüfung, und das Kammfesthalten. Wir vermuten oder nehmen an, daß die Position des Kammes irgendwie mechanisch oder optisch geprüft wird. Nehmen wir z. B. an, wir haben eine kleine Glasskala. Dann können wir durch einen Lichtstrahl feststellen, welche Teilstriche der Skala unter der Aufnahmelupe stehen und so die Kopfposition steuern.

Bei dem Calcomp-Plattengerät haben wir eine induktive Methode, und zwar arbeitet diese auf folgende Weise:

s. Abb. 57 - S. 175 -

Wir haben eine zahnförmige Oberfläche. Darüber bewegen sich nun zwei Polschuh-Paare mit den Spulen A und B, die fest miteinander verbunden sind. Auf jedem dieser Polschuhe haben wir einmal eine Primärwicklung und eine Sekundärwicklung. Dieses Polschuhpaar ist mit dem Kamm verbunden. Sie können sich vorstellen, je nach dem, wo das Polschuhpaar liegt, wird mehr oder weniger Spannung induziert. Wenn nun einer dieser Polschuhe auf dem Berg dieses welligen Gebildes liegt, ist die Induktion größer als wenn der Polschuh über einem Tal liegt. Das ist das Prinzip mit

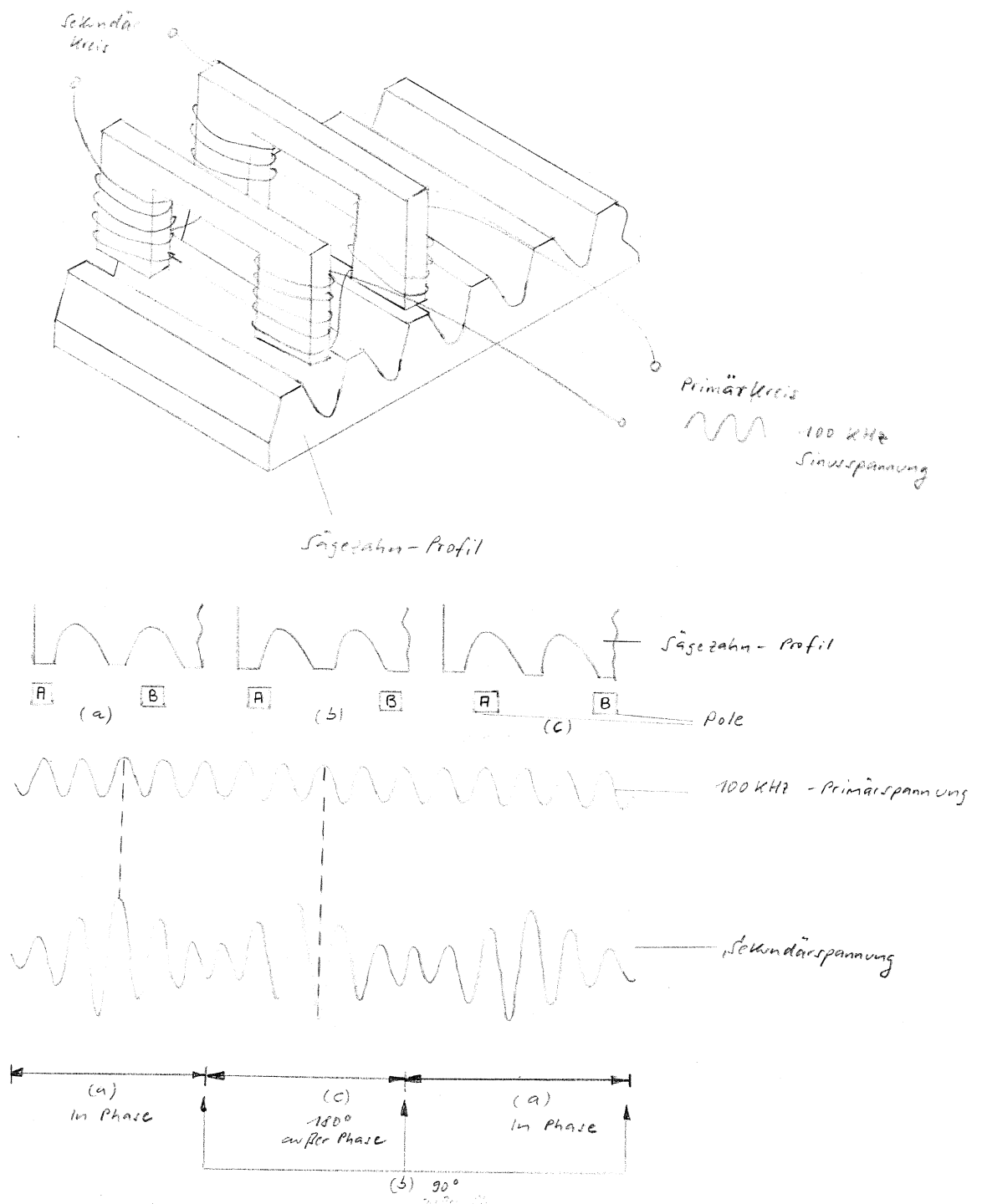


Abb. 57: Prinzip einer Kammpositionsprüfung

dem hier abgefühlt wird. Die Primärspannung ist eine 100 KHz-Wechselspannung, die in den Sekundärspulen abgefühlt wird. Wenn die Pole in Stellung a) stehen, ist die Induktion in der Sekundärentwicklung von A maximal, und in der Sekundärentwicklung von B minimal. Dann haben wir die nächste Stellung b), in der A und B jeweils auf der Flanke liegen. Weiter haben wir die Stellung c) in der A in dem Tal und B auf der Höhe liegen. Das sind die drei möglichen Stellungen, die

grundverschieden sind. Wir haben etwas zu differenzieren, je nach dem, wie weit A und B auf der Flanke liegen, aber immerhin werden wir sehen, daß sich das auskompensieren wird. Wenn A hoch auf der Flanke liegt, liegt B nieder auf der Flanke und umgekehrt. Das wichtige ist nun, daß, sobald ein Zahn unter die Pole kommt, sich jeweils die Sekundärspannung ändert. Die Sekundärspannungen von A und B sind gegeneinander geschaltet. Damit können wir feststellen, wo nun der Kamm liegt. Die Spannungen der Wicklungen A und B überlagern sich, d.h., wir haben eine Interferenzerscheinung. In den Fällen a) und c) haben wir eine hohe Amplitude der Sekundärspannung. Die Interferenzbilder sind jedoch spiegelbildlich. Im Falle b) haben wir einen Tiefstwert, denn wenn die Pole auf den Flanken liegen, sind die Spannungen außer Phase, d.h., sie werden sich gegenseitig auslöschen. Je nach Höhe der Sekundärspannung kann die Lage des Kamms bestimmt werden.

Bei der Bewegung des Kamms wird der Weg, den der Kamm zurücklegen muß, aus der alten und der verlangten Position berechnet. Wie kann die alte Position festgestellt werden? Sagen wir z. B., wir haben jetzt den Kamm auf Zylinder 3 stehen, und nun soll der Kamm auf Zylinder 114 bewegt werden. Wie kann festgestellt werden, daß der Kamm auf Zylinder 3 steht? Die Position halten wir jeweils in einem Register durch Abspeichern fest. Dann wissen wir z. B., daß der Kamm auf Zylinder 3 steht. Wenn wir ihn auf Zylinder 114 bewegen, dann geben wir diesen Wert in das Register ein, d.h., dieser Wert ist im Zylinderadressenregister gespeichert.

Die Größe der verlangten Bewegung berechnet sich nun aus der alten Position und aus dem Wert im Register der Kontrolleinheit. Die Kontrolleinheit gibt an, wo-

hin der Kamm bewegt werden muß, d.h., welchen Zylinder wir lesen wollen. Als das Ergebnis dieser Berechnung wird der Wert als Analogwert auf den Linearmotor gegeben, der dann die Kammvorrichtung in der groben Bewegung steuert. Während der Bewegung wird die Induktion beobachtet, die von den Sekundärwicklungen dieser Spulenpaare erzeugt wird, d.h., ich zähle nun diese Spitzen und Tiefen der Überlagerungsspannung. An jeder Interferenz wird ein neuer Zylinder überquert. Während der Kamm sich nun bewegt, haben wir die beiden Spulen und das Interferenz-Profil. Damit können wir nun feststellen, wo die Spulen sich befinden. Es wird eine Spannung in den Sekundärwicklungen induziert. Diese kann ich nun prüfen und dann in einem Differenzregister jedesmal, wenn nun eine Null erscheint, eine Ziffer abziehen. Damit können wir feststellen, wie weit wir von der gewünschten Spur im Zylinder weg sind. Wie geht nun im einzelnen die Feinbewegung vor sich?

c) Feineinstellung des Kamms

Wir bewegen den Kamm bis die verstärkte Analogspannung Null ist. Durch diese Grobbewegung braucht sich der Kamm nicht ganz auf diese Spur stellen, aber immerhin in die Nähe, und das ist der wesentliche Teil der Grobeinstellung. Sie sehen, es ist relativ einfach. Alles was wir hier brauchen, ist, eine Differenzspannung und einen Analog-Converter und einen Verstärker. Die Spannung wird dann auf die Spule gegeben. Wenn eine positive Spannung gegeben wird, erfolgt die Bewegung in der einen Richtung und umgekehrt in der anderen. Es ist äußerst einfach. Das Schwierige ist die Feineinstellung. Zur Feineinstellung verwenden wir die Amplitude der demodulierten Induktionsspannung, natürlich ebenfalls verstärkt. Mit dieser Spannung steuern wir den Linearmotor in der Nähe der gesuchten Spur.

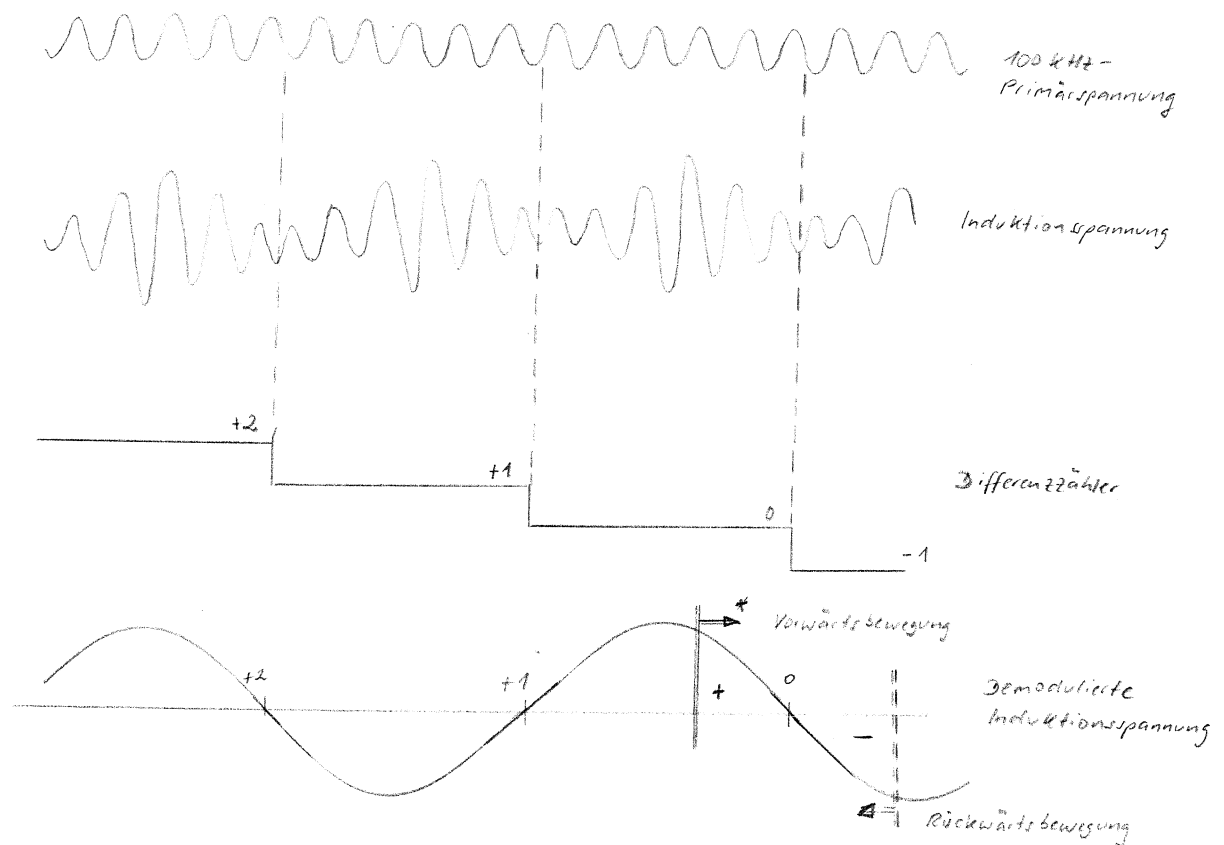


Abb. 58: Feineinstellung des Kamms

Nehmen wir an, daß an der markierten Stelle^{*} der gesuchte Zylinder ist. In dem Differenzähler haben wir dann den Wert 0. Nun haben wir noch die demodulierte Induktionsspannung, d.h., diese Spannung wird einfach gleichgerichtet, demoduliert. Wir bekommen positive und negative Nulldurchgänge. Wenn der Kamm durch die Grobbewegung hier in die Umgebung der gesuchten Spur gebracht wurde, so können wir nun mit Hilfe dieser demodulierten Induktionsspannung den Kamm vorwärts bewegen, oder, wenn der Kamm in der schraffierten Region steht, zurückbewegen, weil ja automatisch in dieser Region die Spannung negativ herauskommen wird, also umgepolt ist. Sie sehen, das ist eine sehr einfache Methode. Bedingung ist nur, daß der Kamm zwischen dem Maximum der vorhergehenden Position und dem Minimum der nachfolgenden Position oder umgekehrt liegen muß. Sobald dies der Fall ist, arbeitet diese Methode einwandfrei. Nun erkennen Sie auch, weshalb wir eine

zweistufige Bewegungsvorrichtung haben. Zweistufig deshalb, um über die vorhergehenden Maxima und Minima erst einmal hinwegzukommen, falls sich der Kamm über mehrere Spuren hinweg bewegen muß. Sobald wir in der Nähe sind, können wir die demodulierte Induktionsspannung auf den Linearmotor geben. Die Spannung des Linearmotors muß also von der Spannung des Differenzzählers auf die Spannung des Demodulators umgeschaltet werden. Dies ist elektronisch einfach zu realisieren.

d) Sicherheitskammentzug

Dieser Sicherheitskammentzug ist lebensnotwendig für den Betrieb der Platteneinheit und zwar bei Verlust der Netzspannung. Sie wissen, die Platte hält nur dann den Luftfilm zwischen Plattenoberfläche und dem Lese-/Schreibkopf, solange sich die Platte bewegt, und zwar sehr schnell bewegt. Sobald sich die Bewegung verlangsamt, geht dieser Film verloren, und wenn sich die Bewegung noch weiter verlangsamt, dann wird die Oberfläche der Platte von dem Lese-/Schreibkopf zerkratzt. Um dies zu vermeiden, muß eine Vorrichtung existieren, die die Kammvorrichtung und die Köpfe aus den Platten entfernt, sobald die Spannung abfällt. Dies geht bei den IBM Einheiten sehr einfach und zwar deshalb, weil die hydraulischen Pumpen noch genügend Druckvorrat in den Behältern gesammelt haben und selbst, wenn die Pumpen stehen bleiben, ist noch genügend Druck da, um den Kamm zu entfernen. Was bei den elektrischen Methoden zu geschehen hat, ist, daß die Netzspannung abgefühlt wird. Von einem Schwellwert an, d.h., sobald die Netzspannung unter einen Schwellwert sinkt, wird die ganze noch verbleibende Spannung sofort auf den Linearmotor gegeben, um den Kern sofort völlig zurückzuziehen. Die Frage ist, wie wird der Schwellwert gewählt? Es kommt sehr häu-

fig vor, daß die Netzspannung etwas schwankt, es muß nun ein geeigneter Schwellwert herausgesucht werden, der so niedrig wie möglich liegt, aber doch noch genügend hoch ist, um mit Sicherheit den Kamm aus dem Plattenstapel zu entfernen. Dies ist technisch realisierbar. Zurückblickend können wir vielleicht vergleichend sagen, daß die elektrischen Methoden den hydraulischen mindestens ebenbürtig sind.

6.7.4 Die Datenzelle

6.7.4.1 Charakteristiken

Die Datenzelle ist eine Großspeichereinheit, d.h., wir haben hier die größten Speichermöglichkeiten (zwar ist es auch nicht mehr ganz richtig, denn wenn Sie zurücksehen, es gibt große Plattenspeicher, die 500 Mio. Zeichen speichern und Sie werden sehen, diese Datenzellen speichern auch nicht wesentlich mehr und eben auch in dieser Größenordnung). Nun, der Unterschied ist, daß die Datenzelle für eine vergleichbare Datenmenge billiger ist als eine vergleichbare Platte. Die Datenmenge, die in Datenzellen gespeichert werden kann, ist vergleichbar mit der Datenmenge, die auf größten Bändern oder in sehr grossen Dateien zu speichern sind, in der Größenordnung von mehreren hundert Megazeichen (100 - 1000 MBytes). Das ist allerdings nicht die allergrößte existierende Speichereinheit. Es existiert ein optischer Speicher, genannt UNICON. Dieser Speicher hat die Eigenschaft, daß er nur gelesen werden kann, er ist nur ein Lesespeicher. Optisch arbeitet er mit Laserstrahlen und zwar geschieht hier die Speicherung durch Zerstörung der Oberfläche einer Kunststoffolie. Diese Oberfläche wird mit dem Laser zerstört und somit beschrieben, und dann mit Laserstrahlen abgetastet. Bei diesem Speicher werden 10^{12} Zeichen zu einer Million Dollar gespei-

chert. Es wurde ausgerechnet (um eine Vorstellung dieser Kapazität zu bekommen), daß wenn wir 1000 dieser Einheiten hätten, alles gespeichert werden könnte, was jemals von der Menschheit überhaupt geschrieben worden ist (wird auf 10^{15} Zeichen geschätzt). Das ist eine bedeutende Leistung, daß alles was überhaupt jemals geschrieben worden ist, mit relativ kleiner Zugriffszeit zur Verfügung stünde. Es gibt Institutionen, denen diese Möglichkeit der Preis wert ist, und zwar ist es die US-Army. Z.Zt. arbeiten die Entwicklungsingenieure mit holographischen Methoden. Dabei wird dann die Speichermöglichkeit wesentlich größer. Immerhin, das sind die größten Speicher, die wir zu berücksichtigen haben.

6.7.4.2 Der Aufbau der Datenzelle

Bei dem Aufbau der Datenzelle beschränke ich mich auf IBM 2321. Es wird von Univac/RCA, eine sog. RACE-Unit hergestellt, die im Prinzip auf derselben Basis arbeitet, allerdings nicht in einem runden Behälter untergebracht ist, wie bei den IBM-Geräten, sondern in einem viereckigen Gehäuse.

Wir haben hier, wie schon erwähnt, eine Art Karussell, und dieses Karussell hat nun 10 Sektoren, d.h., 10 Zellen.

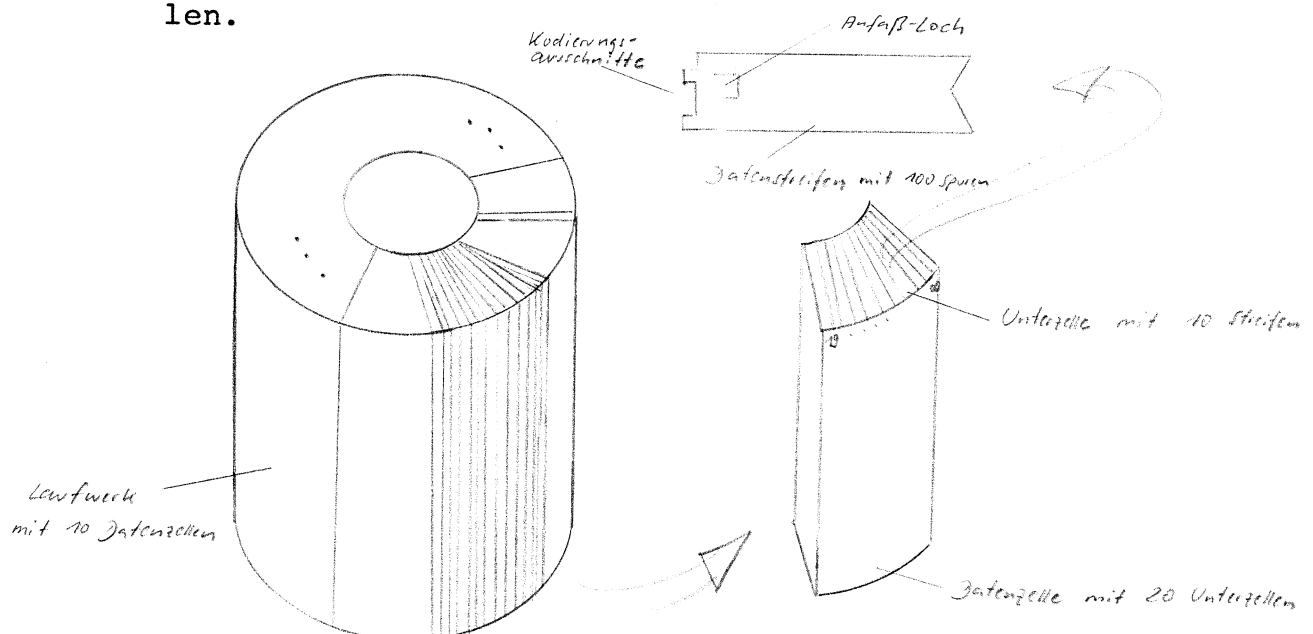


Abb. 59: Aufbau der Datenzelle

Ein Laufwerk hat 10 Zellen und jede dieser Zellen 20 Unterzellen. Betrachten wir nun eine Unterzelle. Jede Unterzelle enthält 10 Streifen. Die Oberkante ist absichtlich nicht symmetrisch. Die Kerbungen oder diese Absätze, die hier herausstehen, dienen zur Kodierung dieses Streifens. Ein Anfaßloch dient dazu, um den Streifen herauszuziehen. Der Streifen ist insgesamt 13 Zoll lang und 2 1/4 Zoll breit. Ein Streifen enthält nun 100 Spuren à 2000 Zeichen. Sie sehen, wir haben hier eine hierarchische Unterteilung dieser Speichereinheit. Fangen wir mit den Zeichen an.

Die Spur hat 2000 Zeichen. Wir haben 100 Spuren, insgesamt also 200.000 Zeichen auf einem Streifen.

Eine Unterzelle enthält 10 Streifen, d.h., 2 Mio Zeichen.

Eine Zelle hat 20 Unterzellen, d.h., insgesamt haben 40 Mio Zeichen in der Zelle Platz.

Eine Datenzelleneinheit hat 10 Zellen, d.h., wir haben 400 Mio Zeichen in der Datenzelle.

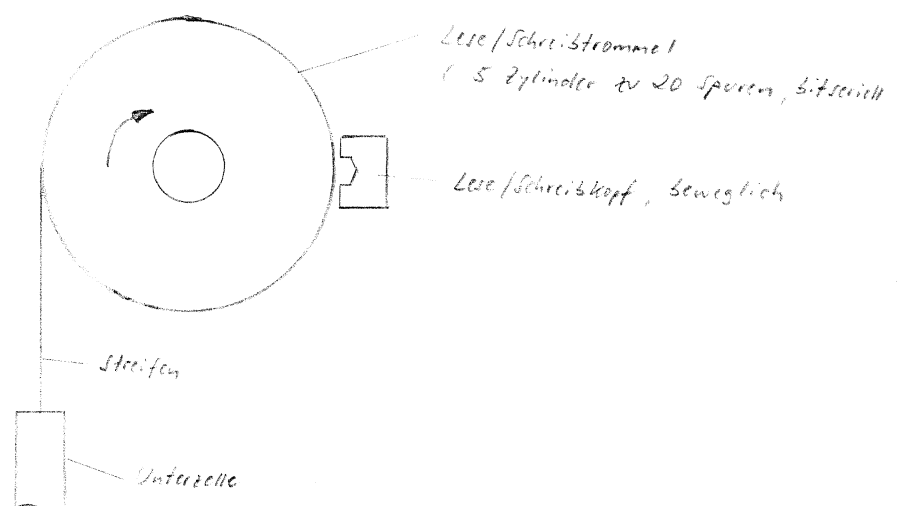


Abb. 60: Aufspannen eines Datenstreifens

6.7.4.3 Arbeitsweise

Die Zelle ist drehbar, und nun haben wir eine Trommel, über der Zelle liegend (Abb. 60). In der Oberfläche dieser Trommel ist ein Greifer eingelassen. Dieser greift nun die Karte, die gesucht wird, aus dem Stapel heraus, d.h., die Karten werden bei der Auswahl durch eine Steuereinheit verschoben, und die Haken greifen nun die durch die Adresse gesuchte Karte heraus. Je nach der Kodierung ist die Stellung der Haken verschieden, d.h., die Kodierungseinschnitte liegen eben nur für eine Karte an der richtigen Stelle und dort greift nun der entsprechende Haken ein und zieht dann eine Karte heraus. Diese Karte wird nun von einem Halter, der aus der Trommel herauskommt, herausgezogen und festgehalten. Auf der Oberfläche der Trommel befinden sich Löcher, und die Karte, die herausgezogen wird, wird nun auf der Oberfläche durch einen Luftsog angesaugt, und dann auf der Trommel festgehalten. Sie läuft dann um. Es existieren 5 feste Schreibköpfe. Wir haben somit 5 Zylinder à 20 Spuren, d.h., ein Lese-/Schreibkopf ist für 20 Spuren in bitserieller Arbeitsweise da, um die Zeit zur Verschiebung eines Lese-/Schreibkopfes entsprechend niedrig zu halten. Es gehen etwa 200-650 ms verloren, bis die Karte herausgezogen ist, festgehalten wird und umläuft. Dann muß noch gleichzeitig der Lese-/Schreibkopf bewegt werden, während die Karte umläuft.

Nehmen wir an, daß die Lese-/Schreibtrommel leer ist, wir können nun eine Karte heraufziehen, wenn wir eine brauchen. Dazu muß nun erstmal, um zum gesuchten Streifen zu kommen, der Zellaufbau bewegt werden, d.h., die Zelle muß bis zur richtigen Unterzelle rotiert werden.

Jetzt haben wir die Unterzelle unter der Abgriff- und

und Auswahlvorrichtung. Dann wird der gewünschte Streifen herausgesucht und Drahtfinger ziehen mittels der Kodierungsausschnitte den richtigen Streifen heraus. Jeder Streifen hat einen verschiedenen Ausschnitt. So bleibt nur eine Karte stehen. Die gesuchte Karte kann die erste sein, diese Karte wird nun aufgegriffen mit dem Haken, der aus der Trommel herauskommt, wird auf die Trommel heraufgezogen und durch den Luftsog festgehalten. Nun, sobald der Streifen auf der Trommel liegt, wird die Trommel zum Motor eingekuppelt. Die Trommel ruht vorher und bewegt sich nun und der Streifen, der auf der Trommel liegt, wird an den Lese-/Schreibköpfen vorbeigeführt. Dafür werden etwa 200 bis 650 msec verbraucht, d.h., eine viertel bis eine halbe Sekunde geht für den ganzen Vorgang verloren. Nach der Bearbeitung wird der Streifen an seine Stelle zurückgegeben. Und zwar deshalb, weil wir diese noch festgehalten haben mit den Fingern. Alles was wir zu tun haben, ist, die Trommel anzuhalten, und den Streifen fallen zu lassen.

Das ist schematisch die Arbeitsweise der Datenzelle. Wir könnten sehr viel auf Einzelheiten eingehen, aber das ist nicht so interessant, die Hauptsache ist, das Prinzip zu verstehen. Ähnlich arbeiten auch die Univac/RCA RACE-Unit. Allerdings ist es dort der Kasten, der vor- und zurückbewegt wird (Also, wir haben eigentlich eine Trommel mit auswechselbarer Oberfläche).

6.8. Datenstationen

6.8.1 Prinzipieller Aufbau

Zuerst betrachten wir die Konsolschreibmaschine und gleichzeitig auch die Fernschreibgeräte. Bei der Konsolschreibmaschine haben wir die Aufgabe der Kommunikation zwischen dem Operateur und dem System. Bei der Fernschreibeinheit

haben wir die Aufgabe der Kommunikation zwischen dem Benutzer und dem System. Wir haben Zweiwegoperationen, d.h., einmal vom Rechnersystem zum Operateur oder zum Benutzer, für Aktionen, die für den Ablauf der Programme nötig sind. Programm ist hier in sehr allgemeinem Sinn gebraucht; "Programm" bedeutet für den Operateur hier der Gesamtablauf des Systems, die Gesamtaufgaben des Systems. Wenn wir einen gewöhnlichen Benutzer einer Datenstation betrachten, oder einer Fernschreibeinheit, dann ist hier die Datenausgabe zu nennen.

Der andere Weg wäre vom Operateur oder Benutzer zum Rechner. Dies ist für besondere Aktionen, z.B. für den Operateur die Änderung der Priorität der Programme, Feststellung des Betriebsverhaltens, Änderung der Systemkonfiguration, usw. Bei dem Benutzer wäre es die Anweisung, welches Programm laufen soll oder die Dateneingabe. Die Fernschreibmaschine arbeitet analog der gewöhnlichen Schreibmaschine. Vom Rechner jedoch wird nur die Schreibeinheit benutzt, während der Operateur oder der Programmierer, der an der Datenstation sitzt, die Schreibeinheit und die Tastatur benutzt, denn er möchte das, was er schreibt, auch geschrieben sehen.

Dagegen benützt der Rechner die Tastatur nicht. Das geht folgendermaßen vor sich:

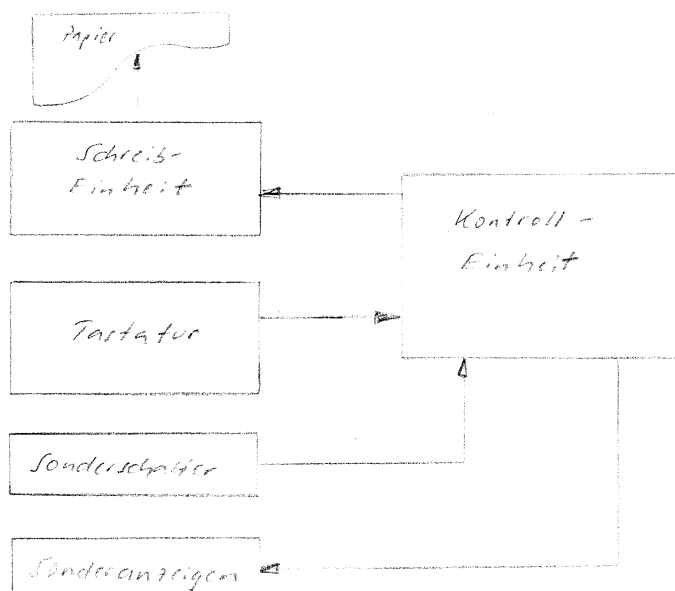


Abb. 61: Blockschaltbild einer Konsolschreibmaschine

Wir haben hier die Schreibeinheit mit dem Papier. Zu dieser Schreibeinheit kommen wir von der Kontrolleinheit. Die Tastatur erlaubt mir eine Eingabe in die Kontrolleinheit. Dies ist der Aufbau einer Datenstation für den Normalbenutzer. Manchmal gibt es noch Sonderschalter, die in die Kontrolleinheit eingehen (z.B. bei der Konsolschreibmaschine), und es gibt auch häufig noch Sonderanzeigengeräte, die von der Kontrolleinheit kommen. Vor einiger Zeit waren noch unbedingt Sonderschalter nötig. Heutzutage ist das nicht mehr so häufig der Fall.

Am häufigsten verwendet wird der ASCII-Code, z.B. bei allen neueren Fernschreibmaschinen. Dann haben wir noch EBCDIC, vor allem bei den IBM Geräten. Bei älteren Fernschreibgeräten gibt es noch den Baudot-Code. Ein Code, der bei den 360/70 Konsolschreibmaschinen vorkommt, ist der PTTC Code, das ist der "paper tape and transmission code". Jeder dieser Codes ist gleichwertig mit Ausnahme des ASCII und des EBCDIC, beides sind 8-bit-Codes und die anderen 6-bit-Codes. Dieser PTTC-Code kommt übrigens von der IBM 1401 oder von der IBM 1601, ist also ein veralteter Code. IBM hatte einen Großteil ihrer Geräte auf diesen Code abgestimmt. Deshalb hat sie diesen Code behalten.

6.8.2 Arbeitsweise

Ohne Sonderschalter und ohne Sonderanzeige arbeiten diese Einheiten wie normale Schreibmaschinen oder normale E/A-Einheiten, d.h., die Arbeitsweise der Schreibmaschinen muß vom Programm abgefragt werden. Die einfachen Datenstationen können nur auf diese Art und Weise arbeiten.

Mit Sonderschaltern haben wir eine Änderung, und zwar haben wir das bei den Maschinen der IBM 360/370 und bei der Univac/RCA. Es gibt Sonderschalter und zwar folgende: READY, REQUEST, NOT READY, CANCEL, EOB, ALTN. Was ist die

bedeutung dieser Sonderschalter?

Der READY-Schalter setzt die Kontrolleinheit in Empfangszustand, wenn möglich oder sobald wie möglich, d.h., dies geht über das Unterbrecherwerk. READY setzt also von der Tastatur aus die Kontrolleinheit in Empfangszustand: wir möchten etwas auf der Tastatur eingeben und zwar deshalb, weil die Kontrolleinheit zuvor die Tastatur gesperrt hatte. Sobald die Kontrolleinheit die Schreibeinheiten benötigt, wird die Tastatur gesperrt und man kann nichts mehr eingeben. Deshalb wird der Sonderschalter READY benötigt.

Nun haben wir REQUEST. Das verlangt Unterbrechung der Arbeitsweise des Rechners.

NOT READY setzt die Kontrolleinheit in den Wartezustand.

CANCEL löscht die bisherige Eingabe, z.B. wenn ich mich vertippt habe, oder an der Tastatur den falschen Knopf gedrückt habe, kann ich durch CANCEL die ganze Eingabe löschen und neu anfangen.

EOB bewirkt die Freigabe der Konsole für Empfang vom System.

ALTN (alternate) ist eine Kontrolltaste, die zusammen mit anderen Tasten betätigt wird, um spezielle Vorgänge zu bewerkstelligen. Das sind die Sonderschalter.

Wir haben noch die Sonderanzeigen, und zwar dienen diese erstens für die Instandhaltung, also für den Wartungsdienst, und für die Operation. Damit kann ich feststellen, was gerade vor sich geht, z.B. wenn PROCEED aufleuchtet ist eine Eingabe von der Tastatur möglich; dann haben wir ALTN, sie ist mit der REQUEST-Taste verbunden, d.h., wenn ich REQUEST drücke, leuchtet die ALTN-Lampe auf, das bedeutet, ich möchte etwas unterbrechen. Weiter

gibt es noch Command Reject, Intervention required, BUS OUT und Equipment. Diese Lampen dienen zur Anzeige von Schwierigkeiten bei der Übertragung. Sie sind speziell für den Wartungsdienst gedacht.

Betrachten wir nun speziell die Konsolschreibmaschine und die Eingabe unter Verwendung von Sondertasten mit dem Unterbrecherwerk. Die Tasten werden in folgender Reihenfolge gedrückt:

Erst einmal READY, d.h., ich muß dem System mitteilen, daß ich etwas tun möchte. Dann kommt die REQUEST-Taste. Dann READY-REQUEST, hier leuchtet die Attention-Lampe auf, und nun muß ich warten, bis die PROCEED-Lampe aufleuchtet. Sobald die PROCEED-Lampe aufleuchtet, ist die Eingabe der Information oder des Befehls möglich. Falls ein Fehler bei der Eingabe entsteht, benutze ich die CANCEL-Taste und wiederhole meine Eingabe. Am Ende meiner Eingabe drücke ich ALTN plus EOB, d.h., Ende der Eingabe.

Bei der Ausgabe vom Rechner ist die Tastatur gesperrt. Andererseits ist die Ausgabe vom Rechner unterbrochen, solange die PROCEED-Lampe leuchtet. Solange die PROCEED-Lampe leuchtet, kann also der Rechner die Konsole nicht benützen.

6.8.3 Kontrolleinheit

In die Schreibeinheit hinein geht ein Übersetzer. Der Übersetzer bekommt das Zeichen aus dem Datenpuffer. Zum Datenpuffer kommt dieses Zeichen über einen Schalter vom Ausgangsbuss des Rechners. Wir haben noch einen weiteren Ausgang aus dem Ausgangsbuss, und zwar zum Befehlsregister, wor der gerade ausgeführte Lese/Schreibbefehl zur Kontrolle und Nachprüfung aufgewahrt wird. Weiter haben wir noch einen Ausgang vorher, und zwar geht es hier um den Adressenvergleich.

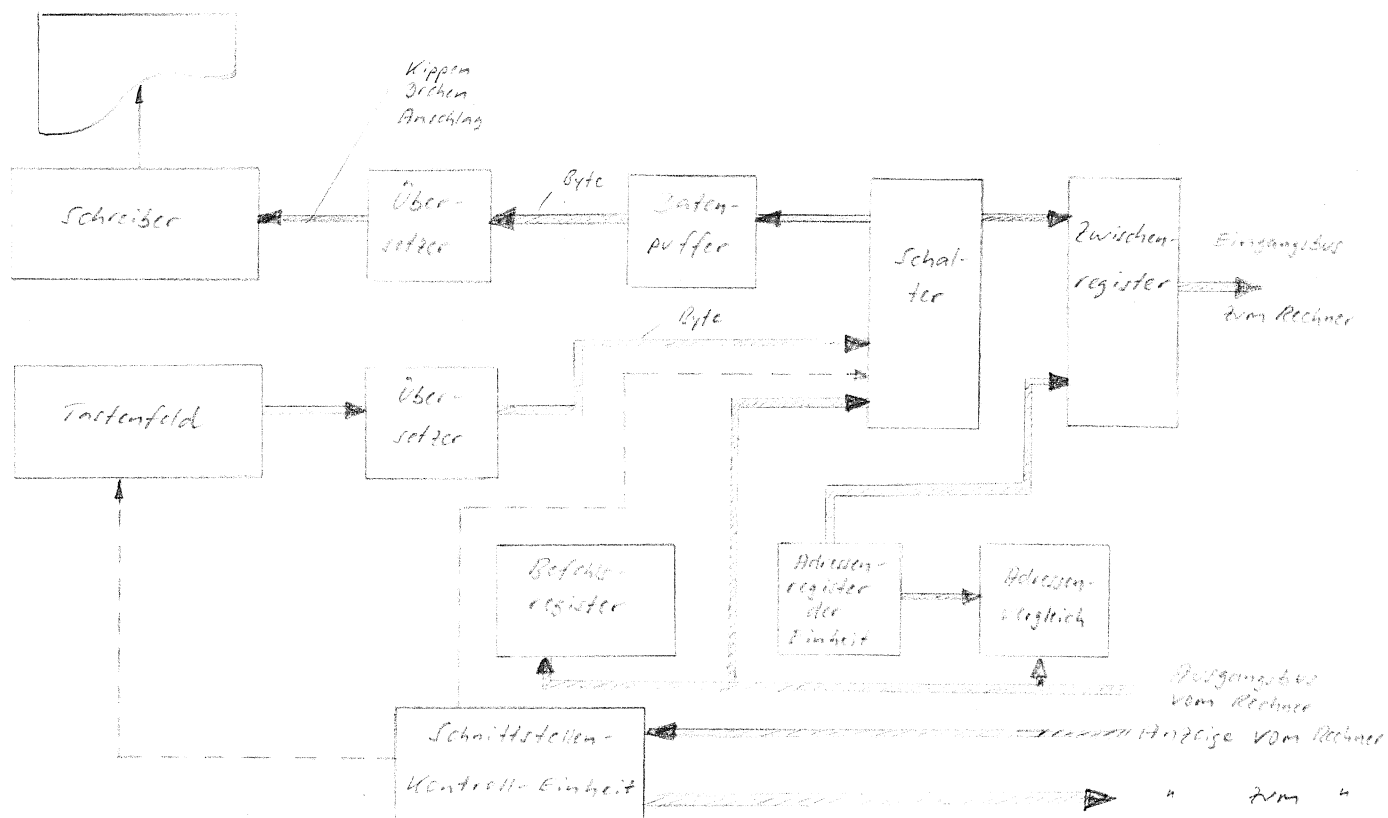


Abb. 62: Blockschaltbild einer Kontrolleinheit

Aus dem Schalter heraus kommen wir in ein Zwischenregister. In dieses Zwischenregister geht auch das Adressenregister der Einheit ein. Vom Zwischenregister geht ein Eingang dann in den Rechner. In den Schalter geht nun der Ausgang von einem Übersetzer hinein, der die Zeichen vom Tastenfeld übernimmt. Dies ist nicht alles, wir haben noch zwei Busse, die für die Anzeigen vom und zum Rechner. Diese gehen in die Schnittstellenkontrolleinheit ein. Diese Einheit bestimmt, wann gelesen und wann geschrieben wird, d.h., hier haben wir die Tastenfeldkontrolle. Das Tastenfeld wird gesperrt, wenn der Rechner schreiben will, und wird entsperrt, wenn der Benutzer schreiben darf.

Wenn wir ein Zeichen in das Tastenfeld eingeben, dann geht es in den Übersetzer, d.h., von der mechanischen

Bewegung wird es übersetzt in elektronische Impulse. Wir hatten das PTTC-Code System hier bei der IBM verwendet, und dann geht es über den Schalter in den Zwischenspeicher und zum anderen Eingangsbuss des Rechners.

Wenn ein Zeichen vom Rechner kommt, dann wird dieses Zeichen über den Schalter vom Ausgangsbuss des Rechners in den Datenpuffer gegeben und als Zeichen in den Übersetzer gebracht. Der Übersetzer muß das Zeichen in eine Kippung, eine Drehung und in den Anschlag des Schreibkopfes übersetzen.

6.8.4 Konsoleingabe

Die nächste Frage ist nun, wie geben wir Daten ein, d.h., wie ist die Dateneingabe von einer Konsole oder einer Schreibmaschine, über ein Terminal, und zwar ohne Unterbrecherwerk und ohne Sondertasten? Wir haben in Abschnitt 6.8.2 gesehen, wie die Eingabe mit Sondertasten verläuft, und nun wollen wir den entsprechenden Fall betrachten. Hier können wir nicht den Rechner benachrichtigen, daß wir Daten zur Eingabe bereit haben. Mit den Sondertasten war dies möglich, wir konnten durch die Sondertaste angeben, daß wir jetzt einige Daten eingeben möchten. Das Unterbrecherwerk kann ich auch nicht benützen, wenn ich es nicht habe. Deshalb muß hier der Rechner durch das Programm abfragen, ob Daten vorhanden sind, z.B. ein kleines Programm:

```
EIN: KSF      "Keyboard skip on flag"
      JMP EIN
      KRB      "Keyboard read buffer"
      .
      .        Befehle zum Aufbereiten des Zeichens
      .
```

Der Rechner prüft, ob ein Zeichen aus dem Tastenfeld er-

schienen ist. Sobald dies der Fall ist, wird die nächste Instruktion übersprungen. Die nächste Instruktion ist ein Sprungbefehl zurück, d.h., wir müssen warten, bis ein Zeichen erscheint. Wenn diese Instruktion übersprungen ist, bedeutet dies, daß ein Zeichen im Lesebuffer vorhanden ist. Hier können wir nun den Puffer mit dem KRB-Befehl lesen. Wir lesen also den Puffer in den Akkumulator. Das ist in jedem Falle ein einfaches Programm, das ich herausgewählt habe, um das Wesentliche darzustellen. Solch eine Instruktionsfolge würden wir nur dann verwenden, wenn wir vom Benutzer etwas anfordern. Z.B. lassen wir ein Programm laufen, und plötzlich muß der Benutzer sich entscheiden, welchen Fall von verschiedenen er auswählen will, d.h., er muß nun etwas eingeben. In diesem Fall wäre es vielleicht sinnvoll, auf diese Weise zu verfahren.

Nach diesen Instruktionen in dem kleinen Programm, folgen noch Befehle zum Aufbereiten und zum Abspeichern der Zeichen, und dann auch noch um festzustellen, wann die Eingabe zu Ende ist. Wie kann nun der Rechner feststellen, daß das Ende der Eingabe erfolgt ist? Es wird entweder ein Sonderzeichen verwendet, um das Ende der Eingabe anzuzeigen, z.B. ein Dollar-Zeichen, dann weiß der Rechner also, jetzt ist die Eingabe zu Ende, oder noch einfacher, das Wagenrücklauf-Zeichen. D.h., wir haben nun eine Zeile eingegeben. Es ist dann sinnvoll, an diesem Punkt zu unterbrechen, und jetzt erst einmal den Rechner untersuchen zu lassen, was eingegeben worden ist.

Wenn wir in unserem Programm noch mehr Zeichen einlesen möchten, so müssen wir wieder zurückspringen nach EIN. Zwischen diesen beiden Zeichen sind mindestens 100 msec für andere Zwecke verfügbar, denn es können maximal nur 10 Zeichen pro Sekunde gelesen werden. Wahrscheinlich ist hier noch mehr Zeit zur Verfügung, da der normale

Programmierer vielleicht nur 3 bis 4 Zeichen pro Sekunde eingeben kann, d.h., es ist noch wesentlich mehr Zeit für andere Zwecke zur Verfügung, die eigentlich ausgenutzt werden sollte. Das ist in diesem Fall aber nicht möglich, in diesem Fall wird die Zeit mit dem Warten auf die Eingabe verbraucht. Man könnte ausrechnen, wie viele Instruktionen in irgendeinem Programm durchgeführt werden können, und dann, sobald diese durchgeführt sind, springen wir wieder zurück und warten, bis das nächste Zeichen da ist, usw. Aber was wir dann hier machen, ist nichts anderes als eine etwas unelegante Version eines Unterbrecherwerks.

Die Eingabe mit Unterbrecherwerk und ohne Sonderschalter läßt diese Wartezeit vermeiden, dadurch, daß das flag mit dem Unterbrecherwerk verbunden ist, d.h., sobald eine Eingabe vorhanden ist, tritt das Unterbrecherwerk in Aktion und kann nun herausfinden, was zu tun ist. Sagen wir z.B. wir haben das Unterbrecherwerk beschäftigt gehabt, und nun wollen wir neu einlesen, d.h., wir arbeiten mit irgendeinem anderen Programm, und haben vorher schon das KSF gegeben. Das braucht einige Zeit, mindestens 100 msec, währenddessen etwas anderes getan werden kann. Kommt das flag herauf, wird das derzeitige Programm unterbrochen und wir können nun die Daten einlesen. Einzelheiten werden wir später sehen.

6.8.5 Konsolausgabe

Bei der Konsolausgabe haben wir ebenfalls wieder verschiedene Fälle zu unterscheiden, einmal ohne Unterbrecherwerk und mit Unterbrecherwerk. Betrachten wir ein kleines Programm zur Ausgabe:

```
AUS: TLS "type load select";  
      lösche Ausgabeflag, lade Schreibpuffer  
      vom Akku und steuere Schreiber an  
      TSF "typer skip on flag"  
      JMP .-1
```

* } Befehle zum Feststellen des Ausgabe-Ende,
 * * Neuladen des Akku vor Rücksprung nach AUS

Mit dem TLS steuern wir den Schreibvorgang an. Mit dieser Instruktion sind viele Vorgänge verbunden. Falls das flag noch da ist, wird es gelöscht, dann wird das Zeichen vom Akkumulator geholt, das geschrieben werden soll, meistens die Bits, die rechtsbündig im Akkumulator stehen, und dann wird der Schreibvorgang initiiert. Dann haben wir das TSF, das Überspringen der nächsten Instruktion, beim Erscheinen des flags. Die nächste Instruktion ist ein Sprungbefehl zurück, d.h., wir warten, um sicher zu sein, daß das Zeichen ausgegeben wurde. Wiederum folgen Befehle, um das Ende der Ausgabe festzustellen, das Aufbereiten von neuen Zeichen, die geschrieben werden sollen, und das Neuladen des Akkumulators.

Wie wird hier nun das Ende der Ausgabe festgestellt?

Vorhin haben wir gesehen, daß wir bei der Eingabe, einmal mit speziellen Sonderzeichen arbeiten können, oder mit dem Wagenrücklauf, denn das ist ja auch nichts anderes als ein Sonderzeichen. Doch wie ist es hier, wenn ich eine Nachricht ausdrucken will?

Da gibt es keine Schwierigkeit, und zwar deshalb, weil das Programm wissen muß, welche Zeichen und wieviele Zeichen ausgeschrieben werden sollen. Dies ist im Programm angegeben. Die Nachricht, die ausgegeben werden soll, muß vorher gespeichert sein, dann wird ein Zeichen nach dem anderen ausgegeben und die Zahl mit dem Indexregister verglichen.

Wir haben bei der Ausgabe dazwischen ebenfalls viel Zeit zur Verfügung, normalerweise auch wieder mindestens 100 msec pro Zeichen, wenn wir eine Schreibmaschine mit 10 Zeichen pro Sekunde haben.

6.9 Datensichtgeräte

6.9.1 Aufbau

Auch hier existiert eine Tastatur; gegenüber der Schreibmaschine wird zur Anzeige anstelle des Schreibers eine Kathodenstrahlröhre verwendet. Darauf ist eine rechteckige Fläche mit den Abmessungen von jeweils Null bis X_0 und Y_0 ausleuchtbar, siehe Abb. 63.

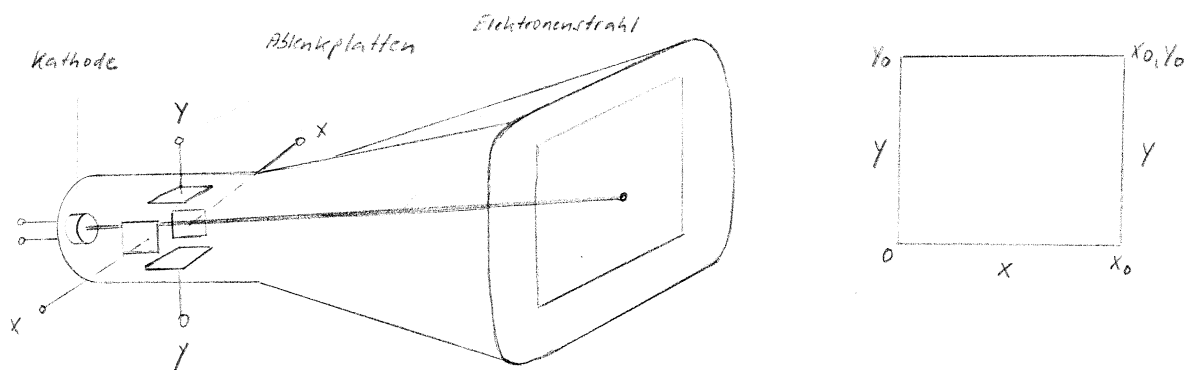


Abb. 63: Kathodenstrahlröhre für Datensichtgeräte

Die Anordnung für die Bilderzeugung zeigt Abb. 64:

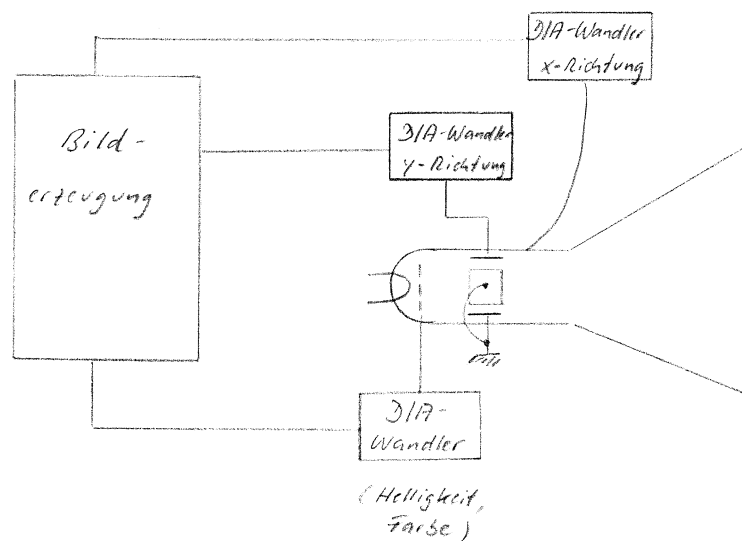


Abb. 64: Anordnung der Bilderzeugung

In die Kathodenstrahlröhre kommen die Zuleitungen über Digital/Analogwandler herein zur Bilderzeugung. Und zwar braucht man diese Wandler, um die Intensität, evtl. die Farbe des Geschriebenen bei Farbströhen zu bestimmen. Wir haben einen Wandler in der Y-Richtung und dann entsprechend in der X-Richtung. Diese Bilderzeugung stritt nun an die Stelle der Übersetzung in der Kontrolleinheit in Abb. 62.

Zusatzeinrichtungen sind Zeichengeneratoren und Lichtgriffel. Die Anordnung folgt in Abb. 65.

Wir haben eine Kontrolleinheit, für die Anzeige auf der Bildröhre. Die Verbindung geht von dort zum Rechner. Weiterhin existieren Puffer für die x- und y-Adressen, sowie für die Leuchtstärke. Der Zeichengenerator ist an die Kontrolleinheit angeschlossen. Der Lichtgriffel mit einem Schalter ist über einen Verstärker ebenfalls an die Kontrolleinheit angeschlossen.

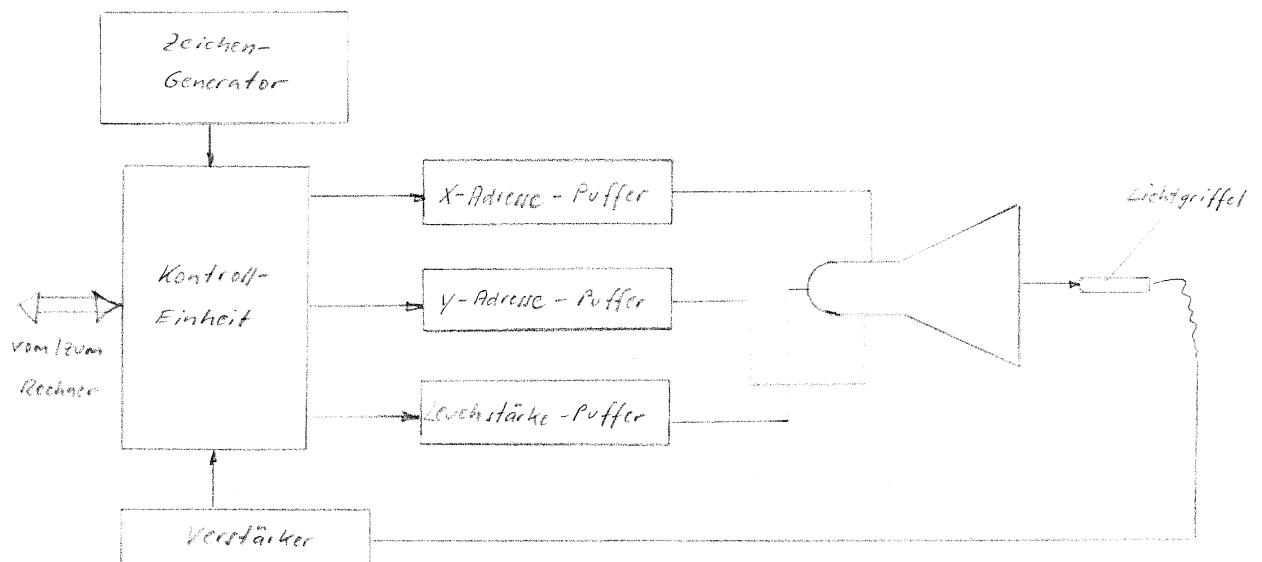


Abb. 65: Zusatzeinrichtungen

6.9.2 Bilderzeugung

6.9.2.1 Punktverfahren

Die Arbeitsweise dieser Kathodenstrahlröhre ist folgende: Der Elektroden-Strahl erzeugt einen Punkt auf der Bildfläche. Durch eine Änderung der Spannungen an den Horizontal- und Vertikal-Ablenkplatten kann der Punkt bewegt werden. Dies ist das Prinzip der Punkterzeugung. Aus vielen Punkten kann man dann ein Bild zusammensetzen, dies ist das sog. Punktverfahren. Außer diesem Punktverfahren werden andere, leistungsfähigere Verfahren verwendet, z.B. das Vektor-Verfahren, das Zeilenverfahren, und auch die Zeichengeneratoren.

6.9.2.2 Vektorverfahren

Ein Bild kann aus Vektoren gegebener Länge und Richtung zusammengesetzt werden. Es existiert eine obere Grenze der Zahl der möglichen Vektoren in einem Gerät, dies ist ein Entwurfskriterium für das Gerät. Außerdem sind spezifisch die Bildröhre, die Auflösung und die Größe des Wiederholungsspeichers. Die größte Zahl der Vektoren ist bestimmt durch die Größe des Wiederholungsspeichers. Normalerweise sind es 100 bis 1000 Vektoren, denn in dieser Größenordnung liegen die Kapazitäten dieser Wiederholungsspeicher.

Für die Darstellung der Vektoren sind folgende Daten nötig: Die Länge eines Vektors in x- und y-Richtung, dazu kommt die Farbe, die Helligkeit, die Wiederholungsrate und dann noch das Flackern. Einen Vektor kann man flackern lassen, dadurch, daß z.B. alle Sekunden der Vektor gezeigt wird. Die Daten, die dazu nötig sind, werden nun in dem Bildwiederholungsspeicher gespeichert, und zwar für jeden Vektor der dargestellt werden soll

in aufeinanderfolgenden Zellen. Dieser Bildwiederholungsspeicher wird nun zyklisch abgefragt, d.h., jede der Zeilen wird durchlaufen und steuert nun die Anordnung aus. Sobald die letzte Zeile durchgelaufen ist, fängt das Gerät wieder mit der ersten Zeile an, um das Bild stationär zu halten.

Diese Größen, die angegeben sind, werden über Digital-Analogwandler in analoge Größen umgesetzt, auf Integratoren zusammengefaßt und dann auf die Platten gegeben. Diese Spannungen steuern dann den Strahl.

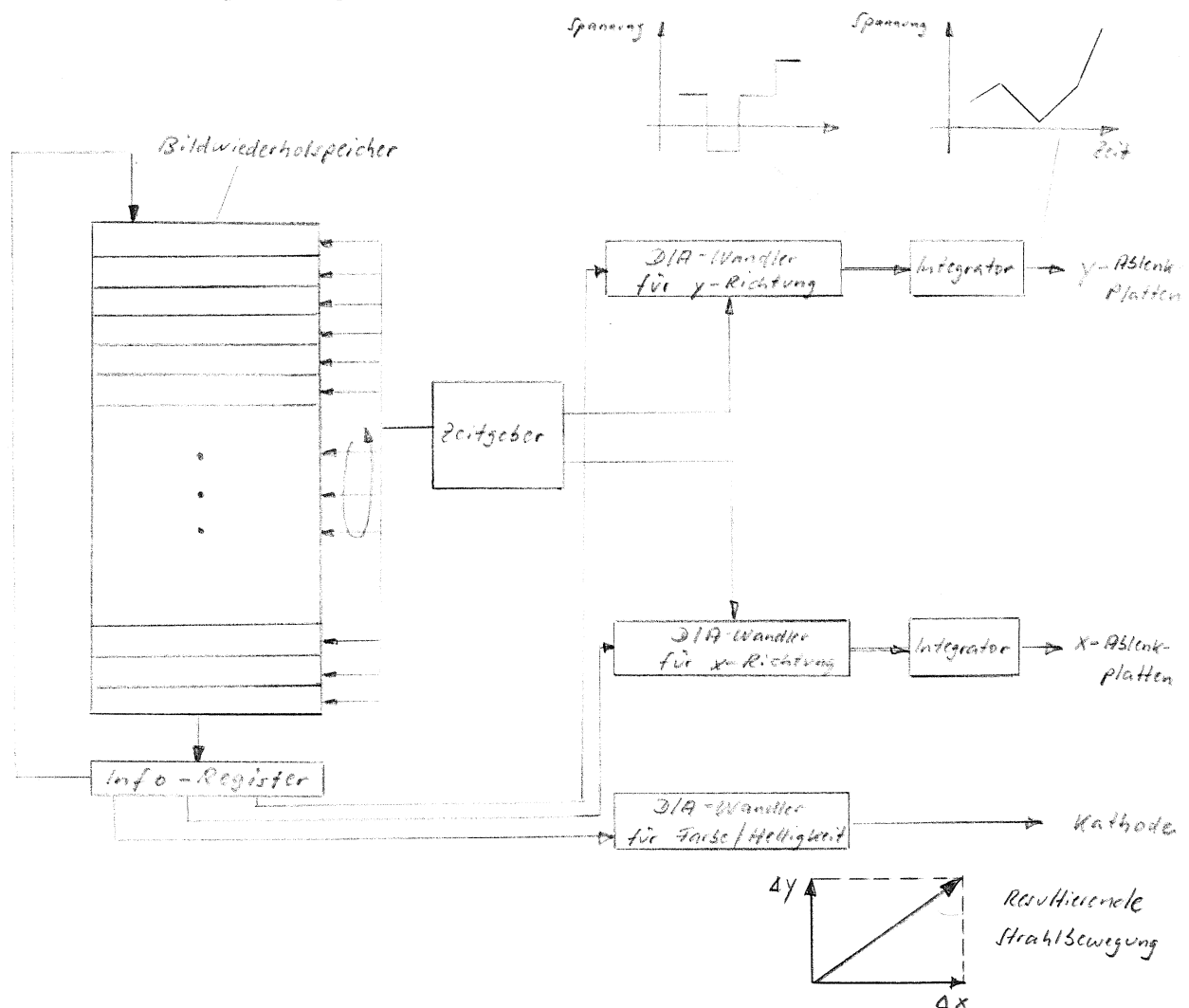


Abb. 66: Bilderzeugung mittels Vektorverfahren

In Abb. 66 ist eine solche Anordnung dargestellt.

Die Daten werden vom Bildwiederholungsspeicher abgegriffen. Durch einen Zeitgeber wird der Abgriff gesteuert. Der Zeitgeber hat auch Ausgänge in die Digitalwandler. Das Informationsregister enthält die gerade zu verwendenden Daten. Der Zeitgeber zeigt zu einem gewissen Zeitpunkt auf ein Datenwort und diese Daten werden in das Informationsregister eingelesen. Sie steuern nun zu diesem Zeitpunkt die Strahlenerzeugung über die Wandler für die Helligkeit und Farbe, sowie für die Strahlablenkung.

6.9.2.3 Zeilenverfahren

Das Zeilenverfahren ist wesentlich einfacher zu verstehen, weil es bereits vom Fernsehen bekannt ist. Allerdings kann es nicht so direkt verwendet werden, sondern es braucht eine Variation. Hier wird das Bild durch ein Zeilenraster erzeugt, das vom Elektronenstrahl durchlaufen wird. Das Raster wird durch zwei Zähler durch die x- und y-Richtung bestimmt, siehe Abb. 67.

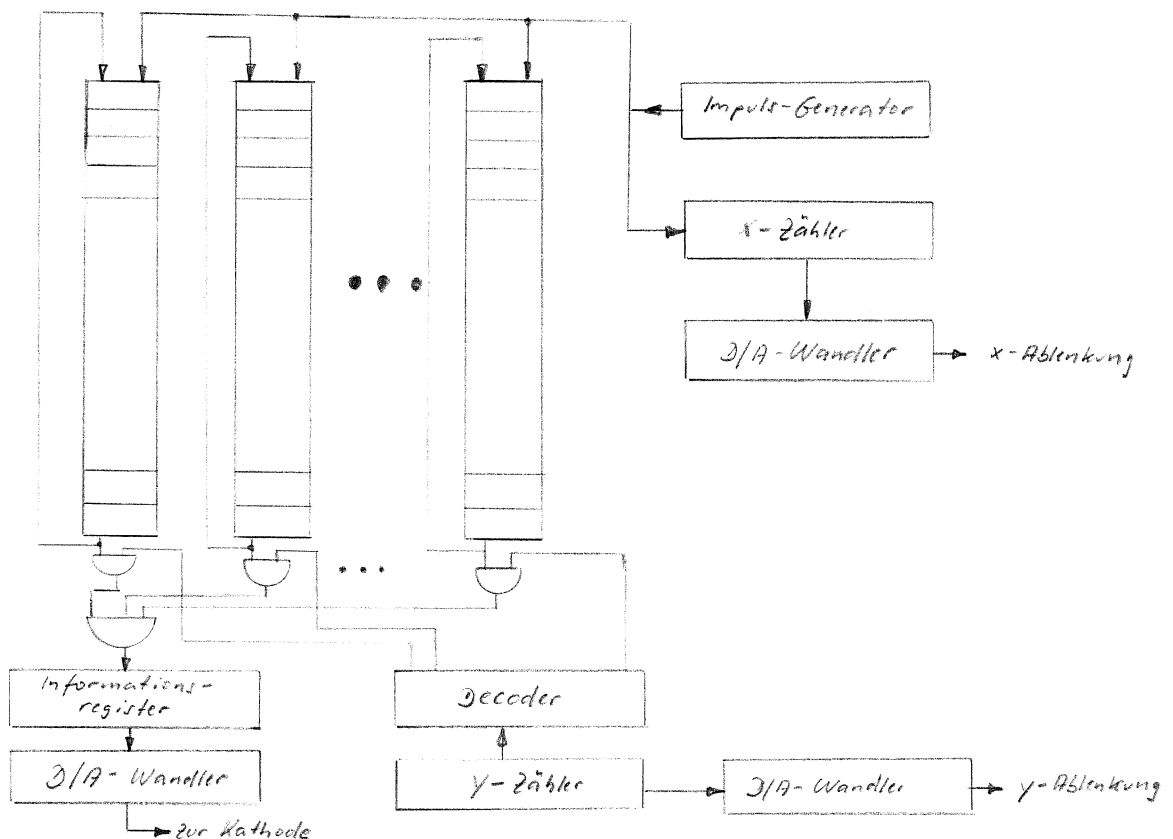


Abb. 67: Zeilenverfahren

In dem Bild erkennt man die Bildwiederholungspeicher für Helligkeit und Farbe. Die Koordinaten brauchen nicht mehr gespeichert zu werden, dafür ist aber jetzt ein Bildwiederholungspeicher für jede Zeile nötig. Z.B., wenn wir in sehr einfacher Weise nur eine Helligkeitsstufe haben, dann geben wir eine 1 bzw. 0. Wenn wir eine 1 geben, ist ein Punkt auf dem Raster da, dagegen bei 0 nicht. Bei einer größeren Zahl von Abstufungen sind entsprechend größere Speicher nötig. Auf jeden Fall braucht man pro Zeile einen Speicher. Die Ausgänge gehen nun in einen Impulsgeber. Der Impulsgeber bewirkt die Weiterschaltung von einer Zeile zur nächsten. Als Ausgänge existieren die x- und y-Zähler mit D/A-Wandlern. Wenn nun eine Zeile ausgewählt wird, dann haben wir anschließend eine kleine logische Schaltung. Die Ausgänge gehen in das Informationsregister für die Helligkeit und Farbe, und wiederum in einen Digital/Analogumwandler.

Der Nachteil dieses Verfahrens ist, daß sehr große Bildwiederholungsspeicher benötigt werden, z.B. bei nur Hell-Dunkelsteuerung und 1024 Punkten, benötigen wir für das rechteckige Bild mit 1024 x 1024 Bit, d.h., wir müssen 1.048.576 Bits bei einer einfachen Hell-Dunkel-Schaltung speichern. Dann sollte man wenigstens 3 Helligkeitsstufen haben. Der Speicherbedarf vervielfältigt sich sehr rasch, d.h., relativ große Bildwiederholungsspeicher werden gebraucht. Dies ist der Nachteil des Zeilenverfahrens. Warum sind beim Fernsehen nicht so viele Wiederholungsspeicher nötig? Beim Fernsehapparat bleibt das originale Bild verfügbar, z.B. bei einer Live-Sendung. Es kann dauernd abgegriffen werden, und was abgegriffen wird, wird in Echtzeit gesendet. Ähnlich bei Filmübertragungen. Hier haben wir aber das Bild nicht dastehen, es muß erst erzeugt werden, deshalb brauchen wir die Speicher.

Speicher sind nötig für die Vektoren, für die Raster oder dann, wenn ich Punkte erzeugen will, für die Koordinaten bei den jeweiligen Verfahren.

6.9.2.4 Zeichengeneratoren

Beim Zeichengenerator besteht eine Standardisierung auf bestimmte Bilder, d.h., wir wollen nur eine bestimmte Anzahl verschiedener Bilder erzeugen (die Zeilen). In diesem Fall können wir durch die Hardware eine Unterstützung bekommen, ohne daß alle Koordinaten oder die Raster gespeichert werden müssen. Diese Lösung erfolgt allerdings auf Kosten der Vielfältigkeit. Zur Erzeugung eines Zeichens wird der Bildschirm in eine Reihe von Punktmatrizen unterteilt. Jede dieser Punktmatrizen hat eine feste Zahl Punkte in Richtung der Achsen. Für jedes Zeichen kann man nun ein festes Unterprogramm vorsehen. Es existieren so nur Zeichen bestimmter Größen.

Die Zahl der Punkte in einer Matrix ist meist ein Vielfaches der Bitzahl im Wort des Rechners, zu dem der Bildschirm gedacht ist. Wenn das zugehörige Bit 1 ist, so wird der Punkt hier angesteuert, sonst nicht. Auf diese Weise ist es sehr einfach, ein Bild, ein festes Zeichen zu erzeugen. Dazu kommt noch ein Größenregister. Das Größenregister gibt den Abstand zwischen den Punkten und damit zwischen den Zeichen an (wenn der Abstand zwischen den Punkten größer wird, dann habe ich also ein größeres Zeichen). Eine Vorrichtung ist noch nötig, damit das letzte Zeichen in einer Zeile noch Platz findet oder, daß Teile vom Zeichen unterdrückt werden.

Jede der Punktmatrizen hat einen Referenzpunkt, meist der linke oberste Punkt, der dann die Lage des Zeichens angibt. So sind dann nur die Koordinaten dieses Punktes nötig, sowie ein Code, um ein Zeichen zu bestimmen.

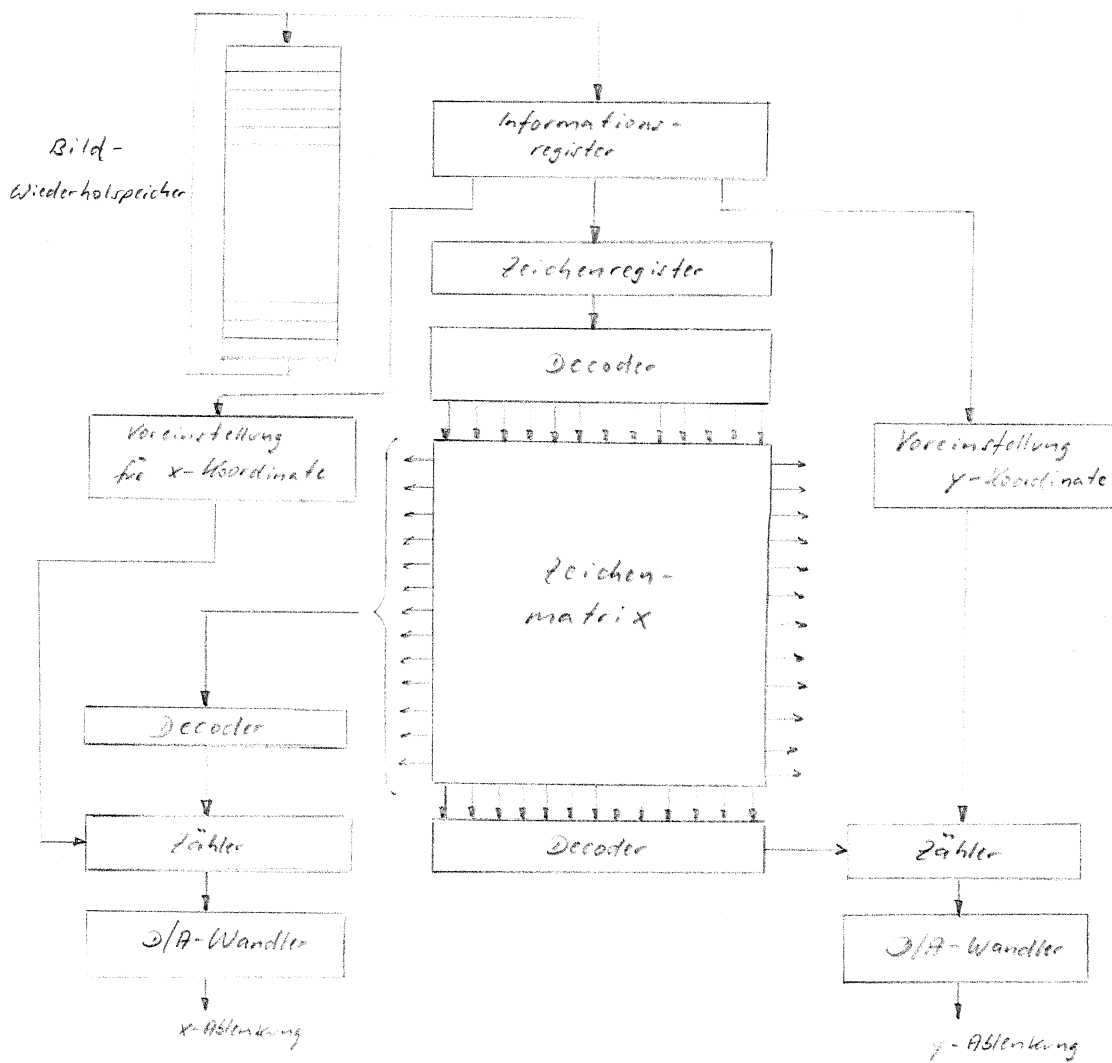


Abb. 68: Zeichenerzeugung

Wir haben in Abb. 68 wiederum einen Wiederholungsspeicher und daraus gehen die Daten in ein Informationsregister. Das Informationsregister hat 3 Ausgänge, einer davon in das Zeichenregister. Der Inhalt des Zeichenregisters muß dekodiert und dann in die Zeichenmatrix herausgeführt werden. Dann existiert noch der Ausgang in die Voreinstellung für die x-Koordinate in einem Zähler, d.h., das ergibt dann die Stellung für die x-Koordinate des linken obersten Punktes des Zeichens. Dazu kommen Ausgänge aus der Zeichenmatrix, ähnlich erfolgt die Auslenkung für die y-Koordinate. Weitere Ausgänge aus der Zeichenmatrix ergeben die Helligkeit der einzelnen Punkte des entsprechenden Zeichens.

Diese Zeichenmatrix ist ein fester Speicher, denn Zeichen ändern sich normalerweise nicht. Entweder ist es

ein Halbleiterspeicher oder ein Kernspeicher. Ein sehr billiger Speicher reicht hier schon aus. Das Interessante hierbei ist, daß nicht nur die standardisierten Zeichen erzeugt werden können, sondern es können auch noch eigene Zeichen entwickelt werden. Besondere Instruktionen greifen den Wert des Akkumulators als Intensitäten für die Punkte einer Zeichenmatrix heraus und projizieren das Zeichen.

6.9.3 Programmierung der Bilderzeugung

6.9.3.1 Zeichengenerator

Wir haben dafür folgende Instruktionen: Einmal das "Lade das Format", d.h., die Zeichengröße, Intensität, Abstand zwischen den Zeichen und das Weiterschalten um die Zeichengröße und dann eventuell noch eine symbolische Repräsentation des Zeichens. Falls ich mit einer Zeichenmatrix die Zeichen generiere, wird das Symbol dekodiert und dann dadurch das Zeichen generiert. Dann haben wir gesehen, daß das nicht unbedingt nötig ist, wir können selbst Zeichen generieren die in der Symbol-Matrix nicht existieren, d.h., andere als standardisierte Zeichen durch Zusammensetzen des Inhalts, z.B. des Akkumulators. Dabei haben wir dann z.B. "Plotte linken Teil des Zeichens", dabei bezieht sich dann die Zeichengröße, Intensität, Abstand, usw. alles auf diese Teile eines Zeichens, die hier erzeugt werden. Dann muß ich warten, nach dem Zeichen des linken Teils, dann plotte ich den mittleren Teil, muß wieder warten, usw, bis ich dann den rechten Teil plotte. Hier muß ich ebenfalls warten und dann kommt noch folgendes hinzu: Nachher muß ich unter Umständen noch einen Zwischenraum plotten. Natürlich müssen hier spezielle Instruktionen gegeben sein, "Plotte den linken Teil" ist eine Instruktion, "Plotte mittleren Teil",

oder "Plotte rechten Teil", usw. genauso, ebenso ist der Zwischenraum eine Instruktion, d.h., durch die Änderung der Befehlscodes kann ich dann entsprechend andeuten, was ich und wo ich plotten möchte.

6.9.3.2 Punktverfahren

Alle Bilder, die wir auf dem Bildschirm erzeugen möchten, werden durch eine fortlaufende Reihe von Punkten erzeugt. Hierbei läuft die Programmierung folgendermaßen:

1. Leuchtstärkeregister laden,
2. Löschen des Anzeigeflags,
3. Laden der x-Koordinate für den Punkt der erzeugt werden soll,
4. Laden der y-Koordinate
Dafür müssen unter Umständen 2 Instruktionen vorgesehen werden, falls der Akkumulator als Zwischenspeicher verwendet werden muß. Dies hängt von den Gegebenheiten des Geräts ab.
5. Anzeige, zeichne den Punkt, der durch die Daten vorher gegeben worden ist
6. Warten auf das Anzeigeflag, d.h. durch das Anzeigeflag wird angedeutet, daß der Punkt nun erzeugt worden ist, genauso wie das Warten bei dem Zeichengenerationsverfahren wird es in derselben oder ähnlichen Weise von statten gehen.
7. Springe nach 6.

Nach 6 werden dann weitere Punkte zu erzeugen sein, falls wir eine Linie oder irgendein Bild erzeugen möchten. Die weiteren Punkte werden meist durch Indexregister angesteuert, d.h., ich habe die Werte der x- und y-Koordinaten im Indexregister, die sich sehr rasch erhöhen lassen und dabei kann ich dann

sehr einfach die Änderung dieser Koordinaten hervorrufen, falls ich eine Linie erzeugen möchte. Je nach der Übertragungsrate hier bei dem Punktverfahren, z.B. haben wir 20, 50 oder 100 kHz Übertragungsrate, muß auch entsprechend lang hier in 6) gewartet werden. Z.B. bei 20.000 Zeichen/s muß ich 50 μ s warten, bei 50 K muß ich 20 μ s warten bei 100 K muß ich 10 μ s warten, bis ich das nächste Zeichen erzeugen kann.

Wie wird nun ein Bild auf dem Bildschirm erscheinen? Das Bild wird erzeugt durch das Nachleuchten der Phosphorschicht. Dies war das Punktverfahren.

6.9.3.3 Lichtgriffel

Dieser Lichtgriffel ist prinzipiell ein Röhrchen, das eine Fotozelle enthält und einen Schalter, d.h., eine Blende, die durch einen Schalter betätigt wird, um dann das Licht auf die Fotozelle fallen zu lassen, wenn es zu einem gewissen Zeitpunkt gewünscht wird. Wir haben eine lichtempfindliche Diode, die eben den Strom erzeugt, wenn Licht darauf fällt. Der Strom muß dann verstärkt werden, um ausreichend zu sein. Die Programmierung des Lichtgriffels ist entsprechend einfach. Wir haben nur eine Skip-Instruktion. D.h., wenn Licht auf die Fotozelle kommt, und der Fotostrom erzeugt wird, steuert dieser Strom ein Flag in dem Datensichtgerät. Wie verwenden wir nun diesen Lichtgriffel?

Die Arbeitsweise ist folgendermaßen (s. Abb. 69), z.B. ich habe einen Raster auf dem Bildschirm, z.B. 9 Flags. Diese Flags kann ich, falls ich einen Zeichengenerator zur Verfügung habe, durch einen Zeichengenerator sehr einfach erzeugen. Andernfalls muß ich eben dann durch das Punktverfahren diese Punktfolge

erzeugen und dann jeweils einen Block erzeugen, usw.

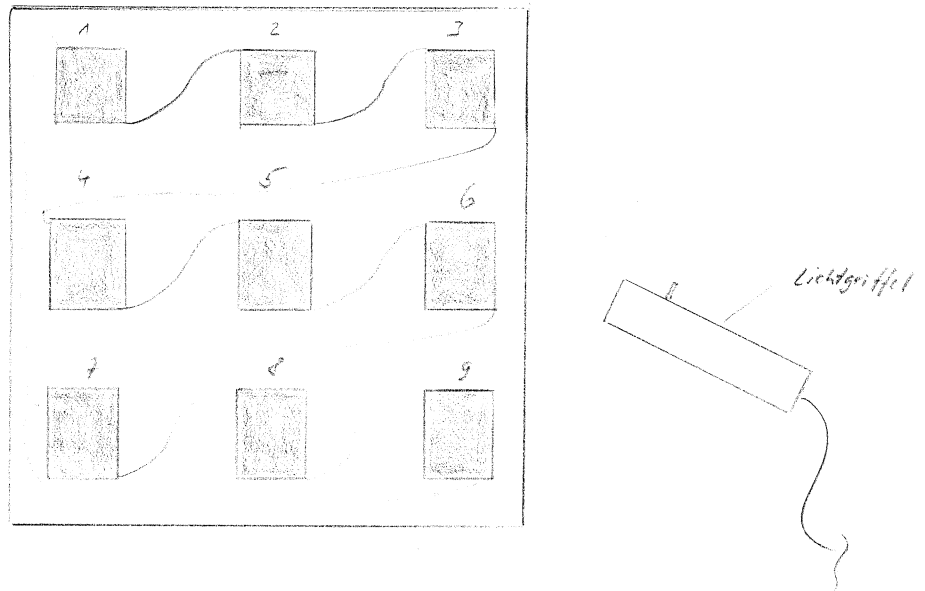


Abb. 69: Rastererzeugung

Die Hauptfrage ist nun, wie kann ich feststellen, daß ich mit meinem Lichtgriffel auf diesen Punkt 6 zeige. Wie wird festgestellt, daß der Lichtgriffel auf Punkt 6 zeigt? Die Antwort darauf ist sehr einfach, was wir tun müssen ist folgendes: In dem Programm, in dem die Punkte erzeugt werden, wird nach jedem Punkt gefragt, ob das Griffelflag da ist. Falls es da ist, dann heißt das, daß der Lichtgriffel auf dieses Flag gezeigt hat. Nun, eine Schwierigkeit existiert beim einmaligen Erzeugen, denn ich habe die Folge 1, 2, 3, 4, usw., dann kommt 6. Nun, Punkt 6 kommt erst dann auf, wenn 1, 2, 3, 4, 5 gezeichnet sind, d.h., erzeugt worden sind, dann erst wird Punkt 6 gezeichnet. Wenn ich nun nachprüfe, nach Punkt 6, ob ein Flag da ist, wird dieses Flag zu diesem Zeitpunkt erscheinen. Punkt 7, 8, 9 werden nachher gezeichnet, wo dieses Flag schon da ist, d.h., dies wird uns nicht mehr bei einmaliger

Erzeugung berühren. Wir müssen, nachdem wir Flag 9 erzeugt haben, zur Erzeugung der früheren Punkte zurückkehren.

Nun gibt es verschiedene Möglichkeiten, diese Punkte zu erzeugen, nehmen wir an, z.B., punktweise. Ich kann auch linienweise erzeugen, das ist aber nicht sehr gut, und wie Sie sofort sehen werden, aus folgenden Gründen: ich werde vorzugsweise so vorgehen, daß ich erst die Punkte im Block 1 erzeuge, dann zum Block 2 gehe, dann nach Block 3 gehe und dann nach Block 4 usw., und dann wieder von vorne anfangen.

Nun, die Abfragen haben wir immer als letzte Instruktion. Wenn ich einen Block erzeugt habe, frage ich nun nach, ob das Flag erschienen ist. Wenn das Flag erschienen ist, dann kann ich wieder eine Instruktion überspringen, d.h., ich kann herauspringen und irgendwelche Aktionen durchführen. Die Intensität dieses Bildes hängt von der Dichte der Punkte ab und der Übertragungsrate: $I = D \times R$, Dichte \times Übertragungsrate.

Die Arbeitsweise des Lichtgriffels mit dem Erkennen des Flags ist im allgemeinen etwas schwierig.

Betrachten wir dazu Abb. 70.

In diesem Diagramm haben wir die Intensität als Ordinate und die Zeit als Abszisse.

Beginnen wir mit Block 1, wir haben zuerst die Intensität für Block 1. Sie wissen, wir haben Nachleuchten, d.h., der Schweif dieses Blocks zieht sich etwas nach rechts heraus. Und nun, irgendwann nachdem ich dies erzeugt habe, möchte ich Punkt 2 erzeugen. Dieser hat ebenfalls Nachleuchten, dann Punkt 3, Punkt 4 usw.

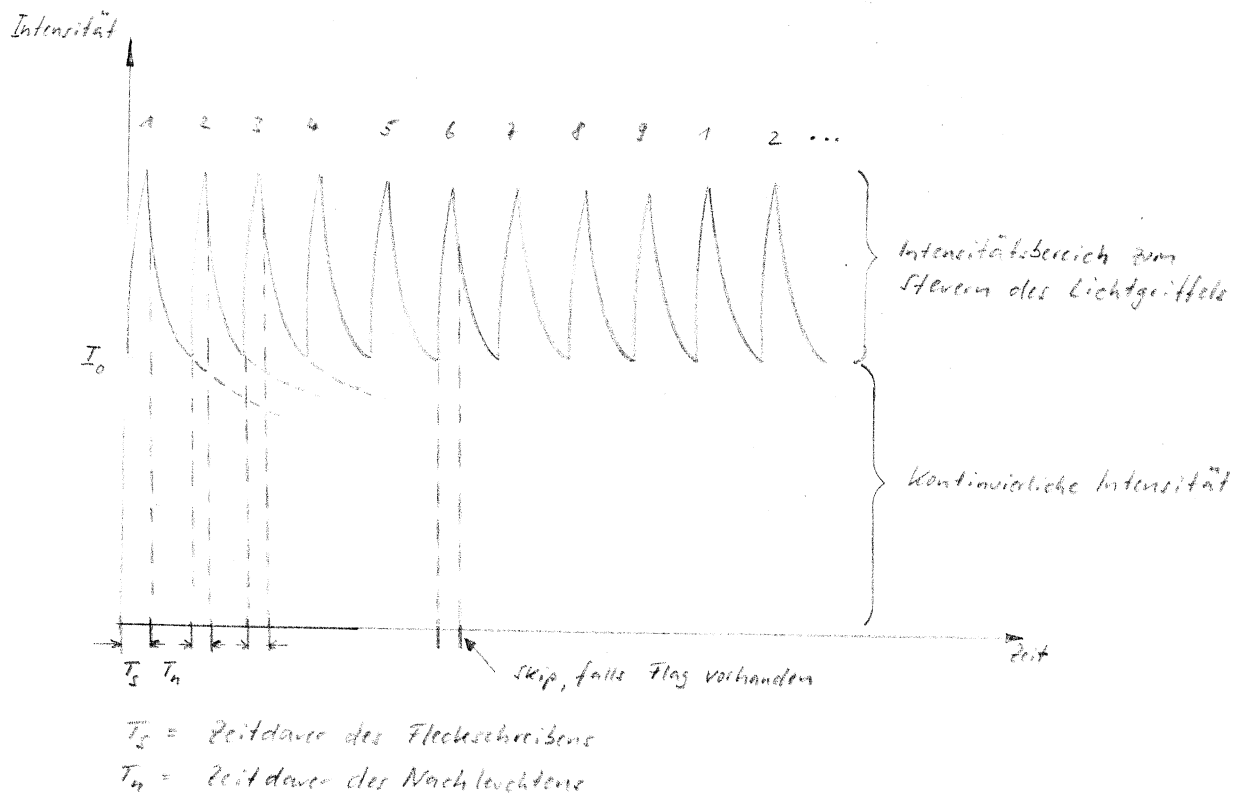


Abb. 70: Intensitätsverlauf bei der Rastererzeugung

Also, wir haben die Folge 1, 2, 3, 4, 5, 6, 7, 8, 9, 1 usw., hier wiederholt sich nun dasselbe in einem Zyklus. Unter den Schnittpunkten die, falls ich mit derselben Intensität arbeite, alle auf einer Linie liegen, zwischen der Intensität 0 und der Intensität I_0 , habe ich kontinuierliche Intensität, d.h., hier habe ich das Nachleuchten aller Punkte. Und nun muß ich die Frequenz der Erzeugung so klein halten, daß vom Zeitpunkt 0 bis zum Zeitpunkt t_g weniger als 1/16 Sekunde vergeht, dann wird das Bild stehend erscheinen. Nun haben wir die Intensitäten I_1, I_2, \dots, I_9 , die in etwa alle gleich groß sein sollen. Natürlich bleibt es mir unbelassen, die Intensität zu ändern. Ich kann irgendeinen dieser Blocks mit der Intensität herausheben; z.B., wenn ich erkannt habe, daß mein Lichtgriffel auf diesen Punkt zeigt, könnte ich von diesem Moment an durch Abprüfen

des Griffelflags die Intensität dieses Blocks erhöhen und dadurch diesen Block von den anderen unterscheiden oder ich könnte z.B. Flackern erzeugen. Immerhin, das Wichtige ist, daß ich hier zwischen I_1 und I_0 diese Intensität habe. Diesen Bereich der Intensität kann ich zum Steuern des Lichtgriffels verwenden. Die Zeit t_0 bis t_1 ist die Zeit des Schreibens von Block 1, t_1 bis t_2 ist die Zeit des Schreibens von Block 2, usw. Diese Zeiten verwende ich zum Steuern des Lichtgriffels. Am Ende der Erzeugung habe ich jeweils das Abfragen des Griffelflags. D.h., ich frage hier zum Zeitpunkt t_6 in unserem Beispiel das Flag ab, nachdem der Block 6 erzeugt worden ist, d.h., in diesem Moment habe ich Block 6 erzeugt und nun frage ich in meinem Programm, ob das Flag da ist. Wenn ich den Schalter betätigt habe, fällt Licht auf die Fotozelle und die Intensität erzeugt nun ein Flag, das ich nun abprüfe auf Punkt 6 und nun kann ich herausspringen.

Wie Sie hier sehen, reicht es insgesamt noch nicht aus, ich muß auch noch die Intensität des Lichtgriffels, d.h., die Empfindlichkeit des Lichtgriffels einstellen, und zwar so einstellen, daß nur die stärksten Impulse, I_0 plus ΔI das Griffelflag erzeugen können. Andernfalls, wenn ich mit meiner Empfindlichkeit im unteren Bereich arbeite, d.h., wenn das Flag so empfindlich ist, daß sehr geringe Intensitäten es aussteuern können, dann kann ich nicht mehr feststellen, wohin ich zeige. Bei zu großer Empfindlichkeit des Lichtgriffels würde über einen großen Bereich hinweg das Flag erzeugt. Die Empfindlichkeit ist eine sehr wesentliche Größe, die hier eingeht.

Im Programm, das das Muster erzeugt, wird nach jedem Zeichen auf das Flag abgefragt, wenn das Griffelflag da ist, dann heißt dies, daß der Griffel auf das verlangte Zeichen gezeigt hat, oder der andere Fall, daß

die Empfindlichkeit des Lichtgriffels zu groß war. Wenn nicht, bedeutet dies für unser Programm, daß der Griffel woanders hinzeigte, oder daß die Empfindlichkeit, d.h., die Intensität des Verstärkers für die Fotozelle, zu gering war. Auf diese Weise kann ich nun sehr elegant mit dem Lichtgriffel arbeiten.

7. Satelliten-Rechner

Die meisten Datenstationen arbeiten auf Zeichenbasis, d.h., wir geben Zeichen ein, die dann in die Datenstation gehen, und ebenso wird in der Ausgabe ein Zeichen von der Datenstation angezeigt oder gedruckt. Im Dialogbetrieb haben wir normalerweise sehr viele Stationen und dabei wird dann die Arbeitsweise unwirtschaftlich. Deshalb werden Satellitenrechner dazwischen geschaltet, und zwar zwischen das Rechnersystem mit den Kanälen und den Datenstationen. Der Zweck dieser Satellitenstationen ist, daß ganze Sätze von Zeichen übertragen werden und nicht mehr die Zeichen einzeln. Betrachten wir dazu Bild 71; wir haben den Satellitenrechner mit den Terminals T_1, T_2, \dots, T_n . Zwischen dem Rechnersystem und dem Satellitenrechner haben wir die Übertragung von Sätzen von Zeichen, was durch einen Block angezeigt wird, und dann von dem Satellitenrechner zu den Terminals, den Datenstationen, haben wir nun jeweils zeichenweise Übertragung.

Dadurch wird eine Entlastung des Zentralrechnersystems und des Hauptspeichers erreicht. Eine Entlastung der Kanäle wird dadurch erreicht, daß weniger oft unterbrochen werden muß, und eine Entlastung des Hauptspeichers wird dadurch erreicht, daß der Satellitenrechner selbst Pufferspeicher besitzt. Diese Satellitenrechner sind im allgemeinen Minicomputer.

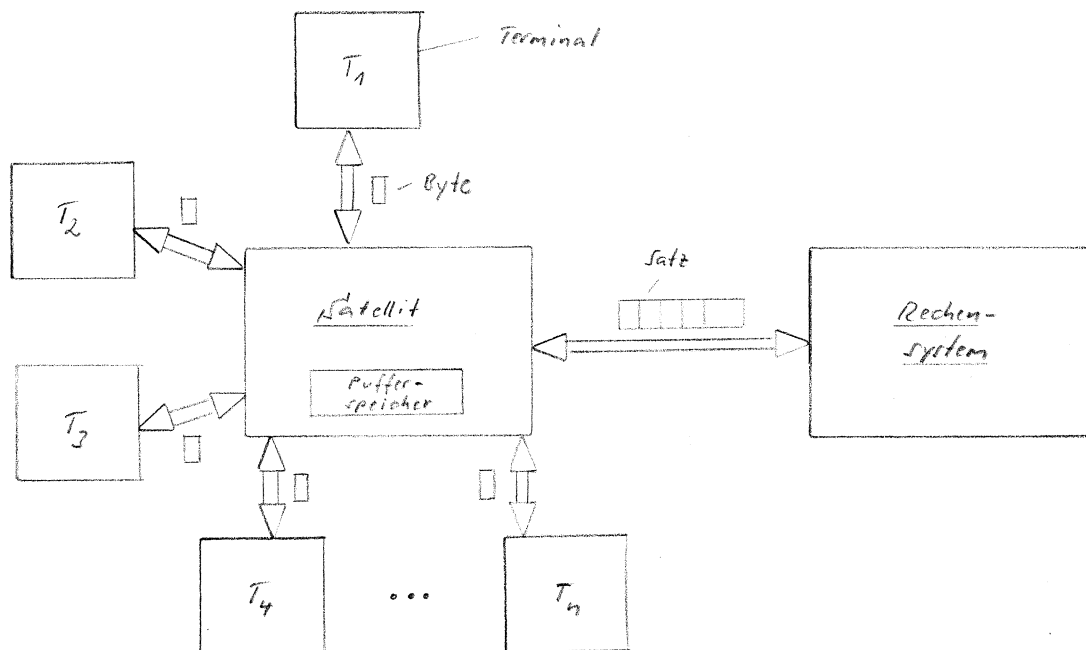


Abb. 71: Zusammenwirken zwischen Rechnersystem und Satelliten

Die Übertragung von Zeichen erfolgt folgendermaßen:

1. Vom Zentralrechner zum Satelliten, wenn Daten bereit liegen für eine Station; d.h., nur dann wird der Zentralrechner mit dem Satelliten arbeiten, wenn vom Zentralrechner aus Daten übertragen sind
2. Zum Zentralrechner vom Satelliten, nach Anzeige des Satelliten, daß Daten bereit liegen.

Die Daten werden übertragen, sobald der Zentralrechner Zeit hat. Unter Berücksichtigung der Priorität in dem System können wir uns folgende Arbeitsweise vorstellen. Einmal kann es durch das Unterbrecherwerk, über das wir noch sprechen werden, oder über ein Semaphorensystem realisiert werden. Wir haben Nachrichtenanzeigen, die angeben, ob etwas vorliegt. Diese Anzeigen werden dann vom Rechner periodisch abgefragt.

8. Plotter

8.1 Analog-Plotter

Plotter werden sehr häufig als Ausgabegeräte gebraucht. Sie sind nur Ausgabegeräte für Digital- und Analog-Systeme, wenn Daten als Diagramm oder als Figuren darstellbar sind. Die allgemeine Charakteristik der Plotter ist folgende: Plotter sind sehr langsam, und aus diesem Grunde wird ein Plotter meist im off-line-Betrieb betrieben und mit einem Magnetband als Zwischenspeicher verwendet. Das Magnetband als Zwischenspeicher hat sich hier sehr bewährt und zwar deshalb, weil die Daten nur sehr langsam verlangt werden, d.h., die Daten von dem Plotter werden nur in sehr großen Zeitabständen verwendet, und außerdem immer nur seriell, deshalb eignet sich das Magnetband dafür ausgezeichnet. Nun zur Arbeitsweise und zum Aufbau: Wir haben zwei Typen von Plottern zu unterscheiden, einmal analoge Plotter und digitale Plotter. Beim analogen Plotter werden die Spannungswerte in eine Brückenschaltung eingegeben, wobei die Differenzspannung, die auftritt auf Nachstellmotoren für den Zeichenstift gegeben wird. Je größer die Differenzspannung ist, desto größer ist der Strom für den Nachstellmotor, und der Nachstellmotor läuft damit rascher. Die Bewegung des Zeichenstifts ist mit einem Kompensationspotentiometer gekoppelt, das in dieser Brücke arbeitet (siehe Abb. 72). Wir haben das Kompensationspotentiometer, das in dieser Brücke liegt und die Speisespannung für die Brücke U_0 . Weiter haben wir die unbekannte Spannung U_x , d.h., die Spannung, die geteilt werden soll, bzw. die Größe, um die der Zeilenstift bewegt werden soll, und zwar ist dazwischen ein Verstärker geschaltet, dessen Ausgang auf das Kompensationspotentiometer geht. Von dem Verstärker aus geht es auf die Nachstellmotoren. und diese Motoren steuern den Zeichenstift, der mit dem Arm des Kompensationspotentiometers gekoppelt ist, d.h.,

je mehr der Zeichenstift bewegt wird, um so mehr wird hier die Spannung auf dem Kompensationspotentiometer verändert, bis die Spannung U_x zu Null wird. Wir haben zwei dieser Brückenschaltungen und zwar für die x-Koordinate sowie für die y-Koordinate.

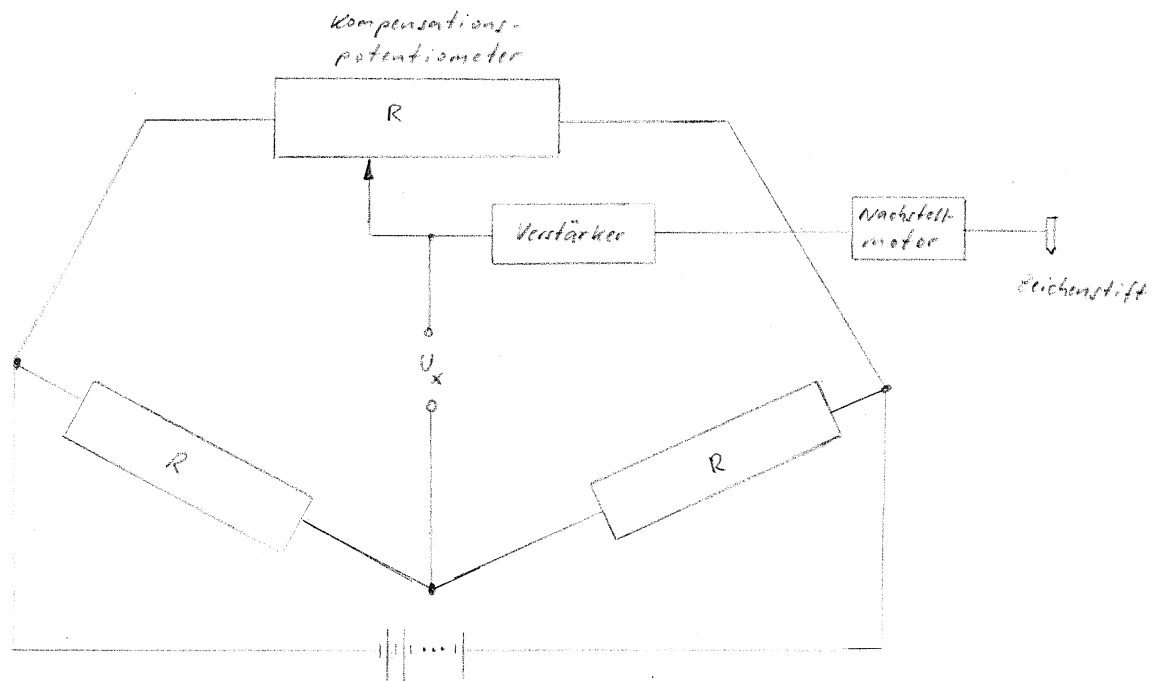


Abb. 72: Prinzip der Steuerung beim Analog-Plotter

8.2 Digital-Plotter

Die Digital-Plotter haben Schrittmotoren, die den Zeichenstift in inkrementalen Weglängen bewegen. Ein Inkrement ist normalerweise so groß wie die Schriftbreite, 1/10 mm pro Impuls. Mit einem Impuls an den Schrittmotor wird der Zeichenstift um 1/10 mm bewegt. Größere Wegstrecken werden durch Zähler bestimmt, die den Inhalt über variable Impulsfolgefrequenzen an die Motoren weitergeben. Damit lassen sich alle möglich Zeichen und Fi-

guren darstellen. Es existiert wieder jeweils ein Motor für die x- und für die y-Richtung. Damit können auch wiederum alle Zeichen und Figuren dargestellt werden. Statt der Schrittmotoren werden zuweilen auch Motoren mit konstanter Drehzahl und Zeitzählung verwendet. Dabei wird dann das Schreiben durch eine Zeitzählung gesteuert, d.h., ich habe den Motor mit konstanter Drehzahl über eine gewisse zeitlang angeschaltet, und durch die Zeitzählung wird eben nicht der Motor in inkrementalen Weglängen, sondern in kontinuierlichen Weglängen bewegt. Wenn ich sehr kleine Wegstrecken beschreiben möchte, überwiegen die Vorteile des inkrementalen Schreibens, und wenn ich große Zeichnungen, große Wegstrecken bewegen möchte, dann haben die Motoren mit konstanter Geschwindigkeit ihre Vorteile und zwar deshalb, weil dann das Schriftbild nicht so oft unterbrochen werden muß. Man kann leicht anhand einer Zeichnung feststellen, mit welchem Verfahren es hergestellt worden ist. Bei digitalen inkrementalen Plottern finden Sie, daß die Striche immer ein klein wenig zackig verlaufen, wogegen die Verfahren mit konstanter Geschwindigkeit glatte Weglängen erzeugen, jedoch an den Ecken etwas abhaken, und zwar bedingt durch das Nachhinken der Motoren.

9. Kanalwerke

9.1 Grundproblem

Das Problem bei der Datenübertragung ist folgendes (siehe Abb. 73): Einmal ist die Datenübertragung zwischen E/A-Geräten und den Speichern sehr langsam, außerdem sind die Datenquanten, die übertragen werden, verschieden. In Abb. 73 haben wir z.B. den Zentralrechner, dann einen Pufferspeicher. In diesem Pufferspeicher ist gerade ein Zeichen mit dem ich arbeite. Die Daten im Speicher sind alle von einer bestimmten Größe, d.h., in Wörtern oder Doppelwörtern. Ich arbeite nun in diesem Moment mit diesem Zeichen, das von einem E/A-Gerät über ein Medium kommt, z.B., einen Lochstrei-

fen. Hier bei der Übertragung zwischen E/A-Geräten und Zentralrechner haben wir ein Zeichen, bei der Übertragung vom Zentralrechner zum Speicher haben wir ein Aggregat von Zeichen, z.B. ein Wort oder ein Doppelwort usw. Es gibt noch ein Adressregister, das die Adresse angibt, wo das Aggregat, wenn es voll ist, abgespeichert werden muß. Der Pufferspeicher ist nichts anderes als ein Sammel- und Verteilerregister, wo die Zeichen herausgegriffen werden aus den Aggregaten, wenn vom Speicher zum Zentralrechner und vom Speicher zum E/A-Gerät übertragen wird. Dieses Zeichen wird herausgegriffen und dann aus- oder eingegeben.

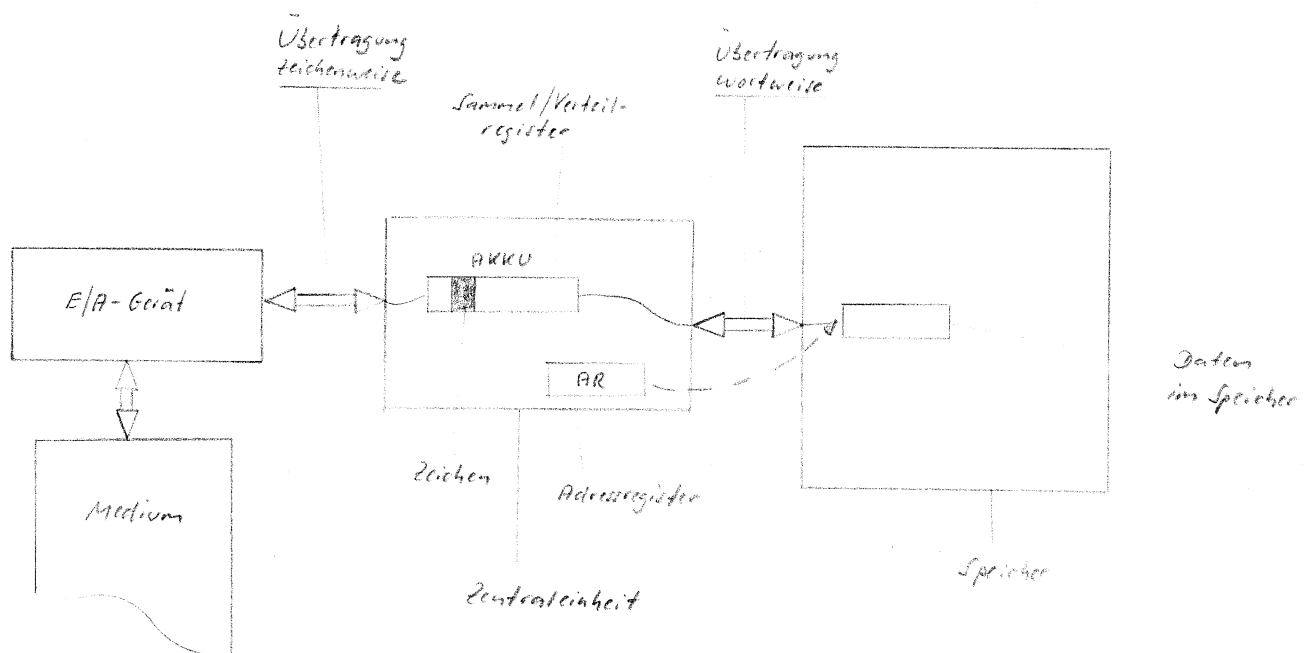


Abb. 73: Datenübertragung zwischen E/A-Geräten und Speicher

Die Information, die ich für diese Übertragung von Daten brauche, ist folgende:

1. die Anfangsadresse der Daten im Speicher,
2. die Art der Operation, die stattfinden soll, z.B. E/A,

3. Angabe des Geräts,
4. Zahl der Wörter oder Zeichen, die übertragen werden sollen,
5. Angabe über die Fortsetzung, d.h., wie und wo fortgesetzt werden soll, nach Vollendung dieser Anforderungen.

Diese Arbeiten wurden seither vom Zentralrechner übernommen, und nun tritt das Problem auf, daß weiterhin bei relativ schnellen E/A-Geräten und langsamen Zentralrechnern, oder bei mehreren E/A-Geräten und langsamen Zentralrechner, der Zentralrechner nur für die Datenübertragung arbeitet. Dies ist natürlich für die Arbeit eines Systems und die Gesamtleistung sehr unwirtschaftlich. Wir können aber auf die Daten nicht verzichten. In der übrigen Zeit sollte der Rechner rechnen, das kann er aber nicht tun, wenn er keine Daten hat. Deshalb haben die Daten generell Priorität vor dem Rechnen, und nun da wir Datenpriorität haben, müssen die Daten natürlich hereinkommen, aufbereitet werden. Wir haben ein Beispiel gesehen, wo der Zentralrechner eines relativ großen Rechners nur für die Datenübertragung arbeitet. Z.B. bei nur einem Magnetband mit 320 kBytes Übertragungsrate, mit einer IBM 360 Modell 50. In diesem Fall überträgt der Zentralrechner fast nur noch Daten. Wenn nun der Zentralrechner nicht nur die Wörter übertragen muß, sondern auch noch die Zeichen, wird dieses Problem bei einer Größenordnung kleinerer Datenübertragungsraten auftreten. Deshalb müssen wir auf dieses Problem eingehen. Die Lösung hierzu ist das Kanalwerk. Wir haben bereits in der Einführung darüber gesprochen, daß auch das Kanalwerk das Problem nur teilweise und nicht vollständig lösen kann, nun betrachten wir noch Einzelheiten. Das Kanalwerk übernimmt die Datenübertragung relativ unabhängig vom Zentralrechner. Die eigentliche Datenübertragung ist vollständig unabhängig vom Zentralrechner, jedoch

muß der Zentralrechner noch das Kanalwerk beaufsichtigen.

Wir haben folgende Aufgaben des Kanalwerks:

1. Befreiung des Zentralrechners von der Datenübertragung,
2. Time-Sharing der Systemkomponenten für paralleles Arbeiten,
3. Datenpufferung .

Es werden für E/A-Vorgänge mehrere Speicherbereiche verwendet. Wenn mehrere Speicherbereiche für eine Datenübertragung verwendet werden, spricht man von Datenpufferung.

Unter den Kanalwerken unterscheiden wir zwei Arten, Standardkanäle und Schnellkanäle.

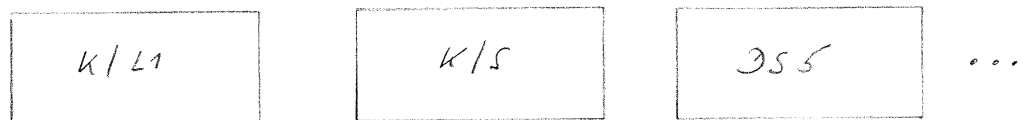
9.2 Standardkanäle

Dieser Standardkanal ist normalerweise der erste Kanal, der an ein System angeschlossen ist. Wenn eine Konfiguration zusammengestellt wird, wird i.a. zuerst ein Standardkanal angeschafft, und zwar deshalb, weil der Standardkanal mehrere Unterkanäle besitzt, z.B. bei den Modellen IBM 360, 370 haben die Kanäle bis zu 256 Unterkanäle. Diese Standardkanäle sind gedacht für langsame Einheiten, z.B. Kartenleser, Schnelldrucker, Datenstationen usw., wobei wir als langsame Einheit solche Einheiten verstehen, die etwa eine Datenübertragungsrate der Größe 10^4 bits/s erzeugen. Diese Einheiten teilen sich nun den Kanal auf folgende Weise (siehe Abb. 74):

Wir haben einen Block mit der Bez. K/L1, dieses Zeichen ist von der Karteneinheit des Lesers L1, das nächste Zeichen ist von der Karteneinheit Stanzen S, das nächste von der Datenstation DS5 kommend, und ein weiteres Zeichen K/L2, das ist ein Zeichen, das vom Kartenleser L2 stammt, usw.

Dies ist das sog. Multiplex-Verfahren, d.h., diese Einheiten "teilen" sich den Kanal. Dies bedeutet aber nicht, daß der Standardkanal nur für langsame Einheiten verwendet werden kann, der Standardkanal kann auch für schnelle Einheiten verwendet werden, z.B. ein Magnetband, eine Trommel, eine Platte, usw., also schnelle Einheiten mit etwa 10^7 bits/s. Wir haben also hier 3 Größenordnungen Unterschiede. Wir haben bis zu 256 Unterkanäle, d.h., wenn wir die Gesamtkapazität eines Kanals auf etwa 10 Mio Bits/s veranschlagen, dann muß man pro Unterkanal etwa 2 Größenordnungen darunter bleiben.

Multiplex-Verfahren:



"Burst"-Verfahren:

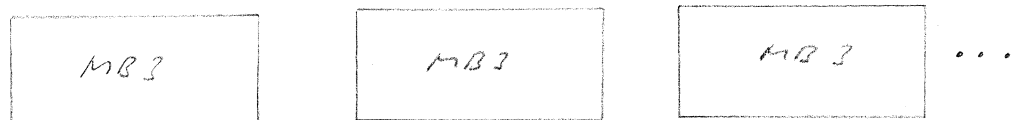


Abb. 74: Datenfluß bei den verschiedenen Übertragungsverfahren

In diesem Fall ist der Zeichenfluß folgendermaßen: wir haben nun ein Zeichen vom Magnetband MB3, das nächste Zeichen ist ebenfalls vom Magnetband 3, das nächste wieder, usw. Sie sehen, hier haben wir nur Zeichen von einer E/A-Einheit, dies ist der sog. Burst-Mode ("Explosions-Verfahren"). Nun verstehen Sie, warum der Standardkanal immer der erste Kanal ist bei einem System, denn ich kann sowohl langsame Geräte in einem System damit betreiben als auch schnelle Geräte. Allerdings ist der Nachteil, daß, wenn ich mit einer schnellen E/A-Einheit arbeite,

während dieser Zeit nichts anderes mehr durch den Kanal hindurchgehen kann. D.h., eine schnelle Einheit verwendet die gesamte Kapazität eines Standardkanals. Dies ist ein Nachteil, denn während z.B. eine Datenübertragung stattfindet, könnte es sein, daß der Operator über die Konsole benachrichtigt werden muß. Die Konsole hängt aber ebenfalls an dem Standardkanal. Wenn die Kapazität des einzigen Standardkanals dann aufgebraucht ist, können keine Nachrichten mehr gesendet werden. Deshalb werden auch Schnellkanäle benötigt. Schnellkanäle sind solche Kanäle, die ähnlich den Standardkanälen aufgebaut sind, nur daß sie einen einzigen Unterkanal besitzen. Mehr wird auch nicht nötig, denn die Idee des Schnellkanals ist, jüngerade ausschließlich, für schnelle E/A-Einheiten verwendet zu werden.

9.3 Aufbau der Kanäle

9.3.1 Allgemeiner Aufbau

Benötigt werden Registerpuffer zur Datenaufnahme und Datenabgabe, weiterhin ein Leitwerk, um die Steuerung der E/A-Vorgänge vorzunehmen, und eine Anpaßelektronik zur Auswahl des betreffenden E/A-Geräts, mit dem gearbeitet werden soll. Dies heißt nicht, daß nur ausschließlich ein Gerät angeschlossen werden kann, sondern es kann an einen Schnellkanal von z.B. 10 Magnetbändern eines angeschlossen werden; allerdings, wenn eines arbeitet, kann kein anderes arbeiten. Die Geräte sind also auswechselbar, das ist also nicht so, daß man für ein schnelles Gerät ausschließlich einen Kanal benötigt. Der Aufbau ist einfach, siehe Abb. 75, es gibt 2 Sammelschienen, eine für die Information, eine für die Adressen und weiterhin die Anpaßelektronik, die jeweils Ausgänge zum Informationsbus hat und einen Eingang von der Adreßschiene, weiterhin das Leit-

werk, das das Herz des Kanals ist. Das Leitwerk hat Ein- und Ausgänge zu der Informationsschiene sowie einen Ausgang zur Adreßschiene. Das Befehlswerk hat einen Eingang vom Leitwerk her und hat einen Ausgang zum Unterbrecherwerk, außerdem das Leitwerk ebenfalls zum Unterbrecherwerk. Ebenso existiert ein Ausgang vom Leitwerk zur peripheren Einheit. Schließlich haben wir die Pufferspeicher und die Register. Die Register und die Pufferspeicher haben jeweils einen Ausgang zu der Sammelschiene für Ein- und Ausgabe. Es gibt noch einen Ausgang zum Leitwerk, der anzeigt, wenn der Puffer voll ist, dann wird der Inhalt des Puffers an die Informationsschiene abgegeben. Vom Puffer gibt es den Ausgang von und zur peripheren Einheit.

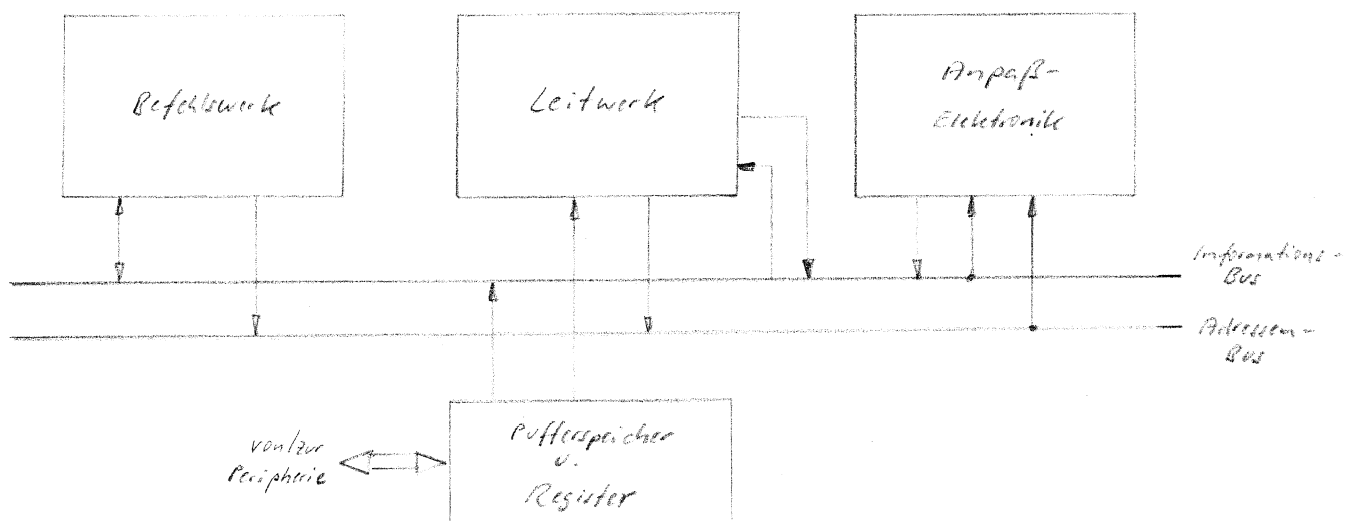


Abb. 75: Genereller Aufbau des Kanalwerks

9.3.2 Funktioneller Aufbau

In Abb. 76 sehen wir nun den funktionellen Aufbau eines Kanals. Wir haben die Kanalkontrolleinheit, und weiter ein Gerätereister. Das Gerätereister enthält die Adresse des Geräts mit dem ich gerade arbeiten möchte. Das Unterbefehladressregister speichert den Befehl, d.h., die Adresse des Unterbefehls, mit dem gerade gearbeitet wird. Der Befehl steht im Speicher. Das Unterbefehlsregister nimmt den Unterbefehl auf. Weiter gibt es ein Zwischenspeicherregister und darunter ein Sammelregister mit dem Kommutator. Unter der Kontrolleinheit ist noch ein Kanalstatuswortregister eingezeichnet.

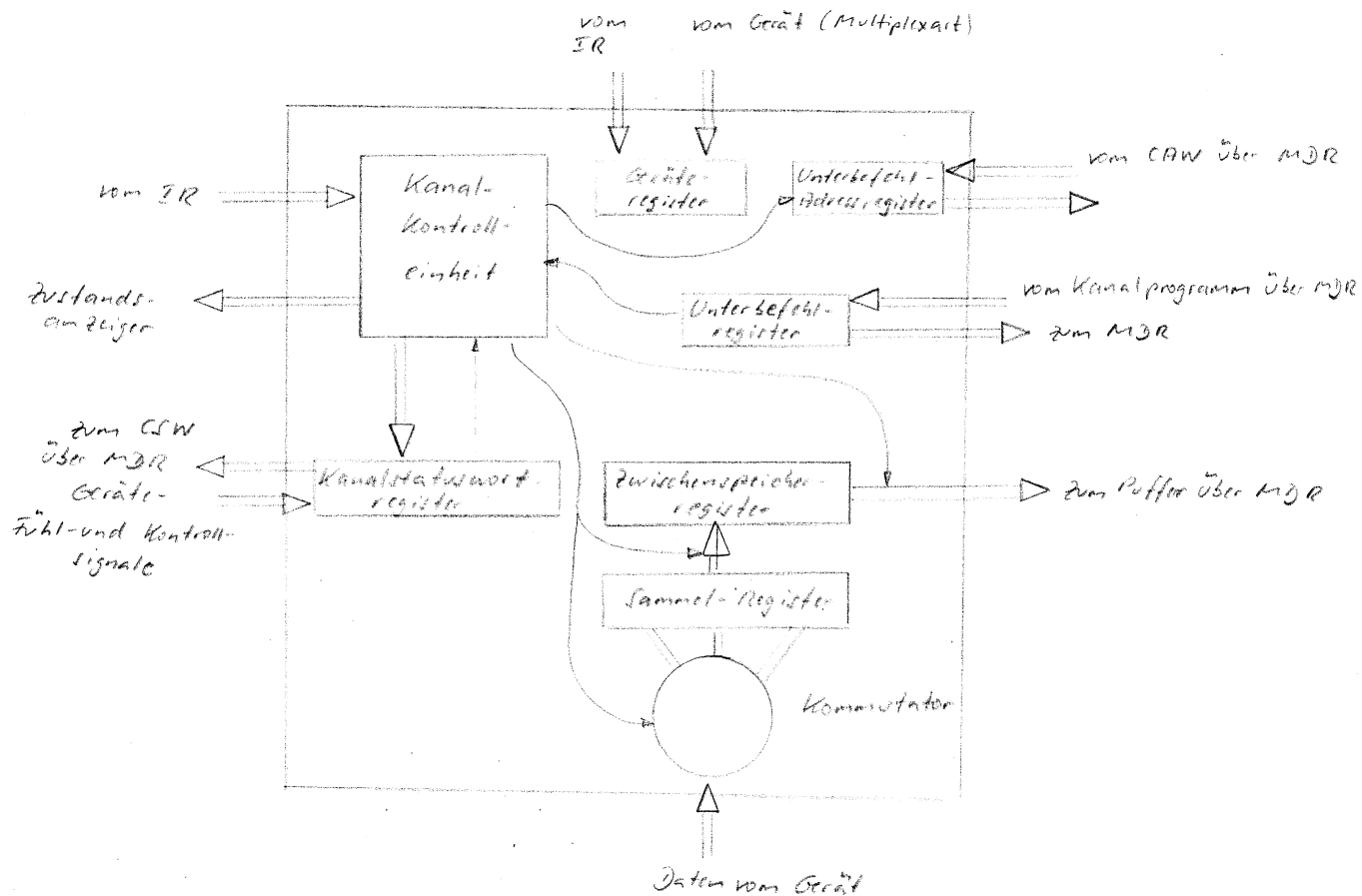


Abb. 76: Funktioneller Aufbau des Kanalwerks

Wir wollen nun die Funktionen untersuchen, die hier stattfinden. Beginnen wir mit dem Gerätere-gister. Wir haben hier Eingänge von Instruktionsregister. Vom Instruktionsregister bringen wir nun die Adresse des Geräts in das Gerätere-gister mit dem gearbeitet werden soll. Weiter haben wir noch vom Gerät her Eingänge, falls wir im Multiplexverfahren arbeiten. Im Multiplexverfahren kommen die einzelnen Teile vermischt herein (Doppelpfeile symbolisieren Datenübertragung und einfache Pfeile eine Kontrollübertragung). Hier haben wir nun eine Datenübertragung und zwar haben wir hier die Unterbefehle, z.B. die Kanaladressenwörter bei der IBM 360 über das Speicherdatenregister MDR. Dies ist das Register, wo erst mal alles hereingeht, was aus dem Speicher herauskommt. Aus dem Speicher kommt nun die Unterbefehlsadresse heraus über dieses Kanaladressenwort. Dann haben wir den Ausgang zum Speicheradressenregister. Das Unterbefehlsregister übernimmt den Unterbefehl vom Kanalprogramm über das Speicherdatenregister und gibt die Unterbefehle, falls nötig an das Speicherdatenregister. Vom Zwischenspeicherregister haben wir nur einen Ausgang, und zwar geht es zum Pufferspeicher im Speicher über das Speicherdatenregister. Das sind die Daten, die über das Speicherregister hereinkommen. In das Speicherdatenregister kommen die Daten vom Sammelregister. Das Sammelregister empfängt die Daten vom Kommutator. Die Daten werden zeichenweise übertragen, und der Kommutator schaltet um ein Zeichen weiter, sobald ein Zeichen ein- oder ausgegeben ist. Betrachten wir die Ein- und Ausgabe der Daten vom Gerät. Das Kanalstatuswortregister geht hinaus zum Kanalstatuswort über das Speicherdatenregister. Im Speicher haben wir eine Anzeige, über den Status des Kanals, dieser wird nun über das Speicherdatenregister hergeholt. Diese Information

stammt von den Geräten und Kontrollgeräten. z.B., wenn ein Gerät stehen bleibt. Weiter haben wir eine Übertragung von der Kontrolleinheit zum Instruktionsregister. Im Instruktionsregister wird mitgeteilt, welche Operation ausgeführt werden soll. Die Kontrollsignale gehen von der Kontrolleinheit aus in das Unterbefehlsadressregister. Vom Unterbefehlsadressregister kommen Signale in die Kontrolleinheit. Die Kontrollsignale gehen wieder aus. Von der Kontrolleinheit wird die Datenübertragung gesteuert, und der Kommutator zeigt an, daß ein Aggregat bereit liegt, das übertragen werden kann.

9.4 Arbeitsweise

9.4.1 E/A-Befehle

Betrachten wir dazu Abb. 77.

Wir haben hier den Speicher, den Zentralrechner und die Kanäle 1 bis n. An den Kanal 1 sind einige E/A-Geräte angeschlossen, im Beispiel hier 3, d.h., dieser Kanal 1 muß ein Multiplex-Kanal sein, an den Kanal n schließen wir ein Magnetband an, d.h., dieser Kanal n ist ein Schnellkanal.

Wir haben nun in dem Speicher ein Programm, das den Zentralrechner mit Instruktionen beauftragt, außerdem sind noch Daten vorhanden, die ebenfalls im Zentralrechner in die Ausführung dieses Programms eingehen. Ein Programm wird hier befehlsweise ausgeführt mit den dazugehörigen Daten. Der Kanal hat hier im Speicher einen Pufferbereich, d.h., hier haben wir Datenübertragung zwischen Kanal 1 dieses Pufferbereichs und dann zwischen den Ein- und Ausgabegeräten ebenfalls. So hat jeder Kanal einen gewissen Pufferbereich, min-

destens einen Pufferbereich.

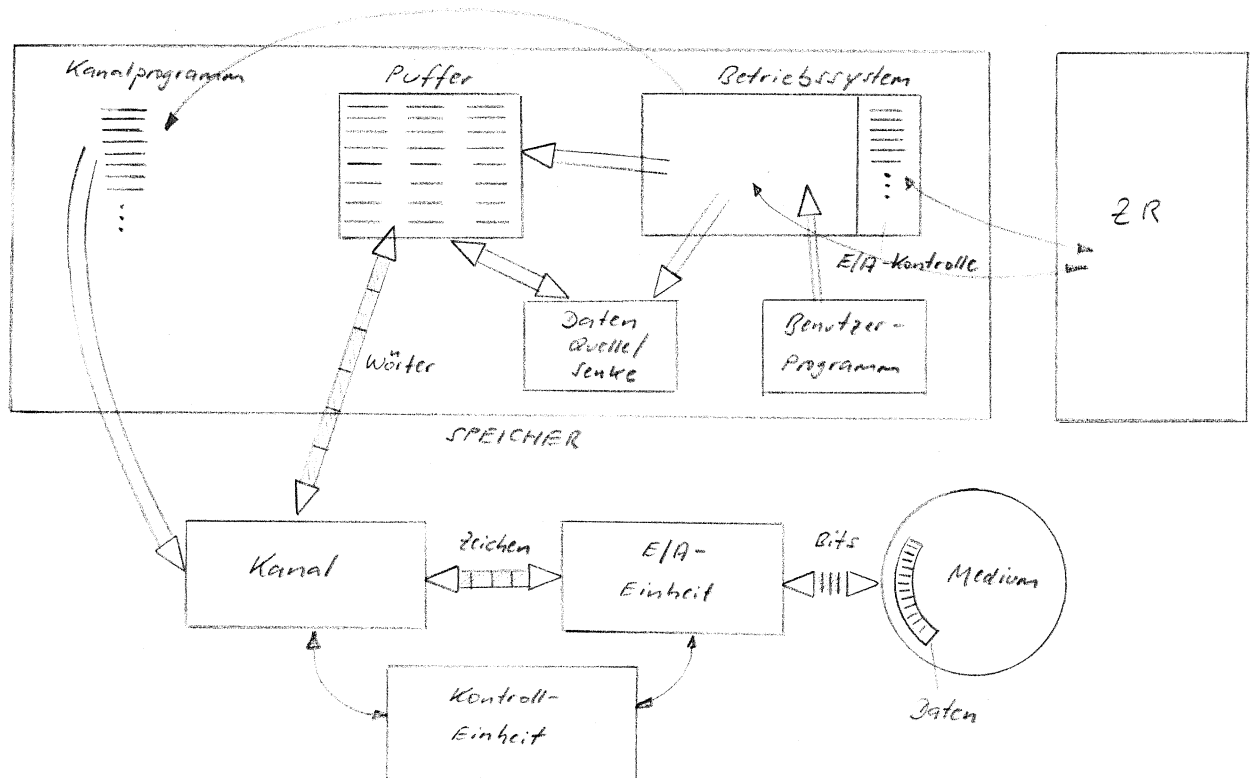


Abb. 77: Zusammenwirken zwischen Kanal und Zentralrechner

Die Kanäle und der Zentralrechner können relativ unabhängig arbeiten, mit Ausnahme des momentanen Speicherzugriffs. Sobald kein Speicherzugriff erforderlich ist, kann der Zentralrechner für sich arbeiten, ebenso der Kanal. Falls nur einer dieser aktiven Komponenten den Speicher braucht, entsteht ebenfalls noch kein Konflikt. Sie sehen, es gibt sehr große Bereiche, wo keinerlei Schwierigkeiten entstehen. Nur in dem Augenblick, wo sowohl der Kanal als auch Zentralrechner Zugriff zum Speicher verlangen, tritt ein Konfliktfall auf, den wir bereits anfangs in der Einleitung erwähnt haben. Natürlich muß irgendwie eine Zuordnung, eine hierarchische Zuordnung zwischen Kanälen und Zentralrechner bestehen und zwar ist es so, daß die Kanäle durch den Zentralrechner kontrolliert werden. Der Zentralrechner hat also Kontrolle über die

Kanäle, d.h., die Indizierung der Kanäle. Die Beaufsichtigung und die Auflösung von Konflikten bei den Kanälen fällt dem Zentralrechner als Aufgabe anheim. Für diese Funktionen hat der Zentralrechner spezielle Befehle, d.h., im Instruktionsrepertoire eines Rechners unterscheiden wir zwei Arten von E/A-Instruktionen. Dies sind die Instruktionen, die vom Zentralrechner ausgeführt werden und dann die E/A-Befehle oder auch Kanalbefehle genannt; diese werden von den Kanälen ausgeführt. In anderen Worten, die E/A-Instruktionen, die der Zentralrechner ausführt, dienen zur Kontrolle der E/A-Vorgänge, die wir eben erwähnt hatten. Die Kanalbefehle dienen den Kanälen zum Bearbeiten der E/A-Funktionen. Diesen entsprechen z.B. die CCW's ("Channel Control Words") bei IBM 360/370 Maschinenreihen. Diese Kanalbefehle werden meistens zusammengefaßt zu Kanalprogrammen. Wir haben jetzt eine Schwierigkeit. Wie arbeiten diese verschiedenen Arten von Befehlen zusammen, d.h., welche Reihenfolge müssen wir bei der Ausführung dieser Befehle beachten. Wir sprechen nun über das Zusammenwirken dieser Befehle. Dazu betrachten wir Abb. 77: Wir haben hier wieder den Speicher, ebenso einen Zentralrechner, einen Kanal und eine Kontrolleinheit für die E/A-Einheit, und dann haben wir noch das Medium mit den Daten. Dies sind die Komponenten dieses Bildes. Nun zu den Einzelheiten: Im Speicher haben wir das Kanalprogramm, d.h., die Instruktionen, die der Kanal ausführt. Weiter haben wir wieder wie vorher Puffer und dann ebenfalls im Speicher das Betriebssystem, sowie der Bestimmungs- oder Quellort der Daten. Wir wollen also Daten auf dem Medium in den Bestimmungsort übertragen, oder umgekehrt, Daten vom Quellort in den Bestimmungsort auf dem Medium übertragen. Das ist der Sinn dieser Datenübertragung. Nun, wie geht diese Datenübertragung vor sich? Wir haben noch im Betriebssystem einen Anteil, der die E/A-Kontrolle bewirkt. In der Ausführung des Benutzerprogramms verlangt nun der Benutzer, eine E/A-Operation durch das Betriebssystem auszuführen. Das Betriebssystem wiederum arbeitet

natürlich auch mit dem Zentralrechner zusammen. Wird eine E/A-Operation verlangt, hält dies der Zentralrechner fest und übergibt die Kontrolle dann dem Betriebssystem. Das Betriebssystem bestimmt nun anhand der Befehle, die es erhält, den Bestimmungs- oder Quellort der Daten und auch den Puffer, der für die Datenübertragung benötigt wird. Dann nimmt das Betriebssystem Zugriff zu dem Kanalprogramm und führt das Kanalprogramm aus. Das Kanalprogramm referiert in dem Kanal und der Kanal hat nun Zugriff zu der Kontrolleinheit, die Kontrolleinheit zu der E/A-Einheit und die E/A-Einheit zu dem Medium. Das sind die Verbindungen die zuerst hergestellt werden müssen. Dann erfolgt die Datenübertragung. Die Datenübertragung, z. B. Eingabe, erfolgt nun vom Medium. Vom Medium kommen Bits heraus in die E/A-Einheit. Diese Bits werden in der Kontrolleinheit zu Zeichen zusammengefaßt, die dann in den Kanal eingehen, und vom Kanal werden diese Zeichen wieder zusammengefaßt und als Wörter in den Puffer übertragen. Bei der Ausgabe erfolgt dasselbe in der umgekehrten Richtung. Es werden die Wörter vom Puffer in den Kanal übertragen und zur E/A-Einheit gegeben und als Bits auf das Medium übertragen. Der Zweck des Puffers ist, daß hier die Daten vom Kanal an das Betriebssystem übergeben werden oder umgekehrt. Das ist eine Art Schnittstelle zwischen Betriebssystem und Kanal und zwischen der Software und der Hardware.

Nun, die erste Frage ist, wie werden diese E/A-Probleme kontrolliert? Sie werden kontrolliert durch die E/A-Instruktionen, die bereits erwähnt worden sind (Befehle der IBM 360/370). Die erste Instruktion, die wir beachten müssen, ist "start I/O", SIO (Code), dann haben wir "Test I/O" (TIO), dann "Test channel" (TCH) und dann "Halt I/O" (HIO). Dies sind die E/A-Instruktionen. Das sind alle E/A-Instruktionen, die bei der

360/370 vorkommen. Diese vier dienen dazu, die E/A der Datenübergänge zu kontrollieren.

Das Format dieser Instruktionen ist in Abb. 78 dargestellt: 9 c ist der Operationscode in hexadezimaler Notierung, dann folgen 00, dies ist jeweils ein Byte von 8 Bits, dann haben wir die Basisadresse und die Versetzung zur Basis. Dies zusammen ist die Adresse für eines dieser Befehle.

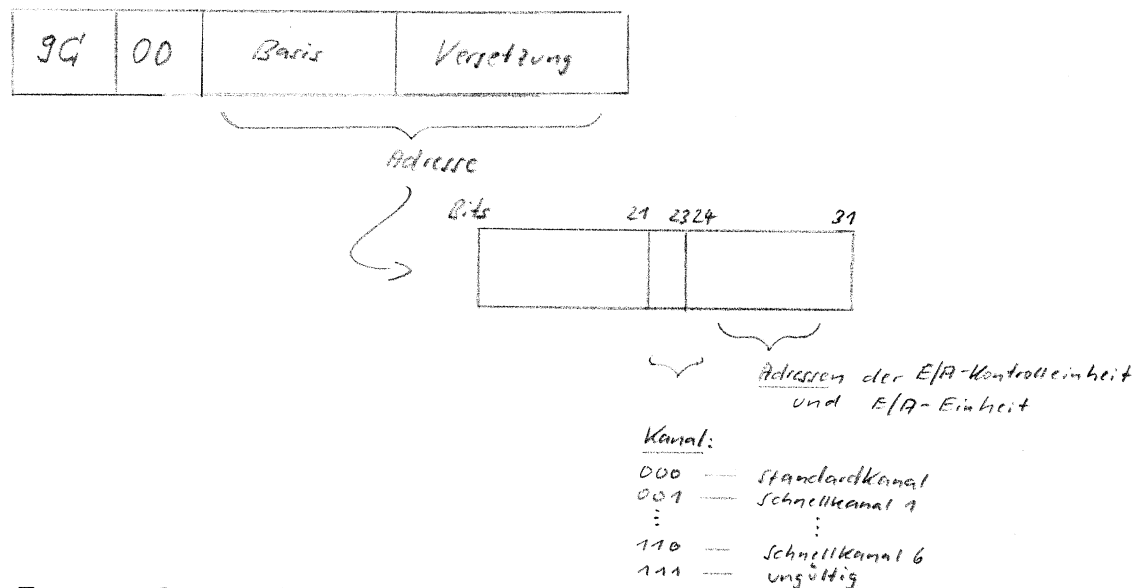


Abb. 78: Format des Befehls SIO

Nun, wie Sie sehen, reichen diese Befehle lange nicht aus, um all das auszusagen, was ausgesagt werden muß, z.B. haben wir nun wohl das Adressfeld, in dem wir z.B. die Adresse eines E/A-Gerätes angeben können. Doch reicht das keineswegs aus, wir müssen angeben, welche Art Operation wir verlangen, welche Art Bewegungsrichtung verlangt wird, usw. Sie sehen, diese Information ist nicht genug, deshalb zeigt diese Adresse nicht auf das E/A-Gerät, sondern auf einen anderen Speicherplatz.

Es ist also ein Zeiger, der auf einen Speicherplatz verweist und dieser Speicherplatz enthält in den Bits 21 bis 23 die Kanalidentifizierung und dann die Adresse der Kontrolleinheit und des E/A-Geräts, die einfach zusammengesetzt werden. Der linke Teil dient für die E/A-Kontrolleinheit, der rechte Teil für das E/A-Gerät. Wir haben genügend Platz in den Bits 24 bis 31, für den Kanal und für die Kontrolleinheit, d.h., wir haben einmal die Codierung 000, dies ist der Standardkanal, 001 ist der Schnellkanal 1 usw., bis zu 110, das ist der Schnellkanal 6 und 111 ist dann die ungültige Kombination zur Kontrollierung dieser Adressen. Das reicht uns jedoch immer noch nicht aus, denn wir haben noch keine Aussage über die Operation und Bewegungsrichtung, d.h., wir müssen weiterhin Information bekommen können, allerdings nicht mehr über eine andere Stufe der indirekten Adressierung, sondern über einen festen Speicherplatz im Hauptspeicher, und zwar wird Speicherplatz 72₁₀ in dezimaler Zählung verwendet für weitere Angaben. Dieser Speicherplatz 72 enthält das Kanaladressenwort.

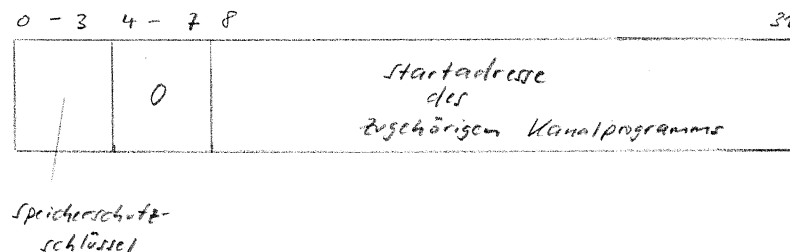


Abb. 79: Kanaladressenwort

Dieses Kanaladressenwort sieht folgendermaßen aus (siehe Abb. 79):

Von Bit 8 bis 31 ist eine Adresse enthalten, und zwar ist hier die Startadresse des zugehörigen Kanalprogramms angegeben. Dazu kommen noch die Bits 4 bis 7 mit der Besetzung 0 und die Bits 0 bis 3 als Speicherschutzschlüssel. Nun, Sie sehen, diese E/A-Instruktionen haben eine etwas seltsame Funktion und zwar gehen sie erstmal in einer indirekten Stufe durch, und dann greifen sie noch auf einen festen Speicherplatz als Adressenplatz zu. Was ist nun der Vorteil dieser Anordnung? Der Vorteil dieser Anordnung ist der, daß bei Zugang zu einem anderen Kanalprogramm nur dieser Speicherplatz 72 geändert werden muß und ich kann automatisch dann wieder dieselbe Befehlsfolge hier verwenden. Dies ist der Sinn der Sache und zwar wurde die Maschine IBM 360 von vorneherein so entworfen, daß die E/A-Vorgänge mit geringstem Aufwand möglich sein sollen. Deshalb nun diese Anordnung.

Wie ist nun die Befehlsfolge in einem E/A-Programm, d.h., wie werden nun diese E/A-Instruktionen mit den Kanalbefehlen, die im Kanalprogramm sind, verwendet?

In Abb. 80 haben wir wieder den Speicher. In diesem Speicher haben wir ein Benutzerprogramm, weiter ein Betriebssystem, und dann haben wir noch außerhalb den Zentralrechner, einen Puffer, und den Kanal und Zugang zu dem Kanal von mehreren Kontrolleinheiten. Über die Kontrolleinheit könnten ebenfalls mehrere E/A-Geräte angeschlossen sein, davon ist hier nur eines wiedergegeben.

Nun wir haben bereits gesehen, daß wir den Speicherplatz 72 benützen als das Kanaladressenwort.

Gehen wir nun die Reihenfolge der Ausführungen eines E/A-Gerätes durch:

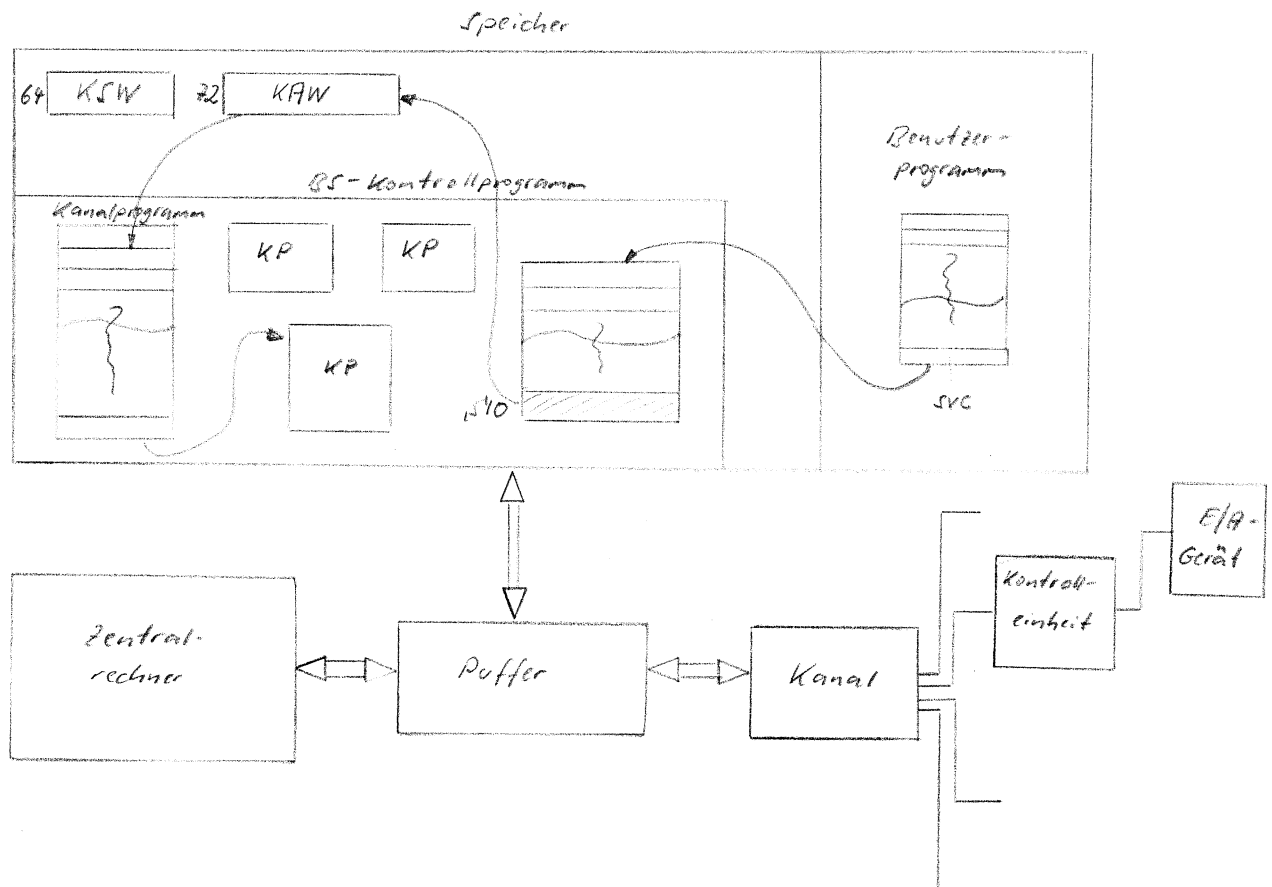


Abb. 80: Befehlsfolge einer E/A-Operation

Ein Benutzerprogramm muß, wenn es E/A verlangt, einen Dienstbefehl SVC "Service Call" an das Betriebssystem abgeben. Dieser "SVC" deutet nun in das Betriebssystem. Im Betriebssystem existiert ein Kontrollprogramm. Dieses Kontrollprogramm wird durchlaufen, und in dem Kontrollprogramm existiert nun ein SIO-Befehl. Wie wir bereits wissen, verwendet diese SIO-Instruktion eine Adresse, und ebenso das Kanaladressenwort. Der Platz, an den das SIO direkt zeigt, gibt die Adresse des Kanals und der Kontroll- und E/A-Einheit an. Das sind also die Adressen des E/A-Geräts, und das Kanaladressenwort zeigt in das Kanalprogramm. Wir haben das Kanalprogramm, bestehend aus Kanalbefehlen, zusammen mit mehreren anderen Kanalprogrammen. Die Ausführung des Befehls erfolgt nun dadurch, daß das Kanalprogramm durchlaufen wird. Dieses Kanalprogramm verlangt nun die Arbeitsweise dieses Ka-

nals, d.h., bestimmt die Arbeitsweise und die Charakteristiken des Kanals und der Datenübertragung von dem E/A-Gerät zum Kanal und zum Puffer und dann vom Puffer in den Speicher. Hier haben wir wiederum die Pufferbereiche, wo die Daten gespeichert werden. Außerdem haben wir einen reinen Hardware-Puffer zwischen Kanal und Zentralrechner, um zu bestimmen, wann der Rechner Zugriff hat und wann der Kanal Zugriff zum Speicher haben soll. Sie haben miteinander nichts gemein, mit der Ausnahme, daß die Bezeichnung gleich ist. Der Speicherplatz 64 enthält das Kanalstatuswort KSW, auf das wir noch zu sprechen kommen.

Betrachten wir die Kanalbefehle etwas näher. Das Kanalprogramm besteht aus Kanalbefehlen, und die Kanalbefehle sind die sogenannten Kanalprogramm Wörter, die CCW's (Channel Command Words). Dieses Format müssen wir etwas näher in den Einzelheiten ansehen und zwar deshalb, weil hier nun die Variationen ins Spiel kommen (siehe Abb. 81).

Wir haben hier ein Doppelwort, d.h., die Bitzählung geht von 0 bis 63. Wir haben von 0 bis 7 den Befehl, d.h., hier wird angegeben, welche Operation ausgeführt werden soll. Dann haben wir von 8 bis 31 die Datenadresse, von 32 bis 36 Flags, von 37 bis 39 Null, und dann von 40 bis 47 haben wir eine Identifizierung, die normalerweise immer in der Ausführung dieser Befehle ignoriert wird. Dann haben wir zuletzt von 48 bis 63 die Zeichenzahl, d.h., die Zahl der Zeichen, die ein- oder ausgelesen werden sollen. Diese Befehle sind einer aus der Gruppe "Lesen", "Schreiben" oder "Fühlen".

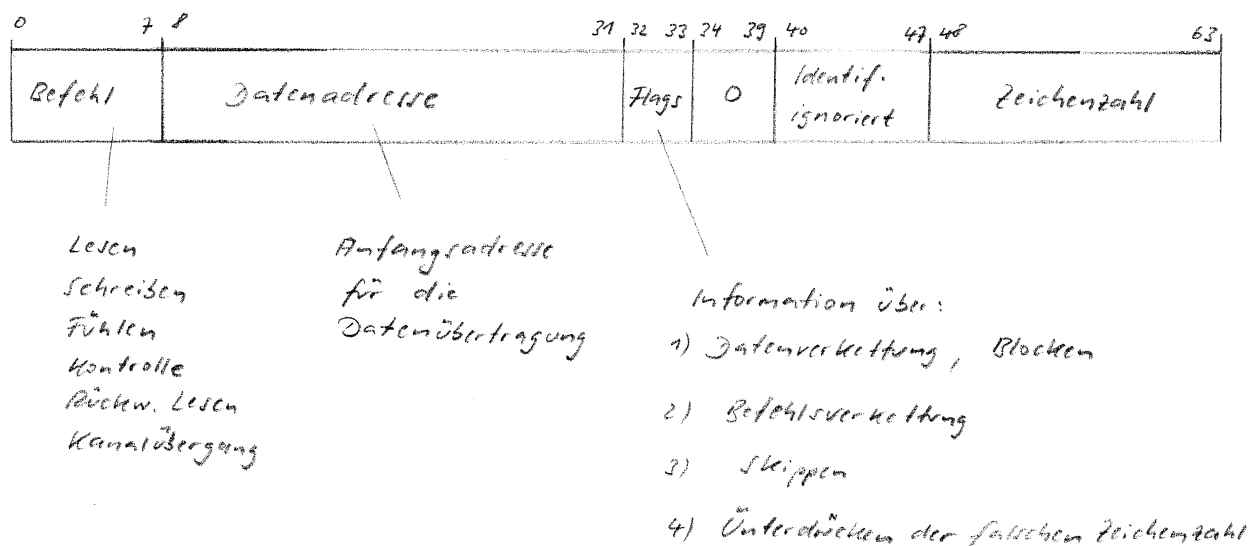


Abb. 81: Aufbau der Kanalprogrammworter

Nun, "Lesen" und "Schreiben" sind selbstverständlich. Weiter haben wir "Fühlen", d.h., hier wird das E/A-Gerät abgefragt, um zu sehen, wie dessen Zustand ist. Die Kontrolle ändert den Zustand der Geräte, z.B. "Rückwärts lesen", dies ist möglich bei Bandgeräten. Der Kanalübergang ist nichts anderes als ein unbedingter Sprungbefehl, d.h., wenn wir ein Kanalprogramm durchlaufen haben, können wir nun weiterspringen in ein anderes Kanalprogramm. Dies ist der Sinn dieses Kanalübergangs, d.h., wir können also Kanalprogramme damit zusammensetzen. Während dieses Kanalübergangs findet an dem Gerät, das gerade arbeitet, keinerlei Änderung des Zustands oder der Arbeitsweise statt. Das nächste ist die Datenadresse. Hier haben wir die Anfangsadresse für die Datenübertragung, d.h., das ist die Speicheradresse, der Speicherplatz des Worts, das

übertragen werden soll. Die nächste Gruppe sind die Flags. Diese Flags sind besonders interessant und zwar aus folgendem Grund: Sie geben uns Information über die Arbeitsweise, z.B., wenn weitere Kanalbefehlswörter für denselben E/A-Block folgen, wie Starten oder Lesen von der Trommel oder von der Platte. Dann haben wir einen Kanalbefehl für Starten, und einen für z.B. Weiterlesen.

Wir haben hier eine sogenannte Datenverkettung. Es ist nichts anderes, als was wir früher mit Blocken bezeichnet haben. Zum Zweiten können wir weitere Kanalbefehle folgen lassen für eine neue Operation am selben Gerät. Dies wird als Befehlsverkettung bezeichnet. Der Unterschied ist einfach der, daß in dem ersten Fall die Daten zusammenhängend bearbeitet werden, und im zweiten Fall wir keine zusammenhängende Bearbeitung der Daten mehr haben, sondern eine Verkettung der Kanalbefehle. Zum dritten haben wir eine Angabe für das Überspringen eines Teils des Datenblocks, und viertens eine Unterdrückung der unrichtigen Zeichenzahl. Falls die angegebene Zahl der Zeichen in dem Kanalbefehl von der wirklichen abweicht, dann muß hier durch ein flag angegeben werden, daß dies unterdrückt werden soll, andernfalls wird ein Fehlerzustand erfolgen. Die Identifizierung dient zur Beschreibung des einzelnen Befehls. Die Zeichenzahl gibt an, wieviel Zeichen bearbeitet werden sollen.

Das nächste ist nun, wie stellen wir fest, was am Gerät oder im System vorgeht. Dies ist vor allem wichtig, falls ein Fehlerzustand eintritt, z.B. die falsche Zeichenzahl, oder wenn ein Teil des Datenblocks übersprungen werden soll, oder andere Fehlerzustände, die eventuell bei diesem komplizierten Zusammenwirken der Komponenten auftreten können. Dazu verwenden wir für das Feststellen der wirklichen Vorgänge den bereits an-

gegebenen Speicherplatz 64 mit dem Kanalstatuswort. Dieses Kanalstatuswort ist nun dieser Platz, wo wir sehen können, was los ist. Dieses Kanalstatuswort wird vom Zentralrechner nach jeder Unterbrechung, sobald eine Änderung des Arbeitszustandes im System mit E/A-Vorgängen stattfindet, durch den Zentralrechner verändert, bzw. auf den neuen Weg gesetzt. Erstens einmal nach Beendigung von normalen E/A-Vorgängen. Beispielsweise wollen wir eine Karte lesen und wir haben diese Karte gelesen und haben die Daten im Puffer stehen. Irgendwo muß dann angezeigt werden, daß dies nun der Fall ist. Dies wird mit Hilfe dieses Kanalstatuswortes getan, und zwar setzt der Zentralrechner dort einen bestimmten Wert ein. Zum zweiten nach abnormaler Beendigung eines E/A-Vorganges, d.h., es treten Fehler auf, z.B. sind wir am Ende eines Magnetbandes angelangt, und wir wollen noch mehr Daten ablesen. Dies ist ein abnormaler Vorgang, der nicht möglich ist und angezeigt werden muß. Drittens nach einem Befehl TIO("Test Input/Output"), dies ist eine E/A-Instruktion, d.h., in diesem Moment möchte z.B. das Betriebssystem feststellen, welcher Status der E/A-Geräte zur Zeit vorliegt. Der Sinn dieser TIO-Instruktion ist, daß der Zentralrechner dieses Kanalstatuswort setzt und es nachher abgefragt werden kann. Schließlich nach einem zurückgewiesenen SIO. Falls das Betriebssystem eine "Starte I/O"-Instruktion gibt, die nicht ausgeführt werden kann, wird diese zurückgewiesen und natürlich muß irgendwo festgehalten werden, weshalb dies der Fall ist. Diese Gründe werden hier auch ebenfalls im Kanalstatuswort registriert. Der Zweck hier ist also, Einzelheiten über die Ursache einer Unterbrechung zu geben und ebenfalls auch über den Zustand (den Status) des Kanals und der Geräte. Wir wollen wissen, welchen Grund eine Unterbrechung hat, die erfolgt ist, und möchten auch wissen, welcher Zustand an einem augenblicklich in Betrieb befindlichen Kanal und den Geräten vorliegen.

Das Kanalstatuswort hat nun folgenden Aufbau (siehe Abb. 82):

Von 0 bis 3 haben wir einen Schlüssel, dies ist ein Schlüssel wie bei dem Kanalstatuswort zum Speicherschutz. Von 4 bis 7 Null, dann von 8 bis 31 haben wir die Adresse des Befehls, von 32 bis 37 und von 40 bis 47 haben wir den Status, und dann von 48 bis 63 die Zeichenzahl.

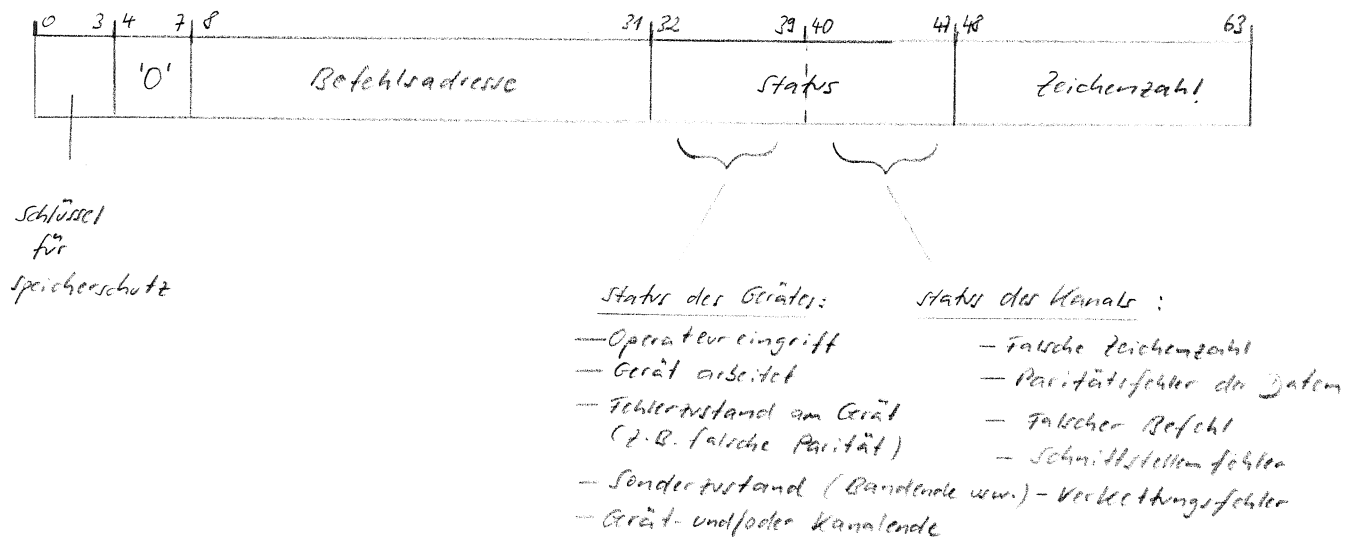


Abb. 82: Aufbau des Kanalstatuswortes

Was ist die Bedeutung dieser Bitfelder? Die Befehlsadresse ist die Adresse des zuletzt ausgeführten Kanalkontrollwortes CCW plus 8. Diese Befehlsadressen werden inkrementiert, wie normalerweise auch die Adressen im Benutzerprogramm. Im Benutzerprogramm haben wir Adressen, und sobald ein Befehl ausgeführt wird, wird sofort noch während der Ausführung des Befehls inkrementiert, genauso geht es bei der Ausführung der Kanalbefehle. Wenn ein Kanalbefehl ausgeführt wird, wird dieser Adressenzähler sofort inkrementiert. Hier haben wir nichts anderes als den Adressenzähler des Kanalprogramms. Nur, wie gesagt, hat er keinen fe-

sten Platz, sondern der Adressenzähler wird irgendwo im Kanal der physische Platz eines Speichers ein. Nur in dem Moment, wo ein Sonderzustand eintritt, wird dieser übertragen in das Kanalstatuswort und zwar von 8 bis 31. Die Statusanzeige besteht aus zwei Feldern, zuerst von 32 bis 37, hier wird der Status des Geräts, z.B. "Eingriff eines Operateurs", angegeben: Wir haben eine Datenübertragung über Magnetband und plötzlich drückt der Operateur den Knopf irgendwo an einem System an dem Laufwerk, es erfolgt ein Eingriff und es kann nicht mehr weitergelesen werden. Diese Störung muß festgestellt werden. Wir haben einen Status, "Gerät arbeitet normal", wie kann dieser Status festgestellt werden? Dies kann mit einem "Test Input/Output" festgestellt werden, dann erwarten wir die Antwort, daß das Gerät entweder normal arbeitet oder ein Sonderfall auftritt. Deshalb kommt dieser Status "Gerät arbeitet normal" zustande. Dann haben wir den "Fehlerzustand am Gerät", z.B. falsche Parität usw. Zur weiteren Analyse werden wir hier einen Fühlbefehl einfügen. Der Sonderzustand ist kein eigentlicher Fehlerzustand, sondern kann nur das Gerät irgendwie nicht arbeiten, wie es soll, z.B. durch Bandende, oder Dateienende oder Gerät- oder Kanalende, d.h., beide, der Kanal und das Gerät arbeiten nicht. Es kann auch vorkommen, daß der Operateur vergessen hat, ein Gerät auszuschalten, dann kommt Geräte-Kanalende. Im zweiten Feld der Statusanzeige wird der Kanalzustand eingetragen, z.B. "falsche Zeichenzahl". Es wurde die falsche Zeichenzahl in dem Kanalbefehl angegeben, oder wir haben Paritätsfehler der Daten, oder falsches Kontrollwort, Kanalkontrollwort, Kanaladressenwort, d.h., Fehler in den Kanalbefehlen. Weiter gibt es noch Schnittstellenfehler, d.h., das Gerät arbeitet, jedoch mit falscher Schnittstelle zwischen Kanal und Kontrolleinheit, "Gerät arbeitet falsch", und schließlich noch einen Verkettungsfehler, z.B. das Gerät arbeitet zu

schnell. D.h., die Verkettung der Daten kann nicht vonstatten gehen wie es sein soll.

Die Zeichenzahl im Kanalstatuswort gibt nun die wirkliche Zahl der Zeichen, die übertragen worden sind, an. D.h., in dem Kanalbefehl haben wir angegeben, wieviele Zeichen übertragen werden sollen. Hier sind nun angegeben, wieviele Zeichen noch verbleiben, also Zeichen, die noch nicht verarbeitet worden sind. Wenn alles glatt verläuft, muß diese Zahl Null sein.

Dies waren nun die Fehlerzustandsanzeigen. Als nächstes betrachten wir noch die Reihenfolge der E/A-Programmierung. Zuerst haben wir das Kanaladressenwort auf den neuen Wert zu setzen, d.h., die Adresse des Kanalprogramms, das wir ausführen möchten, das ist hier natürlich ein E/A-Vorgang. Dann geben wir das SIO an den Zentralrechner und dies verwendet das CAW für die Ausführung der Kanalkontrollwörter im Kanalprogramm, und dies setzt auch das Kontrollstatuswort. Nun, auf der IBM 360/370 haben wir noch folgendes. Dies sind nicht alle Anzeiger, die uns zur Verfügung stehen, wir haben noch einen Zustandsanzeiger, in dem sog. "CC", mit folgenden Bedeutungen: Null bedeutet "Operation hat begonnen", 1 heißt "Operation hat nicht begonnen", und das Kanalstatuswort ist gesetzt worden. Das Kanalstatuswort kann abgefragt werden. 2 heißt, der Kanal arbeitet. 3 heißt, das Gerät ist nicht arbeitsfähig. Sie sehen, wir haben hier eine gewisse Duplizität der Information für "Gerät ist abgeschaltet" und zwar rührt es daher, daß diese Zustandsanzeige sehr sehr einfach im Programm abgefragt werden kann. Wohingegen die Abfrage des Kanalstatusworts sehr kompliziert ist. Der Zustand, daß das Gerät nicht arbeitsfähig ist, ist ein sehr einschneidender Zustand, der so früh wie möglich festgestellt werden sollte. Wir haben weiterhin Tests zum Status der E/A und der Kanäle. Die E/A wird z.B. durch TIO getestet und diese TIO setzt nun die

Zustandsanzeige ebenfalls, mit folgender Bedeutung: Null bedeutet "Gerät verfügbar", 1 entspricht "Gerät arbeitet", und CSW ist gespeichert worden, d.h., es kann abgefragt werden; 2 bedeutet, daß das Gerät nicht arbeitet, das Kontrollstatuswort kann abgefragt werden, und 3 bedeutet, daß das Gerät nicht arbeitsfähig ist. Sie sehen, es deckt sich im wesentlichen mit der Zustandsanzeige des SIO. Und zwar deshalb, weil das TIO eine sehr verwandte Instruktion zu dem SIO ist, es hat dasselbe Format. Das SIO benützt allerdings natürlich nicht das Kanaladressenwort.

Der Status der Kanäle wird durch die TCH-Instruktion geprüft. Dabei wird dasselbe Format verwendet, jedoch wird nur der Kanaladressenteil benutzt. Die TCH-Instruktion setzt ebenfalls wieder die Zustandsanzeige mit der Bedeutung, daß Null "verfügbar" bedeutet, 1 heißt, eine "Unterbrechung wartet", an dem Kanal ist eine Unterbrechung, die allerdings im Moment nicht bearbeitet werden konnte, weil wir schon im unterbrochenen Zustand sind; 2 heißt, der "Kanal arbeitet voll", d.h., wir haben Vollkapazität und 3 wiederum, der "Kanal ist nicht arbeitsfähig". Sie sehen, hier ist eine relativ große Überschneidung, etwas abgewandelt in der Bedeutung dieser Zustandsanzeige und zwar rührt das daher, daß auch die Befehle sehr stark verwandt sind. Alle diese Instruktionen, sowohl die E/A-Instruktionen als auch die Kanalbefehle, können nur durch das Betriebssystem ausgeführt werden, d.h., sie sind privilegiert, der Benutzer kann sie nicht verwenden. Falls der Benutzer in seinem Programm einen solchen Befehl verwendet, wird er unterbrochen, für ihn ist das ein ungültiger Befehlscode.

9.4.2 Abspeichern und Holen der Daten

Wir haben bereits früher gesehen, die wirkliche Abspeicherung der Daten erfolgt durch Zyklenstehlen, d.h., mit anderen Worten, jeder Kanal hat Priorität vor dem Zentralrechner, das ist alles, was Zyklenstehlen bedeutet. Falls der Kanal Zugriff zum Speicher braucht, wird der Zentralrechner für diesen Zyklus ausgeschlossen. Die nächste Frage ist nun, wie verhalten sich die Kanäle unter sich, falls mehrere Kanäle gleichzeitig Zugriff zum Speicher haben wollen? Betrachten wir dazu Abb. 83.

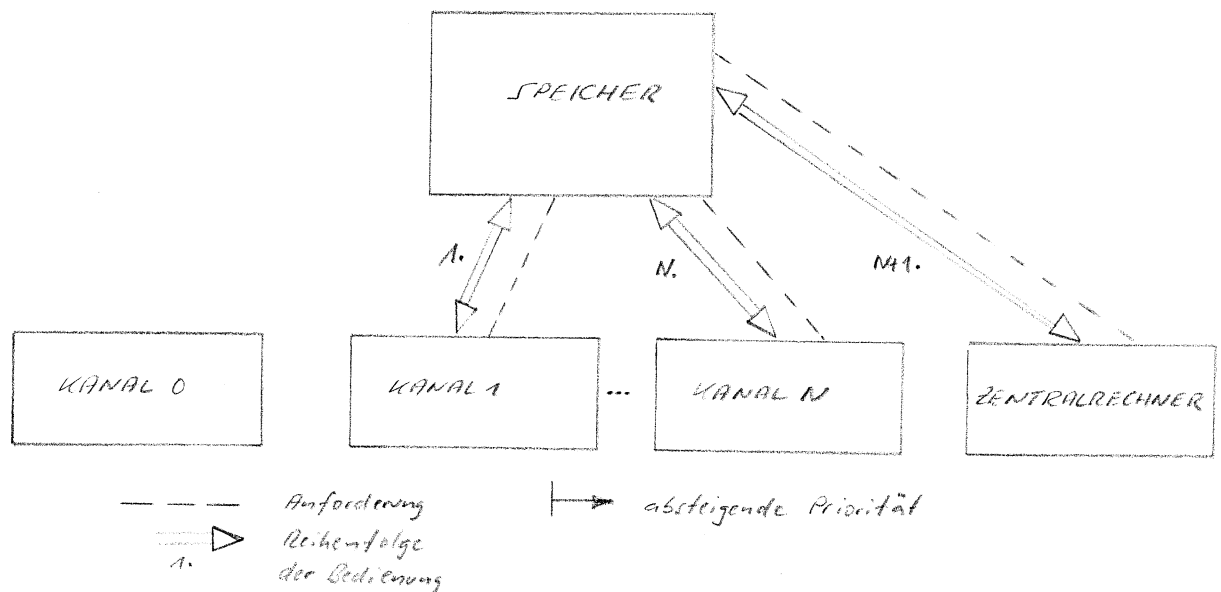


Abb. 83: Bedienung der Kanäle

Wir haben den Speicher, den Zentralrechner und dann Kanal 1, Kanal 2, Kanal 3 und Kanal 4. Und nun sagen wir, daß Kanal 1, 2, 3, 4, falls ein Zugriff existiert, oder eine Anforderung existiert, pro Zugriff Priorität vor dem Zentralrechner haben. Nehmen wir an, daß diese Kanäle nun gleichzeitig oder etwa gleichzeitig Zugriff zum Speicher verlangen. So sehen wir, daß wir eine Prioritätsfolge vereinbaren müssen. Bei den 360/370-Systemen ist folgendes, daß die Kanalnummer die Priorität angibt. Mit höherer Kanalnummer haben wir absteigende Priorität. D.h., die Rei-

Reihenfolge der Bedienung ist folgende: Kanal 1 wird zuerst bedient, Kanal 2 als zweiter, usw. Dies ist sinnvoll, denn nun haben wir die Reihenfolge der Bedienung in der Reihenfolge dieser Nummern. Die Kanaladressen können sehr leicht verglichen werden, und nur derjenige, mit der kleineren Adresse hat Vortritt. Hinzuzufügen ist noch, daß der Speicher nur in vollen Zyklen arbeitet. Während eines Zyklus wird nicht unterbrochen. Der Kanal oder Kanalrechner nimmt nur volle Zyklen aus dem Speicher ab, keine Teile. Lassen Sie mich noch kurz auf die Schnittstelle zwischen Kanal und Geräten eingehen (siehe Abb. 84). Hier haben wir den Kanal, an den eine Kontrolleinheit angeschlossen ist. An dieser Kontrolleinheit sitzen nun ein oder mehrere Geräte. Was als Schnittstelle bezeichnet wird, ist die Verbindung zwischen Kanal und Kontrolleinheit.

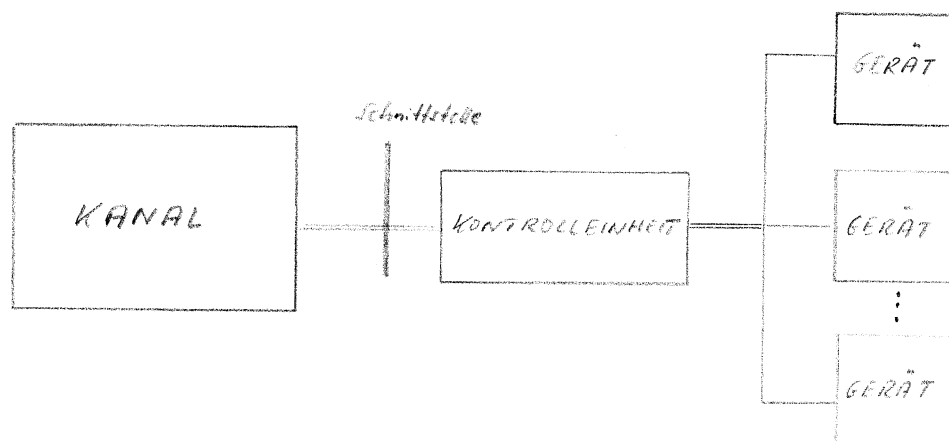


Abb. 84: Schnittstelle zwischen Kanal und Kontrolleinheit

Normalerweise sind alle E/A-Geräte über eine Kontrolleinheit und nicht direkt an den Kanal angeschlossen. Falls wir eine solche Kontrolleinheit einführen, als eine Überstufe über den Geräten, dann erhebt sich die nächste Frage, warum können wir nicht eine einheitliche Schnittstelle definieren? Was wir hier haben wollen, ist also eine

ANHANG

Standardisierung der Schnittstelle jedes E/A-Gerätes über eine entsprechende Kontrolleinheit sofort an das System angekoppelt oder abgekoppelt werden kann, ohne jegliche Änderung im Kanal oder im System selber. Durch das generelle Einführen der Kontrolleinheit ist es möglich, die Schnittstelle zu standardisieren, d.h., wir haben hier dadurch die Steckkompatibilität erhalten. Dies war für die IBM ein großer Fortschritt, eines der umwälzendsten Neuerungen im System 360, und inzwischen hat dies jeder Rechnerhersteller nachkopiert. Es war dazu gedacht, daß die IBM ihre Geräte mit Leichtigkeit an jedes System anschließen kann. Dies wurde nun von den Herstellern der peripheren Einheit dazu ausgenutzt, auswechselbare Einheiten unter den 360-er Geräten zu produzieren, die dann anstelle der IBM-eigenen Geräte angeschlossen werden können, ohne die Arbeiten des Systems zu beeinflussen, weil sie dieselben Charakteristiken gegenüber den Kanälen und dem übrigen System haben.