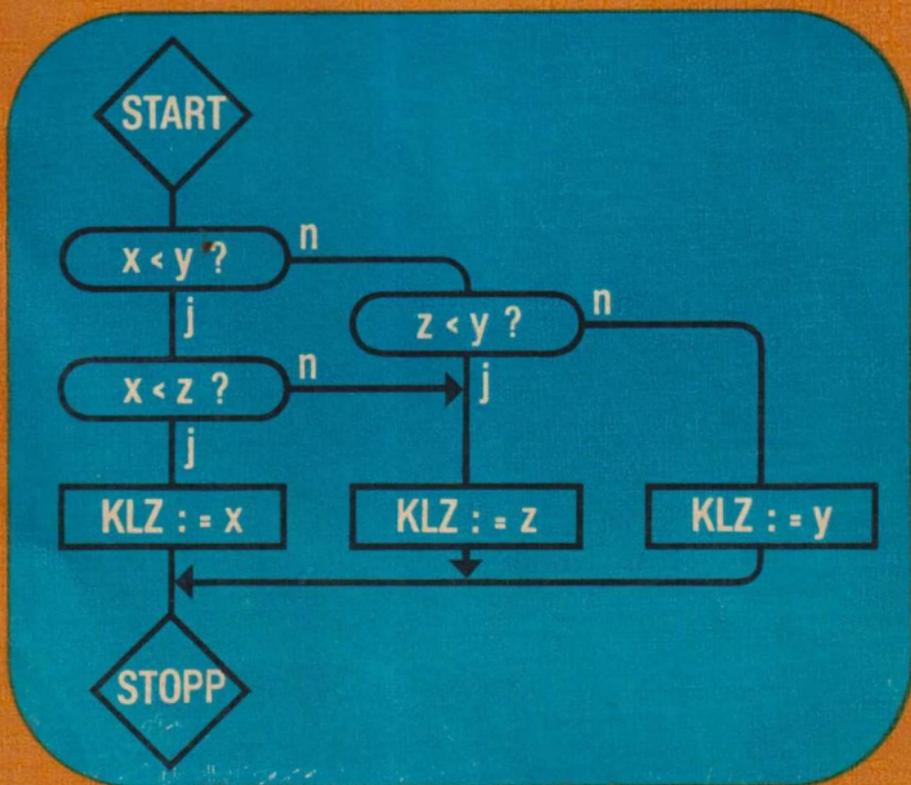


Elektronische Daten- verarbeitung



FRANK STAHL
ERIKA WENZEL



Frank Stahl · Erika Wenzel

Elektronische Daten- verarbeitung



Volk und Wissen
Volkseigener Verlag Berlin
1975

Autoren:

Frank Stahl — Kapitel 4. bis 6.

Erika Wenzel — Kapitel 1. bis 3.; 7.

Dieses Buch wurde unter Nr. 85 in die mathematische Schülerbücherei aufgenommen und kann für Arbeitsgemeinschaften der Klassen 9 und 10 verwendet werden.

© Volk und Wissen Volkseigener Verlag 1975

1. Auflage

Ausgabe 1975

Lizenz Nr. 203/1000/75 (E 001711—1)

LSV 1081

Redaktion: Heinz Junge

Zeichnungen: Heinz Grothmann

Einband und typografische Gestaltung: Atelier vvv, Wolfgang Lorenz

Printed in the German Democratic Republic

Gesamtherstellung: VEB Druckerei „Thomas Müntzer“, Bad Langensalza

Schrift: 10/11 Gill Monotype

Redaktionsschluß: 5. Juli 1974

Bestell-Nr.: 706 491 7

EVP: 3,00 Mark

Inhaltsverzeichnis

1.	Zur Notwendigkeit, zu den Möglichkeiten und Grenzen des Einsatzes von elektronischen Datenverarbeitungsanlagen in der entwickelten sozialistischen Gesellschaft	7
2.	Zur historischen Entwicklung der Datenverarbeitung	15
3.	Zu einigen Grundlagen für die elektronische Datenverarbeitung	25
3.1.	Einige Begriffe, die der Kybernetik entlehnt sind und für die Datenverarbeitung Bedeutung erlangt haben	25
3.2.	Zu einigen Grundlagen aus der Arithmetik	28
3.3.	Zu einigen Grundlagen aus der mathematischen Logik	35
3.4.	Zur technischen Realisierung einfacher logischer Strukturen	39
3.5.	Zu einigen Schaltelementen von EDVA	46
4.	Zum Aufbau und zur Arbeitsweise von EDVA	50
4.1.	Zum Gerätesystem einer EDVA	50
4.2.	Zum Gerätesystem des Kleinrechners Cellatron SER 2d	58
4.3.	Zum Gerätesystem der EDVA Robotron 300	60
5.	Daten, Datenerfassung und Datenträger, Schlüsselssysteme	66
5.1.	Daten	66
5.2.	Datenerfassung und Datenträger	66
5.3.	Codierung und Schlüsselssysteme	70

6.	Zu einigen Grundlagen der Programmierung	74
6.1.	Sprachebenen im Rahmen der Programmierung	74
6.2.	Programmablaufplanung	76
6.3.	Strukturen von Programmablaufplänen	81
6.4.	Maschinensprache	83
6.5.	Maschinenorientierte Programmiersprache (Assemblersprache)	89
6.6.	Beschreibung der internen Operationen und Ein-/Ausgabe- Operationen in MOPS	90
6.7.	Problemorientierte Programmiersprachen	92
7.	Zur Einsatzvorbereitung der EDV in sozialistischen Betrieben	98
7.1.	Allgemeine Probleme	98
7.2.	Organisation der Einsatzvorbereitung von EDVA	99
	Literaturverzeichnis	105
	Register	107

1

Zur Notwendigkeit, zu den Möglichkeiten und Grenzen des Einsatzes von elektronischen Datenverarbeitungsanlagen in der entwickelten sozialistischen Gesellschaft

Ein neues Schuljahr beginnt in der Deutschen Demokratischen Republik. Lehrer und Schüler kehren nach den Sommerferien mit neuen Kräften in ihre Schulen zurück, um gemeinsam die Bildungs- und Erziehungsaufgaben des neuen Schuljahres zu lösen.

Es erhebt sich die Frage, welcher Vorbereitungen es bedarf, damit der Unterricht und alle anderen Tätigkeiten in der Schule pünktlich und reibungslos bereits am ersten Tage anlaufen können.

Die entsprechenden Überlegungen seien hier nur auf einen kleinen Ausschnitt aus dem materiell-organisatorischen Bereich dieser Vorbereitungen gerichtet: Wer sorgt mit welchen Mitteln dafür, daß alle Schüler in allen Klassen 1 bis 12 der allgemeinbildenden polytechnischen Oberschulen für alle Unterrichtsfächer, alle Lehrlinge in den Berufsschulen und Betriebsberufsschulen ihre Schulbücher, Lesestoffe, Arbeitshefte bzw. -blätter, die Materialien für Arbeitsgemeinschaften usw., daß alle Lehrer ihre Unterrichtshilfen und andere pädagogische Literatur zum Schuljahresbeginn in Händen haben?

Verantwortlich für die Verteilung und den rechtzeitigen Versand sind die Mitarbeiter des Leipziger Kommissions- und Großbuchhandels (abgekürzt LKG). In den Schuljahren 1972/73 und 1973/74 beispielsweise mußten sie etwa 7700 Schulen der genannten Arten mit Büchern aus der Lernmittelfreiheit unseres Staates und etwa 4500 Verkaufsstellen in den Schulen und bestimmten Buchhandlungen mit Verkaufstiteln beliefern.

Nun fordert jede Schule und jede Verkaufsstelle mit Hilfe besonderer Bestelllisten andere Anzahlen der benötigten Bücher an. Alle diese Anforderungen fließen beim LKG zusammen. Sie müssen dort zusammengestellt und ausgewertet werden, denn jeder Packer in den Versandabteilungen muß Listen mit genauen, bis auf das Einzelstück stimmenden Angaben, mit genauen Anschriften der Empfänger usw. erhalten.

Die Reihenfolge der Lieferungen an die betreffenden Schulen und Buchhandlungen ist nach einem besonderen Schlüssel sehr sorgfältig festzulegen. Es hat sich in den letzten Jahren auch bewährt, die Auslieferung der Schulbücher klassenstufenweise vorzunehmen. Man geht jetzt ferner dazu über, unter den Bezeichnungen „Volumenversand“ (Bücherkartons mit vollen Stückzahlen) und

„Einzelversand“ (Nachsenden der Bücher, die jeweils die Differenz zu der tatsächlich bestellten Anzahl des betreffenden Titels ausmachen) den Versand weiter zu rationalisieren.

Man kann hier gar nicht alle Besonderheiten auführen, die dabei zu beachten sind. Man muß jedoch wissen, daß der LKG außer den oben betrachteten Büchern, die zum überwiegenden Teil vom Volkseigenen Verlag Volk und Wissen in Berlin entwickelt werden, auch die Buchproduktion aller anderen Verlage der DDR und die aus dem Ausland importierten Bücher zu verteilen hat.

Man kann sich vorstellen, daß ein solcher Aufwand an Arbeit und Arbeitsorganisation in den erforderlichen Zeiträumen von den vorhandenen Mitarbeitern nicht allein mit deren menschlicher Arbeitskraft zu bewältigen ist.

Arbeiter, Techniker und Wissenschaftler haben in kollektiver Arbeit moderne Hilfsmittel geschaffen, die, gelenkt von der schöpferischen Arbeit des Menschen, ihm die langwierigen, ermüdenden rechnerischen Routinearbeiten abnehmen. Es sind „Elektronische Datenverarbeitungsanlagen“ (kurz EDVA), die auch dem LKG die Arbeit erleichtern, ja unter den heutigen Erfordernissen überhaupt erst möglich machen. Die in den Bestelllisten enthaltenen Angaben, die Beispiele für **Daten** sind, werden auf **Datenträger** (→ 66ff.) übertragen, in die **Anlage** eingegeben und von dieser mit Hilfe **elektronischer Schaltelemente** zu neuen, zu den gesuchten Daten **verarbeitet**.

Solche Anlagen sollen in diesem Buch in Aufbau und Wirkungsweise vorgestellt werden, und es soll gezeigt werden, wie einfache Aufgaben mit Mitteln der EDV zu lösen sind.

Vor der Darstellung der wissenschaftlich-technischen Voraussetzungen für die Entwicklung von EDVA seien zunächst *Notwendigkeit* und *Möglichkeit* ihres Einsatzes unter den konkreten historischen Bedingungen der entwickelten sozialistischen Gesellschaft untersucht, aber auch *Grenzen* des Einsatzes von EDVA abgesteckt. Dabei werden weitere Beispiele einbezogen werden. Auch auf die Rolle der EDV in den kapitalistischen Ländern unter den dort noch vorhandenen gesellschaftlichen Bedingungen soll hingewiesen werden. Verweise auf Stellen aus den Lehrbüchern insbesondere der Unterrichtsfächer Staatsbürgerkunde und ESP werden die Betrachtungen unterstützen.

Wenn man die Frage stellt, in welche Bereiche des gesellschaftlichen Lebens unseres Landes elektronische Rechenanlagen bereits Einzug gehalten haben, so kann man viele Antworten erhalten.

Viele volkseigene Betriebe benutzen solche Anlagen zur Planung ihrer Produktion, zur Lenkung der entsprechenden Produktionsprozesse, zur Berechnung von Löhnen und Gehältern, die Sparkassen führen damit die bei ihnen eingerichteten Konten, die Deutsche Volkspolizei regelt vielfach auf diese Weise die Verkehrsströme („Grüne Welle“), die Deutsche Reichsbahn stellt so ihre Fahrpläne auf, staatliche zentrale und kommunale Institutionen setzen EDVA zur planmäßigen Versorgung der Einrichtungen des Handels, zur Planung und Abrechnung der Energieversorgung, zur Bereitstellung von Medikamenten und anderen Leistungen des Gesundheitswesens ein, um nur einige bekannte Beispiele zu nennen.

Die Frage nach der Notwendigkeit der Entwicklung und Benutzung von EDVA kann man nur dann richtig beantworten, wenn man von den Erfordernissen unserer sozialistischen Gesellschaftsordnung ausgeht, also von der Hauptaufgabe, die der VIII. Parteitag der SED im Jahre 1972 stellte. Sie besteht „in der weiteren Erhöhung des materiellen und kulturellen Lebensniveaus des Volkes auf der Grundlage eines hohen Entwicklungstempos der sozialistischen Produktion, der Erhöhung der Effektivität, des wissenschaftlich-technischen Fortschritts und des Wachstums der Arbeitsproduktivität.“ [4; S. 81]

Die objektiven Voraussetzungen für die Erreichung dieser Ziele sind die sozialistischen Produktionsverhältnisse, die unter anderem charakterisiert sind durch das gesellschaftliche Eigentum an den Produktionsmitteln. [↗ 3; S. 14f.]

Entsprechend dem derzeitigen Stand der Produktivkräfte ist der Hauptweg zur Erfüllung der Hauptaufgabe die Intensivierung der sozialistischen Produktion. [↗ 4; S. 95ff.]

„Intensivierung der Produktion heißt . . . , die Erzeugung zu steigern, indem wir die vorhandenen Produktionsanlagen und Gebäude besser nutzen und modernisieren, indem wir mit der gleichen Anzahl von Arbeitskräften mehr produzieren.“ [1; S. 45]

Die wichtigste Methode der Intensivierung ist die sozialistische Rationalisierung der Produktion. [↗ 6; S. 94ff.; 4; S. 99]

Die Rationalisierung der Produktion kann nicht sofort und überall gleichzeitig stattfinden, sondern sie vollzieht sich in verschiedenen Stufen. Das sind beispielsweise

- die Modernisierung der vorhandenen Technik;
- die Mechanisierung;
- die Teilautomatisierung;
- die Automatisierung ausgewählter volkswirtschaftlich bedeutsamer Vorhaben. [↗ 5; S. 102ff.]

Die einzelnen Stufen der sozialistischen Rationalisierung der Produktion werden mit den vielfältigsten Mitteln realisiert, beispielsweise auch durch den Einsatz der Datenverarbeitungstechnik. In der entwickelten sozialistischen Gesellschaft ist die Einführung von EDVA eine hochaktuelle Aufgabe. Diese Tatsache fand ihren Ausdruck in den Direktiven des XXIV. Parteitages der KPdSU. Es ist vorgesehen, die Produktion von EDVA im gegenwärtigen Planjahr fünfmal auf das 2,6-fache zu steigern. So entsteht das größte Zentrum für Datenverarbeitung Europas in der sowjetischen Hauptstadt Moskau. Es befindet sich zur Zeit noch im Bau. Nach seiner endgültigen Fertigstellung wird es 108 m hoch sein und damit zu den höchsten und architektonisch schönsten Gebäuden Moskaus zählen.

Insgesamt kann es 2000 große und kleine Rechenanlagen aufnehmen. Dort werden Rechenanlagen arbeiten, die durch die Gemeinschaftsarbeit der sozialistischen Länder auf dem Gebiet der elektronischen Rechentechnik im Rahmen des Komplexprogramms des Rates für Gegenseitige Wirtschaftshilfe (RGW) entstanden sind. Man nennt die Familie dieser Rechner kurz **ESER** (Abkürzung für „**E**inheitliches **S**ystem der **e**lektronischen **R**echentechnik“; ↗ 20).

Einsatz		
in Wissenschaft und Technik	zum Lösen ökonomischer Probleme	zum Überwachen und Steuern von Prozessen
z. B. für das Berechnen statischer Eigenschaften von Brücken, Wohnhäusern und anderen Bauwerken; für das Berechnen von Linsensystemen	z. B. beim Festlegen der Kennziffern für die Volkswirtschaftspläne; beim Bilanzieren finanzieller Mittel	z. B. durch Prozeßrechner für das Fahren von Taktstraßen, für das Regeln von Verkehrsströmen

Zur Erläuterung des Begriffes „Prozeßrechner“ und als Beispiel für den Einsatz von Prozeßrechnern sei zunächst das Problem der Verkehrsregelung herausgegriffen.

Um auf allen verkehrsreichen Kreuzungen in allen Städten der DDR einen zügigen, unfallfreien Verkehrsstrom nach althergebrachter Weise zu sichern, wäre eine solche Vielzahl von Volkspolizisten nötig, die gar nicht zur Verfügung stehen. An deren Stelle lenken heute vielfach Prozeßrechner, für den Verkehrsteilnehmer unsichtbar installiert, die Leiteinrichtungen (Ampeln, Geschwindigkeitsanzeiger) aller angeschlossenen Kreuzungen einer ganzen Stadt oder eines Stadtbezirks. Zuvor jedoch müssen die Verkehrsverhältnisse im Einsatzgebiet eines solchen Prozeßrechners vom Menschen genau analysiert werden. Zum Beispiel ist die Verkehrsdichte an der betreffenden Kreuzung zu bestimmten Tageszeiten zu messen und zu registrieren, es ist zu prüfen, ob es zulässig ist, die Kraffahrzeuge in alle Richtungen abbiegen zu lassen, es ist zu beachten, an welchen Stellen und in welchen Zeitabständen die Fußgänger die Fahrbahnen überschreiten sollen, und anderes mehr.

Aus der Gesamtheit der festgestellten vorhandenen Bedingungen und der gesammelten Erfahrungen — also der notierten Daten — wird dann vom Menschen ein **Modell** (↗ 26) entwickelt. Es wird eine Handlungsvorschrift für den Prozeßrechner, ein **Programm**, erarbeitet (↗ 74f.). Denn nur auf Grund eines solchen Programmes, das der Mensch dem Rechner „eingeben“ muß, kann dieser seine logischen Entscheidungen treffen, also zum Beispiel die Fahrtrichtungen freigeben oder sperren.

Das bedeutet also allgemein, daß der Mensch plant, den Verlauf eines Prozesses vorausdenkt und damit die Varianten der Lösungen einer Aufgabe dem Prozeßrechner vorgibt. Man kann sagen, daß Prozeßrechner und die EDVA überhaupt taktische Instrumente zur Lösung der vom Menschen gestellten Aufgaben sind, denn der Mensch und nicht die Datenverarbeitungstechnik ist für die Ergebnisse verantwortlich.

Nun heißt „Einsatz von Prozeßrechnern als Verkehrsregler“ natürlich nicht, daß die Verkehrspolizisten überflüssig werden. Im Gegenteil, sie können sich anderen Aufgaben zuwenden, die die Sicherheit auf unseren Straßen zusätzlich erhöhen. Dazu gehört beispielsweise der erzieherische Einfluß auf das richtige

Verkehrsverhalten der Bürger und besonders der Kinder. Ein anderer Aspekt für den Einsatz von Prozeßrechnern ist die Verbesserung der Arbeits- und Lebensbedingungen der Verkehrspolizisten. Dabei sei bedacht, daß sich mit zunehmender Verkehrsdichte die Abgasentwicklung besonders an den Kreuzungen erhöht und daß der Verkehr nicht nur bei schönem Wetter rollt.

Ein weiteres Beispiel zeige die Notwendigkeit, bei der Erfüllung der vom VIII. Parteitag der SED formulierten Hauptaufgabe die Energieerzeugung zu erhöhen und zu rationalisieren.

Der Bedarf an Elektroenergie verdoppelt sich im Zeitraum von jeweils etwa zehn Jahren. Große Energiemengen müssen produziert und für unsere Volkswirtschaft bereitgestellt werden. Vorhandene Kraftwerke werden modernisiert, aber auch neue gebaut, zum Beispiel das Kraftwerk Boxberg II, das Kraftwerk Hagenwerder III, das Kernkraftwerk Nord. [7; S. 8 und 32ff.]

Beim Neubau von Kraftwerken wird die Tendenz sichtbar, die Kraftwerke selbst leistungsfähiger zu machen. Es kommen Kraftwerksblöcke — so nennt man die Einheiten von Dampferzeuger, Turbine, Generator — von 500 Megawatt zum Einsatz.

Um die Blöcke zu beherrschen, müssen Tausende von Informationen in kurzer Zeit verarbeitet werden. Zu bedenken ist auch, daß der Wert des dem Menschen anvertrauten Volkseigentums und damit seine Verantwortung der Gesellschaft gegenüber gestiegen ist.

Zur Steuerung und Überwachung dieser Anlagen werden Prozeßrechner eingesetzt. Wie bereits am Beispiel der Verkehrsregelung gezeigt, sind auch hier gründliche Analysen des gesamten Prozesses der Energieerzeugung notwendig, bevor ein Rechner zum Einsatz kommt.

Das nächste Beispiel für den Einsatz von EDVA in der DDR und in komplexem Maßstab in den sozialistischen Ländern bezieht sich auf die sich immer mehr durchsetzende Erkenntnis, daß es für den Sieg des Sozialismus über den Imperialismus darauf ankommt, den Frieden in Europa und in der ganzen Welt zu sichern. Deshalb muß die Einsatzbereitschaft und Schlagkraft der Streitkräfte der sozialistischen Armeen stets unter anderem mit Hilfe der modernsten Militärtechnik gewährleistet sein. Das heißt beispielsweise, den Luftraum über den Territorien der sozialistischen Länder gegen überraschende feindliche Angriffe optimal zu sichern. So sind auch die Kräfte der Luftverteidigung der DDR mit Prozeßrechnern ausgestattet, die eintreffende Radarimpulse in Sekundenschnelle in Berechnungen zur Lenkung der Fliegerabwehrmittel umsetzen. Denn bei den hohen Geschwindigkeiten der modernen Kampfflugzeuge wäre es mit den früher üblichen Mitteln gar nicht möglich, rechtzeitig zu reagieren.

Stäbe moderner Armeen brauchen zur Führung großer Truppenverbände, die insbesondere das reibungslose Zusammenwirken von Land-, See- und Luftstreitkräften erfordert, unbedingt solche Hilfsmittel wie die EDV. Das haben vor allem alle großen Manöver der Armeen der Warschauer Vertragsstaaten gezeigt. Wir brauchen auch nur an die komplizierten Probleme der Versorgung der Armeen mit Verpflegung, Treibstoff, Munition, Pioniermaterial und vielen anderen Ausrüstungen zu denken.

Viele junge Menschen unserer Republik leisten zur Sicherung der sozialistischen Errungenschaften der DDR ihren Ehrendienst in den Reihen der Nationalen Volksarmee. Darunter gibt es nicht wenige, die unmittelbar an Geräten der militärischen EDVA eingesetzt sind. Es ist deshalb von Nutzen, sich durch den Erwerb entsprechender Kenntnisse und Fähigkeiten darauf vorzubereiten. Auch die Armeen der imperialistischen Staaten verfügen über eine moderne Militärtechnik, darunter über EDVA. Vergangenheit und Gegenwart beweisen jedoch, daß die imperialistischen Machthaber diese Mittel vor allem für militaristische Zwecke mißbrauchen, nach dem Wolfsgesetz, nach dem der Imperialismus von jeher angetreten ist, seine Profitinteressen zum Schaden und auf Kosten fremder Völker durchzusetzen. Dazu kommt die Angst der Imperialisten vor der zunehmenden Ausbreitung der Ideen des Sozialismus und Kommunismus. Das vietnamesische Volk mußte die schwersten Kämpfe unter größten Opfern gegen die US-amerikanische Militärmacht und die Söldnertruppen des Saigoner Regimes führen. Die USA hatten Riesenmengen an Kriegsmaterial nach Südvietnam hineingepumpt, wozu sie auch ihre militärischen EDVA einsetzten.

Zur Nutzung der Errungenschaften des wissenschaftlich-technischen Fortschritts unter kapitalistischen Verhältnissen heißt es im Rechenschaftsbericht des XXIV. Parteitagés der KPdSU:

„Die Monopole nutzen in großem Umfang die Errungenschaften des wissenschaftlich-technischen Fortschritts, um ihre Position zu festigen, um die Effektivität und das Entwicklungstempo der Produktion zu erhöhen sowie die Ausbeutung der Werktätigen und ihre Unterdrückung zu verstärken.“ [2; S. 21]
Mit dieser Zielsetzung stößt der Einsatz der EDV in den imperialistischen Ländern auf gesellschaftliche Schranken.

Es ist in der Tat so, daß zwar die EDV als Ausdruck eines hohen Entwicklungsstandes der Produktivkräfte ein geeignetes Mittel ist, um komplizierte Produktionsprozesse zu beherrschen. Aber auch der Einsatz der EDV dient stets den Interessen der jeweils herrschenden Klasse, im Sozialismus also den Interessen der Arbeiterklasse und des ganzen werktätigen Volkes, im Kapitalismus den Profitinteressen der Feinde der Arbeiterklasse.

Auf beiden Seiten trägt der Einsatz von EDVA zwar zur Erhöhung der Arbeitsproduktivität bei, jedoch unter völlig entgegengesetzten Zielstellungen und Auswirkungen für die Masse der Werktätigen. Die Streikbewegungen in den hochentwickelten kapitalistischen Ländern richten sich zum Beispiel auch gegen die „Herausrationalisierung“ zahlreicher Arbeiter aus ihren Arbeitsplätzen, an denen sie oft jahrzehntlang für die Unternehmer gearbeitet hatten. In vielen Betrieben gibt es mit Hilfe von EDVA ausgeklügelte Antreibersysteme, die den Gesundheitszustand vieler Arbeiter bis zur Frühinvalidität untergraben und eine erschreckend hohe Anzahl von Arbeitsunfällen zur Folge haben.

Das Grundproblem der kapitalistischen Produktionsweise, der Widerspruch zwischen Kapital und Arbeit, kann nicht durch den Einsatz der EDV gelöst werden, sondern nur durch den revolutionären Kampf der arbeitenden Menschen zur Herstellung sozialistischer Produktionsverhältnisse (↗ [6; S. 107f.] und [3; S. 39 und 71]).

Die Politik von Partei und Regierung in der DDR und in den sozialistischen Ländern orientiert ausdrücklich auf die schrittweise Verbesserung der Arbeits- und Lebensbedingungen der Werktätigen auch bei der Einsatzvorbereitung und beim Einsatz von EDVA in der Produktion und allen anderen Lebensbereichen der entwickelten sozialistischen Gesellschaft (↗ auch Teil 7.).

Es sei abschließend betont, daß auch in der entwickelten sozialistischen Gesellschaft dem Einsatz von EDVA Grenzen gesetzt sind. Diese Grenzen entsprechen jedoch dem Charakter unserer Gesellschaftsordnung.

Unsere leitenden Wirtschaftsorgane müssen sorgfältig überlegen, in welchen Betrieben, bei welchen Produktionsprozessen der sehr hohe Aufwand, der mit dem Einsatz von EDVA verbunden ist, tatsächlich lohnt. Wir müssen sehr genau mit den erforderlichen finanziellen Investitionen haushalten. Allein die Kosten für die Datenerfassung machen einen sehr hohen Prozentsatz der erforderlichen Gesamtkosten aus (↗ 67). Wir können auch mit Rücksicht auf die Gesamtplanung unserer Arbeitskräfte nicht in unbegrenzter Anzahl Facharbeiter bzw. Ingenieure für EDV ausbilden. Nicht jeder Jugendliche, der persönliche Neigung dazu verspürt, wird einen entsprechenden Beruf erlernen und ausüben können. Hier gilt es also, die persönlichen Wünsche mit den gesellschaftlichen Erfordernissen sinnvoll zu koordinieren, um höchste Effektivität auf möglichst rationelle Weise zu erzielen.

Man muß vor allem bedenken: Eine Maschine, auch wenn sie eine EDVA von hoher Vollkommenheit im Sinne des wissenschaftlich-technischen Fortschritts ist, kann stets nur ein Hilfsmittel sein, das hilft, die klugen Ideen, die Schöpferkraft seines Erfinders und Beherrschers, des Menschen, in die Realität umzusetzen, im Dienste der Weiterentwicklung der sozialistischen und kommunistischen Gesellschaft.

ZUSAMMENFASSUNG

1. Die Entwicklung und Einführung von EDVA in zahlreichen industriell hochentwickelten Ländern der Erde ist ein wichtiges Ereignis des wissenschaftlich-technischen Fortschrittes seit Mitte unseres Jahrhunderts. Bestimmte Prozesse könnten bei dem heutigen Entwicklungsstand der Technik ohne EDVA gar nicht mehr oder nur in unzureichendem Maße gesteuert werden. Bei dieser Einschätzung sind stets die historisch konkreten gesellschaftlichen Bedingungen zu beachten, unter denen die EDV in dem betreffenden Land zum Einsatz gelangt, und damit die Unterschiede in den Zielstellungen und die Auswirkungen auf die werktätigen Menschen.
2. Die sozialistischen Produktionsverhältnisse in der DDR erfordern und begünstigen auch eine umfassende Anwendung der EDV unter der Beachtung der gegebenen ökonomischen Möglichkeiten. Als Mittel zur Intensivierung der Produktion wirkt sich der Einsatz von EDVA positiv auf die Verbesserung der Arbeits- und Lebensbedingungen der Arbeiterklasse und aller Werktätigen aus.
Das gilt auch für die anderen sozialistischen Staaten und insbesondere für die Realisierung des Komplexprogrammes im Rahmen des RGW.

3. Es wird möglich, die Menschen in zunehmendem Maße von zahlreichen ermüdenden Routinearbeiten zu befreien. Dadurch können sie sich unter verstärkter Ausnutzung ihrer schöpferischen Fähigkeiten mit der Lösung ihrer Aufgaben befassen. Das wiederum wirkt sich positiv auf die Entwicklung sozialistischer Persönlichkeiten aus.
4. In der DDR werden bereits in zahlreichen Bereichen des gesellschaftlichen Lebens, vor allem in der Produktion materieller Güter, EDVA eingesetzt
 - als leistungsstarke Rechenmaschinen;
 - zur Speicherung und Verarbeitung großer Mengen von Informationen;
 - zur Überwachung und Steuerung von Prozessen.
5. Diese Entwicklung erfordert eine sorgfältige Ausbildung von Wissenschaftlern, Technikern und Facharbeitern, die den Erwerb entsprechender Kenntnisse aus den Gesellschaftswissenschaften, insbesondere aus der Politischen Ökonomie des Sozialismus, aus der Mathematik und den Naturwissenschaften und aus bestimmten technischen Disziplinen einschließt.
Die Grundlagen dafür vermittelt der Unterricht der sozialistischen Oberschule, vor allem in den Fächern ESP, Mathematik, Physik und Staatsbürgerkunde.



AUFTRÄGE

1. Begründen und erläutern Sie an Beispielen aus dem Gesundheitswesen, dem Militärwesen und aus Bereichen der materiellen Produktion den Einsatz von EDVA in der DDR!
Gehen Sie dabei von der gesellschaftlichen Zielsetzung aus und stellen Sie besonders die Aufgaben des Menschen in diesem Zusammenhang dar!
2. Informieren Sie sich über die Rechnerarten der ESER-Reihe!
 - Durch welche Merkmale sind alle Modelle gekennzeichnet?
 - Welche Vorteile bietet dieses Rechnersystem den Anwendern?
 - Wodurch werden die Vorteile der sozialistischen internationalen Arbeitsteilung im RGW bei der Entwicklung und Produktion der ESER-Rechner sichtbar?

2

Zur historischen Entwicklung der Datenverarbeitung

Seit frühester Zeit haben sich die Menschen darum bemüht, Rechenhilfsmittel zu entwickeln. Das einfachste Hilfsmittel zum Rechnen und zum Darstellen von Mengen sind die Finger der menschlichen Hand. Auch andere Hilfsmittel können zum Darstellen von Zahlenmengen genutzt werden, zum Beispiel zehn Holzstäbchen oder zehn Kugeln. Bereits im Altertum reihte man Kügelchen in Gruppen zu je zehn Stück auf Fäden und spannte sie in einen Rahmen. Die Kugeln konnte man leicht und schnell bewegen. Ein solches Rechengerät nannte man *Abakus*¹⁾. In manchen Ländern werden noch heute derartige Rechenhilfsmittel verwendet.

Von diesen einfachen Rechenhilfsmitteln bis hin zur elektronischen Datenverarbeitung war ein langer Weg zurückzulegen, und zahlreiche Erfindungen waren notwendig, um schrittweise den heutigen Stand zu erreichen. Im folgenden werden die wichtigsten Etappen bis zur Entwicklung moderner Datenverarbeitungsanlagen beschrieben.

Die erste Etappe beginnt im 16. Jahrhundert und dauert bis zur Mitte des 19. Jahrhunderts. Das ist die Zeit, in der die **mechanischen Rechenmaschinen** entstanden. Zu ihren Konstrukteuren gehören unter anderen die Mathematiker *John Neper* (1550 bis 1617), *Blaise Pascal* (1623 bis 1662) und *Gottfried Wilhelm Leibniz* (1646 bis 1716). In dem von *Pascal* im Jahre 1642 konstruierten Rechenapparat wurden einige Grundmechanismen verwendet, die auch heute noch in Rechenmaschinen zu finden sind.

Pascal benutzte beispielsweise ein Zahnrad mit zehn Zähnen, die die zehn Ziffern von 0 bis 9 darstellen (Bild 1). Mit Hilfe eines solchen sogenannten Dezimal-Zählrades kann man jede gewünschte Zahl von 0 bis 9 einstellen. Stellt man mehrere Dezimal-Zählräder in geeigneter Weise zusammen, so kann man auch Zahlen mit mehreren Stellen speichern. Im Bild 2 ist die Zahl 583 eingestellt.

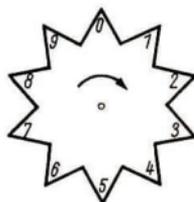


Bild 1

¹⁾ *Abakus* bedeutet „Rechenbrett“.

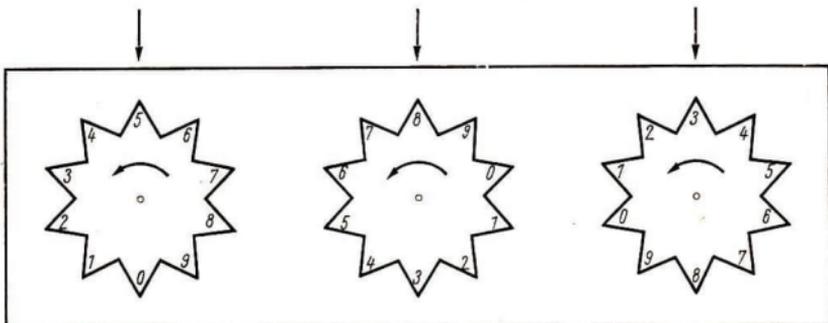


Bild 2

Man sieht: Zum Einstellen der Zahl 583 wird soweit gedreht, bis der Zahn der gewünschten Ziffer einer Markierung gegenübersteht. Die Endstellung der drei Dezimal-Zählräder zeigt dann die Zahl 583 an.

Will man zu der eingestellten Zahl eine andere Zahl addieren, beispielsweise die Zahl 312, so dreht man das erste Rad um 3 Schritte weiter — auf die Ziffer 8 —. Das zweite Rad wird um 1 Schritt weitergedreht — Ziffer 9 —, und schließlich das dritte Rad um 2 Schritte weiter, auf Ziffer 5. Die drei Dezimal-Zählräder zeigen dann die Summe an: 895.

Bei diesem Verfahren der Darstellung einer Addition mit Hilfe von Dezimal-Zählrädern hatte *Pascal* die Schwierigkeit der automatischen Zehnerübertragung zu überwinden, die die Voraussetzung dafür ist, daß *alle* Additionsaufgaben mit Hilfe der Maschine gerechnet werden können. Das Problem der Zehnerübertragung ist ein Grundproblem für jeden Konstrukteur von Rechenmaschinen, gleichgültig, ob es sich dabei um eine mechanische oder um eine elektronische Rechenmaschine handelt. Eine mechanische Additionsmaschine muß also „selbst erkennen“, wann ein Zehnerübertrag zu erfolgen hat. Ein Beispiel dafür, daß eine Lösung möglich ist, ist der Kilometerzähler. Beim Über-

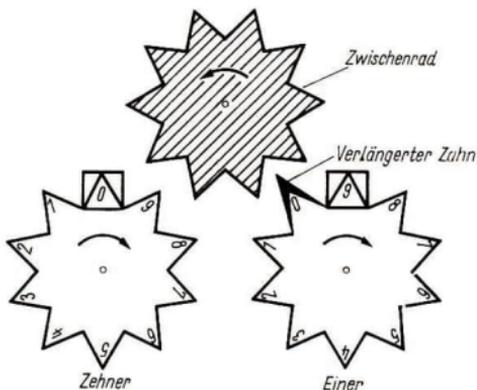


Bild 3

gang von einer 9 zu einer 0 überträgt er selbsttätig eine 1 in die nächsthöhere Stelle.

Durch die Entwicklung der im Bild 3 dargestellten Zehnerübertragung durch *Pascal* war es gelungen, eine Funktion, die vorher der Mensch ausübte, auf die Maschine selbst zu übertragen. Die von *Pascal* und den anderen Konstrukteuren jener Zeit entwickelten Maschinen blieben jedoch ohne Bedeutung für die Praxis. Es fehlten noch andere technische Hilfsmittel für die Herstellung zuverlässig arbeitender Maschinen.

Das Ziel, die Rechenarbeit zu vereinfachen, wurde auch nach diesen ersten, weniger erfolgreichen Versuchen nicht aufgegeben. *Charles Babbage* (1792 bis 1871), ein Universitätsprofessor an der Universität in Cambridge, versuchte Anfang des 19. Jahrhunderts, eine automatische Rechenmaschine zu bauen. In England nannte man damals diese Maschine *Babbage's -Folly*.¹⁾ Die Konstruktionsprobleme, die dabei auftraten, waren zu groß, als daß sie mit den damals zur Verfügung stehenden technischen Mitteln zu bewältigen waren, so daß der Versuch scheitern mußte. Die Gedanken, die *Babbage* für sein Werk entwickelte, bildeten jedoch auch in späterer Zeit, als der angestrebte Rechner auf elektronischer Basis endlich realisiert werden konnte, eine wichtige Grundlage.

Babbage nannte als grundlegende Bauteile den **Speicher**, die **Recheneinheit** und die **Steuereinheit**. Diese drei Elemente gehören heute zur **Zentraleinheit** einer jeden EDVA.

Babbage wies auch schon darauf hin, daß das Stoppen der Maschine nach jedem Rechenvorgang zu einem entscheidenden Zeitverlust führt und daß zur Erhöhung der Rechengeschwindigkeit sowohl die Dateneingabe als auch der Rechenvorgang beschleunigt werden müssen. Er erkannte bereits die Notwendigkeit eines **Datenspeichers** und eines **Befehlsspeichers**, denn zur effektiven Ausnutzung der Rechengeschwindigkeit müssen sowohl Daten als auch Befehle sofort zur Verfügung stehen.

Die zweite Etappe umfaßt den Zeitraum von 1880 bis 1930. In dieser Zeit entstanden die Lochkartenmaschinen, die auf mechanischer bzw. elektromechanischer Grundlage arbeiteten. Der Erfinder dieser Lochkartenmaschinen war der Amerikaner *Hollerith*. Bei diesen Maschinen werden in Lochkarten gestanzte Informationen (Zahlenwerte) automatisch nach einem vorher festgelegten Programm verarbeitet. Diese Lochkartenmaschinen wurden zuerst bei der zehnten amerikanischen Volkszählung in den Jahren 1880 bis 1882 angewendet. Hierbei wurden damals Zahlplättchen verwendet, die sich mit den Maschinen auswerten ließen und als Vorgänger der heutigen Lochkarten anzusehen sind.

Die Lochkartenmaschinen, genannt „Statistikmaschinen“, wurden auch 1910 bei der deutschen Volkszählung mit Erfolg eingesetzt. Damit wurde die **Lochkarte** zum wichtigen Datenträger für die maschinelle Auswertung von statistischen Arbeiten.

¹⁾ *folly* (engl.) — Narretei

Eine neue Etappe begann um 1930. Sie ist gekennzeichnet durch die Entwicklung von elektromechanischen Relaisrechenautomaten, durch elektronische Büromaschinen und Datenverarbeitungsanlagen.

Die Notwendigkeit der Weiterentwicklung ergab sich durch die rasch wachsende Anzahl von Informationen, die zu verarbeiten war. Die manuelle Auswertung dieser Datenträger war zu zeitaufwendig geworden und verlangte einen zu großen Kreis von Mitarbeitern. Durch die Weiterentwicklung der Relais-technik und die Erfindung der Elektronenröhre waren die technischen Voraussetzungen für eine Weiterentwicklung der Datenverarbeitungsrichtungen gegeben.

Die Entwicklung begann in mehreren Ländern gleichzeitig. Der Berliner Ingenieur *Konrad Zuse* baute 1941 einen komplizierten Rechner. Sein Ziel war, die Routineberechnung bei Bauwerken zu automatisieren. Der **Relaisrechner**, genannt Z 1, arbeitete mit etwa 3000 Relais. Ein anderer Relaisrechenautomat war die **OPREMA (Optikrechenmaschine)**. Sie wurde von den Wissenschaftlern des VEB Carl Zeiss Jena im Jahre 1954 entwickelt und war für optische Berechnungen vorgesehen. Der Automat war mit annähernd 17000 Relais und mit 90000 Selengleichrichtern ausgerüstet.

In dieser Etappe entstanden auch die elektronischen Rechenanlagen, bei denen Elektronenröhren an die Stelle der Relais traten. Einer der ersten elektronischen Rechenautomaten, ENIAC, wurde 1946 an der Pennsylvania-Universität (USA) in Betrieb genommen. Er war mit etwa 18000 Elektronenröhren ausgerüstet und arbeitete mit einer damals relativ hohen Rechengeschwindigkeit. Jedoch durch die große Anzahl von Lötstellen und die starke Wärmeentwicklung war er außerordentlich störanfällig und daher nicht genügend praxiswirksam geworden.

Das stürmische Fortschreiten von Wissenschaft und Technik beeinflusste wesentlich auch die Weiterentwicklung der Rechenanlagen.

In diesem Zusammenhang entstand der Begriff **Rechnergeneration**.

Zum gegenwärtigen Zeitpunkt unterscheidet man fünf Entwicklungsstufen von Digitalrechnern (↗ 23). Augenblicklich arbeitet die dritte Rechnergeneration in der Praxis, die vierte und fünfte Generation befinden sich in der Entwicklung beziehungsweise existieren zunächst als Modellvorstellungen. Sie werden jedoch Weiterentwicklungen der dritten Rechnergeneration sein und mit Sicherheit neue Anwendungsbereiche erschließen.

Im folgenden werden wesentliche Merkmale der ersten drei Rechnergenerationen beschrieben und typische Vertreter vorgestellt.

Die **1. Generation** entstand im Zeitraum etwa von 1950 bis 1956.

Als aktive elektronische Bauelemente wurden Elektronenröhren verwendet. Daraus ergab sich für die Anlagen ein großer Platz- und Energiebedarf, etwa 10 bis 100 cm³ beziehungsweise 0,1 bis 10 Watt je Röhre. Durch die große Wärmeentwicklung waren leistungsfähige Kühlaggregate notwendig, die zusätzlich weitere Räumlichkeiten in Anspruch nahmen.

Die Operations- und Rechengeschwindigkeiten lagen im Vergleich zu den heute arbeitenden elektronischen Rechenanlagen niedrig, etwa in der Größen-

ordnung von Millisekunden. Als Hauptspeicher wurden vorwiegend Magnetrommelspeicher (↗ 53f.) verwendet. Das hatte relativ lange Zugriffszeiten zu den gespeicherten Informationen beziehungsweise eine Begrenzung der Speicherplatzanzahl zur Folge. Außerdem waren die Programme für die Rechner in der jeweiligen Sprache des Rechners geschrieben (↗ 75f.) und damit nicht für andere Rechenanlagen verwendbar.

Typische Vertreter dieser Generation waren in der Sowjetunion der **BESM 1** und in der DDR der **ZRA 1**.

Die **2. Generation** kam im Zeitraum etwa von 1957 bis 1965 zur Anwendung. Durch die Entwicklung der Transistorentechnik in den fünfziger Jahren wurde das Entstehen der 2. Generation der elektronischen Rechenautomaten stark beeinflusst, so daß als aktive Bauelemente Transistoren und Halbleiterdioden verwendet wurden. Daraus ergaben sich viele Vorteile gegenüber der 1. Rechengeneration.

Die Abmessungen einer Anlage der 2. Generation sind wesentlich kleiner. Je Bauelement werden etwa 1 bis 10 cm³ benötigt. Der Energiebedarf sinkt auf etwa 10 Milliwatt. Die Rechner arbeiten sicherer und weniger störanfällig, ihre Wärmeentwicklung ist geringer, die Rechengeschwindigkeit erhöht sich erheblich und liegt im Bereich von Mikro- bis Millisekunden. Außerdem ist der Zeitaufwand für die Wartung der Anlagen geringer. Ferritkernspeicher (↗ 54f.) kommen als Speicher zum Einsatz. Dadurch ist die Speicherkapazität erheblich größer, etwa in der Größenordnung von 10⁶ Zeichen.

Ein weiterer Vorteil liegt in der Vereinfachung der Programmierung.

Die Sowjetunion brachte in dieser Zeit die großen leistungsfähigen Datenverarbeitungsanlagen der **URAL-SERIE** heraus. Eine andere bewährte EDVA aus der Sowjetunion ist die **MINSK 22**. Diese Rechner waren maßgeblich bei der Erforschung des erdnahen Raumes und des Mondes beteiligt und dokumentierten die große Leistungsfähigkeit der Sowjetunion auf diesem Gebiet.

Auch die DDR entwickelte einen typischen Vertreter dieser Generation, die EDVA **ROBOTRON 300** (R 300). Im Jahre 1966 wurde sie in Moskau anlässlich der internationalen Fachausstellung „interorg-technica 66“ vorgestellt. Die Leistung der Wissenschaftler, Techniker und Arbeiter der DDR, die auf dem Gebiet der elektronischen Datenverarbeitung tätig sind, fand durch die positive Aufnahme dieser Anlage hohe Anerkennung. Bild 4 zeigt das Gebäude des VEB Kombinat Robotron in Dresden.

In der gegenwärtigen Zeit arbeiten bereits elektronische Datenverarbeitungsanlagen der **3. Generation**.

Diese modernen Rechenanlagen werden

- in Prozessen der Leitung und Planung,
- in Forschung, Entwicklung, Konstruktion und Technologie,
- in Produktionsprozessen,
- für Information und Dokumentation

eingesetzt.



Bild 4

Die Anlagen zeichnen sich durch hohe Rechengeschwindigkeit, große Speicherkapazität und ihre universellen Anwendungsmöglichkeiten aus. Die Speicherkapazität hat beispielsweise die Größenordnung von 10^8 Zeichen, und die Rechengeschwindigkeiten liegen im Nanosekundenbereich.

In der DDR gehört zu diesen Rechnern die EDVA **ROBOTRON 21** (R 21). Die Bauelemente dieses Rechners sind Bausteine der Mikroelektronik. Auf 1 cm^3 kommen jetzt einige Schaltelemente. Diese Anlage ist nahezu stör- und wartungsfrei. Der R 21 (Bild 5) wurde erstmalig 1972 auf der Leipziger Frühjahrsmesse vorgestellt.

Zur 3. Generation der Elektronenrechner gehört ihrem technischen Aufbau nach auch die ESER-Reihe (**E**inheitliches **S**ystem der **e**lektronischen **R**echnetechnik). Sie stellt jedoch qualitativ, von ihrer Konzeption und ihren Anwendungsbereichen her, eine neue Entwicklungsetappe auf dem Gebiet der elektronischen Datenverarbeitungsanlagen dar.

Dieses System von Rechnern entstand unter Führung der Sowjetunion in Zusammenarbeit mit den anderen sozialistischen Ländern mit dem Ziel der Vertiefung der sozialistischen ökonomischen Integration der Mitgliedsländer des RGW auf dem Gebiet der Rechentechnik. Für die EDV-Anwendung sowohl in der DDR als auch in den anderen Partnerländern wird es für die nächsten Jahre typisch sein.

Das System besteht aus verschiedenen Modellen, einer in sich geschlossenen Typenreihe von gleichem Aufbau, mit gleichen Bauelementen und Programmiersprachen, von gleichem Bedienungs- und Wartungsaufwand. Die Anlagen erlauben beispielsweise Verarbeitungsgeschwindigkeiten von 10^4 bis mehr als 10^6 Operationen in der Sekunde.



Bild 5

Die Rechenanlagen der ESER-Reihe sind ein Beitrag der Mitgliedsstaaten des RGW zur systematischen Kooperation in Wissenschaft, Technik und Produktion (↗ [4, S. 122 ff.]).

Die Datenverarbeitungsanlagen der ESER-Reihe wurden auf der Leipziger Frühjahrsmesse 1973 vorgestellt. Sie sind im einzelnen:

EDVA	Operationsgeschwindigkeit in Operationen/s	Hauptspeicherkapazität in K-Bytes ¹⁾
VR Ungarn ES 1010	10000	8 bis 64
UdSSR ES 1020 VR Bulgarien	20000	64 bis 256
UdSSR ES 1030 VR Polen	100000	128 bis 512
DDR ES 1040	380000	256 bis 1024
UdSSR ES 1050	500000	128 bis 1024

Bild 6 zeigt die EDVA ES 1040.

¹⁾ „Byte“ ist die Grundeinheit für den Datenaufbau. Sie besteht für die ESER-Rechner aus 8 Binärzeichen (↗ 30). „K-Byte“ ist ein Vielfaches dieser Größe (1 K ≙ 1024).

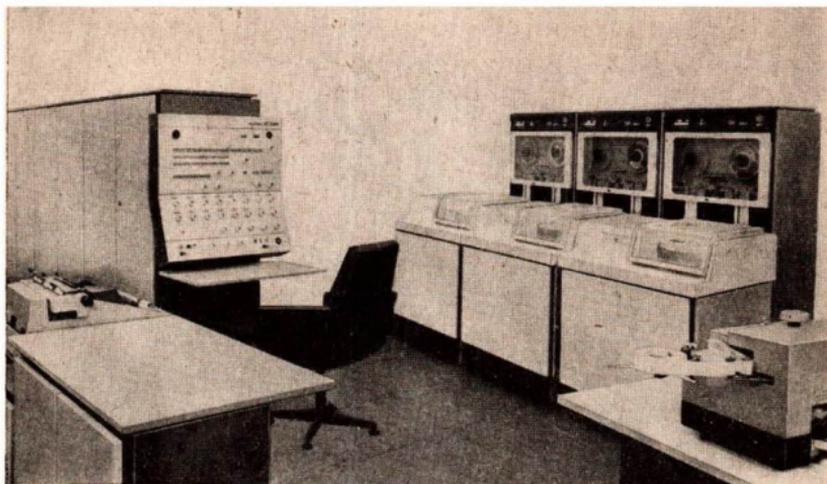


Bild 6

Datenverarbeitungsanlagen können auch für die Informationserfassung, -verarbeitung und -nutzung bei materiellen Prozessen eingesetzt werden. Man nennt sie dann **Prozeßrechner** (↗ Teil 1.). Dabei können sie auch Steuerungs- und Regelungsfunktionen übernehmen.

Die Technik der Prozeßrechenanlagen ist erst wenige Jahre alt. Die Entwicklung begann nach 1960, nachdem durch die Weiterentwicklung der Meß-, Steuerungs-, Regel- und Nachrichtentechnik die Voraussetzungen für den Aufbau derartiger Anlagen geschaffen waren. Prozeßrechner gewinnen mehr und mehr an Bedeutung. Während im Jahre 1964 etwa 450 Prozeßrechner in der Welt eingesetzt waren, schätzte man 1973 ihre Zahl bereits auf etwa 22.000. Die Prozeßrechentechnik umfaßt Einrichtungen der Informationsverarbeitung in unmittelbarer Verbindung mit ablaufenden Produktionsprozessen. Die wichtigsten Anwendungsgebiete von Prozeßrechnern sind

- die Eisen- und Stahlindustrie (Bergwerke, Hochöfen, Stahl- und Walzwerke),
- die Nichteisenmetallurgie (Herstellung von Aluminium und Kupfer),
- die Zementindustrie,
- die chemische Industrie,
- die Lebensmittelindustrie,
- die Erdölindustrie (Förderung, Transport und Verarbeitung),
- die Elektroenergieerzeugung und die Energieverteilung (↗ 11),
- die Automatisierung der Arbeiten in Lagerhäusern,
- das Verkehrswesen (Ampelsteuerung) (↗ 10f.).

Wesentlich ist, daß bei der Prozeßrechentechnik die Vorgänge der unmittelbaren Informationsverarbeitung im allgemeinen mit den Vorgängen der be-

treffenden Prozesse im Produktionsablauf parallel verlaufen. Dabei werden laufend Informationen über den jeweiligen Stand des Prozeßablaufes erfaßt und dem Prozeßrechner zur Auswertung zugeleitet. Dieser vergleicht Sollwerte mit Istwerten und führt, wenn nötig, Korrekturen am Prozeßverlauf durch.

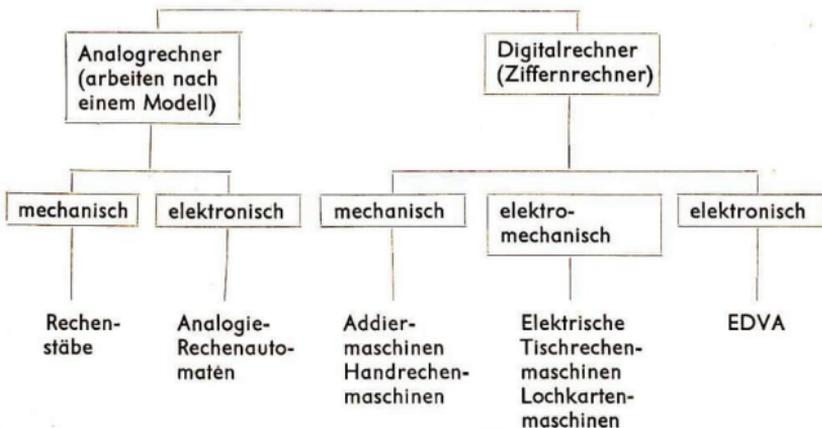
Zu den Hauptaufgaben der Prozeßrechner gehören die Registrierung von Prozeßdaten, die Überwachung eines Prozesses und die Steuerung des Prozeßablaufes.

Man unterscheidet vom Aufbau her zwischen **Analogrechnern** und **Digitalrechnern**.

Analogrechner arbeiten im Unterschied zu Digitalrechnern mit kontinuierlich veränderlichen Größen. Zur Darstellung mathematischer Zusammenhänge werden physikalische Vorgänge benutzt, die denselben Gesetzmäßigkeiten unterliegen, zum Beispiel werden mathematische Operationen und Größen in analoge elektrische Schaltsysteme und Spannungen übersetzt. Die Geräte sind meist aus Recheneinheiten zusammengesetzt; jede Einheit für sich kann eine Rechenoperation oder auch mehrere Rechenoperationen ausführen. Entsprechende Kombinationen solcher Einheiten ermöglichen das Lösen auch komplizierterer mathematischer Probleme.

Digitalrechner arbeiten mit einzelnen Ziffern, von denen jede durch ein besonderes Bauelement dargestellt wird. Prinzipiell wird jedes Problem als eine Folge von Additionen und Subtraktionen dargestellt. Der Hauptvorteil der Digitalrechner gegenüber den Analogrechnern besteht in einer weitaus höheren Rechengenauigkeit. Die elektronischen Rechenmaschinen, die wir in diesem Buch behandeln, sind sämtlich digitale Rechenanlagen.

Übersicht über mathematische Rechengерäte





AUFTRÄGE

1. — Erläutern Sie an Hand einfacher Rechenhilfsmittel die Zehnerübertragung!
 - Beobachten Sie die Zehnerübertragung beim Kilometerzähler (zum Beispiel am Fahrrad) und erklären Sie das Prinzip!
2. Stellen Sie die historische Entwicklung der Datenverarbeitung im Zusammenhang mit der allgemeinen Entwicklung auf gesellschaftlichem und wissenschaftlich-technischem Gebiet dar!
 - Untersuchen Sie die Triebkräfte ihrer Entwicklung!
 - Stellen Sie besonders die Entwicklung der EDV in den sozialistischen Ländern dar! Beachten Sie dabei die führende Rolle der Sowjetunion!
 - Zeigen Sie an Beispielen die Überlegenheit der sozialistischen Produktionsverhältnisse beim Einsatz von EDVA!
3. Erläutern Sie das Funktionsprinzip des Rechenstabes als Beispiel für einen Analogrechner!

3

Zu einigen Grundlagen für die elektronische Datenverarbeitung

Im Teil 3. werden einige grundlegende Erkenntnisse aus denjenigen Wissenschaftsbereichen erarbeitet, die für das Verständnis der Arbeitsweise von EDVA unbedingt notwendig sind, beispielsweise aus der Arithmetik und der mathematischen Logik.

Einige der zu behandelnden Begriffe und Gesetzmäßigkeiten sind aus dem Fachunterricht bekannt. Dem Bekannten sollen neue Erkenntnissen hinzugefügt werden, um zu verstehen

1. Aufbau und Arbeitsweise von EDVA;
2. Einsatzmöglichkeiten von EDVA unter sozialistischen Produktionsverhältnissen, insbesondere solche Möglichkeiten, die sich durch Nutzung von EDVA als Rationalisierungsmittel für die Steigerung der Arbeitsproduktivität und die Befreiung zahlreicher Werktätiger von Routinearbeiten ergeben;
3. Grenzen, die der Anwendung von EDVA gesetzt sind.

3.1. **Einige Begriffe, die der Kybernetik entlehnt sind und für die Daten- verarbeitung Bedeutung erlangt haben**

Die **Kybernetik** ist eine noch sehr junge Wissenschaft. Sie bildete sich erst in den vierziger Jahren unseres Jahrhunderts heraus. Das Wort „Kybernetik“ leitet sich aus einem griechischen Wort ab, das mit „Steuermannskunst“ übersetzt werden kann. Der amerikanische Mathematiker *Norbert Wiener* (1894 bis 1964) gilt als einer der Begründer der Kybernetik.

Wir wollen die Kybernetik als die Wissenschaft bezeichnen, die allgemeine wesentliche Eigenschaften und Gesetzmäßigkeiten der *Regelung* und der *Informationsverarbeitung* in dynamischen biologischen und bestimmten technischen Systemen sowie diese Systeme selbst untersucht (↗ [30 S. 640]).

Der Ablauf eines Vorganges nach einem vorher festgelegten **Programm** wird als **Steuerung** bezeichnet (↗ [6, S. 36 ff.]). EDVA sind „programmgesteuerte“ Maschinen.

Die Analyse der realen Prozesse erfolgt schrittweise. Dabei werden zunächst nur die wesentlichen Seiten untersucht. Man nutzt dazu die folgenden Möglichkeiten.

Modelle

Mit Hilfe eines Modells kann man gewisse Ähnlichkeiten zwischen zwei Objekten veranschaulichen. Dabei sind die Begriffe „Ähnlichkeit“ und „Objekt“ nicht so eng zu sehen, wie man es aus der Umgangssprache gewöhnt ist. Die Ähnlichkeit kann beispielsweise rein äußerlich sein, oder sie bezieht sich auf die innere Struktur äußerlich vollkommen verschiedener Objekte. Die Ähnlichkeit kann sich auch auf bestimmte Verhaltensweisen von Objekten, die weder in Form noch Struktur etwas Gemeinsames haben, beziehen. Wenn zwischen zwei Objekten eine Ähnlichkeit in der eben dargestellten Weise besteht, so daß eines dieser Objekte als das **Original** und das andere als das **Modell** betrachtet werden kann, so spricht man von dem Verhältnis Original – Modell.

Äußere Ähnlichkeit (Ähnlichkeit der Form) haben zum Beispiel ein Haus (Original) und seine Darstellung in einer Zeichnung oder einem räumlichen Gebilde (Modell). Ein Modell ist ein vom Menschen geschaffenes Abbild des betreffenden Originals.

Um mit einem Modell arbeiten zu können, muß es *nicht alle* Ähnlichkeiten mit dem Original aufweisen, sondern beide brauchen *nur in den wesentlichen* Merkmalen übereinzustimmen. Ein derartiges Modell heißt **vereinfachtes Modell**.

Mathematische Modelle sind vereinfachte Modelle. Sie können dargestellt werden zum Beispiel durch

- Gleichungen,
- Tabellen oder graphische Darstellungen.

Andere Modellarten sind beispielsweise

- **verbale Modelle** in Form von Aussagen,
- **physikalische Modelle** (Fahrzeugmodell im Windkanal, physikalische Unterrichtsmodelle),
- **graphische Modelle** (technische Zeichnungen, Schaltpläne, Blockschaltbilder).

Blockschaltpläne

Mit Hilfe eines **Blockschaltplanes** kann man beispielsweise die Funktionsbeschreibung eines einzelnen Gerätes oder einer ganzen Anlage veranschaulichen.

Das kann auf ziemlich einfache Weise erfolgen, indem man zur Darstellung der einzelnen Teile der Anlage rechteckige Symbole oder andere Sinnbilder

(nach TGL 22 451) verwendet. So kann man den Produktionsprozeß in einem Betrieb leichter überschauen, indem man alle wesentlichen Fertigungsstationen dieses Betriebes in Blöcken darstellt, also einen Blockschaltplan anfertigt. Die zwischen den Blöcken bestehenden wichtigen Beziehungen werden mit Hilfe von Pfeilverbindungen angedeutet. Die Pfeilspitzen geben dabei die Orientierungsrichtungen an.

Diese Darstellungsart macht es möglich, den Gesamtinformationsfluß innerhalb einer Anlage zweckmäßig aufzuzeichnen, wobei auch wichtige Kopplungen zu Objekten außerhalb dieser Anlage erfaßt werden können.

Ein Blockschaltplan eignet sich *nicht*,

- um den zeitlichen Ablauf von Prozessen in der Datenverarbeitung darzustellen und
- um die dafür verwendeten Datenträger sichtbar zu machen.

Datenflußpläne

In einem **Datenflußplan** werden die Maßnahmen zur Datenerfassung, Datenverarbeitung und Datenausgabe übersichtlich in der Reihenfolge ihres Ablaufs, mit Terminangaben versehen, dargestellt.

Zum Beispiel wird für die Überwachung eines Produktionsprozesses dargestellt, welche Meldungen über die Bereitstellung von Ausgangsprodukten erfolgen müssen, welche Datenträger für die Aufnahme dieser Meldungen vorgesehen sind, welche Programme für die Beurteilung der Produktionsabläufe verwendet werden, wann der Rechner in Anspruch genommen werden muß und welche Ausdrucklisten ausgegeben werden sollen.

Der Datenflußplan veranschaulicht den *äußeren Ablauf*. Für die Darstellung werden Sinnbilder benutzt, die in der TGL 22 451 festgelegt sind. Danach werden einheitliche Symbole für Ein- und Ausgabevorgänge, für Datenträger (Lochkarte, Lochband, Magnetband) und Speicher sowie für den zeitlichen Ablauf benutzt.

Die graphische Darstellung in Form eines Datenflußplanes ist sehr übersichtlich. Trotzdem können *nicht alle* auftretenden Probleme im Datenflußplan vermerkt sein. Deshalb ist es in der Praxis notwendig, den Datenflußplänen Ergänzungen in Form von Erläuterungen beizufügen.

Datenflußpläne finden vorrangig Verwendung bei der Einsatzvorbereitung, zum Beispiel beim Erarbeiten von Grobprojekten und Feinprojekten (↗ Teil 7.). Sie dienen der Analyse und der Neugestaltung von Informationsprozessen.

Programmablaufpläne

In einem **Programmablaufplan** werden die einzelnen Arbeitsschritte, die vom Rechner auszuführen sind, präzisiert. Auch hierfür gibt es festgelegte Sinnbilder, die im Teil 6. erläutert werden.

Datenflußplan und Programmablaufplan müssen als Einheit betrachtet werden, wobei der Programmablaufplan Einzelheiten der im Datenflußplan ausgewiesenen Arbeitsgänge enthält. Hierzu muß der gesamte Rechenweg in Teilschritte zerlegt und mit Hilfe von Symbolen veranschaulicht werden.

Ein Programmablaufplan entspricht einem **Algorithmus**, einem aus dem Mathematikunterricht bekannten Begriff. Mit Hilfe eines Algorithmus werden vorgegebene Eingangsgrößen nach bestimmten Regeln in bestimmte Ausgangsgrößen umgewandelt. Das Verfahren, nach dem man eine schriftliche Division ausführt, ist zum Beispiel ein solcher Algorithmus. Die Vorschriften, nach denen man vorgegebene Gleichungen oder Ungleichungen in dazu äquivalente Gleichungen bzw. Ungleichungen umformt, sind ebenfalls Algorithmen. Ein Algorithmus kann in unterschiedlicher Weise formuliert sein, beispielsweise:

- verbal,
- mit Hilfe von Zeichenreihen,
- mit Hilfe von Tabellen.

In der EDV hat sich die Darstellung als Programmablaufplan, im folgenden kurz **PAP** genannt, wegen ihrer Eindeutigkeit und Übersichtlichkeit durchgesetzt (↗ 77).

Programmablaufpläne veranschaulichen den *inneren Ablauf* der Operationen in der EDVA, und zwar in seiner zeitlichen und logischen Aufeinanderfolge.

Wir fassen zusammen.

1. EDVA sind **programmgesteuerte** Maschinen.
2. EDVA arbeiten mit **Modellen** beziehungsweise **vereinfachten Modellen** von Objekten der Realität.
3. **Blockschaltpläne** veranschaulichen das System der Informationsverbindungen innerhalb von einzelnen Geräten, von Anlagen oder von ganzen Betrieben. Sie können auch Verbindungen zu außerhalb liegenden Einrichtungen aufzeigen.
4. **Datenflußpläne** veranschaulichen den *äußeren Ablauf* in EDVA in seiner zeitlichen Reihenfolge, von der Datenerfassung über die Datenverarbeitung bis zur Datenausgabe.
5. **Programmablaufpläne** (PAP) veranschaulichen den *inneren Ablauf* in EDVA, das heißt, sie präzisieren die einzelnen Teilschritte der entsprechenden Datenflußpläne in ihrer zeitlichen und logischen Reihenfolge.

3.2. Zu einigen Grundlagen aus der Arithmetik

Positionssysteme

Die Darstellung der natürlichen Zahlen erfolgt im allgemeinen im **dekadischen Positionssystem**. Dabei wird jede Zahl als Summe von Vielfachen von Zehnerpotenzen dargestellt. Als Faktoren der Zehnerpotenzen treten nur die Zahlen 0, 1, 2, 3, 4, 5, 6, 7, 8 und 9 auf.

□ BEISPIEL 3/1:

Die Zahl Neuntausendeinhundertdrei wird wie folgt als Summe dargestellt.

$$9 \cdot 10^3 + 1 \cdot 10^2 + 0 \cdot 10^1 + 3 \cdot 10^0$$

Wenn man vereinbart, die Operationszeichen und die Zehnerpotenzen wegzulassen, so erhält man die Ziffer

9103,

von der man weiß, daß jeder Stelle eine bestimmte Zehnerpotenz als Stellenwert zugeordnet ist. Beispielsweise ist der dritten Stelle von rechts die Potenz $10^3 = 100$ zugeordnet, man spricht deshalb auch von „Hundertern“.

Da im dekadischen Positionssystem die Zahl 10 als Basis der Potenzen auftritt, spricht man auch vom **Dezimalsystem**.

Allgemeine Darstellung:

$$a_n \cdot 10^n + a_{n-1} \cdot 10^{n-1} + \dots + a_2 \cdot 10^2 + a_1 \cdot 10^1 + a_0 \cdot 10^0$$

mit $a \in \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$ und $n \in \mathbb{N}$.

Analog sind unendlich viele Zahlensysteme denkbar, denn jede positive ganze Zahl kann die Basis eines Zahlensystems sein. Die allgemeine Darstellung für ein **Zahlensystem** mit der Basis 8 lautet zum Beispiel:

$$a_n \cdot 8^n + a_{n-1} \cdot 8^{n-1} + \dots + a_2 \cdot 8^2 + a_1 \cdot 8^1 + a_0 \cdot 8^0$$

mit $a \in \{0, 1, 2, 3, 4, 5, 6, 7\}$ und $n \in \mathbb{N}$.

In diesem Zahlensystem sind die Grundziffern die Ziffern 0, 1, 2, 3, 4, 5, 6 und 7. Beim Übergang vom Zahlensystem mit der Basis 8 zum dekadischen Zahlensystem kann man die Vielfachen der Potenzen mit der Basis 8 ausrechnen und dann die Summe bilden.

□ BEISPIEL 3/2:

Die Zahl $(13027)_8$ soll im dekadischen Zahlensystem dargestellt werden.

$$\begin{aligned} & 1 \cdot 8^4 + 3 \cdot 8^3 + 0 \cdot 8^2 + 2 \cdot 8^1 + 7 \cdot 8^0 \\ &= 1 \cdot 4096 + 3 \cdot 512 + 0 \cdot 64 + 2 \cdot 8 + 7 \cdot 1 \\ &= 4096 + 1536 + 0 + 16 + 7 \\ &= 5655 \\ &(13027)_8 = (5655)_{10} \\ &= 5655 \end{aligned}$$



BEISPIEL 3/3:

Es soll die Zahl 333, dargestellt im Dezimalsystem, also $(333)_{10}$, im System mit der Basis 8 dargestellt werden. Dazu muß 333 in eine Summe von Vielfachen der Potenzen von 8 zerlegt werden.

$$8^0 = 1$$

$$8^1 = 8$$

$$8^2 = 64$$

$$8^3 = 512$$

$$8^4 = 4096$$

Die Potenzen 8^3 und größere kommen nicht mehr in Frage, denn $8^3 = 512 > 333$.

Das heißt, die größte vorhandene Potenz ist $8^2 = 64$. Sie ist 5mal in 333 enthalten, denn $333:64 = 5$ Rest 13. Die nächstkleinere Potenz ist $8^1 = 8$. Sie ist 1mal im Rest 13 enthalten, denn $13:8 = 1$ Rest 5. Die nächstkleinere Potenz ist $8^0 = 1$. Sie ist 5mal im Rest 5 enthalten, denn $5:1 = 5$ ohne Rest.

$$\begin{aligned} 333 &= 5 \cdot 64 + 1 \cdot 8 + 5 \cdot 1 \\ &= 5 \cdot 8^2 + 1 \cdot 8^1 + 5 \cdot 8^0 \\ &= (515)_8 \end{aligned}$$

Für die elektronische Datenverarbeitung hat sich besonders das Zahlensystem mit der 2 als Basis bewährt, da zu seiner Darstellung nur zwei Grundziffern erforderlich sind. Die elektronischen Bauelemente können gerade zwei Schaltzustände, die den beiden Grundziffern jeweils zugeordnet werden, unterscheiden.

- „Aus“ — „Ein“
- „Kein Strom“ — „Strom“
- „Kleine Spannung“ — „Große Spannung“

Ein solches System wird **Dualsystem** genannt. Um Verwechslungen zu vermeiden, wird verabredet, daß die duale 1 als „L“ geschrieben wird. Die Zeichen L und 0 heißen **Binärzeichen**¹⁾ (↗ [9, S. 82 f.]).

Beim Zählen im dekadischen System ist nach der 9 der Übergang zu einer zweistelligen Zahl bzw. nach der 99 zu einer dreistelligen Zahl usw. erforderlich. Beim Zählen im Dualsystem ist der erste Übergang bereits nach 2 Schritten erforderlich.

0 — L	Ziffernvorrat erschöpft, es kommt eine Stelle hinzu.
L0 — LL	Nun kommt noch eine Stelle hinzu.
L00 — L0L — LLO — LLL	Die nächste Stelle ist erforderlich.
L000 — L00L — L0LO	usw.

Damit ergibt sich auch die zugehörige Potenzschreibweise:

¹⁾ Binärzeichen ist das kleinste, nicht weiter teilbare Datenelement [bis (lat.) — zweimal].

Dezimalzahl	Dualzahl	Dualzahl in Potenzschreibweise
0	0	$0 \cdot 2^0 = 0$
1	L	$1 \cdot 2^0 = 1$
2	LO	$1 \cdot 2^1 = 2$
3	LL	$1 \cdot 2^1 + 1 \cdot 2^0 = 2 + 1$
4	LOO	$1 \cdot 2^2 = 4$
5	LLOL	$1 \cdot 2^2 + 1 \cdot 2^0 = 4 + 1$
6	LLLO	$1 \cdot 2^2 + 1 \cdot 2^1 = 4 + 2$
7	LLLL	$1 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0 = 4 + 2 + 1$
8	LOOO	$1 \cdot 2^3 = 8$
9	LOOL	$1 \cdot 2^3 + 1 \cdot 2^0 = 8 + 1$
10	LOLO	$1 \cdot 2^3 + 1 \cdot 2^1 = 8 + 2$
11	LOLL	$1 \cdot 2^3 + 1 \cdot 2^1 + 1 \cdot 2^0 = 8 + 2 + 1$
12	LLOO	$1 \cdot 2^3 + 1 \cdot 2^2 = 8 + 4$
13	LLLOL	$1 \cdot 2^3 + 1 \cdot 2^2 + 1 \cdot 2^0 = 8 + 4 + 1$
14	LLLLO	$1 \cdot 2^3 + 1 \cdot 2^2 + 1 \cdot 2^1 = 8 + 4 + 2$
15	LLLLL	$1 \cdot 2^3 + 1 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0 = 8 + 4 + 2 + 1$
16	LOOOO	$1 \cdot 2^4 = 16$

Die allgemeine Darstellung für ein Zahlensystem mit der Basis 2 lautet:

$$a_n \cdot 2^n + a_{n-1} \cdot 2^{n-1} + \dots + a_2 \cdot 2^2 + a_1 \cdot 2^1 + a_0 \cdot 2^0$$

mit $a \in \{0, L\}$ und $n \in \mathbb{N}$.

□ BEISPIEL 3/4:

Die Zahl L0L0L0 soll im dekadischen System dargestellt werden.

$$\begin{aligned} \text{L0L0L0} &= 1 \cdot 2^5 + 0 \cdot 2^4 + 1 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0 \\ &= 32 + 0 + 8 + 0 + 2 + 0 \\ &= 42 \end{aligned}$$

□ BEISPIEL 3/5:

Die Zahl 333 soll im Dualsystem dargestellt werden.

$$\begin{aligned} 333 &= 2^8 + 77 \\ &= 2^8 + 2^6 + 13 \\ &= 2^8 + 2^6 + 2^3 + 5 \\ &= 2^8 + 2^6 + 2^3 + 2^2 + 1 \\ &= 2^8 + 2^6 + 2^3 + 2^2 + 2^0 \\ &= 1 \cdot 2^8 + 0 \cdot 2^7 + 1 \cdot 2^6 + 0 \cdot 2^5 + 0 \cdot 2^4 + 1 \cdot 2^3 + 1 \cdot 2^2 \\ &\quad + 0 \cdot 2^1 + 1 \cdot 2^0 \\ &= \text{L0L00LLOL} \end{aligned}$$



Wir prägen uns ein:

Informationen können durch Folgen von **Binärzeichen** dargestellt werden.

Eine Kombination von 4 Binärzeichen wird **Tetrade** genannt.

Einen Satz aus 5 bis 9 Binärzeichen bezeichnet man als **Byte**.

Die spezielle Zuordnung der Binärzeichen zu den Dezimalziffern heißt **Normalverschlüsselung** oder **rein-duale Verschlüsselung**.

Die Darstellung der Dezimalziffern 0, 1, 2, ..., 9 in Normalverschlüsselung als Tetrade sieht wie folgt aus.

Dezimalziffer	0	1	2	3	4	5	6	7	8	9
Tetrade	0000	000L	00L0	00LL	0L00	0L0L	0LL0	0LLL	L000	L00L

Eine andere Möglichkeit der Verschlüsselung ist die **ziffernweise Verschlüsselung** oder **dezimal-duale Verschlüsselung** von Dezimalzahlen. Bei dieser Verschlüsselung werden die einzelnen Grundziffern jeweils dual verschlüsselt.



BEISPIEL 3/6:

a) Dezimalzahl: 8235

8	2	3	5
L000	00L0	00LL	0L0L

Dezimal-duale Verschlüsselung: L000 00L0 00LL 0L0L.

b) Dezimalzahl: 73

7	3
0LLL	00LL

c) Vergleichsweise Darstellung der Dezimalzahl 89 in beiden Verschlüsselungsarten:

- Rein-duale Verschlüsselung: L0LL 00L
- Dezimal-duale Verschlüsselung: L000 L00L

Beide Darstellungsarten sind gleich wichtig. Beide werden in EDVA angewendet. Bei der dezimal-dualen Verschlüsselung ist von Vorteil, daß die Umrechnung ins dezimale System, man nennt diesen Vorgang **Konvertieren**, nicht so aufwendig ist. Bei der rein-dualen Darstellung werden demgegenüber weit weniger Binärstellen benötigt.

Die Konvertierung von einem Zahlensystem in das andere wird von der Datenverarbeitungsanlage selbst durchgeführt.

Grundrechenoperationen mit Zahlen in dualer Darstellung

Für die **Addition im Dualsystem** sind folgende Rechenregeln erklärt, die sich folgerichtig aus der Übersicht auf Seite 31 ergeben.

$$\begin{aligned} 0 + 0 &= L \\ 0 + L &= L \\ L + 0 &= L \\ L + L &= L0 \end{aligned}$$

□ BEISPIEL 3/7:

$$\begin{array}{r} \text{a) } 13 \quad \text{LLOL} \\ + 5 \quad \text{+ LOL} \\ \hline 18 \quad \text{L00L0} \end{array}$$

Aus der letzten Spalte ergibt sich L als Übertrag, ebenso in den Spalten 3 und 4.

$$\begin{array}{r} \text{b) } 3 \quad \text{LL} \\ + 1 \quad \text{+ L} \\ \hline 4 \quad \text{L00} \end{array}$$

$$\begin{array}{r} \text{c) } 10 \quad \text{LOLO} \\ + 6 \quad \text{+ LLO} \\ \hline 16 \quad \text{L0000} \end{array}$$

$$\begin{array}{r} \text{d) } 4 \quad \text{L00} \\ + 5 \quad \text{+ LOL} \\ + 6 \quad \text{+ LLO} \\ \hline 15 \quad \text{LLLL} \end{array}$$

$$\begin{array}{r} \text{e) } 11 \quad \text{LOLL} \\ + 6 \quad \text{+ LLO} \\ + 9 \quad \text{+ L00L} \\ + 2 \quad \text{+ LO} \\ \hline 28 \quad \text{LLL00} \end{array}$$

Da die *Subtraktion* die Umkehroperation der „ersten Stufe“ ist, rechnen wir bei der **Subtraktion im Dualsystem** wie mit natürlichen Zahlen nach der Methode des Ergänzens.

□ BEISPIEL 3/8:

$$\begin{array}{r} 13 \quad \text{LLOL} \\ - 6 \quad \text{- LLO} \\ \hline 7 \quad \text{LLL} \end{array}$$

Lösungsweg: Die halbfett hervorgehobenen Ziffern schreiben wir jeweils als Ergebnis auf.

$$\begin{aligned} 1. \text{ Spalte: } 0 + \mathbf{L} &= L \\ 2. \text{ Spalte: } L + \mathbf{L} &= L0 \\ 3. \text{ Spalte: } L + \mathbf{L^*} &= L0 \\ &L0 + \mathbf{L} = LL \end{aligned}$$

Übertrag nach links: L*

□ BEISPIEL 3/9:

$$\begin{array}{r} 10 \quad \text{LOLO} \\ - 5 \quad \text{- LOL} \\ \hline 5 \quad \text{LOL} \end{array}$$

Lösungsweg:

1. Spalte: $L + L = L0$

2. Spalte: $0 + L^* = L$

$L + 0 = L$

3. Spalte: $L + L = L0$

Übertrag nach links: L^*

Für die **Multiplikation im Dualsystem** sind folgende Rechenregeln erklärt.

$0 \cdot 0 = 0$

$0 \cdot L = 0$

$L \cdot 0 = 0$

$L \cdot L = L$

Zu achten ist nur auf die Stellenverschiebung.

Die so entstandenen Teilprodukte werden dann addiert.

□ BEISPIEL 3/10:

$$\begin{array}{r} \text{a) } 13 \cdot 5 \\ \hline 65 \\ \hline \end{array} \quad \begin{array}{r} \text{LL0L} \cdot \text{LOL} \\ \hline \text{LL0L} \\ 0000 \\ \text{LL0L} \\ \hline \text{L00000L} \end{array}$$

$$\begin{array}{r} \text{b) } 11 \cdot 13 \\ \hline 33 \\ 143 \\ \hline \end{array} \quad \begin{array}{r} \text{LOLL} \cdot \text{LL0L} \\ \hline \text{LOLL} \\ \text{LOLL} \\ 0000 \\ \text{LOLL} \\ \hline \text{L000LLLL} \end{array}$$

○ AUFGABEN

1. Stellen Sie die folgenden Zahlen jeweils im Dezimalsystem dar!

a) $(123)_8$ b) $(123)_5$ c) $LLL0L0$ d) $LLL0L$

2. Stellen Sie die folgenden im Dezimalsystem gegebenen Zahlen jeweils im angegebenen System dar!

a) 514 (System mit der Basis 8) b) 83 (System mit der Basis 5)
c) 130 (Duales System)

3. Stellen Sie die Zahl 400 in Systemen mit den Basen 8, 5, 4, 3 und 2 dar!

4. Führen Sie die Rechenoperationen im dualen System aus!

$$\begin{array}{lll} \text{a) } 6 + 7 = 13 & \text{b) } 7 + 3 = 10 & \text{c) } 15 \cdot 4 = 60 \\ 5 + 5 = 10 & 8 + 5 = 13 & 8 \cdot 7 = 56 \\ 13 - 6 = 7 & 13 - 8 = 5 & 11 \cdot 12 = 132 \\ 4 - 1 = 3 & 10 - 7 = 3 & 5 \cdot 3 = 15 \end{array}$$

5. Stellen Sie die Zahlen 63 und 514 jeweils in rein-dualer Verschlüsselung und in dezimal-dualer Verschlüsselung dar!

6. Lösen Sie die folgenden Aufgaben im Dualsystem!

Überprüfen Sie die Ergebnisse im Dezimalsystem!

a) $LL0LL + LOLOL$ b) $LOLL0L + LLL + LOLOL$
c) $LOLOLL \cdot LOOLO$ d) $LOLOLOLO \cdot LLLL$

3.3. Zu einigen Grundlagen aus der mathematischen Logik

Zum Verständnis der Arbeitsweise von elektronischen Rechnern sind einige Kenntnisse aus der mathematischen Logik erforderlich.

Wir wissen bereits, daß EDVA auf Grund ihres technischen Aufbaus und durch die Eingabe von Programmen in der Lage sind, logische Entscheidungen zu fällen.

Wir werden im folgenden einige Grundlagen aus der mathematischen Logik, zunächst ohne Bezug zur EDV, behandeln.

Aussagen

Der Begriff „Aussage“ ist bereits aus dem Mathematikunterricht bekannt [8, S. 9ff.]. **Aussagen** sind sinnvolle sprachliche Gebilde, die *entweder wahr oder falsch* sind.

Es gibt keine Aussage, die sowohl wahr als auch falsch ist. „Wahr“ (W) und „Falsch“ (F) heißen in Verbindung mit Aussagen **Wahrheitswerte**.

Welchen Wahrheitswert eine Aussage hat, hängt davon ab, ob diese Aussage die Realität richtig widerspiegelt oder nicht.

□ BEISPIEL 3/11:

a) Für alle reellen Zahlen a und b gilt:

$$a + b = b + a.$$

b) Leipzig ist die größte Stadt der DDR.

c)  ist das Symbol für einen elektrischen Widerstand.

d)  ist das Symbol für einen Kondensator.

Die Aussagen a) und c) haben den Wahrheitswert W (wahr), die Aussagen b) und d) den Wahrheitswert F (falsch).

Die Tatsache, daß jede Aussage *entweder* den Wahrheitswert W *oder* den Wahrheitswert F hat, wird „Satz von der Zweiwertigkeit der Aussagenlogik“ genannt.

Logische Operationen

Einzelassertionen lassen sich durch Bindewörter, zum Beispiel durch „und“ bzw. „oder“, zu Aussagenverbindungen verknüpfen. Ferner ist es möglich, Aussagen zu verneinen, beispielsweise: „Es ist nicht 5 gleich 8“.

Aussagenverbindungen und die **Verneinung (Negation)** sind Gegenstände der weiteren Betrachtungen. Es wird dabei vom konkreten Inhalt der Aussagen abstrahiert, und es werden deshalb Aussagenvariable verwendet, zum Beispiel p , q oder r .

Auf Grund des Satzes über die Zweiwertigkeit der Aussagenlogik kann man nunmehr sagen:

Für die Aussage p trifft entweder W oder F zu.

$$\begin{array}{c} p \\ W \\ F \end{array}$$

Negation

Wenn man sagen will, daß der durch eine Aussage gegebene Sachverhalt *nicht* existiert, verneint (negiert) man diese Aussage.

- BEISPIEL 3/12:
Aussage p : Es regnet. Aussage q : Es regnet nicht.
 q ist die Negation von p .
Man schreibt in diesem Fall $\sim p$ (gelesen: nicht p).

Es gilt: Wenn eine Aussage p wahr ist, so ist ihre Verneinung $\sim p$ falsch.
Ist aber die Aussage p falsch, dann ist die Verneinung $\sim p$ wahr.
Diesen Zusammenhang stellt man in einer **Wahrheitstabelle** der Negation folgendermaßen dar.

p	$\sim p$
W	F
F	W

- ▷ Wir stellen fest:
„Nicht p “ ist wahr genau dann, wenn p falsch ist.
„Nicht p “ ist falsch genau dann, wenn p wahr ist.

Konjunktion

Wir betrachten nun zwei Aussagen p und q , die durch „und“ miteinander verbunden sind. Man nennt eine solche Aussagenverbindung **Konjunktion** oder **UND-Operation**.

- BEISPIEL 3/13:
Gegeben sei die Konjunktion:
„Es regnet, und die Straße ist naß.“
Erste Aussage: Es regnet.
Zweite Aussage: Die Straße ist naß.
Den Wahrheitswert einer zusammengesetzten Aussage kann man aus den Wahrheitswerten der Einzelaussagen ermitteln.
Danach gibt es vier Möglichkeiten.
1. Es regnet (W), und die Straße ist naß (W).
Die Konjunktion hat den Wahrheitswert W .

2. Es regnet (W), und die Straße ist naß (F).
Die Ursache hierfür spielt keine Rolle (vielleicht ist eine Überdachung vorhanden).
Es wird hier nur angenommen, daß die Straße trotz des Regens trocken ist.
In diesem Fall hat die Konjunktion den Wahrheitswert F; denn es wird ja gesagt, daß es regnet und daß auch die Straße naß ist.
3. Es regnet (F), und die Straße ist naß (W).
(Es kann zum Beispiel ein Sprengwagen vorbeigefahren sein.)
In diesem Fall ist auch nur eine der beiden Aussagen wahr, und die Konjunktion hat den Wahrheitswert F.
4. Es regnet (F), und die Straße ist naß (F).
In diesem Fall ist natürlich auch die Verbindung durch „und“ nicht wahr, folglich hat die Konjunktion den Wahrheitswert F.

Für das „und“ schreiben wir das Zeichen \wedge .

Die Wahrheitswertetabelle sieht wie folgt aus.

p	q	$p \wedge q$
W	W	W
W	F	F
F	W	F
F	F	F

$p \wedge q$ gelesen: p und q



Wir stellen fest:

Die Konjunktion „ $p \wedge q$ “ ist eine Aussagenverbindung.

Sie ist dann und nur dann wahr, wenn sowohl p als auch q wahr ist.

Sie ist falsch genau dann, wenn mindestens eine der beiden Aussagen falsch ist.

Wir vergleichen die Multiplikationstabelle mit den Faktoren 1 und 0 mit der Wahrheitswertetabelle in Beispiel 3/13.

$1 \cdot 1 = 1$	W	W	W
$1 \cdot 0 = 0$	W	F	F
$0 \cdot 1 = 0$	F	W	F
$0 \cdot 0 = 0$	F	F	F

Der Zusammenhang mit der Konjunktion wird deutlich.

Die Konjunktion wird auch als **logische Multiplikation** bezeichnet.

Alternative

Wir betrachten nun eine weitere Aussagenverbindung, bei der die Aussagen p und q durch „oder“ verbunden sind.

Bemerkung: Das Bindewort „oder“ kann das ausschließende „entweder ... oder“ bzw. das nichtausschließende „oder“ sein.
 Im ersten Falle heißt die Aussagenverbindung **Disjunktion**, im zweiten Falle **Alternative**.

Für die späteren Betrachtungen von EDVA wird das nichtausschließende „oder“, also die **Alternative**, benötigt.

- ▷ Wir stellen fest:
 Die Alternative „ p oder q “ ist wahr genau dann, wenn mindestens eine der Aussagen p und q wahr ist.
 Sie ist falsch genau dann, wenn p falsch und q falsch ist.

- **BEISPIEL 3/14:**
 Die Aussagenverbindung
 „Heute ist der 31. April, oder Wasser siedet unter normalen Bedingungen bei $+100\text{ }^\circ\text{C}$ “
 ist wahr, weil die Aussage „Wasser siedet unter normalen Bedingungen bei $+100\text{ }^\circ\text{C}$ “ wahr ist.

Wir abstrahieren wie vereinbart vom konkreten Inhalt der einzelnen Aussagen und stellen eine Wahrheitstabelle auf. Für das „oder“ (nichtausschließendes „oder“) setzen wir das Zeichen \vee .

p	q	$p \vee q$
W	W	W
W	F	W
F	W	W
F	F	F

$p \vee q$ gelesen: p oder q

- ▷ Wir stellen noch einmal fest:
 Die Alternative oder ODER-Operation ist nur dann falsch, wenn beide Teilaussagen falsch sind, in den anderen Fällen ist sie wahr.

Wir vergleichen die Additionstabelle mit den Summanden 1 und 0 mit der Wahrheitstabelle in Beispiel 3/14.

$0 + 1 = 1$	W	W	W
$1 + 0 = 1$	W	F	W
$0 + 1 = 1$	F	W	W
$0 + 0 = 0$	F	F	F

Der Zusammenhang mit der Alternative wird deutlich.
 Die Alternative wird auch als **logische Addition** bezeichnet.

3.4. Zur technischen Realisierung einfacher logischer Strukturen

In EDVA müssen die im Teil 3.3. behandelten logischen Operationen durch technische Mittel nachgebildet und realisiert werden. Die Wissenschaftsdisziplin, die sich damit beschäftigt, ist die **Schaltalgebra**¹⁾, manchmal auch als „Algebra der Logik“ bezeichnet.

Als Mittel zur Darstellung von Verknüpfungen werden **Schaltelemente** benutzt, die mechanisch, elektromagnetisch oder elektronisch betätigt (gesteuert) werden (→ 46ff.).

Wir gehen von der Feststellung auf Seite 35 aus.

Danach ist eine Aussage entweder wahr oder falsch.²⁾

Dieser Sachverhalt läßt sich in einem elektrischen Stromkreis einfach realisieren:

Der Strom ist entweder eingeschaltet oder ausgeschaltet.

Auf ein Schaltelement bezogen heißt das: Entweder es fließt Strom durch das Schaltelement, oder es fließt *kein* Strom durch das Schaltelement.

Die Aussage ist wahr .	Die Aussage ist falsch .
<ul style="list-style-type: none"> — Der Wert „L“ wird zugeordnet. — Der Schalter ist geschlossen. — Das Relais ist gezogen. — Der Ferritkern ist magnetisiert. — Die Stelle auf dem Magnetband ist magnetisiert. 	<ul style="list-style-type: none"> — Der Wert „0“ wird zugeordnet. — Der Schalter ist offen. — Das Relais ist abgefallen. — Der Ferritkern ist nicht magnetisiert. — Die Stelle auf dem Magnetband ist nicht magnetisiert.
	
Bild 7	Bild 8

Die **UND-Schaltung** wird im Bild 9 veranschaulicht. Es werden hierzu zwei Schaltkontakte A und B benötigt. Das in Schaltskizzen benutzte Symbol für diese Schaltung zeigt Bild 10.

Für die Kontaktstellungen und ihre Verknüpfungen gilt:

A	B	$A \wedge B$
L	L	L
L	0	0
0	L	0
0	0	0

Wir schreiben: $A \cdot B = C$.

¹⁾ Die *Schaltalgebra* hat die Verknüpfung von Wahrheitswerten von Aussagen zum Gegenstand. In der „normalen Algebra“ sind die zu verknüpfenden Grundelemente Zahlen.

²⁾ Diese Aussagenverbindung mit „oder“ ist keine Alternative, sondern eine Disjunktion.

Strom fließt nur dann, wenn beide Kontakte geschlossen sind (\nearrow [7, S. 93]).

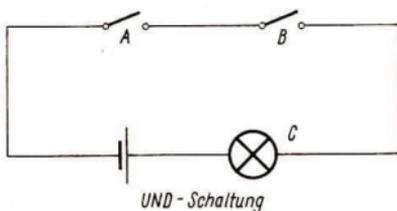
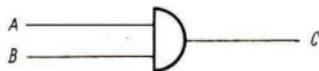


Bild 9



Signalflußbild der UND-Schaltung

Bild 10

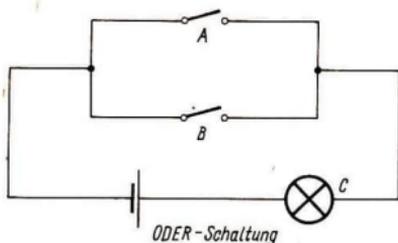
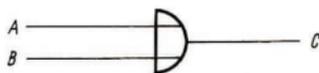


Bild 11



Signalflußbild der ODER-Schaltung

Bild 12

Die **ODER-Schaltung** wird im Bild 11 veranschaulicht. Während bei der UND-Schaltung die Schalter in Reihe liegen, sind sie hier parallel angeordnet. Das Sinnbild für Schaltskizzen zeigt Bild 12.

Für die Kontaktstellungen und ihre Verknüpfungen gilt:

A	B	$A \vee B$
L	L	L
L	0	L
0	L	L
0	0	0

Wir schreiben: $A + B = C$.

Das bedeutet: Strom fließt sowohl, wenn beide Kontakte geschlossen sind, als auch, wenn nur einer von beiden geschlossen ist. Sind beide Kontakte geöffnet, fließt kein Strom (\nearrow [7, S. 92f.]).

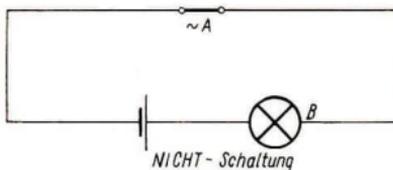


Bild 13



Signalflußbild der NICHT-Schaltung

Bild 14

Die **NICHT-Schaltung** mit dem zugehörigen Sinnbild zeigen die Bilder 13 und 14. In diesem Stromkreis fließt Strom, wenn der Schalter nicht betätigt wird.

Wir schreiben: $\sim A = B$ (gelesen: nicht A ist gleich B).



Wir halten fest:

Die UND-Schaltung ist eine Reihenschaltung.

Die ODER-Schaltung ist eine Parallelschaltung.



AUFGABEN

1. Wiederholen Sie die Gesetzmäßigkeiten im unverzweigten und im verzweigten Stromkreis! Benutzen Sie dazu [11, S. 80ff.]!
2. Geben Sie eine Übersicht über die Gesetzmäßigkeiten in Form einer Tabelle!

Verknüpfung logischer Operationen

Die logischen Operationen treten in ihrer praktischen Anwendung in **zusammengesetzten Schaltungen** oder **Schaltnetzen** auf. Ein einfaches Beispiel soll uns das verdeutlichen (A, B, C, ... verwenden wir als Variable für Aussagen).



BEISPIEL 3/15:

A wenn es regnet UND

B wenn ich auf der Straße bin UND

$\sim C$ meinen Regenschirm *nicht* aufspanne ODER

D wenn ich mich unter die Fontänen eines Springbrunnens stelle,

E werde ich naß.

Dafür schreibt man unter Verwendung der Zeichen für die logischen Operationen:

$$A \wedge B \wedge \sim C \vee D = E,$$

oder in der anderen Schreibweise:

$$A \cdot B \cdot \sim C + D = E.$$

Bild 15 veranschaulicht die Schaltung für diesen Fall.

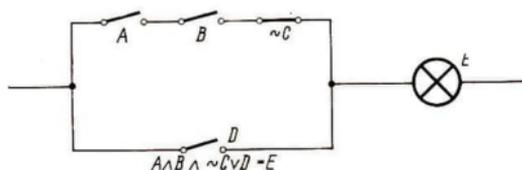
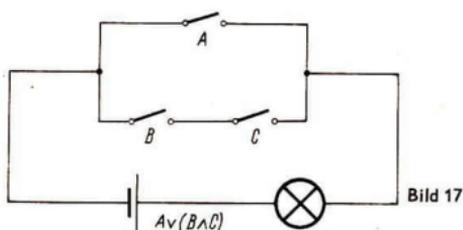
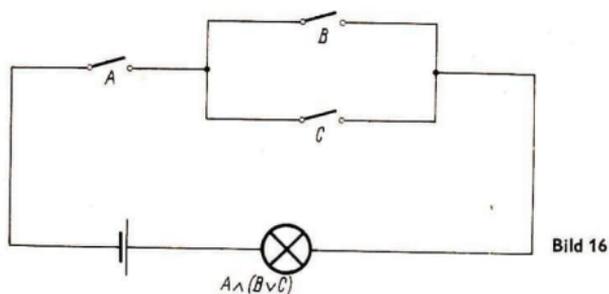


Bild 15

□

BEISPIEL 3/16:

In den Bildern 16 und 17 sind die Verknüpfungen $A \wedge (B \vee C)$ (in anderer Schreibweise $A \cdot (B + C)$) bzw. $A \vee (B \wedge C)$ (in anderer Schreibweise $A + (B \cdot C)$) als Schaltungen dargestellt.



Bemerkung: In den Beispielen 3/15 und 3/16 wurden vorgegebenen logischen Ausdrücken Schaltkreise zugeordnet. Umgekehrt kann man vorgegebenen Schaltungen logische Ausdrücke zuordnen. Diesen Fall erörtern wir hier nicht.

Wir wollen nun noch zu **Wahrheitwertetabellen** bzw. **Schaltbelegungstabellen** die entsprechenden **Schaltungen** skizzieren.

Für die Beispiele 3/17 a) bis d) wählen wir solche Wahrheitwertetabellen aus, die nur in einer Zeile den Wahrheitswert W haben. Das entspricht dem Gesamtzustand L in Schaltbelegungstabellen.

□

BEISPIEL 3/17:

a)

A	B	Gesuchte Schaltung
0	0	L
0	L	0
L	0	0
L	L	0

b)

A	B	Gesuchte Schaltung
0	0	0
0	L	L
L	0	0
L	L	0

c) A	B	Gesuchte Schaltung
0	0	0
0	L	0
L	0	L
L	L	0

d) A	B	Gesuchte Schaltung
0	0	0
0	L	0
L	0	0
L	L	L

Zum Lösen der Aufgaben bedenken wir:

Eine Reihenschaltung hat genau dann den Zustand L, wenn alle ihre einzelnen Schaltelemente den Zustand L haben. Deshalb führen in den vier Fällen folgende Überlegungen zum Ziel.

Fall a): Wenn A und B den Zustand 0 haben, dann haben

$\sim A$ und $\sim B$ den Zustand L.

$\sim A \wedge \sim B = L$

Aus analogen Überlegungen folgen:

Fall b): $\sim A \wedge B = L$,

Fall c): $A \wedge \sim B = L$,

Fall d): $A \wedge B = L$.

Die in den Bildern 18 bis 21 dargestellten Schaltungen widerspiegeln die in den Fällen a) bis d) jeweils anzustellenden Überlegungen.



Bild 18



Bild 19



Bild 20



Bild 21

Vereinigt man die in den Bildern 19 und 20 dargestellten Schaltungen zu einer Parallelschaltung, so erhält man Bild 22.

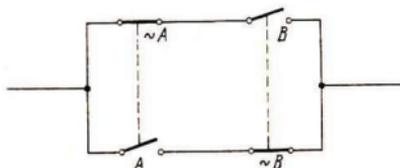


Bild 22

Diese Schaltung hat genau dann den Zustand L, wenn entweder $\sim A \wedge B$ oder $A \wedge \sim B$ gilt.

Die entsprechende Schaltbelegungstabelle sieht wie folgt aus.

A	B	Schaltungen
0	0	0
0	L	L
L	0	L
L	L	0

Wir sind nunmehr in der Lage, eine Schaltung zu entwickeln, mit deren Hilfe wir die Addition zweier Dualzahlen realisieren können.

Wir wählen einstellige Dualzahlen, also 0 und L.

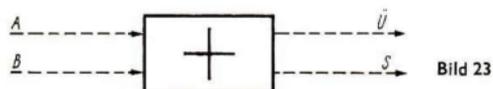
Für die Addition der Dualzahlen 0 und L gilt die folgende Tabelle.

$$\begin{aligned} 0 + 0 &= 00 \\ 0 + L &= 0L \\ L + 0 &= 0L \\ L + L &= L0 \end{aligned}$$

Als Schaltbelegungstabelle, in der A und B die Summanden, Ü einen Übertrag und S die Summe darstellen sollen, erhalten wir:

A	B	Ü	S	(*)
0	0	0	0	
0	L	0	L	
L	0	0	L	
L	L	L	0	

Zur Herstellung der Schaltung benötigen wir außer zwei Eingängen A und B auch zwei Ausgänge, einen für die Summe S und einen für den Übertrag Ü. Ein Blockschaltbild (Bild 23) soll die prinzipielle Lösung verdeutlichen.



Aus der Tabelle (*) ergeben sich zwei Schaltungen, eine Schaltung für den Übertrag Ü und eine Schaltung für die Summe S.

Wir betrachten die Schaltungen zunächst getrennt. Die Schaltungen werden aus den Schaltbelegungstabellen abgeleitet.

a) Schaltbelegungstabelle für den Übertrag:

A	B	Ü
0	0	0
0	L	0
L	0	0
L	L	L

Die dazugehörige Schaltung ist die UND-Schaltung (Bild 9).

b) Schaltbelegungstabelle für die *Summe*:

Für die Summe S ist die logische Verknüpfung nicht so ganz leicht zu erkennen. Weder die UND-Schaltung noch die ODER-Schaltung erfüllt die Schaltbelegungstabelle.

A	B	S
0	0	0
0	L	L
L	0	L
L	L	0

Jedoch die im Bild 22 dargestellte Schaltung entspricht der Schaltbelegungstabelle für die Summe. Nun werden, ausgehend von unserer Aufgabenstellung, beide Schaltungen miteinander verbunden (Bild 24).

A	B	\bar{U}	S
0	0	0	0
0	L	0	L
L	0	0	L
L	L	L	0

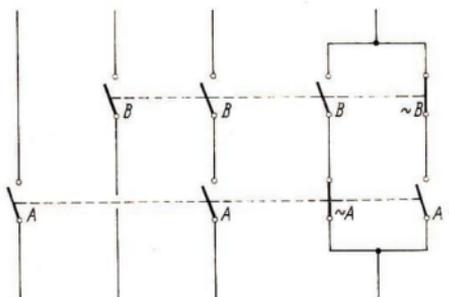


Bild 24

Mit dieser Schaltung läßt sich die Addition zweier einstelliger Dualzahlen verwirklichen.¹⁾

○ AUFGABEN

1. Informieren Sie sich in der angegebenen Literatur, insbesondere in [25], über die Addition einstelliger Dualzahlen!
2. Stellen Sie unter Verwendung von Elektronikbaukästen oder anderer Materialien Schaltungen her, die der unter 1. genannten Operation entsprechen!

¹⁾ Wer sich näher mit dieser Schaltung beschäftigen möchte, lese in [24] nach.

3.5. Zu einigen Schaltelementen von EDVA

Wir wollen uns einen Überblick über diejenigen Schaltelemente verschaffen, mit deren Hilfe man **logische Operationen** in **Digitalrechnern** (↗ 23) darstellen kann. Dabei ist das Wesentliche, daß die Übergänge von einem Zustand in den anderen schrittweise und sprunghaft erfolgen. Wir geben zunächst eine Übersicht.



Beispiele für Anlagen zur digitalen Datenverarbeitung

Anlage	Arbeitsweise	Schaltelemente
Rechenbrett	Mechanisch	Kugeln
Additionsmaschine	Mechanisch	Zahnräder/Zahnstangen
Tischrechenmaschine	Elektromechanisch	Relais
EDVA	Elektronisch	Elektronenröhre Transistor Halbleiterdiode Ferritkern

Wie wir wissen (↗ 18), waren in den ersten Rechenmaschinen *Relais* die hauptsächlichsten Schaltelemente. Sie wurden durch *Elektronenröhren* abgelöst.

Die Rechenanlagen der 3. Generation arbeiten im wesentlichen mit *Halbleiterdioden*, *Transistoren* und *Ferritkernen*.

Bemerkung: Bei den folgenden Erläuterungen der Eigenschaften und Funktionen von Schaltelementen in EDVA gehen wir nicht auf den physikalisch-technischen Aufbau dieser Schaltelemente ein, die aus dem Physikunterricht bekannt sind. (↗ [12, S. 121ff.]).

Zur **Röhrentriode** in EDVA

Zwischen der Verwendung einer Röhrentriode beispielsweise als Verstärker in einem Rundfunkapparat und der als Schaltelement in einer EDVA besteht ein wesentlicher Unterschied. Wir wissen, daß man mit Hilfe einer veränderlichen negativen Spannung am Gitter die Anodenstromstärke zwischen Null und einem Höchstwert kontinuierlich-stufenlos steuern kann.

In Rechenanlagen ist die Triode ein sogenanntes **binäres Element**, das heißt, man nutzt hier nur die beiden Extremfälle der Anodenstromstärke aus, also Null und den Höchstwert. Dadurch ist man in der Lage, die Binärzeichen 0 und 1 darzustellen. Dabei ist wichtig zu prüfen, ob die Röhrentriode diese beiden

Zustände annehmen und den Übergang von einem Zustand in den anderen in der erforderlichen äußerst kurzen Zeit vollziehen kann.

In der Praxis ist ferner zu beachten, daß mit zunehmendem Alter der Triode der Anodenstrom nachläßt. Das wirkt sich negativ auf die Fehlerfreiheit des Rechnens der EDVA aus. Ein weiterer Nachteil ist die sehr hohe Wärmeentwicklung, die beim Betrieb der Triode eintritt.

Zur Halbleiterdiode in EDVA

Die Wirkungsweise einer Diode kann man annähernd mit der eines mechanisch betätigten elektrischen Schalters oder Kontaktes vergleichen, mit dessen Hilfe man abwechselnd zwei unterschiedliche elektrische Spannungszustände herstellen kann.

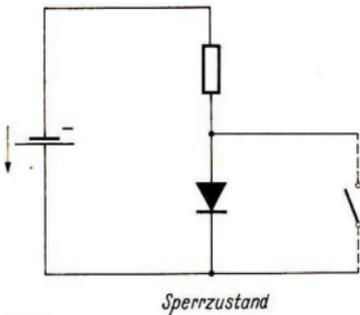


Bild 25

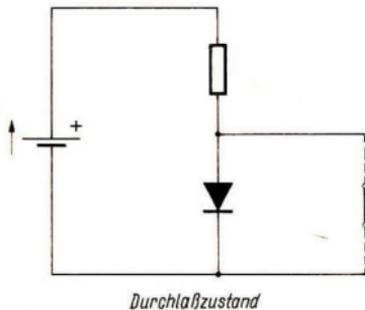


Bild 26

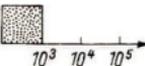
<i>Vergleich von Bauelementen</i>	
 <i>Relais</i>	 <i>Transistor</i>
 25 cm ³ <i>Volumen</i>	 5 cm ³ <i>Volumen</i>
 100 g <i>Masse</i>	 18 g <i>Masse</i>
 10 ³ 10 ⁴ 10 ⁵ <i>Lebensdauer</i>	 10 ³ 10 ⁴ 10 ⁵ <i>Lebensdauer</i>
 max 200/s <i>Schalzhäufigkeit</i>	 ≈ 100 000/s <i>Schalzhäufigkeit</i>

Bild 27

Eine Diode kann in **Sperrichtung** (es fließt kein elektrischer Strom) und in **Durchlaßrichtung** (es fließt ein elektrischer Strom) geschaltet werden. Der Sperrzustand entspricht etwa der Stellung eines geöffneten Schalters (Bild 25), der Durchlaßzustand etwa der Stellung eines geschlossenen Schalters (Bild 26). Verwendet werden dementsprechend die Binärzeichen 0 bzw. 1.

Zu **Transistoren** in EDVA

Transistoren werden ebenfalls als elektrisch gesteuerte Schalter eingesetzt. Dabei nutzt man den unterschiedlichen Grad der Leitfähigkeit solcher Transistoren. Bild 27 ermöglicht Vergleiche hinsichtlich der Vorteile, die das Verwenden von Transistoren an Stelle von Relais bietet.

Zu **Ferritkernen** in EDVA

Da die Ferritkerne in ihrer Anwendung als elektronische Schalter nicht Gegenstand unseres Physikunterrichts sind, wollen wir sie etwas ausführlicher behandeln.

Ein Ferritkern ist ein Ring aus einem Material, das die Eigenschaft hat, magnetische Zustände zu speichern. Sie wurden anfangs nur für das Speichern von Informationen verwendet. Man hat jedoch bald ihre Bedeutung als Schaltelement erkannt. Ferritkerne sind sehr klein, ihr Durchmesser beträgt etwa 1 bis 2 mm. Ihr Einsatz ermöglicht deshalb sehr hohe Umschaltgeschwindigkeiten. Heute werden Ferritkerne in vielen logischen Schaltungen von Datenverarbeitungsanlagen verwendet.

Wenn man einen Ferritkern in völlig unmagnetisiertem Zustand einer magnetischen Erregung H aussetzt, dann wächst seine Magnetisierung B bis zu einem gewissen Höchstwert. Die Magnetisierung wird durch einen Magnetisierungsstrom hervorgerufen.

Soll die magnetische Erregung abklingen, soll also H wieder kleiner werden, so verläuft die Entmagnetisierung nicht in der gleichen Weise wie die Magnetisierung, sondern es tritt eine Verzögerung auf. Selbst beim Fehlen einer

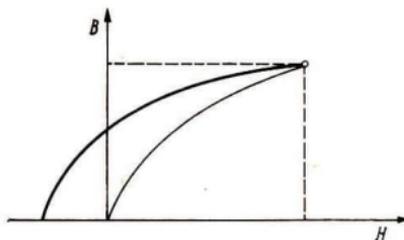
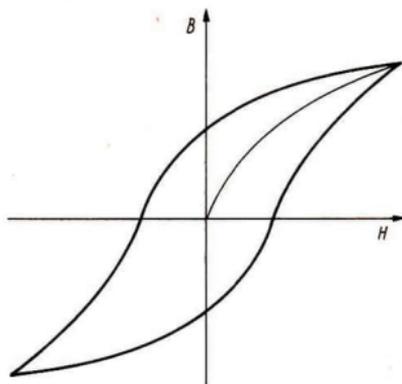


Bild 28

Bild 29



magnetischen Erregung, das heißt bei $H = 0$, bleibt das Material weiter im magnetisiertem Zustand. Bild 28 zeigt, daß die entsprechenden Graphen einander nicht decken.

Die Größe der zurückbleibenden Magnetisierung hängt vom Material ab. Wird nun das magnetische Feld H umgekehrt, dann erreicht die Magnetisierung ihren Wert 0 erst bei einer bestimmten Größe der entgegengesetzten Erregung.

Danach verläuft die graphische Darstellung der Magnetisierung der Erregung in einer Kurve, die man **Hysteresisschleife**¹⁾ nennt (Bild 29).

Wir betrachten noch einmal eine Eigenschaft der Ferritkerne: Sie speichern magnetische Zustände. Dabei ist wichtig, daß Ferritkerne bei der magnetischen Erregung $H = 0$ entweder den Zustand 0 oder den Zustand 1 haben können. In welchem der beiden Zustände sie sich befinden, hängt von der „Vorgeschichte“ der Magnetisierung ab.

Für EDVA werden die Ferritkerne mit einer Schreibwicklung und einer Lesewicklung versehen (Bild 30).

Durch die Schreibwicklung kann der Ferritkern in einen bestimmten Zustand überführt werden.

Die Lesewicklung liefert die Information über den vorgefundenen Zustand.

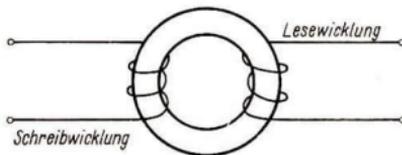


Bild 30

○ AUFGABEN

1. Bereiten Sie einen Kurzvortrag über Aufbau und Arbeitsweise von Relais vor!
2. Vergleichen Sie die Schaltzeiten von Relais und elektronischen Schaltelementen!
3. Bereiten Sie Kurzvorträge über Aufbau und Arbeitsweise von Röhrendioden und Röhrentrioden vor!
4. Bereiten Sie Kurzvorträge über Aufbau und Arbeitsweise von Halbleiterschaltelementen (Halbleiterdioden und Transistoren) vor!
5. Sprechen Sie über die Anwendung der unter 3. und 4. genannten Schaltelemente insbesondere in EDVA!
6. Vergleichen Sie die Magnettrommel mit dem Ferritkern hinsichtlich ihrer Eigenschaft als Speicherelement in EDVA (↗ 53f.)! Informieren Sie sich darüber in [28]!

¹⁾ hysteresis (griech.) bedeutet so viel wie „nachhinken, zurückbleiben“.

4

Zum Aufbau und zur Arbeitsweise von EDVA

4.1. Zum Gerätesystem einer EDVA

Prinzipieller Aufbau

Die Tätigkeit eines Menschen bei der Lösung einer Aufgabe kann durch fünf Merkmale charakterisiert werden (Bild 31).

1. Auslösen verschiedener Funktionen nach einer Verarbeitungsvorschrift — **Steuerfunktion**
2. Aufstellen der Eingabeinformationen — **Eingabefunktion**
3. Ausführen von Rechenoperationen — **Rechenfunktion**
4. Merken von Zwischenergebnissen — **Speicherfunktion**
5. Aufschreiben des Endergebnisses — **Ausgabefunktion**

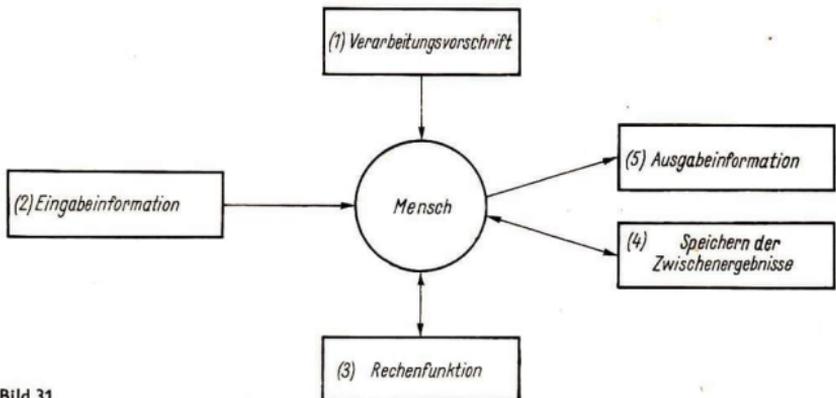


Bild 31



BEISPIEL 4/1:

Ein Schüler löst eine Gleichung.

1. Die *Steuerfunktion* übernimmt das menschliche Gehirn. Sämtliche folgenden Funktionen werden vom Gehirn ausgelöst.
2. Der *Eingabefunktion* entspricht das Stellen der konkreten Aufgabe.
3. Die *Rechenfunktion* wird durch die zum Lösen der Gleichung erforderlichen Rechenoperationen realisiert.
4. Die *Speicherfunktion* wird durch das Merken von Zwischenergebnissen realisiert.
5. Die *Ausgabefunktion* wird durch das Aufschreiben der Lösungsmenge realisiert.

Soll die Aufgabe mit Hilfe einer EDVA erledigt werden, so müssen die fünf angeführten Funktionen von der EDVA ausgeübt werden. Daraus lassen sich die Gesichtspunkte, die den **prinzipiellen Aufbau einer EDVA** charakterisieren, ableiten (Bild 32).

1. Die EDVA muß eine Einrichtung haben, in der nach einer Verarbeitungsvorschrift (*Programm*) in einer zeitlichen Reihenfolge der gesamte Arbeitsablauf gesteuert wird. Diese Einrichtung heißt **Steuerwerk**.
2. In die EDVA müssen die zu verarbeitenden Daten und das Programm *einggegeben* werden. Die EDVA muß also über **Geräte zur Eingabe** von Informationen verfügen.

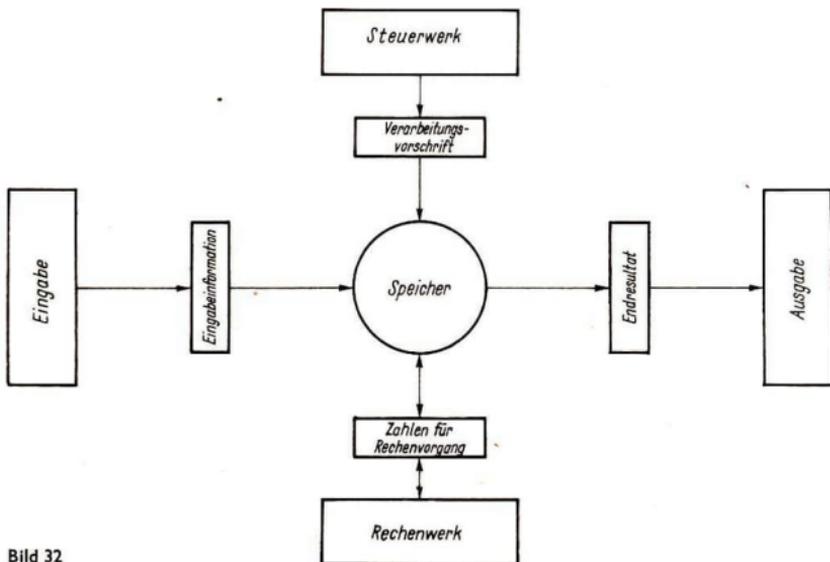


Bild 32

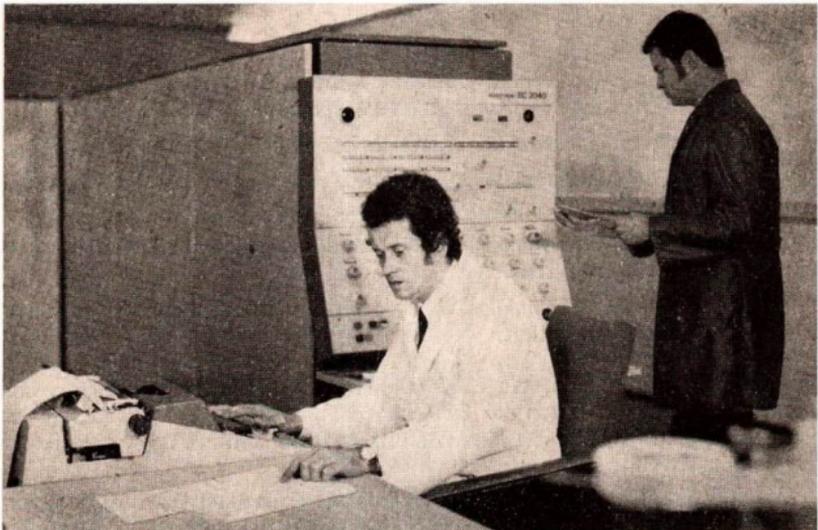


Bild 33

3. Die EDVA muß in der Lage sein, entsprechend den Anweisungen des Programmes *Rechenoperationen* auszuführen. Die Einrichtung hierfür bezeichnet man als **Rechenwerk**.
4. Die EDVA muß das Programm und die zur Verarbeitung benötigten Daten *speichern* können. Sie muß also über einen **Speicher** verfügen.
5. Die von der Maschine ermittelten Resultate müssen *aufgezeichnet* werden können. Die EDVA muß also **Geräte zur Ausgabe** haben.

Speicher, Rechenwerk und Steuerwerk bezeichnet man als **Zentraleinheit** einer EDVA, Geräte zur Ein- und Ausgabe zählt man zur **Peripherie**.

Die Zentraleinheit einer EDVA

Bild 33 zeigt die Zentraleinheit einer EDVA.

Im Speicher der Zentraleinheit werden die zu verarbeitenden Daten und das Programm gespeichert.

Ein Speicher ist in sogenannte Speicherzellen unterteilt, die zur Kennzeichnung der gespeicherten Informationen numeriert sind. Die Zahl, die bei dieser Nummerierung einer Zelle zugeordnet ist, heißt **Adresse**.

Das Aufrufen von Informationen aus dem Speicher heißt **Lesen**, das Eingeben von Informationen heißt **Schreiben**.

Daten müssen auf dem Speicher trotz mehrmaligen Lesens erhalten bleiben. Soll eine Information in eine Speicherzelle geschrieben werden, in der schon eine Information enthalten ist, so wird der alte Inhalt überschrieben (d. h. ge-

löscht). Die Zeit, in der eine Speicherzelle aufgerufen und deren Inhalt an die zur Ausführung der Operation zuständige Stelle transportiert wird, ist die **Zugriffszeit**. Es wird von den Konstrukteuren der EDVA angestrebt, diese Zugriffszeit so gering wie möglich zu halten. Besonders geringe Zugriffszeiten werden bei Magnettrommel- und bei Ferritkernspeichern erzielt. Zum besseren Verständnis und zum Vergleich mit der Wirkungsweise eines Speichers erörtern wir in Beispiel 4/2 die Benutzung eines Gepäckaufbewahrungsautomaten.

- **BEISPIEL 4/2:**
- Wir untersuchen die Vorgänge, die in einem Gepäckaufbewahrungsautomaten erfolgen.
1. Es sind mehrere Boxen vorhanden.
 2. Die Boxen sind durch die Beschriftung mit Ziffern eindeutig unterschieden.
 3. Zu jeder Box gehört genau ein Schlüssel zum Öffnen und Schließen des Faches.
 4. Man kann Gepäckstücke in die Box hineinstellen.
 5. Das Gepäck kann aufbewahrt werden.
 6. Bei Bedarf kann das Gepäck wieder herausgenommen werden.

Wir übertragen diese Funktionen nun sinngemäß auf den Speicher einer EDVA.

- ▷ Wir prägen uns ein:
1. Ein Speicher hat mehrere Speicherzellen.
 2. Die Speicherzellen sind nummeriert.
 3. Es existiert eine eindeutige Zuordnung Speicherzelle – Nummer (**Adresse**).
 4. Die Speicherzelle kann Informationen aufnehmen (**Schreiben**).
 5. Die Speicherzelle behält eine Information so lange, bis eine neue Information in die Zelle gebracht wird (**Speichern**).
 6. Die Information kann aus der Speicherzelle entnommen werden (**Lesen**).

Der **Magnettrommelspeicher** besteht aus einem Zylinder, auf dessen Oberfläche sich eine magnetisierbare Schicht befindet. Diese Schicht kann magnetische Impulse aufnehmen. Die Oberfläche der Magnettrommel ist in Spuren eingeteilt, in der die Informationen gespeichert werden können. Je nach Ausführung beträgt der Spurbstand 1 bis 2 mm, so daß 50 bis 250 Spuren auf einer Trommel vorhanden sind. Jede Spur besitzt mindestens einen Lese-

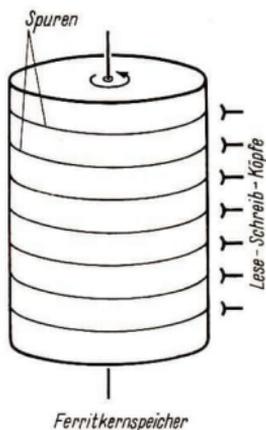


Bild 34

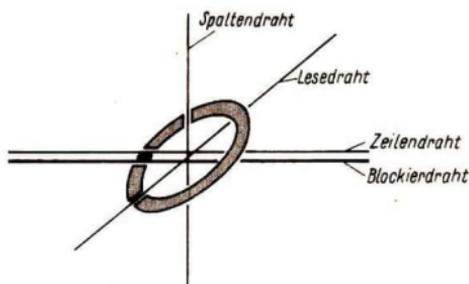


Bild 35

Schreibkopf, der die Informationen auf die Trommel aufzeichnet (Schreiben) oder von der Trommel holt (Lesen). Bei diesem Vorgang dreht sich die Trommel mit einer Geschwindigkeit von 3000 bis 18000 Umdrehungen je Minute (Bild 34).

Im **Ferritkernspeicher**¹⁾ sind kleine Ringe aus einem ferromagnetischen Material enthalten. Die Ringe, die einen Durchmesser von nur 1 bis 2 mm haben, haben zwei stabile Magnetisierungszustände und können mit Hilfe eines Impulses sehr schnell von einem Zustand in den anderen geschaltet werden. Nach dem Umschalten verbleiben sie nahezu in diesem Zustand. Man verwendet für Ferritkerne ein Material mit sehr großem Restmagnetismus (Remanenz). Die Magnetisierung im Ferritkern ist der Windungszahl und der Stromstärke proportional. Bei entsprechend ausgelegter Stromstärke benötigen die sehr kleinen Ferritkerne nur eine geringe Windungszahl. So werden die Leiter ganz einfach in Form eines Gitters durch die Ringe geschoben, so daß sich in jedem Kreuzungspunkt eines Zeilendrahtes mit einem Spaltendraht ein Ferritkern befindet (Bild 35). Damit der Kern umgeschaltet werden kann, muß sowohl durch den Zeilendraht als auch durch den Spaltendraht ein Strom fließen. Ein Leiter allein genügt nicht zur Änderung der Magnetisierung. Es ist also durch Zeilendraht und Spaltendraht eine genaue Zuordnung jedes einzelnen Ferritkerns innerhalb eines Speicherteils möglich.

Der Ferritkernspeicher hat gegenüber dem Magnettrommelspeicher einen großen Vorteil, es fällt die rotierende Bewegung der Trommel weg. Das bedeutet, daß die damit verbundene Wartezeit, bis die Information am Lesekopf vorbeikommt, entfällt. Die Umschaltzeit der Ferritkerne ist sehr klein. Daraus ergibt sich eine äußerst kurze Zugriffszeit des Ferritkernspeichers. Da keine

¹⁾ Zum Schaltelement Ferritkern vergleiche Seite 48f.

Bewegung mechanischer Teile erforderlich ist, nutzen sich Ferritkerne nicht ab. Demgegenüber muß beim Ferritkernspeicher als Nachteil in Kauf genommen werden, daß er eine geringere Speicherkapazität als der Magnetrommelspeicher hat. Ein Trommelspeicher ist auch bedeutend billiger als ein Ferritkernspeicher.

Das Steuerwerk hat die Aufgabe, die Befehle in der vom Programm vorgeschriebenen Reihenfolge zu lesen und die vorgeschriebene Operation auszuführen.

Grundsätzlich besteht ein Befehl aus einem Operationsteil und aus einem Adreßteil.

□ BEISPIEL 4/3:

Im Operationsteil eines Befehls sei angegeben:

„Schreibe den abgefühlten Inhalt einer Lochkarte in Speicherzellen ein!“

Im Adreßteil dieses Befehls steht die Hauptspeicheradresse als Kennzeichen für den Ausgangspunkt der Abspeicherung des Lochkarteninhaltes.

Das Steuerwerk führt folgende Operationen aus.

1. Der Lochkartenleser wird zu einer Eingabe veranlaßt.
2. Der abgefühlte Inhalt der Lochkarte wird in die angegebenen Hauptspeicherzellen eingeschrieben.

Vom Aufbau her unterscheidet man fünf Rechnertypen.

Hat der Adreßteil eine Adressenangabe, so handelt es sich um eine **Einadreßmaschine**. Bei zwei Adressenangaben spricht man von einer **Zweiadreßmaschine**. Dieses Prinzip läßt sich bis zur **Fünfadreßmaschine** fortsetzen.

Die folgenden Ausführungen beziehen sich nur auf die Einadreßmaschine, da diese am häufigsten Verwendung findet.

Das Rechenwerk hat die Aufgabe, die arithmetischen Operationen, logischen Verknüpfungen, Vergleichsoperationen usw. auszuführen.

Da sich letztlich alle Grundrechenoperationen auf die Addition zurückführen lassen, ist das **Addierwerk** das wichtigste Element eines Rechenwerkes (↗ 52). Damit bei der Abarbeitung einer Rechenoperation vom Rechenwerk laufend auf die jeweiligen Operanden zurückgegriffen werden kann, hat das Rechenwerk besonders schnelle Speicher mit kurzer Zugriffszeit (↗ 19), die sogenannten Register. Während der Arbeit des Rechenwerkes werden im ROBOTRON 300 ständig Operanden aus dem Hauptspeicher in die Register transportiert und dort zwischengespeichert. Befehle aus dem Steuerwerk bewirken diesen Transport. Weitere Befehle veranlassen dann die Verknüpfung der Operanden und den Transport zum Hauptspeicher.



BEISPIEL 4/4:

Zwei Zahlen a und b , die auf zwei Hauptspeicherzellen stehen, sollen addiert werden. Das Ergebnis ist in eine dritte Hauptspeicherzelle zu transportieren.

Schritt	Operation	Operand	Gerät
1	Transport der Zahl a in das Register	a	Steuerwerk
2	Addition der Zahl b	b	Rechenwerk
3	Transport des Ergebnisses aus dem Register in die Hauptspeicherzelle	Ergebnis	Steuerwerk

Die Peripherie einer EDVA

Peripheriegeräte sind alle Geräte, die die Eingabe und Ausgabe von Informationen realisieren.

Sie stellen die Verbindung zwischen dem Anwender der EDV und der Zentraleinheit der EDVA her, das heißt, sie erfüllen eine Vermittlerfunktion zwischen Mensch und Maschine. Mit Hilfe der peripheren Geräte können Daten und Programme ein- bzw. ausgegeben werden. Es können auch Daten und Programme auf externe Speicher, also auf zusätzliche Speicher, die nicht zur Zentraleinheit gehören, übermittelt werden.

Lochkartenleser sind Geräte zur Eingabe von Daten, die auf Lochkarten gespeichert sind.

Das Abfühlen der Lochungen auf der Lochkarte kann elektromechanisch oder fotoelektrisch erfolgen. Die Lochkarte gelangt über Transportrollen zur Abfühlstation, wird dort gelesen und anschließend in ein Ablagefach transportiert. Die Abfühlstation besteht bei elektromechanischer Arbeitsweise aus einem Satz federnder Metallbürsten. Für jede Spalte der Lochkarte ist genau eine Abfühlbürste vorhanden. Der Karton der Lochkarte stellt einen Isolator dar. Eine vorhandene Lochung in der Lochkarte schließt einen Stromkreis. Bei fotoelektrischer Abföhlung werden die Metallbürsten durch Fotowiderstände ersetzt. Eine vorhandene Lochung in der Lochkarte läßt einen Lichtstrahl auf den Fotowiderstand treffen. Dadurch wird ein Stromkreis geschlossen. Die Lesegeschwindigkeit liegt bei etwa 500 bis 1500 Karten in der Minute.

Lochkartenstanzer sind Geräte zur Ausgabe von Lochkarten. Das Stanzen der Lochkombinationen erfolgt durch Stanzstempel. Die Geschwindigkeit eines Lochkartenstanzers liegt bei 250 bis 300 Lochkarten in der Minute.

Lochstreifenleser sind Geräte zur Eingabe von Lochstreifen. Man unterscheidet Lochstreifenleser nach der Lesegeschwindigkeit, der Art der Abfüh-

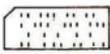
lung und nach der Streifenbreite. Zu den langsamen Lochstreifenlesern zählt man alle Geräte, deren Lesegeschwindigkeit unter 50 Zeichen je Sekunde liegt. Die Abführung erfolgt elektromechanisch. Bei schnelleren Lochstreifenlesern arbeitet die Abfühleinheit fotoelektrisch. Diese Leser erreichen eine Lesegeschwindigkeit bis zu 1500 Zeichen je Sekunde.

Streifenstanzer sind Geräte zur Ausgabe von Lochstreifen. Der Stanzvorgang erfolgt über Stanzstempel, das heißt mechanisch. Die Stanzgeschwindigkeit liegt bei maximal 300 Zeichen je Sekunde.

Elektrische Schreibmaschinen sind Geräte zur Ausgabe in Klartext. Ein solches Gerät ist bei jeder EDVA vorhanden. Mit ihm wird die **Bedienerverständigung** durchgeführt. Die Anlage übermittelt dem Bediener Nachrichten des Systems, oder der Bediener übermittelt dem System Nachrichten. Die elektrische Schreibmaschine ist in beschränktem Maße also auch für die Eingabe geeignet. Die Geschwindigkeit bei der Eingabe richtet sich nach der Fertigkeit des Bedieners. Die Geschwindigkeit bei der Ausgabe liegt bei 10 bis 20 Zeichen je Sekunde. Die Ausgabe größerer Datenmengen wird über **Schnelldrucker** realisiert.

Neben den genannten Ein- und Ausgabegeräten gehören auch **externe Speicher** zu den peripheren Geräten einer EDVA. Die externen Speicher untergliedert man in sequentielle Speicher und in Speicher mit direktem Zugriff. Bild 36 enthält eine Gegenüberstellung der beiden Speicherungsformen.

Bild 36

	Sequentielle Speicher			Speicher mit direktem Zugriff		
Merkmale	Kein direkter Zugriff, Speicherung der Daten wird von der Reihenfolge ihrer Eingabe bestimmt			Direkter Zugriff Wahlfreier Zugriff zu allen gespeicherten Daten		
	Eingabegeschwindigkeit durch Zuführmechanik begrenzt Begrenzte Kapazität		Hohe Eingabegeschwindigkeit durch hohe Zeichendichte und hohe Bandgeschwindigkeit	Hohe Zugriffsgeschwindigkeit Einsatz nur sinnvoll, wenn direkter Zugriff erforderlich		
Typ	 Lochband	 Lochkarte	 Magnetband	 Wechselplattenspeicher	 Plattenspeicher	 Magnettrommel
Kapazität	400 Zeichen/m	80 Zeichen / Karte	20 Mill. / Rolle	7,25 Mill. / Stapel	700 Mill. / Stapel	1...40 Mill. / Trommel
Zugriffzeit	ca. 1 ms je Zeichen	ca. 60 ms je Karte	70 min je Rolle	0,1 s	0,1...0,2 s	0,1 s

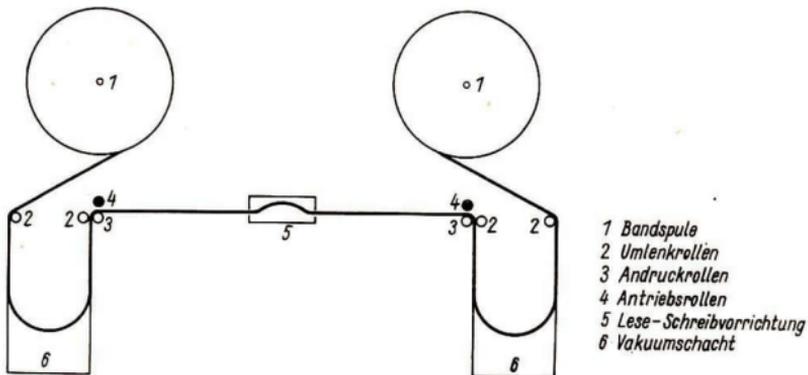


Bild 37

Magnetbandspeicher haben als Speichermedium ein dünnes mit einer Magnetschicht versehenes Kunststoffband.

Auf dieses Band werden mit Hilfe von Magnetköpfen Daten aufgezeichnet und gespeichert. Das Prinzip der Datenspeicherung auf Magnetband entspricht dem der Speicherung auf der Magnettrommel. Der prinzipielle Aufbau eines Magnetbandspeichers ist im Bild 37 dargestellt.

Die Informationen werden auf das Band in mehreren in Längsrichtung nebeneinanderliegenden Spuren aufgezeichnet. Bei der Aufzeichnung oder Wiedergabe von Informationen wird das Band an der Lese-Schreibeinrichtung vorbeigeleitet.

Magnetplattenspeicher haben als Informationsträger mit magnetisierbaren Schichten versehene Scheiben. Diese drehen sich mit sehr hoher Geschwindigkeit. Die Informationen werden in konzentrischen Kreisen (Zylindern) auf der Ober- und Unterseite der Platten (Spuren) untergebracht. Die beiden äußeren Plattenoberflächen sind mit Schutzplatten versehen, so daß für die Informationsaufzeichnung 10 Spuren zur Verfügung stehen. Für jede Plattenoberfläche ist ein Lese-Schreibkopf vorhanden. Die 10 Lese-Schreibköpfe sind zu einem Zugriffskamm zusammengefaßt.

4.2. Zum Gerätesystem des Kleinrechners Cellatron SER 2d

Der **Kleinrechenautomat CELLATRON SER 2d** eignet sich zur Ausführung sich häufig wiederholender technischer und ökonomischer Berechnungen. Die Ein- und Ausgabe von Informationen erfolgt über eine Zehnertastatur rein



Bild 38

dezimal mit Komma und Vorzeichen oder über einen Lochstreifenleser. Befehle und konstante Werte werden über einen Lochstreifenleser eingegeben. Die Abarbeitung umfangreicherer Programme ist durch die Benutzung des Lochstreifens als externen Befehlsspeicher möglich. Bild 38 zeigt ein Kleinrechner-system mit einem Lochstreifenleser. Das im Bild 39 dargestellte Blockschaltbild zeigt den Aufbau des Rechenautomaten in sehr vereinfachter Form.

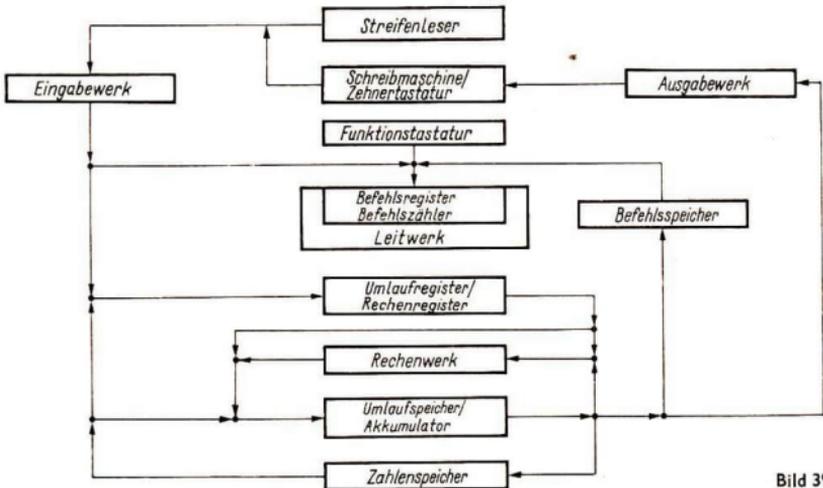


Bild 39

Alle über die Eingabegeräte ankommenden Informationen gelangen als Dezimalziffern zum Eingabewerk. Dort werden sie verschlüsselt (↗ 32). Befehle werden zum Befehlsregister und Rechengrößen werden zum Rechenregister transportiert. Das Rechenwerk führt alle vier Grundrechenoperationen aus. Alle Zahlen, die als Zwischen- oder Endresultate entstanden sind, können im Zahlenspeicher zwischengespeichert werden, bevor sie über das Ausgabewerk zur Schreibmaschine gelangen.

Beim CELLATRON SER 2d unterscheidet man zwei Formen der Programmierung. Bei der ersten Form ist das Programm auf der Magnettrommel gespeichert und wird von dort Befehl für Befehl abgearbeitet. Die Kapazität der Magnettrommel beträgt 189 Befehle. Bei der zweiten Form werden die Befehle vom Lochstreifen in der programmierten Reihenfolge direkt in das Steuerwerk transportiert und abgearbeitet.

Als Speicher hat die Maschine eine Magnettrommel, die in mehrere Spuren eingeteilt ist.

Der **Befehlsvorrat** des SER 2d umfaßt **arithmetische Befehle** (Addition, Subtraktion usw.), **Transportbefehle** (Eingabe in zwei Register, Ausgabe usw.), **Sprungbefehle** (unbedingter Sprung, Null-Sprung usw.) und **Schreibmaschinenbefehle** (Leerschritt, Tabulator usw.).

Die Leistungsfähigkeit einer CELLATRON SER 2d geht aus folgender Übersicht hervor.

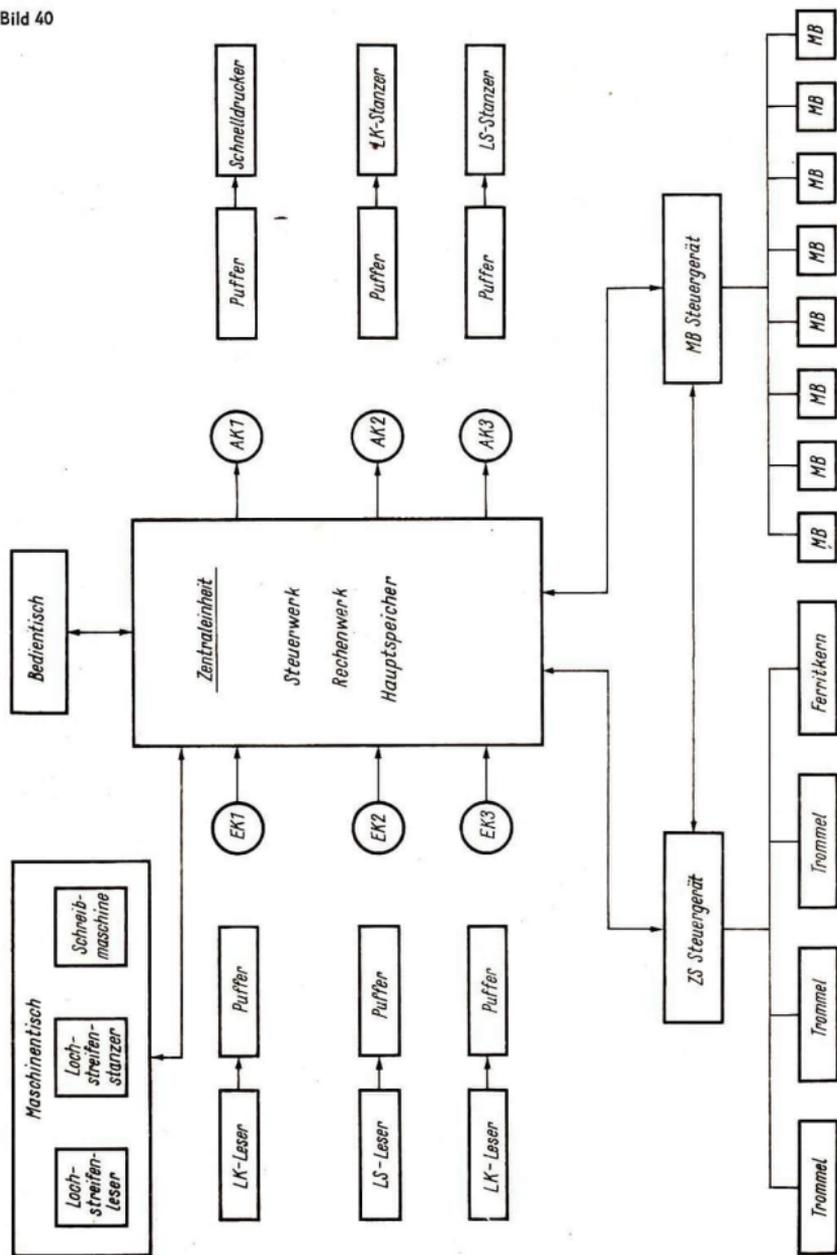
Elektrische Schreibmaschine	10 Anschläge/s
Elektromechanischer Streifenleser	20 Zeichen/s (5 Kanäle)
Festwertspeicher	127 Zahlworte
Befehlsspeicher	381 Einzelbefehle
Trommeldrehzahl	1480/min
Mittlere Zugriffszeit zum Trommelspeicher	11 ms
Mittlere Zugriffszeit zum	
Lochstreifen je Lochkombination	50 ms
Rechenzeit für Addition und Subtraktion	2,5 bis 50 ms
Rechenzeit für Multiplikation und Division	140 bis 180 ms
Rechenverhältnis zur Handrechenmaschine	1:30 bis 1:120
Ausführbare Befehle	11

4.3. Zum Gerätesystem der EDVA ROBOTRON 300

Allgemeines

Die EDVA ROBOTRON 300 (R 300) ist eine Datenverarbeitungsanlage mittlerer Größe.

Sie besteht aus der Zentraleinheit und einer Anzahl von peripheren Geräten. Die Zentraleinheit ist das Kernstück der Anlage, sie besteht aus dem Steuerwerk, dem Rechenwerk und dem Hauptspeicher. Zu den peripheren Geräten



zählt man Lochkartenleser, Lochstreifenleser, Lochkartenstanzer, Lochstreifenstanzer und Magnetbandspeichergeräte.

Die Zusammenarbeit der Zentraleinheit mit den peripheren Geräten wird über **Kanäle** realisiert, das heißt, Kanäle übernehmen den Datentransport vom bzw. zum peripheren Gerät. Die EDVA R 300 hat drei Kanäle für die Dateneingabe (z. B. Lochkarte, Lochstreifen) und drei Kanäle für die Datenausgabe (z. B. Lochkarte, Lochstreifen und Drucker). Die Ein- und Ausgabe von Magnetband wird über ein Magnetbandsteuergerät durchgeführt. Weiterhin besteht die Möglichkeit, einen Ferritkernspeicher und drei Magnettrommelspeicher anzuschließen. Der Aufbau der Anlage und die Zusammenarbeit der einzelnen Geräte wird im Bild 40 als Blockschaltbild dargestellt.

Dieser R 300 ist voll transistorisiert. Mit Transistoren werden hohe Schaltgeschwindigkeiten erreicht. Daraus resultiert auch die relativ hohe Dichte der Informationsübertragung auf dem Hauptspeicher von $10\mu\text{s}$ je Zeichen.

Zum Blockschaltbild der EDVA ROBOTRON 300 (Bild 40)

Bedientisch und Maschinentisch sind direkt an die Zentraleinheit angeschlossen. Über diese Geräte werden Datentransporte (kleine Datenmengen), Bedienfunktionen und Anzeigen der Anlage realisiert. Alle übrigen externen Geräte sind über Puffer oder Steuergeräte an die Zentraleinheit angeschlossen.

Puffer sind Ferritkernspeicher, die die Steuerung der langsam arbeitenden Ein- oder Ausgabegeräte übernehmen. Während der Datenübertragung vom Puffer zum Ein- oder Ausgabegerät und umgekehrt ist die Zentraleinheit für andere Aufgaben frei. Die Zentraleinheit ist nur für den Zeitraum der elektronischen Übertragung vom **Pufferspeicher** zum Hauptspeicher belegt. Die Verwendung von Pufferspeichern gleicht die hohe interne Verarbeitungsgeschwindigkeit mit der relativ langsamen Verarbeitungsgeschwindigkeit der externen Geräte aus. Die Magnetbandspeichergeräte und der Zusatzspeicher sind über Steuergeräte mit der Zentraleinheit verbunden. Die Steuergeräte übernehmen die Steuerung des Datentransports zwischen Hauptspeicher, Zusatzspeicher und Magnetbandspeichern.

Die Zentraleinheit

Die Zentraleinheit der EDVA R 300 besteht aus Hauptspeicher, Rechenwerk und Steuerwerk.

Der **Hauptspeicher** ist ein Ferritkernspeicher ($\approx 54\text{f.}$), auf dem 40000 Zeichen gespeichert werden können.

Er dient zur Aufnahme von Programmdateien und Problemdateien. Programmdateien umfassen die Anweisungen, die die einzelnen Teile der EDVA zur Arbeit anregen. Als Problemdateien bezeichnet man diejenigen Daten, die zur Verarbeitung auf einer EDVA bestimmt sind (\approx Teil 5).

Das **Rechenwerk** besteht aus Rechenwerk-Grundausrüstung und Rechenwerk-Ergänzung. Die Grundausrüstung gestattet die Ausführung von Additionen, Subtraktionen und logischen Operationen. Mit der Rechenwerk-Ergänzung ist es möglich, die Operationen Multiplikation und Division durchzuführen.

Die EDVA ROBOTRON 300 ist eine Einadreßmaschine. Sie ist deshalb mit einem Rechenregister ausgerüstet. Dieses Register ist der **Akkumulator** (Kurzzeichen AC). Der Akkumulator hat die Aufgabe, für eine Operation einen der Operanden aufzunehmen und das Resultat zu speichern. Der Akkumulator ist ebenfalls als Ferritkernspeicher angelegt und hat eine Kapazität von 120 Zeichen.

Das *Steuerwerk* hat die Aufgabe, die Befehlsabarbeitung zu steuern. Die Befehle sind im Hauptspeicher aufgesprochen.

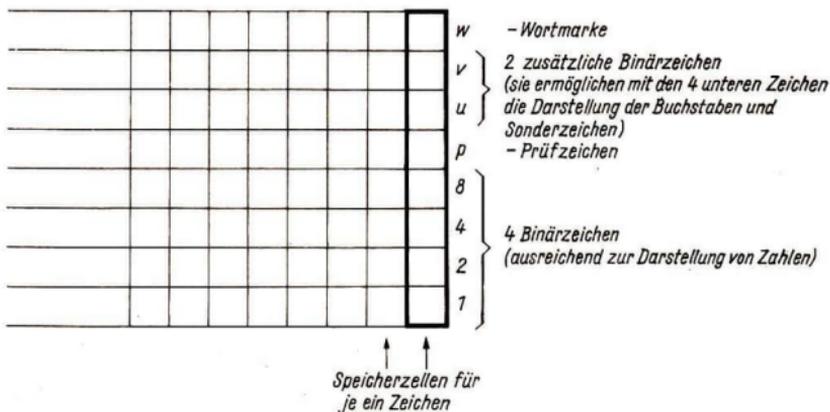
Datenstruktur und Adressierung auf dem Hauptspeicher des R 300

Während für die Darstellung der Zahlen in dezimal-dualer Verschlüsselung 4 Binärzeichen je Grundziffer genügt, müssen zur Darstellung der Buchstaben (A, B, C usw.) sowie Sonderzeichen (z. B. ;, , ;, *) zwei weitere Binärzeichen hinzugenommen werden. Weiterhin benötigt man ein Prüfzeichen, das zur automatischen Fehlererkennung dient. Die Funktion dieses Zeichens wird nachfolgend noch erläutert. Schließlich wird noch ein achttes Zeichen, die Wortmarke, zur Markierung des Wortendes benötigt. Und zwar bedeutet „L“ das Ende des Wortes, während andernfalls „0“ eingegeben wird. Ein Zeichen wird somit durch 8 Binärzeichen (L bzw. 0) angegeben. Das ist in Bild 41 dargestellt.

Jedes Zeichen auf dem Hauptspeicher erhält eine Adresse, die das Aufsuchen des gewünschten Zeichens sicherstellt. Das am weitesten rechts stehende Zeichen erhält die Adresse „00000“ zugewiesen, das am weitesten links stehende Zeichen die Adresse „39999“.

Im Beispiel 4/5 soll die Belegung des Hauptspeichers mit Daten verdeutlicht werden.

Bild 41





BEISPIEL 4/5:

Ab Adresse 00000 soll das Wort „ROBOTRON“ abgespeichert werden.

Die interne Darstellung gibt Bild 42 wieder.

	R	O	B	O	T	R	O	N	
	L	0	0	0	0	0	0	0	w - Wortmarke
	L	L	0	L	L	L	L	L	v
	0	0	L	0	L	0	0	0	u
	L	0	L	0	L	0	0	0	p - Prüfzeichen
	L	0	0	0	0	L	0	0	8
	0	L	0	L	0	0	L	L	4
	0	L	L	L	L	0	L	0	2
	L	0	0	0	L	L	0	L	1
	00008	00007	00006	00005	00004	00003	00002	00001	00000
									- Adresse

Bild 42

Aus einer Tabelle, die auf Seite 71 zu finden ist, entnimmt man die Verschlüsselung für die einzelnen Buchstaben. Danach hat das „R“ den Schlüssel (die Zeichen sind hier hintereinander aufgeführt):

v	u	8	4	2	1
L	0	L	0	0	L

Für das „0“ entnehmen wir der Liste den Schlüssel:

v	u	8	4	2	1
L	0	0	L	L	0

Die Wortmarke w ist in den Zellen mit den Adressen 00000 bis 00006 das Zeichen 0. Lediglich in der Zelle 00007 finden wir ein L, weil hier die Wortbegrenzung angezeigt werden muß. Für das Prüfzeichen kommt natürlich auch eines der Binärzeichen 0 oder L in Frage. Und zwar setzt man bei einem Zeichen ein L, wenn die Anzahl der in diesem Zeichen in den Ebenen w, v, u, 8, 4, 2, 1 auftretenden L gerade ist. Andernfalls setzt man ein 0. Damit ist gesichert, daß die Anzahl der L in allen 8 Ebenen stets ungerade ist. So ist beispielsweise beim „N“ in der Zelle 00000 bereits ein L in den Ebenen 1, 4 und v vorhanden.

Mit 3 Zeichen L ist eine ungerade Anzahl gesichert, so daß als Prüfzeichen 0 gewählt werden muß. Beim „R“ in der Zelle 00002 sind ebenfalls 3 Zeichen L vorhanden, und wieder folgt als Prüfzeichen 0. Dagegen sind beim „R“ in der Zelle 00007 durch das L in der

Ebene *w* insgesamt 4 Zeichen *L* vorhanden, also eine gerade Anzahl. Deshalb wurde hier als Prüfzeichen *L* gewählt.

Auch die Befehle werden im Hauptspeicher gespeichert. Ein Befehl belegt im Hauptspeicher 6 Zeichen (↗ Teil 6.1.).

Einige periphere Geräte des R 300

Auf dem Maschinentisch befinden sich eine Schreibmaschine, ein Lochstreifenleser und ein Lochstreifenstanzer. Diese Geräte sind in erster Linie zur Bedienung und zu Kontrollzwecken vorhanden. Wegen der geringen Arbeitsgeschwindigkeit dieser Geräte ist eine Eingabe bzw. Ausgabe nur geringer Datenmengen sinnvoll.

Die **Lochkarten-Lese-Stanz-Einheit** (Kurzzeichen LSE) dient zur Eingabe bzw. Ausgabe von Lochkarten.

Sie hat zwei gegenläufige Kartenbahnen (Lesebahn, Stanzbahn). Beide Kartenbahnen führen zu einer Ablageeinheit mit 5 Ablagefächern. Auf beiden Bahnen befinden sich 5 Stationen. Auf der Lesebahn sind es zwei Abfühlstationen und drei Leerstationen und auf der Stanzbahn zwei Abfühlstationen, eine Leerstation, eine Stanzstation und eine weitere Abfühlstation. Jede Information, die gelesen bzw. gestanzt wird, wird durch die nachfolgende Abfühlstation kontrollgelesen. Lese- oder Stanzfehler werden erkannt und können vom Programm bearbeitet werden. Entsprechend ihrer Konstruktion ist die LSE mit der Zentraleinheit über drei Puffer verbunden. Es sind ein Eingabepuffer an der Lesebahn, ein Eingabepuffer an der Stanzbahn und ein Ausgabepuffer an der Stanzbahn angeschlossen. Die LSE ist für die Arbeit mit 80-spaltigen Lochkarten ausgelegt und erreicht eine Geschwindigkeit von 24000 Karten je Stunde.

Der **Schnelldrucker** der EDVA R 300 ist über den Puffer an die Zentraleinheit anschließbar. Es können je Druckposition 57 verschiedene Zeichen ausgegeben werden. Das sind die Ziffern von 0 bis 9, die 26 Großbuchstaben und 21 Sonderzeichen (+, -, usw.). Die Druckbreite beträgt 156 Zeichen. Es ist eine Unterteilung der Druckbreite in zwei Bahnen möglich, wobei die Breite der Bahnen nach Bedarf festgelegt werden kann. Dadurch ist es möglich, zwei Drucklisten unabhängig voneinander zu drucken. Die Leistung des Schnelldruckers liegt bei 21000 Zeilen je Stunde.

Die Eingabe oder Ausgabe von **Lochstreifen** bei der EDVA R 300 erfolgt ebenfalls über Puffer. Der Lochstreifenpuffer kann sowohl als Eingabe- als auch als Ausgabepuffer arbeiten. An den Puffer können maximal zwei Lochstreifenleser und ein Lochstreifenstanzer angeschlossen werden, wobei die Arbeitsgeschwindigkeit der peripheren Geräte unterschiedlich sein kann. An die Zentraleinheit können bis zu 8 Magnetbandspeichergeräte angeschlossen werden, die über das Steuergerät mit der Zentraleinheit zusammenarbeiten.

5

Daten, Datenerfassung und Datenträger, Schlüsselsysteme

5.1. Daten

In allen Bereichen des gesellschaftlichen Lebens muß der Mensch Informationen verarbeiten (↗ Teil 1.).

Auf entsprechenden Belegen empfängt oder sendet er Informationen, und zwar in visuell lesbarer und allgemein verständlicher Form, oft in Gestalt von sprachlichen Texten.

Soll für die Informationsverarbeitung eine EDVA eingesetzt werden, so müssen die zu verarbeitenden Informationen aufbereitet werden. Man spricht dann von **Daten**.

▷ Wir prägen uns ein:

Daten sind Signale in Form von Zeichen oder Zeichenreihen zur Darstellung von Informationen.

Sie sind bereits auf Speichermedien als Zwischenträger dargestellt und sollen mit Hilfe von Datenverarbeitungsanlagen verarbeitet werden.

5.2. Datenerfassung und Datenträger

Die in den Rechner eingehenden Informationen müssen eine der Maschine entsprechende Form haben. Es können **Lochkarten, Lochstreifen** oder **Magnetbänder** sein. Man bezeichnet sie als **maschinenlesbare Datenträger**.

Die Daten werden auf der Lochkarte und dem Lochstreifen durch ausgestanzte Löcher gespeichert. Auf dem Magnetband wird die Datenspeicherung durch Magnetisierung realisiert. Dieser Vorgang wird als **Datenerfassung** bezeichnet.

Die Datenerfassung erfolgt maschinell mit Hilfe von **Datenerfassungsgeräten**.

Im folgenden soll die *Datenerfassung mit Hilfe von Lochkarten* erläutert werden. Der Mitarbeiter am Locher liest die Informationen von Belegen ab und gibt sie über ein Tastenfeld ein.

Durch Drücken der Taste wird ein Stanzstempel ausgelöst, und an der entsprechenden Stelle in der Lochkarte wird ein Loch eingestanz. Danach rückt die Lochkarte um eine Spalte weiter, und die nächste Taste kann betätigt werden. So wird die Lochkarte spaltenweise von Spalte 1 bis Spalte 80 gelocht. Die Fehlerhäufigkeit ist beim Lochen naturgemäß sehr hoch. Das würde sich auf den nachfolgenden Prozeß der Aufbereitung und Auswertung negativ auswirken. Deshalb wendet man das Prinzip der **Doppelerfassung** von Daten an. Die gelochten Karten werden in einem zweiten Arbeitsgang von einem anderen Gerät, dem **Prüfer**, auf Richtigkeit geprüft. So wie beim Lochen werden die Daten über ein Tastenfeld eingegeben. Jedoch wird diesmal kein Stanzstempel ausgelöst, sondern es wird festgestellt, ob auch tatsächlich an der richtigen Stelle eine Lochung ist.

Nach dem Lochen und Prüfen sind die Daten ordnungsgemäß auf Lochkarten erfaßt und können dem weiteren Datenverarbeitungsprozeß zugeführt werden. Bei der Datenerfassung benutzt man *numerische* (durch Ziffern dargestellte) oder *alphanumerische* (durch Buchstaben-Ziffern-Kombinationen dargestellte) Abkürzungen und bezeichnet diese als **Schlüssel** (↗ 72). Einige solcher Schlüssel kennen wir aus dem täglichen Leben, zum Beispiel Telefonnummern, Kontonummern der Sparkasse, Abnehmernummern bei Abrechnungen über den Energieverbrauch (↗ 72), polizeiliche Kennzeichen von Kraftfahrzeugen.

▷ Wir prägen uns ein:

1. Das Übertragen von Daten aus der visuell lesbaren Form in die maschinenlesbare bezeichnet man als **Datenerfassung**.
2. Das Ersetzen von Texten durch ein festgelegtes System von Kurzbezeichnungen bezeichnet man als **Verschlüsseln** (↗ „Code“, Seite 69).
3. Die wichtigsten maschinenlesbaren Datenträger sind **Lochkarte**, **Lochstreifen** und **Magnetband**.

Soll der Einsatz einer EDVA rationell erfolgen, muß der Prozeß der Datenerfassung besonders gut durchdacht und gestaltet sein. Man kann sagen: Je höher die Arbeitsproduktivität in der Datenerfassung ist, desto höher ist der ökonomische Nutzen der gesamten EDVA. Interessant sind in diesem Zusammenhang folgende Angaben: „Gegenwärtig sind etwa 40 bis 50 Prozent der Mitarbeiter eines Rechenzentrums mit dem Erfassen der Daten beschäftigt. Etwa 30 bis 55 Prozent der Gesamtkosten der Datenverarbeitung werden durch die Datenerfassung verursacht.“ [31, S. 20].

Zur Lochkarte

Lochkarten bestehen aus einem elastischen Spezialkarton und sind einheitlich 187,3 mm lang und 82,5 mm breit.

Eine Lochkarte ist in 12 **Lochzeilen** eingeteilt (Bild 43). Die Lochzeilen 12 und 11 bezeichnet man als **Überlochzone**, die Lochzeilen 0 bis 9 als **Normallochzone**. Die Kapazität der Lochkarte ist mit 80 **Lochspalten** begrenzt. Mehrere nebeneinanderliegende Lochspalten werden zu einem **Lochfeld** zusammengefaßt (Bild 44).

Für die Darstellung von Ziffern wird die Normallochzone benutzt. Eine Lochung in der Lochzeile 4 beispielsweise entspricht der Ziffer 4.

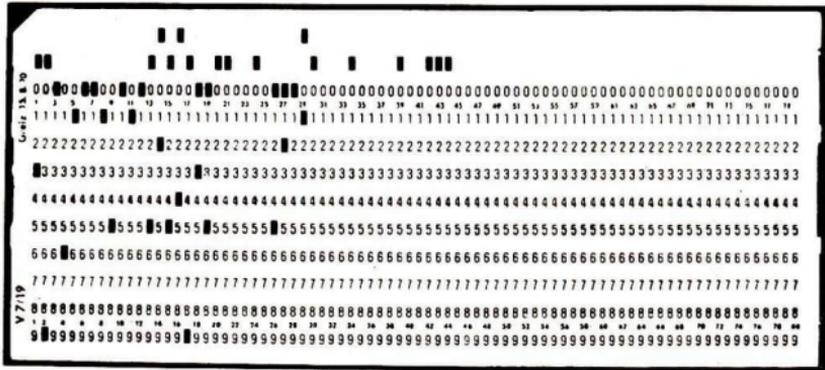
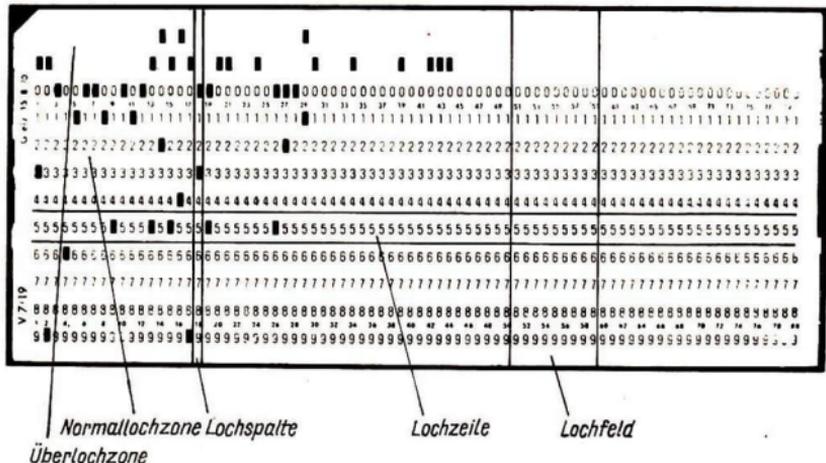


Bild 43

Bild 44



Für die Darstellung von Alphazeichen werden mehrere Lochungen in ein und derselben Spalte benötigt. Man bezeichnet diese Art der Lochung als **Lochkombination**.

- ▷ Wir prägen uns ein:
Die Vorschrift zur eindeutigen Zuordnung von Lochkombination und Zeichen bezeichnet man als **Code**.¹⁾

Lochkarten werden nach der Art ihres *Aufdrucks* und nach ihrer *Funktion* klassifiziert.

Ziffernkarte heißt eine Lochkarte, die als Aufdruck in allen Positionen die entsprechende Ziffer hat.

Normallochkarten enthalten als Aufdruck neben dem Ziffernspiegel die Einteilung der Lochkarte in Lochfelder und deren Bezeichnung.

Verbundkarten enthalten zusätzlich noch Platz für handschriftliche Eintragungen. Die Verbundkarte ist eine Kombination von Beleg und Lochkarte.

Bei der Klassifizierung der Lochkarten nach der Funktion unterscheidet man:

Datenkarten enthalten die zu verarbeitenden Daten.

Programmkarten enthalten die Befehle für die EDVA.

Zum Lochstreifen

Der Lochstreifen ist ein Papierband von 17,5 mm bis 24,5 mm Breite. Er ist längs in **Spuren** oder **Kanäle** und quer in **Positionen** eingeteilt. Man unterscheidet je nach der Anzahl der Kanäle 5-, 6-, 7- und 8-Kanal-Lochstreifen.

Wir befassen uns hier mit dem 8-Kanal-System.

Die Darstellung eines Zeichens auf dem Lochstreifen erfolgt durch Lochung einer entsprechenden **Lochkombination** in der jeweiligen Position. Bei Verwendung von 8-Kanal-Lochstreifen lassen sich 256 Lochkombinationen darstellen. Zur Datensicherung auf dem Lochstreifen wird ein besonderer Kanal für das Prüfbit benutzt. Es wird nach den gleichen Regeln wie im Hauptspeicher der EDVA gebildet (↗ 62).

Zum Magnetband

Das Magnetband ist ein dünnes (0,049 mm) Kunststoffband, das auf einer Seite mit einer magnetisierbaren Schicht versehen ist. Es ist in Längsrichtung in mehrere Spuren eingeteilt. Das Aufzeichnen und Wiedergeben von Informationen erfolgt durch Vorbeiführen des Bandes an Lese-Schreib-Köpfen, wobei das Band bei Aufzeichnungen lokal magnetisiert wird. Bei dem Magnetbandspeichergerät der EDVA R 300 liegt die Bandgeschwindigkeit bei 1,52 m/s. Die Zeichendichte auf dem Magnetband beträgt 22 Zeichen/mm.

¹⁾ Man beachte den Unterschied zu den Begriffen „Schlüssel“ bzw. „Verschlüsseln“ (↗ 72 bzw. 67).

5.3. Codierung und Schlüsselssysteme

Für die auf Seite 69 formulierte Definition könnte man auch sagen:

- ▷ Die Vorschrift zur eindeutigen Zuordnung zweier Zeichensmengen bezeichnet man als **Codierung**.

Sämtliche in der EDVA vorkommenden Informationen seien in der Menge A zusammengefaßt. Die Darstellung der Informationen als Zeichenkombinationen auf dem Datenträger oder auf dem Hauptspeicher der EDVA sei in der Menge B zusammengefaßt.

Die Zuordnungsvorschrift, die Elementen von A genau ein Element von B zuordnet, heißt **Code**.

$$A = \{0, 1, 2, \dots, x, y, z, \dots, ?, !, \dots\}$$

$$B = \{00000000, \dots, LLLLLLLL\}$$

Die Zusammenstellung auf Seite 71 ist eine Code-Übersicht für die EDVA ROBOTRON 300.

- **BEISPIEL 5/1:**
Wir stellen im Bild 45 die Zeichenreihe „EDVA R 300“ im R 300 Interncode dar.

L	0	0	0	0	0	0	0
0	0	L	0	L	0	0	0
L	L	L	L	0	0	0	0
L	L	L	L	0	L	L	L
0	0	0	0	L	0	0	0
L	L	L	0	0	0	0	0
0	0	0	0	0	L	0	0
L	0	L	L	L	L	0	0
E	D	V	A	R	3	0	0

Bild 45

Zeichen	R 300 Interncode	R 300 LK-Code	R 300							LS-Code							
			8	7	6	5	4	3	2	1	8	7	6	5	4	3	2
0	WVUP8421	0															
1	000L0000	1															
2	000000L0	2															
3	0000000L	3															
4	000L00LL	4															
5	00000L00	5															
6	000L0L0L	6															
7	000L0LL0	7															
8	00000LLL	8															
9	0000L000	9															
A	000LL00L	12-1															
B	00LL000L	12-2															
C	00LL00L0	12-3															
D	00L000LL	12-4															
E	00LL0L00	12-5															
F	00L00L0L	12-6															
G	00L00LL0	12-7															
H	00L0L0L0	12-8															
I	00L0L0LL	12-9															
J	00L0L00L	11-1															
K	0L0L00L0	11-2															
L	0L0000LL	11-3															
M	0L0L0L00	11-4															
N	0L000L0L	11-5															
O	0L000LL0	11-6															
P	0L0L0LLL	11-7															
Q	0L0LL000	11-8															
R	0L00L00L	11-9															
S	0LL000L0	0-2															
T	0LLL000L	0-3															
U	0LL00L00	0-4															
V	0LLL0L0L	0-5															
W	0LLL0LL0	0-6															
X	0LL00LLL	0-7															
Y	0LL0L000	0-8															
Z	0LLL000L	0-9															
.	00L0L0LL	12-8-3															
<	0L0L0LLL0	11-8-6															
(000L0000	8-4															
+	00L00000	12															
!	00L0000L	12-8-5															
*	0L00L000	11-8-4															
)	0L0L0L0L	11-8-3															
;	00L0L000	12-8-4															
-	0L000000	11															
>	0LL0L0LL	0-8-6															
?	0L00L0LL	11-8-7															
=	0L0LLL0L	11-8-5															

**BEISPIEL 5/2:**

Die der Zeichenreihe „ROBOTRON“ entsprechenden Lochkombinationen im R 300 Lochstreifencode sind im Bild 46 dargestellt.

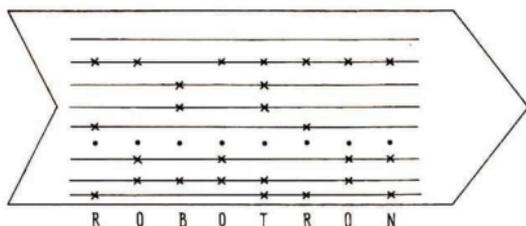


Bild 46

Viele Probleme, die heute von Datenverarbeitungsanlagen gelöst werden, sind ohne **Schlüsselsystematik** undenkbar. Es würde beispielsweise einen zu großen Speicherplatz und zu lange interne Sortierzeiten kosten, würden die Ordnungsbegriffe der Kunden eines Energiekombinates aus Name und Anschrift des Energieabnehmers bestehen. Rechnet man für den Namen maximal 20 Zeichen und für die Adresse maximal 25 Zeichen, so müßte bei der Auswahl eines Kunden nach einem 45 Zeichen langen Begriff abgefragt werden.

Der Abrechnungskassierer des Energiekombinates führt den Nachweis für die Energieabrechnung in seinem Abrechnungsbuch. Dieses Buch trägt eine dreistellige Nummer. Jeder Haushalt steht auf einem besonderen Blatt in diesem Buch. In Verbindung mit der laufenden Nummer des Abrechnungskreises wird die Abnehmernummer des Energieabnehmers gebildet. Diese Abnehmernummer ist ein **Schlüssel**.

**BEISPIEL 5/3:**

29-164014	bedeutet	Kreis	29
		Buch	164
		Blatt	14

Man unterscheidet grundsätzlich zwei Arten von Schlüsseln:

Numerische Schlüssel und **alphanumerische Schlüssel** (↗ 67).

Numerische Schlüssel teilt man wiederum in **dekadische** und **nichtdekadische Schlüssel** ein.

Bei dekadischen Schlüsseln wird jedem Begriff eine Ziffer mit gleicher Stellenzahl zugeordnet. Man kann dadurch in zweckmäßiger Weise Gruppen zusammengehöriger Begriffe bilden. Die einfachste Form eines nichtdekadischen Schlüssels ist die fortlaufende Numerierung. Man kommt mit einer relativ kleinen Anzahl von Stellen aus, muß aber den Nachteil der Unübersichtlichkeit in Kauf nehmen.

Alphanumerische Schlüssel werden aus Kombinationen von Buchstaben und Zahlen gebildet. Da die 26 Buchstaben des Alphabets a, ..., z mehr Möglich-

keiten zulassen als die 10 Grundziffern 0, ..., 9, ist der alphanumerische Schlüssel flexibler einsetzbar. Typische Beispiele sind die polizeilichen Kennzeichen von Kraftfahrzeugen.

Jeder Schlüssel hat eine Reihe von Vor- und Nachteilen. Welcher Schlüssel eingesetzt wird, ist letztlich immer vom jeweiligen Problem abhängig.

○ **AUFGABEN**

1. Nennen Sie die wichtigsten maschinenlesbaren Datenträger!
2. Erläutern Sie die Begriffe „Lochzeile“, „Lochspalte“ und „Lochfeld“!
3. Stellen Sie den Unterschied zwischen „Code“ und „Schlüssel“ dar!
4. Nennen Sie zu den bekannten Schlüsseln je ein Beispiel!
5. Stellen Sie die Zeichenreihe „Datenverarbeitung“ im R 300-Code (Lochkarte) dar!
6. Warum legt man besonderen Wert auf sorgfältige Datenerfassung?
7. Welche Möglichkeit kennen Sie, um die Fehlerhäufigkeit bei der Datenerfassung zu senken?
8. Weshalb werden Texte im Datenverarbeitungsprozeß verschlüsselt?
9. In eine Lochkarte soll die Zeichenreihe „ROBOTRON 300“ ge-
locht werden. Wie viele Lochspalten und wie viele Lochungen
werden benötigt?

6

Zu einigen Grundlagen der Programmierung

6.1. Sprachebenen im Rahmen der Programmierung

Eine EDVA muß durch den Menschen in die Lage versetzt werden, eine Aufgabe zu lösen, das heißt, einen geplanten Prozeß durchzuführen. Sie ist nicht in der Lage, nach eigenem Ermessen zu arbeiten.

Für das Lösen einer Aufgabe benötigt eine EDVA eine exakte Arbeitsvorschrift, in der in vielen kleinen Einzelschritten vorgegeben ist, welche Operationen in welcher Reihenfolge auszuführen sind.

Stellt man einer EDVA zum Beispiel die Aufgabe, eine Zahl a zu einer Zahl b zu addieren, so erfordert dieser Prozeß eine Folge von Anweisungen.

1. Die Zahl a muß in den Hauptspeicher eingegeben werden.
2. Die Zahl b muß ebenfalls in den Hauptspeicher eingegeben werden.
3. Es ist der EDVA mitzuteilen, daß sie zur Zahl a die Zahl b addieren soll.
4. Das Rechenergebnis soll deutlich sichtbar ausgegeben werden.

Diese einzelnen Schritte bezeichnet man als **Befehle**. Sie bilden in ihrer Gesamtheit das **Programm**.¹⁾

Jeder einzelne Befehl wird auf dem Hauptspeicher der EDVA abgespeichert. Die Befehle werden nacheinander vom Steuerwerk übernommen und interpretiert. Das Steuerwerk ist in der Lage zu unterscheiden, ob es sich beispielsweise um einen Eingabebefehl oder einen Sprungbefehl handelt (↗ 60).

Vom Steuerwerk aus werden die vom Befehl angesprochenen Teile der EDVA zur Ausführung angeregt.

- ▷ Wir prägen uns ein:
Eine EDVA arbeitet nicht nach eigenem Ermessen.
Sie benötigt zum Lösen einer Aufgabe eine Arbeitsvorschrift, die als **Programm** bezeichnet wird.

¹⁾ παραγραφειν (griech.) — vorher schreiben

Das Programm besteht aus einer Vielzahl von Einzelschritten, den **Befehlen**.

Die Befehle werden nacheinander in der Reihenfolge, wie sie auf dem Hauptspeicher stehen, abgearbeitet.

Der Programmierer bedient sich beim Aufstellen eines Programmes einer besonderen Sprache, der **Programmiersprache**.

Eine Programmiersprache hat ebenso wie die natürlichen Sprachen (z. B. Deutsch, Russisch, Englisch) ein **Alphabet**, das heißt, eine geordnete Menge von Zeichen. Nach syntaktischen Regeln werden aus dem Alphabet Wörter gebildet.

Diese Wörter werden als **Ausdrucksmittel** einer Programmiersprache bezeichnet.

□ **BEISPIEL 6/1:**

Elemente des Alphabets von Programmiersprachen sind Ziffern und Buchstaben.

Reiht man Ziffern und Buchstaben so aneinander, daß sich die Zeichenreihe „ZAHL 1“ ergibt, so hat man ein Wort gebildet, das man als Ausdrucksmittel einer Programmiersprache verwenden könnte.

Maschinensprache ist die einfachste Form einer Programmiersprache. Mit ihr arbeitet die Maschine unmittelbar. Alphabet und Ausdrucksmittel der Maschinensprache sind vom Konstrukteur der Anlage vorgegeben und genau auf die Anlage zugeschnitten.

Das Programmieren in Maschinensprache hat einige Nachteile. Sie setzt die genaue Kenntnis der speziellen EDVA voraus und ist sehr zeitaufwendig.

Assemblersprachen wurden deshalb eingeführt. Sie sind zwar immer noch an einen speziellen Maschinentyp gebunden, lassen sich aber vom Programmierer relativ leicht erlernen. Die auszuführende Operation wird mnemotechnisch¹⁾ verschlüsselt. Beispielsweise nimmt man „AD“ für Addition und „M“ für Multiplikation.

Hauptspeicheradressen, auf die sich das Programm bezieht, werden symbolisch definiert. Das heißt, im Programm werden keine echten Hauptspeicheradressen angegeben, sondern Symbole. Soll zum Beispiel ab Adresse 500 die Variable A gespeichert werden, so benutzt man nicht die Adressenangabe 500, sondern den Variablennamen A.

Ein in Assemblersprache formuliertes Programm ist der Anlage nicht mehr direkt verständlich. Es existiert ein **Übersetzungsprogramm**, das jeden Assemblerbefehl in einen Maschinenbefehl übersetzt. Assemblersprachen bezeichnet man auch als **maschinenorientiert**.

¹⁾ *mnemonisch* (aus dem Griech.) — das Gedächtnis stärkend

Problemorientierte Programmiersprachen sind für die Formulierung bestimmter Probleme geeignet. Sie existieren völlig unabhängig von der Struktur der Anlage. Entsprechend den Hauptanwendungsgebieten der problemorientierten Sprachen unterscheidet man Sprachen für wissenschaftlich-technische Verfahren und Sprachen für ökonomische Verfahren. Auch für problemorientierte Programmiersprachen existieren Übersetzungsprogramme, die das Programm in die für die entsprechende Anlage verständliche Maschinensprache übertragen.

▷ Wir merken uns:

Eine Sprache, mit deren Hilfe Programme für Rechenautomaten formuliert werden, bezeichnet man als **Programmiersprache**. Es existieren drei Ebenen der Programmiersprachen:

1. **Maschinensprache;**
2. **Assemblersprache;**
3. **Problemorientierte Sprache.**

6.2. Programmablaufplanung

Eine Anweisung in der Form

$$a := b \quad (a - \text{Variable}; \quad b - \text{Ausdruck})$$

bezeichnet man als **Ergibtanweisung**.¹⁾

Einer Variablen wird der Wert *zugeordnet*, der sich aus den *gerade gültigen* Werten des Ausdrucks ergibt.

Die Richtung des Ergibtzeichens wird durch den Doppelpunkt bestimmt.

Im Gegensatz zur Gleichung ist die Ergibtanweisung nicht symmetrisch. Es ist ein Unterschied, ob es $y := a$ oder $a := y$ heißt.

Im ersten Fall wird der Variablen y der Wert von a zugewiesen. Im zweiten Fall wird der Variablen a der Wert von y zugewiesen.

In der folgenden Übersicht werden Gleichung und Ergibtanweisung voneinander unterschieden.

Gleichung	Ergibtanweisung
Eine Gleichung ist <i>symmetrisch</i> . Wenn $a = b$, so $b = a$. Vertauscht man die Seiten einer Gleichung, so ist diese Umformung äquivalent.	Eine Ergibtanweisung ist <i>asymmetrisch</i> . Wenn $a := b$, so nicht $b := a$. Vertauscht man die Seiten einer Ergibtanweisung, so ist diese Umformung nicht äquivalent.

¹⁾ Das Ergibtzeichen $:=$ (gelesen: ergibt sich zu) wurde von Konrad Zuse (geb. 1910) eingeführt.

Eine Gleichung ist eine <i>Aussage</i> oder eine <i>Aussageform</i> .	Eine Ergibtanweisung widerspiegelt einen <i>Prozeß</i> .
Eine Gleichung ist <i>statisch</i> . $k = k + 1$ gilt für kein k .	Eine Ergibtanweisung ist <i>dynamisch</i> . Die Anweisung $k := k + 1$ erhöht den Wert k um eins.

□ BEISPIEL 6/2:

- a) $x := 1$ Der Variablen x wird der Wert 1 zugewiesen.
 b) $s := \emptyset$ Der Variablen s wird der Wert $\emptyset^1)$ zugewiesen.
 c) $a := b + c$ Es wird die Summe der Werte b und c gebildet.
 Der entstandene Wert wird der Variablen a zugewiesen.
 d) $k := k + 1$ Der Variablen k wird der um 1 erhöhte Wert der Variablen k zugewiesen.

▷ Wir halten fest:

Die diagrammartige Darstellung des Lösungsweges einer Aufgabe bezeichnet man als **Programmablaufplan** (PAP).

Ein PAP soll dem Programmierer eine bildhafte Vorstellung über die Art und die Reihenfolge der im Rechner ablaufenden Operationen geben.

Für die Aufstellung von PAP gibt es zwei Darstellungen, die *Kästchen-* und die *Leitlinienmethode*.

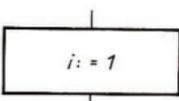
Wir befassen uns mit der am weitesten verbreiteten **Kästchenmethode**. Sie ist im DDR-Standard (TGL 22541) festgelegt.

Die einzelnen PAP-Schritte werden durch Angabe der entsprechenden Operationen innerhalb umrandeter Felder dargestellt. Diese einzelnen Symbole sind miteinander durch Programmlinien so verbunden, daß der Ablauf der Operationen eindeutig hervorgeht.

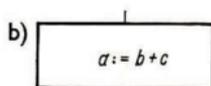
Wir stellen im folgenden die Arten der Kästchen vor.

In das **Operationskästchen** wird eine Ergibtanweisung eingetragen.

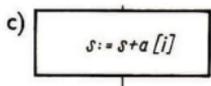
□ BEISPIEL 6/3:

- a)  Der Variablen i wird der Wert 1 zugewiesen.

¹⁾ Zur besseren Unterscheidung zwischen der „Ziffer 0“ und dem „Buchstaben 0“ wird die Null durchgestrichen.



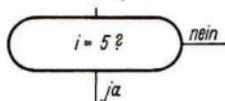
Es soll die Summe der Werte von b und c berechnet werden. Das Ergebnis wird der Variablen a zugewiesen.



Zum Wert von s wird der Wert von a , addiert, das Ergebnis ist der neue Wert von s .

In das **Alternativkästchen** wird eine Testanweisung in Form einer Frage eingetragen. Die Antworten sind „ja“ oder „nein“, mit je einem Ausgang am Alternativkästchen.

□ BEISPIEL 6/4:



Es wird gefragt, ob i den Wert 5 hat. Bei der Antwort „ja“ wird der PAP im JA-Zweig fortgesetzt, im anderen Fall im NEIN-Zweig.

In das **Eingabekästchen** bzw. **Ausgabekästchen** werden die Größen eingetragen, die ein- bzw. ausgegeben werden sollen.

□ BEISPIEL 6/5:



Es soll der Wert a eingegeben werden.



Es soll der Wert b ausgegeben werden.



Symbol für **Programm**anfang



Symbol für **Programm**ende

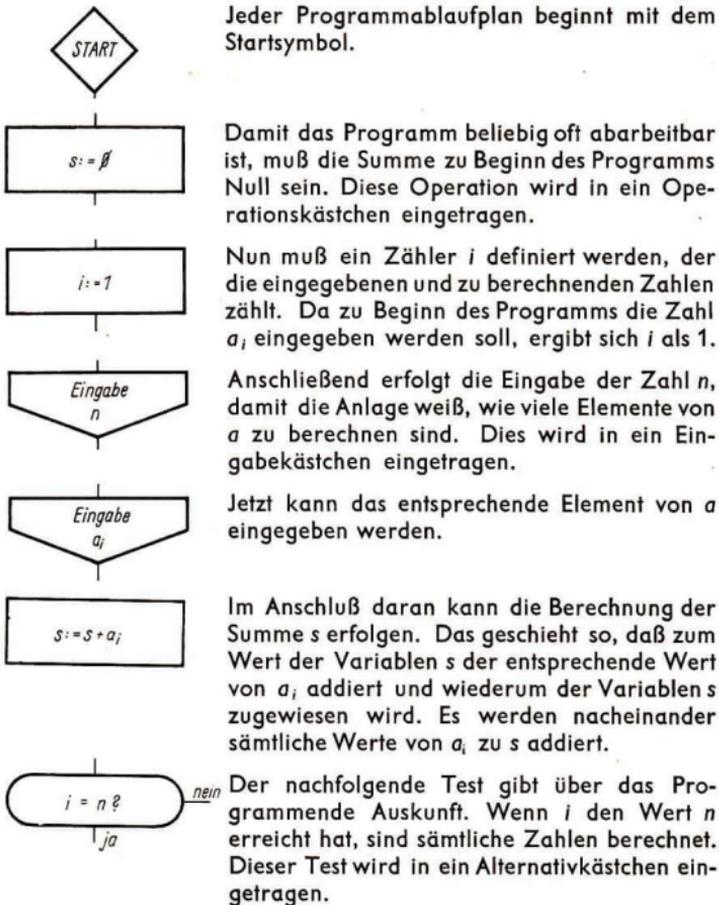


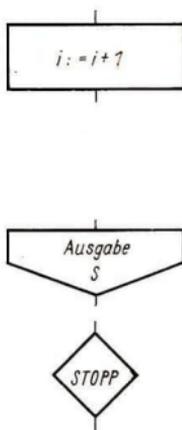
BEISPIEL 6/6:

Einer EDVA soll folgende Aufgabe übertragen werden:
Von einer vorgegebenen Menge ganzer Zahlen ist die Summe zu bilden und auszugeben.

$$S = \sum_{i=1}^n a_i$$

Wir erläutern zunächst die einzelnen Schritte des aufzustellenden PAP.





In seiner Gesamtdarstellung ergibt der PAP Bild 47.

Wir verfolgen den NEIN-Zweig weiter. Das heißt, i hat den Wert n noch nicht erreicht. In diesem Fall muß der Zähler i um 1 erhöht werden, damit der nächste Wert a_i eingegeben und berechnet werden kann. Nach der Erhöhung des Zählers wird zur Eingabe des nächsten Elementes von a_i verzweigt.

Im JA-Zweig ist die Berechnung sämtlicher Werte von a_i beendet, und die Summe s kann ausgegeben werden.

Der PAP wird mit dem STOPP-Symbol abgeschlossen.

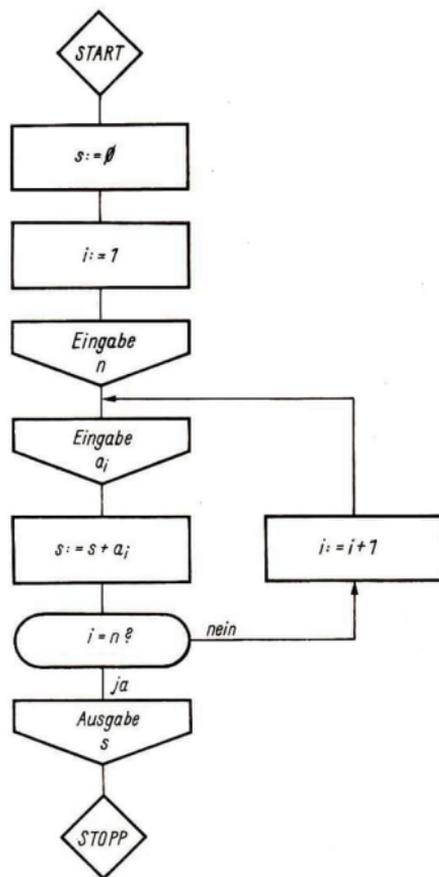


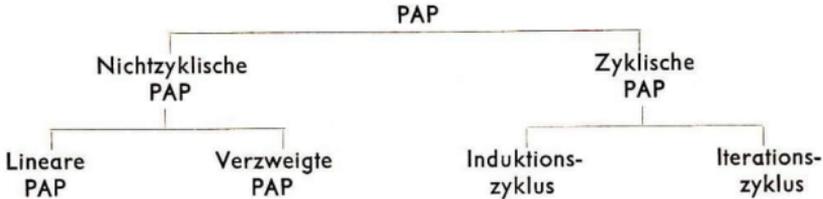
Bild 47

6.3. Strukturen von Programmablaufplänen

Man unterscheidet zwei prinzipielle *Strukturen* von PAP:

Nichtzyklische PAP;
Zyklische PAP.

Diese beiden Strukturen lassen sich noch weiter differenzieren.



Linearer PAP heißt der einfachste Fall eines nichtzyklischen PAP. Man bezeichnet ein solches Programm auch als **Geradeausprogramm**. Es ist dadurch gekennzeichnet, daß *jeder Schritt genau einmal* durchlaufen wird. In der Regel sind solche Geradeausprogramme nur Ausschnitte innerhalb eines komplexen PAP.

Bild 48 ist ein einfaches Schema eines linearen PAP.

Verzweigter PAP ist ein weiterer nichtzyklischer PAP. Er enthält als typische Merkmale logische Entscheidungen (**Alternativkästchen**), wobei entsprechend der Resultate dieser Entscheidungen nur bestimmte Zweige

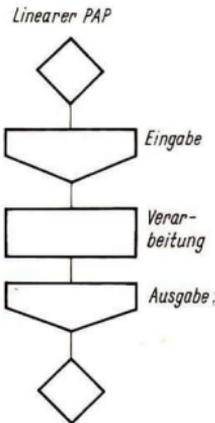


Bild 48

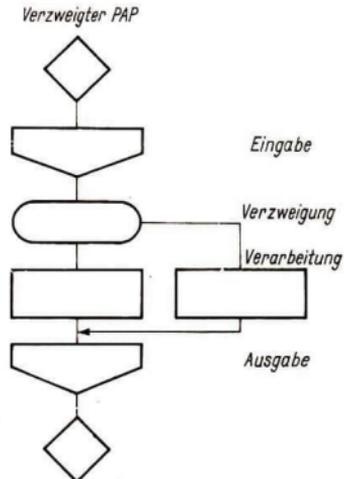


Bild 49

des PAP durchlaufen werden. Die anderen Zweige sind dann von der Verarbeitung ausgeschlossen. Die Schritte im verzweigten Programm werden also *höchstens einmal* durchlaufen.

Bild 49 ist ein einfaches Schema eines verzweigten PAP.

Bemerkung: Weitaus komplizierter sind zyklische PAP. Auf deren Behandlung verzichten wir deshalb in diesem Buch.

□ **BEISPIEL 6/7:**

Es ist ein PAP für die Berechnung des Flächeninhaltes eines Rechtecks aufzustellen (Bild 50). Wir beginnen mit dem Startsymbol. Anschließend erfolgt die Eingabe der Werte für die Länge der Seiten. Im folgenden Operationskästchen wird angegeben, wie sich die Größe A aus den Größen a und b ergibt. Im Anschluß an die Ausgabe von A ist das Programm beendet.

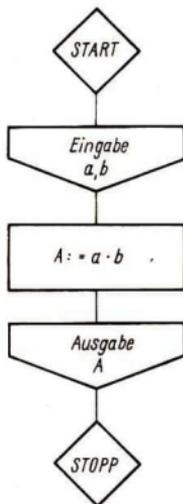


Bild 50

□ **BEISPIEL 6/8:**

Von drei Zahlen x , y , z , die sich bereits auf dem Speicher befinden, soll die kleinste ausgewählt werden und auf einem besonderen Bereich (KLZ) abgespeichert werden (Bild 51). Zunächst vergleichen wir die Zahlen x und y . Für den Fall, daß x kleiner als y ist, vergleichen wir x mit z . Ergibt sich nun, daß x kleiner ist als z , so ist x die kleinste Zahl. Im anderen Fall ist z die

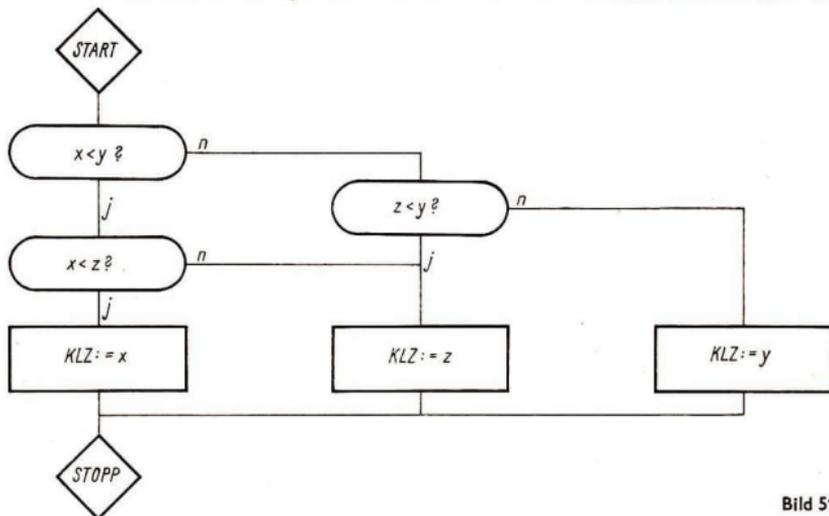


Bild 51

kleinste Zahl. Ergibt sich aus dem Vergleich der Zahlen x und y , daß x nicht kleiner als y ist, so wird z mit y verglichen. Ist z kleiner als y , so ist z die kleinste Zahl. Im anderen Fall ist y die kleinste Zahl.

□

BEISPIEL 6/9:

Es liegen Lochkarten vor, die je Artikelnummer den Preis je Stück und die Stückanzahl enthalten. Essoll der Gesamtpreis berechnet und in die Lochkarte eingestanzt werden. Das Ende der Datenkarten ist durch eine negative Artikelnummer gekennzeichnet (Bild 52).

Als erstes erfolgt die Eingabe der Lochkarte. Danach wird abgefragt, ob es sich um die letzte Karte handelt. Ist das nicht der Fall, so wird der Gesamtpreis errechnet und in die Lochkarte ausgegeben. Anschließend kann zur nächsten Lochkarteneingabe gesprungen werden. Handelt es sich bei der Abfrage um die letzte Karte, so wird das Programm beendet.

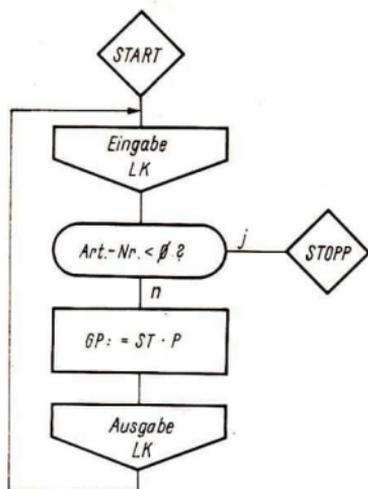


Bild 52

6.4. Maschinensprache

Das Programmieren in **Maschinensprache** setzt immer die Kenntnis eines speziellen Anlagentyps voraus.

Die einzelnen Anlagentypen unterscheiden sich grundsätzlich im **Alphabet**, **Befehlsvorrat**, **Befehlsaufbau**, **Speicherbelegungsplan** und vielem anderen mehr.

Die nachfolgenden Ausführungen beziehen sich auf die EDVA R 300.

Zum Alphabet

Das Alphabet der EDVA R 300 umfaßt 64 Zeichen.

26 Großbuchstaben A, ..., Z

10 Ziffern 0, ..., 9

28 Sonderzeichen, z. B.:

- Leerzeichen
- : Doppelpunkt

- (Runde Klammer auf
-) Runde Klammer zu
- [Eckige Klammer auf
-] Eckige Klammer zu
- Punkt
- + Plus
- Minus
- ; Semikolon
- ! Ausrufezeichen
- ? Fragezeichen
- * Stern
- = Gleichheitszeichen
- % Prozent
- ' Apostroph
- / Schrägstrich
- < Kleiner als
- > Größer als
- , Komma

Im Bild 53 sind sämtliche Zeichen des R 300 und die zugeordneten Bitkombinationen zusammengestellt.

VU	0000	0	+	-	od-0	,	/
	000L	1	A		Kod-1	S	
	00LL	2	B		Kod-2		
	00LL	3	C		Lod-3	T	
	0L0L	4	D		M od-4	U	
	0L0L	5	E		N od-5	V	
	0L7L	6	F		O od-6	W	
	0L7L	7	G		P od-7	X	
	L000	8	H		Q od-8	Y	
	L00L	9	I		R od-9	Z	
	L0L0		~	≈		≈	
	L0L0		□				
	L0L7		#	·)	,	
	L077		(,	*	%	
	L777		:	!	=	Δ	
	L777		∇	"	<	>	
	L777]]	"	?	[

Bild 53

Zum Befehlsvorrat

Unter **Befehlsvorrat** versteht man die Menge aller vorhandenen Befehle einer EDVA.

Zur Veranschaulichung des Befehlsvorrates der EDVA R 300 betrachten wir noch einmal Bild 47 (→ 80).

Im Bild 54 ist erkennbar, welche Operationen eine EDVA ausführen kann.

Addition, Subtraktion, Multiplikation und Division werden als **arithmetische Operationen** bezeichnet.

Zu den **Vergleichsoperationen** zählt man die numerischen oder alphanumerischen Vergleiche zwischen zwei Variablen. Das Ergebnis ist dann eine Vergleichsaussage in der Form „kleiner als“, „gleich“ oder „größer als“.

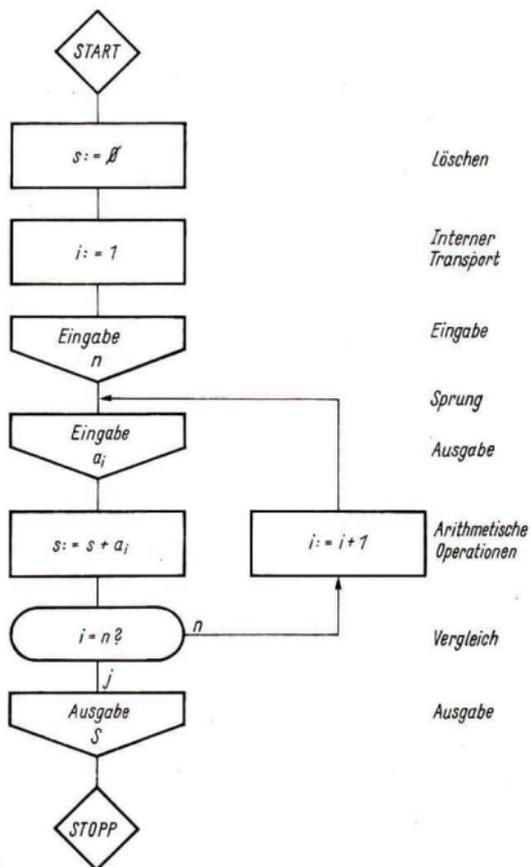


Bild 54

Als **interne Transportoperationen** bezeichnet man die Transporte vom Hauptspeicher zum Akkumulator oder umgekehrt.

Ein- und Ausgabeoperationen sprechen das periphere Gerät an und leiten den Datentransport vom Eingabegerät bzw. zum Ausgabegerät ein.

Alle diese Befehle besitzen ein gemeinsames Merkmal. Sie werden von der EDVA nacheinander, d. h. **sequentiell**, abgearbeitet. Nachdem ein Befehl abgearbeitet wurde, erfolgt der Aufruf an den sich unmittelbar auf dem Hauptspeicher anschließenden Befehl.

Diese Regel wird durch die **Sprungbefehle** unterbrochen. Nach der Abarbeitung eines Sprungbefehls wird das Programm an der Hauptspeicheradresse fortgesetzt, die im Adreßteil des Befehls angegeben wurde.

Zum Befehlsaufbau

Ein Befehl der EDVA R 300 hat eine konstante Länge von 6 Zeichen. Im höchsten Zeichen wird der **Operationshauptteil (OHT)**, im nächsten der **Operationszusatzteil (OZT)** angegeben. Die restlichen 4 Zeichen bilden den **Adreßteil** (Bild 55).

Sämtliche Operationen, die die Anlage auszuführen in der Lage ist, werden im Operationshauptteil verschlüsselt angegeben. Im Operationszusatzteil erfolgt eine genauere Beschreibung der Operation. Der Adreßteil gibt an, auf welche Hauptspeicheradresse sich die Operation bezieht.

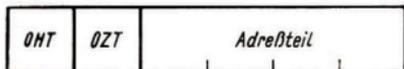


Bild 55

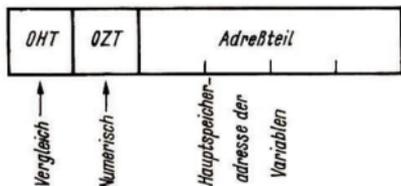


Bild 56



BEISPIEL 6/10:

Numerischer Vergleich mit der Variablen einer Hauptspeicheradresse (Bild 56)

Wir geben einen auszugweisen Überblick über die wichtigsten Befehle in R 300-Maschinensprache.

Befehle		Operations- hauptteil	Operations- zusatzteil
Transport- befehle	$\langle HS \rangle \rightarrow \langle AC \rangle^1$	+	0
	$\langle AC \rangle \rightarrow \langle HS \rangle$	A	0
Arithmetische Befehle	Addition	-	0
	Subtraktion	J	0
Vergleichs- befehle	Numerisch $\langle HS \rangle$ mit $\langle AC \rangle$	8	0
Sprung- befehle	Sprung unbedingt	1	0
	Sprung bei Vergleichs- aussage „gleich“	1	5
	Sprung bei Vergleichs- aussage „kleiner als“	1	6
	Sprung bei Vergleichsaussage „größer als“	1	7
	Sprung unbedingt mit HALT	1	+

□ BEISPIEL 6/11:

Folgende Befehle sollen in Maschinensprache programmiert werden.

a) Transport einer Variablen, die ab Adressen 4181 gespeichert ist, in den Akkumulator + 0 4 1 8 1

b) Addition einer Variablen ab Adresse 7524 zu einer Variablen im Akkumulator - 0 7 5 2 4

c) Transport des Ergebnisses vom Akkumulator zum Hauptspeicher ab Adresse 5000

A 0 5 0 0 0

d) Vergleich des Akkumulatorinhaltes mit einer Variablen der Hauptspeicheradresse 6000

8 0 6 0 0 0

e) Sprung bei Vergleicheraussage „gleich“ zur Adresse 500

1 5 0 5 0 0

Zum Speicherbelegungsplan

Die Daten, die vom Programm verarbeitet werden sollen, belegen ebenso wie die Befehle selbst Speicherplätze. Es muß genau festgelegt werden, unter welchen Adressen die Operanden im Hauptspeicher zu finden sind. Man verschafft sich zunächst einen Überblick, für welche Daten Speicherplatz zu reservieren ist. Im Anschluß an die Daten können dann die Befehle abgespeichert werden.

¹⁾ Man verwendet für die Angabe des Inhaltes eines Speicherplatzes x das Symbol $\langle x \rangle$ und für die Angabe der Adresse des Speicherplatzes x das Symbol $\rangle x \langle$.



BEISPIEL 6/12:

Wir erarbeiten zu dem im Bild 51 (↗ 82) dargestellten PAP den Speicherbelegungsplan und das Maschinenprogramm.

Zum Speicherbelegungsplan überlegen wir:

Das Programm benötigt Speicherplatz für die Zahlen x, y, z und KLZ. Wir setzen die Länge der Zahlen von 10 Zeichen als gegeben voraus. Das Programm soll ab Adresse 1000 beginnen.

Es ergibt sich folgende Speicherbelegung.

>x< — 1000
 >y< — 1010
 >z< — 1020
 >KLZ< — 1030

Der erste Befehl beginnt bei Adresse 1040.

Für das Maschinenprogramm erläutern wir in der folgenden Übersicht die Folge der Befehle.

Maschinenprogramm	Befehlsfolge
+01010 801000 161082	Transport der Zahl y in den Akkumulator (AC) Vergleich der Zahl y mit der Zahl x Sprung bei Vergleichsaussage „kleiner“ zu dem Befehl mit der Adresse 1082
801020 161100	Vergleich der Zahl y mit der Zahl z Sprung bei Vergleichsaussage „kleiner“ zu dem Befehl mit der Adresse 1100
A01030 1+1040 +01020	Transport der Zahl y in den HS-Bereich der Zahl KLZ Sprung mit Halt zum Programmstart Transport der Zahl z in den AC. Dieser Befehl ist auf der Adresse 1082 abgespeichert und wird bei erfüllter Sprungbedingung vom Befehl 3 angesprungen.
801000 161118	Vergleich mit der Zahl x Sprung bei Vergleichsaussage „kleiner“ zu dem Befehl mit der Adresse 1118
+01020	Transport der Zahl z in den AC. Dieser Befehl ist auf der Adresse 1100 abgespeichert und wird bei erfüllter Sprungbedingung vom Befehl 5 angesprungen.
A 01030 1+1040 +01000	Transport der Zahl z in den HS-Bereich der Zahl KLZ Sprung mit Halt zum Programmstart Transport der Zahl x in den AC. Dieser Befehl ist auf der Adresse 1118 abgespeichert und wird bei erfüllter Sprungbedingung vom Befehl 10 angesprungen.
A01030 1+1040	Transport der Zahl x in den HS-Bereich der Zahl KLZ Sprung mit Halt zum Programmstart

6.5. Maschinensorientierte Programmiersprache (Assemblersprache)

Wir befassen uns im folgenden mit einer einfachen Darstellung der **Assemblersprache MOPS**¹⁾ (↗ 75f.).

Die Assemblersprache stellt für den Programmierer eine Hilfe dar, in der die Nachteile der Maschinensprache nicht auftreten.

Der Befehl „Transport vom Hauptspeicher in den Akkumulator“ lautet in Maschinensprache $+0$. In der Assemblersprache werden die beiden Anfangsbuchstaben der Worte „Transport“ und „vom“ angegeben, also „TV“.

Da sich alle Befehle auf den Hauptspeicher beziehen, ist hierfür eine zusätzliche Angabe nicht erforderlich. Soll beispielsweise eine Variable, die ab Adresse 1000 abgespeichert ist, in den Akkumulator transportiert werden, so lautet der vollständige Befehl in Maschinensprache $+01000$. In der Assemblersprache wird diese Variable vereinbart und mit einem **Merkmal** versehen. Die Zuordnung eines Merkmals zu einer Variablen bezeichnet man als **symbolische Adressierung**.

Die **Vereinbarung** von Variablen ist gleichbedeutend mit Reservierung von Speicherplatz.

Die mit einem Merkmal versehene Variable kann von jedem Assemblerbefehl angesprochen werden. Wurde beispielsweise die Variable, die in den Akkumulator zu transportieren ist, mit dem Merkmal A versehen, so lautet der Assemblerbefehl TV A.

Merkmale können aber auch an Befehle selbst vergeben werden. Das ist für Programmverzweigungen von besonderer Bedeutung.

▷ Wir prägen uns ein:

1. Der Operationscode wird **mnemonisch verschlüsselt**.
2. Die Zuordnung von Speicherplatz für Variable bzw. Konstanten erfolgt durch **Vereinbarung**.
3. Der Zugriff zu Variablen, Konstanten oder Befehlen erfolgt durch **symbolische Adressierung**.

□ BEISPIEL 6/13:

PS1	TV	BER	Transport der Variablen BER in den AC
	AD	ZAHL	Addition der Variablen ZAHL zur Variablen BER
	TN	SUM	Transport des Ergebnisses nach dem HS in die Variable SUM
	S	PS1	Rücksprung zum Befehl mit dem Merkmal PS1

¹⁾ Der Name MOPS ist die Abkürzung für „Maschinensorientiertes Programmiersystem“.

6.6. Beschreibung der internen Operationen und Ein-/Ausgabe-Operationen in MOPS

Die folgende Übersicht erläutert die Funktionen der internen Operationen und Ein-/Ausgabeoperationen.

Bemerkung: Die Befehle, die zum Verständnis des Beispiels 6/14 (↗ 91) beitragen sollen, sind ausführlich erläutert, zu jeder Klassifikation ist jedoch mindestens ein Vertreter angegeben.

Klassifikation	Operations-code	Funktion	Erläuterung
Transport-befehle	TV	$\langle HS \rangle \rightarrow \langle AC \rangle$	Transport vom Hauptspeicher zum Akkumulator Transport vom Akkumulator zum Hauptspeicher
	TN	$\langle AC \rangle \rightarrow \langle HS \rangle$	
Arithmetische Befehle	AD	$\langle AC \rangle := \langle AC \rangle + \langle HS \rangle$	Addition des Inhalts des Akkumulators zum Inhalt des Hauptspeichers Ergebnis im Akkumulator
Vergleichsbefehle	VG N	$\langle HS \rangle$ mit $\langle AC \rangle$	Numerischer Vergleich zwischen dem Inhalt des Akkumulators und dem Inhalt eines Hauptspeicherbereiches
Sprungbefehle	S		Unbedingter Sprung zur angegebenen Adresse Bedingter Sprung bei vorangegangener Vergleichsaussage „gleich“ Bedingter Sprung bei vorangegangener Vergleichsaussage „kleiner als“
	SJ =		
	SJ <		

	SJ >		Bedingter Sprung bei vorangegangener Vergleichsaussage „größer als“ Unbedingter Sprung mit anschließendem HALT
	S H		
Ein-/Ausgabeoperationen	E SR		Eingabe über Schreibmaschine
	A SR		Ausgabe über Schreibmaschine
	TN Epi		Eingabe über den i-ten Eingabepuffer in den Hauptspeicher
	TV APi		Ausgabe über den i-ten Ausgabepuffer aus dem Hauptspeicher



BEISPIEL 6/14:

Beispiel 6/12 (↗ 88) würde in MOPS programmiert wie folgt aussehen (Verzicht auf den sogenannten Vereinbarungsteil).

(1)	START	TV	Y	Transport von Y in den AC
(2)		VG N	X	Vergleich mit X
(3)		SJ <	PS2	Sprung bei $X < Y$ zu PS2 (Befehl 8)
(4)		VG N	Z	Vergleich mit Z
(5)		SJ <	PS5	Sprung bei $Z < Y$ zu PS5 (Befehl 11)
(6)		TN	KLZ	Transport von Y nach KLZ
(7)		S	H START	Sprung mit HALT zu START (Befehl 1)
(8)	PS2	TV	Z	Transport von Z in den AC
(9)		VG N	X	Vergleich mit X
(10)		SJ <	PS3	Sprung bei $X < Z$ zu PS3 (Befehl 14)
(11)	PS5	TV	Z	Transport von Z in den AC
(12)		TN	KLZ	Transport von Z nach KLZ
(13)		S	H START	Sprung mit HALT zu START (Befehl 1)
(14)	PS3	TV	X	Transport von X in den AC
(15)		TN	KLZ	Transport von X nach KLZ
(16)		S	H START	Sprung mit HALT zu START (Befehl 1)

6.7. Problemorientierte Programmiersprachen

Problemorientierte Programmiersprachen beziehen sich auf das von der Rechenanlage zu bearbeitende Problem.

In der EDV-Praxis werden vorwiegend mathematisch-technische und ökonomische Probleme bearbeitet. Die Programmiersprachen ALGOL, FORTRAN und PL/I sind von den Entwicklungskollektiven für die Bearbeitung solcher Probleme konzipiert worden.

ALGOL ist die Abkürzung für „**Algorithmic Language**“, eine Sprache, die speziell für mathematisch-technische Probleme entwickelt wurde. 1960 erschien der „Bericht über die algorithmische Sprache ALGOL 60“.

FORTRAN steht für „**Formula Translation**“. Auch diese Sprache eignet sich für mathematisch-technische Berechnungen. FORTRAN wurde 1954 entwickelt.

COBOL ist die Abkürzung für „**Common Business Oriented Language**“. Diese Sprache wird hauptsächlich für die Formulierung ökonomischer Probleme verwendet.

PL/I ist die Abkürzung für „**Programming Language One**“. Diese Sprache vereinigt alle Vorteile der bisher erschienenen Sprachen in sich. PL/I ist sowohl für mathematische als auch ökonomische Probleme geeignet.

Alle diese Sprachen haben gemeinsame Eigenschaften, die sie von Programmiersprachen tieferer Ebenen unterscheiden.

Problemorientierte Programmiersprachen existieren unabhängig von einem bestimmten Rechenautomaten. Sie sind in der Lage, ein Problem genau zu beschreiben, sofern es zu dem Bereich gehört, für das die Sprache konzipiert wurde.

Die folgenden Ausführungen beziehen sich aus den oben unter PL/I genannten Gründen auf die Sprache PL/I.

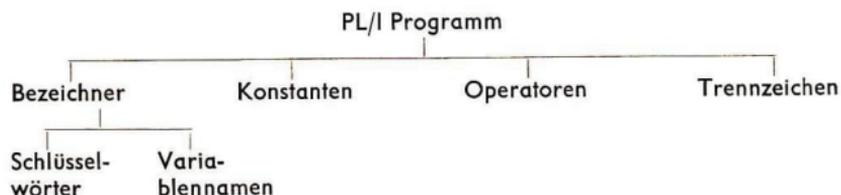
Es ist unser Ziel, einen Einblick in die Problematik der Beschreibung mathematischer beziehungsweise ökonomischer Aufgaben mit Hilfe einer Programmiersprache zu geben. PL/I bietet für diesen Zweck die günstigsten Voraussetzungen.

In der folgenden Übersicht sind zunächst einige der wichtigsten Zeichen des **Alphabets der Sprache PL/I** dargestellt.

Bezeichnung des Zeichens	Zeichen	Verwendung
Buchstaben	A, B, ..., Z	Text
Ziffern	0, 1, ..., 8, 9	Ziffern im Dezimal- und Dualsystem
Leerzeichen	b	Trennzeichen
Gleichheitszeichen	=	Gleichheitszeichen, Ergibtzeichen
Plus-Zeichen	+	Vorzeichen oder Addition
Minus-Zeichen	-	Vorzeichen oder Subtraktion

Stern-Zeichen	*	Multiplikationszeichen
Schrägstrich	/	Divisionszeichen
Öffnende Klammer	(Klammerung von Ausdrücken
Schließende Klammer)	
Komma	,	Dezimalpunkt
Punkt	.	
Apostroph	'	Kennzeichnung für Ende einer Anweisung
Prozentzeichen	%	
Semikolon	;	Steht hinter Marken
Doppelpunkt	:	„größer als“
Größer-Zeichen	>	„kleiner als“
Kleiner-Zeichen	<	
Fragezeichen	?	

Wir geben ferner eine Übersicht über die Programmelemente der Sprache PL/I.



Da es nicht beabsichtigt ist, den vollen Umfang der Sprache zu behandeln, beschränken wir uns bei der Erläuterung der Programmelemente auf eine kleine Auswahl, die zum Verständnis eines kleinen PL/I Programms völlig ausreicht.

Ein **Bezeichner** besteht aus alphanumerischen Zeichen.

Man unterscheidet in PL/I **Schlüsselwörter** und **Variablennamen**.

Variablen werden im Verlauf des Programmes Werte zugewiesen. Sie erhalten einen **Namen**.

- BEISPIEL 6/15:
 ZAHL
 WURZEL
 A
 BEGINN

Schlüsselwörter sind selbst Bestandteil der Sprache PL/I. Sie sind fest vorgegeben und können im Gegensatz zu Variablennamen nicht frei gewählt werden. Mit Schlüsselwörtern werden Anweisungen gegeben, zum Beispiel Sprünge, logische Vergleiche, Eingaben.

- **BEISPIEL 6/16:**
GOTO Sprunganweisung
GET Eingabe
DECLARE Datenvereinbarung

Datenelemente eines PL/I-Programmes, die bereits zu Programmbeginn Werte haben, bezeichnet man als **Konstanten**. Im Gegensatz zur Umgangssprache verwendet man in PL/I kein Komma, sondern einen Dezimalpunkt.

- **BEISPIEL 6/17:**
 1
 12.5
 0.47

Bei der Abarbeitung eines PL/I-Programmes werden den vereinbarten Variablen Werte zugewiesen. Das geschieht durch die Ergibtanweisung. In PL/I verwendet man dafür das Zeichen = . Auch in PL/I hat die Ergibtanweisung die Form

Variable = Arithmetischer Ausdruck ;

Die Ergibtanweisung hat nicht die Bedeutung des Gleichheitszeichens. Ein Vertauschen der beiden Seiten ist nicht erlaubt.

Arithmetische Ausdrücke sind mit Hilfe von Operatoren verknüpfte Variable oder Konstanten.

Operation	Operator	repräsentiert durch	Symbol
Addition	plus	Pluszeichen	+
Subtraktion	minus	Minuszeichen	-
Multiplikation	mal	Multiplikationszeichen	*
Division	durch	Divisionszeichen	/
Potenzierung	hoch	Potenzierungszeichen	**

- **BEISPIEL 6/18:**
 1/B * 0.3
 B ** N
 A * B + C

Wir wollen das vorliegende Programm nicht bis in alle Einzelheiten durchdringen. Es kommt vielmehr darauf an zu erkennen, mit welchen Mitteln bei der Programmierung im PL/I gearbeitet wird.

Der folgende Merkstoff soll noch einmal eine Zusammenfassung der wesentlichen Merkmale der problemorientierten Programmiersprache PL/I sein.

ZUSAMMENFASSUNG

1. Die Sprache PL/I ist an keinen speziellen Anlagentyp gebunden.
2. Man bedient sich in PL/I beim Beschreiben der Operationen der englischen Sprache.
3. Man schreibt für
Sprung GOTO
Wenn IF
Dann THEN
Dezimal DECIMAL
und bezeichnet diese Wörter als **Schlüsselwörter**.
4. Variable, die bei der Programmabarbeitung einen Wert zugewiesen erhalten, werden durch ihre **Variablennamen** beschrieben.
5. Durch Operatoren verknüpfte Variable bezeichnet man als **arithmetische Ausdrücke**.
 $A + B \quad X - Y$
6. Rechenergebnisse können mit Hilfe einer **Ergibtanweisung** einer Variablen zugewiesen werden.
 $A = B + C \quad X = Y/Z$
7. Das Programm kann auf einer EDVA nicht direkt abgearbeitet werden. Es muß in die jeweilige **Maschinensprache übersetzt** werden.

Neben den arithmetischen Operatoren gibt es in PL/I **Vergleichsoperatoren**, zum Beispiel = (gleich), < (kleiner als), > (größer als). Der durchgeführte logische Vergleich liefert eine wahre Aussage bei erfüllter Bedingung oder eine falsche Aussage bei nicht erfüllter Bedingung.

BEISPIEL 6/19:

$$A < B$$

$$B = 1 \text{ (im Gegensatz zur Ergibtanweisung)}$$

$$A > C$$

Jede Anweisung wird in PL/I mit einem Semikolon abgeschlossen.
Im Beispiel 6/20 wollen wir ein einfaches PL/I-Programm darstellen.

BEISPIEL 6/20:

Wir greifen wiederum auf den im Bild 51 (↗ 82) enthaltenen PAP zurück.

Das PL/I Programm hat dann folgendes Aussehen:

```
BEISP2:  PROCEDURE OPTIONS (MAIN);  
         DECLARE (X, Y, Z, KLZ) DECIMAL FIXED (10);  
         GET EDIT (X, Y, Z);
```

```

IF X < Y THEN GOTO PS2;
IF Z < Y THEN GOTO PS5;
KLZ = Y; GOTO PSENDE;
PS2: IF X < Z THEN GOTO PS3;
PS5: KLZ = Z;
      GOTO PSENDE;
PS3: KLZ = X;
PSENDE: PUT EDIT (KLZ); END BEISP2;

```

Sprachübersetzer

Sowohl Assemblersprachen als auch problemorientierte Sprachen sind von der EDV-Anlage nicht ohne weiteres zu verstehen. Beide müssen erst in die jeweilige Maschinensprache übersetzt werden.

Diese Übersetzung wird von der Maschine mit Hilfe eines speziellen Programms, des **Compilers**, durchgeführt. Das in der Assemblersprache oder in der problemorientierten Sprache geschriebene Programm wird auf Lochkarten übertragen. Man bezeichnet es dann als **Quellprogramm** oder auch **Ursprungsprogramm**. Das übersetzte und abarbeitungsfähige Maschinenprogramm heißt **Objektprogramm**.

Die Technik des Compilerens ist sehr kompliziert. Das Quellprogramm wird während des Übersetzungsvorganges syntaktischen und, soweit möglich, logischen Kontrollen unterzogen. Auf dem Drucker wird für den Programmierer ein **Compilerprotokoll** ausgegeben. Das Protokoll gibt Auskunft über den Verlauf der Übersetzung und stellt das wichtigste Arbeitsmittel für die **Programmtestung** dar.



AUFGABEN

1. Was versteht man unter einem Programm?
Was versteht man unter einem Befehl?
Welche Angaben muß ein Befehl enthalten?
2. Erläutern Sie den Unterschied zwischen Gleichheitszeichen und Ergibtzeichen!
3. Nennen Sie Unterscheidungsmerkmale zwischen Maschinensprache, Assemblersprache und problemorientierter Sprache!
4. Ordnen Sie folgende Anweisungen in die entsprechenden Sinnbilder für PAP ein:
 - a) $LK \rightarrow HS$
 - b) $s := \emptyset$
 - c) $s := s + a_i$
 - d) $i = 100?$
 - e) $HS \rightarrow ZS$
 - f) $K = \emptyset?$
5. Nennen Sie Strukturen von PAP!

6. Um welche Programmiersprache handelt es sich jeweils bei den folgenden Befehlen?

- a) 10 1350
- b) AD ZAE
- c) IF X = .5 THEN Y = 1;
- d) 050500
- e) TV BERA
- f) PROCEDURE OPTIONS (MAIN);
- g) IF X = 0 THEN GOTO NAME;
- h) DECLARE A DECIMAL FIXED (5);

7. Es sei die Speicherbelegung

Adresse 50 51 52 53 54 55

Inhalt A 0 1 9 x %

gegeben.

Was bedeuten dann $\langle 50 \rangle$, $\langle 55 \rangle$, $\rangle x \langle ?$

8. Es sei folgende Befehlsfolge gegeben.

```
START E SR A
      TV A
      M A
      TN A
      A SR A
      S H START
```

Beschreiben Sie, was die Anlage bei der Abarbeitung dieser Befehlsfolge ausführt!

- 9. Stellen Sie Programmablaufplan und Programm in Maschinensprache für die Berechnung des Flächeninhaltes eines Rechteckes auf!
- 10. Stellen Sie Programmablaufplan und Programm in Assemblersprache für die Berechnung der Lösungsmenge der linearen Gleichung $mx + n = 0$ auf!

7

Zur Einsatzvorbereitung der EDV in sozialistischen Betrieben

7.1. Allgemeine Probleme

Mit dem Einsatz einer EDVA oder eines Prozeßrechners in einem sozialistischen Betrieb gehen eine Reihe allgemeiner Probleme einher.

Beispielsweise sind mit der Einführung der EDV umfangreiche und gründliche Vorbereitungen der Werktätigen, verbunden mit entsprechenden Qualifizierungsmaßnahmen, erforderlich.

Vor allem sind folgende Fragen zu klären.

- Welche Mitarbeiter werden in welcher Weise von den Veränderungen betroffen?
- Welche Qualifizierungsanforderungen müssen erfüllt werden?
- In welchem Umfange wird Schichtarbeit nötig sein, und wie können die sich daraus ergebenden ideologischen und sozialen Probleme gelöst werden?
- Besteht nach der Qualifizierung für den betreffenden Mitarbeiter die Möglichkeit, im alten Kollektiv weiter zu arbeiten?
- Welche Voraussetzungen muß der einzelne Mitarbeiter für bestimmte Qualifizierungsvorhaben mitbringen?

Die verschiedenen gesellschaftlichen Organisationen des Betriebes, insbesondere die Betriebsparteiorganisation der Sozialistischen Einheitspartei Deutschlands, die Betriebsorganisation des Freien Deutschen Gewerkschaftsbundes und die Grundorganisation der Freien Deutschen Jugend, haben dabei eine hohe Verantwortung zu übernehmen. Es sind also nicht nur Aussprachen zwischen staatlichen Leitern und Mitarbeitern notwendig, sondern auch entsprechende Beschlüsse, die in Beratungen der genannten Organisationen gefaßt werden. Beiträge in der Betriebszeitung und differenzierte Schulungen, zum Beispiel in den Schulen der sozialistischen Arbeit, tragen zur Klärung grundsätzlicher und spezieller Probleme bei und sind Methoden, zwangsläufig auftretende Widersprüche lösen zu helfen.

▷ Wir erkennen:

Bei allen Fragen wird davon ausgegangen, welche Veränderungen und welche beruflichen Perspektiven sich für die Menschen im Prozeß der Einführung der EDV im sozialistischen Betrieb ergeben.

Der Mensch als Hauptproduktivkraft steht im Mittelpunkt aller Überlegungen bei der Einsatzvorbereitung der EDV. Das zeigt sich vor allem auch darin, daß durch den Einsatz der EDV der Mensch von vielen manuellen und geistigen Routinearbeiten befreit werden kann, beispielsweise in der Produktionsvorbereitung oder bei Berechnungsarbeiten. Infolgedessen verändert sich der Charakter seiner Arbeit und damit er selber. Der Mensch übernimmt mehr und mehr organisierende und regulierende beziehungsweise kontrollierende Aufgaben.

Durch die Befreiung von Routinearbeit hat der Mensch mehr Zeit zum Durchdenken der technologischen, ökonomischen und gesellschaftlichen Auswirkungen seiner persönlichen Tätigkeit. Er bekommt mehr Zeit für die Lösung geistig-schöpferischer Aufgaben.

7.2. Organisation der Einsatzvorbereitung von EDVA

Welchen Nutzen eine EDVA für einen Betrieb hat, hängt in starkem Maße von der gründlichen Vorbereitung des Einsatzes ab.

Die Organisation der Einsatzvorbereitung verläuft in verschiedenen **Etappen**. Diese Etappen können entweder in zeitlicher Aufeinanderfolge durchlaufen werden oder auch teilweise einander überlagern.

I	II	III	IV	V	VI
Ausarbeiten einer Studie	Aufstellen eines Arbeitsplanes	Grobprojekt	Feinprojekt	Programmierung	Umstellen der betrieblichen Organisation
Erfassen und Analysieren des Ist-Zustandes	Bilden eines Einsatzkollektivs		für die einzelnen Teilkomplexe	Programmablaufplan Codierung Aufbau von Testsätzen Maschinen-test der Programme	Erproben der Projekte und ihre Einführung

Zur Etappe I

Eine Studie ist eine Voruntersuchung. Man analysiert den bestehenden Zustand und entscheidet, ob sich der Einsatz einer eigenen EDVA für den Betrieb lohnt oder nicht.

Dabei sind folgende Aspekte zu berücksichtigen.

- Die Datenverarbeitungsprozesse, die automatisierungswürdig sind, sind zu ermitteln.
- Der zweckmäßigste Typ einer Datenverarbeitungsanlage ist zu ermitteln.
- Für die Etappen der Einführung sind mehrere Varianten zu entwickeln.
- Die Wirtschaftlichkeit der einzelnen Varianten ist zu prüfen.
- Der ökonomische Nutzen für den Betrieb ist zu bilanzieren.
- Die Auswirkungen auf die Arbeits- und Lebensbedingungen der Werk­­tätigen sind zu erkennen.

Auf der Grundlage der Prüfungsergebnisse sind Entscheidungsvorschläge zu machen.

Wir betrachten das Ausarbeiten einer Studie als eine wesentliche Grundlage der Entscheidungsfindung für den Einsatz einer EDVA in einem sozialistischen Betrieb und beziehen sie deshalb in die Folge der zu durchlaufenden Etappen ein.¹⁾

Zur Etappe II

Damit die Arbeit zielgerichtet ablaufen kann, muß die Vorbereitung des Einsatzes einer EDVA gründlich geplant werden. Es muß festgelegt werden, wie die einzelnen Teilaufgaben gelöst werden sollen, und es muß genau untersucht werden, welche Verflechtungen es zwischen ihnen gibt.

Als Methoden für die Arbeitsplanung der Einsatzvorbereitung eignen sich

- die graphische Darstellung;
- die Erläuterung.

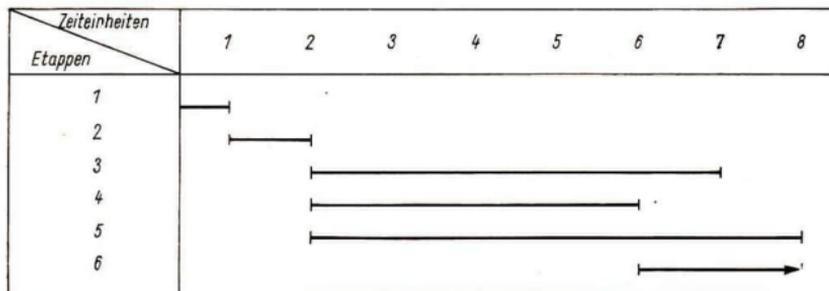


Bild 57

¹⁾ In einigen Titeln der Fachliteratur wird, abweichend von unserer Auffassung, das Ausarbeiten einer Studie nicht als besondere Etappe eingeordnet.

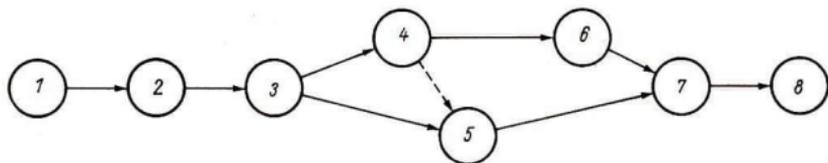


Bild 58

Beide Methoden gehören zusammen. Es genügt nicht, nur eine von beiden anzuwenden.

Die graphische Darstellung bildet den Hauptbestandteil des Arbeitsplanes.

Man verwendet dafür **Liniendiagramme** (Bild 57) und **Netzpläne** (Bild 58). Diese Hilfsmittel sind notwendig, um die Planung und Ablaufkontrolle rationell und sicher durchführen zu können. Mit Hilfe graphischer Darstellungen können nacheinander und parallel verlaufende Tätigkeiten besser übersehen werden. Man kann auch erkennen, welche Voraussetzungen für die Durchführung der weiteren Arbeitsschritte notwendig sind, welche Arbeitsverzweigungen möglich sind und welche Auswirkungen eintreten können, wenn Zeitverzögerungen auftreten.

Wir fassen zusammen:

Um rationell und effektiv die Aufgaben bei der EDV-Einsatzvorbereitung lösen zu können, ist es zweckmäßig, mit graphischen Methoden zu arbeiten. Die Praxis hat die Richtigkeit solcher Maßnahmen bewiesen.

Wir wenden uns nun den Problemen beim Bilden eines **Einsatzkollektivs** zu. In einem Einsatzkollektiv werden die Mitarbeiter zusammengefaßt, die für die gesamte Einsatzvorbereitung verantwortlich sind. Wie viele Mitarbeiter im Einsatzkollektiv tätig sind, hängt von der Aufgabenstellung ab.

Einem Einsatzkollektiv sollten angehören:

- Mitarbeiter des Betriebes, in dem die EDVA eingesetzt werden soll;
- Mitarbeiter von außerbetrieblichen Institutionen, zum Beispiel Vertreter des VEB Kombinat ROBOTRON, Vertreter der übergeordneten wirtschaftsleitenden Organe wie der Vereinigung Volkseigener Betriebe (VVB);
- andere Spezialisten.

Wird zum Beispiel der Einsatz eines Prozeßrechners vorbereitet, dann arbeiten im Einsatzkollektiv Ökonomen, Mathematiker, Verfahrenstechniker, Chemiker, Physiker, BMSR-Fachleute¹⁾, Programmierer, Facharbeiter u. a. mit. Voraussetzung für eine hohe Leistungsfähigkeit innerhalb eines solchen Kollektivs ist eine gut organisierte sozialistische Gemeinschaftsarbeit.

Die Mitglieder dieser Einsatzkollektive müssen ein hohes Maß an Einsicht in die ihnen gestellte gesellschaftliche Aufgabe haben. Willensstark, verantwortungsfreudig und begeistert müssen sie an die Verwirklichung der Aufgabe gehen.

¹⁾ BMSR-Technik als Abkürzung für „Betriebs-, Meß-, Steuerungs- und Regelungs-Technik“

Zur Etappe III

Unter der Ausarbeitung eines Grobprojektes verstehen wir die Beschreibung der konkret einzusetzenden Datenverarbeitungssysteme in ihren Grundzügen. Dabei sind folgende Aufgaben zu lösen.

1. Der objektive Informationsbedarf der einzelnen Leitungsorgane ist zu ermitteln.
2. Aufgabengebiete, die in das EDV-System einbezogen werden sollen, sind genau abzugrenzen, und diese Datenverarbeitungskomplexe sind in Teilkomplexe zu untergliedern.
3. Die im Betrieb bestehenden Benummerungssysteme sind auf ihre Weiterverwendbarkeit unter den Bedingungen der EDV zu überprüfen, oder es sind neue Benummerungssysteme zu entwickeln.
4. Die Komplexe, die durch die EDVA rationalisiert werden sollen, sind endgültig festzulegen.
5. Das EDV-System ist vorzubereiten.
6. Die notwendigen Ein- und Ausgabeinformationen sind festzulegen.
7. Zweckmäßige Datenträger sind zu bestimmen.
8. Die erforderliche Anlagenkonfiguration (Zentraleinheit, Geräte der 1. und 2. Peripherie) ist zu konzipieren.

Das Grobprojekt ist vom Einsatzkollektiv vor dem zuständigen Leitungskollektiv und den Mitarbeitern der Anwendungsbereiche zu verteidigen.

Zur Etappe IV

Unter der Ausarbeitung der Feinprojekte verstehen wir die Bereitstellung von Unterlagen, die den genauen Ablauf der künftigen Datenverarbeitung im Betrieb widerspiegeln. Auf ihrer Grundlage erfolgt dann die Programmierung. Dabei sind folgende Aufgaben zu lösen.

- Ermitteln aller erforderlichen Arbeitsgänge
- Festlegen konkreter Teilschritte für jeden Arbeitsgang
- Bestimmen des Inhaltes und des Aufbaues der vorgesehenen Datenträger

Zur Etappe V

Die Programmierer erarbeiten die Programme für die EDVA und überprüfen diese Programme zunächst im **Trockentest**. Anschließend werden sie nach Umsetzen in die „Sprache“ der EDVA auch einem **Maschinentest** unterzogen. Unter *Trockentest* verstehen wir das Prüfen der Programmablaufpläne vor der Programmierung, das heißt, ohne Benutzung des Rechners. Es werden dazu Beispiele so aufgebaut, daß alle Programmzweige durchlaufen werden. Dabei kommen sowohl Beispiele für Extremfälle als auch für Normalfälle in Betracht. Beim Testen der Beispiele wird vor allem der logische Aufbau des PAP überprüft und wenn nötig korrigiert. Besonders Verzweigungen und Entscheidungsstellen sind zu prüfen, da hier am häufigsten Fehler auftreten können.

Zur Etappe VI

Nachdem alle vorhergehenden fünf Etappen gründlich und gewissenhaft bearbeitet worden sind, erfolgt schrittweise die Einführung der projektierten EDV-Arbeitsweise in die Praxis.

Wir überblicken mit Hilfe des Bildes 59 noch einmal den Ablaufplan der Einsatzvorbereitung einer EDVA.

Nunmehr stehen den Menschen, die im Produktionsprozeß eines sozialistischen Betriebes arbeiten, die hohen Rechen- und Schaltgeschwindigkeiten und die große Speicherfähigkeit der Automaten zur Verfügung.

Sie werden durch ihre durch keine Maschine zu ersetzenden Eigenschaften wie Phantasie und Schöpferum neue Möglichkeiten der Anwendung von Datenverarbeitungssystemen erforschen. Sie werden Ihre Fähigkeit festigen, mit EDVA zu arbeiten, und es wird sich die Überzeugung herausbilden, daß die moderne sozialistische Großproduktion und die Informationsverarbeitung ohne Anwendung der EDV unmöglich ist.

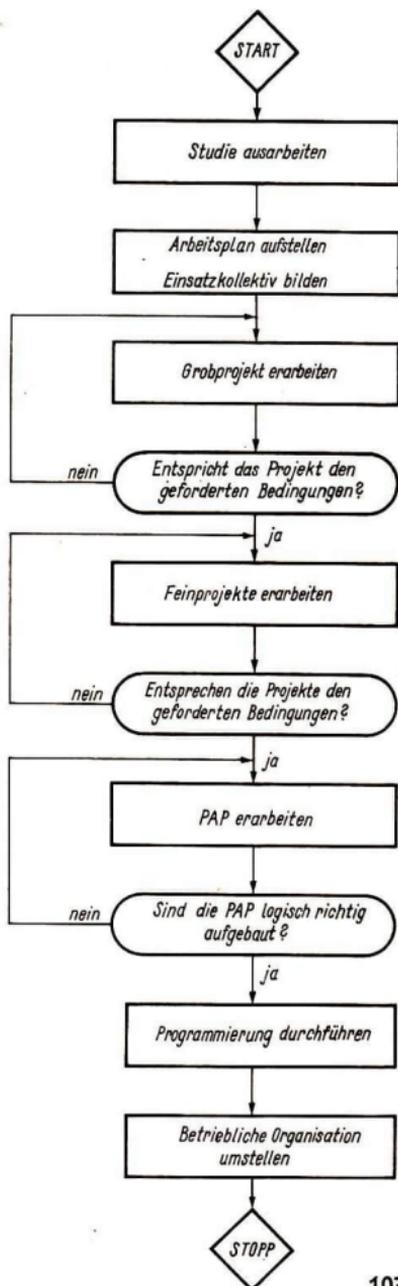


Bild 59



Forschungsaufträge

1. Informieren Sie sich darüber, ob in Ihrem Betrieb (UTP) die EDV bereits eingesetzt ist!
Beachten Sie dabei folgende Aspekte!
 - Wie wurde der Einsatz vorbereitet?
 - Welche Anwendungsbereiche wurden ausgewählt?
 - Wie wurden die Menschen in die Vorbereitung und in den Einsatz einbezogen?
 - Welchen ökonomischen Nutzen brachte die EDV seit dem Zeitpunkt ihres Einsatzes?
 - Welche Auswirkungen auf die Arbeits- und Lebensbedingungen der Werk tätigen haben sich ergeben? Wie wurden dabei auftretende Schwierigkeiten überwunden?
2. Beraten Sie in Ihrer Arbeitsgemeinschaft unter Hinzuziehung der Lehrer von UTP und ESP über Möglichkeiten, den Patenbetrieb oder andere Betriebe beim Einsatz der EDV zu unterstützen!
Solche Möglichkeiten könnten sein
 - Mithilfe beim Aufstellen kleiner PAP;
 - Mithilfe beim Verschlüsseln von Daten;
 - Mithilfe beim Bearbeiten von Lochkarten oder Lochstreifen;
 - Mithilfe beim Auswerten von EDV-Unterlagen.

Literaturverzeichnis

- [1] Bericht des Zentralkomitees an den VIII. Parteitag der Sozialistischen Einheitspartei Deutschlands. Berichterstatter: Genosse Erich Honecker. Dietz Verlag, Berlin 1971
- [2] Rechenschaftsbericht des Zentralkomitees der KPdSU an den XXIV. Parteitag der Kommunistischen Partei der Sowjetunion. Referent: L. J. Breshnew, Generalsekretär des ZK der KPdSU. APN Verlag, Moskau/Dietz Verlag, Berlin 1971
- [3] Staatsbürgerkunde Klasse 9. Volk und Wissen Volkseigener Verlag, Berlin 1973
- [4] Staatsbürgerkunde Klasse 10. Volk und Wissen Volkseigener Verlag, Berlin 1973
- [5] Einführung in die sozialistische Produktion, Lehrbuch für die Klassen 7 und 8. Volk und Wissen Volkseigener Verlag, Berlin 1972
- [6] Einführung in die sozialistische Produktion, Lehrbuch für Klasse 9, Industrie. Volk und Wissen Volkseigener Verlag, Berlin 1972
- [7] Einführung in die sozialistische Produktion, Lehrbuch für Klasse 10. Volk und Wissen Volkseigener Verlag, Berlin 1972
- [8] Mathematik, Lehrbuch für Klasse 6. Volk und Wissen Volkseigener Verlag, Berlin 1973
- [9] Mathematik, Lehrbuch für Klasse 9. Volk und Wissen Volkseigener Verlag, Berlin 1974
- [10] Mathematik in Übersichten, Wissensspeicher für die Klassen 8 bis 10. Volk und Wissen Volkseigener Verlag, Berlin 1974
- [11] Physik, Lehrbuch für Klasse 8. Volk und Wissen Volkseigener Verlag, Berlin 1972
- [12] Physik, Lehrbuch für Klasse 9. Volk und Wissen Volkseigener Verlag, Berlin 1972
- [13] Physik in Übersichten, Wissensspeicher für die Klassen 9 und 10. Volk und Wissen Volkseigener Verlag, Berlin 1974
- [14] Anleitung zum Computerspielzeug PIKO dat — Computer, Spielzeug und Lernmaschine. VEB Piko Sonneberg, 1969
- [15] Autorenkollektiv: Datenverarbeitung — Grundlagen und Einsatzvorbereitung. Staatsverlag der DDR, Berlin 1967
- [16] Autorenkollektiv: Einführung in die mathematische Logik — Einführung in die Mengenlehre — Aufbau der Zahlenbereiche. Volk und Wissen Volkseigener Verlag, Berlin 1973

- [17] *Budden, F. J.*: Zahlensysteme und Rechenautomaten. B. G. Teubner Verlagsgesellschaft, Leipzig 1972
- [18] *Bürger, E./Wittmar, G.*: Was ist – was soll Datenverarbeitung? Urania-Verlag, Leipzig · Jena · Berlin 1969
- [19] *ESEG Begriffe*, Ausgabe Dezember 1971. Herausgeber: Institut für Rationalisierung und Organisation der Elektroindustrie Dresden.
- [20] *Fachkunde für Datenverarbeiter*, Teil 2, 2. bearbeitete Auflage. Verlag Die Wirtschaft, Berlin 1970
- [21] *Glade, H./K. Manteuffel*: Am Anfang stand der Abacus. Urania-Verlag, Leipzig · Jena · Berlin 1973
- [22] *Gluschkow, W. M.*: Vier Generationen der EDVA. In: „Blickpunkt 2000“, Leipzig/Jena/Berlin 1972, S. 76 bis 86
- [23] *Gnedenko, B. W./W. S. Koroljuk und J. L. Justschenko*: Elemente der Programmierung. B. G. Teubner Verlagsgesellschaft, Leipzig 1964
- [24] *Kitow, A. J./N. A. Krinitzki*: Wie arbeitet eine elektronische Rechenmaschine? VEB Fachbuchverlag, Leipzig 1960
- [25] *Kovács, M.*: Rechenautomaten und logische Spiele. Gemeinschaftsausgabe des VEB Fachbuchverlages Leipzig und des Akademiai Kiadó Budapest, Budapest 1971
- [26] *Mathematik für die Praxis*, ein Handbuch, Teil II. Herausgegeben von K. Schröder. VEB Fachbuchverlag, Berlin 1966, S. 429 bis 553
- [27] *Mucke, G.*: Zu einigen technischen Problemen bei der Weiterentwicklung digitaler Rechenanlagen im militärischen Bereich. In: „Militärtechnik“, herausgegeben im Militärverlag der Deutschen Demokratischen Republik, Heft 7/1971, S. 296f.
- [28] *Murphy, J. S.*: Elektronische Ziffernrechner. VEB Verlag Technik, Berlin 1965
- [29] *Paulin, G.*: Kleines Lexikon der Rechentechnik und Datenverarbeitung, Reihe Automatisierungstechnik, Band 52. VEB Verlag Technik, Berlin 1971
- [30] *Philosophisches Wörterbuch*, Band 2. Herausgegeben von Georg Klaus und Manfred Buhr. VEB Bibliographisches Institut, Leipzig 1969
- [31] *Smers, H.*: Datenerfassung und Datenübertragung. In: Fernsehkurs des Deutschen Fernsehfunks, Themenkomplex 5/1. Verlag Die Wirtschaft, Berlin 1969
- [32] *Trachtenbrot, B. A.*: Wieso können Automaten rechnen? VEB Deutscher Verlag der Wissenschaften, Berlin 1968
- [33] *Varga, T.*: Logik für Anfänger, Aussagenlogik. Volk und Wissen Volkseigener Verlag, Berlin 1970
- [34] *Wissenspeicher Grundlagen der Datenverarbeitung*, 2. bearbeitete Auflage. Verlag Die Wirtschaft, Berlin 1970

Register

A

Ablauf
 Äußerer — 27, 28
 Innerer — 28
 Linearer Programm-plan 81
 Nichtzyklischer Programm-plan 81
 Programm-plan 27, 28, 77, 102
 Verzweigter Programm-plan 81
 Zyklischer Programm-plan 81
AC 63
Addition
 — im Dualsystem 33
 -smaschine 16, 23, 46
 Logische — 38
Adresse 52, 53, 63
 Symbolische — 89
Adreßteil 55, 86
Akkumulator 63
ALGOL 92
Algorithmus 28
Alphabet 75, 83, 84, 92
Alternative 37
Analogrechner 23
Anlage 8
 -nkonfiguration 102
Anlage 8
 -nkonfiguration 102
Arbeitsplan 99, 100
Assemblersprache 75, 76, 89
 — MOPS 89
Ausdruck
 Arithmetischer — 94, 95
 -smittel 75
Ausgabe 52
 -funktion 50
 -gerät 52

 Ausgabekästchen 78
 -operation 86, 91
Aussage 35
 -nverbindung 35
 Falsche — 35
 Wahre — 35

B

Bedienerverständigung 57
Befehl 74, 75
 Arithmetischer — 60, 87, 90
 -saufbau 83, 86
 -sfolge 88
 -sspeicher 17
 -svorrat 60, 83, 85
 Schreibmaschinen- 60
 Sprung- 60, 86, 87, 90
 Transport- 60, 87, 90
 Vergleichs- 87, 90
BESM 1 19
Bezeichner 93
Binäres Element 46
Blockschaltbild 62
Blockschaltplan 26, 28
BMSR 101
Byte 32

C

COBOL 92
Code 69, 70

Mnemischer Operations- 89
Operations- 89
Codierung 70, 99
Compiler 96
-protokoll 96

D

Daten 8, 66
-erfassung 66, 67
-erfassungsgerät 66
-flußplan 27, 28
-karte 69
-speicher 17
-struktur 63
-träger 8, 17, 66
Maschinenlesbarer -träger 66
Digitalrechner 23, 46
Disjunktion 38
Doppelerfassung 67
Dualsystem 30
Addition im — 33
Multiplikation im — 34
Subtraktion im — 33
Dualzahl 30, 31
Durchlaßrichtung 47, 48

E

EDVA 8, 46
Eingabe 51
-funktion 50
-gerät 51
-kästchen 78
-operation 86, 91
Einheit
Lochkarten-Lese-Stanz — 65
Rechen- 17
Steuer- 17
Zentral- 17, 52, 62
Einsatzkollektiv 99, 101
Einsatzvorbereitung 98, 99
Etappen der — 99, 100
Organisation der — 99, 103
Elektronenrechner 23
Elektronenröhre 46
ENIAC 18
Ergibtanweisung 76, 95

Ergibtzeichen 76
ESER 9, 20, 21

F

Feinprojekt 99, 102
Ferritkern 46, 48
-speicher 19, 54
FORTRAN 92
Funktion
Ausgabe- 50
Eingabe- 50
Rechen- 50
Speicher- 50
Steuer- 50

G

Generation 18, 19
Rechner- 18
Grobprojekt 99, 102

H

Halbleiterdiode 46, 47
Hysteresisschleife 48, 49

K

Kästchen
Alternativ- 78
Ausgabe- 78
Eingabe- 78
-methode 77
Operations- 77
Kanal 62, 69
Karte
Daten- 69
Loch- 17, 66, 67, 68
Programm- 69
Verbund- 69
Ziffern- 69
Konjunktion 36
Konstante 93, 94
Konvertieren 32
Kybernetik 25

L

- Leitlinienmethode 77
- Lesen 52, 53
- Liniendiagramm 101
- Lochfeld 68
- Lochkarte 17, 66, 67, 68
 - nleser 56
 - n-Lese-Stanz-Einheit 65
 - nmaschine 17
 - nstanzer 56
 - Normal- 69
- Lochkombination 68, 69
- Lochspalte 68
- Lochstreifen 65, 66, 67, 69
 - leser 56
 - stanzer 57
- Lochzeile 68
- Lochzone 68
 - Normal- 68
 - Über- 68

M

- Magnetband 66, 67, 69
 - speicher 58
- Maschine
 - Additions- 16, 23, 46
 - Automatische Rechen- 17
 - Einadreß- 55
 - Elektrische Schreib- 57
 - Elektronische Rechen- 16, 23
 - Fünfadreß- 55
 - Lochkarten- 17
 - nlesbarer Datenträger 66
 - n-orientierte Programmiersprache 75, 89
 - nprogramm 88
 - nsprache 75, 76, 83, 95
 - ntest 99, 102
 - Mechanische Rechen- 15, 16
 - Programmgesteuerte Rechen- 28
 - Schreib-nbefehl 60
 - Statistik- 17
 - Zweiadreß- 55
- Merkmal 89
- MINSK 22 19
- Modell 10, 26, 28
 - Graphisches — 26
 - Mathematisches — 26

- Physikalisches — 26
- Verbales — 26
- Vereinfachtes — 26, 28
- MOPS 89, 90
- Multiplikation
 - Logische — 37
 - im Dualsystem 34

N

- Negation 35, 36
- Netzplan 101

O

- Operation 94
 - Arithmetische — 85
 - Ausgabe- 86, 91
 - Eingabe- 86, 91
 - Interne — 90
 - Interne Transport- 86
 - Logische — 35, 46
 - Mnemonischer -scode 89
 - NICHT- 36
 - ODER- 38
 - scode 89, 90
 - shauptteil 86, 87
 - skästchen 77
 - steil 55
 - szusatzteil 86, 87
 - UND- 36
 - Vergleichs- 85
- Operator 93, 94
 - Arithmetischer — 94
 - Vergleichs- 95
- OPREMA 18
- Original 26

P

- PAP 28
- Peripherie 52, 56
 - gerät 56, 65
- PL/I 92, 95
- Position 69
 - Dekadisches -ssystem 28
 - ssystem 28

Programm 10, 26, 74
 Geradeaus- 81
 Linearer -ablaufplan 81
 Maschinenorientierte -iersprache 75, 89
 Maschinen- 88
 Nichtzyklischer -ablaufplan 81
 Objekt- 96
 Problemorientierte -iersprache 76, 92
 -ablaufplan 27, 28, 77, 102
 -anfang 78
 -ende 78
 -iersprache 75, 76
 -ierung 99
 -karte 69
 -steuerung 28
 -testung 96
 Quell- 96
 Übersetzungs- 75, 96
 Ursprungs- 96
 Verzweigter -ablaufplan 81
 Zyklischer -ablaufplan 81
 Prozeßrechner 10, 22
 Prüfer 67
 Puffer 62
 -speicher 62

R

R 10 21
 R 20 21
 R 21 20
 R 30 21
 R 40 21
 R 50 21
 R 300 19, 60

Rechner

Analog- 23
 Digital- 23, 46
 Elektromechanische — 23
 Elektronen- 23
 Klein- 58
 Mechanische — 15, 23
 Prozeß- 10, 22
 -generation 18
 Relais- 18
 Ziffern- 23

Relais 46, 47, 48
 -rechner 18
 Röhrentriode 46

S

Schaltalgebra 39
 Schaltbelegungstabelle 42
 Schaltelement 39, 46
 Elektromechanisches — 46
 Elektronisches — 8, 46
 Magnetisches — 46
 Mechanisches — 46
 Schaltnetz 41
 Schaltung 42
 NICHT- 41
 ODER- 40
 UND- 39
 Zusammengesetzte — 41
 Schlüssel 67, 72
 Alphanumerischer — 67, 72
 Dekadischer — 72
 Nichtdekadischer — 72
 Numerischer — 67, 72
 -system 70, 72
 -systematik 72
 -wort 93, 95
 Schnelldrucker 57, 65
 Schreiben 52, 53
 SER 2d 58
 Speicher 17, 52, 53
 Befehls- 17
 Daten- 17
 Externer — 57
 Ferritkern- 19, 54
 Haupt- 19, 62, 63
 Magnetband- 58
 Magnetplatten- 58
 Magnettrommel- 53
 Puffer- 62
 -belegungsplan 83, 87, 88
 -funktion 50
 — mit direktem Zugriff 57
 Sequentieller — 57, 86
 Trommel- 19, 53
 Sperrichtung 47, 48
 Spur 69
 Sprache
 Assembler- 75, 76, 89

Assembler- MOPS 89
Maschinen- 75, 76, 83, 95
Maschinenorientierte Programmier-
75, 89
Problemorientierte Programmier- 76,
92
Programmier- 75, 76, 89, 92
Sprachübersetzer 96
Sprung 60
Null- 60
-befehl 60, 86, 87, 90
Unbedingter — 60
Startsymbol 79
Steuerung 26
Programm- 28
Stoppsymbol 80
Studie 99, 100
Subtraktion im Dualsystem 33
System
Addition im Dual- 33
Dekadisches Positions- 28
Dezimal- 29
Dual- 30
Multiplikation im Dual- 34
Positions- 28
Schlüssel- 70, 72
Subtraktion im Dual- 33
Zahlen- 29

T

Tetrade 32
Transistor 46, 47, 48
Trockentest 102

U

URAL-Serie 19

Bildnachweis

VEB Kombinat Robotron DDR Fotos auf
den Seiten 20, 21, 22, 52, 59

V

Variablenname 93, 95
Vereinbarung 89
Verneinung 35, 36
Verschlüsseln 67
Verschlüsselung 32
Dezimalduale — 32
Mnemonische — 89
Normal- 32
Rein-duale — 32
Ziffernweise — 32

W

Wahrheitswert 35
-etabelle 36, 42
Werk
Addier- 65
Rechen- 52, 62
Steuer- 51, 63

Z

Zeichen
Binäres — 30, 32
Ergibt- 76
Trenn- 93
ZRA 1 19
Zugriffszeit 19, 53