
Der Personalcomputer 1715

Der Personalcomputer 1715



Verlag Die Wirtschaft Berlin

Herausgegeben vom VEB Kombinat Robotron

Autorenkollektiv unter Leitung von Prof. Dr.-Ing. Rolf Zeth:

U. Arnold, D. Bade, Dr.-Ing. M. Brode, B. Elpelt,
K. Fahr, K.-H. Fuchs, C. Görner, E. Hase, Dr. R. Helling,
A. Henning, Dr.-Ing. R. Hesse, U. Klein, Dr.-Ing. K. Mäder,
Dr.-Ing. H. Mestenhauser, B. Müller, P. Nicolaus,
R. Nürnberger, P. Polster, S. Radestock, M. Schlegel,
U. Schulze, E. Schwarz, H.-D. Sporbert, B. Steinicke,
K. Stumpp, Prof. Dr.-Ing. R. Zeth, V. Ziller

Lektor:

Gerhard Ziese

2. Auflage

Redaktionsschluß: 30. 9.1986

Der Personalcomputer 1715 [siebzehnhundertfünfzehn]/
hrsg. vom Kombinat Robotron. – 2. Aufl. –
Berlin: Verl. Die Wirtschaft, 1988. –
224 S. : 11 Fotos, 40 Abb., 43 Tab.

ISBN 3-349-00231-5

© Verlag Die Wirtschaft 1988

Am Friedrichshain 22, Berlin, 1055

Lizenz-Nr. 122; Druckgenehmigungs-Nr. 195/132/88

LSV 0395

Einbandgestaltung: Marlies Hawemann

Typografie: Verlag Die Wirtschaft

Printed in the German Democratic Republic

Gesamtherstellung: (140) Druckerei Neues Deutschland, Berlin

Bestellnummer: 676 061 1

01400

Vorwort

Mit der Produktion von Personal- und Arbeitsplatzcomputern leistet der VEB Kombinat Robotron einen wichtigen Beitrag zur Verwirklichung der Beschlüsse des XI. Parteitages der SED. Die Realisierung der Verpflichtungen, Zehntausende Rechner zusätzlich zum Plan zur Verfügung zu stellen, wird den Prozeß der Beschleunigung des wissenschaftlich-technischen Fortschritts zur Intensivierung volkswirtschaftlicher Prozesse und zur Steigerung der Arbeitsproduktivität wirksam unterstützen. Es werden damit die gerätetechnischen Voraussetzungen geschaffen, um durch den Aufbau und die Nutzung von rechnergestützten Arbeitsplätzen entscheidende Rationalisierungsschübe bei Produktions-, Konstruktions- und Verwaltungsprozessen zu erreichen.

Die Schaffung der gerätetechnischen Basis ist jedoch nur eine Seite dieser Entwicklung. Die intensive Nutzung der Gerätetechnik erfordert u. a. auf Gebieten der Einsatzvorbereitung, der Modellierung arbeitsplatzbezogener Prozesse, der Ausnutzung bereits vorhandener leistungsstarker Standardsoftware, des Entwurfs und der Programmierung von Software für spezielle Anwendungsfälle erhebliche Anstrengungen, um das Entwicklungstempo auf diesen Gebieten dem Tempo der Gerätebereitstellung anzupassen.

Ein Beitrag dazu ist die Herausgabe der hier vorgelegten Broschüre, die im Verlag Die Wirtschaft zusammen mit weiteren Schriften wie Datenbanken mit Personalcomputern u. a. erscheint. Diese Broschüren sollen es allen potentiellen Nutzern von Personal- und Arbeitsplatzcomputern ermöglichen, den in ihrem Arbeitsbereich bevorstehenden Einsatz der modernen Informationsverarbeitungstechnik aktiv zu unterstützen. Insbesondere können die Broschüren die vielfältigen Qualifizierungsmaßnahmen in Betrieben und Einrichtungen fördern. Jeder Werktätige, dessen Arbeitsplatz früher oder später vom Einsatz eines Computers beeinflusst wird, kann mit Gewinn zu der jeweils in Frage kommenden Broschüre greifen. Diesem Anliegen entsprechend sind die Broschüren allgemeinverständlich abgefaßt und setzen zu ihrem Verständnis keine speziellen Kenntnisse voraus.

Die hier vorliegende Broschüre über den Personalcomputer 1715 soll den Leser erstmalig mit dem Gerät und seiner Software bekannt machen. Schwerpunkt der Dokumentation sind die technische Beschreibung des Gerätesystems sowie eine Übersicht über seine Einsatzmöglichkeiten.

Neben dem Leser, der Erstinformationen zum Personalcomputer 1715 sucht, wird auch der fortgeschrittene Nutzer eines Personalcomputers wichtige Informationen finden. Diese sind – meist technisches Spezialwissen – in Tabellenform oder in Übersichten komprimiert zusammengestellt; dadurch wird die Lesbarkeit der Broschüre für den Anfänger verbessert: er kann entsprechende Zusammenstellungen beim ersten Lesen leicht übergehen.

Der Inhalt der vorliegenden Broschüre entspricht dem technischen Stand von 1986. Änderungen im Sinne des technischen Fortschritts bleiben dem Hersteller vorbehalten.

Herausgeber und Verlag danken allen Kolleginnen und Kollegen, die durch unermüdlichen Einsatz das Entstehen dieser Broschüre ermöglichten. Stellvertretend für alle anderen sei namentlich den Kollegen Dipl. oec. H.-D. Sportbert für seinen persönlichen Einsatz bei der Gesamtreaktion und Dr.-Ing. sc. nat. Kuno Schmidt für die Durchsicht des Manuskripts und seine Hinweise für die Erstellung der Endfassung ausdrücklich gedankt.

VEB Kombinat Robotron
Verlag Die Wirtschaft

Inhaltsverzeichnis

Einführung	10
1. Die Hardware des PC 1715	13
1.1. Systemeinheit	15
1.1.1. Zentrale Recheneinheit	16
1.1.2. Ansteuereinheit für Diskettenlaufwerke	29
1.1.3. Minifolienspeicher	31
1.1.4. Stromversorgung	32
1.2. Die Bildschirmtypen K 7221 und K 7222	33
1.3. Tastatur	35
1.4. Peripherieerweiterungen	40
1.4.1. Interfaceerweiterungssteckeinheit für 2 × V.24-Schnittstelle	40
1.4.2. Interfaceerweiterungssteckeinheit für 2 × IFSS-Schnittstelle	44
1.4.3. Externe Folienspeichereinheit für 8-Zoll-Disketten	48
1.4.4. Externe Folienspeichereinheit für 5,25-Zoll-Disketten	50
1.5. Die Drucker des PC 1715	50
1.5.1. Die Matrix-Nadeldrucker K 6313 und K 6314	54
1.5.2. Der Typenraddrucker 1152/Typ 257	59
1.5.3. Der Schalterdrucker K 6316	64
1.6. Technisches Datenblatt zum PC 1715	67
1.7. Erweiterungen der Hardwarekonfiguration	68
2. Die Betriebssysteme des PC 1715	69
2.1. Das Betriebssystem SCP für den PC 1715	69
2.1.1. Start des Betriebssystems	73
2.1.2. Aufbau des Datenträgers und Diskettenformate	74
2.1.3. Die CCP-Kommandos	77
2.1.4. Beschreibung der BDOS-Funktionen	85
2.1.5. BDOS-Fehlermeldungen	90
2.1.6. Beschreibung der BIOS-Funktionen	91

2.1.7.	Der Verständigungsbereich	96
2.1.8.	Die Prinzipien und Möglichkeiten der Installation des Betriebssystems SCP	97
2.2.	Das Betriebssystem BROS	98
2.3.	Das Betriebssystem JAMB	101
2.4.	Vergleichende Betrachtungen zu den Betriebssystemen des PC 1715	102
3.	Datenübertragung und Terminalnachbildungen mit dem PC 1715 unter dem Betriebssystem SCP	104
3.1.	Der serielle Ein-/Ausgabetreiber SERIO	104
3.2.	Das Kommunikationsprogramm TLC	107
3.3.	EM62 – Emulation des Terminals EC 8562/64	113
4.	Die Entwicklung von Systemprogrammkomponenten mit dem PC 1715	118
4.1.	Der Assembler ASM	121
4.2.	Der Programmverbinder LINK	124
4.3.	Das Bibliotheksverwaltungsprogramm LIB	124
4.4.	Der Debugger DU	125
5.	Die Programmiersprachen im Betriebssystem SCP	127
5.1.	BASIC-Interpreter und BASIC-Compiler	127
5.2.	Die Programmiersprache PASCAL-MP	134
5.3.	Die Programmiersprache FORTRAN	141
6.	Standardsoftware im Betriebssystem SCP	148
6.1.	Das relationale Datenbanksystem REDABAS	152
6.2.	Das Kalkulationsprogramm KP (SCP)	158
6.3.	Der Textprozessor TP	164
6.4.	Grafische Darstellungen mit dem Programm PCGRAF	171
7.	Übersicht über Einsatzbereiche des PC 1715	180
7.1.	Einsatzbereiche und Einsatzfälle	181
7.1.1.	Einsatz des PC 1715 für die Erstellung von Binnenmarktrechnungen	182
7.1.2.	Einsatz des PC 1715 für die Erstellung von Exportfakturen	182
7.1.3.	Das Programm „SALDOKONTO“	183
7.1.4.	Lohn- und Gehaltsrechnung mit dem PC 1715	183
7.1.5.	Arbeitsplatzbezogene Information und Dokumentation mit Hilfe des PC 1715	184

7.1.6.	Einsatz des PC 1715 für die Auftragsbearbeitung in der Kfz-Instandsetzung	186
7.1.7.	Einsatz des PC 1715 in den CENTRUM-Warenhäusern für den Warenabzug der Verkaufsabteilungen aus den Außenlagern . . .	186
7.1.8.	Einsatz eines PC 1715 im Betrieb Projektierung des WBK Erfurt	187
7.1.9.	Einsatz des PC 1715 in der Forstwirtschaft der DDR	189
7.2	Fertigungslenkungs- und Informationssystem auf Basis PC 1715	191
7.3.	Einsatz eines PC 1715 zur Unterstützung der Konstrukteurarbeit	196
7.4.	Rechnergestützter Arbeitsplatz für den Augenoptiker auf der Basis PC 1715	200
7.5	Einsatz der Nadeldrucker K 6313/K 6314 für Grafikzwecke am PC 1715	202
7.5.1.	Darstellung von Kurvenverläufen	205
7.5.2.	Darstellung von Stabdiagrammen	207
7.5.3.	Darstellung von Kreisdiagrammen	208
8.	Serviceleistungen für den PC 1715	210
8.1.	Montage, Inbetriebnahme, Prüfung	211
8.2.	Wartung	212
8.3.	Instandsetzungsmaßnahmen am Standort des PC 1715	213
	Weiterführende Literatur	215
	Sachwortverzeichnis	218

Einführung

Grundsätzlich versteht man unter dem Begriff „Personalcomputer“ einen Rechner, der als „persönliches“ Arbeitsmittel von jedermann benutzt werden kann. Bestehend aus Hardware (Gerätetechnik) und Software (Programme) ist er ein Rechner, der im Unterschied zu zentralen Datenverarbeitungsanlagen als komplettes System direkt am Arbeitsplatz des Nutzers aufgestellt werden kann. In seinen Funktionen ist er mit einer großen Datenverarbeitungsanlage vergleichbar, lediglich in seiner Leistungsfähigkeit ist er dieser unterlegen.

Personalcomputer sind im allgemeinen Einzelplatz-Computer; sie lassen sich aber ebenso als Terminal oder Endgerät mit einem Großrechner, in einer Punkt-zu-Punkt-Verbindung mit einem anderen Mikrorechner oder im Einsatz als Endgerät in öffentlichen Netzen ringförmig bzw. stern- oder linienförmig zu lokalen Netzen (LAN) zusammenschalten. Über ein solches Netzwerk können die Nutzer miteinander kommunizieren. Vor allem aber wird es dadurch möglich, teure Peripherie, wie Plotter, Drucker, externe Speicher u. a., durch mehrere Anwender zu nutzen.

Entscheidende Bedeutung für die Nutzung und weitere Entwicklung der Personalcomputer hat die Realisierung der Software. Hierbei nimmt das Betriebssystem als Schnittstelle zwischen Anwendersoftware und Hardware eine zentrale Stellung ein. Unter einer Vielzahl von Betriebssystemen haben sich heute „Standardsysteme“ herausgebildet. Für 8-Bit-Mikrocomputer steht das seit 1975 entwickelte Betriebssystem CP/M an erster Stelle. CP/M ist ein eingetragenes Warenzeichen der Fa. Digital Research, Corp./USA.

Mit der Nutzung von „Standardsystemen“ kann sich der Nutzer eines Personalcomputers das international herausgebildete breite Feld der weiteren Systemsoftware sowie der Standard-Anwender-Software erschließen.

In den sozialistischen Ländern ist eine schnelle Entwicklung von Personalcomputern und ein vielseitiges Angebot an Soft- und Hardwareleistungen zu verzeichnen. Durch den VEB Robotron Büromaschinenwerk „Ernst Thälmann“ Sömmerda wurde der Personalcomputer 1715 entwickelt und im

Jahre 1985 in die Produktion übergeleitet. Dieses Gerät wird in der vorliegenden Broschüre in seinem Hard- und Softwareaufbau beschrieben.

Entsprechend der entscheidenden Bedeutung der Software für die Nutzung eines Personalcomputers wurden mit Beginn der Entwicklung des PC 1715 folgende Zielrichtungen gestellt:

- Gewährleistung der Softwarekompatibilität zum Vorläuferprodukt, dem Bürocomputer A 5110 mit dem Betriebssystem BROS;
- Erreichung der Kompatibilität zum Programmiersystem JAMB der sowjetischen Buchungsmaschinen „EBT Newa 501“ und „EBM Iskra 555“;
- Realisierung eines Betriebssystems, welches die Anwenderschnittstelle von CP/M besitzt.

Dementsprechend wurden mit der Softwarekonzeption des PC 1715 folgende Grundsätze verwirklicht:

- Realisierung eines Rechners mit Urladerkonzept und maximal 64 KByte verfügbarem RAM-Speicher, der durch die Ladbarkeit der Systemsoftware den unterschiedlichen Anforderungen der zu implementierenden Betriebssysteme gerecht wird;
- Gewährleistung der Generierbarkeit bzw. Installierbarkeit der Betriebssysteme zur Anpassung an verschiedene Gerätekonfigurationen und Diskettenformate;
- maximale Vereinheitlichung der physischen Gerätetreiber, die eine quasi-standardisierte Software-Zwischenschicht für die Ein-/Ausgabe bilden, um den Entwicklungs- und Implementationsaufwand für die verschiedenen Betriebssysteme zu reduzieren.

Auf Grund seiner Leistungsfähigkeit und seines modularen Gerätekonzeptes wird der PC 1715 vielfältigen Anwenderforderungen gerecht. In einer großen Zahl sehr unterschiedlicher Einsatzfälle hat der PC 1715 bereits seine Einsatzstärke unter Beweis stellen können. Ein knapper Überblick über bisherige Erfahrungen und mögliche Einsatzbereiche wird mit dieser Broschüre gegeben (siehe vor allem das 7. Kapitel). Dieser Überblick soll in erster Linie Erstnutzer des PC 1715 anregen, die Lösung von Aufgaben im eigenen Wirkungsbereich mit Hilfe dieser Gerätetechnik und der zur Verfügung stehenden Software schöpferisch in Angriff zu nehmen.

Entsprechend seinen Anwendungsmöglichkeiten für

- wissenschaftlich-technische Berechnungen in Forschung, Entwicklung und Technologie,
- ökonomische Berechnungen in Buchhaltung, Lagerverwaltung, Planung und Statistik und vielen anderen Bereichen,
- Textverarbeitung in jedem Büro,

- on-line- und off-line-Datenerfassung,
- CAD/CAM-Lösungen

findet der PC 1715 heute seinen Einsatz in einer Vielzahl von Kombinat und Betrieben, in wissenschaftlich-technischen sowie Bildungseinrichtungen, in der Landwirtschaft, im Transportwesen, bei der Post, in Banken und Sparkassen, im Bereich Handel und Versorgung, in zahlreichen staatlichen und gesellschaftlichen Einrichtungen, nicht zuletzt im Gesundheitswesen und weiteren Bereichen. Das Einsatzfeld des PC 1715 erscheint damit praktisch unbegrenzt.

1. Die Hardware des PC 1715

Unter Hardware versteht man allgemein die gerätetechnischen Gegebenheiten eines Computersystems. Die Beschreibung der Hardware des Personalcomputers 1715 (PC 1715) gibt einen Überblick über seine Baugruppen und peripheren Geräte. Dabei wird vor allem aus nutzerseitiger Sicht besonderes Augenmerk auf die Schnittstellen gelegt, und es werden ausgewählte Komponenten behandelt.

In Abbildung 1.1 ist das Computersystem PC 1715 aus struktureller Sicht dargestellt.

Wie aus Abbildung 1.1 ersichtlich ist, besteht der PC 1715 aus einem System modularer Baugruppen,

- der Systemeinheit mit den Komponenten
 - Zentrale Recheneinheit (ZRE)
 - Ansteuereinheit für Diskettenlaufwerke (FD, Abk. von Floppy-Disk)
 - 2 Diskettenlaufwerke
 - Stromversorgung,
- dem Bildschirm (Monitor) und
- der Tastatur.

Peripherieerweiterungen können wahlweise aus den folgenden Komponenten bestehen:

- Interfaceerweiterungssteckeinheit für $2 \times V. 24$
- Interfaceerweiterungssteckeinheit für $2 \times IFSS$
- externe Folienspeichereinheit für 8-Zoll-Laufwerke
- externe Folienspeichereinheit für 5,25-Zoll-Laufwerke
- Systemdrucker (wahlweise Typenrad- oder Nadeldrucker mit entsprechender Formulartechnik).

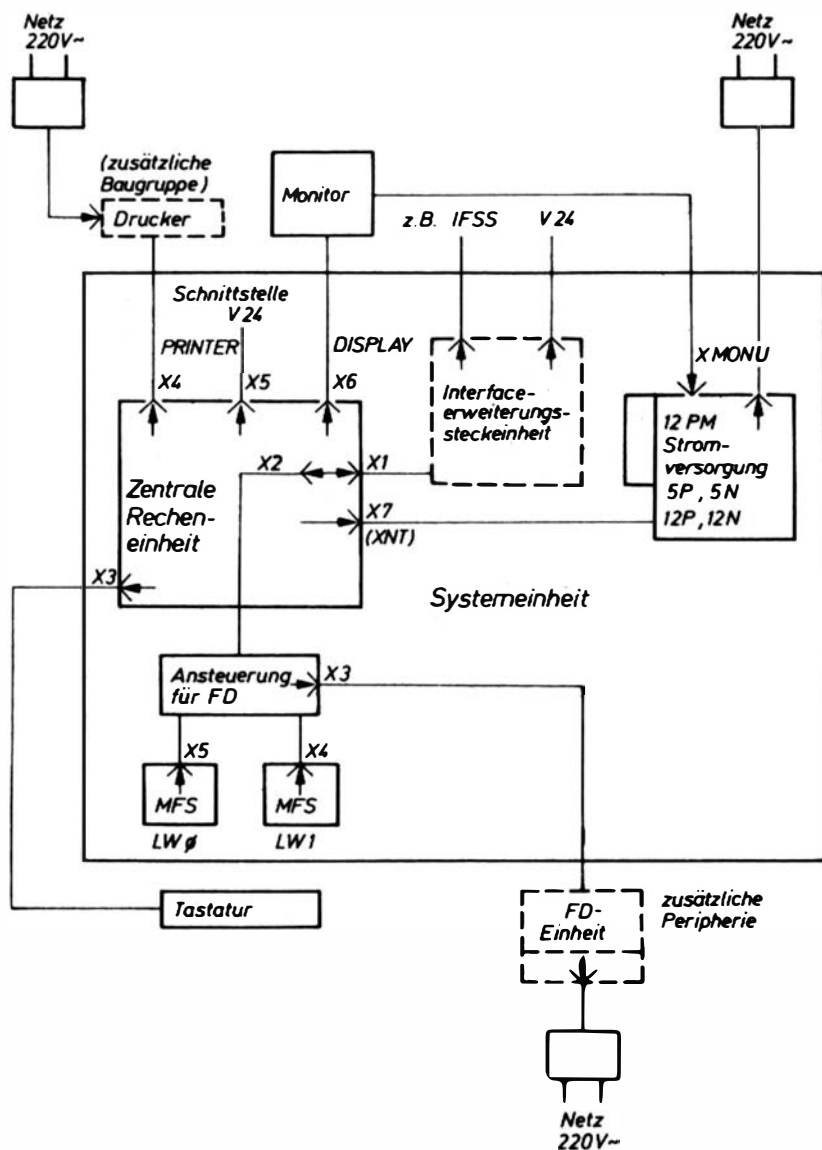


Abbildung 1.1
Modulare Baugruppen des PC 1715 (Blockschaltbild)

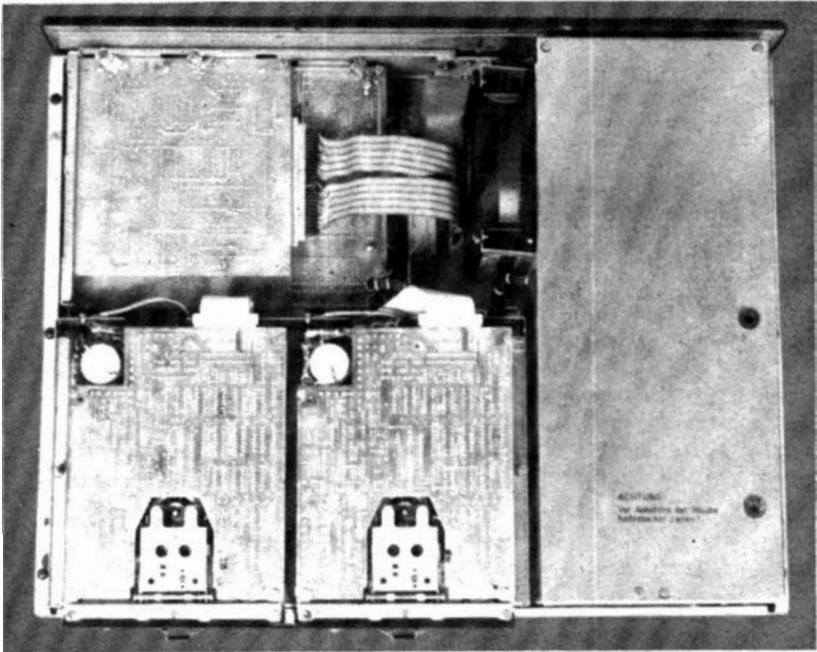


Abbildung 1.2
Systemeinheit des Rechners

1.1. Systemeinheit

Die Systemeinheit (Abb. 1.2) ist die Hauptbaugruppe des Rechners.

Abmessungen:

Breite: 500 mm

Tiefe: 400 mm

Höhe: 130 mm (bei MFS 1.2 und bei MFS 1.6)
156 mm (bei MFS 1.4)

Gewicht: 12,6 kg (bei MFS 1.2 oder MFS 1.6)

1.1.1. Zentrale Recheneinheit

Die Zentrale Recheneinheit (Abb. 1.3) ist auf der Systembasis des Mikroprozessors U 880 aufgebaut.

Abmessungen:

Breite: 240 mm

Tiefe: 360 mm

Höhe: 30 mm

Die Zentrale Recheneinheit besteht aus den Komponenten

- Mikroprozessor U 880
- ROM-Anfangslader
- RAM-Speicher
- Systembus
- Bildschirmsteuerung
- Integrierte Ein-/Ausgabe-Kanäle (V. 24, Printerausgang, Tastatur).

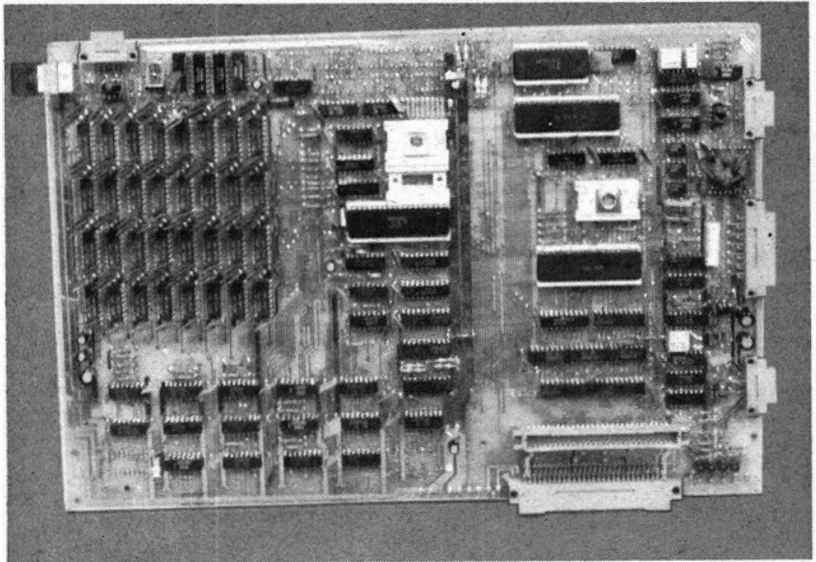


Abbildung 1.3
Zentrale Recheneinheit

Übersicht A

Bedeutung und Funktion der Bestandteile des Systembus

- AB0–AB15:** Der *Adreßbus* liefert die Adressen für Speicher-Daten-Transporte (bis zu 64 KByte) und Datentransporte für E/A-Geräte. Die E/A-Adressierung benutzt die unteren 8-Adressen-Bit, um eine direkte Anwahl von 256 Eingangs- oder 256 Ausgangskanälen zu ermöglichen. Während der Auffrischzeit enthalten die unteren 7 Bit die gültige Adresse für die Speicherauffrischung. Die von der ZRE (CPU) erzeugten Adressen-Bit A0 bis A15 werden über Treiberschaltkreise zum Adreßbus AB0 bis AB15 verstärkt. Während der Datenholzeit (RFSH-Zeit) vom Bildwiederholtspeicher werden die Adreßbus-Treiber in den hochohmigen Zustand versetzt.
- DB0–DB7:** DB0 bis DB7 bilden den *bidirektionalen 8-Bit-Datenbus*. Der Datenbus dient den Datentransporten von oder zum Speicher und von oder zu E/A-Geräten. Der Datenbus DB0 bis DB7 steht über einen Treiberschaltkreis mit dem Datenbus D0 bis D7 der ZRE in Verbindung. Während der Datenholzeit (RFSH-Zeit) vom Bildwiederholtspeicher wird der Datenbustreiber in den hochohmigen Zustand versetzt.
- /M1:** (*Maschinenzyklus 1*). /M1 ist low-aktiv und zeigt an, daß sich der laufende Maschinenzyklus der Befehlsabarbeitung im Zustand „Aufruf des Operationscodes“ befindet.
/M1 wird während der Ausführung eines 2 Byte langen Operationscodes beim Aufruf des OP-Codes eines jeden Byte erzeugt. /M1 tritt auch zusammen mit /IORQ auf, um einen Interrupt-Aannahmezyklus anzuzeigen.
- /MREQ:** *Speicheranforderung* (memory request). Das low-aktive Signal zeigt an, daß der Adreßbus eine gültige Adresse für eine Speicher-Lese- oder -Schreiboperation enthält.
- /IORQ:** *Ein-Ausgabeanforderung* (input/output request). /IORQ ist low-aktiv und zeigt an, daß die untere Hälfte des Adreßbusses eine gültige E/A-Adresse für eine E/A-Lese- oder -Schreiboperation enthält. Ein /IORQ-Signal wird auch während der /M1-Zeit aktiv generiert, um bei Interrupt-Aufnahme anzuzeigen, daß ein entsprechender Interruptvektor auf den Datenbus gelegt werden kann. Interrupt-Aufnahme-Operationen treten während der /M1-Zeit auf, wogegen E/A-Operationen niemals während der /M1-Zeit durchgeführt werden.
- /RD:** *Lesen* (read). /RD ist low-aktiv und zeigt an, daß die ZRE Daten vom Speicher oder von einem E/A-Gerät lesen will.
- /WR:** *Schreiben* (write). /WR ist low-aktiv und zeigt an, daß der Datenbus gültige Daten enthält, die im adressierten Speicherplatz oder E/A-Gerät gespeichert werden sollen.
- /RFSH:** *Speicherauffrischen* (refresh). Das low-aktive Signal zeigt an, daß die unteren 7 Bit des Adreßbusses eine Auffrischadresse für den dynamischen Speicher enthalten.
- /HALT:** *Halt-Zustand* (HALT-state). Befindet sich dieser Ausgang im Zustand low, zeigt die ZRE an, daß sie einen Software-HALT-Befehl ausführt und nun entweder einen nicht maskierten oder einen maskierbaren In-

errupt erwartet. Nur dadurch kann die ZRE diese Operation wieder verlassen. Die ZRE führt in diesem Zustand NOP-Befehle aus, um die Auffrischung der Speicher durchzuführen.

- /WAIT:** *Warten* (WAIT). Low-aktiver Eingang an der ZRE. /WAIT zeigt an, daß der adressierte Speicherplatz oder das E/A-Gerät noch nicht für einen Datentransport bereit sind. Solange dieses Signal aktiv ist, generiert die ZRE WAIT-Zustände.
Mit Hilfe dieses Signals können besonders E/A-Geräte mit abweichender Geschwindigkeit mit der ZRE synchronisiert werden. Dabei ist zu beachten, daß durch die zusätzlichen WAIT-Zyklen das Auffrischen des Speichers nicht beeinträchtigt wird.
- /INT:** *Interrupt-Aufforderung* (interrupt-request) Eingang, low-aktiv. Das Interrupt-Anforderungssignal wird durch ein E/A-Gerät erzeugt. Eine Anforderung wird am Ende des laufenden Befehls beachtet, wenn das Interrupt-Aufnahme-Flip-Flop, das durch die interne Software gesteuert wird, bereit ist und wenn das /BUSRQ-Signal nicht aktiv ist. Nimmt die ZRE den Interrupt an, so wird das Interrupt-Aufnahmesignal bei Beginn des nächsten Befehlszyklus (/IORQ während M1) ausgesandt.
- /NMI:** *Nicht maskierter Interrupt* (non maskable interrupt) Eingang, triggert auf Low-Flanke. Triggerflanke aktiviert ein internes NMI-Flip-Flop. Die Funktion NMI hat eine höhere Priorität als das INT und wird am Ende des anliegenden Befehls getestet, unabhängig von der Lage des Interrupt-Aufnahme-Flip-Flop. /NMI zwingt die ZRE automatisch zu einem RESTART.
Der Befehlszähler wird automatisch im Kellerspeicher gerettet, so daß der Anwender zu dem Programm zurückkehren kann, das unterbrochen wurde. Es muß beachtet werden, daß zusätzliche WAIT-Zyklen das Ende des anliegenden Befehls verhindern und ein /BUSRQ ein /NMI überschreibt.
- /BUSRQ:** *Busanforderung* (bus request). Eingang low-aktiv. Das Busanforderungssignal wird benutzt, um die ZRE aufzufordern, den Adreß- und Datenbus und die Drei-Zustands-Ausgangssignale in den hochohmigen Zustand zu bringen. Das erfolgt, sobald der laufende Maschinenzklus der ZRE abgeschlossen ist.
- /BUSAK:** *Busbestätigung* (bus acknowledge). Ausgang low-aktiv. Die Busbestätigung wird benutzt, um die Bustreiberschaltkreise im Bedarfsfall (Test) in den hochohmigen Zustand zu bringen.
- /RESET:** *Eingang*, low-aktiv, /RESET stellt den Befehlszähler auf Null und weist der ZRE die Anfangswerte zu.

Der *Mikroprozessor U 880*, im weiteren CPU genannt, hat die Aufgabe, sämtliche Systeminformationen zu erzeugen, auszuwerten oder im Dialogbetrieb die peripheren Baugruppen zu steuern oder deren Informationen auszuwerten.

Die CPU ist zur Synchronisation mit dem Systemtakt verbunden. Die Taktzeugung besteht aus dem Quarzgenerator, der mit einer Frequenz von

$f = 9,832 \text{ MHz}$ arbeitet. Der Systemtakt schwingt mit einer Frequenz von $2,458 \text{ MHz}$ und ist somit zur Erzeugung der gewünschten Übertragungsraten für die Versorgung von seriellen Schnittstellen geeignet.

Nach dem Einschalten oder dem Betätigen der RESET-Taste wird das System in den Zustand „Anfangsladen“ versetzt: der *ROM-Anfangslader* (ein in einem PROM (ROM) enthaltenes Mikroprogramm) transportiert ein auf einer Diskette enthaltenes Maschinenprogramm (Betriebssystem) in den RAM-Speicher.

Der *RAM-Speicher* (Kapazität 64 KByte) dient als Arbeitsspeicher sowie als Speicher für das Betriebssystem und für Nutzerprogramme. Er ist als dynamischer Speicher mit 16-KBit-Schaltkreisen aufgebaut.

Verwendbar sind u. a. folgende Schaltkreistypen: U 256, K 565 RU3A, K 565 RU3G, K 565 RU6.

Der *Systembus* besteht aus den Leitungsbündeln Adreßbus AB0–AB15 und Datenbus DB0–DB7 sowie den Steuersignalleitungen /M1, /MREQ, /IORQ, /RD, /WR, /RFSH, /WAIT, /INT, /RESET, /BUSRQ, /BUSAK, TAKT, TAKT-X1, TAKT-X2.

Bedeutung und Funktion dieser Bestandteile des Systembusses ergeben sich aus der Übersicht A.

Eine Übersicht über die I/O-Toradressen gibt die Tabelle 1.1.

Die *Bildschirmsteuerung* hat die Aufgabe, in einem Teilbereich des RAM-Speichers befindliche Daten auf dem Monitor K 7221 (MON1) oder K 7222 (MON2) abzubilden. Die Lage des Bildwiederholerspeichers im Adreßraum wird durch das jeweilige Betriebssystem bestimmt. Die Anfangsadresse ist in einem 1024-Zeichenformat (K 7221) in 1-KByte-Schritten bzw. in einem 1920-Zeichenformat (K 7222) in 2-KByte-Schritten modifiziert.

Den Kern der Bildschirmsteuerung bildet der CRT-Controller KR 580 WG75 (Intel 8275). Da dieser Schaltkreis in seiner Anwendung ein relativ neues Bauelement darstellt, das gleichzeitig die Leistungsfähigkeit der Gesamtschaltung charakterisiert, soll etwas detaillierter auf die technischen Möglichkeiten eingegangen werden.

Der CRT-Controller KR 580 WG 75 besteht aus folgenden wesentlichen Funktionsgruppen: Zeichenzähler, Linienzähler, Zeilenzähler, Raster- und Videosteuerung, Zeilenpuffer, FIFO-Speicher.

Zeichenzähler

wird betrieben mit dem Zeichentakt CCLK;

programmierbar auf 1–80 Zeichen/Zeile;

bestimmt auch die Länge des horizontalen Strahlrücklauf-Intervalles (programmierbar 2–32 Zeichentakte).

Tabelle 1.1
Toradressenübersicht

Signal	Zuordnung	Codierungen	
/PIOCSO	FD-Datentransport	Steuerregister: Kanal A: 01H Kanal B: 03H	Datenregister: Kanal A: 00H Kanal B: 02H
/PIOCS1	FD-Steuersignale	Steuerregister: Kanal A: 05H Kanal B: 07H	Datenregister: Kanal A: 04H Kanal B: 06H
/KRFD	FD-Steuersignale	Datenwort 20H (21H, 22H, 23H)	
/CTCCSO	CTC (TAKT für SIO)	Kanal 00: 08H Kanal 01: 09H	Kanal 02: 0AH Kanal 03: 0BH
/SIOCSO	SIO Drucker, Tastatur V. 24	Steuerregister: Kanal A: 0EH Kanal B: 0FH	Datenregister: Kanal A: 0CH Kanal B: 0DH
/CTCCS1	CTC (Takt für SIO)	Steuerregister: Kanal 00: 10H Kanal 01: 11H	Datenregister: Kanal 02: 12H Kanal 03: 13H
/SIOCS1	SIO	Steuerregister: Kanal A: 16H Kanal B: 17H	Datenregister: Kanal A: 14H Kanal B: 15H
/CRTCS	Bildschirmcontroller CRT	Kommandoregister: Parameterregister:	19H, 1BH 18H, 1AH
/MEMCSO	PROM aktiv	24H, (25H, 26H, 27H)	
/MEMCS1	PROM inaktiv	28H, (29H, 2AH, 2BH)	
/LT107CS	1. Abfrage Ltg.107	2DH, (2FH)	
	2. Setzen Ltg. 107	2CH, (2EH)	
/LT111CS	Setzen Ltg. 111 ZRE	30H, (31H, 32H, 33H)	
/BWSCS	Bitwiederholungspeicher- anfangsadresse und Umschaltung Zeich- generator (PROM)	34H, (35H, 36H, 37H)	

Linienzähler

programmierbar auf 1–16 Linien/Zeile;
seine Ausgänge LC0–LC3 werden benutzt, um den externen Zeichengenerator zu adressieren.

Zeilenzähler

programmierbar auf 1–64 Zeilen/Bild;
bestimmt außerdem die auf 1–4 Zeilen programmierbare Länge des vertikalen Strahlrücklauf-Intervalles.

Raster- und Videosteuerung

Die Rastersteuerung erzeugt die Ausgangssignale HRTC (horizontaler Strahlrücklauf) und VRTC (vertikaler Strahlrücklauf).

Die Videosteuerung erzeugt die Ausgangssignale

HGLT: Einschaltung einer 2. Helligkeitsstufe

RVV: Inversdarstellung

LTEN: Strahleinschaltung (z.B. Unterstreichen)

VSP: Videounterdrückung (z.B. bei Strahlrücklauf)

GPA0–1: Ausgänge für allgemeine Verwendungszwecke.

Zeilenpuffer

2 Zeilenpuffer für je 80 Zeichen zu je 8 Bit. Während einer die Zeichen für eine Zeile über die Ausgänge CC0–6 bereitstellt, gesteuert durch den Zeichentakt CCLK, wird der andere im DMA-Betrieb aus dem RAM-Speicher neu gefüllt.

FIFO-Speicher

Jedem Zeilenpuffer ist als Erweiterungsmöglichkeit ein FIFO-Speicher (First In – First Out) für 16 Zeichen zu je 7 Bit zugeordnet, der im transparenten Attributmodus das auf ein Feldattribut (Steuerzeichen, das z.B. Blinken, Inversdarstellung, Unterstreichen ein- oder ausschaltet) folgende darstellbare Zeichen aufnimmt. Dadurch ist es möglich, pro Zeile außer maximal 80 darstellbaren Zeichen auch 16 Steuerzeichen einzulesen.

Die *Programmierung des CRT-Controllers* erfolgt durch insgesamt 8 Befehle (teilweise mit Parametern), von denen beim PC 1715 nur die in Tabelle 1.2 angegeben verwendet werden. Zwischen Befehlen und Parametern wird durch den Eingang A0 (niedrigstes Adreßbit AB0) unterschieden:

A0 = 1: Befehl

A0 = 0: Parameter

Tabelle 1.2 a
CRT-Controller-Befehle beim PC 1715

Befehl	Zahl der Parameter
Reset	4
Start Display	0
Stop Display	0
Load Cursor	2

Tabelle 1.2 b
Aufbau der CRT-Controller-Befehle

Befehl/Parameter	A0	Datenbus
Resetbefehl	1	0 0 0 0 0 0 0 0
Parameter 1	0	S H H H H H H H
Parameter 2	0	V V R R R R R R
Parameter 3	0	U U U U L L L L
Parameter 4	0	M F C C Z Z Z Z
Start Display	1	0 0 1 S S S B B
Stop Display	1	0 1 0 0 0 0 0 0
Load Cursor	1	1 0 0 0 0 0 0 0
Zeichen-Nr.	0	Zeichen-Pos. in Zeile
Zeilen-Nr.	0	Zeilen-Nr.

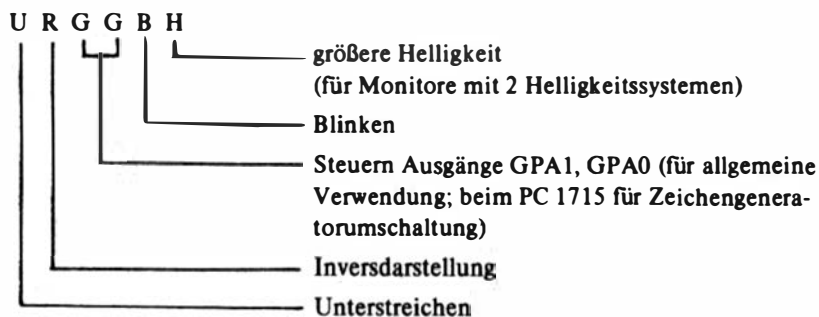
Die Parameter in Tabelle 1.2b haben folgende Bedeutung:

- S = 0 normale Zeilen
 S = 1 Zeilen mit Zwischenraum
 HHHHHH Anzahl der Zeichen pro Zeile (1–80)
 VV Länge des vertikalen Strahlrücklauf-Intervalles (1–4 Zeilen)
 RRRRRR Anzahl Zeilen/Bild (1–64)
 UUUU Linien-Nr. des Unterstreichstrichs
 LLLL Anzahl der Linien/Zeichenzeile
 M Linienzähler-Modus
 M = 0 duale Zählweise, mit 0000 beginnend
 M = 1 Zählweise um 1 versetzt (mit höchster Linien-Nr. beginnend, dann 0000, 0001, ...)
 F Feldattribut-Modus
 F = 0 Transparent (Feldattribute werden auf dem Bildschirm nicht sichtbar; Benutzung des FIFO-Speichers)
 F = 1 Nicht-Transparent (Feldattribute stehen an Stellen darstellbarer Zeichen und werden als Blank dargestellt)

CC	Cursor-Modus
00	Blinkend invers
01	Blinkend unterstreichen
10	Nicht-blinkend invers
11	Nicht-blinkend unterstreichen
ZZZZ	Länge des horizontalen Strahlrücklauf-Intervalles (1–32 Zeichentakte)
SSS	Anzahl der Zeichentakte zwischen 2 DMA-Anforderungen (beim PC 1715: 0 Takte, zu programmieren 000)
BB	Anzahl der DMA-Zyklen pro Busanforderung (beim PC 1715: 1 DMA-Zyklus, zu programmieren 00)

Ein *Feldattributcode* (als Blank dargestellt oder transparent) schaltet einen bestimmten Zustand ein, der vom folgenden Zeichen an solange wirkt, bis er durch einen weiteren Feldattributcode, in dem das betreffende Bit 0 ist, wieder aufgehoben wird bzw. bis das Bildende erreicht ist.

Codierung: 1 0



z. B. B = 1: Blinken ein B = 0: Blinken aus

Die technischen Daten der Bildschirmsteuerung bei Verwendung der Monitore K 7221 und K 7222 sind in der Tabelle 1.3 dargestellt.

Die *integrierten Ein-Ausgabe-Kanäle* sind Schnittstellen, über die verschiedenartige Peripheriegeräte angeschlossen werden können. Die prinzipielle Arbeitsweise dieser Schnittstellen wird in dem Blockschaltbild von Abbildung 1.4 wiedergegeben.

Die Anpassung des parallel arbeitenden Systembusses an die Schnittstelle für serielle Datenübertragung V.24 übernimmt eine Anschlußsteuerung ent-

Tabelle 1.3
Technisches Datenblatt zur Bildschirmsteuerung

Parameter	K 7221	K 7222
Anzahl Zeichen/Bild	16×64	24×80
Zeichenumfeld (Bildpunkte)	8×15	8×12
Zeichen (Bildpunkte)	6×9	6×9
Punktfrequenz	10,7 MHz	13,8 MHz
Punkttaktperiode (BPT1/2)	93,5 ns	72,3 ns
Zeichentaktperiode (CCLK)	750 ns	580 ns
Bildwechselfrequenz	55 Hz	51 Hz

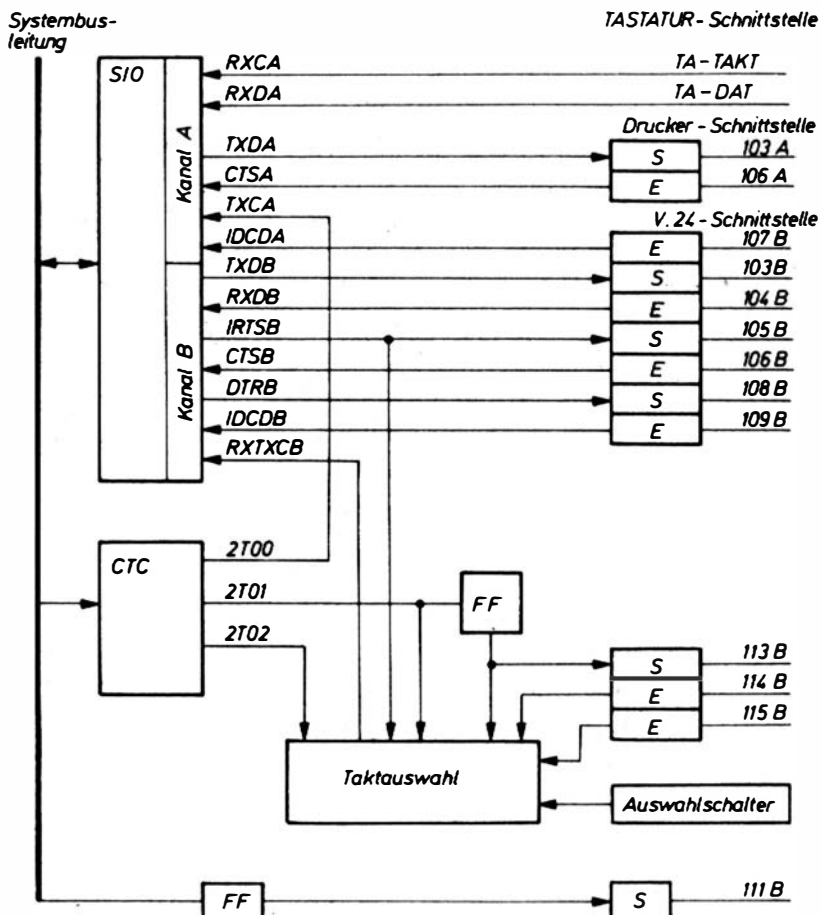
Tabelle 1.4
Technisches Datenblatt zur V.24-Schnittstelle

Merkmalsbezeichnung	Merkmalswerte
Steckverbinder	Steckerleiste 103 -13 (13polig) TGL 28331/04
Betriebsweisen	duplex, halbduplex
Gleichlaufverfahren	synchron, asynchron
Übertragungsgeschwindigkeit	50, 75, 150, 300, 600, 1200, 2400, 4800, 9600 Bd
Zeichenformat	5 ... 8 Bit/Zeichen
Stoppbitlänge	1, 1 1/2, 2 Bit
Paritätsprüfung	ohne, gerade, ungerade
Anschlußgeräte	Modem, GDN oder Terminals mit Schnittstellen nach V.24
Länge der Anschlußkabel	max. 15 m
Art des Kabels	HYF (C) Y(12×2×0,14) mm ²

sprechend ESER-Standard für S2 bzw. TGL 29077/01 (CCITT – V.24). Die Schnittstelle bzw. das Gerät stellen eine Datenendstelle (DEE) dar, die über Datenübertragungseinrichtungen (DUE) mit fernaufgestellten DEE oder mit nahaufgestellten DEE direkt verbunden werden kann. Die für diese Schnittstelle charakteristischen technischen Daten sind in der Tabelle 1.4 zusammengestellt.

Die *Schnittstellensteuerung* besteht aus den Funktionskomplexen

- Takterzeugung für Takteingänge des SIO,
- Schnittstellensteuerung V.24 durch SIO-Kanal B und
- Treiber- und Empfängerschaltkreise.



S = Sender

E = Empfänger

AB0 ≙ Kanalauswahl SIO und KSO beim CTC

AB1 ≙ Daten- oder Steuerwort beim SIO und KS1 beim CTC

Abbildung 1.4

Prinzipielle Arbeitsweise der Schnittstellen

(Blockschaltbild)

Tabelle 1.5
Taktvarianten V.24 (S1.2, S2.2)

Steuerung durch Ltg 105	RXTXCB		S1:2			S2:2	
	Sendetakt	Empfangstakt	1-8	2-7	3-6	1-4	2-3
-	Ltg 114	Ltg 114	-	+	+	-	+
-	Ltg 113	Ltg 113	-	+	+	+	-
x	Ltg 114	Ltg 115	+	-	+	-	+
x	Ltg 113	Ltg 115	+	-	+	+	-
-	Ltg 115	Ltg 115	-	-	+	-	-
-	ZC/T02	ZC/T02	-	+	-	-	-
-	ZC/T01	ZC/T01	-	-	-	-	-
x	ZC/T02	ZC/T01	+	-	-	-	-

- + Schalter geschlossen
- Schalter geöffnet

Die *Taktbereitstellung* für den SIO-Kanal B erfolgt über einen Multiplexerschaltkreis, der durch Schalter programmierbar ist. Es ist möglich, den gewünschten Takt entweder durch die zwei CTC-Kanäle oder durch die Leitungen 113, 114 sowie 115 zur Verfügung zu stellen. Es ist möglich, bei unterschiedlichen Taktfrequenzen des Sende- und Empfangstaktes über den Taktmultiplexer durch Steuerung mit der Leitung 105 den gewünschten Takt zur Wirkung zu bringen. Die Zuordnung der Sende- und Empfangstakte ist aus Tabelle 1.5 zu entnehmen. Eine Darstellung der Schalter findet sich in Abbildung 1.5. In Tabelle 1.6 wird ein Überblick über die Schnittstellenleitungen und die Kontaktbelegung für eine V.24-Schnittstelle gegeben.

Die serielle Schnittstelle *Printerausgang* wird zur Ansteuerung des Systemdruckers genutzt. Dabei wird der Sendeteil des SIO-Kanals A zur Bildung einer seriellen Schnittstelle zur Ansteuerung eines Druckers verwendet (vgl. Abbildung 1.4). Es werden die Leitungen 102 (Betriebserde), 103 (Sendedaten) und 106 (Bereit zum Senden) einer V.24-Schnittstelle genutzt (vgl. Tabelle 1.7). Der Sendetakt wird durch den CTC-Kanal 00 erzeugt.

Der Systemdrucker muß eine V.24-Schnittstelle mit folgenden Einstellwerten besitzen:

Schnittstelle V.24 mit TGL-Buchsenleiste 223-13

Baudrate	9600
Zahl der Datenbit pro Byte	8
Parität	ohne
Zahl der Stoppbit	1
Übertragungsart	duplex
Protokoll	DTR-Protokoll

Die Baudrate kann als einziger Parameter mit Hilfe des Programmes INSTSCP geändert werden.

Der Informationsaustausch zwischen Tastatur und Systemeinheit findet über eine *serielle Schnittstelle zur Ansteuerung der Tastatur* statt. Die Serien-Parallel-Wandlung erfolgt über den Kanal A des SIO. Die seriellen Daten der Tastatur gelangen über den Eingang RXDA des SIO zur weiteren Umwandlung in parallele Daten ins Empfangsdatenregister und werden durch die CPU von dort gelesen. Der dazugehörige Empfangstakt wird vom /WR-Steuersignal des Mikroprozessors der Tastatur abgeleitet.

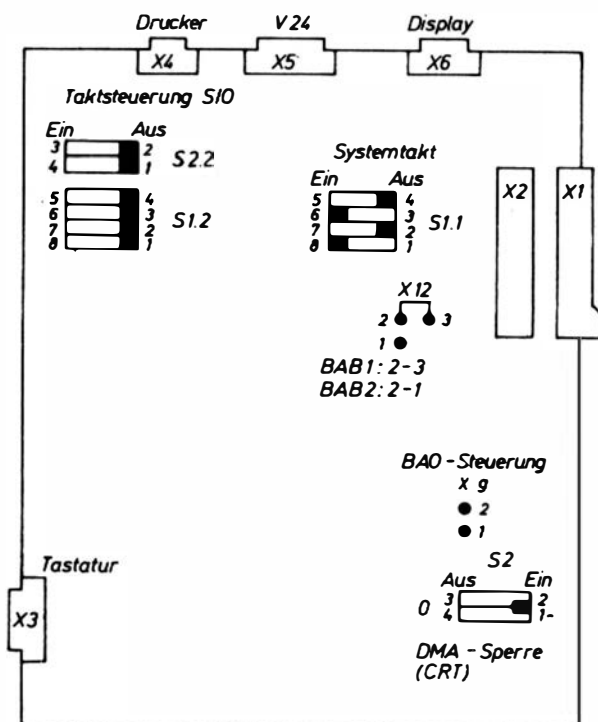


Abbildung 1.5
Darstellung der Schalter

Tabelle 1.6
Leitungen und Kontaktbelegung für eine V.24-Schnittstelle

SIO-Anschlüsse Kanal B	Schnittstellenleitung	Schnitt- stellen- kontakt X5	Übertragungs- richtung DEE – DUE	
	102 Betriebserde	A1	↔	
TxDB	103 Sendedaten	A3	→	
RxDB	104 Empfangsdaten	B4	←	
/RTSB	105 Aufforderung zum Senden	A5	→	
/CTSB	106 Bereit zum Senden	B6	←	
/DCDA	107 Betriebsbereitschaft (DUE)	A7	←	
/DTRB (nur bei Bondvariante 0)	108/1 Datenendstelle mit Übertragungsweg verbinden 108/2 Datenendstelle betriebsbereit	B8	→	
/DCDB	109 Empfangssignalpegel	A9	←	
Bondvar. 0	Bondvar. 1 /TxCB	113 Sendeschrittakt (Quelle DEE)	A11	→
/RxTxCB		114 Sendeschrittakt (Quelle DUE)	B12	←
	/RxCB	115 Empfangsschrittakt (Quelle DUE)	A13	←
Flip – Flop D174 Toradresse /LT111CS 30H, 31H, 32H oder 33H DBI = 0 = 111 AUS DBI = 1 = 111 EIN	111 Wahl der Über- tragungs- geschwindigkeit	B10	→	

Tabelle 1.7

Leitungen und Kontaktbelegung für eine serielle Schnittstelle zur Ansteuerung eines Druckers

SIO-Anschlüsse Kanal A	Schnittstellenleitung	Schnittstellenkontakt X4	Übertragungsrichtung DEE – DUE
	102 Betriebserde	A1	↔
TxDB	103 Sendedaten	B2	→
/CTS	106 Bereit zum Senden	A3	←

1.1.2. Ansteuereinheit für Diskettenlaufwerke

Diese Steckeinheit dient der Ansteuerung von maximal vier Diskettenlaufwerken.

Abmessungen:

Breite: 215 mm

Tiefe: 170 mm

Höhe: 15 mm

Die Ansteuereinheit umfaßt die Komponenten

- Schnittstelle zum Rechnerbus
- Belegung der PIO und Register
- Schnittstelle zu den Laufwerken
- Phasenregelkreis (PLL)
- Lesedatenaufbereitung
- Seriell-Parallel-Wandlung und Markenerkennung
- Parallel-Seriell-Wandlung
- Schreibsteuerung
- Synchronisierung der Datenübertragung.

Es können folgende Typen von Diskettenlaufwerken angeschlossen werden: MF 6400, K 5600.10 (MFS 1.2), K 5600.20 (MFS 1.4) und MFS 1.6. Dabei können 5,25-Zoll- bzw. 8-Zoll-Disketten in den Aufzeichnungsverfahren FM und MFM bearbeitet werden. Auf der Steckeinheit befinden sich 2 Kabel mit 26poligen Steckverbindern, über die je ein geräteinternes Diskettenlaufwerk der Typen K 5600.10 (MFS 1.2), K 5600.20 (MFS 1.4) und MFS 1.6 angeschlossen werden kann. Über einen weiteren 39poligen Steckverbinder

können noch zwei externe Diskettenlaufwerke der Typen MF 6400 oder K 5600.10 (MFS 1.2) angeschlossen werden.

In der Tabelle 1.8 werden die Parameter der verwendeten Diskettenlaufwerke und der sich daraus ergebenden Kennwerte der Disketten unter Berücksichtigung der Anforderungen des Betriebssystems SCP dargestellt.

Tabelle 1.8
Übersicht über die Parameter/Kennwerte von Diskettenformaten

Parameter/Kennwerte	Diskettenformate			
	8" DD	5,25" DD einseitig	5,25" DD einseitig	5,25" DD zweiseitig
Spurdichte	48 tpi	48 tpi	96 tpi	96 tpi
FD-Laufwerk	MF 6400	MFS 1.2	MFS 1.4	MFS 1.6
Anzahl der Spuren	77	40	80	80
Anzahl der Systemspuren	2	3	3	2
Sektoren/Spur	8	16	16	16
Bytes/Sektor	1024	256	256	256
Seiten/Diskette	1	1	1	2
Speicherkapazität (in K Byte)				
Diskette				
- unformatiert	784	244	488	976
- formatiert	616	160	320	640
- nutzbar	600	148	308	624

Berechnungsgrundlagen:

Bytes/Spur unformatiert:	5 ¹ / ₄ "	6.250
	8" SD	5.208
	8" DD	10.416

unformatiert:	Bytes/Spur (unform.) × Anzahl Spuren
formatiert:	Bytes/Sektor × Sektor/Spur × Anzahl Spuren × Seiten
nutzbar:	Bytes/Sektor × Sektor/Spur × (Anzahl Spuren – Systemspuren) × Seiten

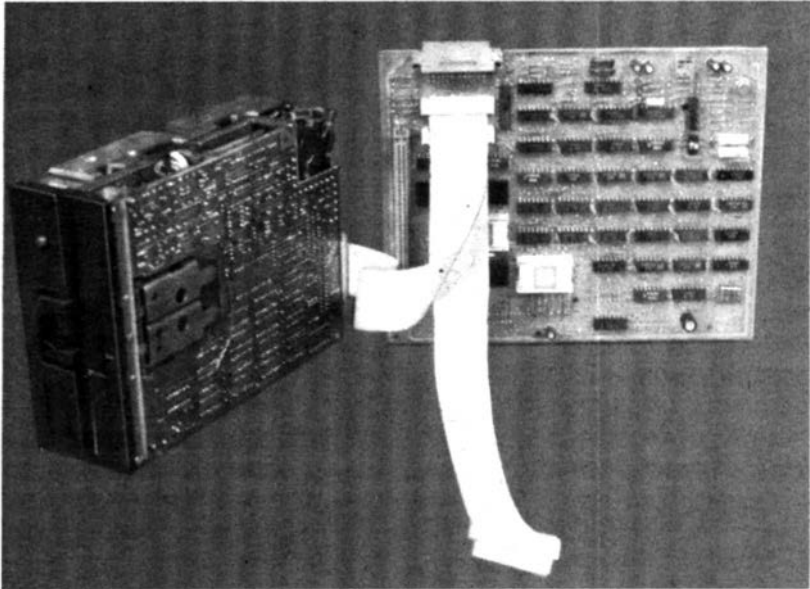


Abbildung 1.6
Steckeinheit Ansteuerung der Diskettenlaufwerke

1.1.3. Minifolienspeicher

Als Komponenten der Systemeinheit können zwei Diskettenlaufwerke der Typen K 5600.10 (MFS 1.2) oder K 5600.20 (MFS 1.4) oder MFS 1.6 zum Einsatz kommen. Diese Laufwerke sind elektromotorische Speicher mit Direktzugriff.

Das Laufwerk K 5600.10 erhält über den 26poligen Steckverbinder der Ansteuersteckeinheit direkt die erforderlichen Signale, die für den Informationsaustausch erforderlich sind, und Versorgungsspannungen von 5P und 12P. Die notwendigen Steuer- und Regelvorgänge werden durch die MFS-eigene Logik realisiert. Als Datenträger werden Disketten mit 5,25 Zoll Durchmesser eingesetzt. Die Aufzeichnungsformate sind in der Tabelle 1.8 übersichtlich dargestellt.

Das Laufwerk K 5600.20 erhält über den 26poligen Steckverbinder der Ansteuersteckeinheit die erforderlichen Signale, die für den richtigen Informationsaustausch adaptiert werden müssen. Die Versorgungsspannungen von 5P und 12P werden über den Adapter aus der direkten Verbindung abgeleitet

und auf den Steckverbinder des Laufwerkes geführt. Die notwendigen Steuer- und Regelvorgänge werden durch die MFS-eigene Logik realisiert. Als Datenträger werden ebenfalls Disketten mit 5,25 Zoll Durchmesser eingesetzt (zu den Aufzeichnungsformaten vgl. Tabelle 1.8).

Das Laufwerk MFS 1.6 erhält über den 26poligen Steckverbinder der Ansteuersteckeinheit die erforderlichen Signale, die für den richtigen Informationsaustausch adaptiert werden müssen. Die Versorgungsspannungen von 5P und 12P werden über den Adapter aus der direkten Verbindung abgeleitet und auf den Steckverbinder des Laufwerkes geführt. Der Adapter für den MFS 1.6 ist *nicht* kompatibel zum Adapter des K 5600.20. Die notwendigen Steuer- und Regelvorgänge werden durch die MFS-eigene Logik realisiert. Als Datenträger werden Disketten mit 5,25 Zoll Durchmesser eingesetzt (zu den Aufzeichnungsformaten vgl. Tabelle 1.8).

1.1.4. Stromversorgung

Das Netzteil des PC 1715 wurde in seinen Parametern so ausgelegt, daß es alle Baugruppen (einschließlich Interfaceerweiterungen) versorgen kann.

Abmessungen:

Breite: 160 mm

Tiefe: 370 mm

Höhe: 115 mm

Die Stromversorgung umfaßt die Komponenten

- Netzgleichrichtung
- Anlaufschaltung und Hilfsspannungserzeugung
- 5P-Regelstrecke mit Überstromsicherung und Überspannungskontrolle
- 5N- und 12N-Erzeugung
- 12P-Regelstrecke mit Strombegrenzung und Verzögerungsschaltung.

Das Gerät ist für den Betrieb an einem 220-V-Wechselstromnetz von 47 Hz bis 63 Hz ausgelegt. Die Netzspannung darf bis zu +10 % und -15 % vom Nennwert abweichen.

Die Baugruppe Stromversorgung beinhaltet zwei Schaltnetzteile sowie zwei Analogregler und stellt für den PC 1715 die erforderlichen Gleichspannungen (5P, 12PM, 12P, 5N und 12N) bereit.

Die Netzspannung wird an der Rückseite über einen Schuko-Kaltgerätestecker und zwei Schmelzsicherungen zugeführt. An der Vorderseite befindet sich der Einbauwippschalter. Seitlich ist ein in der Drehzahl reduzierter

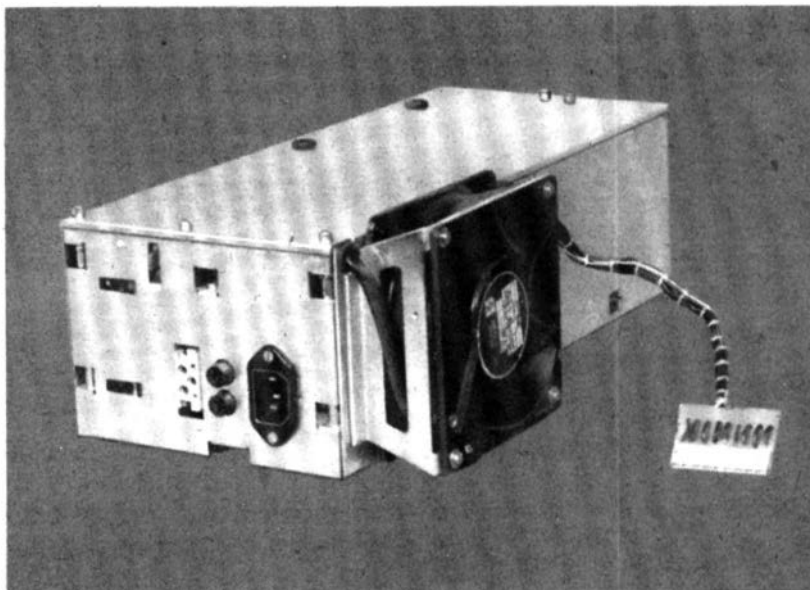


Abbildung 1.7
Netzteil des PC 1715

Axiallüfterbaustein angebracht, der die Luftzirkulation im Gesamtgerät gewährleistet. Die Abgabe der Ausgangsspannungen erfolgt über die Buchsenleiste XNT an die ZRE, auf der zur Kontrolle LEDs installiert wurden. Für die Bildschirmeinheit wird die Betriebsspannung 12PM an der Rückseite der Baugruppen Stromversorgung über eine dreipolige Buchsenleiste (DISPLAY) bereitgestellt.

1.2. Die Bildschirmtypen K 7221 und K 7222

Die folgende Beschreibung gilt für beide Bildschirmtypen.

Abmessungen:

Breite: 320 mm
Tiefe: 350 mm
Höhe: 330 mm
Gewicht: 10,5 kg

Das Bildschirmgerät umfaßt die Komponenten

- Kabelempfänger und Pegelwandler
- Hellstastverstärker
- Intensitätssteuerung
- Synchronimpulstrennung und Vertikalablenkstufe
- Horizontalgenerator mit Phasenvergleich
- Horizontalendstufe und Hochspannungserzeugung
- Erzeugung der Betriebsspannung.

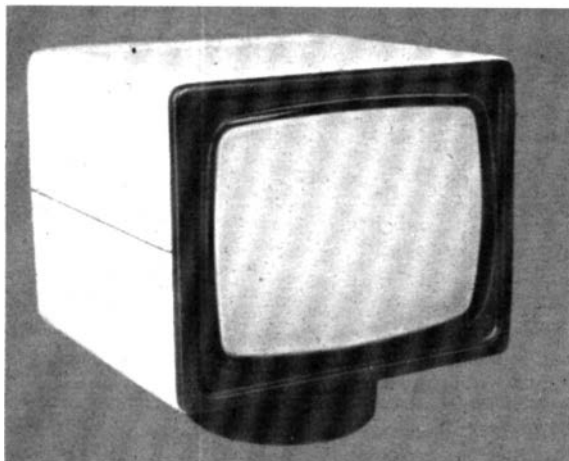


Abbildung 1.8
Bildschirm des PC1715

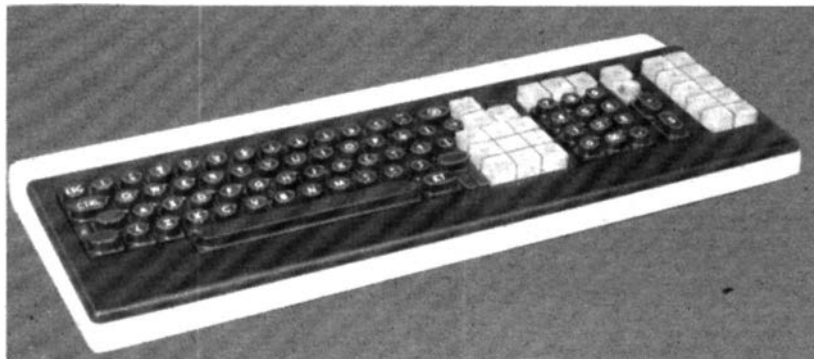


Abbildung 1.9
Tastatur des PC1715

Der Bildschirm ist eine Baugruppe zur visuellen Informationsdarstellung mittels einer 31-cm-Bildröhre. Er enthält nur die elektrischen Funktionsgruppen, die zur Erzeugung eines Bildfeldes auf der Bildröhre erforderlich sind (Ablenkstufen, Hochspannungserzeugung, Videosignal-Endverstärker) sowie die dazu erforderlichen Stromversorgungsblöcke. Außerdem sind Empfangsstufen für die Informationen bzw. Steuersignale übertragenden Kabel vorhanden. Die Aufbereitung der auf diesen Kabeln übertragenen Signale erfolgt auf der ZRE-Steckeinheit, die in der Systemeinheit untergebracht ist. Die Anzahl der darzustellenden Zeichen, der Zeichenvorrat und das Format des zur Zeichenerzeugung dienenden Punktrasters werden ausschließlich von der Ansteuerung bestimmt.

Außer Bildröhre, Ablenkeinheit, Helligkeitsregler und Kontrastregler bei K 7222 sind alle elektrischen Bauelemente auf Leiterplatten angeordnet. Die in der Anschlußsteuerung erzeugten Signale KSYN und KVID für K 7221 bzw. BSYN, VIDEO und INTENS für K 7222 werden über Koaxialkabel zum Bildschirm übertragen.

1.3. Tastatur

Die Tastatur dient der manuellen Eingabe von alphanumerischen und numerischen Zeichen, Ruf- und Steuerinformationen sowie zur Auslösung gerätespezifischer Funktionen.

Abmessungen:

Breite:	500 mm
Tiefe:	200 mm
Höhe:	40 mm
Neigung:	5 Grad
Gewicht:	2,5 kg

Die Tastatur umfaßt die Komponenten

- Mikroprozessor U 880
- Tastaturmatrix
- Programmspeicher und Zeichengenerator
- Octal-Latch mit Treiber.

Die Baugruppe Tastatur enthält alle Elemente zur Eingabe von Informationen sowie zur Auslösung gerätespezifischer Funktionen. Ergänzt wird die Tastatureinheit durch zwei Anzeigeelemente (LED).

Die vollelektronische Tastatur arbeitet nach dem Elastomer-Prinzip und ist

gertaste gestaltet. Bei der ersten Betätigung erfolgt das Umschalten von Klein- auf Großbuchstaben. Durch die LED-Anzeige wird das Umschalten optisch angezeigt. Bis zur erneuten Tastenbetätigung bleibt die Tastatur in dieser Umschaltstellung. Mit der nächsten Betätigung wird der Grundzustand wieder hergestellt. Die LED-Anzeige verlischt.

Sonderzeichen und Ziffern werden in den Umschaltvorgang nicht mit einbezogen.

- SI Diese Taste bewirkt in Abhängigkeit vom Betriebssystem eine Umschaltung in den 2. Zeichensatz. Die Umschaltung wird durch ein optisches Signal verdeutlicht (Leuchtdiode neben der Umschalttaste).
- R Die „Repeat“-Taste löst Dauerfunktion aus für die Taste, die zugleich mit der Repeat-Taste betätigt wird.
- ESC Die „Escape“-Taste bewirkt die Ausgabe der Codierung H'1B', wobei das jeweilige Betriebssystem die Bedeutung festlegt.
- CTRL Die „CTRL“-Taste veranlaßt die Aktivierung einer zusätzlichen Code-Ebene. Diese Taste ist immer gemeinsam mit einer weiteren Taste zu betätigen.
- ET Die „ET“-Taste hat die Bedeutung einer Abschlußtaste (z. B. Ende des Textes oder Ende einer Eingabe) und bewirkt die Ausgabe der Codierung H'9E'.
- INS Die „Insert“-Taste gestattet das Einfügen von Zeichen bis zum Betätigen einer Cursorfunktion.
- DEL Mit der „Delete“-Taste lassen sich Zeichen im Speicher bzw. vom Display löschen.

Die *Kursortasten* gestatten die Bewegung des Cursors auf dem Bildschirm. Die Bedeutung der einzelnen Tasten wird in den Beschreibungen des jeweiligen Betriebssystems erläutert. (Vgl. Tabelle 1.9.)

Die *Zehnertastatur* ist unmittelbar neben der Funktionstastatur angeordnet und besteht aus 15 verschiedenen Tasten. (Vgl. Tabelle 1.10.)

Die *Funktionstastatur* enthält 14 Funktionstasten, deren Bedeutung durch das jeweilige Betriebssystem bzw. das Programm festgelegt wird. Die Tastenbeschriftung wurde so gewählt, daß mit dem Textprogramm Vers 1/3 BWS optimal gearbeitet werden kann (vgl. 6. Kapitel und Tabelle 1.11).

Tabelle 1.9
Übersicht über die Kursortasten

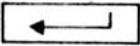

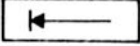
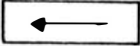
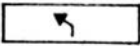
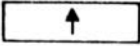
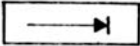

Tastensymbole	Codierung (hex)	Bedeutung (allgemein)
	9D	Return/New-Line
	86	Zeichen nach rechts
	87	linker Rand
	88	Zeichen nach links
	8C	Format-Anfang
	8B	Zeile hoch
	89	rechter Rand
	8A	Zeile nach unten
	8E	Format Ende

Tabelle 1.10
Übersicht über die Zehnertastatur

Tastenbezeichnung	Codierung (hex)	Bedeutung
1	B1	Ziffer 1
2	B2	Ziffer 2
3	B3	Ziffer 3
4	B4	Ziffer 4
5	B5	Ziffer 5
6	B6	Ziffer 6
7	B7	Ziffer 7
8	B8	Ziffer 8
9	B9	Ziffer 9
0	B0	Ziffer 0
00	BB	Doppelnull
,	AC	Komma
-	BD	Minus
CE	CE	-
S	D0	Endtaste nach der Eingabe (analog ET)

Tabelle 1.11
Übersicht über die Funktionstastatur

Tastenbezeichnung (allgemein)	Tastenbezeichnung (Textversion)	Codierung (hex)
F 1	^OI ^OL	D1
F 2	^OF ^OC	D2
F 3	^ON ^OR	D3
F 4	^R ^R	D4
F 5	F 5	CF
F 6	^KN ^KH	A0
F 7	^QB ^KB	A1
F 8	^QK ^KK	A2
F 9	^KV ^KC	A3
F 10	Formularvorschub	
F 11	^QA ^QF	C1
F 12	^QV ^QP	C0
F 13	^KQ ^KS	C2
F 14	^KX ^KD	CD

1.4. Peripherieerweiterungen

1.4.1. Interfaceerweiterungssteckeinheit für 2×V.24-Schnittstelle

Die Anschlußsteuerung übernimmt die Anpassung des parallel arbeitenden Systembusses an die serielle Schnittstelle (entsprechend ESER-Standard für S2 bzw. TGL 29077/01 (CCITT-V.24)). Sie stellt, von der Schnittstelle aus betrachtet, eine Datenendstelle (DEE) dar, die über Datenübertragungseinrichtungen (DUE) mit fernaufgestellten DEE direkt verbunden werden kann. Charakteristische technische Daten sind der Tabelle 1.12 zu entnehmen.

Die Anschlußsteuerung besteht aus den Funktionskomplexen

- Takterzeugung,
- Taktverteilung,
- Schnittstellensteuerung für V.24 durch SIO und
- Treiber- und Empfängerschaltkreise.

Tabelle 1.12

Technisches Datenblatt zur Interfaceerweiterungssteckeinheit für 2 × V.24

Merkmalsbezeichnung	Merkmalswerte
Abmessungen	150 mm × 150 mm
Steckverbinder	1 × Buchsenleiste 202-58 (58polig) TGL 29331/03 2 × Steckerleiste 102-13 (13polig) TGL 29331/04
Stromversorgung	+ 5 V +/- 5 % typ 0,4 A + 12 V +/- 5 % typ 0,03 A - 12 V +/- 5 % typ 0,02 A
Kanäle	2 unabhängige V.24-Kanäle
Betriebsweisen	duplex, halbduplex
Gleichlaufverfahren	synchron, asynchron
Übertragungsgeschwindigkeit	50, 75, 150, 300, 600, 1200, 2400, 4800, 9600 Bd.
Zeichenformat	5...8 Bit/Zeichen
Stoppbitlänge	1, 1½, 2 Bit
Paritätsprüfung	ohne, gerade, ungerade
Anschlußgeräte	Modem, GDN oder Terminals mit Schnittstellen nach V.24
Schnittstellenleitungen	102, 103, 104, 105, 106, 107, 108, 109, 111, 113 114, 115 nach V.24
Länge der Anschlußkabel	max. 15 m
Art des Kabels	HYF (C) Y(12×2×0,14) mm ²
Anschluß zum Systembus	2 Adreßleitungen (AB0, AB1) 8 Datenleitungen (DB0...DB7) 9 Steuerleitungen (/M1, /IORQ, /RD, TAKT, /RESET, IEI, IE0, /WAIT, /INT)

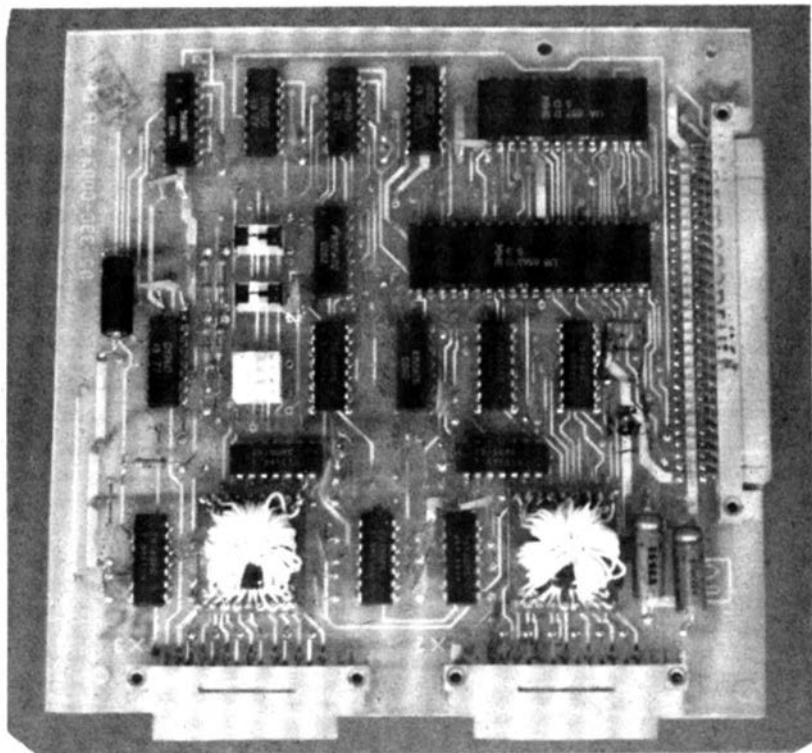


Abbildung 1.11
Interfaceerweiterungssteckeinheit für V.24

Bei der Taktauswahl können Takteingänge des SIO über Multiplexerschaltkreise mit den gewünschten Takten verbunden werden. Es ist möglich, den jeweiligen Takt durch die drei CTC-Kanäle oder durch die Leitungen 113, 114 sowie 115 zur Verfügung zu stellen. Am Kanal B steht bei der Bondvariante 0 nur ein Takteingang für Sende- und Empfangstakt zur Verfügung. Es ist möglich, bei unterschiedlichen Taktfréquenzen des Sende- und Empfangstaktes über den zum Kanal B gehörenden Taktmultiplexer durch Steuerung mit der Leitung 105 den gewünschten Takt zur Wirkung zu bringen.

Die Zuordnung der Sende- und Empfangstakte ist aus Tabelle 1.13 zu entnehmen; in Tabelle 1.14 wird die Bedeutung der Schnittstellenleitungen und die Kontaktbelegung dargestellt.

Tabelle 1.13
Zuordnung der Sende- und Empfangstakte
bei der Interfaceerweiterungssteckeinheit für 2 × V.24

SIO-Kanal A

Sendetakt	Empfangstakt	S1:1	
		1-4	2-3
TXCA	RXCA		
Ltg 114	Ltg 115	x	x
Ltg 113	Ltg 115	x	-
ZC/T00	ZC/T02	-	x
ZC/T00	ZC/T00	-	-

SIO-Kanal B

Steuerung durch Ltg 105	RXTXCB Sendetakt	Empfangstakt	S1:2		S2		
			1-4	2-3	1-8	2-7	3-6
-	Ltg 114	Ltg 114	-	x	x	x	-
-	Ltg 113	Ltg 113	x	-	x	x	-
x	Ltg 114	Ltg 114	-	x	x	-	x
x	Ltg 113	Ltg 115	x	-	x	-	x
-	Ltg 115	Ltg 115	-	-	x	-	-
-	ZC/T02	ZC/T02	-	-	-	x	-
x	ZC/T02	ZC/T01	-	-	-	-	x
-	ZC/T01	ZC/T01	-	-	-	-	-

x geschlossener Schalter
- geöffneter Schalter

Tabelle 1.14
Bedeutung der Schnittstellenleitungen und Kontaktbelegungen
SIO-Anschlüsse

Kanal A	Kanal B	Schnittstellenleitung	Schnittstellenkontakt X2, X3	Übertragungsrichtung DEE – DUE
		102 Betriebserde	A1	↔
TxDA	TxDB	103 Sendedaten	A3	→
RxDA	RxDB	104 Empfangsdaten	B4	←
/RTSA	/RTSB	105 Aufforderung zum Senden	A5	→
/CTSA	/CTSB	106 Bereit zum Senden	B6	←
/DTRA	/DTRB	108/1 Datenendstelle mit Übertragungsweg verbinden 108/2 Datenendstelle betriebsbereit	B8	→
/DCDA	/DCDB	109 Empfangssignalpegel	A9	←
/TxCA		113 Sendeschrittakt	A11	→
	/RxTxCB	114 Sendeschrittakt	B12	←
/RxCA		115 Empfangsschrittakt	A13	←
Register-Abfrage				
Datenbit				
DB0	DB2	107 Betriebsbereitschaft (DUE)	A7	←
Flip-Flop				
Datenbit				
DB0	DB2	111 Wahl der Übertragungsgeschwindigkeit	B10	→

Leitung 107 und 111

Leitung	Kanal A (DBO)		Kanal B (DB2)		Toradresse
	EIN	AUS	EIN	AUS	
107	0	1	0	1	/Ltg.107CS 20H oder 2FH
111	1	0	1	0	/Ltg.107CS 2CH oder 2EH

1.4.2. Interfaceerweiterungssteckeinheit für 2×IFSS-Schnittstelle

Diese Steckeinheit enthält die Anschlußsteuerung für 2 IFSS-Datenübertragungskanäle. Die Anschlußsteuerung realisiert 2 Interfaces zum sternförmigen Anschluß von Ein-/Ausgabegeräten mit serieller Informationsübertragung (IFSS). Das angewendete Interface entspricht dem im System der Kleinrechner (SKR) vereinbarten System zur seriellen Informationsübertragung zwischen Ein-/Ausgabegerät (E/A-Gerät) und Anschlußsteuerung eines SKR-Rechners in der speziellen Auslegung für eine 20 mA-Stromschleife. Charakteristische technische Daten sind in Tabelle 1.15 zusammengestellt. Die Anschlußsteuerung besteht aus den Funktionskomplexen

- Takterzeugung durch CTC,
- Schnittstellensteuerung für IFSS durch SIO,
- Kabelstufen für IFSS und
- Konstantstromquellen.

Die Takterzeugung erfolgt durch den Baustein U 857 (CTC). Er wird als programmierbarer Frequenzteiler zur Bereitstellung der vom Baustein U 856 benötigten Sende- und Empfangstakte benutzt. Die Kanäle 0, 1 und 2 des CTC sind zu diesem Zweck als Zeitgeber (entsprechend der zu realisierenden Übertragungsgeschwindigkeit der seriellen Daten (Baudrate)) zu programmieren. Dabei gilt folgende feste Zuordnung:

Kanal A	Sendetakt	Kanal 0
	Empfangstakt	Kanal 1
Kanal B	Sende-/Empfangstakt	Kanal 2

Die Aufgabe des SIO U 856 besteht in der Parallel-Serien-Wandlung der Ausgabedaten der ZRE sowie der Rückwandlung der über die Kabelstufen

empfangenen seriellen Eingabedaten und der zugehörigen Schnittstellensteuerung. Die Kanäle A und B des SIO sind unabhängig voneinander zu betreiben und zu programmieren. (Hinsichtlich der spezifischen Arbeitsweise des SIO U 856 wird auf die umfangreiche technische Beschreibung des Schaltkreisherstellers verwiesen.)

Das IFSS ist ein serielles Interface zur direkten Kopplung von Ein-/Ausgabe-Geräten über Entfernungen bis zu 500 m in der Auslegung als 20 mA-Stromschleife. Der Datenaustausch erfolgt asynchron im Start-Stopp-Verfahren über eine vieradrige Duplexverbindung. Je 2 Leitungen bilden eine Stromschleife (Sende- und Empfangsschleife), die über optoelektrische Koppler mit der Ein- und Ausgabe-Schaltungsanordnung verbunden ist. Der Strom in der Schleife beträgt im Zustand der logischen „1“ (Ruhezustand) 15 mA... 25 mA (Nennwert 20 mA) und im Zustand logisch „0“

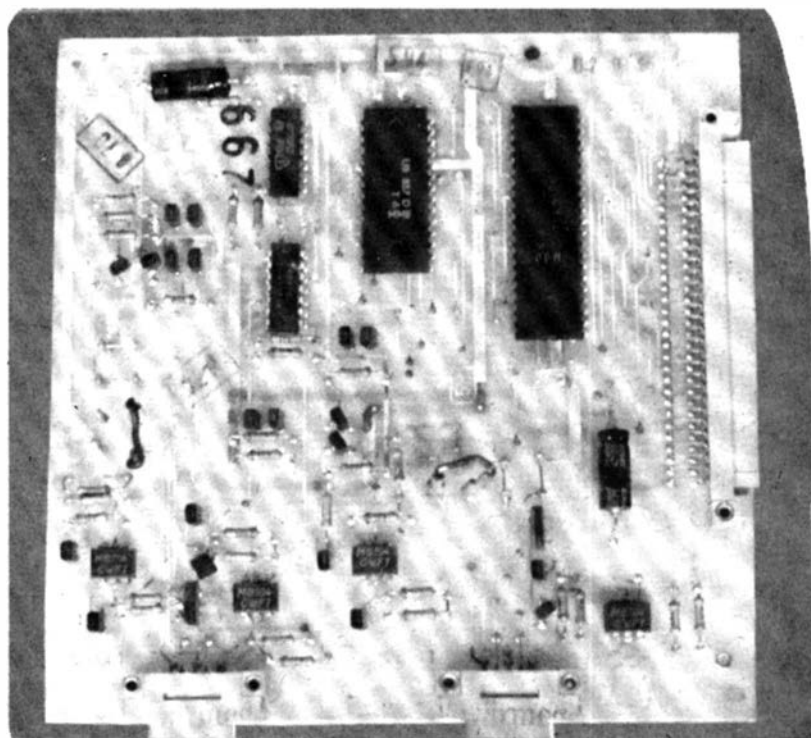


Abbildung 1.12
Interfaceerweiterungssteckeinheit für IFSS

Tabelle 1.15
Technisches Datenblatt zur Interfaceerweiterungssteckereinheit für 2 × IFSS

Merkmalbezeichnung	Merkmalswerte
Abmessungen	150 mm × 150 mm
Steckverbinder	1 × Buchsenleiste 202-58 (58polig) TGL 29331/03 2 × Steckerleiste 103-5 (5polig) TGL 29331/04
Stromversorgung	+ 5 V +/- 5 % typ 0,3 A + 12 V +/- 5 % typ 0,1 A
Kanäle	2 unabhängige IFSS-Kanäle Betriebsweisen: duplex, halbduplex Gleichlaufverfahren: asynchron Zeichenformat: 5...8 Bit/Zeichen
Stoppbitlänge	1, 1 ^{1/2} , 2 Bit
Parität	gerade, ungerade, ohne
Übertragungsgeschwindigkeit	150, 200, 300, 600, 1200, 2400, 4800, 9600 Bd
Übertragungsentfernung	max. 500 m
Elektrische Bedingungen der Stromschleifen	Logisch "0": 0... 3 mA Logisch "1": 15...25 mA
Anschluß zum Systembus (X1)	2 Adreßleitungen (AB0, AB1) 8 Datenleitungen (DB0...DB7) 9 Steuerleitungen (/M1, /IORQ, /RD, TAKT, /RESET, IEI, IEO, /WAIT, /INT

1 mA ... 3 mA (Nennwert 2 mA). Die Einspeisung des Schleifenstromes kann sowohl von der Anschlußsteuerung (Aktivmodus) als auch auf der jeweiligen Gegenstelle (Passivmodus) erfolgen. Die Anschlußsteuerung besitzt zur Realisierung des gewünschten Arbeitsmodus zwei Konstantstromquellen, die in die Sende- und Empfangsschleife eingeschaltet werden können. Die Kabelstufen der Anschlußsteuerung (IFSS) sind so ausgelegt, daß Vertauschungen der Anschlüsse oder Schlüsse auf den Übertragungsleitungen nicht zu Schäden führen.

Die Anschlüsse der beiden Datenübertragungskanäle befinden sich an der Griffseite der Anschlußsteuerung. Die beiden 5poligen Steckerleisten X2 (Kanal B) sowie X3 (Kanal A) sind durch ein Fenster in der Verkleidung erreichbar.

Die Schnittstellenleitungen der IFSS-Kanäle sind folgenden Kontakten der beiden Steckverbinder zugeordnet:

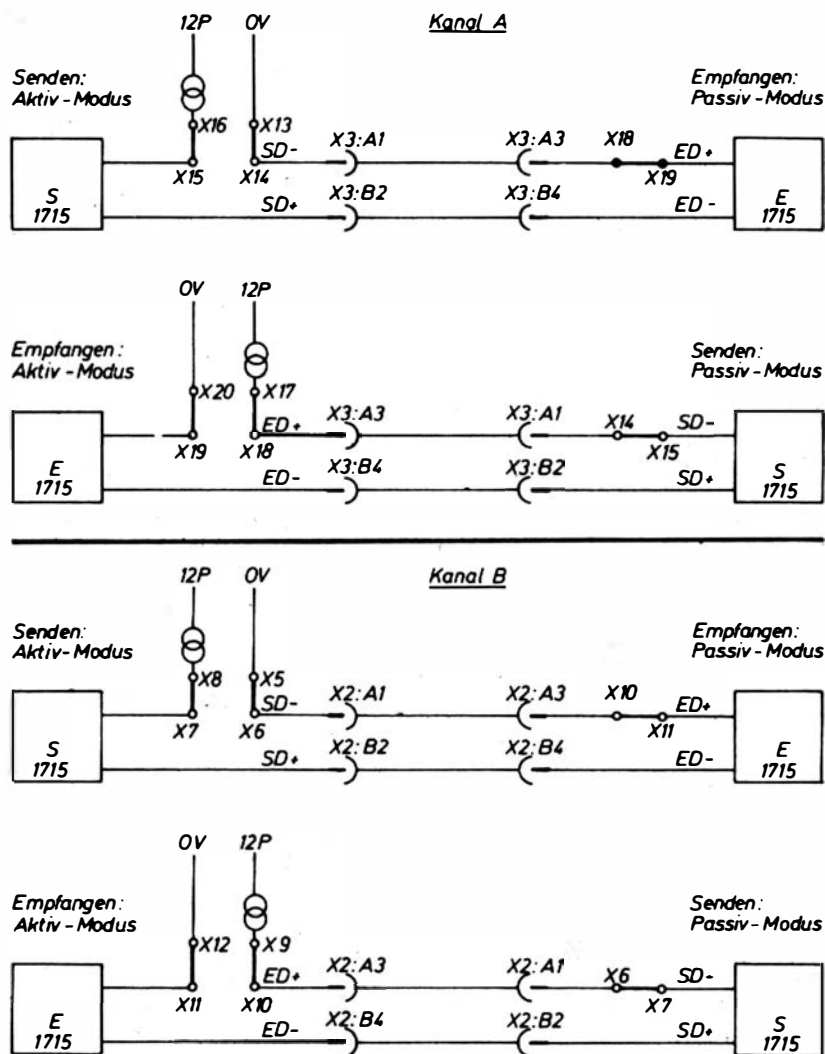


Abbildung 1.13
Zusammenschaltungsmöglichkeiten (Beispiel)

<i>Kontakt</i>	<i>Leitung</i>
A01	SD-
B02	SD+
A03	ED+
B04	ED-
A05	Schirm

Die Zusammenschaltungsmöglichkeiten sind in der Abbildung 1.13 dargestellt.

Die Konstantstromquellen sind für den Betriebsfall einer maximalen Leitungslänge von 500 m (200 Ohm) konzipiert. Sie liefern einen Strom von 15 bis 25 mA bei einer Spannung (gegen Masse) von 2 V bis 9 V. Dieser Spannungsabfall entsteht durch die Summe der Spannungen über Sendestufe, Kabelwiderstand und Empfängerstufe. Im Leerlauf bzw. „0“-Zustand stellt sich eine Spannung von ca. 12 V ein. Die Schaltungen in der Stromschleife (einschließlich der Stromquellen) sind so dimensioniert, daß beim Vertauschen von Leitungen keine Schäden entstehen.

1.4.3. Externe Folienspeichereinheit für 8-Zoll-Disketten

Diese Einheit wurde speziell zur Erweiterung der Gesamtspeicherkapazität des PC 1715 entwickelt. Sie enthält zwei 8-Zoll-Standardlaufwerke. Vorrangig werden Laufwerke des Typs MF 6400 eingesetzt. Durch Anschluß dieser externen Folienspeichereinheit erweitert sich die Speicherkapazität des PC 1715 um 600 KByte (Betriebsart FM) bzw. 1200 KByte (Betriebsart MFM). Die als Beistellgerät gestaltete Einheit besteht im wesentlichen aus folgenden Baugruppen:

- Stromversorgung 5 V und 12 V,
- Verteilereinheit und
- zwei 8-Zoll-Standardlaufwerke.

Das Beistellgerät wird separat an die Stromversorgung (220 V/50 Hz) angeschlossen.

Über eine fest am Beistellgerät angebrachte Interfaceleitung erfolgt der Anschluß an der Rückseite der Systemeinheit unter Ausnutzung des mit FD (Floppy-Disk) gekennzeichneten Steckverbinders. Die Ansteuerung der Laufwerke übernimmt die FD-Ansteuersteckeinheit, die im Grundgerät des PC 1715 untergebracht ist. Als Datenträger fungiert die 8-Zoll-Standard-Diskette.

Charakteristische technische Daten sind in der Tabelle 1.16 zusammengestellt.

Tabelle 1.16
Technisches Datenblatt zur externen Folienspeichereinheit für 8-Zoll-Disketten

Merkmalsbezeichnung	Merkmalswerte
Einsatzbedingungen	Temperaturbereich + 10 °C bis + 35 °C relative Luftfeuchte 40... 80 %
Lagerungsbedingungen	zulässiger Staubgehalt der Luft max. 1 mg/m ³ Lagertemperatur + 5 °C bis + 35 °C relative Luftfeuchte 85 % Lagerungsdauer max. 1 Jahr
Transportbedingungen	Transporttemperatur - 40 °C bis + 50 °C relative Luftfeuchte 95 % Luftdruck 84 kPa ... 107 kPa
Abmessungen	Breite: 340 mm Tiefe: 450 mm Höhe: 282 mm
Gewicht	24 kg
Versorgungsspannung	220 V + 10 % / - 15 %; 50 ± 1 Hz
Leistungsaufnahme	320 W
Schutzgrad	IP 20
Speicherkapazität	max. 2 × 600 KByte nutzbar

Tabelle 1.17
Technisches Datenblatt zur externen Folienspeichereinheit für 5,25-Zoll-Disketten

Merkmalsbezeichnung	Merkmalswerte
Einsatzbedingungen	Temperaturbereich + 10 °C bis + 35 °C relative Luftfeuchte: 40... 80 %
Lagerungsbedingungen	zulässiger Staubgehalt der Luft max. 1 mg/m ³ Lagertemperatur + 5 °C bis + 35 °C relative Luftfeuchte 85 % Lagerungsdauer max. 1 Jahr
Transportbedingungen	Transporttemperatur - 40 °C bis + 50 °C relative Luftfeuchte 95 % Luftdruck 84 kPa ... 107 kPa
Abmessungen	Breite: 174 mm Tiefe: 400 mm Höhe: 215 mm
Gewicht	10 kg
Versorgungsspannung	220 V + 10 % / - 15 %; 50 ± 1 Hz
Leistungsaufnahme	190 W
Schutzgrad	IP 20
Speicherkapazität	max. 2 × 148 KByte nutzbar

1.4.4. Externe Folienspeichereinheit für 5,25-Zoll-Disketten

Diese Einheit wurde speziell zur Erweiterung der Gesamtspeicherkapazität des PC 1715 entwickelt. Da sie über zwei Laufwerke K 5600.10 (MFS 1.2) verfügt, erweitert sich die Speicherkapazität des Systems um 2×148 KByte (je verwendete Diskette). Die als Beistellgerät gestaltete Einheit besteht im wesentlichen aus folgenden Baugruppen:

- Stromversorgung 5 V und 12 V,
- Verteilereinheit und
- zwei 5,25-Zoll-Standard-Minifolienspeicher-Laufwerke.

Das Beistellgerät wird separat an die Stromversorgung (220 V/50 Hz) angeschlossen.

Über eine fest am Beistellgerät angebrachte Interfaceleitung erfolgt der Anschluß an die Rückseite der Systemeinheit unter Nutzung des mit „Floppy-Disk“ gekennzeichneten Steckverbinders. Die Ansteuerung der Laufwerke übernimmt die FD-Ansteuersteckeinheit im PC 1715. Als Datenträger fungiert die 5,25-Zoll-Standard-Diskette. Die in Tabelle 1.17 angegebenen technischen Daten charakterisieren dieses Beistellgerät.

1.5. Die Drucker des PC 1715

Entsprechend dem universellen Einsatzkonzept für den PC 1715 stehen dem Anwender mehrere für seinen speziellen Einsatzfall zu spezifizierende Drucker zur Verfügung.

Tabelle 1.18
Einstellung der Paritätskontrolle

Paritätskontrolle	K 6313/14 DIL-Schalter		1152/Typ 257 DIL-Schalter	
	16-1	16-2	5-6	5-7
ungerade	0	0	0	1
ohne Prüfung*	1	0	1	0
gerade	0	1	0	0
ohne Prüfung*	1	1	1	1

1 = ON, 0 = OFF

* für Betrieb mit PC 1715



Abbildung 1.14
Matrixdrucker K 6313 in der Ausführung
„Schreibwalze mit Stachelsystem und Handrad“

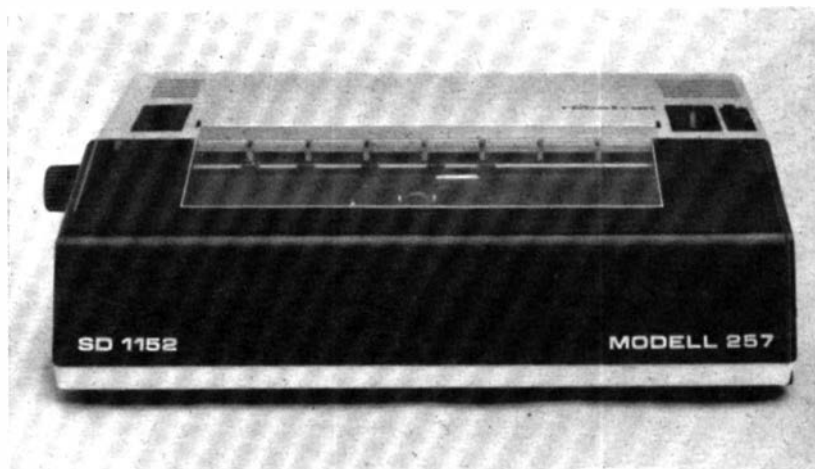


Abbildung 1.15
Typenradrunder 1152/Typ 257

Tabelle 1.19
Einstellung der Baudrate

Baudrate	K 6313/14 DIL-Schalter			1152/Typ 257 DIL-Schalter			
	14-2	15-1	15-2	5-3	5-2	5-1	5-0
9600*	0	0	0	1	1	0	0
4800	0	1	0	1	0	1	1
3200	-	-	-	1	0	1	0
2400	1	1	0	1	0	0	1
1600	-	-	-	1	0	0	0
1200	0	0	1	0	1	1	1
800	-	-	-	0	1	1	0
600	1	0	1	0	1	0	1
300	0	1	1	0	1	0	0
200	1	0	0	-	-	-	-
150	1	1	1	0	0	1	1
100	-	-	-	0	0	1	0
75	-	-	-	0	0	0	1
50	-	-	-	0	0	0	0

1 = ON, 0 = OFF

* für Betrieb mit PC 1715

Tabelle 1.20
Einstellung von Übertragungs- und Betriebsarten

	K 6313/14	1152/Typ 257
Übertragungsprotokoll		
- DTR*	17-1 = OFF	4-1 = OFF
- XON/XOFF	17-1 = ON	4-1 = ON
		4-0 = OFF
Betriebsart XON/XOFF		
- Duplex*	17-2 = OFF	
- Halbduplex	17-2 = ON	
Datenbits		
- 7 bit*	18-1 = OFF	5-4 = OFF
- 8 bit	18-1 = ON	5-4 = ON
Stoppbits		
- 1 bit*	18-2 = OFF	5-5 = OFF
- 2 bit	18-2 = ON	5-5 = ON

* für Betrieb mit PC 1715



Abbildung 1.16
Schalterdrucker K 6316

Für Hardcopy-Ausgabe und Grafik-Darstellung, auch für Nebenbeitextverarbeitung in „near letter quality“ sind die Nadeldrucker K 6313 (Abbildung 1.14) und K 6314 geeignet.

Für die Textverarbeitung in „letter-quality“ steht der leistungsfähige Typenraddrucker 1152/Typ 257 zur Verfügung.

Für off-line- und Terminalbetrieb mit schalterplatzspezifischer Belegverarbeitung ist der Schalterdrucker K 6316 geeignet.

Grundsätzlich kann der PC 1715 neben dem Systemausgabedrucker, der über den „Printer“-Stecker gekoppelt wird, auch mit einem zweiten (oder auch dritten) Drucker (über einen Interface-Kanal gekoppelt) betrieben werden. Damit ist die Möglichkeit gegeben, unterschiedliche Druckertypen gleichzeitig anzuschließen, wobei über die Systeminitialisierung eine entsprechende Kanalfestlegung zu treffen ist und über das Anwenderprogramm die Druckerauswahl vorgenommen werden kann.

Die Spezifizierung der Drucker bezüglich des Zeichensatzes oder der Typenscheibe ist unter Beachtung der Tastaturvorlage des PC 1715 vorzunehmen. Wird der ausgewählte Drucker über den „Printer“-Stecker gekoppelt, ist eine V.24-Gerätevariante des Druckers zu spezifizieren und diese auf folgende Übertragungsparameter einzustellen (DTR-Protokoll):

Baudrate	9600
Zahl der Datenbit/Byte	8
Parität	ohne
Zahl der Stoppbit	1
Übertragungsart	duplex

Die Einstellungen der Baudrate, der Paritätskontrolle sowie von Übertragungs- und Betriebsarten sind den Tabellen 1.18, 1.19 und 1.20 zu entnehmen.

Die mit einer V.24-Schnittstelle ausgelieferte Variante des K 6316 wird bei der Auslieferung auf die oben genannten Parameter eingestellt.

Bei einer Kopplung über Schnittstellen der Interfaceerweiterungssteckeinheit sind die Drucker entsprechend der gewählten Spezifikation der Erweiterungseinheit zu wählen und die Parameter der Übertragungssteuerung durch Einstellung der DIL-Schalter und/oder durch Veränderung der Installationsparameter des PC 1715 mit Hilfe des Programms INSTSCP aufeinander abzustimmen.

1.5.1. Die Matrix-Nadeldrucker K 6313 und K 6314

Die Drucker K 6313 (mit einer Zeilenbreite von 80 Zeichen Normalschrift) und K 6314 (mit einer Zeilenbreite von 136 Zeichen Normalschrift) arbeiten mit einer Druckgeschwindigkeit von 100 Zeichen/s; sie sind mit einem Nadeldruckkopf (neun Nadeln) ausgestattet. Bedruckt werden können ein Original und zwei Durchschläge. Die Drucker bieten gute anwendungstechnische Leistungen durch einfache konstruktiv-technologische Gestaltung, geringe Masse, kleine Abmessungen, gute Druckbildqualität, hohe Zuverlässigkeit, lange Wartungsabstände, niedrige Geräuschpegel. Eine vielseitige Formular-technik ermöglicht die Anwendung von Journalrollenpapier, Journalrollenpapier mit Randlochung, randgelochtem Endlosformular als Leporello und die Einzelblattverarbeitung. Die für den Anwendungszweck benötigte Ausführung ist bei der Bestellung zu spezifizieren.

Speziell für den K 6313 existiert die universell einsetzbare Ausführungsvariante mit Schreibwalze und Handrad. Mit diesem Universaldrucker können sowohl Endlospapier von der Rolle, randgelochtes Leporellopapier sowie Einzelblätter verarbeitet werden.

Für die ausschließliche Verarbeitung von Faltpapier mit Randlochung existiert sowohl für den K 6313 als auch den K 6314 die Ausführungsvariante Matrixdrucker mit Papiertraktoren.

Das Gerät wird über den Netzschalter an der Rückseite ein- bzw. ausgeschaltet. Nach dem Einschalten leuchtet die Glimmlampe innerhalb des Schal-

ters. Der Druckwagen bewegt sich in seine Grundstellung an den linken Rand (Synchronisation). Der Drucker befindet sich automatisch im „ON-LINE-Zustand“ (falls keine Papierendemeldung oder kein technischer Fehler vorliegen) und ist zum Datenempfang bereit.

An der rechten Vorderseite des Gerätes befinden sich die Bedientasten und Kontrolleuchtanzeigen.

Ausgehend vom Anwendungsspektrum bieten die Drucker K 6313 und K 6314 eine Vielfalt von softwaregesteuerten Ausgabemöglichkeiten (vgl. Abbildung 1.17).

Die Steuerung erfolgt auf der Grundlage des ASCII-Codes. Buchstaben, Zahlen und spezielle Symbole werden codiert von 32 bis 126 und von 160 bis 254.

Spezielle Funktionen werden mittels Escape-Folgen zum Drucker übertragen. Eine Escape-Folge besteht aus dem Escape-Code (27) und alphanumerischen oder symbolischen Zeichen (vgl. Übersicht B).

Wurde eine gesamte Zeile von Druckdaten (einschließlich Leerzeichen) übertragen und das folgende Zeichen ist gültig und druckbar, dann wird der Inhalt des Druckpuffers ausgedruckt.

Spezielle Befehle im Einzelpunktmodus und die unterschiedlich setzbaren Zeilenabstände ermöglichen einen leistungsfähigen Grafikdruck. (Vgl. auch die Tabellen 7.1 und 7.2.)

Die charakteristischen technischen Daten der Matrix-Nadeldrucker K 6313 und K 6314 sind in der Tabelle 1.21 zusammengestellt.

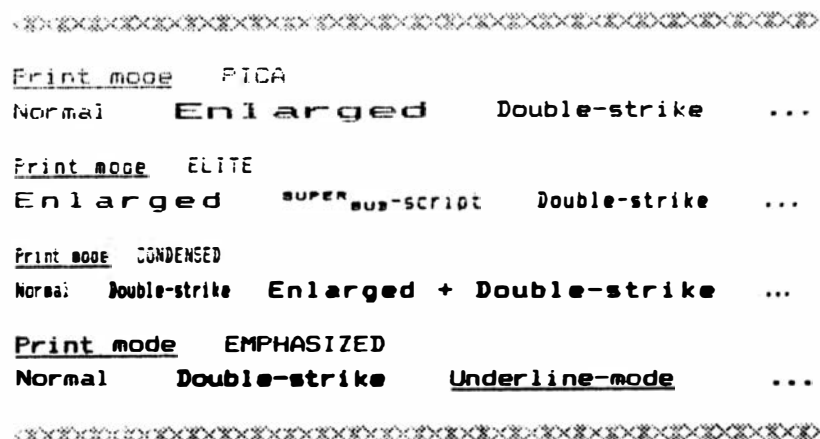


Abbildung 1.17

Die Drucker K 6313 und K 6314 drucken Text
 in unterschiedlicher Schriftgestaltung und Grafik

Übersicht B

Befehlskodierung

Nadeldrucker K 6313 und K 6314

SO	- Einschalten Sperrschrift
ESC SO	- Einschalten Sperrschrift
SI	- Einschalten komprimierte Schrift
ESC SI	- Einschalten komprimierte Schrift
DC 2	- Ausschalten komprimierte Schrift
DC 4	- Ausschalten Sperrschrift
ESC E	- Einschalten Fettschrift
ESC F	- Ausschalten Fettschrift
ESC G	- Einschalten Doppeldruck
ESC H	- Ausschalten Doppeldruck
ESC M	- Einschalten ELITE-Schrift
ESC P	- Einschalten PICA-Schrift
ESC S	- Einschalten Hoch- oder Tiefschrift
ESC T	- Ausschalten Hoch- oder Tiefschrift
ESC W	- Ein- oder Ausschalten Sperrschrift
ESC !	- Auswahl Schriftart
ESC -	- Ein- oder Ausschalten Unterstreichmodus
ESC K	- Einzelpunktmodus 480 Sprossen je 8 Zoll
ESC L	- Einzelpunktmodus 960 Sprossen je 8 Zoll; 6 Zoll je s
ESC Y	- Einzelpunktmodus 960 Sprossen je 8 Zoll; 10 Zoll je s
ESC Z	- Einzelpunktmodus 1920 Sprossen je 8 Zoll
ESC *	- Auswahl Einzelpunktmodus
ESC 0	- Setzen des Zeilenabstandes auf $\frac{1}{8}$ "
ESC 1	- Setzen des Zeilenabstandes auf $\frac{7}{72}$ "
ESC 2	- Setzen des Zeilenabstandes auf $\frac{1}{6}$ "
ESC 3	- Setzen des Zeilenabstandes auf $\frac{n}{216}$ "
ESC A	- Setzen des Zeilenabstandes auf $\frac{n}{72}$ "
LF	- Zeilenvorschub
VT	- Vertikaltabulation
FF	- Formularvorschub
ESC J	- Ausführung eines Zeilenvorschubs von $\frac{n}{216}$ "
ESC j	- Ausführung eines Zeilenvorschubs rückwärts von $\frac{n}{216}$ "
ESC N	- Einstellen Formularendezeile
ESC O	- Ausschalten Formularendezeile
NUL	- Ende Tabulatorsetzung
BS	- Rückschritt
HT	- Horizontal-Tabulation
CR	- Druckposition auf Zeilenanfang setzen
ESC B	- Setzen von Vertikaltabulations-Marken
ESC C	- Einstellen Formularlänge (n Zeilen)
ESC C0	- Einstellen Formularlänge (n Zoll)
ESC D	- Setzen von Horizontaltabulations-Marken
ESC Q	- Einstellen Zeichenanzahl pro Zeile
CAN	- Löschen des Druckpuffers

DEL	- Löschen des letzten Zeichens im Druckpuffer
BEL	- Summer
ESC 8	- Ausschalten Papierendkontrolle
ESC 9	- Einschalten Papierendkontrolle
ESC R	- Auswahl Zeichensatz
ESC U	- Einstellen uni- oder bidirektionaler Druck
ESC @	- Druckerinitialisierung
ESC <	- Einstellen unidirektionaler Druck (eine Zeile links beginnend)
ESC x	- Ein- oder Ausschalten NLQ-Modus

Tabelle 1.21

Technische Datenblätter zu den Matrix-Nadeldruckern K 6313 und K 6314

Mechanische Daten

Druckprinzip	serieller Matrixdruck mit Nadeldrucksystem
Druckgeschwindigkeit	100 Zeichen/Sekunde
Druckrichtung	Bidirektional mit logischer Druckwegoptimierung Unidirektional mit Grafikmodus oder wenn programmiert
Anzahl der Nadeln	9
Text-Modus	
- Zeichenraster	9×9
- Zeichengröße	
· Normalschrift	2,1 mm×3,1 mm
· Elite	1,5 mm×3,1 mm
· komprimierte Schrift	1,1 mm×3,1 mm
(Schriftarten über DIL-Schalter oder über Programmierung einstellbar)	

	Zeichen/Zoll	Zeichen/Zeile K 6313	Zeichen/Zeile K 6314
Normalschrift	10	80	136
Fettschrift	10	80	136
Sperrschrift	5	40	68
Elite	12	96	163
Elite Sperrschrift	6	48	81
komprimierte Schrift	17	132	233
komprimierte Sperrschrift	8,5	66	116

Grafik-Modus

- Grundteilung des Punktrasters vertikal	72 Pkt./Zoll	
- Grundteilung des Punktrasters horizontal	480 Pkt./8 Zoll	720 Pkt./8 Zoll
	576 Pkt./8 Zoll	960 Pkt./8 Zoll
	640 Pkt./8 Zoll	1920 Pkt./8 Zoll

Papierverarbeitung		
- Zeilenvorschub	$\frac{1}{6}$ Zoll (4,23 mm) oder $\frac{1}{8}$ Zoll (3,18 mm) einstellbar über DIL-Schalter	
- Transporteinrichtung	$\frac{1}{6}$ Zoll, $\frac{1}{8}$ Zoll, $\frac{7}{72}$ Zoll, $\frac{5}{72}$ Zoll, $\frac{3}{216}$ Zoll programmierbar	
- Formulararten	vorwärts rückwärts in Einzelschritten über Taste rückwärts bei Traktorsystemen programmierbar	
	Walze	Traktor
· Endlospapier Leporello	230–252 mm (9–10 Zoll)	75–265 mm (3–10,4 Zoll) max. 410 mm (K 6314)
· Endlospapier Rolle	85–216 mm (3,4–8,5 Zoll) 85–375 mm (K 6314)	
Rollendurchmesser	max. 128 mm	
Rollenkerndurchmesser	bis 16 mm	
· Einzelblatt	85–216 mm (3,4–8,5 Zoll)	
Papierhöhe	100–305 mm	
- Kopien	1 Original + 2 Durchschläge (Die Gesamtstärke darf 0,3 mm nicht überschreiten; Flächengewicht max. 80 g/m ²)	
Farbband		
Standard	Farbbandkassette FBK 1 Farbbandkassette Tally 1000	
Zusatz	Farbband 13 mm mit Umschaltösen	
Tränkung	zart (bis 24 %)	
Spulendurchmesser	54 mm	
Abmessungen	K 6313	K 6314
Breite	370 mm	520 mm
Tiefe	280 mm	280 mm
Höhe	130 mm	130 mm
Masse	7 kg	9 kg
Elektrische Daten		
Netzspannung	220 V + 10 %/–15 % 110 V + 10 %/–15 %	
Netzfrequenz	47 bis 63 Hz	
Leistungsaufnahme	max. 70 VA	
Betriebsbedingungen		
Temperatur	10 °C bis 35 °C	
relative Luftfeuchte	40 bis 90 %	
Schalldruckpegel	max. 58,5 db (AS)	

1.5.2. Der Typenraddrucker 1152/Typ 257

Der Typenraddrucker 1152/Typ 257 (siehe Abbildung 1.15) ist ein Ausgabe-
gerät für alphanumerische Informationen in repräsentativer Schriftbildquali-
tät für die Textverarbeitung (vgl. Abbildung 1.18).

Der Drucker ist mit unterschiedlichen Formulartransporteinrichtungen aus-
gestattet. Es können Rollenpapier, randgelochtes Endlospapier als Leporello
sowie Einzelbelege bei manuellem Einlegen über den Hintereinzug der
Schreibwalze verarbeitet werden. Wesentliche Leistungsmerkmale sind:

- hohe Ausgabeleistung bis zu 45 Zeichen/s bei ausgezeichneter Schrift-
bildqualität,
- hohe Zuverlässigkeit und Servicefreundlichkeit,
- umfangreiches Typenscheibenangebot,
- Druck vorwärts/rückwärts (1 Original und max. 5 Kopien),
- Zeilenschaltung vorwärts/rückwärts,
- internes Testprogramm,
- Verwendung marktüblicher Farbbandkassetten.

Der Typenträger ist ein leicht auswechselbares Typenrad mit 96 Druckzei-
chen. Die Typenscheiben werden in unterschiedlichen Schriftarten bereitge-
stellt.

Neben den Schriftarten unterscheiden sich die Typenscheiben in dem zur
Verfügung stehenden Zeichenvorrat. Dieser Zeichenvorrat ist in einer Reihe
von Zeichensätzen standardisiert. Der aufgeprägte Zeichensatz der Typen-
scheibe muß mit der Tastaturbelegung des PC 1715 und mit dem eingestell-
ten Zeichensatz übereinstimmen.

Als Farbbandtypen können Mehrfachcarbonband (Multi-Strike), Gewebe-
band oder Einfachcarbonband in kassettierter Form verwendet werden. Der
Drucker bietet die Möglichkeit, den Vorschub des Farbbandes pro Druckzei-
chen einzustellen. Für Einfachcarbonbänder ist die Stellung SINGLE zu
wählen, für Multi-Strike und Nylon-Farbbänder reicht ein kurzer Vorschub
aus, der in Stellung MULTI erreicht wird.

Die Papierzuführung kann auf zwei verschiedene Arten erfolgen, als Einzel-
blatt halbautomatisch zugeführt und als Endlospapier. Die Einzelblattverar-
beitung erfolgt über den Hintereinzug ähnlich wie bei einer Schreibma-
schine, wobei durch eine einfache (optionale) Zusatzeinrichtung das
Einlegen und Ausrichten des Blattes unterstützt wird und über Tastendruck
(Taste IN/OUT) der Blatteinzug vorgenommen wird.

Für den Betrieb des Druckers mit Endlospapier (Leporello) wird die mitgelie-
ferte Traktorbaugruppe (innerhalb des Druckers) eingesetzt. Das randge-
lochte Endlospapier wird durch das Traktorpaar beidseitig sowohl zum Ein-

Beispiel: Typendrucker **robotron SD 1152, Modell 257**

VVB Robotron Elektronik Zella-Mehlis
Gen. Meier

Str. der Anlaufschichten 63/69
PSI 96

Zella-Mehlis

6060

FF/2607

3845

12.10.86

Software/Typendrucker

Werter Gen. Meier !

Wie vereinbart, sende ich Ihnen in der Anlage eine Zusammenstellung von Hinweisen und Ergänzungen zu unserer Software, wie sie sich aus den bisher gesammelten Erfahrungen der Anwendung ergeben haben.

Wir haben unserem Absatzbereich vorgeschlagen, dieses Material als Systeminformation an unsere Kunden auszugeben. Es wäre günstig, wenn sie gleichermaßen verfahren würden.

Außerdem möchten wir darauf hinweisen, daß der Seriendrucker **robotron SD 1152, Modell 257** mit seiner ausgezeichneten Schriftqualität, der Möglichkeit der Einzelblattverarbeitung und vielfältiger Schriftgestaltung für diesen Anwendungsfall hervorragend geeignet ist, da neben dem geschlossenen Schriftcharakter auch Formelausdrücke und bestimmte graphische Ausdrucksmöglichkeiten für ihre Dokumentationszwecke möglich sind.

Beispiele:

I	I		I
I	I		I
I	I	$x = \frac{1 * (t^2) - (t*1) * t}{n*(t^2) - (t)^2}$	I
I	I		I
I	I		I
I	I	$y = \frac{n * (t*1) - (1) * (t)}{n * (t^2) - (t)^2}$	I
I	I		I
I	I		I

Mit sozialistischem Gruß

Abbildung 1.18

Brief-Qualitätsdruck des Typendruckers 1152/Typ 257

Übersicht C
Befehlskodierung
Typenraddrucker 1152/Typ 257

LF	LINEFEED
ESC LF	NEGATIV LINEFEED
ESC U	HALFLINE
ESC D	NEGATIV HALFLINE
ESC RS m	LINESPACING
ESC FF n	LINES PER PAGE (1/6")
FF	FORMFEED
ESC T	SET TOP MARGIN
ESC L	SET BOTTOM MARGIN
ESC C	CLEAR TOP AND BOTTOM MARGIN
ESC US m	CHARACTER SPACING
ESC S	SWITCH CHARACTER SPACING
SP	SPACE
BS	BACKSPACE
CR	CARRIAGE RETURN
ESC 5	FORWARD PRINT
ESC 6	BACKWARD PRINT
ESC 9	SET LEFT MARGIN
ESC 0	SET RIGHT MARGIN
ESC P	PROPORTIONAL SPACE ON
ESC Q	PROPORTIONAL SPACE OFF
ESC E	UNDERLINE ON
ESC R	UNDERLINE OFF
ESC A	BOLD PRINT ON
ESC O	BOLD PRINT ON
ESC W	SHADOW PRINT ON
ESC B	BOLD/SHADOW PRINT OFF
ESC X	CLEAR WORDPROCESSING MODE
ESC EM NUL	CHARACTER SET 1
ESC EM SOH	CHARACTER SET 2
ESC Y	95. Druckzeichen
ESC Z	96. Druckzeichen
ESC I	SET HORIZONTAL TAB STOP
HT	HORIZONTAL TAB
ESC HT n	ABSOLUTE HORIZONTALTABULATION
ESC -	SET VERTICAL TAB STOP
VT	VERTICAL TAB
ESC VT n	ABSOLUTE VERTICALTABULATION
ESC 2	CLEAR ALL HORIZONTAL AND VERTICAL TAB
ESC 8	CLEAR HORIZONTAL TAB
ESC SUB I	RESET
ESC CR P	RESET

lauf als auch zum Auslauf geführt. Der Drucker hat für die Papierendeerkennung eine Reflexlichtschranke, die in der jeweiligen Druckposition unterhalb der Walze das Vorhandensein von Papier prüft.

Der Drucker bietet entsprechend dem Anwendungszweck eine Vielfalt von softwaregesteuerten Ausgabemöglichkeiten (vgl. Übersicht C) und unterscheidet zwischen den Verarbeitungsarten „Serialmode“ (sofortige Verarbeitung ohne Druckwegoptimierung) und „Linemode“ (Druckzeilenaufbereitung).

Die in der Tabelle 1.22 zusammengestellten technischen Daten des Typenraddruckers charakterisieren dieses Gerät.

Tabelle 1.22

Technisches Datenblatt zum Typenraddrucker 1152/Typ 257

Mechanische Daten

Druckprinzip	serieller Typenraddruck
Druckgeschwindigkeit	96 Zeichen je Typenscheibe 55 Zeichen/s (3A-Methode) 45 Zeichen/s (Shannon)
Druckbreite	10 CPI 132 Zeichen 12 CPI 158 Zeichen 15 CPI 197 Zeichen
Schriftqualität	Proportionalschrift letter-quality (breites Angebot von über 100 Typenscheiben verschiedener Schriftarten und Länderspezifikationen)
kleinste Schrittweiten	– horizontal (einstellbar) $\frac{1}{120}$ Zoll, $\frac{1}{60}$ Zoll – vertikal (einstellbar) $\frac{1}{144}$ Zoll, $\frac{1}{72}$ Zoll, $\frac{1}{48}$ Zoll
Tabulationsgeschwindigkeit	20 Zoll/s
Papierdurchlaßbreite	Leporello 390 mm Einzelblatt 380 mm
Druckrichtung	bidirektional
Formularrandbestimmung	einstellbar – linker Rand – rechter Rand – beide Ränder – keine Vermessung
Farbband	Gewebeband 6,35 mm endlos Carbonbänder als Multistrike oder Singlestrike
Bedientasten	5 (ON/OFF, RESET, LF/FF, IN/OUT, INTENSITY)
Sensoren zur Drucküberwachung	Farbbandende Papierende

	Druckerhaube offen
	interne Fehlererkennung und Behandlung
Interface-Module	– seriell RS 232 C (V.24) IFSS (20 mA Stromschleife)
	– parallel Centronics
Pufferspeicher	2,8 KByte (Standard) 14,8 KByte (Option)
Zeichensätze	spezifiziert nach Ländern und Zeichenbelegung zwei frei wählbare Zeichensätze in jedem Gerät vorhanden (leicht veränderbar)
Befehlssatz	Diablo-kompatibel ISO
leistungsfähige Modes	Schattendruck Doppeldruck Unterstreichen Setzen von vertikalen und horizontalen Tabulatoren und Marken Formatsteuerung
Papierzuführungsarten	– Endlospapier Leporello Journalrolle – Einzelblatt halbautomatisch
Masse	25 kg
Abmessungen	555 mm (Länge) 435 mm (Tiefe) 165 mm (Höhe)

Elektrische Daten

Leistungsaufnahme	100 VA
Netzanschluß	220 V + 10 % / - 15 % (Standard) 110 V + 10 % / - 15 % (Option)

Betriebsbedingungen

Temperatur	10 °C bis 40 °C
relative Luftfeuchte	40 bis 90 %
Schalldruckpegel	60 dB

1.5.3. Der Schalterdrucker K 6316

Die technische Konzeption des Schalterdruckers K 6316 (siehe Abbildung 1.16) ist so gehalten, daß sie vielfältigen kundenspezifischen Anforderungen gerecht werden kann.

Der vor allem für den Bereich der Formulartechnik sehr variabel gestaltete Drucker kommt als Auftischgerät vorwiegend an Terminalarbeitsplätzen zum Einsatz.

Grundbestandteil ist ein 80stelliger Matrix-Nadeldrucker mit einer Druckgeschwindigkeit von 100 Zeichen/Sekunde. Die Informationen werden zeitoptimal vorwärts und rückwärts ausgedruckt.

Bedingt durch eine Vielfalt von Formulartechnik-Baugruppen, die jeweils mit dem Drucker kombiniert werden können, stellt der Schalterdrucker selbst eine Baureihe dar. Durch die verschiedenen Kombinationen entstehen Gerätevarianten, die den unterschiedlichsten anwendungstechnischen Forderungen gerecht werden. Alle Varianten besitzen eine programmgesteuerte Trenneinrichtung und eine Vorsteckeinrichtung für Formulare, Karten und Belegsätze.

Bedientasten sind bei diesem Drucker nur für Servicezwecke (z. B. Einlegen des Endospapiers) vorgesehen.

Eine Anwendung innerhalb normaler Bedienroutinen bei Rechnersteuerung wird nicht empfohlen, da in diesem Falle Störungen des Betriebsablaufes zwischen Gesamtsystem und Drucker nicht auszuschließen sind.

Die charakteristischen technischen Daten des Schalterdruckers K 6316 sind als technisches Datenblatt in Tabelle 1.23 zusammengestellt.

Tabelle 1.23
Technisches Datenblatt zum Schalterdrucker K 6316

Mechanische Daten	
Druckposition/Zeile	80 ($\frac{1}{10}$ Zoll) 100 ($\frac{1}{12,5}$ Zoll) 120 ($\frac{1}{15}$ Zoll)
Druckfrequenz	100 Zeichen/s
Druckrichtung	vorwärts/rückwärts mit Druckwegoptimierung
Abstand der Druckpositionen	2,54 mm (10 Zeichen/Zoll) 2,03 mm (12,5 Zeichen/Zoll) 1,69 mm (15 Zeichen/Zoll)
Grundteilung des Zeichenrasters	
- horizontal	50 Pkt./Zoll (0,508 mm bei $\frac{1}{10}$ Zollteilung)
- vertikal	72 Pkt./Zoll (0,352 mm)
Matrix zur Zeichendarstellung (vertikal \times horizontal)	9 \times 7 (Halbschrittmatrix)

Zeichenbreite	1,9 mm ($1/10$ Zollteilung)
Zeichenhöhe	2,5 mm
Zeichenhöhe mit Unterlängen	3,2 mm
Schriftarten	Normalschrift Breitschrift Schrägschrift
Zeichenvorrat	95 158 (Option)
Zeichenspeicher	160 Byte
Papiervorschub	
- Standardzeilenvorschub	$1/6$ Zoll (4,23 mm)
- Vertikaltabulation (erfolgt über programmierbaren Vorwärts-/Rückwärts-Grund- schritt)	$1/12$ Zoll (2,12 mm)
- Transportrichtung (unter Beachtung der aktuellen Betriebsart)	
- Formatsteuerung über Interface	beliebig (bis 255 Grundschr. itte)
- Formateinstellung nach Synchronisation	144 Grundschr. itte (12 Zoll)
Papierdurchlaßdicke	in Stufen einstellbar von 0,2 mm bis 0,6 mm
Papierdickenunterschied entlang der Zeile	$\leq 0,25$ mm
Anzahl der Nutzen	≤ 3 (Papier mit Flächengewicht 40 g/m ² bis 60 g/m ²)
Transporteinrichtung für Rollenpapier	
Rollenbreite ohne Transportloch	85...210 mm
Rollenbreite mit Transportloch	130...250 mm
Anzahl der Nutzen	≤ 2
Aufwickleinrichtung für Rollenpapier	
Rollenbreite	85...210 mm
Außendurchmesser der aufgewickelten Journalrolle	max. 120 mm
-Option	Verschlüßeinrichtung (Sicherheitsgrad 4)
Anzahl der Lagen	1
Transporteinrichtung für Leporellopapier	
Stapelbreite	130...250 mm
Flächengewicht	55...80 g/m ²
Anzahl der Nutzen	≤ 2
Vorsteckeinrichtung für Einzelformulare, Karten und Formularsätze	
Formulartransport	getrenntes Transportsystem
- Vorstecken	manuell
- Einziehen und Positionieren	programmgesteuert
· Vertikaltabulation	$1/12$ Zoll (2,12 mm)

(erfolgt über programmierbaren Grundschrift)

- Auswurf **programmgesteuert**
- Transportrichtung **vorwärts/rückwärts**

Abmessungen der Formulare und Karten

Höhe	74...297 mm
Breite	74...210 mm
Belegformate	Höhe/Breite $\leq 1,5$
Flächengewichte	
Einzelformulare	40...120 g/m ²
Mehrlagenformulare geheftet	40... 60 g/m ² (Kohlepapier 20 g/m ²)
Karten	120...170 g/m ²
Anzahl der Nutzen	≤ 3

Programmgesteuerte Trenneinrichtung (horizontal)

Trennbreite	85...250 mm
Beleghöhe	mindestens 38 mm (9 Zeilen)
Anzahl der Nutzen	1

(Schneiden von zweilagigem Papier ohne Kohlepapier ist möglich. Bei der Verarbeitung von Rollenpapier ist ein Hülseninnendurchmesser von 30 mm erforderlich.)

Abmessungen

Tiefe	410 mm
Breite	440 mm
Höhe	210 mm für Leporello 226 mm für Rolle mit Aufwickeln (ohne aufsetzbare Einrichtungen für Belegführung und -ablage)
Masse	max. 17,5 kg

Elektrische Betriebsbedingungen

Netzspannung	220 V + 10 % - 15 %
Netzfrequenz	47...63 Hz
Leistungsaufnahme	max. 100 VA
Schutzart	Schutzleiter

Betriebsbedingungen

Temperatur	10 °C bis 40 °C
relative Luftfeuchte	40 bis 90 %
Schalldruckpegel	60 dB

1.6. Technisches Datenblatt zum PC 1715

Zum Abschluß der Hardwarebeschreibung werden die wichtigsten technischen Daten für den PC 1715 zusammengestellt, die beim Einsatz des Gerätes zu berücksichtigen sind.

Lagerung (ohne Disketten)

- Temperatur: - 40 bis + 50 Grad C
- relative Luftfeuchte: max. 85 %
- max. Lagerdauer: 1 Jahr

Transport (ohne Disketten)

- Temperatur: - 40 bis + 50 Grad C
- relative Luftfeuchte: max. 95 %
- atmosphärischer Druck: 84 KPa ... 107 KPa
(630 Hgmm ... 800 Hgmm)

Klimatische Bedingungen (Einsatz)

- Umgebungstemperatur: + 10 bis + 35 Grad C
- zuverlässiger
Temperaturgradient: 5 Grad C/h
- relative Luftfeuchte: 20 ... 80 %

Einsatzbedingungen

- Vibration im Frequenzbereich von 5 bis 25 Hz und eine Amplitude von 0,1 mm beeinflussen die Funktionstüchtigkeit nicht
- Störsicherheit bzw. Funkstörpegel entsprechend dem Standard RS 1352-78 und GOST 23511/79 Kurve 1
- Schutzgüte entsprechend STP Z 50.094.100
- Schutzklasse entsprechend TGL 21 366
- Schutzgrad IP 20 entsprechend TGL RGW 778
- Schutzleiteranschluß entsprechend TGL RGW 1110

Geräuschemission (Schalldruckpegel)

- Arbeitsgeräusch
ohne Tastaturbetätigung: 60 dB (AI)
- Arbeitsgeräusch
bei Tastaturbetätigung: 75 dB (AI)
- Leerlaufgeräusch: 50 dB (AI)

Elektrische Bedingungen

- Anschlußart: Netzspannung 220 V + 10/- 15 %
Frequenz 50 Hz + 1/- 2 %
- Leistungsaufnahme: 120 VA

1.7. Erweiterungen der Hardwarekonfiguration

Der PC 1715 kann als universeller Computer über die vorhandenen Interfaceschnittstellen auch mit weiterer Peripherie ausgestattet werden. Diese Kopplungen werden jedoch nicht vom Herstellerwerk vorgenommen. Einfache Kopplungsmöglichkeiten für den Anwender bestehen für solche Geräte, die die für den PC 1715 definierten Schnittstellenausstattungen und Übertragungsverfahren realisieren, wobei durch das Installationsprogramm INSTSCP Modifikationen ermöglicht und die entsprechenden Kanäle definiert werden.

So können z. B. neben den im Abschnitt 1.5. vorgestellten Seriendruckern auch Geräte der älteren Generation als Systemausgabedruker (Voraussetzung ist die Ausstattung des Druckers mit V.24-Schnittstelle) eingesetzt werden. Dies trifft zu auf robotron 1152/Typ 251 oder Typ 252 sowie robotron 1157/Typ 264 oder Typ 267 oder Typ 269. Als Peripheriedruker können sie ebenso wie robotron 1156 über die Interfacereweiterungssteckeinheit entweder mit V.24- oder mit IFSS-Schnittstelle eingesetzt werden.

Darüber hinaus lassen sich moderne elektronische Typenrad Schreibmaschinen in Ausstattungsvarianten mit V.24-Schnittstelle bedingt (eingeschränkter Auslastungsgrad, niedrigere Druckleistung) als Systemausgabedruker oder Peripheriedruker anschließen, z. B. die Schreibmaschinen S 6120, S 6130, S 6005.10 (S 6006), S 6055.

Für den Aufbau von CAD/CAM-Arbeitsplätzen besteht die Möglichkeit, Digitalisiergeräte wie HDG K 6401 oder Plotter (z. B. K 6418) an den PC 1715 anzuschließen.

Auch die Kopplung mit Großrechnern oder anderen leistungsfähigen Rechnern ist möglich. Dies wird gegenwärtig in der UdSSR (Kraftwerksanlagen) und in der VR Bulgarien (landwirtschaftliche Betriebe) realisiert. Hierbei leistet der PC 1715 Erfassungsaufgaben, nutzt das Leistungsvermögen des Großrechners und übernimmt anschließend die Resultate.

Um den Einsatz des PC 1715 für Text- und Datenkommunikation zu ermöglichen, wird für 1987 der Einsatz des Gerätes im lokalen Rechnernetz ROLANET 1 vorbereitet. Dieses Netz gestattet eine maximale Übertragungsrate von 0,5 MBit/s. Es können bis zu 100 Personalcomputer angeschlossen werden.

Im Rahmen einer Weiterentwicklung des PC 1715 wird

- die interne Speicherkapazität auf maximal 256 KByte erhöht (wovon 58 KByte für den Anwender als nutzbarer Speicherbereich zur Verfügung stehen),
- ein größerer Speicherbereich als RAM-Floppy verwendet,
- ein ladbarer Zeichengenerator für semigrafische Zeichen realisiert.

2. Die Betriebssysteme des PC 1715

Mit dem breiten Einsatz des PC 1715 entsteht für den Nutzer die Frage: Wie sieht das Softwareangebot für diesen Rechner aus und wie portabel ist diese Software? Im Mittelpunkt steht dabei die Frage nach verfügbaren Betriebssystemen, die als Mittler zwischen Anwendersoftware und der rechner-spezifischen Hardware sowie als Schnittstelle zwischen Mensch und Maschine eine maßgebliche Rolle spielen.

In diesem Kapitel werden die drei Betriebssysteme des PC 1715 vorgestellt und bewertet (zur Kompatibilität dieser Betriebssysteme vgl. die Übersicht in Tabelle 2.1). Schwergewicht wird auf das Betriebssystem SCP gelegt, das Hauptbetriebssystem für den Einsatz des PC 1715 in der DDR.

2.1. Das Betriebssystem SCP für den PC 1715

Für die Entwicklung eigener Anwendersoftware existieren in SCP eine Reihe Compiler und Interpreter, die auf dem jeweiligen Sprachniveau auch die Portabilität von SCP-Anwendersoftware ermöglichen. Für die Systemprogramm-entwicklung bzw. Entwicklung spezieller Anwenderprogramme bietet das System einen U-880-Assembler, -Binder, -Bibliotheksverwalter und ein komfortables Testhilfsprogramm. Ein Satz leistungsfähiger Systemdienstprogramme gestattet die Diskettenformatierung, einfache Systeminstallation, Kopieren von Dateien und Diskettenkonvertierung. Eine Übersicht über die Diskettenformate, auch im Vergleich mit den anderen Betriebssystemen des PC 1715, gibt die Tabelle 2.2.

Der Systemkern von SCP besteht aus den drei Teilen

- CCP – Console Command Processor,
- BDOS – Basic Disk Operating System,
- BIOS – Basic Input/Output System.

Der nicht vom Systemkern belegte Speicherbereich gliedert sich in den Ver-

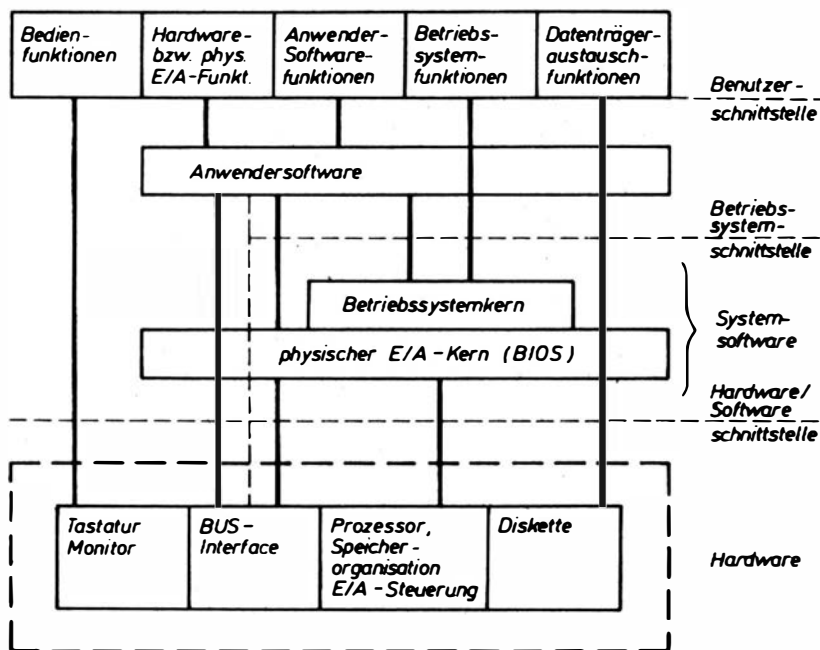


Abbildung 2.1
Schematische Darstellung zu Softwareschnittstellen

Tabelle 2.1
Übersicht zur Kompatibilität der Betriebssysteme des PC 1715

Computer	BS	PC 1715 mit BS	voll software- kompatibel	teilweise software- kompatibel	datenträger- kompatibel
A 5110	BROS	BROS	x ¹	x	x
	SCP5110	SCP1715	x ¹	x	x
A 5120/	SCP1520	SCP1715	x ²	x	x
A 5130	SIOS	BROS		x	x ³
NEWA 501	JAMB	JAMB	x	x	x
ISKRA 555	JAMB	JAMB		x	x

1 für äquivalente Konfigurationen

2 bei Einhaltung von BIOS- bzw. BDOS-Schnittstellen in der Anwendersoftware

3 auf physischer Ebene; mit Einschränkungen auf Dateiebene aufwärtskompatibel

Adresse	SCP	BROS	JAMB
0000	Verständigungsbereich	Betriebssystemkern	Betriebssystem "NEWA 501"
0100			
2800 4C00	TPA	Anwenderspeicher	Adaption NEWA-BS --> E/A-Kern
6000			E/A-Kern
8000			Systemzellen
8640			Betriebssystem - erweiterung
9640			Anwenderspeicher
BC00			
D000	Bildwiederholer		
D800	BIOS	Betriebssystemkern	Bildwiederholer
F800	Bildwiederholer		
FFFF			

Abbildung 2.2
Speicheraufteilung der Betriebssysteme des PC 1715

ständigungsbereich und den TPA (Transient Program Area), den Nutzerbereich. In der Abbildung 2.2 ist die Speicheraufteilung des Betriebssystems SCP (auch im Vergleich mit den anderen Betriebssystemen des PC 1715) dargestellt.

CCP ist ein Kommandointerpreter, der die Kommunikation zwischen Betriebssystem und Nutzer realisiert. Es werden residente und transiente Kommandos unterschieden. Während die Routinen der residenten Kommandos im Systemkern selbst enthalten sind, werden die Routinen der transienten Kommandos von Disketten in den

Tabelle 2.2
Diskettenformate der Betriebssysteme des PC 1715

Diskettenformat	Spuren	Sektoren je Spur	Byte je Sektor	KByte für Nutzer	Betriebssystem			
					SCP*	BROS	JAMB	
5,25"	FM (K 5600.10)	40	9	256	92	-	-	X
5,25"	SS (48 tpi)	38	26	128	126	-	X	-
5,25"	MF (K 5600.10)	40	16	256	148	X (3)	-	-
5,25"	SS (48 tpi)	80	16	256	300	X (3)	-	-
5,25"	MF (K 5600.20)	80	16	256	624	X (2)	-	-
5,25"	SS (MFS 1.6)	77	26	128	256	X (3)	X	X
8"	DS (96 tpi)	77	4	1024	296	X (3)	-	-
8"	FM (MF 3200)	77	8	1024	600	X (2)	-	-
8"	SS (MF 6400)	77	8	1024	600	X (2)	-	-

* Anzahl Systems Spuren

Speicher (TPA) geladen und gestartet. Nach Abarbeitung der Kommandos wird die Steuerung an das CCP-Programm zurückgegeben.

- BDOS** führt alle logischen Operationen zur Behandlung von Diskettendateien und mit den logischen Ein-/Ausgabegeräten durch. Zur Nutzung dieser Routinen, auch BDOS-Funktionen genannt, gibt es ein standardisiertes Verfahren zu deren Aufruf. BDOS bedient sich der physischen BIOS-Routinen.
- BIOS** realisiert die physische Schnittstelle zu den einzelnen Geräten, also das Senden und Empfangen von Daten und Steuerinformationen. Dieser Betriebssystembereich enthält die hardwarespezifischen Programmteile. Am Beginn steht eine Sprungtabelle, die zu den einzelnen Routinen zur Bedienung der im System eingebundenen physischen E/A-Geräte führt. Der Aufruf von BIOS-Funktionen ist ebenfalls vereinbart.

In den transienten Speicherbereich (TPA) werden alle Anwenderprogramme geladen und dort ausgeführt.

2.1.1. Start des Betriebssystems

Der PC 1715 verfügt über einen residenten Urlader (auf PROM). Beim Einschalten des Gerätes (bzw. beim Betätigen der RESET-Taste) wird der *Kaltstart* ausgelöst, indem die Systemsteuerung an das Urladerprogramm übergeben wird.

Es erfolgt eine zyklische Abfrage aller angeschlossenen Diskettenlaufwerke. In eines der Laufwerke ist ein Datenträger einzulegen. Vom Systembereich dieser Diskette wird das Betriebssystem (CCP, BDOS und BIOS) an die entsprechenden Stellen im Speicher geladen, der Verständigungsbereich initialisiert und die Steuerung an das Betriebssystem übergeben. Das gesamte Betriebssystem bleibt bis zum nächsten Kaltstart (Betätigung der RESET-Taste) bzw. bis zum Überschreiben durch Nutzermanipulationen im Speicher resident. Nach erfolgtem Kaltstart gibt das System eine Kaltstartmeldung, und die Steuerung wird an das CCP übergeben. Auf der Konsole erscheint die Meldung

A)

Damit wird der Bediener aufgefordert, durch Eingabe eines CCP-Kommandos mit dem Betriebssystem in Dialog zu treten.

Speziell kann das Betriebssystem auch so installiert werden, daß nach Ablauf des Kaltstarts schon ein CCP-Kommando automatisch aufgerufen und ausgeführt wird, bevor der CCP-Dialog beginnt.

2.1.2. Aufbau des Datenträgers und Diskettenformate

Disketten sind mit einer Magnetschicht versehene Folienscheiben. Sie können ein- oder zweiseitig (SS/DS) mit Daten beschrieben werden. Auf jeder Seite sind um den Scheibenmittelpunkt konzentrisch von außen nach innen Datenspuren in gleicher Anzahl angeordnet. Auf jeder Spur befindet sich eine konstante Anzahl von Sektoren. Diese Sektoren wiederum enthalten eine konstante Anzahl von Datenbyte. Die Anzahl der benutzten Disk-Seiten, der vorhandenen Spuren, der Sektoren in der Spur und die Sektorengröße sind abhängig vom Laufwerks- und Diskettentyp. Eine Übersicht gibt die Zusammenstellung in Tabelle 2.3. Auf der Diskette ergibt sich also ein dreidimensionales physisches Speicherfeld für die Sektoren.

Ausgehend von diesem physischen Aufbau wird die Diskette logisch strukturiert. Dazu wird angenommen, daß die Diskette logisch nur eine Seite hat. Das hat zur Konsequenz, daß sich bei DS-Disketten entweder die Spuranzahl oder antivalent die Spurkapazität verdoppelt. In diesem nunmehr zweidimensionalen Feld gibt es zwei Bereiche. Die ersten Spuren (Anzahl ist installierbar) sind als Systembereich reservierbar. In diesem Bereich können die Daten abgelegt werden, die vom Urlader beim Kaltstart in den Speicher geladen werden. Der zweite Bereich auf der Diskette ist der Datenbereich, auf den durch Lese-/Schreiboperationen über BDOS und BIOS Daten zwischen Diskette und Speicher ausgetauscht werden.

Die elementare Datenmenge, die über diesen Weg ausgetauscht wird, ist der Satz (Record) mit einer Datenmenge von 128 Byte.

Auf jeder Spur befindet sich dann eine konstante Anzahl von Sätzen, die sequentiell auf der Spur angeordnet sind. Die Zuordnung zu den Sektoren erfolgt über Zuordnungstabellen.

Mit den BIOS-Zugriffen wird ein zweidimensionales Feld (Spurnummer/Satznummer in der Spur) von Sätzen adressiert.

Abkürzungen zu Tabelle 2.3:

MFM	– modify frequency modulation	(doppelte Dichte)
FM	– frequency modulation	(einfache Dichte)
DD	– double density	(doppelte Dichte)
SD	– single density	(einfache Dichte)
DS	– double side	(zweiseitig)
SS	– single side	(einsseitig)

Tabelle 2.3
Zusammenstellung der Diskettenformate im Betriebssystem SCP

Format (Zoll)	8	8	8	5,25	5,25	5,25	5,25
Seiten	1	1	1	2	1	1	1
Spuren/Seite	77	77	77	80	80	40	40
Sektoren/Spur	8	4	26	16	16	16	26
Byte/Sektor	1024	1024	128	256	256	256	128
Aufzeichnungsverfahren	MFM	FM	FM	MFM	MFM	MFM	MFM
Laufwerke	MF 6400	MF 6400	MF 6400	MFS 1.6	MFS 1.6	MFS 1.2	MFS 1.2
				MFS 1.4	MFS 1.4	MFS 1.4	MFS 1.4
				MFS 1.6	MFS 1.6	MFS 1.6	MFS 1.6
INIT-Mnemonic	DD-SS	SD-SS	SD-SS	DD-DS	DD-SS	DD-SS	DD-SS
	8×1024	4×1024	26×128	16×256×80	16×256×80	16×256×40	26×128×40
INSTSCP-Mnemonic	1024×8×77	1024×4×77	128×26×77	256×16×80	256×16×80	256×16×40	128×26×40
	ROB 8"	ROB 8"	CPM 8"	S2 5"	ROB 5"	ROB 5"	CPM 5"
	DD-SS	SD-SS	SD-SS	DD-DS	DD-SS	DD-SS	DD-SS
Datenkapazität ¹ (KByte)	600 (596)	296 (294)	243 (241)	624 (620)	308 (306)	148 (146)	123 (121)
max. Verzeichniseintragen	128	64	64	128	64	64	64
Blockgröße (KByte)	2	2	1	2	2	2	1
Skew-Faktor	-	-	6	-	-	-	6
Systemspuren	2	3	2 ²	2	3	3	2 ²

¹ Datenkapazität gibt die Speicherkapazität der Diskette ohne Systemspuren an; der Wert in Klammern ist die Speicherkapazität abzüglich der Kapazität des Verzeichnisses.

² Dieses Diskettenformat ist nicht zum Umladen geeignet.

Abkürzungen siehe Seite 74

Für die BDOS-Zugriffe wird die Blockstruktur eingeführt. Dazu wird der Datenbereich in eine Sequenz von Sätzen so aufgeteilt, daß nach den Sätzen einer Spur die der nächsten Spur folgen. Eine festgelegte Anzahl von aufeinanderfolgenden Sätzen bildet einen SCP-Block. Damit ist der gesamte Datenbereich der Diskette in ein eindimensionales Feld von Blöcken aufgeteilt. Die Anzahl der Blöcke ergibt die Gesamtkapazität der SCP-Diskette.

Der Datenaustausch über BDOS als Nutzerschnittstelle erfolgt dateiunterstützt. Der Datenbereich wird von BDOS blockweise dynamisch verwaltet. Der Zugriff auf die Dateien erfolgt über einen Katalog, Directory, der die ersten Blöcke (Anzahl installierbar) des Datenbereiches okkupiert.

Jedes logische SCP-Laufwerk wird durch eine Tabelle beschrieben, die die physischen, die BIOS- und die BDOS-Parameter enthält. Die Disketten müssen parameterkompatibel sein.

Programme und Daten sind auf der Diskette in Dateien abgelegt. Der Zugriff in SCP wird prinzipiell über diese Datenströme organisiert, d. h. eine Datenmenge wird unter einem Namen (Dateispezifikation) auf der Diskette abgelegt und kann auch nur über diesen Namen vom Betriebssystem abgerufen werden.

Mit der SCPX-Dateispezifikation kann eine spezielle Datei (*ufn*) oder eine Gruppe von Dateien (*afn*) gekennzeichnet werden.

Eine Dateispezifikation setzt sich aus

- der Laufwerkspezifikation,
 - dem Dateinamen,
 - der Dateitypbenennung
- zusammen.

Die *Laufwerkspezifikation* wird aktualisiert durch "a:", "b:" usw. Es können nur die im Betriebssystem installierten Laufwerke selektiert werden. Wenn diese Angabe weggelassen wird, dann wird auf das aktuelle Laufwerk Bezug genommen.

Ein Dateiname kann aus 1 bis 8 Zeichen bestehen. (Intern wird der Begriff mit Leerzeichen aufgefüllt.) Als Zeichen sind zugelassen:

A B ... Z

a b ... z

0 1 ... 9

! # \$ % & ' () + - / \ ^ _ @ { } | ~

Die Dateitypbenennung besteht aus einem Punkt und 1 bis 3 weiteren Zeichen. Als Zeichen sind die gleichen wie für den Dateinamen zugelassen. Wird keine Typbenennung eingegeben, so werden 3 Leerzeichen angenommen.

Neben den eindeutigen Dateispezifikationen existieren die mehrdeutigen Dateispezifikationen. Diese enthalten die Gruppenkennzeichen "?" und "*". Das "?" steht für ein beliebiges Zeichen in genau dieser einen Position. Der "*" steht für eine beliebige Anzahl von Zeichen ab genau dieser Position bis zum Ende des Dateinamens oder der Typbenennung.

Außer Dateien werden von einigen System- und Anwenderprogrammen auch Datenströme über die logischen E/A-Geräte akzeptiert. Angesprochen werden die Datenströme mit den Schlüsselwörtern

con: Eingabe oder Ausgabe über Konsole,
 rdr: Eingabe über den sequentiellen Kanal,
 pun: Ausgabe über den sequentiellen Kanal,
 list: Ausgabe über das List-Gerät.

Es werden damit bytewise Datenströme transportiert, deren Ende durch das EOF-Kennzeichen "^Z" (1AH) markiert ist.

2.1.3. Die CCP-Kommandos

Nach dem erfolgten Start des Betriebssystems SCP wird der CCP-Status eingestellt. Auf dieser Kommandoebene kann

- ein Warmstart ausgelöst,
- ein anderes aktuelles Laufwerk selektiert,
- ein residentes CCP-Kommando aufgerufen und ausgeführt,
- ein transientes Dienstprogramm in den TPA geladen und aktiviert,
- ein transientes Nutzerprogramm in den TPA geladen und aktiviert werden.

Nach Ausführung des Kommandos wird die Steuerung wieder an den CCP-Kommandointerpreter übergeben, der sich mit

d)

meldet.

Die CCP-Kommandos werden als Kommandozeile eingegeben. Neben den abbildbaren Zeichen, die als Echo auf der Konsole erscheinen, sind zur Korrektur und Steuerung der Eingabe die in Tabelle 2.4 erklärten Steuerzeichen wirksam.

Vor der Interpretation durch CCP werden alle Kleinbuchstaben der Kommandozeile in Großbuchstaben umgewandelt.

Beginnt die Kommandozeile mit ";", so handelt es sich um eine Kommentarzeile, die keine Aktionen auslöst.

Tabelle 2.4
Steuerzeichen im CCP-Status

Zeichen	Wirkung
(DEL)	löscht das zuletzt eingegebene Zeichen und zeigt das Echo des gelöschten Zeichens auf der Konsole an
^R	beendet die Anzeige der gerade editierten Zeile mit "*" und protokolliert den tatsächlichen Pufferinhalt auf der nächsten Konsolenzeile
(BS) (auch ^H)	löscht das zuletzt eingegebene Zeichen sowohl auf der Konsole als auch im Puffer
^U	löscht die gesamte Zeile im Puffer und kennzeichnet das durch Ausgabe von "*" auf der Konsole; die Edition kann in der folgenden Konsolenzeile neu begonnen werden
^X	löscht die eingegebene Zeile sowohl im Puffer als auch auf der Konsole; die Edition kann neu begonnen werden
^E	beginnt eine neue Zeile auf der Konsole; das Zeichen wird nicht in den Puffer übernommen, sondern steuert nur das Konsolenbild
(CR) (auch ^M)	schließt die Puffereingabe ab
(LF) (auch ^J)	beendet den BDOS-Ruf
(HT) (auch ^I)	gibt die Zahl von Leerschritten auf der Konsole aus, die zum Erreichen der nächsten Spalte im 8er-Raster (1, 9, 17, 25, ...) notwendig ist
^S	laufende Ausgabe unterbrechen bis zur nächsten Konsoleneingabe
^P	schaltet den LIST-Kanal parallel zum CON-Kanal; Bedingung bleibt gültig bis zum nächsten ^P
^C	bricht das gesamte Programm ab und verzweigt zur Reinitialisierung des Betriebssystems durch Ansprung des Warmstartprogrammes (WBOOT); diese Funktion wird nur aktiv, wenn dieses Zeichen als 1. Zeichen in den Puffer eingegeben wird

Mit dem Kommando

d:

wird das *aktuelle Laufwerk* ausgewählt.

Nur im Betriebssystem installierte Laufwerke sind selektierbar.

Residente Kommandos sind Bestandteil des speicherresidenten CCP-Moduls. Sie sind im CCP-Status abrufbar und beeinträchtigen den Zustand des TPA nicht.

Es sind im einzelnen die Kommandos DIRectory, ERase, REName, SAVE, TYPE und USER, die nachfolgend erklärt werden.

DIR Disketten-Verzeichnis-Anzeigen

DIR gibt die Namen und Typen aller angesprochenen Dateien aus. Dateien mit dem Attribut „SYS“ werden nicht angezeigt. Als Dateispezifikation ist auch ein Gruppenmerkmal zulässig; fehlende Dateispezifikation meint alle Dateien (d. h. *.*). Existiert keine Datei aus der ausgewählten Gruppe, erscheint die Konsolenmeldung „NO FILE“.

Syntax: DIR d:dateiname.typ

ERA Datei löschen

ERA löscht alle Dateien der Gruppe, auf die sich die vorgegebene Dateispezifikation bezieht.

Dateien mit dem R/O-Attribut können nicht gelöscht werden. Für *.* muß zur Sicherheit vor Bedienfehlern noch die Konsolenanfrage ALL FILES (Y/N) vom Bediener beantwortet werden.

Syntax: ERA d:dateiname.typ

REN Datei umbenennen

REN gibt einer existierenden Datei einen neuen Namen. Unterschiedliche Laufwerkspezifikationen und Gruppenmerkmale sind nicht zugelassen. Wenn die alte Datei nicht existiert, erfolgt die Konsolenmeldung NO FILE. Wenn die neue Datei schon existiert, erfolgt Meldung FILE EXIST, und die Umbenennung findet nicht statt. Dateien mit dem Attribut R/O können nicht umbenannt werden. Enthält die alte Datei eine Laufwerkspezifikation, muß sie mit der der neuen Datei identisch sein.

Syntax: REN d:dateiname.typ = dateiname.typ
 neuer Name = alter Name

- TYPE** Text-Datei auf Konsole anzeigen
TYPE gibt den Inhalt einer Textdatei (ASCII-Zeichen) zeichenweise auf die Konsole aus.
Es sind keine Gruppenmerkmale zugelassen. Die Datei sollte nur Zeichen einer Textdatei enthalten. Die Funktion wird beim ersten auftretenden EOF-Kennzeichen (^Z) beendet.
Syntax: TYPE d:dateiname.typ
- SAVE** Arbeitsspeicherabzug in Datei
SAVE legt eine Datei an und legt einen Abzug des TPA-Inhaltes in dieser Datei ab. Ab TPA-Anfang (Speicheradresse 0100H) werden die dort befindlichen Byte nacheinander in die Datei eingeschrieben.
Die Menge ist in der Anzahl *n* von Speicherseiten (jeweils 256 Byte) angebar.
Syntax: SAVE n d:dateiname.typ
"n" wird dezimal angegeben. Es sind keine Gruppenmerkmale zugelassen. Existiert bereits eine Datei mit dem gleichen Namen, dann wird diese zuerst gelöscht und danach eine neue Datei angelegt.
- USER** Wahl des aktuellen Nutzerbereiches
USER erlaubt verschiedenen Nutzern einer Diskette, ihren eigenen Katalog und damit ihre eigene Nutzeroberfläche auf dieser Diskette anzulegen. Das Kommando stellt den aktuellen Nutzerbereich ein. Für Nutzerbereich *n* sind dezimale Werte von 0...15 zugelassen. Vom SCP wird standardmäßig beim Kaltstart 0 als aktueller Nutzerbereich eingestellt. Mit STAT USR kann die Nummer des aktuellen Nutzerbereiches abgefragt werden.
Syntax: USER n

Außer den in CCP residenten Systemdiensten stehen dem Nutzer im Betriebssystem SCP eine Reihe von *transienten Systemprogrammen* zur Verfügung. Diese Programme werden genau wie die Anwenderprogramme durch CCP-Kommando in den TPA geladen und ausgeführt. Nach Abarbeitung wird die Steuerung wieder an CCP übergeben. Wird als erstes Kommando der Zeile ein Begriff eingegeben, der kein residentes CCP-Kommando darstellt, dann wird dieser als Dateiname einer Dateispezifikation vom Typ .COM interpretiert. Diese Datei wird von der Diskette des aktuellen Laufwerkes (oder bei zusätzlicher Laufwerk-Spezifikation vom spezifizierten LW) ab der Adresse 0100H in den TPA geladen. Anschließend wird die Steuerung an dieses Programm übergeben und zur Ausführung gebracht.

Zu den transienten Systemprogrammen gehören die Dateien DUMP.COM, PIP.COM, STAT.COM, SUBM.COM, XSUB.COM, INIT.COM und SGEN.COM, die nachfolgend erklärt werden.

DUMP – beliebige Datei anzeigen

DUMP gibt den Inhalt einer beliebigen Datei sequentiell als Hex-Zeichen auf dem Bildschirm aus. Es werden vom Anfang der Datei jeweils 16-Byte-Blöcke in einer Zeile ausgegeben. Davor steht die Speicheradresse des Blockanfanges.

Erlaubt ist nur eine eindeutige Dateispezifikation.

Syntax: DUMP d:dateiname.typ

PIP – Datenaustausch

Das Dienstprogramm PIP kann Quelldatenströme verketteten und zu einem Zieldatenstrom zusammenfassen. Als Quelle können Dateien oder logische Eingabegeräte, als Ziel eine Datei oder ein logisches Ausgabegerät angegeben werden. Für die Anfügung jeder Quelldatendatei können optionale Bedingungen angegeben werden.

Syntax: PIP pipkommando

oder

PIP

Wird das *pipkommando* im CCP-Kommando mit angegeben, erfolgt nach der Ausführung ein Übergang in den CCP-Modul. Anderenfalls wird nach Laden des Programms mit dem Promptzeichen "*" die Zeilen-Eingabe des PIP-Kommandos vom Programm angefordert. Nach Ausführung des Kommandos werden im Programm wieder mit "*" weitere PIP-Kommandos abgefordert. Erst die Eingabe einer Leerzeile (nur RETURN) oder Ausführung des Warmstartes mit "^C"-Taste führt in den CCP-Modul zurück.

Disketten, auf denen nur Quelldateien entnommen werden, können auch ohne Warmstart gewechselt werden.

Das allgemeine PIP-Kommando lautet

ziel=quelle[option _]quelle[option],...

Links vom Gleichheitszeichen steht das Ziel. Rechts können eine oder mehrere Quellen angegeben werden. Zu jeder Quelle können in eckigen Klammern beliebig viele Options angegeben werden. Die Reihenfolge der Options ist beliebig. Zwischen mehreren Options können Leerzeichen stehen. Für Ziel und Quelle kommen Dateinamen oder Gerätenamen in Frage.

STAT – Systeminformationen

Das Kommando liefert allgemeine statistische Informationen über Dateien und logische Geräte. Außerdem lassen sich den Dateien Attribute und den logischen Geräten jeweils ein Subkanal zuordnen.

Syntax: STAT

STAT argumente

Benötigte *argumente* müssen in der CCP-Kommandozeile eingegeben werden. Zugelassen sind auch Gruppenmerkmale.

Wenn STAT aufgerufen wird, sollten nach dem letzten vorangegangenen Warmstart keine Disketten mehr gewechselt werden, da sich die Auswertung in einigen Fällen sonst auf die Vorgängerdiskette bezieht. Hinsichtlich der einzelnen Argumente sei auf die Systemdokumentation verwiesen.

SUBM – Ausführen einer Befehlsfolge

Das Kommando erlaubt, verschiedene CCP-Kommandos und transiente Programme aus einer Kommando-Datei in der dort angegebenen Reihenfolge abzuarbeiten (Stapelbetrieb).

Die Kommando-Zeilen können formale Parameter (\$1...\$9) enthalten. Diese formalen Parameter werden mit einer einfachen Zeichenketten-Substitution durch die aktuellen Parameter par1...par9 beim Aufruf des Kommandos ersetzt. Beim Abarbeiten der Kommando-Datei wird jede Kommando-Zeile mit den aktuellen Parametern auf der Konsole angezeigt.

Syntax: SUBM d:dateiname par1 par2...

Die in der Kommando-Datei *dateiname.SUB* stehenden CCP-Kommando-Zeilen werden nacheinander aufgerufen.

Beim Aufruf von SUBM muß A das aktuelle Laufwerk sein. Auf diesem Laufwerk wird die Zwischen-Datei "\$\$\$.SUB" mit der aktuellen Befehlsfolge angelegt, abgearbeitet und danach wieder gelöscht.

Jede beliebige Tastenbetätigung bricht die Ausführung der Kommandofolge beim Aufruf des nächsten Kommandos ab und löscht die Zwischendatei wieder.

Kommando-Dateien dürfen nur aus Textzeichen und Zeilenende bestehen. Leerzeilen in der Kommandodatei sind nicht zulässig.

XSUB – Erweiterung der SUBM-Funktion

Mit SUBM können nur CCP-Kommandos aus der Kommando-Datei entnommen und abgearbeitet werden. Wenn XSUB in einer Kommando-Datei aufgerufen wird, werden alle zeilenweisen Eingaben in den nachfolgenden Stapel-Programmen statt von der Tastatur von der Kommandodatei abgefordert; der Eingabedatenstrom ist hiermit vollständig von der Tastatur auf die

Kommando-Datei umgelenkt. Werden nach Verarbeitung des Dateieingabestromes noch weitere Eingaben von der Konsole abgefordert, wird die Eingabe automatisch wieder auf Konsole umgeschaltet. Nach Abarbeitung der Kommandodatei ist XSUB nicht mehr aktiv.

Syntax: XSUB

Es existieren jedoch auch Programme, die unter XSUB nicht arbeiten. Wird XSUB benötigt, so muß es das erste Kommando in einer Kommando-Datei sein. Um nach Abarbeiten der Kommando-Datei sicher den ursprünglichen Systemzustand wieder herzustellen, sollte ein Warmstart ausgelöst werden.

INIT – Diskettenformatieren

Das Kommando INIT dient zum Formatieren von Disketten. Dies ist ein Verfahren, bei dem die magnetisierbaren Oberflächen der Diskette in Bereiche eingeteilt werden, in denen die Informationen gespeichert werden.

Außer INIT können alle anderen Programme nur mit formatierten Disketten arbeiten. Disketten müssen vor dem Gebrauch formatiert sein. Wird eine bereits beschriebene Diskette wieder formatiert, gehen alle gespeicherten Informationen verloren!

Die Aufteilung der Diskette in Bereiche geschieht nach folgendem Prinzip: Eine Diskette kann ein- oder zweiseitig beschrieben werden. Die Diskettenseiten werden in konzentrische Spuren eingeteilt. Innerhalb der Spuren sind die Datenbereiche (Sektoren) angeordnet.

Jedem Datenfeld eines Sektors ist ein Adreßfeld zugeordnet, welches folgende Angaben enthält: Diskettenseite, Spurnummer, Sektornummer, Sektorenlänge. Bei Vereinbarung dieser Angaben kann das Betriebssystem eindeutig auf die Datenfelder der Diskette zugreifen.

Die möglichen Diskettenformate in Abhängigkeit von Diskettengröße und Laufwerkstyp sind in Tabelle 2.3 beschrieben.

INIT arbeitet im Dialog mit dem Bediener. Die Arbeitsweise wird in Abbildung 2.4 am Beispiel des Formatierens einer einseitigen 8-Zoll-Diskette im Format mit 4 Sektoren zu 1024 Byte, einfache Aufzeichnungsdichte, gezeigt.

SGEN – Systemgenerierung

Das Kommando dient zur Generierung des Betriebssystems SCPX. Das SCPX kann in drei verschiedenen Formen vorliegen:

- auf den Systemspuren einer Diskette (zum Umladen),
- als transientes Kommando (COM-File),
- als residentes Betriebssystem im Speicher des Computers.

SGEN kann aus jeder dieser Formen (als Quelle) Betriebssysteme in jede

dieser Formen (als Ziel) kopieren. In den meisten Fällen wird eine neue Systemdiskette generiert, indem die Systemspuren einer bereits vorliegenden Systemdiskette kopiert werden.

Zuerst wird das Betriebssystem aus einer der 3 wählbaren Quellen in den Speicher eingelesen (LOAD). Diese Daten können dann beliebig oft zu einem der 3 Ziele transferiert werden (SAVE).

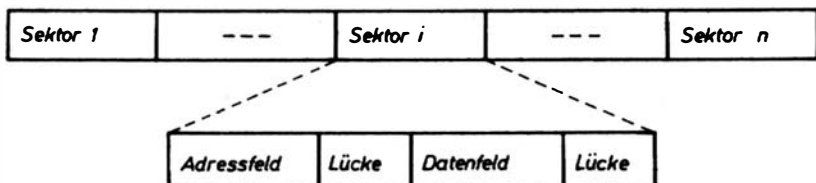


Abbildung 2.3
Prinzipieller Spuraufbau einer Diskette

A) INIT

```

INIT - SCP      V 0.4      DEVICE: ROBOTRON 1715
=====
PLEASE ENTER DRIVE:  C (<----- Auswahl des Laufwerks)
DISK FORMAT CAN BE:  0 = DD - SS   8*1024
                    1 = SD - SS   4*1024
                    2 = SD - SS  26* 128
PLEASE SELECT FORMAT: 1 (<----- Auswahl des Formats)
WARNING    ==>  ALL FILES WILL BE SCRATCHED (Y/N): Y
=====
WAIT   FORMATTING TRACK ( nn ) (<----- Protokoll des Formatierens)
==>  FORMATTING COMPLETE

```

Abbildung 2.4
Menübild des Systemprogramms INIT

2.1.4. Beschreibung der BDOS-Funktionen

Das Diskettenbasisbetriebssystem (BDOS) stellt den Kern des Betriebssystems SCP dar. Es unterstützt die Ein- und Ausgabe auf den logischen Geräten (Konsole, Listgerät, sequentielle Datenkanäle) sowie die Dateiarbeit auf den Disketten.

Über den BDOS-Ruf können folgende Aufträge von BDOS angefordert werden:

- Zeichenorientierte Ein- und Ausgabe
 - Zuordnung der logischen zu den physischen Ein- und Ausgabekanälen
 - zeichenweise Ein- und Ausgabe über die Konsole
 - zeichenweise Ein- und Ausgabe über die sequentiellen Datenkanäle
 - zeichenweise Ausgabe über den Listkanal
- Arbeit mit Diskettendateien
 - allgemeine Dateimanipulationen und -verwaltung
 - sequentieller Dateizugriff
 - direkter Dateizugriff
- Systemverwaltung
 - Initialisierung
 - Ermittlung des Systemzustandes
 - weitere Hilfsfunktionen.

Der BDOS-Ruf erfolgt über einen UP-Aufruf der absoluten Speicher-Adresse 5 im Verständigungsbereich. An dieser Stelle befindet sich ein Sprungbefehl, der bei der Systeminitialisierung eingetragen wird. Dieser Sprungbefehl setzt im BDOS-Körper auf. Die Rückkehr aus dem BDOS wird dort selbst durch einen RET-Befehl realisiert.

Der BDOS-Ruf muß durch Belegung von Registern spezifiziert werden. Die BDOS-Funktionen sind mit laufenden Nummern versehen (vgl. die Zusammenstellung der BDOS-Funktionen in Tabelle 2.5). Im CPU-Register C muß vor Ausführung des Rufes diese Code-Nummer eingestellt werden, die der gewünschten BDOS-Funktion zugeordnet ist.

Die zusätzlichen Eingabeparameter werden

- bei 8-Bit-Werten in das CPU-Register E,
 - bei 16-Bit-Werten in das CPU-Doppelregister DE
- eingetragen. Nach Ausführung der BDOS-Funktion wird das rufende Programm hinter der Aufrufstelle fortgesetzt.

Ausgabeparameter nach Abarbeitung der BDOS-Funktion werden als

- 8-Bit-Werte im CPU-Register A,
- 16-Bit-Werte im CPU-Doppelregister HL,

Tabelle 2.5
BDOS – Funktionen

Nr.	Bezeichnung	Eingang	Ausgang
0	Warmstart	---	---
1	Konsoleneingabe mit Konsolenecho	---	A: Zeichen ¹
2	Konsolenausgabe	E: Zeichen (mit ^S, ^P)	---
3	READER – Eingabe	---	A: Zeichen
4	PUNCH - Ausgabe	E: Zeichen	---
5	LIST – Ausgabe	E: Zeichen	---
6	direkte Konsolen E/A	E = FF: Status E = /FF: aus E Zeichenausgabe	A: Konsolen Status ² ---
7	IOBYTE abfragen	---	A: IOBYTE
8	IOBYTE belegen	E: IOBYTE	---
9	Zeichenkette ausgeben auf Konsole	DE: → Kette (EKZ = \$, mit ^S, ^P)	---
10	Eingabe Konsolpuffer	DE: → Puffer ³	Pufferinhalt ⁴
11	Konsolenstatus	---	A: Konsolen-Status ²
12	Version ermitteln	---	HL: Version
13	Diskettensystem zurücksetzen	---	A: Batchmode-Flag ⁹
14	Auswahl Bezugs-LW	E: LW * (0...15) für A...P	---
15	Datei eröffnen	DE: → FCB	A: DC ⁷ , = FF: Datei nicht vorh.
16	Datei schließen	DE: → FCB	A: DC ⁷ , = FF: Datei nicht vorh.
17	erste Eintragung suchen	DE: → FCB	A: DC ⁷ , = FF: Datei nicht vorh.
18	folgende Eintragung suchen	---	A: DC ⁷ , = FF: Datei nicht vorh.
19	Dateien löschen	DE: → FCB	A: DC ⁷ , = FF: Datei nicht vorh.
20	nächsten Satz lesen	DE: → FCB	A: EC ⁵ , = /00: EOF
21	nächsten Satz schreiben	DE: → FCB	A: EC ⁵ , = /00:Disk. voll
22	Datei erzeugen	DE: → FCB	A: DC ⁷ , = FF: Verzeichn. voll
23	Datei umbenennen	DE: → FCB ⁴	A: DC ⁷ , = FF: Datei nicht vorh.
24	Abfrage angeschlossener LW	---	HL: LW-Vektor
25	Abfrage Bezugs-LW	---	A: LW * (0-15)

26	Datenpuffer adressieren	DE: → Puffer	---
27	Adresse von ALLOC (ALV) ermitteln	---	HL: → ALLOC
28	Schutz des Bezugs-LW	---	---
29	Abfrage geschützte LW	---	HL: LW-Vektor
30	Dateimerkmale setzen	DE: → FCB	A: DC', = FF: Datei nicht vorh.
31	Adresse des DPB ermitteln	---	HL: → DPB des selekt. LW
32	Nutzernummer abfragen/setzen	E = FF: Abfrage E = /FF: Setzen mit Wert von E	A: Nutzer* (0-15)
33	direkt adressierten Satz lesen	DE: → FCB	A: RC ⁶
34	direkt adressierten Satz schreiben	DE: → FCB	A: RC ⁶
35	Dateigröße berechnen	DE: → FCB	Eintrag in FCB
36	Berechnung der aktuellen Satzadresse	DE: → FCB	Eintrag in FCB + 33,34,35: $r0-2 = fkt(ex,cr)$
37	ausgewählte LW zurücksetzen	DE: LW-Vektor	A: 00
40	direkt adressierten Satz schreiben und Block initialisieren	DE: → FCB	A: RC ⁶

Erläuterung zu den Parametern:

- Die Zeichen "→" sind zu interpretieren als „Zeiger auf“.

- Die Zeichen "=/" sind zu interpretieren als „ungleich“.

BDOS-Interface:

Eingang: CPU-Register C : BDOS-Funktionsnummer
 CPU-Register DE : Feldadressen, Vektoren
 E : Eingabezeichen
 UP-Ansprung : 0005H
 Ausgang: CPU-Register A : Status, Zeichen
 CPU-Register HL : Vektoren, Adressen

Fußnoten:

1 Sonderzeichen

^H: Rückschritt

(CR): Wagenrücklauf

(LF): Zeilenschaltung

^I: Tabulation auf nächste 8. Stelle

^S: start/stop Konsolenausgabe

^P: List - Echo

2 Status = 00 kein Zeichen im Konsoleneingabepuffer
 = /00 Zeichen im Konsoleneingabepuffer

3 Steuerzeichen

(Systemsteuerzeichen bei Pufferedition)

4 Struktur des Puffers

DE +0 +1 +2 +3 ... +n+1

mx nx Z1 Z2 ... Zn

mx : Kapazität des Puffers (ist voreinzustellen)

nx : Füllstand des Puffers

Z : Zeichen im Puffer

5 Fehler-Code

A = 00 : Operation ok

= /00 : Operation nicht ausführbar

6 Direktzugriffs-Code

A = 01 : Lesen ungeschriebener Daten

= 03 : Extent-Wechsel nicht möglich

= 06 : Zugriff auf EOD-Satz der Datei

7 Verzeichnis-Code

A = FF : Fehler

= 0,1,2,3 : Nummer des Eintrages am aktuellen Verzeichnis-Satz

8 Aufbau des Dateibesreibers für Umbenennung

FCB +0...+15: Datei alt (LW = 0, 1, ... 16)

+16...+31: Datei neu (LW = 0 oder LW-alt)

9 Batchmode

A = 00 : keine Batchmodedatei

A = FF: Batchmodedatei vorhanden

Aufbau der LW-Vektoren

Vektor im CPU-Doppelregister

Bit 0 : LW A

1 : LW B

⋮

15 : LW P

– Feldeintragen in einem Feld, das durch das CPU-Doppelregister DE vor Aufruf adressiert wurde, bereitgestellt.

Das Betriebssystem benutzt einen eigenen Stapel, so daß das aufrufende Programm nur mit einer Stapel-Ebene durch den Aufruf belastet wird. Alle CPU-Register werden durch Bearbeitung einer BDOS-Funktion verändert. Bei Zugriff auf Dateien ist es notwendig, im Arbeitsspeicher einen *Dateisteuerblock* anzulegen. Dieser Steuerblock wird bei BDOS-Rufen vom SCP herangezogen und verwaltet. Der Block besteht aus einem Feld von 33 Byte (bei sequentiellem Zugriff) bzw. von 36 Byte bei wahlfreiem Zugriff. Bei den BDOS-Rufen ist die Basisadresse des Blockes mit zu übergeben. Die Struktur des Dateisteuerblocks ist aus Tabelle 2.6 zu entnehmen.

Tabelle 2.6
Struktur des Dateisteuerblockes (FCB)

Offset	Symbol	Inhalt
+00H	dr	Laufwerk-Codierung dr= 0: Bezugs-LW 1: LW A ⋮ 16: LW P
+01H	n1-n8	Dateiname in ASCII Dateiattribute in den 8. Bits n1' - n8' n1' - n4': für den Nutzer reserviert n5' - n8': nicht genutzt, aber reserviert
+09H	t1-t3	Dateityp in ASCII Dateiattribute in den 8. Bits t1' - t3' t1' = 1/0: read only / read-write t2' = 1/0: SYS-Datei / DIR-Datei t3' = : nicht genutzt, aber reserviert
+0CH	ex	laufende Nr. des Dateiteils (Extent) ex = 00... 1Fh
+0DH	s1-s2	reserviert für internen Systemgebrauch (ggf. S2 = 0000 vor BDOS-fkt. * 15,17,18,22 setzen)
+0FH	rc	Satzzähler im Extent rc := 00... 80h
+10H	d00-d15	Verzeichnis der für den zugehörigen Extent reservierten SCP-Blöcke
+20H	cr	Nummer des Satzes im Extent, der beim nächsten sequentiellen Zugriff im Eingriff ist (wird nur bei wahlfreiem Zugriff benutzt)
+21H	r0-r2	Satznummer bezüglich der gesamten Datei
+24H		ENDE DES BEREICHES

2.1.5. BDOS-Fehlermeldungen

Während der Arbeit mit BDOS können 4 Arten von Fehlern auftauchen. Diese werden dem Bediener durch die Meldung

BDOS ERR ON d: fehlerart

auf der Konsole angezeigt.

Mit d wird das entsprechende SCP-Laufwerk spezifiziert.

Als *fehlerart* kann alternativ

Bad Sector,
Select,
R/O,
File R/O

auftreten. Die Fehlerspezifikationen haben folgende Bedeutung.

Bad Sector

Der BDOS-Zugriff auf einen Datensatz (Lese- oder Schreiboperation) ist nicht möglich, da entweder das Laufwerk oder die Diskette defekt ist oder Laufwerk und Diskettenparameter unterschiedlich sind. Diese Fehlermeldung muß vom Bediener quittiert werden. Geschieht dieses durch Betätigung der Taste RETURN, wird der Fehler ignoriert, allerdings mit der Konsequenz, daß der fehlerhafte Satz ohne weitere Kennzeichnung im Ein- oder Ausgabedatenstrom ausgelassen wird. Erfolgt die Fehlerquittierung durch die "C"-Taste, wird das laufende Programm ohne Aktualisierung der Daten auf dem Datenträger abgebrochen und mit „Warmstart“ in den Dialogverkehr in CCP zurückgekehrt.

Select

Dieser Fehler taucht auf, wenn ein Laufwerk angesprochen wird, das im vorhandenen Betriebssystem nicht installiert ist. Durch Quittierung mit einer beliebigen Taste wird „Warmstart“ ausgeführt und in den CCP-Dialogstatus zurückgekehrt, ohne daß eine Systemverklemmung auftritt.

R/O

Dieser Fehler erscheint, wenn auf eine Diskette geschrieben werden soll, die durch BDOS-Kommando schreibgeschützt wurde, oder wenn die Diskette zuvor gewechselt wurde, ohne daß danach durch entsprechende BDOS-Kommandos bzw. „Warmstart“ das Laufwerk entkoppelt wurde. Mit Betätigung der "C"-Taste wird ein Warmstart ausgelöst und in den CCP-Status zurückgekehrt.

File R/O

Dieser Fehler tritt auf, wenn ein Schreibversuch auf eine schreibgeschützte Datei unternommen wird. Die Quittierung mit beliebiger Taste durch den Bediener führt zum Warmstart.

2.1.6. Beschreibung der BIOS-Funktionen

Das BIOS (Basic Input/Output System) ist der hardwareabhängige Teil des SCP. Es besteht aus Unterprogrammen zur Initialisierung und Steuerung der Ein- und Ausgabekomponenten des Personalcomputers, den sogenannten Drivern. Diese Unterprogramme sind über einen Sprungvektor, der sich am Beginn des BIOS befindet, zu erreichen (vgl. Tabelle 2.7). Ein Sprungbefehl auf der Speicheradresse 0 verzweigt zum Warmstartsprungbefehl des BIOS-Vektors. Somit führt jeder Sprung zur Adresse 0 zu einem System-Warmstart. Außerdem dient die Verzweigungsadresse in dem Sprungbefehl (Speicheradresse 1, 2) zur Ermittlung der Adresse des BIOS-Vektors.

Tabelle 2.7
Sprungvektor und Interface im BIOS

Offset	Fkt	Name	Funktion
+0000H	00H	BOOT	Initialisierung des Systems (Kaltstart)
+0003H	01H	WBOOT	Reinitialisierung (Warmstart)
+0006H	02H	CONST	Statusabfrage für CONIN
+0009H	03H	CONIN	Eingabe 1 Zeichen von CON
+000CH	04H	CONOUT	Ausgabe 1 Zeichen auf CON
+000FH	05H	LIST	Ausgabe 1 Zeichen auf LIST
+0012H	06H	PUNCH	Ausgabe 1 Zeichen auf PUNCH
+0015H	07H	READER	Eingabe 1 Zeichen von READER
+0018H	08H	HOME	Initialisierung FD-LW
+001BH	09H	SELDSK	FD-LW selektieren
+001EH	0AH	SETTRK	Spur adressieren
+0021H	0BH	SETSEC	SCP-Satz adressieren
+0024H	0CH	SETDMA	Satzpuffer im Speicher adressieren
+0027H	0DH	READ	adressierten Satz in Puffer lesen
+002AH	0EH	WRITE	adressierten Satz aus Puffer schreiben
+002DH	0FH	LISTST	Statusabfrage des Listgerätes
+0030H	10H	SECTAN	Transformation der Satznummer
+0033H	11H	CONST	Statusabfrage CONOUT
+0036H	12H	READERST	Statusabfrage READER
+0039H	13H	PUNCHST	Statusabfrage PUNCH
+003CH		***ENDE DES SPRUNGVEKTORS***	

Tabelle 2.8
Belegung des E/A-Byte und Zuordnung der Subkanäle

Bit	Kanal	Subkanäle			
7/6	LIST	TTY:	CRT:	LPT:	UL1:
5/4	PUNCH	TTY:	PUN:	UP1:	UP2:
3/2	READER	TTY:	RDR:	UR1:	UR2:
1/0	CON	TTY:	CRT:	BAT:	UC1:
Bitbelegung		0/0	0/1	1/0	1/1

Im SCP gibt es 4 logische Kanäle für die *zeichenweise Ein-/Ausgabe (E/A)*:

- CON Dialog mit Bediener
- LIST Ausgabegerät (u. a. Drucker)
- PUNCH sequentielle Zeichenausgabe
- READER sequentielle Zeicheneingabe.

Diesen Kanälen sind über das E/A-Byte auf Adresse (0003H) Subkanäle zugeordnet (Tabelle 2.8).

Die Zuordnung der Driver zu den Subkanälen ist abhängig von der BIOS-Installation. Es können bis zu 4 Driver einem Subkanal zugeordnet werden.

Eine kurze Beschreibung der BIOS-Funktionen erfolgt in Übersicht D.

Die Tabellen der Diskettenparameter sind Bestandteil des BIOS und beschreiben die Organisation der Disketten zur Benutzung im Betriebssystem SCP.

Die Eigenschaften jedes installierten logischen Laufwerkes und dessen BDOS-typische Parameter werden in einer laufwerkspezifischen DPH-Tabelle (disk parameter header) eingetragen (vgl. Tabelle 2.9). Jeder DPH besteht standardmäßig aus 16 Byte, auf die vom BDOS zugegriffen wird. Jeder BIOS-Nutzer (auch BDOS) erhält bei Ausführung des BIOS-Rufes SELDSK als Rückmeldung die Adresse des dem selektierten Laufwerk zugeordneten DPH. Wird als Adresse 0000H übergeben, ist das angesprochene logische Laufwerk nicht installiert.

Innerhalb des DPH wird eine Adresse dpb eingetragen. Diese Adresse verweist auf eine Tabelle DPB (disk parameter block), durch welche die logische und physische Charakteristik des Laufwerkes beschrieben wird (vgl. Tabelle 2.10).

Tabelle 2.9
Aufbau des DPH (disk parameter header)

Offset	Name	Bedeutung
+00H	XLT	Adresse des Transformationsvektors zur Umwandlung der logischen in die physische Satznummer. Wenn XLT:=0000, dann wird keine Transformation durchgeführt.
+02H	0000	vom System reserviert
+04H	0000	vom System reserviert
+06H	0000	vom System reserviert
+08H	DIRBUF	Adresse für einen internen Satzpuffer (Länge 128 Byte), den BDOS-Rufe zur Arbeit mit dem Verzeichnis benötigen.
+0AH	DPB	Adresse des Disketten-Parameterblockes
+0CH	CSV	Adresse des Prüfsummenvektors der Verzeichnissätze
+0EH	ALV	Adresse des Blockbelegungsplanes der Diskette
+10H		ENDE DES DPH

Tabelle 2.10
Aufbau des DPB (disk parameter block)

Offset	Name	Bedeutung
+00H	SPT	Anzahl SCP-Sätze pro Spur
+02H	BSH	Blockverschiebefaktor
+03H	BLM	Funktion der Blockgröße BG: wenn $BG = 128 \times 2^n$ Byte, dann $BSH := n$ Blockmaske (abhängig von Blockgröße BG) wenn $BG := 128 \times 2^n$ Byte, dann $BLM := 2^n - 1$
+04H	EXM	Extentmaske abhängig von BG und DSM: wenn $BG := 1024 \times 2^m$, dann $EXM := 2^m - 1$ für $DSM < 256$ $EXM := 2^{(m-1)} - 1$ für $DSM > 255$
+05H	DSM	Anzahl der Blöcke der Diskette ist $DSM + 1$
+07H	DRM	Anzahl der möglichen Verzeichnis-Eintragungen in den durch ALO/AL1 reservierten Blöcken ist $DRM + 1$
+09H	ALO	Initialwert des 1. Bytes des Blockbelegungsplanes zur Reservierung der Verzeichnisblöcke (Bit 7 gesetzt reserviert Block #0 usw.)
+0AH	AL1	Initialwert des 2. Bytes des Blockbelegungsplanes
+0BH	CKS	Länge des Prüfsummenvektors $CKS := (DRM + 1)/4$
+0DH	DFP	Anzahl der reservierten Systemspuren auf Disk
+0FH		ENDE DES DPB

Übersicht D

BIOS-Funktionen

BOOT	Initialisierung des Systems <ul style="list-style-type: none"> - Kaltstart der Treiber - Voreinstellung Stapel und Verständigungsbereich - Übergang zu Warmstart
WBOOT	Reinitialisierung des Systems <ul style="list-style-type: none"> - Normalisierung Stapel - Einstellung Standard-LW und Nutzer-Nr. aus dem Verständigungsbereich - Einstellung der Adresse Ende-TPA und WBOOT-Adresse - Übergabe der Steuerung CCP
CONST	Statusabfrage von CON <ul style="list-style-type: none"> - Rückgabeparameter in Register A: <ul style="list-style-type: none"> 00h - kein Zeichen im Eingabepuffer FFh - Zeichen liegt vor
CONIN	Eingabe eines Zeichens von CON <ul style="list-style-type: none"> - Rückgabeparameter in Register A: Eingabezeichen
CONOUT	Ausgabe eines Zeichens auf CON <ul style="list-style-type: none"> - Aufrufparameter in Register C: Ausgabezeichen
LIST	Ausgabe eines Zeichens auf LIST <ul style="list-style-type: none"> - Aufrufparameter in Register C: Ausgabezeichen
PUNCH	Ausgabe eines Zeichens auf PUNCH <ul style="list-style-type: none"> - Aufrufparameter in Register C: Ausgabezeichen
READER	Eingabe eines Zeichens von READER <ul style="list-style-type: none"> - Rückgabeparameter in Register A: Eingabezeichen
HOME	Initialisierung des selektierten Laufwerks (LW)
SELDISK	Selektion des LW (Voreinstellung) <ul style="list-style-type: none"> - Aufrufparameter in Register C: LW-Nr. 0, für A, ... 3 für D - Rückgabeparameter in Register HL: Adresse des DPH wenn HL=0, dann existiert das LW nicht
SETTRK	Selektion der Spur (Voreinstellung) <ul style="list-style-type: none"> - Aufrufparameter in Register BC: Spurnummer
SETSEC	Selektion des SCP-Satzes (Voreinstellung) <ul style="list-style-type: none"> - Aufrufparameter in Register BC: Satz-Nummer
SETDMA	Satzpuffer adressieren <ul style="list-style-type: none"> - Aufrufparameter in Register BC: Adresse Satzpuffer, - der Satzpuffer ist ein 128-Byte-Bereich im Arbeitsspeicher für read-/write-Operationen, - Initialbereich 80h - 0FFh
READ	Lesen eines 128-Byte-Satzes von Diskette <ul style="list-style-type: none"> - Quelle ist der mit SELDSK, SETTRK, SETSEC adressierte Satz auf der Diskette, - Ziel ist der mit SETDMA adressierte Puffer - Rückgabeparameter in Register A: <ul style="list-style-type: none"> 00h - Operation OK 01h - Fehler

-
- WRITE** Schreiben eines Satzes auf Diskette
- Quelle ist der mit SETDMA adressierte Puffer
 - Ziel ist der mit SELDSK, SETTRK, SETSEC adressierte Satz auf der Diskette
 - Aufrufparameter in Register C:
Art der Schreiboperation
0: (für Blockung) beim Schreiben bereits zugeordneter Sätze wird vor dem Schreiben gelesen
1: Schreiben Dateiverzeichnis
2: Schreiben neuer (noch nicht zugeordneter) Sätze
 - Rückgabeparameter in Register A:
00h - Operation OK
01h - Fehler
- LISTST** Statusabfrage von LIST
- Rückgabeparameter in Register A:
00h - Gerät noch besetzt
FFh - Gerät frei
- SECTRAN** Transformation der Satzadresse in die SCP-Satznummer für die Funktion SETSEC
- Funktion wird von BDOS automatisch aufgerufen
 - XLT befindet sich in DBP
 - Aufrufparameter in Register BC: alte Satzadresse (0...SPT-1)
DE: Adresse Transformationstabelle
 - Rückgabeparameter in Register HL: transformierte Satzadresse
- CONOST** Statusabfrage von CONOUT
- Rückgabeparameter in Register A:
00h - Gerät noch besetzt
FFh - Gerät frei
- READERST** Statusabfrage von READER
- Rückgabeparameter in Register A:
00h - Kein Zeichen im Eingabepuffer
FFh - Zeichen im Eingabepuffer
- PUNCHST** Statusabfrage von PUNCH
- Rückgabeparameter in Register A:
00h - Gerät noch besetzt
FFh - Gerät frei

2.1.7. Der Verständigungsbereich

Die erste Seite des Hauptspeicherbereiches (Adressenbereich 0000–00FFH) ist als Verständigungsbereich eingerichtet. Dieser Bereich dient der Verständigung zwischen den Betriebssystemkomponenten und den transienten Programmen, die in den TPA geladen und abgearbeitet werden. Zwischen beiden werden Parameter ausgetauscht und Zustandsvariablen eingetragen. Außerdem sind hier Speicherbereiche reserviert, die aufgrund der Hardwarebedingungen der Software nicht zur Verfügung stehen. (Vgl. Tabelle 2.11.)

Tabelle 2.11
Aufbau des Verständigungsbereiches

Adressen	Bedeutung
00H – 02H	Sprungbefehl zum Warmstarteintrittspunkt (WBOOT); damit ist eine einfache Reinitialisierung des Betriebssystems auch vom Nutzerprogramm aus möglich
03H	E/A-Byte zur Zuordnung der Subkanäle für die zeichenweisen logischen E/A-Geräte
04H	enthält das aktuelle Standard-Laufwerk und den aktuellen Nutzerbereich
05H – 07H	enthält den Sprungbefehl nach BDOS; damit können die BDOS-Funktionen aufgerufen werden. Die Sprungadresse kennzeichnet das erste Byte außerhalb des aktuellen TPA-Bereiches.
08H – 2FH	reserviert für die Verzweigungen von RESTART 1 – 5; Bereich wird vom System nicht genutzt
30H – 37H	Ansprungstelle für RESTART 6. (z. Zt. ungenutzt)
38H – 3AH	Ansprungstelle für RESTART 7.; wird vom Debugger genutzt zum Umschalten vom Echtzeitmode in den Debuggermode an gesetzten Unterbrechungspunkten
3BH – 3FH	nicht benutzt, aber reserviert
40H – 4FH	von CCP und BDOS nicht benutzt; kann für BIOS-Eintragungen genutzt werden
50H – 5BH	nicht benutzt, aber reserviert
5CH – 6BH	Kopfzeile des ersten Standard-FCB; Eintragung wird von CCP vorgenommen, dient der Übergabe des ersten Parameters des CCP-Kommandos an transientes Programm
6CH – 7BH	Kopfzeile des zweiten-FCB; Eintragung von CCP als Parameterübergabe des 2. Parameters
5CH – 7FH	Bereich des vollständigen Standard-FCB; Bereich steht transientem Programm zur Verfügung
80H – FFH	Standard-Diskettenpuffer für einen SCP-Satz; Bereich kann vom transienten Programm genutzt werden. Beim Aufruf des transienten Programmes wird vom CCP die Zeichenkette der Parameter der CCP-Kommandozeile eingetragen.

2.1.8. Die Prinzipien und Möglichkeiten der Installation des Betriebssystems SCP

Mit der Bereitstellung des Betriebssystems SCP für den PC 1715 ergibt sich die Notwendigkeit der Variation des SCP, um eine Anpassung an die vorhandenen Hardwareeigenschaften eines speziellen PC 1715 mit geringem Aufwand zu gestatten und den Verwaltungsaufwand im Betrieb klein zu halten. Zu diesem Zweck wurde das Dienstprogramm INSTSCP geschaffen. INSTSCP ermöglicht einerseits die schnelle Installation eines SCP für eine spezielle Maschinenkonfiguration durch den Kundendienst des Herstellerbetriebes und erlaubt andererseits dem Nutzer, bei Erwerb des Dienstprogramms die Änderung am Betriebssystem selbst vorzunehmen.

Eine Installation durch den Nutzer ist z. B. sinnvoll, wenn

- mit verschiedenen Diskettenformaten gearbeitet werden soll
- mehrere Drucker mit verschiedenen Schnittstellen zu bedienen sind
- mit Zusatzlaufwerken gearbeitet werden soll
- eine Erweiterung der Peripherie zu einem späteren Zeitpunkt erfolgt.

Das Betriebssystem SCP ist ein *diskettenorientiertes Betriebssystem*, d. h., das SCP selbst und die zugehörigen Dienstprogramme sowie die Anwenderprogramme sind auf Diskette gespeichert.

Als *Systemspuren* werden auf einer SCP-Diskette die Spuren bezeichnet, die das Betriebssystem, also CCP, BDOS und BIOS, enthalten. Die Systemspuren beginnen stets auf Spur 0, und ihre Anzahl schwankt in Abhängigkeit vom jeweiligen Diskettenformat zwischen 1 und 3.

Der Urlader transportiert nach Einschalten des PC 1715 bzw. nach Betätigen der Taste RESET den Inhalt dieser Spuren an die entsprechenden Betriebssystembereiche im RAM-Speicher.

Das Betriebssystem SCP kann auf Diskette auch in Form einer Programmdatei (COM-File) wie jedes andere ausführbare Programm gespeichert werden. Dadurch besteht z. B. die Möglichkeit, das Betriebssystem wie ein beliebiges transientes Programm zu behandeln und zu aktivieren.

Ein SCP-Betriebssystem, das von den Systemspuren einer Diskette oder durch Eingabe eines transienten Kommandos in den RAM-Speicher des PC 1715 geladen und gestartet wurde, wird als *aktives SCP* bezeichnet.

Der *Installationsvorgang* bezieht sich auf das Variieren der Parameter des SCP. Er wird von INSTSCP in einem Arbeitspuffer vorgenommen. Zu diesem Zweck muß das zu verändernde SCP in diesen Arbeitspuffer geladen werden. Nach erfolgter Variation der Parameter wird der Inhalt des Arbeitspuffers wieder in die gewünschte Speicherform gebracht.

Befindet sich das SCP im Arbeitspuffer, so können die vorhandenen System-

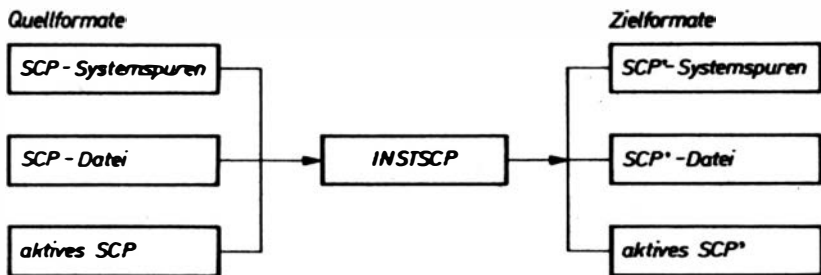


Abbildung 2.5
Arbeitsweisen des Installationsprogramms INST SCP

parameter geändert, angezeigt oder gedruckt werden. Variierbare Parameter sind

- die Diskettenparameter
- die Parameter der byteweisen Ein-/Ausgabetreiber
- die Zuordnung Kanal-Subkanal-Treiber
- die IO-Byte-Belegung
- der Kaltstarttext und das Kaltstartkommando.

Nach Änderung der gewünschten Parameter kann der Inhalt des Arbeitspuffers in die oben genannten Zielformate überführt werden. Beim Aktivieren des SCP direkt aus dem Arbeitspuffer wird INSTSCP verlassen. Diese Variante ist sinnvoll, wenn eine bestimmte Installation nur vorübergehend benötigt wird, um z. B. eine Konvertierung zwischen verschiedenen Diskettenformaten vorzunehmen.

2.2. Das Betriebssystem BROS

Zur Sicherung der Programm- und Datenkompatibilität zwischen dem BC A 5110 und dem PC 1715 wurde das Betriebssystem BROS implementiert. Die Hauptcharakteristika von BROS sind folgende:

- ein auf kommerzielle Probleme optimal zugeschnittener Interpreterbefehlssatz, der eine Untermenge der Makroassemblersprache MABS 1520 unterstützt;
- für Disketten (8" und 5,25") werden Formate entsprechend KROS 5108/01 bzw. KROS 5110/01 realisiert. Damit besteht die Möglichkeit des Datenträger austausches zu allen Geräten mit dieser Norm;
- leistungsfähiges Paket von Dienst- und Hilfsprogrammen, die eine effektive Arbeit zur Entwicklung eigener Anwenderprogramme gestatten;

- kommerzielles PASCAL, das speziell für Bürocomputer entwickelt wurde und die Lösung von Anwenderproblemen auf dem Niveau einer höheren Programmiersprache erlaubt.

BROS gliedert sich in ein residentes Grundsystem und ein variables Teilsystem mit verschiedenen nachladbaren Sprachaufbereitungsprogrammen sowie Dienst- und Hilfsprogrammen.

Das Grundsystem besteht aus

- Steuerprogramm
- Makrobefehlsinterpreter für MABS 1520 Subset
- Makroprogrammtestmonitor.

Der variable Teil umfaßt:

- ASSEM - Makrobefehlsaufbereitungssystem für MABS 1520-Subset
- AS880 - Mikroprogrammsystem für Befehlssatz U880
- DB880 - Mikroprogrammtestmonitor für U880-Programme
- PASCAL 1520 - Compiler für „kommerzielles PASCAL“

Dienst- und Hilfsprogramme

- INIT - Initialisierprogramm für Disketten
- FSERV - Dateiservice
- PSERV - Phasenbibliotheksservice
- SSERV - Quellbibliotheksservice

Als weitere Programme stehen zur Verfügung:

- TEXT - Textprogramm für einfache Nebenbeitextverarbeitung
- REASSY - MABS 1520-Rückübersetzer
- FCOMP - Dateivergleich
- FCHAN - Änderung von Dateiinhalten
- FSOR - Sortieren von Daten im Speicher
- COPYP - Programm zum physischen Kopieren von Disketten gleichen Typs
- TAP62 - Emulationsprogramm zur Nachbildung EC8562/64 asynchrone Übertragung mit der Prozedur AP62/64
- FTS, FTM - Dateiübertragungsprogramme zwischen PC 1715 bzw. BC A 5110 auf Basis der Prozedur AP62/64.

Das Betriebssystem BROS unterscheidet zwei Systembibliotheken

- Quelltextbibliotheken (SOURCELB=SL)
- Phasenbibliotheken (PHASELB=PL)

Für beide Bibliotheken gibt es entsprechende Serviceprogramme zum Einrichten und Verwalten. Die SL dient zur Speicherung von Quelltexten des MABS 1520 Subset. Die PL enthält abarbeitbaren Objektcode.

Das Grundsystem wird von der BROS-Systemdiskette nach Einschalten des PC 1715 durch ein Ladeprogramm in den Speicher geladen. Damit stehen alle Funktionen des Grundsystems dem Nutzer zur Verfügung. Die Speicherbelegung des Grundsystems ist in der Abbildung 2.2 dargestellt.

Der variable Teil des Betriebssystems befindet sich wie auch alle Nutzerprogramme in der Phasenbibliothek auf Diskette und wird durch den Systemlader bei Bedarf in den RAM-Nutzerbereich geladen und gestartet.

Es erscheint notwendig, auf einige Besonderheiten der Implementation von BROS auf dem PC 1715 einzugehen.

Ausgehend von der in Abbildung 2.1 dargestellten Benutzerschnittstelle läßt sich feststellen:

- Nutzerprogramme, die in MABS 1520 geschrieben sind, sichern vollkompatible Anwendersoftwarefunktionen unter folgenden Bedingungen:
 - analoge Hardwarekonfiguration wie beim BC A 5110
 - E/A-Operationen mit MABS 1520-Befehlen
 - ausreichender RAM-Speicherplatz
- Die Betriebssystemfunktionen sind voll kompatibel.

Zu beachten ist, daß die Betriebssystemimplementation eine andere interne Systemstruktur als beim BC A 5110 aufweist und damit der Zugriff auf interne BS-Zellen in Anwenderprogrammen zu Inkompatibilität führt.

- Der Datenträgere Austausch ist unter allen Bedingungen gewährleistet.
- Hardware- bzw. physische E/A-Funktionen sind im allgemeinen in Anwenderprogrammen nicht kompatibel. Eine Ausnahme bildet bedingt der direkte Zugriff auf den Bildwiederholpeicher, der auf den gleichen Adressen wie beim BC A 5110 angeordnet ist. Zu beachten ist jedoch, daß auf Grund der unterschiedlichen Hardware der Bildschirmsteuerung Zeichen außerhalb des ISO-7-Bit-Codes Steuerfunktionen (z. B. inverse Darstellung) auslösen.
- Die Bedienfunktionen sind beim PC 1715 in der beschriebenen Art auf die Tastatur des PC 1715 implementiert, was eine optimale Vereinheitlichung des Bedienkonzepts gewährleistet.

BROS wird für den PC 1715 nicht weiterentwickelt. Es wurden jedoch die notwendigen Betriebssystemerweiterungen zur Nutzung der möglichen Hardwarekomponenten des PC 1715 (z. B. Einbindung eines zweiten Druckers) realisiert.

2.3. Das Betriebssystem JAMB

Das Steuerprogramm des Betriebssystems JAMB wurde auf Basis einer Lizenz von der Leningrader Produktionsvereinigung LEMS, dem Entwickler der in der UdSSR produzierten Bürocomputer „EBM ISKRA 555“ und des Buchungsterminals „EBT NEWA 501“, erworben, um eine Programm- und Datenträgerkompatibilität zwischen diesen Geräten und dem PC 1715 für Anwender in der UdSSR zu erzielen.

Dieses Steuerprogramm wurde mit einem speziell angepaßten E/A-Kern adaptiert, um die Lauffähigkeit von JAMB auf der Hardware des PC 1715 zu gewährleisten.

Das JAMB-Steuerprogramm gestattet

- dialogorientierte Eingabe, Korrektur und Inbetriebnahme von Programmen;
- Abspeichern von Quell- und Objektprogrammen auf Datenträger;
- Laden, Starten, Unterbrechen und Abbrechen von Programmen;
- Ausführen der Programme durch den integrierten Befehlsinterpreter;
- Durchführen von E/A-Funktionen mit verschiedenen physischen Geräten;
- Kontrolle der Programmdurchführung, Behandlung der aufgetretenen Fehler und Ausgabe entsprechender Bedienerinformationen.

Zur Unterstützung der Terminalarbeit enthält der variable Teil des Steuerprogramms einen Modul zur Realisierung der Prozedur AP 70 für ein V.24-Interface und eine rechnerspezifische Prozedur zur Kopplung zweier JAMB-Rechner über IFSS-Interface.

JAMB wurde speziell als Programmiersprache für Buchungsmaschinen entwickelt und ist besonders zur Programmierung von ökonomischen, kommerziellen und statistischen Aufgaben mit belegorientierter Ein-/Ausgabe geeignet. JAMB benutzt auf Quellniveau für die Sprachoperatoren eine russische Mnemonik, und Systemmeldungen erscheinen in russischer Sprache. Nachteilig ist, daß im Betriebssystem JAMB keine Dateioorganisation auf Systemniveau realisiert ist und die Ein- und Ausgabe von Programmen und Daten auf Diskette nur auf physischem Niveau erfolgt.

Die Speicherbelegung des Systems ist in der Abbildung 2.2 dargestellt.

Der Anwenderbereich ist in Programm- und Datensegment unterteilt, die sich jeweils vom Anfang bzw. Ende des Anwenderbereiches gegeneinander aufbauen. Die Struktur des Datensegmentes wird im Programm definiert.

Zwischen beiden Segmenten befindet sich ein sogenannter „freier Bereich“, der mit sequentiell organisierten Datensätzen variabler Wortstruktur belegt

werden kann. Eine Einschätzung zur Kompatibilität ist im Abschnitt 2.4. enthalten.

Zum Betriebssystem gehören außer dem Steuerprogramm die Systemprogramme KOPI und RASM.

- KOPI gewährleistet das Kopieren von JAMB-Systemdisketten, die im Gegensatz zu JAMB-Nutzerdisketten im FM-Verfahren (analog der BROS-Formate) initialisiert und beschrieben werden.

JAMB-Nutzerdisketten sind aus Kompatibilitätsgründen zu den ISKRA- bzw. NEWA-Maschinen im FM-Verfahren beschrieben.

Das Kopieren von JAMB-Nutzerdisketten erfolgt mit entsprechenden JAMB-Operatoren.

- RASM ist ein Programm zum Initialisieren von unformatierten bzw. fehlerhaften Disketten in den entsprechenden Formaten für JAMB-Nutzer- bzw. Systemdisketten. Dabei erfolgt das Erkennen fehlerhafter Spuren und eine entsprechende Anzeige.

Das Betriebssystem JAMB ist für Anwender außerhalb der UdSSR nicht notwendig, da andere adäquatere und leistungsfähige Betriebssysteme zur Verfügung stehen.

2.4. Vergleichende Betrachtungen zu den Betriebssystemen des PC 1715

Die nachfolgende Übersicht zeigt das erreichte Kompatibilitätsniveau der Betriebssysteme des PC 1715. Es kann festgestellt werden, daß die Betriebssysteme des PC 1715 eine ausreichende Portabilität von schon beim Anwender vorhandener Software gewährleisten. Mit dem Betriebssystem SCP wird perspektivischen Entwicklungen in der Anwendung Rechnung getragen.

Die Betriebssysteme BROS und JAMB wurden für den PC 1715 ausschließlich aus Kompatibilitätsgründen bzw. -forderungen entwickelt. Eine Weiterentwicklung dieser Betriebssysteme und dazugehöriger Basissoftware ist nicht vorgesehen. Die Weiterentwicklung von Basissoftware bezieht sich ausschließlich auf das Betriebssystem SCP.

Unter Beachtung der Möglichkeiten der Hardware bzw. deren Weiterentwicklung sind Softwareerweiterungen zur Unterstützung von

- LAN-Lösungen und
 - Semigrafik- bzw. Grafiklösungen
- vorgesehen.

Computer	BS	PC 1715 mit BS	voll Software- kompat.	teilw. Software- kompat.	Datenträger kompatibel
A 5110	BROS SCP5110	BROS SCP1715	x 1) x 1)	x x	x x
A 5120/ A 5130	SCP1520 SIOS	SCP1715 BROS	x 2) x	x x	x x 3)
NEWA 501	JAMB	JAMB	x	x	x
ISKRA 555	JAMB	JAMB		x	x

1) fuer äquivalente Konfigurationen

2) bei Einhaltung von BIOS- bzw. BDDS-Schnittstellen in der Anwendersoftware

3) auf physischer Ebene; mit Einschränkungen auf Dateiebene aufwärtskompatible

Abbildung 2.6

Übersicht zur Kompatibilität der Betriebssysteme des PC 1715

3. Datenübertragung und Terminalnachbildungen mit dem PC 1715 unter dem Betriebssystem SCP

Am PC 1715 stehen die Schnittstellen V.24 und IFSS zur Verfügung (vgl. 1.4.1. und 1.4.2.). Dies sind serielle Schnittstellen, d. h., die Zeichen werden als Kette von Bit gesendet oder empfangen.

Es besteht also die Möglichkeit, mit anderen Computern, die ebenfalls mit diesen Schnittstellen ausgerüstet sind, Daten auszutauschen. Dafür existieren Vorschriften, die auch als Datenübertragungsprotokoll oder Prozedur bezeichnet werden. Insbesondere können Datenübertragungen zwischen dem PC 1715 und Bürocomputern bzw. anderen Personalcomputern sowie anderen SKR-Rechnern und ESER-Rechnern realisiert werden. Dazu steht für den PC 1715 folgende Software zur Verfügung:

- Serieller Ein-/Ausgabetreiber SERIO zur Datenübertragung zwischen PC/BC oder SKR-Rechnern;
- Telekommunikationsprogramm TLC zum Dateiaustausch und zur Arbeit als SKR-Terminal;
- Emulationsprogramm EM 62 zur Nachbildung des Terminals EC 8562/64.

3.1. Der serielle Ein-/Ausgabetreiber SERIO

Im BIOS des SCP sind standardmäßig nur serielle Ausgabetreiber enthalten. Für den Datenaustausch mit anderen Computern sind jedoch bidirektionale Treiber nötig.

SERIO ist eine Betriebssystemerweiterung, die den Datenaustausch asynchron in duplexer Betriebsart ermöglicht. SERIO besteht aus den Komponenten

- Treiber zur Realisierung des Datenaustausches,
- Ladeprogramm zum Einfügen des Treibers in das Betriebssystem,
- Installationsprogramm zur Anzeige/Manipulation der Parameter des Treibers.

SERIO kann als transientes CCP-Kommando auf 2 Arten gestartet werden:

- (1) SERIO (ET) – Aktivierung des Treibers
- (2) SERIO I (ET) – Installation des Treibers

Die Parameter für den Treiber werden der Parameterdatei SERIO.PAR entnommen. (Ist diese Datei auf dem aktuellen Laufwerk nicht vorhanden, so werden Vorgabewerte für die Parameter angenommen.) Zunächst übernimmt das Ladeprogramm die Steuerung. Es fügt den Treiber in das Betriebssystem ein und initialisiert ihn. Die Größe des TPA wird um den Speicherbedarf des Treibers verkürzt.

Nach Ausgabe der Meldung

NEW END OF TPA: xx06H = nn KBYTE

gibt das Ladeprogramm die Steuerung an das CCP zurück.

Mit den spezifizierten BIOS-Rufen kann danach der Datenaustausch erfolgen.

Das Installationsprogramm ermöglicht die Anzeige und Veränderung der logischen und physischen Parameter des Treibers. Die Installation erfolgt menügesteuert. Die einzelnen Funktionen können unabhängig voneinander gerufen werden.

Das Programm meldet sich mit dem Menü:

SERIO	INSTALL-MENUE	Vx.y (PC 1715/BC A 5110)
1	– DEVICE	PC 1715 – V.24
2	– PROCEDURE	DTR
3	– BIT-RATE	9600
4	– BITS/CHARACTER	8
5	– PARITY	NO
6	– STOP-BITS/CHARACTER	1
7	– INPUT	READER
8	– OUTPUT	PUNCH
A	– SAVE NEW PARAMETERS	
B	– START DRIVER	
E	– EXIT	

SELECT:

Nach Auswahl der Funktion werden spezielle Auswahlbilder angezeigt. Im Rahmen der angebotenen Möglichkeiten können die Parameter geändert werden.

1 – DEVICE

Mit dieser Funktion werden das Gerät und der serielle Kanal eingestellt.

2 – PROCEDURE

Auswahl der Prozedur für die Datenübertragung.

3 – BIT-RATE

Die Übertragungsgeschwindigkeit kann damit festgelegt werden.

4 – BITS/CHARACTER

Mit dieser Funktion kann die Anzahl der Bit pro Zeichen eingestellt werden.

5 – PARITY

Hiermit wird die Parität eingestellt, mit der die Zeichen übertragen werden.

6 – STOP-BITS/CHARACTER

Für die Datenübertragung ist die Anzahl der STOP-Bit einzustellen.

7 – INPUT

Zuordnung der Empfangsroutine des Treibers zu einem SCP-Kanal. (Bei der Zuordnung zu CONIN werden die Daten sowohl vom Datenkanal als auch von der Tastatur angenommen.)

8 – OUTPUT

Zuordnung der Ausgaberroutine des Treibers zu einem SCP-Kanal.

A – SAVE NEW PARAMETERS

Diese Funktion speichert die installierten Parameter in der Datei SERIO.PAR. Bei der nächsten Aktivierung von SERIO werden die Parameter dieser Datei entnommen.

B – START DRIVER

Aktivieren des Treibers

E – EXIT

Diese Funktion verzweigt zum CCP, ohne den Treiber zu aktivieren.

3.2. Das Kommunikationsprogramm TLC

Mit dem Programm TLC (Telecommunication) können Daten über serielle Schnittstellen übertragen werden. Dabei wird asynchron und duplex gearbeitet. TLC realisiert dabei folgende Hauptfunktionen:

- Austausch von Dateien,
- Arbeit als Terminal.

Für die Datenübertragung sind zwei Arbeitsweisen möglich:

Computer mit Programm TLC – Computer mit Programm TLC oder
Computer mit Programm TLC – beliebiger Computer mit analogem physischem Übertragungsprotokoll.

Das Programm TLC enthält das Datenaustauschprogramm und das Installationsprogramm. Mit dem Installationsprogramm können die Datenübertragungsparameter angezeigt und geändert werden.

Das Programm TLC kann als CCP-Kommando auf zwei Arten gerufen werden:

- TLC (ET) parameterloser Aufruf,
- TLC par (ET) direkter Aufruf einer TLC-Funktion.

Nach Aufruf des Programms wird eine Datei mit den Datenübertragungsparametern gesucht und geladen. Falls diese Datei nicht existiert, werden Standardwerte initialisiert. TLC meldet sich anschließend mit dem Grundbild (Abbildung 3.1).

Durch Eingabe von N oder Y wird das Programm fortgesetzt, dabei bedeutet

N – Datenaustauschniveau

Y – Installation.

Der Programmteil Datenaustauschniveau meldet sich mit

*

Jetzt steht eine Reihe von Funktionen zur Verfügung, die durch die Eingabe von ? (ET) auch angezeigt werden (Abbildung 3.2).

Die Funktionen lassen sich in Funktionen für Dateitransfer und Terminalarbeit einteilen.

Dateitransfer

Beim Dateitransfer arbeitet auf dem Lokal- und dem Ferncomputer das Programm TLC.

SEND

Mit diesem Kommando können Dateien von einem Computer zu einem anderen Computer übertragen werden. Für die Datei und Laufwerkbezeichnung

LDIR

Das LDIR-Kommando veranlaßt TLC zur Anzeige des Verzeichnisses der angegebenen Diskette des Lokalcomputers.

RDIR

Das RDIR-Kommando veranlaßt TLC zur Anzeige des Verzeichnisses der angegebenen Diskette des Ferncomputers.

LUSER

Mit diesem Kommando kann die aktuelle Nutzerbereichsnummer angezeigt oder ein anderer Bereich eingestellt werden.

RUSER

Die Bedeutung ist analog LUSER und gilt für den Ferncomputer.

NOCONSOLE

Mit NOCONSOLE wird die Ausgabe von Bereitschaftszeichen und anderer Meldungen auf dem Bildschirm verhindert.

BINARY

Nach der Eingabe dieses Kommandos können Daten im 8-Bit-Format gesendet und empfangen werden.

ASCII

Die Daten werden im 7-Bit-Format gesendet und empfangen. ASCII bewirkt, daß ein Byte als zwei ISO-Byte gesendet bzw. zwei empfangene Byte zu einem Byte werden.

MESSAGE

Das MESSAGE-Kommando ermöglicht das Senden einer Nachricht zur Konsole des Ferncomputers.

TRIES

Mit diesem Kommando kann man die Anzahl der Wiederholungen, die TLC beim Senden und Empfangen ausgeführt hat, ermitteln. Nach Abarbeitung der beschriebenen Kommandos wird der Grundzustand des Dateiaustauschniveaus (*) eingenommen.

Terminalarbeit

Dieser Modus wird durch das Kommando TALK erreicht. Auf dem Fern-

computer muß im Programm zur seriellen Datenübertragung mit einfachen Protokollen gearbeitet werden. TLC ist nicht erforderlich.

* ta

CURRENT SETTINGS ARE: FULL DUPLEX, NO PARITY.
SIZE OF TRAPPING BUFFER: 28411 CHARACTERS.

Im Terminalmodus wird jedes über die Tastatur eingegebene Zeichen zum Ferncomputer gesendet. Alle empfangenen Daten werden auf dem Bildschirm angezeigt. Außer diesen einfachen Funktionen können Funktionen über die Taste ESC aktiviert werden. Die Möglichkeiten sind durch ESC? anzeigbar.

* ta

CURRENT SETTINGS ARE: FULL DUPLEX, NO PARITY.
SIZE OF TRAPPING BUFFER: 28411 CHARACTERS.

VALID SEQUENCES ARE:

(ESC) (ESC) SEND 1 ESCAPE TO THE REMOTE COMPUTER
(ESC) S SEND A FILE TO THE REMOTE COMPUTER
(ESC) G GET A FILE FROM THE REMOTE COMPUTER
(ESC) A TOGGLE AUTO LINE FEED ON/OFF
(ESC) C TOGGLE REMOTE ECHO ON/OFF
(ESC) T TOGGLE FILE TRAPPING ON/OFF
(ESC) P TOGGLE PRINTER ON/OFF
(ESC) X CHANGE XON AND XOFF CHARACTERS
(ESC) E EXIT TALK MODE (RETURN TO PROMPT)

(ESC) S

Damit wird das Senden von ASCII-Dateien ermöglicht, deren Name einzugeben ist. Für die Übertragung stehen mehrere logische Übertragungsprotokolle zur Auswahl, wie z. B.

Echoprotokoll

XON/XOFF-Protokoll.

(ESC) G

Die ankommenden Daten werden in einem Puffer und anschließend auf Diskette gespeichert. Ebenso wie beim Senden stehen mehrere Protokolle zur Verfügung.

(ESC) A

Es bewirkt das Einfügen eines zusätzlichen Zeilenvorschubkommandos nach jedem Wagenrücklaufkommando des Nutzers.

(ESC) C

Die Angabe der empfangenen Zeichen auf der Nutzerkonsole wird unterdrückt.

(ESC) T

Der Datei-Auffangpuffer kann hiermit zu- oder abgeschaltet werden.

(ESC) P

Damit kann im TALK-Modus der Drucker ein- und abgeschaltet werden.

(ESC) X

Zum Senden und Empfangen von Dateien kann das XON/XOFF-Protokoll benutzt werden. Die Codes der Zeichen sind

XON = 11Hex und XOFF = 13Hex

Mit (ESC) X kann die Codierung geändert werden.

(ESC) E

Mit (ESC) E wird in den Kommando-Mode * zurückgekehrt.

Das Installationsprogramm meldet sich mit einem Grundbild (MAIN MENUE) (Abbildung 3.3).

Die aktuellen Werte der Parameter werden angezeigt. Nach Auswahl des

TLC – INSTALLATION

MAIN MENUE

=====

ACTUALLY

1 – EXIT	
2 – DEVICE	PC 1715 – V.24
3 – SPEED	9600 BAUD
4 – PARITY	NO
5 – STOP-BITS	1
6 – BITS/CHARACTER	8
7 – PROCEDURE	DC1/DC3
8 – CR-ECHO	CR
9 – SAVE	
SELECT: 2	

Abbildung 3.3
Grundbild des
Installationsprogramms
(TLC)

Parameters werden spezielle Auswahlbilder angezeigt. Im Rahmen der angebotenen Möglichkeiten können die Parameter geändert werden.

1 – EXIT

Das Installationsprogramm kann damit verlassen und in das Grundbild von TLC zurückgekehrt werden.

2 – DEVICE

Mit dieser Funktion können das Gerät und der serielle Kanal eingestellt werden.

3 – SPEED

Die Übertragungsgeschwindigkeit kann damit festgelegt werden.

4 – PARITY

Hiermit wird die Parität eingestellt, mit der die Zeichen übertragen werden sollen.

5 – STOP-BITS

Für die Datenübertragung ist die Anzahl der STOP-Bit einzustellen.

6 – BITS/CHARACTER

Mit dieser Funktion kann die Anzahl der Bit pro Zeichen eingestellt werden.

7 – PROCEDURE

Die eingestellte Prozedur ist nur im Terminalmode wirksam. Wird in diesem Modus eine Datei gesendet oder empfangen, ist die eingestellte Prozedur nur bei den Protokollen "No Protocol", "Time Wait" und "XON/XOFF" gültig. In allen anderen Fällen wird ohne Prozedur gearbeitet.

8 – CR-ECHO

Mit dieser Funktion wird eingestellt, wie im Terminalmode auf ein CR vom Ferncomputer geantwortet wird.

9 – SAVE

Die eingestellten Parameter können in einer Datei abgespeichert werden. Diese wird bei Programmstart ausgewertet.

Die Parameter, die beim Aufruf übergeben werden können, entsprechen den unter Dateitransfer beschriebenen Funktionen (*). Nach der Ausführung wird TLC verlassen und CCP die Steuerung übergeben.

3.3. EM62 – Emulation des Terminals EC 8562/64

EM62 bildet das Anschluß- und Geräteverhalten des Terminals EC 8562/64 auf dem PC 1715 nach. Damit ist es möglich, mit dem PC 1715 als Terminal an ESER- oder SKR-Rechnern, die die Geräte EC 8562/64 in ihren Zugriffsmethoden unterstützen, zu arbeiten. Daneben können Textfiles mit den ESER/SKR-Rechnern ausgetauscht werden.

Das Programm besteht aus folgenden Teilen:

- Emulation des EC 8562/64,
- Dateiaustausch,
- Installation der Parameter für den Datenaustausch.

EM62 wird als transientes CCP-Kommando in der Form

```
EM62(ET)
```

aktiviert.

Es meldet sich mit der in Abbildung 3.4 gezeigten Displayanzeige.

In dem Bild werden die aktuellen Parameterwerte des EM62 angezeigt. Diese Werte werden einer Parameterdatei EM62.PAR entnommen (bei Fehlen der Datei werden Vorgabewerte angenommen). Die Konsoleingabe Y verzweigt zur Parameterinstallation, N zur Emulation.

Der Datenaustausch zwischen Rechnern und den Terminals EC 8562/64 erfolgt mittels der Prozedur AP62. Bei dieser Prozedur werden alle Aktivitäten zum Datenaustausch vom Rechner (Steuerrechner oder steuernde Station)

```

*****  **      **  *****  *****  (C) RBWS V x.y
**      ****  ****  **      **  **  **
**      ** ** ** ** ** **      **
*****  **  ***  **  *****  *****
**      **      **  **  **  **  **
**      **      **  **  **  **  **
*****  **      **  *****  *****

```

```

DEVICE: PC 1715 – V.24          FORMAT DISPLAY: 24 * 80
SPEED:  9600 BAUD
MODE:   HALF-DUPLEX           TERMINAL:      31H
L111:   LOW                   DISPLAY:     32H
BLOCK:  OFF                    PRINTER:    33H

```

CHANGE PARAMETERS (Y/N): N

Abbildung 3.4
Displayanzeige EM62

über Kommandos gesteuert. Das Terminal (abhängige Station) kann nur auf Kommandos des Steuerrechners Daten senden bzw. empfangen.

Im Status der Emulation kann der Bediener

- als Terminal EC 8562/64 arbeiten,
- durch Eingabe von CTRL-F in einen Filemodus wechseln,
- mit CTRL-C in das Grundmenü zurückkehren.

Terminalmodus

EM62 realisiert aus Sicht des steuernden Rechners folgende Funktionen.

- (1) Die Übertragung von Nachrichten des Terminals zum Steuerrechner (polling).
 - Lesen bedient
 - Lesen mit Position bedient
 - Lesen des gesamten Displays unbedient.
- (2) Die Übertragung von Nachrichten des Steuerrechners zum Terminal (select).
 - Schreiben ab Cursorposition
 - Schreiben ab adressierter Zeile
 - Schreiben ab adressierter Position
 - Schreiben mit vorherigem Löschen des gesamten Displays
 - Schreiben mit vorherigem Löschen der ungeschützten Displaybereiche.
- (3) Das Übertragen von Nachrichten ist auch auf das Listgerät möglich.

Für die Bedienung des Terminals wurden folgende Funktionstasten nachgebildet:

- Tasten für horizontale und vertikale Cursorpositionierung bzw. Tabulierung,
- Tasten zum Löschen von Zeichen, Zeilen und Gesamtdisplay,
- Tasten zum Einfügen von Zeichen und Zeilen,
- Tasten für die Nachrichtensteuerung (Beginn Nachricht, Ende Nachricht, Senden Nachricht, Stornieren Nachricht),
- Taste zum Anfertigen einer Kopie des Displays auf dem Listgerät.

Für die Nachbildung dieser Funktionstasten werden die Kursortasten und die Tasten F1...F12 genutzt.

Filemodus

Der Filemodus wird aus dem Terminalmodus erreicht. Beim Verlassen des Filemodus wird wieder in den Terminalmodus zurückgekehrt. Im Filemodus besteht die Möglichkeit, Daten von Diskettenfiles zum steuernden Rechner zu senden bzw. empfangene Daten in Diskettenfiles abzuspeichern. Für den

Datenaustausch wird die Prozedur AP62 genutzt. Als Kommandos werden jedoch nur die Kommandos für Schreiben und Lesen ohne Positionierung und Löschen interpretiert. Im Filemodus rollt das Display, die Funktionstasten des Terminalmodus sind unwirksam.

Beim Erreichen des Filemodus wird folgendes Bild ausgegeben:

=== EM62 - FILE-TRANSMISSION ===

THE POSSIBLE COMMANDS ARE:

<ESC> <ESC> - ENTER ESC
<ESC> S - SEND A TEXT-FILE
<ESC> G - GET A TEXT-FILE
<ESC> R - EXIT TO EMULATION
<ESC> D - DISPLAY DIRECTORY
<ESC> P - PRINTER ON/OFF

Durch die Eingabe von <ESC>, gefolgt von einem Zeichen des Menüs, können folgende Funktionen aktiviert werden.

<ESC> <ESC>

Ermöglicht es, ein <ESC> in den Sendetext einzufügen.

<ESC> S

Führt zur Eröffnung eines vom Bediener zu spezifizierenden Files. Lesege-
suche eines steuernden Rechners werden mit Daten aus dem eröffneten File
beantwortet.

<ESC> G

Führt zur Eröffnung eines Empfangsfiles, das vom Bediener zu spezifizieren
ist. Alle Empfangsnachrichten werden zunächst zu einem Empfangspuffer
und später auf Diskette gespeichert.

<ESC> R

Kehrt in den Terminalmodus zurück.

<ESC> D

Gibt das Dateiverzeichnis aus.

<ESC> P

Schaltet das Listgerät zu oder ab.

Das Installationsprogramm ermöglicht die Anzeige und Veränderung der lo-
gischen und physischen Parameter des Emulators. Die Installation erfolgt

menügesteuert. Die einzelnen Funktionen können unabhängig voneinander gerufen werden.

Das Programm meldet sich mit dem Menü (Abbildung 3.5).

Nach Auswahl der Funktion werden spezielle Auswahlbilder angezeigt. Im Rahmen der angebotenen Möglichkeiten können die Parameter geändert werden.

1 – EXIT

Das Installationsprogramm kann damit verlassen und in das Grundbild von EM62 zurückgekehrt werden.

2 – DEVICE

Mit dieser Funktion können das Gerät und der serielle Kanal eingestellt werden.

3 – FORMAT DISPLAY

Das Format des nachzubildenden Displays wird ausgewählt.

4 – SPEED

Die Übertragungsgeschwindigkeit kann damit festgelegt werden.

5 – MODE

Auswahl, ob duplex oder halbduplex gearbeitet werden soll.

EM62 – INSTALLATION

MAIN MENUE

=====

	ACTUALLY
1 – EXIT	
2 – DEVICE	PC 1715 – V.24
3 – FORMAT DISPLAY	24 * 80
4 – SPEED	9600 BAUD
5 – MODE	HALF-DUPLEX
6 – L111	LOW
7 – BLOCK	OFF
8 – TERMINAL	31H
9 – DISPLAY	32H
A – PRINTER	33H
B – SAVE	

SELECT:

Abbildung 3.5

Grundbild des Installationsprogramms (EM62)

6 - L111

Dient zur Einstellung der Arbeitsweise des Modems.

7 - BLOCK

Stellt ein, ob die Nachrichten zeilenweise geblockt werden oder nicht.

8 - TERMINAL

Auswahl der Terminaladresse.

9 - DISPLAY

Auswahl der Displayadresse.

A - PRINTER

Auswahl der Adresse des Listgerätes.

B - SAVE

Diese Funktion dient zum Abspeichern der Parameter in der Datei EM62.PAR. Bei einem erneuten Aktivieren vom EM62 werden dann die Parameterwerte dieser Datei entnommen.

4. Entwicklung von Systemprogrammkomponenten mit dem PC 1715

Wie im 1. Kapitel ausgeführt wurde, arbeitet der PC 1715 auf der Grundlage eines Mikroprozessors, der eine gewisse Menge von Kommandos versteht. Die Kommandos, die dem Befehlssatz des Mikroprozessors entsprechen, befinden sich im Programmspeicher. Durch eine Folge von solchen Kommandos, die Maschinenbefehle genannt werden, entsteht ein Programm.

Eine Programmerstellung in der sogenannten Maschinensprache ist sehr zeitaufwendig und erfordert eine hohe Konzentration und genaueste Kenntnis des Befehlssatzes des Mikroprozessors. Auftretende Fehler sind nur sehr umständlich zu beseitigen. Um die Programmentwicklung zu vereinfachen und leichter erlernbar zu gestalten, wurde jedem Kommando in Maschinensprache, welches ein- oder mehrbytig sein kann, eine symbolische Darstellung zugeordnet. Eine solche Darstellung wird Mnemonik genannt. Neben den Maschinenbefehlen können Adressen und Parameter im Programm symbolisch dargestellt werden. Die Gesamtheit dieser symbolischen Zeichen (das Alphabet), ihre Verknüpfung (die Grammatik oder Syntax) und inhaltliche Bedeutung (die Semantik) bilden eine Assemblersprache.

Die Darstellung eines Programms in einer Assemblersprache versteht jedoch nicht der Rechner. Um das Programm in eine dem Rechner verständliche Maschinensprache zu übersetzen, wird ein Übersetzungsprogramm, der sogenannte Assembler, verwendet.

Grundeigenschaften einer Assemblersprache sind:

- Sie ist an einen bestimmten Prozessortyp gebunden, so daß nur eine maschinenbezogene Programmierung möglich ist. Jede Änderung der Gerätetechnik bedeutet auch eine unmittelbare Änderung des Programms.
- Sie spiegelt den Befehlssatz und die Architektur (Register, Organisation des Statusspeichers und des Unterbrechungssystems, Adressierungsarten usw.) des Prozessors wider.
- Die Programmstruktur bleibt beim Übersetzungsvorgang unverändert, da jede in Assemblersprache geschriebene Anweisung genau einem Maschinenbefehl entspricht.

Über die Tastatur wird das in einer bestimmten Assemblersprache geschriebene Programm, das sogenannte Quellprogramm, eingegeben. Die Gesamtmenge der eingegebenen Zeichen wird als Quelltext gespeichert. Beim PC 1715 erfolgt die Eingabe über einen der verfügbaren Editoren, der sowohl die Eingabe als auch die Korrektur des Quelltextes im Dialog ermöglicht. Der Assembler analysiert während des Übersetzungsvorganges das eingegebene Quellprogramm Zeichen für Zeichen und bildet als Ergebnis der Übersetzung eine Folge von Maschinenbefehlen, die jedoch noch nicht abarbeitbar ist, da noch alle Speicheradressen relativ zu einer sogenannten Leitadresse aufgeführt sind. Dadurch ist das Programm im Speicher frei beweglich. Für Dokumentationszwecke können der Quelltext und das Maschinenprogramm ausgedruckt werden.

Durch einen Programmverbinder (auch Linkprogramm genannt) werden die noch vorhandenen relativen Adressen in absolute umgewandelt und das Programm durch die Angabe der Leitadresse an die gewünschte Stelle im Speicher transformiert. Die entstandene Programmphase (Lademodul) wird auf der Diskette gespeichert.

Nach Entwurf und Entwicklung beginnt das Testen des Programms. Dafür steht ein leistungsfähiges Testsystem, der sogenannte Debugger, zur Verfügung. Das Programm wird mit diesem Testsystem schrittweise, in Einzelschritten oder Komplexen, in Betrieb genommen. Die internen Zustände des Mikroprozessors (CPU-Register) können jederzeit betrachtet werden.

Werden Fehler im Programm festgestellt, so wird das Quellprogramm entsprechend korrigiert, neu übersetzt, gebunden, und die Testung kann fortgesetzt werden.

Es besteht aber auch die Möglichkeit, das Programm auf dem Maschinenniveau zu korrigieren und als COM-Datei zu speichern. Die Abbildung 4.1 gibt einen Überblick über die Struktur einer Programmdarstellung von der Programmeingabe bis zur Programmtestung.

Für die verwendeten Dateitypen gilt:

MAC : in Assemblersprache geschriebenes Quellprogramm

REL : verschiebbliche Objektcodedatei

PRN : druckbare Datei (aus Übersetzung entstanden)

CRF : Cross-Referenz-Datei

HEX : Programmdatei im hexadezimalen Format

SYM : Symboldatei (verwendet globale Symbole)

COM : ausführbare Datei (in Maschinensprache)

Die Assemblerprogrammierung wird nur im Zusammenhang mit der Lösung

spezieller Aufgabenstellungen eingesetzt. Solche speziellen Anforderungen können unter anderem sein:

- Entwicklung von Betriebssystemkomponenten (zum Beispiel Treiberprogramme für spezielle Gerätetechnik),
- Lösung von Echtzeitaufgaben,
- Steuer- und Regelprozesse,
- Aufgaben der Laborautomatisierung,
- allgemeine Automatisierungsaufgaben.

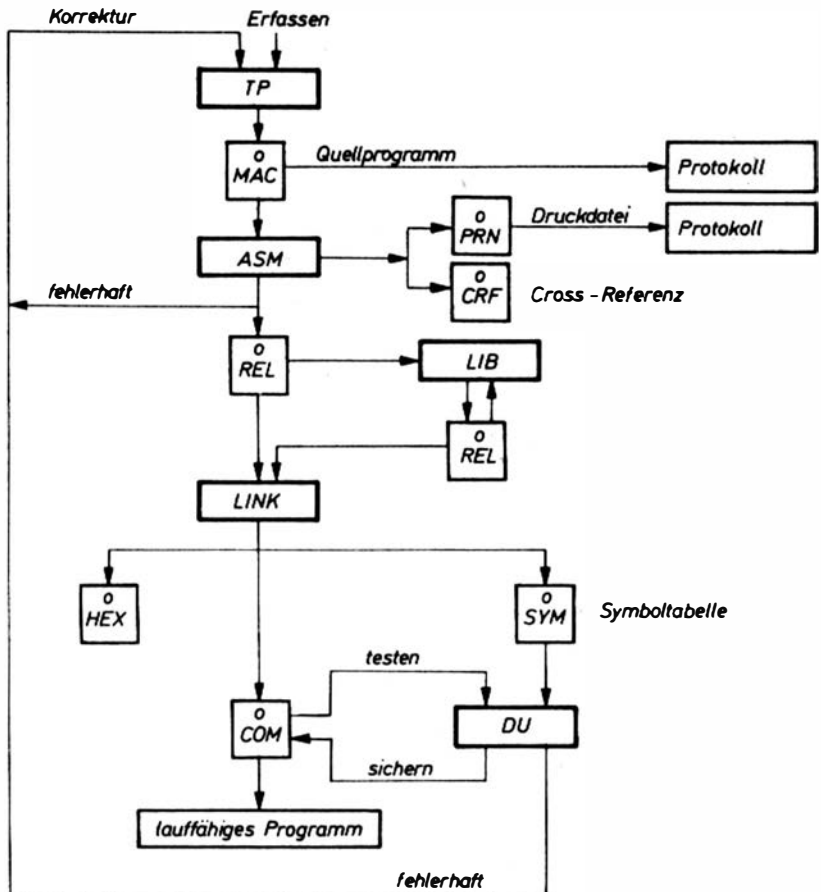


Abbildung 4.1

Überblick über die Erstellung eines Assemblerprogramms

Mit aller Deutlichkeit soll an dieser Stelle jedoch festgestellt werden: Es entspricht nicht mehr dem heutigen Erkenntnis- und Entwicklungsstand der Programmiermethodik und Softwaretechnologie, wenn beliebige Nutzerprobleme in einer Assemblersprache entwickelt werden. Für den PC 1715 stehen dafür eine Reihe von Programmiersprachen (siehe 5. Kapitel) und vor allem leistungsfähige Standardsoftware (siehe 6. Kapitel) zur Verfügung. Das Verhältnis von Aufwand und Nutzen ist auf der Ebene der Assemblerprogrammierung am geringsten, während es auf der Ebene der Standardsoftware am günstigsten ausfällt.

Im folgenden werden die für eine effektive Assemblerprogrammentwicklung für den PC 1715 erforderlichen Softwaresysteme charakterisiert. Als einzusetzender Editor für die Erfassung und Korrektur des Quelltextes eines Assemblerprogramms wird dabei der Textprozessor TP empfohlen, der im 6. Kapitel beschrieben wird.

4.1. Der Assembler ASM

Als Übersetzer für in der Assemblersprache geschriebene Programme steht für den PC 1715 der Assembler ASM zur Verfügung.

Seine Leistungsfähigkeit wird im wesentlichen durch

- die Einzelfunktions-Pseudooperationen,
- die Möglichkeiten der Makroprogrammierung,
- die Möglichkeit der bedingten Assemblierung bestimmt.

Einzelfunktions-Pseudooperationen weisen den Assembler auf die Ausführung einer bestimmten Funktion hin (sie stellen keine Maschineninstruktion für den Mikroprozessor dar). Sie dienen z. B. zur

- Auswahl der verwendeten Assemblersprache (8080- oder Z80-Mnemonic),
- Definition von Daten und Symbolen,
- Verwaltung des Speicherplatzes,
- Steuerung des Formats der Übersetzungsliste,
- Steuerung des Inhaltes der Übersetzungsliste,
- Eintragung von ein- und mehrzeiligen Kommentaren,
- Einfügung von anderen Quelldateien.

Der Assembler unterstützt die Möglichkeit, mit Makros zu arbeiten. Ein Makro ist eine Anzahl von Anweisungen mit einem bestimmten Namen. Die Anweisungen werden an der Stelle im Programm eingetragen, wo sie durch den Namen aufgerufen werden.

Bei Verwendung von Makros wird der Umfang der Quelldatei reduziert, da die Definition des Makros in einer anderen Quelldatei stehen kann. Die Anweisungen werden nur in den zu übersetzenden Modul übernommen, wenn sie während der Assemblierung gefordert werden. Bei bedingter Assemblierung werden im Quellprogramm Bedingungen vorgegeben, nach denen dann Teile des nachfolgenden Programms assembliert oder nicht assembliert werden.

Die Quelldatei für den Assembler besteht aus mehreren Anweisungszeilen, die in der entsprechenden Assemblersprache geschrieben sind.

Die letzte Zeile der Quelldatei muß die END-Anweisung enthalten. Außerdem muß sie durch die Taste (ET) abgeschlossen sein.

Solche Anweisungszeilen sind in maximal 4 Teile untergliedert:

Symbol	Operation	Argument	Kommentar
--------	-----------	----------	-----------

Für Symbol können

- interne Marken (gelten nur für diesen einen Modul),
 - globale Marken (können von anderen Modulen und intern verwendet werden),
 - externe Marken (sind in einem anderen Modul definiert)
- stehen.

Nach dem Symbol folgt sonst unmittelbar ein Doppelpunkt. Eine Ausnahme bilden Anweisungen für den Assembler. Hier darf kein Doppelpunkt stehen (z. B. bei MAKRO- oder EQU-Anweisung). Eine Anweisungszeile kann auch ohne Symbol sein.

Der Teil Operation enthält eine Mnemonik des Prozessorbefehls, eine Pseudoperation, einen Makro-Namen oder einen Ausdruck. Für Argument können Ausdrücke, Variablen, Registernamen, Operanden und Operatoren stehen.

Der Kommentarteil muß mit Semikolon beginnen und dient der Erläuterung von Programmteilen.

Wenn der Quelltext vollständig erfaßt wurde, kann zum Assemblieren übergegangen werden.

Der Assembler übersetzt die gesamte Menge der Anweisungen der Quelldatei einschließlich der Pseudowiederholungsoperationen und der eingesetzten Makros.

Die Übersetzung der Quelldatei erfolgt in zwei Durchläufen, die auch als Pässe bezeichnet werden.

Während des 1. Durchlaufes (Paß 1) wertet der Assembler die Programmanweisungen aus, berechnet, wieviel Byte die Anweisungen erzeugen, sucht die verwendeten symbolischen Darstellungen für Konstanten, Variablen, Adres-

sen sowie für Äquivalenzen und bildet daraus eine Symboltabelle. In dieser ist jedem Symbol ein Wert zugewiesen. Es erfolgt eine Syntaxkontrolle.

Während des 2. Durchlaufes (Paß 2) werden für die symbolischen Darstellungen die Werte aus der gebildeten Symboltabelle eingesetzt. Es erfolgt wiederum eine Syntaxkontrolle.

Festgestellte Fehler werden auf der Konsole ausgegeben und entsprechend der Fehlerart durch ein Merkmal in der ersten Position der Zeile gekennzeichnet.

Als Ergebnis der beiden Durchläufe entsteht ein verschieblicher Objektcode, der mit dem Programmverbinder gebunden und geladen werden kann.

Der verschiebliche Objektcode kann in eine Diskettendatei, die vom Assembler die Dateitypkennung .REL bekommt, geschrieben werden. Die übersetzte Datei ist noch nicht ausführbar. Sie wird es erst nach der Bearbeitung durch den Programmverbinder.

Es ist möglich, für die Übersetzung Schalter am Ende der Kommandozeile einzugeben. Sie bewirken z. B.

- das Anlegen einer Cross-Referenzliste in einer Datei mit der Typkennung .CRF (sie enthält alle Adreßverweise innerhalb des Quellprogramms mit den Speicherplätzen, wo sie definiert sind),
- die Ausgabe von Adressen und Befehlscode in eine Druckdatei oktal oder hexadezimal,
- das Erzeugen einer druckbaren Datei mit der Typkennung .PRN,
- die Initialisierung von Speicherbereichen mit dem Wert 00H.

Der Assembler kann entsprechend den SCP-Konventionen auf zwei Arten aufgerufen werden:

- ASM <kommandozeile>
- ASM

Im zweiten Fall ist die Kommandozeile erst nach Aufforderung durch den Assembler anzugeben. Dabei ist zu beachten, daß in diesem Falle nur Großbuchstaben akzeptiert werden. Durch einen bestimmten Aufbau der Kommandozeile kann gesteuert werden, ob

- die Übersetzung in eine Datei vom Typ .REL,
- die Übersetzung auf den Bildschirm,
- die Übersetzung auf den Drucker oder nur
- ein reiner Syntaxkontrolllauf

erfolgen soll. Eine ausführliche Kommandozeile hat die folgende Struktur: Objektdatei, Druckdatei=Quelldatei/Assemblerschalter ...

4.2. Der Programmverbinder LINK

Um eine ".REL"-Datei ausführbar zu machen, muß sie mit dem Programmverbinder LINK geladen und gebunden werden. *Laden* bedeutet hierbei, daß jedes übersetzte Objektprogramm in den Speicher geladen wird und daß die vorhandenen relativen Adressen in absolute umgewandelt werden. Das ist notwendig, um ein verschiebliches ".REL"-Programm in ein ausführbares Programm (Datei mit der Dateitypkennung .COM) umzuwandeln. *Binden* bedeutet, daß alle externen Bezüge aufgelöst werden. Es wird mit den entsprechenden Objektprogrammen gekoppelt, in denen diese Bezüge definiert sind. Im Programmverbinder können in der Kommandozeile noch Schalter gesetzt werden, die besondere Funktionen veranlassen, wie z. B.

- Schreiben der gebundenen ".COM"-Datei auf die Diskette,
- Festlegen des Namens der ".COM"-Datei,
- Anlegen einer ".SYM"-Datei (für alle global definierten Adressen),
- Durchsuchen von bestimmten Bibliotheken nach Modulen oder Definitionen für globale Symbole,
- sofortige Ausführung des Programmes nach dem Binden,
- Setzen der Leitadressen für Programm und Daten,
- Festlegen der Zahlendarstellung (hexadezimal oder oktal),
- Ausgabe der undefinierten globalen Symbole,
- Grundzustand des Programmverbinders herstellen (Rücksetzen).

Es ist zu beachten, daß die Objektprogramme in der Reihenfolge geladen werden, wie sie in der Kommandozeile angegeben sind, und somit auch in dieser Reihenfolge die externen Bezüge aufgelöst werden. Begonnen wird ab Adresse 103H. Auf Adresse 100H-102H steht ein Sprung zum logischen Programmanfang.

Die Objektprogrammnamen sowie die Schalterangaben sind in der Kommandozeile in Großschreibung einzugeben.

4.3. Das Bibliotheksverwaltungsprogramm LIB

Im Zusammenhang mit dem Programmverbinder sei hier auf das Dienstprogramm LIB zum Bilden und Verwalten einer Bibliothek aus übersetzten Objektprogrammen hingewiesen.

Vor der Benutzung von LIB wird empfohlen, eine Sicherheitskopie der zu verwendenden Bibliotheken anzulegen, da auch Bibliotheken leicht zerstört werden können.

Beim Bilden einer Bibliothek werden mehrere Quelldateien vom Typ .REL

zu einer unter einem gemeinsamen Namen stehenden Datei zusammengefaßt.

Dabei sollte eine solche Reihenfolge der einzelnen Moduln gewählt werden, daß externe Bezüge durch nachfolgende Teile aufgelöst werden und nicht umgekehrt.

In der Kommandozeile können Schalter gesetzt werden, wie z. B.

- Ausgabe einer Liste der Modulnamen in der Bibliothek mit deren globalen und externen Bezügen,
- Ausgabe einer Liste von Symbolen, die in einem Durchlauf nicht aufgelöst werden,
- Zahlendarstellung des Anzeigemodus.

Der Vorteil der Bildung einer Bibliothek liegt darin, daß alle Routinen, die für die Ausführung eines Programmes benötigt werden, in einer gemeinsamen „Quelle“ zusammengefaßt sind. Das können beispielsweise gewisse Unterprogrammrountinen sein, die dann jedem zugänglich sind. Es müssen aber die entsprechenden Modulnamen bekannt sein.

Sie werden dann mit dem Aufruf innerhalb des Programmverbinders LINK (Bibliotheksname mit Schalter /S) zu einer .COM-Datei zusammengefaßt.

4.4. Der Debugger DU

Das Systemprogramm „DU“ dient zur Unterstützung des Programmierers bei der Inbetriebnahme und Testung seiner Maschinenprogramme im Echtzeit-Modus.

Das Inbetriebnahmesystem wird im Dialog über Kommandos angesprochen. Der Debugger ist ein verschiebliches Programm und wird mit dem Laden zunächst ab Adresse 100H im Speicher abgelegt und anschließend an das Ende des TPA geladen. Die untere Grenze des Debuggers wird im Verständigungsbereich als neue obere Grenze des TPA gesetzt. Für den Bediener steht somit wieder der Bereich ab Adresse 100H bis zum Ende des TPA für seine Programmtestung zur Verfügung. Standardmäßig werden durch „DU“ geladene Programme ab der Adresse 100H geladen, wobei aber auch verschiebliches Laden möglich ist.

Folgende Funktionskomplexe werden durch „DU“ realisiert:

- a) Vergleich, Suche und Modifikation von Daten im Speicher
 - E – Vergleich der Belegung zweier Speicherbereiche
 - K – Suche von bestimmten Bytefolgen im Speicher
 - D – Ausgabe eines Speicherbereiches

- S – byteweise Modifikation des Speicherbereiches
- F – Füllen eines Speicherbereiches mit einer Konstanten
- M – Kopieren eines Speicherbereiches in einen anderen Speicherbereich
- b) Assemblierung und Reassemblierung
 - A – Befehlseingabe in mnemonischer Form über Konsole in den Speicher
 - L – Anzeige eines Programmbereiches in mnemonischer Form
- c) Echtzeittest von Programmen und Manipulation der CPU-Register
 - P – Setzen, Löschen und Anzeige von Haltepunkten
 - G – Starten bzw. Fortsetzen im Echtzeitbetrieb
 - T – Ausführung von Befehlsschritten mit Spurprotokollierung
 - U – Ausführung von Befehlsschritten ohne Spurprotokollierung
 - X – Anzeige und Modifikation der Test-CPU
- d) Aufruf von Systemunterprogrammen
 - C – Aufruf von zusätzlichen Systemunterprogrammen
- e) Dateiarbeit
 - I – Eingabe von CCP-Kommando Parametern, Belegen von Dateisteuerblöcken und Zeichenpuffer im SCPX-Verständigungsbereich
 - R – Einlesen von Dateien in den Arbeitsspeicher des Debuggers
- f) Hilfsfunktionen
 - H – Anzeige der Symboltabelle, hexadezimale Addition und Subtraktion.

Die Kommandozeile hat eine Kapazität von 64 Zeichen und wird über die BDOS-Funktion 10 realisiert.

Als Trennzeichen zwischen den einzelnen Parametern sind Leerzeichen oder Komma erlaubt.

In den Parametern in der Kommandozeile können auch symbolische Adressen verwendet werden. Das setzt aber voraus, daß die entsprechende Symboldatei (SYM) geladen wurde. Diese wird beim Binden durch das Setzen des entsprechenden Schalters gebildet. Die Symboldatei wird in den freien TPA-Bereich unmittelbar unterhalb des Debuggers geladen. Dem Nutzer geht dies als freier Bereich verloren.

Der Debugger kann durch Wegnahme des Assemblerteiles um 3,5 KByte verkleinert und somit der freie TPA für den Nutzer um diesen Bereich vergrößert werden. Sollen diese Funktionen wieder aktiviert werden, so muß DU erneut geladen werden.

5. Die Programmiersprachen im Betriebssystem SCP

Im Betriebssystem SCP für den PC 1715 steht dem Nutzer eine Auswahl leistungsfähiger Programmiersprachen zur Verfügung. Dies ermöglicht, die zu lösenden Aufgaben in jener Programmiersprache zu bearbeiten, in der die Lösung effizient beschrieben werden kann. Auch an dieser Stelle muß aber darauf aufmerksam gemacht werden, daß nicht a priori jede Softwareentwicklung nur in einer dieser verfügbaren Programmiersprachen entwickelt werden muß. Vielmehr muß stets sorgfältig geprüft werden, ob nicht eine noch einfachere Aufgabenlösung unter Ausnutzung der im 6. Kapitel beschriebenen Standardsoftware möglich ist. Erst wenn diese Prüfung ergibt, daß dies aus objektiven Gründen nicht möglich ist, sollte die benötigte Software in einer der hier aufgeführten oder sonst für das Betriebssystem SCP verfügbaren Programmiersprachen entwickelt werden.

5.1. BASIC-Interpreter und BASIC-Compiler

Die Programmiersprache BASIC (*Beginners All Purpose Symbolic Instruction Code*) wurde 1964 als Dialogsprache für die Ausbildung von Studenten entwickelt. Erst mit der Entwicklung der Mikroelektronik und ihrem Einsatz in der Rechen- und Informationsverarbeitungstechnik gelangte BASIC dann zu internationaler Verbreitung. BASIC zählt heute zu den am meisten verwendeten Programmiersprachen in der Welt. Die Sprache BASIC zeichnet sich durch große Leistungsfähigkeit, einfache Handhabbarkeit und problemadäquate Notierungen aus, welche dem Nutzer eine einfache und schnelle Programmentwicklung ermöglicht, ohne sich vorher umfassende Kenntnisse in der Rechentechnik/Informationsverarbeitung aneignen zu müssen.

Durch eine *interaktive Dialogprogrammierung* können einzelne Anweisungen, Programmfragmente oder ein vollständiges Programm sofort nach dem Editieren ausgeführt und getestet werden.

BASIC ist nicht nur eine Programmiersprache, sondern beinhaltet auch alle

notwendigen Funktionen eines dialogorientierten Betriebssystems. Dabei wird unterschieden zwischen den Programmanweisungen zur Programmierung einer Anwenderaufgabe und den Programmierdirektiven (Kommandos) zur Unterstützung des Programmierers.

Im Betriebssystem SCP für den PC 1715 wird für die Programmiersprache BASIC sowohl ein Interpreter (BASI) als auch ein Compiler (BASC) bereitgestellt. Beide Komponenten des Systems haben ihre Vor- und Nachteile. Während Interpreterprogramme relativ langsam arbeiten (jede Anweisung muß vor der Ausführung übersetzt werden) und viel Speicherraum belegen, lassen sich Programmänderungen sehr schnell im Dialog realisieren. Durch den Compiler übersetzte Programme laufen schneller ab und belegen weniger Speicher. Bei Programmänderungen muß allerdings jedesmal das Programm neu übersetzt und gelinkt werden.

Da beide BASIC-Komponenten bis auf geringfügige Abweichungen quellcode-kompatibel sind, können BASIC-Programme mit dem Interpreter im Dialog erarbeitet, ausgetestet und anschließend mit dem Compiler in ein laufzeitoptimales Programm übersetzt werden.

Folgende *Datenformate* werden in BASIC unterstützt:

Ganze Zahlen (INTEGER)

- 2 Byte Dualzahl
- Wertebereich: - 32768 ... + 32767
- Variablen werden durch den Typkennner "%" gekennzeichnet
- hexadezimalen Konstanten (0 ... FFFF) wird zu ihrer Kennzeichnung die Zeichenfolge &H vorangestellt (z. B. &H6A7C)
- oktalen Konstanten (0 ... 177777) wird die Zeichenfolge &O bzw. & vorangestellt (z. B. &O341)

Gleitkommazahlen (einfache Genauigkeit)

- 4 Byte
- Wertebereich: $\pm 10^{-38}$... $\pm 10^{+38}$ mit 7 Stellen Genauigkeit
- Typkennner: "!"

Gleitkommazahlen (doppelte Genauigkeit)

- 8 Byte
- Wertebereich: 10^{-38} ... 10^{+38} mit 16 Stellen Genauigkeit
- Typkennner: "*"

Zeichenketten

- maximal 255 Zeichen
- Konstanten werden in " " eingeschlossen (z. B. "OTTO")
- Typkennner: "\$"

Operationszeichen:

- **Arithmetische Operatoren:** +
-
*
/ (ganzahlige Division)
↑ (Potenzierung)
MOD (Modulo-Operation)
- **Vergleichsoperatoren:** > < >= <= <> =
- **Logische Operatoren:** NOT AND OR XOR IMP EQV

Der *Sprachumfang* von BASIC setzt sich zusammen aus Kommandos (Programmierdirektiven), Anweisungen und Standardfunktionen.

Kommandos unterstützen den Programmierprozeß. Mit ihrer Hilfe kann man Programme erfassen, modifizieren, auf Diskette ausgeben, von Diskette laden und starten.

Folgende Kommandos stehen u. a. zur Verfügung:

NEW	Löschen des gesamten Speichers
AUTO	Automatische Generierung von Zeilen-Nummern bei der Programmerrfassung
EDIT	Redigieren einer BASIC-Programmzeile mit den Unterfunktionen <ul style="list-style-type: none"> - Cursor-Steuerung - Einfügen von Zeichen - Löschen von Zeichen - Suchen von Zeichen - Austauschen von Zeichen - Beenden der Redigierung
DELETE	Löschen von Programmzeilen
RENUM	Umnummerierung von Programmzeilen
MERGE	Mischen von Programmen
LIST	Programmausgabe auf Bildschirm
LLIST	Programmausgabe auf Drucker
SAVE	Programm auf Diskette ausgeben
LOAD	Programm von Diskette laden
KILL	Programm oder Datei auf Diskette löschen
NAME	Programm- oder Dateinamen auf Diskette ändern
RUN	Programmstart
CLEAR	Löschen der Speichervariablen
CONT	Fortsetzen eines unterbrochenen Programms

TRON/ TROFF	Ein-/Ausschalten einer TRACING-Routine (Ausgabe der abgearbeiteten Programmzeilennummern) zur Unterstützung des Programmtests
----------------	---

Jedes BASIC-Programm besteht aus einer Folge von *Anweisungen*, wobei jede Anweisung durch eine maximal 5stellige Nummer gekennzeichnet ist. Die Ausführung der Anweisungen nach dem Kommando "RUN" erfolgt in aufsteigender Folge. Die Notation der Anweisungen kann in jeder beliebigen Folge vorgenommen werden.

Unter anderen stehen folgende Anweisungen zur Verfügung:

Vereinbarungsanweisungen

DEFINT, DEFSNG	Implizite Typvereinbarung für Variablenbezeichner
DEFDBL, DEFSTR	
DEF FN	Funktionsvereinbarung
DIM	Vereinbarung für ein- bzw. mehrdimensionale Felder unterschiedlicher Typen

Ein-/Ausgabenanweisungen

INPUT	Eingabe von Daten in Variablen
LINE INPUT	Eingabe beliebiger Zeichenfolgen
PRINT	Nichtformatierte Ausgabe von Informationen auf Bildschirm
LPRINT	Nichtformatierte Ausgabe auf Drucker
PRINT USING	Formatierte Ausgabe auf Bildschirm
LPRINT USING	Formatierte Ausgabe auf Drucker
WRITE	Ausgabe auf Bildschirm
WIDTH	Festlegung der Länge einer Ausgabezeile

Programmanweisungen

LET	Wertzuweisung
SWAP	Tauscht Werte zweier Variablen aus
RANDOMIZE	Einstellen Zufallsgenerator
READ	Lesen konstanter Daten aus der programminternen Datei
DATA	Angabe von Programm-Konstanten
GOTO	Unbedingter Sprung
GOSUB...RETURN	Unterprogramm-Sprung bzw. Rücksprung
ON...GOTO	Bedingte Programmverzweigung
ON...GOSUB	Bedingte Unterprogramm-Verzweigung

FOR...NEXT	Mehrfaches Abarbeiten einer Folge von Anweisungen
IF...THEN...ELSE	Bedingte Abarbeitung von Anweisungen
WHILE...WEND	Zyklusanweisung
CHAIN...MERGE	Programmüberlagerungen
ERROR/ON ERROR	Fehlerbehandlung
GOTO	

Zur Lösung mathematischer bzw. wissenschaftlich-technischer Probleme stehen eine Reihe von *Standardfunktionen* zur Verfügung:

- Arithmetische Funktionen (ABS, INT, SGN, FIX)
- Konvertierfunktionen (CINT, CSNG, CDBL, MKI\$, MKS\$, MKD\$, CVI, CVS, CVD)
- Exponentialfunktionen (EXP, LOG, SQR)
- Trigonometrische Funktionen (SIN, COS, TAN, ATN)
- Zeichenkettenfunktionen (ASC, CHR\$, HEX\$, OCT\$, STR\$, VAL, LEN, INSTR, LEFT\$, RIGHT\$, MID\$, MID\$, STRING\$, SPACES\$, INKEY\$, INPUT\$, LSET, RSET)
- Steuerfunktionen (RND, FRE, TAB, SPC, LPOS, POS, RND)

Auf Diskette können Datenbestände variabler Satzlänge in sequentiell oder direkt organisierten Dateien gespeichert werden.

In *sequentiellen Dateien* werden die Daten im ASCII-Format abgespeichert. Eine Dateieröffnung ist nur als Lese- oder Schreibdatei möglich. Eine bestehende sequentielle Datei kann nicht dynamisch erweitert werden.

Bei *Direktdateien* erfolgt ein Zugriff (Lese-/Schreibdatei) über eine relative Satznummer. Die Daten werden in gepackter Form abgespeichert.

Während sequentielle Dateien leichter zu erstellen sind, wird bei direkten Dateien weniger Speicherplatz benötigt und ein schnellerer Datenzugriff erreicht.

Folgende Anweisungen bzw. Funktionen stehen u. a. für die Dateiarbeit zur Verfügung:

OPEN	Dateieröffnung
INPUT #, LINE INPUT #	Eingabe von sequentiellen Dateien
PRINT #, PRINT# USING	Ausgabe in sequentielle Dateien
WRITE #	
FIELD	Strukturierung von Sätzen für Direktdateien
GET	Lesen eines Satzes einer Direktdatei
PUT	Schreiben eines Satzes einer Direktdatei
CLOSE	Schließen von Dateien
EOF	Funktion zur Abfrage des Dateiendes

LOF	Funktion zur Abfrage der Anzahl gelesener/ geschriebener Sätze
LOC	Funktion zur Abfrage der aktuellen Satz-Nr.

In der Sprache BASIC besteht die Möglichkeit, aus einem Interpreter- oder Compilerprogramm heraus Routinen (*Unterprogramme*) aufzurufen, welche in der Assemblersprache geschrieben wurden.

Wird ein derartiges Unterprogramm im Interpretermodus aufgerufen, dann muß dieses als .COM-Datei vorliegen und vor dem Laden des BASIC-Interpreters in den Speicher eingelesen werden.

Im Compilermodus wird der entsprechende .REL-File während des Bindeprozesses mit dem BASIC-Programm verbunden.

Folgende Anweisungen/Funktionen können zur Prozessorarbeit verwendet werden:

DEF USR	Spezifizierung der Start-Adresse einer USR-Funktion (Assembler-Routine)
USR	Aufruf eines Unterprogramms in Assemblersprache (Adresse ist mit DEF USR spezifiziert)
CALL	Aufruf eines Unterprogramms in Assemblersprache (Adresse wird mit CALL spezifiziert)
VARPTR	Übergabe der Speicheradresse einer Variablen
WAIT	Unterbrechung der Programmdurchführung
INP	Lesen eines Byte von einer spezifizierten Port-Adresse
OUT	Ausgabe eines Byte über eine spezifizierte Port-Adresse
PEEK	Lesen eines Byte vom Speicher (phys. Adresse)
POKE	Schreiben eines Byte in den Speicher (phys. Adresse)

Während der Übersetzung eines Programms und während der Laufzeit wird eine Reihe von *Fehlerkontrollen* durchgeführt.

Werden vom System Fehler erkannt, dann wird eine entsprechende Fehler-Nummer und eine verbale Beschreibung des Fehlers ausgegeben. Darüber hinaus hat der Programmierer mit nachfolgenden Ausdrucksmitteln die Möglichkeit, nutzereigene Fehlerbehandlungsroutinen zu programmieren, um Fehler ohne Programmunterbrechung schnell und effizient behandeln zu können:

ERROR	Aufruf von Standardfehlermitteilungen bzw. Fehlersimulation
ON ERROR	Verzweigung in Fehlerrountinen bei aufgetretenen Fehlern
GOTO	

- RESUME** Abschluß einer Fehlerbehandlungsroutine
ERL-Variable Enthält bei Fehlern die Zeilen-Nummer der Anweisung, die den Fehler verursacht hat
ERR-Variable Enthält die Fehler-Nummer

Der *BASIC-Compiler* bietet die Möglichkeit, Programme zu erstellen, die im Vergleich zum Interpreter schneller ablaufen, weniger Speicherplatz erfordern und eine Sicherung des Quellcodes gestatten. Der BASIC-Compiler übersetzt Quellprogramme (in einer Datei mit der Bezeichnung prog. BAS abgelegt) und erzeugt ein verschiebliches Objektprogramm (in einer Datei mit der Bezeichnung prog. REL abgelegt).

Diese verschieblichen Objektprogramme müssen vor der eigentlichen Laufzeit in einem gesonderten Programmverbinder-Prozeß mit dem Linkprogramm LINK zu einem lauffähigen Programm (in einer Datei mit der Bezeichnung prog. COM abgelegt) gebunden werden.

Während des Bindeprozesses werden Moduln aus einer Laufzeit-Bibliothek mit der Bezeichnung BASLIB.REL bzw. OBSLIB.REL und eventuelle Assembler-Moduln mit eingebunden. Dieser COM-File kann ohne BASIC-Interpreter/-Compiler abgearbeitet werden.

Zur *Arbeit mit BASIC* werden folgende Softwarekomponenten benötigt:

- | | |
|-------------------|---|
| BASI.COM | BASIC-Interpreter |
| BASC.COM | BASIC-Compiler |
| BRUN.COM | Laufzeit-Modul (wird während der Laufzeit des Programms benötigt und enthält die wichtigsten BASIC-Routinen) |
| BASLIB.REL | Laufzeit-Bibliothek (wird während des Bindeprozesses verwendet und enthält Routinen und Funktionen, die nicht in BRUN enthalten sind) |
| BCLOAD | Laufzeit-Ladeinformationsdatei |
| OBSLIB.REL | Laufzeit-Bibliothek (zur Arbeit ohne Laufzeit-Modul) |
| LINK.COM | Programm-Verbinder |

Als zusätzliche Programme können verwendet werden:

- | | |
|----------------|---|
| ASM.COM | Assembler für U880-Programme |
| LIB.COM | Bibliotheksverwalter zur Erstellung und Modifizierung von Anwender-Laufzeitbibliotheken |

Das *Haupt Einsatzgebiet* für BASIC ist die Programmierung von mathematischen und wissenschaftlich-technischen Problemen.

Neben einfachen ökonomischen Aufgaben können auch Probleme der Mas- sendatenverarbeitung (durch die leistungsfähige Dateiverarbeitung) und Textverarbeitungsprobleme (durch eine ausgebaute Zeichenkettenarbeit) gelöst werden.

5.2. Die Programmiersprache PASCAL-MP

Die Programmiersprache PASCAL wurde 1970 mit dem Ziel entwickelt, eine Sprache zur Verfügung zu stellen, welche ein einfaches und systematisches Programmieren ermöglicht und welche zuverlässig und effektiv zu implementieren ist. Seit dieser Zeit hat sich PASCAL sowohl im Schulbetrieb als auch im praktischen Einsatz für die unterschiedlichsten Anwendungsgebiete durchgesetzt.

PASCAL gehört zu den wenigen, weltweit intensiv genutzten Programmiersprachen, die aufgrund ihrer klaren, überschaubaren logischen Grundkonzeption ohne zusätzliche Spezialkenntnisse leicht erlernbar sind.

PASCAL ist eine formatfreie Sprache. Durch diese Eigenschaft wird es dem Programmierer möglich, das Programm in einer strukturierten, übersichtlichen und dadurch gut lesbaren Form zu schreiben und zu editieren.

Die Möglichkeit der modularen Programmierung in PASCAL gestattet es, zusammengehörige Gruppen von Programmschritten in Bausteinen zusammenzufassen und getrennt zu übersetzen. Diese Bausteine werden als Blöcke, Funktionen oder Prozeduren bezeichnet, je nachdem, wie sie im Programm eingesetzt werden.

Um Mehrdeutigkeiten auszuschließen und um die Übersetzung des Programms durch den Compiler zu vereinfachen, wurden alle Regeln der Sprache in einer streng definierten Syntax (beschrieben in der Backus-Naur-Form bzw. als Syntax-Diagramme) zusammengefaßt.

Das für den PC 1715 bereitstehende Programmiersystem PASCAL-MP realisiert das *Standard-PASCAL* (ISO) und stellt darüber hinaus zahlreiche *Erweiterungen* für die Anwendungs- und Systemprogrammierung zur Verfügung:

- Zerlegung eines Programms in Moduln und getrennte Übersetzung in verschiebbliche Objektmoduln;
- Verbindung mit verschiebblichen Assemblermoduln;
- Programmierung von Überlagerungsstrukturen (Overlays) in mehreren Überlagerungsgruppen bei kompletter Steuerung einer Programmwurzel;
- Zusätzliche Prozeduren und Funktionen:
 - MOVE-Prozeduren zum schnellen Datentransport
 - Bit- und Bytemanipulationen

- schnelle Daten-Ein-/Ausgabe
- Adress- und Längenfunktionen
- STRING-Manipulationen
- direkter zeichen- und blockweiser Dateizugriff
- Unterbrechungs- und externe Prozeduren
- Zusätzliche vordefinierte Standarddatentypen (BYTE, WORD, STRING);
- Realisierung von BCD- und Gleitkomma-Arithmetik;
- Rekursiver Aufruf von Prozeduren und Funktionen.

Für PASCAL-Programme existiert eine feste *Programmstruktur*. Alle Marken, Konstanten, Datentypen, Variablen, Prozeduren und Funktionen sind unmittelbar nach dem Programmkopf zu deklarieren bzw. zu definieren. Innerhalb einer Prozedur bzw. Funktion kann wiederum ein lokaler Deklarations- bzw. Definitionsteil programmiert werden.

Grundaufbau:

```
PROGRAM ...
LABEL ...      (Markendeklaration)
CONST ...     (Konstantendefinition)
TYPE ...      (Typ-Definitionen)
VAR ...       (Variablendeklaration)
PROCEDURE ... (Prozedur-Deklaration)
FUNCTION ...  (Funktions-Deklaration)
BEGIN
  <Anweisungsteil>
END.
```

Folgende Datenformate können in PASCAL-MP-Programmen verwendet werden:

- INTEGER:**
- ganze Zahlen
 - Länge: 2 Byte
 - Wertebereich: $- 32768 \dots + 32767$
 - hexadezimale Konstanten werden durch das "\$"-Zeichen gekennzeichnet (\$3A64)
- REAL:**
- Gleitkommazahlen
 - Länge: 4 Byte
 - Wertebereich: $\pm 10^{-17} \dots \pm 10^{+17}$
- REAL:**
- BCD-Zahlen mit 4 Dezimalstellen
 - Länge: 10 Byte
 - Wertebereich: $-9999999999999999.9999 \dots + 9999999999999999.9999$

- CHAR: – 1 alphanumerisches Zeichen
 – Länge: 1 Byte
 – Wertebereich: 0 ... 255
- STRING: – Zeichenkette
 – Länge: (1...255) + 1 Byte (Längenangabe)
- BYTE: – Länge: 1 Byte
 – Wertebereich: 0 ... 255
- WORD: – vorzeichenloses Maschinenwort
 – Länge: 2 Byte
 – Wertebereich: 0 ... 65535

Operationszeichen

NOT-Operatoren: NOT, ? (bei bitweiser Bearbeitung)

Arithmetische Operatoren: +
 –
 /
 *
 DIV MOD

Vergleichsoperatoren: = <> < > <= >= IN

Logische Operatoren: NOT (bei bitweiser Bearbeitung ?)
 AND (bei bitweiser Bearbeitung &)
 OR (bei bitweiser Bearbeitung !)

*Alle Marken, Konstanten, Datentypen und Variablen sowie alle Prozeduren und Funktionen, welche in einem PASCAL-Programm benötigt werden, müssen am Anfang des Programms bzw. am Anfang der Prozedur/Funktion **deklariert** bzw. **definiert** werden.*

Folgende Ausdrucksmittel stehen in PASCAL-MP zur Verfügung:

- LABEL Deklaration von Marken
- CONST Definition von Konstanten
- TYPE Definition von Datentypen
- Aufzählungstypen
 - Teilbereichstypen
 - Feldtypen (ARRAY... OF)
 - Recordtypen (RECORD)
 - Filetypen (FILE OF...)
 - Mengentypen (SET OF...)
 - Zeigertypen

VAR	Deklaration von Variablen (einschließlich EXTERNAL-Variable und ABSOLUTE-Variable) Möglichkeiten des Variablenzugriffs durch: <ul style="list-style-type: none"> - Variablenbezeichner - Index-Variable - Recordkomponenten-Variable - Filepuffer-Variable - Dynamische Variable
PROCEDURE	Deklaration von Prozeduren (einschließlich EXTERNAL-Prozeduren und INTERRUPT-Prozeduren) Parameter-Übergabe in Form von: <ul style="list-style-type: none"> - Werteparameter (call by value) - Variablenparameter (call by reference) - Prozedurparameter - Funktionsparameter
FUNCTION	Deklaration nutzeigener Funktionen (einschließlich EXTERNAL-Funktionen)

Algorithmische Aktionen in PASCAL-Programmen werden durch *Anweisungen* beschrieben. In PASCAL-MP stehen folgende Anweisungen zur Verfügung:

GOTO	unbedingte Verzweigung
:=	Ergibt-Anweisung (Zuweisung)
prozedurname (...)	Prozedur-Anweisung
funktionsname (...)	Funktions-Anweisung
BEGIN... END	Verbund-Anweisung
IF... THEN... ELSE	bedingte Anweisung
CASE... OF... ELSE	bedingte Anweisung für Mehrfachentscheidungen
WHILE... DO	Anweisung zur Programmierung von Bedingungs-schleifen
REPEAT... UNTIL	Anweisung zur Programmierung von Bedingungs-schleifen
FOR... TO(DOWNTO)... DO	Anweisung zur Programmierung von Zählschleifen
WITH	Anweisung zum verkürzten Schreiben von Verbundbezeichnern

In PASCAL-MP werden die nachfolgenden *Standardprozeduren/Standardfunktionen* bereitgestellt:

Ein-/Ausgabeprozeduren/-funktionen

ASSIGN	Zuweisung eines externen Dateinamens an eine Filevariable
OPEN, RESET	Eröffnen einer Lese-Datei
REWRITE	Eröffnen einer Schreib-Datei
READ, READLN	Dateneingabe über Tastatur bzw. Lesen aus einer sequentiellen Datei
WRITE, WRITELN	Datenausgabe auf Bildschirm/Drucker bzw. Schreiben in eine sequentielle Datei
GET	Lesen des nächsten Satzes einer sequentiellen Datei
PUT	Schreiben eines Satzes in eine sequentielle Datei
IORESULT	Die Funktion liefert den Status der letzten E/A-Operation
EOLN	Abfrage des Zeilenendes einer Text-Datei
EOF	Abfrage des Dateiendes einer sequentiellen Datei
SEEKREAD	Lesen eines Satzes einer Direktdatei
SEEKWRITE	Schreiben eines Satzes einer Direktdatei
BLOCKREAD	blockweise Eingabe einer Direktdatei
BLOCKWRITE	blockweise Ausgabe einer Direktdatei
GNB	bytewise Dateieingabe
WNB	bytewise Dateiausgabe
READHEX	Lesen hexadezimaler Daten aus einer Datei
WRITEHEX	Schreiben hexadezimaler Daten in eine Datei
CLOSE, CLOSEDEL	Schließen einer Datei
PURGE	Löschen einer Datei
PAGE	Löschen Bildschirm/Seitenvorschub Drucker

Zeichenketten-Routinen

MOVE, MOVELEFT, MOVERIGHT	Transport-Prozeduren
COPY, CONCAT, INSERT	Teilstring, Zusammenfügung, Einfügung
DELETE	Löschen von Zeichen in Zeichenketten
LENGTH, POS	Längen- und Suchfunktion

Bit- und Byte-Routinen

TSTBIT	Testen von Bitpositionen
SETBIT, CLRBIT	Setzen/Löschen von Bitpositionen
SHR, SHL	Rechts-/Linksverschiebung (bitweise)
HI, LO	Übertragung HIGH-/LOW-Teil
SWAP	Vertauschen HIGH-/LOW-Teil

Arithmetische Funktionen

ABS	Absolutwert
TRUNC, ROUND	ganzzahliger Teil bzw. Rundung
SQR, SQRT	Potenzieren, Quadrieren
SIN, COS, ARCTAN	trigonometrische Funktionen
LN, EXP	natürlicher Logarithmus, Exponentialfunktion

Übertragungsfunktionen

ODD	Kontrolle geradzahlgiger INTEGER-Werte
ORD	Bilden des Ordinalwertes
CHR	Konvertieren INTEGER in CHAR
WRD	Liefert WORD-Typ
PRED, SUCC	Liefert Vorgänger bzw. Nachfolger

Sonstige Funktionen

ADDR, SIZEOF	Liefert die Adresse bzw. Länge einer Variablen
@CMD	Zugriff auf Kommandozeile SCP
@BDOS	Ruf einer BDOS-Funktion
EXIT	Vorzeitiges Beenden von Prozeduren bzw. Funktionen

Die *Dateiarbeit* in PASCAL-MP bezieht sich auf Datenbestände, welche aus logisch gegliederten, gleich großen Elementen (Sätzen) bestehen und auf Disketten geführt werden.

Der Zugriff über Zeiger erfolgt wahlweise direkt oder sequentiell. Grundsätzlich wird unterschieden zwischen temporären und externen Dateien.

Die Standard-Dateien INPUT (Tastatur-Eingabe) und OUTPUT (Bildschirm-Ausgabe) sind vordefiniert.

Jeder Datei wird ein Typ zugeordnet, welcher die Größe und das Format der individuellen Dateielemente spezifiziert.

Maximal können 16 Dateien gleichzeitig eröffnet werden, wobei für jede Datei ein Fileinformationsblock angelegt wird, welcher die Dateiparameter und den E/A-Puffer enthält.

Die Prozeduren und Funktionen zur Dateiarbeit sind in „Standardprozeduren/-funktionen“ zusammengestellt.

Zur Erweiterung der Programmiermöglichkeiten und zur Programmierung komplexer Lösungen können in PASCAL-MP nachfolgende *Programmiertechnologien* genutzt werden:

- Modulprogrammierung/Modulcompilierung:
Dieses Verfahren bietet die Möglichkeit, Programme in einzelne Moduln zu zerlegen, die getrennt übersetzt und während des Bindeprozesses zusammengelinkt werden. Dadurch besteht die Möglichkeit der Erstellung nutzeigener Modul-Bibliotheken.
- Verbinden mit Assembler-Programmen:
Programmrountinen, welche mit dem Assembler „ASM“ übersetzt wurden, können durch den PASCAL-Linker mit PASCAL-Programmen bzw. -Moduln zu einem lauffähigen Programm verbunden werden.
- Programmüberlagerungen:
Zur Programmierung komplexer Programmstrukturen können Programm-Moduln geschrieben werden, welche bei Bedarf während der Programmaufzeit in maximal 3 verschiedene Overlaygebiete von Diskette nachgeladen werden können. Insgesamt können für ein Programm maximal 50 Overlaymoduln bereitgestellt werden. Die Steuerung der Überlagerungen erfolgt von der Programmwurzel.
- Programmverkettung:
Durch die Programmverkettungs-Routine CHAIN ist es möglich, daß aus einem PASCAL-Programm heraus ein weiteres Programm aufgerufen, von Diskette geladen und gestartet werden kann.
- INLINE- und Miniassembler:
Mit diesem Programmierhilfsmittel besteht die Möglichkeit, konstante Daten in eine Prozedur/Funktion einzufügen. Dabei können auch Assemblerbefehle verwendet werden, um kleine CPU-Routinen direkt in PASCAL-Programme einzufügen.

Das Programmiersystem PASCAL-MP läuft unter Steuerung des Betriebssystems SCP und besteht aus folgenden Programmkomponenten:

PASCMP.COM	PASCAL-Compiler mit den Overlays PASCMP.000... PASCMP.006
LINKMP.COM	PASCAL-Linker zum Binden von PASCAL-Programmen, PASCAL-Moduln, Assembler-Moduln und Routinen der Bibliotheksprogramme:

	- REALBCD.ERL (BCD-Arithmetik)
	- REALFP.ERL (Gleitkomma-Arithmetik)
	- TRANC.ERL (Transzendente Funktionen)
	- RANDOM.ERL (Direkter Dateizugriff)
	- MPMON.ERL (Debugger-Routinen)
	- HEAPFULL.ERL (Dynamische Listen)
	- MPLIB.ERL (E/A- und arithmetische Routinen)
	- OVERLMGR.ERL (Overlay-Routinen)
	- SORT.ERL (Sortier-Routinen)
	- INI00.ERL (Initialisier-Routinen)
	- INIFF.ERL (Initialisier-Routinen)
DISA880.COM	Disassembler zur Untersuchung des erzeugten Compiler-Codes
DISZ80.COM	
LIBMP.COM	Bibliotheksverwalter zur Verkettung von .ERL-Dateien (es entstehen durchsuchbare Bibliotheken) und zur Umwandlung von .ERL-Dateien in .REL-Dateien.
PB.COM	Formatierungsprogramm für PASCAL-Quelltext
XREF.COM	Programm zur Ausgabe der PASCAL-Cross-Referenzen

Durch vorhandene INTERRUPT-Routinen, Datenein- und -ausgabe über physische E/A-Tore u. a. lassen sich Probleme der Echtzeitsteuerung lösen. Besonders durch die Erweiterungen in PASCAL-MP gegenüber dem Standard-PASCAL (Bit-/ Byteroutinen, INLINE-Assembler usw.) wird PASCAL in zunehmendem Maße als Sprache für Systementwicklungen genutzt. Neben der Sprache PASCAL-MP werden noch folgende PASCAL-Implementationen für den PC 1715 (SCP) bereitgestellt:

- PPS 880/S
(kompatible Sprachversion zu Turbo-PASCAL)
- CAS-PASCAL
(quellcode-kompatible Sprachversion zum kommerziellen PASCAL 1520).

5.3. Die Programmiersprache FORTRAN

FORTRAN (*FOR*mula *TRAN*slator) ist eine Sprache, welche nach strengen syntaktischen Regeln aufgebaut ist und vom Programmierer fordert, sein zu lösendes Problem in einer Folge von exakten Anweisungen zu formulieren. Der Quelltext eines FORTRAN-Programms wird mit einem geeigneten

Editor (empfohlen wird der Textprozessor TP) erfaßt und editiert, mit dem FORTRAN-Compiler (FOR) in einen verschieblichen Maschinencode übersetzt und mit dem Linkprogramm (LINK) zu einem lauffähigen Programm gebunden.

Zum Programmiersystem FORTRAN wird eine umfangreiche Bibliothek (FORLIB) mit einer Reihe effizienter Unterroutinen zur Verfügung gestellt. Die vorliegende FORTRAN-Implementation entspricht dem FORTRAN IV und damit prinzipiell der ANSI-66-FORTRAN-Norm (Ausnahme: Datentyp COMPLEX) und bietet darüber hinaus verschiedene Erweiterungen zur Erhöhung der Leistungsfähigkeit.

Für alle FORTRAN-Programme ist eine feste *Programmstruktur* vorgesehen:

```

Programmart <Name>
Typvereinbarung
EXTERNAL-Anweisungen
DIMENSION-Anweisungen
COMMON-Anweisungen
EQUIVALENCE-Anweisungen
DATA-Anweisungen
Anweisungsfunktions-Definitionen
:
Anweisungen
:
END.
```

Dabei können folgende *Programmarten* verwendet werden:

```

PROGRAM
SUBROUTINE
FUNCTION
BLOCK DATA.
```

In FORTRAN werden *Daten* in der Form von Konstanten, Variablen, Feldern und Feldelementen verwendet.

Folgende Datentypen stehen zur Verfügung:

INTEGER (Ganze Zahlen)	– 2-Byte-Dualzahl Wertebereich: – 32768 ... + 32767
REAL (Gleitkommazahlen)	– 4 Byte Wertebereich: $\pm 10^{-17}$... $\pm 10^{+17}$ mit 7 signifikanten Ziffern

DOUBLE PRECISION (Gleitkommazahlen)	- 8 Byte Wertebereich: $\pm 10^{-17} \dots \pm 10^{+17}$ mit 16 signifikanten Ziffern
LOGICAL HOLLERITH	Logischer Wert TRUE oder FALSE (1 Byte) Zeichenkette (1 Byte/Zeichen; maximale Länge 255 Zeichen)

Daneben sind folgende Äquivalenzen bei Typbezeichnungen möglich, wobei n die Anzahl der Byte angibt, die der Datentyp belegt (z.B. INTEGER * 1). Felder in FORTRAN stellen eine geordnete Menge von Daten dar und können in den Dimensionen 1, 2 oder 3 definiert werden.

Folgende Operatoren sind vorgesehen:

a) *Arithmetische Operatoren*

- + (Addition)
- (Subtraktion)
- * (Multiplikation)
- ** (Potenzierung)

b) *Logische Operatoren*

- .NOT. (Negation)
- .AND. (logisches UND)
- .OR. (logisches ODER)
- .XOR. (logisches exklusives ODER)

c) *Vergleichsoperatoren*

- .LT. (kleiner als)
- .LE. (kleiner oder gleich)
- .EQ. (gleich)
- .NE. (nicht gleich)
- .GT. (größer als)
- .GE. (größer oder gleich)

Die Anweisungen in der Sprache FORTRAN lassen sich in nichtausführbare und in ausführbare Anweisungen einteilen.

Während die nichtausführbaren Anweisungen (*Vereinbarungsanweisungen*) unter anderem die Art und Anordnung von Daten und die Ein- und Ausgabeformate beschreiben, realisieren die *ausführbaren Anweisungen* den konkreten Programmalgorithmus.

Alle *Vereinbarungsanweisungen* werden am Anfang einer Programmeinheit angeordnet.

Es wird unterschieden zwischen der arithmetischen (IF(A)m1, m2, m3) und der logischen IF-Anweisung (IF (u) S).

- 4) DO-Anweisung:
Ermöglicht das wiederholte Abarbeiten einer Anweisungsfolge (Schleifenanweisung).
- 5) STOP-Anweisung:
Kennzeichnung des logischen Programm-Endes.
- 6) PAUSE-Anweisung:
Unterbrechung der Programm-Abarbeitung.
- 7) END-Anweisung:
Kennzeichnung des physischen Programm-Endes.
- 8) CALL-Anweisung:
Aufruf von Unterprogrammen mit oder ohne Parameterübergabe.
- 9) RETURN-Anweisung:
Rücksprung aus Unterprogrammen.

In FORTRAN wird eine Reihe von Anweisungen bereitgestellt, welche die Steuerung und die Bedingungen der *Datenübertragung* definieren. Die Datenübertragung erfolgt zwischen dem internen Speicher und den externen Geräten (Tastatur, Drucker, Bildschirm, Floppy-Disk).

Folgende Gruppen stehen zur Verfügung:

- 1) Formatgebundene READ-Anweisung
- 2) Formatgebundene WRITE-Anweisung
- 3) Unformatierte READ-Anweisung
- 4) Unformatierte WRITE-Anweisung
- 5) FORMAT-Anweisung.

Formatgebundene READ-/WRITE-Anweisungen realisieren einen Datentransport zwischen Speicher und externen Geräten, wobei die ein-/ausgehenden Daten mit der FORMAT-Anweisung zu beschreiben sind.

SCP-FORTRAN ermöglicht einen sequentiellen bzw. direkten Zugriff auf Disketten-Dateien.

Folgende Anweisungen können verwendet werden:

- | | |
|--|--|
| 1) READ-Anweisung/
WRITE-Anweisung: | Lesen/Schreiben eines Satzes einer sequentiellen Datei. |
| 2) REC-Anweisung: | Für den direkten Zugriff kann die Satz-Nummer durch die REC-Anweisung (REC=nnnnn) in der READ-/WRITE-Anweisung angegeben werden. |

- 3) OPEN-Subroutine: Eröffnen einer Datei (CALL OPEN).
- 4) REWIND-Anweisung: Schließen einer Datei, erneutes Öffnen der Datei und Positionieren auf den Dateianfang.
- 5) ENDFILE-Anweisung: Schließen einer Datei.

Bei Direktdateien erfolgt die Ein-/Ausgabe stets mit einer Satzlänge von 128 Byte. Bei sequentiellen Dateien werden maximal 127 Byte verarbeitet. Fehler und EOF-Unterbrechungen können bei READ/WRITE durch ERR=- und END=-Zusätze behandelt werden.

ENCODE-Anweisung/ Datentransport zwischen verschiedenen Speicherbereichen unter Kontrolle der FORMAT-Anweisung mit gleichzeitiger Konvertierung des angegebenen Formates in das ASCII-Format (ENCODE) bzw. vom ASCII-Format in das gegebene Format (DECODE).

In FORTRAN können durch den Programmierer *Unterprogramme* definiert werden, welche nur einmal beschrieben und beliebig oft aufgerufen werden können. Diese Unterprogramme können sein:

- Anweisungsfunktionen,
- FUNCTION-Unterprogramme,
- SUBROUTINE-Unterprogramme,
- Bibliotheksfunktionen.

Anweisungsfunktionen werden durch einen arithmetischen oder logischen Ausdruck innerhalb einer Programmeinheit definiert und können nur hier verwendet werden.

FUNCTION- und SUBROUTINE-Unterprogramme sind in sich abgeschlossene Programmeinheiten, die getrennt übersetzt werden können.

Bibliotheksfunktionen sind arithmetische oder andere Funktionen, die in das System eingebaut sind und bei Bedarf eingebunden werden.

Unter anderen stehen folgende Funktionen zur Verfügung:

a) In-line-Standardfunktionen:

ABS	MAX	IFIX	DBLE
INT	MIN	SIGN	SNGL
MOD	FLOAT	DIJ	

b) Out-of-line-Funktionen:

EXP	COS	SQRT	ATAN2
DMOD	TANH	ALOG	
SIN	ATAN	ALOG10	

- c) Sonstige Unterprogramme:
PEEK POKE INP OUT

Zur Zuweisung von Anfangswerten an Variablen und Felder in COMMON-Bereichen können BLOCK-DATA-Unterprogramme aufgerufen werden.

Zum *Programmiersystem* gehören folgende Bestandteile:

FOR.COM = FORTRAN-Compiler
LINK.COM = Programmverbinder
FORLIB.REL = Bibliotheksprogramme.

Zur Editierung der Quellprogramme wird zusätzlich ein entsprechender SCP-Editor (z. B. TP) benötigt.

FORTRAN ist eine höhere, international weit verbreitete Programmiersprache, welche auf Grund der universellen und problemorientierten Eigenschaften auf vielen Anlagen implementiert wurde.

Die *Haupteinsatzgebiete* der Sprache liegen in der rechen-technischen Lösung von Ingenieuraufgaben auf mathematischem und wissenschaftlich-technischem Gebiet.

6. Standardsoftware im Betriebssystem SCP

Die in diesem Kapitel vorzustellenden Programmsysteme sind branchenneutrale Anwenderprogramme. Sie zeichnen sich durch ein breites Nutzungsspektrum aus. Das ihrer Wirkungsweise zugrunde liegende Generatorprinzip ermöglicht es, daß die zu lösenden Aufgaben in einer der Ausdrucksweise des Nutzers weitgehend anpaßbaren Sprache für das jeweilige Programmsystem formuliert werden können. Das bietet zum einen den bildungsökonomischen Vorteil, mit einem Minimum an Lernaufwand ein Maximum an Leistungszuwachs zu erreichen, und zum anderen den Nutzungsvorteil, daß jedermann seine Aufgaben selbständig, ohne aufwendige Hilfeleistungen von Spezialisten, am Personalcomputer bearbeiten kann. Praktisch bedingen sich der Personalcomputer und die Standardanwendersoftware gegenseitig und bilden deshalb einen entscheidenden Faktor bei der praktischen Umsetzung der Schlüsseltechnologien. Dem interessierten Nutzer wird empfohlen, bei der Konzipierung seiner Aufgabenlösungen immer erst zu prüfen, inwieweit die zu bearbeitende Aufgabe mit den in diesem Kapitel beschriebenen und weiteren ähnlichen Programmsystemen bereits gelöst werden kann.

Die bisherige Praxis hat gezeigt, daß beispielsweise bei ökonomischen Aufgabenstellungen etwa 90 bis 95 % aller praktisch relevanten Aufgaben mit Standardsoftware gelöst werden können. Nur für die restlichen Aufgaben sind dann noch in einer höheren Programmiersprache, wie sie im 5. Kapitel beschrieben wurden, entsprechende Softwarelösungen zu gestalten. In anderen Nutzungsbereichen liegen die Prozentwerte niedriger, aber in *jedem* Nutzungsbereich ist eine große Zahl von Aufgaben unter Einsatz der hier zu beschreibenden Systeme lösbar.

Prinzipiell lassen sich alle Aufgabenklassen so in Teilaufgabenklassen zerlegen, daß folgende typischen Teilaufgaben auftreten:

- a) Erfassung, Verwaltung, Ergänzung und Aktualisierung von Daten und Datenbeständen,
- b) Recherche und Selektion von Daten aus einem bestehenden Datenbestand (Erstellung eines temporären Teildatenbestandes),

- c) Verknüpfung von Daten (eines temporären Teildatenbestandes), d. h. Berechnungen, Auswertungen, Analysen usw.,
- d) Darstellung der Arbeitsergebnisse des Aufgabenlösungsprozesses in verschiedenen Formen (Listen, Tabellen, Berichte, graphische Darstellungen usw.).

Die konkreten Erscheinungsformen der Teilaufgaben werden von Aufgabenklasse zu Aufgabenklasse sehr differenziert und unterschiedlich umfangreich ausgeprägt sein. Von diesen Eigenschaften einer Aufgabenstellung gehen die standardisierten Anwenderprogramme aus und bieten Unterstützung bei ihrer Lösung. Eine Übersicht gibt Tabelle 6.1.

Daraus können die folgenden methodischen Orientierungen für die Einsatzvorbereitung eines Personalcomputers abgeleitet werden.

1. Jede Softwarelösung für PC 1715 kann vom REDABAS-System ausgehen, indem die Teilaufgaben des Einrichtens von Datenbanken (Datenbasen), der Erfassung, der Auswahl, der Änderung und Aktualisierung von Daten einer vorgegebenen Datenbasis sowie der Archivierung von Ergebnissen von Datenverknüpfungen in einer vorgebbaren Datenbasis in der REDABAS-Programmiersprache gestaltet werden.
2. Die Recherche und Selektion von Daten aus einer vorgebbaren Datenbasis wird zweckmäßig ebenfalls in REDABAS formuliert, da diese Aufgaben zum direkten Leistungsumfang des REDABAS-Systems gehören. Dabei können sowohl einzelne Datensätze, die angebbaren Bedingungen genügen, als auch einzelne Satzelemente eines Datensatzes ausgewählt werden.
3. Die arithmetische und logische Verknüpfung von Daten kann je nach Art, Kompliziertheit und Komplexität der zu beherrschenden Operationen in REDABAS selbst (z. B. bei der Berechnung von Elementen eines Datensatzes aus anderen Elementen des Datensatzes einer vorgebbaren Datenbasis), im Kalkulationsprogramm (z. B. bei normalen Kennziffernberechnungen) oder in einem Teilprogramm, das in einer der im Betriebssystem SCP verfügbaren Programmiersprachen gestaltet wurde, abgefaßt werden.
4. Die Darstellung der Ergebnisse einer Datenverknüpfung und -analyse wird durch verschiedene Komponenten der Standardsoftware unterstützt. Mit dem REPORT-Generator bietet das REDABAS-System eine Unterstützung an, die für eine Reihe von Aufgaben mit geringeren Ansprüchen gut genutzt werden kann. Die für die praktische Interpretation von Ergebnissen häufig benötigte Flexibilität in den Darstellungsformen läßt sich allerdings mit dem REPORT-Generator nur schwer oder gar nicht erreichen. Hier bietet sich für eine flexible Darstellung von Tabellen, Listen, Berichten usw. der Textprozessor mit seinen vielfältigen Möglichkeiten

des Mischens von Dateien und der gezielten Darstellung durch die optimale Ausschöpfung aller Möglichkeiten der Druckgeräte an. Außerdem bietet der Textprozessor die Möglichkeit, manuelle Eingriffe in den Darstellungsprozeß zu organisieren, die innerhalb einer komplexeren oder automatisierten Lösung nur schwer oder sehr aufwendig programmierbar

Tabelle 6.1

Typische Teilaufgabenklassen einer Aufgabenklasse und ihre Realisierbarkeit durch Basissoftware (Standardanwendersoftware) im Rahmen des Betriebssystems SCP für den PC 1715

Grund- aufgabe	Standardsoftware			
	REDABAS	Kalkulations- programm	Textprozessor	PCGRAPH
a)	Hauptaufgabe, sehr gute Unterstützung in allen Teilprozessen	Erfassung gut, Änderung gut, Verwaltung gebunden an die spezielle Aufgabe (Tableau)	Erfassung vielseitig, Änderung und Verwaltung dem Nutzer überlassen	Erfassung möglich
b)	Hauptaufgabe, sehr gute Unterstützung bei Recherche und Selektion	bedingte Unterstützung bei Selektion, bei Recherche keine Unterstützung	keine Unterstützung	möglich
c)	Bei Inhalt von Dateien sehr gut, sonst schwach entwickelt	sehr gute Unterstützung, operative Methoden	keine Unterstützung	bedingt möglich
d)	über REPORT-Generator für Übersichten gut geeignet, nicht alle Ansprüche können befriedigt werden	Für einfache Tabellen und Listen gut geeignet, relativ starr bezüglich der Darstellung	Vielfältige Darstellung von Tabellen, Berichten und Listen programmierbar, manuelle Eingriffe bei operativer Anforderung möglich	Vielfältige graphische Darstellungen möglich, relativ starr in der Darstellung

sind. Graphische Ausgaben können in vielfältiger Form mit dem PCGRAPH-System realisiert werden.

Es kann festgestellt werden, daß eine umfassende, den Qualitätsanforderungen genügende Lösung einer komplexen Aufgabenstellung für einen PC 1715 dann gestaltet werden kann, wenn die im Betriebssystem SCP zur Verfügung stehenden Standardanwenderprogramme in ihrer Nutzungsvielfalt und ihren charakteristischen leistungsbezogenen Besonderheiten voll ausgeschöpft werden. Das führt zu einer zweiten Generation von Softwarelösungen, die besonders durch folgende Merkmale charakterisiert ist:

- Einsatz der verschiedenen Programmsysteme entsprechend den Anforderungen der zu lösenden Aufgabenklasse,
- höherer Komplexitätsgrad der praktischen Lösung,
- Bereitstellung von Kopplungsinformationen zur Verbindung unterschiedlicher Programmsysteme (Aufbau einer Kommandodatei innerhalb eines Programmsystems für ein weiteres System),
- automatisierter Übergang zwischen den Systemen unter Ausnutzung der Möglichkeiten des Betriebssystems SCP und des REDABAS-Datenbanksystems (Aufbau von SUBMIT- und XSUB-Dateien),
- Einrichten von Schnittstellen für off-line-Kopplung über Daten- und Dateistrukturen, um gegebenenfalls später weitere Prozesse integrieren zu können (offene Lösung im Gegensatz zu einer in sich geschlossenen Lösung entsprechend den Möglichkeiten der ersten Generation von Softwarelösungen),
- Menüsteuerung der gesamten Lösung vom REDABAS-System aus, also Rückkehr nach jedem Teilprozeß zum Hauptmenü der Softwarelösung (Entscheidungstabellennormalformulardarstellung der Lösung auf der Grundlage der Architekturtheorie von Softwarelösungen).

Das dargestellte Softwarelösungskonzept wurde bereits erfolgreich auf dem PC 1715 in Grundkonfiguration mit zwei 1.2-Minidiskettenlaufwerken realisiert. Erweiterte Gestaltungs- und Nutzungsmöglichkeiten bieten sich mit den 1.6-Laufwerken. Bei Nutzung der 1.2-Laufwerke muß die strukturelle Gliederung der Datenbasen sorgfältig durchdacht und geplant werden, um den Wechsel von Disketten im Aufgabenlösungsprozeß auf ein Minimum zu begrenzen.

Die vorliegenden Einführungen in Standardanwendersysteme geben einen Überblick über die Einsatz- und Nutzungsmöglichkeiten. Detaildarstellungen zum Aufbau und zur Arbeitsweise dieser Programmsysteme werden in gesonderten Broschüren zum PC 1715 angeboten (vgl. die Literaturhinweise am Ende des Bandes).

Die nachfolgend beschriebenen Systeme sind Programme, die einen Dialog

mit dem Nutzer führen. Dazu erscheint nach dem Programmstart ein sogenanntes Anfangsmenü (auch Grundmenü genannt) auf dem Bildschirm. (Im weiteren Verlauf der Programmabarbeitung können weitere (Teil-)Menüs auftreten.)

Ein Menü bedeutet in diesem Zusammenhang ein Angebot an vom Programmsystem lösbaren Aufgaben, aus denen der Nutzer eine Aufgabe auswählen kann. Dieser Prozeß des Anbietens und Auswählens wird als Dialog bezeichnet. Der Computer bietet dabei also eine Reihe von Funktionen, die in den Menüs angezeigt werden, und der Nutzer wird darauf antworten, indem er eine der Funktionen über die Tastatur auslöst. Hat er das getan, so führt der Computer diese Funktion aus, oder er braucht noch Informationen und fordert durch Bildschirmausschriften den Nutzer zu weiteren Eingaben (z. B. Dateiname) auf. Es kommt also zu einem Dialog zwischen Computer und Nutzer (Bediener); der Bediener muß dabei immer darauf achten, wo sich der Cursor befindet. Dieser steht im allgemeinen immer am Ende einer Bildschirmausschrift, die der Bediener lesen soll und auf die er eine Antwort geben muß.

Für Anfänger ist es oft schwierig zu unterscheiden, ob eine Eingabe mit der Abschlußtaste (ET) beendet werden muß oder nicht. Es gilt folgende Regel: Menü-Funktionen verlangen nur eine Tastencodierung. Sie werden nicht durch die (ET)-Taste abgeschlossen. Dagegen muß die Eingabe von Dateinamen oder ähnlichen Angaben, die eine variable Länge haben, stets durch die (ET)-Taste abgeschlossen werden.

In vielen Bildschirmmeldungen kommen die Begriffe „RETURN“ und „ESCAPE“ vor. Damit sind nur einzelne Tasten gemeint, nämlich die Abschlußtaste (ET) und die (ESC)-Taste. (Bei manchen Tastaturlösungen hat die (ET)-Taste auch die Beschriftung „RETURN“ oder „ENTER“ oder „ET1“.)

6.1. Das relationale Datenbanksystem REDABAS

Der stetige Zuwachs an Informationen verlangt rationelle Verfahrensweisen zu deren Verarbeitung. Für die eingesetzten 8-Bit-Mikrorechner hat sich die Arbeit mit Datenbanken bewährt. Wo zuvor verschiedene Programme zur Speicherung, Verwaltung und Auswertung von verschiedenen Datenbeständen nötig waren, kann nun die gesamte Informationsmenge zu einem Datenbestand zusammengefaßt werden. Es ist dann nur dieser eine Datenbestand zu verwalten, und er kann zu verschiedenen Zwecken ausgewertet werden.

Das relationale Datenbanksystem REDABAS eignet sich für eine derartige Verwaltung kleiner und mittlerer Datenmengen. Ein besonderer Vorteil besteht darin, daß mit REDABAS sowohl den Anwendern ohne Spezialkennt-

nisse in der Datenverarbeitung als auch versierten Programmierern ein leistungsstarkes Hilfsmittel zur Verfügung steht.

Die Übereinstimmung von Daten ist sehr schwierig zu gewährleisten, wenn dieselben Daten in verschiedenen Datenbeständen mehrfach vorhanden sind. Durch die Konzentration der vorhandenen Daten zu einer gemeinsamen Datenbasis, der Datenbank, wird dieser Nachteil beseitigt. Sowohl der Zugriff zu den Daten als auch deren Bearbeitung erfolgen durch das Datenbanksystem selbst und nicht durch individuelle Anwenderprogramme. Alle Anwenderprogramme können aber mit allen in der Datenbank vorhandenen Daten arbeiten. Für eine neue Auswertung der Daten muß lediglich ein neues Auswerteprogramm geschaffen werden; die Datenbank bleibt unverändert. Das Datenbanksystem kümmert sich in einem solchen Fall selbst um die neue Interpretation der Daten. Werden zu einem Datensatz neue Datenfelder hinzugefügt, dann haben diese Erweiterungen bzw. Veränderungen keinen Einfluß auf existierende Auswerteprogramme, die diese neuen Daten nicht verarbeiten.

Das relationale Datenbanksystem REDABAS zeichnet sich durch eine einfache Darstellungsweise der Daten aus. Mit zweidimensionalen Tabellen können die Beziehungen zwischen einzelnen Datensätzen veranschaulicht werden. Dabei wird jede Zeile der Tabelle als Datensatz und jede Spalte in der Zeile als Datenfeld bezeichnet. Jede Eintragung innerhalb der Tabelle besteht aus einzelnen Werten. Alle Eintragungen in einer Spalte der Tabelle müssen vom gleichen Typ sein. Überträgt man nun diese Darstellungsweise der Daten auf die mathematische Theorie der Relationen, dann entspricht eine solche Tabelle einer Relation, d. h. einer Menge gleichartiger Datensätze. Dabei stellen die Zeilen der Tabelle die Elemente der Relation, die Tupel, und die Spalten die Merkmalswerte der Relation dar. Mit der zur Verfügung stehenden Datenbanksprache, die wiederum auf der Relationenalgebra basiert, können die Relationen mathematisch bearbeitet werden. Für den Sprachgebrauch in der Datenverarbeitung stellt eine Relation eine Datei dar. Für die Arbeit mit den Datenbanken gibt es einerseits Elemente der Datenbanksprache, die sich auf die Verwaltung und Auswertung der Daten beziehen, und andererseits gibt es Elemente der Datenbanksprache, mit denen komplette Verwaltungs- und Auswertungsprogramme geschrieben werden können. Zur Einarbeitung in REDABAS wird man sich zunächst auf die Befehle zur Verwaltung und Auswertung von Datenbanken beschränken. Erst danach wird man die Erstellung komplexer Anwenderprogramme in Angriff nehmen. Durch ein Anwenderprogramm läßt sich die Arbeit mit Datenbanken auf einen menügesteuerten Dialog anheben. Die Datenmanipulationen selbst können auch Bestandteil von Anwenderprogrammen sein.

Zur Installation von REDABAS auf dem PC 1715 steht das Programm RDBASINS.COM zur Verfügung. Dabei wird die Programmdatei REDABAS.COM variiert. Mit diesem Installationsprogramm können die Anpassung an den Bildschirm und die Veränderung spezieller Parameter erfolgen. Die menügesteuerte Programmführung erleichtert die Handhabung. Zur Anpassung von REDABAS an den Bildschirm kann man aus einer angegebenen Liste den geeigneten Typ auswählen oder sämtliche Steuercodes neu definieren.

Die Veränderung von Parametern bezieht sich auf

- die Definition des als Makroanzeiger verwendeten Zeichens (Standard ist „&“),
- das Ein- bzw. Ausschalten des Fehlerkorrekturdialogs (Standard ist „ausgeschaltet“),
- die Auswahl des Datumformates (Standard ist TT.MM.JJ).

Entspricht die ausgelieferte Version des REDABAS mit einer Bildschirmgröße von 24 Zeilen zu je 80 Zeichen und den Standardwerten der oben angegebenen Parameter den geforderten Anwendungsbedingungen, dann ist eine Neuinstallation nicht notwendig.

Das *Einrichten einer Datenbank* setzt den Entwurf der Datenbank voraus (d.h. die Struktur der Datenbank muß festgelegt werden). Sie hängt natürlich im wesentlichen von den gewünschten Auswertungen ab. Mit der Datenbankstruktur wird der Rahmen festgelegt, in den die Daten aufgenommen werden. Übertragen auf die Darstellungsweise der Daten in Tabellenform werden mit der Struktur die Spaltenbezeichnungen der Tabelle festgelegt. Eine Änderung der Spaltenbezeichnungen der Datenbank ist vor dem Erfassen der Daten noch leicht möglich. Wird eine Veränderung der Datenbankstruktur nach Erfassen der Daten notwendig, dann ist der bisherige Datenbestand zunächst zu kopieren, sodann ist die Strukturänderung vorzunehmen, und die ausgelagerten Daten sind schließlich in die neue Struktur zurückzukopieren.

Mit dem Befehl

```
CREATE <dateiname>
```

beginnt der Dialog zur Definition der Struktur einer Datenbank, welche schließlich unter <dateiname> auf der Diskette abgelegt wird. Vom Anwender ist dann zu entscheiden, ob sofort Daten erfaßt werden sollen oder erst zu einem späteren Zeitpunkt.

Die *Datenbanksprache* bietet eine Reihe leistungsstarker Befehle, mit denen auf einfache Weise Daten einer Datenbank hinzugefügt, gelöscht, geändert, angezeigt oder ausgedruckt werden können.

Befehle wie

APPEND	(Zufügen von Datensätzen)
EDIT	(Ändern von Datensätzen)
DELETE	(Löschen von Datensätzen)
PACK	(Verdichten der Datenbank)
COPY	(Kopieren einer Datenbank)
BROWSE	(Anzeigen/Ändern von Daten)
REPLACE	(Ersetzen Feldinhalt)
CHANGE	(Inhalt einzelner Datenfelder ändern)

ermöglichen ein Aktualisieren der Datenbank. In sehr vielen Befehlen kann durch das Hinzufügen einer Bedingung die Manipulation auf bestimmte Datensätze eingeschränkt werden. Die Angabe von Bedingungen ermöglicht nicht nur die gezielte Veränderung, sondern auch eine gezielte Auswertung bestimmter Datensätze.

Die Ausgabe von Daten auf den Drucker kann einmal durch ein Zuschalten des Druckers zur Bildschirmausgabe (mit SET PRINT ON oder CTRL-P) oder aber über den integrierten Listengenerator REPORT erfolgen. Bei der ersten Möglichkeit werden alle Bildschirmausgaben auf dem Drucker mitprotokolliert.

Die zweite Möglichkeit ist die elegantere. Sie bietet die Festlegung eines Druckformates mit Angaben über das verwendete Papierformat, das Definieren von Seiten- und Spaltenüberschriften sowie der Berechnung von Spaltenwerten beim Drucken. Ferner können über numerische Felder Zwischensummen und Endsummen in der Druckliste gebildet werden. Dieses im Dialog mit dem Datenbanksystem erstellte Druckformat wird auf Diskette abgespeichert und steht jederzeit zum Aufruf bereit. Eine Änderung von Feldinhalten oder das Hinzufügen von Datensätzen zur Datenbank ziehen keine Änderung dieses Druckformates nach sich.

Der Aufruf eines definierten Druckformates kann erweitert werden durch die Angabe einer Bedingung. Es werden dann nur noch die Datensätze im definierten Druckformat ausgegeben, die dieser Bedingung genügen. Für das Suchen eines bestimmten Datensatzes in der Datenbank setzen bestimmte Befehle (z. B. LOCATE und FIND) eine Sortierung oder Indizierung der Datenbank voraus. Damit verkürzt sich auch die Zeit, die zum Auffinden der gewünschten Daten notwendig ist.

Das Sortieren der Datenbank bewirkt eine Umordnung der Datensätze in eine neue Datenbank. Um aber mehrere Sortierordnungen gleichzeitig führen zu können, wird eine Datenbank indiziert. Die Indizierung bewirkt das Anlegen einer Hilfsdatei (Indexdatei), die nur ein Datenfeld pro Datensatz

beinhaltet (das sogenannte Schlüsselfeld). Zu jedem Schlüsselfeld wird ein Zeiger angelegt, der auf den zugehörigen Datensatz in der Datenbank verweist. Demzufolge kann eine solche Indexdatei nur in Verbindung mit der zugehörigen Datenbank genutzt werden. Wird nun ein bestimmter Wert des Schlüsselfeldes gesucht, dann wird diese Suche in der Indexdatei vorgenommen und über den Zeiger zur Datenbank der vollständige Datensatz verfügbar gemacht. Zu einer Datenbank können beliebig viele solcher Indexdateien gehören.

Das Datenbanksystem REDABAS ermöglicht die gleichzeitige Arbeit mit zwei Datenbanken. Damit kann sich ein Auswertungsprogramm auch auf die Informationen aus zwei miteinander in Verbindung stehenden Datenbanken beziehen. Ein weiterer Aspekt der Arbeit mit zwei Datenbanken ist das Aktualisieren einer Datenbank unter Zuhilfenahme einer zweiten. Möglich ist das sowohl mit dem Befehl APPEND als auch mit dem Befehl UPDATE. Die Vereinigung zweier Datenbanken oder Teilen davon zu einer dritten ist mit dem Befehl JOIN möglich. Bei diesem Befehl kommt der Auswahl der Datenfelder und der Angabe einer exakten Bedingung besondere Bedeutung zu, um ein Überschreiten der Diskettenkapazität durch die neue Datenbank zu vermeiden.

Ein wichtiges Hilfsmittel zum Erlernen der Datenbanksprache ist die HELP-Einrichtung. Mit dem Befehl

HELP <parameter>

können Informationen über Befehle, Funktionen, Installation, Leistungsparameter, verwendete Dateiformate, Steuertasten und verwendete Begriffe abgerufen werden. Die Angabe <parameter> bewirkt den Abruf der speziellen Information. Wird nur der Befehl HELP eingegeben, erhält man zunächst eine Liste der möglichen Parameterangaben und kann sich den gewünschten Parameter auswählen.

Zur *Programmierung* stellt die Datenbanksprache REDABAS zusätzlich zu den Befehlen für die Verwaltung und Auswertung von Datenbanken eine ganze Reihe weiterer leistungsfähiger Funktionen und Befehle bereit.

Die *Funktionen* ermöglichen unter anderem den Zugriff auf die aktuelle Satznummer der Datenbank, das Bestimmen von Teilzeichenketten, die Umwandlung numerischer Werte in alphanumerische und umgekehrt sowie die Längenbestimmung einer Zeichenkette.

Mit dem Befehl

MODIFY COMMAND <programmdateiname>

wird die Eingabe und die Speicherung einer Befehlsfolge unter dem durch

<programmdateiname> angegebenen Namen gestartet. Ist eine solche Programmdatei einmal erfaßt und auf der Diskette abgespeichert, kann deren Abarbeitung jederzeit mit dem Befehl

DO <programmdateiname>

veranlaßt werden.

Der erste Befehl eines Programms sollte

ERASE

sein. Damit wird der Bildschirm gelöscht und der Cursor in die linke obere Ecke des Bildschirms positioniert. Den Abschluß eines Programms bildet der Befehl

RETURN

Er beendet ein Programm und veranlaßt die Rückkehr in die Ebene, aus der das Programm gerufen wurde.

Eine exakte Positionierung von Daten auf dem Bildschirm oder Drucker wird mit dem Befehl

@ zeile, spalte

erreicht. Dieser Befehl kann um die Ausgabe von Daten oder Text erweitert und somit für eine *formatierte Ausgabe* genutzt werden. Mit ihm kann durch eine weitere Ergänzung auch eine *formatierte Dateneingabe* realisiert werden. Programmeingaben sind ferner über die Befehle

ACCEPT (Eingabe beliebiger alphanumerischer Zeichen),

INPUT (Eingabe numerischer, alphanumerischer und logischer Werte) und

WAIT (Eingabe eines einzelnen Zeichens)

möglich.

Programmverzweigungen und *Programmschleifen* können mittels Konstruktionen wie

DO CASE

ENDCASE

sowie

IF

ELSE

ENDIF

und

DO WHILE bedingung
ENDDO

ausgeführt werden.

Für die Programmierung sehr wichtig ist die Benutzung der *Speichervariablen*. REDABAS erlaubt die Arbeit mit bis zu 64 Speichervariablen, deren Name, Typ und Länge durch eine Wertzuweisung festgelegt werden. Die Speichervariablen dürfen eine Gesamtlänge von 254 Zeichen nicht überschreiten. Eine Übersicht über die vorhandenen Speichervariablen, deren Typ und die Gesamtlänge wird mit dem Befehl

DISPLAY MEMORY

geliefert. Der Anwender kann über die von ihm festgelegten Namen auf diese Speichervariablen zugreifen.

Von Bedeutung für die Programmierung ist die Möglichkeit, diese Speichervariablen in eine spezielle Datei zu sichern (Befehl SAVE) und sie zu einem späteren Zeitpunkt wieder laden (Befehl RESTORE) und verarbeiten zu können.

Da diese Variablen nach ihrer Definition bis zum Verlassen von REDABAS (mittels Befehl QUIT) oder ihrer ausdrücklichen Löschung (mit RELEASE oder CLEAR) Gültigkeit besitzen, können alle Programme während der Arbeit mit REDABAS darauf zugreifen. Damit kann also auch eine *Parameterübergabe* zwischen Programmen realisiert werden.

6.2. Das Kalkulationsprogramm KP (SCP)

Das Kalkulationsprogramm ist ein leistungsfähiges System von Programmteilen, das ein Arbeitstableau realisiert. Dieses erlaubt,

- Berechnungsvorschriften zu speichern,
- Texte und Überschriften zu ergänzen, ohne Berechnungen zu beeinflussen,
- Reports zu entwickeln und diese in andere einzufügen,
- Kalkulationsrechnungen schnell zu erledigen.

Das Tableau besteht aus einer zweidimensionalen Matrix mit 63 Spalten und 254 Zeilen. Damit stehen dem Nutzer 16 002 Elemente (auch Felder genannt) zur Verfügung, in die er Daten, Texte oder Formeln eingeben und/oder verändern kann. Maximal können etwa 22 000 Zeichen (Byte) in ein Tableau eingetragen werden.

Das Kalkulationsprogramm ersetzt Arbeitsbogen, Taschenrechner und Bleistift. Es ist selbst erläuternd, da man durch Drücken der Fragezeilentaste die erforderliche Unterstützung (Menü-Bedienerführung) durch das Programm erhält.

Der Cursor kann auf jedes Element des Arbeitstableaus bewegt werden. Einzelne oder mehrere Elemente können über entsprechende Befehle gegen Verfälschung geschützt werden.

Das Kalkulationsprogramm umfaßt eine Reihe arithmetischer und trigonometrischer Funktionen, die an beliebiger Stelle des Arbeitstableaus verwendet werden können. Die Editiermöglichkeiten sind umfangreich und haben eine große Leistungsfähigkeit. Wenn z. B. das Arbeitsblatt geändert werden soll, können Spalten und/oder Zeilen verschoben, eingefügt und auch gelöscht werden. Die im Kalkulationsprogramm enthaltenen Berechnungsvorschriften werden hierbei automatisch angepaßt.

Wenn eine Aufgabe gelöst und in einem Arbeitstableau programmiert ist, können die Daten schnell verändert und die darin enthaltenen Berechnungen sofort wieder durchgeführt werden. Für die Anzeige lassen sich die Ergebnisse wahlfrei formatieren.

Das einmal festgelegte Arbeitstableau kann auf Diskette abgelegt und jederzeit wieder geladen, editiert und gedruckt werden. Das Kalkulationsprogramm ist branchenneutral unter anderem einsetzbar zur Bilanzierung, Buchhaltung, Marktanalyse, Investitionskontrolle, Gehaltsabrechnung, Kostenrechnung, Kontenbewegung, Marktplanung und -prognose.

Das Kalkulationsprogramm setzt sich aus folgenden Bestandteilen zusammen.

KP.COM beinhaltet das Kalkulationsprogramm.

KP.OVL beinhaltet die Überlagerungsdatei für das Kalkulationsprogramm, die für einige Kommandos und Funktionen gebraucht wird.

KP.HLP beinhaltet die Datei, die dem Anwender zusätzliche Hinweise zu Funktionen und Kommandos anzeigt.

Diese drei Dateien müssen grundsätzlich auf der Diskette sein, die sich im aktuellen Laufwerk befindet. Es ist zu empfehlen, daß sich auf dieser Diskette auch das Betriebssystem befindet. Das Kalkulationsprogramm arbeitet unter Steuerung des Betriebssystems SCP. Die Bedienhinweise (Menü) erfolgen in deutscher Sprache.

Durch Eingabe von KP (ET) wird das Kalkulationsprogramm geladen; es erscheint auf dem Bildschirm das Anfangsmenü. Wird ein Fragezeichen eingegeben, dann wird das Hilfsmenü angezeigt (Abbildung 6.1). Nach dem Betätigen von (ET) wird das Grundmenü (Abbildung 6.2) angezeigt.

In der Abbildung 6.1 stehen am linken Rand die Zeilennummern (1...20...254); in der obersten Zeile stehen die Spaltennummern (A, B, C, ..., H, ..., Z, AA, AB, ... AZ, BA, BB, ... BK.)

Die Spalten sind durch :| :| begrenzt.

() verkörpert den Cursor, der das aktive Element im Arbeitsbereich kennzeichnet.

Kalkprog. Spezielle Tasten und ihre Funktionen

- / → Eingabe Kommando
- = → Sprung zum Element xx
- ! → Neuberechnung starten
- ; → Wechsel BS-Fenster

Möglichkeiten der Kursorbewegung im Arbeitsbereich sind:

CTRL-E	CTRL-X	CTRL-S	CTRL-D
HOCH	TIEF	LINKS	RECHTS

- CTRL-C → löscht akt. Eingabezeile
- " → Kennz. Textelement
- / → Kennz. wiederh. Textelement

andere Zeichen kennzeichnen Formelemente

(ET) f. Weiterarbeit,

Abbildung 6.1
Hilfsmenü von KP 1715
(SCP)

: A :: B :: C :: D :: E :: F :: G :: H

1 : ()

2 :

3 :

4 :

5 :

16 :

17 :

18 :

19 :

20 :

) A1

Breite x

1)_

Speich: xx

Letzte Sp/Zeil A1

? → HILF

Abbildung 6.2
Das Tableau
in seiner
Grunddarstellung

Die letzten Zeilen in der Abbildung 6.2 haben folgende Bedeutung:

Statuszeile

Anzeigezeile

Eingabezeile

Die *Arbeitsweisen* des Kalkulationsprogramms gliedern sich in die drei Arbeitsmodi:

- *Tableaumodus*,
- *Dateneingabemodus*,
- *Kommandomodus*.

Zu einem bestimmten Zeitpunkt kann nur in einem dieser Modi gearbeitet werden. Die Auswahl des Arbeitsmodus erfolgt in der Eingabezeile mit der Eingabe des ersten Zeichens. Der Grundmodus, zu dem stets zurückgekehrt wird, ist der *Tableaumodus*.

Im *Tableaumodus* ist der Cursor im Tableau (1. Cursor) aktiv; zugleich ist der Cursor in der Eingabezeile (2. Cursor) inaktiv. Der (1.) Cursor (Tableaueursor) läßt sich über das gesamte Tableau bewegen. Auf diese Weise können Elementinhalte oder -werte angesehen werden.

In der Statuszeile wird angezeigt

- der Status des aktiven Elementes,
- der globale Status,
- die Position des Tableaueursors,
- die Richtung der Bewegung des Tableaueursors.

Der *Dateneingabemodus* gestattet, Daten direkt in die Eingabezeile zu bringen.

Der *Dateneingabemodus* wird gestartet, indem ein Text oder eine Formel eingegeben wird. Die Texteingabe wird mit einem doppelten Anführungszeichen begonnen. Alle Zeichen, die druckbar sind, können über die Tastatur eingegeben werden. Die Textlänge kann maximal 115 Zeichen betragen.

Wird eine Formel eingegeben, so wird sie formal auf Richtigkeit geprüft.

Eine Fehlermeldung erfolgt bei falschen Eingaben. Die Eröffnung kann mit einer Ziffer, einem Buchstaben oder einem Sonderzeichen, z. B. „“, -, +, oder (erfolgen. Beendet wird die Eingabe mit (ET); die Daten werden dann aus der Eingabezeile in das aktive (Tableau-)Element transferiert.

Formeln können aus folgenden Bestandteilen zusammengesetzt sein:

- numerische Konstanten, Vorzeichen, Klammern, Dezimalpunkt
- Elementadressen
- arithmetische Operatoren
- mathematische Funktionen (z. B. Mittelwert, Absolutwert, Logarithmus, Exponentialfunktionen usw.)
- spezifische Funktionen (z. B. die logischen Funktionen IF, AND, OR und NOT).

Die Formeln können eine Länge bis zu 116 Zeichen besitzen. Eine Formel wird in das Feld eingetragen, in das der Funktionswert kommen soll.

Felder, in die Formeln eingetragen wurden, können nicht gleichzeitig als Eingabefelder für Daten verwendet werden.

Numerische Konstanten können aus bis zu 16 Ziffern, einem Dezimalpunkt, einem Vorzeichen sowie einem Exponenten bestehen. Wenn die Eingabe größer ist als 16 Stellen, wird diese rechts abgeschnitten und gerundet.

In der Eingabezeile sind bei Dateneingaben eine Reihe von Cursorfunktionen möglich:

- EDIT-Kursorbewegung
- Löschen von Zeichen
- Einfügen von Leerzeichen
- Heranführen der Adresse an Cursorposition
- Transport des Inhalts der Eingabezeile in das aktive Element
- Löschung der Eingabezeile und Rücksprung in den Tableaumodus.

Der *Kommandomodus* wird durch 5 Schlüsselzeichen aus dem Tableaumodus ausgelöst:

- ! Neuberechnung starten
- , Wechseln des Bildschirmfensters
- = Sprung zum Element $\times \times$
- / Steuerkommando
- ? Hilfestellung.

Bei der Eingabe eines "/" werden gleichzeitig drei Vorgänge abgearbeitet:

- Über die Statuszeile wird der Kommandomodus, als Cursorposition eine 1, danach der "/" angezeigt.
- Die Anzeigezeile gibt alle möglichen Anfangsbuchstaben der Kommandos sowie das "?" als Bedienerunterstützung an.
- Der Cursor in der Eingabezeile wird aktiviert.

Wird ein Steuerkommando eingegeben, genügt der erste Buchstabe des Befehls; der Rest des Wortes wird automatisch in der Eingabezeile ergänzt, auch alle Kommas und Abtrennzeichen.

Folgende Kommandos stehen dem Nutzer zur Verfügung:

- | | |
|---------------|---|
| /A(usführung) | Ausführungen einer Kommandodatei von Diskette |
| /B(lank) | Löscht den Inhalt von Elementen |
| /D(atens.) | (Datenschutz) Verhinderung des Überschreibens eines Elementes, einer Zeile oder einer Spalte oder eines Bereiches |
| /E(dit) | Editieren des Inhalts eines Elements |

/ F(ormat)	Einstellen des Anzeigeformats einer Zeile oder einer Spalte oder eines Bereiches
/ G(lobal)	Änderung der globalen Anzeige oder von Berechnungsmöglichkeiten
/ I(ni)	Löschen des Tableaus und Einstellung der Grunddarstellung
/ K(opi)	Kopieren des Inhalts von Elementen
/ L(oe)	Löschen von Zeilen oder Spalten
/ M(ehr ZS)	Einfügen von Zeilen oder Spalten
/ O(hne DS)	Aufheben des Schutzes von Elementinhalten
/P(rot.)	Ausgabe des Bildschirminhalts oder der Werte der Elemente auf Drucker, Bildschirm oder Diskette
/ R(ett)	Abspeichern des Arbeitsbereiches auf Diskette
/ S(ys)	Beenden des Kalkulationsprogramms
/ T(itel)	Ständiges Anzeigen der Titelzeile
/ U(eb.)	Übernehmen (Laden) der CAL-Datei in den Arbeitsbereich
/ V(ers.)	Verschieben von Zeilen oder Spalten
/ W(iederh.K)	Mehrfachkopie von Zeilen oder Spalten
/ Z(weit)	Teilen/Aufhebung: Teilung der Anzeige in zwei Fenster bzw. Aufhebung dieser Teilung

Im Kalkulationsprogramm gibt es eine Reihe von *Fehlermeldungen*. Für jeden Fehler existiert eine Kurzbeschreibung des Fehlerursprungs und ein Programm zu dessen Beseitigung. Folgende Fehler werden angezeigt:

- Kopier-Definitions-Fehler
- Laufwerk nicht bereit
- Formel-Fehler
- Datei nicht ladbar
- Bereichsfehler
- Fenster-Parameter-Fehler
- Arbeitsbereich voll
- Spaltenfehler
- Zeilenfehler
- Diskette voll
- Datei nicht vorhanden
- Speicher voll
- Geschützte Eingabe.

Das Kalkulationsprogramm bietet die Möglichkeit einer einfachen *grafischen Ausdrucksweise*. Zur Darstellung wird ein Stern (*) für das grafische Format

eingegeben. Die Anwendung der erforderlichen Kommandos führt auf dem Bildschirm zu einer Balkengrafik. Hierbei wird eine 1:1-Grafik dargestellt. Ein wichtiges Kommando stellt die Titelanzeige dar. Sie gestattet eine ständige Anzeige von festgelegten Zeilen und/oder Spalten des Arbeitstableaus; die restlichen Zeilen und/oder Spalten können gerollt werden. So können Spaltenüberschriften oder Zeilenbezeichnungen fixiert werden. Bei Zweiteilung der Anzeige kann jeder Teil selbständige Anzeigeformen haben (Formeln, Grenzen). Soll das Arbeitstableau zur späteren Weiterverarbeitung gerettet werden, wird das Kommando /Rett angewandt. Das Kommando veranlaßt eine Kopie des gesamten Tableaus und speichert es auf der Diskette des zugewiesenen Laufwerks.

Es gibt drei Möglichkeiten zum Kopieren von Dateien in das Tableau:

- Übernehmen einer Datei von der Diskette in das Arbeitstableau;
- vollständiges Kopieren eines einzelnen Elements einer Spalte, einer Zeile oder eines Bereiches in einen Zielbereich des Arbeitstableaus;
- Kopieren der Wertbelegung eines einzelnen Elementes, einer Zeile, einer Spalte oder eines Bereiches in einen Zielbereich des Arbeitstableaus.

Beim Protokollieren wird eine Kopie des Arbeitsbereiches oder eines Teiles davon erzeugt. Drei Protokollierungen sind möglich: auf Papier, auf Bildschirm und als Datei mit dem Dateityp .PRN (Druckdatei) auf Diskette.

Eine Kommandodatei beinhaltet die Zeichen, die bei manueller Erstellung des Kalkulationsprogramms nacheinander über die Tastatur eingetastet werden, um gezielt Kommandos zu aktivieren. Das Kommando „Ausführen“ realisiert das Laden und Ausführen von Kommandodateien.

Nach dem Laden des Kalkulationsprogramms beginnt automatisch die Abarbeitung der Kommandodatei.

Das Beenden der Arbeit mit dem Kalkulationsprogramm erfolgt durch das Kommando /SYS (Eingabe von S reicht aus). Danach wird vom Programm gefragt:

Beenden KP ? J (a) / N (ein) .

Bei Eingabe von J wird die Arbeit mit dem Kalkulationsprogramm beendet, und das Betriebssystem SCP erwartet eine neue Aufgabe.

6.3. Der Textprozessor TP

Die Notwendigkeit, Aussagen zu fixieren und dauerhaft darzustellen, gibt es schon seit mehreren tausend Jahren. Die Entwicklung reicht von Felsmalereien, Keilschriften, über das Schreiben mit der Feder, Buchdruckkunst, Er-

findung der Schreibmaschine und Setzmaschine bis in die heutige Zeit mit Computern und deren großen Speichermöglichkeiten für alle Arten von Daten.

Für die allgemeine praktische Büroarbeit gab es bis vor einigen Jahren prinzipiell nur eine Möglichkeit, amtliche Texte herzustellen, nämlich das Schreiben auf einer Schreibmaschine. Tippfehler mußten geschickt auf dem Papier korrigiert oder das ganze Blatt mußte neu geschrieben werden. Die Anwendung von Kohlepapier machte es möglich, etwa 5 Durchschläge gleichzeitig zu erstellen. Reichte die Anzahl der Exemplare nicht aus bzw. war die Qualität der Durchschläge nicht akzeptabel für den jeweiligen Zweck, so war es meistens notwendig, den gleichen Text mehrfach einzutippen. Vervielfältigungsmöglichkeiten wie Lichtpause, Xerographie, Thermokopie o. ä. brachten in jedem Fall Qualitätsverluste oder waren teuer bzw. aufwendig oder standen gar nicht zur Verfügung.

Einen gewissen Vorteil brachte die Anwendung von elektronischen Schreibmaschinen. Hiermit können verschiedene Schriftarten und Schreibdichten gewählt werden, so daß der optische Eindruck eines Schriftstückes aufgewertet werden kann. Der Hauptvorteil elektronischer Schreibmaschinen liegt jedoch in der Fähigkeit, einen Text speichern und diesen beliebig oft ausdrucken zu können. Solch eine Schreibmaschine besitzt meist einen internen Speicher, der aber nur einen bestimmten Textumfang aufnehmen kann (z. B. 4000 Zeichen, das ist etwa eine A4-Seite).

Das Speichervermögen einer Diskette des PC 1715 liegt zwischen etwa 150 000 und 700 000 Zeichen; die Disketten können zudem gewechselt werden. Ein weiterer wesentlicher Unterschied zu Schreibmaschinen besteht darin, daß beim Bearbeiten (Editieren) der Text nur auf dem Bildschirm erscheint und nicht auf dem Papier. Alle Eingaben und Korrekturen wie Streichen, Einfügen und Überschreiben oder Neugestaltung der Zeilenaufteilung, Einrücken von Absätzen o. ä. werden direkt auf dem Bildschirm im Dialog zwischen dem Computer und dem Nutzer ausgeführt. Der fertige Text wird automatisch auf Diskette dauerhaft gespeichert und kann zu einem beliebigen Zeitpunkt beliebig oft ausgedruckt werden.

Der Textprozessor TP (auch als Textprogramm bezeichnet) arbeitet im Rahmen des Betriebssystems SCP und ist ein bildschirmorientiertes Textbearbeitungssystem. Umfangreiche Menüs und Hilfstexte zeigen dem Bediener alle TP-Funktionen an, so daß er, wenn er die prinzipielle Arbeitsweise erkannt hat, alle TP-Möglichkeiten vom Bildschirm ablesen kann.

Nach dem Aufruf des Textprozessors erscheint ein Kennungstext (Versions-Nr. u. a.). Dieser kann mit der ESC-Taste quittiert werden.

Danach wird das Anfangsmenü auf dem Bildschirm sichtbar.

Alle Funktionen des Anfangsmenüs werden durch Drücken einer einzelnen Buchstabentaste ausgelöst. Dabei ist es gleichgültig, ob ein Groß- oder Kleinbuchstabe eingegeben wird. Im folgenden sind die Funktionen des Anfangsmenüs zusammengestellt:

D, N	Textbearbeitung
P, M	Druck
O, Y, E	Datei kopieren, löschen, umbenennen
L, F	Wahl des Laufwerkes, Anzeige aller Dateinamen
H	Umfang der Menüunterstützung einstellen
R, X	Start eines anderen Programmes, Ausgang aus TP.

Die mit der Funktion H wählbaren verschiedenen Hilfggrade zur Unterstützung des Nutzers gelten nur während einer Textbearbeitung, also innerhalb der Funktionen D oder N. Die maximale Menüunterstützung ist bei Stufe 3 gegeben. Sie wird dem ungeübten Nutzer empfohlen. Für die praktische Bearbeitung von normalen Texten hat sich Hilfggrad 2 als sinnvoll erwiesen. Das Editiermenü ist hierbei abgeschaltet, so daß fast der gesamte Bildschirm zur Textbearbeitung frei ist; die Gruppenmenüs können aber noch angezeigt werden. Hilfggrad 0 wird vom professionellen Anwender bevorzugt.

Üblicherweise befinden sich auf Laufwerk A die TP-Programmdateien und auf einem anderen Laufwerk die zu bearbeitenden Textdateien. Mit der Funktion L sollte dieses Laufwerk selektiert werden, damit das Verzeichnis der Textdiskette angezeigt wird. Außerdem kann dann bei der Textdateibezeichnung die Laufwerksangabe entfallen.

Um einen gewöhnlichen Text (z. B. Brief, Bericht usw.) zu bearbeiten, drückt man im Anfangsmenü die Taste D. Danach erscheint die Aufforderung zur Eingabe der Dateibezeichnung (Laufwerk: Dateiname. Typkennung). Nach dem Drücken der Taste <ET> erscheint die Aufschrift „Neue Datei“, wenn die Datei noch nicht existiert. Danach wird der Bildschirm zur Textbearbeitung aufbereitet. Die erste Bildschirmzeile zeigt verschiedene Zustände an, z. B. den aktuellen Befehl, die Dateibezeichnungen der Textdatei, die Seitennummer, die Text-Zeilenummer dieser Seite, die Spaltennummer, auf der sich der Cursor befindet, und ob das Einfügeregime eingeschaltet ist. (Im Gegensatz zum Überschreiben wird in diesem Regime der neue Text vor der aktuellen Cursorposition eingefügt, indem der schon vorhandene Text einfach nach hinten geschoben wird.)

In der ersten Zeile erscheint auch die Ausschrift „WARTE“, solange der Textprozessor keine (schnellen) Tastatureingaben annehmen kann (Diskettenzugriff).

Bei Hilfggrad 3 zeigen die folgenden Bildschirmzeilen das Edit-Menü an.

Dann kommt das sogenannte Tabulator-Lineal mit linkem (L) und rechtem (R) Rand.

Der Cursor steht am Anfang des Textbereiches. Dieser kann bereits gefüllt sein (wenn keine neue Datei angelegt wird), andernfalls kann beliebiger Text geschrieben werden, der auch auf dem Bildschirm erscheint. Im Textprozessor ist standardmäßig Wortumbruch eingestellt, d. h. das Programm schiebt ganze Wörter, die nicht mehr auf die Zeile passen, automatisch auf den Anfang der nächsten Zeile. Deshalb sollte man nicht alle Zeilen mit der (ET)-Taste abschließen, sondern nur Einzelzeilen (z.B. Überschriften und Leerzeilen); am Absatzende muß selbstverständlich immer die (ET)-Taste benutzt werden.

Für Textkorrekturen, Cursor- und Bildbewegungen, Suchfunktionen u. a. steht eine Vielzahl von Kommandos zur Verfügung. Diese Editbefehle müssen sich jedoch von normalen Textzeichen unterscheiden. Dafür gibt es die Umschalttaste CTRL (control = steuern). Diese ist in gleicher Weise zu handhaben wie eine normale Umschalttaste für Klein-/Großbuchstaben. Für CTRL wird allgemein und auch in diesem Buch die Kurzschreibweise $\hat{\text{G}}$ genutzt, also CTRL-G = $\hat{\text{G}}$.

Die meisten CTRL-Codes wirken sofort nach Tastenbetätigung. Es gibt jedoch fünf CTRL-Codes ($\hat{\text{Q}}$, $\hat{\text{J}}$, $\hat{\text{K}}$, $\hat{\text{O}}$, $\hat{\text{P}}$), die allein noch nicht wirksam sind. Sie beinhalten jeder eine ganze Gruppe von Befehlen (Gruppenmenü). Erst durch Drücken einer weiteren Taste ist der Befehl vollständig und wird ausgeführt.

Das Cursor- und Funktionstastenfeld der Tastatur wird bei der TP-Version V1/3 dazu genutzt, häufig gebrauchte Editierbefehle leicht auslösen zu können. So kann „Cursor eine Zeile tiefer“ sowohl mit $\hat{\text{X}}$ ausgeführt werden, aber auch mit der Kursortaste, die durch den nach unten zeigenden Pfeil gekennzeichnet ist.

Hier sei bemerkt, daß sich der Cursor nur innerhalb eines bereits geschriebenen Textes positionieren läßt.

Die Vielfalt der Editierbefehle kann man nach folgenden Funktionsgruppen unterteilen.

Funktionsgruppen

Kursorpositionierung

Bildbewegung

Texteingabe-Kommandos

Löschkommandos

Beispiele

$\hat{\text{A}}$ ein Wort nach links

$\hat{\text{C}}$ ein Bild weiter

$\hat{\text{V}}$ Einfügen ein/aus

(DEL) ein Zeichen links vom Cursor löschen

Tabulatorkommandos	^I	Tabulatorsprung
Formatierungskommandos	^OR	rechten Rand einstellen
Suchen und Ersetzen	^QF	Zeichenkette suchen
Beenden und Sichern	^KS	Speichern und Weiterarbeit
Marken und Blockkommandos	^KV	Block verschieben
Dateikommandos	^KR	Datei einfügen
Druck-Steuerkommandos	^PD	Doppeldruck ein/aus
sonstige Kommandos	^J	Hilfsmenü aufrufen

Das ^J-Gruppenmenü (Hilfsmenü) beinhaltet viele Erläuterungen zu den TP-Funktionen, die der Bediener während einer Textbearbeitung abrufen kann. Dadurch und in Verbindung mit den anderen Menüs erklärt sich der Textprozessor selbst.

Ist der Text fertig bearbeitet, so muß das dem Textprozessor mitgeteilt werden. Innerhalb des ^K-Gruppenmenüs drückt man Taste D zum Beenden der Textbearbeitung. Diese Editfunktion (^KD) schreibt den Text vollständig auf die Diskette und kehrt zum Anfangsmenü zurück.

Ist das Textprogramm passend zum Drucker installiert, so kann mit P die Druckfunktion gestartet werden. Nach der Eingabe des Dateinamens erscheint eine Reihe von Fragen. Jede Frage hat ihre Standardantwort, die durch die (ET)-Taste ausgelöst werden kann. Sollen alle Fragen mit der Standardantwort beantwortet werden, so kann bereits der Dateiname anstelle von (ET) mit (ESC) abgeschlossen werden.

Die Aktivierung der Druckfunktion P bringt folgende 7 Ausschriften auf den Bildschirm, die entsprechend den Wünschen des Nutzers zu quittieren sind.

– Druckausgabe auf eine Datei? (J/N)

Standardantwort ist N (Nein) (Text wird auf dem Drucker ausgegeben.)

Die Beantwortung mit J (Ja) überträgt den Text nicht zum Drucker, sondern in eine andere Datei. Damit läßt sich eine Druckformatkontrolle (Überprüfung der Wirkung der Punktkommandos) durchführen, ohne daß Papier bedruckt wird. Man sieht sich diese „gedruckte“ Datei mit der D-Funktion auf dem Bildschirm an, fügt aber ganz am Anfang 3 Zeilen mit folgenden Punktkommandos ein.

.pl nn (Gesamtzeilenzahl pro Blatt, z. B. .pl 72)

.mt 0 (kein oberer Rand)

.mb 0 (kein unterer Rand)

Nach dem Durchsehen wird nicht gespeichert, sondern mit ^KQ und J die Datei zurückgewiesen.

- Ab welcher Seitennummer soll der Druck beginnen?
Bei Standardantwort wird die Textdatei von Anfang an ausgedruckt. Anderenfalls wird die Seitennummer eingegeben (mit (ET)-Taste abschließen), ab der ausgedruckt werden soll.
- Nach welcher Seitennummer soll der Druck enden?
Bei Standardantwort wird die Textdatei bis zum Ende ausgedruckt. Anderenfalls muß die letzte auszudruckende Seitennummer (mit (ET)-Taste abgeschlossen) eingegeben werden.
- Seitenvorschub-Befehl FF (OCHex) ? (J/N)
Standardantwort ist Nein; dann wird der Papiertransport nur durch Zeilenvorschubbefehle ausgeführt. Das ist die Einstellung für Endlospapier (Journalrolle oder Leporello) oder Einzelblattzuführung von Hand. Bei Beantwortung mit Ja wird bei jeder Seite der Befehl FF (form feed, OCHex) zum Drucker gesendet. Bei Druckern mit automatischer Einzelblattzuführung (sheet feeder) ist dieser Befehl notwendig zum Auswurf des alten und Einzug des neuen Blattes. Man kann aber auch bei Endlospapier oder Einzelblattzuführung von Hand mit diesem Befehl arbeiten. Im letzten Fall dient der Befehl vorwiegend zum Blattauswurf.
- Seitenformatierung unterdrücken? (J/N)
Standardantwort ist Nein (die Punktcommandos sind wirksam und werden deshalb nicht mit ausgedruckt). Bei Beantwortung mit Ja sind die Punktcommandos unwirksam. Die entsprechenden Zeilen werden mit ausgedruckt.
- Pause für Papierwechsel zwischen den Seiten? (J/N)
Standardantwort ist Nein. Sollen Einzelblätter von Hand zugeführt werden, muß mit Ja geantwortet werden.
- Drucker bereitmachen, RETURN drücken.
Der Druck beginnt, sobald diese Ausschrift mit (ET) quittiert wird.

Nachdem alle Druckfragen beantwortet wurden, erscheint wieder das Anfangsmenü.

Zu beachten ist, daß die Taste P verschiedene Funktionen auslösen kann. Die jeweils aktuelle Funktion von P wird im Menü angezeigt (Druck Datei, Stop Druck, Weiter Druck). Die anderen Tasten-Funktionen (außer R, X, und M) wirken auch während eines Druckes.

Wenn die Arbeit mit dem Textprozessor abgeschlossen werden soll, wird mit der Taste X das Betriebssystem SCP erreicht.

Die Funktion N dient (ähnlich wie D) zum Bearbeiten von Textdateien. Im Unterschied zur D-Funktion werden nur 7 Bit pro Zeichen in die Textdatei geschrieben. Deshalb sind Wortumschlag und andere Textformierung nicht möglich und nicht zulässig. Programmiersprachen wie BASIC und PASCAL

verlangen als Quelldaten nur Zeichen mit 7 Bit; das 8. Bit darf nicht gesetzt sein! Man spricht deshalb auch von Programmdateien, die in diesem Arbeitsregime bearbeitet werden können.

Über die Funktion M wird der KOMBO-Druck ausgelöst. M ist gegenüber P eine erweiterte Druckfunktion, bei der die Kombo-Druck-Punktcommandos (z. B. .fi text2) wirken und die Anzahl der zu druckenden Kopien angebar ist. Während des Kombo-Druckes sind andere Funktionen nicht möglich.

Mit der Funktion O können beliebige Dateien (nicht nur Textdateien) kopiert werden. Nach einem nicht abgeschlossenen Kopiervorgang (wenn z. B. die Diskette voll ist) ist die neu entstandene Datei zu löschen, da diese nicht vollständig ist, aber der Dateiname schon in das Diskettenverzeichnis eingetragen wurde. Mit Y kann eine beliebige Datei gelöscht werden.

Mit E können Name und Typ einer Datei geändert werden.

Mit F wird die Anzeige des Diskettenverzeichnisses unter dem Anfangsmenü ein- bzw. ausgeschaltet.

Mit der Funktion R kann ein anderes SCP-Programm (z. B. DIENST) geladen und gestartet werden. Nach dem Ende dieses Programms lädt sich das Textprogramm automatisch wieder nach und zeigt das Anfangsmenü. Außerdem sind alle vorherigen Einstellungen wie Hilfegrad, Wortumschlag, Einfügen, Randeinstellungen u. a. erhalten geblieben.

Damit das Ausdrucken eines Textes mit allen nutzbaren TP-Druckfunktionen (Hoch- und Tiefschrift, anderer Zeichenabstand usw.) richtig funktioniert, müssen die speziellen Druckercommandos (z. B. halbe Zeilenschaltung) dem Textprozessor bekannt sein. Außerdem muß für alle TP-Funktionen das Bildschirmsformat (Anzahl Zeilen und Spalten) passend zum Betriebssystem eingestellt sein. Weiterhin gibt es noch einige Änderungsmöglichkeiten für Standardeinstellungen (z. B. kann Wortumschlag (^OW) je nach Installation den Grundzustand ein oder aus haben).

Der Textprozessor wird mit dem Installationsprogramm TPINSTD installiert. Dieses Programm ist ebenfalls menügesteuert und erklärt sich selbst.

Der TP mit der Version V1/3 enthält eine integrierte *Tastenumcodierung*, damit die Cursor- und Funktionstasten automatisch mit einbezogen sind. Diese Tastenumcodierung ist jedoch starr und betrifft alle Tasten. Andere Tasteninstallationen sind wirkungslos. Die TP-Version V1/3A ist ohne eigene Tastenumcodierung. Eine Tastenumcodierung kann als gesonderte Datei vorher geladen werden. Zum Erstellen solch einer Datei gibt es das Tasteninstallationsprogramm KEYINST. Damit ist es möglich, von beliebigen Tasten ein beliebiges Zeichen oder eine Zeichenfolge auslösen zu lassen. Solch eine Tasteninstallation läßt sich für ein beliebiges SCP-Programm

(z. B. BASI) anwenden und ist nicht auf den Textprozessor beschränkt.

Der Textprozessor besteht aus folgenden vier Dateien:

- TPG.COM (oder ein anderer, durch TPINSTD erzeugter Name)
- TPOVLY0.OVR
- TPHT .OVR
- TPDRUCK.OVR

Aufgerufen wird nur die .COM-Datei. Die übrigen Dateien bzw. Teile von ihnen werden bei Bedarf automatisch in den Arbeitsspeicher geladen. Diese Dateien müssen sich auf dem selektierten Laufwerk oder auf Laufwerk A befinden.

Die Datei TPOVLY0.OVR wird für alle TP-Funktionen benötigt und muß deshalb immer verfügbar sein. Die Datei TPHT.OVR beinhaltet ausschließlich Menü-Bilder und andere Hilfstexte. Für einige Anwendungsfälle können versierte TP-Benutzer auf diese Datei verzichten. TPDRUCK.OVR ist eine Datei, die nur für den KOMBO-Druck (Funktion M) benötigt wird. Die Druckfunktion P braucht diese Datei nicht.

TP wird mit verschiedenen Versionsnummern vertrieben. Die Version muß bei allen vier Dateien gleich bzw. kompatibel sein. Mit Hilfe der CCP-Funktion TYPE können die Versionsnummern der einzelnen Dateien angezeigt werden (z. B.: A> type tpht.ovr).

6.4. Grafische Darstellungen mit dem Programm PCGRAF

Grafiken dienen in erster Linie dazu, Größenverhältnisse von Daten bildlich darzustellen. Es gibt 4 Haupttypen grafischer Darstellungen, die als Leitungsgrafiken (auch Geschäftsgrafiken genannt) bezeichnet werden:

- **Balkendiagramme**
Sie werden verwendet, um zu zeigen, wie sich eine bestimmte Größe (in der Regel über einen bestimmten Zeitraum) ändert.
(Es sollten nicht mehr als 5–10 Größen oder Zeiträume einbezogen werden, weil das Diagramm sonst unübersichtlich wird.)
- **Kreisdiagramme**
Sie werden verwendet, wenn von einer bestimmten Größe Anteile darzustellen sind. Die Gesamtgröße wird als Kreis, die Anteile werden als Sektoren im Kreis dargestellt.
- **Liniendiagramme**
Sie zeigen die Veränderung oder den Fortschritt einer Größe (in der Regel über einen bestimmten Zeitraum) für eine große Anzahl von Zeitperioden.

– Säulendiagramme

Sie werden verwendet, wenn von einer Größe Anteile darzustellen sind. Die Gesamtgröße wird als waagerechter Balken, die Anteile werden als Sektoren des Balkens dargestellt.

Alle diese Diagramme enthalten eine große Anzahl von Informationen in einfachen Bildern, die es ermöglichen, Trends und Relationen in den Informationen viel leichter zu überblicken als in einer Liste von Zahlen.

PCGRAF verkörpert ein modernes druckerorientiertes Softwarewerkzeug (Programmsystem), mit dessen Hilfe eine Vielfalt von Diagrammen erzeugt werden kann. Der Nutzer benötigt dazu kein spezielles Wissen oder Training. Die darzustellenden Daten können eingegeben oder von anderen Programmen wie REDABAS (oder in Programmiersprachen wie BASIC oder PASCAL geschriebenen Programmen) übernommen werden.

Im einzelnen werden mit Hilfe von PCGRAF folgende Leistungen erbracht:

1. Erstellung von Balken-, Linien-, Kreis- oder Säulendiagrammen,
2. Eingabe von Daten über Tastatur oder aus anderen Datendateien,
3. Änderungen, Einfügungen und Umordnungen von Datendateien durch Nutzung des Editors,
4. Hinzufügen von Überschriften, Daten, Schattierungen, Legenden und Texterklärungen zu den Diagrammen,
5. Auswahl von Daten bei existierenden Diagrammen, die bei einer weiteren grafischen Darstellung einbezogen oder ausgeschlossen werden sollen,
6. Sicherung, Korrektur oder Druck von bereits definierten Diagrammen,
7. Erzeugung von Abbildern eines Diagramms und Übertragung zu einem anderen Rechner,
8. Automatische oder manuelle Bestimmung aller Details eines Diagramms durch Autograf (Bestimmung der Maßstabgröße eines Diagramms),
9. Automatisches Einfügen von Diagrammen aus dem Inhalt anderer Programme.

Frei-Hand-Entwürfe, Urbilder oder spezielle Symbole können mit PCGRAF nicht bearbeitet werden.

Ein Diagramm hat immer die Größe 11 cm × 19 cm.

PCGRAF erscheint mit dem in Abbildung 6.3 dargestellten Hauptmenü. Hierbei gilt:

1. Alle Kommandos sind einfache Zeichen (Eingabe als Klein- oder Großbuchstaben).
2. Das Zeichen für das Kommando steht direkt davor.
3. Jede der 4 Gruppen von PCGRAF-Kommandos steht in einem getrennten Kästchen.

Eine wichtige Kommandotaste ist die Taste ESC. Damit ist die Wahl ver-

PCGRAF – LEITUNGSGRAFIK – RBWS V: 002
Druckerorientiert (K6313/K6314)

```

=====
( ) Start/Parameter -PCGRAF | Diagrammdefinition ( )
( ) ----- | ----- ( )
( ) T Tagesdatum ( ) | C Diagr.Name ( NAME ) ( )
( ) S Systemparameter | B Balkendiagramm ( )
( ) H Hilfsmenue | L Liniendiagramm ( )
( ) ESC Programm beenden | P Kreisdiagramm ( )
( ) | S Saeulendiagramm ( )
=====
( ) Datenaufbereitung | Diagrammerstellung ( )
( ) ----- | ----- ( )
( ) F Dateiname ( NAME ) | D Diagrammberechnung/Druck ( )
( ) N Neuaufbau | R Druck eines vorh. Diagramms ( )
( ) A Alte Datei | ( )
( ) 2 REDABAS-Datei ( NAME ) | ( )
( ) | ( )
=====
Kommando – Eingabe ---->

```

Abbildung 6.3
Das Hauptmenü von PCGRAF

schiedener Menüs, aber auch der Abschluß von PCGRAF möglich. ESC führt immer zu einem Menü zurück. Wurde eine Datei benannt, erscheint ihr Name anstelle von „NAME“.

Im Programmsystem PCGRAF werden folgende Dateitypkennungen benutzt:

“.DGF” (PCGRAF)	Diagrammdefinitionsdatei
“.DAT” (DATA)	Datendatei
“.BTP” (Bit Picture)	Diagrammbilder oder Bitabbilder
“.TXT” (Text)	ASCII-Textdateien
“.EXC” (Excaption)	Datei, die die ausgeschlossenen Sätze enthält
“.DBD” (Datenbasis)	REDABAS-Datenbasisdatei

Wenn Daten, die in Diagrammen dargestellt werden sollen, bereits in mit REDABAS erzeugten Dateien enthalten sind, kann die Eingabe dieser Daten direkt aus der REDABAS-Datei erfolgen. Zur Verbindung von PCGRAF mit REDABAS-Dateien sind folgende Schritte nötig:

- im Hauptmenü von PCGRAF wird mit Hilfe des Kommandos „2“ der Name der REDABAS-Datenbasisdatei angegeben („.DBD“-Datei);

- mit Hilfe des Kommandos „F“ werden die Felder definiert, die im Diagramm dargestellt werden sollen;
- im Anschluß an das Kommando „D“ werden die benötigten Daten aus der REDABAS-Datei automatisch herausgelöst und in die „.DAT“-Datei von PCGRAF übernommen. Diese entstehende Datei hat den gleichen Aufbau wie eine durch Eingabe der Daten erzeugte Datei.

Um den Leistungsumfang von PCGRAF zu demonstrieren, wurde eine REDABAS-Beispieldatei mit dem Namen "DEMO.DBD" erstellt. Die Struktur dieser Datei ist aus der Abbildung 6.4 ersichtlich.

Die nachfolgend wiedergegebenen Diagramme (Abbildungen 6.6 bis 6.10) sind Ableitungen aus der vorgegebenen REDABAS-Datei. Diese Diagramme berechnete PCGRAF aus der in Abbildung 6.5 beschriebenen Diagrammdefinition (DEMO1) und einer mit dem PCGRAF-Editor erstellten Datendatei (DEMO1.DAT).

Dargestellt werden unter anderem die prozentualen Anteile der Erlöse und Kosten an den Gesamterlösen bzw. Gesamtkosten für die Monate Januar bis März.

Das Kreisdiagramm in Abbildung 6.8 stellt nur die Berechnung der ersten Eingabespalte (Erlöse) dar, während das Liniendiagramm in Abbildung 6.9 die Erlöse und Kosten in ihrer Entwicklung wieder grafisch einander gegenüberstellt.

Unter Nutzung der REDABAS-Verbindung in PCGRAF wurde mit Hilfe der Definitionsdatei DEMO.DAT und der REDABAS-Datei DEMO.DBD das Diagramm in Abbildung 6.10 berechnet, das die Ergebnisse aus Erlösen minus Kosten für die Monate Januar bis März über die drei Abteilungen A, B und C darstellt.

Das Programmsystem PCGRAF arbeitet im Betriebssystem SCP. Es stellt spezifische Anforderungen an die Gerätekonfiguration:

- 2 Diskettenlaufwerke, je Laufwerk mindestens 240 KByte Speicherkapazität, wobei auf einem Laufwerk mindestens 90 KByte freie Kapazität verfügbar sein muß,
- grafikfähiger Drucker (z. B. K 6313/K 6314).

PCGRAF umfaßt eine Programm- und eine Arbeitsdiskette mit den nachfolgend aufgeführten Dateien:

- Dateien auf der Programmdiskette

PCGRAF.COM	Hauptprogramme PCGRAF
PCGRAF.PRГ	
GRCHART.COM	Diagramm-Berechnungsmodule
GRCHART.PRГ	

REDBAS-Datei "DEMO.DBD"
 ~~~~~

DATEISTRUKTUR FUER: B:DEMO .DBD  
 ANZAHL DER SAETZE: 00038  
 DATUM DER LETZTEN AENDERUNG: 00.00.00  
 DATENBANKDATEI IM PRIMAERZUGRIFF  
 FELD NAME TYP LAENGE DEZ  
 001 MONAT C 003  
 002 ERFDATUM C 008  
 003 ABTEILUNG C 001  
 004 ERLOESE N 008 001  
 005 KOSTEN N 008 001  
 006 BETRIEB C 005  
 \*\* SUMME \*\* 00034

| FELD  | NAME         | TYP | LAENGE  | DEZ           |
|-------|--------------|-----|---------|---------------|
| 00001 | JAN 31.01.86 | A   | 35000.0 | 20000.0 WERK1 |
| 00002 | JAN 31.01.86 | B   | 40230.0 | 25690.0 WERK1 |
| 00003 | JAN 31.01.86 | C   | 37287.0 | 22000.0 WERK1 |
| 00004 | FEB 28.02.86 | A   | 45040.0 | 27900.0 WERK1 |
| 00005 | FEB 28.02.86 | B   | 39000.0 | 17230.0 WERK1 |
| 00006 | FEB 28.02.86 | C   | 31250.0 | 27890.0 WERK1 |
| 00007 | MAR 31.03.86 | A   | 43745.0 | 33670.0 WERK1 |
| 00008 | MAR 31.03.86 | B   | 40285.0 | 34999.0 WERK1 |
| 00009 | MAR 31.03.86 | C   | 31134.0 | 19871.0 WERK1 |
| 00010 | JAN 31.01.86 | A   | 9500.0  | 6000.0 WERK1  |
| 00011 | JAN 31.01.86 | B   | 11230.0 | 8300.0 WERK1  |
| 00012 | JAN 31.01.86 | C   | 13270.0 | 9000.0 WERK1  |
| 00013 | FEB 28.02.86 | A   | 11490.0 | 8400.0 WERK1  |
| 00014 | FEB 28.02.86 | B   | 9010.0  | 8005.0 WERK1  |
| 00015 | FEB 28.02.86 | C   | 12000.0 | 7900.0 WERK1  |
| 00016 | MAR 31.03.86 | A   | 9300.0  | 4600.0 WERK1  |
| 00017 | MAR 31.03.86 | B   | 13530.0 | 9300.0 WERK1  |
| 00018 | MAR 31.03.86 | C   | 11400.0 | 6790.0 WERK1  |
| 00019 | JAN 31.01.86 | A   | 22340.0 | 16000.0 WERK1 |
| 00020 | JAN 31.01.86 | B   | 19300.0 | 13000.0 WERK1 |
| 00021 | JAN 31.01.86 | C   | 21940.0 | 12100.0 WERK1 |
| 00022 | FEB 28.02.86 | A   | 28500.0 | 14000.0 WERK1 |
| 00023 | FEB 28.02.86 | B   | 19350.0 | 8000.0 WERK1  |
| 00024 | FEB 28.02.86 | C   | 24300.0 | 12900.0 WERK1 |
| 00025 | MAR 31.03.86 | A   | 17900.0 | 11000.0 WERK1 |
| 00026 | MAR 31.03.86 | B   | 14600.0 | 6700.0 WERK1  |
| 00027 | MAR 31.03.86 | C   | 20100.0 | 10000.0 WERK1 |
| 00028 | JAN 31.01.86 | A   | 7900.0  | 4000.0 WERK1  |
| 00029 | JAN 31.01.86 | B   | 9450.0  | 3020.0 WERK1  |
| 00030 | JAN 31.01.86 | C   | 12130.0 | 6030.0 WERK1  |
| 00031 | FEB 28.02.86 | A   | 9000.0  | 5000.0 WERK1  |
| 00032 | FEB 28.02.86 | B   | 11990.0 | 7200.0 WERK1  |
| 00033 | FEB 28.02.86 | C   | 12130.0 | 8410.0 WERK1  |
| 00034 | MAR 31.03.86 | A   | 11000.0 | 6000.0 WERK1  |
| 00035 | MAR 31.03.86 | B   | 15000.0 | 5000.0 WERK1  |
| 00036 | MAR 31.03.86 | C   | 10340.0 | 3340.0 WERK1  |
| 00037 | APR 30.04.86 | A   | 10743.0 | 4380.0 WERK1  |
| 00038 | JAN 31.01.86 | Z   | 10390.0 | 6000.0 WERK1  |

Abbildung 6.4  
 Struktur und  
 Belegung der  
 Demonstrations-  
 datei

GRREADAB.COM    Verbindungsmodule zu REDABAS  
 GRREDAB.PRG  
 GREDIT.COM     Daten-Editor  
 GREDIT.PRG  
 PCGRAF.CFG     Daten der aktuellen PCGRAF-Konfiguration

– Dateien auf der Arbeitsdiskette

GRSCR.LIB        Hauptmenü, Hilfe und Diagramm-Entwurfsbilder  
 GR001.WRK       Hilfsdateien/Arbeitsdaten  
 GR002.WRK  
 GR003.WRK  
 GR004.WRK

\* \* \* \* \*    Diagramm - Definition - DEMO1    \* \* \* \* \*

Datum -30.04.1986

Diagr.-Typ - B

Dateinamen:

Datendatei - DEMO1  
 REDABAS    - NAME  
 Textdatei  - NAME  
 Bilddatei  - DEMO1

Titel :

Ueberschr. - ERLOESE UND KOSTEN  
 Horizontal - MONATE  
 Vertikal    - WERT IN M  
 Bezeichner - ERLOESE KOSTEN

Grenzen:

Minim. Wert    - 500.000  
 Maxim. Wert    - 2000.000  
 gr.Unterteil. - 500.000  
 kl.Unterteil. - 100.000

Option :

|          |     |             |     |
|----------|-----|-------------|-----|
| Shade    | - Y | Autograf    | - Y |
| Zomm     | - N | Stack/Balk. | - N |
| Accum    | - N | Datum       | - Y |
| Legende  | - Y | Textdatei   | - N |
| Mittelw. | - N | REDABAS     | - N |
| Absetzen | - N | Spalt.ausw. | - N |
| Save BTP | - Y | Druck       | - Y |

Abbildung 6.5

Definition eines Diagrammbeispiels, das in der Datei DEMO1.DGF abgelegt wird

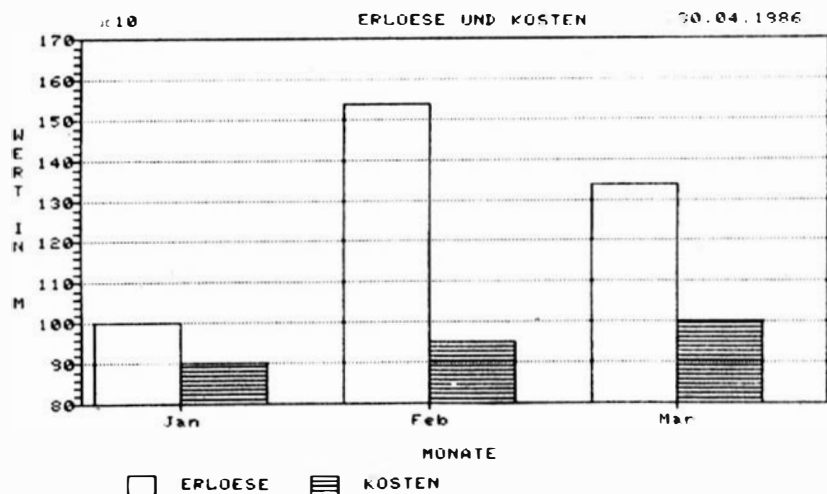


Abbildung 6.6  
Balkendiagramm – Erlöse und Kosten

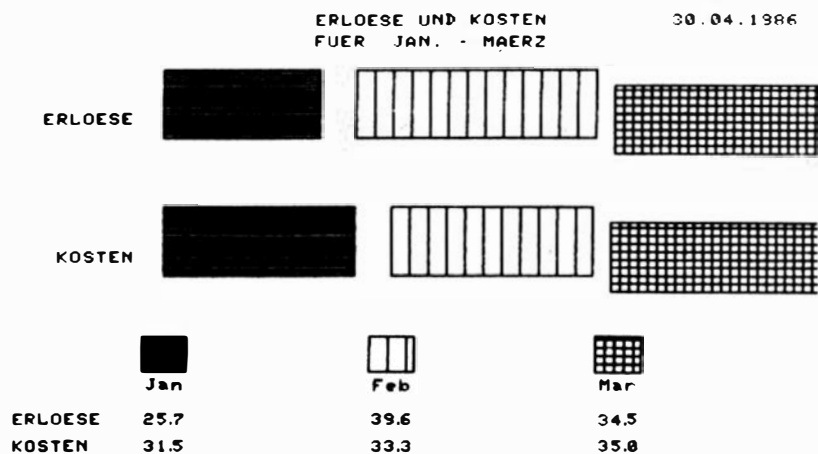


Abbildung 6.7  
Säulendiagramm – Erlöse und Kosten

ANTEIL DER MONATE AM

GESAMTERLÖS JAN. - MÄRZ

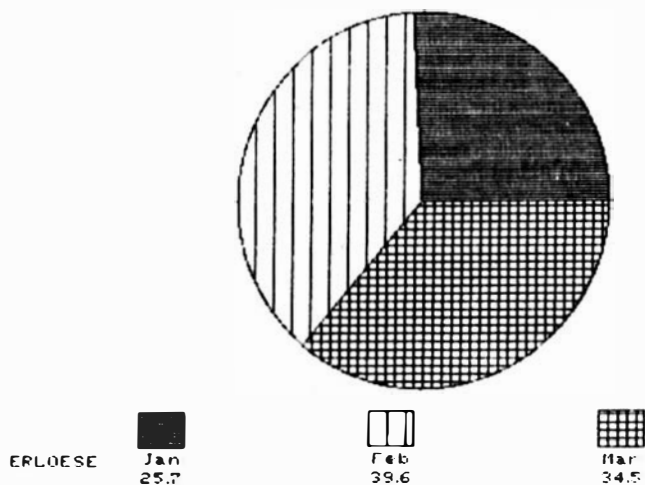


Abbildung 6.8  
Kreisdiagramm - Monatserlöse

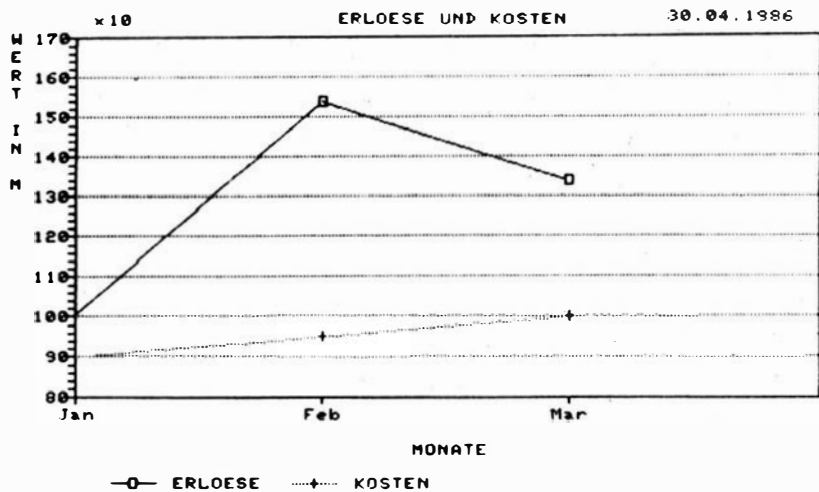


Abbildung 6.9  
Liniendiagramm - Erlöse und Kosten



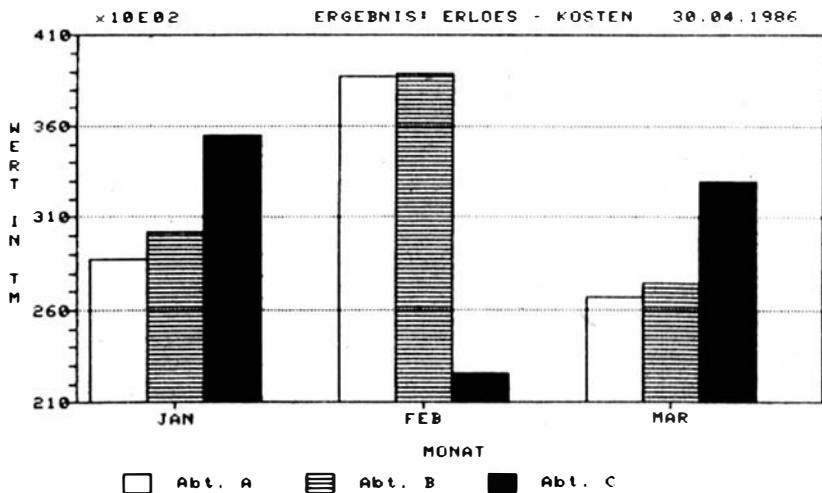


Abbildung 6.10

Balkendiagramm des Ergebnisses Erlöse - Kosten

## 7. Übersicht über Einsatzbereiche des PC 1715

Der PC 1715 hat bereits in einer Vielzahl sehr unterschiedlicher Einsatzfälle seine hohe und flexible Leistungsstärke unter Beweis stellen können. Im vorliegenden Kapitel soll ein Überblick über mögliche Einsatzbereiche gegeben werden. Dieser Überblick soll in erster Linie Anregung für Erstnutzer dieser Technik sein, die Hinweise benötigen, wie die Aufgaben des eigenen Wirkungsbereiches mit dem PC 1715 bearbeitet werden können. Es kann hier kein Vollständigkeitsprinzip angestrebt werden, da dies den Rahmen der Broschüre völlig sprengen müßte. Deshalb werden bisherige Einsatzbereiche nur kurz charakterisiert; auf wichtige gelöste Schwerpunktaufgaben wird hingewiesen. Dabei handelt es sich zum überwiegenden Teil um „klassische“ Einsatzfälle, in denen der Personalcomputer andere Gerätetechnik ablöste oder in bereits bestehende EDV-Lösungssysteme als dezentrale Gerätetechnik ergänzend oder vervollständigend eingeordnet wurde. Darüber hinaus sind aber auch neue Einsatzbereiche bekannt, von denen vier detaillierter behandelt werden.

Zuerst wird der Einsatz des Personalcomputers als erste CAD/CAM-Lösung in der Schrittmotorenfertigung des Büromaschinenwerkes „Ernst Thälmann“ in Sömmerda dargestellt. Hier liegt eine interessante Lösung für den Einsatz eines lokalen Rechnernetzes auf der Grundlage von PC-1715-Technik vor. Das zweite Beispiel beschreibt einen Einsatzfall aus der Konstruktion, in dem beträchtliche Rationalisierungseffekte und eine spürbare Entlastung von aufwendiger Routinearbeit zu verzeichnen sind.

Der dritte detaillierter behandelte Einsatzfall beschreibt einen rechnergestützten Arbeitsplatz für Augenoptiker auf der Grundlage des PC 1715. Dieses Einsatzbeispiel zeigt die prinzipielle Lösungsmethodik für eine große Klasse von Aufgaben, die als rechnergestützte Arbeitsplätze (vor allem auch in der Verwaltung) bezeichnet werden. Es handelt sich um typische Aufgabenstellungen, die besonders effizient mit Hilfe des PC 1715 bearbeitet werden können.

Schließlich soll mit dem vierten Einsatzfall auf weitere grafische Darstel-

lungsmöglichkeiten (Sektor- und Säulendiagramme) bei ökonomischen und bei wissenschaftlich-technischen Berechnungen hingewiesen werden. Damit sollen fortgeschrittene Nutzer angeregt werden, aus ihrem eigenen Wirkungsbereich resultierende Aufgaben entsprechend zu gestalten, gegebenenfalls unter Einbeziehung der dargestellten Softwarekomponenten.

## 7.1. Einsatzbereiche und Einsatzfälle

Die informationsverarbeitende Technik in Form des Personalcomputers und der mit ihm immanent verbundenen Software verändert die Stellung des Menschen im Produktionsprozeß und ermöglicht eine bedeutende Steigerung der Arbeitsproduktivität. Die sozialistische Rationalisierung und Automatisierung ist darauf gerichtet, den Menschen von Prozessen zu entlasten, in denen sein Reaktionsvermögen und seine Leistungsfähigkeit überfordert sind. Voraussetzung dafür ist, daß Funktionen menschlicher Intelligenz an die informationsverarbeitende Technik übertragen werden können. Diese Übertragung erfolgt durch die Software, also durch Programme. Im folgenden werden verschiedene Beispiele angeführt, die diesen Übertragungsprozeß demonstrieren. Dabei lassen sich zwei Typen unterscheiden. Der eine Typ umfaßt fachgebietsbezogene Programme, während der zweite Typ zu zweig- und bereichsbezogenen Beispielen führt.

Es wurden unter anderem folgende Softwarelösungen für den Einsatz des PC 1715 entwickelt:

- Erstellung von Binnenmarktrechnungen,
- Erstellung von Exportfakturen,
- Programm „SALDOKONTO“,
- Lohn- und Gehaltsrechnung,
- Arbeitsplatzbezogene Information und Dokumentation,
- Auftragsbearbeitung in der Kfz-Instandsetzung,
- Warenabzug der Verkaufsabteilungen in CENTRUM-Warenhäusern aus den Außenlagern.

Darüber hinaus gibt es eine Fülle weiterer Lösungen, aus der beispielsweise angeführt werden sollen

- der Einsatz eines PC 1715 im Betrieb Projektierung des WBK Erfurt und
- der Einsatz des PC 1715 in der Forstwirtschaft der DDR.

### **7.1.1. Einsatz des PC 1715 für die Erstellung von Binnenmarktrechnungen**

Grundlage für die Fakturierung bilden die vom Fertigwarenlager erstellten Versandscheine. Konstante Daten werden einmalig auf Diskette abgespeichert und können dann zu jeder Zeit abgerufen werden. Dabei handelt es sich um folgende Daten: 1. Absender, 2. Bankkontonummer, 3. Postschecknummer, 4. Artikelbezeichnung, 5. ZAK-Nummer, 6. HSL-Nummer. Suchkriterium für die Artikel sind Artikelnummern. Die Kundenanschriften liegen auf Diskette vor. Damit wird die mehrfache Erfassung eingespart und die Fehlerquote auf ein Minimum reduziert. Suchkriterium bildet die Kundennummer. Eine Erweiterung der Kundendatei ist jederzeit möglich. Entsprechend dem Auswahlprogramm können mehrere Programme abgearbeitet werden.

### **7.1.2. Einsatz des PC 1715 für die Erstellung von Exportfakturen**

Das Projekt dient zur Fakturierung von Rechnungen, die Exportaufträge (EA) zur Grundlage haben, deren Daten für alle Teillieferungen weitestgehend konstant bleiben.

Bei der ersten Teillieferung zu einem Exportauftrag werden alle konstanten Daten bei der Eingabe in externen Dateien erfaßt, so daß bei jeder der nachfolgend zu fakturierenden Teillieferungen zu diesem Exportauftrag ein hoher Automatisierungsgrad durch Ausnutzung von Käufer-, Vertreter-, Empfänger- und Spezifikationsstammdaten erreicht wird.

Die gesamte Bedienung erfolgt über den Bildschirm.

Im Zuge der Fakturierung kann über eine Wiederholung der Ausschreibung von Faktur und Spezifikation/Packliste entschieden werden. Da die in der Exportfakturierung vorkommenden Rechnungsarten von ihrem inneren Aufbau sehr voneinander abweichen können, zum Teil sogar völlig unterschiedlich sind, wurde das Gesamtprojekt in einzelne autonome Teilprojekte gegliedert. Neben den unterschiedlichen Programmen zur Fakturierung für alle Wirtschaftsgebiete und Erzeugnisse werden folgende Programme angeboten, mit denen ein sicherer Dauerbetrieb garantiert wird: Buchung zahlungsauslösend und intern; Nachberechnung; Umsatzberichtigung Vormonat; Austausch EA-Nr.; Verdichtungsprogramm.

### 7.1.3. Das Programm „SALDOKONTO“

Die Aufgabe des Programms ist die rationelle Erfassung und Verarbeitung von Produktionsdaten der Getränkeindustrie.

Das Programm besteht aus vier Programmteilen: 1. einlaufende Fakturen, 2. Korrekturen, 3. ausgestellte Fakturen, 4. Deckungen. Innerhalb dieser Programmteile können die Produktionsdaten von den Primärbelegen erfaßt und auf Diskette geschrieben werden. In jedem dieser Programmteile können die erfaßten Werte auf dem Bildschirm unter Angabe der Satznummer gelesen und korrigiert werden. Alle Programmteile können über das Programmauswahlbild angewählt werden.

Bei der Projektierung des Programmpaketes „SALDOKONTO“ wurde davon ausgegangen, daß die Eingabe der Datensätze sortiert erfolgt. Die Sortierung der Primärbelege erfolgt vor der Eingabe nach folgenden Prämissen.

| <i>Programmteil</i>   | <i>Sortierkriterien</i>                                         |
|-----------------------|-----------------------------------------------------------------|
| einlaufende Fakturen  | 1. Faktorensorte<br>2. Werk/Betrieb<br>3. Produktionsabteilung  |
| Korrekturen           | 1. Nummer des Belegs                                            |
| ausgestellte Fakturen | 1. Typ des Belegs<br>2. Werk/Betrieb<br>3. Produktionsabteilung |
| Deckungen             | 1. Sorte der Daten                                              |

Die Primärbelege werden sortiert erfaßt. Bei nur einem Sortierkriterium, wie bei den Programmteilen Korrekturen und Deckungen, sind die Primärbelege aufsteigend sortiert. Wählt man innerhalb der Programmteile die Funktion 'Druck' an, so werden bei Erkennen eines neuen Sortierkriteriums sowie am Ende der Datei Summen gebildet und ausgedruckt.

Die Bedienungsführung des Programms „SALDOKONTO“ erfolgt über den Bildschirm. Alle erforderlichen Maßnahmen, Eingaben und Entscheidungen werden auf dem Bildschirm angezeigt.

### 7.1.4. Lohn- und Gehaltsrechnung mit dem PC 1715

Das Projekt ist vorgesehen für die Abrechnung von Prämienstücklohn, Prämienzeitlohn, Monatslohn und Gehalt auf der Basis von Stunden.

Die Anwendungslösung teilt sich in zwei Hauptkomplexe. Der erste Kom-

plex beinhaltet Dateiaufbau und -pflege sowie die Bruttorechnung Lohn mit anschließender Errechnung der Normerfüllung. Im zweiten Komplex ist nochmals die Bewertung fehlender Arbeitsnachweise Lohn möglich. Weiterhin erfolgen die Bewertung der Arbeitsnachweise Gehalt, die Nettolohn- und Nettogehaltsrechnung, die Berechnung des Durchschnittslohnes, der Druck der Auszahlungs- und Überweisungslisten bis hin zur Geldstückelung und der Nachweis seit Jahresbeginn. In jedem Komplex wird eine Zusammenfassung nach Kostenstellen und Betrieb-Gesamt durchgeführt.

Die Abarbeitungszeit am PC liegt für Prämienzeitlohn, Monatslohn und Gehalt unter einer Minute pro Mitarbeiter. Bei Prämienstücklohn liegt sie in Abhängigkeit von der Anzahl der geleisteten Arbeitsgänge zwischen einer und vier Minuten.

### 7.1.5. Arbeitsplatzbezogene Information und Dokumentation mit Hilfe des PC 1715

Der moderne Forschungs-, Entwicklungs- und Konstruktionsprozeß ist durch die Nutzung moderner Rechentechnik und automatisierter Meßtechnik geprägt. Damit verbunden ist ein Datenaufkommen in solchen Ausmaßen, daß eine effektive Speicherung und Verfügbarmachung der Daten mit konventionellen Methoden nicht mehr möglich ist. Die Anwendung automatisierter Lösungen ist unumgänglich. Neben den im Prozeß anfallenden Daten sind im Rahmen einer Problemstellung immer Literaturinformationen zu verarbeiten. Soll Parallelarbeit vermieden werden, müssen Literaturrecherchen unverzichtbarer Bestandteil jeder Forschungs-, Entwicklungs- und Konstruktionsstätigkeit sein.

Im Zuge der immer breiteren Nutzung moderner Computertechnik am Arbeitsplatz bietet es sich an, diese Probleme mit Hilfe der informationsverarbeitenden Technik zu lösen. Zu diesem Zweck wurde das Programmpaket KOMKAR I entwickelt. KOMKAR I wurde auf der Grundlage des Datenbanksystems REDABAS erstellt. Es beansprucht etwa 25 KByte Speicherplatz.

Zur Verwaltung der Dokumentationseinheiten werden Datenbankdateien mit folgender Strukturbeschreibung angelegt:

| FELDNAME | FELDTYP | FELDLÄNGE |
|----------|---------|-----------|
| DOKNR    | C       | 003       |
| TITEL    | C       | 080       |
| AUTOR    | C       | 040       |

| FELDDNAME | FELDTYP | FELDLÄNGE |
|-----------|---------|-----------|
| ZEITSCHR  | C       | 030       |
| JHRGJHRNR | C       | 010       |
| SEITE     | C       | 006       |
| DESKR     | C       | 120       |
| STANDORT  | C       | 004       |

Ein Datensatz besteht aus 8 Feldern mit 294 Byte Gesamtlänge. Mit Hilfe eines Eingangsmenüs werden die beiden Grundfunktionen „SUCHEN im Datenbestand“ und „ÄNDERN des Datenbestandes“ angeboten.

Die SCHLAGWORTSUCHE als wesentliche Funktion erlaubt das Suchen nach 1...3 Schlagwörtern wahlweise im Feld TITEL bzw. im Feld DESKR. Mit "AND" bzw. "OR" können die Suchbegriffe logisch verknüpft werden. Außer SCHLAGWORTSUCHE bietet KOMKAR I die AUTORENSUCHE und die JAHRUCHE bzw. die Kombination beider an. Damit wird es möglich, folgende Ausgabeleistungen mühelos zu realisieren:

- Selektion der abgespeicherten Publikationen nach dem Alter,
- Ausscheiden veralteter Publikationen,
- Ausgabe von Jahresübersichten.

Bei Verwendung eines Handkarteisystems wäre dazu entweder ein unvertretbar hoher Zeitaufwand notwendig, oder es müßte eine weitere Kartei (nach Jahren sortiert) angelegt werden.

Über die ÄNDERUNGS-Funktion können folgende notwendige Manipulationen des Datenbestandes vorgenommen werden:

- Eröffnen (Anlegen) von Dateien,
- Einschreiben in Dateien,
- Ändern von Daten in Dateien,
- Löschen einzelner Datensätze,
- Löschen ganzer Dateien.

Sowohl beim SUCHEN als auch beim ÄNDERN werden die einzelnen Teilfunktionen wie Bildschirmanzeige, Druck und Rücksprünge jeweils über geeignete Menüs angeboten und durch die Menüprogramme angesteuert. Die Erfassung einzugebender Daten erfolgt über geeignete Editiermasken. Damit gestaltet sich der Umgang mit dem System für den Nutzer völlig problemlos. Durch Tests wurde die „Laienbenutzbarkeit“ mehrfach nachgewiesen.

### **7.1.6. Einsatz des PC 1715 für die Auftragsbearbeitung in der Kfz-Instandsetzung**

Kraftfahrzeuginstandsetzungsbetriebe sind auf Grund ihrer durchschnittlichen Größenstruktur, ihrer Leistungsspezifik und der anfallenden Datenmengen zu den klassischen Einsatzfällen zu rechnen. Der unseren Betrachtungen zugrunde liegende Betrieb (bestehend aus einem Hauptwerk und mehreren Meisterbereichen sowie einigen auswärtigen Betriebsteilen) führt neben der Reparatur verschiedener Pkw- und Lkw-Typen auch Baugruppenregenerierungen, Diagnosen und Handelsleistungen aus. Bei einer Betriebsgröße von 350 Beschäftigten werden allein in der Leistungssäule Kfz-Reparaturen täglich 70 Fahrzeuge instandgesetzt. Insgesamt fallen pro Tag Ausgangsrechnungen in diesen Größenordnungen an.

Die Abrechnung der Reparaturaufträge in der Kfz-Instandhaltung berührt prinzipiell alle Betriebsbereiche. Beginnend bei der Auftragsannahme durchlaufen die Belege, die aus dem Auftrag resultieren, die Meisterbüros, die Materialbuchhaltung, das Lohnbüro, die Abrechnungsabteilung und die Rechnungslegung. Den Abschluß findet der Belegdurchlauf im Debitorenkontokorrent. Die Abarbeitung aller betrieblichen Abrechnungsprozesse erfolgte bisher nahezu vollständig auf manueller Basis.

Mit dem Einsatz der modernen Informationsverarbeitungstechnik und dem vorliegenden Projekt Auftragsbearbeitung in der Kraftfahrzeuginstandsetzung werden folgende Abarbeitungsprozesse rationalisiert: Fakturierung, Gutschriften, Abrechnung der Barverkaufserlöse, Errechnung der Gesamterlöse je Meisterbereich, Leistungsart und Bedarfsträger, Errechnung der Buchwerte der Erlöse im Haben, der Gegenkonten im Soll, Buchung der Zahlungseingänge, Berechnung der Verspätungszinsen, Mahnwesen.

Das Projekt ist in die beiden Teilprojekte Fakturierung und Debitorenkontokorrent gegliedert.

### **7.1.7. Einsatz des PC 1715 in den CENTRUM-Warenhäusern für den Warenabzug der Verkaufsabteilungen aus den Außenlagern**

In den CENTRUM-Warenhäusern mit breitem Warensortiment in über 30 Verkaufsabteilungen muß täglich Ware von den Außenlagern abgerufen werden. Der Warenabzug erfolgt von der Verkaufsabteilung auf der Basis der Dispokarten (enthalten die Bestände); der Warenabzug erfolgt mit Warenanforderungsscheinen.



Nach der Warenanlieferung wird in der Verkaufsabteilung die tatsächlich gelieferte Menge mit dem Warenanforderungsschein verglichen; bei auftretenden Differenzen muß die Dispokarte korrigiert werden, um Übereinstimmung mit der Lagerfachkarte herzustellen.

Das Projekt „Warenabzug vom Fachlager“ wurde unter Nutzung des Datenbanksystems „REDABAS“ realisiert; es umfaßt folgende Funktionen:

- Erstellen einer Personaldatei
- Inventur
- Erfassung der Bestände und Wareneingänge
- Ergänzung/Korrektur der Datensätze
- Änderungen durch Protokolle, Warenausgangsrechnungen und Umlagerungsscheine
- Löschung von Datensätzen mit Nullbestand
- Warenabzug
- Bestandsübersichten.

Das Gesamtprojekt gliedert sich in 2 Teilprojekte: Erstellen der Personaldatei/Inventur und Warenabzug.

### **7.1.8. Einsatz eines PC 1715 im Betrieb Projektierung des WBK Erfurt**

Im Betrieb Projektierung des WBK Erfurt wurden mit dem Einsatz einer neuen Wohnungsbaureihe die bisher üblichen Projektierungstechnologien verlassen, und es wurde auf den Einsatz von CAD-Lösungen für den Wohnungs- und Gesellschaftsbau orientiert. Durch die differenzierten Anforderungen aus dem innerstädtischen Bauen als Montagebau mußte eine Projektierungsform gefunden werden, mit der in Arbeitsplatznähe des Projektanten weitgehend individuelle Projektlösungen aus katalogisierten Teillösungen automatengestützt entstehen.

Es werden CAD-Lösungen erarbeitet und genutzt,

- bei denen „Bausteine“ für die verschiedenen Projektteile vollständig durchgearbeitet und EDV-technisch katalogisiert werden und
- mit denen komplette Gebäudelösungen durch Aggregation dieser „Bausteine“ mit den notwendigen Modifikationen der Daten entstehen.

Die Projektdokumentationen werden als Listendruck und als Zeichnungen bereitgestellt.

Die im Betrieb eingesetzten PC 1715 sind in diesen Prozeß je nach dem zu bearbeitenden Projektteil unterschiedlich integriert. Es existieren Automatisierungslösungen in 4 Abarbeitungsformen:

**A – Systeme, die im alphanumerischen Dialog abgearbeitet werden**

**B – Systeme, die einen Datenerfassungs- und Vorverarbeitungsprozeß im alphanumerischen Dialog am PC 1715 realisieren. Deren Daten werden für den Hauptverarbeitungsprozeß auf dem ESER-Rechner als Jobstrom im offline Betrieb bzw. für die Datenfernübertragung genutzt.**

**C – Systeme, die den PC 1715 als Ausgabegerät für auf dem ESER-Rechner bzw. am CAD-Arbeitsplatz erzeugte Drucklisten und Projektzeichnungen nutzen**

**D – Systeme, die am CAD-Arbeitsplatz Bauwesen, bestehend aus Personalcomputer, Rastersichtgerät, Digitalisiergerät, Grafikdrucker bzw. Plotter und DFÜ-Anschluß im grafischen und alphanumerischen Dialog abgearbeitet werden.**

Die Nutzung des PC 1715 erfolgt im Betrieb Projektierung des WBK Erfurt grundsätzlich durch den Projektanten selbst in Arbeitsplatznähe.

Für folgende Teilprozesse werden Lösungen gemäß obigen Abarbeitungsformen genutzt:

- Entwurf und Konstruktion von Rohbauelementen (Abarbeitungsform D)
- Bewertung von Bebauungslösungen mittels Kennzahlen (Abarbeitungsform A)
- Gebäudestatik für Konstruktionen in Wandbauweise (Abarbeitungsform B)
- Montagerohbau in Wandbauweise (Abarbeitungsformen B und C)
- Ausbau (Abarbeitungsform A)
- hydraulische Berechnung für Heiznetze (Abarbeitungsform A)
- Dimensionierung von Zweirohrheizungsanlagen (Abarbeitungsform A)
- Preisberechnung im Wohnungsbau (Abarbeitungsform A)
- bauwirtschaftliche und bautechnologische Bearbeitung von Gebäuden (Abarbeitungsform B)
- Textverarbeitung zur Projektdokumentation (Abarbeitungsform A).

Für die einzelnen Teilprozesse sind Weiterentwicklungen in Arbeit, die die vorhandenen Lösungen zu CAD-Lösungen qualifizieren werden.

Gegenwärtig sind ergänzende Entwicklungen in Arbeit, die dem Projektanten für weitere Projektteile die Nutzung des PC 1715 zur automatengestützten Projektierung gestatten, wie

- Projektteil Gründung

- Projektteil Vermessung
- Projektteil Lüftung
- Projektteil Sanitär
- Projektteil Elektroanlagen
- Projektteil Informations- und Antennenanlagen.

### **7.1.9. Einsatz des PC 1715 in der Forstwirtschaft der DDR**

Wie vor jedem anderen Wirtschaftszweig steht auch vor der Forstwirtschaft die Aufgabe, bei im wesentlichen gleichbleibenden Ressourcen die Leistungen in der Produktion ständig zu erhöhen. Außerdem muß die Forstwirtschaft den erhöhten Anforderungen an Landeskultur, Ertragssicherheit der Wälder usw. nachkommen. Um diese anspruchsvollen Aufgaben erfüllen zu können, muß nach neuen Möglichkeiten zur Produktivitätssteigerung gesucht werden. Die konsequente Entwicklung und Nutzung moderner Informationstechnologien als zusätzliche Ressource ist deshalb auch für die Forstwirtschaft ein objektives Erfordernis.

Das Ziel besteht darin, die Rechner so weit wie möglich in die Produktion zu integrieren und ihre Effektivität durch zusätzliche Nutzung moderner Kommunikationstechnik weiter zu erhöhen. Moderne Mikrorechentchnik wie der PC 1715 und entsprechende Datenmodems (z. B. VM 2400) können hierfür heute schon die technische Basis bieten.

Für die etwa 100 Betriebe und Einrichtungen der Forstwirtschaft wird seit 1980 die Einsatzvorbereitung für Mikrorechner betrieben. 1981 verfügte das Organisations- und Rechenzentrum der Forstwirtschaft über den ersten BC A 5110 (Vorgänger des PC 1715). Heute sind im Wirtschaftszweig zahlreiche Personalcomputer stabil im Einsatz.

1980 wurde als Hauptziel festgelegt, mit Hilfe dieser leistungsfähigen DE/DV-Technik die Nachteile des bestehenden zentralen Datenverarbeitungssystems (z. B. Monatsrhythmus, Schwerfälligkeit) abzubauen, ohne auf dessen Vorteile (z. B. Stabilität, große Rechnerressourcen) verzichten zu müssen. Das gelang durch den Aufbau eines zwei-, teilweise mehrstufigen Datenverarbeitungssystems. Daneben sollten weitere Rationalisierungslösungen für bisher nicht durch die EDV erschlossene Bereiche entwickelt werden.

Im Ergebnis dieser Arbeiten stehen den Nutzern folgende Projekte zur Verfügung:

- Datenerfassungsprogramme für alle zentral auf ESER abzuarbeitenden Projekte, erweitert um umfangreiche formale und logische Tests und um Lösungen für Sofortinformationen

- Kostenrechnung
- Leistungsrechnung
- Fakturierung
- Arbeitskräfterechnung
- Grundmittelrechnung
- Materialrechnung
- Datenspeicher Waldfonds
- selbständige Abrechnungsprojekte
  - Materialdisposition
  - Finanzbuchhaltung
  - Inventur Forstbauschulen (anspruchsvolles Programm auf der Basis eines modernen mathematisch-statistischen Inventurverfahrens; finanzielle Bewertung der Pflanzenbestände)
- spezielle Programme für Planung, Bilanzierung, Produktionsvorbereitung, Produktionsplanung und Produktionssteuerung
  - normative Produktionsplanung
  - jährliche Waldnutzungskontrolle
  - Jahresdifferenzierung (Projekt zur zentralen Planung mit Optimierungseffekten)
- allgemeine Lösungen mit vielfachen Nutzungsmöglichkeiten
  - Programmpaket PLAN (universell einsetzbare Lösung zur Berechnung von Matrizen verschiedenster Dimensionen, Verdichtungsebenen und Generationen; Ausgabe verschiedenster Listenklassen; frei vorzuzugende Spalten-/Zeilenalgorithmen, Listenbezeichnungen und Spalten-/Zeilenbezeichnungen; bis jetzt über 50 Nachnutzungsfälle im Zweig und bei anderen Nachnutzern)
  - allgemeines Datenerfassungsprogramm
  - allgemeines Sortierprogramm, DTSO
  - Listengenerator, LISTG
  - Textverarbeitungssystem
- REDABAS-Lösungen
  - Waggonauslastung (automatische Erkennung des Waggontyps aus der Nummer (national und international; Gegenüberstellung zu Auslastungsnormativen)
  - Personaldatei
  - Forstpflanzenbilanz.

Diese schon recht umfangreiche Softwarebasis ist eine gute Grundlage für eine Reihe von selbständigen oder integrierten Lösungen wie z.B. CAM-Vorhaben in den Bereichen Produktionssteuerung, Material- und Lagerwirtschaft oder Lösungen für die zentrale Wirtschaftszweig-, aber auch für die betriebliche Planung.

## **7.2. Fertigungslenkungs- und Informationssystem auf Basis PC 1715**

Im VEB Robotron-Büromaschinenwerk „Ernst Thälmann“ (BWS) wurde in der Schrittmotorenfertigung (als Erstanwendung einer CAM-Lösung unter Steuerung des PC 1715) eine moderne automatisierte Produktionslinie aufgebaut, um über 1 Million Schrittmotoren bedienarm zu fertigen. Diese Motoren sind eine wichtige Baugruppe für alle Druckertypen, Diskettenlaufwerke und Fernschreiber. Die technologischen Ausrüstungen gliedern sich in 16 Automatenkomplexe, die teilweise direkt voneinander abhängen, deren Maschinen jedoch nicht durch Transportroboter gekoppelt sind. Ein UE-Lager versorgt die Produktion in den unterschiedlichen Fertigungsstufen (Vorfertigung, Montage) mit Arbeitsgegenständen. Es werden gegenwärtig etwa 25 Motorenvarianten in 3 Baugrößen hergestellt. Jeder Motor besteht aus etwa 35 Einzelteilen bzw. Baugruppen.

Von der Lenkung der Stoff- und Informationsflüsse des Produktionssystems und von der Beherrschung der Organisation der Fertigung hängen u. a. die Steigerung der Arbeitsproduktivität, die Senkung der Produktionskosten und die Termintreue ab.

Mit dem Fertigungslenkungs- und Informationssystem (FLIS 1715) wurde ein erweiterungsfähiges Softwarepaket entwickelt, das dem Fertigungsleiter bzw. seinen Mitarbeitern in Disposition und UE-Lager über Bildschirmterminals zu allen wesentlichen Daten des Produktionsprozesses im Echtzeitbetrieb Zugriff gewährt. Das dafür notwendige Hardwaresystem sollte diesem Anspruch genügen, aber auch bezüglich der einmaligen und laufenden Kosten ökonomisch vertretbar sein.

Das Datenerfassungssystem A 6422 vom VEB Robotron-Elektronik Zella-Mehlis erfüllte die Bedingungen der Prozeßebene (Maschinenhalle, Lager). Der Einsatz von PC 1715 als Steuerrechner des Gesamtsystems und als Arbeitsplatzrechner des Fertigungsleiters bot sich vor allem deshalb an, weil verschiedene Bedingungen von „größeren“ Rechnerstationen (klimatisierter Rechnerraum, Bedienpersonal) in der Schrittmotorenfertigung zu größeren Problemen geführt hätten. Durch die Ausnutzung der Intelligenz des Datenerfassungssystems (selbständige Prüfungen, arithmetische Operationen) und der Verteilung von Funktionen der übergeordneten Ebene auf 2 untereinander gekoppelte PC 1715 konnte eine hohe Leistungsfähigkeit und Verfügbarkeit des Gesamtsystems erreicht werden.

Die Hardware des FLIS 1715 setzt sich aus 3 Hauptkomponenten (siehe Abbildung 7.1) zusammen:

- Steuerrechner PC 1715 (Master)

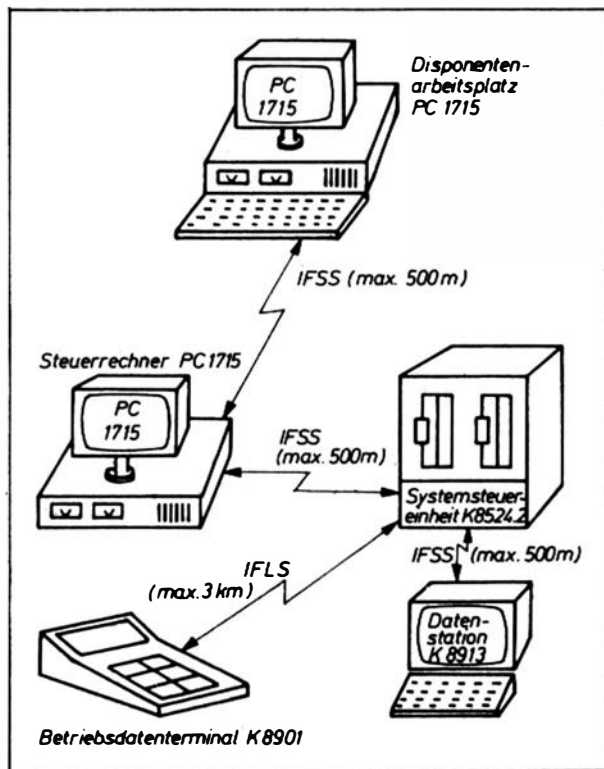


Abbildung 7.1  
Die Hardware-  
konfiguration  
des FLIS 1715

- Disponentenarbeitsplatz PC 1715 (Slave)
- Datenerfassungssystem A 6422.

Diese sind über das Interface IFSS untereinander derart verbunden, daß der Disponentenarbeitsplatz und die Steuereinheit des Datenerfassungssystems jeweils eine IFSS-Schnittstelle des Steuerrechners PC 1715 belegen. Der PC 1715, der als Steuerrechner eingesetzt ist, arbeitet mit dem Betriebssystem SCP. Als Mindestkonfiguration muß er über folgende Module verfügen:

- Grundgerät, Bildschirm, Tastatur
- 2 Diskettenlaufwerke (5,25")
- 2 IFSS-Schnittstellen (Zusatzsteckeinheit ist erforderlich).

Die Tastatur wird nur zum Systemstart und für die Dateikopierfunktion benötigt. Ansonsten sind keine Eingaben erlaubt.

Der PC 1715 hat 2 wesentliche Aufgaben, die er im 3-Schichtbetrieb zuverlässig erfüllen muß. Das Grundanliegen eines Rechnernetzes, das Ermögli-

chen eines ständigen Informationsaustausches zwischen verschiedenen Komponenten des Netzes, realisiert die DFÜ-Prozedur AP62/64 zwischen Steuerrechner PC 1715 einerseits und jeweils Disponentenarbeitsplatz PC 1715 und Steuereinheit des Datenerfassungssystems andererseits. Dabei sind über eine Systemschleife wechselseitig beide Hardwarekomponenten abzufragen, um vor allem einem Time-out-Zustand der Steuereinheit vorzubeugen. Die zweite Hauptaufgabe des PC 1715 ist das Verwalten der im Echtzeitbetrieb von der Fertigungsdisposition, dem UE-Lager und dem Produktionsprozeß ständig aktualisierten Dateien.

Ein zweiter PC 1715 arbeitet als Disponentenarbeitsplatz ebenfalls mit dem Betriebssystem SCP. Als Konfiguration wird vorausgesetzt:

- Grundgerät, Bildschirm, Tastatur
- 2 Diskettenlaufwerke (5,25")
- 2 externe Diskettenlaufwerke (5,25" bzw. 8")
- 1 Drucker K6312
- 1 IFSS-Schnittstelle (Zusatzsteckeinheit ist erforderlich).

Der Disponentenarbeitsplatz PC 1715 hat 2 wesentliche Betriebsarten, die die Universalität dieses speziellen Personalcomputers unterstreichen. Wie jeder andere PC 1715 kann er als Sologerät arbeiten und neben der SCP-Standardsoftware (TP, REDABAS usw.) die produktionsvorbereitenden Aufgaben (Dateipflege, Monatsplanung, Druck von Dateiausdrucken usw.) unterstützen. Die für das FLIS wichtigere Betriebsart ist die Verwendung des Disponentenarbeitsplatzes PC 1715 als intelligentes Terminal am Steuerrechner PC 1715. Die ohne Übertragungsverstärker mögliche Entfernung zwischen den beiden PC 1715 von 500 m für „kleinere“ Produktionsbereiche ist akzeptabel.

Das Datenerfassungssystem A 6422 ist ein modulares System zur Erfassung und Vorverarbeitung von Daten aus dem Produktionsprozeß und dem UE-Lager des Betriebsteiles Schrittmotoren. Das in diesem Betriebsteil angewendete System besteht aus folgenden Komponenten:

- Systemsteuereinheit K 8524.20
- Datenstation K 8913
- Betriebsdatenterminal K 8901.

Die Systemsteuereinheit koordiniert und steuert die Datenstation im UE-Lager und ca. 35 Betriebsdatenterminals in der Produktion und ist Mittler für den Informationsaustausch zwischen Prozeßdaten und den Dateien des Steuerrechners des Gesamtsystems. Die Datenstation prüft eingegebene Daten vor (z. B. Prüzfiffernkontrolle) und führt arithmetische Grundrechenarten eigenständig durch. Bildschirmbilder und Datenerfassungs- und Auswerteprogramme seitens der Datenstation werden auf einer Diskette der Systemsteuereinheit K 8524.20 gespeichert.

Die Betriebsdatenterminals sind zum großen Teil direkt an die Maschinen in der Schrittmotorenfertigung gekoppelt, so daß die Stückzählung automatisch erfolgen kann. Auftrags- und Personaldaten werden über die vorhandene Tastatur in das Betriebsdatenterminal eingegeben. Die Möglichkeiten, Prüflingen und arithmetische Operationen durchzuführen und Daten zu speichern, führen zu einem autonomen Einsatz der Betriebsdatenterminals. Durch die Nutzung der Intelligenz der Datenerfassungstechnik wurde es also möglich, anstelle eines leistungsfähigeren Rechners 2 gekoppelte PC 1715 einzusetzen.

Die Software des FLIS 1715 ist für die beschriebene Hardware weitgehend modular aufgebaut. Die für das Zusammenspiel notwendigen Schnittstellen sind in Form von Dateien und DFÜ-Telegrammen exakt definiert. Die Verwendung verschiedener Implementationssprachen ergab sich aus den Zielen der jeweiligen Teilaufgaben des FLIS 1715 und den konkreten Hard- und Systemsoftwarebedingungen. Die Programme am Steuerrechner PC 1715 wurden ausschließlich in Assembler (Z80- bzw. U 880-Mnemonic) realisiert. Das ist vor allem darin begründet, daß alle Programme so klein und so schnell wie möglich sein sollten. Am Disponentenarbeitsplatz PC 1715 waren andere Merkmale zu beachten, vor allem die höhere Komplexität der Programme und eine große Programmanzahl; dabei spielt die Abarbeitungszeit nicht die vorherrschende Rolle, da es sich vorwiegend um Dialogprogramme handelt, die den Menschen als langsameren Partner haben. Hier wird erfolgreich die Sprache BASIC angewendet. Das Datenerfassungssystem beruht auf einem eigenen Betriebssystem (EIEX 1521) und einer den speziellen Programmierbedingungen gut angepaßten Formatsprache, die ein einfaches und schnelles Programmieren ermöglicht.

Um die oben genannten Ziele zu erfüllen, ist ein enges Zusammenwirken der Nutzer des FLIS 1715 unbedingt erforderlich. Durch entsprechend wahrheitsgetreue und zeitgerechte Eingaben an der Datenstation K 8913 im UE-Lager bzw. an den Betriebsdatenterminals K 8901 im Produktionsprozeß ist es möglich, daß in den Dateien des Steuerrechners PC 1715 alle notwendigen Informationen zur weiteren Bearbeitung am Disponentenarbeitsplatz PC 1715 zusammenfließen können.

Der Echtzeitbetrieb garantiert eine hohe Aktualität der Daten, die den PC 1715 des Disponenten als Produktionsleitstand befähigen, den Leitungsprozeß wirksam zu unterstützen. Die Software des Disponentenarbeitsplatzes PC 1715 gliedert sich entsprechend den möglichen Betriebsarten in folgende Hauptfunktionen auf:

a) *autonome Betriebsart*

- Aufbau und Pflege von systemeigenen Stammdateien (Stücklisten-, Parameter-, Auftragsdatei)



- Erfassung und nachfolgende Auflösung von Planzahlen der zu produzierenden Enderzeugnisse, Baugruppen u. a.
- tägliche Auswertung der Produktionsabrechnung einschließlich der Aktualisierung der Auftragsdatei und der Lieferkontrolle an das UE-Lager
- Plan-Ist-Vergleich einzelner Planpositionen bzw. über das gesamte Auftragsortiment.

**b) interaktive Betriebsart**

- Einsteuerung von Fertigungsaufträgen auf der Basis von Informationen, wie disponible UE-Lagerbestände aller notwendigen Zulaufteile
- Fortschrittskontrolle bereits eingesteuerter Aufträge
- Informationsdienst über UE-Lagerbestände einzelner Arbeitsgegenstände, Zustände von Fertigungsaufträgen, Fehlmengen u. a.

Der Steuerrechner und das Datenerfassungssystem bilden die Quelle der Informationen aus dem Produktionsprozeß für die Arbeit des Disponenten mit seinem PC 1715. Die wesentlichen Dateien und ihre Funktionen sind:

- Bestandsdatei (Nachweis der Bestände aller Arbeitsgegenstände im UE-Lager)
- Fehlteildatei (Übersicht über Fehlmengen einzelner Arbeitsgegenstände von für die Produktion vorbereiteten Aufträgen)
- Dispodatei (Übersicht über alle disponierten Aufträge eines Produktionszyklus mit Kennzeichnung ihres Fortschrittsstandes)
- Maschinendatei (Verwaltung der maschinenspezifischen Daten, die über Betriebsdatenterminals erfaßt wurden, wie Auftrags-Nr., Stückzahl, Personal-Nr.).

Über eine hauptspeicherresidente Indexdatei wurde die Möglichkeit des direkten Zugriffs zu jedem Dateisatz realisiert. Die Datenstation K 8913 und die Betriebsdatenterminals K 8901 sind im Echtzeitbetrieb ständig mit den Dateien des Steuerrechners PC 1715 verbunden. Im FLIS kommen ihnen folgende Hauptfunktionen zu:

**a) Datenstation K 8913**

Die Software realisiert den Informationsaustausch zwischen UE-Lager und Steuerrechner PC 1715. Dazu wurden folgende Funktionen implementiert:

- Buchung von Lagerzu- und -abgängen einzelner Arbeitsgegenstände und kompletter Aufträge
- Informationsdienst (Anzeige des Bedarfs mit Priorität bei Lagerzugängen, Bestandsnachweis, Fehlmengennachweis)
- Prüfungen.

**b) Betriebsdatenterminal K 8901**

Die Software ist vorrangig auf die Kontrolle der Auftragsdurchführung und

deren Fortschrittsstand gerichtet. Um den Menschen von Stückzählung und deren Weitermeldung an die Produktionsleitung zu entlasten, wurden diese Prozesse weitgehend automatisiert. Alle Maschinen geben die produzierten Stückzahlen über Impulsgeber an das Betriebsdatenterminal weiter. Manuelle Eingaben für Handarbeitsplätze sind ebenfalls möglich. Es wurden dazu folgende Funktionen implementiert:

- An- bzw. Abmeldung der Arbeitskraft beim System,
- An- bzw. Abmeldung disponierter Aufträge,
- automatische bzw. manuelle Stückzahlerfassung und deren Anzeige,
- Prüfungen.

Das Hard- und Softwaresystem FLIS 1715 ist seit Anfang 1986 in der Schrittmotorenfertigung des VEB Robotron-Büromaschinenwerk „Ernst Thälmann“ Sömmerda im Einsatz. Es hat sich dabei gezeigt, daß die PC 1715 den Anforderungen, insbesondere der dreischichtigen Auslastung des Steuerrechners PC 1715, voll gerecht werden. Der Personalcomputer, der als Disponentenarbeitsplatz eingesetzt wurde, arbeitet in 2 Schichten. Nach einer kurzen Gewöhnungszeit (Nutzer – PC) akzeptieren ihn die Fertigungsdisponenten als „ihren“ Rechner, der durch seine vor allem zeitlich aktuellen Daten ein wichtiges Hilfsmittel darstellt.

Mit dem Einsatz des FLIS 1715 in der Schrittmotorenfertigung konnten neben der Gewinnung von 6 Arbeitskräften für andere Tätigkeiten folgende ökonomische Effekte verzeichnet werden:

- Verringerung der Lagerbestände,
- Senkung der Stillstandszeiten der Automaten,
- Verringerung der Durchlaufzeiten von Aufträgen,
- Qualifizierte Leitungsentscheidungen durch bessere Informationen.

Das Fertigungslenkungs- und Informationssystem 1715 erbrachte weiterhin den Nachweis, daß der PC 1715 in der Lage ist, in einem Rechnerverbundsystem für kleinere Fertigungsabschnitte die wesentlichen Funktionen zu übernehmen.

### **7.3. Einsatz eines PC 1715 zur Unterstützung der Konstrukteurarbeit**

Jeder Konstrukteur ist zu einem hohen Prozentsatz mit Routinearbeiten belastet, die sich formalisieren lassen. Dazu gehören alle Berechnungen nach festen Algorithmen sowie die Recherchen in Nachschlagewerken, Standards, Tabellen und Materialkatalogen. Neben der zeitlichen Belastung ist dies auch Ansatzpunkt für Fehler und ungünstige Lösungswege.









- Anzeige der Bestände aller Durchmesser/Güten eines Linsensatzes
- Löschen Linsen-/Fassungssätze
- Korrektur Linsen-/Fassungssätze
- Druck Linsen-/Fassungssätze.

Da für jeden Optiker *individuelle Identdaten* benötigt werden, können diese mit Hilfe einer speziellen Programmfunktion erfaßt bzw. korrigiert werden:

- Betriebs-Nummer
- Handelsobjekt-Nummer
- Ident-Nummer
- SV-Abrechnungs-Nummer.

Die Auftrags-Nummer stellt den Ordnungsbegriff für einen *Kundenauftrag* dar. Diese Nummer kann entweder eingegeben (wenn bereits vorhanden) oder vom Rechner auf Grund der einzugebenden Kundendaten (Anschrift, Geb.-Datum) neu errechnet werden. Nach der Eingabe des Verordners und des Kostenträgers können entsprechend den ermittelten optischen Daten für das rechte bzw. das linke Glas die jeweiligen Linsensätze auf Diskette gesucht werden. Dabei werden alle gespeicherten Durchmesser/Güten mit dem aktuellen Bestand angezeigt.

Nach der Auswahl der gewünschten Fassung (Anzeige von Bestand, EVP, SV-Leistung) kann der dazu passende Durchmesser (Güte) für das rechte und linke Glas ausgewählt werden (Anzeige: EVP und SV-Leistung).

Nach Bestätigung wird der entsprechende Auftrag gedruckt. Wenn die benötigten Linsen bzw. der notwendige Durchmesser nicht vorhanden sind, wird für den Kunden die Bestellung vorgemerkt.

In einem Kundenauftragssatz können maximal 2 Brillen bestellt werden. Nach Erfassung der letzten Brille des Auftrages wird der Abschnitt für den Kunden gedruckt. Die erfaßten Auftragsdaten werden in der Auftragsdatei gespeichert. Die Bestandsdaten der ausgewählten Linsen bzw. Fassungssätze werden korrigiert (Bestand -1, kum. Abgang +1).

Die Auswahl der Linsen-/Fassungssätze kann durch Eingabe der 11stelligen Code-Nummer oder durch Eingabe der Einzelparameter (Schlüssel-Nr., Durchmesser, Brennwert, Zusatzangaben bzw. Modell, Ausführung, Scheiben-/Stegbreite) erfolgen.

Werden für bestimmte Brillen Rezeptgläser benötigt, dann werden die dafür erforderlichen Daten in einer Hilfsdatei (REZEPT) gespeichert. In einem vom Nutzer festlegbaren Zeitraum wird eine automatische Bestellung von Lagergläsern ausgelöst. Die Bestellung wird nach der 5stelligen Schlüssel-Nr. vorgenommen. Dazu werden alle Linsensätze angezeigt, bei welchen der Mindestbestand unterschritten bzw. der Lagerbestand = 0 ist.

Zur Kontrolle werden die Bestände aller Durchmesser/Güten der jeweiligen

Brennweite ausgegeben. Vom Rechner wird die zu bestellende Menge (= Losgröße) vorgegeben. Diese Menge kann der Nutzer bestätigen oder korrigieren.

Ist die zu bestellende Menge ungleich 0, dann wird die Bestellung gedruckt, in die Datei „Bestellung“ übernommen, und es wird der Bestell-Bestand in der Linsen-Datei korrigiert. Die Bestellung der Fassungen erfolgt analog.

Aufgrund der in der Hilfsdatei (REZEPT) gespeicherten Rezeptglasdaten werden täglich die Rezeptgläser für jedes Rezept separat bestellt.

Nach Eingang einer Bestellung sind die Bestellnummer und die jeweiligen Code-Nummern mit den gelieferten Mengen einzugeben. Zunächst werden die Bestellungen in der Bestelldatei gesucht und die gelieferten Mengen abgebucht. Bei vollständiger Auslieferung einer Code-Nummer bzw. des gesamten Auftrages wird gelöscht.

Die eingegebenen Mengen werden in der Linsen-Datei bzw. in der Fassungs-Datei automatisch als Zugang gebucht. Liegen für Linsen bzw. Fassungen Vormerkungen für Kundenaufträge vor, dann wird eine entsprechende Druckliste erstellt.

Bei Bedarf kann vom Rechner eine Liste abgefordert werden, welche alle offenen Bestellungen ausweist.

Bei Fertigstellung eines Auftrages bzw. evtl. Rückfragen an den Kunden kann durch Eingabe der Auftrags-Nr. und der Textauswahl die Benachrichtigung gedruckt werden.

Bei der Auslieferung eines Kundenauftrages wird der Auftrag in eine spezielle Abrechnungsdatei übertragen. In einem vom Nutzer festzulegenden Zeitraum werden alle ausgeführten Aufträge gegenüber dem Kostenträger abgerechnet und dabei eine Druckliste erstellt. Abgerechnete Aufträge werden gelöscht.

Zur Durchführung der Jahresinventur können Inventurlisten über die vorhandenen Bestände ausgedruckt werden. Dabei werden alle Linsen nach den ersten 5 Stellen der Code-Nummer sortiert und der IAP je Linse, je Linsen-gruppe und insgesamt ausgewiesen.

## **7.5. Einsatz der Nadeldrucker K 6313/K 6314 für Grafikzwecke am PC 1715**

In fast allen Bereichen der Gesellschaft fallen ständig große Mengen Daten an, die in Tabellenform oder als Diagramme und Kurven dargestellt werden. Dies ist von Hand sehr zeitaufwendig und gelingt nicht immer anschaulich genug. Außerdem sind die Fehlereinflüsse bei manueller Bearbeitung be-



trächtlich. Mit dem verstärkten Einsatz von Personalcomputern in der Volkswirtschaft, an Hochschulen und Institutionen bietet es sich geradezu an, Lösungen zum direkten Umsetzen von statistischen und wissenschaftlichen Daten in Grafiken und Kurven zu schaffen.

Im Direktionsbereich Forschung und Entwicklung des VEB Robotron BWS entstanden Programme zum Drucken von Stab- und Kreisdiagrammen sowie von Meßkurven, wobei die Werte sowohl direkt über Tastatur eingegeben als auch von Diskette gelesen werden können. Außer dem PC 1715 mit dem Betriebssystem SCP ist dazu ein grafikfähiger Drucker K 6313/K 6314 notwendig.

Mit den an dieser Stelle gegebenen allgemeinen Hinweisen zur Erarbeitung grafikfähiger Programme wird dem Programmierer selbstverständlich nicht das Studium der Systemunterlagen und der Druckermanuals erspart.

Im Grafikmodus wird die Druckerausgabe nicht über den Zeichengenerator an Hand einer festen Codierung gesteuert. Der Programmierer muß selbst festlegen, wie viele der acht zur Grafik nutzbaren, senkrecht untereinander angeordneten Nadeln des Druckers anschlagen sollen und somit eine Sprosse auf dem Papier bilden.

Eine Zeile besteht dann aus  $n$  solcher Sprossen. Die maximal mögliche Anzahl ist in Tabelle 7.1 dargestellt. Dabei sind vertikal 72 Punkte pro Zoll möglich.

In Abbildung 7.8 wird eine solche Sprossenzeile vergrößert wiedergegeben. Die interne Verschlüsselung einer Sprosse erfolgt in einem Byte, in dem jeder Nadel ein Bit zugeordnet wird. Die unterste Nadel erhält den Wert 1, die oberste den Wert 128. Somit ist eine Codierung von 0 (keine Nadel) bis 255 (alle 8 Nadeln) möglich. Programmtechnisch ist dann dafür zu sorgen, daß pro Sprosse genau ein Byte (im Programm kann zur besseren Rechnung eine 2-Byte-Integergröße verwendet werden) mit der binären Darstellung X'00 bis X'FF zum Drucker gesendet wird. Vorher muß der Drucker jedoch in den Grafikmodus umgeschaltet werden und die Anzahl der als Grafiksprossen zu deutenden folgenden Byte mitgeteilt bekommen.

|         |          |          |          |          |          |          |
|---------|----------|----------|----------|----------|----------|----------|
| Nadel 8 | o        | o        | o        | o        | o        | o        |
| Nadel 7 | oo       | oo       | oo       | oo       | oo       | oo       |
| Nadel 6 | ooo      | ooo      | ooo      | ooo      | ooo      | ooo      |
| Nadel 5 | oooo     | oooo     | oooo     | oooo     | oooo     | oooo     |
| Nadel 4 | ooooo    | ooooo    | ooooo    | ooooo    | ooooo    | ooooo    |
| Nadel 3 | oooooo   | oooooo   | oooooo   | oooooo   | oooooo   | oooooo   |
| Nadel 2 | ooooooo  | ooooooo  | ooooooo  | ooooooo  | ooooooo  | ooooooo  |
| Nadel 1 | oooooooo | oooooooo | oooooooo | oooooooo | oooooooo | oooooooo |

Abbildung 7.8

Vergrößerte Darstellung einer Grafikzeile

Tabelle 7.1  
Horizontale Punktdichten der Drucker

| Typ    | Punkte/Zoll | max. Punkte/Zeile |
|--------|-------------|-------------------|
| K 6313 | 60          | 480               |
|        | 72          | 576               |
|        | 80          | 640               |
|        | 90          | 720               |
|        | 120         | 960               |
|        | 240         | 1920              |
| K 6314 | 60          | 816               |
|        | 72          | 979               |
|        | 80          | 1088              |
|        | 90          | 1224              |
|        | 120         | 1532              |
|        | 240         | 3264              |

Tabelle 7.2  
Grafikarten des K 6313/14

| m | Modus                   | Punkte/Zoll |
|---|-------------------------|-------------|
| 0 | normale Dichte          | 60          |
| 1 | doppelte Dichte (6"/s)  | 120         |
| 2 | doppelte Dichte (10"/s) | 120         |
| 3 | vierfache Dichte (6"/s) | 240         |
| 4 | CRT Grafik I (10"/s)    | 80          |
| 5 | Plottergrafik           | 72          |
| 6 | CRT Grafik II (8"/s)    | 90          |

Bei m=2 und m=3 werden aufeinanderfolgende horizontale Punkte nicht gedruckt.

Der Einfachheit halber werden die jetzt folgenden Kommandos in BASIC dargestellt; sie gelten für K 6313/14.

Umschalten von TEXT in GRAFIK: `LPRINT CHR$(27);"✱";CHR$(m);`

m ist der Parameter zur Steuerung der horizontalen Punktdichte (Sprossen pro Zoll); er ist Tabelle 7.2 zu entnehmen.

Längeninformation:

`LPRINT CHR$(n1);CHR$(n2);`

n = Anzahl der zu druckenden Sprossen

n2 = INT(n/256)

n1 = (n/256 - n2)\*256

Grafikinformation:

LPRINT CHR\$(I(1));CHR\$(I(2)); ...; CHR\$(I(n));

I(n) ist ein INTEGER-Feld und enthält die Kodierung der Zeile, wobei der Inhalt eines Elementes nicht größer als 255 sein darf. Die Längeninfor- mation und die Anzahl der folgenden Grafik-Byte müssen dabei genau überein- stimmen.

Der Wert CHR\$(9) ist nicht zulässig, da BASIC dies als Tabulation deutet. Ebenso darf bei 7-Bit-Übertragung (A 5120/30) die Länge und auch die Sprossenkodierung den Wert 127 nicht überschreiten.

Nachfolgend sollen einige Lösungen beschrieben werden, die auf dieser Grundlage entstanden.

### 7.5.1. Darstellung von Kurvenverläufen

Für die Darstellung von Kurvenverläufen gibt es zur Zeit zwei Grundvarian- ten:

- MESSKURV** – Darstellung von halblogarithmischen Kurvenverläufen (z. B. Dämpfungskurven elektrischer Schaltungen oder Meßkurven der Störspannung elektronischer Geräte als Funktion der Frequenz)
- GRAPH** – einfache Darstellung von Kurvenverläufen (Polygonzüge) in einer vorgegebenen Größe

Diese Programme wurden in der Programmiersprache PASCAL erstellt und bestehen jeweils aus 4 Teilprogrammen, einem Wurzelprogramm und drei funktionell getrennten Overlay-Routinen.

Der Leistungsumfang beider Programme ist ungefähr gleich, sieht man von der Spezialisierung des Programms MESSKURV einmal ab. Folgendes Lei- stungsspektrum wird durch die Programme erfaßt:

- Eingabe der Daten wahlweise über Tastatur oder von einer Diskettenda- tei,
- darstellbar sind maximal 50 verschiedene Kurven mit insgesamt 300 Stützstellen,
- die Darstellung kann in 4 verschiedenen Linienarten erfolgen,
- Angabe des Formates des zu druckenden Bildes in Anzahl Rasterpun- ken,
- Ausschnittdarstellung, Verkleinerung und Vergrößerung der zu drucken- den Bilder,
- Auswahl der Grafikdrucker (K 6313/14) und der Personalcomputer (PC 1715, A 5110/20/30),

- Löschen, Korrigieren und Erweitern der Daten bei Dateieingabe,
- Abspeicherung neu erstellter oder korrigierter Datenbestände auf Diskette,
- separates Mischen und Korrigieren der bestehenden Grafikdateien.

Zur internen Darstellung der grafischen Daten und zu deren Ausdruck gibt es verschiedene Möglichkeiten.

Eine Abspeicherung aller Grafikdaten im Hauptspeicher hat den Vorteil, daß eine schnelle Manipulation dieser Informationen erfolgen kann. So ist es möglich, in einem Hauptspeicherbereich von 30 KByte etwa 185 000 Bildpunkte abzuspeichern (Speicherung von 6 Bildpunkten pro Byte und reine Schwarz-Weiß-Darstellung). Dieses entspricht einem möglichen Bildformat von  $430 \times 430$  Punkten oder  $150 \times 150$  mm. Der verbleibende Speicherplatz ist dann aber zu gering, um effektive und leistungsfähige Grafikprogramme zu erstellen, da die Summe von Programmdatenbereich und Grafikdaten konstant ist. Dieses Problem kann unterschiedlich gelöst werden. Möglichkeiten dazu sind:

- Verwendung eines PC mit größerem RAM-Bereich,
- Abspeicherung der Daten auf Diskette (Nachteil: sehr langsame Erstellung des Bildes wegen großer Anzahl an Diskettenzugriffen)
- Verwendung eines anderen Druckprinzips.

Für die hier beschriebenen Druckprogramme MESSKURV und GRAPH wurde die dritte Möglichkeit gewählt. Diese soll hier kurz erläutert werden. Das Grundprinzip besteht darin, daß eine Druckzeile von 6 Punkten Höhe und maximal 990 Punkten Breite definiert wird. Während des Drucks wird diese Druckzeile bzgl. der Y-Achse vom Maximalwert zum Minimalwert verschoben. Die Schrittweite beträgt 6 Punkte. Zu jeder Druckzeile werden die anteiligen Kurvenelemente berechnet und anschließend gedruckt. Dieses Druckprinzip ist mit dem Bildaufbau einer Bildröhre vergleichbar. Die maximale Breite des Druckfensters von 990 Punkten resultiert aus den technischen Parametern. Die Druckgeschwindigkeit bei diesem Verfahren ist stark von der Anzahl der Punkte und vom Bildformat abhängig. Tests mit dem Programm MESSKURV haben gezeigt, daß für eine Darstellung von 4 Kurven mit insgesamt 100 Meßpunkten und einem Bildformat von  $500 \times 400$  Punkten die reine Druckdauer (Berechnung und Druck der Kurvenelemente) etwa 3 Minuten beträgt.

Es wird eingeschätzt, daß dieses Grafikverfahren eine Möglichkeit darstellt, auf einem 8-Bit-Personalcomputer relativ schnell Übersichtsgrafiken darzustellen, ohne jedoch die Qualität und den Komfort großer Systeme zu erreichen.

Ein Beispiel für eine Meßkurve ist in der Abbildung 7.9 dargestellt.

Abszisse : 2.0000E+01 - 3.0000E+01 MHz  
 Ordinate : 1.0000E+01 - 4.0000E+01 dB  
 Teilung : 5.0000E+00 dB

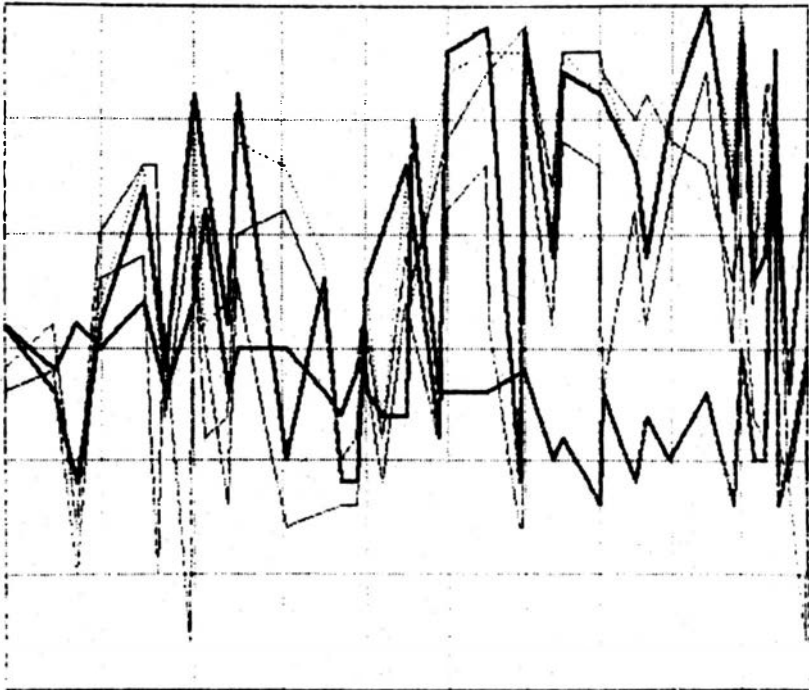


Abbildung 7.9  
 Darstellung einer Meßkurve

## 7.5.2. Darstellung von Stabdiagrammen

Dieses Programm existiert in drei Varianten:

- STABDIA1 – einfaches Stabdiagramm mit 11 wählbaren Grauwerten
- STABDIA2 – geteiltes Stabdiagramm (zwei Werte pro Balken) in Schwarz-Weiß-Darstellung mit Nullpunkt links (Abbildung 7.10)
- STABDIA3 – wie STABDIA2, aber mit Nullpunkt zwischen den Werten (Lebensbaumform).

Die Dateneingabe ist über Tastatur oder von einer Diskettendatei möglich

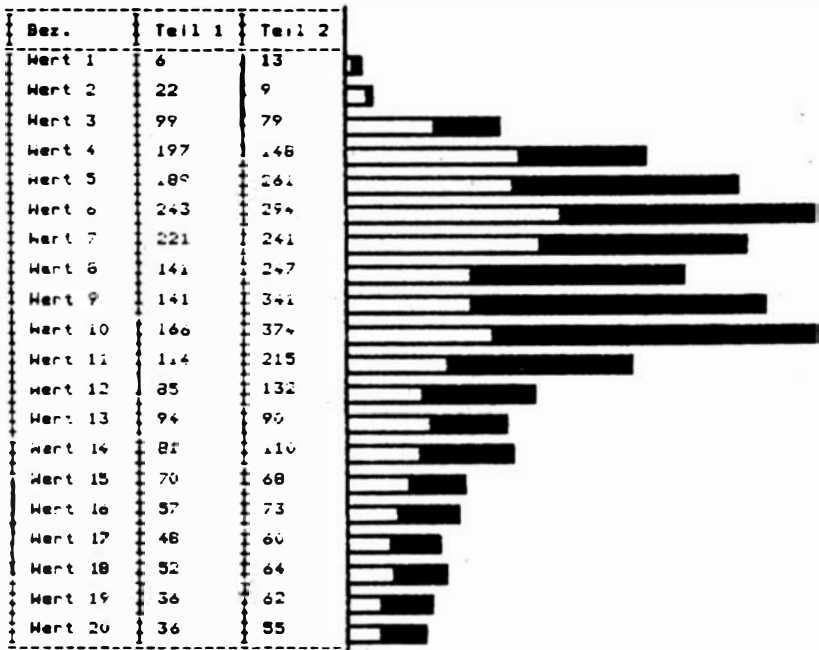
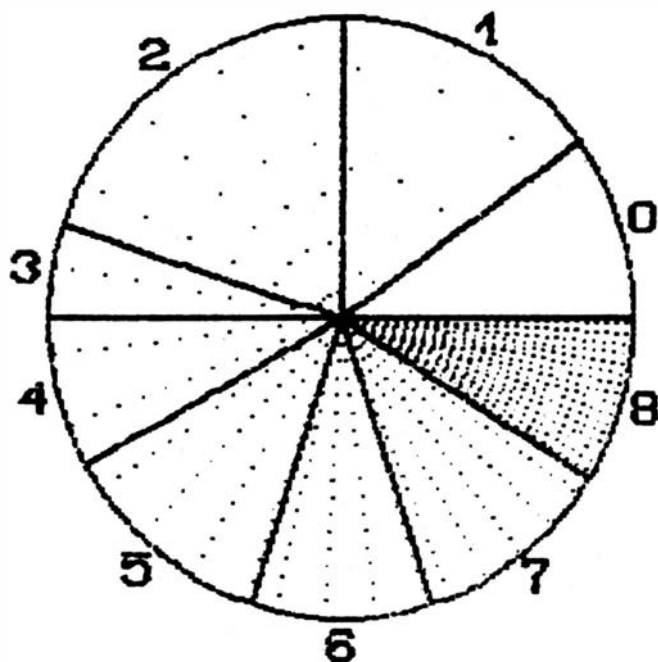


Abbildung 7.10  
Stabdiagramm nach Variante 2 (STABDIA2)

(maximal 200 Werte bzw. Wertepaare mit Bezeichnung). Wählbar ist auch die maximale Stablänge und die Balkenbreite. Der erläuternde Untertitel kann ebenfalls von der Tastatur oder von Diskette kommen.

### 7.5.3. Darstellung von Kreisdiagrammen

In einem Kreisdiagramm sind maximal 10 Werte in 11 verschiedenen Schraffurarten (Grauwerte) möglich. Die Erfassung der Daten und der zugehörigen Benennungen erfolgt mit dem Programm DIAERFAS. Mit DIADRUCK kann das als Diskettendatei abgespeicherte Diagramm dann beliebig oft gedruckt werden.



Erläuterung :

| Nr. | Bezeichnung | Betrag [%] |
|-----|-------------|------------|
| 1   | Anteil 1    | 10.00000   |
| 2   | Anteil 2    | 15.00000   |
| 3   | Anteil 3    | 20.00000   |
| 4   | Anteil 4    | 5.00000    |
| 5   | Anteil 5    | 8.00000    |
| 6   | Anteil 6    | 12.00000   |
| 7   | Anteil 7    | 10.00000   |
| 8   | Anteil 8    | 11.00000   |
| 9   | Anteil 9    | 9.00000    |

Abbildung 7.11  
Kreisdiagramm

## 8. Serviceleistungen für den PC 1715

In diesem Kapitel wird u. a. das Spektrum der möglichen Leistungen des Technischen Kundendienstes beschrieben. Dabei muß beachtet werden, daß die konkreten organisatorischen Regelungen territorial unterschiedlich sein können. Bei der Lieferung eines PC 1715 werden diese Besonderheiten durch den territorial zuständigen (vgl. Übersicht E) Vertriebsbetrieb mitgeteilt.

Das modular aufgebaute Systemkonzept des PC 1715 gestattet den Austausch ganzer Baugruppen. Dadurch ist es möglich, den Nutzer bei einem Ausfall seines Gerätes nicht mit einer Reparatur vor Ort belasten zu müssen. Bei schwierig zu diagnostizierenden Fehlern werden gleich mehrere Module ausgetauscht, damit die Verfügbarkeit des PC nicht durch eine lange Fehlersuche beeinträchtigt wird.

Die Methode des Baugruppentausches ist bei Monoprozessorsystemen, welche durch ein Betriebssystem gesteuert werden, günstig, weil bei Ausfall der Zentraleinheit oder bei Problemen mit dem Betriebssystem der gesamte PC ausfällt. Erst nach Beseitigung des Fehlers kann das Betriebssystem neu geladen oder gestartet werden.

---

### Übersicht E

#### Vertriebsbetriebe und Betreuungsgebiete

| Vertriebsbetrieb                        | Betreuungsgebiet                                                                     |
|-----------------------------------------|--------------------------------------------------------------------------------------|
| ROBOTRON-Vertrieb<br>Berlin             | Bezirke: Berlin, Neubrandenburg, Potsdam, Schwerin,<br>Rostock, Frankfurt, Magdeburg |
| ROBOTRON-Bürotechnik<br>Karl-Marx-Stadt | Bezirke: Dresden, Cottbus, Gera, Karl-Marx-Stadt                                     |
| ROBOTRON-Anlagenbau<br>Leipzig          | Bezirke: Leipzig, Halle                                                              |
| ROBOTRON-Vertrieb<br>Erfurt             | Bezirke: Erfurt, Suhl                                                                |

---



Zu den Serviceleistungen gehört auch ein Programm von Lehrgängen und Kursen, die die ROBOTRON-Schulungszentren anbieten:

- Technikerlehrgang PC 1715  
Dauer: 5 Wochen (davon 1 Woche am Drucker)
- Programmierung REDABAS  
Dauer: 1 Woche
- Programmierung TEXTPROGRAMM  
Dauer: 1 Woche
- Programmierung PASCAL  
Dauer: 3 Wochen
- Programmierung BASIC  
Dauer: 3 Wochen
- Programmierung BROS/ASSEMBLER  
Dauer: 3 Wochen

Die anlagenbezogene Aus- und Weiterbildung erfolgt in den Schulungszentren Leipzig und Berlin des VEB Kombinat Robotron. Zum Programm der Lehrveranstaltungen gehören die Ausbildung auf den Gebieten Organisation, Programmierung, Bedienung, Instandhaltung, Einsatzvorbereitung, Betrieb und Betreuung der Personalcomputer. Die Erstausbildung für den betriebseigenen Kundendienst und für Dozenten der ROBOTRON-Schulungszentren erfolgt im Unterrichtszentrum in Sömmerda. Hier erfolgt auch die Ausbildung der Techniker, Spezialisten und Programmierer ausländischer Serviceorgane.

## 8.1. Montage, Inbetriebnahme, Prüfung

Die Montage und Erstinbetriebnahme erfolgt je nach Liefervertrag entweder durch den Robotron-Kundendienst oder durch den Nutzer selbst. Beim Einbringen in Räume mit höherer Temperatur (Unterschiede größer als 5°C) sind die Geräte in verpacktem Zustand an die neue Temperatur anzugleichen. Das Entpacken erfolgt erst, wenn die Geräte zum Einsatz kommen sollen. Das Grundgerät des Personalcomputers besteht bekanntlich aus den Hauptkomponenten Systemeinheit, Tastatur und Bildschirm. In dieser Konfiguration ist der PC 1715 bereits arbeitsfähig, wenn auf einer Diskette das Betriebssystem zur Verfügung steht. Vorzugsweise in dieser Grundvariante kann der Nutzer anhand des mitgelieferten Manuals die Gerätekomponenten miteinander koppeln.

Es ist generell zu beachten, daß alle Gerätekomponenten nur in spannungslosem Zustand an- bzw. abgekoppelt werden dürfen!

In allen Fällen, bei denen die Logik-Konfiguration nicht fertig geprüft ange-

liefert, sondern beim Nutzer (z. B. durch Nachkauf von Logikmoduln) modifiziert wird oder wenn von der Grundvariante durch Anschluß von weiteren Laufwerken, Druckern, Plottern usw. erheblich abgewichen wird, übernehmen die Kundendienst-Abteilungen der Vertriebsbetriebe die Montage, Inbetriebnahme und Prüfung des Gesamtsystems.

Bei umfangreicheren Konfigurationen wird empfohlen, die Geräte des PC 1715 über einen separaten Verteiler ans Netz anzuschließen (dies kann eine Mehrfach-Schutzkontaktsteckdose sein). Der Verteiler darf einen Hauptschalter zum gemeinsamen Schalten aller Geräte enthalten. Für die Verbindung zu den einzelnen Geräten sind die gelieferten Schutzkontakt-Kabel zu verwenden. Der Verteiler ist nahe der Systemeinheit anzuordnen, die Leitungen zu den Geräten sind sternförmig zu führen. Fremde Geräte (außer zur Fehlersuche) sind an den Verteiler nicht anzuschließen.

Es ist ein Netzanschluß auszuwählen, an dem wenig Einflüsse durch äußere Störer (Schaltvorgänge, HF-Störungen) vorliegen. Eine Parallelführung von störungsführenden Leitungen mit der Verteilerzuleitung ist zu vermeiden, nötigenfalls sind Leitungen der sich beeinflussenden Netze abzuschirmen. Im Rahmen des PC 1715 werden nur solche Geräte verwendet, die den Status Sicherheits-Kleinspannung (IEC 380, VDE 0806) für alle Kleinspannungskreise besitzen.

Geräteexterne Interfaceverbindungen sind nur mit den für den PC 1715 gelieferten geschirmten Rundkabeln und geschirmten Steckern zulässig.

Interfaceleitungen dürfen mit Netzleitungen nicht parallel geführt werden bzw. es ist bei geschirmten Netzleitungen 0,3 m und bei ungeschirmten Netzleitungen 1,0 m Abstand einzuhalten.

Unter Beachtung dieser Hinweise kann der PC in Betrieb genommen werden. Die notwendigen Bedienerhandlungen sind im Manual beschrieben.

## 8.2. Wartung

Für den PC 1715 ist eine zyklische Wartung nicht vorgesehen. Die Techniker des Kundendienstes entscheiden in Verbindung mit einer Reparatur an einer Systemkomponente, in welchem Umfang Wartungsarbeiten vorzusehen sind. Dies geschieht unter Berücksichtigung der Auslastung und Ausfallhäufigkeit des jeweiligen Geräts. Nachfolgend sind einige Wartungsarbeiten, die am Standort des Gerätes notwendig werden können, aufgeführt:

|               |                                                                                |
|---------------|--------------------------------------------------------------------------------|
| Systemeinheit | Reinigung und Fetten sowie Justierung und Prüfen der Einbau-Minifolienspeicher |
|---------------|--------------------------------------------------------------------------------|

|                  |                                             |
|------------------|---------------------------------------------|
| Drucker 1152/251 |                                             |
| 1152/252         |                                             |
| 1152/257         | Reinigung und Ölen                          |
| Drucker 1157     | Justierung und Abarbeitung von Prüfroutinen |
| Drucker 6313     |                                             |
| 6314             | Reinigung und Fetten, Druckkopfwechsel      |
| Drucker 6316     | Reinigung und Fetten, Druckkopfwechsel      |

Tastatur und Monitor sind wartungsfrei.

Der konkrete Ablauf von Wartungsarbeiten sowie die Empfehlungen der Hersteller zum Austausch von Verschleißteilen sind den jeweiligen Servicehandbüchern bzw. Technischen Instruktionen zu entnehmen. Einige Pflegearbeiten kann der Bediener selbst vornehmen. Dazu gehören die regelmäßige Reinigung der Verkleidung mit einem neutralen Lackpflegemittel, die Reinigung des Bildschirms mit einem fusselfarmen Tuch, die Reinigung und das Fetten oder Ölen der Drucker (nach entsprechender Einweisung).

Alle Reinigungsarbeiten dürfen nur bei ausgeschalteten Geräten durchgeführt werden!

### 8.3. Instandsetzungsmaßnahmen am Standort des PC 1715

Der Baugruppentausch erfolgt, wenn Justage- und Einstellmaßnahmen nicht zum Erfolg führen, fehlerhafte Bauelemente prüftechnisch nicht ermittelt werden können bzw. die technologischen Voraussetzungen für eine qualitätsgerechte Reparatur an Ort und Stelle nicht gegeben sind.

Zur programmtechnischen Fehlerlokalisierung defekter Baugruppen wurden Servicetest- und Hilfsprogramme für den PC 1715 geschaffen, die mit dem Betriebssystem BROS arbeiten. Dieses Programmpaket für Techniker besteht aus einer Programmdiskette und aus einer Datendiskette. Es erfordert selbstverständlich, daß der „harte Kern“ des PC einsatzfähig ist (mindestens 1 Laufwerk, Tastatur, Bildschirm, ZRE, Stromversorgung). Nach bisherigen Erfahrungen werden mit diesen Hilfsmitteln sowie mit dem Prüfsatz (Steckeinheiten: ZRE, AS-MFS, StE V.24 bzw. IFSS) 70–80 % aller Fehler durch den Techniker lokalisiert. Dabei ist die Fehlersuchzeit bis zur kleinsten austauschbaren Einheit auf maximal 2 Stunden begrenzt.

Eine weitergehende Fehlersuche am Standort ist nicht sinnvoll, weil hierzu meist hochwertige Meß- und Prüftechnik notwendig ist. Diese Technik sowie entsprechende technologische Ausrüstungen haben die Vertriebsbetriebe in speziellen „Servicezentren“ konzentriert. Hier kann der Techniker die Feh-

ersuche an defekten Moduln und Geräten mit Hilfe einer Muttermaschine fortsetzen.

Der Nutzer kann die Verfügbarkeit seines Gerätes spürbar erhöhen, wenn er die Serviceeinrichtung bei einem Ausfall des PC 1715 mit einer detaillierten Fehlermeldung unterstützt, so daß der Servicetechniker bereits mit allen benötigten Baugruppen anreisen kann.

---

# Weiterführende Literatur

Schmidt, Kuno

*Das Betriebssystem SCP*

Verlag Die Wirtschaft, Berlin, 1987

Hempel, Ursula / Loley, Hans

*Datenbanken mit Personalcomputern*

Verlag Die Wirtschaft, Berlin, 1987

Petschke, Dieter / Müntefering, Peter

*Textverarbeitung mit Personalcomputern*

Verlag Die Wirtschaft, Berlin, 1987

AK, Leitung Bernstein, Klaus

*Tabellenkalkulation mit Personalcomputern*

Verlag Die Wirtschaft, Berlin, 1987

Der Hersteller des PC 1715 kann für Käufer von Hardware bzw. Software folgende Literatur zur Verfügung stellen:

- Als Nachschlagewerke für den Programmierer:
  - . Heft 1 Betriebssystem SCPx
  - . Heft 2 BASIC-Interpreter/BASIC-Compiler
  - . Heft 3 Assembler-Programmierung
  - . Heft 4 Tabellen und Arbeitsblätter

- für den Bereich Drucktechnik

Manual K 6313

K 6316

1152

1157

1156

sowie

- Manual robotron 1715
- Programmieranleitung zur Programmiersprache BASIC im Betriebssystem SCP
- Bedienungsanleitung und Sprachbeschreibung BASIC-Interpreter BASI
- Bedienungsanleitung BASIC-Compiler BASC
- Systemhandbuch SCP Anleitung für den Programmierer
- Systemhandbuch SCP Anleitung für den Bediener
- Systemhandbuch SCP Anleitung für den Programmierer  
Teil 2 – Assemblerprogrammierung
- Installationshandbuch für SCPx  
(Bedienungsanleitung für INSTSCP)
- Bedienungsanleitung und Sprachbeschreibung  
PASCAL-Compiler PASCMP
- Anwendungsbeschreibung und Bedienungsanleitung  
für das Kalkulationsprogramm KP  
KP 1715 (SCP)
- REDABAS Relationales Datenbankbetriebssystem für SCP
- Textprogramm „TP“ Anwenderbeschreibung und Bedienungsanleitung
- DIENST Diskettenserviceprogramm  
Anwendungsbeschreibung und Bedienungsanleitung
- Anwendungsbeschreibung und Bedienungsanleitung für das Sortierprogramm SORT  
SORT 1715 (SCP)
- Bedienungsanleitung und Sprachbeschreibung FORTRAN-Compiler  
FOR 1715 (SCP)
- Telekommunikation „TLC“  
Programm zur Datenübertragung
- KEYINST  
Tasteneinrichtung  
Anwenderbeschreibung und Bedienungsanleitung
- Konvertierung Basis-Austausch-Niveau  
Programm KONV
- Emulation EM 62

**Programm zur Datenübertragung****Nachbildung des Terminals EC 8562****Anwenderbeschreibung und Bedienungsanleitung**

- **Konvertierung von IBM-Disketten in das SCP-Format**

**Programm IBMTRANS**

- **SERIO**

**Programm zur seriellen Ein- und Ausgabe****Anwendungsbeschreibung und Bedienungsanleitung**

- **Bedienungsanleitung und Sprachbeschreibung für das Programmiersystem PPS 880/S (kompatible Sprachversion zu Turbo-PASCAL)**
- **Anwendungsbeschreibung und Bedienungsanleitung für das druckerorientierte Softwarepaket PCGRAF**
- **Bedienungsanleitung und Sprachbeschreibung CAS-PASCAL (quellcodekompatible Sprachversion zum kommerziellen PASCAL 1520)**

# Sachwortverzeichnis

- Abschlußtaste 152  
 Abspeicherung 156  
 Abtrennzeichen 162  
 Adreßfeld 83  
 Aktualisieren 155  
 Anfangsmenü 152  
 Anweisung 130, 145  
 Anweisungszeile 122  
 Anwender-Laufzeitbibliothek 133  
 Anzeige 159  
 Anzeigemodus 125  
 Anzeigezeile 161  
 Arbeitspuffer 97f.  
 Arbeitstableau 158  
 Assembler 69, 121ff.  
 Assemblermoduln 133f., 140  
 Assemblerprogrammierung 119, 121  
 Assemblersprache 118, 121, 132  
 ASSIGN-Anweisung 144  
 Aufzeichnungsdichte 83  
 Ausgabeparameter 85  
 Auswerteprogramm 153
- Balkendiagramm 171, 177  
 Balkengrafik 164  
 BASIC 127ff.  
 BASIC-Compiler 127f., 133  
 BASIC-Interpreter 127f., 133  
 Baugruppentausch 210  
 Baustein 134  
 BDOS-Funktionen 73, 85ff.  
 BIOS-Funktionen 73, 91ff.  
 Bedingungsschleifen 137  
 Belegdurchlauf 186  
 Benutzerschnittstelle 100  
 Bereiche, Aufteilung der Diskette in 83
- Betriebsdatenterminal 193  
 Betriebssystem, SCP 5, 30, 69ff., 77, 79, 83f., 97, 126, 140, 148, 151, 215  
 Betriebssystem, BROS 98f.  
 Betriebssystem, JAMB 101f.  
 Betriebssystembereich 97  
 Betriebssystemerweiterung 100, 104  
 Bibliothek 124f.  
 Bibliotheksfunktion 146  
 Bibliotheksverwalter 141  
 Bildformat 206  
 Bildschirm 33f., 154  
 Bildschirmausgabe 155  
 Bildschirmausschrift 152  
 Bildschirmfenster 162  
 Bildschirmformat 170  
 Bildschirminhalt 163  
 Bildschirmmeldung 152  
 Bildschirmsteuerung 19, 24, 35, 100  
 Bildwiederholpeicher 19, 71, 100  
 Binden 124  
 Bindeprozeß 140  
 Bytefolge 125
- CAD-Lösung 187  
 CALL-Anweisung 145  
 CCP-Kommandos 73, 77ff., 80, 82  
 Code-Nummer 85, 202  
 COMMON-Anweisung 142, 144  
 Compiler 69, 99, 128, 133f., 142  
 Compilermodus 132  
 Compilerprogramm 132  
 CRT-Controller 19, 21f.
- DATA-Anweisung 142, 144  
 Datei löschen 79, 86



- Datei, sequentielle 131
- Datei umbenennen 79, 86
- Dateiarbeit 126, 139
- Dateibezeichnung 166
- Dateieröffnung 131
- Dateiname 152
- Dateispezifikation 76f. 79f.
- Dateisteuerblock 88f.
- Dateistrukturen 151
- Dateitransfer 107f.
- Dateitypkennung 123, 173
- Dateiübertragungsprogramme 99
- Dateizugriff, blockweiser 135
- Dateizugriff, direkter 85, 131
- Dateizugriff, sequentieller 85
- Dateizugriff, zeichenweiser 135
- Datenaustauschniveau 107, 109
- Datenaustauschprogramm 107
- Datenbank 5, 149, 152 ff., 215
- Datenbanksprache 153f.
- Datenbankstruktur 154
- Datenbanksystem, relationales 152
- Datenbasis 149
- Datenbestand 131, 148
- Dateneingabemodus 161
- Datenerfassungsprogramm 189
- Datenerfassungssystem 192
- Datenfelder 83, 153
- Datenformate 128, 135
- Datenkanal, sequentieller 85
- Datenmanipulation 153
- Datenmodem 189
- Datensatz 90, 149
- Datenstation 193
- Datentyp 136, 143
- Datentypaustausch 98
- Datenübertragung 145
- Datenübertragungsprotokoll 104
- Datenverknüpfung 149
- Datumformat 154
- Debugger 119, 125f.
- Definitionsdatei 135, 174
- Deklarationsteil 135
- Diagramm 172
- Diagrammdefinition 174
- Dialog 128, 152
- Dialogprogrammierung, interaktive 127
- Dialogsprache 127
- Dienstprogramm, LIB 124
- Dienstprogramm, PIP 81
- Dienstprogramm, INSTSCP 97
- Digitalisierungsgerät 68, 188
- DIMENSION-Anweisung 142, 144
- Dimensionierung 197
- Direktdatei 74f., 131, 146
- Disassembler 141
- Diskette 29ff., 74, 76, 131
- Diskettenformat 69, 72, 74f.
- Diskettenformatieren 83
- Diskettenkapazität 156
- Diskettenlaufwerk 29ff.
- Displayadresse 117
- Disponentenarbeitsplatz 192
- DO-Anweisung 145
- Drucker des PC 1715 50ff.
- Druckformat 155
- Druckformatkontrolle 168
- Druckfunktion 168
- Druckliste 155
- Druckprogramm 206
  
- Echtzeitbetrieb 126, 191
- Echtzeitmodus 125
- Echtzeitsteuerung 141
- Echtzeittest 126
- Editfunktion 168
- Editierbefehl 167
- Editieren/Editierung 147, 162
- Editiermenü 166
- Ein-/Ausgabe-Kanal 21, 23, 85
- Einfügeregime 166
- Eingabeparameter 85
- Eingabezeile 161f.
- Einsatzbedingungen 49, 67
- Einsatzbereich 180
- Einsatzvorbereitung 149
- Empfangsfile 115
- Emulationsprogramm 99, 104
- Emulator/Emulation 113, 115
- END-Anweisung 145
- EQUIVALENCE-Anweisung 142, 144
- Ergibt-Anweisung 142, 144
- Erstbenutzer 180
- Erstinbetriebnahme 211
- Exponentialfunktion 151
- EXTERNAL-Anweisung 142, 144

- Fakturierung 182
- Fehlerkontrolle 132
- Fehlerkorrekturdialog 154
- Fehlerlokalisierung 213
- Fehlermeldung 90, 163
- Fehlerquittung 50
- Fehlerroutine 132
- Fehlerspezifikation 90, 131
- Fehlersuchzeit 213
- Fehlerursprung 163
- Feldeintragung 88
- Felder 142
- Feldinhalt 155
- Ferncomputer 107ff.
- FIFO-Speicher 21
- Fileinformationsblock 139
- Filemodus 114f.
- Filevariable 138
- Formatieren 83
- Formatierungsprogramm 141
- Formulartechnik 64
- FORTRAN 141ff.
- Funktionen, BDOS 73, 85ff.
- Funktionen, BIOS 73, 91ff.
- Funktionen, arithmetische 159
- Funktionen, trigonometrische 159
  
- Generatorprinzip 148
- Generierung 129
- Gerät, logisches 85
- Gerätekonfiguration 174
- Gerätetreiber 11
- Geräuschemission 67
- Gleitkomma-Arithmetik 135
- Gleitkommazahlen 128, 135
- GOTO-Anweisung 144
- Grafikdaten 206
- Grafikmodus 203
- Grafikspalten 203
- Grundkonfiguration 151
- Grundmenü 159f.
- Grundmodus 161
  
- Hardware 13ff., 70
- Hauptmenü 172
- Hauptspeicherbereich 96
- HELP-Einrichtung 156
- hexadezimal 123
- Hilfsdatei 155
- Hilfsfunktion 126
- Hilfsprogramm 98
  
- IF-Anweisung 144
- Implementieren 98
- Indexdatei 195
- Indexdaten 156
- Indizierung 155
- Informationsmenge 152
- Initialisierungsprogramm 68, 99
- Installation 97, 154
- Installationsprogramm 68, 104f., 107  
111f., 115f., 154
- Installationsvorgang 97
- Interfaceverbindungen 212
- Interpretation der Daten 153
- Interpreter 128
- Interpreterbefehlssatz 98
- Interpretermodus 132
- Interpreterprogramm 132
  
- Kalkulationsprogramm 149, 158ff.
- Kaltstart 73f., 91
- Kennungstext 165
- KOMBO-Druck 170
- Kommando-Datei 82f.
- Kommandointerpreter 71, 77
- Kommandomodus 161
- Kommandos, CCP 73f., 77ff., 80, 82
- Kommandos, residente 79
- Kommandotaste 172
- Kommandozeile 77, 82, 123ff., 126
- Kommentarteil 122
- Kommunikationsprogramm TLC 107
- Konvertierfunktion 131
- Konvertierung 98, 146
- Kopplungsinformation 151
- Korrektur 165
- Kreisdiagramme 171, 174, 208
- Kursor 37
- Kursorposition 162
- Kursorstasten 37f.
- Kurven 202ff.
  
- Laden 124
- Ladeprogramm 100, 104f.
- Längenbestimmung einer Zeichenkette  
156

- Längeninformation 204  
 Laufwerke 31, 32, 75, 76  
 Laufzeit-Bibliothek 133  
 Laufzeit-Ladeinformationsdatei 133  
 Laufzeit-Modul 133  
 Lehrgänge und Kurse der Robotron-  
   Schulungszentren 211  
 Leistungsparameter 156  
 Leistungsumfang 174  
 Leitadresse 119, 124  
 Leitungsgrafiken 171  
 Liniendiagramm 171, 178  
 Linienzähler 21  
 Listengenerator 155  
 Listkanal 85  
 Lokalcomputer 107ff.  
 Löschen 162
- Makro** 121ff.  
**Makroanzeiger** 154  
**Makroassemblersprache** 98  
**Makrobefehlsaufbereitungssystem** 99  
**Makrobefehlsinterpreter** 99  
**Makroprogrammtestmonitor** 99  
 Marke 122  
 Maschinensprache 118  
 Matrix 158  
 Matrixdrucker 51, 54f., 57, 64  
 Mehrfachcarbonband 59, 62  
 Mehrfachkopie 163  
 Menü 105, 115  
 Menüsteuerung 151  
 Mikroprogrammssystem 99  
**Mikroprogrammtestmonitor** 99  
 Mikroprozessor 18  
 Miniassembler 140  
 Minidiskettenlaufwerk 151  
 Mnemonik 118, 122  
 Modem 117  
 Modulbibliothek 140  
 Modulcompilierung 135  
 Moduln 124, 134  
 Modulname 125  
 Modulprogrammierung 140  
**MOVE-Prozedur** 134
- Nachrichtensteuerung** 114  
 Nadeldrucker 53ff., 56f., 64, 202f.  
 Netzanschluß 212
- Netze, lokale 10, 68, 180  
 Neuinstallation 154  
 Nutzerbereich 80  
 Nutzerbereichsnummer 109
- Objektcode 100, 123  
 Objektcodedatei 119, 123  
 Objektmoduln 134  
 Objektprogramm 124, 133  
 oktal 123  
 Operand 122  
 Operationszeichen 129  
 Operatoren 122, 129, 136, 143  
 Option 81  
 Overlaygebiete 140  
 Overlaymoduln 140
- Papierformat** 155  
**Papierzuführung** 59  
**Parameter** 96  
**Parameterinstallation** 113  
 Parität 106  
**PASCAL** 134f.  
**PAUSE-Anweisung** 145  
**PCGRAF** 171ff.  
**Personalcomputer** 5, 10  
**Phasenbibliothek** 99f.  
**Phasenbibliotheksservice** 99  
**PIP-Kommando** 81  
 Portabilität 102  
 Port-Adresse 132  
 Positionierung 157  
**Programmanfang** 124  
**Programmanweisung** 128  
**Programmdatei** 97, 119, 154, 157  
**Programmierdirektive** 128  
**Programmeinheit** 146  
**Programmiermethodik** 121  
**Programmiersprachen** 99, 101, 127ff.,  
   147, 149, 169  
**Programmierung** 156  
**Programmkomponente** 140  
**Programmkopf** 135  
**Programmlaufzeit** 140  
**Programmpaket** 190  
**Programmphase** 119  
**Programmschleife** 157  
**Programmstart** 152  
**Programmstruktur** 118, 155

- Programmsystem 148, 151  
Programmüberlagerung 131, 140  
Programmverbinder 119, 123ff., 133  
Programmverketzung 140  
Programmverzweigung 157  
Programmwurzel 140  
Prozedur 104, 112, 134ff., 140  
Prozessorarbeit 132
- Quellbibliotheksservice 99  
Quellcode 133  
Quelldatei 121ff., 124  
Quellprogramm 118f., 123, 133, 147  
Quelltextbibliothek 99
- RAM-Speicher 19  
Raster- und Videosteuerung 21  
Rechnernetz 192  
REDABAS 152, ff., 190  
Redigieren 129  
Regeln, syntaktische 141  
Register 85  
Registernamen 122  
Relation 153  
Reparatur 210  
REPORT-Generator 149  
RETURN-Anweisung 145  
ROM-Anfangslader 19  
Routine 132f.
- Säulendiagramm 172, 177  
Satzelement 149  
Satz, relativer 131, 156  
Schalterangabe 124  
Schalterdrucker 53, 64ff.  
Schlüselfeld 156  
Schlüsselzeichen 162  
Schnittstelle 19, 23f., 26f., 40ff.  
Schreibmaschine 165  
Seitenformatierung 169  
Sektor 72, 74f., 83  
Sektorlänge 83  
Sektornummer 83  
Serviceleistungen 210  
Sicherheitskopie 124  
Softwarekompatibilität 11, 70  
Softwarelösung 151, 181  
Softwaretechnologie 121  
Softwarewerkzeug 172
- Sonderzeichen 161  
Sortierkriterium 183  
Sortierordnung 155  
Spaltenbezeichnung 154  
Spaltennummer 160  
Spaltensumme 155  
Spaltenüberschrift 155, 164  
Spaltenwert 155  
Speicheradresse 81, 91, 132  
Speicherbereich 96, 123  
Speicherbereich, transienter (TPA) 73, 77, 80, 96  
Speichervariable 129, 158  
Sprachumfang 129  
Sprungbefehl 91  
Sprungmarke 144  
Spur 74f., 94, 97  
Spurnummer 83  
Spurprotokollierung 126  
Standardantwort 168  
Standardfehlermitteilung 132  
Standardfunktion 146  
Standardsoftware 5, 121, 148f.  
Startadresse 132  
Statuszeile 161  
Steueranweisung 144  
Steuercode 154  
Steuerfunktion 131  
Steuerprogramm 99, 101  
Steuerrechner 114, 191  
STOP-Anweisung 145  
STRING-Manipulation 135  
Stromversorgung 32  
Strukturbeschreibung 184  
Subkanal 92  
Symboldatei 119, 126  
Symboltabelle 123, 126  
Syntax 134  
Syntaxkontrolle 123  
Systembus 17, 19, 40  
Systemeinheit 13, 15, 48  
Systemgenerierung 83  
Systemkern 69  
Systemmeldung 101  
Systemprogramm, DU 125  
Systemprogramm, transientes 80f., 125  
Systemspur 84, 97  
Systemsteuereinheit 193

- Tabelle (DPH) 92, 149  
Tabellenkalkulation 5  
Tableaukursor 161  
Tableaumodus 161 f.  
Tabulator-Linear 167  
Tastatur 35 ff., 110  
Tastatur, alphanumerische 35 f.  
Tastaturcodierung 152  
Tastatureingaben 166  
Tastenumcodierung 170  
Teilzeichenketten 156  
Telekommunikationsprogramm 104  
Terminaladresse 117  
Terminalarbeit 101, 109  
Terminalmodus 110, 112, 114 f.  
Textbearbeitungssystem 165  
Textdatei 166, 169  
Textfiles 113  
Textlänge 161  
Textprogramm 99  
Textprozessor (TP) 121, 142, 164 ff.  
Textverarbeitung 5, 166, 215  
Titelzeile 163  
Trenneinrichtung 64, 66  
Trennzeichen 126  
Typbezeichnung 143  
Typenraddrucker 51, 53, 59 ff.  
Typenscheibe 53, 59  
Typkenner 128  
Typkennung 123  
Typvereinbarung 130, 140, 144
- Überlagerungsdatei 159  
Überlagerungsstruktur 134  
Übersetzungsliste 121  
Übersetzungsvorgang 118 f.  
Übertragungsfunktion 139  
Übertragungsgeschwindigkeit 106, 116  
Übertragungsprotokoll 107, 110  
Unterprogramm 91, 132  
Unterprogrammroutine 125  
Unterprogrammssprung 130  
Unterprogrammverzweigung 130  
Urlader, residenter 73, 97
- Variable 122, 128  
Variablenbezeichner 130  
Variablenzugriff 137  
Vereinbarungsanweisung 130, 143  
Vergleichsoperatoren 129, 136, 143  
Verschlüsselung 203  
Verständigungsbereich 71, 96
- Wagenrücklaufkommando 111  
Wartung 212  
Wertbelegung 164  
Wertebereich 135  
Wertzuweisung 130  
Wortumbruch 167  
Wortumschlag 169
- Zählschleifen 137  
Zehnerastatur 36 ff.  
Zeichenausgabe, sequentielle 92  
Zeicheneingabe, sequentielle 92  
Zeichenfolge 128, 130  
Zeichenketten 128, 135  
Zeichenkettenfunktionen 131  
Zeichenkettenroutinen 138  
Zeichenpuffer 126  
Zeichensatz 53  
Zeichenzähler 19  
Zeiger 139, 156  
Zeilennummern 160  
Zeilenpuffer 21  
Zeilenvorschubkommando 111  
Zeilenzähler 21  
Zentrale Recheneinheit 16 ff.  
Zufallsgenerator 130  
Zustandsvariable 96  
Zyklusanweisung 131

# Die Wirtschaft

**Bis 1990 werden mehr als 150 000 Personalcomputer produziert und damit wesentliche Voraussetzungen für die weitere Intensivierung in allen Zweigen unserer Volkswirtschaft geschaffen. Die zusätzliche Bereitstellung von Personal- und Arbeitsplatzcomputern und die rasch steigende Produktion dieser Geräte ist eine wesentliche Voraussetzung für den massenhaften Einsatz moderner Rechentechnik am Arbeitsplatz.**

**Jetzt kommt es darauf an, für den effektiven Einsatz der Informationsverarbeitungstechnik unter Nutzung der vorhandenen Standardsoftware zu sorgen und immer neue, wirksamere Anwendungen vorzubereiten. Diesen Prozeß soll die vorliegende Broschüre unterstützen. Sie hilft allen derzeitigen und künftigen Nutzern, den Einsatz der Rechner im jeweiligen Bereich aktiv vorzubereiten und sie mit dem Ziel steigender Leistungen bei erleichterten Arbeitsbedingungen so wirksam wie möglich zu nutzen.**